Пример конфигурации TLS SIP Unified Border Element

Содержание

Введение
Предварительные условия
Требования
Используемые компоненты
Условные обозначения
Настройка
Схема сети
RFC Support для TLS в CUBE
Порядок действий для настройки
Примечания реализации TLS
Примеры конфигураций
Проверка
Устранение неполадок
Дополнительные сведения

Введение

Cisco Unified Border Element (CUBE) поддерживает Протокол SIP к вызовам SIP с Transport Layer Security (TLS). TLS предоставляет конфиденциальность и целостность данных сообщений о передаче сигнала SIP между двумя приложениями, которые связываются. TLS разделен на уровни поверх надежного транспортного протокола, такого как TCP.

TLS на CUBE может быть настроен на основе на участок для разрешения TLS вызову SIP не-TLS. В то время как участок SIP использует TLS, точно так же CUBE использует IPSec для обеспечения сигнализации и поддержек вызова от H.323 до SIP с участком H.323.

Предварительные условия

Требования

Убедитесь, что вы обеспечили выполнение следующих требований, прежде чем попробовать эту конфигурацию:

- Базовые знания о том, как настроить и использовать Обмен голосовыми данными с помощью Cisco IOS (такой как точки вызова)
- Базовые знания о том, как настроить и использовать CUBE
- Знакомство с понятиями базовых мер безопасности, такими как шифрование,

сертификация, центры сертификации, PKI (ключи) и аутентификация

Используемые компоненты

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

- CUBE освобождают на ISR, который использует Cisco IOS Release 12.4T
- Маршрутизатор Cisco IOS, настроенный как центр сертификации (CA)

Сведения, представленные в этом документе, были получены от устройств, работающих в специальной лабораторной среде. Все устройства, описанные в этом документе, были запущены с чистой (стандартной) конфигурацией. В рабочей сети необходимо изучить потенциальное воздействие всех команд до их использования.

Условные обозначения

Более подробную информацию о применяемых в документе обозначениях см. в описании условных обозначений, используемых в технической документации Cisco.

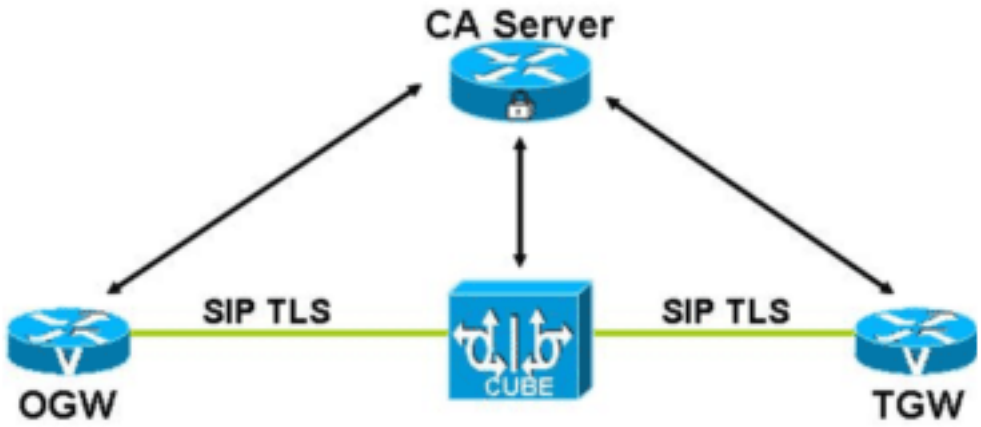
Настройка

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

Примечание: Используйте инструмент Command Lookup (только для зарегистрированных пользователей) для того, чтобы получить более подробную информацию о командах, использованных в этом разделе.

Схема сети

Эти данные показывают пример CUBE с TLS подключение SIP.



- Исходный шлюз (OGW), конечный шлюз (TGW) и устройства CUBE аутентифицируется и регистрируется с сервером CA. Сертификаты подписаны сервером CA.
- Когда вызов выполнен, подтверждение связи TLS инициируется между устройствами (например, OGW и CUBE), и инфраструктура PKI IOS используется для обмена сертификатами, подписанными общим доверяемым CA во время квитирования.

- Во время подтверждения связи TLS о динамично генерируемом симметричном ключе и алгоритмах шифра выполняют согласование между устройствами.
- После того, как подтверждение связи TLS успешно, устройства устанавливают сеанс SIP между ними. Ключи, которыми обмениваются во время процесса подтверждения связи TLS, используются, чтобы зашифровать или дешифровать все сообщения о передаче сигнала SIP. Схема URI “sip”: используется для сообщений TLS SIP.

RFC Support для TLS в CUBE

Наборы шифров, требуемые для TLS согласно SIP RFC 3261, включают:

- (Обязательный) TLS_RSA_WITH_AES_128_CBC_SHA
- (Дополнительный) TLS_RSA_WITH_3DES_EDE_CBC_SHA — Требуемый для серверов сети (таких как прокси и серверы перенаправления для обратной совместимости)

Только комплект TLS_RSA_WITH_AES_128_CBC_SHA применим к CUBE и поддерживается. Точно так же реализация TLS в CUBE поддерживает только обязательные наборы шифров RFC 2246.

Протокол SIP использует одноранговую модель. Therfore, CUBE может быть или сервером или клиентом TLS подключение и внедряет обе стороны. CUBE всегда выполняет обоюдную проверку подлинности, когда это - сторона сервера.

Порядок действий для настройки

Настройте сервер CA

Можно использовать эту команду в режиме глобальной конфигурации для настройки маршрутизатора Cisco IOS, загруженного криптографическим образом:

```
router(config)#crypto pki server <ca-server-name> router(cs-server)#no shutdown
```

Примечания:

- Используйте **команду ip http server** в режиме глобальной конфигурации, чтобы гарантировать, что сервер HTTP работает на маршрутизаторе, настроенном как сервер CA. Это требуется начиная с клиентских точек доверия (CUBE/OGW/TGW) HTTP использования для получения сертификатов от сервера CA.
- Часы в сервере CA и клиентских точках доверия (CUBE/OGW/TGW) должны синхронизироваться. В противном случае могли бы быть проблемы с законностью сертификатов, выполненных сервером CA. Можно использовать **команды show clock и clock set** для синхронизации часов на маршрутизаторах Cisco IOS. Также можно развернуть сервер NTP для синхронизации часов.

Базовая конфигурация для CUBE

Используйте эти команды для включения функциональных возможностей шлюза IP-to-IP CUBE. Это позволяет terminatation входящего вызова VoIP и reoriginatiion вызова с внешним одноранговым узлом набор IP - телефонии.

```
voice service voip
  allow-connections h323 to sip
  allow-connections sip to h323
```

```
allow-connections sip to sip
allow-connections h323 to h323
```

Конфигурация TLS

Выполните эти шаги для настройки TLS на CUBE (и другие устройства как OGW и TGW):

1. **Генерируйте криптографическую пару RSA**Используйте эту команду в режиме глобальной конфигурации для генерации криптографической пары RSA:`router(config)#`**`crypto key generate rsa general-keys label <`***`label`***`> modulus 1024`**
2. **Создайте точку доверия PKI (CUBE)**Используйте эту команду в режиме глобальной конфигурации для создания точки доверия PKI (CUBE):`router(config)#`**`crypto pki trustpoint <`***`ca-server-name`***`>`** `router(ca-trustpoint)#`**`enrollment url <`***`http://ca-server-ip`***`>`** `router(ca-trustpoint)#`**`rsakeypair <`***`rsa keypair label`***`>`**
3. **Аутентифицируйте точку доверия PKI (CUBE) с сервером CA**Используйте эту команду в режиме глобальной конфигурации для аутентификации точки доверия PKI (CUBE) с сервером CA:`router(config)#`**`crypto pki authenticate <`***`ca-server-name`***`>`** Этот шаг инициирует сервер CA для передачи его сертификата к точке доверия (CUBE), который должен быть принят.
4. **Зарегистрируйте точку доверия PKI (CUBE) с Сервером CA**Используйте эту команду в режиме глобальной конфигурации:`router(config)#`**`crypto pki enroll <`***`ca-server-name`***`>`** Для этого шага необходимо ввести пароль вызова. Неполадки сервера CA два сертификата к точке доверия (CUBE): один, чтобы сертифицировать, что сервер CA и другой сертифицируют точку доверия (CUBE). Можно проверить сертификаты с **командой show run**.
5. **Настройте TLS как транспортный протокол сеанса**Транспортный протокол сеанса может быть настроен к TLS с **командой session transport tcp tls** или на глобальном уровне под “voip голосового сервиса” или в соответствующих VoIP одноранговых соединение.Если транспортный протокол сеанса настроен для VoIP однорангового соединение (поступление или выход или оба), то транспорт TLS используется только для настроенного участка. Транспорт TLS поддерживается на основе от участка к участку.
6. **Настройте точку доверия по умолчанию для UA SIP**Используйте эту команду в режиме “sip-ua” для настройки точки доверия по умолчанию для UA SIP:`router(config-sip-ua)#`**`[`***`no`***`] crypto signaling [(`***`remote-addr subnet mask`***`) | default] trustpoint <`***`label`***`> [`***`strict-cipher`***`]`** Метка точки доверия обращается к сертификату CUBE, который генерируется с командами Cisco IOS PKI как часть proces регистрации. *строгий шифр* означает, что процесс TLS SIP использует только те наборы шифров, которые получают мандат RFC SIP.В настоящее время RFC 3261 задает комплекты TLS_RSA_WITH_3DES_EDE_CBC_SHA и TLS_RSA_WITH_AES_128_CBC_SHA. То, когда вы используете аргумент команды *строгого шифра*, избегает изменений к конфигурации, если SIP должен передать под мандат более новые шифры.Уровень SSL в Cisco IOS не поддерживает TLS_RSA_WITH_3DES_EDE_CBC_SHA. Поэтому CUBE активно использует только комплект TLS_RSA_WITH_AES_128_CBC_SHA в строгом режиме. Когда *строгий шифр* не задан, процесс TLS SIP использует больший набор шифров в зависимости от поддержки на уровне SSL.*Пример 1*Команда ниже настраивает CUBE для использования его метки точки доверия **mylabel**, когда это устанавливает или принимает TLS подключение с удаленным устройством в 1.2.3.0 подсетях. Набор шифров в этом случае является полным набором, который поддерживается уровнем SSL на CUBE.

`crypto signaling remote-addr 1.2.3.0 255.255.255.0 trustpoint mylabel` *Пример 2*Команда ниже настраивает CUBE для использования его метки точки доверия **повар**, когда это устанавливает или принимает TLS подключение с любым удаленным устройством, пока не совпадают с отдельной конфигурацией метки подсети.
`crypto signaling default trustpoint chef` *Пример 3*Команда ниже настраивает CUBE для использования его метки точки доверия **mylabel**, когда это устанавливает или принимает TLS подключение с удаленным устройством в 1.2.3.0 подсетях. Набор шифров, используемый во время подтверждения связи TLS, ограничен комплектом TLS_RSA_WITH_AES_128_CBC_SHA.
`crypto signaling remote-addr 1.2.3.0 255.255.255.0 trustpoint mylabel strict-cipher`

7. **Включение порта прослушивания TLS**Выполните эту команду в режиме “sip-ua”, чтобы позволить порту TLS на TCP 5061 слушать:
`transport tcp tls`
8. **Схема URL SIP Настройки**“Sip”: схема URL может быть настроена или под уровнем узла коммутации VoIP или на глобальном уровне. Эта команда используется для настройки “sip”: в VoIP одноранговом соединение:
`voice-class sip url sips` Для настройки “sip”: схема URL под глобальным уровнем, используйте эту команду в режиме “sip” “voip голосового сервиса”:
`voice service voip sip url sips` Использование URL SIP требует, чтобы все переходы в сигнальном пути использовали TLS и SIP. Это становится важным для SRTP, как ключи находятся в SDP и для безопасного соединения, которое информация не должна быть передана в открытом тексте. Если прокси получает INVITE с SIP (например, INVITE sips:123@proxy SIP/2.0), прокси должен использовать SIP для следующего перехода. Когда TLS используется с простым адресом URL SIP, нет никакой гарантии, что все переходы будут использовать TLS, потенциально ставя под угрозу сквозную безопасность вызова.Если URL “sip” будет настроен, то транспорт автоматически будет TLS.

Примечания реализации TLS

- Когда безопасные среды настроены (SRTP), текущая операция CUBE требует использования TLS как транспорт. Будущее усовершенствование может снять это требование.
- Когда SRTP настроен для обеспечения соединения сред, или TLS или IPSec *должны* также быть настроены для обеспечения сообщений о передаче сигнала SIP. Ключами, используемыми для шифрования SRTP, обмениваются с помощью сообщений о передаче сигнала – не обеспечение результатов канала сигнализации в ключах SRTP, которыми обмениваются в открытом тексте, и это инвертирует безопасность SRTP для соединения сред.
- Текущая операция CUBE требует использования “sip”: схема URI вызова TLS. Будущее усовершенствование может снять это требование.
- Текущая операция CUBE была проверена с одиночным сервером CA только.

Примеры конфигураций

CUBE

ipipgw

```
ipipgw#show run Building configuration... Current
configuration : 5096 bytes ! version 12.4 service
timestamps debug datetime msec service timestamps log
datetime msec no service password-encryption ! hostname
ipipgw ! boot-start-marker boot system flash c3845-
adventerprisek9_ivs-mz.124-3.9.PI3a boot-end-marker !
logging buffered 10000000 debugging no logging console !
no aaa new-model ! resource policy ! ip subnet-zero ip
cef ! no ip domain lookup ! voice-card 0 no dspfarm !
voice service voip allow-connections sip to sip sip url
sips ! crypto pki trustpoint ca-server enrollment url
http://9.13.46.14:80 serial-number revocation-check crl
rsakeypair kkp ! crypto pki certificate chain ca-server
certificate 04 3082020D 30820176 A0030201 02020104
300D0609 2A864886 F70D0101 04050030 14311230 10060355
04031309 63612D73 65727665 72301E17 0D303530 39323231
37333435 315A170D 30363039 32323137 33343531 5A303431
32300F06 03550405 13084337 33323231 3333301F 06092A86
4886F70D 01090216 1270696E 612D3338 34352D69 70697067
77312E30 819F300D 06092A86 4886F70D 01010105 0003818D
00308189 02818100 BBCC2977 637E8E42 17EB7C26 FB2BA0A3
6E1ECECB E01A64F8 8F18200F 9837E4FA 7D908B3C 1297A4DE
A403D315 C7BB96C6 50D95291 0433FA7B CB8FFFFD 8FC1C211
CCC7BCA9 140FF942 C3ACF4BC 3EDCE2DC 28FCEA87 AA83629F
D217F833 A727940A 0BBB8624 3EA9D1EC 1F69228F E1DFC113
243246B7 BF57696C 2278F5C3 674EE0E1 02030100 01A34F30
4D300B06 03551D0F 04040302 05A0301F 0603551D 23041830
16801486 7414D5D6 9B8299C1 787211AB 1B265B06 D2B62D30
1D060355 1D0E0416 0414FED1 97051946 D2F870D8 0DE819C3
AA1F3830 AD35300D 06092A86 4886F70D 01010405 00038181
00845AB8 F6589AED 17D0BB10 2AEA48AA 9299C130 4B358EA1
96632C84 0387D2DE 4774C776 6A14F25B 5D062E12 45EF730D
27D45795 62C17F55 A0428259 B13669BC 022201C7 EB6B7ACF
4C7143FA 8A038301 CEA17A0B D0662887 26BA8F0E C44410BB
4F982706 11F0D248 77D8A0E5 4417F0F4 3F993CE3 F62F6BDE
BA2DD6BB B843391D 6D quit certificate ca 01 30820201
3082016A A0030201 02020101 300D0609 2A864886 F70D0101
04050030 14311230 10060355 04031309 63612D73 65727665
72301E17 0D303530 39323031 37303335 375A170D 30383039
31393137 30333537 5A301431 12301006 03550403 13096361
2D736572 76657230 819F300D 06092A86 4886F70D 01010105
0003818D 00308189 02818100 BE7F0760 70D3B5C3 923D59FB
C10AED17 71C6F477 7580851A 282FFAEB 43B918A1 2D867C1B
63963B36 F779FE18 D5DFFDB6 5E436276 459FC5EA A729C386
CDDD922B 2A0439AE 68A5F4C4 3B05F168 5BB93EF2 DF737F11
0BA3F5EB 3E62F423 CB5364D3 C39CCA09 8ADECBFF 4C0515A6
0750A283 ABA39ED2 F5866B98 D3361C1A B88AA62B 02030100
01A36330 61300F06 03551D13 0101FF04 05300301 01FF300E
0603551D 0F0101FF 04040302 0186301F 0603551D 23041830
16801486 7414D5D6 9B8299C1 787211AB 1B265B06 D2B62D30
1D060355 1D0E0416 04148674 14D5D69B 8299C178 7211AB1B
265B06D2 B62D300D 06092A86 4886F70D 01010405 00038181
00AC7DAF 0DF589CA C6175EC0 8F976C5F E08C3C91 85282FFA
94EE6F30 02EEE5B9 E60198ED 643151E0 CCE192FA A352BA3D
8BC5C006 EF89CFCF 59DA9B12 D729102C 3D6ADC3C 09931B96
3F1FB48C C0A85FDB 4F9A7C16 028673C3 91786D57 9D7C1016
62F9D4E9 78FED276 0C404815 B1FE3A11 4D215FCF 573536B4
477ECDB7 7060E221 31 quit ! interface GigabitEthernet0/0
ip address 9.13.46.12 255.255.255.0 duplex auto speed
auto media-type rj45 negotiation auto ! interface
GigabitEthernet0/1 no ip address shutdown duplex auto
speed auto media-type rj45 negotiation auto ! ip
classless ip route 0.0.0.0 0.0.0.0 9.13.46.1 ! ip http
server no ip http secure-server ! no cdp log mismatch
```

```
duplex ! control-plane ! call treatment on ! dial-peer
voice 1 voip session protocol sipv2 incoming called-
number 9000 codec g711ulaw ! dial-peer voice 2 voip
destination-pattern 9000 session protocol sipv2 session
target ipv4:9.13.46.200 codec g711ulaw ! dial-peer voice
3 voip session protocol sipv2 incoming called-number
4000 codec g711ulaw ! dial-peer voice 4 voip
destination-pattern 4000 session protocol sipv2 session
target ipv4:9.13.32.75 codec g711ulaw ! dial-peer voice
5 voip destination-pattern 5000 session protocol sipv2
session target ipv4:9.13.0.10 codec g711alaw ! dial-peer
voice 7 voip destination-pattern 9999 session protocol
sipv2 session target ipv4:9.13.2.36 codec g711alaw !
dial-peer voice 12 pots destination-pattern 8400 ! dial-
peer voice 10 voip destination-pattern 50000 session
protocol sipv2 session target ipv4:9.13.2.150 codec
g711alaw ! dial-peer voice 11 voip session protocol
sipv2 session transport tcp tls incoming called-number
8004 codec g711ulaw ! dial-peer voice 13 voip
destination-pattern 8004 session protocol sipv2 session
target ipv4:9.13.2.70 codec g711ulaw ! dial-peer voice
20 voip destination-pattern 4444 session target
ipv4:9.13.46.111 codec g711ulaw ! dial-peer voice 21
voip incoming called-number 4444 codec g711ulaw ! sip-ua
retry invite 10 crypto signaling default trustpoint ca-
server ! gatekeeper shutdown ! line con 0 stopbits 1
line aux 0 stopbits 1 line vty 0 4 login ! scheduler
allocate 20000 1000 ! end
```

Сервер CA IOS

ca-server

```
ca-server#show run Building configuration... Current
configuration : 2688 bytes ! ! Last configuration change
at 17:11:41 UTC Tue Sep 20 2005 ! NVRAM config last
updated at 16:57:43 UTC Tue Sep 20 2005 ! version 12.4
service timestamps debug datetime msec service
timestamps log datetime msec no service password-
encryption ! hostname ca-server ! boot-start-marker boot
system flash c2800nm-adventerprisek9_ivs-mz.124-3.9.PI3a
boot-end-marker ! no aaa new-model ! resource policy !
ip subnet-zero ! ip cef ! voice-card 0 no dspfarm !
crypto pki server ca-server grant auto ! crypto pki
trustpoint ca-server revocation-check crl rsakeypair ca-
server ! crypto pki certificate chain ca-server
certificate ca 01 30820201 3082016A A0030201 02020101
300D0609 2A864886 F70D0101 04050030 14311230 10060355
04031309 63612D73 65727665 72301E17 0D303530 39323031
37303335 375A170D 30383039 31393137 30333537 5A301431
12301006 03550403 13096361 2D736572 76657230 819F300D
06092A86 4886F70D 01010105 0003818D 00308189 02818100
BE7F0760 70D3B5C3 923D59FB C10AED17 71C6F477 7580851A
282FFAEB 43B918A1 2D867C1B 63963B36 F779FE18 D5DFFDB6
5E436276 459FC5EA A729C386 CDDD922B 2A0439AE 68A5F4C4
3B05F168 5BB93EF2 DF737F11 0BA3F5EB 3E62F423 CB5364D3
C39CCA09 8ADECBFF 4C0515A6 0750A283 ABA39ED2 F5866B98
D3361C1A B88AA62B 02030100 01A36330 61300F06 03551D13
0101FF04 05300301 01FF300E 0603551D 0F0101FF 04040302
0186301F 0603551D 23041830 16801486 7414D5D6 9B8299C1
787211AB 1B265B06 D2B62D30 1D060355 1D0E0416 04148674
14D5D69B 8299C178 7211AB1B 265B06D2 B62D300D 06092A86
4886F70D 01010405 00038181 00AC7DAF 0DF589CA C6175EC0
```

```
8F976C5F E08C3C91 85282FFA 94EE6F30 02EEE5B9 E60198ED
643151E0 CCE192FA A352BA3D 8BC5C006 EF89CFCF 59DA9B12
D729102C 3D6ADC3C 09931B96 3F1FB48C C0A85FDB 4F9A7C16
028673C3 91786D57 9D7C1016 62F9D4E9 78FED276 0C404815
B1FE3A11 4D215FCF 573536B4 477ECDB7 7060E221 31 quit !
interface FastEthernet0/0 ip address 9.13.46.14
255.255.255.0 duplex auto speed auto ! interface
FastEthernet0/1 no ip address shutdown duplex auto speed
auto ! ip classless ip route 0.0.0.0 0.0.0.0 9.13.46.1 !
ip http server no ip http secure-server ! no cdp log
mismatch duplex ! control-plane ! gatekeeper shutdown !
line con 0 line aux 0 line vty 0 4 login ! scheduler
allocate 20000 1000 ! end
```

Проверка

После того, как вызов выполнен, эта **команда показа** может использоваться, чтобы проверить, является ли транспорт, используемый для вызова, TLS:

```
router#show sip-ua connections tcp tls ? brief Show summary of connections detail Show detail
connection information
```

Пример выходных данных для этой команды показывают в этих примерах:

Пример 1: Подробные выходные данные

```
==========================================================================
router#show sip-ua connections tcp tls detail Total active connections : 1 No. of send failures
: 0 No. of remote closures : 3 No. of conn. failures : 0 No. of inactive conn. ageouts : 0 Max.
tls send msg queue size of 0, recorded for 0.0.0.0:0 TLS client handshake failures : 0 TLS
server handshake failures : 0 ---------Printing Detailed Connection Report--------- Note: **
Tuples with no matching socket entry - Do 'clear sip <tcp[tls]/udp> conn t ipv4:<addr>:<port>'
to overcome this error condition ++ Tuples with mismatched address/port entry - Do 'clear sip
<tcp[tls]/udp> conn t ipv4:<addr>:<port> id <connid>' to overcome this error condition Remote-
Agent:9.13.46.12, Connections-Count:1 Remote-Port Conn-Id Conn-State WriteQ-Size ===========
======= =========== =========== 5061 1 Established 0
==========================================================================
```

Пример 2: Краткие выходные данные

```
==========================================================================
router#show sip-ua connections tcp tls brief Total active connections : 2 No. of send failures :
0 No. of remote closures : 0 No. of conn. failures : 0 No. of inactive conn. ageouts : 0 Max.
tls send msg queue size of 0, recorded for 0.0.0.0:0 TLS client handshake failures : 0 TLS
server handshake failures : 0
==========================================================================
```

Также **команда debug ccsip messages** может использоваться для проверки “Через”: заголовок для TLS включен. Эти выходные данные являются типовым запросом INVITE вызова, который использует TLS SIP и “sip”: схема URI:

```
INVITE sips:777@172.18.203.181 SIP/2.0
Via: SIP/2.0/TLS 172.18.201.173:5060;branch=z9hG4bK2C419
From: <sips:333@172.18.201.173>;tag=581BB98-1663
To: <sips:5555555@172.18.197.154>
Date: Wed, 28 Dec 2005 18:31:38 GMT
Call-ID: EB5B1948-770611DA-804F9736-BFA4AC35@172.18.201.173
Remote-Party-ID: "Bob" <sips:+14085559999@1.2.3.4>
Contact: <sips:123@host>
Allow: INVITE, OPTIONS, BYE, CANCEL, ACK, PRACK, COMET, REFER, SUBSCRIBE, NOTIFY, INFO
Max-Forwards: 70
Cseq: 104 INVITE
```

```
Expires: 60
Timestamp: 730947404
Content-Length: 298
Content-Type: application/sdp

v=0
o=CiscoSystemsSIP-GW-UserAgent 8437 1929 IN IP4 172.18.201.173
s=SIP Call
c=IN IP4 1.1.1.1
t=0 0
m=audio 18378 RTP/AVP 0 19
c=IN IP4 1.1.1.1
a=rtpmap:0 PCMU/8000
a=rtpmap:19 CN/8000
a=ptime:20
```

Устранение неполадок

Некоторые советы по устранению проблем для вызовов TLS включают:

- Чтобы позволить серверу CA выполнять сертификаты к точкам доверия, удостоверьтесь, что маршрутизатор IOS, который настроен как сервер CA, имеет включенная **(команда ip http server)** HTTP.
- Часы на Сервере CA и точках доверия должны синхронизироваться.
- Двумя устройствами если сбои подтверждения связи TLS между, (например, OGW и CUBE), проверяют законность сертификатов на устройствах. **Команда debug crypto pki** может использоваться для решения проблем во время подтверждения связи TLS.
- Иногда, когда устройства (например, OGW и CUBE) находятся на других подсетях, там может проблема согласования размера окна TCP, которое вызывает эти ошибки: *ввод-вывод Передает Ошибка чтения ввода-вывода* и *Ошибка*. Этот вопрос может быть решен с **командой ip tcp path-mtu-discovery** на обоих устройствах. Эта проблема могла бы произойти после успешного подтверждения связи TLS.
- “Ясная команда” соединений sip-ua в режиме sip-ua может использоваться для очистки TLS подключение.`Router#`**`clear sip-ua tcp [tls] connections <id <`***`conn id`***`> | target <`***`ipv4:ip address:port`***`>`** Опция **tls** появляется после **tcp** начиная с поездок TLS поверх TCP. Эта команда работает как существующие команды clear для TCP и UDP.

Дополнительные сведения

- Поддержка голосовых технологий
- Поддержка продуктов Голосовой и Унифицированной связи
- Устранение неполадок в системах IP-телефонии Cisco
- Cisco Systems – техническая поддержка и документация