

# Configuración del control de admisión de llamadas básico de gatekeeper

## Contenido

[Introducción](#)

[prerrequisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenciones](#)

[Antecedentes](#)

[Comando bandwidth \(control de acceso\)](#)

[Configurar](#)

[Diagrama de la red](#)

[Configuraciones](#)

[Verificación](#)

[Troubleshooting](#)

[Comandos para resolución de problemas](#)

[Resultado de ejemplo de show y debug](#)

[Información Relacionada](#)

## [Introducción](#)

Este documento aporta una configuración de muestra para el Control de admisión de llamadas básico de gatekeeper.

## [prerrequisitos](#)

### [Requisitos](#)

Hay varias condiciones que se encontrarán antes de que el gateway pueda obtener el address resolution correcto del portero. Hay varios puntos importantes que se verificarán para cada solución de VoIP cuando los links de baja velocidad están implicados.

Antes de utilizar esta configuración, asegúrese de que cumple con estos requisitos:

- Todas las gateways deberían estar registradas en los gatekeepers correspondientes.
- Todos los porteros deben tener plan de marcación correcto así que pueden decidir sobre la ruta para las llamadas.
- El control de admisión se puede configurar para restringir el número de llamada entre ciertas zonas.

Pues las primeras dos puntas se consideran en la sección de la [configuración](#), nos centraremos en el control de admisión en la sección de [información previa](#).

## [Componentes Utilizados](#)

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Tres Cisco 2600 Router.
- EMPRESA PLUS/H323 MCM de 12.2.8.5 de la versión del Cisco IOS ® Software.

La información que se presenta en este documento se originó a partir de dispositivos dentro de un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si la red está funcionando, asegúrese de haber comprendido el impacto que puede tener un comando antes de ejecutarlo.

## [Convenciones](#)

Para obtener más información sobre las convenciones del documento, consulte [Convenciones de Consejos Técnicos de Cisco](#).

## [Antecedentes](#)

Esta configuración de muestra estudia una red VoIP con una topología de la dos-zona, que es manejada por un portero con tres gateways en ambas zonas. El objetivo de este documento es proporcionar un ejemplo simple de una configuración de control de admisión que aplique una directiva al número de llamadas entre las zonas y dentro de ellas. Este documento incluye la información previa técnica en las funciones configuradas, las pautas de diseño, y la verificación básica y las estrategias de Troubleshooting.

**Note:** En esta configuración, el cuatro Routers está situado en el mismo LAN. No obstante, en su topología real todos los dispositivos pueden estar en diferentes partes de la red.

Muy a menudo, hay varias fuentes de tráfico de prioridad alta en las redes reales. Es una tarea compleja distinguir todas estas condiciones porque son numerosas, y fácil pasar por alto. Sin embargo, hay varias situaciones comunes que suceden muy a menudo en la vida real eso valen el considerar. El control de admisión se convierte en un problema cuando el Routers que proporciona el priorización del tráfico es ellos mismos no las fuentes de tal tráfico. La topología típica involucra diversas gateways de voz en dos sitios conectados a través del link provisto por un par de routers. Otra topología implica los Ciscos CallManagers con los Teléfonos IP en dos sitios, junto con los gateways al PSTN o al PBX. En ambas situaciones tenemos varias fuentes de tráfico de voz provenientes de ambos lados del link.

A veces, podría haber un problema con la Calidad de voz, si la cantidad de tráfico de voz excede el configuré el ancho de banda para el priority queue. Esto es porque el Routers y los Teléfonos IP de Cisco CallManager/que originen el tráfico no tienen una administración centralizada para la admisión de llamadas en el diseño dado arriba. En este caso, los paquetes que exceden el ancho de banda serán caídos.

Existen varias maneras para evitar esta situación. La solución más simple es configurar el ancho de banda de voz en la cola de tiempo de latencia bajo (LLQ) para validar el número máximo de

llamadas de todas las fuentes. En ausencia de tráfico de voz, el ancho de banda sin utilizar le será otorgado a los flujos de datos. Esto se puede hacer cuando el ancho de banda total del link es más alto que el ancho de banda requerido para el número máximo de llamadas.

Un más enfoque sensato es aplicar las restricciones en cada fuente de tráfico de voz de los ambos extremos del link. Cuando usted lo hace así pues, el ancho de banda sumario de todos no excederá del 75% recomendado del ancho de banda real del link entre los sitios. Para aplicar esas restricciones, utilice el **comando max-conn** bajo configuración del VoIP dial-peer. Si asumimos que hay un Cisco CallManager solamente en un sitio central, podemos utilizar sus capacidades para restringir el número de llamadas al sitio secundario sin un CallManager. Este enfoque nos permite manejar la situación donde los orígenes del tráfico de voz pueden suscribir en exceso el link. La desventaja de este acercamiento es el uso inflexible de ancho de banda concedido a las fuentes. Este acercamiento no permite que algunos de los gateways pongan una llamada adicional incluso si hay ancho de banda libre disponible en ese momento.

El enfoque más flexible consiste en utilizar una entidad separada para el control de admisión de llamadas centralizado: el portero. Las ayudas del portero para atar dos sitios con dos Ciscos CallManagers (o los clústeres del CallManager).

**Note:** El uso del portero no significa siempre la compra de un nuevo router separado. De acuerdo con el número de llamadas, y la carga del Routers, usted puede configurar a un portero en uno de los routers existentes con el conjunto de características de Cisco IOS apropiado como Enterprise/PLUS/H323. Esto puede ayudar a manejar las pequeñas bifurcaciones, y permite a un portero dedicado en el sitio central solamente.

El acercamiento del portero se debe considerar con el cuidado, para no abrumar al router con una carga adicional. Además, usted debe marcar si la topología permitirá el situar del portero de esta manera para evitar el tráfico adicional en el link crítico.

La recomendación general es que utilice routers de Cisco separados como gatekeepers dedicados en su red en la cantidad apropiada para su topología.

Considere la topología arriba. Aquí, usted puede poner todos los dispositivos en las dos zonas locales manejadas por un solo gatekeeper. Esto permite que usted tenga un número alto de llamadas en cada zona, pero restringe el número de llamadas entre ellas. En nuestro ejemplo de la prueba, restringiremos el ancho de banda entre las dos zonas a una llamada, y permitimos hasta dos (un número más elevado de) llamadas en una de ellas.

Para información más detallada sobre esto, vea el [Control de admisión de Llamadas VoIP](#).

Para completar la tarea, utilice el **comando bandwidth (gatekeeper)** descrito en el [portero de alto rendimiento de Cisco](#)

## **Comando bandwidth (control de acceso)**

Para especificar el ancho de banda total máximo para tráfico H.323, utilice el comando bandwidth gatekeeper configuration. Para inhabilitar la característica, no utilice la **ninguna** forma de este comando.

**Note:** Este comando permite que usted restrinja el ancho de banda a través de un solo link de la zona. Si la topología permite que usted ponga una llamada a través de varias trayectorias a partir de una zona a otra, los links podrían llegar a ser fácilmente oversubscribed. Tenga en cuenta esta

topología: dos zonas están conectadas a través de dos trayectorias, permitiendo solamente una llamada a través de cada trayectoria. Si el ancho de banda es restringido por una llamada, la segunda trayectoria nunca será utilizada. Pero si el ancho de banda es restringido por dos llamadas, uno de los links puede ser oversubscribed. Entonces, este comando puede aplicarse a las zonas que sólo poseen un trayecto a todas las otras zonas. La topología del “hub and spoke” es una excepción. Aunque el concentrador tenga trayectos múltiples, no oversubscribe los links, pues el número de llamadas será restringido en el spokes para cada link.

**ancho de banda {interzone | total | sesión} {omita | divida el ancho de banda-*tamaño del nombre de zona en zonas*}**

**ningún ancho de banda {interzone | total | sesión} {omita | divida el ancho de banda-*tamaño del nombre de zona en zonas*}**

### Descripción de la Sintaxis

La tabla siguiente describe el sintaxis:

Sintaxis	Descripción
<b>interzone</b>	Especifica la cantidad total de ancho de banda para el tráfico H.323 desde la zona hasta cualquier otra zona.
<b>total</b>	Especifica la cantidad total de ancho de banda para el tráfico de H.323 permitido en la zona.
<b>sesión</b>	Especifica el ancho de banda máximo permitido para una sesión en la zona.
<b>predeterminado</b>	Especifica el valor predeterminado para todas las zonas.
<b>nombre de zona de la zona</b>	Especifica una zona particular. Asigna un nombre a la zona en particular.
<b>ancho de banda-tamaño</b>	Ancho de banda máximo. Para interzona y total el rango es desde 1 hasta 10.000.000 kbps. Para la <b>sesión</b> , el rango es a partir 1 a 5,000 kbps.

### Valores predeterminados

Ninguno

### Modos de comando

Configuración de control de acceso

### Historial de comando

La tabla siguiente describe el comando history:

Versión	Modificación
12.1(3)XI	Este comando fue ingresado.
12.1(5)XM	Hicieron el <b>comando bandwidth</b> reconocible sin usar el <b>comando zone gatekeeper</b> .
12.2(2)T	Este comando se incorporó a la versión 12.2(2)T del software del IOS de Cisco.
12.2(2)XB1	Este comando fue implementado en el Cisco AS5850 Universal Gateway.

## [Pautas de uso](#)

En las versiones de Cisco IOS Software anteriores, las funciones del **comando bandwidth** fueron obtenidas usando el **comando zone gatekeeper**.

## [Ejemplos](#)

En el siguiente ejemplo se configura el ancho de banda máximo para la zona en 5,000 kbps:

```
Router(config)# gatekeeper
Router(config-gk)# bandwidth total default 5000
```

## [Comandos relacionados](#)

[ancho de banda remote](#) — Especifica el ancho de banda total para el tráfico de H.323 entre este portero, y a cualquier otro portero.

## [Configurar](#)

En esta sección encontrará la información para configurar las funciones descritas en este documento.

**Note:** Para obtener información adicional sobre los comandos que se utilizan en este documento, use la Command Lookup Tool (solo para clientes [registrados](#)).

## [Diagrama de la red](#)

En este documento, se utiliza esta configuración de red:

## [Configuraciones](#)

El objetivo es restringir el ancho de banda disponible entre zone1 y zone3 a una llamada, y permite un número más elevado de las llamadas (hasta dos en este ejemplo) en zone1. Entonces, cumpliremos todos los requisitos generales para una tarea de admisión de llamada típica. Los mensajes de protocolo de registro, admisión y estado (RAS) van delante de los mensajes de configuración de llamadas H225. Luego sigue la negociación H4245 que verdaderamente define las capacidades de los lados. El ancho de banda real de la llamada se define tan después de la etapa de la admisión de llamadas y del intercambio de los mensajes RAS. Por eso es que el

gatekeeper trata a cada llamada como una llamada de 64 kb. Por lo tanto, el aumento en las limitaciones de ancho de banda entre las zonas para las llamadas de voz se debe hacer en incrementos de 64kb.

**Note:** GW\_3 se configura en el mismo router como el portero para ilustrar tal posibilidad de las sucursales de menor capacidad.

**Note:** La verificación de las configuraciones del gatekeeper y gateway es la parte importante de resolver problemas los problemas GK-GW. Por lo tanto, para simplificar la comprensión de las configuraciones, han quitado a todos los comandos configuration sin relación.

### GW\_1 ECV-2600-17

```
IOS (tm) C2600 Software (C2600-JSX-M), Version 12.2(7a),  
RELEASE SOFTWARE (fcl)  
!  
hostname ECV-2610-17  
!  
!  
interface Ethernet0/0  
 ip address 10.52.218.49 255.255.255.0  
 h323-gateway voip interface  
 h323-gateway voip id gk-zone1.test.com ipaddr  
10.52.218.47 1718  
 h323-gateway voip h323-id gw_1  
 h323-gateway voip tech-prefix 1#  
 h323-gateway voip bind srcaddr 10.52.218.49  
!  
voice-port 1/1/0  
!  
voice-port 1/1/1  
!  
!  
dial-peer voice 1 voip  
 destination-pattern ....  
 session target ras  
!  
dial-peer voice 2 pots  
 destination-pattern 1711  
 port 1/1/1  
 no register e164  
!  
gateway  
!  
end
```

### GW\_2 ECV-2600-16

```
!  
hostname ECV-2610-16  
!  
!  
interface Ethernet0/0  
 ip address 10.52.218.48 255.255.255.0  
 h323-gateway voip interface  
 h323-gateway voip id gk-zone3.test.com ipaddr  
10.52.218.47 1718  
 h323-gateway voip h323-id gw_3  
 h323-gateway voip tech-prefix 1#
```

```
h323-gateway voip bind srcaddr 10.52.218.48
!
!
voice-port 1/1/0
!
voice-port 1/1/1
!
dial-peer voice 1 voip
 destination-pattern ....
 session target ras
!
dial-peer voice 2 pots
 destination-pattern 1611
 port 1/1/1
 no register e164
!
gateway
!
!
end
```

## GK\_1 ECV-2600-15

```
hostname ECV-2610-15
!
boot system tftp c2600-jsx-mz.122-7a.bin 10.52.218.2
!
interface Ethernet0/0
 ip address 10.52.218.47 255.255.255.0
 half-duplex
 h323-gateway voip interface
 h323-gateway voip id gk-zone1.test.com ipaddr
 10.52.218.47 1718
 h323-gateway voip h323-id gw_1b
 h323-gateway voip tech-prefix 1#
 h323-gateway voip bind srcaddr 10.52.218.47
!
!
voice-port 1/1/0
!
voice-port 1/1/1
!
!
dial-peer voice 6 pots
 destination-pattern 1511
 port 1/1/1
 no register e164
!
!
dial-peer voice 5 voip
 destination-pattern ....
 session target ras
!
gateway
!
!
gatekeeper
 zone local gk-zone1.test.com test.com 10.52.218.47
 zone local gk-zone3.test.com test.com
 zone prefix gk-zone1.test.com 15.. gw-priority 10 gw_1b
 zone prefix gk-zone3.test.com 16.. gw-priority 10 gw_3
 zone prefix gk-zone1.test.com 17.. gw-priority 10 gw_1
```

```
gw-type-prefix 1#* default-technology
bandwidth interzone zone gk-zone1.test.com 64
!--- Applies the restriction between gk-zone1, and all
!--- other zones to 64bk. That allows one call only.
bandwidth total zone gk-zone1.test.com 128
!--- Applies the restriction to the total number of
calls in zone1, !--- and allows two call in the gk-
zone1. no shutdown
!
end

ECV-2610-15#
```

## Verificación

En esta sección encontrará información que puede utilizar para confirmar que su configuración esté funcionando correctamente.

La herramienta [Output Interpreter](#) (sólo para clientes [registrados](#)) permite utilizar algunos comandos “show” y ver un análisis del resultado de estos comandos.

- **gateway de la demostración** — visualiza el estado de registro del gateway.
- **muestre los puntos finales del portero** — enumera todos los gateways registrados al portero.
- **muestre el prefijo de la zona de gatekeeper** — visualiza todos los prefijos de zona configurados en el portero.
- **show gatekeeper call** — llamadas activas de las demostraciones procesadas por el portero.

## Troubleshooting

En esta sección encontrará información que puede utilizar para solucionar problemas de configuración.

### Comandos para resolución de problemas

La herramienta [Output Interpreter](#) (sólo para clientes [registrados](#)) permite utilizar algunos comandos “show” y ver un análisis del resultado de estos comandos.

**Note:** Antes de ejecutar un comando debug, consulte [Información Importante sobre Comandos Debug](#).

- **haga el debug del asn1 del h225** — los mensajes de las visualizaciones H225 (la configuración de la llamada RAS y Q931).
- **debug cch323 h225** — mensajes de configuración de llamada de las visualizaciones H225.

Aquí están algunos links útiles:

- [Troubleshoot y Debug de Llamadas VoIP – Conceptos Básicos](#)
- [Comandos de depuración de VoIP](#)
- [Referencia de Comandos de Fax, Video y Voz de Cisco IOS, versión 12.2](#)



## Resultado de ejemplo de show y debug

!--- First step is to check the gateway registrations. !--- On the first gateway:

```
ECV-2610-17#show gateway
Gateway gw_1 is registered to Gatekeeper gk-zone1.test.com
```

```
Alias list (CLI configured)
  H323-ID gw_1
Alias list (last RCF)
  H323-ID gw_1
```

```
H323 resource thresholding is Disabled
ECV-2610-17#
```

```
-----
!--- And on the second Gateway: ECV-2610-16#show gateway
Gateway gw_3 is registered to Gatekeeper gk-zone3.test.com
```

```
Alias list (CLI configured)
  H323-ID gw_3
Alias list (last RCF)
  H323-ID gw_3
```

```
H323 resource thresholding is Disabled
ECV-2610-16#-----
```

```
!--- The same on the third Gateway: ECV-2610-15#show gateway
Gateway gw_1b is registered to Gatekeeper gk-zone1.test.com
```

```
Alias list (CLI configured)
  H323-ID gw_1b
Alias list (last RCF)
  H323-ID gw_1b
```

```
H323 resource thresholding is Disabled
ECV-2610-15#-----
```

```
!--- And on the corresponding Gatekeeper: ECV-2610-15#show gatekeeper end
```

```
      GATEKEEPER ENDPOINT REGISTRATION
      =====
CallSignalAddr  Port RASSignalAddr Port  Zone Name          Type F
-----
10.52.218.47    1720 10.52.218.47    58841 gk-zone1.test.com  VOIP-GW
H323-ID: gw_1b
10.52.218.48    1720 10.52.218.48    59067 gk-zone3.test.com  VOIP-GW
H323-ID: gw_3
10.52.218.49    1720 10.52.218.49    52887 gk-zone1.test.com  VOIP-GW
H323-ID: gw_1
Total number of active registrations = 3
```

```
ECV-2610-15#
```

```
-----
!--- To check the dial plan on the Gatekeeper:
```

```
ECV-2610-15#show gatekeeper zone pre
ZONE PREFIX TABLE
=====
GK-NAME E164-PREFIX
-----
gk-zone1.test.com 15..
```

gk-zone3.test.com 16..  
gk-zone1.test.com 17..

ECV-2610-15#

*!--- All configured prefixes should be seen in the zone list. -----*  
*----- !--- To check the zone status on the Gatekeeper: !--- The*  
output shows one permitted interzone call.

ECV-2610-15#show gatekeeper zone st

GATEKEEPER ZONES  
=====  
GK name    Domain Name    RAS Address    PORT    FLAGS  
-----    -----    -----    -----    -----

*!--- The output shows the bandwidth restrictions for this zone. gk-zone1.test.com*  
10.52.218.47 1719 LS

BANDWIDTH INFORMATION (kbps) :

**Maximum total bandwidth : 128**

**Current total bandwidth : 64**

**Maximum interzone bandwidth : 64**

**Current interzone bandwidth : 64**

Maximum session bandwidth :

**Total number of concurrent calls : 1**

SUBNET ATTRIBUTES :

All Other Subnets : (Enabled)

PROXY USAGE CONFIGURATION :

Inbound Calls from all other zones :

to terminals in local zone gk-zone1.test.com : use proxy

to gateways in local zone gk-zone1.test.com : do not use proxy

to MCUs in local zone gk-zone1.test.com : do not use proxy

Outbound Calls to all other zones :

from terminals in local zone gk-zone1.test.com : use proxy

from gateways in local zone gk-zone1.test.com : do not use proxy

from MCUs in local zone gk-zone1.test.com : do not use proxy

*!--- There are no bandwidth restrictions for this zone. gk-zone3.test.com 10.52.218.47 1719*  
LS

BANDWIDTH INFORMATION (kbps) :

**Maximum total bandwidth :**

**Current total bandwidth : 64**

**Maximum interzone bandwidth :**

**Current interzone bandwidth : 64**

Maximum session bandwidth :

**Total number of concurrent calls : 1**

SUBNET ATTRIBUTES :

All Other Subnets : (Enabled)

PROXY USAGE CONFIGURATION :

Inbound Calls from all other zones :

to terminals in local zone gk-zone3.test.com : use proxy

to gateways in local zone gk-zone3.test.com : do not use proxy

to MCUs in local zone gk-zone3.test.com : do not use proxy

Outbound Calls to all other zones :

from terminals in local zone gk-zone3.test.com : use proxy

from gateways in local zone gk-zone3.test.com : do not use proxy

from MCUs in local zone gk-zone3.test.com : do not use proxy

ECV-2610-15#

ECV-2610-15#show gatekeeper call

**Total number of active calls = 1.**

GATEKEEPER CALL INFO

=====

LocalCallID	Age(secs)	BW
5-0	1	64(Kbps)

Endpt(s): Alias	E.164Addr	CallSignalAddr	Port	RASSignalAddr	Port
src EP: gw_3	1611	10.52.218.48	1720	10.52.218.48	59067
dst EP: gw_lb	1511	10.52.218.47	1720	10.52.218.47	58841

ECV-2610-15#

!--- The output shows that we reach maximum number of calls for gk-zone1. ECV-2610-15# ECV-2610-15#show gatekeeper zone st

```

GATEKEEPER ZONES
=====
GK name      Domain Name  RAS Address  PORT  FLAGS
-----
gk-zone1.tes test.com    10.52.218.47  1719  LS
BANDWIDTH INFORMATION (kbps) :
  Maximum total bandwidth : 128
  Current total bandwidth : 128
  Maximum interzone bandwidth : 64
  Current interzone bandwidth : 64
Maximum session bandwidth :
  Total number of concurrent calls : 2
SUBNET ATTRIBUTES :
  All Other Subnets : (Enabled)
PROXY USAGE CONFIGURATION :
  Inbound Calls from all other zones :
to terminals in local zone gk-zone1.test.com : use proxy
to gateways in local zone gk-zone1.test.com : do not use proxy
to MCUs in local zone gk-zone1.test.com : do not use proxy
  Outbound Calls to all other zones :
from terminals in local zone gk-zone1.test.com : use proxy
from gateways in local zone gk-zone1.test.com : do not use proxy
from MCUs in local zone gk-zone1.test.com : do not use proxy

```

```

gk-zone3.tes test.com    10.52.218.47  1719  LS
BANDWIDTH INFORMATION (kbps) :
  Maximum total bandwidth :
  Current total bandwidth : 64
  Maximum interzone bandwidth :
  Current interzone bandwidth : 64
Maximum session bandwidth :
  Total number of concurrent calls : 1
SUBNET ATTRIBUTES :
  All Other Subnets : (Enabled)
PROXY USAGE CONFIGURATION :
  Inbound Calls from all other zones :
to terminals in local zone gk-zone3.test.com : use proxy
to gateways in local zone gk-zone3.test.com : do not use proxy
to MCUs in local zone gk-zone3.test.com : do not use proxy
  Outbound Calls to all other zones :
from terminals in local zone gk-zone3.test.com : use proxy
from gateways in local zone gk-zone3.test.com : do not use proxy
from MCUs in local zone gk-zone3.test.com : do not use proxy

```

```

gk-zone2.tes test.com    10.52.218.46  1719  RS

```

ECV-2610-15#

ECV-2610-15#show gatekeeper call  
Total number of active calls = 2.

```

GATEKEEPER CALL INFO
=====
LocalCallID      Age(secs)  BW
20-33504         49         64(kbps)
Endpt(s): Alias  E.164Addr  CallSignalAddr  Port  RASSignalAddr  Port
src EP: gw_3     1611       10.52.218.48   1720  10.52.218.48   49762

```

```

dst EP: gw_1b      1510      10.52.218.47   1720  10.52.218.47   52344
LocalCallID                               Age(secs)      BW
21-22720 36 64(Kbps)
  Endpt(s): Alias      E.164Addr      CallSignalAddr  Port  RASSignalAddr  Port
src EP: gw_1          1711           10.52.218.49   1720  10.52.218.49   54114
dst EP: gw_1b        1511           10.52.218.47   1720  10.52.218.47   52344

```

ECV-2610-15#

-----

*!--- The conversation between the gateway and gatekeeper consists of !--- an exchange of RAS messages. !--- We start call to 1511 from GW\_3.* ECV-2610-16#deb h225 asn1  
H.225 ASN1 Messages debugging is on  
ECV-2610-16#

\*Mar 1 14:22:20.972: RAS OUTGOING PDU ::=

```

value RasMessage ::= admissionRequest :
{
  requestSeqNum 970
  callType pointToPoint : NULL
  callModel direct : NULL
  endpointIdentifier {"8262B76400000019"}
  destinationInfo
  {
    e164 : "1511"
  }
  srcInfo
  {
    h323-ID : {"gw_3"}
  }
  bandwidth 640
  callReferenceValue 23
  nonStandardData
  {
    nonStandardIdentifier h221NonStandard :
  {
    t35CountryCode 181
    t35Extension 0
    manufacturerCode 18
  }
    data '000000'H
  }
    conferenceID '00000000000000000000000000000000'H
    activeMC FALSE
    answerCall FALSE
    canMapAlias TRUE
    callIdentifier
  {
    guid '00000000000000000000000000000000'H
  }
    willSupplyUUIEs FALSE
  }

```

```

*Mar 1 14:22:20.992: RAS OUTGOING ENCODE BUFFER ::= 27 8803C900 F0003800 32003600
32004200 37003600 34003000 30003000 30003000 30003100 39010180 48440140 03006700
77005F00 33400280 001740B5 00001203 00000000 00000000 00000000 00000000 00000004
E0200180 11000000 00000000 00000000 00000000 00000100
*Mar 1 14:22:21.008:
*Mar 1 14:22:21.073: RAS INCOMING ENCODE BUFFER ::= 2B 0003C940 0280000A 34DA2F06
B800EF14 00C00100 020000
*Mar 1 14:22:21.077:

```

\*Mar 1 14:22:21.081: RAS INCOMING PDU ::=

!--- The GW\_3 gets permission to proceed with that call. value RasMessage ::= admissionConfirm :

```
{
  requestSeqNum 970
  bandwidth 640
  callModel direct : NULL
  destCallSignalAddress ipAddress :
  {
    ip '0A34DA2F'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
  }
}
```

!--- The Call setup message from GW\_3 follows. \*Mar 1 14:22:21.105: H225.0 OUTGOING PDU ::=

```
value H323_UserInformation ::=
{
  h323-uu-pdu
  {
    h323-message-body setup :
    {
      protocolIdentifier { 0 0 8 2250 0 2 }
      sourceAddress
      {
        h323-ID : {"gw_3"}
      }
      sourceInfo
      {
        gateway
        {
          protocol
          {
            voice :
            {
              supportedPrefixes
              {
                {
                  prefix e164 : "1#"
                }
              }
            }
          }
        }
      }
      mc FALSE
      undefinedNode FALSE
    }
    activeMC FALSE
    conferenceID '00000000000000000000000000000000'H
    conferenceGoal create : NULL
    callType pointToPoint : NULL
    sourceCallSignalAddress ipAddress :
```

```

{
  ip '0A34DA30'H
  port 11018
}
callIdentifier
{
  guid '00000000000000000000000000000000'H
}
fastStart
{
  '0000000D4001800A040001000A34DA3041C5'H,
  '400000060401004D40018011140001000A34DA30...'H
}
mediaWaitForConnect FALSE
canOverlapSend FALSE
}
h245Tunneling FALSE
}
}

```

```

*Mar 1 14:22:21.141: H225.0 OUTGOING ENCODE BUFFER ::= 20
A0060008 914A0002 01400300
67007700 5F003308 80013C05 04010020 40000000 00000000 00000000 00000000 00000045
1C07000A 34DA302B 0A110000 00000000 00000000 00000000 00000032 02120000 000D4001
800A0400 01000A34 DA3041C5 1D400000 06040100 4D400180 11140001 000A34DA 3041C400
0A34DA30 41C50100 01000680 0100

```

```
*Mar 1 14:22:21.161:
```

```

*Mar 1 14:22:21.417: H225.0 INCOMING ENCODE BUFFER ::= 21
80060008 914A0002 00048811
00000000 00000000 00000000 00000000 00390219 0000000D 40018011 14000100 0A34DA2F
486E000A 34DA2F48 6F1D4000 00060401 004D4001 80111400 01000A34 DA3041C4 000A34DA
2F486F06 800100

```

```
*Mar 1 14:22:21.429:
```

```
*Mar 1 14:22:21.429: H225.0 INCOMING PDU ::=
```

```

!--- The GW_3 gets Call Proceeding from GW_1b. value H323_UserInformation ::= { h323-uu-pdu {
h323-message-body callProceeding :

```

```

{
  protocolIdentifier { 0 0 8 2250 0 2 }
  destinationInfo
{
  mc FALSE
  undefinedNode FALSE
}
  callIdentifier
{
  guid '00000000000000000000000000000000'H
}
  fastStart
{
  '0000000D40018011140001000A34DA2F486E000A...'H,
  '400000060401004D40018011140001000A34DA30...'H
}
}
h245Tunneling FALSE
}
}

```

```

*Mar 1 14:22:21.617: H225.0 INCOMING ENCODE BUFFER ::= 28
001A0006 0008914A 00020000
00000000 00000000 00000000 00000000 06A00100 120140B5 0000120B 60011000 011E041E
028188

```

```
*Mar 1 14:22:21.626:
```

```
*Mar 1 14:22:21.626: H225.0 INCOMING PDU ::=
!--- The GW_3 geta Call Progress from GW_1b. value H323_UserInformation ::= { h323-uu-pdu {
h323-message-body progress :
{
  protocolIdentifier { 0 0 8 2250 0 2 }
  destinationInfo
{
  mc FALSE
  undefinedNode FALSE
  }
  callIdentifier
{
  guid '00000000000000000000000000000000'H
  }
  }
  h245Tunneling FALSE
  nonStandardControl
{
  {
  nonStandardIdentifier h221NonStandard :
{
t35CountryCode 181
t35Extension 0
manufacturerCode 18
  }
  data '60011000011E041E028188'

  }
  }
  }
  }
}
```

```
*Mar 1 14:22:21.642: H225 NONSTD INCOMING ENCODE BUFFER::= 60
01100001 1E041E02 8188
```

```
*Mar 1 14:22:21.646:
```

```
*Mar 1 14:22:21.646: H225 NONSTD INCOMING PDU ::=
```

```
*Mar 1 14:22:22.831: %SYS-3-MGDTIMER: Running timer, init, timer = 81F1AC08.
```

```
-Process= "Virtual Exec", ipl= 0, pid= 61
```

```
-Traceback= 803250A4 80325214 80325318 80EB12C0
```

```
80EB17DC 802A65F0 802B5080 8033D818
```

```
*Mar 1 14:22:22.835: H225 NONSTD OUTGOING PDU ::=
```

```
value ARQnonStandardInfo ::=
```

```
{
  sourceAlias
{
  }
  sourceExtAlias
{
  }
}
```

```
*Mar 1 14:22:22.839: H225 NONSTD OUTGOING ENCODE
BUFFER::= 00 0000
*Mar 1 14:22:22.839:
*Mar 1 14:22:22.839: RAS OUTGOING PDU ::=
!--- The GW_3 starts the second Call to 1711 now we send RAS message to GK. value RasMessage ::=
```

```
admissionRequest :
{
  requestSeqNum 971
  callType pointToPoint : NULL
  callModel direct : NULL
  endpointIdentifier {"8262B76400000019"}
  destinationInfo
{
  e164 : "1711"
}
  srcInfo
{
  h323-ID : {"gw_3"}
}
  bandwidth 640
  callReferenceValue 24
  nonStandardData
{
  nonStandardIdentifier h221NonStandard :
{
  t35CountryCode 181
  t35Extension 0
  manufacturerCode 18
}
  data '000000'H
}
  conferenceID '00000000000000000000000000000000'H
  activeMC FALSE
  answerCall FALSE
  canMapAlias TRUE
  callIdentifier
{
  guid '00000000000000000000000000000000'H
}
  willSupplyUUIEs FALSE
}
```

```
*Mar 1 14:22:22.860: RAS OUTGOING ENCODE BUFFER::= 27 8803CA00 F0003800 32003600
32004200 37003600 34003000 30003000 30003000 30003100 39010180 4A440140 03006700
77005F00 33400280 001840B5 00001203 00000000 00000000 00000000 00000000 00000004
E0200180 11000000 00000000 00000000 00000000 00000100
```

```
*Mar 1 14:22:22.876:
*Mar 1 14:22:22.940: RAS INCOMING ENCODE BUFFER::= 2B 0003CA40 0280000A 34DA3106
B800EF14 00C00100 020000
```

```
*Mar 1 14:22:22.944:
*Mar 1 14:22:22.944: RAS INCOMING PDU ::=
!--- The GW_3 gets permission to proceed as there are no restrictions on zone3. value RasMessage
::= admissionConfirm :
```

```
{
  requestSeqNum 971
  bandwidth 640
  callModel direct : NULL
  destCallSignalAddress ipAddress :
{
  ip '0A34DA31'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
}
}
```

\*Mar 1 14:22:22.972: H225.0 OUTGOING PDU ::=

*!--- The GW\_3 sends setup message to GW\_1.* value H323\_UserInformation ::= { h323-uu-pdu { h323-  
message-body **setup** :

```
{
  protocolIdentifier { 0 0 8 2250 0 2 }
  sourceAddress
  {
    h323-ID : {"gw_3"}
  }
  sourceInfo
  {
    gateway
  {
    protocol
  {
    voice :
  {
    supportedPrefixes
  {
    {
    prefix e164 : "1#"
    }
    }
    }
    }
  }
  mc FALSE
  undefinedNode FALSE
  }
  activeMC FALSE
  conferenceID '00000000000000000000000000000000'H
  conferenceGoal create : NULL
  callType pointToPoint : NULL
  sourceCallSignalAddress ipAddress :
  {
    ip '0A34DA30'H
    port 11019
  }
  callIdentifier
  {
    guid '00000000000000000000000000000000'H
  }
  fastStart
  {
```

```
'0000000D4001800A040001000A34DA30402F'H,  
'400000060401004D40018011140001000A34DA30...'H  
}  
mediaWaitForConnect FALSE  
canOverlapSend FALSE  
}  
h245Tunneling FALSE  
}  
}
```

```
*Mar 1 14:22:23.008: H225.0 OUTGOING ENCODE BUFFER::= 20  
A0060008 914A0002 01400300
```

```
67007700 5F003308 80013C05 04010020 40000000 00000000 00000000 00000000 00000045  
1C07000A 34DA302B 0B110000 00000000 00000000 00000000 00000032 02120000 000D4001  
800A0400 01000A34 DA30402F 1D400000 06040100 4D400180 11140001 000A34DA 30402E00  
0A34DA30 402F0100 01000680 0100
```

```
*Mar 1 14:22:23.028:
```

```
*Mar 1 14:22:23.220: H225.0 INCOMING ENCODE BUFFER::= 25  
80060008 914A0002 01110000  
00000000 00000000 00000000 00000006 800100
```

```
*Mar 1 14:22:23.224:
```

```
*Mar 1 14:22:23.224: H225.0 INCOMING PDU ::=
```

```
!--- The GW_1 replies with Release Complete message after asking GK !--- for permission to  
accept that call. !--- When the permission is denied, we set bandwidth limit. value
```

```
H323_UserInformation ::= { h323-uu-pdu { h323-message-body releaseComplete :
```

```
{  
  protocolIdentifier { 0 0 8 2250 0 2 }  
  callIdentifier  
{  
  guid '00000000000000000000000000000000'H  
  }  
  }  
  h245Tunneling FALSE  
  }  
}
```

```
*Mar 1 14:22:23.236: RAS OUTGOING PDU ::=
```

```
!--- The GW_3 notifies GK that the call does not exist anymore. value RasMessage ::=
```

```
disengageRequest :
```

```
{  
  requestSeqNum 972  
  endpointIdentifier {"8262B76400000019"}  
  conferenceID '00000000000000000000000000000000'H  
  callReferenceValue 24  
  disengageReason normalDrop : NULL  
  callIdentifier  
{  
  guid '00000000000000000000000000000000'H  
  }  
  answeredCall FALSE  
}
```

```
*Mar 1 14:22:23.248: RAS OUTGOING ENCODE BUFFER::= 3E 03CB1E00 38003200 36003200  
42003700 36003400 30003000 30003000 30003000 31003900 00000000 00000000 00000000  
00000000 18216111 00000000 00000000 00000000 00000000 000100
```

```
*Mar 1 14:22:23.256:
```

```
*Mar 1 14:22:23.288: RAS INCOMING ENCODE BUFFER::= 40  
03CB
```

```
*Mar 1 14:22:23.288:
*Mar 1 14:22:23.288: RAS INCOMING PDU ::=
!--- The GK confirms that message. value RasMessage ::= disengageConfirm :
{
  requestSeqNum 972
}
```

```
ECV-2610-16#u all
All possible debugging has been turned off
ECV-2610-16#
```

```
-----
!--- The incoming RAS message to the GK from GW_3. ECV-2610-15#debug h225 asn1
H.225 ASN1 Messages debugging is on
ECV-2610-15#
*Mar 11 21:54:28.313: RAS INCOMING PDU ::=
```

```
value RasMessage ::= admissionRequest :
{
  requestSeqNum 970
  callType pointToPoint : NULL
  callModel direct : NULL
  endpointIdentifier {"8262B76400000019"}
  destinationInfo
  {
    e164 : "1511"
  }
  srcInfo
  {
    h323-ID : {"gw_3"}
  }
  bandwidth 640
  callReferenceValue 23
  nonStandardData
  {
    nonStandardIdentifier h221NonStandard :
    {
      t35CountryCode 181
      t35Extension 0
      manufacturerCode 18
    }
    data '000000'H
  }
  conferenceID '00000000000000000000000000000000'H
  activeMC FALSE
  answerCall FALSE
  canMapAlias TRUE
  callIdentifier
  {
    guid '00000000000000000000000000000000'H
  }
  willSupplyUUIEs FALSE
}
```

```
*Mar 11 21:54:28.334: H225 NONSTD INCOMING ENCODE BUFFER::= 00 0000
*Mar 11 21:54:28.334:
*Mar 11 21:54:28.334: H225 NONSTD INCOMING PDU ::=
```

```
value ARQnonStandardInfo ::=
```

```
{
  sourceAlias
{
}
  sourceExtAlias
{
}
}
```

*!---* The outgoing RAS message fro GK to GW\_3 with permission to start call. \*Mar 11 21:54:28.338: **RAS OUTGOING PDU** ::=

```
value RasMessage ::= admissionConfirm :
{
  requestSeqNum 970
  bandWidth 640
  callModel direct : NULL
  destCallSignalAddress ipAddress :
{
  ip '0A34DA2F'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
}
}
```

\*Mar 11 21:54:28.350: RAS OUTGOING ENCODE BUFFER::= 2B 0003C940 0280000A 34DA2F06 B800EF14 00C00100 020000

\*Mar 11 21:54:28.354:

\*Mar 11 21:54:28.446: H225.0 INCOMING ENCODE BUFFER::= 20

A0060008 914A0002 01400300

67007700 5F003308 80013C05 04010020 40000000 00000000 00000000 00000000 00000045

1C07000A 34DA302B 0A110000 00000000 00000000 00000000 00000032 02120000 000D4001

800A0400 01000A34 DA3041C5 1D400000 06040100 4D400180 11140001 000A34DA 3041C400

0A34DA30 41C50100 01000680 0100

\*Mar 11 21:54:28.466:

\*Mar 11 21:54:28.470: **H225.0 INCOMING PDU** ::=

*!---* The incoming H323(Q931) message from GW\_3 to GW\_1b on the same router as GK. value

H323\_UserInformation ::= { h323-uu-pdu { h323-message-body **setup** :

```
{
  protocolIdentifier { 0 0 8 2250 0 2 }
  sourceAddress
{
  h323-ID : {"gw_3"}
}
  sourceInfo
{
  gateway
{
  protocol
```

```

{
  voice :
  {
    supportedPrefixes
    {
      {
        prefix e164 : "1#"
      }
    }
  }
  mc FALSE
  undefinedNode FALSE
  activeMC FALSE
  conferenceID '00000000000000000000000000000000'H
  conferenceGoal create : NULL
  callType pointToPoint : NULL
  sourceCallSignalAddress ipAddress :
  {
    ip '0A34DA30'H
    port 11018
  }
  callIdentifier
  {
    guid '00000000000000000000000000000000'H
  }
  fastStart
  {
    '0000000D4001800A040001000A34DA3041C5'H,
    '400000060401004D40018011140001000A34DA30...'H
  }
  mediaWaitForConnect FALSE
  canOverlapSend FALSE
  h245Tunneling FALSE
}

```

```

*Mar 11 21:54:28.514: H225 NONSTD OUTGOING PDU ::=
value ARQnonStandardInfo ::=

```

```

{
  sourceAlias
  {
  }
  sourceExtAlias
  {
  }
}

```

```

*Mar 11 21:54:28.518: H225 NONSTD OUTGOING ENCODE BUFFER ::= 00 0000

```

```

*Mar 11 21:54:28.518:

```

```

*Mar 11 21:54:28.518: RAS OUTGOING PDU ::=

```

```

!--- The GW_1b asks GK if it can accept call from GW_3. value RasMessage ::= admissionRequest :
{
  requestSeqNum 1347
  callType pointToPoint : NULL
  callModel direct : NULL
  endpointIdentifier {"82717F5C0000001B"}
  destinationInfo

```

```

{
  e164 : "1511"
}
srcInfo
{
  h323-ID : {"gw_3"}
}
srcCallSignalAddress ipAddress :
{
  ip '0A34DA30'H
  port 11018
}
bandWidth 640
callReferenceValue 29
nonStandardData
{
  nonStandardIdentifier h221NonStandard :
{
  t35CountryCode 181
  t35Extension 0
  manufacturerCode 18
}
}
data '000000'H
}
conferenceID '00000000000000000000000000000000'H
activeMC FALSE
answerCall TRUE
canMapAlias TRUE
callIdentifier
{
  guid '00000000000000000000000000000000'H
}
willSupplyUUIEs FALSE
}

```

```

*Mar 11 21:54:28.542: RAS OUTGOING ENCODE BUFFER ::= 27 98054200 F0003800 32003700
31003700 46003500 43003000 30003000 30003000 30003100 42010180 48440140 03006700
77005F00 33000A34 DA302B0A 40028000 1D40B500 00120300 00000000 00000000 00000000
00000000 000044E0 20018011 00000000 00000000 00000000 00000000 000100

```

```

*Mar 11 21:54:28.558:

```

```

*Mar 11 21:54:28.562: RAS INCOMING ENCODE BUFFER ::= 27
98054200 F0003800 32003700 31003700 46003500 43003000 30003000 30003000 30003100
42010180 48440140 03006700 77005F00 33000A34 DA302B0A 40028000 1D40B500 00120300
00000000 00000000 00000000 00000000 000044E0 20018011 00000000 00000000 00000000
00000000 000100

```

```

*Mar 11 21:54:28.578:

```

```

*Mar 11 21:54:28.582: RAS INCOMING PDU ::=

```

```

!--- That is the same RAS message. The GK gets it, and sees the sequence number. !--- The GK is
on the same router as GW_1b, so all messages can be seen twice. value RasMessage ::=

```

```

admissionRequest :

```

```

{
  requestSeqNum 1347
  callType pointToPoint : NULL
  callModel direct : NULL
endpointIdentifier {"82717F5C0000001B"}
  destinationInfo
{
  e164 : "1511"
}
  srcInfo
{
  h323-ID : {"gw_3"}
}
}

```

```
}
srcCallSignalAddress ipAddress :
{
  ip '0A34DA30'H
  port 11018
}
bandWidth 640
callReferenceValue 29
nonStandardData
{
  nonStandardIdentifier h221NonStandard :
{
  t35CountryCode 181
  t35Extension 0
  manufacturerCode 18
}
}
data '000000'
}
conferenceID '00000000000000000000000000000000'H
activeMC FALSE
answerCall TRUE
canMapAlias TRUE
callIdentifier
{
  guid '00000000000000000000000000000000'H
}
willSupplyUUIEs FALSE
}
```

\*Mar 11 21:54:28.606: H225 NONSTD INCOMING ENCODE BUFFER ::= 00 0000

\*Mar 11 21:54:28.606:

\*Mar 11 21:54:28.606: H225 NONSTD INCOMING PDU ::=

value ARQnonStandardInfo ::=

```
{
  sourceAlias
{
}
  sourceExtAlias
{
}
}
```

\*Mar 11 21:54:28.610: RAS OUTGOING PDU ::=

*!--- The GK grants the permission to GW\_1b. !--- This is a message in the GK debug outgoing*

value RasMessage ::= **admissionConfirm** :

```
{
  requestSeqNum 1347
  bandWidth 640
  callModel direct : NULL
  destCallSignalAddress ipAddress :
{
  ip '0A34DA2F'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
}
}
```

```
*Mar 11 21:54:28.622: RAS OUTGOING ENCODE BUFFER::= 2B 00054240 0280000A 34DA2F06
B800EF14 00C00100 020000
```

```
*Mar 11 21:54:28.626:
```

```
*Mar 11 21:54:28.630: RAS INCOMING ENCODE BUFFER::= 2B 00054240 0280000A 34DA2F06
B800EF14 00C00100 020000
```

```
*Mar 11 21:54:28.634:
```

```
*Mar 11 21:54:28.634: RAS INCOMING PDU ::=
```

```
!--- The GK grants the permission to GW_1b. !--- This is a message in the GW_1b debug incoming.
```

```
value RasMessage ::= admissionConfirm :
```

```
{
  requestSeqNum 1347
  bandWidth 640
  callModel direct : NULL
  destCallSignalAddress ipAddress :
```

```
{
  ip '0A34DA2F'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
}
}
```

```
*Mar 11 21:54:28.654: %SYS-3-MGDTIMER: Timer has parent, timer link, timer =
820AE990.
```

```
-Process= "CC-API_VCM", ipl= 6, pid= 93
```

```
-Traceback= 80325850 8032A720 80E74850 8033D818
```

```
*Mar 11 21:54:28.666: H225.0 OUTGOING PDU ::=
```

```
!--- The GW_1b replies to GW_3 setup message. value H323_UserInformation ::= { h323-uu-pdu {
h323-message-body callProceeding :
```

```
{
  protocolIdentifier { 0 0 8 2250 0 2 }
  destinationInfo
```

```
{
  mc FALSE
  undefinedNode FALSE
}
```

```
callIdentifier
{
  guid '00000000000000000000000000000000'H
```



```
}
fastStart
{
'0000000D40018011140001000A34DA2F486E000A...'H,
'400000060401004D40018011140001000A34DA30...'H
}
}
h245Tunneling FALSE
}
}
```

```
*Mar 11 21:54:28.682: H225.0 OUTGOING ENCODE BUFFER::= 21 80060008 914A0002 00048811
00000000 00000000 00000000 00000000 00390219 0000000D 40018011 14000100 0A34DA2F
486E000A 34DA2F48 6F1D4000 00060401 004D4001 80111400 01000A34 DA3041C4 000A34DA
2F486F06 800100
```

```
*Mar 11 21:54:28.694:
```

```
*Mar 11 21:54:28.710: H225 NONSTD OUTGOING PDU ::=
```

```
value H323_UU_NonStdInfo ::=
```

```
{
version 16
protoParam qsigNonStdInfo :
{
iei 30
rawMesg '1E028188'H
}
}
```

```
*Mar 11 21:54:28.714: H225 NONSTD OUTGOING ENCODE BUFFER::= 60 01100001 1E041E02 8188
```

```
*Mar 11 21:54:28.714:
```

```
*Mar 11 21:54:28.714: H225.0 OUTGOING PDU ::=
```

```
!--- The GW_1b replies to GW_3 setup message and sends second message. value
```

```
H323_UserInformation ::= { h323-uu-pdu { h323-message-body progress :
```

```
{
protocolIdentifier { 0 0 8 2250 0 2 }
destinationInfo
{
mc FALSE
undefinedNode FALSE
}
callIdentifier
{
guid '00000000000000000000000000000000'H
}
}
h245Tunneling FALSE
nonStandardControl
{
{
nonStandardIdentifier h221NonStandard :
{
t35CountryCode 181
t35Extension 0
manufacturerCode 18
}
data '60011000011E041E028188'H
}
}
}
```

\*Mar 11 21:54:28.734: H225.0 OUTGOING ENCODE BUFFER::= 28 001A0006 0008914A 00020000  
00000000 00000000 00000000 00000000 06A00100 120140B5 0000120B 60011000 011E041E  
028188

\*Mar 11 21:54:28.742:

\*Mar 11 21:54:30.161: RAS INCOMING ENCODE BUFFER::= 27 8803CA00 F0003800 32003600  
32004200 37003600 34003000 30003000 30003000 30003100 39010180 4A440140 03006700  
77005F00 33400280 001840B5 00001203 00000000 00000000 00000000 00000000 00000004  
E0200180 11000000 00000000 00000000 00000000 00000100

\*Mar 11 21:54:30.177:

\*Mar 11 21:54:30.181: **RAS INCOMING PDU** ::=

*!--- The GK gets ARQ from GW\_3 for the second call.* value RasMessage ::= **admissionRequest**:

```
{
  requestSeqNum 971
  callType pointToPoint : NULL
  callModel direct : NULL
  endpointIdentifier {"8262B76400000019"}
  destinationInfo
  {
    e164 : "1711"
  }
  srcInfo
  {
    h323-ID : {"gw_3"}
  }
  bandwidth 640
  callReferenceValue 24
  nonStandardData
  {
    nonStandardIdentifier h221NonStandard :
  {
    t35CountryCode 181
    t35Extension 0
    manufacturerCode 18
  }
    data '000000'H
  }
    conferenceID '00000000000000000000000000000000'H
    activeMC FALSE
    answerCall FALSE
    canMapAlias TRUE
    callIdentifier
  {
    guid '00000000000000000000000000000000'H
  }
    willSupplyUUIEs FALSE
  }
}
```

\*Mar 11 21:54:30.197: H225 NONSTD INCOMING ENCODE BUFFER::= 00 0000

\*Mar 11 21:54:30.201:

\*Mar 11 21:54:30.201: H225 NONSTD INCOMING PDU ::=

value ARQnonStandardInfo ::=

```
{
  sourceAlias
  {
  }
  sourceExtAlias
  {
  }
}
```



```
}
```

```
*Mar 11 21:54:30.469: H225 NONSTD INCOMING ENCODE BUFFER ::= 00 0000  
*Mar 11 21:54:30.469:  
*Mar 11 21:54:30.469: H225 NONSTD INCOMING PDU ::=
```

```
value ARQnonStandardInfo ::=  
{  
  sourceAlias  
{  
}  
  sourceExtAlias  
{  
}  
}
```

```
!-- The GK does not allow the call to come through, and replies with ARJ. *Mar 11  
21:54:30.473: RAS OUTGOING PDU ::=
```

```
value RasMessage ::= admissionReject :  
{  
  requestSeqNum 1120  
  rejectReason requestDenied : NULL  
}
```

```
*Mar 11 21:54:30.477: RAS OUTGOING ENCODE BUFFER ::= 2C 045F20  
*Mar 11 21:54:30.477:  
*Mar 11 21:54:30.541: RAS INCOMING ENCODE BUFFER ::= 3E 03CB1E00 38003200 36003200  
42003700 36003400 30003000 30003000 30003000 31003900 00000000 00000000 00000000  
00000000 18216111 00000000 00000000 00000000 00000000 000100  
*Mar 11 21:54:30.553:  
*Mar 11 21:54:30.557: RAS INCOMING PDU ::=
```

```
!-- The GW_3 notifies GK that call does not exist anymore. value RasMessage ::=  
disengageRequest :  
{  
  requestSeqNum 972  
  endpointIdentifier {"8262B76400000019"}  
  conferenceID '00000000000000000000000000000000'H  
  callReferenceValue 24  
  disengageReason normalDrop : NULL  
  callIdentifier  
{  
  guid '00000000000000000000000000000000'H  
  }  
  answeredCall FALSE  
}
```

```
*Mar 11 21:54:30.565: RAS OUTGOING PDU ::=  
!-- The GK confirms the message from GW_3  
value RasMessage ::= disengageConfirm :  
{  
  requestSeqNum 972  
}
```

```
-----  
!-- The call setup from GW_1 perspective. ECV-2610-17#deb h225 asn1
```

```
H.225 ASN1 Messages debugging is on
```

```
ECV-2610-17#
```

```
*Mar 2 22:55:40: H225.0 INCOMING ENCODE BUFFER ::= 20 A0060008 914A0002 01400300
```

67007700 5F003308 80013C05 04010020 40000000 00000000 00000000 00000000 00000045  
1C07000A 34DA302B 0B110000 00000000 00000000 00000000 00000032 02120000 000D4001  
800A0400 01000A34 DA30402F 1D400000 06040100 4D400180 11140001 000A34DA 30402E00  
0A34DA30 402F0100 01000680 0100

\*Mar 2 22:55:40:

\*Mar 2 22:55:40: H225.0 INCOMING PDU ::=

*!--- The GW\_1 gets the H323 (Q931) setup message from GW\_3.* value H323\_UserInformation ::= {

```
h323-uu-pdu { h323-message-body setup :
{
  protocolIdentifier { 0 0 8 2250 0 2 }
  sourceAddress
{
  h323-ID : {"gw_3"}
}
  sourceInfo
{
  gateway
{
  protocol
{
  voice :
{
  supportedPrefixes
{
  {
  prefix e164 : "1#"
  }
  }
  }
  }
  }
  mc FALSE
  undefinedNode FALSE
  }
  activeMC FALSE
  conferenceID '00000000000000000000000000000000'H
  conferenceGoal create : NULL
  callType pointToPoint : NULL
  sourceCallSignalAddress ipAddress :
{
  ip '0A34DA30'H
  port 11019
  }
  callIdentifier
{
  guid '00000000000000000000000000000000'H
  }
  fastStart
{
  '0000000D4001800A040001000A34DA30402F'H,
  '400000060401004D40018011140001000A34DA30...'H
  }
  mediaWaitForConnect FALSE
  canOverlapSend FALSE
  }
  h245Tunneling FALSE
  }
}
```

\*Mar 2 22:55:40: H225 NONSTD OUTGOING PDU ::=

value ARQnonStandardInfo ::=

```
{
  sourceAlias
{
}
  sourceExtAlias
{
}
}
```

\*Mar 2 22:55:40: H225 NONSTD OUTGOING ENCODE BUFFER::= 00 0000

\*Mar 2 22:55:40:

\*Mar 2 22:55:40: RAS OUTGOING PDU ::=

*!--- The GW\_1 asks GK for permission to accept the call.* value RasMessage ::= admissionRequest :

```
{
  requestSeqNum 1120
  callType pointToPoint : NULL
  callModel direct : NULL
  endpointIdentifier {"823860D40000001A"}
  destinationInfo
{
  e164 : "1711"
}
  srcInfo
{
  h323-ID : {"gw_3"}
}
  srcCallSignalAddress ipAddress :
{
  ip '0A34DA30'H
  port 11019
}
  bandwidth 640
  callReferenceValue 40
  nonStandardData
{
  nonStandardIdentifier h221NonStandard :
{
  t35CountryCode 181
  t35Extension 0
  manufacturerCode 18
}
  data '000000'H
}
  conferenceID '00000000000000000000000000000000'H
  activeMC FALSE
  answerCall TRUE
  canMapAlias TRUE
  callIdentifier
{
  guid '00000000000000000000000000000000'H
}
  willSupplyUUIEs FALSE
}
```

\*Mar 2 22:55:40: RAS OUTGOING ENCODE BUFFER::= 27 98045F00 F0003800 32003300  
38003600 30004400 34003000 30003000 30003000 30003100 41010180 4A440140 03006700  
77005F00 33000A34 DA302B0B 40028000 2840B500 00120300 00000000 00000000 00000000  
00000000 000044E0 20018011 00000000 00000000 00000000 00000000 000100

\*Mar 2 22:55:41:

\*Mar 2 22:55:41: RAS INCOMING ENCODE BUFFER::= 2C 045F20

```
*Mar 2 22:55:41:
*Mar 2 22:55:41: RAS INCOMING PDU ::=
!--- The GK denies permission to accept the call from GW_3 due to bandwidth limit. value
RasMessage ::= admissionReject :
{
  requestSeqNum 1120
  rejectReason requestDenied : NULL
}

*Mar 2 22:55:41: H225.0 OUTGOING PDU ::=
!--- The GW_1 rejects call setup from GW_3. value H323_UserInformation ::= { h323-uu-pdu { h323-
message-body releaseComplete :
{
  protocolIdentifier { 0 0 8 2250 0 2 }
  callIdentifier
{
  guid '00000000000000000000000000000000'H
  }
  }
  h245Tunneling FALSE
  }
  }

*Mar 2 22:55:41: H225.0 OUTGOING ENCODE BUFFER::= 25 80060008 914A0002 01110000
00000000 00000000 00000000 00000006 800100
*Mar 2 22:55:41:
ECV-2610-17#
ECV-2610-17#
ECV-2610-17#u all
All possible debugging has been turned off
-----
```

## [Información Relacionada](#)

- [Resolución de problemas e introducción a la administración del ancho de banda del Gatekeeper del Cisco](#)
- [Introducción de Gatekeepers para H.323](#)
- [Gatekeeper de alto rendimiento de Cisco](#)
- [Configuración de gateways H.323](#)
- [Configuración de los controladores de acceso H.323](#)
- [Configuración del soporte H.323 para interfaces virtuales](#)
- [Soporte de tecnología de voz](#)
- [Soporte de Productos de Voice and Unified Communications](#)
- [Troubleshooting de Cisco IP Telephony](#)
- [Soporte Técnico - Cisco Systems](#)