

排除GUP，备选终点和负载均衡故障

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[规则](#)

[定义](#)

[实验室拓扑结构和配置](#)

[实验室拓扑结构和配置](#)

[备选网守](#)

[网守更新协议](#)

[debug 及 show 命令用于网守集群](#)

[从"gkb-1"的调试，当它是加入团星的第一](#)

[从"gkb-2"的调试，当它在"gkb-1"以后加入团星](#)

[当终端用集群中的一个网守注册时的调试](#)

[当一个带有有效呼叫的终端移至备用网守时的调试](#)

[Cisco 网关故障转换到备选网守](#)

[排除故障与备选终点的问题](#)

[验证网守有正确备用端点](#)

[验证网守是否在其LCF或ACF RAS消息包括备选终点](#)

[验证OGW是否设法与替代项联系，万一主要目的地终点发生故障](#)

[排除故障负载平衡](#)

[相关信息](#)

简介

本文设计帮助您排除故障和了解这些Cisco网守功能：

- 关守集群和关守升级协议(GUP)
- 备选终点
- 负载平衡

关于这些功能的所有信息参考的[思科高性能网守](#)包括功能概述，支持的平台，需要的Cisco IOS软件版本和如何配置，监控和维护他们。

先决条件

要求

本文档的读者应具备以下方面的知识：

- 网守功能基础知识。
- VoIP、H.323和注册、接纳和状态(RAS)信令基础知识。

使用的组件

本文档中的信息基于以下软件和硬件版本。

- Cisco IOS软件版本12.3(4)T1
- Cisco网关：Cisco AS5300、Cisco AS5400和Cisco3725
- Cisco网守：Cisco3725和Cisco 2611

本文档中的信息都是基于特定实验室环境中的设备创建的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您是在真实网络上操作，请确保您在使用任何命令前已经了解其潜在影响。

规则

有关文档规则的详细信息，请参阅 [Cisco 技术提示规则](#)。

定义

期限	定义
ARQ	从Cisco H.323终端RAS信息发送的Admission Request (ARQ)到请求接纳建立呼叫的网守。
ACF	从网守(ACF) RAS信息发送的准入确认到确认呼叫的接受的终端。
ARJ	准入拒绝(ARJ)从网守是一个RAS信息到拒绝准入请求的终端。
GCF	从网守RAS信息发送的关守确认(GCF)到确认网守的发现的Cisco H.323终端。
GRQ	从Cisco H.323终端RAS信息发送的关守请求(GRQ)发现网守。
GUP	网守更新协议用于共享在网守在一集群关于他们的终端和激活的呼叫之间的信息。
LCF	从一网守RAS信息发送的Location Confirm (LCF)到确认位置请求(LRQ)并且包括终止终端的IP地址的另一个。
LRJ	从一网守RAS信息发送的位置拒绝(LRJ)到拒绝LRQ的另一个。
LRQ	从一网守RAS信息发送的位置请求到请求一个远程终止的终端的IP地址的另一个。
R	RAS协议允许网守执行注册、终端的接纳和状态检查

A S	。
R C F	从网守RAS信息发送的Registration Confirm (RCF)到确认注册的终端。
R R J	从拒绝注册请求的网守RAS信息发送的注册拒绝 (RRJ)。
R R Q	从终端RAS信息发送的注册请求(RRQ)到请求向它登记的网守。
U R Q	从终端RAS信息发送的注销请求(URQ)到请求未注册与它的网守。

实验室拓扑结构和配置

要解释功能如何运作和如何排除故障，实验室设置用此拓扑建立：

实验室拓扑结构和配置

所有网关和网守基本配置在下表。用不同的案件，在配置上的有些变化是需要的。当这发生时，更改指示。下面配置只包含对网关或网守功能是重要的此实验室的零件。

配置"gw-1"和"gw-2"是几乎类似的(除了IP地址和H.323 ID)。所以，仅_{gw-1}如下所示。

<pre> gwa-1 ! controller E1 3/0 pri-group timeslots 1-2,16 ! interface Ethernet0/0 ip address 172.16.13.15 255.255.255.224 half-duplex h323-gateway voip interface h323-gateway voip id gka-1 ipaddr 172.16.13.35 1718 h323-gateway voip h323-id gwa-1 h323-gateway voip tech-prefix 1# ! voice-port 3/0:15 ! ! dial-peer voice 5336 pots incoming called-number destination-pattern 5336 direct-inward-dial port 3/0:15 prefix 21 ! dial-peer voice 3653 voip incoming called-number destination-pattern 3653 session target ras dtmf-relay h245-alphanumeric codec g711ulaw </pre>

```
!  
gateway  
!  
ntp clock-period 17178794  
ntp server 172.16.13.35  
end
```

gka-1

```
!  
gatekeeper  
  zone local gka-1 domainA.com 172.16.13.35  
  zone remote gkb domainB.com 172.16.13.41 1719  
  zone prefix gkb 36*  
  zone prefix gka-1 53*  
  gw-type-prefix 1#* default-technology  
  no shutdown  
!  
no scheduler  
max-task-timentp master  
!  
end
```

gwb-1

```
!  
controller E1 0  
  clock source line primary  
  ds0-group 0 timeslots 1-2 type r2-digital r2-compelled  
!  
interface Ethernet0  
  ip address 172.16.13.23 255.255.255.224  
  h323-gateway voip interface  
  h323-gateway voip id gkb-1 ipaddr 172.16.13.41 1718  
  h323-gateway voip h323-id gwb-1  
  h323-gateway voip tech-prefix 2#  
!  
dial-peer voice 3653 pots incoming called-number  
  destination-pattern 3653  
  port 0:0  
  prefix 21  
!  
dial-peer voice 5336 voip  
  incoming called-number  
  destination-pattern 5336  
  session target ras  
  dtmf-relay h245-alphanumeric  
  codec g711ulaw  
!  
gateway  
!  
ntp clock-period 17179389  
ntp server 172.16.13.35  
end
```

gwb-3

```
!  
interface Ethernet0/0  
  ip address 172.16.13.42 255.255.255.224  
  half-duplex h323-gateway voip interface  
  h323-gateway voip id gkb-1 ipaddr 172.16.13.41 1718  
  h323-gateway voip  
  h323-id gwb-3  
  h323-gateway voip tech-prefix 1#  
!  
voice-port 3/0/0
```

```
!  
voice-port 3/0/1  
!  
dial-peer voice 3653 pots  
  destination-pattern 3653  
  port 3/0/0  
  prefix 21  
!  
dial-peer voice 5336 voip  
  incoming called-number  
  destination-pattern 5336  
  session target ras  
  dtmf-relay h245-alphanumeric  
  
  codec g711ulaw  
  
gateway ! ntp clock-period 17179181 ntp server  
172.16.13.35  
!  
end
```

gkb-1

```
!  
gatekeeper  
  zone local gkb-1 domainB.com 172.16.13.41  
  zone remote gka-1 domainA.com 172.16.13.35 1719  
  zone cluster local gkb gkb-1  
  element gkb-2 172.16.13.16 1719  
  gw-type-prefix 2#* default-technology  
  no shutdown  
!  
ntp clock-period 17179580  
ntp server 172.16.13.35  
!  
end
```

gkb-2

```
!  
gatekeeper  
  zone local gkb-2 domainB.com 172.16.13.16  
  zone cluster local gkb gkb-2  
  element gkb-1 172.16.13.41 1719  
!  
no shutdown  
!  
ntp clock-period 17179199  
ntp server 172.16.13.35  
!  
end
```

备选网守

在Cisco H.323版本2之前，每个区域只是由单个网守控制的。Cisco H.323版本2引入“备选网守”想法提供网守冗余。实现备选网守功能允许多网守控制一个区域。当终端用网守时注册，带有备选网守列表为终端注册的区域在，并且为使用CLI，哪些替代项指定。如果网守出故障，终端可能使用备选网守为了继续操作。

Alternate Gatekeeper列表提供给Cisco网守通过CLI为每个区域和传送对终端通过RCF (包括轻量)和GRQ消息。此列表在其他消息可能也传送，例如ARJ或URQ，实现一受控的网守关闭。

备选网守得知现有呼叫通过在网关和网守之间的中断请求(IRQ) /Information请求答复(IRR)交换并且记录这些呼叫。

检测其网守的失败的终端能从该失败安全恢复通过使用备选网守将来请求的，包括要求现有呼叫。备选网守在集群必须配置。他们共享在TCP运行关于终端的信息和激活的呼叫使用GUP。

网守更新协议

这是GUP的一些主要步骤和警告。这应该也帮助您排除故障。

- 一旦配置是集群的一部分的网守在线路来，打开细听的流入连接一个TCP端口GUP协议的。
- 然后它通过发送GRQ消息宣布其在线状态定期。默认时期是30秒并且使用[timer cluster-element announce命令](#)网守的CLI是可配置。此GRQ消息包含非标准数据给每备选网守。此非标准数据是指示器对替代项GRQ根本确实不是GRQ，但是相当是" announcement "消息。在GRQ消息里面，网守指示有开放细听GUP协议的端口号。
- 当您接收从新的网守时的一GRQ，在集群的其他网守建立TCP信道到该端口。
- GUP GRQ消息可以是其中一下列信息：announcementIndication、announcementReject、registrationIndication、unregistrationIndication和resourceIndication。
- 通知指示也传播关于带宽利用率的信息区域的。这允许备选网守适当地管理单区域的带宽，即使网守在独立的物理设备。
- 要验证是否备选网守适当地通信，请使用[show gatekeeper zone cluster命令](#)。此命令也报告备选网守的带宽信息。
- 网守假设，备选网守失败(和假设所有以前已分配带宽当前是可用的)，如果网守不收到在六个通知周期的通知消息，或者，如果检测中断TCP连接用网守。使用六个通知周期每30秒，时间是三分钟，等同于对什么假设是呼叫的平均的长度。假设应该然后是相当安全的，带宽被释放了。在三分钟之后，此网守宣称其备选作为下来并且派出更新通知所有其已注册终端没有备选网守。
- 当终端用在集群的一网守注册/时未注册，该网守使用registrationIndication/unregistrationIndication消息更新在该集群的其他网守关于此更改。
- 如果终端报告一个资源变化使用资源可用性指示器(RAI)对网守在集群上，通过使用GUP消息resourceIndication，该网守报告对所有备选网守的变化在该集群上。
- GUP消息为在集群的网守是需要的有关于每个终端的满足的知识在的区域(注册、带宽、激活的呼叫，资源)能解决所有查询本地。
- 当终端从一网守交换到备选，备选需要得知是活跃的在终端的呼叫。当网守发送新注册的时一个RCF，也发送IRQ得到所有呼叫列表在终端的。请注意IRQ不在RCF前到达终端。
- 在集群的网守允许关闭，即使有激活的呼叫，只要有为有激活的呼叫的所有区域定义的备选网守。如果任何区域没有定义的一个激活的呼叫和备选网守，网守拒绝关闭。
- 他们不知道的备选网守接受所有断开请求(DRQs)呼叫的并且通过相应的信息到验证、授权和统计(AAA)和Cisco关守事务处理消息协议(GKTMP)服务器。这发生，当该终端移动向备选网守，当有激活的呼叫时。另外，包含呼叫的呼叫信息以前不知道的IRR信息可能传送。对于那些IRRs，呼叫记录被修建，并且相应地分配带宽。
- 网守创建每备选网守的一个唯一通知指示消息。如果备选网守接收包含网守标识符它不认可(可能发生的消息，如果备选网守是一个区域的一备选)，但是不是别的，该信息忽略。然而，备选网守通过检查那些消息在替代项的配置里检测错误，并且那些错误向用户报告。
- 当地址为远程区域时，是解决的GUP的真的电源认识到。而不是需要对于远程区域发送LRQs(依顺序或疾风)对所有网守，因而增加消息传送在头顶上在广域网链路，它当前需要发送此查询到一个在集群的网守。如果接收从任何一个的拒绝加上[新区域集群远程](#)CLI，它能在网守之间的循环在集群和不尝试发送LRQ到在集群的另一网守。

- 万一网关移动向备选网守，总是设法注册到该网守，除非不发出网守然后gateway命令。当终端的主要网守回到联机时，终端不重新登记对它，除非终端丢失了与备选网守的通信。它继续使用备选网守其呼叫路由工艺路线信息。

debug 及 show 命令用于网守集群

下面的调试展示网守如何能参加集群，并且他们如何共享关于他们的终端的信息。显示命令使用显示如何监控集群。使用的调试是[debug gatekeeper gup asn1](#)和[debug h225 asn1](#)。这根据以上提到的拓扑和配置。

从"gkb-1"的调试，当它是加入团星的第一

```
Mar 1 08:15:08.348: gk_gup_listen(): listening port = 11007 !--- Opens a TCP port (here it is
11007) to listen to GUP messages. Mar 1 08:15:08.348: gk_gup_listen(): listening fd = 0 Mar 1
08:15:38.351: H225 NONSTD OUTGOING PDU ::= value GRQnonStandardInfo ::= !--- The non-standard
data that is in the GRQ. { gupAddress { ip 'AC100D29'H !--- Listening IP address 172.16.13.41.
port 11007 !--- Listening TCP port 11007. } } Mar 1 08:15:38.351: H225 NONSTD OUTGOING ENCODE
BUFFER::= 40 AC100D29 2AFF Mar 1 08:15:38.351: Mar 1 08:15:38.351: RAS OUTGOING PDU ::= value
RasMessage ::= gatekeeperRequest : !--- GRQ with the non-standard is sent out. { requestSeqNum
59 protocolIdentifier { 0 0 8 2250 0 3 } nonStandardData { nonStandardIdentifier h221NonStandard
: { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '40AC100D292AFF'H } rasAddress
ipAddress : { ip 'AC100D29'H port 1719 } endpointType { vendor { vendor { t35CountryCode 181
t35Extension 0 manufacturerCode 18 } } mc FALSE undefinedNode FALSE } } Mar 1 08:15:38.359: RAS
OUTGOING ENCODE BUFFER::= 01 00003A06 0008914A 000340B5 00001207 40AC100D 292AFF00 AC100D29
06B72000 B5000012 00 Mar 1 08:15:38.359:
```

从"gkb-2"的调试，当它在"gkb-1"以后加入团星

```
Mar 1 08:16:38.878: gk_gup_listen(): listening port = 11006 !--- Opens a TCP port (here it is
11006) to listen to GUP messages. Mar 1 08:16:38.878: gk_gup_listen(): listening fd = 0 Mar 1
08:17:08.385: RAS INCOMING ENCODE BUFFER::= 01 00003D06 0008914A 000340B5 00001207 40AC100D
292AFF00 AC100D29 06B72000 B5000012 00 Mar 1 08:17:08.385: Mar 1 08:17:08.385: RAS INCOMING PDU
::= value RasMessage ::= gatekeeperRequest : !--- GRQ message is received from gkb-1 gatekeeper
with non-standard information. { requestSeqNum 62 protocolIdentifier { 0 0 8 2250 0 3 }
nonStandardData { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0
manufacturerCode 18 } data '40AC100D292AFF'H } rasAddress ipAddress : { ip 'AC100D29'H !--- RAS
IP address 172.16.13.41 used gkb-1. port 1719 !--- RAS TCP port used by gkb-1. } endpointType {
vendor { vendor { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } } mc FALSE
undefinedNode FALSE } } Mar 1 08:17:08.393: H225 NONSTD INCOMING ENCODE BUFFER::= 40 AC100D29
2AFF Mar 1 08:17:08.393: Mar 1 08:17:08.393: H225 NONSTD INCOMING PDU ::= value
GRQnonStandardInfo ::= !--- gkb-2 extracts the non-standard data from the GRQ. { gupAddress { ip
'AC100D29'H !--- GUP IP address 172.16.13.41 used by gkb-1. port 11007 !--- GUP TCP port 11007
used by gkb-1. } } Mar 1 08:17:08.393: check_connection: checking connection to
172.16.13.41:11007 Mar 1 08:17:08.393: gk_gup_connect(): initiating connection Mar 1
08:17:08.393: gup_connect: connecting to 172.16.13.41:11007 !--- A GUP connection is
established, and updates follow. Mar 1 08:17:08.393: gup_connect, fd = 1 Mar 1 08:17:08.401: GUP
OUTGOING PDU ::= value GUP_Information ::= !--- GUP announcement is sent to alternate GK gkb-1.
{ protocolIdentifier { 1 2 840 113548 10 0 0 2 } message announcementIndication : {
announcementInterval 30 endpointCapacity 100000 callCapacity 100000 hostName '676B622D32'H
percentMemory 8 !--- Below is information about the status of gkb-2. percentCPU 0 currentCalls 0
currentEndpoints 0 zoneInformation { { gatekeeperIdentifier {"gkb-2"} altGKIdentifier {"gkb-1"}
totalBandwidth 0 interzoneBandwidth 0 remoteBandwidth 0 } } } } Mar 1 08:17:08.405: GUP OUTGOING
ENCODE BUFFER::= 00 0A2A8648 86F70C0A 00000220 001E8001 86A08001 86A00467 6B622D32 10000000
00014200 0067006B 0062002D 00320800 67006B00 62002D00 31000000 000000 Mar 1 08:17:08.409: Mar 1
08:17:08.409: Sending GUP ANNOUNCEMENT INDICATION to 172.16.13.41 Mar 1 08:17:08.413: GUP
INCOMING ENCODE BUFFER::= 00 0A2A8648 86F70C0A 00000220 001E8001 86A08001 86A00467 6B622D31
32000000 00014200 0067006B 0062002D 00310800 67006B00 62002D00 32000000 000000 Mar 1
08:17:08.413: Mar 1 08:17:08.413: GUP INCOMING PDU ::= value GUP_Information ::= !--- GUP
announcement is received from alternate GK gkb-1. { protocolIdentifier { 1 2 840 113548 10 0 0 2
```

```
} message announcementIndication : { announcementInterval 30 endpointCapacity 100000
callCapacity 100000 hostName '676B622D31'H percentMemory 25 !--- Below is information about the
status of gkb-1. percentCPU 0 currentCalls 0 currentEndpoints 0 zoneInformation { {
gatekeeperIdentifier {"gkb-1"} altGKIdentifier {"gkb-2"} totalBandwidth 0 interzoneBandwidth 0
remoteBandwidth 0 } } } } Mar 1 08:17:08.421: Received GUP ANNOUNCEMENT INDICATION from
172.16.13.41
```

用[show gatekeeper endpoint命令](#)，没有已注册终端。输出是如下：

```
gkb-1#show gatekeeper endpoints GATEKEEPER ENDPOINT REGISTRATION
===== CallSignalAddr Port RASignalAddr Port Zone Name Type Flags ---
----- Total number of active
registrations = 0 gkb-2# show gatekeeper endpoints GATEKEEPER ENDPOINT REGISTRATION
===== CallSignalAddr Port RASignalAddr Port Zone Name Type Flags ---
----- Total number of active
registrations = 0 gkb-2#
```

[当终端用集群中的一个网守注册时的调试](#)

此调试从"gkb-1"网守采取。终端是注册对有打开的[debug h225 ans1](#)和[debug ras](#)的"gkb-1"网守的"gw-1"。

```
Mar 1 08:22:47.396: RAS INCOMING ENCODE BUFFER ::= 00 A00AAD06
0008914A 000300AC 100D17E0 7D088001 3C050401 00205002 00006700 6B006200
2D003101 40040067 00770062 002D0031
Mar 1 08:22:47.396:
Mar 1 08:22:47.396: RAS INCOMING PDU ::= value RasMessage ::= gatekeeperRequest : !--- GRQ is
received from "gw-1" gateway. { requestSeqNum 2734 protocolIdentifier { 0 0 8 2250 0 3 }
rasAddress ipAddress : { ip 'AC100D17'H !--- gw-1 IP address (172.16.13.23). port 57469 !---
gw-1 TCP port 57469. } endpointType { gateway { protocol { voice : { supportedPrefixes { {
prefix e164 : "2#" } } } } mc FALSE undefinedNode FALSE } gatekeeperIdentifier {"gkb-1"}
endpointAlias { h323-ID : {"gw-1"} } } Mar 1 08:22:47.404: RAS OUTGOING PDU ::= value
RasMessage ::= gatekeeperConfirm : !--- GCF is sent back with alternate gatekeepers included. {
requestSeqNum 2734 protocolIdentifier { 0 0 8 2250 0 3 } gatekeeperIdentifier {"gkb-1"}
rasAddress ipAddress : { ip 'AC100D29'H !--- Gatekeeper gkb-1 IP address (172.16.13.41). port
1719 } alternateGatekeeper !--- List of alternate gatekeepers, here is "gkb-2" only. { {
rasAddress ipAddress : { ip 'AC100D10'H !--- Alternate gatekeeper gkb-2 IP address
(172.16.13.16) port 1719 } gatekeeperIdentifier {"gkb-2"} needToRegister TRUE priority 0 } } }
Mar 1 08:22:47.412: RAS OUTGOING ENCODE BUFFER ::= 06 800AAD06 0008914A 00030800 67006B00
62002D00 3100AC10 0D2906B7 0D001401 40AC100D 1006B708 0067006B 0062002D 003280 Mar 1
08:22:47.412: Mar 1 08:22:47.432: RAS INCOMING ENCODE BUFFER ::= 0E C00AAE06 0008914A 00038001
00AC100D 1706B801 00AC100D 17E07D08 80013C05 04010020 50000140 04006700 77006200 2D003108
0067006B 0062002D 003100B5 00001212 8B000200 3B010001 000180 Mar 1 08:22:47.432: Mar 1
08:22:47.436: RAS INCOMING PDU ::= value RasMessage ::= registrationRequest : !--- RRQ is
received from "gw-1" gateway. { requestSeqNum 2735 protocolIdentifier { 0 0 8 2250 0 3 }
discoveryComplete TRUE callSignalAddress { ipAddress : { ip 'AC100D17'H !--- Gateway gw-1 IP
address (172.16.13.23). port 1720 } } rasAddress { ipAddress : { ip 'AC100D17'H port 57469 } }
terminalType { gateway { protocol { voice : { supportedPrefixes { { prefix e164 : "2#" } } } }
mc FALSE undefinedNode FALSE } terminalAlias { h323-ID : {"gw-1"} } gatekeeperIdentifier {"gkb-
1"} endpointVendor { vendor { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } }
timeToLive 60 keepAlive FALSE willSupplyUIEs FALSE maintainConnection TRUE } Mar 1
08:22:47.448: GUP OUTGOING PDU ::= value GUP_Information ::= !--- A GUP registration indicates a
message is sent to "gkb-2" to inform it !--- about the new registered endpoint. {
protocolIdentifier { 1 2 840 113548 10 0 0 2 } message registrationIndication : { version 3
callSignalAddress { ipAddress : { ip 'AC100D17'H !--- Gateway gw-1 IP address (172.16.13.23).
port 1720 } } rasAddress { ipAddress : { ip 'AC100D17'H !--- Gateway gw-1 IP address
(172.16.13.23). port 57469 } } terminalType { vendor { vendor { t35CountryCode 181 t35Extension
0 manufacturerCode 18 } } gateway { protocol { voice : { supportedPrefixes { { prefix e164 :
"2#" } } } } mc FALSE undefinedNode FALSE } terminalAlias { h323-ID : {"gw-1"} !--- Name/ID
of the new endpoint which has just registered. } gatekeeperIdentifier {"gkb-1"} !--- Name/ID of
the gatekeeper which the new endpoint(gw-1) has registered to. resourceIndicator {
almostOutOfResources FALSE } } } Mar 1 08:22:47.460: GUP OUTGOING ENCODE BUFFER ::= 00 0A2A8648
86F70C0A 00000232 020100AC 100D1706 B80100AC 100D17E0 7D2800B5 00001240 013C0505 01004050
```



```
10000140 04006700 77006200 2D003108 0067006B 0062002D 003100 Mar 1 08:22:47.464: Mar 1
08:22:47.464: Sending GUP REGISTRATION INDICATION to 172.16.13.16 Mar 1 08:22:47.464: RAS
OUTGOING PDU ::= value RasMessage ::= registrationConfirm : !--- RCF is sent back to "gwb-1"
gateway. { requestSeqNum 2735 protocolIdentifier { 0 0 8 2250 0 3 } callSignalAddress { }
terminalAlias { h323-ID : {"gwb-1"} } gatekeeperIdentifier {"gkb-1"} endpointIdentifier
{"61809DB800000001"} alternateGatekeeper { { rasAddress ipAddress : { ip 'AC100D10'H port 1719 }
gatekeeperIdentifier {"gkb-2"} needToRegister TRUE priority 0 } } timeToLive 60 willRespondToIRR
FALSE maintainConnection TRUE } Mar 1 08:22:47.472: RAS OUTGOING ENCODE BUFFER::= 12 C00AAE06
0008914A 00030001 40040067 00770062 002D0031 08006700 6B006200 2D00311E 00360031 00380030
00390044 00420038 00300030 00300030 00300030 00300031 0F8A1401 40AC100D 1006B708 0067006B
0062002D 00328002 003B0100 0180 Mar 1 08:22:47.472:
```

在所有网关在集群后，注册以上输出包含[show gatekeeper endpoint命令](#)输出。上述调试为每终端注册发生。在集群的全部三个网关注册后，[show gatekeeper endpoint命令](#)在两网守如下：

```
gkb-1#show gatekeeper endpoints GATEKEEPER ENDPOINT REGISTRATION
===== CallSignalAddr Port RASignalAddr Port Zone Name Type Flags ---
-----
57469 gkb-1 VOIP-GW H323-ID: gwb-1 172.16.13.26 1720 172.16.13.26 49801 gkb-1 VOIP-GW H323-ID:
gwb-2 172.16.13.42 1720 172.16.13.42 57216 gkb-1 VOIP-GW A !--- A flag set. H323-ID: gwb-3 Total
number of active registrations = 3 gkb-2# show gatekeeper endpoints GATEKEEPER ENDPOINT
REGISTRATION ===== CallSignalAddr Port RASignalAddr Port Zone Name
Type Flags -----
172.16.13.23 57469 gkb-2 VOIP-GW A !--- A flag set. H323-ID: gwb-1 172.16.13.26 1720
172.16.13.26 49801 gkb-2 VOIP-GW A !--- A flag set. H323-ID: gwb-2 172.16.13.42 1720
172.16.13.42 57216 gkb-2 VOIP-GW H323-ID: gwb-3 Total number of active registrations = 3 !---
The "A" under the flag field means that the gatekeeper is an alternate one !--- for this
endpoint.
```

当一个带有有效呼叫的终端移至备用网守时的调试

这些是从启动的网守的调试，当呼叫请求时并且是，直到被断开。某些多余的调试消息省略。这些调试是从"gkb-1"网守。gwa-1安置呼叫注册到"gka-1"对另一个网关(gwb-1)在远程区域集群。调试展示激活的呼叫流如何从备选网守的主要关守被跟踪，当主要的断开。

```
Mar 2 23:59:26.714: RecvUDP_IPSockData successfully rcvd message of length 84
from 172.16.13.35:1719
Mar 2 23:59:26.714: RAS INCOMING ENCODE BUFFER::= 4A 80080801 01806986
40B50000 122C8286 B01100C8 C66C7D1
6 8011CC80 0D882828 5B8DF601 80140204 8073B85A 5C564004 00670077
0061002D 003100AC 100D2306 B70B800D 01400
400 67006B00 61002D00 310180
Mar 2 23:59:26.714:
Mar 2 23:59:26.714: RAS INCOMING PDU ::=
```

```
value RasMessage ::= locationRequest : !--- LRQ is received from "gka-1" gatekeeper from domain
A. { requestSeqNum 2057 destinationInfo { e164 : "3653" !--- E164 number to be resolved by the
this gatekeeper. } nonStandardData { nonStandardIdentifier h221NonStandard : { t35CountryCode
181 t35Extension 0 manufacturerCode 18 } data '8286B01100C8C66C7D168011CC800D8828285B8D...'H }
replyAddress ipAddress : { ip 'AC100D23'H port 1719 } sourceInfo { h323-ID : {"gka-1"} }
canMapAlias TRUE } Mar 2 23:59:26.722: LRQ (seq# 2057) rcvd Mar 2 23:59:26.722: H225 NONSTD
INCOMING ENCODE BUFFER::= 82 86B01100 C8C66C7D 168011CC 800D8828 285B8DF6 01801402 048073B8
5A5C5640 04006700 77006100 2D0031 Mar 2 23:59:26.722: Mar 2 23:59:26.722: H225 NONSTD INCOMING
PDU ::= !--- LRQ nonStandardInfo decoded output. value LRQnonStandardInfo ::= { ttl 6 nonstd-
callIdentifier { guid 'C8C66C7D168011CC800D8828285B8DF6'H } callingOctet3a 128 gatewaySrcInfo {
e164 : "4085272923", h323-ID : {"gwa-1"} } } parse_lrq_nonstd: LRQ Nonstd decode succeeded,
remlen = 84 Mar 2 23:59:26.726: H225 NONSTD OUTGOING PDU ::= !--- LCF nonStandardInfo reply back
to the LRQ nonStandardInfor. value LCFnonStandardInfo ::= { termAlias { h323-ID : {"gwb-1"} }
gkID {"gkb-1"} gateways { { gwType voip : NULL gwAlias { h323-ID : {"gwb-1"} !--- Gateway gwb-1
is the resolved terminating gateway sent back for the request. } sigAddress { ip 'AC100D17'H !---
Gateway gwb-1 IP address (172.16.13.23). port 1720 } resources { maxDSPs 0 inUseDSPs 0
maxBChannels 0 inUseBChannels 0 activeCalls 0 bandwidth 0 inuseBandwidth 0 } } } } Mar 2
23:59:26.734: H225 NONSTD OUTGOING ENCODE BUFFER::= 00 01400400 67007700 62002D00 31080067
```

```
006B0062 002D0031 01100140 04006700 77006200 2D003100 AC100D17 06B80000 00000000 00000000 Mar 2
23:59:26.734: Mar 2 23:59:26.734: RAS OUTGOING PDU ::= value RasMessage ::= locationConfirm : !-
-- LCF is sent back with "gwb-1" as the resolved terminating gateway. { requestSeqNum 2057
callSignalAddress ipAddress : { ip 'AC100D17'H !--- Resolved terminating gateway gwb-1 IP
address (172.16.13.23). port 1720 } rasAddress ipAddress : { ip 'AC100D17'H !--- Resolved
terminating gateway gwb-1 IP address (172.16.13.23). port 51874 } nonStandardData {
nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18
} data '00014004006700770062002D0031080067006B00...'H } destinationType { gateway { protocol {
voice : { supportedPrefixes { } } } } mc FALSE undefinedNode FALSE } } Mar 2 23:59:26.742: RAS
OUTGOING ENCODE BUFFER ::= 4F 080800AC 100D1706 B800AC10 0D17CAA2 40B50000 1239000 1 40040067
00770062 002D0031 08006700 6B006200 2D003101 10014004 00670077 0062002D 003100AC 100D1706 B8000
000 00000000 00000010 40080880 013C0501 0000 Mar 2 23:59:26.746: Mar 2 23:59:26.746:
IPSOCK_RAS_sendto: msg length 91 from 172.16.13.41:1719 to 172.16.13.35: 1719 Mar 2
23:59:26.746: RASLib::RASSendLCF: LCF (seq# 2057) sent to 172.16.13.35 Mar 2 23:59:26.798:
RecvUDP_IPSockData successfully rcvd message of length 129 from 172.16.13.23:51874 Mar 2
23:59:26.798: RAS INCOMING ENCODE BUFFER ::= 27 98172700 F0003600 31003900 36003200 39003600
3800300 0 30003000 30003000 30003000 31010180 69860204 8073B85A 5C564004 00670077 0061002D
003100AC 100D0F2A FA400 500 000E40B5 00001207 80000008 800180C8 C66C7D16 8011CC80 0C882828
5B8DF645 60200180 1100C8C6 6C7D1680 11C C800D 8828285B 8DF60100 Mar 2 23:59:26.802: Mar 2
23:59:26.802: RAS INCOMING PDU ::= value RasMessage ::= admissionRequest : !--- "gwb-1" sent
answerCall ARQ. { requestSeqNum 5928 callType pointToPoint : NULL callModel direct : NULL
endpointIdentifier {"6196296800000001"} destinationInfo { e164 : "3653" !--- E164 number the
caller is trying to reach. } srcInfo { e164 : "4085272923", !--- Caller information. h323-ID :
{"gwa-1"} } srcCallSignalAddress ipAddress : { ip 'AC100D0F'H !--- Originating gateway (gwa-1)
IP address and port. port 11002 } bandWidth 1280 callReferenceValue 14 !--- Remember call
reference, since it is used when the call !--- is disconnected when sending the DRQ.
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 to "gwb-1". { requestSeqNum 5928 bandWidth 1280
callModel direct : NULL destCallSignalAddress ipAddress : { ip 'AC100D17'H !--- gwb-1 IP address
(172.16.13.23). 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 is sent out and it contains the
information !--- about the last call that is still active. { 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 !--- 1280 is 128 Kbps of total bandwidth used for the zone.
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
```

注意：这时"gkb-1"被关闭。这允许(即使有一个激活的呼叫)，因为有该区域的一备选网守。

URQ信息传送对所有终端注册与"gkb-1"。这些终端是"gwb-1"和"gwb-2"网关。这些网关通过发送到UCFs确认URQ。并且，gkb-1传送GUP unregistration征兆信息给集群备选网守然后断开GUP连接。

```
Mar 2 23:59:55.914: RAS OUTGOING PDU ::= value RasMessage ::= unregistrationRequest : {
requestSeqNum 79 callSignalAddress { ipAddress : { ip 'AC100D17'H !--- UnregistrationRequest
(URQ) sent to gwb-1 (172.16.13.23). port 1720 } } } Mar 2 23:59:55.914: RAS OUTGOING ENCODE
BUFFER ::= 18 00004E01 00AC100D 1706B8 Mar 2 23:59:55.914: Mar 2 23:59:55.914: IPSOCK_RAS_sendto:
```

```

msg length 12 from 172.16.13.41:1719 to 172.16.13.23: 51874 Mar 2 23:59:55.914:
RASLib::RASsendURQ: URQ (seq# 79) sent to 172.16.13.23 Mar 2 23:59:55.918: RAS OUTGOING PDU ::=
value RasMessage ::= unregistrationRequest : { requestSeqNum 80 callSignalAddress { ipAddress :
{ ip 'AC100D1A'H !--- URQ sent to gw-2 (172.16.13.26). port 1720 } } } Mar 2 23:59:55.918: RAS
OUTGOING ENCODE BUFFER::= 18 00004F01 00AC100D 1A06B8 Mar 2 23:59:55.918: Mar 2 23:59:55.918:
IPSOCK_RAS_sendto: msg length 12 from 172.16.13.41:1719 to 172.16.13.26: 50041 Mar 2
23:59:55.918: RASLib::RASsendURQ: URQ (seq# 80) sent to 172.16.13.26 Mar 2 23:59:55.922:
RecvUDP_IPSockData successfully rcvd message of length 3 from 172.16.13.23:51874 Mar 2
23:59:55.922: RAS INCOMING ENCODE BUFFER::= 1C 004E Mar 2 23:59:55.922: Mar 2 23:59:55.922: RAS
INCOMING PDU ::= value RasMessage ::= unregistrationConfirm : { requestSeqNum 79 } Mar 2
23:59:55.922: UCF (seq# 79) rcvd Mar 2 23:59:55.926: RecvUDP_IPSockData successfully rcvd
message of length 3 from 172.16.13.26:50041 Mar 2 23:59:55.926: RAS INCOMING ENCODE BUFFER::= 1C
004F Mar 2 23:59:55.926: Mar 2 23:59:55.926: RAS INCOMING PDU ::= value RasMessage ::=
unregistrationConfirm : { requestSeqNum 80 } Mar 2 23:59:55.926: UCF (seq# 80) rcvd Mar 3
00:00:01.922: GUP OUTGOING PDU ::= value GUP_Information ::= { protocolIdentifier { 1 2 840
113548 10 0 0 2 } message unregistrationIndication : { reason explicitUnregister : NULL
callSignalAddress { ipAddress : { ip 'AC100D17'H !--- GUP UnregistrationIndication sent to
alternate gatekeeper !--- gkb-2 (172.16.13.16) in the cluster. port 1720 } } } } Mar 3
00:00:01.922: GUP OUTGOING ENCODE BUFFER::= 00 0A2A8648 86F70C0A 00000238 000100AC 100D1706 B8
Mar 3 00:00:01.926: Mar 3 00:00:01.926: Sending GUP UNREGISTRATION INDICATION to 172.16.13.16
Mar 3 00:00:01.934: gk_gup_close_connection(): closing connection to 172.16.13.16 Mar 3
00:00:01.934: gk_gup_close_listen(): closing listen

```

这是从"gkb-2"的调试。显示移动的终端"gw-1"和"gw-2"注册省略的调试，因为他们看起来象正常注册。当移动向"gkb-2"时，此处目的将显示激活的呼叫的DRQ的接受在"gw-1"的。

```

Mar 3 00:00:24.307: RecvUDP_IPSockData successfully rcvd message of length 77
from 172.16.13.23:51874
Mar 3 00:00:24.307: RAS INCOMING ENCODE BUFFER::= 3E 172C1E00 36003100
38003400 44004300 34004300 30003000 30003000 3
0003000 300033C8 C66C7D16 8011CC80 0C882828 5B8DF600 0E21A100 1100C8C6
6C7D1680 11CC800D 8828285B 8DF60180
Mar 3 00:00:24.311:
Mar 3 00:00:24.311: RAS INCOMING PDU ::=

```

```

value RasMessage ::= disengageRequest : !--- DRQ is received with call reference 14 and normal
clearing !--- disconnect cause code. !--- This information is passed to the accounting server
and the GKTMP !--- server if configured. { requestSeqNum 5933 endpointIdentifier
{"6184DC4C00000003"} conferenceID 'C8C66C7D168011CC800C8828285B8DF6'H callReferenceValue 14
disengageReason normalDrop : NULL callIdentifier { guid 'C8C66C7D168011CC800D8828285B8DF6'H }
answeredCall TRUE } Mar 3 00:00:24.311: DRQ (seq# 5933) rcvd Mar 3 00:00:24.315: RAS OUTGOING
PDU ::= value RasMessage ::= disengageConfirm : !--- DCF is sent to "gw-1". { requestSeqNum
5933 } Mar 3 00:00:24.315: RAS OUTGOING ENCODE BUFFER::= 40 172C Mar 3 00:00:24.315: Mar 3
00:00:24.315: IPSOCK_RAS_sendto: msg length 3 from 172.16.13.16:1719 to 172.16.13.23: 51874 Mar
3 00:00:24.315: RASLib::RASsendDCF: DCF (seq# 5933) sent to 172.16.13.23 gkb-2#

```

Cisco 网关故障转换到备选网守

默认情况下，Cisco网关发送一轻量级RRQ每45秒。万一网守没有发送任何URQ到网关(例如由于一个残破的路由问题)，两次网关(在听不到一个RCF或RRJ其轻量级RRQ的)尝试与五秒在其中每一之间。如果第三尝试失败，立即网守把死者视为并且向使用RRQ的备选网守登记。在网关开始最初的注册过程用网守的方案中，它派出GRQ找出网守IP地址。如果有GCF回复上一步，网关发送RRQ给主要关守指定。如果网守因故拒绝注册请求，网关不设法与其备选网守联系。它再开始此进程(GRQ、GCF和RRQ)与主要关守。

网关只与代替网守联系，当对主要关守的连接丢失时，并且没有回复上一步。如果主要关守不回复回到GRQ消息，当网关首先派出发现网守，则，在三失败的尝试(大约每尝试五分钟)后，网关与代替网守联系。在主要关守断开的情况下，在网关注册与它后，网关丢失从主要关守的保活信息。在未命中三个连续的保活信息以后，网关宣称主要关守作为下来，并且再开始注册过程。

排除故障与备选终点的问题

一个呼叫的终端能从呼叫建立故障恢复通过发送设置信息到其中一备选终点。呼叫可以由于许多原因发生故障：网关发生故障，并且网守不知道它在发送ACF或LCF时，没有在网关的资源，并且没有那向网守报告，呼叫失效由于在主要终端的更一个不正确的配置和。

注意：始发端只设法与备选网守联系，如果呼叫发生故障，在提醒的阶段前(警报或进度)。如果呼叫发生故障由于用户忙或没有答案，始发端不尝试任何其他替代项。

网守得知某一终端的该备选由手动配置使用gatekeeper cli命令[endpoint alt-ep](#)或从所有已接收RAS消息。思科支持最多每个终端的20替代项，无论网守学习他们。

您需要看一看的问题包括：

- 如果网守有如期望的一样正确备用端点。
- 如果网守在其LCF或ACF RAS消息包括备选终点。
- 如果OGW设法与替代项联系，万一主要目的地终点发生故障。

显示如何排除故障这些问题，拓扑和一样以上与在"gkb-1"网守配置的此更改一起使用包括两个备选网关："gwb-3"和"gw-1"网关的"gw-2"。这是"gkb-1"网守的配置：

```
!
gatekeeper
zone local gkb-1 domainB.com 172.16.13.41
zone remote gka-1 domainA.com 172.16.13.35 1719
zone cluster local gkb gkb-1
  element gkb-2 172.16.13.16 1719
!
gw-type-prefix 2#* default-technology
bandwidth total zone gkb-1 512
bandwidth session zone gkb-1 512
no shutdown
endpoint alt-ep h323id gwb-2 172.16.13.42 !--- 172.16.13.42 is gwb-3. endpoint alt-ep h323id
gwb-2 172.16.13.23 !--- 172.16.13.23 is gw-1. !
```

验证网守有正确备用端点

要看到网守是否有正确的备选终点，请使用[show gatekeeper endpoints alternates](#)命令。

```
gkb-1#show gatekeeper endpoints alternates GATEKEEPER ENDPOINT REGISTRATION
===== CallSignalAddr Port RASSignalAddr Port Zone Name Type Flags -
-----
172.16.13.23 1720 172.16.13.23 54670 gkb-1 VOIP-GW H323-ID: gwb-1 172.16.13.26 1720 172.16.13.26
57233 gkb-1 VOIP-GW H323-ID: gwb-2 ALT_EP: 172.16.13.42 <1720> 172.16.13.23 <1720> !--- This
shows the information about all collected endpoints. 172.16.13.42 1720 172.16.13.42 58430 gkb-1
VOIP-GW A H323-ID: gwb-3 Total number of active registrations = 3 ALL CONFIGURED ALTERNATE
ENDPOINTS !--- Only manually configured. ===== Endpoint H323
Id RASSignalAddr Port -----
1720 gwb-2 172.16.13.23 1720 gkb-1#
```

验证网守是否在其LCF或ACF RAS消息包括备选终点

要看到网守是否发送备选终点的IP地址，您能打开debug h225 asn1并且查看ACF消息或LCF。这是从"gkb-1"采取的示例调试。

```
Mar 3 04:12:47.676: H225 NONSTD OUTGOING ENCODE BUFFER::= 00 01400400
67007700 62002D00 32080067 00
6B0062 002D0031 01100140 04006700 77006200 2D003200 AC100D1A 06B80000
```

00000000 00000000

Mar 3 04:12:47.676:

Mar 3 04:12:47.676: RAS OUTGOING PDU ::=

```
value RasMessage ::= locationConfirm : { requestSeqNum 2070 callSignalAddress ipAddress : { ip
'AC100D1A'H !--- This is IP address of main destination. port 1720 } rasAddress ipAddress : { ip
'AC100D1A'H port 50041 } nonStandardData { nonStandardIdentifier h221NonStandard : {
t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data
'00014004006700770062002D0032080067006B00...'H } destinationType { gateway { protocol { voice :
{ supportedPrefixes { } } } } mc FALSE undefinedNode FALSE } alternateEndpoints !--- Alternate
endpoints. { { callSignalAddress { ipAddress : { ip 'AC100D2A'H !--- This is the first alternate
IP address (172.16.13.42 gw-3). port 1720 }, ipAddress : { ip 'AC100D17'H !--- This is the
second alternate IP address (172.16.13.23 gw-1). port 1720 } } } }
```

验证OGW是否设法与替代项联系，万一主要目的地终点发生故障

此部分显示OGW如何起反应，当接收其ACF消息的备选终点。在本例中呼叫被做发生故障，当尝试与主要终止的终端时(千兆瓦)联系。打开的调试这是[debug voip ccapi inout](#)和[debug h225 asn1](#)。

您在调试看的第一事物是显示产生的电话段的ccapi消息。

```
Mar 3 04:12:47.616: cc_api_call_setup_ind (vdbPtr=0x6264A60C,
callInfo={called=3653,called_oct3=0x8
0,calling=4085272923,calling_oct3=0x21,calling_oct3a=0x80,calling_xlated=false,subsc
riber_type_str=R egularLine,fdest=1,peer_tag=5336, prog_ind=0},callID=0x62155454) Mar 3
04:12:47.616: cc_api_call_setup_ind type 13 , prot 0 Mar 3 04:12:47.620:
cc_process_call_setup_ind (event=0x6231C454) Mar 3 04:12:47.620: >>>CCAPI handed cid 51 with
tag 5336 to app "DEFAULT" Mar 3 04:12:47.620: sess_appl: ev(24=CC_EV_CALL_SETUP_IND), cid(51),
disp(0) Mar 3 04:12:47.620: sess_appl: ev(SSA_EV_CALL_SETUP_IND), cid(51), disp(0) Mar 3
04:12:47.620: ssaCallSetupInd Mar 3 04:12:47.620: ccCallSetContext (callID=0x33,
context=0x626EAC9C) Mar 3 04:12:47.620: ssaCallSetupInd cid(51), st(SSA_CS_MAPPING),oldst(0),
ev(24)ev->e.evCallSetupIn.d.nCallInfo.finalDestFlag = 1 Mar 3 04:12:47.620: ssaCallSetupInd
finalDest cllng(4085272923), cllcd(3653) Mar 3 04:12:47.620: ssaCallSetupInd cid(51),
st(SSA_CS_CALL_SETTING),oldst(0), ev(24)dpMatchPeersMo reArg result= 0 Mar 3 04:12:47.620:
ssaSetupPeer cid(51) peer list: tag(3653) called number (3653) Mar 3 04:12:47.620: ssaSetupPeer
cid(51), destPat(3653), matched(4), prefix(), peer(62663E7C), peer ->encapType (2) Mar 3
04:12:47.620: ccCallProceeding (callID=0x33, prog_ind=0x0) Mar 3 04:12:47.620:
ccCallSetupRequest (Inbound call = 0x33, outbound peer =3653, dest=, params=0x62327730 mode=0,
*callID=0x62327A98, prog_ind = 0) Mar 3 04:12:47.624: ccCallSetupRequest numbering_type 0x80 Mar
3 04:12:47.624: ccCallSetupRequest encapType 2 clid_restrict_disable 1 null_orig_clg 0 clid_tra
nsparent 0 callingNumber 4085272923 Mar 3 04:12:47.624: dest pattern 3653, called 3653,
digit_strip 0 Mar 3 04:12:47.624: callingNumber=4085272923, calledNumber=3653, redirectNumber=
display_info= call ing_oct3a=80 Mar 3 04:12:47.624: accountNumber=, finalDestFlag=1,
guid=2d3a.ac33.16a4.11cc.8068.8828.285b.8df6 Mar 3 04:12:47.624: peer_tag=3653 Mar 3
04:12:47.624: ccIFCallSetupRequestPrivate: (vdbPtr=0x621B2360, dest=, callParams={called=3653
,called_oct3=0x80, calling=4085272923,calling_oct3=0x21, calling_xlated=false,
subscriber_type_str= RegularLine, fdest=1, voice_peer_tag=3653},mode=0x0) vdbPtr type = 1 !---
The OGW establishes the second leg. Mar 3 04:12:47.624: ccIFCallSetupRequestPrivate:
(vdbPtr=0x621B2360, dest=, callParams={called=3653 , called_oct3 0x80,
calling=4085272923,calling_oct3 0x21, calling_xlated=false, fdest=1, voice_pee r_tag=3653},
mode=0x0, xltrc=-5) Mar 3 04:12:47.624: ccSaveDialpeerTag (callID=0x33, dialpeer_tag=0xE45) Mar
3 04:12:47.624: ccCallSetContext (callID=0x34, context=0x626EB9A4) Mar 3 04:12:47.624:
ccCallReportDigits (callID=0x33, enable=0x0) Mar 3 04:12:47.624: cc_api_call_report_digits_done
(vdbPtr=0x6264A60C, callID=0x33, disp=0) Mar 3 04:12:47.624: sess_appl:
ev(52=CC_EV_CALL_REPORT_DIGITS_DONE), cid(51), disp(0) Mar 3 04:12:47.624:
cid(51)st(SSA_CS_CALL_SETTING)ev(SSA_EV_CALL_REPORT_DIGITS_DONE) oldst(SSA_CS_MAPPING)cfid(-
1)csz(0)in(1)fDest(1) Mar 3 04:12:47.624: -
cid2(52)st2(SSA_CS_CALL_SETTING)oldst2(SSA_CS_MAPPING) Mar 3 04:12:47.624: ssaReportDigitsDone
cid(51) peer list: (empty) Mar 3 04:12:47.624: ssaReportDigitsDone callid=51 Reporting disabled.
Mar 3 04:12:47.628: H225 NONSTD OUTGOING PDU ::= value ARQnonStandardInfo ::= { sourceAlias { }
sourceExtAlias { } callingOctet3a 128 interfaceSpecificBillingId "ISDN-VOICE" } Mar 3
04:12:47.628: H225 NONSTD OUTGOING ENCODE BUFFER::= 80 000008A0 01800B12 4953444E 2D564F49 43 45
Mar 3 04:12:47.628: Mar 3 04:12:47.628: RAS OUTGOING PDU ::= value RasMessage ::=
```

admissionRequest : *!--- ARQ is sent to the gatekeeper.* requestSeqNum 2210 callType pointToPoint : NULL callModel direct : NULL endpointIdentifier {"81206D2C00000001"} destinationInfo { e164 : "3653" } srcInfo { e164 : "4085272923", h323-ID : {"gwa-1"} } bandWidth 640 callReferenceValue 26 nonStandardData { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '80000008A001800B124953444E2D564F494345'H } conferenceID '2D3AAC3316A411CC80688828285B8DF6'H activeMC FALSE answerCall FALSE canMapAlias TRUE callIdentifier { guid '2D3AAC3316A411CC80698828285B8DF6'H } willSupplyUIEs FALSE } Mar 3 04:12:47.636: RAS OUTGOING ENCODE BUFFER::= 27 8808A100 F0003800 31003200 30003600 44003200 4 3003000 30003000 30003000 30003000 31010180 69860204 8073B85A 5C564004 00670077 0061002D 00314002 80 001A40 B5000012 13800000 08A00180 0B124953 444E2D56 4F494345 2D3AAC33 16A411CC 80688828 285B8DF6 04E 02001 8011002D 3AAC3316 A411CC80 69882828 5B8DF601 00 Mar 3 04:12:47.640: Mar 3 04:12:47.656: RAS INCOMING ENCODE BUFFER::= 80 050008A1 2327 Mar 3 04:12:47.656: Mar 3 04:12:47.656: RAS INCOMING PDU ::= value RasMessage ::= requestInProgress : { requestSeqNum 2210 delay 9000 } Mar 3 04:12:47.704: RAS INCOMING ENCODE BUFFER::= 2B 0008A140 028000AC 100D1A06 B800EF1A 10C01201 1 0000200 AC100D2A 06B800AC 100D1706 B8010002 0000 Mar 3 04:12:47.704: Mar 3 04:12:47.704: RAS INCOMING PDU ::= value RasMessage ::= **admissionConfirm** : *!--- ACF is received.* { requestSeqNum 2210 bandWidth 640 callModel direct : NULL **destCallSignalAddress** **ipAddress** : *!--- Primary destination endpoint.* { **ip** 'AC100D1A'H port 1720 } irrFrequency 240 **alternateEndpoints** *!--- List of alternate endpoints.* { { **callSignalAddress** { **ipAddress** : { **ip** 'AC100D2A'H *!--- 172.16.13.42.* port 1720 }, **ipAddress** : { **ip** 'AC100D17'H *!--- 172.16.13.23.* port 1720 } } } } willRespondToIRR FALSE uuiesRequested { setup FALSE callProceeding FALSE connect FALSE alerting FALSE information FALSE releaseComplete FALSE facility FALSE progress FALSE empty FALSE } } Mar 3 04:12:47.720: H225 NONSTD OUTGOING PDU ::= value H323_UU_NonStdInfo ::= { version 2 protoParam qsigNonStdInfo : { iei 4 rawMesg '04038090A31803A983816C0C2180343038353237...'H } } } Mar 3 04:12:47.720: H225 NONSTD OUTGOING ENCODE BUFFER::= 60 01020001 041F0403 8090A318 03A98381 6C 0C2180 34303835 32373239 32337005 80333635 33 Mar 3 04:12:47.724: Mar 3 04:12:47.724: H225.0 OUTGOING PDU ::= value H323_UserInformation ::= { h323-uu-pdu { **h323-message-body setup** : *!--- H.225 setup sent to primary endpoint.* { protocolIdentifier { 0 0 8 2250 0 2 } sourceAddress { h323-ID : {"gwa-1"} } sourceInfo { gateway { protocol { voice : { supportedPrefixes { { prefix e164 : "1#" } } } } } } mc FALSE undefinedNode FALSE } activeMC FALSE conferenceID '2D3AAC3316A411CC80688828285B8DF6'H conferenceGoal create : NULL callType pointToPoint : NULL sourceCallSignalAddress **ipAddress** : { **ip** 'AC100D0F'H port 11025 } callIdentifier { guid '2D3AAC3316A411CC80698828285B8DF6'H } fastStart { '0000000C6013800A04000100AC100D0F47F1'H, '400000060401004C6013801114000100AC100D0F...'H } mediaWaitForConnect FALSE canOverlapSend FALSE } h245Tunneling TRUE nonStandardControl { { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '6001020001041F04038090A31803A983816C0C21...'H } } } } } Mar 3 04:12:47.740: H225.0 OUTGOING ENCODE BUFFER::= 20 A0060008 914A0002 01400400 67007700 61002D0 0 31088001 3C050401 00204000 2D3AAC33 16A411CC 80688828 285B8DF6 00451C07 00AC100D 0F2B1111 002D3AAC 3316A411 CC806988 28285B8D F6320212 0000000C 6013800A 04000100 AC100D0F 47F11D40 00000604 01004C60 13801114 000100AC 100D0F47 F000AC10 0D0F47F1 01000100 06A00180 2D0140B5 00001226 60010200 01041F04 0 38090A3 1803A983 816C0C21 80343038 35323732 39323370 05803336 3533 Mar 3 04:12:47.744: Mar 3 04:12:47.760: H225.0 INCOMING ENCODE BUFFER::= 25 80060008 914A0004 11001100 2D3AAC33 16A411 C 80698828 285B8DF6 10800180 Mar 3 04:12:47.760: Mar 3 04:12:47.760: H225.0 INCOMING PDU ::= value H323_UserInformation ::= { h323-uu-pdu { **h323-message-body releaseComplete** : *!--- First setup message failed.* { protocolIdentifier { 0 0 8 2250 0 4 } callIdentifier { guid '2D3AAC3316A411CC80698828285B8DF6'H } } h245Tunneling TRUE } } } Mar 3 04:12:47.776: H225 NONSTD OUTGOING PDU ::= value H323_UU_NonStdInfo ::= { version 2 protoParam qsigNonStdInfo : { iei 4 rawMesg '04038090A31803A983816C0C2180343038353237...'H } } } Mar 3 04:12:47.776: H225 NONSTD OUTGOING ENCODE BUFFER::= 60 01020001 041F0403 8090A318 03A98381 6C 0C2180 34303835 32373239 32337005 80333635 33 Mar 3 04:12:47.776: Mar 3 04:12:47.776: H225.0 OUTGOING PDU ::= value H323_UserInformation ::= { h323-uu-pdu { **h323-message-body setup** : *!--- Second setup sent to alternate endpoint.* { protocolIdentifier { 0 0 8 2250 0 2 } sourceAddress { h323-ID : {"gwa-1"} } sourceInfo { gateway { protocol { voice : { supportedPrefixes { { prefix e164 : "1#" } } } } } } mc FALSE undefinedNode FALSE } activeMC FALSE conferenceID '2D3AAC3316A411CC80688828285B8DF6'H conferenceGoal create : NULL callType pointToPoint : NULL sourceCallSignalAddress **ipAddress** : { **ip** 'AC100D0F'H port 11027 } callIdentifier { guid '2D3AAC3316A411CC80698828285B8DF6'H } fastStart { '0000000C6013800A04000100AC100D0F47F1'H, '400000060401004C6013801114000100AC100D0F...'H } mediaWaitForConnect FALSE canOverlapSend FALSE } h245Tunneling TRUE nonStandardControl { { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '6001020001041F04038090A31803A983816C0C21...'H } } } } } Mar 3 04:12:47.796: H225.0 OUTGOING ENCODE BUFFER::= 20 A0060008 914A0002 01400400 67007700 61002D0 0 31088001 3C050401 00204000 2D3AAC33 16A411CC 80688828 285B8DF6 00451C07 00AC100D 0F2B1311 002D3AAC 3316A411 CC806988

```

28285B8D F6320212 0000000C 6013800A 04000100 AC100D0F 47F11D40 00000604 01004C60 13801114
000100AC 100D0F47 F000AC10 0D0F47F1 01000100 06A00180 2D0140B5 00001226 60010200 01041F04 0
38090A3 1803A983 816C0C21 80343038 35323732 39323370 05803336 3533 Mar 3 04:12:47.800: Mar 3
04:12:47.872: H225.0 INCOMING ENCODE BUFFER::= 21 80060008 914A0003 00078E11 002D3AAC 3316A41 1
CC806988 28285B8D F6390219 0000000C 60138011 14000100 AC100D17 479E00AC 100D1747 9F1D4000
00060401 004C6013 80111400 0100AC10 0D0F47F0 00AC100D 17479F01 00010008 800180 Mar 3
04:12:47.872: Mar 3 04:12:47.876: H225.0 INCOMING PDU ::= value H323_UserInformation ::= { h323-
uu-pdu { h323-message-body callProceeding : !--- Call proceeding received. { protocolIdentifier
{ 0 0 8 2250 0 3 } destinationInfo { mc FALSE undefinedNode FALSE } callIdentifier { guid
'2D3AAC3316A411CC80698828285B8DF6'H } fastStart {
'0000000C6013801114000100AC100D17479E00AC...'H, '400000060401004C6013801114000100AC100D0F...'H }
} h245Tunneling TRUE } } Mar 3 04:12:47.884: H225.0 OUTGOING PDU ::= value H323_UserInformation
::= { h323-uu-pdu { h323-message-body empty : NULL h245Tunneling TRUE h245Control {
'0270010600088175000380138000140001000001...'H } } } Mar 3 04:12:47.884: H225.0 OUTGOING ENCODE
BUFFER::= 28 10010006 C0018063 01610270 01060008 8175000 3 80138000 14000100 00010000 0100000C
C0010001 00048000 104810B5 0000120C 52747044 746D6652 656C6179 00008000 16830150 80001583
01408000 12830110 80000020 C0130080 01020000 16020015 00120010 000000 Mar 3 04:12:47.888: Mar 3
04:12:47.888: H225.0 OUTGOING PDU ::= value H323_UserInformation ::= { h323-uu-pdu { h323-
message-body empty : NULL h245Tunneling TRUE h245Control { '01003C4010F3'H } } } Mar 3
04:12:47.892: H225.0 OUTGOING ENCODE BUFFER::= 28 10010006 C0018008 01060100 3C4010F3 Mar 3
04:12:47.892: Mar 3 04:12:47.892: cc_api_call_proceeding(vdbPtr=0x621B2360, callID=0x34,
prog_ind=0x0) Mar 3 04:12:47.896: sess_appl: ev(21=CC_EV_CALL_PROCEEDING), cid(52), disp(0) Mar
3 04:12:47.896: cid(52)st(SSA_CS_CALL_SETTING)ev(SSA_EV_CALL_PROCEEDING)
oldst(SSA_CS_MAPPING)cfid(-1)csz(0)in(0)fDest(0) Mar 3 04:12:47.896: -
cid2(51)st2(SSA_CS_CALL_SETTING)oldst2(SSA_CS_CALL_SETTING) Mar 3 04:12:47.896: ssaCallProc Mar
3 04:12:47.896: ccGetDialpeerTag (callID=0x33) Mar 3 04:12:47.896: ssaIgnore cid(52),
st(SSA_CS_CALL_SETTING),oldst(1), ev(21) Mar 3 04:12:47.900: H225.0 INCOMING ENCODE BUFFER::= 28
10010008 C0018063 01610270 01060008 8175000 6 80138000 14000100 00010000 0100000C C0010001
00048000 104810B5 0000120C 52747044 746D6652 656C6179 00008000 16830150 80001583 01408000
12830110 80000020 C0130080 01020000 16020015 00120010 000000 Mar 3 04:12:47.904: Mar 3
04:12:47.904: H225.0 INCOMING PDU ::= value H323_UserInformation ::= { h323-uu-pdu { h323-
message-body empty : NULL h245Tunneling TRUE h245Control {
'0270010600088175000680138000140001000001...'H } } } !--- Some of the unnecessary H.225 debug
messages are deleted here. Mar 3 04:12:52.116: H225.0 INCOMING ENCODE BUFFER::= 23 80060008
914A0003 000A8600 11002D3A AC3316A 4 11CC8069 8828285B 8DF60100 01000880 0180 Mar 3
04:12:52.120: Mar 3 04:12:52.120: H225.0 INCOMING PDU ::= value H323_UserInformation ::= { h323-
uu-pdu { h323-message-body alerting : !--- Alerting message received. { protocolIdentifier { 0 0
8 2250 0 3 } destinationInfo { mc FALSE undefinedNode FALSE } callIdentifier { guid
'2D3AAC3316A411CC80698828285B8DF6'H } } h245Tunneling TRUE } } Mar 3 04:12:52.124:
cc_api_call_alert(vdbPtr=0x621B2360, callID=0x34, prog_ind=0x8, sig_ind=0x1) Mar 3 04:12:52.124:
sess_appl: ev(7=CC_EV_CALL_ALERT), cid(52), disp(0) Mar 3 04:12:52.124:
cid(52)st(SSA_CS_CALL_SETTING)ev(SSA_EV_CALL_ALERT) oldst(SSA_CS_CALL_SETTING)cfid(-
1)csz(0)in(0)fDest(0)

```

排除故障负载均衡

使用负载均衡功能，您能设置有一特定的阈值的网守呼叫、内存、已注册终端CPU和编号编号的。一旦该阈值达到，网守移动已注册思科H.323终端向备选网守或拒绝新建的呼叫和注册。使用以下 `gatekeeper cli` 命令，负载均衡启用：

```
Router(config-gk)#load-balance [endpoints max-endpoints] [calls max-calls] [cpu max-%cpu][memory max-%mem-used]
```

当阈值满足时，网守使用RAS信息的RRJ通知关于备选网守和拒绝原因的终端。当此消息接收时，终端发送一新的RRQ给备选网守。一旦它注册与备选网守，使用GUP消息通知在集群的所有网守关于新建的已注册终端。

注意的某些问题，当您排除故障时是检查在网守的配置和确保备选网守和负载均衡是工作。使用得以上的拓扑为了实现故障排除目的。"gkb-1"网守的配置更改显示以下案件：

- 网守如何能拒绝呼叫，当阈值满足。

- 网守如何能移动终端的注册向备选网守，当阈值满足。

调试负载均衡功能、使用 [debug gatekeeper load](#) 和 `debug h225 asn1` 发现网守如何起反应，当阈值满足。

这是使用报道以上提到的两个案件"gkb-1"网守的配置(呼叫门限值已注册端点编号和编号)：

```
!
gatekeeper
zone local gkb-1 domainB.com 172.16.13.41
zone remote gka-1 domainA.com 172.16.13.35 1719
zone cluster local gkb gkb-1
  element gkb-2 172.16.13.16 1719
!
security token required-for all
gw-type-prefix 2#* default-technology
bandwidth total zone gkb-1 512
bandwidth session zone gkb-1 512
load-balance endpoints 2 calls 1 !--- maximum of 2 endpoints and call threshold is 1 no
shutdown ! !
```

呼叫通过网守gkb-1被做。当该呼叫是UP时，另一呼叫被做。获取调试显示什么负载平衡调试看上去象，并且网守如何拒绝第二次呼叫，因为阈值满足。您能使用以下命令显示使用网守，多少激活的呼叫运行：

```
gkb-1#show gatekeeper call Total number of active calls = 1. GATEKEEPER CALL INFO
===== LocalCallID Age(secs) BW 5-29514 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
```

这是H.225 asn1和关守负载调试，当第二次呼叫请求时：

```
Mar  3 05:04:55.354: RAS INCOMING ENCODE BUFFER ::= 4A 80080501 01806986
40B50000 12298286 B0110075 7
95BF216 AB11CC80 95882828 5B8DF601 81110201 80866940 04006700 77006100
2D003100 AC100D23 06B70B80 0D
014004 0067006B 0061002D 00310180
Mar  3 05:04:55.358:
Mar  3 05:04:55.358: RAS INCOMING PDU ::=
```

```
value RasMessage ::= locationRequest : !--- LRQ is received. { requestSeqNum 2054
destinationInfo { e164 : "3653" } nonStandardData { nonStandardIdentifier h221NonStandard : {
t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data
'8286B0110075795BF216AB11CC80958828285B8D...'H } replyAddress ipAddress : { ip 'AC100D23'H port
1719 } sourceInfo { h323-ID : {"gka-1"} } canMapAlias TRUE } Mar 3 05:04:55.362: H225 NONSTD
INCOMING ENCODE BUFFER ::= 82 86B01100 75795BF2 16AB11CC 80958828 28 5B8DF6 01811102 01808669
40040067 00770061 002D0031 Mar 3 05:04:55.366: Mar 3 05:04:55.366: H225 NONSTD INCOMING PDU ::=
value LRQnonStandardInfo ::= { ttl 6 nonstd-callIdentifier { guid
'75795BF216AB11CC80958828285B8DF6'H } callingOctet3a 129 gatewaySrcInfo { e164 : "5336", h323-ID
: {"gwa-1"} } } Mar 3 05:04:55.366: gk_load_overloaded: Overloaded due to reaching specified
call limits !--- Number of calls threshold has met. Mar 3 05:04:55.370: RAS OUTGOING PDU ::=
value RasMessage ::= locationReject : !--- LRJ is sent. { requestSeqNum 2054 rejectReason
undefinedReason : NULL }
```

对于第二示例，"gkb-1"网守有两个注册的终端。另一个终端的注册尝试。因为他们在同一集群，网守移动终端，尝试注册对备选网守"gkb-2"。这是 `debug h225 asn1` 和 `debug gatekeeper gup asn1` 的调试消息此案件的：

```
Mar  3 05:21:05.682: RAS INCOMING PDU ::=
```

```
value RasMessage ::= registrationRequest : !--- RRQ message is received. { requestSeqNum 4621
protocolIdentifier { 0 0 8 2250 0 3 } discoveryComplete TRUE callSignalAddress { ipAddress : {
ip 'AC100D2A'H port 1720 } } rasAddress { ipAddress : { ip 'AC100D2A'H port 49998 } }
terminalType { gateway { protocol { voice : { supportedPrefixes { { prefix e164 : "1#" } } } } }
```



```
mc FALSE undefinedNode FALSE } terminalAlias { h323-ID : {"gwb-3"} } gatekeeperIdentifier {"gkb-1"} endpointVendor { vendor { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } }
timeToLive 60 tokens { { tokenOID { 1 2 840 113548 10 1 2 1 } timeStamp 731136065 challenge
'5A70CA112E6C7A3834792BD64FF7AD2F'H random 58 generalID {"gwb-3"} } } cryptoTokens {
cryptoEPPwdHash : { alias h323-ID : {"gwb-3"} timeStamp 731136065 token { algorithmOID { 1 2 840
113549 2 5 } paramS { } hash "B1C1DAD962BEE42B1E53F368238B1D8" } } } keepAlive FALSE
willSupplyUIEs FALSE maintainConnection TRUE } Mar 3 05:21:05.698: gk_load_overloaded:
Overloaded due to reaching specified endpoint limits !--- Endpoint threshold is met. Mar 3
05:21:05.702: RAS OUTGONG PDU ::= value RasMessage ::= registrationReject : !--- RRJ is sent. {
requestSeqNum 4621 protocolIdentifier { 0 0 8 2250 0 3 } rejectReason resourceUnavailable : NULL
!--- Reject reason. gatekeeperIdentifier {"gkb-1"} altGKInfo !--- List of alternate gatekeepers.
{ alternateGatekeeper { { rasAddress ipAddress : { ip 'AC100D10'H port 1719 }
gatekeeperIdentifier {"gkb-2"} needToRegister TRUE priority 0 } } altGKisPermanent TRUE !---
Informs the endpoint that the move is permanent. } Mar 3 05:21:05.706: RAS OUTGOING ENCODE
BUFFER::= 16 80120C06 0008914A 00038101 00080067 006B0062 0 02D0031 07001600 0140AC10 0D1006B7
08006700 6B006200 2D003280 80 Mar 3 05:21:05.706: Mar 3 05:21:05.782: Received GUP REGISTRATION
INDICATION from 172.16.13.16 !--- GUP update for the new endpoint. gkb-1#
```

相关信息

- [语音技术支持](#)
- [语音和 IP 通信产品支持](#)
- [Cisco IP 电话故障排除](#)
- [技术支持 - Cisco Systems](#)