

# VoIP mit PPP über eine Standleitung mit hoher Bandbreite und LLQ

## Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Konventionen](#)

[Hintergrundinformationen](#)

[Konfigurieren](#)

[Netzwerkdigramm](#)

[Konfigurationen](#)

[Überprüfen](#)

[Verifizierung für den San Jose Router](#)

[Überprüfung des Raleigh-Routers](#)

[Fehlerbehebung](#)

[Fehlerbehebung bei Befehlen](#)

[Zugehörige Informationen](#)

## [Einführung](#)

Dieses Dokument enthält Beispielkonfigurationen für zwei Cisco 3640 Router. Die Konfigurationen ermöglichen den Routern die Kommunikation mit VoIP und PPP über eine Mietleitung mit hoher Bandbreite und niedriger Latenz (Low Latency Queuing, LLQ). Weitere Informationen zu LLQ finden Sie im Dokument [VoIP over PPP Links with Quality of Service \(LLQ/IP RTP Priority, LFI, cRTP\)](#).

**Hinweis:** Wenn in diesem Dokument die hohe Bandbreite in Bezug auf VoIP und QoS behandelt wird, beträgt die Bandbreite eine Bandbreite von über 768 Kbit/s.

## [Voraussetzungen](#)

### [Anforderungen](#)

Für dieses Dokument bestehen keine speziellen Anforderungen.

### [Verwendete Komponenten](#)

Die Informationen in diesem Dokument basieren auf den folgenden Software- und Hardwareversionen:

- Cisco IOS® Software Release 12.2(19a) IP Plus oder eine andere Cisco IOS Software-Version von 12.2, 12.2T, 12.3 oder 12.3T
- Zwei Cisco 3640-Router mit mindestens 48 DRAM und 16 MB Flash-Speicher
- Zwei Cisco NM-2V Voice/Fax Interface Card Slot Network-Module plus zwei VIC-2FXS-Schnittstellenkarten
- Zwei serielle Schnittstellen In diesem Beispiel sind die beiden seriellen Schnittstellen NM-1E2Ws mit jeweils einer WIC-1T-WAN-Schnittstellenkarte.
- Analoge Telefone zur Verbindung mit FXS-Ports (Foreign Exchange Station) für Sprachanrufe

**Hinweis:** Die Netzwerkmodule NM-1E2W, NM-1E1R2W und NM-2E2W verfügen nicht über genügend Leistung, um die WIC-2T zu unterstützen. Der fehlende Support ist auf Hardware-Einschränkungen zurückzuführen.

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

## [Konventionen](#)

Weitere Informationen zu Dokumentkonventionen finden Sie in den [Cisco Technical Tips Conventions](#).

## [Hintergrundinformationen](#)

Wenn die Zeit, die zum Senden eines 1500-Byte-Pakets an die Leitung erforderlich ist, mehr als 10 ms beträgt, müssen Sie Pakete fragmentieren. Dieses Dokument stellt eine Konfiguration ohne Fragmentierung dar. Die Konfiguration gilt für eine 1544-Kilobit-Verbindung, bei der die Übertragungsverzögerung für ein 1500-Byte-Paket unter 10 ms liegt.

**Hinweis:** In einigen Fällen, in denen Sie über eine dedizierte, vollständige T1-Verbindung verfügen, kann eine Fragmentierungsfunktion nicht erforderlich sein. Sie benötigen jedoch weiterhin einen QoS-Mechanismus. Verwenden Sie in diesem Fall LLQ. Wenn die zum Versenden eines 1500-Byte-Pakets an die Leitung erforderliche Zeit weniger als 10 ms beträgt, müssen Sie keine Pakete fragmentieren. Die vollständige T1-Leitung bietet genügend Bandbreite, um Sprachpakete in die Warteschlange eindringen und diese unverzüglich verlassen zu können.

**Hinweis:** Wenn Sie die Fragmentierung auf dem Router aktiviert haben, wird der Warteschlangenmechanismus zu 100 Prozent aktiviert. Wenn Sie LLQ konfiguriert haben, schränkt der von Ihnen konfigurierte Wert den Datenverkehr für die Prioritätswarteschlange ein. Wenn Sie die Fragmentierung nicht aktiviert haben, wendet der Router die QoS-Richtlinie nur bei Überlastung an.

Bei Leitungsraten von mehr als 768 Kbit/s ist außerdem ein komprimiertes Real-Time Transport Protocol (cRTP) nicht erforderlich. Weitere Informationen finden Sie im Dokument [VoIP over PPP Links with Quality of Service \[LLQ/IP RTP Priority, LFI, cRTP\]](#). Die Verwendung von cRTP hilft, Bandbreite zu sparen, da cRTP IP-RTP-Header komprimiert. Im Abschnitt [Konfigurationen](#) dieses Dokuments ist die Aktivierung von cRTP nicht erforderlich. Mit T1 können die Sprachpakete problemlos ohne Komprimierung über eine ausreichende Bandbreite auf das Kabel übertragen werden.

**Vorsicht:** Wenn Sie sich für cRTP entscheiden, sollten Sie bedenken, dass cRTP CPU-Ressourcen verwendet. Das cRTP kann einen Router mit hoher Belastung für den Sprachdatenverkehr übersteuern.

**Hinweis:** In dieser Konfiguration werden die beiden Router über eine Standleitung wieder verbunden. In den meisten Topologien können jedoch Router mit Sprachunterstützung überall vorhanden sein. In der Regel sind die Sprach-Router mit LAN-Verbindungen mit anderen Routern verbunden, die mit dem WAN verbunden sind. Wenn Ihre Sprach-Router keine Verbindung über PPP über eine Standleitung herstellen, müssen Sie alle Konfigurationsbefehle für die WAN-Konnektivität auf den Routern konfigurieren, die mit dem WAN verbunden sind. Sie konfigurieren die Befehle auf den Sprach-Routern nicht, wie die [Konfigurationen](#) in diesem Dokument zeigen.

**Hinweis:** Diese Konfiguration kann für Cisco Router der Serien 1700, [2600](#), [3600](#) und [3700](#) verwendet werden.

## [Konfigurieren](#)

In diesem Abschnitt erhalten Sie Informationen zum Konfigurieren der in diesem Dokument beschriebenen Funktionen.

**Hinweis:** Um weitere Informationen zu den in diesem Dokument verwendeten Befehlen zu erhalten, verwenden Sie das [Command Lookup Tool](#) ([nur registrierte](#) Kunden).

## [Netzwerkdigramm](#)

In diesem Dokument wird die folgende Netzwerkeinrichtung verwendet:



## [Konfigurationen](#)

In diesem Dokument werden folgende Konfigurationen verwendet:

- [San José](#)
- [Raleigh](#)

### San José

```
SanJose3640A# show run
Building configuration...

Current configuration : 1425 bytes
!
```

```
version 12.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname SanJose3640A
!
logging buffered 50000 debugging
!
ip subnet-zero
!
!
no ip domain-lookup
!
call rsvp-sync
!
!
!
!
!
!
!
class-map match-all voice-signaling
  match access-group 103
class-map match-all voice-traffic
  match access-group 102
!
!
policy-map voice-policy
  class voice-traffic
    priority 51

!--- These are two uncompressed G729 VoIP calls at 24
kpbs each !--- that have voice activity detection (VAD)
disablement. You also need !--- to consider the Layer 2
(L2) overhead. class voice-signaling bandwidth 16 !---
This assigns a queue for voice signaling traffic that
ensures 8 kbps. !--- Note: This action is optional and
has nothing to do with good voice !--- quality. This
queue assignment is a way to secure signaling.

class class-default
  fair-queue
!--- The class-default class classifies traffic that
does !--- not fall into one of the class definitions.
The fair-queue command !--- associates the default class
weighted fair queuing (WFQ).

!
!
!
interface Ethernet1/0
  ip address 10.89.251.158 255.255.255.192
  half-duplex
!
interface Serial1/0
  bandwidth 1544
  ip address 192.168.1.1 255.255.255.0
  service-policy output voice-policy
  encapsulation ppp
  load-interval 30
  clockrate 2000000
```

```
!  
ip classless  
ip route 0.0.0.0 0.0.0.0 10.89.251.129  
no ip http server  
!  
access-list 102 permit udp any any range 16384 32767  
access-list 103 permit tcp any eq 1720 any  
access-list 103 permit tcp any any eq 1720  
!  
voice-port 3/0/0  
!  
voice-port 3/0/1  
!  
voice-port 3/1/0  
!  
voice-port 3/1/1  
!  
dial-peer cor custom  
!  
!  
!  
dial-peer voice 1 voip  
    incoming called-number .  
    destination-pattern 2...  
    session target ipv4:192.168.1.2  
    dtmf-relay h245-alphanumeric  
    no vad  
!  
dial-peer voice 2 pots  
    destination-pattern 1001  
    port 3/0/0  
!  
dial-peer voice 3 pots  
    destination-pattern 1002  
    port 3/0/1  
!  
!  
line con 0  
line aux 0  
line vty 0 4  
password cisco  
login  
!  
end  
  
SanJose3640A#  
  
SanJose3640A#  
SanJose3640A# show version  
Cisco Internetwork Operating System Software  
IOS (tm) 3600 Software (C3640-IS-M), Version 12.2(19a),  
RELEASE SOFTWARE (fc2)  
Copyright (c) 1986-2003 by cisco Systems, Inc.  
Compiled Mon 29-Sep-03 23:45 by pwade  
Image text-base: 0x60008930, data-base: 0x61134000  
  
ROM: System Bootstrap, Version 11.1(20)AA2, EARLY  
DEPLOYMENT RELEASE SOFTWARE (fc1)  
  
SanJose3640A uptime is 5 minutes  
System returned to ROM by reload  
System image file is "flash:c3640-is-mz.122-19a.bin"  
  
cisco 3640 (R4700) processor (revision 0x00) with
```

```
126976K/4096K bytes of memory.
Processor board ID 15636516
R4700 CPU at 100Mhz, Implementation 33, Rev 1.0
Bridging software.
X.25 software, Version 3.0.0.
SuperLAT software (copyright 1990 by Meridian Technology
Corp).
1 Ethernet/IEEE 802.3 interface(s)
1 Serial network interface(s)
2 Voice FXO interface(s)
2 Voice FXS interface(s)
DRAM configuration is 64 bits wide with parity disabled.
125K bytes of non-volatile configuration memory.
32768K bytes of processor board System flash
(Read/Write)
16384K bytes of processor board PCMCIA Slot1 flash
(Read/Write)

Configuration register is 0x2102

SanJose3640A#
```

## Raleigh

```
Raleigh3640A# show run
Building configuration...

Current configuration : 1406 bytes
!
version 12.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Raleigh3640A
!
logging buffered 50000 debugging
!
ip subnet-zero
!
!
no ip domain-lookup
!
call rsvp-sync
!
!
!
!
!
!
class-map match-all voice-signaling
  match access-group 103
class-map match-all voice-traffic
  match access-group 102
!
!
policy-map voice-policy
  class voice-traffic
    priority 51
!
!--- These are two uncompressed G729 VoIP calls at 24
kpbs each !--- that have VAD disablement. You also need
to consider !--- the L2 overhead. class voice-signaling
```

```
bandwidth 16 !--- This assigns a queue for voice signaling traffic that ensures 8 kbps. !--- Note: This action is optional and has nothing to do with good voice !--- quality. This queue assignment is a way to secure signaling.
```

```
class class-default  
  fair-queue  
!--- The class-default class classifies traffic that does !--- not fall into one of the class definitions.  
The fair-queue command !--- associates the default class WFQ.
```

```
!  
!  
!  
interface Ethernet1/0  
  ip address 10.89.251.159 255.255.255.192  
  half-duplex  
!  
interface Serial1/0  
  bandwidth 1544  
  ip address 192.168.1.2 255.255.255.0  
  service-policy output voice-policy  
  encapsulation ppp  
  load-interval 30  
!  
ip classless  
ip route 0.0.0.0 0.0.0.0 10.89.251.129  
no ip http server  
!  
access-list 102 permit udp any any range 16384 32767  
access-list 103 permit tcp any eq 1720 any  
access-list 103 permit tcp any any eq 1720  
!  
voice-port 3/0/0  
!  
voice-port 3/0/1  
!  
voice-port 3/1/0  
!  
voice-port 3/1/1  
!  
dial-peer cor custom  
!  
!  
!  
dial-peer voice 1 voip  
  incoming called-number .  
  destination-pattern 1...  
  session target ipv4:192.168.1.1  
  dtmf-relay h245-alphanumeric  
  no vad  
!  
dial-peer voice 2 pots  
  destination-pattern 2001  
  port 3/0/0  
!  
dial-peer voice 3 pots  
  destination-pattern 2002  
  port 3/0/1  
!  
!
```

```
line con 0
line aux 0
line vty 0 4
password cisco
login
!
end

Raleigh3640A#
Raleigh3640A#
Raleigh3640A# show version
Cisco Internetwork Operating System Software
IOS (tm) 3600 Software (C3640-IS-M), Version 12.2(19a),
RELEASE SOFTWARE (fc2)
Copyright (c) 1986-2003 by cisco Systems, Inc.
Compiled Mon 29-Sep-03 23:45 by pwade
Image text-base: 0x60008930, data-base: 0x61134000

ROM: System Bootstrap, Version 12.1(17r) [cmong 17r],
RELEASE SOFTWARE (fc1)

Raleigh3640A uptime is 6 minutes
System returned to ROM by reload
System image file is "flash:c3640-is-mz.122-19a.bin"

cisco 3640-A (R4700) processor (revision 0x00) with
94208K/4096K bytes of memory.
Processor board ID 29851759
R4700 CPU at 100Mhz, Implementation 33, Rev 1.0
Bridging software.
X.25 software, Version 3.0.0.
SuperLAT software (copyright 1990 by Meridian Technology
Corp).
1 Ethernet/IEEE 802.3 interface(s)
1 Serial network interface(s)
2 Voice FXO interface(s)
2 Voice FXS interface(s)
DRAM configuration is 64 bits wide with parity disabled.
123K bytes of non-volatile configuration memory.
32768K bytes of processor board System flash
(Read/Write)
16384K bytes of processor board PCMCIA Slot0 flash
(Read/Write)

Configuration register is 0x2102

Raleigh3640A#
```

## Überprüfen

Nachdem Sie diese [Konfigurationen](#) in Ihre Router eingegeben haben, überprüfen Sie deren ordnungsgemäße Funktion. Die Befehle und die entsprechende Ausgabe zeigen hier eine erfolgreiche Implementierung der Konfigurationen.

Bestimmte **show**-Befehle werden vom [Output Interpreter Tool](#) unterstützt (nur [registrierte](#) Kunden), mit dem Sie eine Analyse der **show**-Befehlsausgabe anzeigen können.

- **show interface serial 1/0** - Ermöglicht Ihnen, den Status Ihrer seriellen Schnittstelle zu überprüfen.



- **Übersicht über aktive Sprachnachrichten anzeigen** - Ermöglicht Ihnen, Anrufinformationen während eines Anrufs anzuzeigen.
- **show call active voice** - Ermöglicht Ihnen, Anrufinformationen während eines Anrufs anzuzeigen.
- **show policy-map interface** - Ermöglicht Ihnen, die von der Schnittstelle verwendete QoS-Richtlinie zu überprüfen.
- **show access-list 102** - Ermöglicht Ihnen, die Paketauswahl nach der Zugriffsliste für die Sprachklasse zu überprüfen. Geben Sie den Befehl nach einigen Sekunden ein zweites Mal aus, und überprüfen Sie, ob die Paketanzahl erhöht wird. Geben Sie ggf. den Befehl **clear access-list counter 102** ein.
- **Übersicht über Sprachanrufe anzeigen** - Ermöglicht Ihnen, den Status der Anrufe zu überprüfen. Der Befehl zeigt an, ob die Anrufe über eine Verbindung verfügen.
- **show voice port summary** - Ermöglicht Ihnen, den Status der Sprach-Ports zu überprüfen. Der Befehl zeigt an, dass die Sprach-Ports aufgelegt oder abgehoben sind.
- **show voice dsp**: Ermöglicht die Überprüfung des Status des digitalen Signalprozessors (DSP) und des Codierungs-Decoders (Codec), den jeder Anruf verwendet.

## Verifizierung für den San Jose Router

Überprüfen Sie vor der Überprüfung die Schnittstellen, um sicherzustellen, dass Sie über die erforderliche Verbindung zum Tätigen von Anrufen verfügen. Geben Sie den Befehl **show interface serial 1/0** ein, um den Status der seriellen Schnittstelle zu überprüfen. Achten Sie bei den [Konfigurationen](#) in diesem Dokument darauf, dass die seriellen und Multilink-Schnittstellen den Status `Line Protocol Up` aufweisen. Stellen Sie außerdem sicher, dass Sie Folgendes sehen:

- **LCP Open, Multilink Open (LCP offen, Multilink offen)**: Gibt die Einrichtung der PPP-Verbindung an.
- **offen: IPCP, CDPCP**: Gibt an, dass das Senden von IP-Datenverkehr über die PPP-Verbindung möglich ist.
- **Warteschlangenstrategie: Weighted Fair** - Entspricht der CLI (Service-Policy Output Command Line Interface) unter der seriellen Schnittstelle. Die Strategie besteht in der Konfiguration des LLQ zur Priorisierung von Sprache gegenüber Daten.

```
SanJose3640A# show interface serial 1/0
Serial1/0 is up, line protocol is up
Hardware is QUICC Serial
Internet address is 192.168.1.1/24
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation PPP, loopback not set
Keepalive set (10 sec)
LCP Open
Open: IPCP, CDPCP
Last input 00:00:27, output 00:00:02, output hang never
Last clearing of "show interface" counters 00:00:05
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/64/0 (size/max total/threshold/drops)
Conversations 0/1/256 (active/max active/max total)
Reserved Conversations 1/1 (allocated/max allocated)
Available Bandwidth 1091 kilobits/sec
30 second input rate 0 bits/sec, 0 packets/sec
```

```
30 second output rate 0 bits/sec, 0 packets/sec
1 packets input, 16 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
1 packets output, 16 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 output buffer failures, 0 output buffers swapped out
0 carrier transitions
DCD=up DSR=up DTR=up RTS=up CTS=up
```

SanJose3640A#

Diese Ausgabe zeigt eine erfolgreiche Verbindung zwischen den Routern. Wenn das Leitungsprotokoll nicht aktiviert ist, überprüfen Sie die Taktfrequenz auf der DCE-Schnittstelle. Einige serielle Schnittstellen unterstützen keine Hochgeschwindigkeits-Verbindungen, z. B. NM-8A/S. Überprüfen Sie außerdem, ob die Parameter auf beiden Seiten übereinstimmen und, was am wichtigsten ist, ob die Kapselung übereinstimmt.

Die Ausgabe des Befehls **show call active voice brief** zeigt zwei erfolgreiche Anrufe. Ein Anruf geht vom Router in Raleigh zum Router in San Jose, der andere von San Jose nach Raleigh. In dieser Liste wird die in Fettschrift angezeigte Ausgabe erläutert:

- **Answer 1001 active:** Bedeutet, dass San Jose der Router ist, von dem der Anruf stammt.
- **Tele 3/0/0:** Bedeutet, dass es sich um die Telefonie-Anrufkomponente handelt.
- **2001 active:** Bezeichnet, dass ein Telefon auf der Raleigh-Seite den Anruf empfängt.
- **IP 192.168.1.2:** Dies ist der IP-Anrufabschnitt.
- **Answer 2002 active:** Bedeutet, dass Raleigh der Router ist, an den der Anruf gesendet wird.
- **IP 192.168.1.2:** Dies ist der IP-Anrufabschnitt.
- **Ursprungs 1002 active:** bedeutet, dass ein Telefon auf der San Jose-Seite den Anruf empfängt.
- **Tele 3/0/1:** Bedeutet, dass es sich um den Anrufabschnitt der Telefonie handelt.

SanJose3640A# **show call active voice brief**

```
<ID>: <start>hs.<index> +<connect> pid:<peer_id> <dir> <addr> <state>
dur hh:mm:ss tx:<packets>/<bytes> rx:<packets>/<bytes>
IP <ip>:<udp> rtt:<time>ms pl:<play>/<gap>ms lost:<lost>/<early>/<late>
delay:<last>/<min>/<max>ms <codec>
MODEMPASS <method> buf:<fills>/<drains> loss <overall%> <multipkt>/<corrected>
last <buf event time>s dur:<Min>/<Max>s
FR <protocol> [int dlci cid] vad:<y/n> dtmf:<y/n> seq:<y/n>
<codec> (payload size)
ATM <protocol> [int vpi/vci cid] vad:<y/n> dtmf:<y/n> seq:<y/n>
<codec> (payload size)
Tele <int>: tx:<tot>/<v>/<fax>ms <codec> noise:<l> acom:<l> i/o:<l>/<l> dBm
Proxy <ip>:<audio udp>,<video udp>,<tcp0>,<tcp1>,<tcp2>,<tcp3> endpt: <type>/<manf>
bw: <req>/<act> codec: <audio>/<video>
tx: <audio pkts>/<audio bytes>,<video pkts>/<video bytes>,<t120 pkts>/<t120 bytes>
rx: <audio pkts>/<audio bytes>,<video pkts>/<video bytes>,<t120 pkts>/<t120 bytes>
```

Total call-legs: 4

```
11E8 : 115599hs.1 +318 pid:2 Answer 1001 active
dur 00:00:29 tx:1545/30900 rx:1544/30880
Tele 3/0/0:20: tx:30890/30890/0ms g729r8 noise:0 acom:2 i/o:-35/-44 dBm
```

```
11E8 : 115823hs.1 +94 pid:1 Originate 2001 active
dur 00:00:31 tx:1556/31120 rx:1602/32040
IP 192.168.1.2:17360 rtt:4ms pl:25590/0ms lost:0/1/0 delay:69/69/70ms g729r8
```

```
11F0 : 116855hs.1 +156 pid:1 Answer 2002 active
```

dur 00:00:20 tx:1087/21740 rx:1009/20180  
IP 192.168.1.2:16772 rtt:2ms pl:17270/0ms lost:0/0/0 delay:69/69/70ms g729r8

11F0 : 116855hs.2 +156 pid:3 **Originate 1002 active**

dur 00:00:20 tx:1009/20180 rx:1087/21740

**Tele 3/0/1** (23): tx:21740/21740/0ms g729r8 noise:0 acom:5 i/0:-40/-40 dBm

Total call-legs: 4

SanJose3640A#

Diese Ausgabe des Befehls **show call active voice** enthält weitere Details zum aktiven Anruf:

SanJose3640A# **show call active voice**

Total call-legs: 4

GENERIC:

SetupTime=115599 ms

Index=1

**PeerAddress=1001**

PeerSubAddress=

PeerId=2

PeerIfIndex=9

LogicalIfIndex=4

ConnectTime=115917

**CallDuration=00:05:05**

CallState=4

CallOrigin=2

ChargedUnits=0

InfoType=2

TransmitPackets=15338

TransmitBytes=306760

ReceivePackets=15337

ReceiveBytes=306740

TELE:

ConnectionId=[0x38D3783F 0x14F111CC 0x801CFDB1 0x2D0CC4A5]

IncomingConnectionId=[0x38D3783F 0x14F111CC 0x801CFDB1 0x2D0CC4A5]

TxDuration=306740 ms

VoiceTxDuration=306740 ms

FaxTxDuration=0 ms

CoderTypeRate=g729r8

NoiseLevel=0

ACOMLevel=5

OutSignalLevel=-43

InSignalLevel=-36

InfoActivity=2

ERLLevel=5

SessionTarget=

ImgPages=0

GENERIC:

SetupTime=115823 ms

Index=1

PeerAddress=2001

PeerSubAddress=

PeerId=1

PeerIfIndex=8

LogicalIfIndex=0

ConnectTime=115917

CallDuration=00:05:07

CallState=4

CallOrigin=1

ChargedUnits=0

InfoType=2

TransmitPackets=15357  
TransmitBytes=307140  
ReceivePackets=15403  
ReceiveBytes=308060  
VOIP:  
ConnectionId[0x38D3783F 0x14F111CC 0x801CFDB1 0x2D0CC4A5]  
IncomingConnectionId[0x38D3783F 0x14F111CC 0x801CFDB1 0x2D0CC4A5]  
RemoteIPAddress=192.168.1.2  
RemoteUDPPort=17360  
RemoteSignallingIPAddress=192.168.1.2  
RemoteSignallingPort=1720  
RemoteMediaIPAddress=192.168.1.2  
RemoteMediaPort=17360  
RoundTripDelay=1 ms  
SelectedQoS=best-effort  
tx\_DtmfRelay=h245-alphanumeric  
FastConnect=TRUE

Separate H245 Connection=FALSE

H245 Tunneling=TRUE

SessionProtocol=cisco  
SessionTarget=ipv4:192.168.1.2  
OnTimeRvPlayout=300810  
GapFillWithSilence=0 ms  
GapFillWithPrediction=0 ms  
GapFillWithInterpolation=0 ms  
GapFillWithRedundancy=0 ms  
HiWaterPlayoutDelay=70 ms  
LoWaterPlayoutDelay=69 ms  
ReceiveDelay=69 ms  
LostPackets=0  
EarlyPackets=2  
LatePackets=0

**VAD = disabled**

**CoderTypeRate=g729r8**

CodecBytes=20  
GENERIC:  
SetupTime=116855 ms  
Index=1  
PeerAddress=2002  
PeerSubAddress=  
PeerId=1  
PeerIfIndex=8  
LogicalIfIndex=0  
ConnectTime=117011  
CallDuration=00:04:56  
CallState=4  
CallOrigin=2  
ChargedUnits=0  
InfoType=2  
TransmitPackets=14915  
TransmitBytes=298300  
ReceivePackets=14837  
ReceiveBytes=296740

VOIP:  
ConnectionId[0x6C135AD4 0x14F311CC 0x8024CE4C 0xAA60AB15]  
IncomingConnectionId[0x6C135AD4 0x14F311CC 0x8024CE4C 0xAA60AB15]  
RemoteIPAddress=192.168.1.2  
RemoteUDPPort=16772  
RemoteSignallingIPAddress=192.168.1.2  
RemoteSignallingPort=11004  
RemoteMediaIPAddress=192.168.1.2

RemoteMediaPort=16772  
RoundTripDelay=7 ms  
SelectedQoS=best-effort  
tx\_DtmfRelay=h245-alphanumeric  
FastConnect=TRUE

Separate H245 Connection=FALSE

H245 Tunneling=TRUE

SessionProtocol=cisco  
SessionTarget=  
OnTimeRvPlayout=295580  
GapFillWithSilence=0 ms  
GapFillWithPrediction=0 ms  
GapFillWithInterpolation=0 ms  
GapFillWithRedundancy=0 ms  
HiWaterPlayoutDelay=70 ms  
LoWaterPlayoutDelay=69 ms  
ReceiveDelay=69 ms

**LostPackets=0**

**EarlyPackets=0**

**LatePackets=0**

**VAD = disabled**

**CoderTypeRate=g729r8**

CodecBytes=20

GENERIC:

SetupTime=116855 ms

Index=2

PeerAddress=1002

PeerSubAddress=

PeerId=3

PeerIfIndex=10

LogicalIfIndex=5

ConnectTime=117011

CallDuration=00:04:59

CallState=4

CallOrigin=1

ChargedUnits=0

InfoType=2

TransmitPackets=14952

TransmitBytes=299040

ReceivePackets=15030

ReceiveBytes=300600

TELE:

ConnectionId=[0x6C135AD4 0x14F311CC 0x8024CE4C 0xAA60AB15]

IncomingConnectionId=[0x6C135AD4 0x14F311CC 0x8024CE4C 0xAA60AB15]

TxDuration=300600 ms

VoiceTxDuration=300600 ms

FaxTxDuration=0 ms

CoderTypeRate=g729r8

NoiseLevel=0

ACOMLevel=5

OutSignalLevel=-40

InSignalLevel=-41

InfoActivity=2

ERLLevel=5

SessionTarget=

ImgPages=0Total call-legs: 4

SanJose3640A#\$

Other shows:

Die Ausgabe des Befehls **show policy-map interface** enthält folgende fett formatierte Anweisung:

- **Gebotene Übertragungsrate von 30 Sekunden 51.000 bps** - Zeigt die Bandbreite an, die für die beiden Anrufe erforderlich ist, 51 Kbit/s.

```
SanJose3640A# show policy-map interface
Serial1/0

Service-policy output: voice-policy

Class-map: voice-traffic (match-all)
99403 packets, 6401420 bytes
30 second offered rate 51000 bps, drop rate 0 bps
Match: access-group 102
Queueing
Strict Priority
Output Queue: Conversation 264
Bandwidth 51 (kbps) Burst 1275 (Bytes)
(pkts matched/bytes matched) 407/65676
(total drops/bytes drops) 0/0

Class-map: voice-signaling (match-all)
158 packets, 12926 bytes
30 second offered rate 0 bps, drop rate 0 bps
Match: access-group 103
Queueing
Output Queue: Conversation 265
Bandwidth 16 (kbps) Max Threshold 64 (packets)
(pkts matched/bytes matched) 158/12926
(depth/total drops/no-buffer drops) 0/0/0

Class-map: class-default (match-any)
75 packets, 9221 bytes
30 second offered rate 0 bps, drop rate 0 bps
Match: any
Queueing
Flow Based Fair Queueing
Maximum Number of Hashed Queues 256
(total queued/total drops/no-buffer drops) 0/0/0
SanJose3640A#
```

Die Ausgabe des Befehls **show access-lists 102** enthält folgende fett formatierte Anweisung:

- **100676 Matches (Übereinstimmungen mit 10676)** - Zeigt an, dass die Priorisierung von RTP-Paketen erfolgt, weil die Pakete die Zugriffsliste 102 erreichen.

```
SanJose3640A# show access-lists 102
Extended IP access list 102
permit udp any any range 16384 32767 (100676 matches)
SanJose3640A#
SanJose3640A#
SanJose3640A#
SanJose3640A#
SanJose3640A# show access-lists 102
Extended IP access list 102
permit udp any any range 16384 32767 (100930 matches)
SanJose3640A#
SanJose3640A#
SanJose3640A# show access-lists 102
```

```

Extended IP access list 102
permit udp any any range 16384 32767 (101076 matches)
SanJose3640A#
SanJose3640A#
SanJose3640A#
SanJose3640A# show access-lists 102
Extended IP access list 102
permit udp any any range 16384 32767 (101198 matches)
SanJose3640A#
SanJose3640A#
SanJose3640A# show access-lists 102
Extended IP access list 102
permit udp any any range 16384 32767 (101304 matches)
SanJose3640A#
SanJose3640A#

```

```

SanJose3640A#
SanJose3640A# show voice call sum
PORT CODEC VAD VTSP STATE VPM STATE
=====
3/0/0 g729r8 n S_CONNECT FXSLS_CONNECT
3/0/1 g729r8 n S_CONNECT FXSLS_CONNECT
3/1/0 - - - FXOLS_ONHOOK
3/1/1 - - - FXOLS_ONHOOK

```

```

SanJose3640A#
SanJose3640A#

```

```

SanJose3640A#
SanJose3640A# show voice port sum
IN OUT
PORT CH SIG-TYPE ADMIN OPER STATUS STATUS EC
=====
3/0/0 -- fxs-ls up up off-hook idle y
3/0/1 -- fxs-ls up up off-hook idle y
3/1/0 -- fxo-ls up dorm idle on-hook y
3/1/1 -- fxo-ls up dorm idle on-hook y

```

```

SanJose3640A#

```

```

SanJose3640A# show voice dsp

DSP DSP DSPWARE CURR BOOT PAK TX/RX
TYPE NUM CH CODEC VERSION STATE STATE RST AI VOICEPORT TS ABORT PACK COUNT
=====
C542 001 01 g729r8 3.4.55 busy idle 0 0 3/0/0 NA 0 62487/61902
C542 002 01 g729r8 3.4.55 busy idle 0 0 3/0/1 NA 0 44362/44194
C542 003 01 g711ulaw 3.4.55 IDLE idle 0 0 3/1/0 NA 0 541/546
C542 004 01 g711ulaw 3.4.55 IDLE idle 0 0 3/1/1 NA 0 535/532

```

```

SanJose3640A#

```

## [Überprüfung des Raleigh-Routers](#)

Das Verifizierungsverfahren für den Raleigh-Router ähnelt dem Verfahren für den San Jose-Router.

```

Raleigh3640A# show interface serial 1/0
Serial1/0 is up, line protocol is up

```

Hardware is QUICC Serial  
Internet address is 192.168.1.2/24  
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,  
reliability 255/255, txload 1/255, rxload 1/255  
Encapsulation PPP, loopback not set  
Keepalive set (10 sec)

**LCP Open**

**Open: IPCP, CDPCP**

Last input 00:00:15, output 00:00:00, output hang never  
Last clearing of "show interface" counters 00:12:33  
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0

**Queueing strategy: weighted fair**

Output queue: 0/1000/64/0 (size/max total/threshold/drops)  
Conversations 0/1/256 (active/max active/max total)  
Reserved Conversations 1/1 (allocated/max allocated)  
Available Bandwidth 1091 kilobits/sec  
30 second input rate 0 bits/sec, 0 packets/sec  
30 second output rate 0 bits/sec, 0 packets/sec  
167 packets input, 6849 bytes, 0 no buffer  
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles  
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort  
169 packets output, 6907 bytes, 0 underruns  
0 output errors, 0 collisions, 0 interface resets  
0 output buffer failures, 0 output buffers swapped out  
11 carrier transitions  
DCD=up DSR=up DTR=up RTS=up CTS=up

Raleigh3640A#

Raleigh3640A#

Raleigh3640A#

Raleigh3640A#

Raleigh3640A# **show call active voice**

Total call-legs: 4

**GENERIC:**

SetupTime=209451 ms

Index=1

PeerAddress=1001

PeerSubAddress=

PeerId=1

PeerIfIndex=8

LogicalIfIndex=0

ConnectTime=209543

CallDuration=00:08:20

CallState=4

CallOrigin=2

ChargedUnits=0

InfoType=2

TransmitPackets=25054

TransmitBytes=501080

ReceivePackets=25008

ReceiveBytes=500160

**VOIP:**

ConnectionId[0x38D3783F 0x14F111CC 0x801CFDB1 0x2D0CC4A5]

IncomingConnectionId[0x38D3783F 0x14F111CC 0x801CFDB1 0x2D0CC4A5]

RemoteIPAddress=192.168.1.1

RemoteUDPPort=17210

RemoteSignallingIPAddress=192.168.1.1

RemoteSignallingPort=11006

RemoteMediaIPAddress=192.168.1.1

RemoteMediaPort=17210

RoundTripDelay=3 ms

SelectedQoS=best-effort



tx\_DtmfRelay=h245-alphanumeric  
FastConnect=TRUE

Separate H245 Connection=FALSE

H245 Tunneling=TRUE

SessionProtocol=cisco  
SessionTarget=  
OnTimeRvPlayout=497610  
GapFillWithSilence=0 ms  
GapFillWithPrediction=0 ms  
GapFillWithInterpolation=0 ms  
GapFillWithRedundancy=0 ms  
HiWaterPlayoutDelay=70 ms  
LoWaterPlayoutDelay=69 ms  
ReceiveDelay=69 ms  
LostPackets=0  
EarlyPackets=1  
LatePackets=0

**VAD = disabled**

**CoderTypeRate=g729r8**

CodecBytes=20  
GENERIC:  
SetupTime=209451 ms  
Index=2

**PeerAddress=2001**

PeerSubAddress=  
PeerId=2  
PeerIfIndex=9  
LogicalIfIndex=4  
ConnectTime=209543

**CallDuration=00:08:21**

CallState=4  
CallOrigin=1  
ChargedUnits=0  
InfoType=2  
TransmitPackets=25074  
TransmitBytes=501480  
ReceivePackets=25120  
ReceiveBytes=502400

TELE:

ConnectionId=[0x38D3783F 0x14F111CC 0x801CFDB1 0x2D0CC4A5]  
IncomingConnectionId=[0x38D3783F 0x14F111CC 0x801CFDB1 0x2D0CC4A5]  
TxDuration=502410 ms  
VoiceTxDuration=502410 ms  
FaxTxDuration=0 ms  
CoderTypeRate=g729r8  
NoiseLevel=0  
ACOMLevel=1  
OutSignalLevel=-41  
InSignalLevel=-37  
InfoActivity=2  
ERLLevel=1  
SessionTarget=  
ImgPages=0  
GENERIC:  
SetupTime=210097 ms  
Index=1  
PeerAddress=2002  
PeerSubAddress=  
PeerId=3  
PeerIfIndex=10  
LogicalIfIndex=5

ConnectTime=210638  
**CallDuration=00:08:10**  
CallState=4  
CallOrigin=2  
ChargedUnits=0  
InfoType=2  
TransmitPackets=24606  
TransmitBytes=492120  
ReceivePackets=24605  
ReceiveBytes=492100  
TELE:  
ConnectionId=[0x6C135AD4 0x14F311CC 0x8024CE4C 0xAA60AB15]  
IncomingConnectionId=[0x6C135AD4 0x14F311CC 0x8024CE4C 0xAA60AB15]  
TxDuration=492110 ms  
VoiceTxDuration=492110 ms  
FaxTxDuration=0 ms  
CoderTypeRate=g729r8  
NoiseLevel=0  
ACOMLevel=0  
OutSignalLevel=-46  
InSignalLevel=-33  
InfoActivity=2  
ERLLevel=0  
SessionTarget=  
ImgPages=0  
GENERIC:  
SetupTime=210480 ms  
Index=1  
**PeerAddress=1002**  
PeerSubAddress=  
PeerId=1  
PeerIfIndex=8  
LogicalIfIndex=0  
ConnectTime=210638  
**CallDuration=00:08:11**  
CallState=4  
CallOrigin=1  
ChargedUnits=0  
InfoType=2  
TransmitPackets=24587  
TransmitBytes=491740  
ReceivePackets=24664  
ReceiveBytes=493280  
VOIP:  
ConnectionId[0x6C135AD4 0x14F311CC 0x8024CE4C 0xAA60AB15]  
IncomingConnectionId[0x6C135AD4 0x14F311CC 0x8024CE4C 0xAA60AB15]  
RemoteIPAddress=192.168.1.1  
RemoteUDPPort=18884  
RemoteSignallingIPAddress=192.168.1.1  
RemoteSignallingPort=1720  
RemoteMediaIPAddress=192.168.1.1  
RemoteMediaPort=18884  
**RoundTripDelay=4 ms**  
SelectedQoS=best-effort  
tx\_DtmfRelay=h245-alphanumeric  
FastConnect=TRUE  
  
Separate H245 Connection=FALSE  
  
H245 Tunneling=TRUE  
  
SessionProtocol=cisco  
SessionTarget=ipv4:192.168.1.1  
OnTimeRvPayout=487570

GapFillWithSilence=0 ms  
GapFillWithPrediction=0 ms  
GapFillWithInterpolation=0 ms  
GapFillWithRedundancy=0 ms  
HiWaterPlayoutDelay=70 ms  
LoWaterPlayoutDelay=69 ms  
ReceiveDelay=69 ms  
**LostPackets=0**  
**EarlyPackets=1**  
**LatePackets=0**  
**VAD = disabled**  
**CoderTypeRate=g729r8**  
CodecBytes=20Total call-legs: 4

Raleigh3640A#  
Raleigh3640A#

Raleigh3640A# **show policy interface**  
Serial1/0

Service-policy output: voice-policy

Class-map: voice-traffic (match-all)  
113186 packets, 7289624 bytes  
**30 second offered rate 51000 bps, drop rate 0 bps**  
Match: access-group 102  
Queueing  
Strict Priority  
Output Queue: Conversation 264  
**Bandwidth 51 (kbps) Burst 1275 (Bytes)**  
**(pkts matched/bytes matched) 471/75864**  
**(total drops/bytes drops) 0/0**

Class-map: voice-signaling (match-all)  
162 packets, 13339 bytes  
30 second offered rate 0 bps, drop rate 0 bps  
Match: access-group 103  
Queueing  
Output Queue: Conversation 265  
Bandwidth 16 (kbps) Max Threshold 64 (packets)  
(pkts matched/bytes matched) 162/13339  
(depth/total drops/no-buffer drops) 0/0/0

Class-map: class-default (match-any)  
194 packets, 16761 bytes  
30 second offered rate 0 bps, drop rate 0 bps  
Match: any  
Queueing  
Flow Based Fair Queueing  
Maximum Number of Hashed Queues 256  
**(total queued/total drops/no-buffer drops) 0/0/0**  
Raleigh3640A#

Raleigh3640A# **show access-lists 102**  
Extended IP access list 102  
permit udp any any range 16384 32767 (**113963 matches**)  
Raleigh3640A#  
Raleigh3640A#  
Raleigh3640A# **show access-lists 102**  
Extended IP access list 102  
permit udp any any range 16384 32767 (**114093 matches**)  
Raleigh3640A#  
Raleigh3640A#

```
Raleigh3640A# show access-lists 102
Extended IP access list 102
permit udp any any range 16384 32767 (114188 matches)
Raleigh3640A#
Raleigh3640A#
Raleigh3640A# show access-lists 102
Extended IP access list 102
permit udp any any range 16384 32767 (114404 matches)
Raleigh3640A#
Raleigh3640A#
```

```
Raleigh3640A#
Raleigh3640A# show voice call sum
PORT CODEC VAD VTSP STATE VPM STATE
=====
3/0/0 g729r8 n S_CONNECT FXSLS_CONNECT
3/0/1 g729r8 n S_CONNECT FXSLS_CONNECT
3/1/0 - - - FXOLS_ONHOOK
3/1/1 - - - FXOLS_ONHOOK

Raleigh3640A#
```

```
Raleigh3640A# show voice port sum
IN OUT
PORT CH SIG-TYPE ADMIN OPER STATUS STATUS EC
=====
3/0/0 -- fxs-ls up up off-hook idle y
3/0/1 -- fxs-ls up up off-hook idle y
3/1/0 -- fxo-ls up dorm idle on-hook y
3/1/1 -- fxo-ls up dorm idle on-hook y
```

```
Raleigh3640A#
```

```
Raleigh3640A#
Raleigh3640A# show voice dsp
DSP DSP DSPWARE CURR BOOT PAK TX/RX
TYPE NUM CH CODEC VERSION STATE STATE RST AI VOICEPORT TS ABORT PACK COUNT
=====
C542 001 01 g729r8 3.4.55 busy idle 0 0 3/0/0 NA 0 69615/68771
C542 002 01 g729r8 3.4.55 busy idle 0 0 3/0/1 NA 0 51511/51520
C542 003 01 g711ulaw 3.4.55 IDLE idle 0 0 3/1/0 NA 0 541/546
C542 004 01 g711ulaw 3.4.55 IDLE idle 0 0 3/1/1 NA 0 535/532
```

```
Raleigh3640A#
```

## [Fehlerbehebung](#)

Dieser Abschnitt enthält Informationen zur Fehlerbehebung in Ihrer Konfiguration.

### [Fehlerbehebung bei Befehlen](#)

Bestimmte **show**-Befehle werden vom [Output Interpreter Tool](#) unterstützt (nur [registrierte](#) Kunden), mit dem Sie eine Analyse der **show**-Befehlsausgabe anzeigen können.

**Hinweis:** Bevor Sie **Debugbefehle** ausgeben, lesen Sie [die](#) Informationen [Wichtige Informationen über Debug-Befehle](#).

- **debug voip ccapi inout:** Verfolgt den Ausführungspfad über die API (Call Control Application Programming Interface).
- **debug vpm all:** Aktiviert das Debugging auf allen VPM-Bereichen (Virtual Voice Port Module).
- **show log:** Zeigt die Ausgabe von aktivierten Debuggen an.

Da die Seiten Raleigh und San Jose in der Konfiguration und Konfiguration sehr ähnlich sind, zeigt dieses Dokument die **debug voip ccapi in out** und **debug vpm alle** Befehle nur für den San Jose Router.

Wenn die Anruferichtung ein Problem darstellt, geben Sie die in diesem Abschnitt aufgeführten **Debugbefehle** aus. Vergleichen Sie die Ausgabe mit den Informationen hier. Sie können Software wie Compare it oder Beyond Compare verwenden, um die beiden Textdateien zu vergleichen und die Unterschiede zu finden. Die Ausgabe dient hier als Referenz für einen erfolgreichen Anruf.

Bestimmen Sie zunächst, was während des Anrufs auf dem Router geschieht. Geben Sie die **debug voip ccapi-Ausgabe** und die **debug vpm alle** Befehle ein. Die Ausgabe des Befehls **show debug**, wie hier dargestellt, zeigt die Aktivierung des **Befehls debug vpm all** im San Jose-Router. Sie können die Aktivierung des Befehls **debug vpm all** bestimmen, da in der Ausgabe neben dem Befehl **debug voip ccapi in out** vier aktivierte Debugbefehle angezeigt werden. Diese vier Befehle verfügen über die automatische Aktivierung, wenn Sie den Befehl **debug vpm all** eingeben.

**Vorsicht:** Sie müssen diese **Debugbefehle** deaktivieren, nachdem Sie die erforderliche Ausgabe generiert haben. Deaktivieren Sie die **debug**-Befehle mit dem Problem des Befehls **undebug all**. Wenn Sie die Debugging-Aktivierung verlassen, können Probleme mit der Router-Leistung auftreten. Durch das Debuggen von Befehlen mit Enablement werden CPU-Ressourcen beansprucht.

```
SanJose3640A# show debug
voip:
voip ccAPI function enter/exit debugging is on
Voice Port Module session debugging is on
Voice Port Module DSP message debugging is on
Voice Port Module error debugging is on
Voice Port Module signaling debugging is on
Voice Port Module voaal2 debugging is on
Voice Port Module trunk conditioning is on
SanJose3640A#
SanJose3640A#
SanJose3640A#
SanJose3640A#
SanJose3640A#! Call from 1001 to 2001
SanJose3640A#
SanJose3640A#
SanJose3640A#
SanJose3640A#
*Mar 1 00:05:07.675: htsp_dsp_message: SEND/RESP_SIG_STATUS: state=0xC timestamp=33146
  systime=30767
*Mar 1 00:05:07.679: htsp_process_event: [3/0/0, FXSLS_ONHOOK, E_DSP_SIG_
  1100] fxsls_onhook_offhook htsp_setup_ind
*Mar 1 00:05:07.679: [3/0/0] get_local_station_id calling num= calling name= calling
  time=00/00 00:00
*Mar 1 00:05:07.679: cc_api_call_setup_ind (vdbPtr=0x6217C270, callInfo={called=,called_
  oct3=0x81,calling=,calling_oct3=0x0,calling_oct3a=0x0,calling_xlated=false,
  subscriber_type_str=RegularLine,fdest=0,peer_tag=2, prog_ind=3,callingIE_present 0},
  callID=0x61DAB4F4)
*Mar 1 00:05:07.679: cc_api_call_setup_ind calling number is null, answer addr dest
  pattern 1001 e164_ans_addr 0 e164_dest_pattern 1
```

\*Mar 1 00:05:07.679: cc\_api\_call\_setup\_ind valid dest pattern, copying 1001 to calling number

\*Mar 1 00:05:07.679: cc\_api\_call\_setup\_ind type 3 , prot 0

\*Mar 1 00:05:07.683: cc\_process\_call\_setup\_ind (event=0x62107860)

\*Mar 1 00:05:07.683: >>>CCAPI handed cid 5 with tag 2 to app "DEFAULT"

\*Mar 1 00:05:07.683: sess\_appl: ev(24=CC\_EV\_CALL\_SETUP\_IND), cid(5), disp(0)

\*Mar 1 00:05:07.683: sess\_appl: ev(SSA\_EV\_CALL\_SETUP\_IND), cid(5), disp(0)

\*Mar 1 00:05:07.683: ssaCallSetupInd

\*Mar 1 00:05:07.683: ccCallSetContext (callID=0x5, context=0x620005E8)

\*Mar 1 00:05:07.683: ssaCallSetupInd cid(5), st(SSA\_CS\_MAPPING),oldst(0), ev(24)ev->e.evCallSetupInd.nCallInfo.finalDestFlag = 0

\*Mar 1 00:05:07.683: ccCallSetupAck (callID=0x5)

\*Mar 1 00:05:07.683: ccCallReportDigits (callID=0x5, enable=0x1)

\*Mar 1 00:05:07.683: cc\_api\_call\_report\_digits\_done (vdbPtr=0x6217C270, callID=0x5, disp=0)

\*Mar 1 00:05:07.683: sess\_appl: ev(53=CC\_EV\_CALL\_REPORT\_DIGITS\_DONE), cid(5), disp(0)

\*Mar 1 00:05:07.683: cid(5)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_CALL\_REPORT\_DIGITS\_DONE) oldst(SSA\_CS\_MAPPING)cfid(-1)csize(0)in(1)fDest(0)

\*Mar 1 00:05:07.683: ssaReportDigitsDone cid(5) peer list: (empty)

\*Mar 1 00:05:07.683: ssaReportDigitsDone callid=5 Enable succeeded

\*Mar 1 00:05:07.687: ccGenerateTone (callID=0x5 tone=8)

\*Mar 1 00:05:07.687: dsp\_digit\_collect\_on: [3/0/0] packet\_len=20 channel\_id=128 packet\_id=35 min\_inter\_delay=240 max\_inter\_delay=9760 mim\_make\_time=10 max\_make\_time=100 min\_brake\_time=10 max\_brake\_time=100

\*Mar 1 00:05:07.687: dsp\_soutput: [3/0/0]

\*Mar 1 00:05:07.687: dsp\_digit\_collect\_on: [3/0/0] packet\_len=20 channel\_id=128 packet\_id=35 min\_inter\_delay=240 max\_inter\_delay=9760 mim\_make\_time=10 max\_make\_time=100 min\_brake\_time=10 max\_brake\_time=100

\*Mar 1 00:05:07.687: dsp\_soutput: [3/0/0]

\*Mar 1 00:05:07.687: htsp\_process\_event: [3/0/0, FXSLS\_WAIT\_SETUP\_ACK, E\_HTSP\_SETUP\_ACK]

\*Mar 1 00:05:09.455: cc\_api\_call\_digit\_begin (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x5, digit=2, digit\_begin\_flags=0x1, rtp\_timestamp=0xEB32A6E0 rtp\_expiration=0x0, dest\_mask=0x1)

\*Mar 1 00:05:09.455: sess\_appl: ev(10=CC\_EV\_CALL\_DIGIT\_BEGIN), cid(5), disp(0)

\*Mar 1 00:05:09.455: cid(5)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_DIGIT\_BEGIN) oldst(SSA\_CS\_MAPPING)cfid(-1)csize(0)in(1)fDest(0)

\*Mar 1 00:05:09.455: ssaIgnore cid(5), st(SSA\_CS\_MAPPING),oldst(0), ev(10)

\*Mar 1 00:05:09.515: cc\_api\_call\_digit\_end (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x5,digit=2,duration=95,xruleCallingTag=0,xruleCalledTag=0, dest\_mask=0x1), digit\_tone\_mode=0

\*Mar 1 00:05:09.515: sess\_appl: ev(9=CC\_EV\_CALL\_DIGIT\_END), cid(5), disp(0)

\*Mar 1 00:05:09.515: cid(5)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_CALL\_DIGIT) oldst(SSA\_CS\_MAPPING)cfid(-1)csize(0)in(1)fDest(0)

\*Mar 1 00:05:09.515: ssaDigit

\*Mar 1 00:05:09.515: ssaDigit, 0. sct->digit , sct->digit len 0, usrDigit 2, digit\_tone\_mode=0

\*Mar 1 00:05:09.515: ssaDigit,1. callinfo.called , digit 2, callinfo.calling 1001, xrulecallingtag 0, xrulecalledtag 0

\*Mar 1 00:05:09.515: ssaDigit, 7. callinfo.calling 1001, sct->digit 2, result 1

\*Mar 1 00:05:09.635: cc\_api\_call\_digit\_begin (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x5, digit=0, digit\_begin\_flags=0x1, rtp\_timestamp=0xEB32A6E0 rtp\_expiration=0x0, dest\_mask=0x1)

\*Mar 1 00:05:09.635: sess\_appl: ev(10=CC\_EV\_CALL\_DIGIT\_BEGIN), cid(5), disp(0)

\*Mar 1 00:05:09.635: cid(5)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_DIGIT\_BEGIN) oldst(SSA\_CS\_MAPPING)cfid(-1)csize(0)in(1)fDest(0)

\*Mar 1 00:05:09.635: ssaIgnore cid(5), st(SSA\_CS\_MAPPING),oldst(0), ev(10)

\*Mar 1 00:05:09.695: cc\_api\_call\_digit\_end (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x5,digit=0,duration=95,xruleCallingTag=0,xruleCalledTag=0, dest\_mask=0x1), digit\_tone\_mode=0

\*Mar 1 00:05:09.695: sess\_appl: ev(9=CC\_EV\_CALL\_DIGIT\_END), cid(5), disp(0)

\*Mar 1 00:05:09.695: cid(5)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_CALL\_DIGIT) oldst(SSA\_CS\_MAPPING)cfid(-1)csize(0)in(1)fDest(0)

\*Mar 1 00:05:09.695: ssaDigit

\*Mar 1 00:05:09.695: ssaDigit, 0. sct->digit 2, sct->digit len 1, usrDigit 0,

digit\_tone\_mode=0

\*Mar 1 00:05:09.695: ssaDigit,1. callinfo.called , digit 20, callinfo.calling 1001, xrulecallingtag 0, xrulecalledtag 0

\*Mar 1 00:05:09.695: ssaDigit, 7. callinfo.calling 1001, sct->digit 20, result 1

\*Mar 1 00:05:09.815: cc\_api\_call\_digit\_begin (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x5, digit=0, digit\_begin\_flags=0x1, rtp\_timestamp=0xEB32A6E0 rtp\_expiration=0x0, dest\_mask=0x1)

\*Mar 1 00:05:09.815: sess\_appl: ev(10=CC\_EV\_CALL\_DIGIT\_BEGIN), cid(5), disp(0)

\*Mar 1 00:05:09.815: cid(5)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_DIGIT\_BEGIN)  
oldst(SSA\_CS\_MAPPING)cfid(-1)csz(0)in(1)fDest(0)

\*Mar 1 00:05:09.815: ssaIgnore cid(5), st(SSA\_CS\_MAPPING),oldst(0), ev(10)

\*Mar 1 00:05:09.875: cc\_api\_call\_digit\_end (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x5,digit=0,duration=95,xruleCallingTag=0,xruleCalledTag=0, dest\_mask=0x1), digit\_tone\_mode=0

\*Mar 1 00:05:09.875: sess\_appl: ev(9=CC\_EV\_CALL\_DIGIT\_END), cid(5), disp(0)

\*Mar 1 00:05:09.875: cid(5)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_CALL\_DIGIT)  
oldst(SSA\_CS\_MAPPING)cfid(-1)csz(0)in(1)fDest(0)

\*Mar 1 00:05:09.875: ssaDigit

\*Mar 1 00:05:09.875: ssaDigit, 0. sct->digit 20, sct->digit len 2, usrDigit 0, digit\_tone\_mode=0

\*Mar 1 00:05:09.875: ssaDigit,1. callinfo.called , digit 200, callinfo.calling 1001, xrulecallingtag 0, xrulecalledtag 0

\*Mar 1 00:05:09.875: ssaDigit, 7. callinfo.calling 1001, sct->digit 200, result 1

\*Mar 1 00:05:09.995: cc\_api\_call\_digit\_begin (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x5, digit=1, digit\_begin\_flags=0x1, rtp\_timestamp=0xEB32A6E0 rtp\_expiration=0x0, dest\_mask=0x1)

\*Mar 1 00:05:09.995: sess\_appl: ev(10=CC\_EV\_CALL\_DIGIT\_BEGIN), cid(5), disp(0)

\*Mar 1 00:05:09.995: cid(5)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_DIGIT\_BEGIN)  
oldst(SSA\_CS\_MAPPING)cfid(-1)csz(0)in(1)fDest(0)

\*Mar 1 00:05:09.995: ssaIgnore cid(5), st(SSA\_CS\_MAPPING),oldst(0), ev(10)

\*Mar 1 00:05:10.055: cc\_api\_call\_digit\_end (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x5,digit=1,duration=95,xruleCallingTag=0,xruleCalledTag=0, dest\_mask=0x1), digit\_tone\_mode=0

\*Mar 1 00:05:10.055: sess\_appl: ev(9=CC\_EV\_CALL\_DIGIT\_END), cid(5), disp(0)

\*Mar 1 00:05:10.055: cid(5)st(SSA\_CS\_MAPPING)ev(SSA\_EV\_CALL\_DIGIT)  
oldst(SSA\_CS\_MAPPING)cfid(-1)csz(0)in(1)fDest(0)

\*Mar 1 00:05:10.055: ssaDigit

\*Mar 1 00:05:10.055: ssaDigit, 0. sct->digit 200, sct->digit len 3, usrDigit 1, digit\_tone\_mode=0

\*Mar 1 00:05:10.055: ssaDigit,1. callinfo.called , digit 2001, callinfo.calling 1001, xrulecallingtag 0, xrulecalledtag 0

\*Mar 1 00:05:10.055: ssaDigit, 7. callinfo.calling 1001, sct->digit 2001, result 0

\*Mar 1 00:05:10.055: ccCallReportDigits (callID=0x5, enable=0x0)

\*Mar 1 00:05:10.055: cc\_api\_call\_report\_digits\_done (vdbPtr=0x6217C270, callID=0x5, disp=0)

\*Mar 1 00:05:10.055: ssaSetupPeer cid(5) peer list: tag(1) called number (2001)

\*Mar 1 00:05:10.055: ssaSetupPeer cid(5), destPat(2001), matched(1), prefix(), peer(622FB888), peer->encapType (2)

\*Mar 1 00:05:10.055: ccCallProceeding (callID=0x5, prog\_ind=0x0)

\*Mar 1 00:05:10.059: ccCallSetupRequest (Inbound call = 0x5, outbound peer =1, dest=, params=0x621129C8 mode=0, \*callID=0x6 2112D38, prog\_ind = 3) callingIE\_present 0

\*Mar 1 00:05:10.059: ccCallSetupRequest numbering\_type 0x81

\*Mar 1 00:05:10.059: ccCallSetupRequest encapType 2 clid\_restrict\_disable 1 null\_orig\_clg 1 clid\_transparent 0 callingNumber 1001

\*Mar 1 00:05:10.059: dest pattern 2..., called 2001, digit\_strip 0

\*Mar 1 00:05:10.059: callingNumber=1001, calledNumber=2001, redirectNumber= display\_info= calling\_oct3a=0

\*Mar 1 00:05:10.059: accountNumber=, finalDestFlag=0, guid=3f30.bb8e.14ef.11cc.8008.fdb1.2d0c.c4a5

\*Mar 1 00:05:10.059: peer\_tag=1

**\*Mar 1 00:05:10.059: ccIFCallSetupRequestPrivate: (vdbPtr=0x620BCAF0, dest=, callParams={called=2001,called\_oct3=0x81, calling=1001,calling\_oct3=0x0, calling\_xlated=false, subscriber\_type\_str=RegularLine, fdest=0, voice\_peer\_tag=1},mode=0x0) vdbp**

tr type = 1

```
*Mar 1 00:05:10.059: ccIFCallSetupRequestPrivate: (vdbPtr=0x620BCAF0, dest=, callParams=
{called=2001, called_oct3 0x81, calling=1001,calling_oct3 0x0, calling_xlated=false,
fdest=0, voice_peer_tag=1}, mode=0x0, xltrc=-5)
*Mar 1 00:05:10.059: ccSaveDialpeerTag (callID=0x5, dialpeer_tag=0x1)
*Mar 1 00:05:10.059: ccCallSetContext (callID=0x6, context=0x61DAD8A0)
*Mar 1 00:05:10.059: sess_appl: ev(53=CC_EV_CALL_REPORT_DIGITS_DONE), cid(5), disp(0)
*Mar 1 00:05:10.059: cid(5)st(SSA_CS_CALL_SETTING)ev(SSA_EV_CALL_REPORT_DIGITS_DONE)
oldst(SSA_CS_MAPPING)cfid(-1)csize(0)in(1)fDest(0)
*Mar 1 00:05:10.059: -cid2(6)st2(SSA_CS_CALL_SETTING)oldst2(SSA_CS_MAPPING)
*Mar 1 00:05:10.059: ssaReportDigitsDone cid(5) peer list: (empty)
*Mar 1 00:05:10.059: ssaReportDigitsDone callid=5 Reporting disabled.
*Mar 1 00:05:10.063: dsp_digit_collect_off: [3/0/0] packet_len=8 channel_id=128 packet_id=
36
*Mar 1 00:05:10.063: dsp_soutput: [3/0/0]
*Mar 1 00:05:10.063: htsp_process_event: [3/0/0, FXSLS_OFFHOOK, E_HTSP_PROCEEDING]
*Mar 1 00:05:10.095: cc_api_call_proceeding(vdbPtr=0x620BCAF0, callID=0x6,
prog_ind=0x0)
*Mar 1 00:05:10.099: sess_appl: ev(21=CC_EV_CALL_PROCEEDING), cid(6), disp(0)
*Mar 1 00:05:10.099: cid(6)st(SSA_CS_CALL_SETTING)ev(SSA_EV_CALL_PROCEEDING)
oldst(SSA_CS_MAPPING)cfid(-1)csize(0)in(0)fDest(0)
*Mar 1 00:05:10.099: -cid2(5)st2(SSA_CS_CALL_SETTING)oldst2(SSA_CS_CALL_SETTING)
*Mar 1 00:05:10.099: ssaCallProc
*Mar 1 00:05:10.099: ccGetDialpeerTag (callID=0x5)
*Mar 1 00:05:10.099: ssaIgnore cid(6), st(SSA_CS_CALL_SETTING),oldst(1), ev(21)
*Mar 1 00:05:10.103: cc_api_call_cut_progress(vdbPtr=0x620BCAF0, callID=0x6, prog_ind=0x8,
sig_ind=0x1)
*Mar 1 00:05:10.103: sess_appl: ev(22=CC_EV_CALL_PROGRESS), cid(6), disp(0)
*Mar 1 00:05:10.107: cid(6)st(SSA_CS_CALL_SETTING)ev(SSA_EV_CALL_PROGRESS)
oldst(SSA_CS_CALL_SETTING)cfid(-1)csize(0)in(0)fDest(0)
*Mar 1 00:05:10.107: -cid2(5)st2(SSA_CS_CALL_SETTING)oldst2(SSA_CS_CALL_SETTING)
*Mar 1 00:05:10.107: ssaCutProgress
*Mar 1 00:05:10.107: ccGetDialpeerTag (callID=0x5)
*Mar 1 00:05:10.107: ccCallCutProgress (callID=0x5, prog_ind=0x8, sig_ind=0x1)
*Mar 1 00:05:10.107: ccConferenceCreate (confID=0x6211310C, callID1=0x5,
callID2=0x6, tag=0x0)
*Mar 1 00:05:10.107: cc_api_bridge_done (confID=0x3, srcIF=0x620BCAF0, srcCallID=0x6,
dstCallID=0x5, disposition=0, tag=0x0)htsp_alert_notify
*Mar 1 00:05:10.107: cc_api_bridge_done (confID=0x3, srcIF=0x6217C270, srcCallID=0x5,
dstCallID=0x6, disposition=0, tag=0x0)
*Mar 1 00:05:10.107: cc_api_caps_ind (dstVdbPtr=0x620BCAF0, dstCallId=0x6, srcCallId=0x5,
caps={codec=0x2EBFB, fax_rate=0x7F, vad=0x3, modem=0x2 codec_bytes=0, signal_type=3})
*Mar 1 00:05:10.107: cc_api_caps_ind (Playout: mode 1, initial 60,min 40, max 200)
*Mar 1 00:05:10.111: cc_api_caps_ind (dstVdbPtr=0x6217C270, dstCallId=0x5, srcCallId=0x6,
caps={codec=0x4, fax_rate=0x2, vad=0x1, modem=0x0 codec_bytes=20, signal_type=2})
*Mar 1 00:05:10.111: cc_api_caps_ind (Playout: mode 1, initial 60,min 40, max 200)
*Mar 1 00:05:10.111: cc_api_caps_ack (dstVdbPtr=0x6217C270, dstCallId=0x5, srcCallId=0x6,
caps={codec=0x4, fax_rate=0x2, vad=0x1, modem=0x0 codec_bytes=20, signal_type=2,
seq_num_start=9062})
*Mar 1 00:05:10.111: cc_api_caps_ack (dstVdbPtr=0x620BCAF0, dstCallId=0x6, srcCallId=0x5,
caps={codec=0x4, fax_rate=0x2, vad=0x1, modem=0x0 codec_bytes=20, signal_type=2,
seq_num_start=9062})
*Mar 1 00:05:10.111: cc_api_voice_mode_event , callID=0x5
*Mar 1 00:05:10.111: Call Pointer =620005E8
*Mar 1 00:05:10.115: cc_api_caps_ind (dstVdbPtr=0x6217C270, dstCallId=0x5, srcCallId=0x6,
caps={codec=0x4, fax_rate=0x2, vad=0x1, modem=0x0 codec_bytes=20, signal_type=2})
*Mar 1 00:05:10.115: cc_api_caps_ind (Playout: mode 1, initial 60,min 40, max 200)
*Mar 1 00:05:10.115: cc_api_caps_ack (dstVdbPtr=0x6217C270, dstCallId=0x5, srcCallId=0x6,
caps={codec=0x4, fax_rate=0x2, vad=0x1, modem=0x0 codec_bytes=20, signal_type=2,
seq_num_start=9062})
*Mar 1 00:05:10.123: cc_api_caps_ack (dstVdbPtr=0x620BCAF0, dstCallId=0x6, srcCallId=0x5,
caps={codec=0x4, fax_rate=0x2, vad=0x1, modem=0x0 codec_bytes=20, signal_type=2,
seq_num_start=9062})
*Mar 1 00:05:10.123: cc_api_voice_mode_event , callID=0x5
```



\*Mar 1 00:05:10.123: Call Pointer =620005E8  
\*Mar 1 00:05:10.123: htsp\_process\_event: [3/0/0, FXSLS\_OFFHOOK, E\_HTSP\_VOICE\_CUT\_THROUGH]  
\*Mar 1 00:05:10.123: htsp\_process\_event: [3/0/0, FXSLS\_OFFHOOK, E\_HTSP\_VOICE\_CUT\_THROUGH]  
\*Mar 1 00:05:10.123: sess\_appl: ev(29=CC\_EV\_CONF\_CREATE\_DONE), cid(5), disp(0)  
\*Mar 1 00:05:10.123: cid(5)st(SSA\_CS\_CONFERENCING\_PROGRESS)ev(SSA\_EV\_CONF\_CREATE\_DONE)  
oldst(SSA\_CS\_CALL\_SETTING)cfid(3)csize(0)in(1)fDest(0)  
\*Mar 1 00:05:10.127: -cid2(6)st2(SSA\_CS\_CONFERENCING\_PROGRESS)oldst2(SSA\_CS\_CALL\_SETTING)  
\*Mar 1 00:05:10.127: ssaConfCreateDoneAlert  
\*Mar 1 00:05:10.127: sess\_appl: ev(51=CC\_EV\_VOICE\_MODE\_DONE), cid(5), disp(0)  
\*Mar 1 00:05:10.127: cid(5)st(SSA\_CS\_CONFERENCED\_ALERT)ev(SSA\_EV\_VOICE\_MODE\_DONE)  
oldst(SSA\_CS\_CONFERENCING\_PROGRESS)cfid(3)csize(0)in(1)fDest(0)  
\*Mar 1 00:05:10.127: -cid2(6)st2(SSA\_CS\_CONFERENCED\_ALERT)oldst2(SSA\_CS\_CALL\_SETTING)  
\*Mar 1 00:05:10.127: ssaIgnore cid(5), st(SSA\_CS\_CONFERENCED\_ALERT),oldst(4), ev(51)  
\*Mar 1 00:05:10.127: sess\_appl: ev(51=CC\_EV\_VOICE\_MODE\_DONE), cid(5), disp(2)  
\*Mar 1 00:05:10.127: cid(5)st(SSA\_CS\_CONFERENCED\_ALERT)ev(SSA\_EV\_VOICE\_MODE\_DONE)  
oldst(SSA\_CS\_CONFERENCED\_ALERT)cfid(3)csize(0)in(1)fDest(0)  
\*Mar 1 00:05:10.127: -cid2(6)st2(SSA\_CS\_CONFERENCED\_ALERT)oldst2(SSA\_CS\_CALL\_SETTING)  
\*Mar 1 00:05:10.127: ssaIgnore cid(5), st(SSA\_CS\_CONFERENCED\_ALERT),oldst(4), ev(51)  
\*Mar 1 00:05:10.127: cc\_process\_notify\_bridge\_done (event=0x6210BDB8)  
\*Mar 1 00:05:10.131: cc\_api\_caps\_ind (dstVdbPtr=0x6217C270, dstCallId=0x5, srcCallId=0x6,  
caps={codec=0x4, fax\_rate=0x2, vad=0x1, modem=0x0 codec\_bytes=20, signal\_type=2})  
\*Mar 1 00:05:10.131: cc\_api\_caps\_ind (Playout: mode 1, initial 60,min 40, max 200)  
\*Mar 1 00:05:10.131: cc\_api\_caps\_ack (dstVdbPtr=0x6217C270, dstCallId=0x5, srcCallId=0x6,  
caps={codec=0x4, fax\_rate=0x2, vad=0x1, modem=0x0 codec\_bytes=20, signal\_type=2,  
seq\_num\_start=9063})  
\*Mar 1 00:05:10.131: cc\_api\_caps\_ind (dstVdbPtr=0x6217C270, dstCallId=0x5, srcCallId=0x6,  
caps={codec=0x4, fax\_rate=0x2, vad=0x1, modem=0x0 codec\_bytes=20, signal\_type=2})  
\*Mar 1 00:05:10.131: cc\_api\_caps\_ind (Playout: mode 1, initial 60,min 40, max 200)  
\*Mar 1 00:05:10.131: cc\_api\_caps\_ack (dstVdbPtr=0x6217C270, dstCallId=0x5, srcCallId=0x6,  
caps={codec=0x4, fax\_rate=0x2, vad=0x1, modem=0x0 codec\_bytes=20, signal\_type=2,  
seq\_num\_start=9063})  
\*Mar 1 00:05:10.135: cc\_api\_caps\_ack (dstVdbPtr=0x620BCAF0, dstCallId=0x6, srcCallId=0x5,  
caps={codec=0x4, fax\_rate=0x2, vad=0x1, modem=0x0 codec\_bytes=20, signal\_type=2,  
seq\_num\_start=9063})  
\*Mar 1 00:05:10.135: cc\_api\_voice\_mode\_event , callID=0x5  
\*Mar 1 00:05:10.135: Call Pointer =620005E8  
**\*Mar 1 00:05:10.135: cc\_api\_caps\_ack (dstVdbPtr=0x620BCAF0, dstCallId=0x6,  
srcCallId=0x5, caps={codec=0x4, fax\_rate=0x2, vad=0x1, modem=0x0 codec\_bytes=20,  
signal\_type=2, seq\_num\_start=9063})**  
\*Mar 1 00:05:10.135: cc\_api\_voice\_mode\_event , callID=0x5  
\*Mar 1 00:05:10.135: Call Pointer =620005E8  
\*Mar 1 00:05:10.135: htsp\_process\_event: [3/0/0, FXSLS\_OFFHOOK, E\_HTSP\_VOICE\_CUT\_THROUGH]  
\*Mar 1 00:05:10.135: htsp\_process\_event: [3/0/0, FXSLS\_OFFHOOK, E\_HTSP\_VOICE\_CUT\_THROUGH]  
\*Mar 1 00:05:10.135: sess\_appl: ev(51=CC\_EV\_VOICE\_MODE\_DONE), cid(5), disp(0)  
\*Mar 1 00:05:10.135: cid(5)st(SSA\_CS\_CONFERENCED\_ALERT)ev(SSA\_EV\_VOICE\_MODE\_DONE)  
oldst(SSA\_CS\_CONFERENCED\_ALERT)cfid(3)csize(0)in(1)fDest(0)  
\*Mar 1 00:05:10.135: -cid2(6)st2(SSA\_CS\_CONFERENCED\_ALERT)oldst2(SSA\_CS\_CALL\_SETTING)  
\*Mar 1 00:05:10.135: ssaIgnore cid(5), st(SSA\_CS\_CONFERENCED\_ALERT),oldst(4), ev(51)  
\*Mar 1 00:05:10.135: sess\_appl: ev(51=CC\_EV\_VOICE\_MODE\_DONE), cid(5), disp(0)  
\*Mar 1 00:05:10.135: cid(5)st(SSA\_CS\_CONFERENCED\_ALERT)ev(SSA\_EV\_VOICE\_MODE\_DONE)  
oldst(SSA\_CS\_CONFERENCED\_ALERT)cfid(3)csize(0)in(1)fDest(0)  
\*Mar 1 00:05:10.139: -cid2(6)st2(SSA\_CS\_CONFERENCED\_ALERT)oldst2(SSA\_CS\_CALL\_SETTING)  
\*Mar 1 00:05:10.139: ssaIgnore cid(5), st(SSA\_CS\_CONFERENCED\_ALERT),oldst(4), ev(51)  
\*Mar 1 00:05:18.303: cc\_api\_call\_connected(vdbPtr=0x620BCAF0, callID=0x6), prog\_ind =  
2cc\_api\_call\_connected: setting callEntry->connected to TRUE  
  
\*Mar 1 00:05:18.303: sess\_appl: ev(8=CC\_EV\_CALL\_CONNECTED), cid(6), disp(0)  
\*Mar 1 00:05:18.303: cid(6)st(SSA\_CS\_CONFERENCED\_ALERT)ev(SSA\_EV\_CALL\_CONNECTED)  
oldst(SSA\_CS\_CALL\_SETTING)cfid(3)csize(0)in(0)fDest(0)  
\*Mar 1 00:05:18.307: -cid2(5)st2(SSA\_CS\_CONFERENCED\_ALERT)oldst2(SSA\_CS\_CONFERENCED\_ALERT)  
\*Mar 1 00:05:18.307: ssaConnectAlert  
\*Mar 1 00:05:18.307: ccGetDialpeerTag (callID=0x5)  
**\*Mar 1 00:05:18.307: ccCallConnect (callID=0x5), prog\_ind = 2ccCallConnect:  
setting callEntry->connected to TRUE**

\*Mar 1 00:05:18.307: ssaFlushPeerTagQueue cid(5) peer list: (empty)htsp\_connect: no\_offhook 0

\*Mar 1 00:05:18.307: htsp\_process\_event: [3/0/0, FXSLS\_OFFHOOK, E\_HTSP\_CONNECT]fxsls\_offhook\_connect

\*Mar 1 00:05:18.307: [3/0/0] set signal state = 0x6 timestamp = 0

\*Mar 1 00:05:18.307: dsp\_set\_sig\_state: [3/0/0] packet\_len=12 channel\_id=128 packet\_id=39 state=0x6 timestamp=0x0

\*Mar 1 00:05:18.307: dsp\_soutput: [3/0/0]

SanJose3640A#

SanJose3640A#

SanJose3640A#

SanJose3640A#! call connected

SanJose3640A#

SanJose3640A#

SanJose3640A#

SanJose3640A#

SanJose3640A#! 1001 disconnecting the call

SanJose3640A#

SanJose3640A#

SanJose3640A#

SanJose3640A#

SanJose3640A#

\*Mar 1 00:05:57.019: htsp\_dsp\_message: SEND/RESP\_SIG\_STATUS: state=0x4 timestamp=16952 systime=35702

\*Mar 1 00:05:57.019: htsp\_process\_event: [3/0/0, FXSLS\_CONNECT, E\_DSP\_SIG\_0100]fxsls\_offhook\_onhook, HF duration=500

\*Mar 1 00:05:57.023: htsp\_timer - 500 msec

\*Mar 1 00:05:57.523: htsp\_process\_event: [3/0/0, FXSLS\_CONNECT, E\_HTSP\_EVENT\_TIMER]fxsls\_connect\_wait\_release\_req

\*Mar 1 00:05:57.523: htsp\_timer\_stop

\*Mar 1 00:05:57.523: cc\_api\_call\_disconnected(vdbPtr=0x6217C270, callID=0x5, cause=0x10)

\*Mar 1 00:05:57.523: sess\_appl: ev(11=CC\_EV\_CALL\_DISCONNECTED), cid(5), disp(0)

\*Mar 1 00:05:57.523: cid(5)st(SSA\_CS\_ACTIVE)ev(SSA\_EV\_CALL\_DISCONNECTED)  
oldst(SSA\_CS\_CONFERENCED\_ALERT)cfid(3)csize(0)in(1)fDest(0)

\*Mar 1 00:05:57.523: -cid2(6)st2(SSA\_CS\_ACTIVE)oldst2(SSA\_CS\_CONFERENCED\_ALERT)

\*Mar 1 00:05:57.523: ssa: Disconnected cid(5) state(5) cause(0x10)

\*Mar 1 00:05:57.523: ccConferenceDestroy (confID=0x3, tag=0x0)

\*Mar 1 00:05:57.523: cc\_api\_bridge\_drop\_done (confID=0x3, srcIF=0x620BCAF0, srcCallID=0x6, dstCallID=0x5, disposition=0 tag=0x0)

\*Mar 1 00:05:57.523: cc\_api\_bridge\_drop\_done (confID=0x3, srcIF=0x6217C270, srcCallID=0x5, dstCallID=0x6, disposition=0 tag=0x0)

\*Mar 1 00:05:57.523: sess\_appl: ev(30=CC\_EV\_CONF\_DESTROY\_DONE), cid(5), disp(0)

\*Mar 1 00:05:57.523: cid(5)st(SSA\_CS\_CONF\_DESTROYING)ev(SSA\_EV\_CONF\_DESTROY\_DONE)  
oldst(SSA\_CS\_ACTIVE)cfid(-1)csize(0)in(1)fDest(0)

\*Mar 1 00:05:57.527: -cid2(6)st2(SSA\_CS\_CONF\_DESTROYING)oldst2(SSA\_CS\_CONFERENCED\_ALERT)

\*Mar 1 00:05:57.527: ssaConfDestroyDone

\*Mar 1 00:05:57.527: ccCallDisconnect (callID=0x5, cause=0x10 tag=0x0)

\*Mar 1 00:05:57.527: ccCallDisconnect: existing\_cause = 0x0, **new\_cause = 0x10**

\*Mar 1 00:05:57.527: ccCallDisconnect (callID=0x6, cause=0x10 tag=0x0)

\*Mar 1 00:05:57.527: ccCallDisconnect: existing\_cause = 0x0, new\_cause = 0x10htsp\_release\_req: cause 16, no\_onhook 0

\*Mar 1 00:05:57.531: htsp\_process\_event: [3/0/0, FXSLS\_WAIT\_RELEASE\_REQ, E\_HTSP\_RELEASE\_REQ] fxsls\_waitrls\_req\_rls

\*Mar 1 00:05:57.531: [3/0/0] set signal state = 0x4 timestamp = 0

\*Mar 1 00:05:57.531: dsp\_set\_sig\_state: [3/0/0] packet\_len=12 channel\_id=128 packet\_id=39 state=0x4 timestamp=0x0

\*Mar 1 00:05:57.531: dsp\_soutput: [3/0/0]htsp\_report\_onhook\_sig

\*Mar 1 00:05:57.531: cc\_api\_call\_feature: (vdbPtr=0x6217C270, callID=0x5, feature\_ind.type=5

\*Mar 1 00:05:57.535: cc\_api\_call\_disconnect\_done(vdbPtr=0x6217C270, callID=0x5, disp=0, tag=0x0)

\*Mar 1 00:05:57.535: hdsprm\_close\_cleanup

```
*Mar 1 00:05:57.535: sess_appl: ev(28=CC_EV_CALL_FEATURE), cid(5), disp(0)
*Mar 1 00:05:57.535: cid(5)st(SSA_CS_DISCONNECTING)ev(SSA_EV_CALL_FEATURE)
  oldst(SSA_CS_CONF_DESTROYING)cfid(-1)csize(0)in(1)fDest(0)
*Mar 1 00:05:57.535: -cid2(6)st2(SSA_CS_DISCONNECTING)oldst2(SSA_CS_CONFERENCED_ALERT)
*Mar 1 00:05:57.535: ssaIgnore cid(5), st(SSA_CS_DISCONNECTING),oldst(7), ev(28)
*Mar 1 00:05:57.539: sess_appl: ev(12=CC_EV_CALL_DISCONNECT_DONE), cid(5), disp(0)
*Mar 1 00:05:57.539: cid(5)st(SSA_CS_DISCONNECTING)ev(SSA_EV_CALL_DISCONNECT_DONE)
  oldst(SSA_CS_DISCONNECTING)cfid(-1)csize(0)in(1)fDest(0)
*Mar 1 00:05:57.539: -cid2(6)st2(SSA_CS_DISCONNECTING)oldst2(SSA_CS_CONFERENCED_ALERT)
*Mar 1 00:05:57.539: ssaDisconnectDone
*Mar 1 00:05:57.543: cc_api_icpif: expect factor = 0
*Mar 1 00:05:57.543: g113_calculate_impairment (delay=101,loss=0), Io=0 Iq=0 Idte=0 Idd=0
  Ie=9 Itot=9
*Mar 1 00:05:57.543: cc_api_call_disconnect_done(vdbPtr=0x620BCAF0, callID=0x6, disp=0,
  tag=0x0)
*Mar 1 00:05:57.547: sess_appl: ev(12=CC_EV_CALL_DISCONNECT_DONE), cid(6), disp(0)
*Mar 1 00:05:57.547: cid(6)st(SSA_CS_DISCONNECTING)ev(SSA_EV_CALL_DISCONNECT_DONE)
  oldst(SSA_CS_CONFERENCED_ALERT)cfid(-1)csize(1)in(0)fDest(0)
*Mar 1 00:05:57.547: ssaDisconnectDone
SanJose3640A#
```

SanJose3640A#

## Zugehörige Informationen

- [VoIP über PPP-Links mit Quality of Service \(LLQ/IP RTP-Priorität, LFI, cRTP\)](#)
- [VoIP over Frame Relay mit Quality of Service \(Fragmentierung, Traffic Shaping, LLQ/IP RTP-Priorität\)](#)
- [VoIP QoS für Frame Relay an ATM Interworking mit LLQ, PPP LFI und cRTP](#)
- [Informationen zu DFÜ-Peers und Anruflisten auf Cisco IOS-Plattformen](#)
- [Problembehandlung und Debuggen der Grundlagen von VoIP-Anrufen](#)
- [Unterstützung von Sprachtechnologie](#)
- [Produkt-Support für Sprach- und IP-Kommunikation](#)
- [Fehlerbehebung bei Cisco IP-Telefonie](#)
- [Technischer Support und Dokumentation](#)