

Troubleshooting e Entendendo o Gerenciamento de Largura de Banda do Cisco Gatekeeper

Contents

[Introduction](#)

[Visão geral sobre a operação do gerenciamento de largura de banda](#)

[Como configurar o recurso de gerenciamento de largura de banda no Cisco Gatekeeper](#)

[Comandos show do gatekeeper utilizados para exibir informações de largura de banda](#)

[Mensagens de RAS relacionadas à largura de banda \(BRQ/BCF/BRJ\)](#)

[Mensagens de RAS usadas para o status de largura de banda do relatório](#)

[Como o BRQ é disparado a partir do gateway para notificar o Gatekeeper para reduzir a largura de banda de chamada](#)

[Examples](#)

[Informações Relacionadas](#)

[Introduction](#)

Este documento pressupõe que o leitor está familiarizado com uma compreensão básica dos Gatekeepers do software Cisco IOS® e das Mensagens de Gatekeeper para Gateway H.225 Registration, Admission e Status (RAS). Consulte [Entendendo Gatekeepers H.323](#) para obter mais informações.

De acordo com a recomendação H.323, os Cisco IOS Gatekeepers devem suportar estas mensagens de gerenciamento de largura de banda RAS H.225:

- Requisição de Largura de Banda (BRQ)
- BRJ (Bandwidth Rejection)
- Mensagens de confirmação de largura de banda (BCF)

Esse conceito pode ser baseado no gerenciamento da largura de banda. Também pode ser uma função nula que aceita todas as solicitações de alterações de largura de banda. Em outras palavras, o Gatekeeper pode usar essas mensagens para gerenciar a largura de banda se permitir ou rejeitar solicitações ou simplesmente ignorá-las.

[Visão geral sobre a operação do gerenciamento de largura de banda](#)

O Cisco Gatekeeper pode rejeitar chamadas de um terminal devido a limitações de largura de banda. Isso pode ocorrer se o Gatekeeper determinar que não há largura de banda suficiente disponível na rede para suportar a chamada. Essa função também opera durante uma chamada ativa quando um terminal solicita largura de banda adicional ou informa uma alteração na largura de banda utilizada para a chamada.

O Cisco Gatekeeper mantém um registro de todas as chamadas ativas para poder gerenciar os recursos de largura de banda na sua zona. Em uma configuração de cluster, a mensagem de indicação de anúncio do Protocolo de Atualização de Gatekeeper (GUP) é trocada a cada intervalo de tempo definido e transporta informações sobre a utilização da largura de banda para a zona. Essa troca de mensagens GUP permite que os Gatekeepers alternativos gerenciem corretamente a largura de banda de uma única zona, mesmo que os Gatekeepers estejam em dispositivos físicos separados.

Quando você decide se há largura de banda suficiente para aceitar uma solicitação de admissão de chamada (ARQ), o Cisco Gatekeeper calcula a largura de banda disponível com esta fórmula:

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

Se a largura de banda disponível for suficiente para a chamada, uma Confirmação de Admissão (ACF) será retornada, caso contrário será retornada uma Rejeição de Admissão (ARJ).

Os gateways de voz devem considerar o codec, o encapsulamento da camada 2 e os recursos de compactação, como RTP comprimido [cRTP], quando solicitam largura de banda do Cisco Gatekeeper. Às vezes, esses recursos não são definidos no momento da configuração da chamada, caso em que uma solicitação de alteração de largura de banda pode ser emitida ao gatekeeper após a configuração da chamada para ajustar a quantidade de largura de banda usada pela chamada.

Observação: a partir da versão 12.2(2)XA do software Cisco IOS, a Cisco implementou somente a funcionalidade do relatório de qualquer alteração de largura de banda quando os codecs mudam. Consulte a seção: [Consulte Como o BRQ é disparado a partir do gateway para notificar o Gatekeeper para reduzir a largura de banda de chamada para obter mais informações](#)

[Como configurar o recurso de gerenciamento de largura de banda no Cisco Gatekeeper](#)

A partir do Cisco IOS Software Release 12.3(1), esses tipos de limitações de largura de banda de zona podem ser configurados no Cisco Gatekeeper:

- A largura de banda máxima para todo o tráfego H.323 entre a zona local e uma zona remota especificada. Se desejado, essa configuração pode ser repetida individualmente para cada zona remota.
- A largura de banda máxima permitida para uma única sessão na zona local, normalmente usada para aplicativos de vídeo, não para voz
- A largura de banda máxima para todo o tráfego H.323 permitida coletivamente a todas as zonas remotas
- O novo comando **bandwidth check-destination** verifica a largura de banda do ponto de extremidade de destino antes de responder ao ARQ. Esse comando foi introduzido no Cisco IOS Software Release 12.3(1).

Use estes comandos para configurar a largura de banda da zona do Cisco Gatekeeper:

- **bandwidth {interzone | total | session} {default | zone zone-name} max bandwidth**
- **bandwidth remote max-bandwidth**
- **bandwidth check-destination** Consulte o comando [bandwidth](#) para obter mais detalhes.

Esses valores configurados são usados para processar ARQs e BRQs.

Para um ARQ, o Cisco Gatekeeper deduz a largura de banda especificada na mensagem dos contadores de zona e/ou contadores remotos apropriados. Se isso fizer com que qualquer contador fique negativo, a chamada será negada e uma resposta ARJ será enviada com o motivo ARJ_REQ_DENIED. Se a solicitação de chamada exceder essa largura de banda, o Cisco Gatekeeper retornará uma Rejeição de Admissão (ARJ).

Quando um BRQ solicita um aumento de largura de banda, o Cisco Gatekeeper valida a solicitação em relação à zona e/ou remoto. Se a validação falhar, uma resposta BRJ será enviada com um motivo de BRJ_INSUFFICIENT_RSC e a quantidade máxima de largura de banda permitida.

[Comandos show do gatekeeper utilizados para exibir informações de largura de banda](#)

Insira o comando **show gatekeeper zone status** para exibir as informações de largura de banda para todas as zonas.

```
gkb-1#show gatekeeper zone status
                        GATEKEEPER ZONES
                        =====
GK name          Domain Name    RAS Address      PORT  FLAGS
-----          -
gkb-1            domainB.com    172.16.13.41    1719  LS
BANDWIDTH INFORMATION (kbps) :
  Maximum total bandwidth : 512
  Current total bandwidth : 128
  Current total bandwidth (w/ Alt GKs) : 128
  Maximum interzone bandwidth : 512
  Current interzone bandwidth : 128
  Current interzone bandwidth (w/ Alt GKs) : 128
  Maximum session bandwidth : 512
SUBNET ATTRIBUTES :
  All Other Subnets : (Enabled)
PROXY USAGE CONFIGURATION :
  Inbound Calls from all other zones :
    to terminals in local zone gkb-1 : use proxy
    to gateways in local zone gkb-1  : do not use proxy
    to MCUs in local zone gkb-1     : do not use proxy
  Outbound Calls to all other zones :
    from terminals in local zone gkb-1 : use proxy
    from gateways in local zone gkb-1  : do not use proxy
    from MCUs in local zone gkb-1     : do not use proxy
```

```
gka-1            domainA.com    172.16.13.35    1719  RS
```

Insira o comando **show gatekeeper zone cluster** para exibir as informações de largura de banda, caso o gatekeeper faça parte de um cluster.

```
gkb-1#show gatekeeper zone cluster
                        LOCAL CLUSTER INFORMATION
                        =====
LOCAL GK NAME ALT GK NAME    PRI  TOT BW  INT BW  REM BW  LAST ANNOUNCE  ALT GK STATUS
-----          -

```

gkb-1 gkb-2 0 0 0 0 22s CONNECTED

Insira o comando **show gatekeeper calls** para exibir as chamadas ativas permitidas por esse gatekeeper e a largura de banda que cada um usa.

```
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

Mensagens de RAS relacionadas à largura de banda (BRQ/BCF/BRJ)

A mensagem BRQ é usada para solicitar uma alteração na largura de banda do Cisco Gatekeeper. Este é o procedimento:

1. O Cisco Gatekeeper verifica a solicitação do endpointIdentifier para localizar o endpoint no banco de dados de registro.
2. Ele localiza o registro de chamada por meio do uso de callReferenceValue para localizar uma chamada associada ao endpoint com o mesmo callReferenceValue.
3. Se localizar o registro da chamada, ele calculará a alteração na largura de banda, em seguida adicionará ou subtrairá da largura de banda de zona global, conforme a necessidade. Faz o mesmo para qualquer tipo de recursos de proxy ou gateway em uso.
4. Uma mensagem BCF ou BRJ é enviada de volta ao endpoint, que depende do sucesso ou da falha.

Mensagens de RAS usadas para o status de largura de banda do relatório

O campo "Dados não padrão" de Resposta à solicitação de informações (IRR) também transporta informações sobre a largura de banda usada atualmente em um gateway ou proxy.

Como o BRQ é disparado a partir do gateway para notificar o Gatekeeper para reduzir a largura de banda de chamada

Antes do Cisco IOS Software Release 12.2(2)XA no Cisco H.323 Gateway, as chamadas eram sempre relatadas para exigir uma largura de banda de 64 kbps. Essa é a largura de banda unidirecional de um codec Cisco G.711. Se os pontos finais da chamada escolheram usar um codec mais eficiente, isto não foi relatado ao Cisco Gatekeeper. No Cisco IOS Software Release 12.2(2)XA versão do Cisco H.323 Gateway ou posterior, em conformidade com a versão 3 do H.323, a largura de banda relatada é bidirecional. Inicialmente, são reservados 128 kb. Se os endpoints na chamada selecionarem um codec mais eficiente, o Cisco Gatekeeper será notificado sobre a alteração da largura de banda.

Observação: Configure o Cisco H.323 Gateway com este comando no modo de configuração

global para usar o comportamento de largura de banda relatado usado antes do Cisco IOS Software Release 12.2(2)XA para gerenciamento de largura de banda de zona:

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

Examples

Esta seção aborda estes dois exemplos:

- [Gerenciamento de largura de banda em uma topologia de cluster](#)
- [Usar BRQ para relatar a largura de banda](#)

Exemplo 1: Gerenciamento de largura de banda em uma topologia de cluster

Veja as depurações capturadas de um Cisco Gatekeeper em um cluster. A depuração mostra as mensagens ARQ e ACF, que incluem a largura de banda necessária para a chamada. Depois de receber essas mensagens, o Cisco Gatekeeper atualiza os outros gatekeepers no cluster sobre essa alteração de largura de banda.

Observação: estes comandos são usados para capturar esta saída: `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

```

Exemplo 2: Usar BRQ para relatar a largura de banda

Encontre depurações de um Cisco Gatekeeper em uma configuração onde a largura de banda no gatekeeper remoto é limitada a 144 kbps. Você vê na depuração que o ARQ solicitado é uma largura de banda inicial de 128 kbps. Quando a chamada for configurada, o ponto final relata a alteração na largura de banda com uma mensagem de BRQ e a largura de banda usada é de 16 kbps, o que significa que a chamada foi configurada com o codec Cisco G729. Em seguida, outra chamada é solicitada e tratada da mesma forma.

Observe que se a segunda chamada chegou antes do ponto final solicitar a alteração na largura de banda para a primeira chamada, o Cisco Gatekeeper rejeitará essa chamada, pois $128+128=256$ kbps e isso é mais de 144 kbps configurado.

```

!
!
!
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 saída foi capturada com os comandos `debug h225 asn1` e `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 ::=
{
```

```
t1 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:
```

[Informações Relacionadas](#)

- [Controle de admissão de chamada VoIP](#)
- [Gatekeeper de alto desempenho Cisco](#)
- [Aprimoramentos de escalabilidade e interoperabilidade do Cisco H.323](#)
- [VoIP com Gatekeeper](#)
- [Suporte à Tecnologia de Voz](#)
- [Suporte aos produtos de Voz e Comunicações Unificadas](#)
- [Troubleshooting da Telefonia IP Cisco](#)
- [Suporte Técnico e Documentação - Cisco Systems](#)