

Resolución de problemas e introducción a la administración del ancho de banda del Gatekeeper del Cisco

Contenido

[Introducción](#)

[Información general sobre el funcionamiento de la administración de ancho de banda](#)

[Cómo Configurar la Función Bandwidth Management en Cisco Gatekeeper](#)

[Comandos show de Gatekeeper utilizados para mostrar información sobre el ancho de banda](#)

[Mensajes RAS relacionados con el ancho de banda \(BRQ/BCF/BRJ\)](#)

[Mensajes RAS utilizados para informar sobre el estado de ancho de banda](#)

[Cómo se dispara la petición de ancho de banda desde la puerta de enlace para notificar al Gatekeeper para reducir el ancho de banda de llamada](#)

[Examples](#)

[Información Relacionada](#)

Introducción

Este documento asume que el lector tiene un conocimiento básico de Cisco IOS® Software Gatekeepers y de las mensajerías de Registro, admisión y de estado (RAS) de gatekeeper a gateway H.225. Remítase a Introducción a los Gatekeepers H.323 para obtener más información.

Según la recomendación H.323, los Gatekeepers del IOS de Cisco deben soportar estos mensajes de administración de ancho de banda RAS H.225:

- Petición de ancho de banda (BRQ)
- Rechazo de ancho de banda (BRJ)
- Mensajes de confirmación de ancho de banda (BCF)

Este concepto se puede basar en la administración del ancho de banda. También puede ser una función nula que acepta todas las solicitudes de cambios de ancho de banda. En otras palabras, el Gatekeeper puede utilizar estos mensajes para administrar el ancho de banda si permite o rechaza solicitudes o simplemente ignorarlas.

Información general sobre el funcionamiento de la administración de ancho de banda

El gatekeeper de Cisco puede rechazar llamadas de un terminal debido a limitaciones de ancho de banda. Esto puede ocurrir si el Gatekeeper determina que no hay suficiente ancho de banda disponible en la red para soportar la llamada. Esta función también opera durante una llamada activa cuando una terminal solicita ancho de banda adicional o informa sobre un cambio del

ancho de banda utilizado para la llamada.

El controlador de acceso de Cisco mantiene un registro de todas las llamadas activas a fin de administrar los recursos de ancho de banda en su zona. En una configuración de clúster, el mensaje de indicación de anuncio del protocolo de actualización del control de acceso (GUP) se intercambia cada intervalo de tiempo establecido y transporta información sobre el uso del ancho de banda para la zona. Este intercambio de mensajes GUP permite a los Gatekeepers alternativos administrar correctamente el ancho de banda para una zona única, aunque los Gatekeepers estén en dispositivos físicos separados.

Cuando decide si hay suficiente ancho de banda para aceptar una solicitud de admisión de llamada (ARQ), Cisco Gatekeeper calcula el ancho de banda disponible con esta fórmula:

$$\text{Available_bandwidth} = (\text{total_assigned_bandwidth}) - (\text{bandwidth_used_local}) - (\text{bandwidth_used_by_all_alternates}).$$

Si el ancho de banda disponible es suficiente para la llamada, se devuelve una Confirmación de admisión (ACF); de lo contrario, se devuelve un Rechazo de admisión (ARJ).

Los gateways de voz deben tener en cuenta las funciones de códec, encapsulación de capa 2 y compresión, como RTP comprimido [cRTP], cuando solicitan ancho de banda al gatekeeper de Cisco. A veces estas funciones no se definen en el momento de la configuración de la llamada, en cuyo caso se puede enviar una solicitud de cambio de ancho de banda al gatekeeper después de la configuración de la llamada para ajustar la cantidad de ancho de banda que utiliza la llamada.

Nota: A partir de Cisco IOS Software Release 12.2(2)XA, Cisco ha implementado solamente la funcionalidad del informe de cualquier cambio de ancho de banda cuando cambian los códecs. Consulte la sección: [Cómo se dispara la petición de ancho de banda desde la puerta de enlace para notificar al control de acceso sobre la reducción del ancho de banda de llamada, para obtener más información.](#)

[Cómo Configurar la Función Bandwidth Management en Cisco Gatekeeper](#)

A partir de Cisco IOS Software Release 12.3(1), estos tipos de limitaciones de ancho de banda de zona se pueden configurar en Cisco Gatekeeper:

- El ancho de banda máximo para todo el tráfico H.323 entre la zona local y una zona remota especificada. Si lo desea, esta configuración se puede repetir individualmente para cada zona remota.
- El ancho de banda máximo permitido para una sola sesión en la zona local, normalmente utilizado para aplicaciones de vídeo, no para voz
- El ancho de banda máximo para todo el tráfico H.323 permitido colectivamente a todas las zonas remotas
- El nuevo comando **bandwidth check-destination** verifica el ancho de banda del terminal de destino antes de que responda a ARQ. Este comando se introdujo en la versión 12.3(1) del software del IOS de Cisco.

Utilice estos comandos para configurar el ancho de banda de la zona de Cisco Gatekeeper:

- **bandwidth {interzone | total | session} {default | zone zone-name} max-bandwidth**

| LOCAL GK NAME | ALT GK NAME | PRI | (kbps) | (kbps) | (kbps) | ANNOUNCE | STATUS |
|---------------|-------------|-----|--------|--------|--------|----------|-----------|
| gkb-1 | gkb-2 | 0 | 0 | 0 | 0 | 22s | CONNECTED |

Ingrese el comando **show gatekeeper calls** para mostrar las llamadas activas permitidas por ese gatekeeper y el ancho de banda que cada una utiliza.

```
gkb-1#show gatekeeper calls
Total number of active calls = 1.
                GATEKEEPER CALL INFO
                =====
LocalCallID           Age(secs)   BW
3-63466                9           128 (Kbps)
  Endpt(s): Alias      E.164Addr
    src EP: gwa-1      4085272923
  Endpt(s): Alias      E.164Addr
    dst EP: gwb-1      3653
    CallSignalAddr    Port  RASSignalAddr  Port
    172.16.13.23     1720 172.16.13.23   54670
```

[Mensajes RAS relacionados con el ancho de banda \(BRQ/BCF/BRJ\)](#)

El mensaje BRQ se utiliza para solicitar un cambio en el ancho de banda del Cisco Gatekeeper. Este es el procedimiento:

1. El gatekeeper de Cisco verifica la solicitud por el identificador de punto final para localizar el punto final en la base de datos de registro.
2. Localiza el registro de llamada mediante el uso de callReferenceValue para encontrar una llamada asociada al terminal con la misma llamadaReferenceValue.
3. Si encuentra el registro de llamadas, computa el cambio de ancho de banda, y luego lo suma o resta al ancho de banda de zona global, según sea necesario. Hace lo mismo en relación con cualquier proxy o recurso de gateway en uso.
4. Un mensaje BCF o BRJ se envía de vuelta al terminal, lo que depende del éxito o del fracaso.

[Mensajes RAS utilizados para informar sobre el estado de ancho de banda](#)

El campo Information Request Response (IRR) "Non-Standard Data" (Datos no estándar) también contiene información sobre el ancho de banda utilizado actual en un gateway o proxy.

[Cómo se dispara la petición de ancho de banda desde la puerta de enlace para notificar al Gatekeeper para reducir el ancho de banda de llamada](#)

Antes de Cisco IOS Software Release 12.2(2)XA en el Cisco H.323 Gateway, siempre se informaba de las llamadas para requerir un ancho de banda de 64 kbps. Este es el ancho de banda unidireccional para un códec Cisco G.711. Si los puntos finales en la llamada utilizaron un códec más eficiente, esto no se informó al Gatekeeper de Cisco. En la versión 12.2(2)XA del software Cisco IOS de la gateway Cisco H.323 o posterior, que cumple con la versión 3 de H.323, el ancho de banda informado es bidireccional. Inicialmente, se reservan 128 kb. Si los terminales de la llamada seleccionan un códec más eficiente, se notifica al gatekeeper de Cisco el cambio de ancho de banda.

Nota: Configure el Cisco H.323 Gateway con este comando en el modo de configuración global para utilizar el comportamiento de ancho de banda informado utilizado antes de Cisco IOS Software Release 12.2(2)XA para la administración de ancho de banda de zona:

```
Router(config-gateway)#emulate cisco h323 bandwidth
```

Examples

Esta sección trata estos dos ejemplos:

- [Administración del ancho de banda en una topología de agrupamiento](#)
- [Utilice BRQ para informar sobre ancho de banda](#)

Ejemplo 1: Administración del ancho de banda en una topología de agrupamiento

Vea las depuraciones capturadas de un gatekeeper de Cisco en un clúster. La depuración muestra los mensajes ARQ y ACF, que incluye el ancho de banda necesario para la llamada. Después de recibir estos mensajes, el gatekeeper de Cisco actualiza los otros gatekeepers del clúster acerca de este cambio de ancho de banda.

Nota: Estos comandos se utilizan para capturar este resultado: **debug h225 asn1, debug ras, debug gatekeeper gup asn1, debug gatekeeper gup events.**

```
Mar  2 23:59:26.802:
Mar  2 23:59:26.802: RAS INCOMING PDU ::=

value RasMessage ::= admissionRequest :
!--- ARQ is received. { requestSeqNum 5928 callType pointToPoint : NULL callModel direct : NULL
endpointIdentifier {"6196296800000001"} destinationInfo { e164 : "3653" } srcInfo { e164 :
"4085272923", h323-ID : {"gwa-1"} } srcCallSignalAddress ipAddress : { ip 'AC100D0F'H port 11002
} bandwidth 1280
!--- Intial bandwidth of 128k is requested. callReferenceValue 14 nonStandardData {
nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18
} data '80000008800180'H } conferenceID 'C8C66C7D168011CC800C8828285B8DF6'H activeMC FALSE
answerCall TRUE canMapAlias TRUE callIdentifier { guid 'C8C66C7D168011CC800D8828285B8DF6'H }
willSupplyUUIEs FALSE } Mar  2 23:59:26.810: ARQ (seq# 5928) rcvd Mar  2 23:59:26.810: H225 NONSTD
INCOMING ENCODE BUFFER::= 80 00000880 0180 Mar  2 23:59:26.810: Mar  2 23:59:26.810: H225 NONSTD
INCOMING PDU ::= value ARQnonStandardInfo ::= { sourceAlias { } sourceExtAlias { }
callingOctet3a 128 } parse_arq_nonstd: ARQ Nonstd decode succeeded, remlen = 129 Mar  2
23:59:26.814: RAS OUTGOING PDU ::= value RasMessage ::= admissionConfirm :
!--- ACF is sent back. { requestSeqNum 5928 bandwidth 1280
!--- BW value is included. callModel direct : NULL destCallSignalAddress ipAddress : { ip
'AC100D17'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  2 23:59:26.818: RAS OUTGOING ENCODE BUFFER::=
2B 00172740 050000AC 100D1706 B800EF1A 00C00100 020000 Mar  2 23:59:26.818: Mar  2 23:59:26.818:
IPSOCK_RAS_sendto: msg length 24 from 172.16.13.41:1719 to 172.16.13.23: 51874 Mar  2
23:59:26.822: RASLib::RASsendACF: ACF (seq# 5928) sent to 172.16.13.23 Mar  2 23:59:36.046: GUP
OUTGOING PDU ::=

value GUP_Information ::=
!--- GUP update message is sent to all gatekeepers in the cluster. { protocolIdentifier { 1 2
840 113548 10 0 0 2 } message announcementIndication : { announcementInterval 30
endpointCapacity 46142 callCapacity 68793 hostName '676B622D31'H percentMemory 25 percentCPU 0
currentCalls 1
```

```

currentEndpoints 2
zoneInformation
{
    {
        gatekeeperIdentifier {"gkb-1"}
        altGKIdentifier {"gkb-2"}
        totalBandwidth 1280
!--- BW info is included. interzoneBandwidth 1280
        remoteBandwidth 1280
    }
}
}
}

```

```

Mar  2 23:59:36.050: GUP OUTGOING ENCODE BUFFER::= 00 0A2A8648 86F70C0A
00000220 001E40B4 3E80010C B904676
B 622D3132 00010002 01420000 67006B00 62002D00 31080067 006B0062
002D0032 40050040 05004005 00
Mar  2 23:59:36.054:
Mar  2 23:59:36.054: Sending GUP ANNOUNCEMENT INDICATION to 172.16.13.16

```

[Ejemplo 2: Utilice BRQ para informar sobre ancho de banda](#)

Busque depuraciones de un gatekeeper de Cisco en una configuración donde el ancho de banda en el gatekeeper remoto está limitado a 144 kbps. En la depuración verá que el ARQ solicitado es un ancho de banda inicial de 128 kbps. Cuando se configura la llamada, el punto final informa acerca del cambio en el ancho de banda con un mensaje BRQ y el ancho de banda utilizado en 16 kbps; esto significa que la llamada se configuró con el códec G729 de Cisco. Luego, se solicita otra llamada y se procede de la misma manera.

Tenga en cuenta que si la segunda llamada llega antes de que el terminal solicitara el cambio en el ancho de banda para la primera llamada, el gatekeeper de Cisco rechaza esa llamada, ya que $128+128=256$ kbps y eso es más de 144 kbps configurados.

```

!
!
!
gatekeeper
zone local gka-1 domainA.com 172.16.13.35
zone remote gkb-1 domainB.com 172.16.13.41 1719
zone prefix gkb-1 36*
zone prefix gka-1 53*
gw-type-prefix 1#* default-technology
bandwidth remote 144
no shutdown
endpoint ttl 120
!

```

Esta salida se capturó con los comandos **debug h225 asn1** y **debug ras**:

```

gka-1#show logging
Syslog logging: enabled (0 messages dropped, 0 messages rate-limited, 0
flushes, 0 overruns)
  Console logging: disabled
  Monitor logging: level debugging, 1076 messages logged
  Buffer logging: level debugging, 203860 messages logged

```

Logging Exception size (4096 bytes)

Trap logging: level informational, 66 message lines logged

Log Buffer (9999999 bytes):

Mar 14 20:18:06.385: RAS INCOMING ENCODE BUFFER ::= 27 88039700 F0003800
31004600 36004100 38003900 38003000 30003000 30003000 30003000 31010180
69860140 04006700 77006100 2D003140 0500000B 40B50000 12138000 0008A001
800B1249 53444E2D 564F4943 45DA4A9C E21FCF11 CC802093 7822E08B 6308E020
00018011 00DA4A9C E21FCF11 CC802193 7822E08B 630100

Mar 14 20:18:06.401:

Mar 14 20:18:06.405: RAS INCOMING PDU ::=

value RasMessage ::= **admissionRequest** :

!--- ARQ is received. { requestSeqNum 920 callType pointToPoint : NULL callModel direct : NULL
endpointIdentifier {"81F6A89800000001"} destinationInfo { e164 : "3653" } srcInfo { h323-ID :
{"gwa-1"} } **bandwidth 1280**

!--- Intial BW of 128 kpbs is requested. callReferenceValue 11 nonStandardData {
nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18
} data '80000008A001800B124953444E2D564F494345'H } conferenceID

'DA4A9CE21FCF11CC8020937822E08B63'H activeMC FALSE answerCall FALSE canMapAlias TRUE
callIdentifier { guid 'DA4A9CE21FCF11CC8021937822E08B63'H } willSupplyUUIEs FALSE } Mar 14
20:18:06.425: H225 NONSTD INCOMING ENCODE BUFFER ::= 80 000008A0 01800B12 4953444E 2D564F49 4345

Mar 14 20:18:06.429: Mar 14 20:18:06.429: H225 NONSTD INCOMING PDU ::= value ARQnonStandardInfo
::= { sourceAlias { } sourceExtAlias { } callingOctet3a 128 interfaceSpecificBillingId "ISDN-
VOICE" } Mar 14 20:18:06.433: H225 NONSTD OUTGOING PDU ::= value LRQnonStandardInfo ::= { ttl 6
nonstd-callIdentifier { guid 'DA4A9CE21FCF11CC8021937822E08B63'H } callingOctet3a 128

gatewaySrcInfo { h323-ID : {"gwa-1"} } } Mar 14 20:18:06.437: H225 NONSTD OUTGOING ENCODE
BUFFER ::= 82 86B01100 DA4A9CE2 1FCF11CC 80219378 22E08B63 01800D01 40040067 00770061 002D0031
Mar 14 20:18:06.445: Mar 14 20:18:06.445: RAS OUTGOING PDU ::= value RasMessage ::=

locationRequest : { requestSeqNum 2061 destinationInfo { e164 : "3653" } nonStandardData {
nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18
} data '8286B01100DA4A9CE21FCF11CC8021937822E08B...'H } replyAddress ipAddress : { ip

'AC100D23'H port 1719 } sourceInfo { h323-ID : {"gka-1"} } canMapAlias TRUE } Mar 14
20:18:06.461: RAS OUTGOING ENCODE BUFFER ::= 4A 80080C01 01806986 40B50000 12258286 B01100DA
4A9CE21F CF11CC80 21937822 E08B6301 800D0140 04006700 77006100 2D003100 AC100D23 06B70B80

0D014004 0067006B 0061002D 00310180 Mar 14 20:18:06.469: Mar 14 20:18:06.473: RAS OUTGOING PDU
::= value RasMessage ::= requestInProgress : { requestSeqNum 920 delay 9000 } Mar 14
20:18:06.473: RAS OUTGOING ENCODE BUFFER ::= 80 05000397 2327 Mar 14 20:18:06.473: Mar 14

20:18:06.477: RAS INCOMING ENCODE BUFFER ::= 4F 080C00AC 100D1706 B800AC10 0D17DC0E 40B50000
12390001 40040067 00770062 002D0031 08006700 6B006200 2D003101 10014004 00670077 0062002D
003100AC 100D1706 B8000000 00000000 00000010 40080880 013C0501 0000 Mar 14 20:18:06.489: Mar 14

20:18:06.489: RAS INCOMING PDU ::= value RasMessage ::= locationConfirm : { requestSeqNum 2061
callSignalAddress ipAddress : { ip 'AC100D17'H port 1720 } rasAddress ipAddress : { ip
'AC100D17'H port 56334 } nonStandardData { nonStandardIdentifier h221NonStandard : {

t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data
'00014004006700770062002D0031080067006B00...'H } destinationType { gateway { protocol { voice :
{ supportedPrefixes { } } } } mc FALSE undefinedNode FALSE } } Mar 14 20:18:06.509: H225 NONSTD
INCOMING ENCODE BUFFER ::= 00 01400400 67007700 62002D00 31080067 006B0062 002D0031 01100140

04006700 77006200 2D003100 AC100D17 06B80000 00000000 00000000 Mar 14 20:18:06.517: Mar 14
20:18:06.521: H225 NONSTD INCOMING PDU ::= value LCFnonStandardInfo ::= { termAlias { h323-ID :
{"gwb-1"} } gkID {"gkb-1"} gateways { { gwType voip : NULL gwAlias { h323-ID : {"gwb-1"} }
sigAddress { ip 'AC100D17'H port 1720 } resources { maxDSPs 0 inUseDSPs 0 maxBChannels 0
inUseBChannels 0 activeCalls 0 bandwidth 0 inuseBandwidth 0 } } } } Mar 14 20:18:06.537: RAS

OUTGOING PDU ::= value RasMessage ::= **admissionConfirm** :

!--- ACF is sent back. { requestSeqNum 920 **bandwidth 1280**
!--- BW is included. callModel direct : NULL destCallSignalAddress ipAddress : { ip 'AC100D17'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 14 20:18:06.549: RAS OUTGOING ENCODE BUFFER ::= 2B 00039740

050000AC 100D1706 B800EF1A 00C00100 020000 Mar 14 20:18:06.553: Mar 14 20:18:06.677: RAS
INCOMING ENCODE BUFFER ::= 32 0003981E 00380031 00460036 00410038 00390038 00300030 00300030
00300030 00300031 DA4A9CE2 1FCF11CC 80209378 22E08B63 000B00A0 15080011 00DA4A9C E21FCF11

CC802193 7822E08B 630100 Mar 14 20:18:06.685: Mar 14 20:18:06.689: RAS INCOMING PDU ::= value

```
RasMessage ::= bandwidthRequest :
!--- BRQ message to request bandwidth to be changed to 16 kpbs. { requestSeqNum 921
endpointIdentifier {"81F6A89800000001"} conferenceID 'DA4A9CE21FCF11CC8020937822E08B63'H
callReferenceValue 11 bandwidth 160
!--- 16 kpbs is requested. callIdentifier { guid 'DA4A9CE21FCF11CC8021937822E08B63'H }
answeredCall FALSE } Mar 14 20:18:06.697: RAS OUTGOING PDU ::= value RasMessage ::=
bandwidthConfirm :
!--- BCF is sent back approving the bandwidth request change. { requestSeqNum 921 bandwidth 160
}
```

```
Mar 14 20:18:06.697: RAS OUTGOING ENCODE BUFFER ::= 34 039800A0
Mar 14 20:18:06.701:
Mar 14 20:18:12.066: RAS INCOMING ENCODE BUFFER ::= 0E 40039906 0008914A
00030000 0100AC10 0D0FE511 00040067 006B0061 002D0031 00B50000 12288F00
0002003B 0180211E 00380031 00460036 00410038 00390038 00300030 00300030
00300030 00300031 01000180
Mar 14 20:18:12.074:
Mar 14 20:18:12.078: RAS INCOMING PDU ::=
```

```
value RasMessage ::= registrationRequest :
{
  requestSeqNum 922
  protocolIdentifier { 0 0 8 2250 0 3 }
  discoveryComplete FALSE
  callSignalAddress
  {
  }
  rasAddress
  {
    ipAddress :
    {
      ip 'AC100D0F'H
      port 58641
    }
  }
  terminalType
  {
    mc FALSE
    undefinedNode FALSE
  }
  gatekeeperIdentifier {"gka-1"}
  endpointVendor
  {
    vendor
    {
      t35CountryCode 181
      t35Extension 0
      manufacturerCode 18
    }
  }
  timeToLive 60
  keepAlive TRUE
  endpointIdentifier {"81F6A89800000001"}
  willSupplyUUIEs FALSE
  maintainConnection TRUE
}
```

```
Mar 14 20:18:12.098: RAS OUTGOING PDU ::=
```

```
value RasMessage ::= registrationConfirm :
```



```

{
  requestSeqNum 922
  protocolIdentifier { 0 0 8 2250 0 3 }
  callSignalAddress
  {
  }
  gatekeeperIdentifier {"gka-1"}
  endpointIdentifier {"81F6A89800000001"}
  alternateGatekeeper
  {
  }
  timeToLive 60
  willRespondToIRR FALSE
  maintainConnection TRUE
}

```

```

Mar 14 20:18:12.106: RAS OUTGOING ENCODE BUFFER ::= 12 40039906 0008914A
00030008 0067006B 0061002D 00311E00 38003100 46003600 41003800 39003800
30003000 30003000 30003000 3000310F 8A010002 003B0100 0180

```

```

Mar 14 20:18:12.114:

```

```

Mar 14 20:18:14.586: RAS INCOMING ENCODE BUFFER ::= 5A C0039A08 80013C05
04010020 40078000 38003100 46003600 41003800 39003800 30003000 30003000
30003000 30003100 AC100D0F E5110100 AC100D0F 06B80140 04006700 77006100
2D003101 C100B500 00120570 2BA39307 000BDA4A 9CE21FCF 11CC8020 937822E0
8B630000 A003C000 1100DA4A 9CE21FCF 11CC8021 937822E0 8B630E20 0100

```

```

Mar 14 20:18:14.602:

```

```

Mar 14 20:18:14.602: RAS INCOMING PDU ::=

```

```

value RasMessage ::= infoRequestResponse :

```

```

!--- IRR message is received and it includes the bandwidth used on the gateway. { requestSeqNum
923 endpointType { gateway { protocol { voice : { supportedPrefixes { { prefix e164 : "1#" } } } }
} } mc FALSE undefinedNode FALSE } endpointIdentifier {"81F6A89800000001"} rasAddress ipAddress
: { ip 'AC100D0F'H port 58641 } callSignalAddress { ipAddress : { ip 'AC100D0F'H port 1720 } }
endpointAlias { h323-ID : {"gwa-1"} } perCallInfo { { nonStandardData { nonStandardIdentifier
h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '702BA39307'H }
callReferenceValue 11 conferenceID 'DA4A9CE21FCF11CC8020937822E08B63'H h245 { } callSignaling {
} callType pointToPoint : NULL bandwidth 160
  callModel direct : NULL
  callIdentifier
  {
    guid 'DA4A9CE21FCF11CC8021937822E08B63'H
  }
}
}
}
needResponse FALSE
}

```

```

Mar 14 20:18:14.646: H225 NONSTD INCOMING ENCODE BUFFER ::= 70 2BA39307

```

```

Mar 14 20:18:14.646:

```

```

Mar 14 20:18:14.646: H225 NONSTD INCOMING PDU ::=

```

```

value IRRperCallnonStandardInfo ::=

```

```

{
  startTime 732140295
}

```

```

Mar 14 20:18:28.008: RAS INCOMING ENCODE BUFFER ::= 27 88039B00 F0003800
31004600 36004100 38003900 38003000 30003000 30003000 30003000 31010180

```

```
69860140 04006700 77006100 2D003140 0500000C 40B50000 12030000 00000000
00000000 00000000 00000000 0008E020 00018011 00000000 00000000 00000000
00000000 000100
Mar 14 20:18:28.024:
Mar 14 20:18:28.024: RAS INCOMING PDU ::=
```

```
value RasMessage ::= admissionRequest :
{
  requestSeqNum 924
  callType pointToPoint : NULL
  callModel direct : NULL
  endpointIdentifier {"81F6A89800000001"}
  destinationInfo
  {
    e164 : "3653"
  }
  srcInfo
  {
    h323-ID : {"gwa-1"}
  }
  bandwidth 1280
  callReferenceValue 12
  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 14 20:18:28.044: H225 NONSTD INCOMING ENCODE BUFFER::= 00 0000
Mar 14 20:18:28.044:
Mar 14 20:18:28.044: H225 NONSTD INCOMING PDU ::=
```

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

```
Mar 14 20:18:28.048: H225 NONSTD OUTGOING PDU ::=
```

```
value LRQnonStandardInfo ::=
```

```
{
  ttl 6
  nonstd-callIdentifier
  {
    guid '00000000000000000000000000000000'H
  }
  gatewaySrcInfo
  {
    h323-ID : {"gwa-1"}
  }
}
```

Mar 14 20:18:28.056: H225 NONSTD OUTGOING ENCODE BUFFER::= 82 86901100
00000000 00000000 00000000 00000000 0D014004 00670077 0061002D 0031

Mar 14 20:18:28.060:

Mar 14 20:18:28.060: RAS OUTGOING PDU ::=

value RasMessage ::= locationRequest :

```
{
  requestSeqNum 2062
  destinationInfo
  {
    e164 : "3653"
  }
  nonStandardData
  {
    nonStandardIdentifier h221NonStandard :
    {
      t35CountryCode 181
      t35Extension 0
      manufacturerCode 18
    }
    data '828690110000000000000000000000000000000000000000...'H
  }
  replyAddress ipAddress :
  {
    ip 'AC100D23'H
    port 1719
  }
  sourceInfo
  {
    h323-ID : {"gka-1"}
  }
  canMapAlias TRUE
}
```

Mar 14 20:18:28.076: RAS OUTGOING ENCODE BUFFER::= 4A 80080D01 01806986
40B50000 12238286 90110000 00000000 00000000 00000000 0000000D 01400400
67007700 61002D00 3100AC10 0D2306B7 0B800D01 40040067 006B0061 002D0031
0180

Mar 14 20:18:28.084:

Mar 14 20:18:28.088: RAS OUTGOING PDU ::=

value RasMessage ::= requestInProgress :

```
{
  requestSeqNum 924
  delay 9000
}
```

```
Mar 14 20:18:28.088: RAS OUTGOING ENCODE BUFFER ::= 80 0500039B 2327
Mar 14 20:18:28.088:
Mar 14 20:18:28.097: RAS INCOMING ENCODE BUFFER ::= 4F 080D00AC 100D1706
B800AC10 0D17DC0E 40B50000 12390001 40040067 00770062 002D0031 08006700
6B006200 2D003101 10014004 00670077 0062002D 003100AC 100D1706 B8000000
00000000 00000010 40080880 013C0501 0000
Mar 14 20:18:28.105:
Mar 14 20:18:28.109: RAS INCOMING PDU ::=
```

```
value RasMessage ::= locationConfirm :
{
  requestSeqNum 2062
  callSignalAddress ipAddress :
  {
    ip 'AC100D17'H
    port 1720
  }
  rasAddress ipAddress :
  {
    ip 'AC100D17'H
    port 56334
  }
  nonStandardData
  {
    nonStandardIdentifier h221NonStandard :
    {
      t35CountryCode 181
      t35Extension 0
      manufacturerCode 18
    }
    data '00014004006700770062002D0031080067006B00...'H
  }
  destinationType
  {
    gateway
    {
      protocol
      {
        voice :
        {
          supportedPrefixes
          {
            }
          }
        }
      }
    }
    mc FALSE
    undefinedNode FALSE
  }
}
```

```
Mar 14 20:18:28.129: H225 NONSTD INCOMING ENCODE BUFFER ::= 00 01400400
67007700 62002D00 31080067 006B0062 002D0031 01100140 04006700 77006200
2D003100 AC100D17 06B80000 00000000 00000000
Mar 14 20:18:28.133:
Mar 14 20:18:28.137: H225 NONSTD INCOMING PDU ::=
```

```
value LCFnonStandardInfo ::=
{
  termAlias
  {
```

```

    h323-ID : {"gwb-1"}
  }
  gkID {"gkb-1"}
  gateways
  {
    {
      gwType voip : NULL
      gwAlias
      {
        h323-ID : {"gwb-1"}
      }
      sigAddress
      {
        ip 'AC100D17'H
        port 1720
      }
      resources
      {
        maxDSPs 0
        inUseDSPs 0
        maxBChannels 0
        inUseBChannels 0
        activeCalls 0
        bandwidth 0
        inuseBandwidth 0
      }
    }
  }
}

```

Mar 14 20:18:28.153: RAS OUTGOING PDU ::=

```

value RasMessage ::= admissionConfirm :
{
  requestSeqNum 924
  bandwidth 1280
  callModel direct : NULL
  destCallSignalAddress ipAddress :
  {
    ip 'AC100D17'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 14 20:18:28.169: RAS OUTGOING ENCODE BUFFER ::= 2B 00039B40 050000AC

```
100D1706 B800EF1A 00C00100 020000
Mar 14 20:18:28.169:
Mar 14 20:18:28.289: RAS INCOMING ENCODE BUFFER ::= 32 00039C1E 00380031
00460036 00410038 00390038 00300030 00300030 00300030 00300031 00000000
00000000 00000000 00000000 000C00A0 15080011 00000000 00000000 00000000
00000000 000100
Mar 14 20:18:28.301:
Mar 14 20:18:28.301: RAS INCOMING PDU ::=
```

```
value RasMessage ::= bandwidthRequest :
{
  requestSeqNum 925
  endpointIdentifier {"81F6A89800000001"}
  conferenceID '00000000000000000000000000000000'H
  callReferenceValue 12
  bandwidth 160
  callIdentifier
  {
    guid '00000000000000000000000000000000'H
  }
  answeredCall FALSE
}
```

```
Mar 14 20:18:28.309: RAS OUTGOING PDU ::=
```

```
value RasMessage ::= bandwidthConfirm :
{
  requestSeqNum 925
  bandwidth 160
}
```

```
Mar 14 20:18:28.313: RAS OUTGOING ENCODE BUFFER ::= 34 039C00A0
Mar 14 20:18:28.313:
```

[Información Relacionada](#)

- [Control de admisión de llamadas VoIP](#)
- [Gatekeeper de alto rendimiento de Cisco](#)
- [Mejoras en la escalabilidad e interoperabilidad de Cisco H.323](#)
- [VoIP con Gatekeeper](#)
- [Soporte de tecnología de voz](#)
- [Soporte de Productos de Voice and Unified Communications](#)
- [Troubleshooting de Cisco IP Telephony](#)
- [Soporte Técnico y Documentación - Cisco Systems](#)