

Cisco IOS Gatekeepers in Intrazone H.323 Networks Configuration Example

Document ID: 5413

Introduction

Prerequisites

- Requirements
- Components Used
- Conventions

Background Information

- Intrazone Call Setup Overview
- Gatekeeper Call Routing Based on ARQ Messages

Configure

- Network Diagram
- Gatekeeper Configuration
- Gateway Configuration

Configuration Examples

- Configuration Scenario 1: Gatekeeper with Default Technology Prefixes
- Configuration Scenario 2: Gatekeeper with Technology Prefixes

Verify

- Gatekeeper Verification Commands
- Gateway Verification Commands

Troubleshoot

- Troubleshooting Procedure
- Troubleshooting Commands

Related Information

Introduction

This document introduces the basic concepts in order to configure Cisco IOS® gatekeepers. This document provides a sample configuration that starts with the simplest scenario: the configuration of Cisco IOS H.323 gatekeeper and gateways in an intrazone H.323 voice network.

Note: Refer to Understanding H.323 Gatekeepers before you read this document.

A zone is the collection of H.323 nodes or, in this case, gateways that are registered with a gatekeeper. There can only be one active gatekeeper per zone. Gatekeeper zones can overlay subnets. One gatekeeper can manage gateways in one or more subnets. Therefore, this document configures only one gatekeeper, and there is no interzone or gatekeeper-to-gatekeeper communication.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

The information in this document is based on these software and hardware versions:

- Gatekeeper Cisco 3725 that runs Cisco IOS Software Release (c3725-jsx-mz.123-4.T1.bin)
- Gateway-01 Cisco 3725 that runs Cisco IOS Software Release (c3725-jsx-mz.123-4.T1.bin)

Voice module High Density Voice Network Module (NM-HDV) with T1-multiflex trunk module (MFT) voice WAN interface card (VWIC)

- Gateway-02 Cisco 3640 that runs Cisco IOS Software Release (c3640-jsx-mz.123-19.bin)

Voice module Two Voice/Fax Interface Card Slot Network Modules (NM-2V) with Foreign Exchange Station (FXS) voice interface cards (VICs)

Note: The gatekeeper-gateway configuration concepts that this document presents are applicable to all Cisco IOS Software voice-enable platforms.

Note: Gatekeeper functionality is available in these platforms:

- Cisco 72xx
- Cisco 3600/3700/2600
- Cisco 2500

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

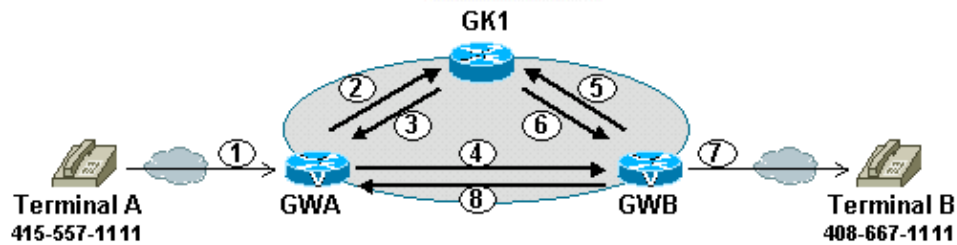
Conventions

Refer to the Cisco Technical Tips Conventions for more information on document conventions.

Background Information

Intrazone Call Setup Overview

This diagram illustrates the gatekeeper-gateway call setup flow, which is H.225 Registration, Admission, and Status (RAS) protocol and H.225 call control signaling.



- 1) Terminal A dials the phone number 408-667-1111 for Terminal B
- 2) GWA sends GK1 an ARQ, asking permission to call Terminal B
- 3) GK1 does a look-up and finds Terminal B registered; returns an ACF with the IP address of GWB
- 4) GWA sends a Q.931 Call-Setup to GWB with Terminal B's phone number
- 5) GWB sends GK1 an ARQ, asking permission to answer GWA's call
- 6) GK1 returns an ACF with the IP address of GWA
- 7) GWB sets up a POTS call to Terminal B at 408-667-1111
- 8) When Terminal B answers, GWB sends Q.931 Connect to GWA
- 9) GWA sends IRR to GK after call is setup

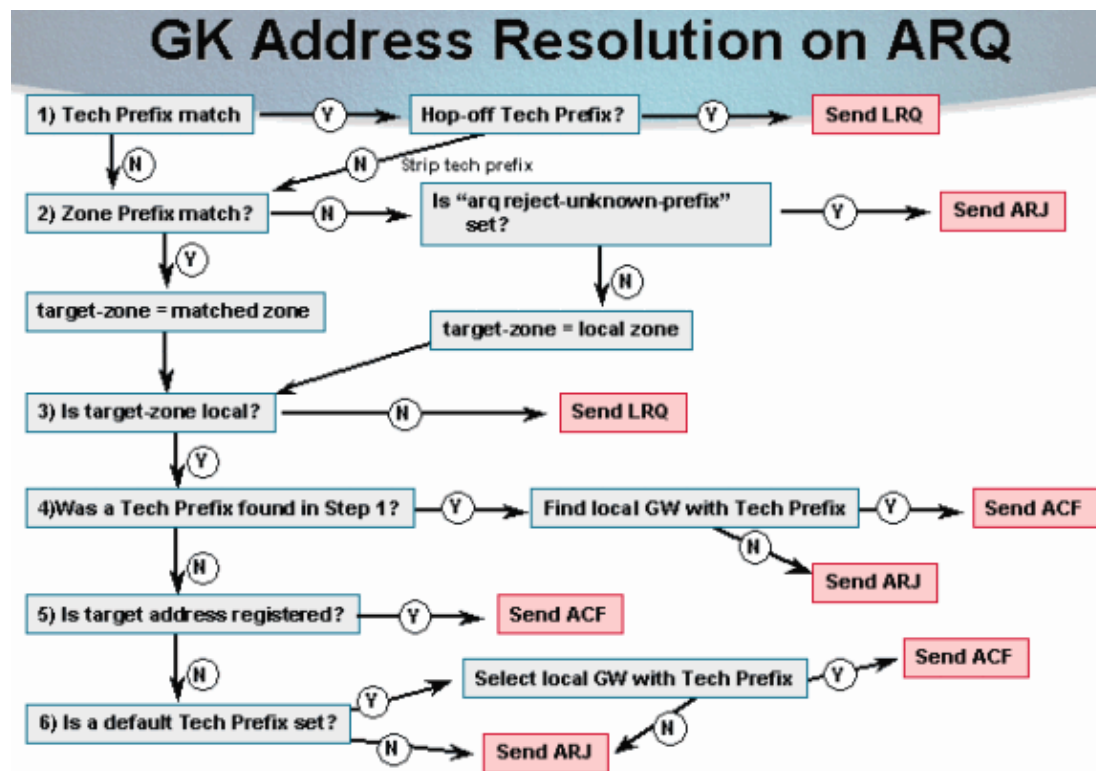
Note: In this diagram:

- ARQ stands for Admission Request
- ACF stands for Admission Confirmation

Refer to Understanding H.323 Gatekeepers for more information on RAS messages.

Gatekeeper Call Routing Based on ARQ Messages

This diagram illustrates the decision algorithm that the gatekeeper goes through when the gatekeeper receives an ARQ message from one of the zone gateways:



Note: In this diagram:

- ARJ stands for Admission Reject
- LRQ stands for Location Request

Note: Only local zone endpoints originate ARQ messages. If a call request arrives at the gatekeeper from another zone, the gatekeeper receives an LRQ message. The gatekeeper address resolution algorithm based on LRQ differs from the algorithm of the ARQ. This document does not present the LRQ algorithm because the document does not cover interzone gatekeeper configurations.

Note: In the diagram, Tech Prefix represents technology prefix. See the Configure section of this document for an explanation of the use of technology prefixes.

Note: This document does not include zone prefixes because the document does not cover interzone gatekeeper configurations.

Gatekeeper Zone Restrictions

- 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.

Gateway Selection Process

- 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 become low. The gatekeeper uses this information to determine which gateway to use to complete a call.

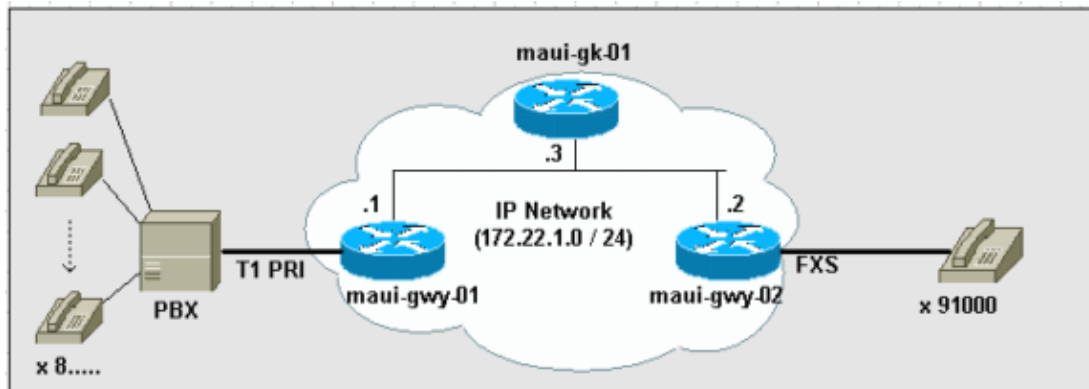
Configure

In this section, you are presented with the information to configure the features described in this document.

Note: Use the Command Lookup Tool (registered customers only) to find additional information on the commands used in this document.

Network Diagram

This document uses this network setup:



Gatekeeper Configuration

Complete these steps:

1. Enable the gatekeeper-gateway discovery and registration process.

Complete these steps:

- a. Enter gatekeeper configuration mode.

```
maui-gk-01#configure terminal
maui-gk-01(config)#gatekeeper
maui-gk-01(config-gk)#
```

- b. Define the gatekeeper local zone of influence.

Note: This command should be on *one* line. It has been moved to a second line in this document due to spatial reasons.

```
maui-gk-01(config-gk)#zone local gatekeeper-name domain-name
[ras-IP-address]
```

The *ras-IP-address* is optional. If you configure this element, the gatekeeper, in response to gatekeeper discovery messages, indicates to endpoints or gateways to use this address for future communications.

Note: This document does not cover H.323 interzone configurations. In order to define interzones, use the **zone remote** command.

- c. Enable gatekeeper functionality.

```
maui-gk-01(config-gk)#no shutdown
```

2. Configure technology prefixes, if you use them.

Note: This command should be on *one* line. It has been moved to a second line in this document due to spatial reasons.

```
maui-gk-01(config-gk)#gw-type-prefix type-prefix [hopoff gk-id]
[default-technology][gw ipaddr ipaddr [port]]
```

Gateway Configuration

Note: This document only deals with a gatekeeper and gateways in the same zone, which is an intrazone setup. Therefore, the document does not cover the zone prefix concept. Refer to the *Remote Zone Call Examples* section of Understanding Cisco IOS Gatekeeper Call Routing for more information about zone

prefixes.

Complete these steps:

1. Enable the gatekeeper–gateway discovery and registration process.

Complete these steps:

- a. Enter gateway configuration mode.

```
maui-gwy-02#configure terminal
maui-gwy-02(config)#gateway
```

- b. Configure the gateway H.323 interface.

```
maui-gwy-02(config)#interface fastethernet 0/0
maui-gwy-02(config-if)#h323-gateway voip interface
maui-gwy-02(config-if)#h323-gateway voip h323-id gateway-id

maui-gwy-02(config-if)#h323-gateway voip id gatekeeper-id
{ipaddr ip-address [port-number] | multicast}
```

Note: The last command should be on *one* line. It has been moved to a second line due to spatial reasons.

- c. Configure the gateway to register to the gatekeeper with a technology prefix, if you use a technology prefix.

```
maui-gwy-02(config-if)#h323-gateway voip tech-prefix prefix
```

The *prefix* defines the numbers that serve as the technology prefixes. Although not strictly necessary, a pound (#) symbol frequently serves as the last digit in a technology prefix.

2. Configure voice ports.
3. Configure plain old telephone service (POTS) dial peers.
4. Configure VoIP dial peers.

Configure the session target as RAS.

Note: If the gateway sends a prefix in the call setup, configure the prefix in the VoIP dial peer that corresponds.

```
maui-gwy-02(config-dial-peer)#session target ras
maui-gwy-02(config-dial-peer)#tech-prefix number
```

WORD A string

Configuration Examples

Configuration Scenario 1: Gatekeeper with Default Technology Prefixes

With the Default Technology Prefixes option, the Cisco gatekeeper assigns default gateways for the route of unresolved call addresses. This assignment is based on the registered technology prefix of the gateways.

```
maui-gk-01 (Cisco 3725- Gatekeeper)
```

```
version 12.3
```

```

!--- Output is suppressed.

!
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname maui-gk-01
!
interface FastEthernet2/0
 ip address 172.22.1.3 255.255.255.0
 duplex half
!
 ip classless
 no ip http server
!
gatekeeper
  zone local GK-01.zone-one.com zone-one.com

!--- Be sure that the gateways have the same gatekeeper name on
!--- their configurations.

gw-type-prefix 1#* default-technology

!--- The gatekeeper treats gateways that are registered with
!--- technology prefix 1# as default when the gatekeeper makes call routing
!--- decisions. There is a default addition of the * character to delimit
!--- the prefix.

no shutdown

!--- Be sure to issue the no shutdown command
!--- in order to enable the gatekeeper functionality.

```

maui-gwy-01 (Cisco 3725)

```

version 12.3

!--- Output is suppressed.

!
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname maui-gwy-01
!
voice-card 3
!
isdn switch-type primary-ni
call rsvp-sync
!
controller T1 3/0
 framing esf
 linecode b8zs
 pri-group timeslots 1-24
!
interface Ethernet0/0
 ip address 172.22.1.1 255.255.255.0
 half-duplex
 h323-gateway voip interface
 h323-gateway voip id GK-01.zone-one.com ipaddr 172.22.1.3 1718

!--- This defines the gatekeeper (GK) ID and the gatekeeper IP address.
!--- In this case, the gateway uses "GK Unicast Discovery".

```

```

!--- Port 1718 is a default assignment.

h323-gateway voip h323-id gwy-01@zone-one.com

!--- This defines the ID of this gateway.

h323-gateway voip tech-prefix 1#

!--- The gateway registers to the gatekeeper with
!--- the technology prefix 1#. In this scenario, the gatekeeper
!--- assigns 1# gateways as default for call routing decisions.

!
interface Serial3/0:23
  no ip address
  no logging event link-status
  isdn switch-type primary-ni
  isdn incoming-voice voice
  no cdp enable
!
voice-port 3/0:23

!--- This is the voice port of the T1 PRI.
!--- Note: The port points to the PRI D-channel (23).

!
dial-peer cor custom
!
dial-peer voice 1 pots
  destination-pattern 8....
  port 3/0:23 prefix 8

!--- This prefix does not relate to gatekeeper-gateway technology prefixes.
!--- This example uses this prefix because, on POTS ports, the explicit defined numbers
!--- in the destination pattern are dropped. Also, the PBX needs the complete
!--- five-digit dial string.

!
dial-peer voice 2 voip
  destination-pattern 91000
  session target ras

!--- Here, you use RAS signaling to point to the gatekeeper.

!
gateway

```

maui-gwy-02 (Cisco 3640)

```

version 12.3

!--- Output is suppressed.

!
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname maui-gwy-02
!
voice-port 1/0/0
!
voice-port 1/0/1
!
dial-peer voice 1 voip

```

```

destination-pattern 8....
session target ras
!
dial-peer voice 2 pots
destination-pattern 91000
port 1/0/0
!
gateway
!
interface FastEthernet0/0
ip address 172.22.1.2 255.255.255.0
duplex auto
speed 10
h323-gateway voip interface
h323-gateway voip id GK-01.zone-one.com multicast

!--- This defines the gatekeeper ID. In this case, the gateway uses
!--- "GK Multicast (autodiscovery)". User Datagram Protocol (UDP) multicast
!--- address 224.0.1.41 is used.

h323-gateway voip h323-id gwy-02@zone-one.com

```

Configuration Scenario 2: Gatekeeper with Technology Prefixes

Cisco gatekeepers use technology prefixes to route calls when there are no E.164 addresses registered by a gateway that match the called number.

maui-gk-01 (Cisco 3725- Gatekeeper)

```

version 12.3

!--- Output is suppressed.

!
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname maui-gk-01
!
interface FastEthernet2/0
ip address 172.22.1.3 255.255.255.0
duplex half
!
ip classless
no ip http server
!
gatekeeper
zone local GK-01.zone-one.com zone-one.com

!--- Be sure that the gateways have the same gatekeeper name on
!--- their configurations.

gw-type-prefix 8#*

!--- The gatekeeper defines the technology prefix 8#.
!--- When the gatekeeper receives an E.164 address (dial string) in
!--- the format "8#...", the gatekeeper routes the call to a gateway that
!--- is registered with 8#.

no shutdown

```

maui-gwy-01 (Cisco 3725)

```

version 12.3

!--- Output is suppressed.

!
service timestamps debug datetime msec
service timestamps log datetime msec
!
hostname maui-gwy-01
!
voice-card 3
!
isdn switch-type primary-ni
call rsvp-sync
!
controller T1 3/0
    framing esf
    linecode b8zs
    pri-group timeslots 1-24
!
interface Ethernet0/0
    ip address 172.22.1.1 255.255.255.0
    half-duplex
    h323-gateway voip interface
    h323-gateway voip id GK-01.zone-one.com ipaddr 172.22.1.3 1718
    h323-gateway voip h323-id gwy-01@zone-one.com
    h323-gateway voip tech-prefix 8#

!--- The gateway registers to the gatekeeper with
!--- the technology prefix 8#.

!
interface Serial3/0:23
    no ip address
    no logging event link-status
    isdn switch-type primary-ni
    isdn incoming-voice voice
    no cdp enable
!
voice-port 3/0:23

!--- This is the voice port of the T1 PRI.
!--- Note: The port points to the PRI D-channel (23).

!
dial-peer cor custom
!
dial-peer voice 1 pots
    destination-pattern 8#.....
    port 3/0:23

!--- Note: The destination pattern starts with 8#.
!--- Incoming calls that the gatekeeper routes based on the 8#
!--- technology prefix come with this number in the dial string.
!--- By the nature of POTS dial peers, the explicitly defined patterns are dropped
!--- before the forward of the call. Therefore, the 8# drops at the transmit
!--- of the digits to the PBX.

!
dial-peer voice 2 voip
    destination-pattern 91000
    session target ras

!--- Here, you use RAS signaling to point to the gatekeeper.

```

```
!  
gateway
```

maui-gwy-02 (Cisco 3640)

```
version 12.3  
  
!--- Output is suppressed.  
  
!  
service timestamps debug datetime msec  
service timestamps log datetime msec  
!  
hostname maui-gwy-02  
!  
voice-port 1/0/0  
!  
voice-port 1/0/1  
!  
dial-peer voice 1 voip  
  destination-pattern 8....  
  tech-prefix 8#  
  
!--- This dial peer appends the 8# pattern to the dial string  
!--- in the gatekeeper ARQ. In this way, the gatekeeper can route the call based on  
!--- the technology prefix 8#. This dial peer also includes the technology  
!--- prefix in the call setup to the terminating gateway which, in this case, is 8#8....  
  
  session target ras  
!  
dial-peer voice 2 pots  
  destination-pattern 91000  
  port 1/0/0  
!  
gateway  
  !  
  interface FastEthernet0/0  
  ip address 172.22.1.2 255.255.255.0  
  duplex auto  
  speed 10  
  h323-gateway voip interface  
  h323-gateway voip id GK-01.zone-one.com multicast  
  h323-gateway voip h323-id gwy-02@zone-one.com
```

Verify

This section provides information you can use to confirm your configuration is working properly.

Certain **show** commands are supported by the Output Interpreter Tool (registered customers only) , which allows you to view an analysis of **show** command output.

Gatekeeper Verification Commands

- **show gatekeeper endpoints** Verifies the registration of the gateways. The information that this command shows includes:
 - ◆ H323-ID
 - ◆ Zone
 - ◆ E164-ID, if applicable

- **show gatekeeper gw-type-prefix** Verifies the gateways that have registered a technology prefix and how the gatekeeper treats the defined technology prefixes.
- **show gatekeeper zone prefix** Indicates the zone to which respective E.164 prefixes are to be routed.
- **show gatekeeper zone status** Verifies zone status and configuration parameters.
- **show gatekeeper status** Displays the overall gatekeeper status, including authorization and authentication status and zone status.
- **show gatekeeper calls** Displays the status of each ongoing call of which a gatekeeper is aware.

Note: Use the Command Lookup Tool (registered customers only) for more information on these commands.

From Configuration Scenario 1

```
!--- Note: Gateway-02 (gwy-02) registers an ID of E164.
!--- This gateway has an FXS port and a number assignment. Gateway-01 (gwy-01) cannot
!--- register E164 numbers because gwy-02 is unaware of the E164 numbers behind
!--- the PBX (T1 PRI).
```

```
maui-gk-01#show gatekeeper endpoints
                        GATEKEEPER ENDPOINT REGISTRATION
                        =====
CallSignalAddr  Port  RASSignalAddr  Port  Zone Name          Type  Flags
-----
172.22.1.1      1720  172.22.1.1     53523  GK-01.zone-one.co VOIP-GW
H323-ID: gwy-01@zone-one.com
172.22.1.2      1720  172.22.1.2     50423  GK-01.zone-one.co VOIP-GW
E164-ID: 91000
H323-ID: gwy-02@zone-one.com
Total number of active registrations = 2
```

```
!-----
!--- Note: The gatekeeper has technology prefix 1#,
!--- which is the default for gateway selection.
!--- Note: Gwy-01 is the only gateway that is registered with
!--- technology prefix 1#.
```

```
maui-gk-01#show gatekeeper gw-type-prefix
GATEWAY TYPE PREFIX TABLE
=====
Prefix: 1#*      (Default gateway-technology)
Zone GK-01.zone-one.com master gateway list:
172.22.1.1:1720 gwy-01
```

```
!-----
maui-gk-01#show gatekeeper status
Gatekeeper State: UP
Load Balancing:   DISABLED
Zone Name:        GK-01.zone-one.com
Accounting:       DISABLED
Security:         DISABLED
Maximum Remote Bandwidth:          unlimited
Current Remote Bandwidth:           0 kbps
Current Remote Bandwidth (w/ Alt GKs): 0 kbps
```

From Configuration Scenario 2

```
maui-gk-01#show gatekeeper gw-type-prefix
GATEWAY TYPE PREFIX TABLE
=====
Prefix: 8#*
Zone GK-01.zone-one.com master gateway list:
```

Gateway Verification Commands

- **show gateway** Displays the current gateway status.
- **show dial-peer voice number** Verifies that the VoIP session protocol is RAS and used to see the technology prefix configurations.

From Configuration Scenario 1

```
maui-gwy-01#show gateway
Gateway gwy-01@zone-one.com is registered to Gatekeeper GK-01.zone-one.com

Alias list (CLI configured)
H323-ID gwy-01@zone-one.com
Alias list (last RCF)
H323-ID gwy-01@zone-one.com

H323 resource thresholding is Disabled
```

From Configuration Scenario 2

```
maui-gwy-02#show dial-peer voice 1

VoiceOverIpPeer1
peer type = voice, information type = voice,
description = '',
tag = 1, destination-pattern = `8....',
answer-address = '', preference=0,
CLID Restriction = None
CLID Network Number = ''
CLID Second Number sent
CLID Override RDNIS = disabled,
source carrier-id = '', target carrier-id = '',
source trunk-group-label = '', target trunk-group-label = '',
numbering Type = `unknown'
group = 1, Admin state is up, Operation state is up,
incoming called-number = '', connections/maximum = 0/unlimited,
DTMF Relay = disabled,
modem transport = system,
huntstop = disabled,
in bound application associated: 'DEFAULT'
out bound application associated: ''
dnis-map =
permission :both
incoming COR list:maximum capability
outgoing COR list:minimum requirement
Translation profile (Incoming):
Translation profile (Outgoing):
incoming call blocking:
translation-profile = ''
disconnect-cause = `no-service'
advertise 0x40 capacity_update_timer 25 addrFamily 4 oldAddrFamily 4
type = voip, session-target = `ras',
technology prefix: 8#
settle-call = disabled
ip media DSCP = ef, ip signaling DSCP = af31, UDP checksum = disabled,
session-protocol = cisco, session-transport = system, req-qos = best-eort,
acc-qos = best-effort,
RTP dynamic payload type values: NTE = 101
Cisco: NSE=100, fax=96, fax-ack=97, dtmf=121, fax-relay=122
CAS=123, ClearChan=125, PCM switch over u-law=0,A-law=8
RTP comfort noise payload type = 19
```

```
fax rate = voice,    payload size = 20 bytes
fax protocol = system
fax-relay ecm enable
fax NSF = 0xAD0051 (default)
codec = g729r8,    payload size = 20 bytes,
Media Setting = flow-through (global)
Expect factor = 10, Icpif = 20,
Playout Mode is set to adaptive,
Initial 60 ms, Max 250 ms
Playout-delay Minimum mode is set to default, value 40 ms
Fax nominal 300 ms
Max Redirects = 1, signaling-type = cas,
VAD = enabled, Poor QOV Trap = disabled,
Source Interface = NONE
voice class sip url = system,
voice class sip rellxx = system,
voice class perm tag = ``
Time elapsed since last clearing of voice call statistics never
Connect Time = 0, Charged Units = 0,
Successful Calls = 5, Failed Calls = 8, Incomplete Calls = 0
Accepted Calls = 0, Refused Calls = 0,
Last Disconnect Cause is "10 ",
Last Disconnect Text is "normal call clearing (16)",
Last Setup Time = 31861243.
```

Troubleshoot

This section provides information you can use to troubleshoot your configuration.

This section is not a complete troubleshooting guide. Instead, the section provides a methodology and series of useful **debug** commands in order to troubleshoot an issue. The purpose of this section is to expose you to the available **debug** commands and provide an understanding of them.

Troubleshooting Procedure

Complete these steps in order to troubleshoot the gatekeeper–gateway scenarios:

1. Assure that the gateway–gatekeeper discovery process is successful.

Use the **debug ras** and **debug h225 asn1** commands. The Troubleshooting Commands section shows these commands.

2. Assure that the gateway–gatekeeper registration process is successful.
3. Assure that the gatekeeper has complete information in order to route calls.

In the gatekeeper–gateway scenarios, this information includes ARQ, replies to ARQ, and no LRQ.

4. Assure the correct configuration of the gateway voice ports, POTS dial peers, and VoIP dial peers for call termination and initiation.

Troubleshooting Commands

The **debug** commands in this section are useful in order to troubleshoot the Troubleshooting Procedure steps.

Note: Refer to Important Information on Debug Commands before you issue **debug** commands.

Gatekeeper

- **debug ras** Displays the RAS messages that exchange between the gatekeeper and the gateway.
- **debug h225 asn1** Provides information in more detail. The command shows ACF and Location Confirm (LCF), along with responses and H.225 call setup/teardown messages.
- **debug h225 events**
- **debug h245 {asn1 | events}**

Gateway

- **debug ras**
- **debug cch323 ras**
- **debug voip ccapi inout**
- **debug cch323 h225**
- **debug cch323 h245**
- **debug h225 asn1**
- **debug h225 events**
- **debug h245 {asn1 | events}**

From Configuration Scenario 1, Discovery and Registration Process

```
!--- This output shows a successful gatekeeper discovery and  
!--- registration process. Output is captured in gwy-01 and the gatekeeper.  
!--- Refer to Understanding H.323 Gatekeepers  
!--- for more information on the gatekeeper discovery and registration process.
```

```
maui-gwy-01# debug ras  
H.323 RAS Messages debugging is on
```

```
RASLib::GW_RASSendGRQ: GRQ (seq# 30779) sent to 172.22.1.3
```

```
!--- Gwy-01 sends a Gatekeeper Request (GRQ) message to the gatekeeper  
!--- (172.22.1.3).
```

```
GCF (seq# 30779) rcvd from h323chan_dgram_send:Sent UDP msg.  
Bytes sent: 131 to 172.22.1.3:1719
```

```
!--- Gwy-01 receives a Gatekeeper Confirmation (GCF) message from  
!--- the gatekeeper (172.22.1.3).
```

```
RASLib::GW_RASSendRRQ: RRQ (seq# 30780) sent to 172.22.1.3
```

```
!--- Gwy-01 sends a Registration Request (RRQ) message to the gatekeeper  
!--- (172.22.1.3).
```

```
h323chan_dgram_rcvdata:rcvd from [172.22.1.3:1719] on sock[1]  
RCF (seq# 30780) rcvd
```

```
!--- Gwy-01 receives a Registration Confirmation (RCF) message from  
!--- the gatekeeper (172.22.1.3).
```

```
!-----
```

```
maui-gk-01#debug ras  
H.323 RAS Messages debugging is on
```

```
!--- Output is suppressed.
```

```

*Oct 31 08:23:29.245: GRQ (seq# 30779) rcvd
!--- The gatekeeper receives a GRQ from gwy-01.

*Oct 31 08:23:29.245:          RASLib::RASSendGCF: GCF (seq# 30779) sent to 172.22.1.1
!--- The gatekeeper sends a GCF to gwy-01.

*Oct 31 08:23:29.249: RRQ (seq# 30780) rcvd
!--- The gatekeeper receives an RRQ from gwy-01.

*Oct 31 08:23:29.249:          RASLib::RASSendRCF: RCF (seq# 30780) sent to 172.22.1.1

!-----
!--- This is gatekeeper output. You can also use this debug

!--- with the gateway.
!--- Output is suppressed. Only the registration process is captured.

maui-gk-01#debug h225 asn1
H.225 ASN1 Messages debugging is on

*Oct 31 09:56:12.980: RAS INCOMING PDU ::=

!--- This is an incoming RAS: RRQ message from gwy-01.

value RasMessage ::= registrationRequest :
  {
    requestSeqNum 30906

!--- The RCF uses the same sequence number.

    protocolIdentifier { 0 0 8 2250 0 2 }
    discoveryComplete TRUE

!--- This indicates that the discovery process is complete.
!--- GRQ and GCF are complete.

    callSignalAddress
    {
      ipAddress :
      {
        ip 'AC160101'H
        port 1720
      }
    }
    rasAddress
    {
      ipAddress :
      {
        ip 'AC160101'H
        port 53523
      }
    }
    terminalType

!--- This is either the gateway or terminal.

    {
      gateway
      {
        protocol
        {

```

```

        voice :
        {
            supportedPrefixes
            {
                {
                    prefix e164 : "1#"
                }
            }
        }
    }
}
mc FALSE
undefinedNode FALSE
}
terminalAlias
{
    h323-ID : {"gwy-01@zone-one.com"}
}

!--- No E.164 IDs are registered for this gwy-01.

}
gatekeeperIdentifier {"GK-01.zone-one.com"}
endpointVendor
{
    vendor
    {
        t35CountryCode 181
        t35Extension 0
        manufacturerCode 18
    }
}
timeToLive 60
keepAlive FALSE
willSupplyUUUIEs FALSE
}

*Oct 31 09:56:12.984: RAS OUTGOING PDU ::=

!--- The gatekeeper sends to gwy-01 a RAS: RCF message.

value RasMessage ::= registrationConfirm :
{
    requestSeqNum 30906
}

!--- The sequence number is the same as RRQ.

protocolIdentifier { 0 0 8 2250 0 2 }
callSignalAddress
{
}
terminalAlias
{
    h323-ID : {"gwy-01@zone-one.com"}
}
gatekeeperIdentifier {"GK-01.zone-one.com"}
endpointIdentifier {"632098E800000001"}
alternateGatekeeper
{
}
timeToLive 60

```

From Configuration Scenario 1, Admission and Call Routing Process

```
!--- Refer to Understanding H.323 Gatekeepers
!--- for more information on the gatekeeper admission process and
!--- gatekeeper-gateway call flows.

!-----
!--- Action: A call is placed from extension x81690 (gwy-02 FXS port) to
!--- x81550 (gwy-01 --> PBX). Call disconnect is not captured.
!--- Output is suppressed.

maui-gwy-02#debug ras
H.323 RAS Messages debugging is on

RASLib::RASSendARQ: ARQ (seq# 1813) sent to 172.22.1.3

!--- An ARQ message goes to the gatekeeper to initiate the call.
!--- Note: The sequence number matches with the gatekeeper.

RASLib::RASRecvData: ACF (seq# 1813) rcvd from [172.22.1.3:1719] on sock[0x81825C9C]

!--- The gatekeeper replies with an ACF message.

maui-gk-01#debug ras
H.323 RAS Messages debugging is on

*Oct 31 10:58:45.620: ARQ (seq# 1813) rcvdparse_arq_nonstd: ARQ Nonstd decode

!--- The gatekeeper receives an ARQ message from gwy-02.
!--- Note: The sequence number matches with gwy-02.

*Oct 31 10:58:45.620:RASLib::RASSendACF: ACF (seq# 1813) sent to 172.22.1.2

!--- The gatekeeper sends an ACF message to gwy-02.

*Oct 31 10:58:45.648: ARQ (seq# 30998) rcvdparse_arq_nonstd: ARQ Nonstd decode

!--- The gatekeeper receives an ARQ message from gwy-01.
!--- Note: The sequence number matches with gwy-01.

*Oct 31 10:58:45.648:RASLib::RASSendACF: ACF (seq# 30998) sent to 172.22.1.1

!--- The gatekeeper sends an ACF message to gwy-01.

maui-gwy-01#debug ras
H.323 RAS Messages debugging is on

RASLib::GW_RASSendARQ: ARQ (seq# 30998) sent to 172.22.1.3
ACF (seq# 30998) rcvdh323chan_dgram_send:Sent UDP msg. Bytes sent: 107
to 172.22.1.3:1719

!-----
!--- This is gatekeeper output. You can also use this debug
!--- with the gateway.
!--- Action: A call is placed from extension x81690 (gwy-02 FXS port) to
!--- x81550 (gwy-01 --> PBX). Call disconnect is not captured.
!--- Output suppressed.

maui-gk-01#debug h225 asn1
```

H.225 ASN1 Messages debugging is on

*Oct 31 11:36:51.416: RAS INCOMING PDU ::= value RasMessage ::= admissionRequest :

!--- The gatekeeper receives an ARQ from gwy-02.

```
{
  requestSeqNum 1885
  destinationInfo
```

!--- The gatekeeper routes the call with the use of the destination address/E.164 number.

!--- Note: There are no technology prefixes.

```
{
  e164 : "81550"
}
srcInfo
{
  e164 : "91000",
  h323-ID : {"gwy-02@zone-one.com"}
}
}
```

*Oct 31 11:36:51.420: RAS OUTGOING PDU ::= value RasMessage ::= admissionConfirm :

!--- The gatekeeper sends an ACF to gwy-02.

```
{
  requestSeqNum 1885
  bandwidth 640
  callModel direct : NULL
  destCallSignalAddress ipAddress :
  {
    ip 'AC160101'H
```

!--- The gatekeeper responds with the destination gateway (gwy-01) IP address.

!--- Note: Because gwy-01 did not register the "e164:81550" address,

!--- the gatekeeper makes the routing decision based on the gwy-01 default

!--- technology prefix registration.

```
    port 1720
  }
}
```

*Oct 31 11:36:51.532: RAS INCOMING PDU ::= value RasMessage ::= admissionRequest :

!--- The gatekeeper receives an ARQ from gwy-01.

!--- Gwy-01 needs authorization to accept an incoming call.

```
{
  requestSeqNum 31077
  callType pointToPoint : NULL
  callModel direct : NULL
  endpointIdentifier {"62B49A4000000001"}
  destinationInfo
  {
    e164 : "81550"
  }
  srcInfo
  {
    e164 : "91000"
  }
  srcCallSignalAddress ipAddress :
```

```

    {
      ip 'AC160102'H
      port 11026
    }
    bandwidth 640
    callReferenceValue 32

```

```

*Oct 31 11:36:51.536: RAS OUTGOING PDU ::=
value RasMessage ::= admissionConfirm :

```

!--- The gatekeeper sends an ACF to gwy-01.

```

    {
      requestSeqNum 31077
      bandwidth 640
      callModel direct : NULL
      destCallSignalAddress ipAddress :
      {
        ip 'AC160101'H
        port 1720
      }
      irrFrequency 240
      willRespondToIRR FALSE
      uuiesRequested
      {
        setup FALSE
        callProceeding FALSE
        connect FALSE
        alerting FALSE
        information FALSE
        releaseComplete FALSE
        facility FALSE
        progress FALSE
        empty FALSE
      }
    }

```

From Configuration Scenario 2, Admission and Call Routing Process

*!--- Refer to Understanding H.323 Gatekeepers
 !--- for more information on the gatekeeper admission process and
 !--- gatekeeper-gateway call flows.*

*!-----
 !--- Action: A call is placed from extension x81690 (gwy-02 FXS port) to
 !--- x81550 (gwy-01 --> PBX). Call disconnect is not captured.
 !--- Output is suppressed.*

GKKK

```

*Oct 31 13:50:49.911: RAS INCOMING PDU ::=

```

```

value RasMessage ::= admissionRequest :
    {
      requestSeqNum 2105
      callType pointToPoint : NULL
      callModel direct : NULL
      endpointIdentifier {"631E269800000002"}
      destinationInfo
      {
        e164 : "8#81550"
      }
      srcInfo
      {

```

```

    e164 : "91000",
    h323-ID : {"gwy-02@zone-one.com"}
}
bandwidth 640
callReferenceValue 195
nonStandardData
{
    nonStandardIdentifier h221NonStandard :
    {
        t35CountryCode 181
        t35Extension 0
        manufacturerCode 18
    }
    data '000000'H
}
conferenceID '76F6F2EEA9AC01AB0000000005B41E78'H
activeMC FALSE
answerCall FALSE
canMapAlias TRUE
callIdentifier
{
    guid '76F6F2EEA9AC01AC0000000005B41E78'H
}
willSupplyUUIEs FALSE
}

```

*Oct 31 13:50:49.915: RAS OUTGOING PDU ::=

```

value RasMessage ::= admissionConfirm :
{
    requestSeqNum 2105
    bandwidth 640
    callModel direct : NULL
    destCallSignalAddress ipAddress :
    {
        ip 'AC160101'H
        port 1720
    }
    irrFrequency 240
    willRespondToIRR FALSE
    uuiesRequested
    {
        setup FALSE
        callProceeding FALSE
        connect FALSE
        alerting FALSE
        information FALSE
        releaseComplete FALSE
        facility FALSE
        progress FALSE
        empty FALSE
    }
}
}

```

maui-gwy-01#debug voip ccapi inout
 voip ccAPI function enter/exit debugging is on

```

maui-gwy-01#
*Mar 17 05:44:48.555: cc_api_call_setup_ind (vdbPtr=0x621EB2C0, callInfo={called=8#81550,
called_oct3=0x91,calling=91000,calling_oct3=0x91,calling_oct3a=0x0,calling_xlated=false,
subscriber_type_str=Unknown,fdest=1,peer_tag=2, prog_ind=0},callID=0x626A6BC8)
*Mar 17 05:44:48.555: cc_api_call_setup_ind type 0 , prot 1
*Mar 17 05:44:48.555: cc_api_call_setup_ind (vdbPtr=0x621EB2C0, callInfo={called=8#81550,
calling=91000, fdest=1 peer_tag=2}, callID=0x626A6BC8)
*Mar 17 05:44:48.555: cc_process_call_setup_ind (event=0x6230CA38)

```

*Mar 17 05:44:48.555: >>>>CCAPI handed cid 134 with tag 2 to app "DEFAULT"
*Mar 17 05:44:48.555: sess_appl: ev(24=CC_EV_CALL_SETUP_IND), cid(134), disp(0)
*Mar 17 05:44:48.555: sess_appl: ev(SSA_EV_CALL_SETUP_IND), cid(134), disp(0)
*Mar 17 05:44:48.555: ssaCallSetupInd
*Mar 17 05:44:48.559: ccCallSetContext (callID=0x86, context=0x626B4A30)
*Mar 17 05:44:48.559: ssaCallSetupInd cid(134), st(SSA_CS_MAPPING),oldst(0),
ev(24)ev->e.evCallSetupInd.nCallInfo.finalDestFlag = 1
*Mar 17 05:44:48.559: ssaCallSetupInd finalDest cllng(91000), cllcd(8#81550)
*Mar 17 05:44:48.559: ssaCallSetupInd cid(134), st(SSA_CS_CALL_SETTING),oldst(0),
ev(24)dpMatchPeersMoreArg result= 0
*Mar 17 05:44:48.559: ssaSetupPeer cid(134) peer list: tag(1) called number (8#81550)
*Mar 17 05:44:48.559: ssaSetupPeer cid(134), destPat(8#81550), matched(1), prefix(),
peer(622FCB48), peer->encapType (1)
*Mar 17 05:44:48.559: ccCallProceeding (callID=0x86, prog_ind=0x0)
*Mar 17 05:44:48.559: ccCallSetupRequest (Inbound call = 0x86, outbound peer =1, dest=,
params=0x62318A18 mode=0, *callID=0x62318D80, prog_ind = 0)
*Mar 17 05:44:48.559: ccCallSetupRequest numbering_type 0x91
*Mar 17 05:44:48.559: dest pattern 8#....., called 8#81550, digit_strip 1
*Mar 17 05:44:48.559: callingNumber=91000, calledNumber=8#81550, redirectNumber=
display_info= calling_oct3a=0
*Mar 17 05:44:48.559: accountNumber=, finalDestFlag=1,
guid=76f6.f2ee.a9ac.01c3.0000.0000.05b7.2984
*Mar 17 05:44:48.559: peer_tag=1
*Mar 17 05:44:48.559: ccIFCallSetupRequestPrivate: (vdbPtr=0x62627630, dest=, callParams=
{called=8#81550,called_oct3=0x91, calling=91000,calling_oct3=0x91, calling_xlated=false,
subscriber_type_str=Unknown, fdest=1, voice_peer_tag=1},mode=0x0) vdbPtr type = 6
*Mar 17 05:44:48.559: ccIFCallSetupRequestPrivate: (vdbPtr=0x62627630, dest=, callParams=
{called=8#81550, called_oct3 0x91, calling=91000,calling_oct3 0x91, calling_xlated=false,
fdest=1, voice_peer_tag=1}, mode=0x0, xltrc=-5)
*Mar 17 05:44:48.559: ccSaveDialpeerTag (callID=0x86, dialpeer_tag=
*Mar 17 05:44:48.563: ccCallSetContext (callID=0x87, context=0x626A2DB0)
*Mar 17 05:44:48.563: ccCallReportDigits (callID=0x86, enable=0x0)
*Mar 17 05:44:48.563: cc_api_call_report_digits_done (vdbPtr=0x621EB2C0, callID=0x86, disp=0)
*Mar 17 05:44:48.563: sess_appl: ev(52=CC_EV_CALL_REPORT_DIGITS_DONE), cid(134), disp(0)
*Mar 17 05:44:48.563: cid(134)st(SSA_CS_CALL_SETTING)ev(SSA_EV_CALL_REPORT_DIGITS_DONE)
oldst(SSA_CS_MAPPING)cfid(-1)csize(0)in(1)fDest(1)
*Mar 17 05:44:48.563: -cid2(135)st2(SSA_CS_CALL_SETTING)oldst2(SSA_CS_MAPPING)
*Mar 17 05:44:48.563: ssaReportDigitsDone cid(134) peer list: (empty)
*Mar 17 05:44:48.563: ssaReportDigitsDone callid=134 Reporting disabled.
*Mar 17 05:44:48.603: cc_api_call_proceeding(vdbPtr=0x62627630, callID=0x87,
prog_ind=0x0)
*Mar 17 05:44:48.603: sess_appl: ev(21=CC_EV_CALL_PROCEEDING), cid(135), disp(0)
*Mar 17 05:44:48.603: cid(135)st(SSA_CS_CALL_SETTING)ev(SSA_EV_CALL_PROCEEDING)
oldst(SSA_CS_MAPPING)cfid(-1)csize(0)in(0)fDest(0)
*Mar 17 05:44:48.607: -cid2(134)st2(SSA_CS_CALL_SETTING)oldst2(SSA_CS_CALL_SETTING)
*Mar 17 05:44:48.607: ssaCallProc
*Mar 17 05:44:48.607: ccGetDialpeerTag (callID=0x)
*Mar 17 05:44:48.607: ssaIgnore cid(135), st(SSA_CS_CALL_SETTING),oldst(1), ev(21)
*Mar 17 05:44:48.607: cc_api_call_alert(vdbPtr=0x62627630, callID=0x87, prog_ind=0x0,
sig_ind=0x1)
*Mar 17 05:44:48.607: sess_appl: ev(7=CC_EV_CALL_ALERT), cid(135), disp(0)
*Mar 17 05:44:48.611: cid(135)st(SSA_CS_CALL_SETTING)ev(SSA_EV_CALL_ALERT)
oldst(SSA_CS_CALL_SETTING)cfid(-1)csize(0)in(0)fDest(0)
*Mar 17 05:44:48.611: -cid2(134)st2(SSA_CS_CALL_SETTING)oldst2(SSA_CS_CALL_SETTING)
*Mar 17 05:44:48.611: ssaAlert
*Mar 17 05:44:48.611: ccGetDialpeerTag (callID=0x)
*Mar 17 05:44:48.611: ccCallAlert (callID=0x86, prog_ind=0x0, sig_ind=0x1)
*Mar 17 05:44:52.363: cc_api_call_connected(vdbPtr=0x62627630, callID=0x87), prog_ind =
1651166880
*Mar 17 05:44:52.363: sess_appl: ev(8=CC_EV_CALL_CONNECTED), cid(135), disp(0)
*Mar 17 05:44:52.363: cid(135)st(SSA_CS_ALERT_RCVD)ev(SSA_EV_CALL_CONNECTED)
oldst(SSA_CS_CALL_SETTING)cfid(-1)csize(0)in(0)fDest(0)
*Mar 17 05:44:52.363: -cid2(134)st2(SSA_CS_ALERT_RCVD)oldst2(SSA_CS_CALL_SETTING)
*Mar 17 05:44:52.363: ssaConnect
*Mar 17 05:44:52.363: ccGetDialpeerTag (callID=0x)
*Mar 17 05:44:52.363: ccConferenceCreate (confID=0x62318E04, callID1=0x86, callID2=0x87,

```
tag=0x0)
*Mar 17 05:44:52.367: cc_api_bridge_done (confID=0x1D, srcIF=0x621EB2C0, srcCallID=0x86,
dstCallID=0
x87, disposition=0, tag=0x0)
*Mar 17 05:44:52.367: cc_api_bridge_done (confID=0x1D, srcIF=0x62627630, srcCallID=0x87,
dstCallID=0
x86, disposition=0, tag=0x0)
*Mar 17 05:44:52.367: cc_api_caps_ind (dstVdbPtr=0x621EB2C0, dstCallId=0x86, srcCallId=0x87,
caps={codec=0x2887F, fax_rate=0x7F, vad=0x3, modem=0x2
codec_bytes=0, signal_type=3})
*Mar 17 05:44:52.367: cc_api_caps_ind (Playout: mode 0, initial 60,min 40, max 200)
*Mar 17 05:44:52.367: cc_api_caps_ind (dstVdbPtr=0x62627630, dstCallId=0x87, srcCallId=0x86,
caps={codec=0x4, fax_rate=0x2, vad=0x2, modem=0x0
codec_bytes=20, signal_type=2})
*Mar 17 05:44:52.367: cc_api_caps_ind (Playout: mode 0, initial 60,min 40, max 200)
*Mar 17 05:44:52.367: cc_api_caps_ack (dstVdbPtr=0x62627630, dstCallId=0x87, srcCallId=0x86,
caps={codec=0x4, fax_rate=0x2, vad=0x2, modem=0x0
codec_bytes=20, signal_type=2})
*Mar 17 05:44:52.367: cc_api_caps_ack (dstVdbPtr=0x621EB2C0, dstCallId=0x86, srcCallId=0x87,
caps={codec=0x4, fax_rate=0x2, vad=0x2, modem=0x0
codec_bytes=20, signal_type=2})
*Mar 17 05:44:52.367: cc_api_voice_mode_event , callID=0x87
*Mar 17 05:44:52.367: Call Pointer =626A2DB0
*Mar 17 05:44:52.371: sess_appl: ev(29=CC_EV_CONF_CREATE_DONE), cid(134), disp(0)
*Mar 17 05:44:52.371: cid(134)st(SSA_CS_CONFERENCING)ev(SSA_EV_CONF_CREATE_DONE)
oldst(SSA_CS_CALL_SETTING)cfid(29)csize(2)in(1)fDest(1)
*Mar 17 05:44:52.371: -cid2(135)st2(SSA_CS_CONFERENCING)oldst2(SSA_CS_ALERT_RCVD)
*Mar 17 05:44:52.371: ssaConfCreateDone
*Mar 17 05:44:52.371: ccCallConnect (callID=0x86), prog_ind = 2
*Mar 17 05:44:52.371: ssaFlushPeerTagQueue cid(134) peer list: (empty)
*Mar 17 05:44:52.371: sess_appl: ev(50=CC_EV_VOICE_MODE_DONE), cid(135), disp(0)
*Mar 17 05:44:52.371: cid(135)st(SSA_CS_ACTIVE)ev(SSA_EV_VOICE_MODE_DONE)
oldst(SSA_CS_ALERT_RCVD)cfid(29)csize(2)in(0)fDest(0)
*Mar 17 05:44:52.371: -cid2(134)st2(SSA_CS_ACTIVE)oldst2(SSA_CS_CONFERENCING)
*Mar 17 05:44:52.371: ssaIgnore cid(135), st(SSA_CS_ACTIVE),oldst(5), ev(50)
*Mar 17 05:44:52.371: cc_process_notify_bridge_done (event=0x6230E2C0)
```

maui-gwy-01#**debug isdn q931**

ISDN Q931 packets debugging is on

maui-gwy-01#

maui-gwy-01#

maui-gwy-01#

```
*Mar 17 05:49:01.451: ISDN Se3/0:23: TX -> SETUP pd = 8 callref = 0x0036
*Mar 17 05:49:01.451: Bearer Capability i = 0x8090A2
*Mar 17 05:49:01.451: Channel ID i = 0xA98381
*Mar 17 05:49:01.451: Calling Party Number i = 0x91, '91000', Plan:ISDN, Type:
International
*Mar 17 05:49:01.455: Called Party Number i = 0x91, '81550', Plan:ISDN, Type:
International
*Mar 17 05:49:01.495: ISDN Se3/0:23: RX <- CALL_PROC pd = 8 callref = 0x8036
*Mar 17 05:49:01.495: Channel ID i = 0xA98381
*Mar 17 05:49:01.499: ISDN Se3/0:23: RX <- ALERTING pd = 8 callref = 0x8036
*Mar 17 05:49:13.563: ISDN Se3/0:23: RX <- CONNECT pd = 8 callref = 0x8036
*Mar 17 05:49:13.563: Progress Ind i = 0x8182 - Destination address is non-ISDN
*Mar 17 05:49:13.567: ISDN Se3/0:23: TX -> CONNECT_ACK pd = 8 callref = 0x0036
```

maui-gk-01#**debug gatekeeper main 5**

maui-gk-01#

maui-gk-01#

maui-gk-01#

maui-gk-01#

maui-gk-01#

maui-gk-01#

maui-gk-01#

```
maui-gk-01#
*Oct 31 14:02:09.747: gk_rassrv_arq: arqp=0x631FCA90, crv=0xD9, answerCall=0
*Oct 31 14:02:09.747: gk_dns_locate_gk(): No Name servers
*Oct 31 14:02:09.747: rassrv_get_addrinfo(8#81550): Matched tech-prefix 8#
*Oct 31 14:02:09.747: rassrv_get_addrinfo(8#81550): unresolved zone prefix, using source
zone GK-01.zone-one.com
*Oct 31 14:02:09.771: gk_rassrv_arq: arqp=0x62E80920, crv=0x3E, answerCall=1
```

Related Information

- [Troubleshooting Gatekeeper Registration Issues](#)
 - [Understanding Inbound and Outbound Dial Peers Matching on IOS Platforms](#)
 - [Troubleshooting Gatekeeper Registration Issues](#)
 - [Understanding Cisco IOS Gatekeeper Call Routing](#)
 - [Configuring H.323 Gatekeepers and Proxies](#)
 - [Gatekeeper Endpoint Control Enhancements](#)
 - [Voice Technology Support](#)
 - [Voice and IP Communications Support Resources](#)
 - [Recommended Reading: Troubleshooting Cisco IP Telephony](#)
 - [Technical Support & Documentation – Cisco Systems](#)
-

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2008 – 2009 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Jun 19, 2006

Document ID: 5413
