Cisco IOS H.323 Configuration Guide

Release 15.0(1)M

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
     800 553-NETS (6387)
Fax: 408 527-0883

Cisco IOS H.323 Feature Roadmap

This guide collects together in one place information about Cisco IOS voice features having to do with the H.323 standard for sending and receiving audio, video, and data on an IP-based internetwork.

This first chapter of the guide is a feature roadmap. It describes how to access Cisco Feature Navigator and lists H.323 features by Cisco IOS release.

Note For information about the full set of Cisco IOS voice features, see the entire Cisco IOS Voice Configuration Library—including library preface, glossary, and other documents—at http://www.cisco.com/en/US/docs/ios/12_3/vvf_c/cisco_ios_voice_configuration_library_glossary/vcl.htm

Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Cisco IOS H.323 Feature List

Table 1 lists H.323 features by Cisco IOS release. Features that are introduced in a particular release are available in that and subsequent releases.

Table 1   H.323 Features by Cisco IOS Release

Release	Features Introduced[1]	Feature Description	Feature Documentation
12.4(4)XC	H.323 VoIP Call Preservation Enhancements for WAN Link Failures	Sustains connectivity for H.323 topologies where signaling is handled by an entity that is different from the other endpoint,	"Configuring H.323 VoIP Call Preservation Enhancements for WAN Link Failures" section on page 111 of this guide

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Table 1 **H.323 Features by Cisco IOS Release (continued)**

Release	Features Introduced[1]	Feature Description	Feature Documentation
12.4(4)T	No Retry on User Busy in an H.323 Gateway	Changes the default behavior of the gateway to not retry alternate endpoints when the release complete reason is user busy.	"Configuring No Retry on User Busy in an H.323 Gateway" section on page 92 of this guide
12.3(11)T	Overlap Signaling on H.323 Terminating Gateways	Introduces overlap signaling.	"Configuring Overlap Signaling on H.323 Terminating Gateways" section on page 91 of this guide
12.3(7)T	Signal ISDN B-Channel ID to Enable Application Control of Voice Gateway Trunks	Enables the H.323 gateway to access B-channel information for all H.323 calls.	"Configuring Signal ISDN B-Channel ID to Enable Application Control of Voice Gateway Trunks" section on page 108 of this guide
12.3(1)	Gatekeeper Enhancements for Managed Voice Services	Adds call control based on available bandwidth and endpoint resources, allowing call rerouting to achieve higher call-completion rates and ensure QoS.	"Configuring H.323 Gatekeepers and Proxies" chapter of this guide
	Gateway Codec Order Preservation and Shutdown Control	Adds a way to shutdown and restart SIP and H.323 gateways.	"Configuring Multiple Codecs" section on page 75 of this guide
12.2(15)T	Gatekeeper Management Statistics	Provides statistics that can be used to monitor a network and troubleshoot network problems.	"Gatekeeper-Management Statistics" section on page 38 section of this guide
	H.323v4 Gateway Zone Prefix Registration Enhancements	Reduces configuration complexity on the gatekeeper by enabling a gateway to report supported E.164 prefixes to the gatekeeper at registration.	"Configuring H.323 Version 4 Zone Prefix Registration" section on page 101 section of this guide
	Tokenless Call Authorization	Creates a trusted access list to handle networks that have not implemented token-security features.	"Configuring Tokenless Call Authorization" section on page 181 section of this guide
12.2(13)T	Multiservice IP-to-IP Gateway with Media Flow-Around	Adds media flow-around capability to the multiservice IP-to-IP gateway.	*"Configuring H.323 Gateways" section on page 43* section of this guide

Table 1 **H.323 Features by Cisco IOS Release (continued)**

Release	Features Introduced[1]	Feature Description	Feature Documentation
12.2(11)T	Basic Service Relationships (H.225 Annex-G)	Implements the minimal set of Annex G features needed to allow Cisco border elements to interoperate with other border elements.	"Configuring Annex G" section on page 79 of this guide
	Call Status Tracking Optimization	Reduces unnecessary messages between gatekeeper and the gateways, reducing network congestion and CPU over-utilization.	"Configuring the IRR Timer and Disable IRQ Requests" section on page 228 of this guide
	Configuring a Gatekeeper to Provide Nonavailability Information for Terminating Endpoints	Describes how to configure a gatekeeper to provide nonavailability information for terminating endpoints.	"Configuring Nonavailability Information for Terminating Endpoints" section on page 225 of this guide
	Gatekeeper Alias Registration and Address Resolution Enhancements	Allows configuration of multiple prefixes for a local zone and registration of an endpoint belonging to multiple zone prefixes.	"Configuring Gatekeeper Alias Registration and Address Resolution" section on page 152 of this guide
	Gatekeeper Endpoint Control Enhancements	Enable finer-grained control of gatekeeper registrations, and enable more capable and robust back-end server applications.	"Configuring Endpoints" section on page 217 of this guide
	Gatekeeper-to-Gatekeeper Authentication	Provides additional security for H.323 networks by introducing the ability to validate intradomain and interdomain gatekeeper-to-gatekeeper LRQ messages on a per-hop basis. When used in conjunction with per-call security using IZCT, protects network resources from attackers and prevents security holes.	"Configuring Security and Authentication" section on page 171 of this guide

Table 1   H.323 Features by Cisco IOS Release (continued)

Release	Features Introduced[1]	Feature Description	Feature Documentation
12.2(11)T	H.323 Dual Tone Multifrequency Relay Using Named Telephone Events	Enhances interoperability of Cisco gateways with equipment from other vendors by implementing H245v7 extensions to support RFC2833 in-band audio telephone events and in-band audio tones as well as support for asymmetrical RTP dynamic payload types.	*"Configuring DTMF Relay" section on page 69* of this guide
	VoIP Gatekeeper Trunk and Carrier Based Routing Enhancements	Allows service providers to route traffic based on trunk and carrier rule sets.	*VoIP Gatekeeper Trunk and Carrier Based Routing Enhancements*
	VoIP Gateway Trunk and Carrier Based Routing Enhancements	Addresses functional enhancements needed to improve routing on Cisco voice-over-packet platforms.	*VoIP Gateway Trunk and Carrier Based Routing Enhancements*
	VoIP Outgoing Trunk Group ID and Carrier ID for Gateways and Gatekeepers	Provides alternate endpoints by adding trunk group labels (outgoing trunk group identification) to endpoints on H.323 VoIP networks.	*"Configuring Border Elements" section on page 216* of this guide
12.2(8)T	Gatekeeper Transaction Message Protocol Interface Resiliency Enhancement	Provides enhanced call-routing and address-translation services.	*"Configuring Gatekeeper Interaction with External Applications" section on page 187* of this guide
12.2(4)T	Cisco H.323 Scalability and Interoperability Enhancements for Gatekeepers	Upgrades the gatekeeper to comply with H.323 Version 3.	*"Configuring H.323 Gateways" section on page 43* of this guide
	Cisco H.323 Scalability and Interoperability Enhancements for Gateways	Upgrades the gateway to comply with H.323 Version 3.	*"Configuring H.323 Gateways" section on page 43* chapter of this guide
	H.323 Version 2 Enhancements	Upgrades Cisco IOS software to comply with the mandatory requirements and several optional features of the version 2 specification. Enhances the existing VoIP gateway and multimedia conference manager gatekeeper and proxy.	*"Configuring Generic Transparency Descriptor for GKTMP Using SS7 Interconnect for Voice Gateways Version 2.0" section on page 95* of this guide
	Inter-Domain Gatekeeper Security Enhancement	Provides a secure mechanism for authenticating and authorizing H.323 calls from other administrative domains.	*"Configuring E.164 Interzone Routing" section on page 183* of this guide
	Sequential Location Request Enhancement	Provides a potentially faster gatekeeper LRQ response when an LRJ response is received while the gatekeeper is sending sequential LRQs.	*"Configuring Gatekeeper Zones" section on page 143* of this guide

Table 1 **H.323 Features by Cisco IOS Release (continued)**

Release	Features Introduced[1]	Feature Description	Feature Documentation
12.2(2)T	Call Admission Control for H.323 VoIP Gateways	Provides call-admission control for voice gateways.	"Configuring Call Admission Control" section on page 108 of this guide
	H.323 Call Redirection Enhancements	Indicates the nature of a call redirection and provides a nonstandard method for using the message to effect call transfer.	"H.323 Call Redirection" section on page 30 of this guide
	High Performance Gatekeeper	Facilitates carrier-class reliability, security, and performance.	"Configuring H.323 Gatekeepers and Proxies" section on page 139 of this guide
	NAT Support of H.323 v2 RAS	Allows embedded IP addresses to be inspected for potential address translation.	"Configuring an H.323 Proxy Server" section on page 196 of this guide
12.1(1)T	Gatekeeper Ecosystem Interoperability	Allows a gateway to move between gatekeepers during GRQ and RRQ messages.	"Ecosystem Gatekeeper Interoperability" section on page 36 of this guide
	Gateway-to-Gatekeeper Billing Redundancy	Provides redundant billing information to an alternate gatekeeper if the primary gatekeeper becomes unavailable.	"Gateway-to-Gatekeeper Billing Redundancy" section on page 36 of this guide

1. Features that are introduced in a particular release are available in that and subsequent releases.

H.323 Overview

This chapter provides an overview of the ITU- H.323 standard for sending and receiving audio, video, and data on an IP-based internetwork.

Note • For information about the full set of Cisco IOS voice features, see the entire Cisco IOS Voice Configuration Library—including library preface, glossary, and other documents—at http://www.cisco.com/en/US/docs/ios/12_3/vvf_c/cisco_ios_voice_configuration_library_glossary/vcl.htm

Contents

Prerequisites for Configuring an H.323 Network

- Establish a working IP network. For information on IP configuration, see the references listed in the "Related Documents" section on page 20.

- Install the appropriate voice network module and voice-interface card for the Cisco router. For information on the module and card, see the *Voice Network Module and Voice Interface Card Configuration Note* that came with the voice network module.

- Configure your H.323 gateways, gatekeepers, and proxies. For information on VoIP configuration, see the resources in the .

- To ensure network security, configure a RADIUS authentication, authorization, and accounting (AAA) server. Configure the following information in your CiscoSecure AAA server:

    - In the /etc/raddb/clients file, ensure that the following information is provided:

      ```
      #Client Name          Key
      #-----------          -------------------
      gk215.cisco.com        testing123
      ```

      Where *gk215.cisco.com* is resolved to the IP address of the gatekeeper requesting authentication

    - In the /etc/raddb/users file, ensure that the following information is provided:

      ```
      taeduk@cisco.com   Password = "thiswouldbethepassword"
      User-Service-Type = Framed-User,
      Login-Service = Telnet
      ```

      Where *taeduk@cisco.com* is the h323-id of the gateway authenticating to gatekeeper gk215.cisco.com.

- Configure an NTP server for your network.

- For gatekeeper-management statistics, do the following:

    - Configure the Simple Network Management Protocol (SNMP) agent in global configuration mode.

    - Update the MIB data files on your management workstations so that the management application knows what the new objects are.

Restrictions for Configuring an H.323 Network

This section describes the following restrictions:

H.323 Version 2 Restrictions

- All systems must be running either Cisco IOS Release 11.3(9)NA and later releases or Cisco IOS Release 12.0(3)T and later releases to interoperate with the Cisco H.323 Version 2 features. Earlier releases contain H.323 Version 1 software that does not support protocol messages that have an H.323 Version 2 protocol identifier. The earlier releases do not interoperate with Cisco H.323 Version 2 Phase 2 features.

- To use H.450 services (call transfer or call deflection), use Cisco IOS Release 12.1(1)T or later on the gatekeeper: H.450 on the gateways is incompatible with previous releases of the Cisco gatekeeper.

- If a Cisco AS5300 is used, the software requires the appropriate version of VCWare.

- The H.323 Version 2 Fast Connect feature is not explicitly configurable. It is assumed that the gateway is capable of sending and receiving fast-connect procedures unless its corresponding dial peer is configured for RSVP (in other words, the req-qos is set to a value other than the default of best-effort). In the latter case, traditional "slow" connect procedures are followed, and the endpoint neither attempts to initiate fast connect nor responds to a fast-connect request from its peer.

H.323 Signaling Enhancement Restrictions

- Supplementary voice services are not supported with ISDN and CAS over an H.323 network—except on the NET5 switch.

- Progress messages require a PI value, and only ITU-T standards are supported.

- Progress indicator 2 is not supported in progress messages for the DMS100 switch.

- TCL 2.0 for IVR supports the interworking signaling enhancements only on the Cisco AS5300. For IVR on other Cisco platforms, select TCL 1.0 as the session application. To use standard IVR applications with TCL 1.0, configure the application name as "session.t.old" by using the **call application voice** command. It is not necessary to do this if customized scripts are used.

- The Cisco AS5300 sends a connect message to the originating gateway after it receives a setup message only when it is configured for one of the following supported switch types:
  - 5ESS
  - NET5
  - NTT
  - QSIG
  - QSIGP

- For the SS7 interconnect for voice gateways solution, the following behavior applies to suspend and resume messages, which are supported on NET5 and NI2+ ISDN interfaces:
  - If the ISDN interface is NET5, the Cisco AS5300 sends a notify message with the notification indicator (NI) set to user-suspended or user-resumed.
  - If the ISDN interface is NI2+, the Cisco AS5300 sends a suspend or resume message to the Cisco SC2200.
  - If the Cisco SC2200 receives an ISUP suspend or resume message, it sends an NI2+ suspend or resume message to the Cisco AS5300.
  - Both the Cisco AS5300 and the Cisco SC2200 timers start when a suspend message is received. The Cisco AS5300 timer, T307, is configurable from 30 to 300 seconds. The Cisco SC2200 timer, T6, is not configurable and has a default of 120 seconds if the ISUP variant Q.761 is used.

- When the Cisco AS5300 and the Cisco SC2200 receive a resume message, the timers are stopped. If either of the timers expires, the call is released with a cause code of normal clearing.

Source Call Signal Address and H.245 Empty Capabilities Set Restrictions

- To use H.450 services (call transfer or call deflection), Cisco IOS Release 12.1(2)T of the gatekeeper must be used. H.450 on the gateways is incompatible with previous releases of the Cisco gatekeeper.
- If a Cisco AS5300 is used, the system requires the appropriate version of VCWare.

Call Transfer Restrictions

- Interactive Voice Response (IVR) must be configured on the router and supplementary services must be provided for processing. For information about configuring IVR and supplementary services, see *Configuring Interactive Voice Response for Cisco Access Platforms*.
- The "session" application must be specified properly for the dial-peers.
- Release 12.1(1)T (or later) of the Cisco H.323 Gatekeeper is required.
- The H.323 Call Redirection Enhancements feature does not provide the ability for a Cisco H.323 Gateway to initiate a call transfer request.

Ecosystem Gatekeeper Interoperability Restrictions

- The maximum number of alternate gatekeepers is eight (including static gatekeepers).
- During the retransmission of the GRQ or RRQ messages, the gateway responds only to the current gatekeeper (regardless of the state of the altGKisPermanent flag).
- The process of retransmission to an alternate gatekeeper can be time-consuming.

H.323 Gatekeepers and Proxies Restrictions

- Both the gateway H323_ID and the generalID in ClearTokens should be same.

Information About H.323

> **Note** When you configure H.323 on a router, the ports on all its interfaces are open by default. This makes the router vulnerable to malicious attackers who can execute toll fraud across the gateway if the router has a public IP address and a public switched telephone network (PSTN) connection. To eliminate the threat, you should bind an interface to private IP address that is not accessible by untrusted hosts. In addition, you should protect any public or untrusted interface by configuring a firewall or an access control list (ACL) to prevent unwanted traffic from traversing the router.

This section contains the following information:

H.323 Standards

Table 1 lists H.323 standards and applicable Cisco VoIP features.

Table 1   H.323 Standards and Applicable Cisco VoIP Features

Standard	Applicable Cisco VoIP Features

H.323 Version 1	• Multimedia conferencing protocol which includes voice, video, and data conferencing for use over packet-switched networks
	• Provides a framework that uses others to describe the actual protocol
	   – H.245 conference-control protocol
	   – H.225 call signaling and communication between endpoints (call signaling) and the gatekeeper (RAS)
	   – Q.931
	   – RTP/RTCP audio and video

H.323 Version 2[1]	• Improved gateway selection process • Gateway resource availability reporting • Support for single proxy configurations • Registration of E.164 addresses for gateway-attached devices • Tunneling of redirecting number information element • DTMF-relay • Hookflash relay • CODEC negotiation
H.323 Version 3	• Caller ID • Language preference • Annex E—Protocol for Multiplexed Call Signaling Transport • Annex F—Simple Endpoint Type • Annex G—Communication Between Administrative Domains
H.323 Version 4	• Gateway decomposition • Additive registrations • Dynamic zone prefix registration • Alternate gatekeepers • Endpoint capacity

1. To learn about restrictions that apply to Version 2, see the "H.323 Version 2 Restrictions" section on page 8.

Network Components

Figure 1 shows a typical H.323 network. Network components are described below.

Figure 1   Gatekeeper in an H.323 Network
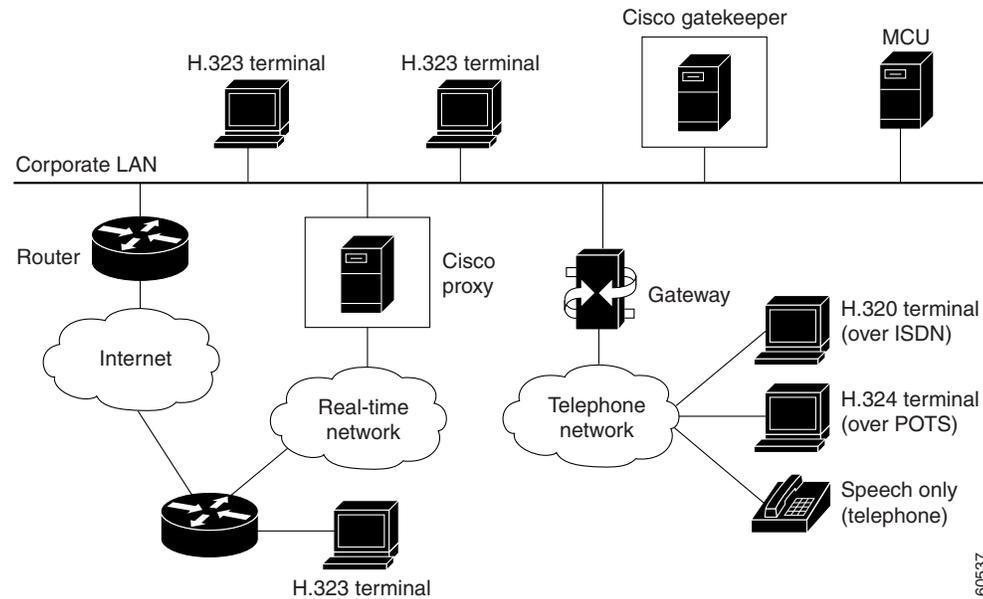
H.323 Terminals

An H.323 terminal is an endpoint in the network that provides for real-time, two-way communications with another H.323 terminal, gateway, or multipoint control unit (MCU). The communications consist of control, indications, audio, moving color video pictures, or data between the two terminals. A terminal may provide audio only; audio and data; audio and video; or audio, data, and video. The terminal can be a computer-based video conferencing system or other device.

A gatekeeper supports a broad variety of H.323 terminal implementations from many different vendors. These terminals must support the standard H.323 Registration, Admission, and Status (RAS) protocol to function with the gatekeeper.

Gatekeepers recognize one of two types of terminal aliases, or terminal names:

- H.323 IDs, which are arbitrary, case-sensitive text strings
- E.164 addresses, which are telephone numbers

If an H.323 network deploys interzone communication, each terminal should at least have a fully qualified e-mail name as its H.323 identification (ID), for example, bob@cisco.com. The domain name of the e-mail ID should be the same as the configured domain name for the gatekeeper of which it is to be a member. As in the previous example, the domain name would be cisco.com.

Multipoint Control Unit

A multipoint control unit (MCU) is an endpoint on the network that allows three or more endpoints to participate in a multipoint conference. It controls and mixes video, audio, and data from endpoints to create a robust multimedia conference. An MCU may also connect two endpoints in a point-to-point conference, which may later develop into a multipoint conference.

Note Some terminals have limited multipoint control built into them. These terminals may not require an MCU that includes all the functionality mentioned.

H.323 Gateways

An H.323 gateway is an endpoint on the LAN that provides real-time communications between H.323 terminals on the LAN and other ITU terminals on a WAN or to other H.323 gateways.

Gateways allow H.323 terminals to communicate with devices that are running other protocols. They provide protocol conversion between the devices that are running different types of protocols. For example, Figure 2 shows a gateway between an H.323 terminal and a non-H.323 terminal.

Figure 2   **Gateway Between an H.323 Terminal and an H.320 Terminal**
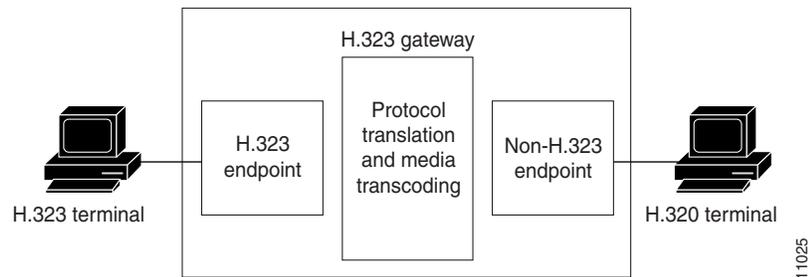
H.323 Proxies

H.323 proxies are special types of gateways that relay H.323 calls to another H.323 endpoint. They can be used to isolate sections of an H.323 network for security purposes, to manage quality of service (QoS), or to perform special application-specific routing tasks.

H.323 Gatekeepers

An H.323 gatekeeper is an H.323 entity on the LAN that provides address translation and that controls access to the LAN for H.323 terminals, gateways, and MCUs.

Gatekeepers are optional nodes that manage endpoints in an H.323 network. The endpoints communicate with the gatekeeper using the RAS protocol.

Endpoints attempt to register with a gatekeeper on startup. When they wish to communicate with another endpoint, they request admission to initiate a call using a symbolic alias for the endpoint, such as an E.164 address or an e-mail address. If the gatekeeper decides that the call can proceed, it returns a destination IP address to the originating endpoint. This IP address may not be the actual address of the destination endpoint, but it may be an intermediate address, such as the address of a proxy or a gatekeeper that routes call signaling.

> ✎
>
> **Note**   Although the gatekeeper is an optional H.323 component, it must be included in the network if proxies are used.

The Cisco gatekeeper provides H.323 call management, including admission control, bandwidth management, and routing services for calls in the network.

The Cisco H.323-compliant Multimedia Conference Manager (MCM) is a subset of gatekeeper functionality available in a special image.

> ✎
>
> **Note**   To learn about MCM and other special images, use Cisco Feature Navigator. Access Cisco Feature Navigator at http://www.cisco.com/go/fn. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Figure 3   Cisco H.323/Gatekeeper Overview
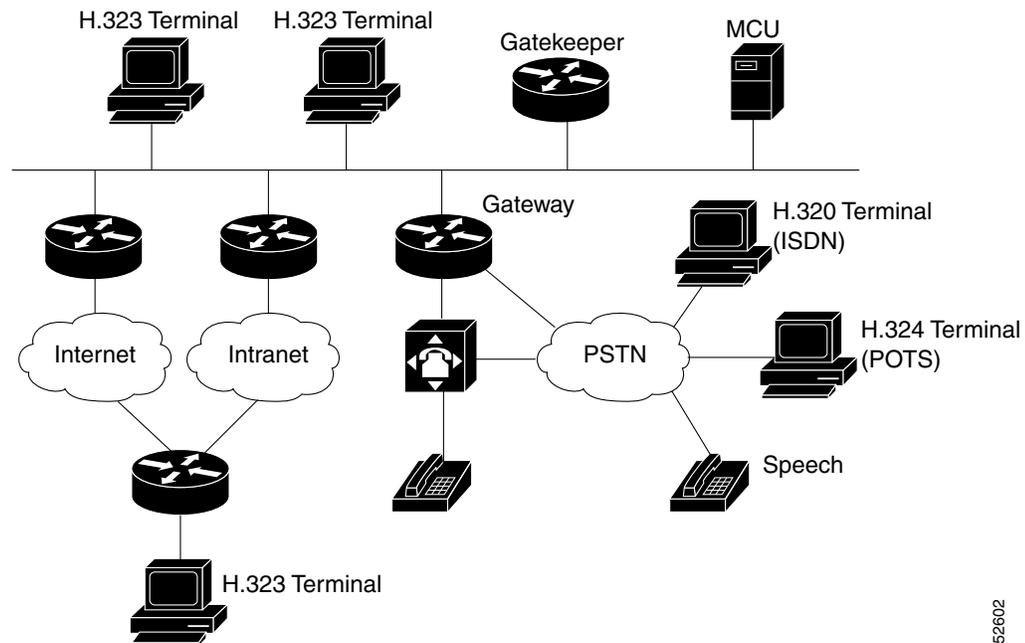
Alternate Gatekeepers

An endpoint that detects the failure of its gatekeeper can safely recover from that failure by utilizing an alternate gatekeeper for future requests, including requests for existing calls. A gateway can only be registered to a single GK at a time. Only one GK is allowed to manage a single zone. The cluster manages up to five similarly configured zones and shares resources between the alternate gatekeepers in the cluster for each zone. You can define up to 100 zones in a single GK.

Alternate Endpoints

A calling endpoint can recover from a call setup failure by sending a setup message to one of the alternate endpoints so that it is possible for a call to finish even if a gateway goes down and the gatekeeper is not yet aware of the problem. Cisco supports a maximum of 20 alternates for each endpoint, and any alternates received through registration, admission, and status protocol (RAS) messages are merged with those entered manually in the gatekeeper command-line interface. If more than 20 alternates are submitted, the total list of alternates reverts back to 20.

GKTMP Messages

The Gatekeeper Transaction Message Protocol (GKTMP) servers can set triggers for disengage request (DRQ) and resource availability indication (RAI) messages. Other messages are extended to contain more parameters for added call control.

Billing Information

The gatekeeper sends detailed call information to a RADIUS distributed client/server system that can be used for billing purposes. RADIUS servers use the vendor-specific attribute (VSA) capability to configure features for individual users.

Least-Cost Routing

Cost and priority fields are included with each remote zone definition, which ensures that the zones with lower cost are given an advantage over zones with higher cost.

Load Balancing

Load balancing allows the gatekeeper to move registered H.323 endpoints to an alternate gatekeeper or to reject new calls and registrations once a certain threshold is met.

Border Elements

Border elements (BE) exchange addressing information and participate in call authorization between the administrative domains. The BEs are often located with a gatekeeper. The BE can reduce the routing information passed though the network by aggregating address information.

Gatekeeper Zones

An H.323 endpoint is an H.323 terminal, gateway, or MCU. An endpoint can call and be called.

H.323 endpoints are grouped into zones. Each zone has one gatekeeper that manages all the endpoints in the zone. A zone is an administrative convenience similar to a Domain Name System (DNS) domain. (Because a zone is, by definition, the area of control of a gatekeeper, the terms "zone name" and "gatekeeper name" are used synonymously in this chapter.)

Note The maximum number of local zones defined in a gatekeeper should not exceed 100.

Discovery and Registration

Gateways and gatekeepers communicate using the Registration, Admission, and Status (RAS) protocol for discovery and registration. When endpoints are brought online, they first attempt to discover their gatekeeper. They discover their gatekeeper either by sending multicast a discovery request or by being configured with the address and, optionally, with the name of the gatekeeper and by sending a unicast discovery request. Following successful discovery, each endpoint registers with the gatekeeper. The gatekeeper keeps track of which endpoints are online and available to receive calls.

Cisco IOS Network Address Translation (NAT) supports all H.225 and H.245 message types, including those sent in the RAS protocol.

Call Setup

In a typical H.323 call setup scenario, after RAS messages are exchanged, H.225 setup messages are sent over a control channel. For example, in Figure 4, both gateways are registered to the same gatekeeper, and the gatekeeper has chosen direct call signaling.

1. Gateway 1 (the calling gateway) initiates the admission request (ARQ) (1)/admission confirmation (ACF) (2) exchange with that gatekeeper.

2. The gatekeeper returns the call signaling channel address of Gateway 2 (the called gateway) in the ACF.

3. Gateway 1 then sends the setup (3) message to Gateway 2 using that transport address.

4. The setup is complete and the call is proceeding (4).

5. If Gateway 2 wishes to accept the call, it initiates an ARQ (5)/ACF (6) exchange with the gatekeeper.

6. The gatekeeper responses with ACF/ARJ (6).

7. Gateway 2 sends an alerting (7) message to Gateway 1. (If Gateway 2 receives an admission reject [ARJ] (6) message instead of an ACF message, it sends a release complete message to Gateway 1 instead of the alerting message.)

8. Gateway 2 responds with the connect (8) messege to Gateway 1

Note An H.245 control channel transport for use in H.245 signalling can send in any of the H.225 messeges: call proceeding, alerting, or connect.

Figure 4   *Both Gateways Registered to the Same Gatekeeper*
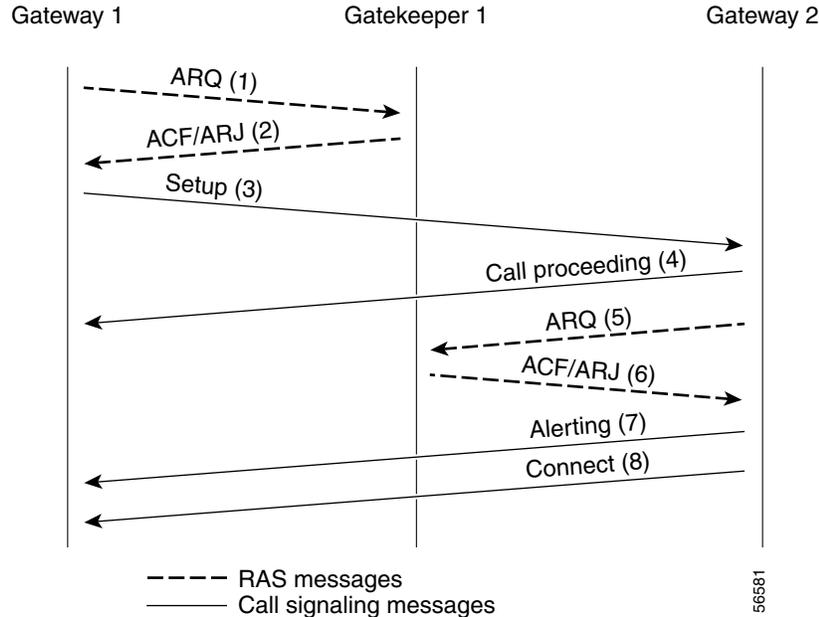
Fast connect allows endpoints to establish media channels without waiting for a separate H.245 connection to be opened. This streamlines the number of messages that are exchanged and the amount of processing that must be done before endpoint connections can be established. A high-level view of the fast-connect procedures within the H.323 protocol follows:

1. The calling endpoint transmits a setup message containing the fastStart element that contains a sequence of encoded logical channel structures, each representing a different capability media type for both "send" and "receive" directions.

2. The called endpoint selects one or more of the media types offered by the calling endpoint for the send and receive directions and returns its selections in a fastStart element in any H.225 message up to and including connect. At this point, the called endpoint must be prepared to receive media along any of the channels it selected.

3. If H.245 procedures are needed and one or both of the endpoints do not support tunneling, a separate H.245 connection is used.

Fast connect is not explicitly configurable. All H.323 Version 2 VoIP endpoints are capable of initiating or accepting fast-connect calls. It is assumed that the gateway is capable of sending and receiving fast-connect procedures unless its corresponding dial peer has been configured for the Resource Reservation Protocol (RSVP). RSVP means the quality of service is set by the **req-qos** command to a value other than the default of best-effort. If the dial peer has been configured for RSVP, traditional "slow" connect procedures are followed, and the endpoint neither attempts to initiate fast connect nor responds to a fast-connect request from its peer.

A terminating endpoint can reject fast connect by simply omitting the fastStart element from all H.225 messages up to and including connect. In this case, normal H.245 procedures are followed and a separate H.245 TCP connection is established. So, if an endpoint does not support the fast-connect procedures, normal H.245 procedures are followed. In addition, certain conditions can cause a fast-connect call to fall back to normal H.245 procedures to complete the call.

Once a media connection has been opened (an audio path has been established), either endpoint has the option of switching to H.245 procedures (if they are needed) by using H.245 tunneling, whereby H.245 messages are encapsulated within the h245Control element of H.225 messages.

The **dtmf-relay** command is the only H.245-cognizant command that can initiate H.245-tunneling procedures from a fast-connect call. If H.245 tunneling is active on the call, switching to a separate H.245 connection is not supported.

A Cisco terminating endpoint accepts a fast-connect request only if a pair of symmetric codecs (codecs that in both directions are equivalent or identical) can be selected from a list that has been offered. The originating endpoint is constrained only by what it can send through the codec (or voice class codec list) associated with the dial peer.

If the Cisco originating endpoint has offered multiple codecs and the terminating endpoint selects a pair of asymmetric (mismatched) codecs, the originating endpoint initiates separate H.245 procedures to correct the asymmetric codec situation.

Fast connect is backward compatible with H.323 Version 1 configurations.

Call Termination

Either gateway may terminate a call in one of the following ways:

1. Discontinuing transmission of video at the end of a complete picture and then closes all logical channels for video.

2. Discontinuing transmission of data and then closes all logical channels for data.

3. Discontinuing transmission of voice and then closes all logical channels for voice.

4. Transmitting the H.245 endSessionCommand message in the H.245 control channel, indicating to the far end that it wishes to disconnect the call and then discontinues H.245 message transmission.

5. Waiting to receive the endSessionCommand message from the other gateway and then closes the H.245 control channel.

6. Sending a release complete message if the call signaling channel is open and the channel is closed.

7. Clearing the call by using the procedures defined below.

An endpoint receiving an endSessionCommand message without first having transmitted it carries out steps 1 and 7 above, except that in Step 5, the gateway waits for the endSessionCommand message from the first endpoint.

Terminating a call may not terminate a conference; a conference may be explicitly terminated using an H.245 message (**dropConference**). In this case, the gateways wait for the multipoint controller to terminate the calls as described.

In networks that contain a gatekeeper, the gatekeeper needs to know about the release of bandwidth. After performing steps 1 to 6 in the preceding section, each endpoint transmits an H.225 disengage request (DRQ) message (3) to its gatekeeper as shown in Figure 5. The gatekeeper responds with a disengage confirm (DCF) message (4). After sending the DRQ message, the endpoints do not send further unsolicited information request response (IRR) messages that relate to that call to the gatekeeper. At this point, the call is terminated. Figure 5 shows the direct call model. The DRQ and DCF messages are sent on the RAS channel.

Cisco IOS H.323 gateways will terminate a call if a TCP connection is closed while the call is in progress, or if a TCP connection error is detected when signaling message are sent or received.

Figure 5   *Call Termination Direct Call Model*
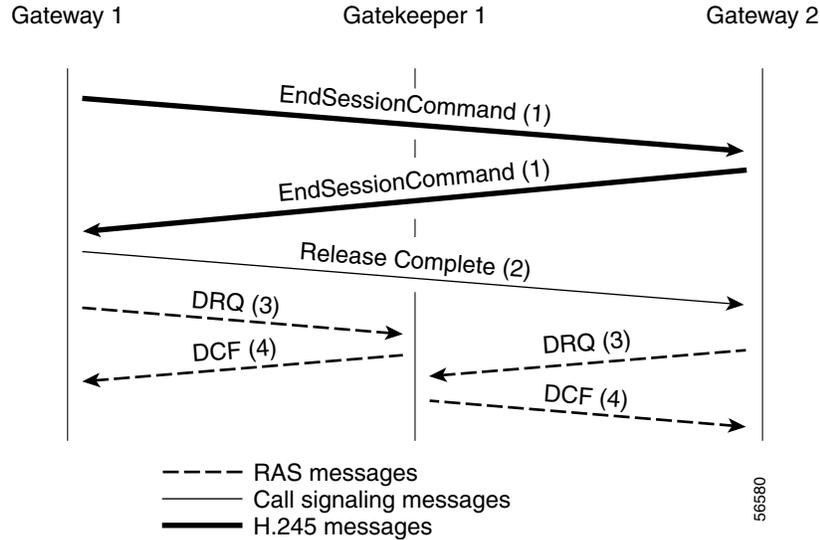
Security

Security for RAS protocol signaling between H.323 endpoints and gatekeepers is enhanced in H.323 Version 2 software by including secure endpoint registration of the Cisco gateway to the Cisco gatekeeper and secure per-call authentication. In addition, it provides for the protection of specific messages related to Open Settlement Protocol (OSP) and to other messages as required via encryption tokens. The authentication type is "password with hashing" as described in the ITU H.235 specifications. Specifically, the encryption method is to use the MD5 algorithm, with password hashing. This functionality is provided by the **security token required-for** command on the gatekeeper and the **security password** command on the gateway.

The gatekeeper can interact with a RADIUS security server to perform the authentications. The gateway can also authenticate an external application by using the Gatekeeper Transaction Message Protocol (GKTMP) application programming interface (API).

Per-call authentication is accomplished by validating account and pin numbers that are entered by the user connected to the calling gateway by using an IVR prompt.

The security mechanisms described above require the gateway and gatekeeper clocks to be synchronized within 30 seconds of each other by using a Network Time Protocol (NTP) server.

Additional References

The following sections provide references related to H.323.

Related Documents

Related Topic	Document Title
Cisco IOS Release 12.4	• Cisco IOS Release 12.4 Configuration Guides • Cisco IOS Release 12.4T Configuration Guides • Cisco IOS Release 12.4 Command References • Cisco IOS Voice Configuration Library http://www.cisco.com/en/US/docs/ios/12_3/vvf_c/cisco_ios_voice_configuration_library_glossary/vcl.htm ✎ **Note** This website contains the library preface and glossary.
Cisco IOS Release 12.3	• Cisco IOS Release 12.3 documentation • Cisco IOS voice commands • Cisco IOS Voice Troubleshooting and Monitoring Guide • Tcl IVR Version 2.0 Programming Guide
Cisco IOS Release 12.2	• *Cisco IOS Voice, Video, and Fax Configuration Guide*, Release 12.2 at http://www.cisco.com/en/US/docs/ios/12_2/voice/configuration/guide/fvvfax_c.html
Troubleshooting and Debugging guides	• Cisco IOS Debug Command Reference, Release 12.4 at http://www.cisco.com/en/US/docs/ios/debug/command/reference/db_book.html • *Troubleshooting and Debugging VoIP Call Basics* at http://www.cisco.com/en/US/tech/tk1077/technologies_tech_note09186a0080094045.shtml • *VoIP Debug Commands* at http://www.cisco.com/en/US/docs/routers/access/1700/1750/software/configuration/guide/debug.html
Cisco Unified Border Element Configuration Examples	• Local-to-remote network using the IPIPGW http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a00801b0803.shtml • Remote-to-local network using the IPIPGW: http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080203edc.shtml • Remote-to-remote network using the IPIPGW: http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080203edd.shtml • Remote-to-remote network using two IPIPGWs: http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080203edb.shtml

Related Topic	Document Title
Related Application Guides	• Cisco Unified Communications Manager and Cisco IOS Interoperability Guide • Cisco IOS Fax, Modem, and Text Support over IP Configuration Guide • "Configuring T.38 Fax Relay" chapter • Cisco IOS SIP Configuration Guide • Cisco Unified Communications Manager (CallManager) Programming Guides at: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_programming_reference_guides_list.html • *Quality of Service for Voice over IP* at http://www.cisco.com/en/US/docs/ios/solutions_docs/qos_solutions/QoSVoIP/QoSVoIP.html

Standards

Standards	Title
ITU-T E.164	Overall network operation, telephone service, service operation and human factors
ITU-T H.225 Version 2	Call signalling protocols and media stream packetization for packet-based multimedia communication systems
ITU-T H.235	Security and encryption for H-Series (H.323 and other H.245-based) multimedia terminals
ITU-T H.323	Packet-based multimedia communications systems
ITU-T H.450	Supplementary services for multimedia

MIBs

MIBs	MIBs Link
• CISCO-GATEKEEPER-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 2833	*RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals*

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Basic H.323 Setup and Management

This chapter describes nonconfigurable H.323 features.

Feature History for Gatekeeper Ecosystem Interoperability

Release	Modification
12.1(1)T	This feature was introduced.

Feature History for Gatekeeper Management Statistics

Release	Modification
12.2(15)T	This feature was introduced, and the CISCO-GATEKEEPER-MIB was enhanced to display gatekeeper-management statistics.

Feature History for Gateway-to-Gatekeeper Billing Redundancy

Release	Modification
12.1(1)T	This feature was introduced.

Feature History for H.323 Call Redirection Enhancements

Release	Modification
12.1(5)XM	This feature was introduced.
12.2(2)T	This feature was integrated into this release.
12.2(2)XB1	This feature was implemented on the Cisco AS5850.
12.2(11)T	This feature was integrated into this release.

Feature History for H.323 Version 2 Enhancements

Release	Modification
12.0(5)T	This feature was introduced.
12.1(5)XM2	Support was added for the Cisco AS5350 and Cisco AS5400.
12.2(2)XA	The **call rscmon update-timer** command was added.

CISCO

12.2(4)T	The **call rscmon update-timer** command was integrated into this release. Support for the Cisco AS5300, Cisco AS5350, and Cisco AS5400 is not included.
12.2(2)XB1	This feature was implemented on the Cisco AS5850.
12.2(11)T	This features was integrated into this release.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Note For more information about these and other related Cisco IOS voice features, see the following:

- "H.323 Overview" section on page 7

- For information about the full set of Cisco IOS voice features, see the entire Cisco IOS Voice Configuration Library—including library preface, glossary, and other documents—at http://www.cisco.com/en/US/docs/ios/12_3/vvf_c/cisco_ios_voice_configuration_library_glossary/vcl.htm

Contents

Prerequisites for Basic H.323 Setup and Management

Prerequisites are described in the "Prerequisites for Configuring an H.323 Network" section on page 7.

Restrictions for Basic H.323 Setup and Management

Restrictions are described in the "Restrictions for Configuring an H.323 Network" section on page 8.

How to Set Up and Manage Basic H.323 Features

This section contains the following information:

Managing Basic (Nonconfigurable) Gateway Features

The following sections describe H.323 features on the gateway that do not require user configuration:

H.323 Signaling

When interworking with ISDN, with T-1 channel-associated signaling (CAS), and with E-1 R2 services from the PSTN, H.323 signaling enables VoIP networks to properly signal the setup and teardown of calls. In-band tones and announcements are generated as needed at the originating or terminating switch. When a tone is played at the destination switch, the backward voice path from the called party to the calling party is cut through early so that the calling party can hear the tone or announcement. To prevent fraudulent calls, the voice path is cut through in both directions only after the connect message is received from the destination. The call progress indicator, which signals the availability of in-band communication, is carried end to end as required when interworking with ISDN and CAS protocols.

The H.323 signaling feature prevents unexpected behavior, such as early alerting (when an alert message is returned immediately after a call proceeding message is sent), to ensure that the calling party does not hear conflicting call progress information, such as a ringback tone followed by a busy tone, and does not miss hearing a tone or announcement when one should play. Support for network-side ISDN and reduction in the risk of speech clipping is also addressed.

The H.323 signaling feature is dependent on Cisco H.323 gateways, gatekeepers, and VoIP features.

H.323 signaling provides the following capabilities:

End-to-End Alerting

Early alerting is prevented in these ways:

- For calls that terminate at an ISDN switch—The terminating gateway sends an alert message to the originating gateway only after it receives an alert message from the terminating switch.

- For calls that terminate at a CAS switch—The terminating gateway sends a progress message, rather than an alert message, to the originating gateway after it receives a setup message.

Cut-Through of Voice Path

When tones and announcements are generated at the destination switch, the backward voice path from the called party to the calling party is cut through before the tones and announcements are played. This allows announcements, such as "The number you have called has been changed," or allows tones for error conditions, such as network congestion, to be forwarded to the calling party. To prevent fraudulent calls, the originating gateway does not perform full cut-through until it receives a connect message from the destination switch. Cut-through is performed as follows:

- For calls that terminate at an ISDN switch—The terminating gateway performs backward cut-through when it receives an alert or progress message and full cut-through (both directions) when it receives a connect message. The originating gateway performs backward cut-through when it receives a call proceeding message and full cut-through when it receives a connect message.

- For calls that terminate at a CAS switch—The terminating gateway performs backward cut-through after it sends a progress message and full cut-through (both directions) when it receives an off-hook signal. The originating gateway performs backward cut-through when it receives a progress message and full cut-through when it receives a connect message.

> **Note** If the originating or terminating gateway sends a call proceeding message and then receives a call proceeding message with a progress indicator of 1, 2, or 8, the gateway converts this call proceeding message into a progress message with a corresponding PI.

H.245 Initiation

To avoid speech clipping, H.245 capabilities are now initiated at the originating gateway at the earliest possible moment, when the originating gateway receives a call proceeding message from the terminating gateway. Previously, call proceeding messages were not passed end to end across the VoIP network; H.245 was initiated only after the originating gateway received an alert message.

Overlap Dialing

To enhance overlap dialing, the call proceeding message is now passed transparently from the terminating switch to the originating switch if the originating switch does not include the sending complete information element in the setup message. The call proceeding message notifies the originating switch that the terminating switch has collected all dialed digits that are required to route the call. If the originating switch sends a sending complete IE, the originating gateway responds with a call proceeding message, and the session application drops the call proceeding message sent by the terminating switch.

H.323 Call Statistics

Beginning with Cisco IOS Release 12.2(4)T, enhancements to H.323 call statistics allow you to clear the gateway counters, display H.323 messages that have been sent and received, obtain statistics on the reasons calls are disconnected, and display debug output for various components within the H.323 subsystem. To enable these enhancements, the following commands are available: **clear h323 gateway** command, **show h323 gateway** command, and **debug cch323** command.

Note Using any of the **debug cch323** commands could slow your system and flood the TTY if there is significant call traffic.

The enhancements to H.323-call-statistics commands do not affect Cisco H.323 configurations. Therefore, there are no configuration tasks in this document.

To display and clear H.323 call statistics, use the following commands in privileged EXEC mode.

SUMMARY STEPS

1. **clear h323 gateway** [**cause-code stats** | **h225** | **ras**]

2. **show h323 gateway** [**cause-code stats** | **h225** | **ras**]

3. **debug cch323** {**all** | **error** | **h225** | **h245** | **ras** | **rawmsg** | **session**}

DETAILED STEPS

	Command	Purpose
Step 1	`clear h323 gateway [cause-code stats	h225	ras]`  **Example:** `Router# clear h323 gateway`	Clears the H.323 gateway counters. Keywords are as follows: • **cause-code stats**—Disconnect cause-code stats counters • **h225**—H.225 counters • **ras**—RAS counters **Note** If this command is entered without any of the optional keywords, all counters are cleared. If the command is entered with an optional keyword, only counters associated with that keyword are cleared.

	Command	Purpose
Step 2	**show h323 gateway** [**cause-code stats** | **h225** | **ras**] **Example:** Router# show h323 gateway	Displays statistics for H.323 gateway messages that have been sent and received and displays the reasons for which H.323 calls have been disconnected. Keywords are as follows: • **cause-code stats**—Displays the disconnect cause codes that the H.323 subsystem has received. A disconnect can originate either from the far-end gateway or from the opposite call leg on the local gateway. • **h225**—Lists cumulative counts of the number of H.225 messages that have been sent and received since the counters were last cleared. • **ras**—Lists the counters for RAS messages that have been sent to and received from the gatekeeper. **Note** If this command is entered without any of the optional keywords, all counters are displayed. If the command is entered with an optional keyword, only counters associated with that keyword are displayed.
Step 3	**debug cch323** {**all** | **error** | **h225** | **h245** | **ras** | **rawmsg** | **session**} **Example:** Router# debug cch323 all	Provides debug output for various components within the H.323 subsystem. Keywords are as follows: • **all**—Enables all **debug cch323** commands. • **error**—Traces errors encountered in the H.323 subsystem and can be used to help troubleshoot problems with H.323 calls. • **h225**—Traces the state transition of the H.225 state machine on the basis of the processed event. • **h245**—Traces the state transition of the H.245 state machine on the basis of the processed events. • **ras**—Traces the state transition of the RAS state machine on the basis of the processed events. • **rawmsg**—Troubleshoots raw message buffer problems. • **session**—Traces general H.323 events and can be used to troubleshoot H.323 problems.

Source Call Signal Address

Note To learn about restrictions that apply, see the "Source Call Signal Address and H.245 Empty Capabilities Set Restrictions" section on page 10.

Source call signal address allows a source call-signal address field to be included in the ARQ.

Previously, in the Cisco IOS implementation of H.323 gateway software, if the terminating gateway was registered to an H.323 gatekeeper and used RAS, the ARQ message sent for each incoming call did not contain the H.225 source call signal address (CSA). The source CSA is an optional parameter in the ARQ message. The source CSA is also an optional parameter in the H.225 call setup message sent by the originating endpoint.

source call signal address also allows for the source CSA parameter to be included in the ARQ message, as illustrated by the message sequence shown in Figure 1.

Figure 1   *Source Call Signal Message Sequence*
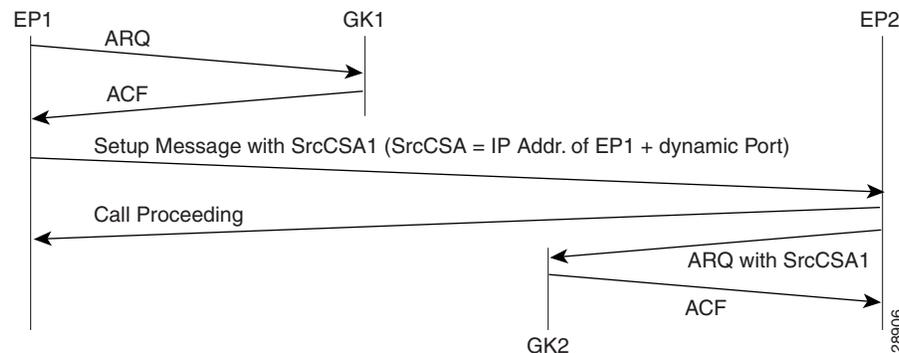
In the message sequence shown in Figure 1, the ARQ messages are enhanced to send the source CSA. The originating gateway (EP1) sends the H.225 setup message to the destination gateway. The setup message contains the source CSA parameter, which is the combination of the IP address of the originator and the dynamic TCP port number used or obtained for the H.225 call signaling channel. If the terminating gateway (EP2) accepts the call upon receipt of the setup message, the gateway sends an ARQ message to the gatekeeper. The terminating gateway retrieves the source CSA parameter sent by the originating gateway in the setup message. It then sends an ARQ message to the gatekeeper with the source CSA parameter. The CSA parameter is optional and has the same value as the source CSA in the received setup message. If the setup message does not contain the source CSA parameter, the terminating gateway determines the source CSA by using the H.225 call-signaling TCP socket connection of the peer endpoint, which it uses in the ARQ message.

If the originating gateway is registered to a gatekeeper and RAS is used as the session target, the originating gateway also sends an ARQ message. This ARQ does not include the optional source CSA parameter.

Tunneling of Redirecting Number Information Element

An incoming PRI setup message may contain either a Redirecting Number (RDN) Information Element (IE) or an Original Called Number (OCN) IE. These IEs indicate that the call has been redirected (forwarded) and that each message contains the following:

- Destination number (DN) that was originally called
- Reason for the call being redirected
- Other related information

OCN IE is a Nortel variant of the RDN IE.

H.323 Version 2

H.323 Version 2 gateway passes the entire RDN or OCN IE from an incoming PRI message into the H.225 SETUP message. The IE is encapsulated in the nonStandardData field within the user-to-user information element (UUIE) of the H.225 SETUP message. The nonStandardData field can contain the encapsulated RDN or OCN IE and a tunneled global, signaling, and control standard QSIG message, or it can contain only the OCN or RDN. Cisco and other third-party H.323 endpoints can access the redirected information by decoding the nonStandardData field. In accordance with the H.225 specification, the nonStandardData is ignored by third-party endpoints and causes no interoperability problems.

For redirected PRI calls that are routed to a Cisco gateway, that are sent using H.323 to another Cisco gateway, and that exit the gateway using PRI, the RDN/OCN IE is tunneled from the source gateway to the destination gateway. The incoming PRI setup message is tunneled through H.225 and is encoded into the outgoing PRI setup message by the destination gateway.

Tunneling the RDN or OCN IE is important for applications such as Unified Messaging servers that need to know the telephone number that was originally dialed so as to access the correct account information.

H.323 Version 4

H.323 Version 4 introduces a standard-based RDN IE in the H.225 SETUP message in Cisco IOS Release 12.3(11)T. The RDN IE is sent as a Q.931 IE in the H.225 SETUP message. The nonStandardData RDN and OCN IE are still supported for backward compatibility. When both H.225 Q.931 RDN IE and nonStandardData IE are received, the RDN in the H.225 Q.931 is decoded and the nonStandardData is not decoded.

H.323 Call Redirection

The user-to-user information element (UUIE) of the Facility message is used primarily for call redirection. The UUIE contains a field, facilityReason, that indicates the nature of the redirection. The H.323 Call Redirection Enhancements feature adds support for two of the reasons: routeCallToGatekeeper and callForwarded. It also provides a nonstandard method for using the Facility message to effect call transfer.

Route Call to Gatekeeper

There are two situations in which the Cisco H.323 gateway might receive or generate a facility message with a routeCallToGatekeeper reason.

- The gateway receives a facility message with routeCallToGatekeeper as a response to its H.225 SETUP message. Upon receiving the Facility message, the Cisco H.323 gateway attempts to route the call to the new gatekeeper, using the new IP address specified in the alternativeAddress field of the facility message.
  - If the IP address is not available, the gateway ignores the facility message and sends a release complete toward the original destination end-point. The release complete message contains a ReleaseCompleteReason of facilityCallDeflection.
  - If the IP address is available, the gateway sends a disengage request (DRQ) message to the gatekeeper and waits for the disengage confirmation (DCF) message before it sends the SETUP message to the new destination gatekeeper.
- During the admission request (ARQ) phase of a call, a gatekeeper might determine that a call, which has come through an intermediate gateway, needs to be routed to another gatekeeper. The gatekeeper sends an admission rejection (ARJ) message with a RejectReason of routeCallToGatekeeper to the gateway. Upon receiving the message, the intermediate Cisco H.323 gateway sends a Facility

message to the originator of the SETUP message. This message indicates that the SETUP message should be sent to another address. (The gateway includes the callSignalAddress from ARJ in the alternativeAddresss field of the Facility message.) Upon receiving the Facility message, the calling gateway terminates the initial call and sends a new SETUP message to the specified gatekeeper, using the new IP address specified in the alternativeAddress field of the facility message. If the callSignalAddress is not provided, the gateway does not send the Facility message and the call is terminated without any rerouting.

Call Forward

In certain cases, an H.323 endpoint might determine that a call needs to be forwarded. The endpoint then sends a Facility message to the gateway with a facilityReason of callForwarded. This message includes the address of the new destination (either an alternativeAddress or alternativeAliasAddress). Upon receiving the Facility message, the Cisco H.323 gateway sends a release complete to the original destination endpoint and initiates a new call using the new destination address supplied in the Facility message. The release complete message contains a ReleaseCompleteReason of facilityCallDeflection. If the gateway is registered with a gatekeeper, the gateway sends a DRQ to the gatekeeper and waits for the DCF before sending a setup message to the destination gatekeeper.

The Facility message must contain an E.164 address in the alternativeAliasAddress field. If no address is included, the Facility message is ignored. The E.164 is required because the call forwarding process initiates a new call, which may be subject to authentication processes that can handle only E.164 addresses.

If the Facility message contains both and IP address (in the alternativeAddress field) and an E.164 address (in the alternativeAliasAddress field), the gateway first attempts to find a match for the new E.164 and the dial-peer. If there is no match, the gateway uses the same incoming peer to determine if there is a matching peer to reroute the call. If there is no match to the incoming peer, the message is ignored.

Call Transfer

Note   To learn about restrictions that apply, see the "Call Transfer Restrictions" section on page 10.

If a Facility message with a facilityReason of callForwarded is received after the call has been accepted, it is considered a call transfer. In this case, the Cisco H.323 gateway places the call on hold and initiates a new call using the address (alternativeAddress or alternativeAliasAddress) supplied in the Facility message.

As with call forwarding, the Facility message must contain an E.164 address in the alternativeAliasAddress field. If no address is included, the Facility message is ignored. The E164 is required because the call forwarding process initiates a new call, which may be subject to authentication processes that can handle only E.164 addresses.

If the Facility message contains both and IP address (in the alternativeAddress field) and an E.164 address (in the alternativeAliasAddress field), the gateway first attempts to find a match for the new E.164 and the dial-peer. If there is no match, the gateway uses the same incoming peer to determine if there is a matching peer to reroute the call. If there is no match to the incoming peer, the message is ignored.

Unlike in call forwarding case, the Facility message is accepted by both the called side and the originating side.

Note This use of call forwarded is not defined by ITU standard.

Multizone Features

Cisco multizone software enables the Cisco gateway to provide information to the gatekeeper using additional fields in the RAS messages. The gatekeeper no longer terminates a call if it is unable to resolve the destination E.164 phone number with an IP address.

Previously, the source gateway attempted to set up a call to a destination IP address as provided by the gatekeeper in an admission confirm (ACF) message. If the gatekeeper was unable to resolve the destination E.164 phone number to an IP address, the incoming call was terminated.

Multizone software allows a gatekeeper to provide additional destination information and modify the destinationInfo field in the ACF message. The gateway includes the canMapAlias-associated destination information in setting up the call to the destination gateway.

The gatekeeper indicates to the gateway that the call should be destined to a new E.164 number by sending an ACF message with an IP address of 10.0.0.0 in the destCallSignalAddress field and the new destination E.164 phone number in the destinationInfo field.

The gateway that receives such an ACF falls back to routing the call on the basis of this new E.164 address and performing another lookup of the configured dial plan for the gateway. If the gateway routes the call on the basis of the new E.164 address, the call might be routed back to the PSTN or to an H.323 endpoint.

Codec Negotiation

Codec negotiation allows the gateway to offer several codecs during the H.245 capability exchange phase and to ultimately settle on a single common codec during the call establishment phase. Offering several codecs increases the probability of establishing a connection because there is a greater chance of overlapping voice capabilities between endpoints. Normally, only one codec can be specified when a dial peer is configured, but codec negotiation allows a prioritized list of codecs associated with a dial peer to be specified. During call establishment, the router uses the highest-priority codec from the list that it has in common with the remote endpoint. It also adjusts to the codec selected by the remote endpoint so that a common codec is established for both the receive and send voice directions.

When a call is originated, all the codecs associated with the dial peer are sent to the terminating endpoint in the H.245 terminal capability set message. At the terminating endpoint, the gateway advertises all the codecs that are available in firmware in its terminal capability set. If there is a need to limit the codecs advertised to a subset of the available codecs, a terminating dial peer must be matched that includes this subset. The **incoming called-number** command in dial peer configuration mode can be used to force this match.

Supported codecs (Table 1) are available for use with Cisco H.323 Version 2 software.

Table 1   Codec Default Packet Size

Codecs	Range (bytes)	Default (bytes)	Bit Rate (kbps)
G.711ulaw	40–240	160	64
G.711alaw	40–240	160	64
G.723r63	24–240	24	6.3
G.723r53	20–240	20	5.3

Table 1   Codec Default Packet Size

Codecs	Range (bytes)	Default (bytes)	Bit Rate (kbps)
G.723ar63	24–240	24	6.3
G.723ar53	20–240	20	5.3
G.726r32	20–240	40	32
G.726r24	15–240	30	24
G.726r16	10–240	20	16
G.728	10–240	10	16
G.729br8	10–240	20	8
G.729r8 pre-ietf	10–240	20	8
G.729r8	10–240	20	8

Note
- A separate codec for G.729 Annex B is included, which adds Annex B functionality to G.729. A separate codec for G.723.1 Annex A adds Annex A functionality to G.723.1.
- The Annex B functionality added to G.729 and the Annex A functionality added to G.723.1 are the built-in, codec-specific voice-activated detection/calling tone (VAD/CNG) functions.

H.245 Empty Capabilities Set

Note   To learn about restrictions that apply, see the "Source Call Signal Address and H.245 Empty Capabilities Set Restrictions" section on page 10.

Empty capabilities set support is a mandatory part of the H.323 Version 2 standard. It is used by applications to redirect the voice media stream. This feature is particularly useful for applications such as the following:

- Selsius IP phones, which rely on a hub or call manager to direct the media stream to IP phones.
- Unified messaging for which it is desirable to redirect the media stream to various message servers for message playout.

The empty capabilities set feature was added to provide a way to redirect RTP streams. The RTP streams are redirected as follows:

- The sequence starts with the an empty capabilities set being received at an endpoint.
- After an open logical channel (OLC) is established (or if in the middle of this process) one of the endpoints sends an empty capabilities set message.
- When the empty capabilities set message is received, the other endpoints close the logical channel if any was opened with that endpoint and move to a pause state, waiting for a nonempty capability set message.

After receiving the nonempty capabilities set message, the endpoint moves to the beginning of Phase B, which is the initial communication and capabilities exchange, as described in H.323 Version 3 (June 1999), item 8.4.6.

In other words, the exchange of the capabilities message determines a master/slave relationship, and a new OLC message is created to open a new logical channel with another endpoint. From this point on, the RTP streams are sent to the new endpoint.

Lightweight Registration

Before the release of its H.323 Version 2 software, Cisco gateways reregistered with the gatekeeper every 30 seconds. Each registration renewal used the same process as the initial registration, even though the gateway was already registered with the gatekeeper. These registration renewals generated considerable overhead at the gatekeeper.

Cisco H.323 Version 2 software defines a lightweight registration procedure that still requires the full registration process for initial registration but that uses an abbreviated renewal procedure to update the gatekeeper and minimize overhead.

Lightweight registration requires each endpoint to specify a time-to-live (TTL) value in its registration request (RRQ) message. When a gatekeeper receives an RRQ message with a TTL value, it returns an updated TTL timer value in a registration confirmation (RCF) message to the endpoint. Shortly before the TTL timer expires, the endpoint sends an RRQ message with the KeepAlive field set to TRUE, which refreshes the existing registration.

It is not required that an H.323 Version 2 endpoint indicate a TTL in its registration request. If the endpoint does not indicate a TTL, the gatekeeper assigns one and sends it to the gateway in the RCF message. No configuration changes are permitted during a lightweight registration, so all fields other than the endpointIdentifier, gatekeeperIdentifier, tokens, and TTL are ignored. In the case of H.323 Version 1 endpoints that cannot process the TTL field in the RCF, the gatekeeper probes the endpoint with information requests (IRQs) for a predetermined grace period to see if the endpoint is still alive.

H.450.2 Call Transfer

Call transfer allows an H.323 endpoint to redirect an answered call to another H.323 endpoint. Cisco gateways support H.450.2 call transfer as the transferring and transferred-to party. The transferring endpoint must be an H.450-capable terminal; the Cisco gateway cannot act as the transferring endpoint. Gatekeeper-controlled or gatekeeper-initiated call transfer is not supported.

> **Note** Certain devices are limited in their support of H.450. The Cisco 1700 and Cisco uBR820 platforms do not support IVR. Therefore, these platforms are not able to act as H.450 transferring endpoints.

H.450.2 specifies two variants of call transfer:

- Transfer without consultation—The transferring endpoint supplies the number of the transferred-to endpoint as part of the transfer request, and the two remote endpoints are transferred together. A Cisco gateway cannot be the transferring endpoint.

- Transfer with consultation—This feature is not currently supported.

H.450.3 Call Deflection

Call deflection is a feature under H.450.3 Call Diversion (Call Forwarding) that allows a called H.323 endpoint to redirect the unanswered call to another H.323 endpoint. Cisco gateways support H.450.3 call deflection as the originating, deflecting, and deflected-to gateway. The Cisco gateway as the deflecting gateway supports invocation of call deflection only by using an incoming PRI QSIG message (call deflection cannot be invoked by using any other trunk type).

If the deflecting endpoint is a Cisco gateway, the telephony endpoint on the PRI of the deflecting gateway invokes call deflection by sending an equivalent QSIG reroute invoke request within a FACILITY message to the gateway. The deflecting gateway then uses the procedures outlined in the H.450.3 call deflection standard to transfer the call to another endpoint. Note that the initiation of deflection using QSIG reroute invoke is valid only on calls that arrived as H.323 calls at the deflecting gateway. In other words, for calls that arrive at the gateway through a telephony interface (such as a hairpin call) or by using a non-H.323 IP protocol, QSIG reroute invoke is ignored.

Cisco H.323 Version 2 software does not support gatekeeper-controlled or gatekeeper-initiated call deflection.

Note Certain devices are limited in their support of the H.450 standard. The Cisco AS5800 is not able to convert QSIG to H.450. The Cisco 1700 and Cisco uBR820 do not support IVR. Therefore, these devices are not able to act as H.450 deflecting endpoints.

Gateway Support for a Network-Based Billing Number

Gateway support for a network-based billing number informs the gatekeeper of the specific voice port or T1/E1 span from which an incoming call entered the ingress gateway. This is done using a Cisco-proprietary, nonstandard field that has been added to the ARQ message sent by the ingress gateway. No configuration is necessary for this feature.

Answer Supervision Reporting

Answer supervision reporting is an enhancement to the information request (IRR) Registration, Admission, and Status (RAS) protocol message that enables gatekeepers to maintain call accounting information by reporting the call connection time of connected calls to the gatekeeper.

In H.323 configurations, the endpoint (gateway) uses direct call-routed signaling. Gatekeepers do not have real-time knowledge or control over the state of a call and are dependent on the endpoints to provide them with necessary real-time information, such as call connect time, call termination time, and call termination reason.

When a call ends, the gateway sends a DRQ message with the BillingInformationToken (which contains the duration of the call) to the gatekeeper. If for some reason the gatekeeper does not receive the DRQ message, the gatekeeper does not have the information about when the call started or the duration of the call, which is necessary to maintain accounting information.

Answer supervision reporting allows the call connection time to be reported to the gatekeeper upon the connection of a call and at periodic intervals thereafter. Answer supervision reporting adds a proprietary Cisco parameter, the call connection time, to the perCallInfo parameter in the nonStandardData field, which is located in the IRR message. When a connect message is received, the originating gateway sends the unsolicited IRR message to its gatekeeper. On sending a connect message, the terminating gateway sends the unsolicited IRR message to its gatekeeper. If the ACF message has a nonzero value for the IRR frequency parameter, the gateway sends the unsolicited IRR message to its gatekeeper at periodic intervals, which are determined by the value in the IRRfrequency parameter.

With the exception of containing the call connection time in the perCallInfo parameter, the IRR message and its functionality remain the same.

Managing Basic (Nonconfigurable) Gatekeeper Features

The following sections describe H.323 features on the gateway that do not require user configuration:

- Gateway-to-Gatekeeper Billing Redundancy, page 36
- Ecosystem Gatekeeper Interoperability, page 36
- Gatekeeper-Management Statistics, page 38

Gateway-to-Gatekeeper Billing Redundancy

Gateway-to-gatekeeper billing enhances the accounting capabilities of the Cisco H.323 gateway and provides support for VocalTec™ gatekeepers. Gateway-to-gatekeeper billing redundancy provides for redundant billing information to be sent to an alternate gatekeeper if the primary gatekeeper to which a gateway is registered becomes unavailable.

During the process of establishing a call, the primary gatekeeper sends an ACF message to the registered gateway. The ACF message includes the billing information of the user and an access token. To provide the billing information to an alternate gatekeeper if the primary gatekeeper is unavailable when the call session ends, the access token information sent in the ACF message is also included in the DRQ message that is sent to the alternate gatekeeper.

This feature enables the alternate gatekeeper to obtain the billing information required to successfully complete the transaction.

Ecosystem Gatekeeper Interoperability

Note To learn about restrictions that apply, see the "Ecosystem Gatekeeper Interoperability Restrictions" section on page 10.

Ecosystem gatekeeper interoperability adds support for the alternate gatekeeper field (altGKInfo) in the gatekeeper rejection (GRJ), registration rejection (RRJ), and admission rejection (ARJ) messages. This allows a gateway to move between gatekeepers during the GRQ, RRQ, and ARQ phases. There is no need for gateway reconfiguration or for a gatekeeper failover in the gateway.

Gateways can be configured to switch from their primary gatekeeper to an alternate gatekeeper if a failure or outage occurs. If an outage occurs and gateways move from one gatekeeper to another, there may be an imbalance in the number of gateways registered to each gatekeeper. The ecosystem gatekeeper interoperability feature helps to restore the balance (when the outage has been corrected) by allowing some of the gateways to be moved back to their proper gatekeepers.

The altGKInfo consists of two subfields: the alternateGatekeeper and the altGKisPermanent flag. The alternateGatekeeper is the list of alternate gatekeepers. The altGKisPermanent is a flag that indicates whether the gatekeepers in the associated alternateGatekeeper field are permanent or temporary.

- If the current state of the altGKisPermanent flag is TRUE, the new altGKInfo of any RAS message received from one of the alternate gatekeepers is accepted and the new list replaces the existing list.
- If the current state of the altGKisPermanent flag is FALSE, the altGKInfo of any RAS message received from one of the alternate gatekeepers is ignored.

If the current permanent gatekeeper becomes nonresponsive and the altGKisPermanent flag is set to FALSE, the gateway sets the internal state of the altGKisPermanent flag to TRUE. This allows the gateway to accept the alternate gatekeeper list from one of the gatekeepers in the existing alternate gatekeeper list.

The handling of the altGKInfo field varies depending on whether it is included in a GRJ or an RRJ message.

AltGKInfo in GRJ Messages

When the gateway accepts the alternate gatekeeper list from the GRJ, the gateway sends a GRQ message to a gatekeeper on the list. The selection is based on priority of the alternate gatekeepers. Each alternate gatekeeper is tried until a GCF message is received.

If the gateway receives a GRJ message without the AltGKInfo field, it accepts the rejection. Because this is the first phase for the gateway to contact a gatekeeper, the gateway is considered lost without a gatekeeper.

During the GRQ phase, the gateway ignores the value of the altGKisPermanent flag in any RAS message and sets the value internally to TRUE.

AltGKInfo in RRJ Messages

When the gateway accepts the alternate gatekeeper list from the first RRJ message, the gateway retransmits an RRQ message to a gatekeeper on the alternate gatekeeper list. The selection is based on priority of the alternate gatekeepers.

The retransmission of the RRQ message depends on the type of RRQ (full or lightweight), the current state of the altGKisPermanent flag, and the current state of the needToRegister flag of each alternate gatekeeper as follows:

- If the state of the altGKisPermanent flag is TRUE and the state of the needToRegister flag is NO, the gateway retransmits the full RRQ to an alternate gatekeeper for full RRQs and a lightweight RRQ for lightweight RRQs.

- If the state of the altGKisPermanent flag is TRUE and the state of the needToRegister flag is YES, the gateway retransmits the full RRQ to an alternate gatekeeper for full RRQs and lightweight RRQs.

- If the state of the altGKisPermanent flag is FALSE and the state of the needToRegister flag is NO, the gateway retransmits a lightweight RRQ for lightweight RRQs and nothing for full RRQs.

- If the state of the altGKisPermanent flag is TRUE and the state of the needToRegister flag is YES, the gateway does not retransmit the RRQ.

If the gateway receives an RRJ message without the AltGKInfo field, it accepts the rejection and returns to the GRQ phase. If the state of the altGKisPermanent flag is FALSE, the gateway sends the GRQ message to the original gatekeeper that sent the first RRJ. If the state of the altGKisPermanent flag is TRUE, the gateway sends the GRQ to the current gatekeeper.

If the current state of the altGKisPermanent flag is TRUE, then the next RAS message is sent to the new gateway. Otherwise, the next RAS message is sent to the original gatekeeper.

If the gateway exhausts the list of alternate gatekeepers without receiving any response from an alternate gatekeeper, the gateway returns to the GRQ phase.

Note For more information on the Cisco ecosystem gatekeeper interoperability feature, see the "Configuring Alternate-Gatekeeper Support" section on page 66.

Gatekeeper-Management Statistics

Performance-management parameters provide gatekeeper-management statistics that may be used to monitor a network and troubleshoot problems on the network. Parameters provide statistics such as the following:

- Number of calls that originate and terminate from a specific location
- Number of ongoing calls
- Aggregate messaging information per zone
- Equipment behavior
- Registration and unregistration information
- Counter information (such as location requests [LRQs]) to gauge the level of activity

Statistics are counted when the Registration, Admission, and Status (RAS) messages are sent and received by the gatekeeper. They are in raw form and reflect only a count of messages. Retries or retransmissions are not counted.

There are two ways to monitor gatekeeper-management statistics:

- Using the MIB module—The MIB module consists of a repository of characteristics and parameters that support the gatekeeper function. The MIB gathers statistics and responds to queries as specified by the Simple Network Management Protocol (SNMP). SNMP operations are supported on the object identifiers (OIDs) for the managed objects. These OIDs can configure, manage, or analyze aspects of SNMP operation. Gatekeeper-management statistics are supported by the CISCO-GATEKEEPER-MIB; parameters for this MIB are shown in a table that you can access on your network management station.
- Using the command-line interface as in the following steps.

Displaying and Clearing Gatekeeper-Management Statistics

To display and clear gatekeeper-management statistics, use the following commands beginning in global configuration mode.

Prerequisites

- Perform the prerequisite tasks listed in the "Prerequisites for Configuring an H.323 Network" section on page 7.

SUMMARY STEPS

1. **show gatekeeper performance stats**
2. **clear h323 gatekeeper statistics**
3. **show h323 gatekeeper statistics aggregate**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`show gatekeeper performance stats`
	
	Example:
	`Router# show gatekeeper performance stats`
		Displays performance statistics gathered from the gatekeeper that include per-gatekeeper and per-zone-level statistics, counters, and other gatekeeper-management statistics.

Step 2	`clear h323 gatekeeper statistics`
	
	Example:
	`Router# clear h323 gatekeeper statistics`
		Clears the counters of H.323 gatekeeper statistics.

Step 3	`show h323 gatekeeper statistics aggregate`
	
	Example:
	`Router# show h323 gatekeeper statistics aggregate`
		Displays the system statistics since it was started, regardless of whether or not the counters have been cleared. Without the aggregate keyword, the counters reflects the activity since the last clear command.

Examples

The following sample output displays BASIC gatekeeper-management statistics.

```
Router# show gatekeeper performance stats

-----Gatekeeper Performance Statistics-----

Performance statistics captured since: 00:17:00 UTC Mon Mar 1 1993

Gatekeeper level Admission Statistics:
        ARQs received: 1
        ARQs received from originating endpoints: 0
        ACFs sent: 1
        ACFs sent to the originating endpoint: 0
        ARJs sent: 0
        ARJs sent to the originating endpoint: 0
        ARJs sent due to overload: 0
        Number of concurrent calls: 0
        Number of concurrent originating calls: 0

Gatekeeper level Location Statistics:
        LRQs received: 1
        LRQs sent: 0
        LCFs received: 0
        LCFs sent: 1
        LRJs received: 0
        LRJs sent: 0
        LRJs sent due to overload: 0

Gatekeeper level Registration Statistics:
        RRJ due to overload: 0
        Total Registered Endpoints: 1

Gatekeeper level Disengage Statistics:
        DRQs received: 1
        DRQs sent: 0
        DCFs received: 0
        DCFs sent: 1
        DRJs received: 0
        DRJs sent: 0
```

```
Load balancing events: 0
```

The following CUMULATIVE sample output is the same as for BASIC output; the difference is that the BASIC counters are cleared by the **clear h323 gatekeeper statistics** command, and CUMULATIVE counters are not.

```
Router# show gatekeeper performance stats zone name voip3-2600-2

Performance statistics for zone voip3-2600-2


-----Zone Level Performance Statistics-----

Performance statistics captured since: 00:17:00 UTC Mon Mar 1 1993

Zone level Admission Statistics:
        ARQs received: 1
        ARQs received from originating endpoints: 0
        ACFs sent: 1
        ACFs sent to the originating endpoint: 0
        ARJs sent: 0
        ARJs sent to the originating endpoint: 0
        Number of concurrent total calls: 0
        Number of concurrent originating calls: 0

Zone level Location Statistics:
        LRQs received: 1
        LRQs sent: 0
        LCFs received: 0
        LCFs sent: 1
        LRJs received: 0
        LRJs sent: 0

Zone level Registration Statistics:
        Full RRQs received: 1
        Light RRQs received: 574
        RCFs sent: 576
        RRJs sent: 0
        Total Registered Endpoints: 1

Zone level UnRegistration Statistics:
        URQs received: 0
        URQs sent: 0
        UCFs received: 0
        UCFs sent: 0
        URJs received: 0
        URJs sent: 0
        URQs sent due to timeout: 0

Zone level Disengage Statistics:
        DRQs received: 1
        DRQs sent: 0
        DCFs received: 0
        DCFs sent: 1
        DRJs received: 0
        DRJs sent: 0
```

Toll Fraud Prevention

When a Cisco router platform is installed with a voice-capable Cisco IOS software image, appropriate features must be enabled on the platform to prevent potential toll fraud exploitation by unauthorized users. Deploy these features on all Cisco router Unified Communications applications that process voice calls, such as Cisco Unified Communications Manager Express (CME), Cisco Survivable Remote Site Telephony (SRST), Cisco Unified Border Element (UBE), Cisco IOS-based router and standalone analog and digital PBX and public-switched telephone network (PSTN) gateways, and Cisco contact-center VoiceXML gateways. These features include, but are not limited to, the following:

- Disable secondary dial tone on voice ports—By default, secondary dial tone is presented on voice ports on Cisco router gateways. Use private line automatic ringdown (PLAR) for foreign exchange office (FXO) ports and direct-inward-dial (DID) for T1/E1 ports to prevent secondary dial tone from being presented to inbound callers.

- Cisco router access control lists (ACLs)—Define ACLs to allow only explicitly valid sources of calls to the router or gateway, and therefore to prevent unauthorized Session Initiation Protocol (SIP) or H.323 calls from unknown parties to be processed and connected by the router or gateway.

- Close unused SIP and H.323 ports—If either the SIP or H.323 protocol is not used in your deployment, close the associated protocol ports. If a Cisco voice gateway has dial peers configured to route calls outbound to the PSTN using either time division multiplex (TDM) trunks or IP, close the unused H.323 or SIP ports so that calls from unauthorized endpoints cannot connect calls. If the protocols are used and the ports must remain open, use ACLs to limit access to legitimate sources.

- Change SIP port 5060—If SIP is actively used, consider changing the port to something other than well-known port 5060.

- SIP registration—If SIP registration is available on SIP trunks, turn on this feature because it provides an extra level of authentication and validation that only legitimate sources can connect calls. If it is not available, ensure that the appropriate ACLs are in place.

- SIP Digest Authentication—If the SIP Digest Authentication feature is available for either registrations or invites, turn this feature on because it provides an extra level of authentication and validation that only legitimate sources can connect calls.

- Explicit incoming and outgoing dial peers—Use explicit dial peers to control the types and parameters of calls allowed by the router, especially in IP-to-IP connections used on CME, SRST, and Cisco UBE. Incoming dial peers offer additional control on the sources of calls, and outgoing dial peers on the destinations. Incoming dial peers are always used for calls. If a dial peer is not explicitly defined, the implicit dial peer 0 is used to allow all calls.

- Explicit destination patterns—Use dial peers with more granularity than .T for destination patterns to block disallowed off-net call destinations. Use class of restriction (COR) on dial peers with specific destination patterns to allow even more granular control of calls to different destinations on the PSTN.

- Translation rules—Use translation rules to manipulate dialed digits before calls connect to the PSTN to provide better control over who may dial PSTN destinations. Legitimate users dial an access code and an augmented number for PSTN for certain PSTN (for example, international) locations.

- Tcl and VoiceXML scripts—Attach a Tcl/VoiceXML script to dial peers to do database lookups or additional off-router authorization checks to allow or deny call flows based on origination or destination numbers. Tcl/VoiceXML scripts can also be used to add a prefix to inbound DID calls. If the prefix plus DID matches internal extensions, then the call is completed. Otherwise, a prompt can be played to the caller that an invalid number has been dialed.

- Host name validation—Use the "permit hostname" feature to validate initial SIP Invites that contain a fully qualified domain name (FQDN) host name in the Request Uniform Resource Identifier (Request URI) against a configured list of legitimate source hostnames.

- Dynamic Domain Name Service (DNS)—If you are using DNS as the "session target" on dial peers, the actual IP address destination of call connections can vary from one call to the next. Use voice source groups and ACLs to restrict the valid address ranges expected in DNS responses (which are used subsequently for call setup destinations).

For more configuration guidance, see the "Cisco IOS Unified Communications Toll Fraud Prevention" paper.

Configuring H.323 Gateways

This chapter describes the configuration of H.323 gateways.

Feature History for Basic Service Relationships (H.225 Annex-G)

Release	Modification
12.2(11)T	This feature was introduced.

Feature History for Cisco H.323 Scalability and Interoperability Enhancements for Gatekeepers

Release	Modification
12.2(2)XA	This feature was introduced.
12.2(4)T	This feature was integrated into this release.
12.2(2)XB1	This feature was implemented on the Cisco AS5850.
12.2(11)T	This feature was integrated into this release.

Feature History for Gateway Codec Order Preservation and Shutdown Control

Release	Modification
12.3(1)	This feature was introduced.

Feature History for H.323 Dual Tone Multifrequency Relay Using Named Telephone Events

Release	Modification
12.2(2)XB	The Dual Tone Multifrequency Relay for SIP Calls Using Named Telephone Events feature was introduced. The Media Gateway Control Protocol-Based Fax (T.38) and Dual Tone Multifrequency (IETF RFC 2833) Relay feature was also introduced.
12.2(11)T	H.323 support for DTMF relay was added.

Feature History for H.323 Version 2 Enhancements

Release	Modification
12.0(5)T	This feature was introduced.

12.1(5)XM2	Support was added for the Cisco AS5350 and Cisco AS5400.
12.2(2)XA	The **call rscmon update-timer** command was added.
12.2(4)T	The **call rscmon update-timer** command was integrated into this release. Support for the Cisco AS5300, Cisco AS5350, and Cisco AS5400 is not included.
12.2(2)XB1	This feature was implemented on the Cisco AS5850.
12.2(11)T	This feature was integrated into this release.

Feature History for H.323v4 Gateway Zone Prefix Registration Enhancements

Release	Modification
12.2(15)T	This feature was introduced.
12.3(3)	The **ras rrq dynamic prefixes** and the **rrq dynamic-prefixes-accept** commands were modified to be disabled by default.
12.3(4)T	This feature was integrated into this release.
12.4(9)T	The **terminal-alias-pattern** command was introduced to send the gateway priority along with dynamic zone prefixes from the gateway.

Feature History for Signal ISDN B-Channel ID to Enable Application Control of Voice Gateway Trunks

Release	Modification
12.3(7)T	This feature was introduced.

Feature History for H.323 VoIP Call Preservation Enhancements for WAN Link Failures

Release	Modification
12.4(4)XC	This feature was introduced.
12.4(9)T	This feature was integrated into this release.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Note For more information about these and other related Cisco IOS voice features, see the following:

- "H.323 Overview" section on page 7

- For information about the full set of Cisco IOS voice features, see the entire Cisco IOS Voice Configuration Library—including library preface, glossary, and other documents—at http://www.cisco.com/en/US/docs/ios/12_3/vvf_c/cisco_ios_voice_configuration_library_glossary /vcl.htm

Contents

Note For complete descriptions of the commands used in this chapter, see the command references listed in the "Additional References" section on page 135.

Prerequisites for Configuring H.323 Gateways

- Perform the prerequisites that are listed in the "Prerequisites for Configuring an H.323 Network" section on page 7.
- Develop a network plan that details the requirements and characteristics of your VoIP network. For more information, see the documents in the "Additional References" section on page 19
- Ensure that the routers you intend to configure as H.323 gateways are running a Cisco IOS software image that contains gateway functionality.
- To use H.323 security and accounting features, do the following:
  - These features use the H.235 standard. Because the standard is broad, ensure that the gatekeeper provides H.235 functionality that specifically complements the gateway implementation described in this document.
  - The H.323 gateway sends accounting information using a nonstandard field in the ClearToken field. Ensure that the gatekeeper can retrieve this information from the ClearToken field.

Restrictions for Configuring H.323 Gateways

Restrictions are described in the Restrictions for Configuring an H.323 Network, page 8

Note The gatekeeper authenticates the endpoint based on the general ID. It does not relate the H.323 ID and general ID. Both the gateway H323_ID and the generalID in ClearTokens should be same.

How to Configure H.323 Gateways

This section contains the following information:

Configuring a Router Interface as a Gateway

To configure a Cisco device as an H.323 gateway in a service provider environment, configure at least one of its interfaces as a gateway interface. Use either an interface that is connected to the gatekeeper or a loopback interface for the gateway interface. The interface that is connected to the gatekeeper is usually a LAN interface: Fast Ethernet, Ethernet, FDDI, or Token Ring.

Configuring a Router Interface

To configure a gateway interface, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gateway**
2. **exit**
3. **ip cef**
4. **interface** *type number* [*nametag*]
5. **h323-gateway voip interface**
6. **h323-gateway voip id** *gatekeeper-id* {**ipaddr** *ip-address* [*port*] | **multicast**} [**priority** *priority*]
7. **h323-gateway voip h323-id** *interface-id*

8. **h323-gateway voip tech-prefix** *prefix*

9. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gateway`
	
	Example:
	`Router(config)# gateway`
		Enters gateway configuration mode and enables the gateway.

Step 2	`exit`
	
	Example:
	`Router(config-gateway)# exit`
		Exits the current mode.

Step 3	`ip cef`
	
	Example:
	`Router(config)# ip cef`
		(Optional) Enables Cisco Express Forwarding routing.

Step 4	`interface` *type number* `[`*nametag*`]`
	
	Example:
	`Router(config)# interface serial 0`
		Enters interface configuration mode for the interface that is connected to the gatekeeper. Keywords and arguments are as follow:
		
		• *type*—Type of interface to be configured.
		
		• *number*—Port, connector, or interface card number. The number is assigned at the factory at the time of installation or when added to a system and can be displayed with the **show interfaces** command.
		
		• *nametag*—Logic name to identify the server configuration so that multiple entries of server configuration can be entered.

Step 5	`h323-gateway voip interface`
	
	Example:
	`Router(config-if)# h323-gateway voip interface`
		Identifies this as a VoIP gateway interface.

	Command	Purpose
Step 6	**h323-gateway voip id** *gatekeeper-id* {**ipaddr** *ip-address* [*port*]| **multicast**} [**priority** *priority*] **Example:** Router(config-if)# h323-gateway voip id gk3.gg-dn1 ipaddr 172.18.0.0 1719	(Optional) Defines the name and location of the gatekeeper for this gateway. Keywords and arguments are as follows: • *gatekeeper-id*—H.323 identification of the gatekeeper. Must exactly match the gatekeeper ID in the gatekeeper configuration. Recommended format: name.domainname. • **ipaddr** *ip-address*—IP address to be used to identify the gatekeeper. • *port*—Port number used. • **multicast**—Gateway uses multicast to locate the gatekeeper. • **priority** *priority*—Priority of this gatekeeper. Range: 1 to 127. Default: 127.
Step 7	**h323-gateway voip h323-id** *interface-id* **Example:** Router(config-if)# h323-gateway voip h323-id name@domainname	(Optional) Defines the H.323 name of the gateway, identifying this gateway to its associated gatekeeper. Usually this ID is the name of the gateway, with the gatekeeper domain name appended: name@domainname.
Step 8	**h323-gateway voip tech-prefix** *prefix* **Example:** Router(config-if)# h323-gateway voip tech-prefix 1#	(Optional) Defines the numbers used as the technology prefix that the gateway registers with the gatekeeper. Can contain up to 11 characters. Although not strictly necessary, a pound symbol (#) is frequently used as the last digit in a prefix. Valid characters: 0 to 9, #, and *.
Step 9	**exit** **Example:** Router(config-if)# exit	Exits the current mode.

Verifying a Router Interface

To verify the router interface, perform the following step.

Step 1 **show gateway**

Use this command to verify gateway configuration by displaying the current registration information and gateway status.

Router# **show gateway**

Shutting Down and Enabling VoIP Services on a Gateway

This section contains the following procedures:

Shutting Down and Enabling VoIP Service

To shut down or enable all VoIP services on a Cisco gateway, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **voice service voip**
2. **no shutdown forced**
3. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**voice service voip** **Example:** `Router(config)# voice service voip`	Enters voice-service-VoIP configuration mode.
Step 2	**no shutdown forced** **Example:** `Router(conf-voi-serv)# shutdown forced`	Shuts down or enables VoIP call services.
Step 3	**exit** **Example:** `Router(conf-voi-serv)# exit`	Exits the current mode.

Shutting Down and Enabling VoIP Submodes

To shut down and enable VoIP submodes, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **voice service voip**
2. **h323**
3. **no call service stop maintain-registration**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`voice service voip`
	
	Example:
	`Router(config)# voice service voip`
		Enters voice-service-VoIP configuration mode.

Step 2	`h323`
	
	Example:
	`Router(conf-voi-serv)# h323`
		Selects H.323-call-processing submode.

Step 3	`no call service stop forced maintain-registration`
	
	Example:
	`Router(conf-voi-serv)# call service stop maintain-registration`
		Shuts down or enables VoIP call services for the selected submode.

Step 4	`exit`
	
	Example:
	`Router(conf-voi-serv)# exit`
		Exits the current mode.

Verifying Gateway Status

To verify gateway status, perform the following step.

Step 1   **show gateway**

Use this command to display gateway status.

The following example displays output after the gateway has been shut down:

```
Router# show gateway

H.323 ITU-T Version: 4.0   H323 Stack Version: 0.1
 H.323 service is shutdown
 Gateway  Router is not registered to any gatekeeper
```

The following example displays output after a graceful shutdown with calls in progress:

```
Router# show gateway

H.323 ITU-T Version: 4.0   H323 Stack Version: 0.1
 H.323 service is shutting down
 Gateway  Router is registered to Gatekeeper GK1
```

The following example displays output when H.323 call service has been shut down with the **call service stop maintain-registration** command:

```
Router# show gateway

H.323 ITU-T Version: 4.0   H323 Stack Version: 0.1
 H.323 service is shutdown
 Gateway  Router is registered to Gatekeeper GK1
```

Configuring Gateway RAS

This section contains the following information:

Registration, Admission, and Status (RAS) signaling performs registration, admissions, status, and disengage procedures between the H.323 VoIP gateway and the H.323 VoIP gatekeeper. RAS tells the gatekeeper to translate a E.164 phone number of the session target into an IP address.

In the RAS exchange between a gateway and a gatekeeper, a technology prefix is used to identify the specific gateway when the selected zone contains multiple gateways. The **tech-prefix** command is used to define technology prefixes.

In most cases there is a dynamic protocol exchange between the gateway and the gatekeeper that enables the gateway to inform the gatekeeper about technology prefixes and where to forward calls. If, for some reason, that dynamic registry feature is not in effect, statically configure the gatekeeper to query the gateway for this information.

Note To configure the gatekeeper to query for prefix and forwarding information, see "Configuring H.323 Gatekeepers and Proxies" section on page 139.

To configure RAS, define specific parameters for the applicable POTS and VoIP dial peers. The POTS dial peer informs the system of which voice port to direct incoming VoIP calls to and (optionally) determines that RAS-initiated calls have a technology prefix prepended to the destination telephone number. The VoIP dial peer determines how to direct calls that originate from a local voice port into the VoIP cloud to the session target. The session target indicates the address of the remote gateway where the call is terminated. There are several different ways to define the destination gateway address:

- By statically configuring the IP address of the gateway.
- By defining the Domain Name System (DNS) name of the gateway.
- By using RAS. If RAS is used, the gateway determines the destination target by querying the RAS gatekeeper.

Configuring Basic RAS

To configure basic RAS, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **dial-peer voice** *tag* **pots**
2. **destination-pattern** *string*[**T**]
3. **port** *controller***:D**
4. **exit**
5. **dial-peer voice** *tag* **voip**
6. **destination-pattern** *string*[**T**]
7. **tech-prefix** *number*

8. **session target ras**

9. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**dial-peer voice** *tag* **pots** **Example:** `Router(config)# dial-peer voice 456 pots`	Enters dial-peer configuration mode for the POTS dial peer designated by *tag*.

Step 2	**destination-pattern** *string*[**T**]
	
	Example:
	`Router(config-dial-peer)# destination-pattern 1513200....`
		Specifies the E.164 address associated with this dial peer. Keywords and arguments are as follows:
		• *string*—E.164 or private dialing plan telephone number. Valid entries: digits 0 to 9, letters A to D, and the following special characters:
		  – Asterisk (*) and pound sign (#)—Keys that appear on standard touchtone dial pads.
		  – Comma (,)—Pause between digits.
		  – Period (.)—Match to any entered digit (used as a wildcard).
		  – Percent sign (%)—The previous digit or pattern zero or multiple times, similar to wildcard usage in the regular expression.
		  – Circumflex (^)—Match to the beginning of the string.
		  – Dollar sign ($)—Match to the null string at the end of the input string.
		  – Backslash (\)—Is followed by a single character matching that character or used with a single character having no other significance (matching that character).
		  – Question mark (?)—The previous digit occurred zero or one time.
		  – Brackets ([ ])—Range of digits. Digits (0 to 9) are enclosed in brackets. Similar to a regular expression rule.
		  – Parentheses (( ))—A pattern. Same as the regular expression rule—for example, 408(555). Use parentheses in conjunction with symbols ? or %.
		  For more information on applying wildcard symbols to destination patterns and the dial strings that result, see Dial Peer Configuration on Voice Gateway Routers.
		• **T**—Control character indicating that the **destination-pattern** value is a variable-length dial string.

	Command	Purpose

Step 3	`port` *controller*`:D`
	
	Example:
	`Router(config-dial-peer)# port 0:D`
		(Cisco AS5300 only) Associates this POTS dial peer with a specific voice port. Keywords and arguments are platform dependent.

Step 4	`exit`
	
	Example:
	`Router(config-dial-peer)# exit`
		Exits the current mode.

Step 5	`dial-peer voice` *tag* `voip`
	
	Example:
	`Router(config)# dial-peer voice 123 voip`
		Enters dial-peer configuration mode for the VoIP peer designated by *tag*.

Step 6	`destination-pattern` *string*[`T`] **Example:** `Router(config-dial-peer)# destination-pattern 1513200....`	See Step 2 above.
Step 7	`tech-prefix` *number* **Example:** `Router (config-dial-peer)# tech-prefix 9#`	Defines the numbers used as the technology prefix that the gateway registers with the gatekeeper. Can contain up to 11 characters. Although not strictly necessary, a pound symbol (#) is frequently used as the last digit in a prefix. Valid characters: 0 to 9, #, and *.

Step 8	`session target ras`
	
	Example:
	`Router (config-dial-peer)# session target ras`
		Specifies that the RAS protocol is being used to determine the IP address of the session target—meaning that a gatekeeper translates the E.164 address to an IP address.

Step 9	`exit`
	
	Example:
	`Router (config-dial-peer)# exit`
		Exits the current mode.

Verifying RAS Configuration

To verify RAS configuration, perform the following step.

Step 1   **show dial-peer voice**

Use this command to verify the POTS and VoIP dial-peer configuration.

The following example shows output for a VoIP dial peer using RAS on a Cisco AS5300:

```
Router# show dial-peer voice 1234

VoiceOverIpPeer1234
tag = 1234, destination-pattern = 1234',
answer-address = ',
group = 1234, Admin state is up, Operation state is up,
incoming called-number = ', connections/maximum = 0/unlimited,
application associated:
type = voip, session-target = ras',
```

```
technology prefix: 8#
ip precedence = 0, UDP checksum = disabled,
session-protocol = cisco, req-qos = controlled-load,
acc-qos = best-effort,
fax-rate = voice, codec = g729r8,
Expect factor = 10, Icpif = 30,
VAD = enabled, Poor QOV Trap = disabled,
```

Troubleshooting Tips

- To display the types and addressing of RAS messages sent and received, use the **debug ras** command. The debug output lists the message type using mnemonics defined in ITU-T specification H.225.

- To display additional information about the actual contents of the H.225 RAS messages, use the **debug h225 asn1** command.

Configuring RAS Retries and Timers

You can configure RAS message timeout values, message retry counter values, and registration request (RRQ) message time-to-live and early transmit time margins on Cisco gateways. This provides greater flexibility in configuring gateways in different network environments.

The **ras timeout** command configures the number of seconds for the gateway to wait before resending a RAS message to a gatekeeper. The **ras retry** command configures the number of times to resend the RAS message after the timeout period expires. The default values for timeouts and retries are acceptable in most networks. You can use these commands if you are experiencing problems in RAS message transmission between gateways and gatekeepers. For example, if you have gatekeepers that are slow to respond to a type of RAS request, increasing the timeout value and the number of retries increases the call success rate, preventing lost billing information and unnecessary switchover to an alternate gatekeeper.

The **ras rrq ttl** command configures the number of seconds that the gateway should be considered active by the gatekeeper. The gateway transmits this value in the RRQ message to the gatekeeper. The **margin** *time* keyword and argument allow the gateway to transmit an early RRQ to the gatekeeper before the time-to-live value advertised to the gatekeeper.

Configuring RAS Timeout and Retry Counters

To configure RAS message timeout values and retry counters, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **voice service voip**

2. **h323**

3. **ras timeout** {**all** | **arq** | **brq** | **drq** | **grq** | **rai** | **rrq**} *value*

4. **ras retry** {**all** | **arq** | **brq** | **drq** | **grq** | **rai** | **rrq**} *value*

5. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`voice service voip`
	
	Example:
	`Router(config)# voice service voip`
		Enters voice-service configuration mode for VoIP.

Step 2	`h323`
	
	Example:
	`Router(conf-voi-serv)# h323`
		Enters voice-service-h323 configuration mode.

Step 3	`ras timeout {all	arq	brq	drq	grq	rai	rrq} value`  **Example:** `Router(conf-serv-h323)# ras timeout all 10`	Sets RAS timeout conditions. Keywords and argument are as follows: • **all**—All RAS message counters that do not have explicit values configured individually. If the **no ras timeout all** command is entered, all values are set to the default except the individual values that were configured separately. • **arq**—Admission request (ARQ) message counter. • **brq**—Bandwidth request (BRQ) message counter. • **drq**—Disengage request (DRQ) message counter. • **grq**—Gatekeeper request (GRQ) message counter. • **rai**—Resource availability indication (RAI) message counter. • **rrq**—Registration request (RRQ) message counter. • *value*—How long the gateway waits for a message from the gatekeeper before timing out, in seconds. Range: 1 to 45.
Step 4	`ras retry {all	arq	brq	drq	grq	rai	rrq} value`  **Example:** `Router(conf-serv-h323)# ras retry grq 5`	Sets RAS retry conditions. Keywords are as in step 3. The argument is as follows: • *value*—Number of times that the gateway resends messages to the gatekeeper after timeout. Range: 1 to 30.

Step 5	`exit`
	
	Example:
	`Router(conf-serv-h323)# exit`
		Exits the current mode.

Configuring RRQ Time-to-Live Value

To configure the RRQ time-to-live value, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **voice service voip**

2. **h323**

3. **ras rrq ttl** *time-to-live* [**margin** *time*]

4. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`voice service voip`
	
	Example:
	`Router(config)# voice service voip`
		Enters voice-service configuration mode for VoIP.

Step 2	`h323`
	
	Example:
	`Router(conf-voi-serv)# h323`
		Enters voice-service-h323 configuration mode.

Step 3	`ras rrq ttl` *time-to-live* [**margin** *time*] **Example:** `Router(conf-serv-h323)# ras rrq ttl 90 margin 30`	Sets time-to-live parameters. Argument and keyword are as follows: • *time-to-live*—How long, in seconds, the gatekeeper considers the gateway active. Range: 15 to 4000 (must be greater than the **margin** *time* value). • **margin** *time*—How long, in seconds, an RRQ message can be transmitted from the gateway before the time-to-live value advertised to the gatekeeper. Range: 1 to 60 (this value times two must be less than or equal to the *time-to-live* value).

Step 4	`exit`
	
	Example:
	`Router(conf-serv-h323)# exit`
		Exits the current mode.

Verifying RAS Retries and Timers

To verify RAS retries and timers, perform the following step.

Step 1 **show running config**

Use this command to verify RAS message retry counters, timeout values, and time-to-live values.

```
Router# show running-config

Current configuration : 925 bytes
!
version 12.3
.
.
.
voice service voip
 h323
  ras rrq ttl 90 margin 30
  ras timeout all 7
  ras timeout grq 10
```

```
    ras timeout drq 30
    ras retry all 10
    ras retry grq 5
.
.
.
```

Examples

The following example shows the GRQ message timeout value set to 10 seconds and all other RAS message timeout values set to 7 seconds:

```
Router(conf-serv-h323)# ras timeout grq 10
Router(conf-serv-h323)# ras timeout all 7
```

The following example shows the GRQ message counter set to 5 and all other RAS message counters set to 10:

```
Router(conf-serv-h323)# ras retry all 10
Router(conf-serv-h323)# ras retry grq 5
```

The following example shows the time-to-live value configured to 90 seconds and the **margin** *time* value configured to 30 seconds:

```
Router(conf-serv-h323)# ras rrq ttl 90 margin 30
```

Configuring Gateway-Resource-Availability Reporting

To allow gatekeepers to make intelligent call-routing decisions, the gateway reports the status of its resource availability to its gatekeeper. Resources that are monitored are digital-signal-level 0 (DS0) channels and digital-signal-processor (DSP) channels.

The gateway reports its resource status to the gatekeeper using the RAS Resource Availability Indication (RAI). When a monitored resource falls below a configurable threshold, the gateway sends a RAI to the gatekeeper indicating that the gateway is almost out of resources. When the available resources then cross over another configurable threshold, the gateway sends an RAI indicating that the resource depletion condition no longer exists.

You can configure resource-reporting thresholds by using the **resource threshold** command. Upper and lower thresholds are separately configurable to prevent the gateway from operating sporadically because of the availability or lack of resources.

Configuring E.164-Address Registration

If phones are connected directly to the gateway, the Cisco H.323 Version 2 gateway allows fully qualified E.164 numbers to be registered with the gatekeeper. When configuring the gateway, use the **register e164** command to register these E.164 numbers.

Configuring In-Band Tones and Announcements

In-band progress tones and announcements are required for PSTN services and for ISDN speech and 3.1-kHz voice services, per Bellcore and ANSI specifications. To guarantee that in-band tones and announcements are generated when required and at the appropriate switch, Cisco H.323 signaling

software ensures that the progress indicator (PI) is carried end to end in call-signaling messages between the called party and the calling party. The PI in outbound dial peers can also be configured at the H.323 VoIP gateway, if necessary.

The PI is an IE that signals when in-band tones and announcements are available. The PI controls whether the local switch generates the appropriate tone or announcement or whether the remote switch is responsible for the generation. For example, if the terminating switch generates the ringback tone, it sends a PI of 1 or 8 in the alerting message. If the originating switch receives an alerting message without a PI, it generates the ringback tone.

The specific PI that a switch sends in call messages, if any, depends on the model of the switch. To ensure that in-band communication is generated appropriately, it may be necessary in some instances to override the default behavior of the switch by manually configuring the PI at the Cisco H.323 gateway.

The PI is configurable in setup messages from the outbound VoIP dial peer, typically at the originating gateway, and in alert, progress, and connect messages from the outbound POTS dial peer, typically at the terminating gateway. The PI is configured by the **progress_ind** command. Table 1 shows the PI values that can be configured on the H.323 gateway.

Table 1   Configurable Progress Indicator Values for H.323 Gateways

PI	Description	Message Type
0	No progress indicator is included.	Setup
1	Call is not end-to-end ISDN; further call progress information may be available in-band.	Alert, setup, progress, connect
2	Destination address is non-ISDN.	Alert, progress, connect
3	Origination address is non-ISDN.	Setup
8	In-band information or appropriate pattern is now available.	Alert, progress, connect

When interworking is between ISDN and non-ISDN networks, the originating gateway reacts as follows:

- If the originating switch does not include a PI in setup messages, the originating gateway assumes that the originating switch is ISDN and expects the switch to generate the ringback tone. Determine which device generates the ringback tone by using the **progress_ind** command in dial-peer configuration mode:
  - To enable the terminating switch to generate the ringback tone, set the PI to 8 in the alert messages on the terminating gateway. The progress indicator is configured in the POTS dial peer.
  - To enable the originating gateway to generate the ringback tone, set the PI to 3 in setup messages on the originating gateway. The PI is configured in the VoIP dial peer.

Note   If the terminating gateway sends an alert message with no PI value, the originating gateway generates the ringback tone. But if the terminating gateway sends an alert message that has a PI of 1, 2, or 8, the originating gateway does not generate ringback tone.

- The originating gateway cuts through the voice path in the backward direction when it receives a progress or alert message that has a PI of 1, 2, or 8.

Pure ISDN calls may use different protocols at the originating and terminating ends. For example, a call may originate on ETSI and terminate on NI2. If the two protocols are not compatible end to end, the gateway drops all IEs from messages, including the progress indicator. Because a progress indicator is required in all progress messages, the originating gateway inserts a PI of 1 in the progress message. To avoid dropping IEs, use the **isdn gateway-max-internetworking** command to prevent the gateway from checking protocol compatibility.

Configuring Gateway AAA

For the gateway to provide authentication and accounting services, enable and configure your gateway to support authentication, authorization, and accounting (AAA) services. AAA enables the gateway to interact with a RADIUS security server to authenticate users (typically incoming calls) and to perform accounting services.

Note

- For information about AAA configuration on a gateway, see *Configuring AAA for Cisco Voice Gateways* at
  http://www.cisco.com/en/US/docs/ios/voice/aaa/configuration/guide/15_0/va_15_0_book.html

- For information about RADIUS and AAA security services, see the *Cisco IOS Security Configuration Guide* at
  http://www.cisco.com/en/US/docs/ios/sec_user_services/configuration/guide/15_0/sec_user_servi
  ces_15_0_book.html.

Configuring H.235 Gateway Security

This section contains the following information:

Information About H.235 Gateway Security

The Cisco H.235-based security and accounting features described in this section can be used by a gatekeeper, which is considered a known and trusted entity, to authenticate, authorize, and route H.323 calls.

The Cisco H.323 gateway supports the use of CryptoH323Tokens for authentication. The CryptoH323Token is defined in the ITU-T H.225 Version 2 standard and is used in a "password-with-hashing" security scheme as described in section 10.3.3 of the H.235 specification.

A cryptoToken can be included in any RAS message to authenticate the sender of the message. A separate database can be used for user ID and password verification.

Cisco H.323 gateways support three levels of authentication:

- Endpoint—The RAS channel used for gateway-to-gateway signaling is not a secure channel. To ensure secure communication, H.235 allows gateways to include an authentication key in their RAS messages. This key is used by the gatekeeper to authenticate the source of the messages. At the endpoint level, validation is performed on all messages from the gateway. The cryptoTokens are validated using the password configured for the gateway.

> **Note** To secure the RAS messages and calls, it is essential that the gatekeeper provides authentication based on the secure key. The gatekeeper must support H.235 security using the same security scheme as the Cisco gateway.

- Per-Call—When the gateway receives a call over the telephony leg, it prompts the user for an account number and PIN. These two numbers are included in certain RAS messages sent from the endpoint to authenticate the originator of the call.

- All—This option is a combination of the other two. With this option, the validation of cryptoTokens in ARQ messages is based on an the account number and PIN of the user making a call. The validation of cryptoTokens sent in all the other RAS messages is based on the password configured for the gateway.

CryptoTokens for RRQs, unregistration requests (URQs), DRQs, and the terminating side of ARQs contain information about the gateway that generated the token. The cryptoTokens include the gateway identification (ID)—which is the H.323 ID configured on the gateway—and the gateway password. The cryptoTokens for the originating-side ARQ messages contain information about the user that is placing the call, including the user ID and PIN.

Although the scenarios in this document describe how to use the security and accounting features in a prepaid call environment, these features may also be used to authorize IP calls that originate in another domain (interservice provider or intercompany calls).

H.235-based security and accounting features can be used with AAA. The gateway can be configured to use the gatekeeper for call authentication or authorization, and AAA can be used for call accounting.

In addition, H.235-based security and accounting features include support for the following:

- Settlement with the gatekeeper, which allows the gateway to obtain, track, and return accounting information

- Call metering, which allows the gateway to terminate a call if it exceeds the allotted time (in the case of prepaid calls)

> **Note** The H.235 security and accounting features described in this document are separate from, and should not be confused with, the standard interactive-voice-response (IVR) and AAA features used to authenticate inbound calls or with the settlement functions provided by the Open Settlement Protocol (OSP).

Settlement with the Gatekeeper

The H.235 security and accounting features are designed to support a variety of situations in which some form of authentication or tracking is required. The security features control access through a userID-password database. The accounting enhancements allow call usage to be tracked at the origin and at the destination.

Fields in the RAS messages allow the gateway to report call-usage information to the gatekeeper. The call-usage information is included in the DRQ message that is sent when the call is terminated.

Call Tracking

With prepaid calling services, an account number and PIN must be entered and the duration of the call must be tracked against the remaining credit of the customer. The Cisco H.323 gateway monitors prepaid account balances and terminates a call if the account is exceeded.

> **Note** Because authentication information includes a time stamp, it is important that all Cisco H.323 gateways and gatekeepers (or other entities that perform authentication) be synchronized. Cisco H.323 gateways must be synchronized using the Network Time Protocol (NTP).

Figure 1 illustrates the flow of a possible call for which H.323 security and accounting features are used.

Figure 1   Flow for a Call That Requires H.323 Security and Accounting Features
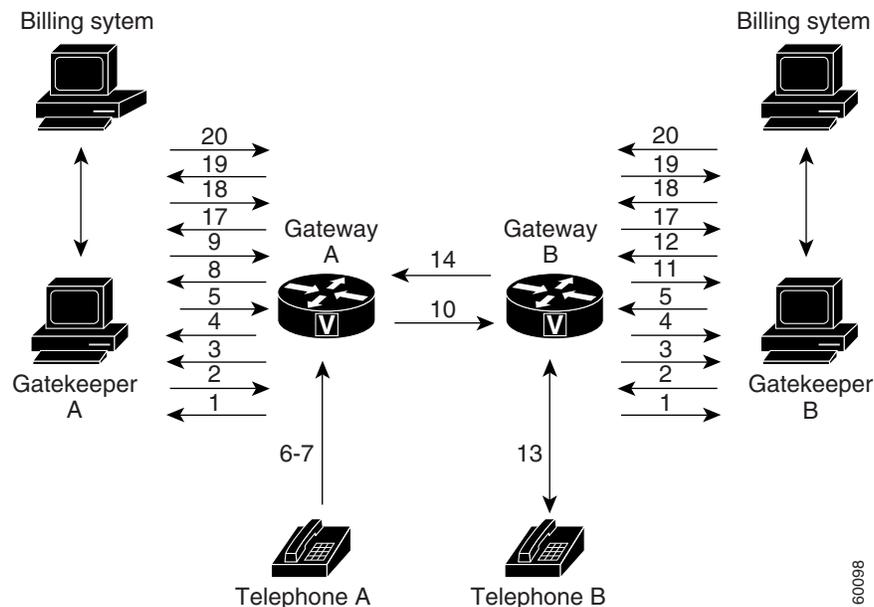
In this example, Telephone A is attempting to establish a phone call to Telephone B. The following numbered explanations correspond to the action taking place at each numbered reference in Figure 1.

Gateways Establish Secure Communication with the Gatekeepers

1. Gateways A and B send GRQ messages to their respective gatekeepers. The GRQ message includes the authentication capability and the algorithm object ID.

2. Gatekeepers A and B respond to their respective gateways with gatekeeper confirmation (GCF) messages. The GCF message includes the authentication capability and the algorithm object ID.

3. If the values for the H.323 security parameters do not match what is expected, the gatekeeper responds with a gatekeeper rejection (GRJ) message that contains a reject reason of securityDenial. This prompts the gateway to resend the GRQ.

4. Gateways A and B send RRQ messages to their respective gatekeepers. The RRQ message includes authentication information in the cryptoToken field.

5. Gatekeepers A and B respond to their respective gateways with registration confirmation (RCF) messages.

If an authentication failure occurs, the gatekeeper responds with a registration rejection (RRJ) message.

Secure Telephone Communications Initiated

6. Telephone A establishes a connection with Gateway A.

7. Gateway A initiates the IVR script to obtain the account number and PIN of the user and the desired destination telephone number.

8. Gateway A sends an ARQ message to Gatekeeper A. The gateway must include additional information in the ARQ message to enable the gatekeeper to authenticate the call. The information included in the ARQ message varies depending on whether the ARQ message is being sent by the source or the destination gateway. At this point in the scenario, it is the source gateway that is requesting admission. Therefore, the ARQ message includes the account number and PIN of the user. This information is encrypted using MD5 hashing and is included in the cryptoTokens field.

9. Gatekeeper A validates the authentication information, resolves the destination telephone number, and determines the appropriate destination gateway (which is Gateway B in this case). Then Gatekeeper A sends an admission confirmation (ACF) message to Gateway A. The ACF message includes the billing information of the user (such as a reference ID and current account balance for prepaid call services) and an access token.

10. Gateway A sends a setup message to Gateway B. The setup message also includes the access token.

11. Gateway B sends an ARQ message to Gatekeeper B. The ARQ message includes the access token received from Gateway A.

12. Gatekeeper B validates the authentication information in the access token and responds to Gateway B with an ACF message.

    If the authentication information is in error, Gatekeeper B sends an admission rejection (ARJ) message to Gateway B with a reject reason of securityDenial.

13. Gateway B initiates a call to the destination telephone.

14. When the destination telephone is answered, Gateway B sends a connect message to Gateway A.

15. Gateways A and B start their timers to meter the call. If the caller is using prepaid call services, the meter is constantly compared to the account balance of the user, which was included in the ACF message sent in Step 9.

Telephone Communications Terminated

16. The call is terminated when one of the parties hangs up or, in the case of prepaid call services, when either of the gateways determines that the account balance of the user has been exceeded.

17. Gateways A and B send DRQ messages to the their respective gatekeepers. The DRQ message contains the resulting billing information.

18. Gatekeepers A and B send disengage confirmation (DCF) messages to their respective gateways.

Communication Between the Gateways and the Gatekeepers Terminated

19. Gateways A and B send URQ messages to their respective gatekeepers.

20. Gatekeepers A and B send unregistration confirmation (UCF) messages to their respective gateways.

Downloading IVR Scripts

Tool Command Language (TCL) IVR scripts are the default scripts for all Cisco voice features that use IVR.

The H.323 security and accounting enhancements described in this document require the use of one of the following IVR scripts:

- voip_auth_acct_pin_dest.tcl
- voip_auth_acct_pin_dest_2.tcl

Note For more information on TCL IVR applications, see the *Cisco IOS TCL and VoiceXML Application Guide* at http://www.cisco.com/en/US/docs/ios/voice/ivr/configuration/guide/tcl_c.html.

voip_auth_acct_pin_dest.tcl Script

The voip_auth_acct_pin_dest.tcl script does the following:

- Prompts the caller to enter an account number, PIN, and destination number. This information is provided to an H.323 gatekeeper, which authenticates and authorizes the call.

  If the caller is using a debit card account number, the following occurs:

  - The gatekeeper returns the remaining credit time amount.
  - The TCL script monitors the time remaining and, based on a configured value, plays a "time running out" message to the caller. The message (such as, "You have only 3 minutes remaining on your credit.") is played only to the calling party. The called party hears silence during this time. For example, if the configured timeout value is 3 minutes, the message is played when the caller has only 3 minutes of credit left.
  - The TCL script plays a warning message when the credit of the user has been exhausted. The message (such as, "Sorry, you have run out of credit.") is played only to the calling party. The called party hears silence during this time.

- Allows the caller to make subsequent calls to different destinations without disconnecting from the call leg. Thus, the caller is required to enter the account ID and PIN only once (during initial authorization). For making subsequent calls, the caller needs to enter only the destination number. After completing a call to one destination, the caller can disconnect the call by pressing the pound (#) key on the keypad and holding it down from 1 to 2 seconds. If the # key is pressed down for more than 1 second, it is treated as a long pound (#). The called party is disconnected, and the caller is prompted to enter a new destination number. Once a new destination number is entered, the call is authenticated and authorized using this number and the previously provided account number and PIN.

  This feature also allows the caller to continue making additional calls if the called party hangs up.

- Reauthenticates and authorizes each new call. Each time a caller enters a new destination number, the TCL script reauthenticates or authorizes the call with the gatekeeper and, if the caller is using a debit card account, obtains the remaining credit time information.

- Allows the caller to enter the necessary information without having to hear all or any of the prompts. The TCL script stops playing (or does not begin playing) the prompt if it detects that the caller wants to enter the information without listening to the prompt.

> **Note** The normal terminating character for the account number, PIN, and destination number is the pound (#) key.

- Allows the caller to interrupt announcements by pressing the touchtone key. This TCL script stops playing announcements when the system detects that the caller has pressed any touchtone key.

- Allows the caller to interrupt partially entered numbers and restart from the beginning by pressing a designated key on the keypad. The asterisk (*) key is configured as the interrupt key in the TCL script. The caller can use the asterisk key to cancel an entry and then reenter the account number, PIN, or destination number. The caller is allowed to re-enter a field only a certain number of times. The number of retries may be configured. The default is three times.

- Can terminate a field by size instead of the terminating character (#). The TCL script allows a specified number of digits to be entered in the account number and PIN fields. This means that the caller can type all the digits (without the terminating character) and the script determines how to extract different fields from the number strings. If the caller uses the terminating character, the terminating character takes precedence and the fields are extracted accordingly.

- Supports two languages. The IVR script supports two languages, which must be similar in syntax. The languages must be similar in the manner in which numbers are constructed—especially for currency, amount, and time. All the prompts are recorded and stored in both languages. The language selection is made when the caller presses a predefined key in response to a prompt (such as, "For English, press 1. For Spanish, press 2."). The TCL script uses the selected language until the caller disconnects.

voip_auth_acct_pin_dest_2.tcl Script

The voip_auth_acct_pin_dest_2.tcl script is a simplified version of the voip_auth_acct_pin_dest.tcl script. It prompts the caller for an account number followed by a PIN. The caller is then prompted for a destination number. This information is provided to the H.323 gatekeeper that authenticates and authorizes the call. This script provides prompts only in English.

If the caller is using a debit account number, it plays a "time running out" message when the caller has 10 seconds of credit time remaining. It also plays a "time has expired" message when the credit of the caller has been exhausted.

Configuring H.235 Gateway Security

To use the H.235 security features for routing H.323 calls as illustrated above, do the following:

- Enable H.323 security on the gateway.
- Download the appropriate TCL IVR scripts from the Cisco Connection Online Software Support Center. The URL to this site is as follows:

  http://www.cisco.com/cgi-bin/tablebuild.pl/tclware

- Configure the IVR inbound dial peer on the gateway router.

To enable security on the gateway, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gateway**

2. **security password** *password* **level** {**endpoint** | **per-call** | **all**}

3. **exit**

4. **dial-peer voice** *tag* **pots**

5. **call application voice** *application-name location word*

6. **destination-pattern** *string*[**T**]

7. **port** *controller-number***:D**

8. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gateway`
	
	Example:
	`Router(config)# gateway`
		Enters gateway configuration mode.

Step 2	`security password` *password* `level` {`endpoint` | `per-call` | `all`} **Example:** `Router(config-gateway)# security password password level all`	Enables security and specifies the level of validation to be performed. • *password*—Gateway password. • **endpoint**—Validation is performed on all RAS messages sent by the gateway using the cryptoTokens that are generated based on the security password configured for the gateway. • **per-call**—Validation is performed only on the admission messages from the H.323 endpoints to the gateway ARQ messages). The gateway prompts the user for an account number and PIN. These two numbers are sent from the endpoint and are used to authenticate the originator of the call. • **all**—Combination of the **endpoint** and **per-call** options. Specifies that validation be performed on all RAS messages sent by the gateway. The validation of cryptoTokens in ARQ messages is based on the account number and PIN of the user making the call, and the validation of cryptoTokens sent in all other RAS messages is based on the password configured for the gateway.

Step 3	`exit`
	
	Example:
	`Router(config-gateway)# exit`
		Exits the current mode.

Step 4	`dial-peer voice` *tag* `pots`
	
	Example:
	`Router(config)# dial-peer voice 1 pots`
		Enters dial-peer configuration mode for the POTS dial peer designated by the *tag* value.

	Command	Purpose
Step 5	**call application voice** *application-name* *location word* **Example:** `Router(config-dial-peer)# call application voice xyz tftp://172.18.16.2/samp/xyz.tcl`	Initiates the IVR application and the selected TCL application name. • *application-name*—Character string that defines the name of the application. • *location*—Location of the TCL file in URL format. Valid values: TFTP, FTP, or flash. • *word*—Text string that defines an attribute-value (AV) pair specified by the TCL script and understood by the RADIUS server.
Step 6	**destination-pattern** *string*[**T**] **Example:** `Router(config-dial-peer)# destination-pattern 1513200....`	Specifies the E.164 address associated with this dial peer. For an explanation of the keywords and arguments, see the "Configuring Gateway RAS" section on page 51, Step 2.
Step 7	**port** *controller-number*:**D** **Example:** `Router(config-dial-peer)# port 0:D`	(Cisco AS5300 only) Configures the voice port associated with this dial peer. Keywords and arguments are as follows: • *controller-number*—The T1 or E1 controller. • **:D**—D channel associated with the ISDN PRI. **Note** Command syntax varies by platform.
Step 8	**exit** **Example:** `Router(config-dial-peer)# exit`	Exits the current mode.

Verifying H.235 Gateway Security

To verify H.235 gateway security, perform the following step.

Step 1 **show running-config**

Use this command to display the security password and level when it is enabled. By default, security is disabled.

```
Router# show running-config

security password 151E0A0E level all
```

Configuring Alternate-Gatekeeper Support

This section contains the following information:

Restrictions for Alternate-Gatekeeper Support

- You can use this feature only with a gatekeeper that supports the alternate gatekeeper functionality.

- The timer/retry number of RAS messages remains internal to the gateway as currently implemented. This feature does not include commands to allow tuning of these parameters.

- The alternate gatekeeper list is volatile—when the gateway loses power or is reset or reloaded, the alternate gatekeeper list that has been acquired from the gatekeeper is lost.

Information About Alternate-Gatekeeper Support

A gatekeeper manages H.323 endpoints in a consistent manner, allowing them to register with the gatekeeper and to locate another gatekeeper. The gatekeeper provides logic variables for proxies or gateways in a call path to provide connectivity with the Public Switched Telephone Network (PSTN), to improve quality of service (QoS), and to enforce security policies. Multiple gatekeepers may be configured to communicate with one another, either by integrating their addressing into the DNS or by using Cisco IOS configuration options.

An alternate gatekeeper provides redundancy for a gateway in a system in which gatekeepers are used. Redundant H.323 zone support in the gateway allows a user to configure two gatekeepers in the gateway (one as the primary and the other as the alternate). All gatekeepers are active. Each alternate gatekeeper, or gatekeeper node, shares its local zone information so that the cluster can effectively manage all local zones within the cluster. Each alternate gatekeeper has a unique local zone. Clusters provide a mechanism for distributing call processing seamlessly across a converged IP network infrastructure to support IP telephony, facilitate redundancy, and provide feature transparency and scalability.

An endpoint that detects the failure of its gatekeeper can safely recover from that failure by utilizing an alternate gatekeeper for future requests, including requests for existing calls. A gateway can only be registered to a single gatekeeper at a time. Only one gatekeeper is allowed to manage a single zone. The cluster manages up to five similarly configured zones and shares resources between the alternate gatekeepers in the cluster for each zone. You can define up to 100 zones in a single gatekeeper.

With gatekeeper clustering there is the potential that bandwidth may be overcommitted in a cluster. For example, suppose that there are five gatekeepers in a cluster and that they share 10 Mbps of bandwidth. Suppose that the endpoints registered to those alternates start placing calls quickly. It is possible that within a few seconds, each gatekeeper could be allocating 3 Mbps of bandwidth if the endpoints on each of the gatekeepers request that much bandwidth. The net result is that the bandwidth consumed in the cluster is 15 Mbps.

The alternate gatekeeper was purposely designed to restrict bandwidth because there is no clear way to sync bandwidth information quickly and efficiently. To work around this problem, "announcement" messages were restricted to intervals as small as 10 seconds. If the gatekeepers get into a situation in which endpoints request bandwidth rapidly, the problem is discovered and corrective action takes place within 10 seconds. Assuming that the gatekeepers are not synchronized on their timers, the announcement messages from the various gatekeepers are likely to be heard more quickly. Therefore, the problem is less severe. The potential exists, however, for overcommitment of the bandwidth between announcement messages if the call volume increases substantially in a short amount of time (as small as 10 seconds).

Note If you monitor your bandwidth, it is recommended that you consider lowering the maximum bandwidth so that if "spikes" such as those described above do occur, some bandwidth is still available.

Configuring Alternate-Gatekeeper Support

To configure alternate gatekeeper support on a gateway, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **interface Ethernet 0/1**

2. **h323-gateway voip interface**

3. **h323-gateway voip id** *gatekeeper-id* {**ipaddr** *ip-address* [*port*]| **multicast**} [**priority** *priority*]

4. **h323-gateway voip id** *gatekeeper-id* {**ipaddr** *ip-address* [*port*] | **multicast**} [**priority** *priority*]

5. **h323-gateway voip h323-id** *interface-id*

6. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**interface Ethernet 0/1** **Example:** Router(config)# interface Ethernet 0/1	Enters interface configuration mode for the selected Ethernet interface.
Step 2	**h323-gateway voip interface** **Example:** Router(config-if)# h323-gateway voip interface	Identifies this as a VoIP gateway interface.
Step 3	**h323-gateway voip id** *gatekeeper-id* {**ipaddr** *ip-address* [*port*]| **multicast**} [**priority** *priority*] **Example:** Router(config-if)# h323-gateway voip id gk3.gg-dn1 ipaddr 172.18.0.0 1719	Identifies the gatekeeper for this gateway interface and sets its attributes. For an explanation of the keywords and arguments, see the "How to Configure H.323 Gateways" section on page 45, step 6.
Step 4	**h323-gateway voip id** *gatekeeper-id* {**ipaddr** *ip-address* [*port*] | **multicast**} [**priority** *priority*] **Example:** Router(config-if)# h323-gateway voip id gk3.gg-dn1 ipaddr 172.18.0.0 1721	Identifies the alternate gatekeeper and sets its attributes.

	Command	Purpose
Step 5	**h323-gateway voip h323-id** *interface-id* **Example:** Router(config-if)$ h323-gateway voip id gk4.gg-dn1 ipaddr 209.165.202.132 1719	Defines the H.323 name of the gateway, identifying this gateway to its associated gatekeeper. Usually this ID is the name of the gateway, with the gatekeeper domain name appended to the end: name@domainname.
Step 6	**exit** **Example:** Router(config-if)# exit	Exits the current mode.

Verifying Configuration of Alternate-Gatekeeper Support

To verify configuration of alternate-gatekeeper support, perform the following step.

Step 1   **show gateway**

Use this command to verify that an alternate gatekeeper is configured.

```
Router# show gateway

Permanent Alternate Gatekeeper List
priority 127 id bmx1 ipaddr 10.77.241.103 1719 register needed
priority 127 id bmx2 ipaddr 10.77.241.117 1719 register needed
Primary gatekeeper ID bmx1 ipaddr 10.77.241.103 1719
```

Configuring DTMF Relay

This section contains the following information:

Information About DTMF Relay

Dual-tone multifrequency (DTMF) is the tone generated on a touchtone phone when the keypad digits are pressed. During a call, DTMF may be entered to access interactive voice response (IVR) systems, such as voice mail and automated banking services.

Although DTMF is usually transported accurately when using high-bit-rate voice codecs such as G.711, low-bit-rate codecs such as G.729 and G.723.1 are highly optimized for voice patterns and tend to distort DTMF tones. As a result, IVR systems may not correctly recognize the tones.

DTMF relay solves the problem of DTMF distortion by transporting DTMF tones "out of band," or separate from the encoded voice stream.

Relay Types

Cisco gateways currently support the following methods of DTMF relay:

- Cisco-proprietary Real-Time Transport Protocol (RTP)—DTMF tones are sent in the same RTP channel as voice data. However, the DTMF tones are encoded differently from the voice samples and are identified by a different RTP payload type code. Use of this method accurately transports DTMF tones, but because it is proprietary, it requires the use of Cisco gateways at both the originating and terminating endpoints of the H.323 call.

- H.245 signal or alphanumeric—These methods separate DTMF digits from the voice stream and send them through the H.245 signaling channel instead of through the RTP channel. The tones are transported in H.245 User Input Indication messages. The H.245 signaling channel is a reliable channel, so the packets that transport the DTMF tones are guaranteed to be delivered. However, because of the overhead of using a reliable protocol, and depending on network congestion conditions, the DTMF tones may be slightly delayed. All H.323 version 2 compliant systems are required to support the "h245-alphanumeric" method, while support of the "h245-signal" method is optional.

- Named Telephone Events (NTEs). Using NTE to relay DTMF tones provides a standardized means of transporting DTMF tones in RTP packets according to section 3 of RFC 2833, RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals, developed by the Internet Engineering Task Force (IETF) Audio/Video Transport (AVT) working group. RFC 2833 defines formats of NTE RTP packets used to transport DTMF digits, hookflash, and other telephony events between two peer endpoints. With the NTE method, the endpoints perform per-call negotiation of the DTMF relay method. They also negotiate to determine the payload type value for the NTE RTP packets. User preference for DTMF relay types is not supported, and DTMF relay forking is not supported.

The ability of a gateway to receive DTMF digits in a particular format and the ability to send digits in that format are independent functions. No configuration is necessary to receive DTMF digits from another H.323 endpoint using any of the methods described. The Cisco gateway is capable of receiving DTMF tones transported by any of these methods at all times.

Capabilities and Priorities

Cisco H.323 gateways advertise capabilities using H.245 capabilities messages. By default, they advertise that they can receive all DTMF relay modes. If the capabilities of the remote gateway do not match, the Cisco H.323 gateway transmits DTMF tones as in-band voice.

Configuring DTMF relay on the Cisco H.323 gateway sets preferences for how the gateway handles DTMF transmission. You can enable more than one DTMF relay option for a particular dial peer. If more than one option is enabled and if the peer indicates that it is capable of receiving DTMF in more than one of these formats, the gateway sends DTMF using the method among the supported formats that it considers to be the most preferred. If the remote device supports multiple formats, the gateway chooses the format according to the following priority:

1. cisco-rtp (highest priority)
2. h245-signal
3. h245-alphanumeric
4. rtp-nte
5. None—DTMF sent in-band

Payload Types

In addition, Cisco gateways provide support for asymmetrical payload types. Payload types can differ between local and remote endpoints. Therefore, the Cisco gateway can transmit one payload type value and receive a different payload type value.

The **dtmf-relay h245-signal** command relays a more accurate representation of a DTMF digit than does the **dtmf-relay h245-alphanumeric** command because tone duration information is included along with the digit value. This information is important for applications requiring that a key be pressed for a particular length of time. For example, one popular calling card feature allows the caller to terminate an existing call by pressing the # key for more than 2 seconds and then making a second call without having to hang up in between. This feature is beneficial because the access number and personal identification number (PIN) code do not need to be dialed again. Outside-line access charges, which are common at hotels, may also be avoided.

The **dtmf-relay h245-alphanumeric** command simply relays DTMF tones as ASCII characters. For instance, the DTMF digit 1 is transported as the ASCII character 1. There is no duration information associated with tones in this mode. When the Cisco H.323 gateway receives a DTMF tone using this method, the gateway generates the tone on the PSTN interface of the call using a fixed duration of 500 ms. All systems that are H.323 Version 2-compliant are required to support the **dtmf-relay h245-alphanumeric** command, but support of the **dtmf-relay h245-signal** command is optional.

H.245 Tunneling of DTMF Relay in Conjunction with Fast Connect

Through H.245 tunneling, H.245 messages are encapsulated within H.225 messages without using a separate H.245 TCP connection. When tunneling is enabled, one or more H.245 messages can be encapsulated in any H.225 message. H.245 tunneling is not supported as a stand-alone feature; initiation of H.245 tunneling procedures can be initiated only by using the **dtmf-relay** command and only from an active fast connect call. Furthermore, if **dtmf-relay** is configured on a Version 2 VoIP dial peer and the active call has been established by using fast connect, tunneling procedures initiated by the opposite endpoint are accepted and supported.

H.245 tunneling is backward compatible with H.323 Version 1 configurations.

Configuring DTMF Relay

To configure DTMF relay on a gateway, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **dial-peer voice tag voip**

2. **dtmf-relay [cisco-rtp] [h245-alphanumeric] [h245-signal] [rtp-nte]**

3. **rtp payload-type nte** *number*

4. **codec** {**clear-channel** | **g711alaw** | **g711ulaw** | **g723ar53** | **g723ar63** | **g723r53** | **g723r63** | **g726r16** | **g726r24** | **g726r32** | **g726r53** | **g726r63** | **g728** | **g729abr8** | **g729ar8** | **g729br8** | **g729r8** | **gsmefr** | **gsmfr**} [**bytes** *payload_size*]

5. **destination-pattern** *string*[**T**]

6. **session target** {**ipv4:***destination-address* | **dns:[s. | d. | e. | u.]** *hostname* | **loopback:rtp** | **loopback:compressed** | **loopback:uncompressed**}

   or

> > session target {**ipv4:***destination-address* | **dns:**[**s.** | **d.** | **e.** | **u.**] *hostname* | **loopback:rtp** | **loopback:compressed** | **loopback:uncompressed** | **mailto:**{*name* | **d.**}**@***domainname*}

> 7. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`dial-peer voice` *tag* `voip`
	
	Example:
	`Router(config)# dial-peer voice tag voip`
		Enters dial-peer configuration mode for the VoIP dial peer designated by *tag*.

Step 2	`dtmf-relay [cisco-rtp] [h245-alphanumeric] [h245-signal] [rtp-nte]`
	
	Example:
	`Router(config-dial-peer)# dtmf-relay cisco-rtp h245-alphanumeric h245-signal rtp-nte`
		Forwards DTMF tones. Keywords are as follows:
		
		• **cisco-rtp**—Forwards DTMF tones by using RTP with a Cisco-proprietary payload type.
		
		• **h245-alphanumeric**—Forwards DTMF tones by using the H.245 "alphanumeric" User Input Indication (UII) method. Range: tones 0 to 9, *, #, and A to D. Use this keyword to configure DTMF relay.
		
		• **h245-signal**—Forwards DTMF tones by using the H.245 "signal" UII method. Range: tones 0 to 9, *, #, and A to D.
		
		• **rtp-nte**—Forwards DTMF tones by using Real-Time Transport Protocol (RTP) with the Named Telephone Event (NTE) payload type.

Step 3	`rtp payload-type nte` *number*
	
	Example:
	`Router(config-dial-peer)# rtp payload-type nte 100`
		Identifies the payload type of a Real-Time Transport Protocol (RTP) packet. Keyword and argument are as follows:
		
		• **nte** *number*—Payload type is a Named Telephone Event (NTE). Range: 96 to 127. Default: 101.
		
		Do not use the following numbers, because they have preassigned values: 96, 97, 100, 121 to 123, and 125 to 127.
		
		Use of these values causes the command to fail. You must first reassign the value in use to a different unassigned number, for example:
		
		`rtp payload-type nse 105`
		`rtp payload-type nte 100`

Step 4	`codec {clear-channel` | `g711alaw` | `g711ulaw` | `g723ar53` | `g723ar63` | `g723r53` | `g723r63` | `g726r16` | `g726r24` | `g726r32` | `g726r53` | `g726r63` | `g728` | `g729abr8` | `g729ar8` | `g729br8` | `g729r8` | `gsmefr` | `gsmfr} [bytes` *payload_size*`]`
	
	Example:
	`Router(config-dial-peer)# codec g711alaw`
		Specifies the voice coder rate of speech for a dial peer.

	Command	Purpose
Step 5	`destination-pattern` *string*[`T`] **Example:** `Router(config-dial-peer)# destination-pattern 1513200....`	Specifies the prefix, the full E.164 telephone number, or an ISDN directory number to be used for a dial peer (depending on the dial plan). For an explanation of the keywords and arguments, see the "Configuring Gateway RAS" section on page 51, Step 2.
Step 6	**Cisco 2600 Series and Cisco 3600 Series** `session target` {`ipv4:`*destination-address* | `dns:[$s$.` | `$d$.` | `$e$.` | `$u$.]` *hostname* | `loopback:rtp` | `loopback:compressed` | `loopback:uncompressed`} **Cisco AS5300** `session target` {`ipv4:`*destination-address* | `dns:[$s$.` | `$d$.` | `$e$.` | `$u$.]` *hostname* | `loopback:rtp` | `loopback:compressed` | `loopback:uncompressed` | `mailto:`{*name* | `$d$.`}@*domainname*} **Example:** `Router(config-dial-peer)# session target ipv4:192.168.0.0`	Specifies a network-specific address for a specified dial peer or destination gatekeeper.

Step 7	`exit`
	
	Example:
	`Router(config-dial-peer)# exit`
		Exits the current mode.

Monitoring and Maintaining DTMF Relay

To monitor and maintain H.323 DTMF relay using NTE, use the following commands.

Step 1 **debug voip rtp session named-event**

Use this command to turn on debugging for RTP NTEs.

Step 2 **show voip rtp connections**

Use this command to display local and remote calling ID and IP address and port information.

Configuring FXS Hookflash Relay

A hookflash indication is a brief on-hook condition that occurs during a call. It is not long enough in duration to be interpreted as a signal to disconnect the call. Create a hookflash indication by quickly depressing and then releasing the hook on your telephone.

PBXs and telephone switches are frequently programmed to intercept hookflash indications and use them as a way to allow a user to invoke supplemental services. For example, your local service provider may allow you to enter a hookflash as a means of switching between calls if you subscribe to a call waiting service.

In the traditional telephone network, a hookflash results in a voltage change on the telephone line. Because there is no equivalent of this voltage change in an IP network, the ITU H.245 standard defines a message representing a hookflash. To send a hookflash indication using this message, an H.323 endpoint sends an H.245 user input indication message containing a "signal" structure with a value of "!". This value represents a hookflash indication.

Cisco H.323 Version 2 software includes limited support for relaying hookflash indications using the H.245 protocol. H.245 user input indication messages containing hookflash indications that are received on the IP call leg are forwarded to the plain old telephone service (POTS) call leg if the POTS interface is Foreign Exchange Office (FXO). If the interface is not FXO, any H.245 hookflash indication that is received is ignored. This support allows IP telephony applications to send hookflash indications to a PBX through the Cisco gateway and thereby invoke the IOS supplementary services of the PBX if the PBX supports access to those features using hookflash.

The gateway does not originate H.245 hookflash indications in this release. For example, it does not forward hookflash indications from foreign-exchange-station (FXS) interfaces to the IP network over H.245.

The acceptable duration of a hookflash indication varies by equipment vendor and by country. Although one PBX may consider a 250-ms on-hook condition to be a hookflash, another PBX may consider this condition to be a disconnect. Therefore, the **timing hookflash-out** command allows the administrator to define the duration of a hookflash signal generated on an FXO interface.

Figure 2 illustrates an FXS hookflash being translated to an H.245 user input.

Figure 2   *Translating an FXS Hookflash to an H.245 User Input*
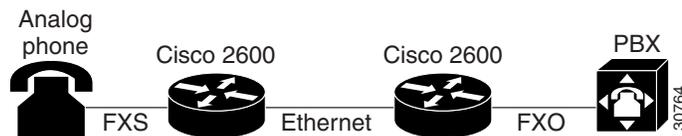
In Cisco H.323 Version 2 software, an FXS hookflash relay is generated only if the following two conditions are met:

- The other endpoint supports the reception of an H.245 hookflash and advertise this using the "Receive User Input Capability" message during H.245 capabilities exchange.

- The call is established with either the **h245-alphanumeric** or **h245-signal** variant of the **dtmf-relay** command.

This implies that the VoIP dial peer is configured for **dtmf-relay h245-alphanumeric** or **dtmf-relay h245-signal**, but not **cisco-rtp**.

Enter the **timing hookflash-input** command on FXS interfaces to specify the maximum length of a hookflash indication. If the hookflash lasts longer than the specified limit, then the FXS interface processes the indication as an onhook.

To configure hookflash relay on a gateway, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **voice-port**

2. **timing hookflash-input** *duration*

3. **timing hookflash-out** *duration*

4. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**Cisco 2600 and 3600 Series** Router(config)# **voice-port** {*slot*/*subunit*/*port*} | {*slot*/*port:ds0-group-no*} **Cisco 7200 Series** Router(config)# **voice-port** {*slot*/*port:ds0-group-no*} | {*slot-number*/*subunit-number*/*port*} **Example:** Router(config)# voice-port 1/0/0	Enters voice-port configuration mode. Keywords and arguments vary by platform. • *slot*—Slot in which the voice interface card or voice port adapter is installed. Range: 0 to 3. • *subunit*—Subunit on the voice interface card in which the voice port is located. Range: 0 to 1. • *port*—Voice port. Range varies by type of router.
Step 2	**timing hookflash-input** *duration* **Example:** Router(config-voice-port)# timing hookflash-input 200	Specifies the maximum duration of a hookflash indication, in ms. If the hookflash lasts longer than the specified limit, the Foreign Exchange Station (FXS) interface processes the indication as an on-hook. Range: 50 to 1550. Default: 600.
Step 3	**timing hookflash-out** *duration* **Example:** Router(config-voice-port)# timing hookflash-out 200	Specifies the duration, in ms, of the hookflash indications that the gateway generates on a Foreign Exchange Office (FXO) interface. Range: 50 to 1550. Default: 400.
Step 4	**exit** **Example:** Router(config-voice-port)# exit	Exits the current mode.

Configuring Multiple Codecs

Normally only one codec is specified when a dial peer is configured on a gateway. However, you can configure a prioritized list of codecs to increase the probability of establishing a connection between endpoints during the H.245 exchange phase.

Codec-order preservation enables a gateway to pass codec preferences to the terminating leg of a VoIP call. This feature was developed primarily for Cisco multiservice IP-to-IP gateways (IPIPGWs), which are configured to use a transparent codec. The transparent codec enables an IPIPGW to pass codecs from the originating endpoint to the terminating endpoint; however, previous versions of the IPIPGW did not preserve the preferential order of the codecs.

With codec-order preservation, the IPIPGW passes codecs transparently from the originating device, listed in order of preference, to the terminating device. It also enables gateways to pass user-configured codecs in their preferred order when the endpoints exchange capabilities, enabling endpoints to use the codec that best suits both devices.

Codec-order preservation is enabled by default in Cisco gateways running Cisco IOS Release 12.3(1) and later releases. No further configuration is needed.

To configure multiple codecs for a dial peer, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **voice class codec** *tag*
2. **codec preference** *value codec-type* [**bytes** *payload-size*]
3. **exit**
4. **dial-peer voice** *tag* **voip**
5. **voice-class codec** *tag*
6. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**voice class codec** *tag* **Example:** Router(config)# voice class codec 123	Enters voice-class configuration mode and assigns an identification tag number for a codec voice class. The *tag* argument is the unique number assigned to the voice class. Range: 1 to 10000. Each tag must be unique on the router.
Step 2	**codec preference** *value codec-type* [**bytes** *payload-size*] **Example:** Router(config-class)# codec preference 1 g711alaw	Adds codecs to the prioritized list of codecs. Keyword and arguments are as follows: • *value*—Order of preference, with 1 being the most preferred and 12 being the least preferred. • *codec-type*—Type of codec preferred. • **bytes** *payload-size*—Size of the voice frame in bytes. Values depend on the codec type and the packet voice protocol.
Step 3	**exit** **Example:** Router(config-class)# exit	Exits the current mode.
Step 4	**dial-peer voice** *tag* **voip** **Example:** Router(config)# dial-peer voice 456 voip	Enters dial-peer configuration mode for the VoIP dial peer designated by *tag*.
Step 5	**voice-class codec** *tag* **Example:** Router(config-dial-peer)# voice-class codec 123	Assigns a previously configured codec selection preference list (codec voice class) to the VoIP dial peer designated by *tag*. Range: 1 to 10000. Maps to the tag number created using the **voice class codec** command.
Step 6	**exit** **Example:** Router(config-dial-peer)# exit	Exits the current mode.

Verifying Preservation

To verify preservation, perform the following step.

Step 1   **show running-config**

Use this command to verify the codecs defined for a particular prioritized list of codecs.

```
Router(config-dial-peer)# show running config
```

Configuring Rotary Calling Pattern

Rotary calling pattern routes an incoming call that arrives over a telephony interface back out through another telephony interface under certain circumstances. Rotary calling pattern primarily provides reliable service during network failures.

Call establishment using rotary calling pattern is supported by rotary group support of dial peers, where multiple dial peers may match a given destination phone number and be selected in sequence. In addition, if the destinations need to be tried in a certain order, preference may be assigned. Use the **preference** command when configuring the dial peers to reflect the preferred order (0 being the highest preference and 10 the lowest).

If several dial peers match a particular destination pattern, the system attempts to place a call to the dial peer configured with the highest preference. If the call cannot be completed because of a system outage (for example, the gatekeeper or gateway cannot be contacted), the rotary call pattern performs the following tasks:

- Lists all the conditions under which this instance occurs.
- Retries the call to the next highest preference dial peer.
- Continues until no more matching dial peers are found.

If there are equal priority dial peers, the order is determined randomly.

Note   You can configure hunting-algorithm precedence. See the **preference** command in the "Dial Peer Features and Configuration" chapter in *Dial Peer Configuration on Voice Gateway Routers at* http://www.cisco.com/en/US/docs/ios/12_3/vvf_c/dial_peer/dpeer_c.html.

Configuring H.323 Support for Virtual Interfaces

H.323 support for virtual interfaces allows the IP address of the gateway to be configured so that the IP address included in the H.323 packet is always the source IP address of the gateway, regardless of the physical interface and protocol used. This single-address feature allows firewall applications to be easily configured to work with H.323 messages.

Configuring the Source IP Address of a Gateway

To configure a source IP address for a gateway, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **interface**
2. **h323-gateway voip bind srcaddr** *ip-address*
3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**interface** *type slot*/*port* **Example:** `Router(config)# interface serial 0/0`	Enters interface configuration mode for the specified interface. Keywords and arguments vary by platform.
Step 2	**h323-gateway voip bind srcaddr** *ip-address* **Example:** `Router(config-if)# h323-gateway voip bind srcaddr 192.168.0.0`	Sets the source IP address to be used for this gateway. The argument is as follows: • *ip-address*—IP address to be used for outgoing H.323 traffic, which includes H.225, H.245, and RAS messages. Typically, this is the IP address assigned to the Ethernet interface.
Step 3	**exit** **Example:** `Router(config-if)# exit`	Exits the current mode.

Verifying the Source IP Address of the Gateway

To verify the source IP address of the gateway, perform the following step.

Step 1   **show running-config**

Use this command to verify the source IP address of the gateway. The output shows the source IP address that is bound to the interface.

```
router# show running-config

interface Loopback0
 ip address 10.0.0.0 255.255.255.0
 no ip directed-broadcast
 h323-gateway voip bind srcaddr 10.0.0.0
!
interface Ethernet0/0
 ip address 172.18.194.50 255.255.255.0
 no ip directed-broadcast
 h323-gateway voip interface
 h323-gateway voip id j70f_2600_gk2 ipaddr 172.18.194.53 1719
 h323-gateway voip h323-id j70f_3640_gw1
 h323-gateway voip tech-prefix 3#
 .
 .
 .
```

In the following example, Ethernet interface 0/0 is used as the gateway interface. For convenience, the **h323-gateway voip bind srcaddr** command has been specified on the same interface. The designated source IP address is the same as the IP address assigned to the interface.

```
interface Ethernet0/0
 ip address 172.18.194.50 255.255.255.0
 no ip directed-broadcast
 h323-gateway voip interface
 h323-gateway voip id j70f_2600_gk2 ipaddr 172.18.194.53 1719
 h323-gateway voip h323-id j70f_3640_gw1
 h323-gateway voip tech-prefix 3#
 h323-gateway voip bind srcaddr 172.18.194.50
```

Configuring Annex G

This section contains the following information:

Information About Annex G

Annex G of the H.323 standard provides address resolution using border elements (BE). The BE (as described in Annex G) is colocated with the Cisco H.323 gatekeeper and provides additional address resolution capabilities. The BE can cache address information from neighboring BEs. When the gatekeeper receives a call that it cannot resolve, it can contact its local BE. If the address is in the BE's cache, the BE on the gatekeeper sends an AccessRequest to the BE in the terminating domain. If the address is not in the BE's cache, then the BE attempts to resolve the address by sending an AccessRequest to each of its neighboring BEs.

Note The Annex G BEs support Hot Standby Routing Protocol (HSRP) for high reliability and availability. You can identically configure multiple gatekeepers and BEs and use HSRP to designate a primary BE and other standby BEs. If the primary BE is down, a standby BE operates in its place. You configure the local address with an HSRP address in BE configuration.

Figure 3 illustrates a call flow for a scenario in which a call has originated in the zone administered by Border Element D, but the address cannot be resolved locally.

Figure 3   Address Resolution Using Border Elements
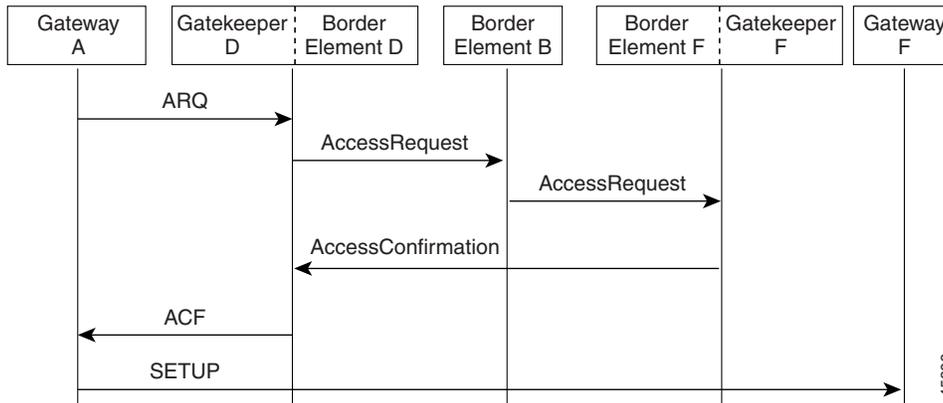
Table 2 describes how address resolution works in the illustration.

Table 2   Address Resolution Using Border Elements

Elements	Action
Gateway A to Gatekeeper D/Border Element D	GW A sends an ARQ to GK D/BE D.
Gatekeeper D/Border Element D to Border Element B	GK D/BE D is a noncaching BE and cannot resolve the address internally. Therefore, BE D sends an AccessRequest to BE B.
Border Element B to Border Element F/Gatekeeper F	BE B searches its cache to for the closest match and locates a descriptor that indicates that the access request should be sent to BE F/GK F.
Border element F/gatekeeper F to Border Element D	BE F/GK F returns an access confirmation to BE D. The access confirmation contains a template with a single address indicating where the SETUP message should be sent.
Gatekeeper D/Border Element D to Gateway A	GK D/BE D sends an ACF to GW A.
Gateway A to Gateway F	GW A sends a SETUP message to GW F.

Configuring and Provisioning an Annex G Border Element

To configure and provision an Annex G border element, use the following commands beginning in global configuration mode.

Note   Cisco supports one BE per gatekeeper.

SUMMARY STEPS

1. **call-router h323-annexg** *border-element-id*

2. **local ip** *ip-address* [**port** *local-port*]

3. **neighbor** *ip-address*

4. **port** *neighbor-port*

5. **id** *neighbor-id*

6. **cache**

7. **query-interval** *query-interval*

8. **exit**

9. Repeat Steps 3 to 8 for each neighbor BE that you configure.

10. **advertise** [**static** | **dynamic** | **all**]

11. **ttl** *value*

12. **hopcount** *value*

13. **no shutdown**

14. **timer accessrequest sequential delay** *value*

15. **exit**

16. **gatekeeper**

17. **h323-annexg** *border-element-id* **cost** *cost* **priority** *priority*

18. **prefix** *prefix*** [**seq** | **blast**]

19. **exit**

20. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**call-router h323-annexg** *border-element-id* **Example:** Router(config)# call-router h323-annexg be20	Enters Annex G configuration mode for the border element.
Step 2	**local ip** *ip-address* [**port** *local-port*] **Example:** Router(config-annexg)# local ip 192.168.0.0	Defines the local domain, including the IP address and port that this BE should use for interacting with remote BEs. Specify a port only if you want to use a nonstandard port number; otherwise, use the default standard well-known port 2099.
Step 3	**neighbor** *ip-address* **Example:** Router(config-annexg)# neighbor 192.168.0.0	Enters neighbor configuration mode to configure a neighboring BE that interacts with the local BE for the purpose of obtaining addressing information and aiding in address resolution.
Step 4	**port** *neighbor-port* **Example:** Router(config-annexg-neigh)# port 2000	(Optional) Specifies the neighbor's port number that is used for exchanging Annex G messages. Default: 2099. Do not use this command if you want to use the default value; use it only if you want a value other than 2099.
Step 5	**id** *neighbor-id* **Example:** Router(config-annexg-neigh)# id be20	(Optional) Sets the local ID of the neighboring BE. The ID is used locally to identify the neighbor and has no global significance in the Annex G network.

	Command	Purpose

Step 6	`cache`
	
	Example:
	`Router(config-annexg-neigh)# cache`
		(Optional) Configures the local BE to cache the descriptors received from its neighbors. If caching is enabled, the neighbors are queried at the specified interval for their descriptors.

Step 7	`query-interval` *query-interval* **Example:** `Router(config-annexg-neigh)# query-interval 20`	(Optional) Sets the interval at which the local BE queries the neighboring BE, in minutes. Default: 30. Do not use this command if you want to use the default query interval; use it only if you want a query interval other than 30 minutes.

Step 8	`exit`
	
	Example:
	`Router(config-annexg-neigh)# exit`
		Exits the current mode.

Step 9	Repeat Steps 3 to 8 for each neighbor BE that you configure.	—
Step 10	`advertise` [`static` | `dynamic` | `all`] **Example:** `Router(config-annexg)# advertise dynamic`	Specifies the type of descriptors that the BE advertises to its neighbors. Keywords are as follows: • **static**—Only the descriptors provisioned on this BE are advertised. This is the default. • **dynamic**—Only dynamically learned descriptors are advertised. • **all**—Both static and dynamic descriptors are advertised.
Step 11	`ttl` *value* **Example:** `Router(config-annexg)# ttl 2600`	Sets the time-to-live value for advertisements, in seconds. Default: 3180 (53 minutes).
Step 12	`hopcount` *value* **Example:** `Router(config-annexg)# hopcount 5`	Specify the maximum number of BE hops through which an address resolution request can be forwarded. Default: 7.

Step 13	`no shutdown`
	
	Example:
	`Router(config-annexg)# no shutdown`
		Starts the BE. By default, when a BE is first configured, it is shut down, so you must use this command after you configure each BE.

Step 14	`timer accessrequest sequential delay` *value* **Example:** `Router(config-annexg)# timer accessrequest sequential delay 3`	Specifies the intermessage delay (in increments of 100 ms). Range: 0 to 10. Default: 1 (100 ms). Setting this to 0 causes AccessRequest messages to be blasted to applicable neighboring BEs.

Step 15	`exit`
	
	Example:
	`Router(config-annexg)# exit`
		Exits the current mode.

	Command	Purpose

Step 16	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters H.323-gatekeeper configuration mode.

Step 17	`h323-annexg` *border-element-id* `cost` *cost* `priority` *priority* **Example:** `Router(config-gk)# h323-annexg be20 cost 35 priority 20`	Enters BE configuration mode and enables the BE on the GK. Keywords and arguments are as follows: • *border-element-id*—Identifier of the border element that you are provisioning. Associates the gatekeeper with the BE identifier that is configured on the BE. Valid values: any International Alphabet 5 (IA5) string, without spaces and up to 20 characters in length. • **cost** *cost*—Cost associated with this border element. When a gatekeeper sends requests to remote zones and to the BE in its attempt to resolve an address, the remote zone or BE that resolves the address and has the lowest cost and highest priority is given preference. Range: 1 to 99. Default: 50. • **priority** *priority*—Priority associated with this border element. When a gatekeeper sends requests to remote zones and to the BE in its attempt to resolve an address, the remote zone or BE that resolves the address and has the lowest cost and highest priority is given preference. Range: 1 to 99. Default: 50.
Step 18	`prefix` *prefix** *** [`seq` | `blast`] **Example:** `Router(config-gk-annexg)# 419*`	(Optional) Specifies the prefixes for which a BE should be queried for address resolution. Default: the GK forwards all remote zone queries to the BE. Do not use this command unless you want to restrict queries sent to the BE to a specific prefix or set of prefixes.

Step 19	`exit`
	
	Example:
	`Router(config-gk-annexg)# exit`
		Exits the current mode.

Step 20	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Configuring Basic Service Relationships

Cisco H.225 Annex G implementation supports the minimal set of Annex G features that are needed to allow Cisco border elements (BE) to interoperate with other BEs per the iNow profile for IP telephony interoperability. The implementation also allows Cisco BEs to interoperate with ClearingHouse and other third-party elements. Figure 4 depicts a basic network configuration of BEs, gatekeepers, and Clearing Houses. This feature addresses the link between the gatekeeper/border element (GK/BE) in a Cisco domain and the ClearingHouse border element that complies with the Annex G specification and the iNow profile.

Figure 4   Basic Network Configuration
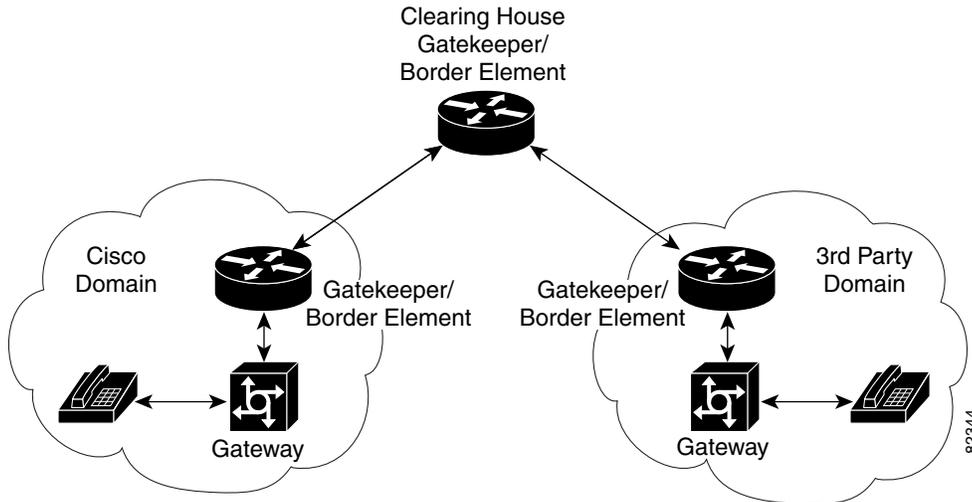
Restrictions for Basic Service Relationships

- Authentication is not supported
- Packet-level integrity checking is not supported.
- ClearingHouse CryptoTokens are not supported.
- Clustered gatekeeper and border element are not supported.
- Interoperation with LRQ-based gatekeeper networks is not supported.
- Layered Annex G networks are not supported.
- Usage indications are supported only within the context of active Service Relationships.

Prerequisites for Basic Service Relationships

- Provision Annex G border elements before configuring Annex G service relationships.

SUMMARY STEPS

1. **call-router h323-annexg** *border-element-id*
2. **access-policy neighbors-only**
3. **domain-name** *id*
4. **neighbor** *ip-address*
5. **service-relationship**
6. **outbound retry-interval** *interval_number*
7. **inbound ttl** *ttl-value*
8. **no shutdown**
9. **exit**
10. **exit**
11. **exit**

DETAILED STEPS

	Command	**Purpose**
Step 1	**call-router h323-annexg** *border-element-id* **Example:** Router(config)# call-router h323-annexg be20	Enters Annex-G configuration mode for the specified border element.
Step 2	**access-policy neighbors-only** **Example:** Router(config-annexg)# access-policy neighbors-only	As a prerequisite for configuring service relationships, sets the **access-policy** to accept requests only from known neighbors. Default: **no access-policy** allows request from any border element.
Step 3	**domain-name** *id* **Example:** Router(config-annexg)# domain-name id	Sets the domain name reported in service relationships.
Step 4	**neighbor** *ip-address* **Example:** Router(config-annexg-neigh)# neighbor 192.168.0.0	Enters neighbor configuration mode to configure a neighboring BE that interacts with the local BE for the purpose of obtaining addressing information and aiding in address resolution.
Step 5	**service-relationship** **Example:** Router(config-annexg-neigh)# service-relationship	Enters service-relationship mode.
Step 6	**outbound retry-interval** *interval_number* **Example:** Router(config-nxg-neigh-svc)# outbound retry-interval 15	(Optional) Defines the retry period for attempting to establish the outbound relationship between border elements, in seconds. Default: 30.
Step 7	**inbound ttl** *ttl-value* **Example:** Router(config-nxg-neigh-svc)# inbound 100	(Optional) Sets the duration of the inbound service relationship and interval in which the remote peer must reestablish the service relationship, in seconds. Default: 120.
Step 8	**no shutdown** **Example:** Router(config-nxg-neigh-svc)# no shutdown	Enables the service relationship.
Step 9	**exit** **Example:** Router(config-nxg-neigh-svc)# exit	Exits the current mode.

	Command	Purpose

Step 10	`exit`
	
	Example:
	`Router(config-annexg-neigh)# exit`
		Exits the current mode.

Step 11	`exit`
	
	Example:
	`Router(config-annexg)# exit`
		Exits the current mode.

Configuring Usage Indication

To enter usage indication submode and configure usage-indicators after service relationships are established, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **call-router h323-annexg** *border-element-id*
2. **neighbor** *ip-address*
3. **usage-indication**
4. **retry interval** *seconds*
5. **retry window** *minutes*
6. **exit**
7. **exit**
8. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	`call-router h323-annexg` *border-element-id* **Example:** `Router(config)# call-router h323-annexg be20`	Enters Annex-G configuration mode for the specified border element.
Step 2	`neighbor` *ip-address* **Example:** `Router(config-annexg)# neighbor 192.168.0.0`	Enters neighbor configuration mode to configure a neighboring BE that interacts with the local BE for the purpose of obtaining addressing information and aiding in address resolution.

Step 3	`usage-indication`
	
	Example:
	`Router(config-annexg-neigh)# usage-indication`
		Enters config-nxg-neigh-usg mode.

Step 4	`retry interval` *seconds*	(Optional) Defines the time, in seconds, between delivery attempts. Default: 900.

	Example:
	`Router(config-nxg-neigh-usg)# retry`
	`interval 600`	

Step 5	`retry window` *minutes*	(Optional) Defines the total time, in minutes, that a border element attempts delivery. Default: 1440 (24 hours).

	Example:
	`Router(config-nxg-neigh-usg)# retry window`
	`1200`	

Step 6	`exit`	Exits the current mode.
	Example: `Router(config-nxg-neigh-usg)# exit`	
Step 7	`exit`	Exits the current mode.
	Example: `Router(config-annexg-neigh)# exit`	
Step 8	`Router(config-annexg)# ` **`exit`**	Exits the current mode.
	Example: `Router(config-annexg)# exit`	

Verifying Annex G Configuration

To verify Annex G configuration, perform the following step.

Step 1 **show call-router status**

Use this command to display Annex G border-element status.

```
Router# show call-router status neighbors

ANNEX-G CALL ROUTER STATUS:
  ===========================
    Border Element ID Tag   : Celine
    Domain Name             : Celine-Domain
    Border Element State    : UP
    Border Element Local IP : 172.18.193.31:2099
    Advertise Policy        : STATIC descriptors
    Hopcount Value          : 7
    Descriptor TTL          : 3180
    Access Policy           : Neighbors only
    Current Active Calls    : 0
    Current Calls in Cache  : 0
    Cumulative Active Calls : 0
    Usage Ind Messages Sent : 0
    Usage Ind Cfm Rcvd      : 0
    IRRs Received           : 0
    DRQs Received           : 0
    Usage Ind Send Retrys   : 0

    NEIGHBOR INFORMATION:
    =====================
```

```
Local Neighbor ID : (none)
Remote Element ID : (unknown)
Remote Domain ID  : (unknown)
IP Addr           : 1.2.3.4:2099
Status            : DOWN
Caching           : OFF
Query Interval    : 30 MIN (querying disabled)
Usage Indications :
  Current Active Calls : 0
  Retry Period        : 600 SEC
  Retry Window        : 3600 MIN
Service Relationship Status: ACTIVE
  Inbound Service Relationship  : DOWN
    Service ID     : (none)
    TTL            : 1200 SEC
  Outbound Service Relationship : DOWN
    Service ID     : (none)
    TTL            : (none)
    Retry interval : 120 SEC (0 until next attempt)
```

Configuring H.225

This section contains the following information:

Associating the H.323 Voice Class with Each VoIP Dial Peer

To associate the H.323 voice class with a dial peer, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **dial-peer voice** *tag* **voip**

2. **voice-class h323** *number*

3. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`dial-peer voice` *tag* `voip`
	
	Example:
	`Router(config)# dial-peer voice 123 voip`
		Enters dial-peer configuration mode for the remote VoIP dial peer designated by *tag*.

Step 2	`voice-class h323` *number* **Example:** `Router(config-dial-peer)# voice-class h323 456`	Associates the specified H.323 voice class (and all of its related attributes) with the dial peer.

Step 3	`exit`
	
	Example:
	`Router(config-dial-peer)# exit`
		Exits the current mode.

Configuring the SETUP Response Timeout Value

To configure the timeout value for the response of the outgoing SETUP message, use the following commands in global configuration mode.

SUMMARY STEPS

1. **voice class h323** *number*
2. **h225 timeout setup** *value*
3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	`voice class h323` *number* **Example:** `Router(config)# voice class h323 123`	Enters voice-class mode to create or modify the specified H.323 voice class.
Step 2	`h225 timeout setup` *value* **Example:** `Router(config-class)# h225 timeout setup 10`	Sets the timeout value, in seconds, for the response of the outgoing SETUP message. If the timer expires, the GK tries an alternate endpoint (if configured and specified in the ACF); otherwise, it terminates the call. Range: 0 to 30. Default: 15.

Step 3	`exit`
	
	Example:
	`Router(config-class)# exit`
		Exits the current mode.

Configuring the Number of Concurrent Calls Per Connection

To limit the number of concurrent calls on an H.225 TCP connection, use the following commands in global configuration mode.

SUMMARY STEPS

1. **voice service voip**
2. **h323**
3. **session transport tcp** [**calls-per-connection** *value*]
4. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**voice service voip** **Example:** `Router(config)# voice service voip`	Enters voice-service configuration mode.
Step 2	**h323** **Example:** `Router(conf-voi-serv)# h323`	Enters H.323-voice-service configuration mode.
Step 3	**session transport tcp** [**calls-per-connection** *value*] **Example:** `Router(conf-serv-h323)# session transport tcp`	Sets the number of concurrent calls for a single TCP connection. Range: 1 to 9999. Default: 5.
Step 4	**exit** **Example:** `Router(conf-serv-h323)# exit`	Exits the current mode.

Changing the Idle Timer for Concurrent Calls

To change the H.225 idle timer for concurrent calls, use the following commands in global configuration mode.

SUMMARY STEPS

1. **voice service voip**
2. **h323**
3. **h225 timeout tcp call-idle** {**value** *value* | **never**}
4. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`voice service voip`
	
	Example:
	`Router(config)# voice service voip`
		Enters voice-service configuration mode.

Step 2	`h323`
	
	Example:
	`Router(conf-voi-serv)# h323`
		Enter s H.323-voice-service configuration mode.

Step 3	`h225 timeout tcp call-idle {value value | never}`
	
	Example:
	`Router(conf-serv-h323)# h225 timeout tcp call-idle never`
		Sets a timer to maintain a connection when no calls are active.

Step 4	`exit`
	
	Example:
	`Router(conf-serv-h323)# exit`
		Exits the current mode.

Configuring Overlap Signaling on H.323 Terminating Gateways

The terminating gateway is responsible for collecting all the called number digits. Overlap signaling is implemented by matching destination patterns on the dial peers. When H.225 signal overlap is configured on the originating gateway, it sends the SETUP to the terminating gateway once a dial-peer match is found. The originating gateway sends all further digits received from the user to the terminating gateway using INFO messages until it receives a sending complete message from the user. The terminating gateway receives the digits in SETUP and subsequent INFO messages and does a dial-peer match. If a match is found, it sends a SETUP with the collected digits to the PSTN. All subsequent digits are sent to the PSTN using INFO messages to complete the call.

To configure overlap signaling on H.323 terminating gateways, perform the following steps.

SUMMARY STEPS

1. **voice service voip**
2. **h323**
3. **h225 signal overlap**
4. **h225 timeout t302**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`voice service voip`
	
	Example:
	`Router(config)# voice service voip`
		Enters VoIP voice-service configuration mode.

Step 2	`h323`
	
	Example:
	`Router(conf-voi-serv)# h323`
		Enters H.323 voice-service configuration mode.

Step 3	`h225 signal overlap`
	
	Example:
	`Router(conf-serv-h323)# h225 signal overlap`
		Activates overlap signaling to the destination gateway.

Step 4	`h225 timeout t302` *seconds* **Example:** `Router(conf-serv-h323)# h225 timeout t302 15`	Sets the t302 timer timeout value. The argument is as follows: • *seconds*— Number of seconds for timeouts. Range: 1 to 30.

Step 5	`exit`
	
	Example:
	`Router(conf-serv-h323)# exit`
		Exits the current mode.

Configuring No Retry on User Busy in an H.323 Gateway

This section describes how to configure the alternate endpoint hunt for failed calls in an IP-to-IP Gateway (IPIPGW) based on Q.850 disconnect cause codes.

The default behavior of the gateway is to retry all alternate endpoints received from the gatekeeper regardless of the ReasonComplete reason. Perform this task if you want to stop the alternate endpoint hunt retry attempts when the ReasonComplete is User-busy or Invalid-number.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **h323**
5. **no h225 alt-ep hunt** [**all** | *cause-code*]

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		
		• Enter your password if prompted.

Step 2	`configure terminal`
	
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`voice service voip`
	
	Example:
	`Router(config)# voice service voip`
		Enters voice service configuration mode and specifies a voice encapsulation type.

Step 4	`h323`
	
	Example:
	`Router(conf-voice-service)# h323`
		Enters H.323 configuration mode.

Step 5	`no h225 alt-ep hunt user-busy`
	
	Example:
	`Router(conf-serv-h323)# no h225 alt-ep hunt user-busy`
		Disables alternate endpoint hunts.
		
		• all—Continue hunt for all disconnect cause codes.
		
		• cause-code—May be entered as standard Q.850 number or as text.
		
		Note Alternate endpoint hunt is enabled for all cause codes by default. Command will be visible only for the negated hunt cause codes (with **no** prefixed).
		
		Note This functionality, requires a Cisco Gatekeeper. See the "Configuring H.323 Gatekeepers and Proxies" chapter of this guide.

Examples

The following example shows a configuration that disables the alternate endpoint hunt for user busy and no answer:

!

voice service voip

h323

no h225 alt-ep hunt user-busy

no h225 alt-ep hunt no-answer

!

Configuring the VoIP Transport Method

To configure the VoIP transport method, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **voice service voip**
2. **h323**
3. **session transport** {**udp** | **tcp**}
4. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**voice service voip** **Example:** `Router(config)# voice service voip`	Enters voice-service configuration mode.
Step 2	**h323** **Example:** `Router(conf-voi-serv)# h323`	Enters H.323-voice-service configuration mode.
Step 3	**session transport** {**udp**	**tcp**} **Example:** `Router(conf-serv-h323)# session transport tcp`	Sets the underlying transport layer protocol for H.323 messages to be used across all VoIP dial peers. If you specify **udp**, Annex E is used. For concurrent calls, you must specify **tcp**.
Step 4	**exit** **Example:** `Router(conf-serv-h323)# exit`	Exits the current mode.

Configuring Zone Bandwidth Management

In the current version of the Cisco H.323 gateway (which conforms with H.323 version 3), the reported bandwidth is bidirectional. Initially, 128 kb is reserved. If the endpoints in the call select a more efficient codec, the gatekeeper is notified of the bandwidth change.

If you prefer to use the behavior of previous Cisco H.323 gateway versions for zone bandwidth management, configure the gateway accordingly.

To configure the Cisco H.323 gateway to use its previous behavior, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gateway**

2. **emulate cisco h323 bandwidth**

3. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gateway`
	
	Example:
	`Router(config)# gateway`
		Enters gateway configuration mode.

Step 2	`emulate cisco h323 bandwidth`
	
	Example:
	`Router(config-gateway)# emulate cisco h323 bandwidth`
		Sets the gateway to use its previous behavior for bandwidth management.

Step 3	`exit`
	
	Example:
	`Router(config-gateway)# exit`
		Exits the current mode.

Configuring Generic Transparency Descriptor for GKTMP Using SS7 Interconnect for Voice Gateways Version 2.0

This section contains the following information:

Information About GTD for GKTMP Using SS7 Interconnect for Voice Gateways

The GTD for GKTMP Using SS7 Interconnect for Voice Gateways feature provides additional functionality to Cisco gateways and gatekeepers in a Cisco SS7 Interconnect for Voice Gateways Solution. The generic transparency descriptor or generic telephony descriptor (GTD) format is defined in the a Cisco-proprietary draft. GTD format defines parameters and messages of existing SS7 ISUP protocols in text format and allows SS7 messages to be carried as a payload in the H.225 RAS messages between gateway and gatekeeper. With the GTD feature, the gatekeeper extracts the GTD message and the external route server derives routing and accounting information based upon the GTD information provided from the Cisco Gatekeeper Transaction Message Protocol (GKTMP).

Currently routing on Cisco gateways is based on generic parameters such as originating number, destination number, and port source. Adding support for SS7 ISUP messages allows the VoIP network to use additional routing enhancements found in traditional TDM switches.

Figure 5 shows an example of a Cisco SS7 Interconnect for Voice Gateways solution using the GTD feature.

Figure 5   *Cisco SS7 Interconnect for Voice Gateways Solution With the GTD Feature*
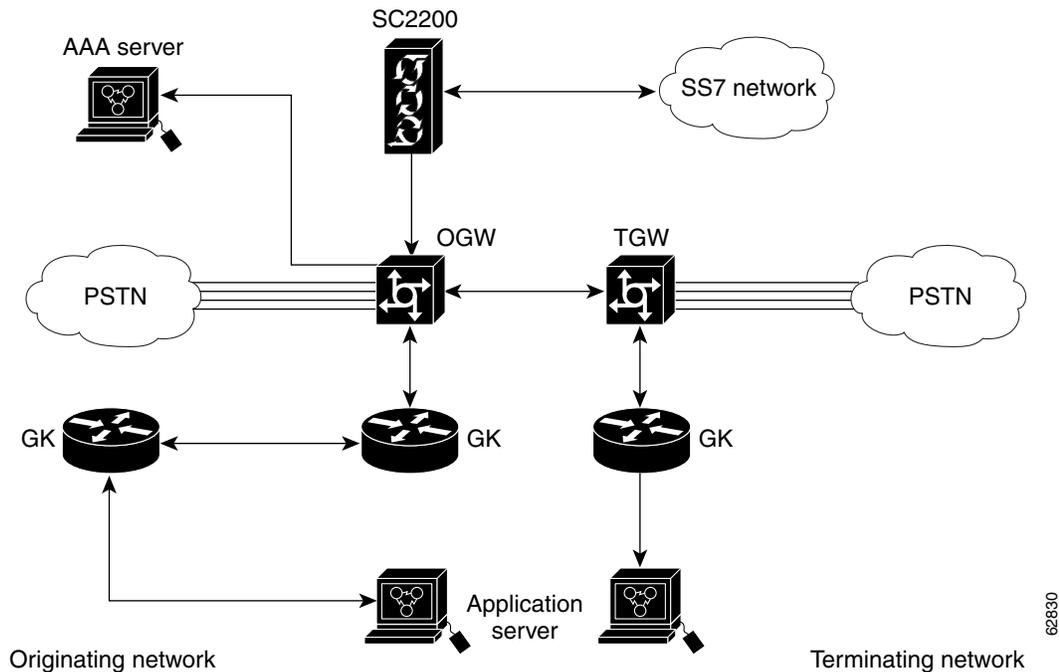
In the originating network, the following events occur:

- The Cisco SC2200 receives SS7 messages from the SS7 network and encapsulates them into GTD format. The messages are then passed to the Cisco originating gateway (OGW).

- Using the GTD feature, the OGW transmits the GTD payload in the Admission Request (ARQ) message to GK1.

- GK1 transmits the GTD payload in a Location Request (LRQ) message to GK2.

- GK 2 uses GKTMP with the GTD feature to decode the GTD payload and transmits it to the route server with the REQUEST LRQ message.

- The route server returns a RESPONSE LCF (Location Confirmation) message that includes the GTD payload to GK2. The route server also returns a service descriptor code (SC) field to GK2. (The SC field is transmitted to the AAA server for billing purposes. The SC field conveys the Carrier ID and trunk number information that is determined by and passed from the Route Server.)

- GK2 passes the LCF that includes the GTD payload and the SC field to GK1.

- GK1 sends an Admission Confirmation (ACF) message that includes the GTD payload to the OGW, along with the SC field.

- The OGW sends the SC field and call detail records (CDRs) to the AAA server.

- When the call ends, the Cisco SC2200 receives the SS7 messages, encodes them into GTD format, and passes them to the OGW.

- The OGW sends a Disengage Request (DRQ) with the GTD payload to GK1.

- GK1 sends the DRQ with the GTD payload to the route server.

In the terminating network, the following events occur:

- The OGW sends the GTD in H.225 the SETUP message to the terminating gateway (TGW).

- The TGW sends regular RAS messages to the gatekeeper.

Prerequisites for GTD for GKTMP Using SS7 Interconnect for Voice Gateways

- Configure your VoIP network and the Cisco SS7 Interconnect for Voice Gateways Solution, including the following components:

  – Cisco SC2200—Cisco MGC Software Release 9.1(5) or higher

  – Cisco IOS gateways—Cisco IOS Release 12.2(2)XU or higher

  – Cisco IOS gatekeepers—Cisco IOS Release 12.2(2)XU or higher

  – Route servers

  – AAA servers

> **Note** For more information on software and components of the Cisco SS7 Interconnect for Voice Gateways Solution, see the release notes and other documentation at
> http://www.cisco.com/univercd/cc/td/doc/product/access/sc/rel7/soln/das/index.htm

Configuring GTD System-Wide

To configure the GTD feature system-wide for a VoIP network, enter the commands shown below. If you want to configure the feature on individual dial peers rather than system-wide, use the commands in the "Configuring GTD for a Dial Peer" section on page 98.

SUMMARY STEPS

1. **voice service voip**

2. **signaling forward** {**none** | **unconditional**}

3. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`voice service voip`
	
	Example:
	`Router(config)# voice service voip`
		Enters voice-service configuration mode.

Step 2	`signaling forward {unconditional	none}`  **Example:** `Router(conf-voi-serv)# signaling forward unconditional`	Chooses whether or not the gateway forwards signaling payload to another gateway. Keywords are as follows: • **unconditional**—Forward payload to the remote end, even if the attached external route server has modified the payload. • **none**—Do not forward payload.

Step 3	`exit`
	
	Example:
	`Router(conf-voi-serv)# exit`
		Exits the current mode.

Configuring GTD for a Dial Peer

To configure the GTD feature on an individual dial peer, follow the steps below.

SUMMARY STEPS

1. **dial-peer voice** *tag* **voip**
2. **signaling forward** {**conditional** | **unconditional** | **none**}
3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**dial-peer voice** *tag* **voip** **Example:** Router(config)# dial-peer voice 4 voip	Enters dial-peer configuration mode for the VoIP dial peer designated by *tag*.

Step 2	**signaling forward** {**conditional** | **unconditional** | **none**}
	
	Example:
	Router(config-dial-peer)# signaling forward conditional
		Chooses whether or not the gateway forwards signaling payload to another gateway. Keywords are as follows:
		
		• **conditional**—Forward payload as defined in the **session target** command.
		
		  – If the target is a non-RAS target, forward to the H.323 endpoint using H.225 messages.
		
		  – If the target is a RAS target, for a non-GTD payload, forward. For a GTD payload, encapsulate the payload in an ARQ/DRQ message and send it to the originating gateway. The gateway conveys the payload to the GKTMP and external route server for a flexible route decision based up the ISUP GTD parameters. The gateway then conditionally forwards the payload based upon the route server's instruction.
		
		• **unconditional**—Forward the payload to the remote end, even if the attached external route server has modified the payload.
		
		• **none**—Do not forward payload.

Step 3	**exit** **Example:** Router(config-dial-peer)# exit	Exits the current mode.

Verifying GTD

To verify GTD, perform the following step.

Step 1   **show running-config**

Use this command to verify that the GTD feature is configured.

The following shows sample output for system-wide employment.

```
Router# show running-config

Building configuration...

Current configuration : 4192 bytes
!
version 12.2
service config
no service single-slot-reload-enable
no service pad
service timestamps debug uptime
```

```
service timestamps log uptime
no service password-encryption
service internal
service udp-small-servers
!
hostname as5300-2
!
voice service voip
 signaling forward unconditional
 h323
.
.
.
```

The following shows sample output for employment on select dial peers.

```
Router# show running-config

Building configuration...

Current configuration : 4192 bytes
!
version 12.2
service config
no service single-slot-reload-enable
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service internal
service udp-small-servers
!
hostname as5300-2
!
.
.
.
!
dial-peer voice 1 pots
 application session
 incoming called-number 25164
 port 0:D
!
dial-peer voice 1513 voip
 destination-pattern 1513.......
 session target ipv4:1.8.156.3
!
dial-peer voice 1408525 voip
 destination-pattern 1408525....
!
dial-peer voice 1800877 voip
 destination-pattern 1800877....
 session target ipv4:1.8.156.3
!
dial-peer voice 2 pots
 destination-pattern 51550
 no digit-strip
 direct-inward-dial
 port 3:D
!
dial-peer voice 51557 voip
 destination-pattern 51557
 signaling forward unconditional
 session target ras
!
```

```
dial-peer voice 52557 voip
 destination-pattern 52557
 signaling forward unconditional
 session target ipv4:1.8.156.3
!
.
.
.
```

Configuring H.323 Version 4 Zone Prefix Registration

The H.323v4 Gateway Zone Prefix Registration Enhancements feature provides support for two capabilities included in H.323 Version 4: additive registration and dynamic zone prefix registration. Additive registration allows a gateway to add to or modify a list of aliases contained in a previous registration without first unregistering from the gatekeeper. Dynamic zone prefix registration allows a gateway to register actual public switched telephone network (PSTN) destinations served by the gateway with its gatekeeper.

Information About H.323v4 Gateway Zone Prefix Registration Enhancements

To configure the H.323v4 Gateway Zone Prefix Registration Enhancements feature, you must understand the following concepts:

- Additive Registration, page 101
- Dynamic Zone Prefix Registration, page 101

Additive Registration

Prior to H.323 version 4, there was no way for a large device, such as a gateway, to register hundreds or thousands of E.164 alias addresses with a gatekeeper. The limiting factor was the size of a User Datagram Protocol (UDP) packet, which does not allow an unlimited number of aliases in a single heavyweight **registration request** (RRQ) RAS message.

To allow an endpoint to register an unlimited number of aliases with the gatekeeper, H.323v4 introduces the concept of *additive registration*. When the gateway registers with a gatekeeper, it provides an initial list of aliases. Additive registration allows the gateway to send subsequent RRQ messages with more lists of aliases until the gatekeeper has the complete list of the gateway's aliases.

When the gatekeeper wants to acknowledge only a subset of the aliases proposed in an additive RRQ, the gatekeeper returns a registration confirm (RCF) RAS message specifying the accepted aliases. The gateway assumes that the aliases not listed in the RCF were rejected.

Dynamic Zone Prefix Registration

H.323v4 allows a gateway to register actual zone prefixes that it can terminate to the PSTN with a gatekeeper. A gateway can register multiple zone prefixes with the gatekeeper via the RRQ message and subsequently remove one or more zone prefixes using an unregistration request (URQ) RAS message indicating the specific prefixes to be removed. When the gatekeeper receives the URQ, it leaves the gateway registered and removes the specified zone prefixes.

When the H.323v4 Gateway Zone Prefix Registration Enhancements feature is enabled on a trunking gateway, all addresses specified by the destination patterns in the plain old telephone service (POTS) dial peers that are operational are advertised to the gatekeeper.

The gatekeeper treats these addresses similarly to configured zone prefixes. The dynamically registered zone prefixes are used in routing decisions just as if they had been entered using the **zone prefix** command. Dynamically registered zone prefixes have a default gateway priority of 5.

Table 3 shows destination patterns on gateway GW1 and how the gatekeeper GK1 views the dynamically registered prefixes.

Table 3   *Gateway Prefixes Dynamically Registered on the Gatekeeper*

GW1 Configuration	GK1 Corresponding Pseudo Configuration

```dial-peer voice 919 pots```
 ```destination-pattern 919.......```
 ```port 0:D```
	```gatekeeper```
	 ```zone local GK1 cisco.com 172.18.197.132```
	 ```zone prefix GK1 919....... gw-priority 5 GW1```

```dial-peer voice 5551001pots```
 ```destination-pattern 5551001```
 ```port 0:D```
	```gatekeeper```
	 ```zone local GK1 cisco.com 172.18.197.132```
	 ```zone prefix GK1 5551001* gw-priority 5 GW1```

```dial-peer voice 408 pots```
 ```destination-pattern 408T```
 ```port 0:D```
	```gatekeeper```
	 ```zone local GK1 cisco.com 172.18.197.132```
	 ```zone prefix GK1 408* gw-priority 5 GW1```

Configuring H.323v4 Gateway Zone Prefix Registration Enhancements

This section contains the following tasks:

Enabling the Dynamic Zone Prefix Registration

This task shows you how to enable the gateway to send an advertisement of dynamic prefixes in additive RRQ RAS messages automatically to the gatekeeper.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **h323**
5. **ras rrq dynamic prefixes**
6. **exit**
7. **gatekeeper**
8. **rrq dynamic-prefixes-accept**
9. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		
		• Enter your password if prompted.

Step 1	`configure terminal`
	
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 2	`voice service voip`
	
	Example:
	`Router(config)# voice service voip`
		Enters voice service configuration mode.

Step 3	`h323`
	
	Example:
	`Router(config-voice-service)# h323`
		Enters the H.323 voice service configuration mode.

Step 4	`ras rrq dynamic prefixes`
	
	Example:
	`Router(conf-serv-h323)# ras rrq dynamic prefixes`
		Enables the gateway to send an advertisement of dynamic prefixes in additive RRQ RAS messages.
		
		Note In Cisco IOS Release 12.2(15)T, this command was enabled by default. Beginning in Cisco IOS Release 12.3(3), this command is disabled by default.

Step 5	`exit`
	
	Example:
	`Router(conf-serv-h323)# exit`
		Exits voice service voip h323 configuration mode and enters global configuration mode.

Step 6	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 7	`rrq dynamic-prefixes-accept`
	
	Example:
	`Router(config-gk)# rrq dynamic-prefixes-accept`
		Enables the gatekeeper to receive the RRQ RAS messages from the gateway.
		
		Note In Cisco IOS Release 12.2(15)T, this command was enabled by default. Beginning in Cisco IOS Release 12.3(3), this command is disabled by default.

Step 8	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits gatekeeper configuration mode.

Enabling the Dynamic Zone Prefix Registration Along with the Gateway Priority

This task shows you how to configure the priority to the dynamic prefixes on the gateway. Allowing you to configure a different priority to each of the dynamic prefix. When configured, the gateway sends the priority along with the prefixes in additive RRQ and the gatekeeper assigns the received priority to the gateway for a given dynamic prefix.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **h323**
5. **terminal-alias-pattern 22... priority 8**
6. **terminal-alias-pattern 23* priority 7**
7. Repeat Step 5 for each prefix on the gateway.
8. **ras rrq dynamic prefixes**
9. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 1	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 2	**voice service voip** **Example:** `Router(config)# voice service voip`	Enters voice-service configuration mode.
Step 3	**h323** **Example:** `Router(config-voice-service)# h323`	Enters H.323 voice-service configuration mode.
Step 4	**terminal-alias-pattern 22... priority 8** **Example:** `Router(conf-serv-h323)# terminal-alias-pattern 23 priority 8`	Assigns priority to a dynamic prefix. The prefixes mentioned in this command should exactly match the prefixes configured in the **destination-pattern** command of POTS dial-peer. **Note** Dynamic zone prefix does not support destination patterns with regular expression. It accepts the patterns ending with dot "." and asterisk "*" only.

	Command or Action	Purpose

Step 5	`terminal-alias-pattern 23* priority 7`
	
	Example:
	`Router(conf-serv-h323)# terminal-alias-pattern 23* priority 7`
		Assigns priority to a dynamic prefix. The prefixes mentioned in this command should exactly match the prefixes configured in the **destination-pattern** command of POTS dial-peer.
		
		Note Dynamic zone prefix does not support destination patterns with regular expression. It accepts the patterns ending with dot "." and asterisk "*" only.

Step 6	Repeat Step 5 for each priority you configure.	—
Step 7	**`ras rrq dynamic prefixes`** **Example:** `Router(conf-serv-h323)# ras rrq dynamic prefixes`	Enables the gateway to send an advertisement of dynamic prefixes in additive RRQ RAS messages. **Note** In Cisco IOS Release 12.2(15)T, this command was enabled by default. Beginning in Cisco IOS Release 12.3(3), this command is disabled by default.
Step 8	**`exit`** **Example:** `Router(conf-serv-h323)# exit`	Exits gatekeeper configuration mode.

Verifying Gateway Advertisement of Dynamic Zone Prefixes

Perform this task to verify that the gateway is advertising dynamic zone prefixes.

SUMMARY STEPS

1. **enable**
2. **show gateway**
3. **show h323 gateway prefixes**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**`enable`** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**`show gateway`** **Example:** `Router# show gateway`	Displays the current status of the gateway.
Step 3	**`show h323 gateway prefixes`** **Example:** `Router# show h323 gateway prefixes`	Displays the status of the gateway destination pattern database and the status of the individual destination patterns along with it's configured priority. • Verify that gateway additive RRQ support is enabled, that the pattern database is active, and that destination patterns have been acknowledged by the gatekeeper.

Verifying Gatekeeper Processing of Additive RRQ Messages

Perform this task to verify that the gatekeeper is processing additive RRQ messages.

SUMMARY STEPS

1. **enable**
2. **show gatekeeper zone prefix** [**all**]
3. **show gatekeeper gw-type-prefix**
4. **show gatekeeper endpoints**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		• Enter your password if prompted.

Step 2	`show gatekeeper zone prefix` [`all`] **Example:** `Router# show gatekeeper zone prefix all`	Displays the gatekeeper zone prefix table. • Use the **all** keyword to display the dynamic zone prefixes registered by each gateway. • Use the **include** filter with the **all** keyword to display the prefixes associated with a particular gateway.

Step 3	`show gatekeeper gw-type-prefix`
	
	Example:
	`Router# show gatekeeper gw-type-prefix`
		Displays the gateway technology prefix table.

Step 4	`show gatekeeper endpoints`
	
	Example:
	`Router# show gatekeeper endpoints`
		Displays the status of all registered endpoints for a gatekeeper.

Troubleshooting H.323v4 Gateway Zone Prefix Registration Enhancements

Use the **debug h225 asn1** command to observe the dynamic registration process. The **debug h225 asn1** command is intended only for troubleshooting purposes because the volume of output generated by the software can result in severe performance degradation on the router.

Prerequisites

Attach a console directly to a router running Cisco IOS Release 12.2(15)T or a later release.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **logging buffered** [*buffer-size* | *level*]

4. **no logging console**

5. **end**

6. **debug h225 asn1**

7. **show logging** [**history** | **slot** *slot-number* | **summary** | **count**]

8. **no debug h225 asn1**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`enable`
	
	Example:
	`Router> enable`
		Enables privileged EXEC mode.
		
		• Enter your password if prompted.

Step 2	`configure terminal`
	
	Example:
	`Router# configure terminal`
		Enters global configuration mode.

Step 3	`logging buffered` [*buffer-size* | *level*] **Example:** `Router(config)# logging buffered 65536`	Limits messages logged to an internal buffer based on severity.

Step 4	`no logging console`
	
	Example:
	`Router(config)# no logging console`
		Disables all logging to the console terminal.
		
		• To reenable logging to the console, use the **logging console** command in global configuration mode.

Step 5	`end`
	
	Example:
	`Router(config)# end`
		Exits to privileged EXEC mode.

Step 6	`debug h225 asn1`
	
	Example:
	`Router# debug h225 asn1`
		Displays ASN1 contents of RAS and Q.931 messages.
		
		⚠
		Caution This command slows down the system considerably. Connections may time out.

Step 7	`show logging` [`history` | `slot` *slot-number* | `summary` | `count`] **Example:** `Router# show logging`	Displays the state of logging (syslog).

Step 8	`no debug h225 asn1`
	
	Example:
	`Router# no debug h225 asn1`
		Disables display of ASN1 contents of RAS and Q.931 messages.

Configuring Call Admission Control

Cisco H.323 gateways provide the ability to support resource-based call admission control (CAC) processes. These resources include system resources such as CPU, memory, and call volume, and interface resources such as call volume.

If system resources are not available to admit the call, two kinds of actions are provided: system denial which busyouts all of T1 or E1 or per call denial, which disconnects, hairpins, or plays a message or tone. If the interface-based resource is not available to admit the call, the call is dropped from the session protocol.

Note For information on CAC, see *Trunk Connections and Conditioning Features* at http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/vvfax_c/vcltrunk.htm.

Configuring Trunk-Based and Carrier-Based Routing

Voice wholesalers use multiple ingress and egress carriers to route traffic. A call coming into a gateway on a particular ingress carrier must be routed to an appropriate egress carrier. As networks grow and become more complicated, the dial plans needed to route the carrier traffic efficiently become more complex and the need for carrier-sensitive routing (CSR) increases.

Note For information on routing, see *VoIP Gateway Trunk and Carrier Based Routing Enhancements* at the following URL: http://www.cisco.com/en/US/docs/ios/12_2t/12_2t11/feature/guide/ftgwrepg.html

Configuring Signal ISDN B-Channel ID to Enable Application Control of Voice Gateway Trunks

This section contains the following information:

Information About Signal ISDN B-Channel ID

The Signal ISDN B-Channel ID to Enable Application Control of Voice Gateway Trunks feature enables call-management applications to identify specific ISDN bearer (B) channels used during a voice-gateway call for billing purposes. With identification of the B channel, H.323 gateways can enable port-specific features such as voice recording and call transfer.

In Cisco IOS releases prior to 12.3(7)T, fields used to store call leg information regarding the telephony port do not include B channel information. B-channel information is used to describe incoming ISDN call legs. The Signal ISDN B-Channel ID to Enable Application Control of Voice Gateway Trunks feature allows H.323 and SIP gateways to receive B-channel information from incoming ISDN calls. The acquired B-channel information can be used during call transfer or to route a call.

SIP and H.323 gateways use two different commands to enable receiving the B channel of a telephony call leg. Using a different command for each protocol allows users to run the two protocols on one gateway simultaneously.

> **Note** For information on using this feature on SIP gateways, see the information on SIP ISDN support features in the *Cisco IOS SIP Configuration Guide* at
> http://www.cisco.com/en/US/docs/ios/voice/sip/configuration/guide/15_0/sip_15_0_book.html

For H.323, if the **billing b-channel** command is configured, the H.323 gateway accesses B-channel information on all calls in the ARQ, LRQ, and GKTMP messages.

Configuring Signal ISDN B-Channel ID

To provide H.323 users with B-channel information, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **voice service voip**
2. **h323**
3. **billing b-channel**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`voice service voip`
	
	Example:
	`Router(config)# voice service voip`
		Enters voice-service configuration mode and specifies a voice-encapsulation type.

Step 2	`h323`
	
	Example:
	`Router(conf-voi-serv)# h323`
		Enters H.323-voice-service configuration mode.

Step 3	`billing b-channel`
	
	Example:
	`Router(conf-serv-h323)# billing b-channel`
		Enables the H.323 gateway to access B-channel information on all H.323 calls.

Step 4	`exit`
	
	Example:
	`Router(conf-serv-h323)# end`
		Exits the current mode.

Troubleshooting Signal ISDN B-Channel ID

To troubleshoot signal ISDN B-channel ID problems, perform the following steps.

Step 1 **debug h245 asn1**

Use this command to display ASN1 contents of H.245 messages.

The following sample command output shows an H.323 ARQ nonstandard message. The format of the B-channel billing information is: 1 is the D-channel ID, 1 is the T1 controller, and 10 is the B-channel.

```
Router# debug h245 asn1

.
.
.
value ARQnonStandardInfo ::=
 {
   sourceAlias
   {
   }
   sourceExtAlias
   {
   }
   interfaceSpecificBillingId 1:D 1:DS1 10:DS0
   gtd '49414D2C0D0A50524E2C6973646E2A2C2...'H
 }
.
.
.
```

Step 2 **debug gatekeeper servers**

Use this command on gatekeeper to trace all the message exchanges between a gatekeeper and an external application. It also displays any errors that occur in sending messages to the external application or in parsing messages from the external application.

The following sample command output also shows B-channel information. The format of the B-channel billing information is as follows: 1 is the D-channel ID, 1 is the T1 controller, and 10 is the B-channel.

```
Router# debug gatekeeper servers

"REQUEST ARQ
Version-id:402
From:voip6-2600-1
To:GKTMP_SERVER
Transaction-Id:81A3EB4000000001
Content-Length:258
i=I:1.3.26.21:1720
s=E:9190001 H:voip6-5300-1
d=E:4080001
b=1280
A=F
C=C13CB8DE-C47F-11D3-80A9-FC0BFCA7B068
c=C13D5506-C47F-11D3-80AB-FC0BFCA7B068
B= 1:D 1:DS1 10:DS0
```

Configuring H.323 VoIP Call Preservation Enhancements for WAN Link Failures

H.323 VoIP call preservation enhancements for WAN link failures sustains connectivity for H.323 topologies where signaling is handled by an entity that is different from the other endpoint, such as a gatekeeper that provides routed signaling or a call agent, such as the Cisco BTS 10200 Softswitch, Cisco PGW 2200, or Cisco CallManager, that brokers signaling between the two connected parties.

Call preservation is useful when a gateway and the other endpoint (typically an Cisco Unified IP phone) are collocated at the same site and the call agent is remote and therefore more likely to experience connectivity failures.

Note If a preserved H.323 call is torn down at a IP PBX, a call-stop record will be generated while Real-time Transport Protocol (RTP) is still flowing. Such an event can be misused to generate a signaling error and allow toll bypass, thus affecting per-call billing integrity.

H.323 call preservation covers the following types of failures and connections:

Failure Types

- WAN failures that include WAN links flapping or degraded WAN links
- Cisco Unified CallManager software failure, such as when the ccm.exe service crashes on a Cisco Unified CallManager server.
- LAN connectivity failure, except when a failure occurs at the local branch

Connection Types

- Calls between two Cisco Unified CallManager controlled endpoints
    - During Cisco Unified CallManager reloads
    - When a Transmission Control Protocol (TCP) connection between one or both endpoints and Cisco Unified CallManager used for signaling H.225.0 or H.245 messages is lost or flapping
    - Between endpoints that are registered to different Cisco Unified CallManagers in a cluster and the TCP connection between the two Cisco Unified CallManagers is lost
    - Between IP phones and the PSTN at the same site
- Calls between Cisco IOS gateway and an endpoint controlled by a softswitch where the signaling (H.225.0, H.245 or both) flows between the gateway and the softswitch and media flows between the gateway and the endpoint.
    - When the softswitch reloads.
    - When the H.225.0 or H.245 TCP connection between the gateway and the softswitch is lost, and the softswitch does not clear the call on the endpoint
    - When the H.225.0 or H.245 TCP connection between softswitch and the endpoint is lost, and the soft-switch does not clear the call on the gateway
- Call flows that involve a Cisco IP in IP (IPIP) gateway running in media flow-around mode that reload or lose connection with the rest of the network

Note that after the media is preserved, the call is torn down later when either one of the parties hangs up or media inactivity is detected. In cases where there is a machine-generated media stream, such as music streaming from a media server, the media inactivity detection will not work and the call may hang. Cisco Unified CallManager addresses such conditions by indicating to the gateway that such calls should not be preserved, but third-party devices or IPIP gateways would not do this.

Flapping is defined for this feature as the repeated and temporary loss of IP connectivity that can be caused by WAN or LAN failures. H.323 VoIP calls between a Cisco IOS gateway and Cisco Unified CallManager may be torn down when flapping occurs. When Cisco Unified CallManager detects that the TCP connection is lost, it clears the call and closes the TCP sockets used for the call by sending a TCP FIN, without sending an "H.225.0 Release Complete" or "H.245 End Session" message. This is called quiet clearing. The TCP FIN sent from the Cisco Unified CallManager could reach the gateway if the network comes up for a short duration, and the gateway will tear the call down. Even if the TCP FIN does not reach the gateway, the TCP keepalives sent from the gateway could reach Cisco Unified CallManager when the network comes up. Cisco Unified CallManager will send TCP RST messages in response to the keepalives as it has already closed the TCP connection. The gateway will tear down H.323 calls if it receives the RST message.

Configuration of H.323 VoIP call preservation enhancements for WAN link failures involves configuring the **call preserve** command. If you are using Cisco Unified CallManager you must enable the "Allow Peer to Preserve H.323 Calls" parameter from Cisco Unified CallManager's Service Parameters window.

The **call preserve** command causes the gateway to ignore socket closure or socket errors on H.225.0 or H.245 connections for active calls, allowing the socket to be closed without tearing down calls using those connections.

Call preservation may be reported through Syslog, which optionally can be obtained through a simple network management protocol (SNMP) trap. New syslog messages are printed when call preservation is applied. An SNMP trap can be configured on this syslog message, so you can be notified when call preservation occurs on a gateway.

Preservation information is displayed through the **show h323 calls preserved** command. The following is an example of the command's output:

```
CallID = 11EC , Calling Number = , Called Number = 3210000 ,
RemoteSignallingIPAddress=9.13.0.26 , RemoteSignallingPort=49760 ,
RemoteMediaIPAddress=9.13.0.11 , RemoteMediaPort=17910 , Preserved Duration = 262 , Total
Duration = 562 , H225 FD = -1 , H245 FD = -1
```

The previous example represents one preserved call. One such display is provided per preserved call. The **show h323 calls preserved** displays active calls only. No history is output.

To obtain additional information about a call, you can also use the **show call active voice** command. Calls can be cleared with the **clear call voice causecode** command.

Prerequisites

- This feature may be used on all Cisco Unified CallManager system hardware configurations. If you are not using Cisco Unified CallManager, this feature can only be configured on the Cisco AS5000 Series.

- For bidirectional silence detection, Cisco IOS gateways with 5510 digital signal processors (DSPs) are needed.

- It is recommended that media inactivity detection be configured so that preserved calls are torn down after conversations are over. Two available media inactivity detection features are discussed in the "Configuring Signal ISDN B-Channel ID" section on page 109. They are RTP and RTP Control Protocol (RTCP) inactivity detection and bidirectional silence detection. For more information about media inactivity detection, see the "Configuring Media Inactive Call Detection" chapter in the *Cisco IOS Tcl IVR and VoiceXML Application Guide—12.3(14)T and Later*.

Restrictions

H.323 VoIP Call preservation enhancements for WAN link failures does not support the following:

- Calls in transient call states
- Calls in for which a H.225.0 connection has not occurred
- Calls on which supplementary services are in progress, such as when one of the parties is on hold.
- Calls that involve a media resource located across a WAN, such as conference resources
- Calls where the two parties are registered to different Cisco Unified CallManager clusters
- The "Do Not Preserve" function (using an H.225 Notify message) on networks without Cisco CallManager.

Configuring H.323 Call Preservation Enhancements for WAN Link Failures

The tasks for configuring H.323 VoIP call preservation enhancements for WAN link failures include the following:

- Configuring the Gateway, page 113
- Configuring Cisco Unified CallManager, page 118 (Cisco CallManager Only)

Configuring the Gateway

The **call preserve** command activates H.323 VoIP call preservation. RTP and RTCP inactivity detection and bidirectional silence detection can be used with this feature. Note that voice activity detection (VAD) must be set to off if you are using RTP and RTCP inactivity detection. VAD may be set to on, for bidirectional silence detection. For configuration examples, see the "RTP and RTCP Inactivity Detection Configuration Example" section on page 134 and "Bidirectional Silence Detection Enable Example" section on page 134.

When bidirectional silence and RTP and RTCP inactivity detection are configured, they are enabled for all calls by default. To enable them for H.323 VoIP preserved calls only, you must use the **call preserve** command's **limit-media-detection** keyword.

H.323 VoIP call preservation can be applied to all calls and to dial peers. The required steps are described in the following sections:

- Configuring H.323 VoIP Call Preservation for All Calls, page 113
- Configuring H.323 VoIP Call Preservation for a Dial Peer, page 114

Configuring H.323 VoIP Call Preservation for All Calls

The following describes how to configure H.323 VoIP call preservation for all calls.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **h323**
5. **call preserve** [**limit-media-detection**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Router> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** Router# configure terminal	Enters global configuration mode.
Step 3	**voice service voip** **Example:** Router (config)# voice service voip	Enters voice-service configuration mode.
Step 4	**h323** **Example:** Router (config-voi-serv)# h323	Enables the H.323 voice service configuration commands.
Step 5	**call preserve** [**limit-media-detection**] **Example:** Router (config-voi-h323)# call preserve	Enables the preservation of H.323 VoIP calls. • **limit-media-detection**—Limits RTP and RTCP inactivity detection and bidirectional silence detection (if configured) to H.323 VoIP preserved calls only.
Step 6	**exit** **Example:** Router# exit	Exits H.323 configuration mode.
Step 7	**exit** **Example:** Router# exit	Exist voice service voip configuration mode.

Examples

The following configuration example enables H.323 VoIP call preservation for all calls.

```
voice service voip
 h323
  call preserve
```

The following configuration example enables H.323 VoIP call preservation and limits RTP and RTCP inactivity detection and bidirectional silence detection (if configured) to preserved calls only:

```
voice service voip
 h323
  call preserve limit-media-detection
```

Configuring H.323 VoIP Call Preservation for a Dial Peer

The following describes how to configure H.323 VoIP call preservation for a dial peer.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice-class h323** *tag*
4. **call preserve** [**limit-media-detection**]
5. **exit**
6. **dial-peer voice** *tag* **voip**
7. **voice-class h323** *tag*
8. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice-class h323** *tag* **Example:** `Router (config)# voice-class h323 4`	Assigns an H.323 voice class to a VoIP dial peer. • tag—Unique number to identify the voice class. Range is from 1 to 10000.
Step 4	**call preserve** [**limit-media-detection**] **Example:** `Router (config-class)# call preserve`	Enables the preservation of H.323 VoIP calls. • **limit-media-detection**—Limits RTP and RTCP inactivity detection and bidirectional silence detection (if configured) to H.323 VoIP preserved calls only.
Step 5	**exit** **Example:** `Router (config)# exit`	Exits H.323 voice class configuration mode.
Step 6	**dial-peer voice** *tag* **voip** **Example:** `Router (config)# dial-peer voice 1 voip`	Defines a particular dial peer.

	Command or Action	Purpose
Step 7	**voice-class h323** *tag* **Example:** `Router (config-dial-peer)# voice-class h323 4`	Assigns an H.323 voice class to a VoIP dial peer. • tag—Unique number to identify the voice class. Range is from 1 to 10000.
Step 8	**exit** **Example:** `Router# exit`	Exits dial-peer voice configuration mode.

Examples

The following configuration example enables H.323 VoIP call preservation for dial peer 1.

```
voice-class h323 4
 call preserve
dial-peer voice 1 voip
 voice-class h323 4
```

Troubleshooting Tips

- Enable the **voice iec syslog** command in global configuration mode to display the reason that a call has disconnected after call preservation. The following is an example of the **voice iec syslog** command output line that display this information:

  `Nov 29 12:39:55.167: %**VOICE_IEC-3-GW: H323: Internal Error (Socket error):**`

- Calls on hold are not preserved and a non-standard message with "callPreserveIE FALSE" is sent in the notify message. Use the **debug h225 asn** command for debug. The following is example output:

```
Router# debug h225 asn
H.225 ASN1 Messages debugging is on
3725-GW1#
*May 3 15:57:27.920: H225.0 INCOMING ENCODE BUFFER::=
28501900060008914A00040000D2D6D6D87EB11D02000000090D194410A00100110140B50000120A80A480
04000101000100
*May 3 15:57:27.920:
*May 3 15:57:27.920: H225.0 INCOMING PDU ::=
value H323_UserInformation ::=
{
h323-uu-pdu
{
h323-message-body notify :
{
protocolIdentifier { 0 0 8 2250 0 4 }
callIdentifier
{
guid '00D2D6D6D87EB11D02000000090D1944'H
}
}
h245Tunneling FALSE
nonStandardControl
{
{
nonStandardIdentifier h221NonStandard :
{
t35CountryCode 181
t35Extension 0
manufacturerCode 18
```

```
}
data '80A48004000101000100'H
}
}
}
}
*May 3 15:57:27.924: H225 NONSTD INCOMING ENCODE BUFFER::= 80A48004000101000100
*May 3 15:57:27.924:
*May 3 15:57:27.924: H225 NONSTD INCOMING PDU ::=
value H323_UU_NonStdInfo ::=
{
callMgrParam
{
interclusterVersion 1
enterpriseID {}
}
callPreserveParam
{
callPreserveIE FALSE
}
}
```

When the call is resumed, "callPreserve" is again set to True as shown in the following output
example:

```
Router# debug h225 asn
*May 3 15:57:32.676: H225.0 INCOMING ENCODE BUFFER::=
28501900060008914A00040000D2D6D6D87EB11D02000000090D194410A001001B0140B50000121480A680
04000101000943004C0580323030300140
*May 3 15:57:32.676:
*May 3 15:57:32.676: H225.0 INCOMING PDU ::=
value H323_UserInformation ::=
{
h323-uu-pdu
{
h323-message-body notify :
{
protocolIdentifier { 0 0 8 2250 0 4 }
callIdentifier
{
guid '00D2D6D6D87EB11D02000000090D1944'H
}
}
h245Tunneling FALSE
nonStandardControl
{
{
nonStandardIdentifier h221NonStandard :
{
t35CountryCode 181
t35Extension 0
manufacturerCode 18
}
data '80A68004000101000943004C0580323030300140'H
}
}
}
}
*May 3 15:57:32.680: H225 NONSTD INCOMING ENCODE BUFFER::=
80A68004000101000943004C0580323030300140
*May 3 15:57:32.680:
*May 3 15:57:32.680: H225 NONSTD INCOMING PDU ::=
value H323_UU_NonStdInfo ::=
{
```

```
callMgrParam
{
interclusterVersion 1
enterpriseID {}
}
callSignallingParam
{
connectedNumber '4C058032303030'H
}
callPreserveParam
{
callPreserveIE TRUE
}
}
```

- Use the **debug cch323 all** command after call setup to see if call is going into preserved state. Note that this command generates verbose output, and a console message is printed for every preserved call. In the following output, the relevant information appears in boldface:

```
Router# debug cch323 all
(CCH323-6-CALL_PRESERVED).
Nov 29 12:39:55.167: //-1/xxxxxxxxxxxx/H323/cch323_ct_main: SOCK 3 Event 0x1
Nov 29 12:39:55.167: //31/A9E0FB268017/H323/cch323_h225_handle_conn_loss:
cch323_h225_handle_conn_loss Call not torn down despite H.225.0 socket error: socket
error status = 1, ccb status = 403760899, fd = 3, pre-V3 = 0
Nov 29 12:39:55.167: %CCH323-6-CALL_PRESERVED: cch323_h225_handle_conn_loss: H.323
call preserved due to socket closure or error, Call Id = 4593, fd = 3
Nov 29 12:39:55.167: %VOICE_IEC-3-GW: H323: Internal Error (Socket error):
IEC=1.1.186.5.7.6 on callID 31 GUID=A9E0FB26600B11DA8017000653455072
Nov 29 12:39:55.167: //-1/xxxxxxxxxxxx/H323/h323_set_release_source_for_peer:
ownCallId[31], src[6]
Nov 29 12:39:55.167: //-1/xxxxxxxxxxxx/H323/h323_gw_clean_send_blocked_watch: fd 3
Nov 29 12:39:55.167: //-1/xxxxxxxxxxxx/H323/cch323_cleanup_xport: hashDestroy for
TcpFDTbl
```

- The following are additional debug commands can be used to troubleshoot the problems associated with H.323 VoIP call preservation:

    – **debug h225 asn1**

    – **debug h225 q931**

    – **debug h245 asn1**

Configuring Cisco Unified CallManager

If you are using Cisco Unified CallManager, you must activate H.323 call preservation through the "Allow Peer to Preserve H.323 Calls" parameter, which preserves the following:

- Active H323 calls with quiet clear triggered by the other half of the call
- Active H323 calls with TCP socket closed on the H.323 end before the H.225 or H.245 release signal is received
- Active H323 calls with a signal distribution layer (SDL) link that is out of service and detected on the H323 end

Procedure

Step 1   Choose **Service > Service Parameters**.

Step 2   From the Service menu select Cisco Unified CallManager.

Step 3   Click **Advanced**.

Step 4   Scroll to the Clusterwide Parameter (Device — H.323) section.

Step 5   Set the "Allow Peer to Preserve H.323 Calls" parameter to True.

Step 6   At the top of the screen click **Update**.

Configuration Examples for H.323 Gateways

This section provides the following configuration examples:

RAS: Example

Figure 6 shows a Cisco 2600 and a Cisco AS5800 as gateways and a Cisco 3640 as a gatekeeper.

Figure 6   *VoIP for the Cisco AS5800*
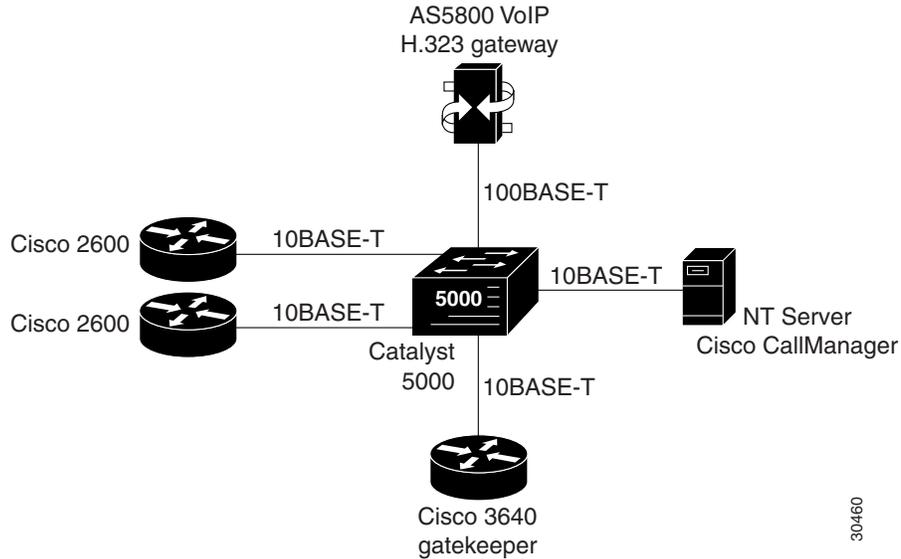
The following example shows a Cisco AS5800 as a gateway using RAS:

```
! Configure the T1 controller. (This configuration is for a T3 card.)
controller T1 1/0/0:1
 framing esf
 linecode b8zs
 pri-group timeslots 1-24
!
! Configure POTS and VoIP dial peers.
dial-peer voice 11111 pots
 incoming called-number 12345
 destination-pattern 9#11111
 direct-inward-dial
 port 1/0/0:1:D
 prefix 11111
!
dial-peer voice 12345 voip
 destination-pattern 12345
 tech-prefix 6#
 session target ras
!
! Enable gateway functionality.
gateway
!
! Enable Cisco Express Forwarding.
ip cef
!
! Configure and enable the gateway interface.
interface FastEthernet0/3/0
 ip address 172.16.0.0.255.255.255.0
 no ip directed-broadcast
 no keepalive
 full-duplex
 no cdp enable
h323-gateway voip interface
 h323-gateway voip id gk3.gg-dn1 ipaddr 172.18.0.0 1719
 h323-gateway voip h323-id gw3@gg-dn1
 h323-gateway voip tech-prefix 9#
!
! Configure the serial interface.(This configuration is for a T3 serial interface.)
```

```
interface Serial1/0/0:1:23
 no ip address
 no ip directed-broadcast
 ip mroute-cache
 isdn switch-type primary-5ess
 isdn incoming-voice modem
 no cdp enable
```

Gateway Security: Example

H.323 Gateway Security

The following example illustrates H.323 security configuration on a Cisco AS5300 gateway.

```
hostname um5300
!
enable password xyz
!
resource-pool disable
!
clock timezone EST -5
clock summer-time EDT recurring
ip subnet-zero
no ip domain-lookup
!
isdn switch-type primary-5ess
isdn voice-call-failure 0
call application voice xyz tftp://172.18.16.2/samp/xyz.tcl
call application voice load xys
mta receive maximum-recipients 1024
!
xgcp snmp sgcp
!
controller T1 0
 framing esf
 clock source line primary
 linecode b8zs
 pri-group timeslots 1-24
!
controller T1 1
 framing esf
 clock source line secondary 1
 linecode b8zs
 pri-group timeslots 1-24
!
controller T1 2
!
controller T1 3
!
voice-port 0:D
!
voice-port 1:D
!
dial-peer voice 4001 pots
 application xyz
 destination-pattern 4003
 port 0:D
 prefix 4001
!
dial-peer voice 513 voip
 destination-pattern 1513200....
 session target ras
```

```
!
dial-peer voice 9002 voip
 destination-pattern 9002
 session target ras
!
dial-peer voice 4191024 pots
 destination-pattern 4192001024
 port 0:D
 prefix 4001
!
dial-peer voice 1513 voip
 destination-pattern 1513.......
 session target ras
!
dial-peer voice 1001 pots
 destination-pattern 14192001001
 port 0:D
!
gateway
 security password 151E0A0E level all
!
 interface Ethernet0
 ip address 10.99.99.7 255.255.255.0
 no ip directed-broadcast
 shutdown
!
interface Serial0:23
 no ip address
 no ip directed-broadcast
 isdn switch-type primary-5ess
 isdn protocol-emulate user
 isdn incoming-voice modem
 fair-queue 64 256 0
 no cdp enable
!
interface Serial1:23
 no ip address
 no ip directed-broadcast
 isdn switch-type primary-5ess
 isdn protocol-emulate user
 isdn incoming-voice modem
 isdn guard-timer 3000
 isdn T203 10000
 fair-queue 64 256 0
 no cdp enable
!
interface FastEthernet0
 ip address 172.18.72.121 255.255.255.192
 no ip directed-broadcast
 duplex auto
 speed auto
 h323-gateway voip interface
 h323-gateway voip id um5300@vgkcisco3 ipaddr 172.18.72.58 1719
 h323-gateway voip h323-id um5300
 h323-gateway voip tech-prefix 1#
!
no ip http server
ip classless
ip route 10.0.0.0 172.18.72.65
!
!
line con 0
 exec-timeout 0 0
 length 0
```

```
 transport input none
line aux 0
line vty 0 4
 password xyz
 login
!
ntp clock-period 17179974
ntp server 172.18.72.124
```

H.235 Gateway Security

The following example shows output from configuring secure registrations from the gatekeeper and identifying which RAS messages the gatekeeper checks to find authentication tokens:

```
dial-peer voice 10 voip
 destination-pattern 4088000
 session target ras
 dtmf-relay h245-alphanumeric
!
gateway
 security password 09404F0B level endpoint
```

The following example shows output from configuring which RAS messages contain gateway-generated tokens:

```
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
radius-server host 10.25.0.0 auth-port 1645 acct-port 1646
radius-server retransmit 3
radius-server deadtime 5
radius-server key lab
radius-server vsa send accounting
!
gatekeeper
 zone local GK1 test.com 10.0.0.3
 zone remote GK2 test2.com 10.0.2.2 1719
 accounting
 security token required-for registration
 no use-proxy GK1 remote-zone GK2 inbound-to terminal
 no use-proxy GK1 remote-zone GK2 inbound-to gateway
 no shutdown
```

Alternate Gatekeeper Support: Example

In the following example, the gateway is configured to have alternate gatekeepers. The primary and secondary gatekeepers are configured with the priority option. The priority range is 1 to 127. The first alternate gatekeeper is configured as priority 120; the second alternate gatekeeper is not configured, so remains at the default setting of 127.

```
interface Ethernet 0/1
 ip address 172.18.193.59 255.255.255.0
 h323-gateway voip interface
 h323-gateway voip id GK1 ipaddr 172.18.193.65 1719 priority 120
 h323-gateway voip id GK2 ipaddr 172.18.193.66 1719
 h323-gateway voip h323-id cisco2
```

DTMF Relay: Example

The following example configures DTMF relay with the **cisco-rtp** keyword when sending DTMF tones to dial peer 103:

```
dial-peer voice 103 voip
 dtmf-relay cisco-rtp
```

The following example configures DTMF relay with the **cisco-rtp** or **h245-signal** keywords when DTMF tones are sent to dial peer 103:

```
dial-peer voice 103 voip
 dtmf-relay cisco-rtp h245-signal
```

The following example configures the gateway to send DTMF in-band (the default) when DTMF tones to are sent dial peer 103:

```
dial-peer voice 103 voip
 no dtmf-relay
```

The following example shows that DTMF relay is configured on an H.323 gateway using NTE RTP and H.245 signaling. In this example, the Named Signaling Event (NSE) value in use is reassigned to a different, unassigned number (110). NTE payload is then assigned to the previously used value (100).

```
dial-peer voice 400 voip
 destination-pattern 400
 dtmf-relay rtp-nte h245-signal
 rtp payload nse 110
 rtp payload-type nte 100
 session target ipv4:172.18.193.181
```

Multiple Codecs: Example

The following configuration shows how to create a list of prioritized codecs and apply that list to a specific VoIP dial peer:

```
 voice class codec 99
  codec preference 1 g711alaw
  codec preference 2 g711ulaw bytes 80
  codec preference 3 g723ar53
  codec preference 4 g723ar63 bytes 144
  codec preference 5 g723r53
  codec preference 6 g723r63 bytes 120
  codec preference 7 g726r16
  codec preference 8 g726r24
  codec preference 9 g726r32 bytes 80
  codec preference 10 g728
  codec preference 11 g729br8
  codec preference 12 g729r8 bytes 50
!
dial-peer voice 1919 voip
  voice-class codec 99
```

Rotary Calling Pattern: Example

The following example configures POTS dial peer 10 for a preference of 1, POTS dial peer 20 for a preference of 2, and Voice over Frame Relay dial peer 30 for a preference of 3:

```
dial-peer voice 10 pots
```

```
destination pattern 5552150
preference 1

dial-peer voice 20 pots
destination pattern 5552150
preference 2

dial-peer voice 30 vofr
destination pattern 5552150
preference 3
```

H.323 Support for Virtual Interfaces: Example

In the following example, Ethernet interface 0/0 is used as the gateway interface. For convenience, the **h323-gateway voip bind srcaddr** command is specified on the same interface. The designated source IP address is the same as the IP address assigned to the interface.

```
interface Ethernet0/0
 ip address 172.18.194.50 255.255.255.0
 no ip directed-broadcast
 h323-gateway voip interface
 h323-gateway voip id j70f_2600_gk2 ipaddr 172.18.194.53 1719
 h323-gateway voip h323-id j70f_3640_gw1
 h323-gateway voip tech-prefix 3#
 h323-gateway voip bind srcaddr 172.18.194.50
```

H.225 Annex-G: Example

The following example shows the gatekeeper border element router with service relationship and usage-reporting functionality turned on:

```
Router# show running config

Building configuration...
.
.
.
call-router h323-annexg boston1
 neighbor 1.2.3.4
  service-relationship
   outbound retry interval 120
   inbound ttl 1200
   no shutdown
  usage-indication
   retry interval 600
   retry window 3600
 domain-name Celine-Domain
 access-policy neighbors-only
 local ip 172.18.193.31
 no shutdown
.
.
.
```

GTD Payload: Examples

GTD Payload System-Wide

The following example shows the GTD feature configured on the system:

```
Router# show running-config

Building configuration...

Current configuration : 4192 bytes
!
version 12.2
service config
no service single-slot-reload-enable
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service internal
service udp-small-servers
!
hostname as5300-2
!
voice service voip
 signaling forward unconditional
 h323
!
.
.
.
```

GTD Payload on a Dial Peer

The following example shows GTD configured with unconditional forwarding on two dial peers:

```
Router# show running-config

Building configuration...

Current configuration : 4192 bytes
!
version 12.2
service config
no service single-slot-reload-enable
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service internal
service udp-small-servers
!
hostname as5300-2
!
.
.
.
!
dial-peer voice 1 pots
 application session
 incoming called-number 25164
 port 0:D
!
dial-peer voice 1513 voip
```

```
 destination-pattern 1513.......
 session target ipv4:1.8.156.3
!
dial-peer voice 1408525 voip
 destination-pattern 1408525....
!
dial-peer voice 1800877 voip
 destination-pattern 1800877....
 session target ipv4:1.8.156.3
!
dial-peer voice 2 pots
 destination-pattern 51550
 no digit-strip
 direct-inward-dial
 port 3:D
!
dial-peer voice 51557 voip
 destination-pattern 51557
 signaling forward unconditional
 session target ras
!
dial-peer voice 52557 voip
 destination-pattern 52557
 signaling forward unconditional
 session target ipv4:1.8.156.3
!
gateway
!
.
.
```

H.323v4 Gateway Zone Prefix Registration Enhancements: Examples

Verifying Gateway Advertisement of Dynamic Zone Prefixes

The following example displays the status of the destination pattern database and the status of the individual destination patterns for Gatekeeper1:

```
Gateway1# 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
```

Verifying Gatekeeper Processing of Additive RRQs Example

The following example displays the zone prefix table, including the dynamic zone prefixes, for Gatekeeper1:

```
Gatekeeper1# show gatekeeper zone prefix all

                    ZONE PREFIX TABLE
        =============================================
```

```
GK-NAME            E164-PREFIX      Dynamic GW-priority
-------            -----------      -------------------
gatekeeper1        1110507*         gateway2 /5
gatekeeper2        1110508*
gatekeeper1        1110509*         gateway1 /5
gatekeeper1        1110511*         gateway1 /5
gatekeeper1        23*              gateway1 /5
gatekeeper1        4666002*
gatekeeper3        55530..
gatekeeper1        7779...
```

Verifying Dynamic Zone Prefix Registration based on Gateway Priority Lists Example

The following example displays the gateway destination-pattern database status:

Router# **show h323 gateway prefixes**

```
GK Supports Additive RRQ  :True
GW Additive RRQ Support   :True
Pattern Database          :Active

Destination Active
Pattern                    Status          Dial-Peers    Priority
================================================================
1110509*                   ADD ACKNOWLEDGED    2             8
1110511*                   ADD ACKNOWLEDGED    2
23*                        ADD ACKNOWLEDGED    2             4
```

Troubleshooting H.323v4 Gateway Zone Prefix Registration Enhancements Example

The following example displays the ASN1 contents of RAS messages sent during the registration process:

Gatekeeper1# **debug h225 asn1**

```
U.S. Eastern time (GMT -5/-4)
voice:(919) 392-6007.Feb 5 16:27:05.894:RAS INCOMING ENCODE BUFFER::= 00 A0004306 0008914A
00040001 07072ACC 3D2800B5 00001240 0238500A 00320036 00300030 002D0031 02400500 33003600
34003000 2D003101 00C4C0
.Feb 5 16:27:05.906:
.Feb 5 16:27:05.906:RAS INCOMING PDU ::=

value RasMessage ::= gatekeeperRequest :
    {
      requestSeqNum 68
      protocolIdentifier { 0 0 8 2250 0 4 }
      rasAddress ipAddress :
      {
        ip '0107072A'H
        port 52285
      }
      endpointType
      {
        vendor
        {
          vendor
          {
            t35CountryCode 181
            t35Extension 0
            manufacturerCode 18
          }
        }
        gateway
```

```
                {
                  protocol
                  {
                    voice :
                    {
                    },            h323 :
                    {
                    }
                  }
                }
                mc FALSE
                undefinedNode FALSE
              }
              gatekeeperIdentifier {"2600-1"}
              endpointAlias
              {
                h323-ID :{"3640-1"},
                dialedDigits :"919"
              }
          }

.Feb 5 16:27:05.926:RAS OUTGOING PDU ::=

value RasMessage ::= gatekeeperConfirm :
      {
        requestSeqNum 68
        protocolIdentifier { 0 0 8 2250 0 4 }
        gatekeeperIdentifier {"2600-1"}
        rasAddress ipAddress :
        {
          ip '01070721'H
          port 1719
        }
      }

.Feb 5 16:27:05.934:RAS OUTGOING ENCODE BUFFER::= 04 80004306 0008914A 00040A00 32003600
30003000 2D003100 01070721 06B7
.Feb 5 16:27:05.938:
.Feb 5 16:27:05.946:RAS INCOMING ENCODE BUFFER::= 0E C0004406 0008914A 00048001 00010707
2A06B801 00010707 2ACC3D28 00B50000 12400238 50024005 00330036 00340030 002D0031 0100C4C0
A0003200 36003000 30002D00 3100B500 0012288B 08000200 3B010001 00018002 7000
.Feb 5 16:27:05.958:
.Feb 5 16:27:05.958:RAS INCOMING PDU ::=

value RasMessage ::= registrationRequest :
      {
        requestSeqNum 69
        protocolIdentifier { 0 0 8 2250 0 4 }
        discoveryComplete TRUE
        callSignalAddress
        {
          ipAddress :
          {
            ip '0107072A'H
            port 1720
          }
        }
        rasAddress
        {
          ipAddress :
          {
            ip '0107072A'H
            port 52285
          }
```

```
      }
      terminalType
      {
        vendor
        {
          vendor
          {
            t35CountryCode 181
            t35Extension 0
            manufacturerCode 18
          }
        }
        gateway
        {
          protocol
          {
            voice :
            {
            },              h323 :
            {
            }
          }
        }
        mc FALSE
        undefinedNode FALSE
      }
      terminalAlias
      {
        h323-ID :{"3640-1"},
        dialedDigits :"919"
      }
      gatekeeperIdentifier {"2600-1"}
      endpointVendor
      {
        vendor
        {
          t35CountryCode 181
          t35Extension 0
          manufacturerCode 18
        }
      }
      timeToLive 60
      keepAlive FALSE
      willSupplyUUIEs FALSE
      maintainConnection TRUE
      usageReportingCapability
      {
        nonStandardUsageTypes
        {
        }
        startTime NULL
        endTime NULL
        terminationCause NULL
      }
    }

.Feb 5 16:27:05.998:RAS OUTGOING PDU ::=

value RasMessage ::= registrationConfirm :
    {
      requestSeqNum 69
      protocolIdentifier { 0 0 8 2250 0 4 }
      callSignalAddress
      {
```

```
        }
        terminalAlias
        {
          h323-ID :{"3640-1"},
          dialedDigits :"919"
        }
        gatekeeperIdentifier {"2600-1"}
        endpointIdentifier {"816F7A1000000001"}
        alternateGatekeeper
        {
        }
        timeToLive 60
        willRespondToIRR FALSE
        maintainConnection TRUE
        supportsAdditiveRegistration NULL
        usageSpec
        {

          {
            when
            {
              end NULL
              inIrr NULL
            }
            callStartingPoint
            {
              connect NULL
            }
            required
            {
              nonStandardUsageTypes
              {
              }
              startTime NULL
              endTime NULL
              terminationCause NULL
            }
          }
        }
      }
```

Signal ISDN B-Channel ID: Example

The following example shows an H.323 and SIP ISDN B-channel configuration example.

```
Current configuration : 3394 bytes
!
version 12.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service internal
!
memory-size iomem 15
ip subnet-zero
!
!
no ip domain lookup
!
voice service voip
 h323
  billing b-channel
 sip
```

```
  ds0-num

ip dhcp pool vespa
 network 192.168.0.0 255.255.255.0
 option 150 ip 192.168.0.1
 default-router 192.168.0.1
!
!
voice call carrier capacity active
!
voice class codec 1
 codec preference 2 g711ulaw
!
!
no voice hpi capture buffer
no voice hpi capture destination
!
!
fax interface-type fax-mail
mta receive maximum-recipients 0
!
!
interface Ethernet0/0
 ip address 10.8.17.22 255.255.0.0
 half-duplex
!
interface FastEthernet0/0
 ip address 192.168.0.1 255.255.255.0
 speed auto
 no cdp enable
 h323-gateway voip interface
 h323-gateway voip id vespa2 ipaddr 10.8.15.4 1718
!
router rip
 network 10.0.0.0
 network 192.168.0.0
!
ip default-gateway 10.8.0.1
ip classless
ip route 0.0.0.0 0.0.0.0 10.8.0.1
no ip http server
ip pim bidir-enable
!
!
tftp-server flash:SEPDEFAULT.cnf
tftp-server flash:P005B302.bin
call fallback active
!
!
call application global default.new
call rsvp-sync
!
voice-port 1/0
!
voice-port 1/
!
mgcp profile default
!
!
dial-peer voice 1 pots
 destination-pattern 5100
 port 1/0
!
dial-peer voice 2 pots
```

```
 destination-pattern 9998
 port 1/1
!
dial-peer voice 123 voip
 destination-pattern [12]...
 session protocol sipv2
 session target ipv4:10.8.17.42
 dtmf-relay sip-notify
!
gateway
!
sip-ua
 retry invite 3
 retry register 3
 timers register 150
 registrar dns:myhost3.cisco.com expires 3600
 registrar ipv4:10.8.17.40 expires 3600 secondary
!
!
telephony-service
 max-dn 10
 max-conferences 4
!
ephone-dn 1
number 4001
!
ephone-dn 2
number 4002
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
login
line vty 5 15
 login
!
no scheduler allocate
end
```

H.323 VoIP Call Preservation Enhancements for WAN Link Failures Examples

This section includes the following configuration examples:

H.323 VoIP Call Preservation for All Calls Example

The following configuration example enables H.323 VoIP call preservation for all calls:

```
voice service voip
 h323
  call preserve
```

H.323 VoIP Call Preservation for a Dial Peer Example

The following configuration example enables H.323 VoIP call preservation for one dial peer:

```
voice class h323 4
 call preserve

dial-peer voice 1
 voice class h323 4
```

H.323 Call Preservation for RTP and RTCP and Silence Detection Example

The following configuration example enables H.323 VoIP call preservation and limits RTP and RTCP inactivity detection and bidirectional silence detection (if configured) to H.323 VoIP preserved calls only:

```
voice service voip
 h323
  call preserve limit-media-detection
```

RTP and RTCP Inactivity Detection Configuration Example

The following configuration example can be used to enable RTP and RTCP inactivity detection for dial peers. Note that for call preservation VAD must be set to off (**no vad** command):

```
dial-peer voice 10 voip
 no vad
gateway
 timer receive-rtcp 4
ip rtcp report-interval 60
```

Bidirectional Silence Detection Enable Example

The following configuration example enables bidirectional silence detection:

```
gateway
 timer media-inactive 5
ip rtcp report interval
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS Release 12.4	• Cisco IOS Release 12.4 Configuration Guides • Cisco IOS Release 12.4T Configuration Guides • Cisco IOS Release 12.4 Command References • Cisco IOS Voice Configuration Library http://www.cisco.com/en/US/docs/ios/12_3/vvf_c/cisco_ios_voice_configuration_library_glossary/vcl.htm ✎ **Note** This website contains the library preface and glossary.
Cisco IOS Release 12.3	• Cisco IOS Release 12.3 documentation • Cisco IOS voice commands • Cisco IOS Voice Troubleshooting and Monitoring Guide • Tcl IVR Version 2.0 Programming Guide
Cisco IOS Release 12.2	• *Cisco IOS Voice, Video, and Fax Configuration Guide*, Release 12.2 at http://www.cisco.com/en/US/docs/ios/12_2/voice/configuration/guide/fvvfax_c.html
Troubleshooting and Debugging guides	• Cisco IOS Debug Command Reference, Release 12.4 at http://www.cisco.com/en/US/docs/ios/debug/command/reference/db_book.html • *Troubleshooting and Debugging VoIP Call Basics* at http://www.cisco.com/en/US/tech/tk1077/technologies_tech_note09186a0080094045.shtml • *VoIP Debug Commands* at http://www.cisco.com/en/US/docs/routers/access/1700/1750/software/configuration/guide/debug.html

Related Topic	Document Title
Cisco Unified Border Element Configuration Examples	• Local-to-remote network using the IPIPGW http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a00801b0803.shtml • Remote-to-local network using the IPIPGW: http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080203edc.shtml • Remote-to-remote network using the IPIPGW: http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080203edd.shtml • Remote-to-remote network using two IPIPGWs: http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080203edb.shtml
Related Application Guides	• Cisco Unified Communications Manager and Cisco IOS Interoperability Guide • Cisco IOS Fax, Modem, and Text Support over IP Configuration Guide • "Configuring T.38 Fax Relay" chapter • Cisco IOS SIP Configuration Guide • Cisco Unified Communications Manager (CallManager) Programming Guides at: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_programming_reference_guides_list.html • *Quality of Service for Voice over IP* at http://www.cisco.com/en/US/docs/ios/qos/configuration/guide/15_0/qos_15_0_book.html

Standards

Standards	Title
ITU-T E.164	Overall network operation, telephone service, service operation and human factors
ITU-T H.225 Version 2	Call signalling protocols and media stream packetization for packet-based multimedia communication systems
ITU-T H.235	Security and encryption for H-Series (H.323 and other H.245-based) multimedia terminals
ITU-T H.323	Packet-based multimedia communications systems
ITU-T H.450	Supplementary services for multimedia

MIBs

MIBs	MIBs Link
• CISCO-GATEKEEPER-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 2833	*RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals*

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport

Configuring H.323 Gatekeepers and Proxies

This chapter describes how to configure Cisco H.323 gatekeepers. It also presents information about gatekeeper features that are not configurable.

Feature History for Call Status Tracking Optimization

Release	Modification
12.2(11)T	This feature was introduced.

Feature History for Configuring a Gatekeeper to Provide Nonavailability Information for Terminating Endpoints

Release	Modification
12.2(11)T	This feature was introduced.
12.3(8)T1	The carrier based routing without the presence of the GKTMP application server was introduced.
12.3(11)T	The carrier based routing without the presence of the GKTMP application server was implemented in this release.

Feature History for Gatekeeper Alias Registration and Address Resolution Enhancements

Release	Modification
12.2(11)T	This feature was introduced.

Feature History for Gatekeeper Endpoint Control Enhancements

Release	Modification
12.2(11)T	This feature was introduced.

Feature History for Gatekeeper Enhancements for Managed Voice Services

Release	Modification
12.3(1)	This feature was introduced.

Feature History for Gatekeeper-to-Gatekeeper Authentication

Release	Modification
12.2(11)T	This feature was introduced.
12.2(11)T2	The encrypted keyword was added to the **security password-group** command.

Feature History for Gatekeeper Transaction Message Protocol Interface Resiliency Enhancement

Release	Modification
12.2(2)XB	This feature was introduced.
12.2(8)T	This feature was integrated into this release. The Cisco 2500 series is not supported in this release.

Feature History for H.323 Version 2 Enhancements

Release	Modification
12.0(5)T	This feature was introduced.
12.1(5)XM2	Support was added for the Cisco AS5350 and Cisco AS5400.
12.2(2)XA	The **call rscmon update-timer** command was added.
12.2(4)T	The **call rscmon update-timer** command was integrated into this release. Support for the Cisco AS5300, Cisco AS5350, and Cisco AS5400 is not included.
12.2(2)XB1	This feature was implemented on the Cisco AS5850.
12.2(11)T	This features was integrated into this release.

Feature History for High Performance Gatekeeper

Release	Modification
12.1(5)XM	This feature was introduced.
12.2(2)T	This feature was integrated into this release.

Feature History for Inter-Domain Gatekeeper Security Enhancement

Release	Modification
12.2(2)XA	This feature was introduced.
12.2(4)T	This feature was integrated into this release.
12.2(2)XB1	This feature was implemented on the Cisco AS5850.
12.2(11)T	This feature was implemented on the Cisco AS5300 and Cisco AS5850 and integrated into this release.

Feature History for NAT Support of H.323 v2 RAS

Release	Modification
12.2(2)T	This feature was introduced.

Feature History for Sequential Location Request Enhancement

Release	Modification
12.2(4)T	This feature was introduced.

Feature History for Tokenless Call Authorization

Release	Modification
12.2(15)T	This feature was introduced.

Feature History for VoIP Outgoing Trunk Group ID and Carrier ID for Gateways and Gatekeepers

Release	Modification
12.2(11)T	This feature was introduced, and the **carrier-id** keyword and *carrier-name* argument were introduced for the **endpoint alt-ep h323id** command.

Finding Support Information for Platforms and Cisco IOS Software Images

Use Cisco Feature Navigator to find information about platform support and Cisco IOS software image support. Access Cisco Feature Navigator at http://www.cisco.com/go/fn. You must have an account on Cisco.com. If you do not have an account or have forgotten your username or password, click **Cancel** at the login dialog box and follow the instructions that appear.

Note For more information about these and other related Cisco IOS voice features, see the following:

- "H.323 Overview" section on page 7

- For information about the full set of Cisco IOS voice features, see the entire Cisco IOS Voice Configuration Library—including library preface, glossary, and other documents—at http://www.cisco.com/en/US/docs/ios/12_3/vvf_c/cisco_ios_voice_configuration_library_glossary /vcl.htm

Contents

Note For complete descriptions of the commands used in this chapter, see the command references listed in the "Additional References" section on page 256.

Prerequisites for Configuring H.323 Gatekeepers and Proxies

- Perform the prerequisites that are listed in the "Prerequisites for Configuring an H.323 Network" section on page 7.

- Install Cisco IOS Release 12.3 or later on your gatekeeper.

Restrictions for Configuring H.323 Gatekeepers and Proxies

Restrictions are described in the "Restrictions for Configuring an H.323 Network" section on page 8

How to Configure H.323 Gatekeepers and Proxies

This section contains the following information:

Configuring Hot Standby

Cisco routers support Hot Standby Router Protocol (HSRP), which allows one router to serve as a backup to another router. Cisco gatekeepers can be configured to use HSRP so that when one gatekeeper fails, the standby gatekeeper assumes its role.

To configure a gatekeeper to use HSRP, perform the following tasks.

Step 1   Select one interface on each gatekeeper to serve as the HSRP interface and configure these two interfaces so that they belong to the same HSRP group but have different priorities. The one with the higher priority becomes the active gatekeeper; the other assumes the standby role. Make a note of the virtual HSRP IP address shared by both of these interfaces.

> **Note**   For more information on HSRP and HSRP configuration, see the *Configuring HSRP* section of the *Cisco IOS IP Application Services Configuration Guide* at http://www.cisco.com/en/US/docs/ios/ipapp/configuration/guide/ipapp_hsrp_ps6441_TSD_Products_Configuration_Guide_Chapter.html

Step 2   Configure the gatekeepers so that the HSRP virtual IP address is the RAS address for all local zones.

Step 3   Make sure that the gatekeeper-mode configurations on both routers are identical.

Step 4   If the endpoints and gateways are configured so that they use a specific gatekeeper address (rather than multicasting), use the HSRP virtual IP address as the gatekeeper address. You can also let the endpoints and gateways find the gatekeeper by multicasting. As long as it is on standby status, the secondary gatekeeper neither receives nor responds to multicast or unicast requests.

As long as both gatekeepers are up, the one with the higher priority on its HSRP interface is the active gatekeeper. If this active gatekeeper fails, or if its HSRP interface fails, the standby HSRP interface assumes the virtual HSRP address and, with it, the active gatekeeper role. When the gatekeeper with the higher HSRP priority comes back online, it reclaims the HSRP virtual address and the gatekeeper function, while the secondary gatekeeper goes back to standby status.

> **Note**   Gatekeeper failover is not completely transparent to endpoints and gatekeepers. When the standby gatekeeper takes over, it does not have the state of the failed gatekeeper. If an endpoint that had registered with the failed gatekeeper now makes a request to the new gatekeeper, the gatekeeper responds with a reject, indicating that it does not recognize the endpoint. The endpoint must reregister with the new gatekeeper before it can continue H.323 operations.

Configuring Gatekeeper Zones

This section contains the following information:

Restrictions for Gatekeeper Zones

- The gateway can register with only one gatekeeper at a time.
- Only E.164 address resolution is supported.
- Because the gateway can register with only one gatekeeper at a time, redundant H.323 zone support provides only redundancy and does not provide any load balancing.

- Although redundant H.323 zone support allows you to configure alternate gatekeepers, it does not insert information in the alternate gatekeeper field of some RAS messages.

Information About Gatekeeper Zones

Zone and Subnet Configuration

A zone is defined as the set of H.323 nodes controlled by a single gatekeeper. Gatekeepers that coexist on a network may be configured so that they register endpoints from different subnets.

Endpoints attempt to discover a gatekeeper and consequently the zone of which they are members by using the Registration, Admission, and Status (RAS) message protocol. The protocol supports a discovery message that may be sent multicast or unicast.

If the message is sent multicast, the endpoint registers nondeterministically with the first gatekeeper that responds to the message. To enforce predictable behavior, where endpoints on certain subnets are assigned to specific gatekeepers, the **zone subnet** command can be used to define the subnets that constitute a given gatekeeper zone. Any endpoint on a subnet that is not enabled for the gatekeeper is not accepted as a member of that gatekeeper zone. If the gatekeeper receives a discovery message from such an endpoint, it sends an explicit reject message.

Gateway Selection Process

Cisco H.323 Version 2 software improves the gateway selection process as follows:

- When more than one gateway is registered in a zone, the updated **zone prefix** command allows selection priorities to be assigned to these gateways on the basis of the dialed prefix.

- Gateway resource reporting allows the gateway to notify the gatekeeper when H.323 resources are getting low. The gatekeeper uses this information to determine which gateway to use to complete a call.

The gatekeeper maintains a separate gateway list, ordered by priority, for each of its zone prefixes. If a gateway does not have an assigned priority for a zone prefix, it defaults to priority 5, which is the median. To explicitly bar the use of a gateway for a zone prefix, the gateway must be defined as having a priority 0 for that zone prefix.

When selecting gateways, the gatekeeper identifies a target pool of gateways by performing a longest zone prefix match; then it selects from the target pool according to priorities and resource availability. If all high-priority gateways are busy, a low-priority gateway might be selected.

Redundant H.323 Zone Support

Redundant H.323 zone support allows for the following:

Gatekeeper Multiple Zone Support

Redundant H.323 zone support allows users to configure multiple remote zones to service the same *zone* or *technology prefix*. A user is able to configure more than one remote gatekeeper to which the local gatekeeper can send location requests (LRQs). This allows for more reliable call completion.

Redundant H.323 zone support is supported on all gatekeeper-enabled IOS images.

Zone Prefixes

The zone prefixes (typically area codes) serve the same purpose as the domain names in the H.323-ID address space.

For example, the local gatekeeper can be configured with the knowledge that zone prefix "212......" (that is, any address beginning "212" and followed by 7 arbitrary digits) is handled by the gatekeeper gatekeeper_2. Then, when the local gatekeeper is asked to admit a call to destination address 2125551111, it knows to send the LRQ to gatekeeper_2.

When gatekeeper_2 receives the request, the gatekeeper must resolve the address so that the call can be sent to its final destination. There may be an H.323 endpoint with that E.164 address that has registered with gatekeeper_2, in which case gatekeeper_2 returns the IP address for that endpoint. However, it is possible that the E.164 address belongs to a non-H.323 device (for example, a telephone or an H.320 terminal). Because non-H.323 devices do not register with gatekeepers, gatekeeper_2 cannot resolve the address. The gatekeeper must be able to select a gateway that can be used to reach the non-H.323 device. This is where the technology prefixes (or "gateway-type") become useful.

Technology Prefixes

The network administrator selects technology prefixes (tech-prefixes) to denote different types or classes of gateways. The gateways are then configured to register with their gatekeepers with these prefixes. For example, voice gateways can register with tech-prefix 1#, H.320 gateways with tech-prefix 2#, and voicemail gateways with tech-prefix 3#. More than one gateway can register with the same type prefix. When this happens, the gatekeeper makes a random selection among gateways of the same type.

If the callers know the type of device that they are trying to reach, they can include the technology prefix in the destination address to indicate the type of gateway to use to get to the destination. For example, if a caller knows that address 2125551111 belongs to a regular telephone, the destination address of 1#2125551111 can be used, where 1# indicates that the address should be resolved by a voice gateway. When the voice gateway receives the call for 1#2125551111, it strips off the technology prefix and bridges the next leg of the call to the telephone at 2125551111.

Configuring Gatekeeper Zones

To configure gatekeeper zones, use the following commands starting in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **zone local** *zonename domainname [ras-ip-address] [port]*
3. **zone remote** *zonename domainname ip-address* [*port*] [**cost** *cost* [**priority** *priority*]]
4. **zone prefix** *gatekeeper-name e164-prefix* [**blast** | **seq**] [**gw-priority** *priority gw-alias* [*gw-alias*, ...]]
5. **use-proxy** *local-zone* **remote-zone** *zone-name* **outbound-from gateway**
6. **zone subnet** *local-gatekeeper-name [**default** | *subnet-address* {*/bits-in-mask* | *mask*} **enable]**
7. Repeat Step 6 for each subnet.
8. **no shutdown**
9. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`zone local` *zone-name domain-name* `[ras-ip-address] [port]`
	
	Example:
	`Router(config-gk)# zone local gk408or650 xyz.com`
		Specifies a zone controlled by a gatekeeper. Arguments are as follows:
		
		• *zone-name*—Gatekeeper name or zone name. This is usually the fully domain-qualified host name of the gatekeeper. For example, if the domain name is cisco.com, the gatekeeper name might be gk1.cisco.com. However, if the gatekeeper is controlling multiple zones, the zone name for each zone should be some unique string that has a mnemonic value.
		
		• *domain-name*—Domain name served by this gatekeeper.
		
		• *ras-ip-address*—IP address of one of the interfaces on the gatekeeper. When the gatekeeper responds to gatekeeper discovery messages, it signals the endpoint or gateway to use this address in future communications. Setting this address for one local zone makes it the address used for all local zones.
		
		• *port*—RAS signaling port number for the local zone. Range: 1 to 65535. Default: 1719.

Step 3	`zone remote` *zone-name domain-name ip-address* `[port] [cost` *cost* `[priority` *priority*`]]`
	
	Example:
	`Router(config-gk)# zone remote zone1 domain 192.168.0.0 123 cost 25 priority 25`
		Defines the remote zone cluster. Keywords and arguments are as follows:
		
		• *zone-name*—ID of the remote zone.
		
		• *domain-name*—ID of the domain the remote zone is serving.
		
		• *ip-address*—IP address for the remote gatekeeper.
		
		• *port*—RAS signaling port number for the remote zone. Range: 1 to 65535. Default: the well-known RAS port number 1719.
		
		• **cost** *cost*—Cost of the zone. Range: 1 to 100. Default: 50.
		
		• **priority** *priority*—Priority of the zone. Range: 1 to 100. Default: 50.
		
		When several remote zones are configured, you can rank them by cost and priority value. A zone with a lower cost value and a higher priority value is given preference over others.

	Command	Purpose
Step 4	**zone prefix** *gatekeeper-name e164-prefix* [**blast** | **seq**] [**gw-priority** *priority gw-alias* [*gw-alias*, ...]] **Example:** `Router(config-gk)# zone prefix gatekeeper1 888 blast`	Adds a prefix to the gatekeeper zone list. For redundant H.323 zone support, you can configure multiple remote gatekeepers for the same prefix, but only one of the gatekeepers defined for any given zone prefix can be local. It is recommended that you limit the number of remote gatekeepers that serve the same zone prefix to two. By default, LRQs are sent sequentially to the remote gatekeepers. With sequential, LRQs are sent one at a time with a delay between them. With blast, LRQs are sent back-to-back in rapid sequence without delay. If you want to specify blast for each gatekeeper, you need to specify **blast** on only one **zone prefix** command per E.164 prefix. The order in which zone and technology prefixes are configured determines the order in which LRQs are sent to the remote gatekeepers. Using zone prefixes as an example, the local gatekeeper routes a call to the first zone that responds with an LCF. If the local gatekeeper is configured for a zone prefix that already has remote gatekeepers configured, the local gatekeeper automatically puts that zone prefix at the top of the list.
Step 5	**use-proxy** *local-zone* **remote-zone** *zone-name* **outbound-from gateway** **Example:** `Router(config-gk)# use-proxy zone123 remote-zone remote456 outbound-from gateway`	Specifies that all calls originating from gateways in the local zone and bound to the remote zone route through a proxy, which should be registered with the gatekeeper. Keywords and arguments are as follows: • *local-zone*—Local zone. • **remote-zone** *zone-name*—Proxy policy for calls to or from a specific gatekeeper or zone. • **outbound-from**—Proxy policy as it applies to calls that are outbound from the local zone to a remote zone. Each use-proxy command defines the policy for only one direction. • **gateway**—Type of local device to which the policy applies. Applies the policy only to local gateways.

	Command	Purpose
Step 6	**zone subnet** *local-gatekeeper-name* [**default** | *subnet-address* {**/***bits-in-mask* | *mask*} **enable]** **Example:** Router(config-gk)# zone subnet gatekeeper3 default	Defines a set of subnets that constitute the gatekeeper zone. Enables the gatekeeper for each of these subnets and disables it for all other subnets. (Repeat for each subnet.) Keywords and arguments are as follows: • *local-gatekeeper-name*—Name of the local gatekeeper. Should be a Domain Name System (DNS) host name if you use DNS to locate remote zones. • **default**—Applies to all other subnets that are not specifically defined by this command. • *subnet-address*—Address of the subnet that is being defined. • */bits-in-mask*—Number of bits of the mask to be applied to the subnet address. You must enter a slash before this argument. • *mask*—Mask (in dotted string format) to be applied to the subnet address. • **enable**—Gatekeeper accepts discovery and registration from the specified subnets. To define the zone as being all but one set of subnets by disabling that set and enabling all other subnets, use the **no** form of the command as follows: Configure **no zone subnet** *local-gatekeeper-name subnet-address* {*/bits-in-mask* | *mask*} **enable**. To accept the default behavior, which is that all subnets are enabled, use the **no** form of the command as follows: **no zone subnet** *local-gatekeeper-name* **default enable**. You can use this command more than once to create a list of subnets controlled by a gatekeeper. The subnet masks need not match actual subnets in use at your site. For example, to specify a particular endpoint, show its address as a 32-bit netmask. If a local gatekeeper name is contained in the message, it must match the *local-gatekeeper-name* argument. **Note** To explicitly enable or disable a particular endpoint, specify its host address using a 32-bit subnet mask.
Step 7	Repeat Step 6 for each subnet.	—
Step 8	**no shutdown** **Example:** Router(config-gk)# no shutdown	Brings the gatekeeper online.
Step 9	**exit** **Example:** Router(config-gk)# exit	Exits the current mode.

Configuring Destination Zones

When a gatekeeper receives an admission request (ARQ) message from a local zone gateway, if bandwidth management is enabled in the gatekeeper checks the bandwidth. The gatekeeper sends an admission reject (ARJ) message to the local zone gateway if the local zone is out of bandwidth and if no remote zone has been configured. The ARJ reject reason is set to "resource unavailable."

If a remote zone has been configured, the gatekeeper sends a location request (LRQ) message to that zone. A check is made of the total, interzone, and session bandwidth limits of the remote zone. If no remote zone has been defined or if the gatekeeper receives location request reject (LRJ) messages from all the remote gateways to which it has sent LRQ messages, the gatekeeper sends an ARJ message (with the reject reason set to "resource unavailable") to the requesting gateway.

> **Note** The ARJ message functionality is available for only tech and zone prefix routing. By default, this functionality is not enabled.

In addition to the gatekeeper maintaining concurrent call counts per zone, after receiving ARQ and LRQ messages from a requesting gateway, the gatekeeper can also check the concurrent call count of the destination zone. If the call count exceeds a preconfigured maximum threshold and if no other remote zone has been configured, the gatekeeper sends an ARJ or LRJ message to the requesting gateway. The ARJ reject reason is shown as "resource unavailable" and the LRJ reject reason is shown as "undefined reason."

If a remote zone has been configured, the gatekeeper sends LRQ messages to the remote zones. If no remote zone has been defined or if the gatekeeper receives LRJ messages from all the remote gateways to which it has sent LRQ messages, the gatekeeper sends an ARJ message (with the reject reason set to "resource unavailable") and an LRJ message (with the reject reason set to "undefined reason") to the requesting gateway.

After receiving an ARQ message from a requesting gateway and if the destination is a local zone, the gatekeeper sends an ACF message to the requesting gateway only if the local destination gateway has resources.

If the local destination gateway is out of resources, the gatekeeper tries to send an LRQ message to remote destination zones until it receives a location confirmation (LCF) message or until no remote zones remain. If no remote zone has been defined or if the gatekeeper receives LRJ messages from remote destinations for all the LRQ messages sent, the gatekeeper sends an ARJ message to the requesting gateway. The reject reason in the ARJ message to the requesting gateway is set to "resource unavailable."

To configure session bandwidth limits of the destination zones and how the gateway should handle requests if resources run low, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **bandwidth check-destination**
3. **arq reject-resource-low**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`bandwidth check-destination`
	
	Example:
	`Router config-gk)# bandwidth check-destination`
		Specifies the maximum aggregate bandwidth for H.323 traffic and enables destination bandwidth checking.

Step 3	`arq reject-resource-low`
	
	Example:
	`Router(config-gk)# arq reject-resource-low`
		(Optional) Configures the gatekeeper to reject an admissions request (ARQ) from a requesting gateway if resources run low.

Step 4	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Configuring Intergatekeeper Communication

You can configure intergatekeeper communication either by means of DNS or manually.

Configuring Intergatekeeper Communication Using DNS

To configure intergatekeeper communication using DNS, use the following commands in global configuration mode.

SUMMARY STEPS

1. **ip name-server** *dns-servername [server-address2...server-address6*
2. **ip domain-name** *name*
3. **ras** [*gk-id@*] *host* [*:port*] [*priority*]

DETAILED STEPS

	Command	Purpose
Step 1	**ip name-server** *dns-servername* *[server-address2...server-address6]* **Example:** Router(config)# ip name-server 192.168.0.0 192.168.1.1	Specifies the DNS server address. Arguments are as follows: • *dns-servername*—IP address of the name server. • *server-address2...server-address6*—IP addresses of up to five additional name servers.
Step 2	**ip domain-name** *name* **Example:** Router(config)# ip domain-name cisco.com	Defines a default domain name that Cisco IOS software uses to complete unqualified host names (names without a dotted-decimal domain name). The argument is as follows: • *name*—Default domain name used to complete unqualified host names. Do not include the initial period that separates an unqualified name from the domain name.
Step 3	**ras** *[gk-id@] host [:port] [priority]*	For all gatekeepers in the system, enter a text record of the form into DNS. Arguments are as follows: • *gk-id*—Optional gatekeeper ID. If the optional gatekeeper ID is not specified, *host* is used as the gatekeeper ID. • *host*—IP address or the actual host name of the gatekeeper in the form *host.some_domain.com*. • *port*—Port number other than RAS port 1719. • *priority*—Order in which the listed gatekeepers are searched for endpoints. Gatekeepers with lower priorities are searched before those with higher priorities. **Note** See the note below about text records.

Note How you enter the text record for a particular domain depends on the DNS implementation. The following examples are for the Berkeley Internet Name Domain (BIND). These records are typically entered into the "hosts" database:

```
zone1.comintxt"ras gk.zone1.com"
zone2.comintxt"ras gk2@gk.zone2.com"
zone3.comintxt"ras gk.3@gk.zone3.com:1725"
zone4.comintxt"ras gk4@gk.zone4.com:1725 123"
zone5.comintxt"ras gk5@101.0.0.1:1725"
```

Configuring Intergatekeeper Communication Manually

If you choose not to use DNS or if DNS is not available, configure intergatekeeper communication manually. To configure intergatekeeper manual communication, use the following command in gatekeeper configuration mode for every other gatekeeper in the network.

SUMMARY STEPS

1. **gatekeeper**

2. **zone remote** *other-gatekeeper-name other-domain-name other-gatekeeper-address* [*port*]

3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** Router(config)# gatekeeper	Enters gatekeeper configuration mode.
Step 2	**zone remote** *other-gatekeeper-name other-domain-name other-gatekeeper-address* [*port*] **Example:** Router(config-gk)# zone remote gatekeeper4 xxx.com 192.168.0.0	Statically specifies a remote zone if Domain Name System (DNS) is unavailable or undesirable. Enter this command for each gatekeeper. Arguments are as follows: • *other-gatekeeper-name*—Name of the remote gatekeeper. • *other-domain-name*—Domain name of the remote gatekeeper. • *other-gatekeeper-address*—IP address of the remote gatekeeper. • *port*—RAS signaling port for the remote zone. Range: 1 to 65 535. Default: the well-known RAS port number 1719.
Step 3	**exit** **Example:** Router(config-gk)# exit	Exits the current mode.

Configuring Gatekeeper Alias Registration and Address Resolution

You can configure multiple prefixes for a local zone and register an endpoint belonging to multiple zone prefixes. Gatekeepers can accept a registration request (RRQ) message with multiple E.164 aliases with different prefixes.

Alias Registration

When a gatekeeper receives an RRQ message from a gateway with a Foreign Exchange Station (FXS) port configured to register its E.164 address, it performs either of the following steps:

• If the E.164 alias is prefix-qualified, the gatekeeper tries to match the prefix with the zone prefixes it has defined. If a prefix is found, the gatekeeper searches its E.164 alias table with the exact alias from the RRQ message, including the prepended prefix, to make sure the alias is unique.

• If no zone prefix is found, the gatekeeper searches its E.164 alias table with the exact alias from the RRQ message:

- If the alias does exist and it is not owned by the same endpoint, the gatekeeper sends a registration reject (RRJ) message.

- If the alias does not exist, the gatekeeper creates an entry in the table for the exact alias name from the RRQ message, with or without the prefix qualifier, and sends a registration confirm (RCF) message.

✎

Note With the Gatekeeper Alias Registration and Address Resolution Enhancements feature, the gatekeeper creates an entry in its E.164 alias table for the exact alias name from the RRQ message. It does not strip off the prefix before creating the entry.

Address Resolution

Resolution for ARQ Messages

When a gatekeeper receives an admission request (ARQ) message from a gateway, it performs either of the following steps:

- If there is a technology prefix specified in the admission request and it is a hopoff technology prefix, the gatekeeper sends a location request (LRQ) message.

- If there is no technology prefix or the technology prefix is not a hopoff technology prefix, the gatekeeper uses the exact E.164 alias in the ARQ message, including the zone prefix, if any, to search its zone prefix table and the E.164 aliases registered by local endpoints:

  - If no zone-prefix match is found and the **arq reject-unknown prefix** command is set, the gatekeeper sends an admission reject (ARJ) message.

  - If a match is found and the destination zone is not local, the gatekeeper sends an LRQ message to the remote zone.

  - If the destination address is an E.164 alias registered by an endpoint, the gatekeeper sends an admission confirm (ACF) message

  - If the destination zone is local and the destination address is not registered but the local gateway is found with the specified technology prefix or the default technology prefix, the gatekeeper sends an ACF. If no local gateway with the specified technology prefix is found, the gatekeeper sends an ARJ message.

    If there is no matching technology prefix and no default technology prefix is set, the gatekeeper sends an ARJ message.

Resolution for LRQ Messages

When a gatekeeper receives an LRQ message from a gateway, it performs either of the following steps:

- 'If a hopoff technology prefix is found in the Location Request and the destination zone is not local, the gatekeeper sends an LRQ message, if the lrq forward-queries command is set.

- If there is no technology prefix or the technology prefix is not a hopoff technology prefix, the gatekeeper uses the exact E.164 alias in the LRQ message to search its zone prefix table and the registered E.164 aliases.

  - If no match is found and the **lrq reject-unknown prefix** command is set, the gatekeeper sends a location reject (LRJ) message.

  - If a match is found and the destination zone is a remote zone, and the **lrq forward-queries** command is set, the gatekeeper sends an LRQ message to the destination zone.

– If the destination zone is local and the destination address is registered, the gatekeeper sends a location confirm (LCF) message.

– If the destination zone is local and the destination address is not registered but the local gateway is found with the specified technology prefix or the default technology prefix, the gatekeeper sends an LCF message. If no local gateway with the specified technology prefix is found, the gatekeeper sends an LRJ message.

– If the destination zone is local and the destination address is not registered, and there is no matching technology prefix and no default technology prefix is set, the gatekeeper sends an LRJ message.

Request Processing

A gatekeeper with the Gatekeeper Alias Registration and Address Resolution Enhancements feature processes requests in a new way, as showing in the following examples. The gatekeeper is configured with two local zones, zone1 and zone2, and three prefixes, as follows:

```
Router(config-gk)#zone local zone1 domain.com
Router(config-gk)#zone local zone2 domain.com
Router(config-gk)#zone prefix zone2 407 .......
Router(config-gk)#zone prefix zone1 408 .......
Router(config-gk)#zone prefix zone1 409 .......
```

Table 1 shows various E.164 alias registration requests and the resulting gatekeeper actions.

Table 1   E.164 Alias Registration Requests and Gatekeeper Actions

RRQ	Without This Feature	With This Feature	Action
4085551000 4095552000	RRJ	RCF	Two entries are created in the E.164 alias hash table: 4085551000 4095552000
4095551000	RCF	RCF	4095551000 is created in the table.
4085553000	RCF	RCF	4085553000 is created in the table.
5551234	RRJ	RCF	5551234 is created in the table.
4085551000	RRJ	RRJ	Gatekeeper rejects the request because it is a duplicate alias.
4085554000 4075554000	RRJ	RRJ	Gatekeeper rejects the request because the two prefixes (407 and 408) have different zone names (zone1 and zone2).

To allow endpoints to communicate between zones, gatekeepers must be able to determine which zone an endpoint is in and be able to locate the gatekeeper responsible for that zone. If the Domain Name System (DNS) mechanism is available, a DNS domain name can be associated with each gatekeeper.

Note For more information on DNS, see the "Configuring Intergatekeeper Communication" section on page 150.

Configuring Load Balancing with Alternate Gatekeepers

This section contains the following information:

Restrictions for Load Balancing with Alternate Gatekeepers

- The gatekeeper-to-gatekeeper redundancy and load-sharing mechanism requires the Cisco H.323 VoIP Gatekeeper for Cisco Access Platforms feature.
- The order in which LRQs are sent to the gatekeepers is based on the order in which the gatekeepers are listed. You cannot specify a priority number for a gatekeeper.
- Regardless of the order in which the LRQs are sent, the gateway still uses the first gatekeeper that sends an LCF.
- The settings for delay between LRQs and the LRQ window are global and cannot be set on a per-zone or technology-prefix basis.
- The number of remote gatekeepers multiplied by the delay per LRQ cannot exceed the Routing Information Protocol (RIP) timeout. Therefore, we recommend that you limit your list of remote gatekeepers to two or three.
- If LRQ forwarding is enabled on the directory gatekeeper, the *sequential* setting for LRQs is ignored.
- Only E.164 address resolution is supported.
- Using redundant H.323 zone support in the "directory gatekeeper" can generate extra RAS messages. Therefore, the number of "directory gatekeeper" levels should be kept to a minimum (two or three at the maximum).
- If a gatekeeper fails, the endpoint might use alternate gatekeepers to continue operation. The example below creates a local cluster associated with a local zone and defines an alternate gatekeeper within the cluster.

Information About Load Balancing with Alternate Gatekeepers

Load balancing allows the gatekeeper to move registered H.323 endpoints to an alternate gatekeeper or to reject new calls and registrations once a certain threshold is met.

If a gatekeeper fails, the endpoint might use alternate gatekeepers to continue operation.

The gatekeeper-to-gatekeeper redundancy and load-sharing mechanism expands the capability that is provided by the redundant H.323 zone support feature. Redundant H.323 zone support allows you to configure multiple gatekeepers to service the same zone or technology prefix by sending LRQs to two or more gatekeepers.

With the redundant H.323 zone support feature, the LRQs are sent simultaneously (in a "blast" fashion) to all of the gatekeepers in the list. The gateway registers with the gatekeeper that responds first. Then, if that gatekeeper becomes unavailable, the gateway registers with another gatekeeper from the list.

The gatekeeper-to-gatekeeper redundancy and load-sharing mechanism allows you to configure gatekeeper support and to give preference to specific gatekeepers. You may choose whether the LRQs are sent simultaneously or sequentially (one at a time) to the remote gatekeepers in the list. If the LRQs are sent sequentially, a *delay* is inserted after the first LRQ and before the next LRQ is sent. This delay allows the first gatekeeper to respond before the LRQ is sent to the next gatekeeper. The order in which LRQs are sent to the gatekeepers is based on the order in which the gatekeepers are listed (using either the **zone prefix** command or the **gw-type-prefix** command).

Once the local gatekeeper has sent LRQs to all the remote gatekeepers in the list (either simultaneously or sequentially), if it has not yet received a location confirmation (LCF), it opens a "window." During this window, the local gatekeeper waits to see whether a LCF is subsequently received from any of the remote gatekeepers. If no LCF is received from any of the remote gatekeepers while the window is open, the call is rejected.

Configuring Load Balancing with Alternate Gatekeepers

To configure load balancing and alternate gatekeepers, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **zone local** *local-zone-name domain-name* [*ras-ip-address*]
3. **zone cluster local** *cluster-name local-zone-name*
4. **element** *alternateGK ip-address* [*port*]
5. **exit**
6. **load-balance** [**endpoints** *max-endpoints*] [**calls** *max-calls*] [**cpu** *max-%cpu*] [**memory** *max-%mem-used*]
7. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`zone local `*`local-zone-name domain-name`* *`[ras-ip-address]`* **Example:** `Router(config-gk)# `**`zone local gk408or650`** **`xyz.com`**	Defines the gatekeeper's name or zone name. This is usually the fully domain-qualified host name of the gatekeeper.

Step 3	`zone cluster local `*`cluster-name local-zone-name`*
	
	Example:
	`Router(config-gk)# zone cluster local`
	`RTPCluster RTPGK1`
		Defines a local cluster for the local zone.

	Command	Purpose

Step 4	`element alternateGK ip-address [port]`
	
	Example:
	`Router(config-gk-cluster)# element alternateGK1 192.168.0.0`
		Defines the alternate gatekeeper in the local cluster. The alternate gatekeeper is an alternate gatekeeper to the local zone. Arguments are as follows:
		
		• *alternateGK*—Name of the alternate gatekeeper.
		• *ip-address*—IP address of the gatekeeper.
		• *port*— RAS signaling port number. Range: 1 to 65535. Default: the well-known RAS port number 1719.

Step 5	`exit`
	
	Example:
	`Router(config-gk-cluster)# exit`
		Exits the current mode.

Step 6	`load-balance [endpoints max-endpoints] [calls max-calls] [cpu max-%cpu] [memory max-%mem-used]`
	
	Example:
	`Router(config-gk)# load-balance endpoints 200 calls 100 cpu 75 memory 80`
		Configures load balancing. Keywords and arguments are as follows:
		
		• **endpoints** *max-endpoints*—Maximum number of endpoints
		• **calls** *max-calls*—Maximum number of calls
		• **cpu** *max-%cpu*—Maximum percentage of CPU usage
		• **memory** *max-%mem-used*—Maximum percentage of memory used

Step 7	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Verifying Load Balancing with Alternate Gatekeepers

To verify load balancing and alternate gatekeeper configuration, perform the following steps.

Step 1 **show gatekeeper status**

Use this command to see if load balancing is configured and if accounting vendor-specific attributes (VSAs) are enabled. The last five lines shown below, starting with Load Balance Count, display only when load balancing is enabled.

```
Router# show gatekeeper status

    Gatekeeper State: UP
    Load Balancing: ENABLED
    Zone Name: RoseGK
    Zone Name: PurpleGK
    Accounting: DISABLED
    Security: DISABLED
    Maximum Remote Bandwidth: unlimited
    Current Remote Bandwidth: 0 kbps
    Current Remote Bandwidth (w/Alt GKs): 0 kbps
    Load Balance Count: 0
    Calls: 0/unlimited
    Endpoints: 0/unlimited
    Memory: 0%/90%
```

```
            CPU: 0%/80%
```

Step 2   **show gatekeeper performance statistics**

Use this command to verify performance statistics.

```
Router# show gatekeeper performance statistics

Performance statistics captured since:19:00:12 EST Sun Feb 28 1993

RAS inbound message counters:
       Originating ARQ:426     Terminating ARQ:306     LRQ:154
RAS outbound message counters:
       ACF:731        ARJ:1  LCF:154        LRJ:0
       ARJ due to overload:0
       LRJ due to overload:0

Load balancing events:0
Real endpoints:5
```

Configuring Remote Clusters

The following commands define a group of associated gatekeepers in a remote cluster. This remote cluster can then be addressed using the **zone prefix** command in the same way that a remote gatekeeper would be addressed to route calls. However, rather than individually addressing each remote gatekeeper within the cluster, you can address the cluster as a single entity. Additionally, location requests (LRQs) are now sent round-robin to each gatekeeper within the remote cluster.

Configuring Remote Clusters

To configure remote clusters, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **zone local** *zonename domainname* [*ras-ip-address*] [*port*]
3. **zone cluster remote** *remote-cluster-name domain-name* [**cost** *cost* [**priority** *priority*]]
4. **element** *alternateGK IP-address* [*port*]
5. **exit**
6. **zone prefix** *remote-clustername e164-prefix*
7. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`zone local` *zonename domainname* [*ras-ip-address*] [*port*] **Example:** `Router(config-gk)# **zone local gk408or650 xyz.com**`	Defines the gatekeeper's name or zone name.
Step 3	`zone cluster remote` *remote-cluster-name domainname* [**cost** *cost* [**priority** *priority*]] **Example:** `Router(config-gk)# zone cluster remote SJCluster cisco.com`	Defines a remote cluster. Keywords and arguments are as follows: • *remote-cluster-name*—Remote cluster name. • *domain-name*—ID of the domain the remote cluster is serving. • **cost** *cost*—Cost. Range: 1 to 100. Default: 50. • **priority** *priority*—Priority value. Range: 1 to 100. Default: 50.
Step 4	`element` *alternateGK IP-address* [*port*] **Example:** `Router(config-gk-cluster)# element alternateGK1 192.168.0.0`	Defines component elements of local or remote clusters.

Step 5	`exit`
	
	Example:
	`Router(config-gk-cluster)# exit`
		Exits the current mode.

Step 6	`zone prefix` *remote-clustername e164-prefix* **Example:** `Router(config-gk)# zone prefix 40_gatekeeper 408*`	Keywords and arguments are as follows: • *remote-clustername*—Name of a local or remote cluster, which must have been defined by using the **zone local** or **zone remote** command. • *e164-prefix*— E.164 prefix in standard form followed by dots (.). Each dot represent a number in the E.164 address. For example, 212....... is matched by 212 and any seven numbers. **Note** Although a dot representing each digit in an E.164 address is the preferred configuration method, you can also enter an asterisk (*) to match any number of digits.

Step 7	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Verifying Remote Clusters

To verify configuration of remote clusters, perform the following steps.

Step 1   **show gatekeeper status cluster**

Use this command to display each element of a cluster. This command shows the health of the elements in a cluster and reports on the percentage of memory and CPU usage, the number of active calls, and the number of endpoints registered on the element. The Last Announce field tells you the time since the last announcement message was received from the alternate gatekeeper. In this example, MsPacman and LavenderGK are part of a local cluster.

```
Router# show gatekeeper status cluster

CLUSTER INFORMATION
                ===================
                              Active   Endpoint  Last
Hostname       %Mem  %CPU   Calls    Count     Announce
--------       ----  ----   ------   --------  --------
MsPacman       17    2      0        1         Local Host
LavenderGK     30    1      0        4         14s
```

Step 2   **show gatekeeper zone status**

Use this command to display the bandwidth information for all zones.

```
Router# show gatekeeper zone status

                        GATEKEEPER ZONES
                        ================
GK name       Domain Name   RAS Address     PORT  FLAGS
-------       -----------   -----------     ----- -----

RoseGK        cisco.com     209.165.201.30  1719  LS
  BANDWIDTH INFORMATION (kbps) :
    Maximum interzone bandwidth :unlimited
    Current interzone bandwidth :0
    Current interzone bandwidth (w/ Alt GKs) :0
    Maximum total bandwidth :unlimited
    Current total bandwidth :0
    Current total bandwidth (w/ Alt GKs) :0
    Maximum session bandwidth :unlimited
  SUBNET ATTRIBUTES :
    All Other Subnets :(Enabled)
  PROXY USAGE CONFIGURATION :
    Inbound Calls from all other zones :
      to terminals in local zone RoseGK :use proxy
      to gateways in local zone RoseGK  :do not use proxy
    Outbound Calls to all other zones :
      from terminals in local zone RoseGK :use proxy
      from gateways in local zone RoseGK  :do not use proxy
```

Step 3   **show gatekeeper zone cluster**

Use this command to display information about alternate gatekeepers. PRI represents the priority value assigned to an alternate gatekeeper. This field ranges from 0 to 127, with 127 representing the lowest priority.

```
Router# show gatekeeper zone cluster

ALTERNATE GATEKEEPER INFORMATION
================================
                                 TOT BW   INT BW   REM BW  LAST      ALT GK
```

```
LOCAL GK NAME ALT GK NAME   PRI (kbps)   (kbps)   (kbps)   ANNOUNCE  STATUS
------------- -----------   --- ------   ------   ------   --------  ------
RoseGK        LilacGK       120 0        0        0        7s        CONNECTED
```

Step 4   **show proxy h323 status**

Use this command to display information about the proxy such as the T.120 mode and what port is being used.

```
Router# show proxy h323 status

      H.323 Proxy Status
      ==================
H.323 Proxy Feature:Enabled
Proxy interface = Ethernet0:UP
Proxy IP address = 209.165.200.254
Proxy IP port = 11720
Application Specific Routing:Disabled
RAS Initialization:Complete
Proxy aliases configured:
  H323_ID:PROXY
Proxy aliases assigned by Gatekeeper:
  H323_ID:PROXY
Gatekeeper multicast discovery:Disabled
Gatekeeper:
    Gatekeeper ID:DVM1
    IP address:209.165.200.254
Gatekeeper registration succeeded
T.120 Mode:PROXY
RTP Statistics:OFF
Number of calls in progress:0
```

Step 5   **show gatekeeper cluster**

Use this command to display all clusters defined in the gatekeeper and with their component elements.

```
Router# show gatekeeper cluster

gatekeeper
    zone local RTPGK1cisco.com
    zone cluster local RTPCluster RTPGK1
        element RTPGK2 209.165.200 1719
        element RTPGK3 209.165.200 1719
    zone cluster remote SJCluster cisco.com
        element SJGK1 209.18.79.23 1719
        element SJGK2 209.18.79.24 1719
        element SJGK3 209.18.79.25 1719
no shutdown

Router# show gatekeeper cluster

                CONFIGURED CLUSTERS
                ===================
Cluster Name    Type      Local Zone   Elements    IP
------------    ----      ----------   --------    --
RTPCluster      Local     RTPGK1       RTPGK2      209.165.200.254 1719
                                       RTPGK3      209.165.200.223 1719
SJCluster       Remote                 SJGK1       209.165.200.257 1719
                                       SJGK2       209.165.200.258 1719
                                       SJGK3       209.165.200.259 1719
```

Configuring Static Nodes

In some cases, registration information is not accessible for a terminal or endpoint from any gatekeeper. This inaccessible registration information may be because the endpoint does not use RAS, is in an area where no gatekeeper exists, or is in a zone where the gatekeeper addressing is unavailable either through DNS or through configuration. These endpoints can still be accessed via a gatekeeper by entering them as static nodes.

To enter endpoints as static nodes, use the following commands beginning in global configuration mode.

Prerequisites for Configuring Static Nodes

- Obtain the address of the endpoint.

SUMMARY STEPS

1. **gatekeeper**
2. **zone local** *gatekeeper-name domain-name* [*ras-ip-address*]
3. **alias static** *ip-signalling-addr* [*port*] **gkid** *gatekeeper-name* [**ras** *ip-ras-addr port*] [**terminal** | **mcu** | **gateway** {**h320** | **h323-proxy** | **voip**}] [**e164** *e164-address*] [**h323id** *h323-id*]
4. Repeat Step 3 for each E.164 address that you want to add for the endpoint.
5. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** `Router(config)# gatekeeper`	Enters gatekeeper configuration mode.
Step 2	**zone local** *gatekeeper-name domain-name* [*ras-ip-address*] **Example:** `Router(config-gk)# zone local gatekeeper1 domain1`	Specifies a zone controlled by a gatekeeper.

	Command	Purpose
Step 3	`alias static` *ip-signaling-addr* `[`*port*`] gkid` *gatekeeper-name* `[ras` *ip-ras-addr port*`]` `[terminal	mcu	gateway {h320	h323-proxy	voip}] [e164` *e164-address*`] [h323id` *h323-id*`]`  **Example:** `Router(config-gk)# alias static` `ip-signalling-addr gkid gatekeeper1`	Creates a static entry in the local alias table for each E.164 address. Keywords and arguments are as follows: • *ip-signaling-addr*—IP address of the H.323 node, used as the address to signal when establishing a call. • *port*—Port number other than the endpoint call-signaling well-known port number (1720). • **gkid** *gatekeeper-name*—*Local gatekeeper of whose zone this node is a member.* • **ras** *ip-ras-addr*—*Node remote access server (RAS) signaling address. Default: ip-signalling-addr parameter is used in conjunction with the RAS well-known port.* • *port*—Port number other than the RAS well-known port number (1719). • **terminal—**Alias is a terminal. • **mcu**—Alias is a multiple control unit (MCU). • **gateway**—Alias is a gateway. • **h320**—Alias is an H.320 node. • **h-323 proxy**—Alias is an H.323 proxy. • **voip**—Alias is VoIP. • **e164** *e164-address*—Node E.164 address. Can be used more than once to specify as many E.164 addresses as needed. A maximum number of 128 characters can be entered for this address. To avoid exceeding this limit, you can enter multiple **alias static** commands with the same call-signaling address and different aliases. • **h323-id** *h323-id*—Node H.323 alias. Can be used more than once to specify as many H.323 identification aliases as needed. A maximum number of 256 characters can be entered for this address. To avoid exceeding this limit, you can enter multiple commands with the same call signaling address and different aliases.
Step 4	Repeat Step 3 for each E.164 address that you want to add for the endpoint.	—

Step 5	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Configuring AAA and RADIUS

Version 1 of the H.323 specification does not provide a mechanism for authenticating registered endpoints. Credential information is not passed between gateways and gatekeepers. However, by enabling AAA on the gatekeeper and configuring for RADIUS and TACACS+, a rudimentary form of identification can be achieved.

In Version 2 and higher, authentication is done using tokens. See "Configuring Security and Authentication" section on page 171 for more information.

If the AAA feature is enabled, the gatekeeper attempts to use the registered aliases along with a password and completes an authentication transaction to a RADIUS and TACACS+ server. The registration is accepted only if RADIUS and TACACS+ successfully authenticates the name.

The gatekeeper can be configured so that a default password can be used for all users. It can also be configured to recognize a password separator character that allows users to piggyback their passwords onto H.323-ID registrations. In this case, the separator character separates the ID and password fields.

> **Note** The names loaded into RADIUS and TACACS+ are probably not the same names provided for dial access because they may all have the same password.

If AAA is enabled on the gatekeeper, the gatekeeper emits an accounting record each time a call is admitted or disconnected.

Configuring H.323 Users via RADIUS

> **Note** For more information about configuring AAA services or RADIUS, see the *Cisco IOS Security Configuration Guide* at http://www.cisco.com/en/US/docs/ios/sec_user_services/configuration/guide/15_0/sec_user_services_15_0_book.html.

To authenticate H.323 users via RADIUS, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **aaa new-model**

2. **aaa authentication login {default |** *listname*} *method1* [*method2...*]

3. **radius-server host {***hostname* | *ip-address*} [**auth-port** *port*] [**acct-port** *port*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]

4. **radius-server key {***0 string* | *7 string* | *string*}

5. **gatekeeper**

6. **security {any | h323-id | e164}** {**password default** *password* | **password separator** *character*}

7. **exit**

8. Enter each user into the RADIUS database.

DETAILED STEPS

	Command	**Purpose**

Step 1	`aaa new-model`
	
	Example:
	`Router(config)# aaa new-model`
		Enables the authentication, authorization, and accounting (AAA) access model.

Step 2	`aaa authentication login {default |` *`listname`*`}` *`method1`* `[`*`method2...`*`]`
	
	Example:
	`Router(config)# aaa authentication login default`
		Sets AAA authentication at login. Keywords and arguments are as follows:
		
		• **default**—Uses the listed authentication methods that follow this keyword as the default list of methods when a user logs in.
		
		• *listname*—Character string used to name the list of authentication methods activated when a user logs in.
		
		• *method1* [*method2...*]—At least one of the following authentication methods:
		
		   – **enable**—Enable password
		   – **krb5**—Kerberos 5
		   – **krb5-telnet**—Kerberos 5 Telnet authentication protocol when using Telnet to connect to the router
		   – **line**—Line password
		   – **local**—Local username database
		   – **local-case**—Case-sensitive local username
		   – **none**—No authentication
		   – **group radius**—List of all RADIUS servers
		   – **group tacacs+**—List of all TACACS+ servers
		   – **group group-name**—Subset of RADIUS or TACACS+ servers as defined by the **group server radius** or **aaa group server tacacs+** command

	Command	Purpose

Step 3	`radius-server host {`*`hostname`* `|` *`ip-address`*`}` `[auth-port` *`port`*`] [acct-port` *`port`*`] [timeout` *`seconds`*`] [retransmit` *`retries`*`] [key` *`string`*`]`
	
	 Example:
	 `Router(config)# radius-server host 10.0.0.1 auth-port 1645 acct-port 1646`
		Specifies the RADIUS server host. Keywords and arguments are as follows:
		
		 • **auth-port** *port*—User Datagram Protocol (UDP) destination port for authentication requests; the host is not used if set to 0. Default: 1645.
		
		 • **acct-port** *port*—UDP destination port for accounting requests; the host is not used if set to 0. Default: 1646.
		
		 • **timeout** *seconds*—Time, in seconds, for which the router waits for the RADIUS server to reply before retransmitting. This setting overrides the global value of the **radius-server timeout** command. Range: 1 to 1000. Default: global value.
		
		 • **retransmit** *retries*—Number of times that a RADIUS request is resent to a server if that server is not responding or responding slowly. Overrides the global setting of the **radius-server retransmit** command. Range: 1 to 100. Default: the global value.
		
		 • **key** *string*—Authentication and encryption key used between the router and the RADIUS daemon running on this RADIUS server. Must match the encryption used on the RADIUS daemon. Overrides the global setting of the **radius-server key** command. Default: the global value.

Step 4	`radius-server key {`*`0`* *`string`* `|` *`7`* *`string`* `|` *`string`*`}`
	
	 Example:
	 `Router(config)# radius-server key 0 143212343`
		Sets the authentication and encryption key for all RADIUS communications between the router and the RADIUS daemon. Keywords and arguments are as follows:
		
		 • **0** *string*—Unencrypted (cleartext) shared key
		
		 • **7** *string*—Hidden shared key
		
		 • *string*—Unencrypted (cleartext) shared key

Step 5	`gatekeeper`
	
	 Example:
	 `Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

	Command	Purpose
Step 6	`security {any	h323-id	e164} {password default password	password separator character}`  **Example:** `Router(config-gk)# security any password default thisismypassword`	Enables authentication and authorization on a gatekeeper and specifies the means of identifying the user to RADIUS/TACACS+. Keywords and arguments are as follows: • **any**—First alias of an incoming Registration, Admission, and Status (RAS) registration, regardless of its type. • **h323-id**—First H.323 ID type alias. • **e164**—First E.164 address type alias. • **password default** *password*—Default password that the gatekeeper associates with endpoints when authenticating them with an authentication server. Must be identical to the password on the authentication server. • **password separator** *character*—Character that endpoints use to separate the H.323-ID from the piggybacked password in the registration. This allows each endpoint to supply a user-specific password. The separator character and password are stripped from the string before it is treated as an H.323-ID alias to be registered. Note that passwords may be piggybacked only in the H.323-ID, not the E.164 address. This is because the E.164 address allows a limited set of mostly numeric characters. If the endpoint does not wish to register an H.323-ID, it can still supply an H.323-ID that consists of just the separator character and password. This is understood to be a password mechanism, and no H.323-ID is registered.

Step 7	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Step 8	Enter each user into the RADIUS database.	Use either of the following: • If using the **security password default** command, use the default password. • If using the piggybacked password mechanism or the actual passwords, use the user H.323-ID or the E.164 address, depending on how the gatekeeper was configured.

Configuring a RADIUS/AAA Server

To configure a RADIUS/AAA server with information about the gatekeeper for your network installation, use the following commands in global configuration mode.

SUMMARY STEPS

1. **aaa new-model**

2. **aaa authentication login** {**default** | *listname*} *method1* [*method2...*]

3. **radius-server deadtime** *minutes*

4. **radius-server host** {*hostname* | *ip-address*} [**auth-port** *port*] [**acct-port** *port*] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]

5. **radius-server key** {*0 string* | *7 string* | *string*}

6. Configure the CiscoSecure AAA server.

DETAILED STEPS

	Command	Purpose

Step 1	`aaa new-model`
	
	Example:
	`Router(config)# aaa new-model`
		Enables the authentication, authorization, and accounting (AAA) model.

Step 2	`aaa authentication login {default	listname} method1 [method2...]`  **Example:** `Router(config)# aaa authentication login default`	Sets AAA authorization at login. For a list of keywords and arguments, see the "Configuring H.323 Users via RADIUS" section on page 164, Step 2.

Step 3	`radius-server deadtime minutes`
	
	Example:
	`Router(config)# radius-server deadtime 120`
		Sets the time, in minutes, for which a RADIUS server is skipped over by transaction requests. Range: 1 to 1440 (24 hours).

Step 4	`radius-server host {hostname	ip-address} [auth-port port] [acct-port port] [timeout seconds] [retransmit retries] [key string]`  **Example:** `Router(config)# radius-server host 10.0.0.1 auth-port 1645 acct-port 1646`	Specifies the RADIUS server host. For a list of keywords and arguments, see "Configuring H.323 Users via RADIUS" section on page 164, Step 3.

	Command	Purpose
Step 5	`radius-server key {0 string	7 string	string}`	Sets the authentication and encryption key for all RADIUS communications between the router and the RADIUS daemon.
	Example: `Router(config) radius-server key 7 anykey`	For a list of arguments, see "Configuring H.323 Users via RADIUS" section on page 164 Step 4.

Step 6	Configure the CiscoSecure AAA server.	• In the /etc/raddb/clients file, provide the following information:
		
		`#Client Name     Key`
		`#--------------   ----------`
		`gk215.cisco.com   testing123`
		
		Where *gk215.cisco.com* is resolved to the IP address of the gatekeeper requesting authentication.
		• In the /etc/raddb/users file, provide the following information:
		
		`h323id@cisco.com  Password = "password"`
		`User-Service-Type = Framed-User,`
		`Login-Service = Telnet`
		
		Where *h323id@cisco.com* is the h323-id of the gateway authenticating to gatekeeper gk215.cisco.com.

Configuring User Activity for RADIUS

After you enable AAA and configure the gateway to recognize RADIUS as the remote security server providing authentication services, the next step is to configure the gateway to report user activity to the RADIUS server in the form of connection accounting records.

To send connection accounting records to the RADIUS server, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **aaa accounting connection h323 {stop-only | start-stop | wait-start | none} [broadcast] group** *groupname*

2. **gatekeeper**

3. **aaa accounting**

4. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`aaa accounting connection h323 {stop-only | start-stop | wait-start | none} [broadcast] group groupname`
	
	Example:
	`Router(config)# aaa accounting connection h323 start-stop group group1`
		Defines the accounting method list H.323 with RADIUS as a method. Keywords and arguments are as follows:
		
		• **stop-only—Sends a "stop" accounting notice at the end of the requested user process.**
		
		• **start-stop—Sends a "start" accounting notice at the beginning of a process and a "stop" notice at the end of a process. The "start" notice is sent in the background. The requested process begins regardless of whether the "start" accounting notice is received by the server.**
		
		• **wait-start—Sends a "start" accounting notice at the beginning of a process and a "stop" notice at the end of a process. The "start" notice is sent in the background. The requested process does not begin until the "start" accounting notice is received by the server.**
		
		• **none—Disables accounting services on this line or interface.**
		
		• **broadcast—Sends accounting records to multiple AAA servers. Simultaneously sends accounting records to the first server in each group. If the first server is unavailable, failover occurs using the backup servers defined within that group.**
		
		• **group *groupname*—Server group to be used for accounting services. The following are valid group names:**
		
		– *string*—Character string used to name a server group
		
		– **radius**—List of all RADIUS hosts
		
		– **tacacs+**—List of all TACACS+ hosts

Step 2	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 3	`aaa accounting`
	
	Example:
	`Router(config-gk)# aaa accounting`
		Enables authentication, authorization, and accounting (AAA) of requested services for billing or security purposes when you use RADIUS or TACACS+.

Step 4	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Note For more information about AAA connection accounting services, see the *Cisco IOS Security Configuration Guide* at
http://www.cisco.com/en/US/docs/ios/sec_user_services/configuration/guide/15_0/sec_user_services_15_0_book.html.

Configuring Security and Authentication

This section contains the following information:

- Restrictions for Security and Authentication, page 171
- Information About Security and Authentication, page 171
- Configuring Domain Zones and the IZCT Password, page 177
- Configuring Cisco Access Tokens, page 178
- Configuring Tokenless Call Authorization, page 181

Restrictions for Security and Authentication

- CAT is a Cisco-proprietary security mechanism and requires a Cisco solution to receive the full end-to-end benefits of the Gatekeeper-to-Gatekeeper Authentication feature.
- LRQ message authentication is done on a hop-by-hop basis. Because a non-Cisco gatekeeper does not support CATs, authentication stops at the non-Cisco gatekeeper. If a non-Cisco gatekeeper can support LRQ forwarding, end-to-end authentication is achieved. However, LRQ message authentication is performed only at the Cisco gatekeepers.
- If IZCT is used for Clustered Gatekeepers, the same IZCT password should be used on all the Gatekeepers belonging to the same cluster

Information About Security and Authentication

This section contains the following information:

- Interzone ClearTokens (IZCTs), page 171
- Configuring Cisco Access Tokens, page 178
- Configuring Tokenless Call Authorization, page 181

Interzone ClearTokens (IZCTs)

The Inter-Domain Gatekeeper Security Enhancement provides a means of authenticating and authorizing H.323 calls between the administrative domains of Internet Telephone Service Providers (ITSPs).

An interzone ClearToken (IZCT) is generated in the originating gatekeeper when a location request (LRQ) is initiated or an admission confirmation (ACF) is about to be sent for an intrazone call within an ITSP's administrative domain. As the IZCT traverses through the routing path, each gatekeeper stamps the IZCT's destination gatekeeper ID with its own ID. This identifies when the IZCT is being passed over to another ITSP's domain. The IZCT is then sent back to the originating gateway in the location confirmation (LCF) message. The originating gateway passes the IZCT to the terminating gateway in the SETUP message.

The terminating gateway forwards the IZCT in the AnswerCall admission request (ARQ) to the terminating gatekeeper, which then validates it.

Within the IZCT format, the following information is required:

- srcCarrierID —Source carrier identification
- dstCarrierID — Destination carrier identification
- intCarrierID — Intermediate carrier identification
- srcZone — Source zone
- dstZone — Destination zone
- interzone type
    - INTRA_DOMAIN_CISCO
    - INTER_DOMAIN_CISCO
    - INTRA_DOMAIN_TERM_NOT_CISCO
    - INTER_DOMAIN_ORIG_NOT_CISCO

Figure 1 shows a simple inter-ITSP diagram of the IZCT flow.

Figure 1   *Inter-ITSP Diagram of the Inter-Domain Gatekeeper Security Enhancement Flow*
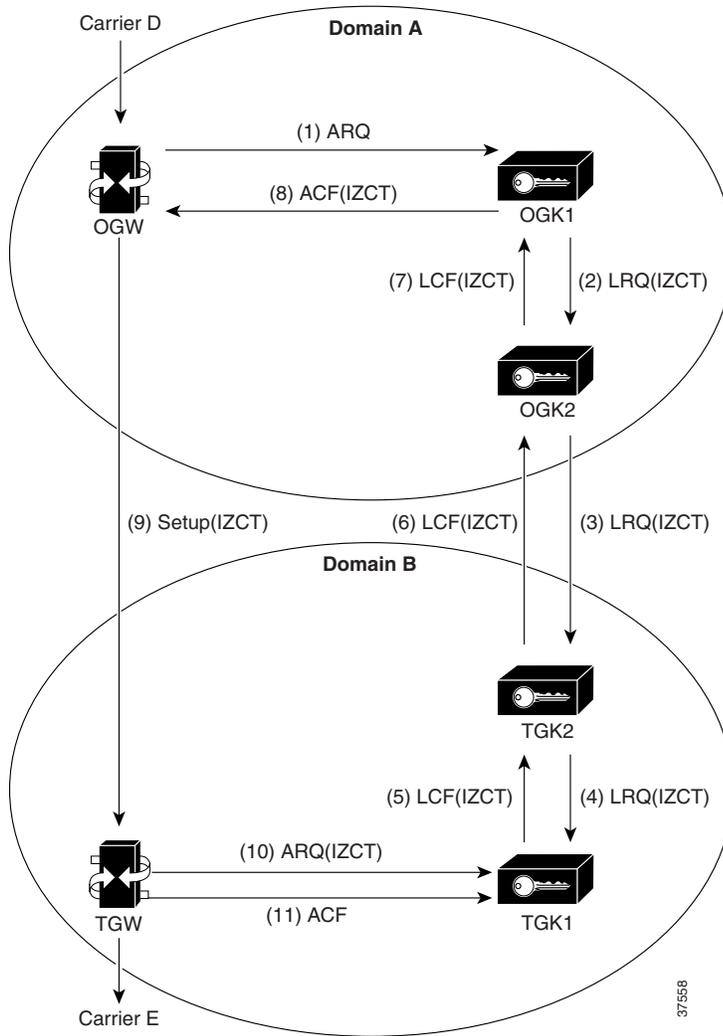
1. The originating gateway sends an ARQ message with an interface description as a nonstandard field to originating gatekeeper 1 (OGK1). The interface description is treated as a source carrier identifier.

2. Upon receiving the ARQ, OGK1 creates an IZCT with the following:

   • srcCarrierID— Source carrier identification, received from the ARQ

   • dstCarrierID—Destination carrier identification, received from the CSR

   • intCarrierID—Intermediate carrier identification, received from the CSR

   • srcZone—Source zone name or a cluster name if the gatekeeper is a member of a cluster

   • dstZone—Destination zone is set to null

   • interZoneType—Interzone type is set to INTRA_DOMAIN_CISCO

   The IZCT is sent in an LRQ to OGK2.

3. OGK2 determines that the LRQ did not come from a foreign domain, replaces the IZCT's srcZoneID with its ID (or cluster name, if the gatekeeper is member of a cluster), and forwards the LRQ with the updated IZCT to terminating gatekeeper 2 (TGK2).

4. TGK2 determines that the LRQ came from a foreign domain, updates the IZCT's dstZone with its own ID (or cluster name, if the gatekeeper is a member of a cluster) and the interZoneType as INTER_DOMAIN_CISCO, and passes the updated IZCT to TGK1. TGK2 treats the zone from which an LRQ is received as foreign-domain zone in either of the following two scenarios:

   a. The TGK2's remote zone list does not contain the zone from which an LRQ is received.

   b. The TGK2's remote zone list contains the zone from which an LRQ is received and the zone is marked with a foreign-domain flag.

5. TGK1 updates the IZCT's dstCarrierID to Carrier E, which is determined by the routing process; generates a hash with the IZCT's password; and sends an LCF with the updated IZCT in it. If TGK1 is a clustered gatekeeper, then the IZCT password is identical across the cluster.

6. TGK2 forwards the LCF to OGK2.

7. OGK2 forwards the LCF to OGK1.

8. OGK1 extracts the IZCT from the LCF and sends it in an ACF to the OGW.

9. The OGW sends the IZCT to the TGW in the H.225 SETUP message.

10. The TGW passes the IZCT to the TGK1 in an ARQ answerCall.

11. TGK1 authenticates the destination IZCT successfully, because TGK1generated the hash in the IZCT.

Note In the case of an inter-ITSP call, border zones (in the above example, OGK2 and TGK2) are identified as the srcZone and dstZone of the IZCT that is returned in the ACF to the OGW. If the call is intra-ITSP, leaf zones are identified as the srcZone and dstZone of the IZCT that is returned in the ACF to the OGW.

The main tasks are marking foreign and local domain zones and setting up an IZCT password for use in all the zones. After the **security izct password** command is issued, the technology prefix for the gatekeepers must be configured for the gateways. The gatekeeper must be enabled to forward LRQ messages that contain E.164 addresses matching zone prefixes controlled by remote gatekeepers.

Cisco Access Tokens

The Gatekeeper-to-Gatekeeper Authentication feature provides additional security for H.323 networks by introducing the ability to validate intradomain and interdomain gatekeeper-to-gatekeeper LRQ messages on a per-hop basis. When used in conjunction with per-call security using the interzone ClearToken (IZCT), network resources are protected from attackers and security holes are prevented.

The Gatekeeper-to-Gatekeeper Authentication feature provides a Cisco access token (CAT) to carry authentication within zones. The CAT is used by adjacent gatekeepers to authenticate each other and is configured on a per-zone basis. In addition, service providers can specify inbound passwords to authenticate LRQ messages that come from foreign domains and outbound passwords to be included in LRQ messages to foreign domains.

The call flows illustrated in Figure 2 and Figure 3 show the steps that occur with a successful LRQ authentication and with an unsuccessful LRQ authentication.

Note Although the IZCT is not required for use with the Gatekeeper-to-Gatekeeper Authentication feature, it is recommended and is shown below in the call flow examples.

Figure 2   *Call Flow with Successful LRQ Authentication*
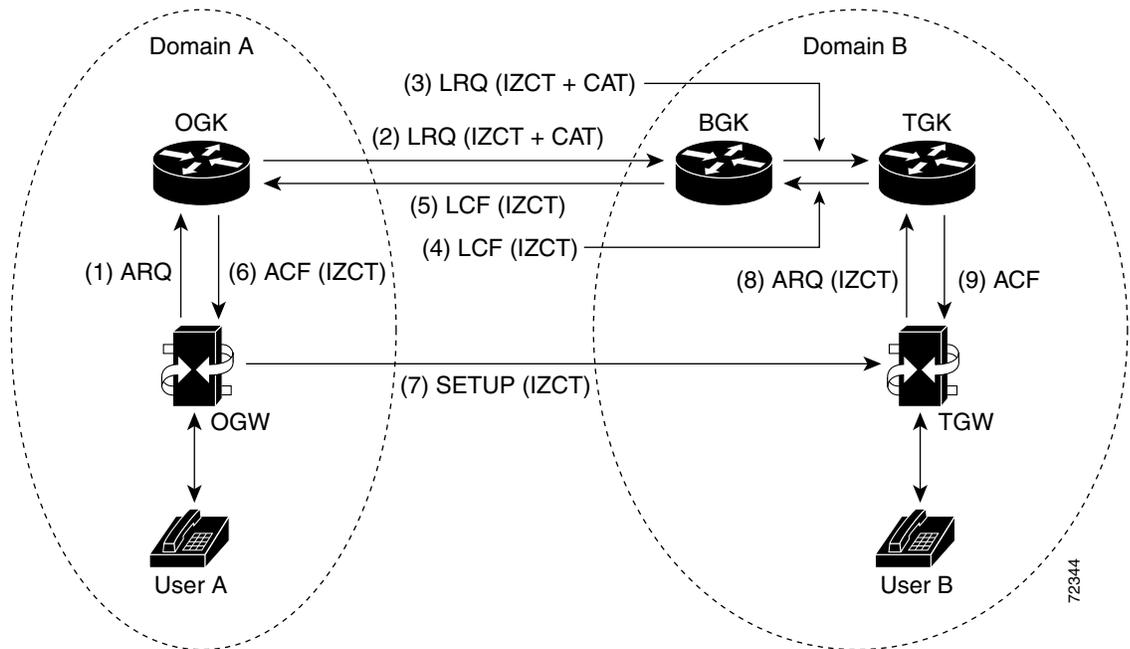
The following sequence occurs in the call flow:

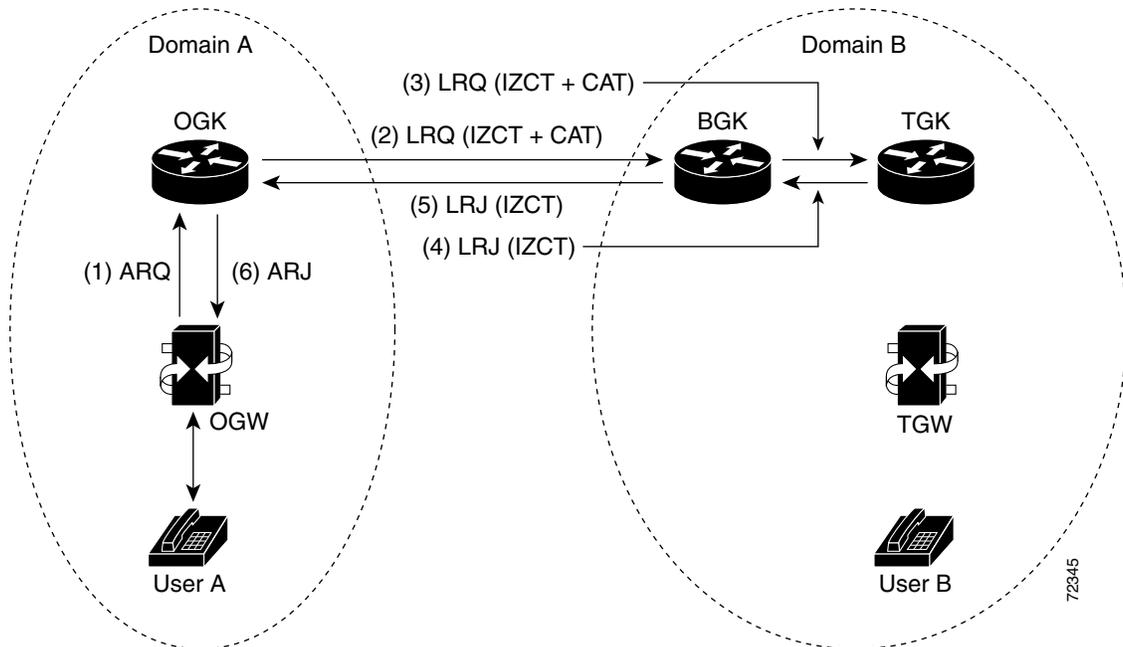1. User A calls User B. The originating dial peer is configured for H.323 Registration, Admission, and Status (RAS) and sends an Admission Request (ARQ) message to the originating gatekeeper (OGK).

2. Assuming the OGK has security enabled, the OGK generates an IZCT and a CAT to include in the LRQ message. The IZCT is used for per-call authorization while the CAT is used for gatekeeper-to-gatekeeper authentication. The CAT includes the following:

   • general_id: gatekeeper ID (OGK)

   • timeStamp: local gatekeeper time

   • randomValue: a random number

   • MD5 hash value

3. The border gatekeeper (BGK) receives the LRQ message, checks its gatekeeper configuration, and determines that the LRQ should be authenticated before forwarding the LRQ message to the terminating gatekeeper (TGK). Once accepted, the BGK creates a new CAT and includes it in the LRQ message sent to the TGK.

4. The TGK receives the LRQ message, checks its gatekeeper configuration, and determines that the LRQ should be authenticated. The E.164 address indicates that the destination is a local gateway, so the TGK acknowledges the request by sending a Location Confirmation (LCF) message, including an updated IZCT, to the BGK.

5. The BGK transfers the LCF to the OGK. Normal call signaling proceeds.

6. The OGK sends an Admission Confirmation (ACF) message to the OGW. The IZCT is copied to the ACF.

7. The OGW sends a SETUP message to the terminating gateway (TGW).

8. The TGW sends an ARQ message to the TGK. The TGK authorizes the call by comparing the IZCT with a locally created IZCT.

9. The TGK sends an ACF to the TGW. The call is set up between the TGW and User B.

Figure 3   Call Flow with Unsuccessful LRQ Authentication

The following sequence occurs in the call flow:

1. User A calls User B. The originating dial peer is configured for H.323 RAS and sends an ARQ to the OGK.

2. Assuming the OGK has security enabled, the OGK generates an IZCT and a CAT to include in the LRQ message. The IZCT is used for per-call authorization while the CAT is used for gatekeeper-to-gatekeeper authentication. The CAT includes the following:

- general_id—Gatekeeper ID (OGK)
- timeStamp—Local gatekeeper time
- randomValue—A random number
- MD5 hash value

3. The BGK receives the LRQ message, checks its gatekeeper configuration, and determines that the LRQ should be authenticated before forwarding the LRQ message to the TGK. Once accepted, the BGK creates a new CAT and includes it in the LRQ message sent to the TGK. However, in this example, an incorrect outbound password is used.

4. The TGK receives the LRQ message, checks its gatekeeper configuration, and determines that the LRQ should be authenticated. Because an incorrect outbound password was used by the BGK, the LRQ CAT and the locally created CAT are not equivalent. The TGK sends a Location Reject (LRJ) message back to the BGK and includes a reject reason of LRJ_INVAL_PERMISSION.

5. The BGK sends the LRJ to the OGK.

6. The OGK sends an Admission Reject (ARJ) message to the OGW and signaling is terminated.

Tokenless Call Authorization

The Tokenless Call Authorization feature is an alternative to using IZCTs and CATs to provide gatekeeper security in an H.323 voice network. ITSPs may not control gatekeepers in other domains to which they connect; for example, if these domains do not have Cisco software installed on the gatekeepers, tokens cannot be used. Additionally, the Tokenless Call Authorization feature can be used with Cisco Call Manager; tokens cannot.

With the Tokenless Call Authorization feature, an access list of all known endpoints is configured on the gatekeeper. The gatekeeper is configured to use the access list when processing calls. Rather than rejecting all calls that do not contain IZCTs or CATs, gatekeepers reject only calls that do not have tokens and are not from endpoints on the access list.

Configuring Domain Zones and the IZCT Password

This section contains the following information:

-
-

Configuring Zones and Password

To configure domain zones and the IZCT password, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. `security izct password` *password*
3. **no shutdown**
4. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** `Router(config)# gatekeeper`	Enters gatekeeper configuration mode.
Step 2	`security izct password` *password* **Example:** `Router(config-gk)# security izct password thisismypassword`	Sets the IZCT password. The password must be from six to eight alphanumeric characters. All gatekeepers in a cluster should have the same IZCT password. To disable the IZCT password, use the **no** form of the command.

	Command	Purpose

Step 3	`no shutdown`
	
	Example:
	`Router(config-gk)# no shutdown`
		Ensures that the gatekeepers are activated.

Step 4	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Verifying Zones and Password

To verify that the IZCT is enabled, perform the following step.

Step 1 **show running-config**

Use this command to display configuration information.

```
Router# show running-config

gatekeeper
 zone local 35_dirgk cisco.com 172.18.198.196
 zone remote 40_gatekeeper cisco.com 172.18.198.91 1719
 zone remote 34_dirgk cisco.com 172.18.198.197 1719 foreign-domain
 zone prefix 40_gatekeeper 408*
 zone prefix 34_dirgk *
 security izct password ABCDEF
 lrq forward-queries
 no shutdown
```

Configuring Cisco Access Tokens

This section contains the following information:

Configuring Tokens

To configure gatekeeper-to-gatekeeper authentication, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **security password-group** *groupname* **lrq** {**receive** *password* [**encrypted**] [**effective** *hh:mm day month year*] | **send** *password* [**encrypted**]}
3. **security zone** {*zonename* | *****} **password-group** *groupname*
4. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`security password-group` *groupname* **`lrq`** {**`receive`** *password* [**`encrypted`**] [**`effective`** *hh:mm day month year*] | **`send`** *password* [**`encrypted`**]} **Example:** `Router(config-gk)# security password-group groupname lrq receive password`	Defines the passwords used by remote gatekeeper zones and associates them with an ID. Keywords and arguments are as follows: • *groupname*—ID given to a group of passwords. The group can contain inbound and outbound passwords. The group name can include up to 16 characters (any characters on the keyboard). • **lrq receive** *password*—Password that is used to validate any LRQ messages that are received from the specified remote zone. The password can be up to 16 characters (any characters on the keyboard) for cleartext format and 34 characters for encrypted format. • **encrypted**—Password is in encrypted format. The password is always displayed in encrypted format. Default: the password is in cleartext format. • **effective** *hh:mm day month year*—Time and date on which the current **lrq receive** *password* expires. Old and new passwords are valid until the configured time value expires. After expiration, only the new password is valid. After you configure the keyword and time (for example, a day later), the following syslog message displays ("china" is the **password-group** name): `%GK-5-RX_LRQ_PASSWORD_UPDATED:LRQ receive password for security password-group 'china' has been updated.` • **lrq send** *password*—Password that is contained in the CAT and sent in the outbound LRQ messages. Can be up to 16 characters (any characters on the keyboard) for cleartext format and 34 characters for encrypted format. If multiple changes are made to the password groups, the latest update takes precedence.

	Command	Purpose

Step 3	`security zone {`*`zonename`* `| *} password-group` *`groupname`*
	
	Example:
	`Router(config-gk)# security zone *`
	`password-group groupname`
		Associates a remote zone gatekeeper with a specific password group. If a remote zone sends an LRQ message to the gatekeeper, the gatekeeper checks to see if there is a security password group configured for that remote zone name. If one exists, the gatekeeper gets the password information from the group name configured for that security zone.
		
		For example, if you used the command in Step 2 to create a password group named "china," you could use this command to associate one or more of your remote gatekeepers with that password group.
		
		Keywords and arguments are as follows:
		
		• *zonename*—Remote zone gatekeeper.
		
		• *—Using the wildcard (*) means that remote zones that do not have a security zone configured defaults to the security zone password group on the receiving gatekeeper and that the received LRQ message is authenticated using the wildcard-related passwords. Using the wildcard does not affect transmitted LRQ messages.
		
		• **password-group** *groupname*—Password group created using the **security password-group** command.

Step 4	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Verifying Tokens

To verify configuration of access tokens, perform the following step.

Step 1   **show running-config**

Use this command to verify configuration of remote zone and security features.

Note   For security reasons, passwords created using the **security password-group** command are encrypted when displayed in the command output.

```
Router# show running-config

gatekeeper
 zone local tsunamiGK cisco 172.18.195.138
 zone remote laharGK cisco 172.18.195.139 1719
 zone prefix laharGK 987*
 security izct password 123456
 security password-group 1 lrq receive 0257550A5A57 encrypted
 security password-group 1 lrq send 144540595E56 encrypted
 security password-group 2 lrq receive 091F1D5A4A56 encrypted
 security password-group 2 lrq send 135143465F58 encrypted
```

```
security zone larharGK password-group 1
no shutdown
```

Configuring Tokenless Call Authorization

This section contains the following information:

Configuring the IP Access List

Perform this task to create a list of endpoints known to the gatekeeper. Calls from these endpoints are accepted by the gatekeeper even if the endpoints are located in a different domain.

To configure the IP access list, use the following command beginning in global configuration mode.

SUMMARY STEPS

1. **access-list** *access-list-number* {**permit** | **deny** | **remark**} *source* [*source-wildcard*] [*log*]

DETAILED STEPS

	Command or Action	Purpose

Step 1	`access-list` *access-list-number* {**permit** | **deny** | **remark**} *source* [*source-wildcard*] [*log*]
	
	Example:
	`Router(config)# access-list 20 permit`
	`172.16.10.190`
		Configures the access list mechanism for filtering frames by protocol type or vendor code. Keywords and arguments are as follows:
		
		• *access-list-number*—Number of an access list. Range: a decimal number 1 to 99 (standard) or 1300 to 1999 (extended). Only standard IP access lists 1 to 99 are supported for the Tokenless Call Authorization feature.
		
		• **permit**—Permits access if the conditions are matched.
		
		• **deny**—Denies access when there is an address match.
		
		• **remark**—Comment that describes the access list entry, up to 100 characters long.
		
		• *source*—Number of the network or host from which the packet is being sent. There are three ways to specify the source:
		
		  – *hostname*—Use the name of the host machine.
		
		  – *A.B.C.D*—Use 32-bit quantity in four-part, dotted-decimal format.
		
		  – *any*—Use the *any* keyword as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255.
		
		• *source-wildcard*—Wildcard bits to be applied to the source. There are two ways to specify the source wildcard:
		
		  – Use a 32-bit quantity in four-part, dotted-decimal format. Place 1s in the bit positions you want to ignore.
		
		  – Use the *any* keyword as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255.
		
		• *log*—Causes an informational logging message about the packet that matches the entry to be sent to the console. (The level of messages logged to the console is controlled by the **logging console** command.)

Configuring IP-Access-List Security on the Gatekeeper

To enable a gatekeeper to use an IP access list to perform tokenless call authorization, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **security acl answerarq** *access-list-number*

3. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	**gatekeeper** **Example:** Router(config)# gatekeeper	Enters gatekeeper configuration mode.
Step 2	**security acl answerarq** *access-list-number* **Example:** Router(config-gk)# security acl answerarq 20	Instructs the gatekeeper to use an IP access list—also known as an access control list (ACL)—to verify calls. Calls received from endpoints listed in the ACL are processed by the gatekeeper regardless of whether they contain IZCTs or CATs in the ARQ message from the endpoint. Rather than sending a Location Reject (LRJ) message for calls without tokens from these endpoints, the gatekeeper sends an admission confirm (ACF) message and accepts the calls.
Step 3	**exit** **Example:** Router(config-gk)# exit	Exits the current mode.

Configuring E.164 Interzone Routing

This section contains the following information:

Information About E.164 Interzone Routing

You can configure interzone routing using E.164 addresses.

Two types of address destinations are used in H.323 calls. You can specify a destination using either an H.323-ID address (a character string) or an E.164 address (a string that contains telephone keypad characters). The way in which interzone calls are routed depends on the type of address being used.

When using H.323-ID addresses, interzone routing is handled through the use of domain names. For example, to resolve the domain name bob@cisco.com, the source endpoint gatekeeper finds the gatekeeper for cisco.com and sends it the location request for the target address bob@cisco.com. The destination gatekeeper looks in its registration database, sees bob registered, and returns the appropriate IP address to get to bob.

When using E.164 addresses, call routing is handled through zone prefixes and gateway-type prefixes, also referred to as technology prefixes. The zone prefixes, which are typically area codes, serve the same purpose as domain names in H.323-ID address routing. Unlike domain names, however, more than one zone prefix can be assigned to one gatekeeper, but the same prefix cannot be shared by more than one gatekeeper.

Use the **zone prefix** command to define gatekeeper responsibilities for area codes. The command can also be used to tell the gatekeeper which prefixes are in its own zones and which remote gatekeepers are responsible for other prefixes.

Note Area codes are used as an example in this section, but a zone prefix need not be an area code. It can be a country code, an area code plus local exchange (NPA-NXX), or any other logical hierarchical partition.

The following sample command shows how to configure a gatekeeper with the knowledge that zone prefix 212....... (that is, any address beginning with area code 212 and followed by seven arbitrary digits) is handled by gatekeeper gk-ny:

```
my-gatekeeper(config-gk)# zone prefix gk-ny 212.......
```

When my-gatekeeper is asked to admit a call to destination address 2125551111, it knows to send the location request to gk-ny.

However, once the query gets to gk-ny, gk-ny still needs to resolve the address so that the call can be sent to its final destination. There could be an H.323 endpoint that has registered with gk-ny with that E.164 address, in which case gk-ny would return the IP address for that endpoint. However, it is more likely that the E.164 address belongs to a non-H.323 device, such as a telephone or an H.320 terminal.

Because non-H.323 devices do not register with gatekeepers, gk-ny has no knowledge of which device the address belongs to or which type of device it is, so the gatekeeper cannot decide which gateway should be used for the *hop off* to the non-H.323 device. (The term *hop off* refers to the point at which the call leaves the H.323 network and is destined for a non-H.323 device.)

Note The number of zone prefixes defined for a directory gatekeeper that is dedicated to forwarding LRQs, and not for handling local registrations and calls, should not exceed 10,000; 4 MB of memory must be dedicated to describing zones and zone prefixes to support this maximum number of zone prefixes. The number of zone prefixes defined for a gatekeeper that handles local registrations and calls should not exceed 2000.

To enable the gatekeeper to select the appropriate hop-off gateway, use the **gw-type-prefix** command to configure technology or gateway-type prefixes. Select technology prefixes to denote different types or classes of gateways. The gateways are then configured to register with their gatekeepers using these technology prefixes.

For example, voice gateways might register with technology prefix 1#, and H.320 gateways might register with technology prefix 2#. If there are several gateways of the same type, configure them to register with the same prefix type. By having them register with the same prefix type, the gatekeeper treats the gateways as a pool out of which a random selection is made whenever a call for that prefix type arrives. If a gateway can serve more than one type of hop-off technology, it can register more than one prefix type with the gatekeeper.

Callers must identify the type of gateway by prepending the appropriate technology prefix for that gateway type to the destination address. For example, callers might request 1#2125551111 if they know that address 2125551111 is for a telephone and that the technology prefix for voice gateways is 1#. The voice gateway is configured with a dial peer (using the **dial-peer** command) so that when the gateway receives the call for 1#2125551111, it strips off the technology prefix 1# and bridges the next leg of the call to the telephone at 2125551111.

In cases in which the call scenario is as shown in Figure 4, voice-gw1 can be configured to prepend the voice technology prefix 1# so that the use of technology prefixes is completely transparent to the caller.

Figure 4   *Call Scenario*
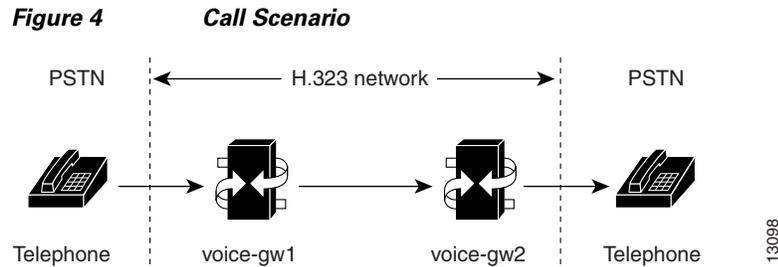
Additionally, in using the **gw-type-prefix** command, a particular gateway-type prefix can be defined as the default gateway type to be used for addresses that cannot be resolved. It also forces a technology prefix to always hop off in a particular zone.

If the majority of calls hop off on a particular type of gateway, the gatekeeper can be configured to use that type of gateway as the default type so that callers no longer have to prepend a technology prefix on the address. For example, if voice gateways are mostly used in a network, and all voice gateways have been configured to register with technology prefix 1#, the gatekeeper can be configured to use 1# gateways as the default technology if the following command is entered:

```
Router(config-gk)# gw-type-prefix 1# default-technology
```

Now a caller no longer needs to prepend 1# to use a voice gateway. Any address that does not contain an explicit technology prefix is routed to one of the voice gateways that registered with 1#.

With this default technology definition, a caller could ask the gatekeeper for admission to 2125551111. If the local gatekeeper does not recognize the zone prefix as belonging to any remote zone, it routes the call to one of its local (1#) voice gateways so that the call hops off locally. However, if it knows that gk-ny handles the 212 area code, it can send a location request for 2125551111 to gk-ny. This requires that gk-ny also be configured with some default gateway type prefix and that its voice gateways be registered with that prefix type.

> ✎
> **Note** For ease of maintenance, the same prefix type should be used to denote the same gateway type in all zones under your administration.

Also, with the **gw-type-prefix** command, a hop off can be forced to a particular zone. When an endpoint or gateway makes a call-admission request to its gatekeeper, the gatekeeper determines the destination address by first looking for the technology prefix. When that is matched, the remaining string is compared against known zone prefixes. If the address is determined to be a remote zone, the entire address, including technology and zone prefixes, is sent to the remote gatekeeper in a location request. That remote gatekeeper then uses the technology prefix to decide on which of its gateways to hop off. In other words, the zone prefix (defined using the **zone prefix** command) determines the routing to a zone, and once there, the technology prefix (defined using the **gw-type-prefix** command) determines the gateway to be used in that zone. The zone prefix takes precedence over the technology prefix.

This behavior can be overridden by associating a forced hop-off zone with a particular technology prefix. Associating a forced hop-off zone with a particular technology prefix forces the call to the specified zone, regardless of what the zone prefix in the address is. As an example, you are in the 408 area code and want callers to the 212 area code in New York to use H.323-over-IP and hop off there because it saves on costs. However, the only H.320 gateway is in Denver. In this example, calls to H.320 endpoints must be forced to hop off in Denver, even if the destination H.320 endpoint is in the 212 area code. The forced hop-off zone can be either a local zone (that is, one that is managed by the local gatekeeper) or a remote zone.

Configuring a Dialing Prefix for Each Gateway

To configure a dialing prefix for each gateway, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **zone local** *gatekeeper-name domain-name* [*ras-ip-address*]
3. **zone prefix** *gatekeeper-name e164-prefix* [**gw-priority** *pri-0-to-10 gw-alias* [*gw-alias, ...*]]
4. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** Router(config)# gatekeeper	Enters gatekeeper configuration mode.
Step 2	**zone local** *gatekeeper-name domain-name* [*ras-ip-address*] **Example:** Router(config-gk)# zone local gatekeeper1 domain1	Specifies a zone controlled by a gatekeeper.
Step 3	**zone prefix** *gatekeeper-name e164-prefix* [**gw-priority** *pri-0-to-10 gw-alias* [*gw-alias, ...*]] **Example:** Router(config-gk)# zone prefix localgk 415....... gw-priority 10 gw1 gw2	Adds a prefix to the gatekeeper zone list. To remove knowledge of a zone prefix, use the **no** form of this command with the gatekeeper name and prefix. To remove the priority assignment for a specific gateway, use the **no** form of this command with the **gw-priority** keyword. To put all of your gateways in the same zone, use the **gw-priority** keyword as described below.
Step 4	**exit** **Example:** Router(config-gk)# exit	Exits the current mode.

To put all of your gateways in the same zone, use the **gw-priority** keyword and specify which gateways are used for calling different area codes. For example:

```
zone local localgk xyz.com
zone prefix localgk 408.......
zone prefix localgk 415....... gw-priority 10 gw1 gw2
zone prefix localgk 650....... gw-priority 0 gw1
```

The above commands accomplish the following:

- Domain xyz.com is assigned to gatekeeper localgk.

- Prefix 408 is assigned to gatekeeper localgk, and no gateway priorities are defined for it; therefore, all gateways registering to localgk can be used equally for calls to the 408 area code. No special gateway lists are built for the 408 prefix; a selection is made from the master list for the zone.

- The prefix 415 is added to gatekeeper localgk, and priority 10 is assigned to gateways gw1 and gw2.

- Prefix 650 is added to gatekeeper localgk, and priority 0 is assigned to gateway gw1.

- A priority 0 is assigned to gateway gw1 to exclude it from the gateway pool for prefix 650. When gw2 registers with gatekeeper localgk, it is added to the gateway pool for each prefix as follows:

    - For gateway pool for 415, gateway gw2 is set to priority 10.

    - For gateway pool for 650, gateway gw2 is set to priority 5.

Configuring Gatekeeper Interaction with External Applications

There are two ways of configuring the gatekeeper for interaction with an external application. You can configure a port number where the gatekeeper listens for dynamic registrations from applications. Using this method, the application connects to the gatekeeper and specifies the trigger conditions in which it is interested.

The second method involves using the command-line interface to statically configure the information about the application and its trigger conditions, in which case the gatekeeper initiates a connection to the external application.

Cisco provides a Gatekeeper Transaction Message Protocol (GKTMP) server and commands to configure the gatekeeper to communicate with the server using GKTMP messages.

Note For configuration information, see *VoIP Gatekeeper Trunk and Carrier Based Routing Enhancements* at http://www.cisco.com/en/US/docs/ios/12_2t/12_2t11/feature/guide/ftgkrenb.html

This section contains the following information:

Configuring Gatekeeper-to-GKTMP Server Flow Control

You can set a timeout value for responses from the GKTMP server to the gatekeeper. Thegatekeeper measures the average time taken by the server to process each transaction. If the time period for processing reaches 80 percent of the configured timeout value, the server is marked as unavailable. Thegatekeeper routes transactions bound for this server to alternate servers if they are available. If no alternate servers are available, thegatekeeper handles the calls.

Configuring Flow Control

To configure server flow control, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**

2. **server flow-control** [**onset** *value*] [**abatement** *value*] [**qcount** *value*]

3. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`server flow-control` [`onset` *value*] [`abatement` *value*] [`qcount` *value*] **Example:** `Router(config-gk)# server flow-control onset 50 abatement 25 qcount 100`	Enables flow control and resets all the thresholds to default. Keywords and arguments are as follows: • **onset** *value*—Percentage of the server timeout value that is used to mark the server as usable or unusable. Range: 1 to 100. Default: 80. • **abatement** *value*—Percentage of the server timeout value that is used to mark the server as unusable or usable. Range: 1 to 100; cannot be greater than or equal to the onset value. Default: 50. For example, if the server timeout value is 3 seconds, **onset** *value* is 50, and **abatement** *value* is 40, when the average response time from the server to the GKTMP reaches 1.5 seconds (the onset percentage of the server timeout value), the server is marked as unusable. During the period that the server is marked as unusable, REQUEST ALV messages are still sent to the unusable server. When the response time is lowered to 1.2 seconds (the abatement percentage of the timeout value), the server is marked usable again and the GKTMP resumes sending messages to the server. • **qcount** *value*—Threshold length of the outbound queue on the GK. The queue contains messages waiting to be transmitted to the server. The TCP socket between the GK and GKTMP server queues messages if it has too many to transmit. If the count of outbound queue length on the server reaches the qcount value, the server is marked unusable. Range:1 to 2000. Default: 400.

Step 3	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Verifying Flow Control

Step 1   **show running-config**

Use this command to verify that server flow-control appears in the output.

```
Router# show running-config

Building configuration...
```

```
Current configuration : 1055 bytes
!
version 12.2
no service single-slot-reload-enable
service timestamps debug datetime msec
service timestamps log uptime
no service password-encryption
!
hostname snet-3660-3
!
.
.
.
gatekeeper
 zone local snet-3660-3 cisco.com
 zone remote snet-3660-2 cisco.com 209.165.200.225 1719
 zone prefix snet-3660-2 408*
 lrq forward-queries
 no use-proxy snet-3660-3 default inbound-to terminal
 no use-proxy snet-3660-3 default outbound-from terminal
 no shutdown
 server registration-port 8000
 server flow-control
!
.
.
.
```

Step 2   **show gatekeeper status**

Use this command to view the status of the GKTMP Interface Resiliency Enhancement feature.

The following example shows that the GKTMP Interface Resiliency Enhancement feature is enabled:

```
Router# show gatekeeper status

Gatekeeper State: UP
    Load Balancing:    DISABLED
    Flow Control:      ENABLED
    Zone Name:         snet-3660-3
    Accounting:        DISABLED
    Endpoint Throttling:        DISABLED
    Security:          DISABLED
    Maximum Remote Bandwidth:            unlimited
    Current Remote Bandwidth:          0 kbps
    Current Remote Bandwidth (w/ Alt GKs): 0 kbps
```

Step 3   **show gatekeeper servers**

Use this command to view the server statistics, including timeout encountered, average response time, and server status.

```
Router# show gatekeeper servers

         GATEKEEPER SERVERS STATUS
         =========================

Gatekeeper Server listening port: 8250
Gatekeeper Server timeout value: 30 (100ms)
GateKeeper GKTMP version: 3.1

Gatekeeper-ID: Gatekeeper1
-----------------------
  RRQ  Priority: 5
```

```
Server-ID: Server43
Server IP address: 209.165.200.254:40118
Server type: dynamically registered
Connection Status: active
Trigger Information:
  Trigger unconditionally

Server Statistics:
REQUEST RRQ Sent=0
RESPONSE RRQ Received = 0
RESPONSE RCF Received = 0
RESPONSE RRJ Received = 0
Timeout encountered=0
Average response time(ms)=0
Server Usable=TRUE
```

Setting the Retry Timer for Failed GKTMP Server Connections

Configuring the Timer

To configure faster reconnection to a GKTMP server when its TCP connection fails, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **timer server retry** *seconds*
3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** `Router(config)# gatekeeper`	Enters gatekeeper configuration mode.

	Command	Purpose
Step 2	`timer server retry` *seconds* **Example:** Router(config-gk)# timer server retry 20	Sets the retry timer for failed GKTMP server connections, in seconds. After the gatekeeper detects that its GKTMP server TCP connection has failed, the gatekeeper retries the server based on the setting of this timer, and keep retrying until the connection is established. Range: 1 to 300. Default: 30. **Note** This timer applies only to deployments where static triggers are used between the gatekeeper and the GKTMP server. If dynamic triggers are used, the server must determine and implement a retry mechanism if the TCP connection to the gatekeeper fails.
Step 3	`exit` **Example:** Router(config-gk)# exit	Exits the current mode.

Verifying the Timer

Step 1   **show gatekeeper servers**

Use this command to verify the retry timer for failed server connections.

```
Router# show gatekeeper servers

        GATEKEEPER SERVERS STATUS
        =========================

Gatekeeper Server listening port:0
Gatekeeper Server response timeout value:30 (100ms)
Gatekeeper Server connection retry timer value:30 (sec)
Gatekeeper GKTMP version:4.1
```

Configuring Registration and Call Rejection

Configuring Registration and Call Rejection

To configure the gatekeeper to reject new registrations or calls when it is unable to reach the GKTMP server because the TCP connection between the gatekeeper and GKTMP server is down, use these commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **server absent reject** {**rrq** | **arq**}
3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** Router(config)# gatekeeper	Enters gatekeeper configuration mode.
Step 2	**server absent reject {rrq	arq}** **Example:** Router(config-gk)# server absent reject rrq	Configures the gatekeeper to reject new registrations or calls when it is unable to reach the GKTMP server because the TCP connection between gatekeeper and server is down. If multiple GKTMP servers are configured, the gatekeeper tries all of them and rejects registrations or calls only if none of the servers responds. Keywords are as follows: • **rrq**—Reject registrations by RRQ messages • **arq**—Reject calls by admission request (ARQ) messages You can also use this feature for security or service denial if a connection with the server is required to complete a registration. Default: this feature is not enabled; the gatekeeper does not reject new registrations or calls. **Note** This command assumes that RRQ and ARQ triggers are used between the gatekeeper and GKTMP server.
Step 3	**exit** **Example:** Router(config-gk)# exit	Exits the current mode.

Verifying Registration and Call Rejection

Step 1 **show running-config**

Use this command to verify that the gatekeeper is rejecting new registrations when unable to reach the GKTMP server.

```
Router# show running-config
.
.
.
gw-type-prefix 1#* default-technology
gw-type-prefix 9#* gw ipaddr 1.1.1.1 1720
no shutdown
server absent reject rrq
.
.
.
```

Use this command to verify that the gatekeeper is rejecting new calls when unable to reach the GKTMP server, use the command.

```
Router# show running-config
```

```
.
.
.
gw-type-prefix 1#* default-technology
gw-type-prefix 9#* gw ipaddr 1.1.1.1 1720
no shutdown
server absent reject arq
.
.
.
```

Configuring Gatekeeper Proxied Access

By default, a gatekeeper offers the IP address of the local proxy when queried by a remote gatekeeper (synonymous with remote zone) or the border element. This is considered proxied access.

Note   The **use-proxy** command replaces the **zone access** command. The **use-proxy** command, configured on a local gatekeeper, affects only the use of proxies for incoming calls (that is, it does not affect the use of local proxies for outbound calls). When originating a call, a gatekeeper uses a proxy only if the remote gatekeeper offers a proxy at the remote end. A call between two endpoints in the same zone is always a direct (nonproxied) call.

Configuring Access

To configure a proxy for inbound calls from remote zones or the border element to gateways in its local zone and to configure a proxy for outbound calls from gateways in its local zone to remote zones or the border element, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**

2. **use-proxy** *local-zone-name* {**default** | **h323-annexg** | **remote-zone** *remote-zone-name*} {**inbound-to** | **outbound-from**} {**gateway** | **terminal**}

3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** Router(config)# gatekeeper	Enters gatekeeper configuration mode.
Step 2	**use-proxy** *local-zone-name* {**default** | **h323-annexg** | **remote-zone** *remote-zone-name*} {**inbound-to** | **outbound-from**} {**gateway** | **terminal**} **Example:** Router(config-gk)# use-proxy zonename default inbound-to gateway	Enables proxy communications for calls between local and remote zones. Keywords and arguments are as follows: • *local-zone-name*—Name or zone name of the gatekeeper, which is usually the fully domain-qualified host name of the gatekeeper. For example, if the domain name is cisco.com, the gatekeeper name might be gk1.cisco.com. However, if the gatekeeper is controlling multiple zones, the name of the gatekeeper for each zone should be a unique string that has a mnemonic value. • **default**—Default proxy policy for all calls that are not defined by a **use-proxy** command that includes the **remote-zone** keyword. • **h323-annexg**—Proxy policy for calls to or from the H.225 AnnexG border element co-located with the gatekeeper. • **remote-zone** *remote-zone-name*—Proxy policy for calls to or from a specific remote gatekeeper or zone. • **inbound-to**—Applies the proxy policy to calls that are inbound to the local zone from a remote zone. Each **use-proxy** command defines the policy for only one direction. • **outbound-from**—Applies the proxy policy to calls that are outbound from the local zone to a remote zone. Each **use-proxy** command defines the policy for only one direction. • **gateway**—Type of local device to which the policy applies. Applies the policy only to local gateways. • **terminal**—Type of local device to which the policy applies. Applies the policy only to local terminals.
Step 3	**exit** **Example:** Router(config-gk)# exit	Exits the current mode.

Verifying Access

Step 1   **show gatekeeper zone status**

Use this command to see information about the configured gatekeeper proxies and gatekeeper zone information (as shown in the following output).

```
Router# show gatekeeper zone status

                      GATEKEEPER ZONES
                      ================

GK name       Domain Name  RAS Address      PORT  FLAGS MAX-BW  CUR-BW
                                                        (kbps)  (kbps)
-------       -----------  -----------      ----  ----- ------  ------
sj.xyz.com    xyz.com      10.0.0.9 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 H.225 AnnexG border element :
      to terminals in local zone germany.xyz.com :use proxy
      to gateways in local zone germany.xyz.com  :do not use proxy
    outbound calls to H.225 AnnexG border element :
      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 :
      from terminals in local zone sj.xyz.com :do not use proxy
      from gateways in local zone sj.xyz.com  :do not use proxy
tokyo.xyz.co xyz.com        172.21.139.89    1719  RS           0
milan.xyz.co xyz.com        172.16.00.00     1719  RS           0
```

Configuring a Forced Disconnect on a Gatekeeper

Configuring Disconnect

To force a disconnect on a gatekeeper, use the following command in privileged EXEC mode.

SUMMARY STEPS

1. **clear h323 gatekeeper call** {**all** | **local-callID** *local-call-id*}

DETAILED STEPS

	Command	Purpose
Step 1	`clear h323 gatekeeper call {all	local-callID` `local-call-id}`    **Example:**   `Router# clear h323 gatekeeper call all`	Forces a disconnect on a specific call or on all calls currently active on this gatekeeper. Keywords and arguments are as follows: • **all**—All active calls associated with this gatekeeper • **local-callID** *local-call-id*—Local call identification number (CallID) that identifies the call to be disconnected

---

Note To force a particular call to disconnect (as opposed to all active calls on the H.323 gateway), use the local call identification number (CallID) to identify that specific call. Find the local CallID number for a specific call by using the **show gatekeeper calls** command; the ID number is displayed in the LocalCallID column.

Verifying Disconnect

Step 1 **show gatekeeper calls**

Use this command to show the status of each ongoing call that a gatekeeper is aware of. If you have forced a disconnect either for a particular call or for all calls associated with a particular H.323 gatekeeper, the system does not display information about those calls.

```
router# show gatekeeper calls

Total number of active calls =1
                    Gatekeeper Call Info
                    ====================
LocalCallID                    Age (secs)      BW
12-3339                        94              768 (Kbps)
 Endpt(s): Alias    E.164Addr   CallSignalAddr   Port   RASSignalAddr   Port
  src EP: epA                   10.0.0.11        1720   10.0.0.11       1700
  dst EP: epB2zoneB.com
  src PX: pxA                   10.0.0.1         1720   10.0.0.11       24999
  dst PX: pxB                   172.21.139.90    1720   172.21.139.90   24999
```

Configuring an H.323 Proxy Server

The following sections describes how the proxy feature can be used in an H.323 network.

When terminals signal each other directly, they must have direct access to each other's addresses. This exposes an attacker to key information about a network. When a proxy is used, the only addressing information that is exposed to the network is the address of the proxy; all other terminal and gateway addresses are hidden.

There are several ways to use a proxy with a firewall to enhance network security. The configuration to be used depends on how capable the firewall is of handling the complex H.323 protocol suite. Each of the following sections describes a common configuration for using a proxy with a firewall:

Proxy Inside the Firewall

H.323 is a complex, dynamic protocol that consists of several interrelated subprotocols. During H.323 call setup, the ports and addresses released with this protocol require a detailed inspection as the setup progresses. If the firewall does not support this dynamic access control based on the inspection, a proxy can be used just inside the firewall. The proxy provides a simple access control scheme, as illustrated in Figure 5.

Figure 5   *Proxy Inside the Firewall*
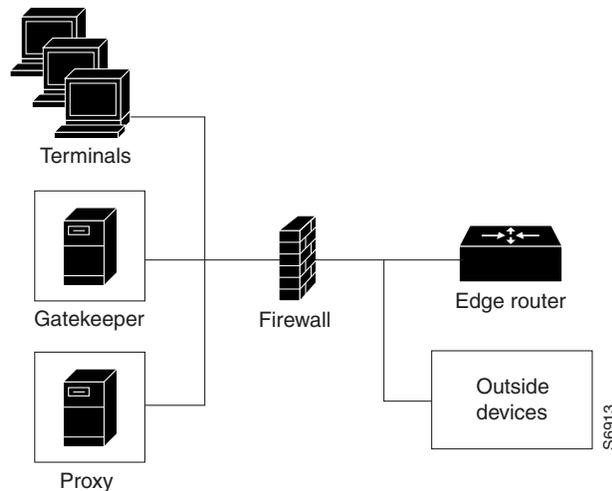
Because the gatekeeper (using RAS) and the proxy (using call setup protocols) are the only endpoints that communicate with other devices outside the firewall, it is simple to set up a tunnel through the firewall to allow traffic destined for either of these two endpoints to pass through.

Proxy in Co-Edge Mode

If H.323 terminals exist in an area with local interior addresses that must be translated to valid exterior addresses, the firewall must be capable of decoding and translating all addresses passed in the various H.323 protocols. If the firewall is not capable of this translation task, a proxy may be placed next to the firewall in a co-edge mode. In this configuration, interfaces lead to both inside and outside networks. (See Figure 6.)

Figure 6　　　*Proxy in Co-Edge Mode*
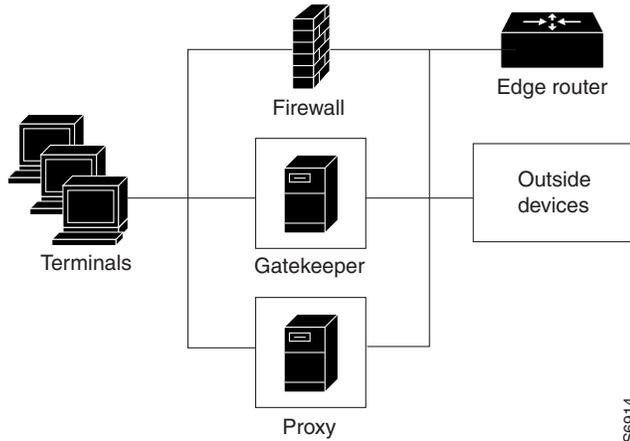
In co-edge mode, the proxy can present a security risk. To avoid exposing a network to unsolicited traffic, configure the proxy to route only proxied traffic. In other words, the proxy routes only H.323 protocol traffic that is terminated on the inside and then repeated to the outside. Traffic that moves in the opposite direction can be configured this way as well.

Proxy Outside the Firewall

To place the proxy and gatekeeper outside the firewall, two conditions must exist. First, the firewall must support H.323 dynamic access control. Second, Network Address Translation (NAT) must not be in use.

If NAT is in use, each endpoint must register with the gatekeeper for the duration of the time it is online. This quickly overwhelms the firewall because a large number of relatively static, internal-to-external address mappings need to be maintained.

If the firewall does not support H.323 dynamic access control, the firewall can be configured with static access lists that allow traffic from the proxy or gatekeeper through the firewall. This can present a security risk if an attacker can *spoof*, or simulate, the IP addresses of the gatekeeper or proxy and use them to attack the network. Figure 7 illustrates proxy outside the firewall.

Figure 7   *Proxy Outside the Firewall*
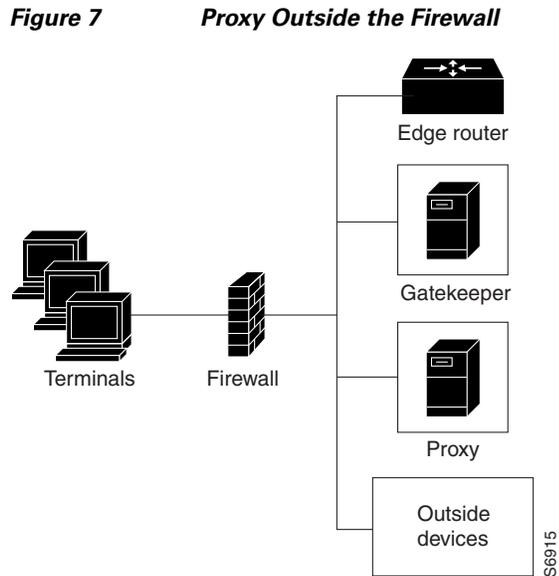
Proxy and NAT

When a firewall is providing NAT between an internal and an external network, proxies may allow H.323 traffic to be handled properly, even in the absence of a firewall that can translate addresses for H.323 traffic. Table 2 and Table 3 provide guidelines for proxy deployment for networks that use NAT.

Table 2   *Guidelines for Networks That Use NAT*

For Networks Using NAT	Firewall with H.323 NAT	Firewall Without H.323 NAT
Firewall with dynamic access control	Gatekeeper and proxy inside the firewall	Co-edge gatekeeper and proxy
Firewall without dynamic access control	Gatekeeper and proxy inside the firewall, with static access lists on the firewall	Co-edge gatekeeper and proxy

Table 3   *Guidelines for Networks That Do Not Use NAT*

For Networks Not Using NAT	Firewall with H.323. NAT	Firewall Without H.323 NAT
Firewall with Dynamic Access Control	Gatekeeper and proxy inside the firewall Gatekeeper and proxy outside the firewall	Gatekeeper and proxy inside the firewall Gatekeeper and proxy outside the firewall
Firewall Without Dynamic Access Control	Gatekeeper and proxy inside the firewall, with static access lists on the firewall	Gatekeeper and proxy inside the firewall, with static access lists on the firewall

Configuring Quality of Service

This section contains the following information:

Prerequisites for QoS

- The proxy is not capable of modifying the Quality of Service (QoS) between the terminal and itself. To achieve the best overall QoS, ensure that terminals are connected to the proxy using a network that intrinsically has good QoS. In other words, configure a path between a terminal and proxy that provides good bandwidth, delay, and packet-loss characteristics without the terminal needing to request special QoS. A high-bandwidth LAN works well for this.

Information About QoS

QoS enables complex networks to control and predictably service a variety of applications. QoS expedites the handling of mission-critical applications while sharing network resources with noncritical applications. QoS also ensures available bandwidth and minimum delays required by time-sensitive multimedia and voice applications. In addition, QoS gives network managers control over network applications, improves cost-efficiency of WAN connections, and enables advanced differentiated services. QoS technologies are elemental building blocks for other Cisco IOS-enabling services such as its H.323-compliant gatekeeper. Overall call quality can be improved dramatically in the multimedia network by using pairs of proxies between regions of the network where QoS can be requested.

RSVP and IP Precedence

When two H.323 terminals communicate directly, the resulting call quality can range from good (for high-bandwidth intranets) to poor (for most calls over the public network). As a result, deployment of H.323 is almost always predicated on the availability of some high-bandwidth, low-delay, low-packet-loss network that is separate from the public network or that runs overlaid with the network as a premium service and adequate QoS.

Adequate QoS usually requires terminals that are capable of signaling such premium services. There are two major ways to achieve such signaling:

- Resource Reservation Protocol (RSVP) to reserve flows having adequate QoS based on the media codecs of H.323 traffic
- IP precedence bits to signal that the H.323 traffic is special and that it deserves higher priority

Unfortunately, the vast majority of H.323 terminals cannot achieve signaling in either of these ways. The proxy can be configured to use any combination of RSVP and IP precedence bits.

Note For more information on RSVP, synchronous reservation timers, and slow connect, see *Quality of Service for Voice* at
http://www.cisco.com/en/US/docs/ios/qos/configuration/guide/15_0/qos_15_0_book.html

Application-Specific Routing

To achieve adequate QoS, a separate network may be deployed that is partitioned away from the standard data network. The proxy can take advantage of such a partitioned network using a feature known as application-specific routing (ASR).

Application-specific routing is simple. When the proxy receives outbound traffic, it directs traffic to an interface that is connected directly to the QoS network. The proxy does not send the traffic using an interface that is specified for the regular routing protocol. Similarly, inbound traffic from other proxies is received on the interface that is connected to the QoS network. This is true if all these other proxies around the QoS network use ASR in a consistent fashion. ASR then ensures that ordinary traffic is not routed into the QoS network by mistake.

Implementation of ASR ensures the following:

- Each time a connection is established with another proxy, the proxy automatically installs a host route pointing at the interface designated for ASR.

- The proxy is configured to use a loopback interface address. The proxy address is visible to both the ASR interface and all regular interfaces, but there are no routes established between the loopback interface and the ASR interface. This ensures that no non-H.323 traffic is routed through the ASR interface.

✎
Note ASR is not supported on Frame Relay or ATM interfaces for the Cisco MC3810.

Configuring QoS Using a Multimedia Backbone

The examples in this section illustrate a separate multimedia backbone network dedicated to transporting only H.323 traffic. The closed functionality of the H.323 proxy is necessary for creating this type of backbone. Place a closed H.323 proxy on each edge of the multimedia backbone to achieve the following goals:

- The proxy directs all inter-proxy H.323 traffic, including Q.931 signaling, H.245, and media stream, to the multimedia backbone.

- The proxy shields the multimedia backbone so that routers on edge networks and other backbone networks are not aware of its existence. In this way, only H.323-compliant packets can access or traverse the multimedia backbone.

- The proxy drops any unintended non-H.323 packets that attempt to access the multimedia backbone.

Figure 8 illustrates a network that has a multimedia backbone. A gatekeeper (not shown) in the edge network (zone) directs all out-of-zone H.323 calls to the closed proxy on the edge of that network. The closed proxy forwards this traffic to the remote zone through the multimedia backbone. A closed proxy and the edge router may reside in the same router or they may be in separate routers, as shown in the figure.

Figure 8   *Sample Network with Multimedia Backbone*
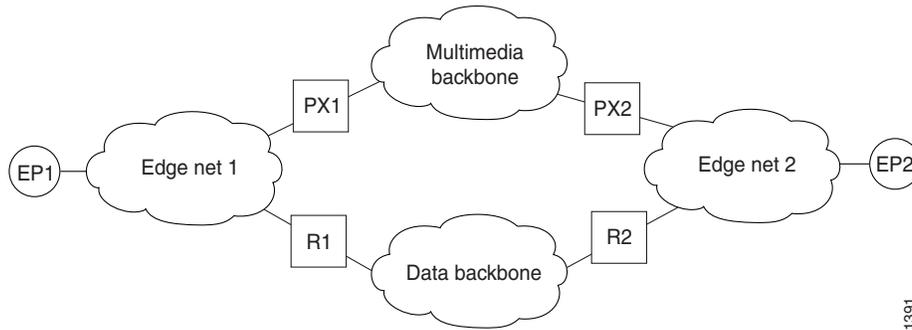
Enabling the Proxy to Forward H.323 Packets

To enable the proxy to forward H.323 packets received from the edge network to the multimedia backbone, designate the interface that connects the proxy to the multimedia backbone to the ASR interface by entering the **h323 asr** command in interface configuration mode. Enabling the proxy to forward H.323 packets satisfies the first goal identified earlier in this section.

Because the proxy terminates two call legs of an H.323 call and bridges them, any H.323 packet that traverses the proxy has the proxy address either in its source field or in its destination field.

To prevent problems that can occur in proxies that have multiple IP addresses, designate only one interface to be the proxy interface by entering the **h323 interface** command in interface configuration mode. Then all H.323 packets that originate from the proxy has the address of this interface in their source fields, and all packets that are destined to the proxy has the address of this interface in their destination fields.

Figure 8 illustrates that all physical proxy interfaces belong either to the multimedia network or to the edge network. These two networks must be isolated from each other for the proxy to be closed; however, the proxy interface must be addressable from both the edge network and the multimedia network. For this reason, a loopback interface must be created on the proxy and configured to the proxy interface.

It is possible to make the loopback interface addressable from both the edge network and the multimedia network without exposing any physical subnets on one network to routers on the other network. Only packets that originate from the proxy or packets that are destined to the proxy can pass through the proxy interface to the multimedia backbone in either direction. All other packets are considered unintended packets and are dropped. This can be achieved by configuring access control lists (ACLs) so that the closed proxy acts like a firewall that only allows H.323 packets to pass through the ASR interface. This satisfies the second goal identified earlier in this section, which is to ensure that only H.323-compliant packets can access or traverse the multimedia backbone.

Isolating the Multimedia Network

The last step is to configure the network so that non-H.323 traffic never attempts to traverse the multimedia backbone and so that it never risks being dropped by the proxy. This is achieved by completely isolating the multimedia network from all edge networks and from the data backbone and by configuring routing protocols on the various components of the networks.

The example provided in Figure 8 requires availability of six IP address classes, one for each of the four autonomous systems and one for each of the two loopback interfaces. Any Cisco-supported routing protocol can be used on any of the autonomous systems, with one exception: Routing Information

Protocol (RIP) cannot be configured on two adjacent autonomous systems because this protocol does not include the concept of an autonomous system. The result would be the merging of the two autonomous systems into one.

If the number of IP addresses are scarce, use subnetting, but the configuration can get complicated. In this case, only the Enhanced IGRP, Open Shortest Path First (OSPF), and RIP Version 2 routing protocols, which allow variable-length subnet masks (VLSMs), can be used.

Assuming these requirements are met, configure the network illustrated in Figure 8 as follows:

- Configure each of the four networks as a separate routing autonomous system and do not redistribute routes between the multimedia backbone and any other autonomous system.

- Create a loopback interface on the proxy and configure it to be the proxy interface. That way no subnets of the multimedia backbone are exposed to the edge network, or the other way around.

- To ensure that the address of the loopback interface does not travel outside the edge network, configure the appropriate distribution list on the edge router that connects the edge network to the data backbone. Configuring the appropriate distribution list guarantees that any ongoing H.323 call is interrupted if the multimedia backbone fails. Otherwise, H.323 packets that originate from one proxy and that are destined to another proxy might discover an alternate route using the edge networks and the data backbone.

In some topologies, the two edge networks and the data backbone may be configured as a single autonomous system, but it is preferable to separate them as previously described because they are different networks with different characteristics.

Configuring QoS on a Proxy Without ASR

To start the proxy without application-specific routing (ASR), start the proxy and then define the H.323 name, zone, and QoS parameters on the interface whose IP address the proxy uses. To start the proxy without ASR, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **proxy h323**
2. **interface** *type number* [*nametag*]
3. **h323 interface** [*port*]
4. **h323 h323-id** *h323-id*
5. **h323 gatekeeper** [**id** *gatekeeper-id*] {**ipaddr** *ip-address* [*port*] | **multicast**}
6. **h323 qos** {*ip-precedence* | **rsvp** {**controlled-load** | **guaranteed-qos**}}
7. **ip route-cache** [**cbus**] **same-interface** [**flow**] **distributed**
8. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`proxy h323`
	
	Example:
	`Router(config)# proxy h323`
		Starts the proxy.

Step 2	`interface` *type number* [*nametag*] **Example:** `Router(config)# interface serial 0`	Enters interface configuration mode for a particular interface or subinterface. Keywords and arguments are platform dependent; for more information, see the IOS interface command reference listed in the "Additional References" section on page 256.

Step 3	`h323 interface [`*port*`]`
	
	Example:
	`Router(config-if)# h323 interface 1`
		Selects an interface whose IP address is used by the proxy to register with the gatekeeper. The argument are as follows:
		
		• *port*—Port on which the proxy listens for incoming call setup requests. Range: 1 to 65356. Default:
		
		– 11720 in -isx- or -jsx- Cisco IOS images
		
		– 1720 in -ix- Cisco IOS images, which do not contain the VoIP gateway

Step 4	`h323 h323-id` *h323-id* **Example:** `Router(config-if)# h323 h323-id PX1@zone1.com`	Configures the proxy name. (More than one name may be configured if necessary.) The argument is as follows: • *h323-id*—Name of the proxy. We recommend that this be a fully qualified e-mail identification (ID), with the domain name being the same as that of its gatekeeper.
Step 5	`h323 gatekeeper [id` *gatekeeper-id*`] {ipaddr` *ip-address* `[`*port*`]	multicast}`  **Example:** `Router(config-if)# h323 gatekeeper ipaddr 10.0.0.0`	Specifies the gatekeeper associated with a proxy and controls how the gatekeeper is discovered. Keywords and arguments are as follows: • **id** *gatekeeper-id*—Gatekeeper name. Typically, this is a Domain Name System (DNS) name, but it can also be a raw IP address in dotted form. If this parameter is specified, gatekeepers that have either the default or the explicit flags set for the subnet of the proxy respond. If this parameter is not specified, only those gatekeepers with the default subnet flag respond. • **ipaddr** *ip-address* [*port*]—Gatekeeper discovery message is unicast to this address and, optionally, to the port specified. • **multicast**—Gatekeeper discovery message is multicast to the well-known Registration, Admission, and Status (RAS) multicast address and port.

	Command	Purpose
Step 6	**h323 qos** {*ip-precedence* | **rsvp** {**controlled-load** | **guaranteed-qos**}} **Example:** `Router(config-if)# h323 qos rsvp guaranteed-qos`	Enables QoS on the proxy. Keywords and arguments are as follows: • *ip-precedence*— Realtime Transport Protocol (RTP) streams set their IP precedence bits to the specified value • **rsvp** [**controlled-load**]—Controlled load class of service • **rsvp** [**guaranteed-qos**]—Guaranteed QoS class of service
Step 7	**ip route-cache [cbus] same-interface [flow] distributed** **Example:** `Router(config-if)# ip route-cache same-interface distributed`	Controls the use of high-speed switching caches for IP routing. Keywords are as follows: • **cbus**—Both autonomous switching and fast switching • **same-interface**—Fast-switching packets to back out through the interface on which they arrived • **flow**—The route switch processor (RSP) performs flow switching on the interface. • **distributed**—Versatile Interface Processor (VIP) distributed switching on the interface. This feature can be enabled on Cisco 7500 series routers with RSP and VIP controllers. If both the **ip route-cache flow** and the **ip route-cache distributed** command are configured, the VIP does distributed flow switching. If only the **ip route-cache distributed** command is configured, the VIP does distributed switching.
Step 8	**exit** **Example:** `Router(config-if)# exit`	Exits the current mode.

Configuring QoS on a Proxy with ASR

To enable ASR on the proxy, start the proxy and then define the H.323 name, zone, and QoS parameters on the loopback interface. Next, determine which interface is used to route the H.323 traffic and configure ASR on it. The ASR interface and all other interfaces must be separated so that routing information never travels from one to the other. There are two different ways to separate the ASR interface and all other interfaces:

• Use one type of routing protocol on the ASR interface and another on all the non-ASR interfaces. Include the loopback subnet in both routing domains.

• Set up two different autonomous systems, one that contains the ASR network and the loopback network and another that contains the other non-ASR networks and loopback network.

To ensure that the ASR interface and all other interfaces never route packets between each other, configure an access control list. (The proxy traffic is routed specially because it is always addressed to the loopback interface first and then translated by the proxy subsystem.)

ASR Enabled on the Proxy Using One Type of Routing Protocol

To start the proxy with ASR enabled on the proxy using one type of routing protocol on the ASR interface and another on all of the non-ASR interfaces, and with the loopback subnet included in both routing domains, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **proxy h323**

2. **interface** *type number* [*nametag*]

3. **ip address** *ip-address mask* [**secondary**]

4. **h323 interface** [*port*]

5. **h323 h323-id** *h323-id*

6. **h323 gatekeeper** [**id** *gatekeeper-id*] {**ipaddr** *ip-address* [*port*] | **multicast**}

7. **h323 qos** {*ip-precedence* | **rsvp** {**controlled-load** | **guaranteed-qos**}}

8. **interface** *type number* [*nametag*]

9. **h323 asr** [**bandwidth** *max-bandwidth*]

10. **ip address** *ip-address mask* [**secondary**]

11. **exit**

12. **interface** *type number* [*nametag*]

13. **ip address** *ip-address mask* [**secondary**]

14. **exit**

15. **router rip**

16. **network** *network-number*

17. **router igrp** *autonomous-system*

18. **network** *network-number*

19. **network** *loopback-addr*

20. **access-list** *access-list-number* {**permit** | **deny**} *source source-mask* [*destination destination-mask*] {**eq** | **neq**} [[*source-object*] [*destination-object*] [*identification*] **any**]

21. **interface** *type number* [*nametag*]

22. **ip access-group** {*access-list-number* | *access-list-name*} {**in** | **out**}

23. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`proxy h323`
	
	Example:
	`Router(config)# proxy h323`
		Starts the proxy.

Step 2	`interface` *type number* **[**`nametag`**]** **Example:** `Router(config)# interface loopback 3`	Enters loopback-interface configuration mode. Keywords and arguments are platform dependent; for more information, see the IOS interface command reference listed in the To configure a proxy with ASR enabled on the proxy using one type of routing protocol, set *type* to **loopback**. The **loopback** type specifies the software-only loopback interface that emulates an interface that is always up. It is a virtual interface supported on all platforms. The *number* argument is the number of the loopback interface that you want to create or configure. There is no limit on the number of loopback interfaces that you can create.
Step 3	`ip address` *ip-address mask* **[**`secondary`**]** **Example:** `Router(config-if)# ip address 192.168.0.0 225.225.225.0`	Sets a primary or secondary IP address for an interface. Keywords and arguments are as follows: • *ip-address—IP address.* • *mask—Mask for the associated IP subnet.* • **secondary**—*Configured address is a secondary IP address. Default: the configured address is the primary IP address.*

Step 4	`h323 interface [`*port*`]`
	
	Example:
	`Router(config-if)# h323 interface`
		Signals the proxy that this interface IP address is the one to use. The argument are as follows:
		
		• *port—Port on which the proxy listens for incoming call setup requests. Range: 1 to 65356. Default:*
		
		  – 11720 in -isx- or -jsx- Cisco IOS images
		
		  – 1720 in -ix- Cisco IOS images, which do not contain the VoIP gateway.

Step 5	`h323 h323-id` *h323-id* **Example:** `Router(config-if)# h323 h323-id PX1@zone1.com`	Configures the proxy name. (More than one name can be configured if necessary.) The argument is as follows: • *h323-id—Name of the proxy. We recommend that this be a fully qualified e-mail identification (ID), with the domain name being the same as that of its gatekeeper.*
Step 6	`h323 gatekeeper [id` *gatekeeper-id*`] {ipaddr` *ip-address* `[`*port*`]	multicast}`  **Example:** `Router(config-if)# h323 gatekeeper ipaddr 10.0.0.0`	Specifies the gatekeeper associated with a proxy and controls how the gatekeeper is discovered. For an explanation of the keywords and arguments, see , Step 5.

	Command	Purpose
Step 7	**h323 qos** {*ip-precedence* | **rsvp** {**controlled-load** | **guaranteed-qos**}} **Example:** Router(config-if)# h323 qos rsvp guaranteed-qos	Enables QoS on the proxy. For an explanation of the keywords and arguments, see "Configuring QoS on a Proxy Without ASR" section on page 203, Step 6.
Step 8	**interface** *type number* [*nametag*] **Example:** Router(config)# interface serial 0	If ASR is to be used, enters the interface through which outbound H.323 traffic should be routed. Keywords and arguments are platform dependent; for more information, see Step 2 above.
Step 9	**h323 asr** [**bandwidth** *max-bandwidth*] **Example:** Router(config-if)# h323 asr bandwidth 5000000	Enables ASR and specifies the maximum bandwidth for a proxy. Keyword and argument are as follows: • **bandwidth** *max-bandwidth*—Maximum bandwidth on the interface, in kbps. Range: 1 to 10,000,000. Default: the bandwidth on the interface. If you specify a value greater than the interface bandwidth, the bandwidth defaults to the interface bandwidth.
Step 10	**ip address** *ip-address mask* [**secondary**] **Example:** Router(config-if)# ip address 192.168.0.0. 225.225.225.0	Sets up the ASR interface network number. For an explanation of the keywords and arguments, see Step 3 in this configuration task table.
Step 11	**exit** **Example:** Router(config-if)# exit	Exits the current mode.
Step 12	**interface** *type number* [*nametag*] **Example:** Router(config)# interface serial 0	Enters interface configuration mode for a non-ASR interface. Keywords and arguments are platform dependent; for more information, see Step 2 above.
Step 13	**ip address** *ip-address mask* [**secondary**] **Example:** Router(config-if)# ip address 192.168.0.0 225.225.225.0	Sets up a non-ASR interface network number. For an explanation of the keywords and arguments, see Step 3 above.
Step 14	**exit** **Example:** Router(config-if)# exit	Exits the current mode.
Step 15	**router rip** **Example:** Router(config)# router rip	Configures Routing Information Protocol (RIP) for a non-ASR interface.

	Command	Purpose
Step 16	**network** *network-number* **Example:** Router(config)# network 192.168.0.0	Specifies a list of networks for the RIP routing process or a loopback interface in an Interior Gateway Routing Protocol (IGRP) domain. The argument is as follows: • *network-number*—IP address of the directly connected networks
Step 17	**router igrp** *autonomous-system* **Example:** Router(config)# router igrp 109	Configures Interior IGRP for an ASR interface. The argument is as follows: • *autonomous-system*—Autonomous system number that identifies the routes to the other IGRP routers. It is also used to tag the routing information.
Step 18	**network** *network-number* **Example:** Router(config)# network 172.16.0.0	Specifies a list of networks for the Routing Information Protocol (RIP) routing process. The argument is as follows: • *network-number*— Includes an ASR interface in an IGRP domain.
Step 19	**network** *loopback-addr* **Example:** Router(config)# network 10.0.0.0	Includes a loopback interface in an IGRP domain.

	Command	Purpose
Step 20	`access-list` *access-list-number* `{permit	deny}` *source source-mask* `[`*destination destination-mask*`]` `{eq	neq}` `[[`*source-object*`]` `[`*destination-object*`]` `[`*identification*`]` `any]`  **Example:** `Router(config)# access-list 20 permit 172.16.10.190 eq`	Creates an access list. Keywords and arguments are as follows: • *access-list-number*—Integer that uniquely identifies the access list. Range: 300 to 399. • **permit**—Permits access when there is an address match. • **deny**—Denies access when there is an address match. • *source source-mask*—Source address and mask in decimal format. DECnet addresses are written in the form area.node. For example, 50.4 is node 4 in area 50. • *destination destination-mask*—DECnet address and mask of the destination node in decimal format. DECnet addresses are written in the form area.node. For example, 50.4 is node 4 in area 50. • **eq	neq**—Item matches the packet if all the specified parts of the source object, destination object, and identification match (or do not match) the data in the packet. • *source-object*—Contains the mandatory keyword **src** and one of the following optional keywords: – **eq	neq	lt	gt**—Equal to, not equal to, less than, or greater than. Must be followed by the argument *object-number*, a numeric DECnet object number. – **exp**—Expression; followed by a regular-expression that matches a string. For more information, see the "Regular Expressions" appendix in the *Cisco IOS Dial Technologies Command Reference*.

Command	Purpose
	• *destination-object*—Contains the mandatory keyword **dst** and one of the following optional keywords:
	– **eq** | **neq** | **lt** | **gt**—Equal to, not equal to, less than, or greater than. These keywords must be followed by the argument *object-number*, a numeric DECnet object number.
	– **exp**—Expression; followed by a regular expression that matches a string. For more information, see the "Regular Expressions" appendix in the *Cisco IOS Dial Technologies Command Reference*.
	– **uic**—User identification code; followed by a numeric UID expression. The argument [*group*, *user*] is a numeric UID expression. In this case, the bracket symbols are literal; they must be entered. The *group* and *user* parts can be specified either in decimal, in octal by prefixing the number with a 0, or in hex by prefixing the number with 0x. The **uic** expression displays as an octal number.
	• *identification*—Any of the following three keywords:
	– **id**—Regular expression; refers to the user ID.
	– **password**—Regular expression; the password to the account.
	– **account**—Regular expression; the account string.
	– **any**—Item matches if *any* of the specified parts *do* match the corresponding entries for *source-object*, *destination-object*, or *identification*.
Step 21 `interface` *type number* `[`*nametag*`]` **Example:** `Router(config)# interface serial 0`	Enters interface configuration mode on an ASR interface. Keywords and arguments are platform dependent; for more information, see Step 2 above.
Step 22 `ip access-group {`*access-list-number* | *access-list-name*`} {in	out}`  **Example:** `Router(config-if)# ip access-group 101 in`	Controls access to an interface. Use this command to set the outbound access group and then the inbound access group. Keywords and arguments are as follows: • *access-list-number*—Number of an access list. Range: 1 to 199 and 1300 to 2699. • *access-list-name*—Name of an IP access list as specified by an IP **access-list** command. • **in**—Filters on inbound packets. • **out**—Filters on outbound packets.
Step 23 `exit` **Example:** `Router(config-if)# exit`	Exits the current mode.

✎

Note   ASR is not supported on Frame Relay or ATM interfaces for the Cisco MC3810.

ASR Enabled on the Proxy Using Two Different Autonomous Systems

To start the proxy with ASR enabled on the proxy using two different autonomous systems (one that contains the ASR network and the loopback network and another that contains the other non-ASR networks and the loopback network), use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **proxy h323**

2. **interface** *type number* [*nametag*]

3. **ip address** *ip-address mask* [**secondary**]

4. **h323 interface** [*port*]

5. **h323 h323-id** *h323-id*

6. **h323 gatekeeper [id** *gatekeeper-id*] {**ipaddr** *ip-address* [*port*] | **multicast**}

7. **h323 qos** {*ip-precedence* | **rsvp** {**controlled-load** | **guaranteed-qos**}}

8. **interface** *type number* [*nametag*]

9. **h323 asr** [**bandwidth** *max-bandwidth*]

10. **ip address** *ip-address mask* [**secondary**]

11. **exit**

12. **interface** *type number* [*nametag*]

13. **ip address** *ip-address mask* [**secondary**]

14. **exit**

15. **router igrp** *autonomous-system*

16. **network** *network-number*

17. **network** *network-number*

18. **router igrp** *autonomous-system*

19. **network** *network-number*

20. **network** *network-number*

21. **access-list** *access-list-number* {**permit** | **deny**} *source source-mask* [*destination destination-mask*] {**eq** | **neq**} [[*source-object*] [*destination-object*] [*identification*] **any**]

22. **interface** *type number* [*nametag*]

23. **ip access-group** {*access-list-number* | *access-list-name*} {**in** | **out**}

24. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**proxy h323** **Example:** Router(config)# proxy h323	Starts the proxy.
Step 2	**interface** *type number* [*nametag*] **Example:** Router(config)# interface loopback 3	Enters loopback-interface configuration mode. Keywords and arguments are platform dependent; for more information, see the IOS interface command reference listed in the "Additional References" section on page 256. To start the proxy with ASR enabled on the proxy using two different autonomous systems, the *type* argument is **loopback**. The **loopback** type specifies the software-only loopback interface that emulates an interface that is always up. It is a virtual interface supported on all platforms. The *number* argument is the number of the loopback interface that you want to create or configure. There is no limit on the number of loopback interfaces you can create.
Step 3	**ip address** *ip-address mask* [**secondary**] **Example:** Router(config-if)# ip address 192.168.0.0 225.225.225.0	Sets a primary or secondary IP address for an interface. Keyword and arguments are as follows: • *ip-address—IP address.* • *mask—Mask for the associated IP subnet.* • **secondary**—*The configured address is a secondary IP address. If this keyword is omitted, the configured address is the primary IP address.*
Step 4	**h323 interface** [*port*] **Example:** Router(config-if)# h323 interface 1	Signals the proxy that this interface IP address is the one to use. The argument are as follows: • *port*—Port on which the proxy listens for incoming call setup requests. Range: 1 to 65356. Default: – 11720 in -isx- or -jsx- Cisco IOS images – 1720 in -ix- Cisco IOS images, which do not contain the VoIP gateway
Step 5	**h323 h323-id** *h323-id* **Example:** Router(config-if)# h323 h323-id PX1@zone1.com	Configures the proxy name. (More than one name can be configured if necessary.) The argument is as follows: • *h323-id*—Name of the proxy. It is recommended that this be a fully qualified e-mail identification (ID), with the domain name being the same as that of its gatekeeper.
Step 6	**h323 gatekeeper [id** *gatekeeper-id*] {**ipaddr** *ip-address* [*port*] | **multicast**} **Example:** Router(config-if)# h323 gatekeeper ipaddr 10.0.0.0	Specifies the gatekeeper associated with a proxy and controls how the gatekeeper is discovered. For an explanation of the keywords and arguments, see Step 5 in the configuration task table in the "Configuring QoS on a Proxy Without ASR" section on page 203.

	Command	Purpose
Step 7	**h323 qos** {*ip-precedence* | **rsvp** {**controlled-load** | **guaranteed-qos**}} **Example:** Router(config-if)# h323 qos rsvp guaranteed-qos	Enables quality of service (QoS) on the proxy. Keywords and arguments are as follows: • **ip-precedence** *value*—Real-time Transport Protocol (RTP) streams should set their IP precedence bits to the specified value • **rsvp** {**controlled-load**}—Controlled load class of service • **rsvp** {**guaranteed-qos**}—Guaranteed QoS class of service
Step 8	**interface** *type number* [*nametag*] **Example:** Router(config)# interface serial 0	If application-specific routing (ASR) is to be used, enters the interface through which outbound H.323 traffic should be routed. Keywords and arguments are platform dependent; for more information, see Step 2 above.
Step 9	**h323 asr** [**bandwidth** *max-bandwidth*] **Example:** Router(config-if)# h323 asr bandwidth 5000000	Enables ASR and specifies the maximum bandwidth for a proxy. The argument is as follows: • *max-bandwidth*—Maximum bandwidth on the interface, in kbps. Range: 1 to 10,000,000. Default: the bandwidth on the interface. If you specify a value greater than the interface bandwidth, the bandwidth defaults to the interface bandwidth.
Step 10	**ip address** *ip-address mask* [**secondary**] **Example:** Router(config-if)# ip address 192.168.0.0 225.225.225.0	Sets up the ASR interface network number. For an explanation of the keywords and arguments, see Step 3 in this configuration task table.
Step 11	**exit** **Example:** Router(config-if)# exit	Exits the current mode.
Step 12	**interface** *type number* [*nametag*] **Example:** Router(config)# interface serial 0	Enters interface configuration mode on a non-ASR interface. Keywords and arguments are platform dependent; for more information, see Step 2 above.
Step 13	**ip address** *ip-address mask* [**secondary**] **Example:** Router(config-if)# ip address 192.168.0.0 225.225.225.0	Sets up a non-ASR interface network number. For an explanation of the keywords and arguments, see Step 3 in this configuration task table.
Step 14	**exit** **Example:** Router(config-if)# exit	Exits the current mode.

	Command	**Purpose**
Step 15	`router igrp` *autonomous-system* **Example:** `Router(config)# router igrp 4`	Configures Interior Gateway Routing Protocol (IGRP) for a non-ASR interface. The argument is as follows: • *autonomous-system*—Autonomous system number that identifies the routes to the other IGRP routers. It is also used to tag the routing information.
Step 16	`network` *network-number* **Example:** `Router(config)# network 192.168.0.0`	Includes a non-ASR interface in an IGRP domain. The argument is as follows: • *network-number*—IP address of the network of the directly connected networks
Step 17	`network` *network-number* **Example:** `Router(config)# network 192.169.0.0`	Includes a loopback interface in an IGRP domain. The argument is as follows: • *network-number*—IP address of the network of the directly connected networks
Step 18	`router igrp` *autonomous-system* **Example:** `Router(config)# router igrp 5`	Configures IGRP for an ASR interface. The argument is as follows: • *autonomous-system*—Autonomous system number that identifies the routes to the other IGRP routers. It is also used to tag the routing information.
Step 19	`network` *network-number* **Example:** `Router(config)# network 192.170.0.0`	Specifies a list of networks for the Routing Information Protocol (RIP) routing process. The argument is as follows: • *network-number*—Should include an ASR interface in an IGRP domain
Step 20	`network` *network-number* **Example:** `Router(config)# network 192.171.0.0`	Specifies a list of networks for the RIP routing process. The argument is as follows: • *network-number*—Should include a loopback interface in an IGRP domain
Step 21	`access-list` *access-list-number* `{permit	deny}` *source source-mask* `[`*destination destination-mask*`] {eq	neq} [[`*source-object*`] [`*destination-object*`] [`*identification*`] any]`  **Example:** `Router(config)# access-list 20 permit 172.16.10.190 eq`	Creates an access list. For an explanation of the keywords and arguments, see Step 20 in the configuration task table in the "Configuring QoS on a Proxy with ASR" section on page 205.

Step 22	`interface` *type number* `[`*nametag*`]`
	
	Example:
	`Router(config)# interface serial 03`
		Enters interface configuration mode on an ASR interface. Keywords and arguments are platform dependent; for more information, see Step 2 above.

	Command	Purpose

Step 23	`ip access-group {`*access-list-number* `| `*access-list-name*`} {in | out}`
	
	Example:
	`Router(config-if)# ip access-group 101 in`
		Controls access to an interface. Use this command to set the outbound access group and then the inbound access group. Keywords and arguments are as follows:
		
		• *access-list-number*—Number of an access list. Range: decimal number 1 to 199 or 1300 to 2699.
		
		• *access-list-name*—Name of an IP access list as specified by an IP **access-list** command.
		
		• **in**—Filters on inbound packets.
		
		• **out**—Filters on outbound packets.

Step 24	`exit`
	
	Example:
	`Router(config-if)# exit`
		Exits the current mode.

Configuring Border Elements

✎

Note   Cisco supports one border element per gatekeeper. For gateway configuration commands, see Configuring Annex G, page 79

To configure and provision an Annex G border element, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**

2. **h323-annexg** *border-element-id* **cost** *cost* **priority** *priority*

3. **prefix** *prefix** [**seq** | **blast**]

4. **exit**

5. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** Router(config)# gatekeeper	Enters gatekeeper configuration mode.
Step 2	**h323-annexg** *border-element-id* **cost** *cost* **priority** *priority* **Example:** Router(config-gk)# h323-annexg h323-annexg be20 cost 10 priority 40	Enables the BE on the GK and enters BE configuration mode. Keywords and arguments are as follows: • *border-element-id*—Identifier of the Annex G border element that you are provisioning. Associates the gatekeeper with the BE identifier that is configured on the BE. Possible values: any International Alphabet 5 (IA5) string, without spaces and up to 20 characters in length. • **cost** *cost*— Cost associated with this Annex G border element. When a gatekeeper sends requests to remote zones and to the BE in its attempt to resolve an address, the remote zone or BE that resolves the address and has the lowest cost and highest priority is given preference. Range: 1 to 99. Default: 50. • **priority** *priority*— Priority associated with this Annex G border element. When a gatekeeper sends requests to remote zones and to the BE in its attempt to resolve an address, the remote zone or BE that resolves the address and has the lowest cost and highest priority is given preference. Range: 1 to 99. Default: 50.
Step 3	**prefix** *prefix** [**seq** | **blast**] **Example:** Router(config-gk-annexg)# prefix 414*	(Optional) Specifies the prefixes for which a BE should be queried for address resolution. Default: the GK forwards all remote zone queries to the BE. Do not use this command unless you want to restrict the queries sent to the BE to a specific prefix or set of prefixes.
Step 4	**exit** **Example:** Router(config-gk-annexg)# exit	Exits the current mode.
Step 5	**exit** **Example:** Router(config-gk)# exit	Exits the current mode.

Configuring Endpoints

This section contains the following information:

Information About Endpoints

This section contains the following information:

Alternate Endpoints

A calling endpoint can recover from a call setup failure by sending a setup message to one of the alternate endpoints so that it is possible for a call to finish even if a gateway goes down and the gatekeeper is not yet aware of the problem. Cisco supports a maximum of 20 alternates for each endpoint, and any alternates received through registration, admission, and status protocol (RAS) messages are merged with those entered manually in the gatekeeper command-line interface. If more than 20 alternates are submitted, the total list of alternates reverts back to 20.

Alternate endpoints are configured using the **endpoint alt-ep h323id command.** This command defines the IP address for an alternate endpoint for the primary endpoint identified by its H.323 ID. The IP address is returned in the alternate endpoint field whenever the primary endpoint is returned in an ACF or LCF. The alternate endpoint gives an alternate address to place the call in case the call to the primary endpoint fails.

This command provides a failover mechanism if a gateway becomes disabled for a period of time before the gatekeeper becomes aware of the problem. After receiving an admission confirmation (ACF) from the gatekeeper with an alternate endpoint list, the Cisco gateway may attempt to use an alternate if a SETUP message results in no reply from the destination. This command causes the alternate endpoints specified to be sent in all subsequent ACF/location confirmation (LCF) messages for the endpoint named in the *h323-id* argument. Gatekeepers that support this **endpoint alt-ep h323id** command also support receiving alternate endpoint information using RAS messages. The gatekeeper accepts IP and port call signal address information in endpoint registration request (RRQ) messages. The gatekeeper list of alternates for a given endpoint is the union of the configured alternates and alternates received in RRQs from that endpoint.

The Outgoing Trunk Group ID and Carrier ID for H.323 VoIP Networks feature provides an enhancement to Registration, Admission, and Status (RAS) Admission Confirmation and Location Confirmation messages. RAS messages include a circuitInfo field that provides trunk group label or carrier ID information for remote endpoints (gateways) in H.323 networks. The Outgoing Trunk Group ID and Carrier ID for H.323 VoIP Networks feature also adds trunk group label and carrier ID support for the alternate endpoint field in the Gatekeeper Transaction Message Protocol (GKTMP) Response Admission Request (ARQ), Admission Confirmation (ACF), Location Request (LRQ), and Location Confirmation (LCF) messages.

This feature allows a gatekeeper to specify a primary route-server trunk group as the destination to which a call is to be routed. The gatekeeper provides the IP address of the terminating gateway and the trunk group label or carrier ID of that gateway (in the circuitInfo field) to the requesting gateway. The GKTMP application server provides the trunk group label or carrier ID of the terminating gateway to the gatekeeper in the RESPONSE ARQ, ACF, LRQ, or LCF messages. The gatekeeper converts the trunk group ID or carrier ID information and sends it in the circuitInfo field of its RAS message to the requesting gateway.

The GKTMP application server may also provide a list of alternate gateways in the RESPONSE ARQ, ACF, LRQ, or LCF messages that the gatekeeper sends to the requesting gateway. The alternate gateway list includes a separate call signal address and circuitInfo field (trunk group label or carrier ID) for each alternate gateway. The gatekeeper removes identical alternate gateway routes from the consolidated alternate gateway list before sending the list to the requesting gateway.

> **Note** The gatekeeper does not validate whether the alternate gateway is valid or whether the target carrier ID will have enough capacity if the destination gateways and their trunk group labels and carrier IDs are registered to the local gatekeeper zone.

Figure 9 illustrates that this feature allows the gatekeeper in Zone 1 to receive routing information from the primary gateway, Gateway 2 in Zone 2, and from the alternate gateway, Gateway 3, also in Zone 2. The routing information is passed from the gatekeeper in Zone 1 to requesting Gateway 1.

Figure 9 *Topology of Routing Between Zone 1 and Zone 2*

GKTMP = Gatekeeper Transaction Message Protocol
PSTN = public switched telephone network
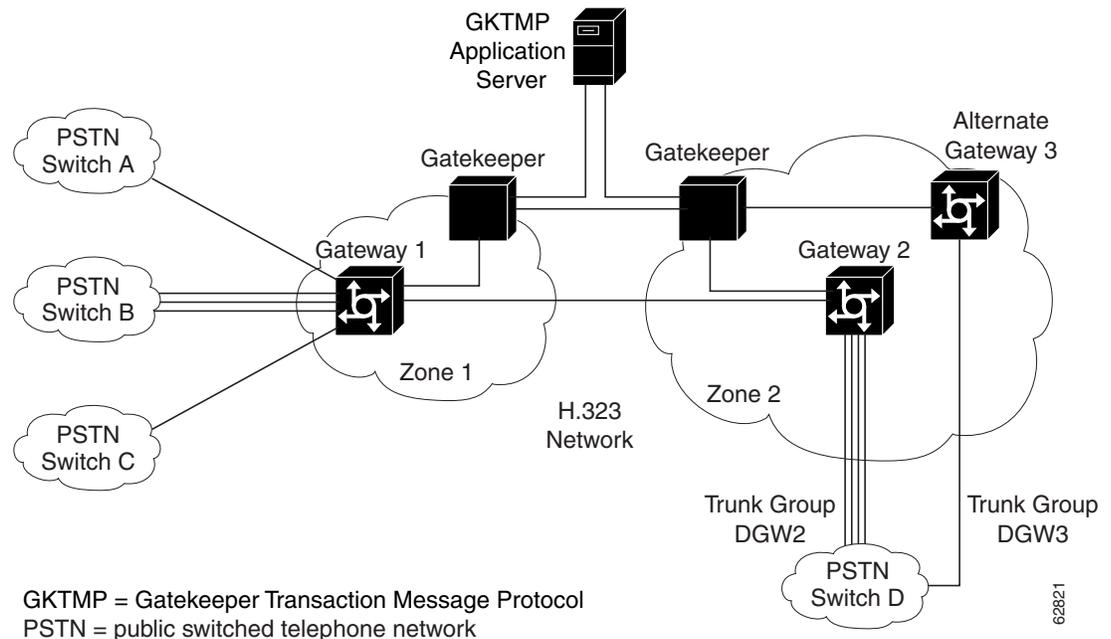
The RAS message includes a new field called circuitInfo. The information in the circuitInfo field corresponds to the information in the Q and J tags in the GKTMP message. The trunk group label (Q tag) or carrier ID (J tag) of the primary gateway is provided in the alternateEndpoint structure of the GKTMP message, along with the call signal address of the primary gateway. The trunk group label or

carrier ID of each alternate gateway is also provided in the alternateEndpoint structure of the GKTMP message. The Q and J tags of each alternate gateway are embedded inside the existing A-tagged fields of the GKTMP message, as shown in the following example:

```
A=c:{I:172.18.194.1:1720} J:CARRIER_ID
A=c:{I:10.1.1.1:1720}    Q:TRUNK_GROUP_LABEL
```

The following is an example of a RAS message from a gatekeeper to a requesting gateway. (The gatekeeper has converted the information in the Q and J fields of the GKTMP message that it received from the GKTMP application server.) The RAS message contains two alternate endpoints, each of which has a circuitInfo field:

```
alternateEndpoints
 callSignalAddress
  ipAddress :
   'AC12C826'H
  port 1720
  circuitInfo
   destinationCircuitID
    group
     group "CARRIER_ID"
!
 callSignalAddress
  ipAddress :
   ip 'AC12C816'H
  port 1720
  circuitInfo
   destinationCircuitID
    group
     group "TRUNK_GROUP_LABEL"
```

Limitations for Alternate Endpoints

The gatekeeper can be instructed by GKTMP servers to send alternate endpoints with same call signaling address and different calling or called numbers in the ACF. When this happens the Cisco gateway acting as the endpoint will send an alternate endpoint attempt to the same call signaling address as the primary call. If the first call is still active on the terminating gateway when the second call arrives the TGW would detect a call loop because the calls share the same GUID, and the second call will be rejected with a 'CALL_LOOP' message printed on syslog.

- Effective with Cisco IOS Release 12.4(9)T2 and before, the first call can be active on the TGW when the second call arrives in the following cases.

    1. A Release Complete message has been sent on the first call, but the TGW keeps the call active till a Release Complete message arrives from OGW or till the release timer expires.

    2. A Release Complete message has been sent out on the first call, but a DRQ has not arrived from the GK.

- Effective with Cisco IOS Release 12.4(9)T3 and later, the first call can be active on the TGW when the second call arrives because:

    1. The TGW keeps the call active if Maintain connection timeout is turned off, even if a Release Complete message has been sent on the first call. The call is kept active till a Release Complete message arrives from OGW or till the release timer expires.

    2. A Release Complete message has been sent out on the first call, but a DRQ has not arrived from the GK.

Carrier-Based Routing Without a GKTMP Application Server

Carrier-based routing is possible without the presence of the GKTMP application server if you have Cisco IOS Release 12.3(8)T1, Cisco IOS Release 12.3(11)T, or higher. The trunk group label or carrier ID of the terminating gateway can also be provided by the destination circuitInfo field in ARQ. Incoming ARQ to the gatekeeper has the destination circuitInfo field. When both GKTMP and incoming ARQ provide the trunk group ID or carrier ID, the ID provided by the GKTMP server is accepted. The GKTMP server can also add, modify, or delete the trunk group ID or carrier ID present in ARQ using RESPONSE ARQ or ACF message. If the RESPONSE ARQ or the ACF message does not include a Q or J tag, only the trunk group or carrier ID provided by the incoming ARQ is used for routing.

Additional Routes to Alternate Endpoints

The Location Confirmation Enhancements for Alternate Endpoints feature allows a Cisco IOS gatekeeper to collect additional routes to endpoints that are indicated by multiple location confirmation (LCF) responses from remote gatekeepers and convey a collection of those routes to the requesting (calling) endpoint. Currently, the originating gatekeeper sends Location Request (LRQ) messages to multiple remote zones. Remote gatekeepers in the zones return LCF responses to the originating gatekeeper. The LCF responses indicate alternate routes to the endpoints of the remote gatekeeper. The consolidation of LCF responses to multiple LRQ messages can provide many alternate routes to reach a given destination. An endpoint can have up to 20 alternate endpoints.

The remote gatekeeper zones have been configured in the originating gatekeeper using the **zone remote** command, specifying the cost and priority to each remote zone. After receiving the LCF responses, the originating gatekeeper determines the best route to an endpoint on the basis of the cost and priority of remote zones returning the responses. The originating gatekeeper then forwards route information to the requesting endpoint in the Admission Confirmation (ACF) message, which contains an ordered list of alternate endpoints.

The Location Confirmation Enhancements for Alternate Endpoints feature allows the originating gatekeeper to discover and relay more possible terminating endpoints to the requesting endpoint, therefore providing alternate routes to endpoints that can be used if the best route is busy or does not provide any alternate routes. The endpoint that receives the list of alternate endpoints tries to reach them in the order in which the alternate endpoints were received. The Location Confirmation Enhancements for Alternate Endpoints feature can be used on gatekeepers that originate LRQ messages and directory gatekeepers that forward LRQ messages.

The Location Confirmation Enhancements for Alternate Endpoints feature allows you to choose the number of alternate routes you want the gatekeeper to collect during the existing LRQ timer window. When the timer expires or the best response and sufficient alternates are received, the resolved address and alternate endpoints from all the LCFs received by the gatekeeper are consolidated in a single list. Identical alternate endpoints are removed from the list. That is, if an alternate endpoint that was received in an LCF message has an identical IP address or trunk group label or carrier ID as any alternate endpoints received in previous LCF messages, the previous duplicate alternate endpoints are removed from the consolidated list.

The address and endpoints are sent as alternate endpoints in the ACF or LCF messages from the gatekeeper. If this feature is not enabled, the gatekeeper stops collecting routes after the LRQ timer expires and then chooses the best LCF and sends it in the ACF message. After you enable the feature, the gatekeeper stops collecting routes after the LRQ timer expires and then consolidates the endpoints from all LCF messages received.

Note Annex G border element (BE) interaction is not affected. The LCF responses from BEs are treated like any remote gatekeeper LCF.

Effective with Cisco IOS Release 12.2(11)T, duplicate alternate endpoints that are received in a Location Confirmation (LCF) message are removed from the consolidated list of endpoints. The current gatekeeper limitations apply:

- Ten LRQ messages can be sent by the gatekeeper; therefore, there is a limit of 10 remote zones that are handled by the gatekeeper.
- ACF and LCF messages can carry up to 20 alternate endpoints.

Nonavailability Information for Terminating Endpoints

An H.323 Location Request (LRQ) message is sent by a gatekeeper to another gatekeeper to request a terminating endpoint. The second gatekeeper determines the appropriate endpoint on the basis of the information contained in the LRQ message. However, sometimes all the terminating endpoints are busy servicing other calls and none are available. If you configure the **lrq reject-resource-low** command, the second gatekeeper rejects the LRQ request if no terminating endpoints are available. If the command is not configured, the second gatekeeper allocates and returns a terminating endpoint address to the sending gatekeeper even if all the terminating endpoints are busy. A call has a higher chance of succeeding if the availability of the endpoint is determined in advance. Returned addresses are only those that have available capacity. Rejecting an LRQ message forces the sending gatekeeper to query other gatekeepers to find an endpoint that has available capacity.

Endpoint-Based Call-Capacity Management

Gatekeepers can currently provide dynamic calculation of maximum calls for endpoints that report v4 call capacity in Registration, Admission, and Status (RAS) messages. This enhancement enables the static assignment of a maximum number of calls to an endpoint and the dynamic calculation of maximum calls to be overridden for an endpoint. You can also statically assign maximum calls to non-v4 endpoints that do not report call capacity and to override the dynamic calculation of maximum calls for an endpoint that does report call capacity. The **endpoint circuit-id h323id** command is used to configure the dynamic calculation of maximum calls.

While managing endpoint call capacity, a gatekeeper uses one of two different fields of the endpoint structure to store endpoint call capacity (based on the flag voice_GwCallsAvailable_present and h323_GwCallsAvailable_present of call capacity reported by an endpoint). If the Endpoint-Based Call Capacity Management feature is used to configure maximum calls, the gatekeeper stores endpoint call capacity in the field that is already in use (e_voiceCallCapacity and e_h323CallCapacity). If no field is in use (if the endpoint is not reporting call capacity), the gatekeeper uses the field associated with the time-division multiplexing (TDM) gateway (this is e_voiceCallCapacity) to store endpoint call capacity.

A gatekeeper also does active call counting for carrier-based routing when an endpoint reports capacity or carrier ID information in an ARQ or disengage request (DRQ) message or is statically configured for carrier ID and maximum call. Call accounting is extended if an endpoint does not report capacity or carrier ID information in the ARQ or DRQ message or is not statically configured for carrier ID and maximum calls. The **show gatekeeper endpoints** command displays the current call count for the endpoint. The current call should be updated for the call reported using the information response (IRR) message.

Gatekeeper resource monitoring is enabled using the **endpoint resource-threshold onset** command. If the command is configured, a gatekeeper currently indicates that a gateway is "out-of-resource" when the available call percentage is less than the configured value. For prefix-based routing, nothing special needs to be configured for the gatekeeper to select a local destination gateway. For carrier-based routing, before selecting a local destination endpoint, a gatekeeper currently checks to ensure that the endpoint is not out-of-resource for the destination carrier. The gatekeeper must perform an additional check to ensure that the endpoint is not out-of-resource (as reported through the Resource Availability Indicator (RAI) out-of-resource flag).

Configuring Alternate Endpoints

This section contains the following information:

Configuring Endpoints

To configure alternate endpoints, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **endpoint alt-ep h323id** *h323-id ip-address* [*port*] [**carrier-id** *carriername*]
3. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router (config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`endpoint alt-ep h323id` *h323-id ip-address* [*port*] [**carrier-id** *carriername*]
	
	Example:
	`Router (config-gk)# endpoint alt-ep h323id`
	`h323-id 192.168.0.0`
		Configures alternate endpoints. Keywords and arguments are as follows:
		
		• *h323-id*—H.323 name ID of the endpoint for which an alternate address is being supplied. This ID is used by a gateway to communicate with the gatekeeper. Usually, this ID is the name given to the gateway, with the gatekeeper domain name appended.
		
		• *ip-address*—IP address of an alternate endpoint.
		
		• *port*—Port number associated with the address of the alternate. Default: 1720.
		
		• **carrier-id** *carriername*—Trunk group label or circuit ID of the alternate endpoint. It may be added in addition to the IP address of the alternate endpoint.

Step 3	`exit`
	
	Example:
	`Router (config-gk)# exit`
		Exits the current mode.

Verifying Endpoints

To verify alternate endpoints, perform the following steps.

Step 1   **show gatekeeper endpoints alternates**

Use this command to display the status of all registered endpoints for a gatekeeper.

The following example shows three carrier IDs (CARRIER_ABC, CARRIER_DEF, and CARRIER_GHI):

```
Router# show gatekeeper endpoints alternates

                  GATEKEEPER ENDPOINT REGISTRATION
                  ==================================
CallSignalAddr  Port  RASSignalAddr   Port  Zone Name  Type   Flags
----------------- ----- ------------------   ---- -------------- ------ ------
!
ALL CONFIGURED ALTERNATE ENDPOINTS
                  ==================================
Endpoint H323 Id             RASSignalAddr   Port    Carrier Id
----------------------        ------------------    -----    -----------
gwid                              1.1.1.1            1720    CARRIER_ABC
gwid                              1.1.1.1            1720    CARRIER_DEF
gwid                              2.2.2.2            1720    CARRIER_GHI
```

Configuring Additional Routes to Alternate Endpoints

To configure a collection of alternate endpoints, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**

2. **endpoint alt-ep collect** *value* [**distribute**]

3. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router (config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`endpoint alt-ep collect` *value* [`distribute`] **Example:** `Router(config-gk)# endpoint alt-ep collect 20`	Configures the number of alternate routes to consolidate from various LCF responses before ending the collection process and sending the LCF message to the requesting endpoint. Keywords and arguments are as follows: • *value*—Number of routes. Range: 1 to 20. Default: 0, which indicates that alternate route consolidation is not enabled. When the feature is not enabled, the gatekeeper gets alternates from only one LCF (the best LCF with the least-cost routing). The gatekeeper ignores the alternates present in other LCF messages even if they are present and there is no consolidation. Identical alternate endpoints are removed from the list. That is, if an alternate endpoint received in an LCF message has an identical IP address or trunk group label or carrier ID as any alternate endpoints received in previous LCF messages, the previous duplicate alternate endpoints are removed from the consolidated list. • **distribute**—Gatekeeper includes alternate routes from as many LCF messages as possible in the consolidated list. Use of this keyword allows the gatekeeper to give fairness to the information of alternate routes present in various LCFs.

Step 3	`exit`
	
	Example:
	`Router (config-gk)# exit`
		Exits the current mode.

Configuring Nonavailability Information for Terminating Endpoints

Configuring the Sending of Nonavailability Information

To configure a gatekeeper to notify a sending gatekeeper on receipt of an LRQ message that no terminating endpoints are available, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **lrq reject-resource-low**
3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** Router (config)# gatekeeper	Enters gatekeeper configuration mode.
Step 2	**lrq reject-resource-low** **Example:** Router (config-gk)# lrq reject-resource-low	Configures the gatekeeper to notify a sending gatekeeper on receipt of an LRQ message that no terminating endpoints are available.
Step 3	**exit** **Example:** Router (config-gk)# exit	Exits the current mode.

Verifying the Sending of Nonavailability Information

To verify gatekeeper configuration, perform the following steps.

Step 1   **show running-config**

Use this command to verify that the gatekeeper is configured to notify a sending gatekeeper on receipt of an LRQ message that no terminating endpoints are available.

Configuring Endpoint-Based Call-Capacity Management

Note   The **endpoint resource-threshold onset command must be configured for the gatekeeper to perform endpoint-based call-capacity management.**

To configure endpoint-based call-capacity management, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**

2. **endpoint max-calls h323id** *endpoint-id max-calls*

3. exit

DETAILED STEPS

	Command or Action	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`endpoint max-calls h323id` *endpoint-id max-calls*
	
	Example:
	`Router(config-gk)# endpoint max-calls h323id`
	`GW-1 1000`
		Sets the maximum number of calls that are allowed for an endpoint. Arguments are as follows:
		• *endpoint-id—ID of the endpoint.*
		• *max-calls—Maximum number of calls allowed to the endpoint. Range: 1 to 100000.*

Step 3	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Forcing Endpoint Unregistration

This section contains the following information:

Prerequisites for Forcing Unregistration

- For gatekeeper cluster configurations, the **clear h323 gatekeeper endpoint** command must be entered on the gatekeeper where the endpoint is registered. Use the **show gatekeeper endpoints** command to locate the endpoint in a gatekeeper cluster.

Forcing Unregistration

To force a gatekeeper to unregister an endpoint, use the **clear h323 gatekeeper endpoint** command as described below. Alternatively, you can issue a command from the GKTMP server to unregister an endpoint.

Note For more information on GKTMP, see the *Cisco Gatekeeper External Interface Reference*, Version 4.2 at http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/rel_docs/gktmp4_2/.

To force endpoint unregistration, use the following command beginning in global configuration mode.

SUMMARY STEPS

1. **clear h323 gatekeeper endpoint** {**alias** {**e164** *name* | **h323id** *name*} | **all** | **id** *number* | **ipaddr** *ip-address* [*port*]}

DETAILED STEPS

	Command	Purpose
Step 1	`clear h323 gatekeeper endpoint` {**alias** {**e164** *name* | **h323id** *name*} | **all** | **id** *number* | **ipaddr** *ip-address* [*port*]} **Example:** `Router# clear h323 gatekeeper endpoint all`	Forces the gatekeeper to send an unregistration request (URQ) message to the specified endpoint or all endpoints and removes the endpoint from the gatekeeper registration database. The endpoint that is unregistered can come back if it sends the RRQ message back to the gatekeeper after unregistration. Keywords and arguments are as follows: • **alias e164** *name*—E.164 alphanumeric address that is specified in the local alias table. • **alias h323id** *name*—H.323 ID name that is specified in the local alias table and is an alternate way to reach an endpoint. • **all**—All endpoints. • **id** *number*—ID of the endpoint. • **ipaddr** *ip-address* [*port*]—Call signaling address and port (optional) of the endpoint. Default: 1720.

Verifying Unregistration

To verify unregistration, perform the following steps.

Step 1 Verify that you did not receive an error message after entering the **clear h323 gatekeeper endpoint** command.

Step 2 **show gatekeeper endpoints**

Use this command to view all endpoints registered to the gatekeeper:

```
Router# show gatekeeper endpoints

GATEKEEPER ENDPOINT REGISTRATION
                  ================================
CallSignalAddr  Port  RASSignalAddr   Port  Zone Name         Type


Flags
--------------- ----- --------------- ----- ---------         ----


-----
1.1.1.1         1720  1.1.1.1         1719  gk-e4-2           VOIP-GW S
    H323-ID: test (static)
Total number of active registrations = 1
```

Step 3 Verify that the unregistered endpoint is not displayed in the list of endpoints.

Configuring the IRR Timer and Disable IRQ Requests

This section contains the following information:

Restrictions for the IRR Timer and Disable IRQ Requests

- If the gatekeeper is configured to not send IRQs with the CRV set to zero, bandwidth control is not supported.
- Adjusting the IRR frequency while there are existing calls should be avoided.
- All gatekeepers should have the same IRR frequency configured to prevent problems during gatekeeper switchover.
- RQ retries from two to nine increases DRQ reliability. This value is not configurable.

Information About the IRR Timer and Disable IRQ Requests

Call Status Tracking Optimization reduces unnecessary messages between gatekeeper and the gateways, reducing network congestion and CPU over-utilization.

In an H.323 VoIP network, gatekeepers use information request (IRQ) messages to obtain information about a certain call or all calls from an endpoint (for example, an originating gateway). The gatekeeper can send an IRQ to request information from the endpoint, which responds with an information request response (IRR). The gatekeeper can also use the irrFrequency field in the initial admission confirm (ACF) message to instruct the endpoint to periodically report with IRR messages during call admission.

Currently, the Cisco gatekeeper maintains the call states of all calls it has admitted, to track bandwidth usage. In addition, the gatekeeper must be able to reconstruct call structures for a newly transferred gateway from an alternate gatekeeper, if a gatekeeper switchover has occurred. In a gatekeeper switchover, the new gatekeeper sends an IRQ message with the call reference value (CRV) set to 0 to the newly registered gateway to obtain information about existing calls before the switchover.

If a gateway supports a large volume of calls, the number of IRR messages as responses to an IRQ with the CRV set to zero could be very CPU intensive and cause congestion. Additionally, if a gatekeeper serves many endpoints or high-capacity gateways, the IRQ requests and the resulting IRR messages received can flood the network, causing high CPU utilization and network congestion.

The Call Status Tracking Optimization feature provides the following methods to address this potential problem:

- A command to configure IRR frequency that is included in the ACF message. Currently, the IRR frequency is set to 240 seconds (4 minutes), based on an average 4-minute call hold time. The IRR allows the gatekeepers to terminate calls for which a disengage request (DRQ) has not been received. If missing DRQs are not a problem, the IRR frequency can be set to a larger value than four minutes, minimizing the number of unnecessary IRRs sent by a gateway.
- A command to disable the gatekeeper from sending an IRQ with the CRV set to zero when the gatekeeper is requesting the status of all calls after its initialization. Disabling the IRQ can eliminate unnecessary IRR messages in cases where the reconstruction of call structures can be postponed until the next IRR, or in cases where the call information is no longer required because calls are terminated before the periodic IRR is sent. Disabling the IRQ is advantageous if direct bandwidth control is not used in the gatekeeper.

- An increase from two to nine in the number of retries for sending the DRQ. If the reliability of DRQ messages is increased, a longer period can be used before the next IRR is sent. Third-party gatekeepers must support this feature.

Configuring IRR Periodic Intervals on the Gatekeeper

To configure IRR periodic intervals on the gatekeeper, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **timer irr period** *value*
3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** Router(config)# gatekeeper	Enters gatekeeper configuration mode.
Step 2	**timer irr period** *value* **Example:** Router(config-gk)# timer irr period 30	Configures the IRR timer, or the periodic interval of IRR messages sent by the gatekeeper, in minutes. The gatekeeper uses this value to populate the irrFrequency field in the ACF message. Range: 1 to 60. Default: 4.
Step 3	**exit** **Example:** Router(config-gk)# exit	Exits the current mode.

Disabling IRQ Requests for All Calls in the Gatekeeper

To disable IRQ requests for all calls in the gatekeeper, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**
2. **no irq global-request**
3. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`no irq global-request`
	
	Example:
	`Router(config-gk)# no irq global-request`
		Prohibits the gatekeeper from sending IRQ requests with a CRV set to zero to endpoints to obtain information about all calls. These IRQ requests are usually sent after a gatekeeper initializes upon switchover.
		
		Default: sends IRQ requests with a CRV set to zero.

Step 3	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Configuring Sequential LRQs

This section contains the following information:

Restrictions for Sequential LRQs

- In a network where LRQs are forwarded through multiple gatekeepers along a single path, a single LRQ sent from a gatekeeper could solicit multiple LRJ and Location Confirmation (LCF) responses. If an LRJ response is received first, a potentially unnecessary LRQ could be sent to the next zone, increasing traffic.

  To avoid this problem, ensure that the gatekeepers do not use the **blast** option, or carefully configure the sequential timer on each gatekeeper along the path. Using sequential LRQs in a directory gatekeeper along the path can also help because sequential LRQs in the directory gatekeeper always send one response back to an LRQ request.

Information About Sequential LRQs

You can configure the gatekeeper to provide a potentially faster LRQ response to the originator of the request when a location reject (LRJ) response is received while the gatekeeper is sending sequential LRQs. The Sequential LRQ Enhancement feature introduces a fixed delay for the gatekeeper to send sequential LRQs to successive zones even when a negative response or an LRJ is received from the current zone.

You configure this fixed delay using the **lrq lrj immediate-advance** command. If an LRJ is received from the current zone, the gatekeeper assumes that the current zone cannot satisfy the request and immediately sends an LRQ to the next zone. This feature works for both typical and directory gatekeepers.

Figure 9 shows how the Sequential LRQ Enhancement feature affects call flows. The originating gatekeeper sends LRQ1 to the first gatekeeper. GK1 responds with LRJ1. Immediately upon receipt of LRJ1, the originating gatekeeper sends LRQ2 to GK2.

Call Flow Using Sequential LRQ Enhancement Feature

Figure 10   *Call Flow Using Sequential LRQ Enhancement Feature*
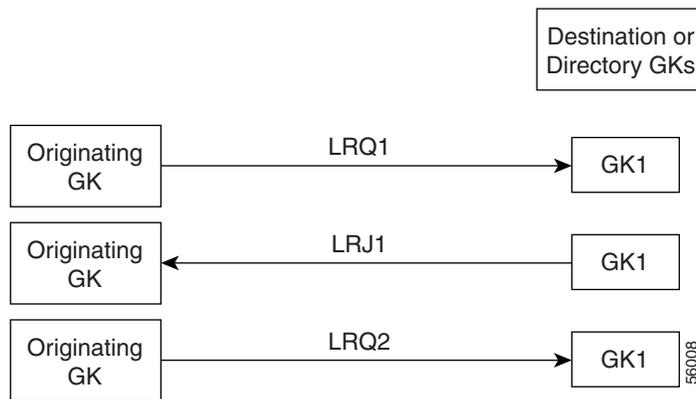
Figure 12 shows a call flow with the Sequential LRQ Enhancement feature when LRJ1 arrives after the delay timer has expired and after LRQ2 has been sent. If this occurs, the originating gatekeeper does not send LRQ3 and ignores LRQ2.

Figure 11   *Call Flow with LRJ1 Arriving After Delay Timer Expiration*
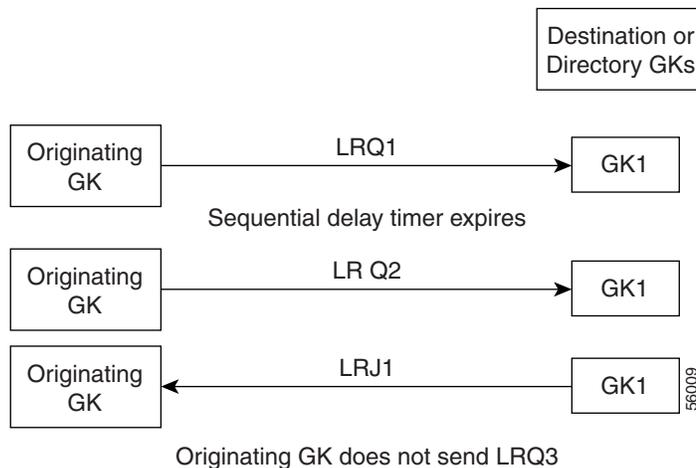
Configuring Sequential LRQ Enhancement

To configure sequential LRQ enhancement, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**

2. **lrq lrj immediate-advance**

3. **exit**

DETAILED STEPS

	Command	Purpose
Step 1	**gatekeeper** **Example:** Router(config)# gatekeeper	Enters gatekeeper configuration mode.
Step 2	**lrq lrj immediate-advance** **Example:** Router(config-gk)# lrq lrj immediate-advance	Enables the GK to immediately send an LRQ to the next zone after it receives an LRJ from a GK in the current zone.
Step 3	**exit** **Example:** Router(config-gk)# exit	Exits the current mode.

Configuring the Sequential LRQ Timer

To configure the sequential LRQ timer, use the following commands beginning in global configuration mode.

SUMMARY STEPS

1. **gatekeeper**

2. **timer lrq seq delay** *time-in-100-ms-units*

3. **exit**

DETAILED STEPS

	Command	Purpose

Step 1	`gatekeeper`
	
	Example:
	`Router(config)# gatekeeper`
		Enters gatekeeper configuration mode.

Step 2	`timer lrq seq delay` *time-in-100-ms-units* **Example:** `Router(config-gk)# timer lrq seq delay 3`	Defines the intervals for the GK to send successive sequential LRQs. The LRQ sequential timing source (SEQ) delay is used to set the time between sending LRQs to remote gatekeepers for address resolution. To resolve an address, the gatekeeper might have several remote zones configured, and it can send the LRQs simultaneously (blast) or sequentially (seq). The gatekeeper chooses the best route based on availability and cost. Using LRQs sequentially results in lower network traffic, but can increase latency of calls when the most preferred route is unavailable. The argument is as follows: • *time-in-100-ms-units*—Timer value, in hundreds of ms. Range: 1 to 10. Default: 5 (500 ms). Lowering the time increases traffic on the network but might reduce call-setup time.

Step 3	`exit`
	
	Example:
	`Router(config-gk)# exit`
		Exits the current mode.

Verifying Sequential LRQ Enhancement

Step 1   **show running-config**

Use this command to verify that the Sequential LRQ Enhancement feature is enabled.

```
Router# show running-config

Building configuration...

Current configuration : 1802 bytes
!
version 12.2
.
.
.
gatekeeper
 zone local Zone1 cisco.com
 zone remote c3620-1-gk cisco.com 209.165.200.225 1719
 zone remote c2514-2-gk cisco.com 209.165.200.228 1719
 zone remote gk-cisco-mn cisco.com 209.165.200.230 1719
 zone remote gkzone3 cisco.com 209.165.200.235
 zone remote gk-catapult cisco.com 209.165.200.229 1719
 zone prefix gkzone3 405.......
```

```
                zone prefix gk-gk5 515....
                zone prefix c2514-2-gk 910.......
                zone prefix c3620-1-gk 917300....
                zone prefix c2514-2-gk 919.......
                zone prefix gk-cisco-mn 919.......
                zone prefix c3620-1-gk 919.......
                lrq reject-resource-low
                lrq lrj immediate-advance
                timer lrq window 6
                no shutdown
                .
                .
                .
```

Configuration Examples for H.323 Gatekeepers and Proxies

This section provides the following configuration examples:

HSRP: Example

This sample sample configuration uses Ethernet 0 as the HSRP interface on both gatekeepers.

Primary Gatekeeper

```
configure terminal
 ! Enter global configuration mode.
interface ethernet 0
 ! enter interface configuration mode for interface ethernet 0.
 standby 1 ip 172.21.127.55
 ! Member of standby group 1, sharing virtual address 172.21.127.55.
 standby 1 preempt
 ! Claim active role when it has higher priority.
 standby 1 timers 5 15
 ! Hello timer is 5 seconds; hold timer is 15 seconds.
 standby 1 priority 110
 ! Priority is 110.
```

Backup Gatekeeper

```
configure terminal
interface ethernet 0
 standby 1 ip 172.21.127.55
 standby 1 preempt
 standby 1 timers 5 15
```

The configurations are identical except that gk2 has no standby priority configuration, so it assumes the default priority of 100—meaning that gk1 has a higher priority.

gk1 and gk2 Gatekeeper Mode Configurations

```
configure terminal
 ! Enter global configuration mode.
gatekeeper
 ! Enter gatekeeper configuration mode.
 zone local gk-sj cisco.com 172.21.127.55
 ! Define local zone using HSRP virtual address as gatekeeper RAS address.
.
.
.
 ! Various other gk-mode configurations.
 no shut
 ! Bring up the gatekeeper.

configure terminal
 ! Enter global configuration mode.
gatekeeper
 ! Enter gatekeeper configuration mode.
 zone local gk-sj cisco.com 172.21.127.55
 ! Define local zone using HSRP virtual address as gatekeeper RAS address.
 ! Note this uses the same gkname and address as on gk1.
.
.
.
! Various other gk-mode configurations.
 no shut
 ! Bring up the gatekeeper.
```

Note The **no shut** command is issued on both gatekeepers, primary and secondary. If the **show gatekeeper status** command is issued on the two gatekeepers, gk1 shows the following:

```
Gatekeeper State: UP
! But gk2 shows the following:
Gatekeeper State: HSRP STANDBY
```

Gatekeeper Zones: Example

Multiple Zones

The following example shows how to define multiple local zones for separating gateways:

```
zone local gk408or650 xyz.com
zone local gk415 xyz.com
zone prefix gk408or650 408.......
zone prefix gk408or650 650.......
zone prefix gk415 415.......
```

All the gateways used for area codes 408 or 650 can be configured so that they register with gk408or650, and all gateways used for area code 415 can be configured so that they register with gk415.

One Zone for Multiple Gateways

The following example shows how to put all the gateways in the same zone and use the **gw-priority** keyword to determine which gateways are used for calling different area codes:

```
zone local localgk xyz.com
zone prefix localgk 408.......
zone prefix localgk 415....... gw-priority 10 gw1 gw2
zone prefix localgk 650....... gw-priority 0 gw1
```

The commands shown accomplish the following tasks:

- Domain xyz.com is assigned to gatekeeper localgk.

- Prefix 408....... is assigned to gatekeeper localgk, and no gateway priorities are defined for it; therefore, all gateways that register to localgk can be used equally for calls to the 408 area code. No special gateway lists are built for the 408....... prefix; selection is made from the master list for the zone.

- The prefix 415....... is added to gatekeeper localgk, and priority 10 is assigned to gateways gw1 and gw2.

- Prefix 650....... is added to gatekeeper localgk, and priority 0 is assigned to gateway gw1.

A priority 0 is assigned to gateway gw1 to exclude it from the gateway pool for prefix 650........ When gateway gw2 registers with gatekeeper localgk, it is added to the gateway pool for each prefix as follows:

- For gateway pool for 415......., gateway gw2 is set to priority 10.

- For gateway pool for 650......., gateway gw2 is set to priority 5.

To change gateway gw2 from priority 10 for zone 415....... to the default priority 5, enter the following command:

```
no zone prefix localgk 415....... gw-pri 10 gw2
```

To change both gateways gw1 and gw2 from priority 10 for zone 415....... to the default priority 5, enter the following command:

```
no zone prefix localgk 415....... gw-pri 10 gw1 gw2
```

In the preceding example, the prefix 415....... remains assigned to gatekeeper localgk. All gateways that do not specify a priority level for this prefix are assigned a default priority of 5. To remove the prefix and all associated gateways and priorities from this gatekeeper, enter the following command:

```
no zone prefix localgk 415.......
```

Session Bandwidth Limits

The following example shows session bandwidth limits and resource information for destination zones configured on the gatekeeper:

```
Router# show running-config
!
Building configuration...

Current configuration : 1329 bytes
!
version 12.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
```

```
!
hostname router
!
username all
memory-size iomem 10
clock timezone GMT 0
aaa new-model
!
aaa accounting connection h323 stop-only group radius
aaa session-id common
ip subnet-zero
!
no ip domain lookup
ip domain name cisco.com
ip host anyname-tftp1 172.18.207.15
ip dhcp smart-relay
!
voice call carrier capacity active
voice service voip
 sip
 session transport tcp
 rel1xx disable
!
interface Ethernet0/0
 ip address 172.18.200.28 255.255.255.0
 half-duplex
 no cdp enable
!
interface TokenRing0/0
 no ip address
 shutdown
 ring-speed 16
 no cdp enable
!
no ip http server
ip classless
ip route 0.0.0.0 0.0.0.0 172.18.200.1
!
radius-server host 172.18.200.30 auth-port 1645 acct-port 1646
radius-server vsa send accounting
!
dial-peer cor custom
!
gatekeeper
 zone local GK-1 cisco.com 172.18.200.28
 zone local GK-2 cisco.com
 zone local word word
 zone remote GK-3 cisco.com 172.18.200.5 1719
 zone prefix GK-2 1..
 gw-type-prefix 1#* default-technology
 bandwidth interzone default 1
 bandwidth session default 5
 bandwidth remote 4
 no shutdown
 server registration-port 21000
!
line con 0
 exec-timeout 0 0
line aux 0
line vty 0 4
 password lab
 line vty 5 15
!end
```

Load Balancing with Alternate Gatekeepers: Example

Redundant Gatekeepers for a Zone Prefix

In the following example, two remote gatekeepers are configured to service the same zone prefix:

```
gatekeeper
 zone remote c2600-1-gk cisco.com 172.18.194.70 1719
 zone remote c2514-1-gk cisco.com 172.18.194.71 1719
 zone prefix c2600-1-gk 919.......
 zone prefix c2514-1-gk 919.......
```

Redundant Gatekeepers for a Technology Prefix

In the following example, two remote gatekeepers are configured to service the same technology prefix:

```
gatekeeper
 zone remote c2600-1-gk cisco.com 172.18.194.70 1719
 zone remote c2514-1-gk cisco.com 172.18.194.71 1719
 gw-type-prefix 3#* hopoff c2600-1-gk hopoff c2514-1-gk
```

Security and Authentication: Example

Domain Zones and the IZCT Password

All of the configuration examples are for Figure 12. One IZCT password is enabled for all of the gatekeepers.

Figure 12   *Set-Up Diagram for the Example Configuration*
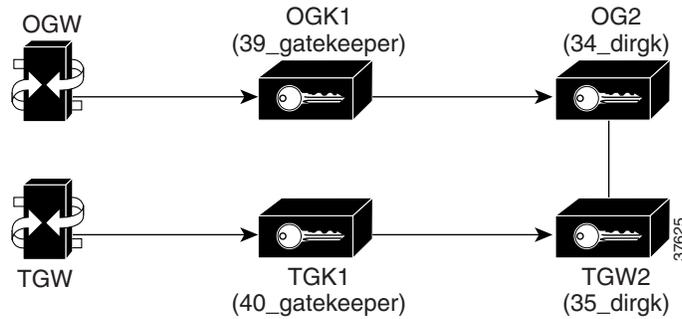
Originating Gatekeeper 1

```
config terminal
gatekeeper
 zone local 39_gatekeeper cisco.com 172.18.198.92
 zone remote 34_dirgk cisco.com 172.18.198.197 1719
 zone prefix 39_gatekeeper 919*
 zone prefix 34_dirgk *
 security izct password cisco
 gw-type-prefix 1#* default-technology
 no shutdown
```

Terminating Gatekeeper 1

```
config terminal
gatekeeper
```

```
zone local 40_gatekeeper cisco.com 172.18.198.91
 zone remote 35_dirgk cisco.com 172.18.198.196 1719
 zone prefix 40_gatekeeper 408*
 zone prefix 35_dirgk *
 security izct password cisco
 gw-type-prefix 1#* default-technology
 no shutdown
```

Originating Gatekeeper 2

```
config terminal
gatekeeper
zone local 34_dirgk cisco.com 172.18.198.197
 zone remote 39_gatekeeper cisco.com 172.18.198.92 1719
 zone remote 35_dirgk cisco.com 172.18.198.196 1719
 zone prefix 39_gatekeeper 919*
 zone prefix 35_dirgk *
 security izct password cisco
 lrq forward-queries
 no shutdown
```

Terminating Gatekeeper 2

```
config terminal
gatekeeper
 zone local 35_dirgk cisco.com 172.18.198.196
 zone remote 40_gatekeeper cisco.com 172.18.198.91 1719
 zone remote 34_dirgk cisco.com 172.18.198.197 1719 foreign-domain
 zone prefix 40_gatekeeper 408*
 zone prefix 34_dirgk *
 security izct password cisco
 lrq forward-queries
 no shutdown
```

Cisco Access Tokens

Note The following examples do not reflect the actual display of the passwords as you would see them in output. Actual displays show the passwords as being encrypted. The displays here show them in cleartext format for clarity purposes only.

Originating Gatekeeper

In this example, LRQ messages received from the border gatekeeper authenticate the LRQ message by using the password "ogk_123." LRQ messages sent to the border gatekeeper contain the password "bgk_123" in the CAT.

```
gatekeeper
 zone remote bgk china 172.18.195.137 1719 foreign-domain
 security password-group china lrq send bgk_123
 security password-group china lrq receive ogk_123
 security zone bgk password-group china
```

Border Gatekeeper Configuration

In this example, LRQ messages received from the originating gatekeeper authenticate the LRQ message by using the password "bgk_123." LRQ messages sent to the originating gatekeeper contain the password "ogk_123" in the CAT. LRQ messages received from the terminating gatekeeper authenticate the LRQ message by using the password "bgk_123." LRQ messages sent to the terminating gatekeeper contain the password "tgk_123" in the CAT.

```
gatekeeper
 zone remote ogk usa 172.18.195.138 1719 foreign-domain
 zone remote tgk china 172.18.195.139 1719
 security password-group usa lrq send ogk_123
 security password-group usa lrq receive bgk_123
 security password-group china lrq send tgk_123
 security password-group china lrq receive bgk_123
 security zone ogk password-group usa
 security zone tgk password-group china
```

Terminating Gatekeeper Configuration

In this example, LRQ messages received from the border gatekeeper authenticate the LRQ message by using the password "tgk_123." LRQ messages sent to the border gatekeeper contain the password "bgk_123" in the CAT.

```
gatekeeper
 zone remote bgk china 172.18.195.137 1719
 security password-group china lrq send bgk_123
 security password-group china lrq receive tgk_123
 security zone bgk password-group china
```

Gatekeeper Configuration Using the Wildcard

In this example, LRQ messages are received from the terminating gatekeeper, which does not have a password group configured. Therefore, the LRQ messages received are authenticated using the password group configured for the originating gatekeeper (in this example, "ogk_123").

```
gatekeeper
 zone remote tgk china 172.18.195.137 1719 foreign-domain
 security password-group china lrq send tgk_123
 security password-group china lrq receive ogk_123
 security zone * password-group china
```

Tokenless Call Authorization

Tokenless Call Authorization

The following example shows how to configure tokenless call authorization. You create an IP ACL containing endpoints from which the gatekeeper should accept calls. After the router enters gatekeeper configuration mode, you instruct the gatekeeper to check the ACL before processing the call.

```
Router# enable
Router# configure terminal
    Router(config)# access-list 20 permit 172.16.10.190
    Router(config)# access-list 20 permit 192.16.18.2
    Router(config)# access-list 20 permit 192.16.10.12
    Router(config)# access-list 20 permit 192.16.12.1
    Router(config)# gatekeeper
    Router(config-gk)# security acl answerarq 20
```

IP Access Lists

The following example shows how to verify the IP access lists and that the gatekeeper has been configured to use them:

```
Router# show running-config
Building configuration...
.
.
.
ip access-list standard WORD
!
```

```
access-list 20 permit 172.16.10.190
access-list 20 permit 192.16.18.2
access-list 20 permit 192.16.10.12
access-list 20 permit 192.16.12.1
.
.
.
gatekeeper
 zone local herndon.cisco.com cisco.com
 security acl answerarq 20
 no shutdown
.
.
.
end
```
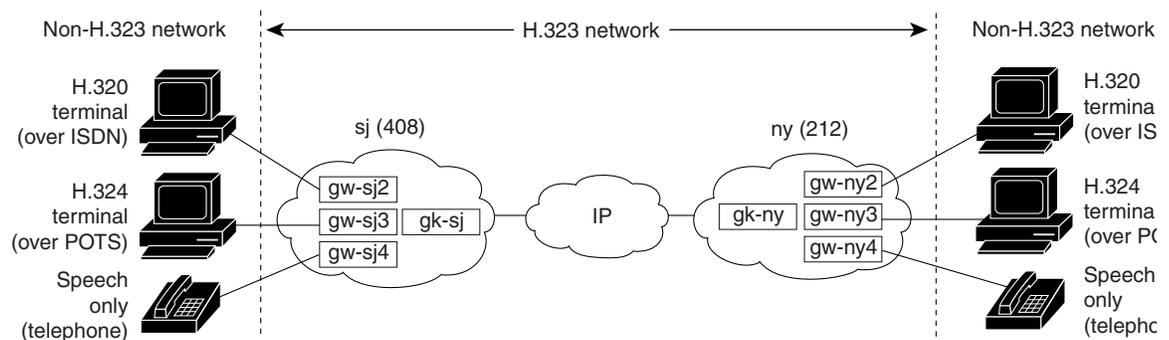
E.164 Interzone Routing: Example

Interzone routing may be configured by using E.164 addresses.

In this example, there are two gatekeepers that need to be able to resolve E.164 addresses. One is in San Jose and the other is in New York. (See Figure 13.)

Figure 13   E.164 Interzone Routing

In sj (San Jose in the 408 area code), the gateways are configured to register with gk-sj as follows:

- gw-sj2 configured to register with technology prefix 2#

- gw-sj3 configured to register with technology prefix 3#

- gw-sj4 configured to register with technology prefix 4#

Similarly, in ny (New York in the 212 area code), gateways are configured to register with gk-ny as follows:

- gw-ny2 configured to register with technology prefix 2#

- gw-ny3 configured to register with technology prefix 3#

- gw-ny4 configured to register with technology prefix 4#

For the gatekeeper for San Jose, the configuration commands are as follows:

```
gatekeeper
 zone local gk-sj cisco.com
 zone remote gk-ny cisco.com 172.21.127.27
 use-proxy gk-sj default direct
 zone prefix gk-sj 408.......
```

```
zone prefix gk-ny 212.......
gw-type-prefix 3# hopoff gk-sj
gw-type-prefix 4# default-technology
```

For the gatekeeper for New York, the configuration commands are as follows:

```
gatekeeper
 zone local gk-ny cisco.com
 zone remote gk-sj cisco.com 172.21.1.48
 use-proxy gk-ny default direct
 zone prefix gk-sj 408.......
 zone prefix gk-ny 212.......
 gw-type-prefix 3# hopoff gk-ny
 gw-type-prefix 4# default-technology
```

When a call is presented to gatekeeper gk-sj with the following target address in San Jose:

```
2#2125551212
```

Gatekeeper gk-sj recognizes that 2# is a technology prefix. It was not configured as such, but because gw-sj2 registered with it, the gatekeeper now treats 2# as a technology prefix. It strips the prefix, which leaves the telephone number 2125551212. This is matched against the zone prefixes that have been configured. It is a match for 212......., so gk-sj knows that gk-ny handles this call. Gatekeeper gk-sj forwards the entire address 2#2125551212 over to Gatekeeper gk-ny, which also looks at the technology prefix 2# and routes it to gw-ny2.

When a call is presented to gatekeeper gk-sj with the following target address in San Jose:

```
2125551212
```

Gatekeeper gk-sj checks it against known technology prefixes but finds no match. It then checks it against zone prefixes and matches on 212....... for gk-ny, and therefore routes this call to gk-ny. Gatekeeper gk-ny does not have any local registrations for this address, and there is no technology prefix on the address, but the default prefix is 4#, and gw-ny4 is registered with 4#, so the call gets routed to gw-ny4.

Another call is presented to gatekeeper gk-sj with the following target address in San Jose:

```
3#2125551212
```

The call has technology prefix 3#, which is defined as a local hopoff prefix, so gk-sj routes this call to gw-sj3, despite the fact that it has a New York zone prefix.

In this last example, a call is presented to gatekeeper gk-sj with the following target address in San Jose:

```
6505551212
```

Gatekeeper gk-sj checks for a technology prefix match but does not find one. It then searches for a zone prefix match and fails again. But there is a match for default gateway prefix of 4#, and gw-sj4 is registered with 4#, so the call is routed out on gw-sj4.

Interaction with External Applications: Example

Gatekeeper Flow Control

In the following example, server flow-control is set with an onset level of 50:

```
Router# server flow-control onset 50

*Mar  8 20:05:34.081: gk_srv_handle_flowcontrol: Flow control enabled

Router# show running-config
```

```
Building configuration...

Current configuration : 1065 bytes
!
version 12.2
no service single-slot-reload-enable
service timestamps debug datetime msec
service timestamps log uptime
no service password-encryption
!
hostname snet-3660-3
!
.
.
.
gatekeeper
 zone local snet-3660-3 cisco.com
 zone remote snet-3660-2 cisco.com 209.165.200.225 1719
 zone prefix snet-3660-2 408*
 lrq forward-queries
 no use-proxy snet-3660-3 default inbound-to terminal
 no use-proxy snet-3660-3 default outbound-from terminal
 no shutdown
 server registration-port 8000
 server flow-control onset 50
!
!
.
.
.
end
```

Retry Timer

The following example shows that the retry timer has been set to 45 seconds:

```
.
.
.
gw-type-prefix 1#* default-technology
gw-type-prefix 9#* gw ipaddr 1.1.1.1 1720
timer server retry 45
no shutdown
.
.
.
```

Registration and Call Rejection

The following example shows that the gatekeeper rejects registrations when it cannot connect to the GKTMP server:

```
.
.
.
gw-type-prefix 1#* default-technology
gw-type-prefix 9#* gw ipaddr 1.1.1.1 1720
no shutdown
server absent reject rrq
.
.
.
```

The following example shows that the gatekeeper rejects calls when it cannot connect to the GKTMP server:

```
.
.
.
gw-type-prefix 1#* default-technology
gw-type-prefix 9#* gw ipaddr 1.1.1.1 1720
no shutdown
server absent reject arq
.
.
.
```

Proxy Use: Example

Proxy for Inbound Calls

In the following example, the local zone sj.xyz.com is configured to use a proxy for inbound calls from remote zones tokyo.xyz.com and milan.xyz.com to gateways in its local zone. The sj.xyz.com zone is also configured to use a proxy for outbound calls from gateways in its local zone to remote zones tokyo.xyz.com and milan.xyz.com.

```
gatekeeper
use-proxy sj.xyz.com remote-zone tokyo.xyz.com inbound-to gateway
use-proxy sj.xyz.com remote-zone tokyo.xyz.com outbound-from gateway
use-proxy sj.xyz.com remote-zone milan.xyz.com inbound-to gateway
use-proxy sj.xyz.com remote-zone milan.xyz.com outbound-from gateway
```

Because the default mode disables proxy communications for all gateway calls, only the gateway call scenarios listed can use the proxy.

Proxy for Outbound Calls

In the following example, the local zone sj.xyz.com uses a proxy for only those calls that are outbound from H.323 terminals in its local zone to the specified remote zone germany.xyz.com:

```
gatekeeper
no use-proxy sj.xyz.com default outbound-from terminal
use-proxy sj.xyz.com remote-zone germany.xyz.com outbound-from
 terminal
```

Note that any calls inbound to H.323 terminals in the local zone sj.xyz.com from the remote zone germany.xyz.com use the proxy because the default applies.

Proxy Removal

The following example shows how to remove one or more proxy statements for the remote zone germany.xyz.com from the proxy configuration list:

```
no use-proxy sj.xyz.com remote-zone germany.xyz.com
```

The command removes all special proxy configurations for the remote zone germany.xyz.com. After the command is entered like this, all calls between the local zone (sj.xyz.com) and germany.xyz.com are processed according to the defaults defined by any **use-proxy** commands that use the **default** option.

H.235 Security

The following example shows output from configuring secure registrations from the gatekeeper and identifying which RAS messages the gatekeeper checks to find authentication tokens:

```
dial-peer voice 10 voip
 destination-pattern 4088000
 session target ras
 dtmf-relay h245-alphanumeric
!
gateway
 security password 09404F0B level endpoint
```

The following example shows output from configuring which RAS messages contain gateway-generated tokens:

```
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipx permit
radius-server host 10.0.0.1 auth-port 1645 acct-port 1646
radius-server retransmit 3
radius-server deadtime 5
radius-server key lab
radius-server vsa send accounting
!
gatekeeper
 zone local GK1 test.com 10.0.0.3
 zone remote GK2 test2.com 10.0.2.2 1719
 accounting
 security token required-for registration
 no use-proxy GK1 remote-zone GK2 inbound-to terminal
 no use-proxy GK1 remote-zone GK2 inbound-to gateway
 no shutdown
```

Prohibition of Proxy Use for Inbound Calls

To prohibit proxy use for inbound calls to H.323 terminals in a local zone from a specified remote zone, enter a command similar to the following:

no use-proxy sj.xyz.com remote-zone germany.xyz.com inbound-to terminal

This command overrides the default and disables proxy use for inbound calls from remote zone germany.xyz.com to all H.323 terminals in the local zone sj.xyz.com.

Co-Edge Proxy: Example

Co-Edge Proxy with ASR Without Subnetting

Figure 14 and the following configuration examples show how to configure RIP on the two edge networks and how to configure IGRP on the two backbone networks.

Figure 14   *Sample Configuration Without Subnetting*
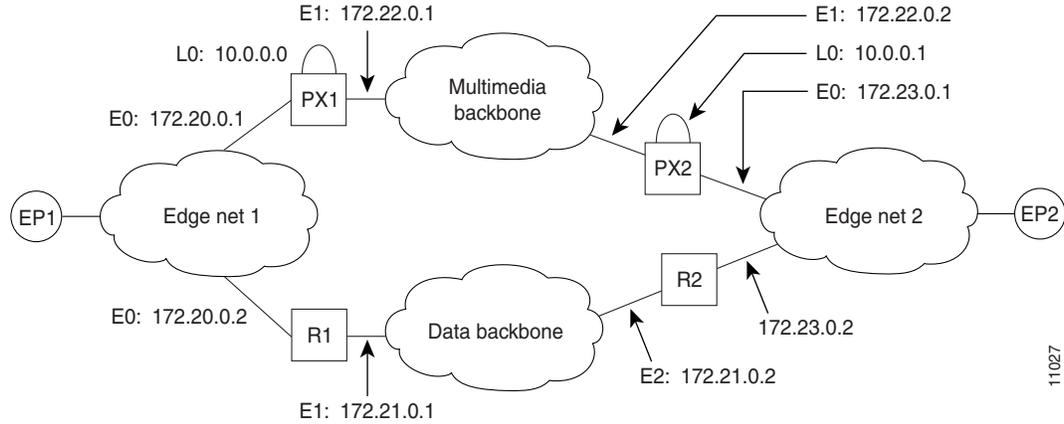
The following output is for the PX1 configuration:

```
!
proxy h323
!
interface Loopback0

 ip address 10.0.0.0 255.0.0.0
!Assume PX1 is in Zone 1, and the gatekeeper resides in the same routers as PX1:
 h323 interface
 h323 h323-id PX1@zone1.com
 h323 gatekeeper ipaddr 10.0.0.0
!
interface Ethernet0
 ip address 172.20.0.1 255.255.0.0
!
interface Ethernet1
 ip address 172.22.0.1 255.255.0.0
 ip access-group 101 in
 ip access-group 101 out
 h323 asr
!
router rip
 network 172.20.0.0
 network 10.0.0.0
!
router igrp 4000
 network 172.22.0.0
 network 10.0.0.0
!
access-list 101 permit ip any host 10.0.0.0
access-list 101 permit ip host 10.0.0.0 any
access-list 101 permit igrp any any
```

The following output is for the R1 configuration:

```
!
interface Ethernet0
 ip address 172.20.0.2 255.255.0.0
!
interface Ethernet1
 ip address 172.21.0.1 255.255.0.0
!
router rip
 redistribute igrp 5000 metric 1
```

```
 network 172.20.0.0
!
router igrp 5000
 redistribute rip metric 10000 10 255 255 65535
 network 172.21.0.0
 distribute-list 10 out
!
access-list 10 deny ip 10.0.0.0 255.255.255
access-list 10 permit any
```

Note The configuration for PX2 and R2 is the same as that for PX1 and R1.

Co-Edge Proxy with Subnetting

Figure 15 and the examples that follow show how to configure Enhanced IGRP on all networks.

Figure 15   Sample Configuration with Subnetting
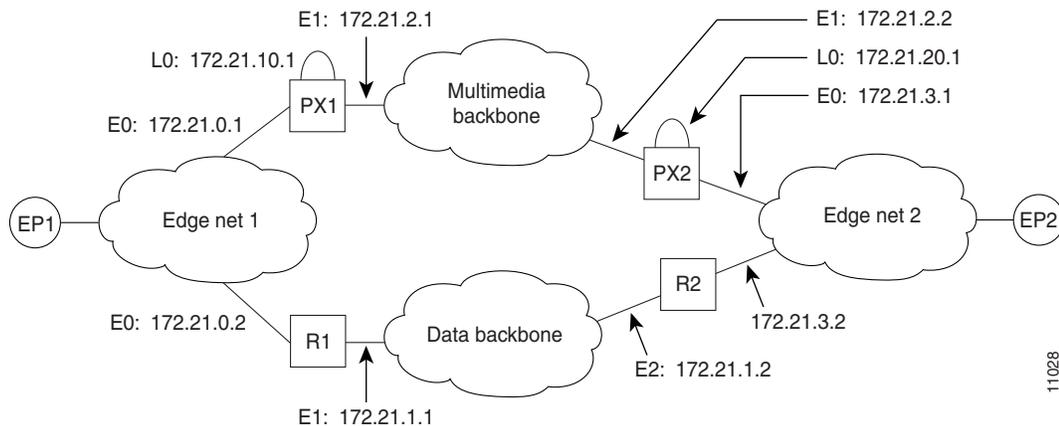
The following output is for the PX1 configuration:

```
!
proxy h323
!
interface Loopback0
 ip address 172.21.10.1 255.255.255.192
 h323 interface
 h323 h323-id PX1@zone1.com
 h323 gatekeeper ipaddr 172.21.20.1
!
interface Ethernet0
 ip address 172.21.0.1 255.255.255.192
!
interface Ethernet1
 ip address 172.21.2.1 255.255.255.192
 ip access-group 101 in
 ip access-group 101 out
 h323 asr
!
router eigrp 4000
 redistribute connected metric 10000 10 255 255 65535
 passive-interface Ethernet1
 network 172.21.0.0
 distribute-list 10 out
 no auto-summary
```

```
!
router eigrp 5000
 redistribute connected metric 10000 10 255 255 65535
 passive-interface Ethernet0
 network 172.21.0.0
 distribute-list 11 out
 no auto-summary
!
access-list 10 deny 172.21.2.0 0.0.0.63
access-list 10 permit any
access-list 11 deny 172.21.0.0 0.0.0.63
access-list 11 permit any
access-list 101 permit ip any host 172.21.10.1
access-list 101 permit ip host 172.21.10.1 any
access-list 101 permit eigrp any any
```
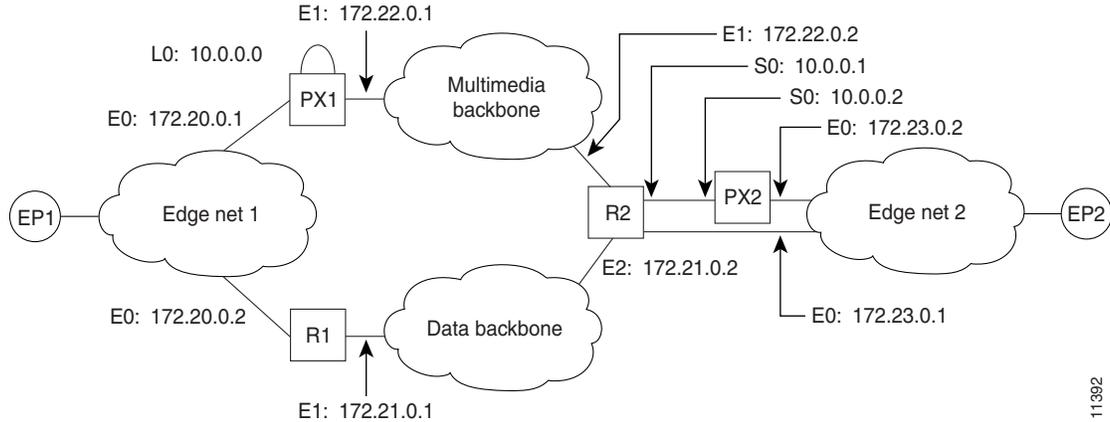
The following output is for the R1 configuration:

```
!
interface Ethernet0
 ip address 172.21.0.2 255.255.255.192
!
interface Ethernet1
 ip address 172.21.1.1 255.255.255.192
!
router eigrp 4000
 redistribute eigrp 6000 metric 10000 10 255 255 65535
 passive-interface Ethernet1
 network 172.21.0.0
 no auto-summary
!
router eigrp 6000
 redistribute eigrp 4000 metric 10000 10 255 255 65535
 passive-interface Ethernet0
 network 172.21.0.0
 distribute-list 10 out
 no auto-summary
!
access-list 10 deny 172.21.10.0 0.0.0.63
access-list 10 permit any
```

Note   The configuration for PX2 and R2 is the same as that for PX1 and R1.

Inside-Edge Proxy with ASR Without Subnetting

The configuration of the co-edge proxy in Edge net 1 has already been presented above. Figure 16 shows
the configuration of the inside-edge proxy PX2 and edge router R2 of Edge net 2. RIP is used on the
edge networks. IGRP is used on the data backbone and the multimedia backbone.

Figure 16   *Edge Net 2 with Inside-Edge Proxy and No Subnetting*

The following output is for the PX2 configuration:

```
!
proxy h323
!
interface Ethernet0
 ip address 172.23.0.2 255.255.0.0
!
interface Serial0
 ip address 10.0.0.2 255.0.0.0
 ip access-group 101 in
 ip access-group 101 out
 h323 interface
 h323 asr
 h323 h323-id PX2@zone2.com
 h323 gatekeeper ipaddr 10.0.0.2
!
router rip
 redistribute connected metric 10000 10 255 255 65535
 network 172.23.0.0
!
access-list 101 permit ip any host 10.0.0.2
access-list 101 permit ip host 10.0.0.2 any
```

The following output is for the R2 configuration:

```
!
interface Ethernet0
 ip address 172.23.0.1 255.255.0.0
!
interface Ethernet1
 ip address 172.22.0.1 255.255.0.0
 ip access-group 101 in
 ip access-group 101 out
!
interface Ethernet2
 ip address 172.21.0.2 255.255.0.0
!
interface Serial0
 ip address 10.0.0.1 255.0.0.0
!
router rip
 redistribute igrp 5000 metric 1
 network 172.23.0.0
!
```

```
router igrp 4000
 network 10.0.0.0
 network 172.22.0.0
!
router igrp 5000
 redistribute rip metric 10000 10 255 255 65535
 network 172.21.0.0
 distribute-list 10 out
!
ip route 10.0.0.2 255.255.255.255 Serial0
access-list 10 deny ip 10.0.0.0 255.255.255
access-list 10 permit any
access-list 101 permit ip any host 10.0.0.2
access-list 101 permit ip host 10.0.0.2 any
```

Note To guarantee that all traffic between the proxy and other proxies is carried over the multimedia backbone, run IGRP 4000 on the 10.0.0.0 network and on the 172.22.0.0 network. Make sure that the H.323 proxy interface address (10.0.0.2) is not advertised over the data network (distribution list 10 in IGRP 5000). Doing this also eliminates the need to configure policy routes or static routes.

QoS-Enforced Open Proxy Using RSVP

Figure 17 shows a proxy configuration that was created on a Cisco 2500 router with one Ethernet interface and two serial interfaces. Only the Ethernet interface is in use.

Figure 17 *Configuring a QoS-Enforced Open Proxy Using RSVP*
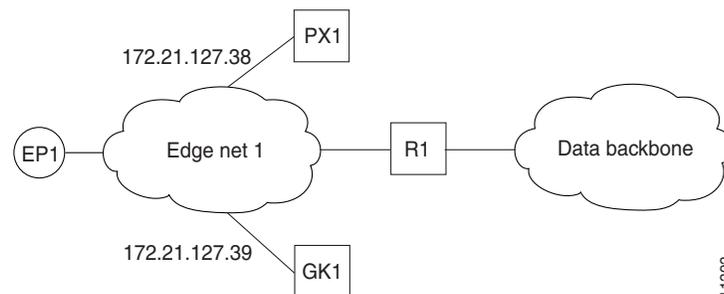
The following output is for the PX1 configuration:

```
!
version 11.3
no service password-encryption
service tcp-small-servers
!
hostname ExampleProxy
!
no ip domain-lookup
!
proxy h323
!
interface Ethernet0
 ip address 172.21.127.38 255.255.255.192
 no ip redirects
 ip rsvp bandwidth 7000 7000
 ip route-cache same-interface
 fair-queue 64 256 1000
 h323 interface
 h323 qos rsvp controlled-load
 h323 h323-id px1@zone1.com
```

```
 h323 gatekeeper ipaddr 172.21.127.39
!
interface Serial0
 no ip address
 shutdown
!
interface Serial1
 no ip address
 shutdown
!
router rip
 network 172.21.0.0
!
ip classless
!
line con 0
 exec-timeout 0 0
line aux 0
 transport input all
line vty 0 4
 password lab
 login
!
end
```

Closed Co-Edge Proxy with ASR Without Subnetting

Figure 18 shows how to configure RIP on the edge networks and IGRP on the two backbone networks. A Cisco 2500 router is used for the proxy.

Figure 18   Configuring a Closed Co-Edge Proxy with ASR
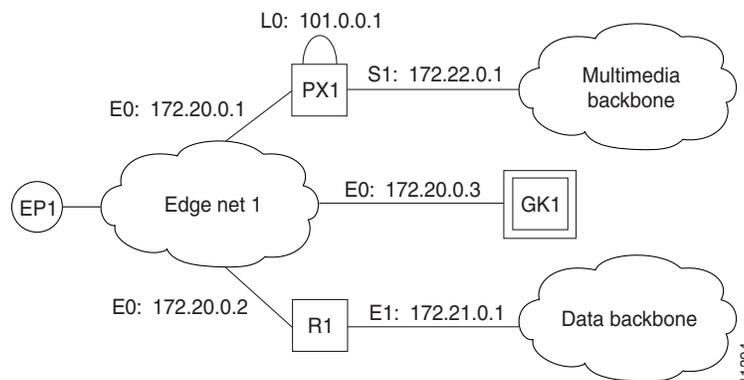
The following output is for the PX1 configuration:

```
!
version 11.3
no service password-encryption
service tcp-small-servers
!
hostname ExampleProxy
!
!
no ip domain-lookup
!
!
proxy h323
!
interface Loopback0
```

```
   ip address 10.0.0.1 255.0.0.0
   h323 interface
   h323 qos ip-precedence 4
   h323 h323-id px1@zone1.com
   h323 gatekeeper ipaddr 172.20.0.3
 !
 interface Ethernet0
  ip address 172.20.0.1 255.255.255.192
  no ip redirects
 !
 interface Serial0
  no ip address
  shutdown
 !
 interface Serial1
  ip address 172.22.0.1 255.255.0.0
  ip access-group 101 in
  ip access-group 101 out
  h323 asr
 !
 router rip
  network 172.20.0.0
  network 10.0.0.0
 !
 router igrp 4000
  network 172.22.0.0
  network 101.0.0.0
 !
 ip classless
 access-list 101 permit ip any host 10.0.0.1
 access-list 101 permit ip host 10.0.0.1 any
 access-list 101 permit igrp any any
 !
 !
 line con 0
  exec-timeout 0 0
 line aux 0
  transport input all
 line vty 0 4
  password lab
  login
```

Endpoints: Example

Alternate Endpoints

The following example shows that an alternate endpoint has been configured. There are three carrier IDs (CARRIER_ABC, CARRIER_DEF, and CARRIER_GHI).

```
gatekeeper
 zone local GK cisco.com 172.16.32.12
 zone remote gk2 cisco.com 172.32.33.44 1719
 zone prefix gk2 414*
 gw-type-prefix 919*
 no shutdown
 endpoint alt-ep h323id gwid 1.1.1.1 carrier-id CARRIER_ABC
 endpoint alt-ep h323id gwid 1.1.1.1 carrier-id CARRIER_DEF
 endpoint alt-ep h323id gwid 2.2.2.2 carrier-id CARRIER_GHI
```

The following example shows that the endpoint at 172.16.53.15 1719 has been configured as an alternate for GW10. There are no carrier IDs:

```
endpoint alt-ep h323id GW10 172.16.53.15 1719
```

Nonavailability

The following example shows that the **lrq reject-resource-low** command has been configured on the gatekeeper:

```
gatekeeper
 lrq reject-resource-low
```

Endpoint-Based Call Capacity Management

The following example shows that the maximum number of calls that GW-1 can handle is 1000:

```
Router(config)# gatekeeper
Router(config-gk)# endpoint max-calls h323id GW-1 1000
```

The following example displays concurrent calls for the endpoint. In the first call example, "Voice Capacity Max.= 10000" means that the maximum calls for the endpoint are 10000. "Avail.= 10000" indicates that currently available calls for the endpoint are 10000. "Current.= 0" shows that current active calls for the endpoint are 0. (If the endpoint is not reporting capacity and the **endpoint max-calls h323id** command is not configured, "Voice Capacity Max." and "Avail." are shown as –1.)

```
Router# show gatekeeper endpoints

        GATEKEEPER ENDPOINT REGISTRATION
        ================================
CallSignalAddr  Port  RASSignalAddr   Port  Zone Name        Type    Flags
--------------  ----  -------------   ----  ---------        ----    -----
172.18.200.27   1720  172.18.200.27   57245 GK-1             VOIP-GW
    H323-ID:GW1
    Voice Capacity Max.= 10000  Avail.= 10000  Current.= 0
172.18.200.29   1720  172.18.200.29   58703 GK-2             VOIP-GW
    H323-ID:GW2
    Voice Capacity Max.= 23  Avail.= 23  Current.= 0
Total number of active registrations = 2
```

Endpoint Unregistration

The following example shows that all endpoints have been unregistered:

```
GATEKEEPER ENDPOINT REGISTRATION
                ================================
CallSignalAddr  Port  RASSignalAddr   Port  Zone Name        Type    Flags
--------------  -----  ---------------  -----  ---------        ----    -----
Total number of active registrations = 0
```

IRR Timer and Disable IRQ Requests: Example

IRQ Messages Are Sent

The following example shows that the endpoint that is registered to the gatekeeper has sent an IRR in response to the IRQ:

```
.
.
.
gatekeeper
.
lrq reject-resource-low
```

```
no irq global-request
timer lrq seq delay 10
timer lrq window 6
timer irr period 45
no shutdown
.
.
.
```

IRQ Messages Are Not Sent

The following example shows that IRQ messages are not sent from the gatekeeper:

```
.
.
.
lrq reject-resource-low
 no irq global-request
 timer lrq seq delay 10
 timer lrq window 6
 timer irr period 6
 no shutdown
.
.
.
```

Sequential LRQ Enhancement: Example

The following example shows a gatekeeper with the Sequential LRQ Enhancement feature enabled:

```
Router# show running-config

Building configuration...

Current configuration : 1802 bytes
!
version 12.2
.
.
.
gatekeeper
 zone local Zone1 cisco.com
 zone remote c3620-1-gk cisco.com 209.165.200.225 1719
 zone remote c2514-2-gk cisco.com 209.165.200.228 1719
 zone remote gk-cisco-mn cisco.com 209.165.200.230 1719
 zone remote gkzone3 cisco.com 209.165.200.235
 zone remote gk-catapult cisco.com 209.165.200.229 1719
 zone prefix gkzone3 405.......
 zone prefix gk-gk5 515....
 zone prefix c2514-2-gk 910.......
 zone prefix c3620-1-gk 917300....
 zone prefix c2514-2-gk 919.......
 zone prefix gk-cisco-mn 919.......
 zone prefix c3620-1-gk 919.......
 lrq reject-resource-low
 lrq lrj immediate-advance
 timer lrq window 6
 no shutdown
.
.
.
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS Release 12.4	• Cisco IOS Release 12.4 Configuration Guides • Cisco IOS Release 12.4T Configuration Guides • Cisco IOS Release 12.4 Command References • Cisco IOS Voice Configuration Library http://www.cisco.com/en/US/docs/ios/12_3/vvf_c/cisco_ios_voice_configuration_library_glossary/vcl.htm > **Note** This website contains the library preface and glossary.
Cisco IOS Release 12.3	• Cisco IOS Release 12.3 documentation • Cisco IOS voice commands • Cisco IOS Voice Troubleshooting and Monitoring Guide • Tcl IVR Version 2.0 Programming Guide
Cisco IOS Release 12.2	• *Cisco IOS Voice, Video, and Fax Configuration Guide*, Release 12.2 at http://www.cisco.com/en/US/docs/ios/12_2/voice/configuration/guide/fvvfax_c.html
Troubleshooting and Debugging guides	• Cisco IOS Debug Command Reference, Release 12.4 at http://www.cisco.com/en/US/docs/ios/debug/command/reference/db_book.html • *Troubleshooting and Debugging VoIP Call Basics* at http://www.cisco.com/en/US/tech/tk1077/technologies_tech_note09186a0080094045.shtml • *VoIP Debug Commands* at http://www.cisco.com/en/US/docs/routers/access/1700/1750/software/configuration/guide/debug.html

Related Topic	Document Title
Cisco Unified Border Element Configuration Examples	• Local-to-remote network using the IPIPGW http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a00801b0803.shtml • Remote-to-local network using the IPIPGW: http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080203edc.shtml • Remote-to-remote network using the IPIPGW: http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080203edd.shtml • Remote-to-remote network using two IPIPGWs: http://www.cisco.com/en/US/tech/tk1077/technologies_configuration_example09186a0080203edb.shtml
Related Application Guides	• Cisco Unified Communications Manager and Cisco IOS Interoperability Guide • Cisco IOS Fax, Modem, and Text Support over IP Configuration Guide • "Configuring T.38 Fax Relay" chapter • Cisco IOS SIP Configuration Guide • Cisco Unified Communications Manager (CallManager) Programming Guides at: http://www.cisco.com/en/US/products/sw/voicesw/ps556/products_programming_reference_guides_list.html • *Quality of Service for Voice over IP* at http://www.cisco.com/en/US/docs/ios/qos/configuration/guide/15_0/qos_15_0_book.html

Standards

Standards	Title
ITU-T E.164	Overall network operation, telephone service, service operation and human factors
ITU-T H.225 Version 2	Call signalling protocols and media stream packetization for packet-based multimedia communication systems
ITU-T H.235	Security and encryption for H-Series (H.323 and other H.245-based) multimedia terminals
ITU-T H.323	Packet-based multimedia communications systems
ITU-T H.450	Supplementary services for multimedia

MIBs

MIBs	MIBs Link
• CISCO-GATEKEEPER-MIB	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
RFC 2833	*RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals*

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport