

Настройка простейшего контроля использования полосы пропускания в привратнике

Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Условные обозначения](#)

[Общие сведения](#)

[команда bandwidth \(сторожевое устройство\)](#)

[Настройка](#)

[Схема сети](#)

[Конфигурации](#)

[Проверка](#)

[Устранение неполадок](#)

[Команды для устранения неполадок](#)

[Показ образца и данные отладки](#)

[Дополнительные сведения](#)

Введение

Этот документ предоставляет пример конфигурации для основных правил приема вызовов сторожевым устройством.

Предварительные условия

Требования

Существует несколько условий, которые будут встречены, прежде чем шлюз будет в состоянии получить разрешение верного адреса из сторожевого устройства. Вот несколько важных пунктов, которые надлежит проверить для каждого решения VoIP, если задействованы низкоскоростные соединения.

Прежде чем использовать эту конфигурацию, убедитесь, что выполняются эти требования:

- Все шлюзы должны быть зарегистрированы на соответствующих сторожевых устройствах
- Все сторожевые устройства должны иметь корректный план соединений, таким

образом, они могут выбрать маршрут для вызовов.

- Можно конфигурировать входной контроль для ограничения количества вызовов между определенными зонами.

[Так как первые два пункта рассматриваются в разделе Конфигурация, мы сосредоточимся на управлении доступа в секции Вспомогательная информация.](#)

Используемые компоненты

Сведения, содержащиеся в данном документе, касаются следующих версий программного обеспечения и оборудования:

- Три Маршрутизатора Cisco 2600.
- MCM PLUS/Н323 ПРЕДПРИЯТИЯ выпуска 12.2.8.5 программного обеспечения Cisco IOS.

Сведения, содержащиеся в данном документе, были получены с устройств в специальной лабораторной среде. Все устройства, описанные в данном документе, были запущены с конфигурацией по умолчанию. При работе с реальной сетью необходимо полностью осознавать возможные результаты использования всех команд.

Условные обозначения

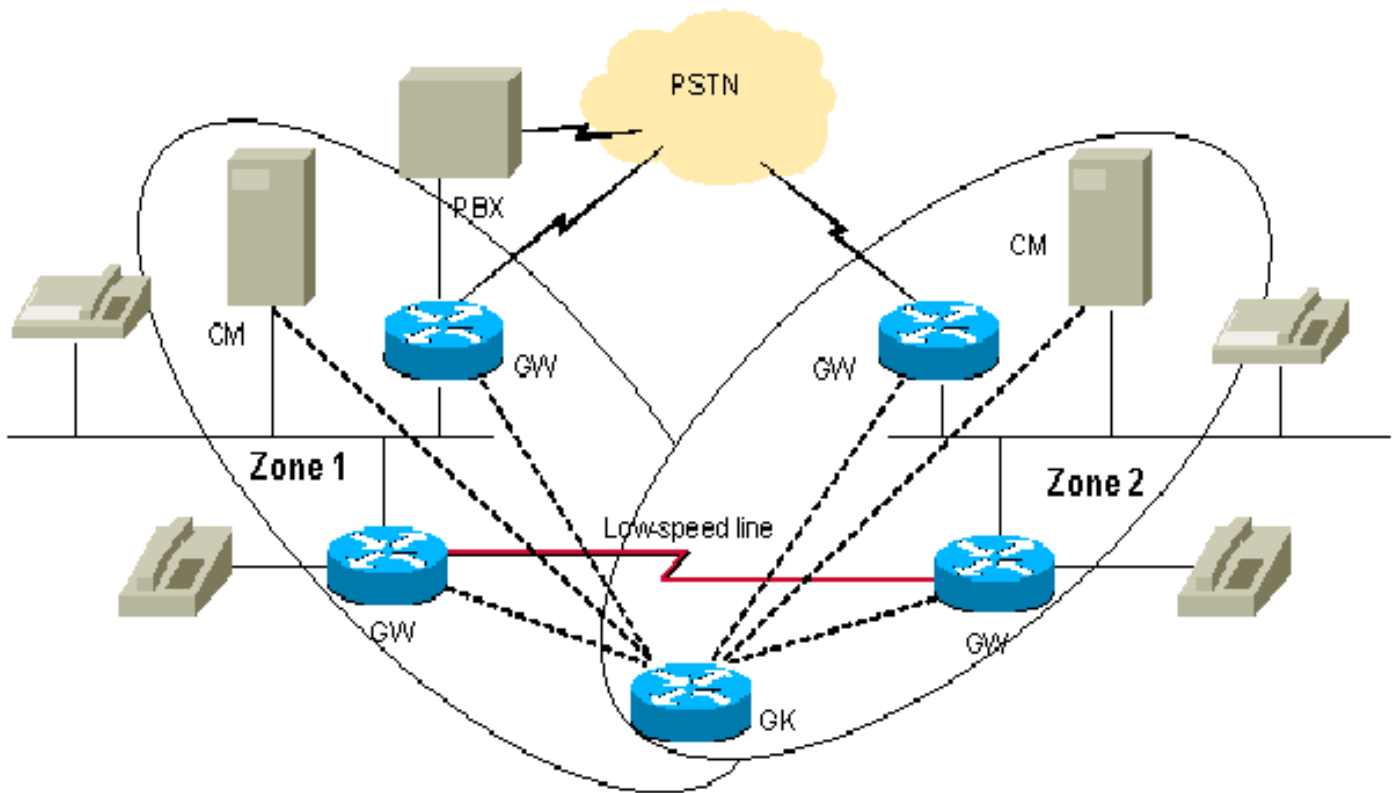
[Дополнительные сведения об условных обозначениях см. в документе Технические рекомендации Cisco. Условные обозначения.](#)

Общие сведения

Этот пример конфигурации рассматривает сеть VoIP с двухзонной топологией, которая управляется одним узлом с тремя шлюзами в двух зонах. Цель этого документа - дать простой пример конфигурации контроля доступа, который применяет policy к количеству звонков между зонами и внутри их. Этот документ включает общие технические сведения о сконфигурированных свойствах, руководства по проектированию, и основных стратегиях верификации и устранения неполадок.

Примечание: В этой конфигурации эти четыре маршрутизатора расположены на той же LAN. Тем не менее, в реальной топологии все устройства могут находиться в разных частях вашей сети.

Очень часто существует несколько источников трафика с высоким приоритетом в реальных сетях. Это - сложная задача для различения всех этих условий, потому что они являются многочисленными, и легкими пропустить. Однако существует несколько обычных ситуаций, которые происходят очень часто в реальной жизни, которые достойны рассмотрения. Контроль доступа становится проблемой, когда маршрутизаторы, которые предоставляют приоритизацию трафика, являются самостоятельно не источниками такого трафика. Стандартная топология включает несколько голосовых шлюзов на двух узлах, подключенных по каналу, предоставленному парой маршрутизаторов. Другая топология связала Cisco CallManager с IP-телефонами на двух узлах, наряду со шлюзами к PSTN или УАТС. И в том, и в другом случае на каждом конце канала имеется несколько источников голосового трафика.



Иногда возможны проблемы с качеством голосовой связи, если количество голосового трафика превышает настраиваемую пропускную способность для приоритетной очереди. Это вызвано тем, что маршрутизаторы и Cisco CallManager / IP-телефоны, которые инициируют трафик, не имеют централизованного управления для доступа вызова в дизайне данным выше. В этом случае пакеты, превышающие пропускную способность, будут отброшены.

Есть несколько способов избежать этого сценария. Наиболее легким решением является установка речевой полосы частот (voice bandwidth) в очереди с низкой задержкой (Low Latency Queue (LLQ)) для того, чтобы принимать максимальное количество звонков из всех источников. При отсутствии голосового трафика неиспользуемая полоса пропускания предоставляется для потока данных. Когда общая пропускная способность ссылки выше, чем пропускная способность, требуемая для максимального числа вызовов, это может быть сделано.

Больше осмысленного подхода должно ввести ограничения на каждом источнике голосового трафика от обоих концов ссылки. Когда вы сделаете так, суммарная пропускная способность от всех них не превысит рекомендуемые 75% действительной пропускной способности ссылки между узлами. Для введения тех ограничений используйте команду **max-conn** под конфигурацией узла коммутации VoIP. Если мы предполагаем, что существует Cisco CallManager только в одном центральном узле, мы можем использовать его возможности ограничить количество вызовов к узлу филиала без CallManager. Этот подход позволяет управлять ситуацией, когда источники голосового трафика могут превысить подписку канала. Недостаток этого подхода является негибким использованием полосы пропускания, предоставленным источникам. Даже если существует свободная пропускная способность, доступная в тот момент, этот подход не позволяет некоторым шлюзам размещать дополнительный вызов.

Наиболее гибким решением будет использование отдельного элемента для централизованного контроля использования полосы пропускания: сторожевое устройство. Сторожевое устройство помогает связывать два узла с двумя Cisco CallManager (или

Кластеры CallManager).

Примечание: Использование сторожевого устройства не всегда означает покупать новый отдельный маршрутизатор. На основе количества вызовов и загрузки маршрутизаторов, можно настроить сторожевое устройство на одном из существующих маршрутизаторов с соответствующим набором функций Cisco IOS как ПРЕДПРИЯТИЕ/ПЛЮС/Н323. Это может помочь управлять маленькими ответвлениями и позволять специализированное сторожевое устройство в центральном узле только.

Подход сторожевого устройства нужно рассмотреть с осторожностью, чтобы не сокрушить маршрутизатор с дополнительной загрузкой. Кроме того, вы должны проверить позволит ли топология расположить гэйткипер таким образом, чтобы избежать дополнительного трафика на критическом соединении.

В общем случае рекомендуется использовать отдельные маршрутизаторы Cisco в качестве коммутируемых привратников, количество которых зависит от топологии сети.

Рассмотрите топологию выше. Здесь, можно поместить все устройства в эти две локальных зоны, которыми управляет одно сторожевое устройство. Это позволяет вам иметь большое число вызовов в каждой зоне, но ограничивает количество вызовов между ними. В нашем тестовом примере мы ограничим пропускную способность между этими двумя зонами к одному вызову и позволим до двух (более высокое количество) вызовы в одном из них.

Для более подробной информации об этом посмотрите [Контроль приема вызовов по технологии VoIP](#).

Для выполнения задачи используйте **пропускную способность** (сторожевое устройство) команда, описанная в [Cisco Высокоэффективное Сторожевое устройство](#)

[команда bandwidth \(сторожевое устройство\)](#)

Для определения максимальной суммарной пропускной способности трафика H.323 используется команда `bandwidth gatekeeper configuration`. Для отключения опции используйте эту команду с параметром `no`.

Примечание: Эта команда позволяет вам ограничивать пропускную способность через одно соединение от зоны. Если топология позволяет вам заказывать телефонный разговор через несколько путей от одной зоны до другого, ссылки могли легко стать превышенными. Можете использовать следующую топологию: две зоны связаны через два пути, позволив только один вызов через каждый путь. Если пропускная способность будет ограничена одним вызовом, то второй путь никогда не будет использоваться. Но если пропускная способность ограничена двумя вызовами, одна из ссылок может быть превышена. Эта команда может применяться для зон, имеющих только один путь ко всем остальным зонам. "Концентратор и лучевая" топология являются исключением. Несмотря на то, что концентратор имеет разнообразные пути, он не превысит намеченную сумму ссылок, поскольку количество вызовов будет ограничено в лучах для каждой ссылки.

`пропускная способность {промежуточная зона | общее количество | сеанс} {по умолчанию | зональное имя зоны} размер пропускной способности`

`никакая пропускная способность {промежуточная зона | общее количество | сеанс} {по умолчанию | зональное имя зоны} размер пропускной способности`

[Описание синтаксиса](#)

Следующая таблица описывает синтаксис:

Синтаксис	Описание
промежуточная зона	Определяет общую полосу пропускания для трафика N.323 из данной зоны в любую другую.
общее количество	Задаёт максимальную пропускную способность для трафика N.323, позволенного в зоне.
сеанс	Указывает, что максимальная пропускная способность обеспечила сеанс в зоне.
по умолчанию	Задаёт значение по умолчанию для всех зон.
зональное имя зоны	Задаёт конкретную зону. Именует отдельную зону.
размер пропускной способности	Максимальная пропускная способность. Для промежуточного пространства и в целом диапазон составляет от 1 до 10000000 кбит/с. Для сеанса диапазон от 1 до 5,000 кбит/с.

[Настройки по умолчанию](#)

Нет

[Командные режимы](#)

Конфигурация привратника

[История команд](#)

Следующая таблица описывает историю команд:

Выпуск	Модификация
12.1 (3) КСИ	Команда включена впервые.
12.1 (5) ХМ	Команда bandwidth была сделана распознаваемой, не используя команду сторожевого устройства зоны .
12.2 (2) Т	Эта команда была использована в Cisco IOS Software Release 12.2(2)Т.

12.2 (2) XB1	Эта команда была реализована в универсальном шлюзе Cisco AS5850.
--------------------	--

[Инструкции по использованию](#)

В предыдущих Cisco IOS Software Release функциональность команды **bandwidth** была получена при помощи команды сторожевого устройства зоны.

[Примеры](#)

Приведенный ниже пример задает максимальную полосу пропускания зоны на 5 000 кбит/с:

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

[Связанные команды](#)

[bandwidth remote](#) — Задает общую пропускную способность для трафика H.323 между этим сторожевым устройством и любым другим сторожевым устройством.

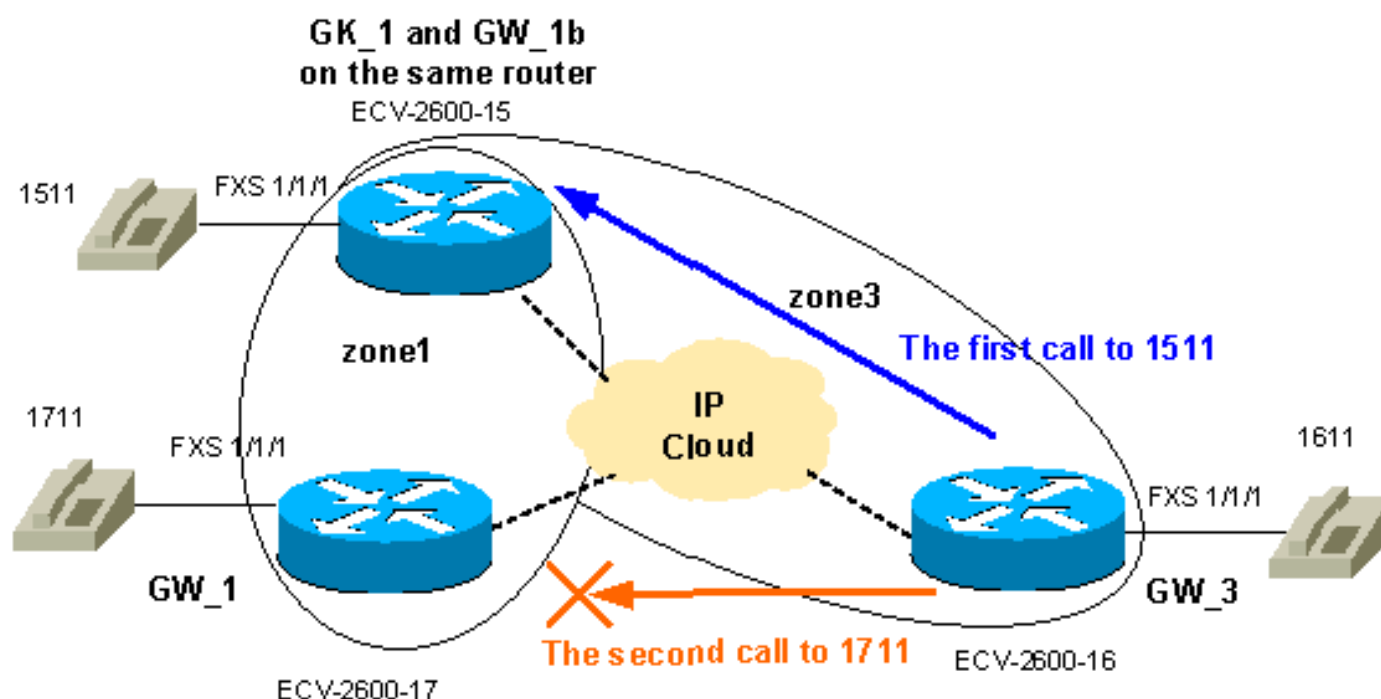
[Настройка](#)

В этом разделе содержатся сведения о настройке функций, описанных в этом документе.

Примечание: [Поиск дополнительной информации о командах в данном документе можно выполнить с помощью средства "Command Lookup" \(Поиск команд\) \(только для зарегистрированных клиентов\).](#)

[Схема сети](#)

В настоящем документе используется следующая схема сети:



Конфигурации

Цель состоит в том, чтобы ограничить доступную пропускную способность между zone1 и zone3 к одному вызову, и позволить более высокое количество вызовов (до двух в данном примере) в zone1. Таким образом, будут удовлетворены общие требования для обычной задачи приема вызова. Сообщения протокола регистрации, доступа и статуса (RAS) идут до сообщений настройки вызова H225. После этого следует согласование H4245, которое фактически определяет возможности обеих сторон. Таким образом, действительная пропускная способность вызова определена после этапа доступа вызова и обмена сообщениями RAS. Поэтому привратник рассматривает каждый вызов как 64КБ-вызов. Поэтому увеличение ограничений пропускной способности между зонами для голосовых вызовов должно быть сделано в инкрементах 64 КБ.

Примечание: GW_3 настроен на том же маршрутизаторе как сторожевое устройство для иллюстрирования такой возможности для филиалов компании начального уровня.

Примечание: Проверка конфигураций сторожевого устройства и шлюза является важной частью устранения проблем GK-GW. Поэтому для упрощения понимания конфигураций все не связанные друг с другом команды настройки были удалены.

ECV-2600-17 GW_1

```
IOS (tm) C2600 Software (C2600-JSX-M), Version 12.2(7a),  
RELEASE SOFTWARE (fc1) ! hostname ECV-2610-17 ! !  
interface Ethernet0/0 ip address 10.52.218.49  
255.255.255.0 h323-gateway voip interface h323-gateway  
voip id gk-zone1.test.com ipaddr 10.52.218.47 1718 h323-  
gateway voip h323-id gw_1 h323-gateway voip tech-prefix  
1# h323-gateway voip bind srcaddr 10.52.218.49 ! voice-  
port 1/1/0 ! voice-port 1/1/1 ! ! dial-peer voice 1 voip  
destination-pattern .... session target ras ! dial-peer  
voice 2 pots destination-pattern 1711 port 1/1/1 no  
register e164 ! gateway ! end
```

ECV-2600-16 GW_2

```
!  
hostname ECV-2610-16  
!  
!  
interface Ethernet0/0  
ip address 10.52.218.48 255.255.255.0  
h323-gateway voip interface h323-gateway voip id gk-  
zone3.test.com ipaddr 10.52.218.47 1718 h323-gateway  
voip h323-id gw_3 h323-gateway voip tech-prefix 1# h323-  
gateway voip bind srcaddr 10.52.218.48 ! ! voice-port  
1/1/0 ! voice-port 1/1/1 ! dial-peer voice 1 voip  
destination-pattern .... session target ras ! dial-peer  
voice 2 pots destination-pattern 1611 port 1/1/1 no  
register e164 ! gateway ! ! end
```

GK_1 ECV-2600-15

```
hostname ECV-2610-15  
!  
boot system tftp c2600-jsx-mz.122-7a.bin 10.52.218.2  
!  
interface Ethernet0/0  
ip address 10.52.218.47 255.255.255.0  
half-duplex  
h323-gateway voip interface h323-gateway voip id gk-  
zone1.test.com ipaddr 10.52.218.47 1718 h323-gateway
```



```
voip h323-id gw_1b h323-gateway voip tech-prefix 1#
h323-gateway voip bind srcaddr 10.52.218.47 !! voice-
port 1/1/0 ! voice-port 1/1/1 !! dial-peer voice 6 pots
destination-pattern 1511 port 1/1/1 no register e164 !!
dial-peer voice 5 voip destination-pattern .... session
target ras ! gateway !! gatekeeper zone local gk-
zone1.test.com test.com 10.52.218.47 zone local gk-
zone3.test.com test.com zone prefix gk-zone1.test.com
15.. gw-priority 10 gw_1b zone prefix gk-zone3.test.com
16.. gw-priority 10 gw_3 zone prefix gk-zone1.test.com
17.. gw-priority 10 gw_1 gw-type-prefix 1#* default-
technology bandwidth interzone zone gk-zone1.test.com 64
!--- Applies the restriction between gk-zone1, and all
!--- other zones to 64bk. That allows one call only.
bandwidth total zone gk-zone1.test.com 128 !--- Applies
the restriction to the total number of calls in zone1,
!--- and allows two call in the gk-zone1. no shutdown !
end ECV-2610-15#
```

Проверка

В этом разделе содержатся сведения, которые помогают убедиться в надлежащей работе конфигурации.

Некоторые команды show поддерживаются Средством интерпретации выходных данных(только зарегистрированные клиенты), которое позволяет просматривать аналитику выходных данных команды show.

- show gateway - вывод состояния регистрации шлюза.
- show gatekeeper endpoints все шлюзы, зарегистрированные к сторожевому устройству.
- show gatekeeper zone prefix - показывает все префиксы зоны, сконфигурированные на привратнике.
- команда show gatekeeper call отображает активные вызовы, обрабатываемые привратником.

Устранение неполадок

В этом разделе описывается процесс устранения неполадок конфигурации.

Команды для устранения неполадок

Некоторые команды show поддерживаются Средством интерпретации выходных данных(только зарегистрированные клиенты), которое позволяет просматривать аналитику выходных данных команды show.

Примечание: Прежде чем применять команды отладки, ознакомьтесь с разделом "Важные сведения о командах отладки".

- debug h225 asn1 H225 (RAS и настройка вызова Q931) сообщения.
- debug scc323 h225 — отображает сообщения настройки вызова H225.

Вот некоторые полезные ссылки:

- [Основы устранения неполадок и отладки вызовов по протоколу VoIP — основные сведения](#)
- [Команды отладки передачи голоса по IP-протоколу \(VoIP\)](#)
- [Справочное руководство по обмену голосовыми, видео и факсимильными данными для Cisco IOS, версия 12.2](#)

Показ образца и данные отладки

```

!--- First step is to check the gateway registrations. !--- On the first gateway: ECV-2610-
17#show gateway Gateway gw_1 is registered to Gatekeeper gk-zone1.test.com Alias list (CLI
configured) H323-ID gw_1 Alias list (last RCF) H323-ID gw_1 H323 resource thresholding is
Disabled ECV-2610-17# ----- !--- And
on the second Gateway: ECV-2610-16#show gateway Gateway gw_3 is registered to Gatekeeper gk-
zone3.test.com Alias list (CLI configured) H323-ID gw_3 Alias list (last RCF) H323-ID gw_3 H323
resource thresholding is Disabled ECV-2610-16#-----
----- !--- The same on the third Gateway: ECV-2610-15#show gateway Gateway gw_1b is
registered to Gatekeeper gk-zone1.test.com Alias list (CLI configured) H323-ID gw_1b Alias list
(last RCF) H323-ID gw_1b H323 resource thresholding is Disabled ECV-2610-15#-----
----- !--- And on the corresponding Gatekeeper: ECV-
2610-15#show gatekeeper end GATEKEEPER ENDPOINT REGISTRATION =====
CallSignalAddr Port RASignalAddr Port Zone Name Type F -----
- ----- -- 10.52.218.47 1720 10.52.218.47 58841 gk-zone1.test.com VOIP-GW H323-ID:
gw_1b 10.52.218.48 1720 10.52.218.48 59067 gk-zone3.test.com VOIP-GW H323-ID: gw_3 10.52.218.49
1720 10.52.218.49 52887 gk-zone1.test.com VOIP-GW H323-ID: gw_1 Total number of active
registrations = 3 ECV-2610-15# -----
----- !--- To check the dial plan on the Gatekeeper: ECV-2610-15#show gatekeeper zone pre ZONE
PREFIX TABLE ===== GK-NAME E164-PREFIX ----- gk-zone1.test.com 15..
gk-zone3.test.com 16.. gk-zone1.test.com 17.. ECV-2610-15# !--- All configured prefixes should
be seen in the zone list. -----
- !--- To check the zone status on the Gatekeeper: !-- The output shows one permitted interzone
call. ECV-2610-15#show gatekeeper zone st GATEKEEPER ZONES ===== GK name Domain Name
RAS Address PORT FLAGS ----- !--- The output shows the
bandwidth restrictions for this zone. gk-zone1.tes test.com 10.52.218.47 1719 LS BANDWIDTH
INFORMATION (kbps) : Maximum total bandwidth : 128 Current total bandwidth : 64 Maximum
interzone bandwidth : 64 Current interzone bandwidth : 64 Maximum session bandwidth : Total
number of concurrent calls : 1 SUBNET ATTRIBUTES : All Other Subnets : (Enabled) PROXY USAGE
CONFIGURATION : Inbound Calls from all other zones : to terminals in local zone gk-
zone1.test.com : use proxy to gateways in local zone gk-zone1.test.com : do not use proxy to
MCUs in local zone gk-zone1.test.com : do not use proxy Outbound Calls to all other zones : from
terminals in local zone gk-zone1.test.com : use proxy from gateways in local zone gk-
zone1.test.com : do not use proxy from MCUs in local zone gk-zone1.test.com : do not use proxy
!--- There are no bandwidth restrictions for this zone. gk-zone3.tes test.com 10.52.218.47 1719
LS BANDWIDTH INFORMATION (kbps) : Maximum total bandwidth : Current total bandwidth : 64 Maximum
interzone bandwidth : Current interzone bandwidth : 64 Maximum session bandwidth : Total number
of concurrent calls : 1 SUBNET ATTRIBUTES : All Other Subnets : (Enabled) PROXY USAGE
CONFIGURATION : Inbound Calls from all other zones : to terminals in local zone gk-
zone3.test.com : use proxy to gateways in local zone gk-zone3.test.com : do not use proxy to
MCUs in local zone gk-zone3.test.com : do not use proxy Outbound Calls to all other zones : from
terminals in local zone gk-zone3.test.com : use proxy from gateways in local zone gk-
zone3.test.com : do not use proxy from MCUs in local zone gk-zone3.test.com : do not use proxy
ECV-2610-15# ----- ECV-2610-
15#show gatekeeper call Total number of active calls = 1. GATEKEEPER CALL INFO
===== LocalCallID Age(secs) BW 5-0 1 64(Kbps) Endpt(s): Alias E.164Addr
CallSignalAddr Port RASignalAddr Port src EP: gw_3 1611 10.52.218.48 1720 10.52.218.48 59067
dst EP: gw_1b 1511 10.52.218.47 1720 10.52.218.47 58841 ECV-2610-15# -----
----- !--- The output shows that we reach maximum number of
calls for gk-zone1. ECV-2610-15# ECV-2610-15#show gatekeeper zone st GATEKEEPER ZONES
===== GK name Domain Name RAS Address PORT FLAGS -----
- ----- gk-zone1.tes test.com 10.52.218.47 1719 LS BANDWIDTH INFORMATION (kbps) : Maximum total
bandwidth : 128 Current total bandwidth : 128 Maximum interzone bandwidth : 64 Current interzone

```

bandwidth : 64 Maximum session bandwidth : **Total number of concurrent calls : 2** SUBNET
ATTRIBUTES : All Other Subnets : (Enabled) PROXY USAGE CONFIGURATION : Inbound Calls from all
other zones : to terminals in local zone gk-zone1.test.com : use proxy to gateways in local zone
gk-zone1.test.com : do not use proxy to MCUs in local zone gk-zone1.test.com : do not use proxy
Outbound Calls to all other zones : from terminals in local zone gk-zone1.test.com : use proxy
from gateways in local zone gk-zone1.test.com : do not use proxy from MCUs in local zone gk-
zone1.test.com : do not use proxy **gk-zone3.test.com 10.52.218.47 1719 LS BANDWIDTH**
INFORMATION (kbps) : **Maximum total bandwidth : Current total bandwidth : 64 Maximum interzone**
bandwidth : Current interzone bandwidth : 64 Maximum session bandwidth : **Total number of**
concurrent calls : 1 SUBNET ATTRIBUTES : All Other Subnets : (Enabled) PROXY USAGE CONFIGURATION
: Inbound Calls from all other zones : to terminals in local zone gk-zone3.test.com : use proxy
to gateways in local zone gk-zone3.test.com : do not use proxy to MCUs in local zone gk-
zone3.test.com : do not use proxy Outbound Calls to all other zones : from terminals in local
zone gk-zone3.test.com : use proxy from gateways in local zone gk-zone3.test.com : do not use
proxy from MCUs in local zone gk-zone3.test.com : do not use proxy **gk-zone2.test.com**
10.52.218.46 1719 RS ECV-2610-15# ECV-2610-15#show gatekeeper call Total number of active calls
= 2. GATEKEEPER CALL INFO ===== LocalCallID Age(secs) BW 20-33504 49 **64(kbps)**
Endpt(s): Alias E.164Addr CallSignalAddr Port RASSignalAddr Port src EP: **gw_3 1611 10.52.218.48**
1720 10.52.218.48 49762 dst EP: gw_1b 1510 10.52.218.47 1720 10.52.218.47 52344 LocalCallID
Age(secs) BW 21-22720 36 **64(Kbps)** Endpt(s): Alias E.164Addr CallSignalAddr Port RASSignalAddr
Port src EP: **gw_1 1711 10.52.218.49 1720 10.52.218.49 54114 dst EP: gw_1b 1511 10.52.218.47 1720**
10.52.218.47 52344 ECV-2610-15# -----
--- !--- The conversation between the gateway and gatekeeper consists of !--- an exchange of RAS
messages. !--- We start call to 1511 from GW_3. ECV-2610-16#deb h225 asn1 H.225 ASN1 Messages
debugging is on ECV-2610-16# *Mar 1 14:22:20.972: **RAS OUTGOING PDU ::= value RasMessage ::=**
admissionRequest : { requestSeqNum 970 callType pointToPoint : NULL callModel direct : NULL
endpointIdentifier {"8262B76400000019"} destinationInfo { e164 : "1511" } srcInfo { h323-ID :
{"gw_3"} } bandwidth 640 callReferenceValue 23 nonStandardData { nonStandardIdentifier
h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '000000'H }
conferenceID '00000000000000000000000000000000'H activeMC FALSE answerCall FALSE canMapAlias
TRUE callIdentifier { guid '00000000000000000000000000000000'H } willSupplyUUIEs FALSE } *Mar 1
14:22:20.992: RAS OUTGOING ENCODE BUFFER::= 27 8803C900 F0003800 32003600 32004200 37003600
34003000 30003000 30003000 30003100 39010180 48440140 03006700 77005F00 33400280 001740B5
00001203 00000000 00000000 00000000 00000000 00000004 E0200180 11000000 00000000 00000000
00000000 00000100 *Mar 1 14:22:21.008: *Mar 1 14:22:21.073: RAS INCOMING ENCODE BUFFER::= 2B
0003C940 0280000A 34DA2F06 B800EF14 00C00100 020000 *Mar 1 14:22:21.077: *Mar 1 14:22:21.081:
RAS INCOMING PDU ::= !--- The GW_3 gets permission to proceed with that call. value RasMessage
::= **admissionConfirm** : { requestSeqNum 970 bandwidth 640 callModel direct : NULL
destCallSignalAddress ipAddress : { ip '0A34DA2F'H 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 } } **!--- The Call setup**
message from GW_3 follows. *Mar 1 14:22:21.105: **H225.0 OUTGOING PDU ::= value**
H323_UserInformation ::= { h323-uu-pdu { h323-message-body setup : { protocolIdentifier { 0 0 8
2250 0 2 } sourceAddress { h323-ID : {"gw_3"} } sourceInfo { gateway { protocol { voice : {
supportedPrefixes { { prefix e164 : "1#" } } } } } mc FALSE undefinedNode FALSE } activeMC FALSE
conferenceID '00000000000000000000000000000000'H conferenceGoal create : NULL callType
pointToPoint : NULL sourceCallSignalAddress ipAddress : { ip '0A34DA30'H port 11018 }
callIdentifier { guid '00000000000000000000000000000000'H } fastStart {
'0000000D4001800A040001000A34DA3041C5'H, '400000060401004D40018011140001000A34DA30...'H }
mediaWaitForConnect FALSE canOverlapSend FALSE } h245Tunneling FALSE } } *Mar 1 14:22:21.141:
H225.0 OUTGOING ENCODE BUFFER::= 20 A0060008 914A0002 01400300 67007700 5F003308 80013C05
04010020 40000000 00000000 00000000 00000000 00000045 1C07000A 34DA302B 0A110000 00000000
00000000 00000000 00000032 02120000 000D4001 800A0400 01000A34 DA3041C5 1D400000 06040100
4D400180 11140001 000A34DA 3041C400 0A34DA30 41C50100 01000680 0100 *Mar 1 14:22:21.161: *Mar 1
14:22:21.417: H225.0 INCOMING ENCODE BUFFER::= 21 80060008 914A0002 00048811 00000000 00000000
00000000 00000000 00390219 0000000D 40018011 14000100 0A34DA2F 486E000A 34DA2F48 6F1D4000
00060401 004D4001 80111400 01000A34 DA3041C4 000A34DA 2F486F06 800100 *Mar 1 14:22:21.429: *Mar
1 14:22:21.429: H225.0 INCOMING PDU ::= !--- The GW_3 gets Call Proceeding from GW_1b. value
H323_UserInformation ::= { h323-uu-pdu { h323-message-body callProceeding : { protocolIdentifier
{ 0 0 8 2250 0 2 } destinationInfo { mc FALSE undefinedNode FALSE } callIdentifier { guid
'00000000000000000000000000000000'H } fastStart {
'0000000D40018011140001000A34DA2F486E000A...'H, '400000060401004D40018011140001000A34DA30...'H }
} h245Tunneling FALSE } } *Mar 1 14:22:21.617: H225.0 INCOMING ENCODE BUFFER::= 28 001A0006
0008914A 00020000 00000000 00000000 00000000 00000000 06A00100 120140B5 0000120B 60011000

011E041E 028188 *Mar 1 14:22:21.626: *Mar 1 14:22:21.626: **H225.0 INCOMING PDU** ::= *!--- The GW_3 gets Call Progress from GW_1b.* value H323_UserInformation ::= { h323-uu-pdu { h323-message-body **progress** : { protocolIdentifier { 0 0 8 2250 0 2 } destinationInfo { mc FALSE undefinedNode FALSE } callIdentifier { guid '00000000000000000000000000000000'H } } h245Tunneling FALSE nonStandardControl { { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '60011000011E041E028188' } } } } *Mar 1 14:22:21.642: H225 NONSTD INCOMING ENCODE BUFFER::= 60 01100001 1E041E02 8188 *Mar 1 14:22:21.646: *Mar 1 14:22:21.646: **H225 NONSTD INCOMING PDU** ::= *!--- The GW_3 get some facility messages from GW_1b.* value H323_UU_NonStdInfo ::= { version 16 protoParam **qsigNonStdInfo** : { iei 30 rawMesg '1E028188'H } } *Mar 1 14:22:22.831: %SYS-3-MGDTIMER: Running timer, init, timer = 81F1AC08. -Process= "Virtual Exec", ipl= 0, pid= 61 -Traceback= 803250A4 80325214 80325318 80EB12C0 80EB17DC 802A65F0 802B5080 8033D818 *Mar 1 14:22:22.835: H225 NONSTD OUTGOING PDU ::= value ARQnonStandardInfo ::= { sourceAlias { } sourceExtAlias { } } *Mar 1 14:22:22.839: H225 NONSTD OUTGOING ENCODE BUFFER::= 00 0000 *Mar 1 14:22:22.839: *Mar 1 14:22:22.839: **RAS OUTGOING PDU** ::= *!--- The GW_3 starts the second Call to 1711 now we send RAS message to GK.* value RasMessage ::= **admissionRequest** : { requestSeqNum 971 callType pointToPoint : NULL callModel direct : NULL endpointIdentifier {"8262B76400000019"} destinationInfo { e164 : "1711" } srcInfo { h323-ID : {"gw_3"} } bandwidth 640 callReferenceValue 24 nonStandardData { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '000000'H } conferenceID '00000000000000000000000000000000'H activeMC FALSE answerCall FALSE canMapAlias TRUE callIdentifier { guid '00000000000000000000000000000000'H } willSupplyUUIEs FALSE } *Mar 1 14:22:22.860: RAS OUTGOING ENCODE BUFFER::= 27 8803CA00 F0003800 32003600 32004200 37003600 34003000 30003000 30003000 30003100 39010180 4A440140 03006700 77005F00 33400280 001840B5 00001203 00000000 00000000 00000000 00000000 00000004 E0200180 11000000 00000000 00000000 00000000 00000100 *Mar 1 14:22:22.876: *Mar 1 14:22:22.940: RAS INCOMING ENCODE BUFFER::= 2B 0003CA40 0280000A 34DA3106 B800EF14 00C00100 020000 *Mar 1 14:22:22.944: *Mar 1 14:22:22.944: **RAS INCOMING PDU** ::= *!--- The GW_3 gets permission to proceed as there are no restrictions on zone3.* value RasMessage ::= **admissionConfirm** : { requestSeqNum 971 bandwidth 640 callModel direct : NULL destCallSignalAddress ipAddress : { ip '0A34DA31'H port 1720 } irrFrequency 240 willRespondToIRR FALSE uuiesRequested { setup FALSE callProceeding FALSE connect FALSE alerting FALSE information FALSE releaseComplete FALSE facility FALSE progress FALSE empty FALSE } } *Mar 1 14:22:22.972: **H225.0 OUTGOING PDU** ::= *!--- The GW_3 sends setup message to GW_1.* value H323_UserInformation ::= { h323-uu-pdu { h323-message-body **setup** : { protocolIdentifier { 0 0 8 2250 0 2 } sourceAddress { h323-ID : {"gw_3"} } sourceInfo { gateway { protocol { voice : { supportedPrefixes { { prefix e164 : "1#" } } } } } mc FALSE undefinedNode FALSE } activeMC FALSE conferenceID '00000000000000000000000000000000'H conferenceGoal create : NULL callType pointToPoint : NULL sourceCallSignalAddress ipAddress : { ip '0A34DA30'H port 11019 } callIdentifier { guid '00000000000000000000000000000000'H } fastStart { '0000000D4001800A040001000A34DA30402F'H, '400000060401004D40018011140001000A34DA30...'H } mediaWaitForConnect FALSE canOverlapSend FALSE } h245Tunneling FALSE } } *Mar 1 14:22:23.008: H225.0 OUTGOING ENCODE BUFFER::= 20 A0060008 914A0002 01400300 67007700 5F003308 80013C05 04010020 40000000 00000000 00000000 00000000 00000045 1C07000A 34DA302B 0B110000 00000000 00000000 00000000 00000032 02120000 000D4001 800A0400 01000A34 DA30402F 1D400000 06040100 4D400180 11140001 000A34DA 30402E00 0A34DA30 402F0100 01000680 0100 *Mar 1 14:22:23.028: *Mar 1 14:22:23.220: H225.0 INCOMING ENCODE BUFFER::= 25 80060008 914A0002 01110000 00000000 00000000 00000000 00000006 800100 *Mar 1 14:22:23.224: *Mar 1 14:22:23.224: **H225.0 INCOMING PDU** ::= *!--- The GW_1 replies with Release Complete message after asking GK !--- for permission to accept that call. !--- When the permission is denied, we set bandwidth limit.* value H323_UserInformation ::= { h323-uu-pdu { h323-message-body **releaseComplete** : { protocolIdentifier { 0 0 8 2250 0 2 } callIdentifier { guid '00000000000000000000000000000000'H } } h245Tunneling FALSE } } *Mar 1 14:22:23.236: **RAS OUTGOING PDU** ::= *!--- The GW_3 notifies GK that the call does not exist anymore.* value RasMessage ::= **disengageRequest** : { requestSeqNum 972 endpointIdentifier {"8262B76400000019"} conferenceID '00000000000000000000000000000000'H callReferenceValue 24 disengageReason normalDrop : NULL callIdentifier { guid '00000000000000000000000000000000'H } answeredCall FALSE } *Mar 1 14:22:23.248: RAS OUTGOING ENCODE BUFFER::= 3E 03CB1E00 38003200 36003200 42003700 36003400 30003000 30003000 30003000 31003900 00000000 00000000 00000000 00000000 18216111 00000000 00000000 00000000 00000000 000100 *Mar 1 14:22:23.256: *Mar 1 14:22:23.288: RAS INCOMING ENCODE BUFFER::= 40 03CB *Mar 1 14:22:23.288: *Mar 1 14:22:23.288: **RAS INCOMING PDU** ::= *!--- The GK confirms that message.* value RasMessage ::= **disengageConfirm** : { requestSeqNum 972 } ECV-2610-16#u all All possible debugging has been turned off ECV-2610-16# ----- *!--- The incoming RAS message to the GK from GW_3.* ECV-2610-15#debug h225 asnl H.225 ASN1 Messages debugging is on ECV-2610-15# *Mar 11 21:54:28.313: **RAS INCOMING PDU** ::= value RasMessage ::= **admissionRequest** : { requestSeqNum 970 callType pointToPoint : NULL callModel

direct : NULL endpointIdentifier {"8262B76400000019"} destinationInfo { e164 : "1511" } srcInfo { h323-ID : {"gw_3"} } bandwidth 640 callReferenceValue 23 nonStandardData { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '000000'H } conferenceID '00000000000000000000000000000000'H activeMC FALSE answerCall FALSE canMapAlias TRUE callIdentifier { guid '00000000000000000000000000000000'H } willSupplyUUIEs FALSE } *Mar 11 21:54:28.334: H225 NONSTD INCOMING ENCODE BUFFER::= 00 0000 *Mar 11 21:54:28.334: *Mar 11 21:54:28.334: H225 NONSTD INCOMING PDU ::= value ARQnonStandardInfo ::= { sourceAlias { } sourceExtAlias { } } *!--- The outgoing RAS message fro GK to GW_3 with permission to start call.* *Mar 11 21:54:28.338: **RAS OUTGOING PDU** ::= value RasMessage ::= **admissionConfirm** : { requestSeqNum 970 bandwidth 640 callModel direct : NULL destCallSignalAddress ipAddress : { ip '0A34DA2F'H port 1720 } irrFrequency 240 willRespondToIRR FALSE uuiesRequested { setup FALSE callProceeding FALSE connect FALSE alerting FALSE information FALSE releaseComplete FALSE facility FALSE progress FALSE empty FALSE } } *Mar 11 21:54:28.350: RAS OUTGOING ENCODE BUFFER::= 2B 0003C940 0280000A 34DA2F06 B800EF14 00C00100 020000 *Mar 11 21:54:28.354: *Mar 11 21:54:28.446: H225.0 INCOMING ENCODE BUFFER::= 20 A0060008 914A0002 01400300 67007700 5F003308 80013C05 04010020 40000000 00000000 00000000 00000000 00000045 1C07000A 34DA302B 0A110000 00000000 00000000 00000000 00000032 02120000 000D4001 800A0400 01000A34 DA3041C5 1D400000 06040100 4D400180 11140001 000A34DA 3041C400 0A34DA30 41C50100 01000680 0100 *Mar 11 21:54:28.466: *Mar 11 21:54:28.470: **H225.0 INCOMING PDU** ::= *!--- The incoming H323(Q931) message from GW_3 to GW_1b on the same router as GK.* value H323_UserInformation ::= { h323-uu-pdu { h323-message-body **setup** : { protocolIdentifier { 0 0 8 2250 0 2 } sourceAddress { h323-ID : {"gw_3"} } sourceInfo { gateway { protocol { voice : { supportedPrefixes { { prefix e164 : "1#" } } } } } mc FALSE undefinedNode FALSE } activeMC FALSE conferenceID '00000000000000000000000000000000'H conferenceGoal create : NULL callType pointToPoint : NULL sourceCallSignalAddress ipAddress : { ip '0A34DA30'H port 11018 } callIdentifier { guid '00000000000000000000000000000000'H } fastStart { '0000000D4001800A040001000A34DA3041C5'H, '400000060401004D40018011140001000A34DA30...'H } mediaWaitForConnect FALSE canOverlapSend FALSE } h245Tunneling FALSE } } *Mar 11 21:54:28.514: H225 NONSTD OUTGOING PDU ::= value ARQnonStandardInfo ::= { sourceAlias { } sourceExtAlias { } } *Mar 11 21:54:28.518: H225 NONSTD OUTGOING ENCODE BUFFER::= 00 0000 *Mar 11 21:54:28.518: *Mar 11 21:54:28.518: **RAS OUTGOING PDU** ::= *!--- The GW_1b asks GK if it can accept call from GW_3.* value RasMessage ::= **admissionRequest** : { requestSeqNum 1347 callType pointToPoint : NULL callModel direct : NULL endpointIdentifier {"82717F5C0000001B"} destinationInfo { e164 : "1511" } srcInfo { h323-ID : {"gw_3"} } srcCallSignalAddress ipAddress : { ip '0A34DA30'H port 11018 } bandwidth 640 callReferenceValue 29 nonStandardData { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '000000'H } conferenceID '00000000000000000000000000000000'H activeMC FALSE answerCall TRUE canMapAlias TRUE callIdentifier { guid '00000000000000000000000000000000'H } willSupplyUUIEs FALSE } *Mar 11 21:54:28.542: RAS OUTGOING ENCODE BUFFER::= 27 98054200 F0003800 32003700 31003700 46003500 43003000 30003000 30003000 30003100 42010180 48440140 03006700 77005F00 33000A34 DA302B0A 40028000 1D40B500 00120300 00000000 00000000 00000000 00000000 000044E0 20018011 00000000 00000000 00000000 00000000 000100 *Mar 11 21:54:28.558: *Mar 11 21:54:28.562: RAS INCOMING ENCODE BUFFER::= 27 98054200 F0003800 32003700 31003700 46003500 43003000 30003000 30003000 30003100 42010180 48440140 03006700 77005F00 33000A34 DA302B0A 40028000 1D40B500 00120300 00000000 00000000 00000000 00000000 000044E0 20018011 00000000 00000000 00000000 00000000 000100 *Mar 11 21:54:28.578: *Mar 11 21:54:28.582: **RAS INCOMING PDU** ::= *!--- That is the same RAS message. The GK gets it, and sees the sequence number. !--- The GK is on the same router as GW_1b, so all messages can be seen twice.* value RasMessage ::= **admissionRequest** : { requestSeqNum 1347 callType pointToPoint : NULL callModel direct : NULL endpointIdentifier {"82717F5C0000001B"} destinationInfo { e164 : "1511" } srcInfo { h323-ID : {"gw_3"} } srcCallSignalAddress ipAddress : { ip '0A34DA30'H port 11018 } bandwidth 640 callReferenceValue 29 nonStandardData { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '000000' } conferenceID '00000000000000000000000000000000'H activeMC FALSE answerCall TRUE canMapAlias TRUE callIdentifier { guid '00000000000000000000000000000000'H } willSupplyUUIEs FALSE } *Mar 11 21:54:28.606: H225 NONSTD INCOMING ENCODE BUFFER::= 00 0000 *Mar 11 21:54:28.606: *Mar 11 21:54:28.606: H225 NONSTD INCOMING PDU ::= value ARQnonStandardInfo ::= { sourceAlias { } sourceExtAlias { } } *Mar 11 21:54:28.610: **RAS OUTGOING PDU** ::= *!--- The GK grants the permission to GW_1b. !--- This is a message in the GK debug outgoing* value RasMessage ::= **admissionConfirm** : { requestSeqNum 1347 bandwidth 640 callModel direct : NULL destCallSignalAddress ipAddress : { ip '0A34DA2F'H port 1720 } irrFrequency 240 willRespondToIRR FALSE uuiesRequested { setup FALSE callProceeding FALSE connect FALSE alerting FALSE information FALSE releaseComplete FALSE facility FALSE progress FALSE empty FALSE } } *Mar 11 21:54:28.622: RAS OUTGOING ENCODE BUFFER::= 2B 00054240 0280000A 34DA2F06 B800EF14 00C00100 020000 *Mar 11 21:54:28.626: *Mar 11 21:54:28.630: RAS INCOMING ENCODE BUFFER::= 2B 00054240

0280000A 34DA2F06 B800EF14 00C00100 020000 *Mar 11 21:54:28.634: *Mar 11 21:54:28.634: **RAS INCOMING PDU** ::= *!--- The GK grants the permission to GW_1b. !---* This is a message in the GW_1b debug incoming. value RasMessage ::= **admissionConfirm** : { requestSeqNum 1347 bandwidth 640 callModel direct : NULL destCallSignalAddress ipAddress : { ip '0A34DA2F'H port 1720 } irrFrequency 240 willRespondToIRR FALSE uuiesRequested { setup FALSE callProceeding FALSE connect FALSE alerting FALSE information FALSE releaseComplete FALSE facility FALSE progress FALSE empty FALSE } } *Mar 11 21:54:28.654: %SYS-3-MGDTIMER: Timer has parent, timer link, timer = 820AE990. -Process= "CC-API_VCM", ip1= 6, pid= 93 -Traceback= 80325850 8032A720 80E74850 8033D818 *Mar 11 21:54:28.666: **H225.0 OUTGOING PDU** ::= *!--- The GW_1b replies to GW_3 setup message.* value H323_UserInformation ::= { h323-uu-pdu { h323-message-body **callProceeding** : { protocolIdentifier { 0 0 8 2250 0 2 } destinationInfo { mc FALSE undefinedNode FALSE } callIdentifier { guid '00000000000000000000000000000000'H } fastStart { '0000000D40018011140001000A34DA2F486E000A...H, '400000060401004D40018011140001000A34DA30...H } } h245Tunneling FALSE } } *Mar 11 21:54:28.682: H225.0 OUTGOING ENCODE BUFFER::= 21 80060008 914A0002 00048811 00000000 00000000 00000000 00000000 00390219 0000000D 40018011 14000100 0A34DA2F 486E000A 34DA2F48 6F1D4000 00060401 004D4001 80111400 01000A34 DA3041C4 000A34DA 2F486F06 800100 *Mar 11 21:54:28.694: *Mar 11 21:54:28.710: H225 NONSTD OUTGOING PDU ::= value H323_UU_NonStdInfo ::= { version 16 protoParam qsigNonStdInfo : { iei 30 rawMesg '1E028188'H } } *Mar 11 21:54:28.714: H225 NONSTD OUTGOING ENCODE BUFFER::= 60 01100001 1E041E02 8188 *Mar 11 21:54:28.714: *Mar 11 21:54:28.714: **H225.0 OUTGOING PDU** ::= *!--- The GW_1b replies to GW_3 setup message and sends second message.* value H323_UserInformation ::= { h323-uu-pdu { h323-message-body **progress** : { protocolIdentifier { 0 0 8 2250 0 2 } destinationInfo { mc FALSE undefinedNode FALSE } callIdentifier { guid '00000000000000000000000000000000'H } } h245Tunneling FALSE nonStandardControl { { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '60011000011E041E028188'H } } } } *Mar 11 21:54:28.734: H225.0 OUTGOING ENCODE BUFFER::= 28 001A0006 0008914A 00020000 00000000 00000000 00000000 00000000 06A00100 120140B5 0000120B 60011000 011E041E 028188 *Mar 11 21:54:28.742: *Mar 11 21:54:30.161: RAS INCOMING ENCODE BUFFER::= 27 8803CA00 F0003800 32003600 32004200 37003600 34003000 30003000 30003000 30003100 39010180 4A440140 03006700 77005F00 33400280 001840B5 00001203 00000000 00000000 00000000 00000000 00000004 E0200180 11000000 00000000 00000000 00000000 00000100 *Mar 11 21:54:30.177: *Mar 11 21:54:30.181: **RAS INCOMING PDU** ::= *!--- The GK gets ARQ from GW_3 for the second call.* value RasMessage ::= **admissionRequest**: { requestSeqNum 971 callType pointToPoint : NULL callModel direct : NULL endpointIdentifier {"8262B76400000019"} destinationInfo { e164 : "1711" } srcInfo { h323-ID : {"gw_3"} } bandwidth 640 callReferenceValue 24 nonStandardData { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '000000'H } conferenceID '00000000000000000000000000000000'H activeMC FALSE answerCall FALSE canMapAlias TRUE callIdentifier { guid '00000000000000000000000000000000'H } willSupplyUUIEs FALSE } *Mar 11 21:54:30.197: H225 NONSTD INCOMING ENCODE BUFFER::= 00 0000 *Mar 11 21:54:30.201: *Mar 11 21:54:30.201: H225 NONSTD INCOMING PDU ::= value ARQnonStandardInfo ::= { sourceAlias { } sourceExtAlias { } } *Mar 11 21:54:30.205: **RAS OUTGOING PDU** ::= *!--- The GK grants permission to GW_3, as there are no restrictions for zone3.* value RasMessage ::= **admissionConfirm** : { requestSeqNum 971 bandwidth 640 callModel direct : NULL destCallSignalAddress ipAddress : { ip '0A34DA31'H *!--- The hexadecimal number is 10.52.218.49, IP of GW_1.* port 1720 } irrFrequency 240 willRespondToIRR FALSE uuiesRequested { setup FALSE callProceeding FALSE connect FALSE alerting FALSE information FALSE releaseComplete FALSE facility FALSE progress FALSE empty FALSE } } *Mar 11 21:54:30.217: RAS OUTGOING ENCODE BUFFER::= 2B 0003CA40 0280000A 34DA3106 B800EF14 00C00100 020000 *Mar 11 21:54:30.221: *Mar 11 21:54:30.429: RAS INCOMING ENCODE BUFFER::= 27 98045F00 F0003800 32003300 38003600 30004400 34003000 30003000 30003000 30003100 41010180 4A440140 03006700 77005F00 33000A34 DA302B0B 40028000 2840B500 00120300 00000000 00000000 00000000 00000000 000044E0 20018011 00000000 00000000 00000000 00000000 000100 *Mar 11 21:54:30.445: *Mar 11 21:54:30.445: **RAS INCOMING PDU** ::= *!--- The incoming request from GW_1 asks for permission to accept call from GW_3.* value RasMessage ::= **admissionRequest** : { requestSeqNum **1120** callType pointToPoint : NULL callModel direct : NULL endpointIdentifier {"823860D40000001A"} destinationInfo { e164 : "**1711**" } srcInfo { h323-ID : {"gw_3"} } srcCallSignalAddress ipAddress : { ip '0A34DA30'H port 11019 } bandwidth 640 callReferenceValue 40 nonStandardData { nonStandardIdentifier h221NonStandard : { t35CountryCode 181 t35Extension 0 manufacturerCode 18 } data '000000'H } conferenceID '00000000000000000000000000000000'H activeMC FALSE answerCall TRUE canMapAlias TRUE callIdentifier { guid '00000000000000000000000000000000'H } willSupplyUUIEs FALSE } *Mar 11 21:54:30.469: H225 NONSTD INCOMING ENCODE BUFFER::= 00 0000 *Mar 11 21:54:30.469: *Mar 11 21:54:30.469: H225 NONSTD INCOMING PDU ::= value ARQnonStandardInfo ::= { sourceAlias { } sourceExtAlias { } } *!--- The GK does not allow the call to come through, and replies with ARJ.* *Mar 11 21:54:30.473: **RAS OUTGOING PDU** ::= value RasMessage ::= **admissionReject** : { requestSeqNum **1120** rejectReason requestDenied : NULL } *Mar

- [Устранение неполадок в системах IP-телефонии Cisco](#)
- [Техническая поддержка - Cisco Systems](#)