

Risoluzione dei problemi e informazioni sulla gestione della larghezza di banda del Gatekeeper Cisco

Sommario

[Introduzione](#)

[Panoramica delle operazioni di gestione della larghezza di banda](#)

[Come configurare la funzione di gestione della larghezza di banda sul Gatekeeper Cisco](#)

[Gatekeeper show Comandi utilizzati per visualizzare le informazioni sulla larghezza di banda](#)

[Messaggi RAS correlati alla larghezza di banda \(BRQ/BCF/BRJ\)](#)

[Messaggi RAS utilizzati per segnalare lo stato della larghezza di banda](#)

[Modalità di attivazione di BRQ dal gateway per notificare al Gatekeeper di ridurre la larghezza di banda delle chiamate](#)

[Esempi](#)

[Informazioni correlate](#)

Introduzione

In questo documento si presume che il lettore abbia familiarità con le nozioni di base dei gatekeeper software Cisco IOS® e dei messaggi di registrazione, ammissione e stato (RAS) da Gatekeeper a gateway H.225. per ulteriori informazioni, fare riferimento a [Descrizione dei gatekeeper H.323](#).

In base alla raccomandazione H.323, i Gatekeeper Cisco IOS devono supportare i seguenti messaggi di gestione della larghezza di banda RAS H.225:

- Richiesta larghezza di banda (BRQ)
- Rifiuto larghezza di banda (BRJ)
- Messaggi BCF (Bandwidth Confirmation)

Questo concetto può essere basato sulla gestione della larghezza di banda. Può inoltre essere una funzione null che accetta tutte le richieste di modifica della larghezza di banda. In altre parole, il Gatekeeper può utilizzare questi messaggi per gestire la larghezza di banda se consente o rifiuta le richieste o semplicemente ignorarle.

Panoramica delle operazioni di gestione della larghezza di banda

Il Gatekeeper Cisco può rifiutare le chiamate da un terminale a causa dei limiti della larghezza di banda. Questa situazione può verificarsi se il Gatekeeper determina che la larghezza di banda disponibile sulla rete non è sufficiente per supportare la chiamata. Questa funzione è operativa anche durante una chiamata attiva quando un terminale richiede una larghezza di banda aggiuntiva o segnala una modifica nell'ampiezza di banda utilizzata per la chiamata.

Il Gatekeeper Cisco conserva un record di tutte le chiamate attive in modo da poter gestire le risorse di larghezza di banda nella sua zona. In una configurazione cluster, il messaggio di indicazione dell'annuncio del Gatekeeper Update Protocol (GUP) viene scambiato ogni intervallo di tempo specificato e contiene informazioni sull'utilizzo della larghezza di banda per la zona. Questo scambio di messaggi GUP consente ai Gatekeeper alternativi di gestire correttamente la larghezza di banda per una singola zona, anche se i Gatekeeper si trovano in dispositivi fisici separati.

Quando si decide se la larghezza di banda è sufficiente per accettare una richiesta di ammissione di chiamata (ARQ), Cisco Gatekeeper calcola la larghezza di banda disponibile con questa formula:

$$\text{Available_bandwidth} = (\text{total_allocation_bandwidth}) - (\text{bandwidth_used_localmente}) - (\text{bandwidth_used_by_all_alternates}).$$

Se la larghezza di banda disponibile è sufficiente per la chiamata, viene restituita una conferma di ammissione (ACF), altrimenti viene restituito un rifiuto di ammissione (ARJ).

I gateway voce devono prendere in considerazione il codec, l'incapsulamento di layer 2 e le funzionalità di compressione, ad esempio il RTP compresso [cRTP], quando richiedono la larghezza di banda al Gatekeeper Cisco. A volte queste funzioni non sono definite al momento della configurazione della chiamata, nel qual caso è possibile inviare una richiesta di modifica della larghezza di banda al gatekeeper dopo la configurazione della chiamata per regolare la quantità di larghezza di banda utilizzata dalla chiamata.

Nota: a partire dal software Cisco IOS versione 12.2(2)XA, Cisco ha implementato solo la funzionalità del report di tutte le modifiche della larghezza di banda quando i codec vengono modificati. Vedere la sezione: [Modalità di attivazione di BRQ dal gateway per notificare al Gatekeeper di ridurre la larghezza di banda delle chiamate](#) per ulteriori informazioni.

[Come configurare la funzione di gestione della larghezza di banda sul Gatekeeper Cisco](#)

A partire dal software Cisco IOS versione 12.3(1), questi tipi di limitazioni della larghezza di banda delle zone possono essere configurati sul Gatekeeper Cisco:

- Larghezza di banda massima per tutto il traffico H.323 tra la zona locale e una zona remota specificata. Se lo si desidera, questa configurazione può essere ripetuta singolarmente per ciascuna zona remota.
- Larghezza di banda massima consentita per una singola sessione nella zona locale, generalmente utilizzata per applicazioni video e non per la voce
- Larghezza di banda massima per tutto il traffico H.323 consentita collettivamente a tutte le zone remote
- Il nuovo comando **bandwidth check-destination** controlla la larghezza di banda dell'endpoint di destinazione prima che risponda ad ARQ. Questo comando è stato introdotto nel software Cisco IOS versione 12.3(1).

Utilizzare questi comandi per configurare la larghezza di banda della zona del Gatekeeper Cisco:

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


```

=====
LOCAL GK NAME ALT GK NAME  PRI  TOT BW  INT BW  REM BW  LAST  ALT GK
-----  -----  ---  (kbps) (kbps)  (kbps)  ANNOUNCE  STATUS
gkb-1      gkb-2      0  0      0      0      22s      CONNECTED

```

Immettere il comando **show gatekeeper call** per visualizzare le chiamate attive consentite dal gatekeeper e la quantità di larghezza di banda utilizzata da ciascuno di essi.

```

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

```

[Messaggi RAS correlati alla larghezza di banda \(BRQ/BCF/BRJ\)](#)

Il messaggio BRQ viene usato per richiedere una modifica nella larghezza di banda al Cisco Gatekeeper. Questa è la procedura:

1. Il Gatekeeper Cisco verifica la richiesta tramite l'identificatore endpoint per individuare l'endpoint nel database di registrazione.
2. Individua il record di chiamata tramite l'utilizzo di callReferenceValue per trovare una chiamata associata all'endpoint con lo stesso callReferenceValue.
3. Se individua il record di chiamata, calcola la modifica nella larghezza di banda, quindi aggiunge o sottrae la larghezza di banda della zona globale, se necessario. Lo stesso avviene per tutte le risorse proxy o gateway in uso.
4. Viene inviato un messaggio BCF o BRJ all'endpoint, che dipende dalla riuscita o meno dell'operazione.

[Messaggi RAS utilizzati per segnalare lo stato della larghezza di banda](#)

Il campo "Dati non standard" di Information Request Response (IRR) contiene anche informazioni sulla larghezza di banda utilizzata corrente su un gateway o proxy.

[Modalità di attivazione di BRQ dal gateway per notificare al Gatekeeper di ridurre la larghezza di banda delle chiamate](#)

Prima del software Cisco IOS versione 12.2(2)XA sul gateway Cisco H.323, le chiamate erano sempre segnalate per richiedere una larghezza di banda di 64 kbps. Questa è la larghezza di banda unidirezionale per un codec Cisco G.711. Se gli endpoint nella chiamata scelgono di utilizzare un codec più efficiente, il problema non viene segnalato al Cisco Gatekeeper. Nel software Cisco IOS versione 12.2(2)XA del gateway Cisco H.323 o successive, conforme alla versione 3, la larghezza di banda rilevata è bidirezionale. Inizialmente, 128 kb sono riservati. Se gli endpoint nella chiamata selezionano un codec più efficiente, il Gatekeeper Cisco viene informato della modifica della larghezza di banda.

Nota: configurare il gateway Cisco H.323 con questo comando in modalità di configurazione globale per utilizzare il comportamento della larghezza di banda segnalato utilizzato prima del software Cisco IOS versione 12.2(2)XA per la gestione della larghezza di banda della zona:

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

Esempi

In questa sezione vengono illustrati due esempi:

- [Gestione della larghezza di banda in una topologia cluster](#)
- [Utilizzo di BRQ per il report della larghezza di banda](#)

Esempio 1: Gestione della larghezza di banda in una topologia cluster

Vedere i debug acquisiti da un Gatekeeper Cisco in un cluster. Il comando debug visualizza i messaggi ARQ e ACF, che includono la larghezza di banda richiesta per la chiamata. Dopo aver ricevuto questi messaggi, Cisco Gatekeeper aggiorna gli altri gatekeeper del cluster in merito a questa modifica della larghezza di banda.

Nota: per acquisire l'output, vengono utilizzati i seguenti comandi: **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 }
willSupplyUIEs 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

```

Esempio 2: Utilizzo di BRQ per il report della larghezza di banda

Trova i debug da un Gatekeeper Cisco in una configurazione in cui la larghezza di banda sul gatekeeper remoto è limitata a 144 kbps. Come si vede nel debug, la richiesta ARQ ha una larghezza di banda iniziale di 128 kbps. Quando la chiamata è configurata, l'endpoint segnala la modifica nella larghezza di banda con un messaggio BRQ e la larghezza di banda utilizzata in 16 kbps, ossia la chiamata è stata configurata con il codec Cisco G729. Viene quindi richiesta un'altra chiamata che viene trattata allo stesso modo.

Se la seconda chiamata è arrivata prima che l'endpoint richiedesse la modifica della larghezza di banda per la prima chiamata, il Gatekeeper Cisco rifiuta tale chiamata, da $128+128=256$ kbps, ossia da più di 144 kbps configurati.

```

!
!
!
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
!

```

Questo output è stato acquisito con i comandi **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 ::=
```

```
{
  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:
```

[Informazioni correlate](#)

- [Controllo dell'ammissione delle chiamate VoIP](#)
- [Cisco High-Performance Gatekeeper](#)
- [Miglioramenti scalabilità e interoperabilità Cisco H.323](#)
- [VoIP con Gatekeeper](#)
- [Supporto alla tecnologia vocale](#)
- [Supporto ai prodotti voce e Unified Communications](#)
- [Risoluzione dei problemi di Cisco IP Telephony](#)
- [Documentazione e supporto tecnico – Cisco Systems](#)