

Multimedia Conference Manager

Feature Summary

Deploying H.323 applications and services requires careful design and planning both on the network infrastructure and the H.323 devices. The Multimedia Conference Manager provides both gatekeeper and proxy capabilities, which are required for service provisioning and management of H.323 networks. The Multimedia Conference Manager feature provides H.323 application options previously unavailable. Using Multimedia Conference Manager, you can implement H.323-compliant applications on existing networks in an incremental fashion without upgrades. This feature also provides a rich list of networking capability, including:

- A means to implement Quality of Service (QoS), which is required for the successful deployment of H.323 applications.
- Inter-zone routing in the E.164 address space. When using H.323-ID format addresses, inter-zone routing is done through domain names.

The Multimedia Conference Manager feature provides network administration mechanisms to support H.323 applications without impacting the mission critical applications running on today's networks. Multimedia Conference Manager is implemented on Cisco IOS software. Multimedia Conference Manager provides the network administrator with these abilities:

- Identify H.323 traffic and apply appropriate policies
- Limit H.323 traffic on the local-area network (LAN) and wide-area network (WAN)
- Provide user accounting for records based on service utilization
- Insert QoS for the H.323 traffic generated by applications such as Voice over IP (VoIP), data conferencing, and video conferencing
- Implement security for H.323 communications

Multimedia Conference Manager has two principal functions: *gatekeeper* and *proxy*. This document describes the value of the Multimedia Conference Manager gatekeeper and proxy functions for end-to-end implementation of H.323-compliant multimedia applications. These functions are unique to Multimedia Conference Manager. Similar robust features are currently not available in other vendor solutions.

Gatekeeper subsystems provide:

- User authorization where authorization, authentication, and accounting (AAA) account holders are permitted to register and use the services of Multimedia Conference Manager
- Accounting using AAA call detail records
- Zone bandwidth management to limit the number of active sessions

- H.323 call routing
- Address resolution

Starting with Cisco IOS Releases 11.3(6)Q and 11.3(7)NA and later, you can configure Cisco gatekeepers to use the Cisco Hot Standby Routing Protocol (HSRP), so that when one gatekeeper fails, the standby gatekeeper assumes its role.

Proxy subsystems provide:

- H.323 traffic consolidation
- Tight bandwidth controls
- QoS mechanisms such as IP Precedence and RSVP
- Secure communication over extranets

Benefits

Use the Multimedia Conference Manager to enable your current internetwork to route bit-intensive data such as audio telephony, video and audio telephony, and data conferencing using existing telephone and ISDN links, without degrading the network's current level of service.

List of Terms

gatekeeper—A gatekeeper maintains a registry of devices in the multimedia network. The devices register with the gatekeeper when they start up, and request admission to a call from the gatekeeper.

gateway—Gateways allow H.323 terminals to communicate with non-H.323 terminals by converting protocols.

H.225.0—An ITU standard that governs H.323 session establishment and packetization. H.225.0 actually describes several different protocols: RAS, use of Q.931, and use of RTP.

H.245—An ITU standard that governs H.323 endpoint control.

H.323—An International Telecommunication Union (ITU) standard that describes packet-based video, audio, and data conferencing. H.323 is an umbrella standard that describes the architecture of the conferencing system, and refers to a set of other standards (H.245, H.225.0, and Q.931) to describe its actual protocol.

H.323 terminal—Endpoints in the multimedia network that provide real-time, two-way communications with another H.323 terminal. H.323 terminals are typically computer-based video conferencing systems.

hairpin—A term used in telephony that means to send a call back in the direction that it came from. For example, if a call cannot be routed over IP to a gateway that is closer to the target phone, the call is typically sent back out the local zone, back the way it came from.

hop off—the point at which a call transitions from H.323 to non-H.323, typically at a gateway.

multipoint control unit— A multipoint control unit (MCU) is an endpoint on the LAN that provides the capability for three or more terminals and gateways to participate in a multipoint conference.

node—An H.323 entity that uses RAS to communicate with the gatekeeper. For example, an endpoint such as a terminal, proxy, or gateway.

proxy—Proxies are special gateways that relay one H.323 session to another.

Q.931—An ITU standard that describes ISDN signaling. The H.225.0 standard uses a variant of Q.931 to establish and disconnect H.323 sessions.

Real-time Transport Protocol (RTP)—An IETF standard protocol. The H.225.0 standard describes how to use RTP to handle the packetization of video and audio in H.323.

Registration, Admission, and Status (RAS) protocol—This is the protocol used between endpoints and the gatekeeper.

T.120—An ITU standard that describes data conferencing. H.323 provides for the ability to establish T.120 data sessions inside of an existing H.323 session.

Restrictions

The Multimedia Conference Manager complies with Version 1 of the H.323 standard. This feature is meant to run with the gatekeeper or proxy functions enabled; it is not meant to be run as a regular router.

Platforms

This feature is supported on the following platforms:

- Cisco 2500 series
- Cisco 3620 and 3640

Prerequisites

To implement the Multimedia Conference Manager in your IP-based internetwork, you need to understand the following features described in this document:

- Proxies
- Gatekeepers
- Gatekeeper Zones
- H.323 Terminals
- MCUs
- Gateways

You also need to understand the following functions described in this document:

- How Terminals, Gatekeepers, and Proxies Work Together
- How Terminals, Gatekeepers, and Gateways Work Together
- How Terminals, Gatekeepers, Proxies, and MCUs Work Together

Supported MIBs and RFCs

None

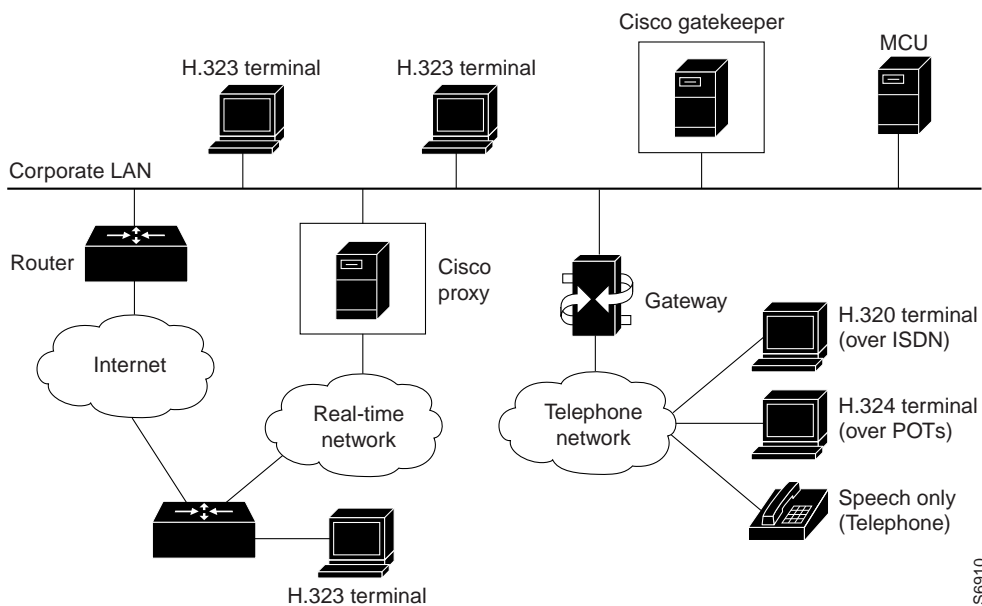
Functional Description

The Multimedia Conference Manager conforms to the H.323 standard for transmitting audio, video, and data conferencing data on an IP-based internetwork. The H.323 standard provides for the following types of *endpoints* in the network:

- Proxies
- Gatekeepers
- Gatekeeper Zones
- H.323 Terminals
- MCUs
- Gateways

Figure 1 shows a typical H.323 network:

Figure 1 Multimedia Conference Manager in an H.323 Network



Proxies

Proxies are a special type of gateway that relay H.323 data to another H.323 session. They can be used to isolate sections of an H.323 network for security purposes, to manage QoS, or to perform special application-specific routing tasks.

Gatekeepers

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

These nodes attempt to *register* with a gatekeeper upon startup. When they wish to communicate with another endpoint, they request *admission* to the call, using a symbolic alias for the endpoint name such as an E.164 address or an e-mail ID. If the gatekeeper decides the call can proceed, it

returns a destination IP address to the originating endpoint. This IP address cannot be the actual address of the target endpoint, but it can be an intermediate address. Finally, a gatekeeper and its registered endpoints exchange *status* information.

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

Gatekeeper Zones

H.323 endpoints are grouped together in zones. Each zone has one gatekeeper that manages all of the endpoints in the zone. A zone is an administrative convenience similar to a DNS domain. (Because a zone is, by definition, the area of control of a gatekeeper, you will find the terms “zone name” and “gatekeeper name” used synonymously in this chapter.)

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

H.323 Terminals

An H.323 terminal is an endpoint in the LAN that provides for real-time, two-way communications with another H.323 terminal, gateway, or MCU. This communication consists 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 videoconferencing 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 RAS protocol in order to function with the gatekeeper features.

MCUs

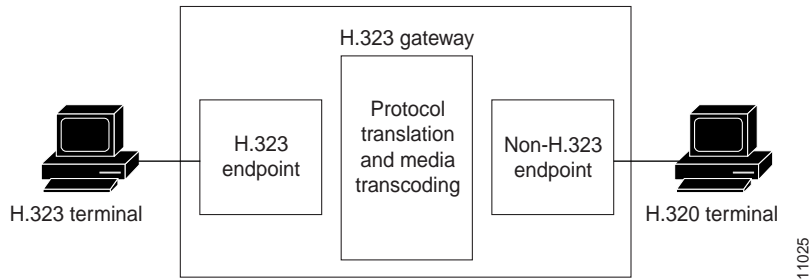
An MCU is an endpoint on the LAN that provides the capability for three or more terminals and gateways to participate in a multipoint conference. It controls and mixes video, audio, and data from terminals to create a robust videoconference. An MCU may also connect two terminals in a point-to-point conference, 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 with all the functionality mentioned above.

Gateways

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

Figure 2 Gateway between an H.323 Terminal and an H.320 Terminal



How Terminals, Gatekeepers, and Proxies Work Together

When terminals and proxies are brought online, they first attempt to discover their gatekeeper. Discovery occurs by either multicasting a discovery request, or by being configured with the name and address of the gatekeeper and unicasting a discovery request. Following successful discovery, each endpoint registers with the gatekeeper. The gatekeeper keeps track of what endpoints are online and available to receive calls.

There are three ways to set up calls between various endpoints, as described in the following sections:

- Intra-Zone Call
- Inter-Zone Call without Proxy
- Inter-Zone Call with Proxy

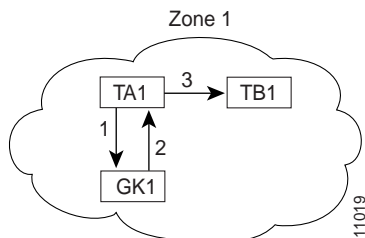
Intra-Zone Call

If terminal TA1 wants to make an intra-zone call to terminal TB1 in Zone 1, the following sequence of events occurs:

- 1 TA1 asks GK1 for permission to call TB1.
- 2 GK1 returns the address of TB1 to TA1.
- 3 TA1 then calls TB1.

Figure 3 illustrates these events.

Figure 3 Intra-Zone Call



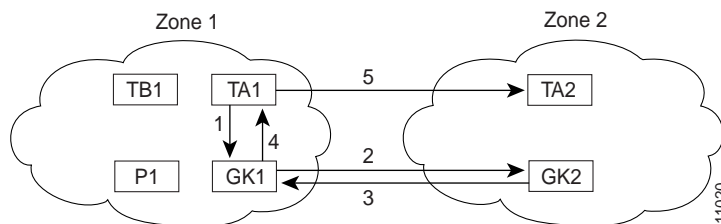
Inter-Zone Call without Proxy

If terminal TA1 in Zone 1 wants to call terminal TA2 in Zone 2 without the use of a proxy, the following sequence of events occurs:

- 1 TA1 asks GK1 for permission to call TA2.
- 2 TA2 is not in GK1's zone. GK1 locates GK2 as TA2's gatekeeper. GK1 then asks GK2 for TA2's address.
- 3 GK2 returns TA2's address to GK1.
- 4 GK1 returns the address to TA1.
- 5 TA1 calls TA2.

Figure 4 illustrates these events.

Figure 4 Inter-Zone Call without Proxy



Inter-Zone Call with Proxy

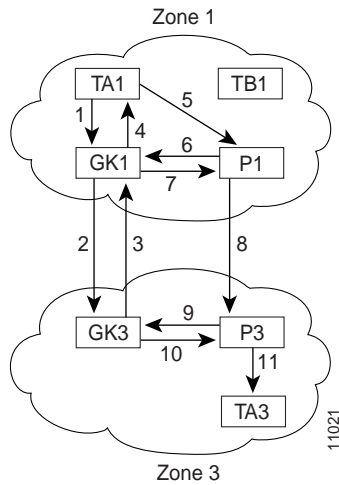
One reason for using a proxy is to isolate addressing information in one zone from another. When such isolation is desired, zones are configured as inaccessible on the gatekeepers. (Other reasons for using proxies are discussed later in this document.)

If terminal TA1 in Zone 1 wants to call terminal TA3 in Zone 3, the following sequence of events occurs:

- 1 TA1 asks GK1 for permission to call TA3.
- 2 GK1 locates GK3 as TA3's gatekeeper. GK1 asks GK3 for TA3's address.
- 3 GK3 responds with P3's address instead of TA3's, to hide TA3's identity.
- 4 GK1 knows that to get to P3, the call must go through P1. So GK1 returns P1's address to TA1.
- 5 TA1 calls P1.
- 6 P1 consults GK1 to discover the true call's destination (which is TA3 in this example).
- 7 GK1 instructs P1 to call P3.
- 8 P1 calls P3.
- 9 P3 consults GK3 for the true destination, which is TA3.
- 10 GK3 gives TA3's address to P3.
- 11 P3 completes the call to TA3.

Figure 5 illustrates these events.

Figure 5 Inter-Zone Call with Proxy

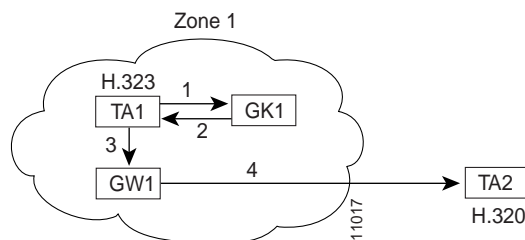


How Terminals, Gatekeepers, and Gateways Work Together

Gateways provide protocol conversion between terminals running different types of protocols. Gateways communicate with gatekeepers using the RAS protocol. The gatekeeper maintains resource computing information, which it uses to select the appropriate gateway during the admission of a call. In Figure 6:

- TA1 is an H.323 terminal registered to GK1.
- GW1 is an H.323-to-H.320 gateway registered to GK1.
- TA2 is an H.320 terminal.

Figure 6 Intra-Zone Call through Gateway



A call from TA1 to TA2 is set up as follows:

- 1 TA1 asks GK1 for permission to connect to TA2's E.164 address.
- 2 The gatekeeper looks through its local registrations and does not find any H.323 terminals registered with that E.164 address, so the gatekeeper assumes that it is an H.320 terminal outside the scope of H.323. The gatekeeper instructs TA1 to connect to the GW1 IP address.
- 3 TA1 connects to GW1.
- 4 GW1 completes the call to TA2.

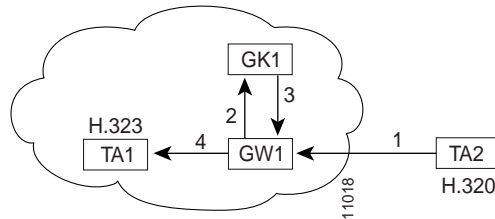
A call from TA2 to TA1 is set up as follows:

- 1 TA2 calls GW1 and provides the TA1 E.164 address as the final destination.

- 2 GW1 sends a message to GK1 asking to connect to that address.
- 3 GK1 gives GW1 the address of TA1.
- 4 GW1 completes the call with TA1.

Figure 7 illustrates these events.

Figure 7 Gateways Provide Translation between Terminal Types



How Terminals, Gatekeepers, Proxies, and MCUs Work Together

When MCUs are brought online, they first attempt to discover their gatekeeper. As with terminals and proxies, this is done by either multicasting a discovery request, or by being configured with the name and address of the gatekeeper and unicasting a discovery request. Following successful discovery, the MCU registers with the gatekeeper. The gatekeeper keeps track of what endpoints are online and available to receive calls.

There are three ways to set up an MCU conference call, as described in the following sections:

- Intra-Zone MCU Conference Call
- Inter-Zone MCU Conference Call without Proxy
- Inter-Zone MCU Conference Call with Proxy

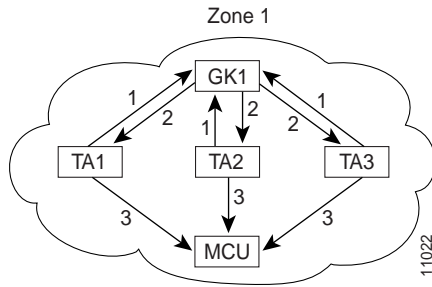
Intra-Zone MCU Conference Call

An MCU conference in *Zone 1* is created with the conference ID *CompanyMeeting*. The MCU re-registers itself with the gatekeeper with the new conference ID appended to its list of existing aliases. If terminals TA1, TA2, and TA3 in *Zone 1* want to join *CompanyMeeting*, the following sequence of events occurs:

- 1 TA1, TA2, and TA3 join the conference by asking GK1 for permission to call the given conference ID.
- 2 GK1 returns the address of the MCU to TA1, TB1, and TC1.
- 3 TA1, TA2, and TA3 then call the MCU.

Figure 8 illustrates these events.

Figure 8 Intra-Zone Call with MCU

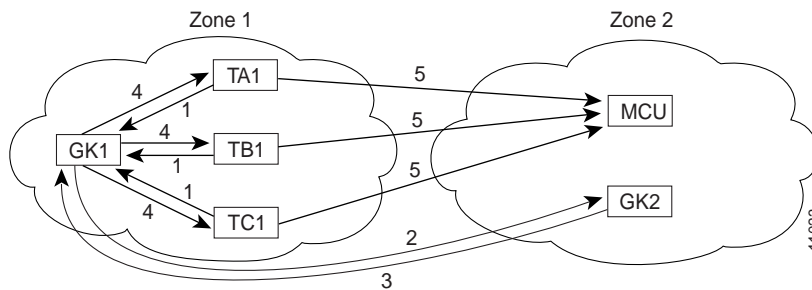


Inter-Zone MCU Conference Call without Proxy

The MCU in Zone 2 creates a conference with conference ID *CompanyMeeting@zone2.com*. The MCU re-registers itself with GK2, with the new conference ID appended to its list of existing aliases. Terminals TA1, TB1, and TC1 in Zone 1 want to join the MCU conference call with the conference ID *CompanyMeeting@zone2.com* in Zone 2. The following sequence of events occurs:

- 1 TA1, TB1, and TC1 ask GK1 for permission to join the conference.
- 2 GK1 locates GK2 for the remote zone containing conference *CompanyMeeting@zone2.com*. This is accomplished using DNS or information configured on GK1. GK1 sends a request to GK2 to recover the MCU's address.
- 3 GK2 gives to MCU's address to GK1.
- 4 GK1 gives the MCU's address to TA1, TB1, and TC1, and instructs these endpoints to set up the call with the MCU.
- 5 TA1, TB1, and TC1 then call the MCU.

Figure 9 Inter-Zone MCU Conference Call without Proxies



Inter-Zone MCU Conference Call with Proxy

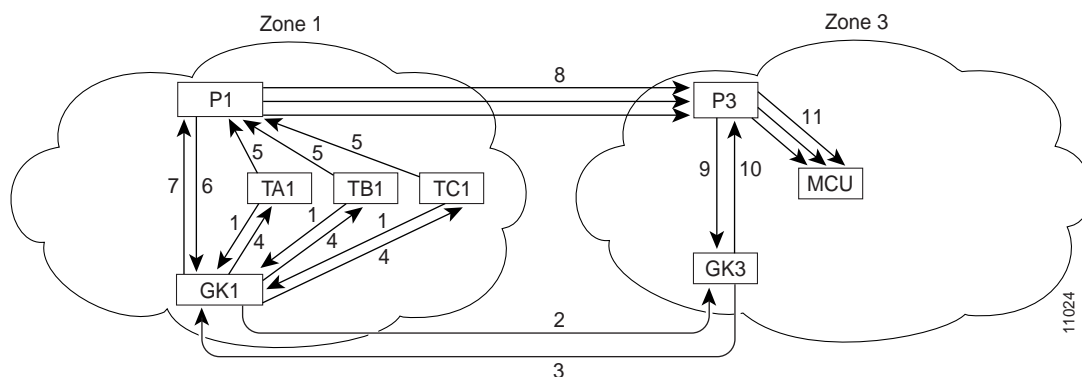
One of the main reasons for using a proxy is to isolate addressing information in one zone from another. When such isolation is desired, zones are configured inaccessible on the gatekeepers.

The MCU in Zone 3 creates a conference with the conference ID *CompanyMeeting@zone3.com*. The MCU re-registers itself with the gatekeeper, with the new conference ID appended to its list of existing aliases. Terminals TA1, TB1, and TC1 in Zone 1 want to join the MCU conference with the conference ID *CompanyMeeting@zone3.com* in Zone 3. The following sequence of events occurs:

- 1 TA1, TB1, and TC1 ask GK1 for permission to join the conference *CompanyMeeting@zone3.com*.
- 2 GK1 locates GK3 for the remote zone containing conference *CompanyMeeting@zone3.com*. GK1 asks GK3 for the MCU's address.
- 3 GK3 responds with PX3's address instead of the MCU's. GK1 knows that to get to PX3 the call should go through P1.
- 4 GK1 gives P1's address to TA1, TB1 and TC1.
- 5 TA1, TB1, and TC1 call P1.
- 6 P1 consults GK1 to discover the true call destination, which is the *CompanyMeeting@zone3.com* in this example.
- 7 GK1 instructs P1 to call P3.
- 8 P1 calls P3.
- 9 P3 consults with GK3 to discover the true call destination, which is the *CompanyMeeting@zone3.com* in this example.
- 10 GK3 gives the MCU's address to PX3.
- 11 P3 completes the call with the MCU.

Figure 10 illustrates these events.

Figure 10 Inter-Zone MCU Conference Call with Proxy



Gatekeeper Features

The following sections describe the main features of a gatekeeper in an H.323 network:

- Zone and Subnet Configuration
- Terminal Name Registration
- Inter-Zone Communication
- Endpoint Identification via RADIUS/TACACS+
- Accounting via RADIUS/TACACS+
- Inter-Zone Routing Using E.164 Addresses

Zone and Subnet Configuration

A zone is defined as the set of H.323 nodes controlled by a single gatekeeper. Gatekeepers co-existing on a network may be configured so that they register endpoints from different subnets.

Endpoints attempt to discover a gatekeeper, and consequently what zone they are members of, by using the 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 non-deterministically with the first gatekeeper to respond. To enforce predictable behavior, where endpoints on certain subnets are assigned to specific gatekeepers, you can use the **zone subnet** command to define the subnets that constitute a given gatekeeper's zone. Any endpoint on a subnet that is not enabled for the gatekeeper will not be accepted as a member of that gatekeeper's zone. If the gatekeeper receives a discovery message from such an endpoint, it will send an explicit reject message.

Terminal Name Registration

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

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

If an H.323 network deploys inter-zone communication, each terminal should at least have a fully-qualified e-mail name as its H.323 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*.

Inter-Zone Communication

To allow endpoints to communicate between zones, gatekeepers must be able to determine which zone an endpoint is in and locate the gatekeeper responsible for that zone. If DNS is available, you can associate a DNS domain name to each gatekeeper. See the DNS configuration task in the section "Setting Up Inter-Gatekeeper Communication" on page 21 to understand how to do this.

Endpoint Identification via RADIUS/TACACS+

Version 1 of the H.323 specification does not provide a mechanism for authenticating registered endpoints. No credential information is passed. However, by enabling authorization, authentication and accounting (AAA) on the gatekeeper and configuring for RADIUS/TACACS+, you can achieve a rudimentary form of identification.

If you enable this feature, the gatekeeper attempts to use the registered aliases along with a password, and do an authentication transaction to a RADIUS/TACACS+ server. The registration will only be accepted if RADIUS/TACACS+ successfully authenticates the name.

The gatekeeper can be configured to use a default password 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 by using it to separate the ID and password fields.

Note The names loaded into RADIUS/TACACS+ are probably not the same names provided for dial access, since they may all have the same password.

Accounting via RADIUS/TACACS+

If you enable AAA on the gatekeeper, the gatekeeper will emit an accounting record each time an endpoint registers or unregisters, or each time a call is admitted or disconnected.

Inter-Zone Routing Using E.164 Addresses

Starting with Cisco IOS Releases 11.3(6)Q and 11.3(7)NA and later, you can configure inter-zone routing using E.164 addresses.

There are two types of address destinations used in H.323 calls. The destination can be specified using either an H.323-ID address (a character string) or an E.164 address (a string containing telephone keypad characters). The way inter-zone calls are routed depends on the type of address being used.

When using H.323-ID addresses, inter-zone routing is handled through the use of domain names. For example, to resolve the domain name *bob@cisco.com*, the source endpoint's gatekeeper finds the gatekeeper for *cisco.com* and sends it the location request for 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 means of 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.

Using the **zone prefix** command, you can define gatekeeper responsibilities for area codes. The command can also be used to both tell the gatekeeper which prefixes are in its own zones, as well as 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, or 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 returns 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.

Since non-H.323 devices do not register with gatekeepers, gk-ny has no knowledge of what device the address belongs to, or what type of device it is, so the gatekeeper cannot decide what gateway should be used to *hop off* to the non-H.323 device. (The term *hop off* refers to the point where 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 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.

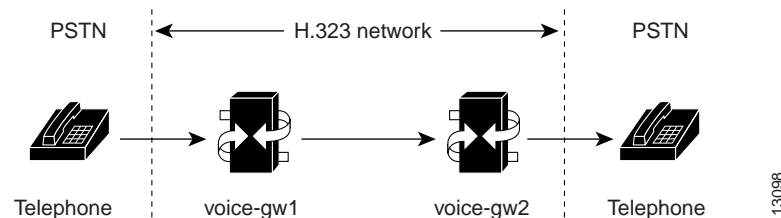
To enable the gatekeeper to select the appropriate hop-off gateway, use the **gw-type-prefix** command to configure technology or gateway-type prefixes. You can select technology prefixes to denote different types or classes of gateways. The gateways are then configured to register with their gatekeepers with these technology prefixes.

For example, voice gateways might register with technology prefix 1#, H.320-gateways with technology prefix 2#, and so on. If you have several gateways of the same type, you should configure them to register with the same prefix type. By doing this, 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.

You will need to let callers know the technology prefixes you define. The callers will need to know the type of device they are trying to reach, and will need to prepend the appropriate technology prefix to the destination address to indicate the type of gateway to reach the destination.

As an example, callers might request 1#2125551111 if they know that address 2125551111 is for a telephone and 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 it 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 where the call scenario is as follows:



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.

Additionally, the **gw-type-prefix** command can provide the ability to define a particular gateway-type prefix as being the default gateway type to be used for unresolveable addresses, and also to force 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 what you use in your network are mostly voice gateways, and you have configured all your voice gateways to register with technology prefix 1#, you can configure your gatekeeper to use 1# gateways as the default technology by entering the following command:

```
my-gatekeeper(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 will be 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 will route the call to one of its local (1#) voice gateways, so 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, it is recommended that you use the same prefix type to denote the same gateway type in all zones under your administration. No more than 50 different technology prefixes should be registered per zone.

The **gw-type-prefix** command also provides the ability to force a hop-off to a particular zone. When an endpoint or gateway makes a call-admission request to its gatekeeper, the gatekeeper resolves 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 resolves to a remote zone, the entire address including both 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 over-ridden by associating a forced hop-off zone with a particular technology prefix. This 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. You can define the gateway type prefix for H.320 gateways as one that is always forced to the Denver zone. The forced hop-off zone can either be a local zone (that is, one that is managed by the local gatekeeper) or a remote zone.

Proxy Features

Each of the following sections describes a reason for using the proxy feature in an H.323 network:

- Security
- Quality of Service
- Application-Specific Routing

Security

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 your network security. The configuration that you choose depends on how capable your firewall is of handling the complex H.323 protocol suite. Each of the following sections describes a common configuration for using the proxy with a firewall:

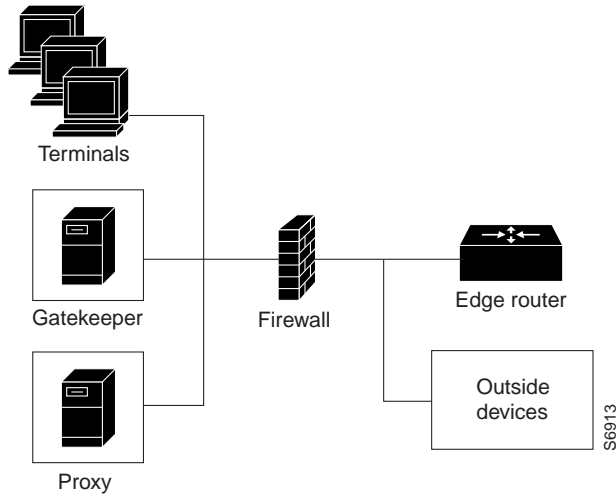
- Proxy inside the Firewall

- Proxy in Co-Edge Mode
- Proxy Outside the Firewall
- Proxies and Network Address Translation

Proxy inside the Firewall

H.323 is a complex, dynamic protocol, consisting 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, you can use a proxy just inside your firewall. The proxy provides a simple access control scheme, as illustrated in Figure 11.

Figure 11 Proxy inside the Firewall

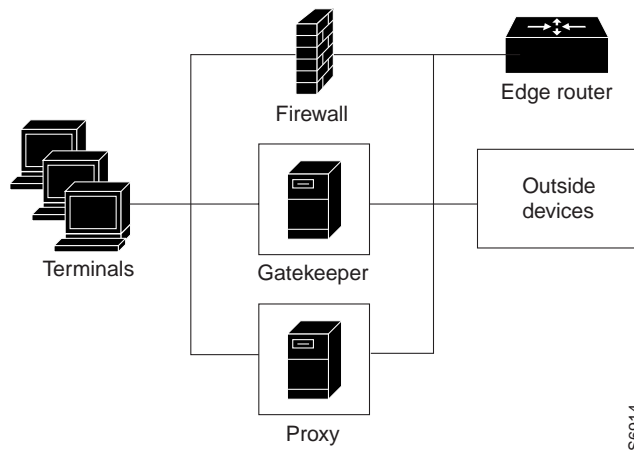


Since the gatekeeper (using RAS) and the proxy (using call setup protocols) are the only endpoints that communicate with other devices outside the firewall, it is simple to setup a tunnel through the firewall to allow traffic destined for either of these two endpoints.

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, then 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 12.

Figure 12 Proxy in Co-edge Mode



In co-edge mode, the proxy can present a security risk. To avoid exposing your network to unsolicited traffic, configure the proxy to route only proxied traffic. In other words, the proxy only routes H.323 protocol traffic that is terminated on the inside and then repeated to the outside. You can configure this for traffic moving in the opposite direction as well.

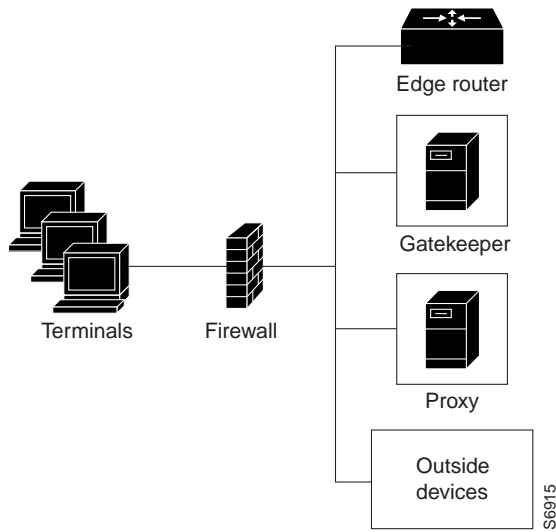
Proxy Outside the Firewall

If you would like to place the proxy and gatekeeper outside your firewall, two conditions must exist. First, your 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 will quickly overwhelm the firewall since a large number of relatively static, internal-to-external address mappings are maintained.

If the firewall does not support H.323 dynamic access control, you could configure the firewall 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 13 illustrates proxy outside the firewall.

Figure 13 Proxy Outside the Firewall



Proxies and Network Address Translation

When a firewall is providing network address translation 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. Table 1 and Table 2 provide guidelines for proxy deployment with NAT.

Table 1 Guidelines for Networks Using 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 2 Guidelines for Networks Not Using NAT

For Networks Not Using NAT	Firewall with H.323. NAT	Firewall without H.323 NAT
	Gatekeeper and proxy inside the firewall.	Gatekeeper and proxy inside the firewall.
Firewall with Dynamic Access Control	Gatekeeper and proxy outside 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.

Quality of Service

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. It also 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 the Multimedia Conference Manager. By using pairs of proxies between regions of the network where QoS can be requested, overall call quality in the multimedia network can be dramatically improved.

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 either separate from the public network or runs overlaid with the network as a premium service with adequate QoS.

Adequate QoS usually requires terminals that know how to signal such premium services. There are two major ways to achieve such signaling:

- RSVP to reserve flows with adequate QoS based on the media codecs of H.323 traffic.
- IP precedence bits to signal that the H.323 traffic is special and deserves higher priority.

Unfortunately, the vast majority of H.323 terminals cannot achieve signaling in either of these ways.

You can configure the proxy to use any combination of RSVP and IP precedence bits.

The proxy is not capable of modifying the QoS between the terminal and itself. To achieve the best overall QoS, you should ensure that terminals are connected to the proxy using a network that intrinsically has good QoS. In other words, you can configure a path between a terminal and proxy that provides good bandwidth, delay, and packet-loss characteristics without the terminal having to request special QoS. A high-bandwidth LAN works well for this.

Application-Specific Routing

In order to achieve adequate QoS, you may deploy a separate network 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).

ASR is simple. When the proxy receives outbound traffic, it directs traffic to an interface connected directly to the QoS network. The proxy does not send the traffic using an interface specified for the regular routing protocol. Similarly, inbound traffic from other proxies is received on the interface connected to the QoS network. This is true as long as all of these other proxies around the QoS network use ASR in a consistent fashion. Finally, ASR 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's address is visible to both the ASR interface and all regular interfaces, but no routes are propagated to or from the ASR interface. This ensures that no non-H.323 traffic is routed through the ASR interface.

Configuration Tasks

The following two sections describe the tasks for configuring the gatekeeper and proxy:

- Gatekeeper Configuration

- Proxy Configuration

Gatekeeper Configuration

This section describes the following configuration tasks for configuring the gatekeeper:

- Starting a Gatekeeper
- Setting Up Inter-Gatekeeper Communication
- Controlling Zone Accessibility
- Defining Static Nodes
- Identifying H.323 Users via RADIUS/TACACS+
- Sending Accounting Records to RADIUS/TACACS+
- Inter-Zone Routing Using E.164 Addresses

Starting a Gatekeeper

To enter gatekeeper configuration mode and start the gatekeeper, complete the following tasks beginning in privileged EXEC mode:

Task	Command
Enter global configuration mode.	configure { terminal memory network }
Enter gatekeeper configuration mode.	gatekeeper
Create the gatekeeper name.	zone local <i>gatekeeper-name domain-name</i>
Define a set of subnets that constitute the gatekeeper's zone. Enable the gatekeeper for each of these subnets, and disable it for all other subnets. or	zone subnet <i>local-gatekeeper-name subnet-address {/bits-in-mask mask-address}</i> enable (repeat for as many subnets as there are) no zone subnet <i>local-gatekeeper-name</i> default enable
Define the zone as being all but a set of subnets by disabling a set of subnets and enabling all others. or	no zone subnet <i>local-gatekeeper-name subnet-address {/bits-in-mask mask-address}</i> enable (repeat for as many subnets as there are)
Accept the default behavior, which is that all subnets are enabled.	zone subnet <i>gatekeeper-name</i> default enable
Bring the gatekeeper online.	no shutdown

The *local-gatekeeper-name* should be a DNS host name if DNS is to be used to locate remote zones.

You can use the **zone subnet** command more than once to create a list of subnets controlled by a gatekeeper. The subnet masks do not have to match actual subnets in use at your site. For example, to specify a particular endpoint, you can supply its address with a 32-bit netmask.

If a local gatekeeper name is contained in the message, it must match the *local-gatekeeper-name*.

Note If you want to explicitly enable or disable a particular endpoint, specify its host address with a 32-bit subnet mask.

Setting Up Inter-Gatekeeper Communication

This section describes two ways for setting up inter-gatekeeper communication:

- Via DNS
- Manual Configuration

Via DNS

To configure inter-gatekeeper communication using DNS, perform the following tasks starting in EXEC mode:

Task	Command
Enter global configuration mode.	configure { terminal memory network }
Specify the DNS server address.	ip name-server <i>dns-server-name</i>
Set the default domain name for the Cisco IOS.	ip domain-name <i>domain-name</i>

For all gatekeepers in the system, enter into DNS a text record of the form:

```
ras [gk-id@] host [:port] [priority]
```

The *gk-id* argument is an optional gatekeeper ID. If this is not specified, *host* is used as the gatekeeper ID.

The *host* argument is either an IP address or the actual host name of the gatekeeper in the form *host.some_domain.com*.

The *port* argument, if specified, should be some port number other than RAS port 1719.

The *priority* argument specifies the order in which the listed gatekeepers should be searched for endpoints. Gatekeepers with lower priorities are searched before those with higher numbers.

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.com      in   txt   “ras gk.zone1.com”
zone2.com      in   txt   “ras gk2@gk.zone2.com”
zone3.com      in   txt   “ras gk3@gk.zone3.com:1725”
zone4.com      in   txt   “ras gk4@gk.zone4.com:1725 123”
zone5.com      in   txt   “ras gk5@101.0.0.1:1725”
```

Manual Configuration

If you choose not to use DNS, or DNS is unavailable, you can configure inter-gatekeeper communication manually. On each gatekeeper in your network, repeat the following task for every other gatekeeper in the network:

Task	Command
Configure inter-gatekeeper communication with other gatekeepers in your network by entering the following command for each gatekeeper.	zone remote <i>other-gatekeeper-name other-domain-name other-gatekeeper-ip-address</i> [<i>port-number</i>]

Controlling Zone Accessibility

By default, a gatekeeper will offer a local proxy's IP address when queried by a remote gatekeeper (synonymous with remote zone). This is considered *proxied* access. By using the **zone access** command, you can configure the local gatekeeper to offer the local endpoint's address instead of the local proxy's address. This is considered *direct* access.

Note The **zone access** command, configured on your local gatekeeper, only affects 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 will use a proxy only if the remote gatekeeper offers a proxy at the remote end. A call between two endpoints in the same zone will always be a direct (non-proxied) call.

To make your local zone directly accessible to a small set of remote zones and accessible using proxies to all other remote zones, perform the following tasks in gatekeeper configuration mode:

Task	Command
Make your local zone directly accessible to a remote zone (proxies not required). Repeat this command for each remote zone that you wish to give direct access to your local zone.	zone access <i>local-zone-name</i> remote-zone <i>remote-zone-name</i> direct
Make your local zone accessible via proxy to all other remote zones.	zone access <i>local-zone-name</i> default proxied

To make your local zone accessible using proxies to a small set of remote zones and directly accessible to all other remote zones, perform the following tasks in gatekeeper configuration mode:

Task	Command
Make your local zone accessible via proxies to a remote zone. Repeat this command for each remote zone that you wish to give proxied access to your local zone.	zone access <i>local-zone-name</i> remote-zone <i>remote-zone-name</i> proxied
Make your local zone directly accessible to all other remote zones.	zone access <i>local-zone-name</i> default direct

Defining Static Nodes

In some cases, a particular terminal or endpoint's registration information is not accessible from any gatekeeper. This may be because the endpoint doesn't 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 do this, complete the following steps:

Step 1 Obtain the endpoint's address.

Step 2 If you have not done so already, define your local zone by performing the following task in gatekeeper configuration mode:

Task	Command
Define your local zone.	zone local <i>gatekeeper-name domain-name</i>

Step 3 For each H.323 ID you wish to add for the endpoint, perform the following task in gatekeeper configuration mode:

Task	Command
Create a static entry in your local alias table for an H.323 ID.	alias static <i>ip-signaling-addr port gkid gatekeeper-name</i> terminal h323id <i>h323-id</i>

Step 4 For each E.164 address you wish to add for the endpoint, perform the following task in gatekeeper configuration mode:

Task	Command
Create a static entry in your local alias table for an E.164 address.	alias static <i>ip-signaling-addr port gkid gatekeeper-name</i> terminal e164 <i>e164-address</i>

Identifying H.323 Users via RADIUS/TACACS+

To identify H.323 users via RADIUS/TACACS+, perform the following steps:

Step 1 Enable the use of AAA accounting in login authentication mode. Perform the following tasks in global configuration mode:

Task	Command
Specify AAA accounting.	aaa new-model
Enable login authentication for H.323 users via RADIUS/TACACS+.	aaa authentication login h323 radius

Step 2 Enable use of the RADIUS/TACACS+ server. Perform the following tasks beginning in EXEC mode:

Task	Command
Enter global configuration mode.	configure { terminal memory network }
Specify the RADIUS/TACACS+ server.	radius-server host { <i>hostname</i> <i>ip-address</i> }
Specify the RADIUS/TACACS+ server key.	radius-server key <i>keystring</i>

Step 3 Configure the gatekeeper to use a particular type of alias as the RADIUS/TACACS+ identification:

Task	Command
Enter gatekeeper configuration mode.	gatekeeper
Identify users by their H.323 ID. or	security h323-id
Identify users by their E.164 address. or	security e164
Identify users by either their E.164 address or H.323 ID.	security any

Step 4 Select a password that will be used by all users:

Task	Command
Pick a default security password. or	security password default <i>password</i>
Pick a character to be used by endpoints to separate the ID from the password, thereby entering both in the H.323-ID registrations.	security password separator <i>character</i>

Step 5 Enter each user into the RADIUS/TACACS+ database, using either the default password if using the **security password default** command, or actual passwords if using the piggybacked password mechanism as the RADIUS/TACACS+ authentication for that user. Enter either the user's H.323-ID or E.164 address, depending on how you configured the gatekeeper.

Sending Accounting Records to RADIUS/TACACS+

To send accounting records to RADIUS/TACACS+, perform the following steps:

Step 1 Enable the use of AAA accounting in connection accounting mode. Perform the following tasks in global configuration mode:

Task	Command
Specify AAA accounting.	aaa new-model
Enable connection accounting mode using RADIUS/TACACS+.	aaa accounting connection start-stop radius

Step 2 Set up the RADIUS/TACACS+ server. Perform the following tasks in global configuration mode:

Task	Command
Enter global configuration mode.	configure { terminal memory network }
Specify the RADIUS/TACACS+ server.	radius-server host { <i>hostname</i> <i>ip-address</i> }
Specify the RADIUS/TACACS+ server key.	radius-server key <i>keystring</i>

Step 3 Enable gatekeeper accounting. Perform the following tasks in global configuration mode:

Task	Command
Enter global configuration mode.	configure { terminal memory network }
Enter gatekeeper configuration mode.	gatekeeper
Enable gatekeeper accounting.	accounting

Enabling E.164 Inter-Zone Routing

Starting with Cisco IOS Releases 11.3(6)Q and 11.3(7)NA and later, you can configure inter-zone routing using E.164 addresses.

To configure inter-zone routing in the E.164 address space, perform the following tasks beginning in gatekeeper configuration mode:

Task	Command
Enter gatekeeper configuration mode.	gatekeeper
Create the local gatekeeper name.	zone local <i>gatekeeper-name domain-name</i>
Create the remote gatekeeper name.	zone remote <i>other-gatekeeper-name other-domain-name other-gatekeeper-ip-address [port-number]</i>
Configure knowledge of the zone's prefixes.	zone prefix <i>gatekeeper-name e164-prefix</i>
Define technology prefixes.	gw-type-prefix <i>type-prefix [hopoff gkid]</i> [default-technology] [[gw ipaddr <i>ipaddr</i> [<i>port</i>]] ...

Proxy Configuration

This section describes the following configuration tasks for configuring the proxy:

- Starting a Proxy without Application-Specific Routing
- Starting a Proxy with Application-Specific-Routing

Starting a Proxy without Application-Specific Routing

To start the proxy without ASR, complete the following tasks.

Step 1 Perform the following tasks in global configuration mode:

Task	Command
Start the proxy.	proxy h323
Enter interface configuration mode on the interface whose IP address is to be used by the proxy.	interface <i>type number</i>
Signal to the proxy that this interface's IP address is the one to use.	h323 interface
Configure the proxy name. (More than one name can be configured if necessary.)	h323 h323-id <i>h323-id</i>
Set up the zone in which the proxy will run.	
Unicast gatekeeper discovery.	h323 gatekeeper [<i>id gatekeeper-id</i>] ipaddr <i>ipaddr</i>
or	
Multicast gatekeeper discovery.	h323 gatekeeper [<i>id gatekeeper-id</i>] multicast
Configure RSVP or IP precedence.	
Set up IP precedence.	h323 qos ip-precedence <i>value</i>
and/or	
Set up RSVP with controlled-load class of service.	h323 qos rsvp controlled-load
or	
Set up RSVP with guaranteed class of service.	h323 qos rsvp guaranteed-qos

Step 2 Perform the following task in interface configuration mode:

Task	Command
For every interface, enable fast switching for packets received and forwarded on the same interface.	ip route-cache same-interface

Starting a Proxy with Application-Specific-Routing

If ASR is to be enabled on the proxy, perform the following steps, starting in global configuration mode:

Step 1 Create a loopback interface and start the proxy.

Task	Command
Start the proxy.	proxy h323
Enter loopback interface configuration mode.	interface Loopback0

Task	Command
Define the interface subnet.	ip address <i>ipaddr subnet-mask</i> or ip address <i>ipaddr/subnet-mask-size</i>
Signal to the proxy that this interface's IP address is the one to use.	h323 interface
Configure the proxy name. (More than one name can be configured if necessary.)	h323 h323-id <i>h323-id</i>
Set up the zone in which the proxy will run.	
Unicast gatekeeper discovery.	h323 gatekeeper [<i>id gatekeeper-id</i>] ipaddr <i>ipaddr</i> or
Multicast gatekeeper discovery.	h323 gatekeeper [<i>id gatekeeper-id</i>] multicast
Configure RSVP or IP precedence.	
Set up IP precedence.	h323 qos ip-precedence <i>value</i> and/or
Set up RSVP with controlled-load class of service.	h323 qos rsvp controlled-load or
Set up RSVP with guaranteed class of service.	h323 qos rsvp guaranteed-qos

Step 2 Determine which interface is to route the H.323 traffic and configure ASR on it.

Task	Command
If application-specific routing is to be used, enter interface configuration mode on the interface through which outbound H.323 traffic should be routed.	interface <i>type number</i>
Enable application-specific routing with no bandwidth limitations.	h323 asr or
Enable ASR and limit the total bandwidth of H.323 traffic through the ASR interface.	h323 asr bandwidth <i>bandwidth</i>

Note ASR is not supported on Frame Relay or ATM interfaces for the Cisco MC3810 platform.

Step 3 The ASR interface and all other interfaces must be separated so that they never propagate routing information between each other. There are two ways to do this:

Method 1

Use one type of routing protocol on the ASR interface and another on all of the non-ASR interfaces, with the loopback subnet included in both routing domains. Example:

Task	Command
Enter interface configuration mode on the ASR interface.	interface <i>type number</i>

Task	Command
Set up the ASR interface network number.	ip address <i>asr-addr asr-subnet-mask</i>
Go back to global configuration mode.	Ctrl-Z
Enter interface configuration mode on a non-ASR interface.	interface <i>type number</i>
Set up a non-ASR interface network number.	ip address <i>non-asr-addr non-asr-subnet-mask</i>
Go back to global configuration mode.	Ctrl-Z
Configure RIP for a non-ASR interface.	router rip
Include a non-ASR interface in RIP domain.	network <i>non-asr-addr</i>
Include a loopback interface in RIP domain.	network <i>loopback-addr</i>
Go back to global configuration mode.	Ctrl-Z
Configure IGRP for an ASR interface.	router igrp <i>asr-autonomous-system-number</i>
Include an ASR interface in an IGRP domain.	network <i>asr-addr</i>
Include a loopback interface in an IGRP domain.	network <i>loopback-addr</i>
Exit.	Ctrl-Z

Method 2

Set up two different autonomous systems, one which contains the ASR network and the loopback network, the other which contains the other non-ASR networks and loopback network. Example:

Task	Command
Enter interface configuration mode on an ASR interface.	interface <i>type number</i>
Set up an ASR interface network number.	ip address <i>asr-addr asr-subnet-mask</i>
Go back to global configuration mode.	Ctrl-Z
Enter interface configuration mode on a non-ASR interface.	interface <i>type number</i>
Set up a non-ASR interface network number.	ip address <i>non-asr-addr non-asr-subnet-mask</i>
Go back to global configuration mode.	Ctrl-Z
Configure IGRP for a non-ASR interface.	router igrp <i>non-asr-autonomous-system-number</i>
Include a non-ASR interface in RIP domain.	network <i>non-asr-addr</i>
Include a loopback interface in RIP domain.	network <i>loopback-addr</i>
Go back to global configuration mode.	Ctrl-Z
Configure IGRP for an ASR interface.	router igrp <i>asr-autonomous-system-number</i>
Include an ASR interface in IGRP domain.	network <i>asr-addr</i>
Include a loopback interface in IGRP domain.	network <i>loopback-addr</i>
Exit.	Ctrl-Z

Step 4 The ASR interface and all other interfaces should never route packets between each other. (The proxy traffic will be routed specially because it is always addressed to the loopback interface first, then translated by the proxy subsystem.) To do this, set up the following access control list:

Task	Command
Create an access list.	access-list <i>access-list-number</i> permit ip host <i>loopback-address</i> any access-list <i>access-list-number</i> permit ip any host <i>loopback-address</i> access-list <i>access-list-number</i> permit igrp any any ¹
Enter interface configuration mode on an ASR interface.	interface <i>type number</i>
Set the outbound access group.	ip access-group <i>access-list-number</i> out
Set the inbound access group.	ip access-group <i>access-list-number</i> in
Leave ASR interface configuration mode.	Ctrl-Z

1. You may replace **igrp** with the appropriate routing protocol in use on the ASR interface.

Configuration Examples

This section includes the following sample configurations:

- Configuring a Gatekeeper Example
- Enabling E.164 Inter-Zone Routing Example
- Configuring a Gatekeeper for HSRP Example
- Using ASR for a Separate Multimedia Backbone Example
- Configuring an Open Proxy, QoS Enforced, Using RSVP Example

Configuring a Gatekeeper Example

Following is an annotated example of how to configure a gatekeeper:

```
hostname gk-eng.xyz.com
! This router serves as the gatekeeper for the engineering community
! at xyz.com.
ip domain-name xyz.com
! domain name of this company
interface Ethernet0
 ip address 172.21.127.27 255.255.255.0
! This gatekeeper can be found at address 172.21.127.27
gatekeeper
! Enter gatekeeper config mode
 zone local gk-eng.xyz.com xyz.com
! Because a zone is, by definition, the area of control of a gatekeeper,
! we tend to use the terms "zone name" and "gatekeeper name" synonymously.
! Here we use the host name as the name of the gatekeeper and zone.
! This is not necessary, but does simplify administration.
 zone remote gk-mfg.xyz.com xyz.com 171.120.1.4 1719
 zone remote gk-corp.xyz.com xyz.com 170.124.3.80 1719
! A couple of other zones within xyz.com. We make lots of calls
! between these departments, so we just configure these so we save
! a little time bypassing DNS lookup to find their gatekeepers.
 zone access gk-eng.xyz.com remote-zone gk-mfg.xyz.com direct
```

```

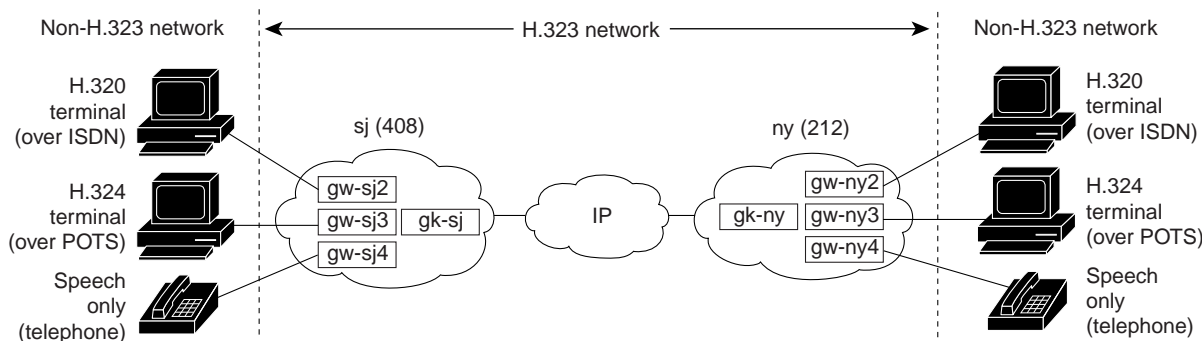
zone access gk-eng.xyz.com remote-zone gk-corp.xyz.com direct
zone access gk-eng.xyz.com default proxied
! We have good QoS on our local network, so we don't need proxies when
! calling between the xyz.com zones. But for all other zones, we want
! to use proxies.
zone subnet gk-eng.xyz.com 172.21.127.0/24 enable
no zone subnet gk-eng.xyz.com default enable
! We will accept registrations from our local subnet as long as they
! do not specify some other gatekeeper name. We will not accept any
! registrations from any other subnet.
zone bw gk-eng.xyz.com 2000
! Preserve our good QoS by not allowing excessive amounts of H.323 traffic
! on the local network. This restricts the traffic within our zone,
! for both intra-zone and inter-zone calls, to 2 kbps at any given time.
alias static 172.21.127.49 gkid gk-eng.xyz.com terminal h323id joeblow ras
172.21.127.49 1719
! The "user" has an H.323 terminal, which does not support RAS. So we have
! to configure his alias manually so that callers can find him.
    
```

Enabling E.164 Inter-Zone Routing Example

Starting with Cisco IOS Releases 11.3(6)Q and 11.3(7)NA and later, you can configure inter-zone routing using E.164 addresses.

In this example, there are two gatekeepers, one in San Jose and one in New York that need to be able to resolve E.164 addresses; see Figure 14.

Figure 14 E.164 Inter-Zone 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
zone access 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
zone access 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 since 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. gk-sj forwards the entire address 2#2125551212 over to gk-ny, who 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, so routes this call to gk-ny. 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 bears 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 bears 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.

Configuring a Gatekeeper for HSRP Example

Cisco routers support HSRP, which allows one router to serve as a backup to another router. Starting with Cisco IOS Releases 11.3(6)Q and 11.3(7)NA and later, you can configure Cisco gatekeepers to use HSRP so that when one gatekeeper fails the standby gatekeeper assumes its role.

To configure a gatekeeper to use HSRP, you need to do the following:

- 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 will be the active gatekeeper, the other assumes the standby role. Make a note of the virtual HSRP IP address shared by both these interfaces. (For details on HSRP and HSRP configuration, refer to the Cisco IOS Release 11.2 *Network Protocols Configuration Guide, Part 1*, in the chapter *IP Routing*.)
- Configure the gatekeepers so that the HSRP virtual IP address is the RAS address for all local zones.
- Make sure that the gatekeeper-mode configurations on both routers are identical.
- If you configure your endpoints and gateways so that they use a specific gatekeeper address (rather than multicasting), use the HSRP virtual IP address as the gatekeeper's address. You can also leave the endpoints and gateways to 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 will be 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 will not be 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 which 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 re-register with the new gatekeeper before it can continue H.323 operations.

Sample Commands for Configuring HSRP on the Gatekeeper

This sample configuration uses Ethernet 0 as the HSRP interface on both gatekeepers.

On the primary gatekeeper, enter these commands:

```
gk1# conf t
  ! enter configuration mode
gk1(config)# int e0
  ! enter interface configuration mode for interface e0
gk1(config-int)# standby 1 ip 172.21.127.55
  ! member of standby group 1, sharing virtual address 172.21.127.55
gk1(config-int)# standby 1 preempt
  ! claim active role when it has higher priority
gk1(config-int)# standby 1 timers 5 15
  ! hello timers is 5 seconds, hold timer is 15 seconds
gk1(config-int)# standby 1 priority 110
  ! priority is 110
```

On the backup gatekeeper, enter these commands:

```
gk2# conf t
gk2(config)# int e0
gk2(config-int)# standby 1 ip 172.21.127.55
gk2(config-int)# standby 1 preempt
gk2(config-int)# 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. This means that gk1 has a higher priority.

On both gk1 and gk2, set up identical gatekeeper mode configurations, as follows:

```
gk1# conf t
  ! enter config mode
gk1(config)# gatekeeper
  ! enter gatekeeper config mode
gk1(config-gk)# zone local gk-sj cisco.com 172.21.127.55
  ! define local zone using HSRP virtual address as gatekeeper's RAS address.
...
  ! various other gk-mode configurations
gk1(config-gk)# no shut
  ! bring up the gatekeeper

gk2# conf t
  ! enter config mode
gk2(config)# gatekeeper
  ! enter gatekeeper config mode
gk2(config-gk)# zone local gk-sj cisco.com 172.21.127.55
  ! define local zone using HSRP virtual address as gatekeeper's RAS address.
  ! Note this uses the same gkname and address as on gk1.
...
  ! various other gk-mode configurations
gk2(config-gk)# no shut
  ! bring up the gatekeeper
```

Note The bring up command **no shut** is issued on both gatekeepers, primary as well as secondary. If you issue the **show gatekeeper status** command on the two gatekeepers, you see on gk1:

Gatekeeper State: UP

but on gk2 you see:

Gatekeeper State: HSRP STANDBY

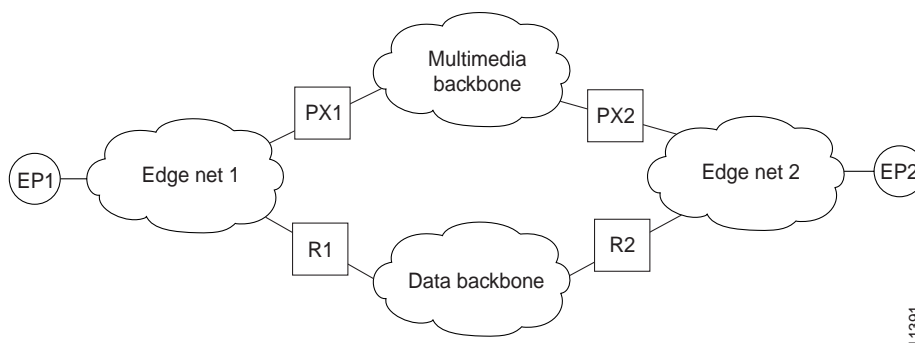
Using ASR for a Separate Multimedia Backbone Example

The examples in this section create 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. You can place a closed H.323 proxy on each edge of the multimedia backbone to achieve the following goals:

- 1 The proxy directs all inter-proxy H.323 traffic, including Q.931 signaling, H.245, and media streams, to the multimedia backbone.
- 2 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.
- 3 The proxy drops any unintended non-H.323 packets that attempt to access the multimedia backbone.

Figure 15 illustrates a network with 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 Cisco router, or may be in separate routers, as shown in the figure.

Figure 15 Sample Network with Multimedia Backbone



11391

Enable 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 connecting the proxy to the multimedia backbone to the ASR interface. Do this by entering the **h323 asr** command in interface configuration mode. This satisfies goal 1 identified earlier in this section.

Since the proxy terminates two call legs of an H.323 call and bridges them together, any H.323 packet traversing the proxy will have the proxy's address either in its source field or in its destination field.

To prevent problems that can occur in proxies with 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 originating from the proxy will have the address of this interface in their source field, and all packets destined to the proxy will have the address of this interface in their destination field.

Figure 15 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 in order 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, you must create a loopback interface on the proxy and configure it 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 originating from the proxy or packet destined to the proxy can pass through the proxy's 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 goal 2 identified earlier in this section.

Isolate the Multimedia Network

Now you need only to configure the network such that non-H.323 traffic never attempts to traverse the multimedia backbone and so 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 is done by configuring routing protocols on the various components of the networks.

The example provided in Figure 15 requires availability of six IP address classes, one for each of the four autonomous systems (ASs) and one for each of the two loopback interfaces. Any Cisco-supported routing protocol can be used on any of the ASs, with one exception: RIP cannot be configured on two adjacent ASs because this protocol does not understand the concept of an AS. The result would be the merging of the two ASs into one.

If IP addresses are scarce, you can use subnetting but configuration gets complicated. In this case only the Enhanced IGRP, OSPF, and RIPv2 routing protocols, which allow variable length subnet masks (VLSM), can be used.

Assuming these requirements are met, you can configure the network depicted in Figure 15 as follows:

- Configure each of the four networks as a separate routing AS and do not redistribute routes between the multimedia backbone and any other AS.
- Create a loopback interface on the proxy and configure it to be the proxy interface. That way you do not expose any subnets of the multimedia backbone to the edge network or the other way around.
- You can choose not to propagate the address of the loopback interface outside the edge network. Do this by configuring the appropriate distribute-list on the edge router connecting the edge network to the data backbone. This guarantees that any ongoing H.323 call will be interrupted if the multimedia backbone fails. Otherwise, H.323 packets originating from one proxy and destined to another proxy might discover an alternate route using the edge networks and the data backbone.

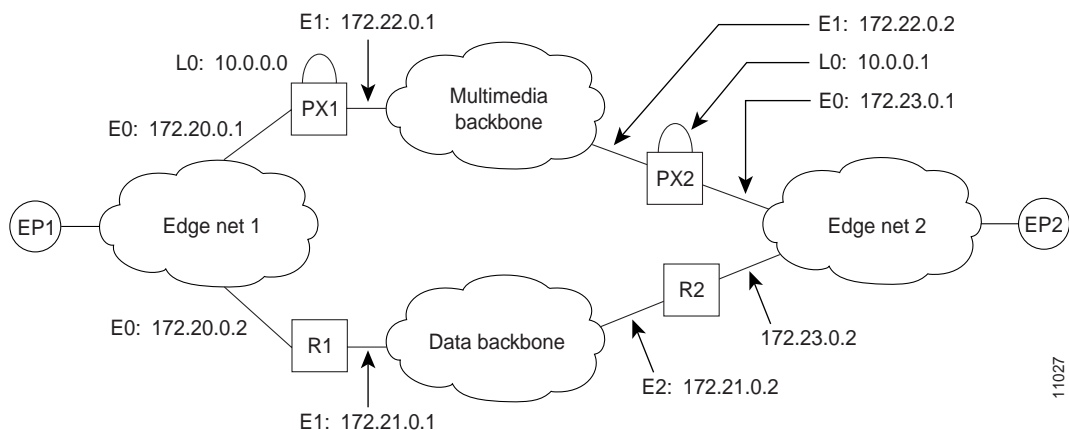
In some topologies you can configure the two edge networks and the data backbone as a single AS, but it is preferable to separate them as previously described because they are different networks with different characteristics.

The following examples illustrate the router configuration that is relevant to the closed proxy operation.

Example: Configuring a Co-Edge Proxy with ASR, without Subnetting

Refer to Figure 16 and the following sample configurations to see how to configure RIP on the two edge networks, and IGRP on the two backbone networks.

Figure 16 Sample Configuration without Subnetting



11027

PX1 Configuration

```
!  
proxy h323  
!  
interface Loopback0  
 ip address 101.0.0.1 255.0.0.0  
!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 101.0.0.1  
!  
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 101.0.0.0  
!  
router igrp 4000  
 network 172.22.0.0  
 network 101.0.0.0  
!  
access-list 101 permit ip any host 101.0.0.1  
access-list 101 permit ip host 101.0.0.1 any  
access-list 101 permit igrp any any
```

R1 Configuration

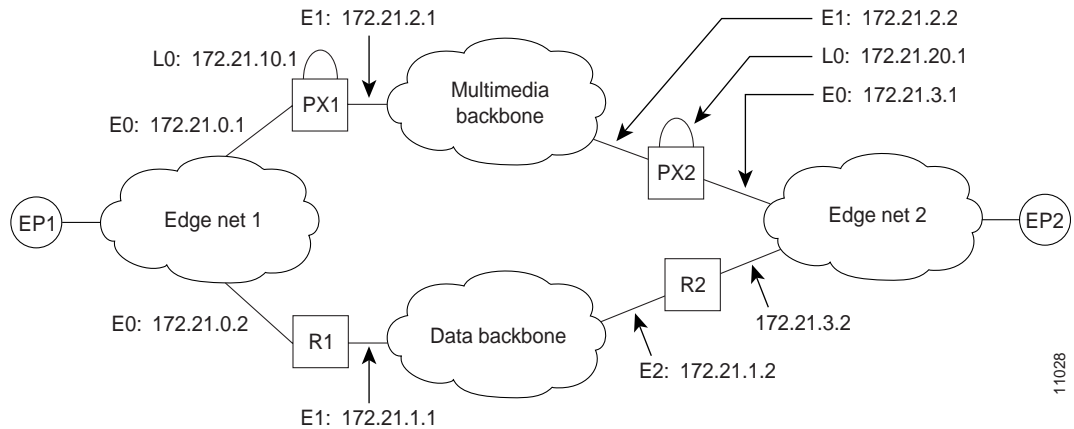
```
!  
interface Ethernet0  
 ip address 172.20.0.2 255.255.0.0  
!  
interface Ethernet1  
 ip address 172.21.0.1 255.255.0.0  
!  
router rip  
 redistribute igrp 5000 metric 1  
 network 172.20.0.0  
!  
router igrp 5000  
 redistribute rip metric 10000 10 255 255 65535  
 network 172.21.0.0  
 distribute-list 10 out  
!  
access-list 10 deny ip 101.0.0.0 0.255.255.255  
access-list 10 permit any
```

Note The configuration for PX2 and R2 is the same as PX1 and R1.

Example: Co-Edge Proxy with Subnetting

Figure 17 illustrates how to configure Enhanced IGRP on all networks.

Figure 17 Sample Configuration with Subnetting



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
```

R1 Configuration

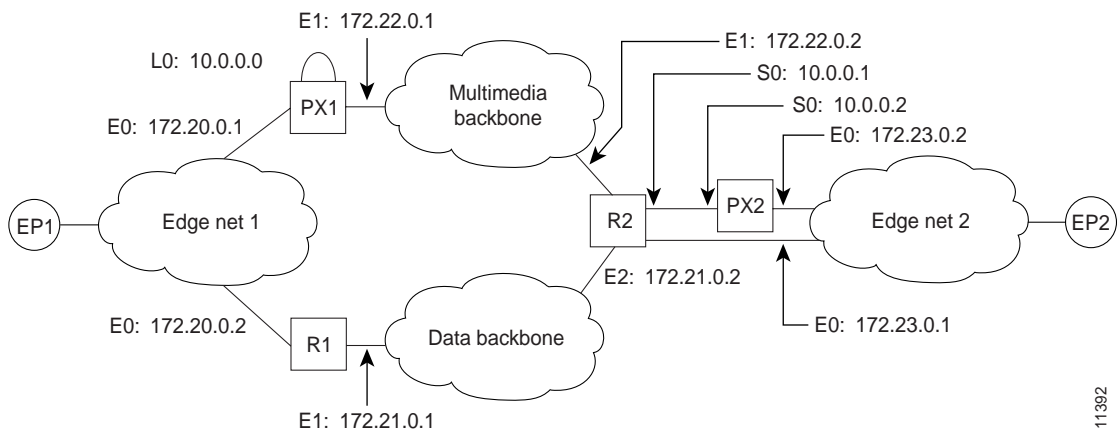
```
!
interface Ethernet0
 ip address 172.21.0.2 255.255.255.192
!
interface Ethernet1
 ip address 172.21.1.1 255.255.255.192
!
router eigrp 4000
 redistribute eigrp 6000 metric 10000 10 255 255 65535
 passive-interface Ethernet1
 network 172.21.0.0
 no auto-summary
!
router eigrp 6000
 redistribute 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 PX1 and R1.

Example: Configuring an Inside-Edge Proxy with ASR, without Subnetting

The configuration of the co-edge proxy in “Edge net 1” has already been presented above. Figure 18 illustrates the configuration of the inside-edge proxy, PX2, and edge router, R2, of “Edge net 2.” RIP routing protocol is used on the edge networks. IGRP is used on the data backbone and the multimedia backbone.

Figure 18 Edge Net 2 with Inside-Edge Proxy, No Subnetting



11392

PX2 Configuration

```
!  
proxy h323  
!  
interface Ethernet0  
 ip address 172.23.0.2 255.255.0.0  
!  
interface Serial0  
 ip address 102.0.0.2 255.0.0.0  
 ip access-group 101 in  
 ip access-group 101 out  
 h323 interface  
 h323 asr  
 h323 h323-id PX2@zone2.com  
 h323 gatekeeper ipaddr 102.0.0.2  
!  
router rip  
 redistribute connected metric 10000 10 255 255 65535  
 network 172.23.0.0  
!  
access-list 101 permit ip any host 102.0.0.2  
access-list 101 permit ip host 102.0.0.2 any
```

R2 Configuration

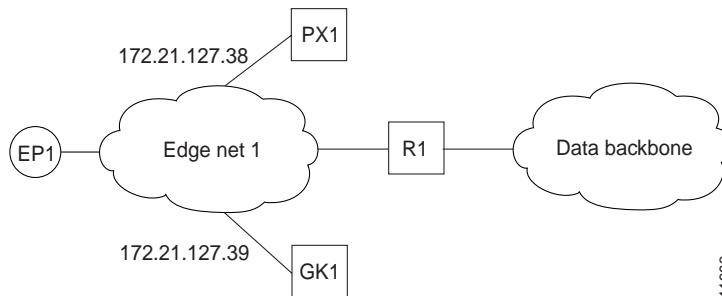
```
!  
interface Ethernet0  
 ip address 172.23.0.1 255.255.0.0  
!  
interface Ethernet1  
 ip address 172.22.0.1 255.255.0.0  
 ip access-group 101 in  
 ip access-group 101 out  
!  
interface Ethernet2  
 ip address 172.21.0.2 255.255.0.0  
!  
interface Serial0  
 ip address 102.0.0.1 255.0.0.0  
!  
router rip  
 redistribute igrp 5000 metric 1  
 network 172.23.0.0  
!  
router igrp 4000  
 network 102.0.0.0  
 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 102.0.0.2 255.255.255.255 Serial0  
access-list 10 deny ip 102.0.0.0 0.255.255.255  
access-list 10 permit any  
access-list 101 permit ip any host 102.0.0.2  
access-list 101 permit ip host 102.0.0.2 any
```

Note To guarantee that all traffic between the proxy and other proxies is carried over the multimedia backbone, run IGRP 4000 on both the 102.0.0.0 network and the 172.22.0.0 network, and make sure that the H.323 proxy interface address (102.0.0.2) is not advertised over the data network (distribute-list 10 in IGRP 5000). Doing this also eliminates the need to configure policy routes or static routes.

Configuring an Open Proxy, QoS Enforced, Using RSVP Example

Figure 19 illustrates 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 19 Configuring a QoS-Enforced Open Proxy Using RSVP



Configuration Commands for PX1

```

!
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

```

```

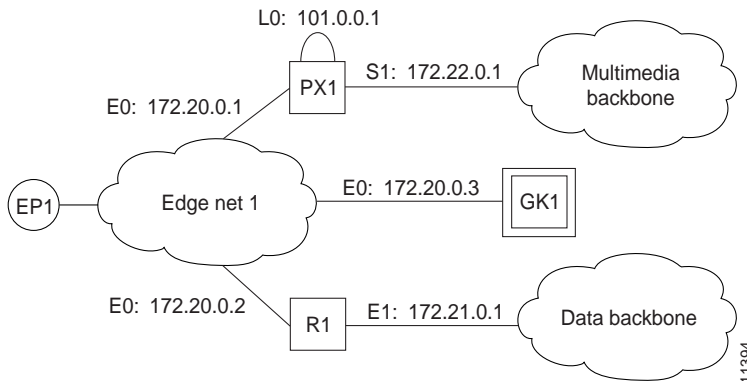
!
router rip
 network 172.21.0.0
!
ip classless
!
!
line con 0
 exec-timeout 0 0
line aux 0
 transport input all
line vty 0 4
 password lab
 login
!
end

```

Example: Configuring a Closed Co-edge Proxy with ASR, Network without Subnetting

Figure 20 illustrates 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 20 Configuring a Closed Co-Edge Proxy with ASR



Configuration Commands for PX1

```

!
version 11.3
no service password-encryption
service tcp-small-servers
!
hostname ExampleProxy
!
!
no ip domain-lookup
!
!
proxy h323
!
interface Loopback0
 ip address 101.0.0.1 255.0.0.0
 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 101.0.0.0
!
router igrp 4000
 network 172.22.0.0
 network 101.0.0.0
!
ip classless
access-list 101 permit ip any host 101.0.0.1
access-list 101 permit ip host 101.0.0.1 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
!
end
```