

Configurazione del controllo di ammissione delle chiamate Gatekeeper di base

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Convenzioni](#)

[Premesse](#)

[Comando larghezza di banda \(gatekeeper\)](#)

[Configurazione](#)

[Esempio di rete](#)

[Configurazioni](#)

[Verifica](#)

[Risoluzione dei problemi](#)

[Comandi per la risoluzione dei problemi](#)

[Output di esempio di show e debug](#)

[Informazioni correlate](#)

[Introduzione](#)

Questo documento fornisce una configurazione di esempio per il controllo di base dell'ammissione delle chiamate del gatekeeper.

[Prerequisiti](#)

[Requisiti](#)

Affinché il gateway sia in grado di ottenere la risoluzione degli indirizzi corretta dal gatekeeper, è necessario che siano soddisfatte diverse condizioni. Quando sono coinvolti collegamenti a bassa velocità, è necessario verificare diversi punti importanti per ogni soluzione VoIP.

Prima di provare la configurazione, verificare che siano soddisfatti i seguenti requisiti:

- Tutti i gateway devono essere registrati ai gatekeeper corrispondenti
- Tutti i gatekeeper devono avere il dial-plan corretto in modo da poter decidere il percorso delle chiamate.
- È possibile configurare il controllo dell'ingresso in modo da limitare il numero di chiamate tra determinate zone.

Poiché i primi due punti sono considerati nella sezione [Configura](#), ci concentreremo sul controllo di ammissione nella sezione [Informazioni di base](#).

[Componenti usati](#)

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Tre router Cisco 2600.
- Software Cisco IOS® versione 12.2.8.5 ENTERPRISE PLUS/H323 MCM.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

[Convenzioni](#)

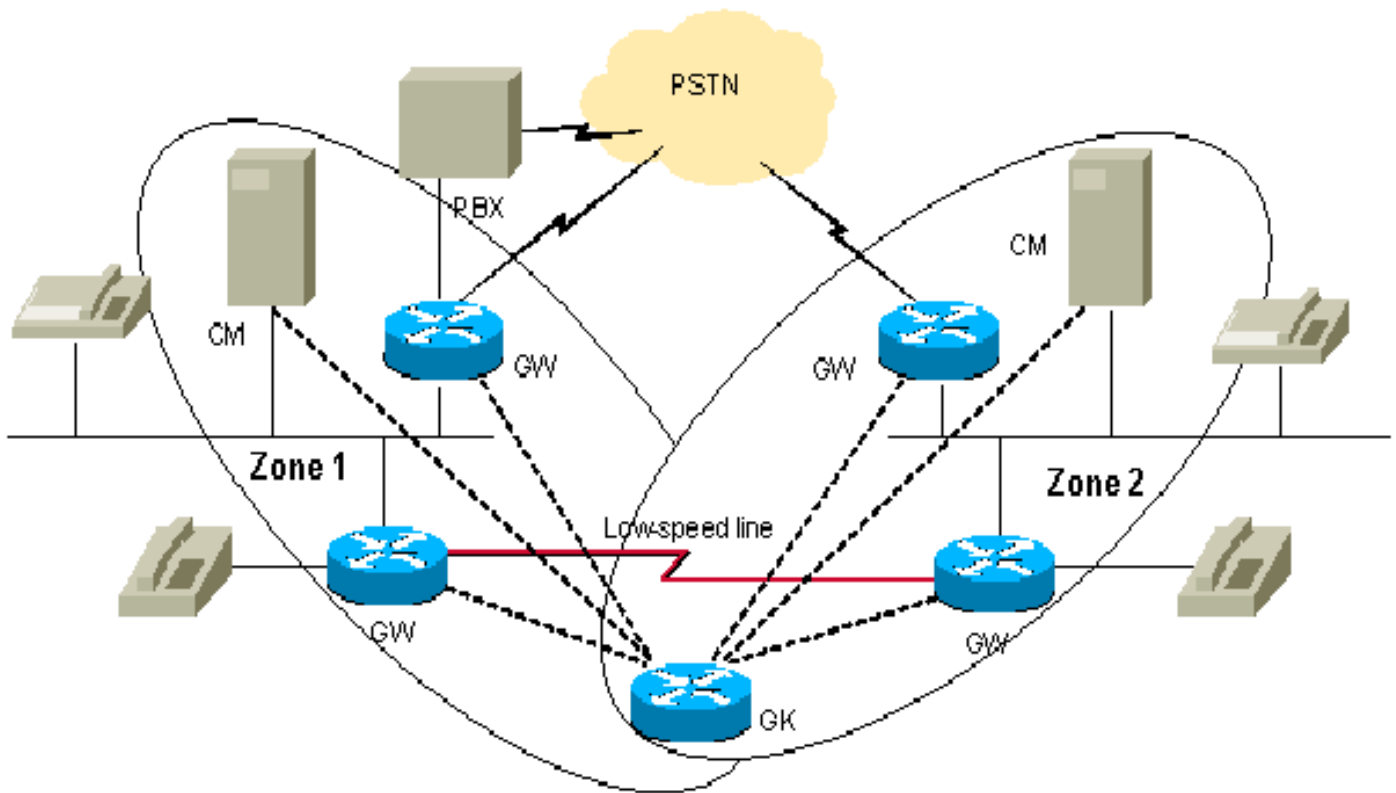
Per ulteriori informazioni sulle convenzioni usate, consultare il documento [Cisco sulle convenzioni nei suggerimenti tecnici](#).

[Premesse](#)

In questa configurazione di esempio viene studiata una rete VoIP con una topologia a due zone, gestita da un gatekeeper con tre gateway in entrambe le zone. Lo scopo di questo documento è quello di fornire un semplice esempio di una configurazione di controllo di ammissione che applica una policy al numero di chiamate tra zone e all'interno di esse. Questo documento include informazioni tecniche di base sulle funzionalità configurate, linee guida di progettazione e strategie di verifica e risoluzione dei problemi di base.

Nota: in questa configurazione, i quattro router si trovano sulla stessa LAN. Tuttavia, nella topologia reale, tutti i dispositivi possono trovarsi in parti diverse della rete.

Molto spesso, nelle reti reali vi sono diverse fonti di traffico ad alta priorità. È un compito complesso distinguere tutte queste condizioni perché sono numerose e facilmente trascurabili. Tuttavia, ci sono diverse situazioni comuni che accadono molto spesso nella vita reale che vale la pena considerare. Il controllo dell'ammissione diventa un problema quando i router che forniscono la definizione di priorità del traffico non sono le origini di tale traffico. La topologia tipica coinvolge diversi gateway voce in due siti connessi tramite il collegamento fornito da una coppia di router. Un'altra topologia interessa i Cisco CallManager con telefoni IP in due siti, insieme ai gateway della PSTN o PBX. In entrambe le situazioni abbiamo diverse fonti di traffico vocale da entrambi i lati del collegamento.



A volte, potrebbe esserci un problema con la qualità della voce, se la quantità di traffico vocale supera la larghezza di banda configurata per la coda di priorità. Infatti, i router e i telefoni IP Cisco CallManager che originano il traffico non dispongono di una gestione centralizzata per l'ammissione delle chiamate nella progettazione descritta sopra. In questo caso, i pacchetti che superano la larghezza di banda verranno scartati.

Esistono diversi modi per evitare questo scenario. La soluzione più semplice consiste nel configurare la larghezza di banda della voce nella coda a bassa latenza (LLQ) in modo che accetti il numero massimo di chiamate da tutte le origini. In assenza di traffico vocale, la larghezza di banda non utilizzata verrà concessa ai flussi di dati. Questa operazione può essere eseguita quando la larghezza di banda totale del collegamento è superiore a quella richiesta per il numero massimo di chiamate.

Un approccio più ragionevole consiste nell'applicare restrizioni a ciascuna origine di traffico vocale da entrambe le estremità del collegamento. Quando lo si fa, la larghezza di banda di riepilogo da tutti loro non supererà il 75% consigliato della larghezza di banda reale del collegamento tra i siti. Per applicare queste restrizioni, usare il comando **max-conn** in Configurazione dial-peer VoIP. Se si presume che Cisco CallManager sia presente solo in un sito centrale, è possibile utilizzare le funzionalità di Cisco CallManager per limitare il numero di chiamate al sito di succursale senza CallManager. Questo approccio ci permette di gestire la situazione in cui le fonti del traffico vocale sono in grado di sovrascrivere il collegamento. Lo svantaggio di questo approccio è l'uso inflessibile della larghezza di banda concessa alle sorgenti. Questo approccio non consente ad alcuni gateway di effettuare una chiamata aggiuntiva anche se al momento è disponibile una larghezza di banda libera.

L'approccio più flessibile consiste nell'utilizzare un'entità separata per il controllo centralizzato dell'ammissione di chiamata: il guardiano. Il gatekeeper aiuta a collegare due siti con due Cisco CallManager (o cluster CallManager).

Nota: L'uso del gatekeeper non sempre indica l'acquisto di un nuovo router separato. In base al numero di chiamate e al carico dei router, è possibile configurare un gatekeeper su uno dei router

esistenti con le funzionalità Cisco IOS appropriate impostate su Enterprise/PLUS/H323. Ciò può aiutare a gestire piccole filiali e consentire l'uso di un gatekeeper dedicato solo nel sito centrale.

L'approccio del gatekeeper deve essere valutato con attenzione, in modo da non sovraccaricare il router con un carico aggiuntivo. Inoltre, è necessario verificare se la topologia consente di posizionare il gatekeeper in questo modo per evitare l'aumento del traffico sul collegamento critico.

Si consiglia di utilizzare router Cisco separati come gatekeeper dedicati nella rete in un numero appropriato per la topologia.

Si consideri la topologia riportata sopra. Qui è possibile posizionare tutti i dispositivi nelle due zone locali gestite da un singolo gatekeeper. In questo modo è possibile disporre di un numero elevato di chiamate in ogni zona, ma il numero di chiamate tra le zone è limitato. Nel nostro esempio di test, limiteremo la larghezza di banda tra le due zone a una chiamata e permetteremo fino a due (più alti) chiamate in una di esse.

Per ulteriori informazioni, vedere [Controllo dell'ammissione di chiamate VoIP](#).

Per completare l'attività, usare il comando **bandwidth** (gatekeeper) descritto in [Cisco High-Performance Gatekeeper](#)

Comando larghezza di banda (gatekeeper)

Per specificare la larghezza di banda aggregata massima per il traffico H.323, usare il comando di configurazione **bandwidth** gatekeeper. Per disabilitare la funzione, utilizzare la forma **no** di questo comando.

Nota: questo comando consente di limitare la larghezza di banda tramite un singolo collegamento dalla zona. Se la topologia consente di effettuare una chiamata attraverso diversi percorsi da una zona all'altra, i collegamenti potrebbero facilmente diventare sovrascritti. Supponiamo di avere questa topologia: due zone sono collegate attraverso due percorsi, consentendo una sola chiamata attraverso ogni percorso. Se la larghezza di banda è limitata da una chiamata, il secondo percorso non verrà mai utilizzato. Ma se la larghezza di banda è limitata da due chiamate, uno dei link potrebbe essere sovrascritto. Questo comando può quindi essere applicato alle zone che hanno un solo percorso verso tutte le altre zone. La topologia "Hub and Spoke" è un'eccezione. Sebbene l'hub abbia più percorsi, non sovrascriverà i collegamenti, poiché il numero di chiamate verrà limitato in corrispondenza degli spoke per ogni collegamento.

larghezza di banda {interzone | totale | session} {predefinito | zone nome-zona} larghezza di banda

nessuna larghezza di banda {interzone | totale | session} {predefinito | zona nome-zona} larghezza di banda

Descrizione della sintassi

Nella tabella seguente viene descritta la sintassi:

Sintassi	Descrizione
interzona	Specifica la quantità totale di larghezza di banda per il traffico H.323 dalla zona a

	qualsiasi altra zona.
totale	Specifica la quantità totale di larghezza di banda per il traffico H.323 consentito nella zona.
sessione	Specifica la larghezza di banda massima consentita per una sessione nella zona.
predefinito	Specifica il valore predefinito per tutte le zone.
zona nome-zona	Specifica una zona specifica. Assegna un nome alla zona specifica.
larghezza di banda	Larghezza di banda massima. Per interzone e totale , l'intervallo è compreso tra 1 e 10.000.000 kbps. Per la sessione , l'intervallo è compreso tra 1 e 5.000 kbps.

[Valori predefiniti](#)

Nessuna

[Modalità dei comandi](#)

Configurazione Gatekeeper

[Cronologia comandi](#)

Nella tabella seguente viene descritta la cronologia dei comandi:

Release	Modifica
12.1(3)XI	Questo comando è stato introdotto.
12.1(5)XM	Il comando bandwidth è stato reso riconoscibile senza usare il comando zone gatekeeper .
12.2(2)T	Questo comando è stato integrato nel software Cisco IOS versione 12.2(2)T.
12.2(2)XB 1	Questo comando è stato implementato sul gateway universale Cisco AS5850.

[Linee guida per l'utilizzo](#)

Nelle versioni precedenti del software Cisco IOS, la funzionalità del comando **bandwidth** è stata ottenuta con il comando **zone gatekeeper**.

[Esempi](#)

L'esempio seguente configura la larghezza di banda massima per la zona a 5.000 kbps:

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

Comandi correlati

bandwidth remote: specifica la larghezza di banda totale per il traffico H.323 tra questo gatekeeper e qualsiasi altro gatekeeper.

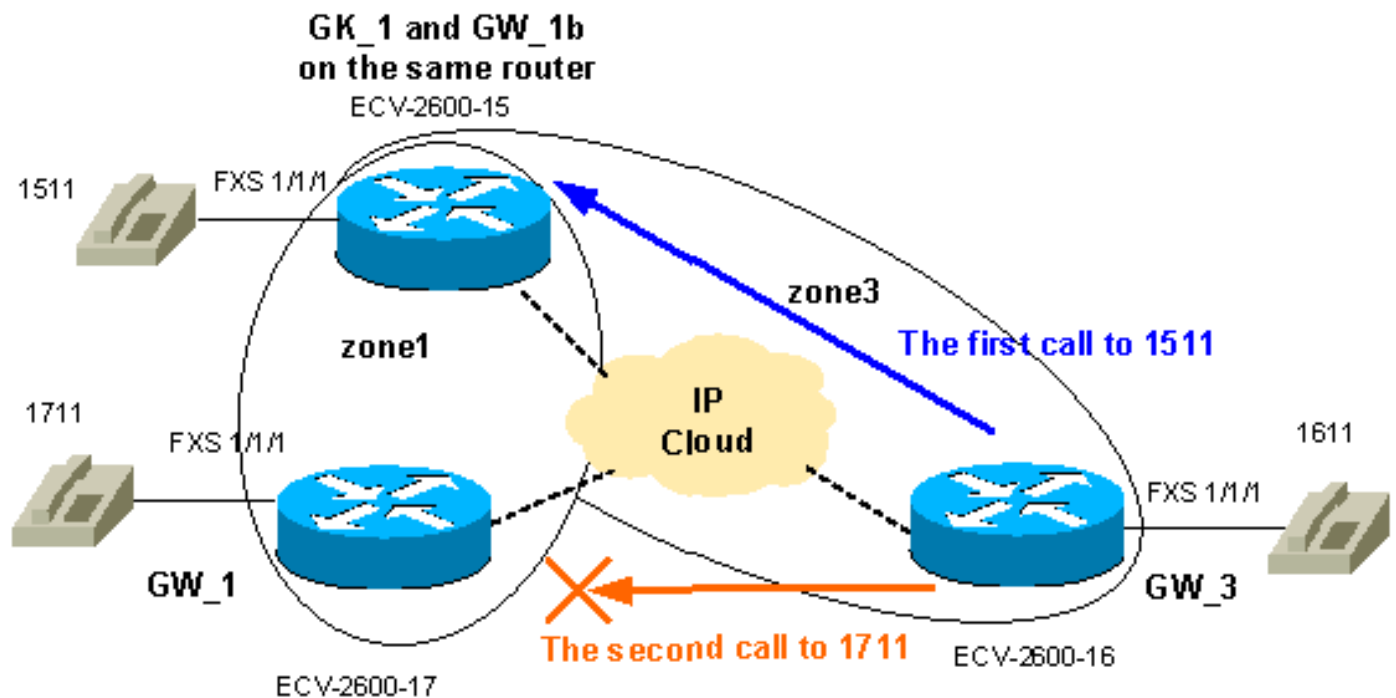
Configurazione

In questa sezione vengono presentate le informazioni necessarie per configurare le funzionalità descritte più avanti nel documento.

Nota: per ulteriori informazioni sui comandi menzionati in questo documento, usare lo [strumento di ricerca dei comandi](#) (solo utenti [registrati](#)).

Esempio di rete

Nel documento viene usata questa impostazione di rete:



Configurazioni

Lo scopo è quello di limitare la larghezza di banda disponibile tra la zona 1 e la zona 3 a una chiamata e consentire un numero maggiore di chiamate (fino a due in questo esempio) nella zona 1. In questo modo verranno soddisfatti i requisiti generali per il tipico task di ammissione di chiamata. I messaggi Registration, Admission, and Status Protocol (RAS) precedono i messaggi di impostazione della chiamata H225. Segue quindi la negoziazione H4245, che in realtà definisce le funzionalità dei due lati. La larghezza di banda reale della chiamata viene definita dopo la fase di ammissione di chiamata e lo scambio dei messaggi RAS. Ecco perché il gatekeeper tratta ogni chiamata come una chiamata a 64 kb. Pertanto, le limitazioni della larghezza di banda tra le zone per le chiamate vocali devono essere incrementate di 64 kb.

Nota: il GW_3 è configurato sullo stesso router del gatekeeper per illustrare questa possibilità per le filiali di fascia bassa.

Nota: la verifica delle configurazioni del gatekeeper e del gateway è la parte importante della risoluzione dei problemi relativi a GK-GW. Pertanto, per semplificare la comprensione delle configurazioni, tutti i comandi di configurazione non correlati sono stati rimossi.

GW_1 ECV-2600-17

```
IOS (tm) C2600 Software (C2600-JSX-M), Version 12.2(7a),  
RELEASE SOFTWARE (fc1)  
!  
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 64kb. 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#
```

Verifica

Le informazioni contenute in questa sezione permettono di verificare che la configurazione funzioni correttamente.

Alcuni comandi **show** sono supportati dallo [strumento Output Interpreter \(solo utenti registrati\)](#); lo strumento permette di visualizzare un'analisi dell'output del comando **show**.

- **show gateway**: visualizza lo stato di registrazione del gateway.
- **show gatekeeper endpoints**: elenca tutti i gateway registrati per il gatekeeper.
- **show gatekeeper zone prefix**: visualizza tutti i prefissi di zona configurati sul gatekeeper.
- **show gatekeeper call**: visualizza le chiamate attive elaborate dal gatekeeper.

Risoluzione dei problemi

Le informazioni contenute in questa sezione permettono di risolvere i problemi relativi alla configurazione.

Comandi per la risoluzione dei problemi

Alcuni comandi **show** sono supportati dallo [strumento Output Interpreter \(solo utenti registrati\)](#); lo strumento permette di visualizzare un'analisi dell'output del comando **show**.

Nota: prima di usare i comandi di **debug**, consultare le [informazioni importanti sui comandi di debug](#).

- **debug h225 asn1**: visualizza i messaggi H225 (RAS e Q931 call setup).
- **debug cch323 h225**: visualizza i messaggi di impostazione della chiamata H225.

Di seguito sono riportati alcuni collegamenti utili:

- [Risoluzione dei problemi e debug delle chiamate VoIP - Nozioni fondamentali](#)
- [Comandi di debug VoIP](#)
- [Guida di riferimento ai comandi voce, video e fax di Cisco IOS, versione 12.2](#)

Output di esempio di show e 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

=====

Table with 6 columns: CallSignalAddr, Port, RASSignalAddr, Port, Zone Name, Type F. It lists three active registrations for gatekeepers gw_1b, gw_3, and 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_1b	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
}
```



```

{
'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 ::=
!--- The GW_3 get some facility messages from GW_1b. value H323_UU_NonStdInfo ::= { version 16
protoParam qsigNonStdInfo :
{
iei 30
rawMesg '1E028188'H
}
}
```

```
*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 21:54:30.205: **RAS OUTGOING PDU** ::=

!--- The GK grants permission to GW_3, as there are no restrictions for zone3. value RasMessage

::= **admissionConfirm** :

```
{
  requestSeqNum 971
```

```

bandWidth 640
callModel direct : NULL
destCallSignalAddress ipAddress :
{
  ip '0A34DA31'H
  !--- The hexadecimal number is 10.52.218.49, IP of GW_1. 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 21:54:30.217: RAS OUTGOING ENCODE BUFFER::= 2B 0003CA40 0280000A 34DA3106 B800EF14 00C00100
020000 *Mar 11 21:54:30.221: *Mar 11 21:54:30.429: RAS INCOMING 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 11 21:54:30.445: *Mar
11 21:54:30.445: RAS INCOMING PDU ::=
!--- The incoming request from GW_1 asks for permission to accept call from GW_3. 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 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
-----
```

[Informazioni correlate](#)

- [Risoluzione dei problemi e informazioni sulla gestione della larghezza di banda del Gatekeeper Cisco](#)
- [Informazioni sui gatekeeper H.323](#)
- [Cisco High-Performance Gatekeeper](#)
- [Configurazione dei gateway H.323](#)
- [Configurazione dei gatekeeper H.323](#)
- [Configurazione del supporto H.323 per le interfacce virtuali](#)
- [Supporto alla tecnologia vocale](#)
- [Supporto ai prodotti voce e Unified Communications](#)
- [Risoluzione dei problemi di Cisco IP Telephony](#)
- [Supporto tecnico – Cisco Systems](#)