

# Configurando o tronco de conexão para gateways VoIP

## Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenções](#)

[Configurar](#)

[Considerações e limitações do design do tronco de conexão](#)

[Diretrizes de configuração](#)

[Diagrama de Rede](#)

[Configurações](#)

[Verificar](#)

[Troubleshooting](#)

[Comandos para Troubleshooting](#)

[Informações Relacionadas](#)

## Introdução

Um tronco (tie-line) é uma linha de comunicação ponto a ponto permanente entre duas portas de voz. [O comando de tronco de conexão cria uma chamada permanente de Voz sobre IP \(VoIP\) entre dois gateways VoIP.](#) Simula uma conexão de tronco através da criação de troncos virtuais (tie-lines) entre dois pontos finais de telefonia. Para os sistemas conectados, é como se um tronco T1 estivesse conectado diretamente entre eles.

## Pré-requisitos

### Requisitos

Estas Plataformas apoiam um tronco da conexão voip:

- Cisco 2600, 3600, e interfaces digitais e analógicas do 3700 Series
- Interfaces digital da série do Cisco 7200/7500
- Interfaces digitais e analógicas de Cisco MC3810
- Cisco 1750/1751 e 1760

**Nota:** As Plataformas AS5300/AS5400/AS5800 não fazem e não apoiarão troncos de conexão, porque não são apropriadas para a conectividade de WAN com volumes de tráfego pesado.

## Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- Cisco IOS® Software Versão 12.2(10a) com Conjunto de Recursos de IP Plus
- Cisco 2610 Series Routers

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a sua rede estiver ativa, certifique-se de que entende o impacto potencial de qualquer comando.

## Convenções

Para obter mais informações sobre convenções de documento, consulte as [Convenções de dicas técnicas Cisco](#).

## Configurar

Nesta seção, você encontrará informações para configurar os recursos descritos neste documento.

**Nota:** Para encontrar informações adicionais sobre os comandos usados neste documento, use a ferramenta [Consulta de Comando](#) (somente para clientes [registrados](#))

## Considerações e limitações do design do tronco de conexão

- O modo de tronco de conexão é suportado nas interfaces CAS (sinalização associada a canal) T1/E1. Um tronco de conexão não é apoiado nas relações T1/E1 que estão usando o Common Channel Signaling (CCS); por exemplo, QSIG e PRI Q.931. Um tronco de conexão não é suportado nas portas FXO (Foreign Exchange Office) configuradas para início terra.
- O modo de tronco de conexão é uma conexão permanente; a chamada VoIP é conectada sempre independentemente da porta do serviço de telefonia tradicional (POTS) que é em-gancho ou fora-gancho. O tronco de conexão tem estaticamente pontos finais configurados e não exige um usuário discar para conectar atendimentos. Igualmente permite que a sinalização de chamada suplementar, tal como o hookflash ou o Point-to-Point hoot-n-holler, seja passada sobre a rede IP entre os dois dispositivos de telefonia.
- O modo de tronco de conexão é apoiado com estas combinações da porta de voz:receba e transmita (E & M) a E & M (o mesmo tipo)FXO para Foreign Exchange Station (FXS)FXS ao FXS (sem a sinalização)**Nota:** Estas combinações da porta de voz são permitidas entre o analógico ao analógico, digital às relações digitais, e analógicas-numéricas. Também, quando você está configurando o FXS ao FXS, sinalizar não pode ser transportada porque não seria um trajeto transparente. Os dispositivos conectados (FXOs) estariam tentando sinalizar um para o outro. É possível conseguir este projeto trabalhar se você ajusta o trajeto da voz sempre para estar aberto. Configurar o `signal-type ext-signal` ao dial peer de VOIP, e o roteador já não esperará a sinalização antes que abra o trajeto da voz.
- Um mapeamento T1 CAS a E1 CAS do tronco de conexão não funciona por padrão. a manipulação da Bit-ordem nos gateways deve ser executada e não pode sempre trabalhar, com base no apoio PBX da vária sinalização de bit ABCD.

- Um tronco de conexão permite a linha privada, tipo automático da ringdown-Fora-Local-extensão (PLAR-OPX) de funcionalidade entre portas FXO e FXS. Isto permite que as estações remotas (conectadas às portas FXS) apareçam ao PBX como estações fisicamente conectadas. Se esta estação remota não responde a um atendimento, pode ser virada ao correio de voz centralizado (se é configurada no PBX).
- Um tronco de conexão, tal como o PLAR, não exige o roteador recolher dígitos do dispositivo de telefonia. A chamada VOIP permanente está criada quando o roteador é carregado e a conectividade IP está estabelecida. Devido a isto, o Plano de discagem do cliente existente não tem de ser alterado.
- Um tronco de conexão pode transferir alguma sinalização telefônica, como hookflash, mas não transmite sinalização PBX proprietária. Não é um recurso Transparent CCS (T-CSS).
- Um tronco de conexão, como o PLAR, é definido por porta de voz. Isso significa que a porta de voz não pode operar no modo Tronco de Conexão e no modo Coletar Dígitos Discados. O único exemplo onde este não pôde ser completamente desejável estaria em um escritório remoto que precisasse de discar igualmente entre extensões locais sem o uso de um PBX centralizado. Isto exigiria o trajeto do atendimento ir sobre a rede voip e a parte traseira, ao contrário dele que está sendo comutado dentro do roteador. Em geral, isso não é um problema.

## Diretrizes de configuração

É necessário configurar o tronco de conexão em suas duas extremidades. Quando você está configurando um tronco de conexão com interfaces analógicas, deve ser definido pela porta de voz. Quando você está configurando um tronco de conexão com interfaces digitais, há diversas opções:

- Você pode definir um [comando ds0-group](#) separado para cada DS0 (cada intervalo de tempo), e você pode usar o [comando connection trunk](#) para definir cada porta de voz que é criada. Isto assegura-se de que o DS0 ao mapeamento DS0 esteja retido em troncos digitais.
- Você pode definir um único [comando ds0-group](#) para segurar todos os DS0, e você pode definir um [comando trunk da conexão única na](#) porta de voz. Isto reduz a quantidade de configuração manual que é exigida, mas não há nenhuma garantia do mapeamento um a um dos DS0 em uma ou outra extremidade do tronco. Além, cada vez que isso os recarregamentos de roteador, o mapeamento pode ser diferente da última vez. Além disso, esta configuração complica o Troubleshooting, porque você não é capaz de isolar o problema aos únicos (ou mesmo alguns) intervalos de tempo sem tomar abaixo do grupo de troncos inteiro. Esta configuração não é recomendada igualmente para o T-CCS com a sinalização de propriedade em um ou outro fim dos PBX, porque não entregaria o canal de sinalização confiantemente sem mapeamento cara-a-cara.
- Recomenda-se que um lado da conexão esteja configurado com o **answer-mode keyword** especificado após o **comando connection trunk string**. Isto faz a um lado do tronco o “lado de mestre.” O gateway (roteador) com o **answer-mode keyword** é então do “o lado escravo.” O **comando answer-mode** especifica que o gateway não tentará iniciar uma conexão de tronco, mas pelo contrário esperará uma chamada recebida antes que estabeleça o tronco. Este esquema da configuração minimiza o tempo que o Roteadores toma para trazer acima a troncos e se assegura de que os troncos vão abaixo de quando as conexões são perdidas entre dois gateways. Se não, os gateways não puderam tentar restabelecer o tronco quando a conexão está acima outra vez.

**Nota:** Quando você emite o **comando connection trunk**, você deve executar uma sequência de

comando shutdown/no shutdown na porta de voz.

## Diagrama de Rede

Este documento usa estas duas instalações de rede:

O diagrama precedente ilustra a encenação digital-à-digital, onde ambos os lados do roteador têm links digitais.

O diagrama precedente ilustra a encenação numérico-analógica, com o digital em uns extremidade e analógico na outra extremidade.

## Configurações

Este documento utiliza as seguintes configurações:

- Digital para digital [maui-slt-01maui-rtr-07](#)
- Digital para analógico [maui-slt-01maui-rtr-07](#)

A primeira configuração (digital-à-digital) mostra uma configuração típica para um tronco de conexão entre dois Roteadores com relações T1 digitais. Neste exemplo, o Roteadores está fornecendo a substituição de tie-line verdadeira entre os PBX.

### **Digital para digital - maui-slt-01**

```
version 12.2
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname maui-slt-01
!
voice-card 1
!
controller T1 1/0
framing esf
linecode b8zs
ds0-group 1 timeslots 1 type e & m-wink-start ds0-group
2 timeslots 2 type e & m-wink-start clock source line !-
-- The ds0-group command creates the logical voice-
ports: !--- voice-port 1/0:1 and voice-port 1/0:2. !
voice-port 1/0:1 connection trunk 2000 !--- "master
side" !--- This starts the trunk connection using digits
2000 to match !--- a VoIP dial-peer. The digits are
generated internally by the !--- router and are not
received from the voice-port. ! voice-port 1/0:2
connection trunk 2001 ! dial-peer voice 2 voip
destination-pattern 200. !--- Matches connection trunk
string 2000 and 2001. dtmf-relay h245-alphanumeric
session target ipv4:192.168.100.2 ip qos dscp cs5 media
! dial-peer voice 1 pots destination-pattern 1000 port
1/0:1 !--- This dial-peer maps to maui-rtr-07's voice-
port 1/0:1. ! dial-peer voice 3 pots destination-pattern
1001 port 1/0:2 !--- This dial-peer maps to maui-rtr-
07's voice-port 1/0:2. ! interface Serial0/1 ip address
192.168.100.1 255.255.255.0
```

### **Digital para digital - maui-rtr-07**

```
version 12.2
```

```

service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
hostname maui-rtr-07
!
voice-card 1
!
controller T1 1/0
    framing esf
    linecode b8zs
ds0-group 1 timeslots 1 type e & m-wink-start ds0-group
2 timeslots 2 type e & m-wink-start clock source line !
voice-port 1/0:1 connection trunk 1000 answer-mode !---
"slave side" !--- The answer-mode specifies that the
router should not attempt !--- to initiate a trunk
connection, but it should wait for an !--- incoming call
before it establishes the trunk. ! voice-port 1/0:2
connection trunk 1001 answer-mode ! dial-peer voice 1
voip destination-pattern 100. dtmf-relay h245-
alphanumeric session target ipv4:192.168.100.1 ip qos
dscp cs5 media ! dial-peer voice 2 pots destination-
pattern 2000 port 1/0:1 !--- This dial-peer terminates
the connection !--- from maui-slt-01 voice-port 1/0:1. !
dial-peer voice 3 pots destination-pattern 2001 port
1/0:2 !--- This dial-peer terminates the connection !---
from maui-slt-01 voice-port 1/0:2. ! interface Serial0/1
ip address 192.168.100.2 255.255.255.0 clockrate 128000
!

```

A segunda configuração (numérico-analógica) mostra uma configuração típica para um tronco de conexão entre dois Roteadores similares, um com relações T1 digitais e outro com interfaces análogas. As relações devem ser o mesmo tipo para que esta trabalhe (por exemplo, permissão e & m à permissão e & m, E & M imediato a E & M imediato, FXO ao FXS e vice-versa). Em nosso exemplo, o início de circuito de FXO está sinalizando na relação T1 digital e há portas FXS Analógicos com a sinalização de loopstart FXS no lado correspondente.

### Numérico-analógico - maui-slt-01

```

version 12.2
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname maui-slt-01
!
voice vad-time 40000
!
voice-card 1
!
controller T1 1/0
    framing esf
    linecode b8zs
ds0-group 1 timeslots 1 type fxo-loopstart clock source
line !--- The ds0-group command creates the logical
voice-ports: !--- voice-port 1/0:1 and voice-port 1/0:2.
! voice-port 1/0:1 connection trunk 2000 !--- "master
side" !--- This starts the trunk connection using digits
2000 to match !--- a VoIP dial-peer. The digits are
generated internally by the !--- router and are not

```

```

received from the voice-port. !!! dial-peer voice 2
voip destination-pattern 200. !--- Matches connection
trunk string 2000 and 2001. dtmf-relay h245-alphanumeric
session target ipv4:192.168.100.2 ip qos dscp cs5 media
! dial-peer voice 1 pots destination-pattern 1000 port
1/0:1 !--- This dial-peer maps to maui-rtr-07's voice-
port 1/0/0. !!! interface Serial0/1 ip address
192.168.100.1 255.255.255.0 !

```

## Numérico-analógico - maui-rtr-07

```

version 12.2
service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
hostname maui-rtr-07
!
!
voice-port 1/0/0 connection trunk 1000 answer-mode !---
"slave side" !--- The answer-mode specifies that the
router should not attempt !--- to initiate a trunk
connection, but it should wait for an !--- incoming call
before it establishes the trunk. !! dial-peer voice 1
voip destination-pattern 100. dtmf-relay h245-
alphanumeric session target ipv4:192.168.100.1 ip qos
dscp cs5 media ! dial-peer voice 2 pots destination-
pattern 2000 port 1/0/0 !--- This dial-peer terminates
the connection !--- from maui-slt-01 voice-port 1/0:1. !
!! interface Serial0/1 ip address 192.168.100.2
255.255.255.0 clockrate 128000 !

```

## Verificar

Esta seção fornece a informação que você pode usar para confirmar que sua configuração está trabalhando corretamente.

A [Output Interpreter Tool \(somente clientes registrados\)](#) oferece suporte a determinados comandos show, o que permite exibir uma análise da saída do comando show.

- [mostre o sumário da chamada de voz](#) — Usado para verificar que todos os troncos estão acima e no estado `S_CONNECT`.

Quando os troncos se tornarem ativos, o console exibirá a mensagem `%HTSP-5-UPDOWN: A porta de tronco (canal) [1/0:1(1)] está habilitada.`

Este é exemplo de saída do comando `show voice call summary`:

PORT	CODEC	VAD	VTSP	STATE	VPM STATE
3/0:0.1	g729r8	n	S_CONNECT		S_TRUNKED
3/0:1.2	g729r8	n	S_CONNECT		S_TRUNKED
3/0:2.3	g729r8	n	S_CONNECT		S_TRUNKED

Um tronco que não esteja acima aparecerá como `S_TRUNK_PEND`:

PORT	CODEC	VAD	VTSP	STATE	VPM STATE
3/0:0.1	-	-	-		<b>S_TRUNK_PEND</b> 3/0:1.2 g729r8 n S_CONNECT
S_TRUNKED	3/0:2.3	g729r8	n	S_CONNECT	S_TRUNKED

# Troubleshooting

Esta seção fornece informações que você pode utilizar para fazer troubleshooting de configuração.

## Comandos para Troubleshooting

A [Output Interpreter Tool \(somente clientes registrados\)](#) oferece suporte a determinados comandos show, o que permite exibir uma análise da saída do comando show.

**Nota:** Antes de emitir **comandos debug**, consulte [Informações importantes sobre comandos debug](#).

- **mostre a Voz do histórico da chamada | Inclua DisconnectText** — Mostra o motivo de desconexão para as últimas chamadas falha.
- **mostre o sumário da chamada de voz** — Mostra o active do atendimento em ambos os trechos de chamada.
- **DSP de voz da mostra** — Mostra que os processadores do sinal digital (DSP) estão no uso e estão processando pacotes.

Para obter mais informações sobre das chamadas VoIP do Troubleshooting, refira [Conceitos Básicos de Troubleshooting e Debugging de Chamada VoIP](#) e [comandos Debug de VoIP](#).

As portas de voz associadas em ambo o Roteadores devem ser **parada programada shutdown/no** depois que você configura o tronco de conexão. Isto igualmente cancela as portas de voz se você vê o usuário ocupado como uma causa da desconexão.

Este é exemplo de saída de comando do **comando show voice dsp**:

BOOT					PAK									
TYPE	DSP	CH	CODEC	VERS	STATE	STATE	RST	AI	PORT	TS	ABORT			TX/RX-PAK-CNT
====	===	==	=====	====	=====	=====	===	==	=====	==	=====			=====
C549	000	01	g729r8	3.4	busy	idle	0	0	3/0:12	13	0			3522765/3578769
		00	g729r8	.41	busy	idle	0	0	3/0:0	1	0			3505023/3560759
C549	001	01	g729r8	3.4	busy	idle	0	0	3/0:13	14	0			3522761/3578601
		00	g729r8	.41	busy	idle	0	0	3/0:1	2	0			3522794/3578579

O exemplo de saída seguinte é o resultado do debug o mais comum para o **comando debug voip ccapl inout**. Isto debuga foi tomado sob o erro comum de um par faltante dos POTENCIÔMETROS no lado chamado. No exemplo, o roteador do lado analógico não tem um par dos POTENCIÔMETROS para terminar o tronco; o lado da chamada digital terá estes debuga nesta situação:

```
maui-slt-01#
```

```
*Mar 1 00:11:19.903: cc_api_call_setup_ind (vdbPtr=0x620B2DE8,
callInfo={called=2000,called_oct3=0x81,calling=,calling_oct3=0x0,
calling_oct3a=0x0,calling_xlated=false,subscriber_type_str=RegularLine
,fdest=1,peer_tag=2, prog_ind=3},callID=0x621C45F0)
*Mar 1 00:11:19.903: cc_api_call_setup_ind type 3 , prot 0
*Mar 1 00:11:19.903: cc_process_call_setup_ind (event=0x62332908)
*Mar 1 00:11:19.903: >>>>CCAPI handed cid 3 with tag 2 to app "DEFAULT"
*Mar 1 00:11:19.907: sess_appl: ev(24=CC_EV_CALL_SETUP_IND), cid(3), disp(0)
*Mar 1 00:11:19.907: sess_appl: ev(SSA_EV_CALL_SETUP_IND), cid(3), disp(0)
*Mar 1 00:11:19.907: ssaCallSetupInd
*Mar 1 00:11:19.907: ccCallSetContext (callID=0x3, context=0x621C4E90)
```

\*Mar 1 00:11:19.907: ssaCallSetupInd cid(3), st(SSA\_CS\_MAPPING),oldst(0),  
ev(24)ev->e.evCallSetupInd.nCallInfo.finalDestFlag = 1  
\*Mar 1 00:11:19.907: ssaCallSetupInd finalDest cllng(1000), cllcd(2000)  
\*Mar 1 00:11:19.907: ssaCallSetupInd cid(3), st(SSA\_CS\_CALL\_SETTING),  
oldst(0), ev(24)dpMatchPeersMoreArg result= 0  
\*Mar 1 00:11:19.907: ssaSetupPeer cid(3) peer list:  
tag(1) called number (2000)  
\*Mar 1 00:11:19.907: ssaSetupPeer cid(3), destPat(2000), matched(1),  
prefix(), peer(61EE565C), peer->encapType (2)  
\*Mar 1 00:11:19.907: ccCallProceeding (callID=0x3, prog\_ind=0x0)  
\*Mar 1 00:11:19.907: ccCallSetupRequest (Inbound call = 0x3, outbound  
peer =1, dest=, params=0x6233BD30 mode=0, \*callID=0x6233C098, prog\_ind = 3)  
\*Mar 1 00:11:19.907: ccCallSetupRequest numbering\_type 0x81  
\*Mar 1 00:11:19.907: ccCallSetupRequest encapType 2 clid\_restrict\_disable 1  
null\_orig\_clg 1 clid\_transparent 0 callingNumber 1000  
\*Mar 1 00:11:19.907: dest pattern 2..., called 2000, digit\_strip 0  
\*Mar 1 00:11:19.907: callingNumber=1000, calledNumber=2000, redirectNumber=  
display\_info= calling\_oct3a=0  
\*Mar 1 00:11:19.907: accountNumber=, finalDestFlag=1,  
guid=1d0d.9a0f.14f0.11cc.8008.b3df.433e.6402  
\*Mar 1 00:11:19.911: peer\_tag=1  
\*Mar 1 00:11:19.911: ccIFCallSetupRequestPrivate: (vdbPtr=0x621D74DC, dest=,  
callParams={called=2000,called\_oct3=0x81, calling=1000,calling\_oct3=0x0,  
calling\_xlated=false, subscriber\_type\_str=RegularLine, fdest=1,  
voice\_peer\_tag=1}, mode=0x0) vdbPtr type = 1  
\*Mar 1 00:11:19.911: ccIFCallSetupRequestPrivate: (vdbPtr=0x621D74DC, dest=,  
callParams={called=2000, called\_oct3 0x81, calling=1000,calling\_oct3 0x0,  
calling\_xlated=false, fdest=1, voice\_peer\_tag=1}, mode=0x0, xltrc=-5)  
\*Mar 1 00:11:19.911: ccSaveDialpeerTag (callID=0x3, dialpeer\_tag=0x1)  
\*Mar 1 00:11:19.911: ccCallSetContext (callID=0x4, context=0x624C3094)  
\*Mar 1 00:11:19.911: ccCallReportDigits (callID=0x3, enable=0x0)  
\*Mar 1 00:11:19.911: cc\_api\_call\_report\_digits\_done (vdbPtr=0x620B2DE8,  
callID=0x3, disp=0)  
\*Mar 1 00:11:19.911: sess\_appl: ev(52=CC\_EV\_CALL\_REPORT\_DIGITS\_DONE),  
cid(3), disp(0)  
\*Mar 1 00:11:19.911: cid(3)st(SSA\_CS\_CALL\_SETTING)ev  
(SSA\_EV\_CALL\_REPORT\_DIGITS\_DONE)oldst(SSA\_CS\_MAPPING)  
cfid(-1)csize(0)in(1)fDest(1)  
\*Mar 1 00:11:19.911: -cid2(4)st2(SSA\_CS\_CALL\_SETTING)oldst2(SSA\_CS\_MAPPING)  
\*Mar 1 00:11:19.911: ssaReportDigitsDone cid(3) peer list: (empty)  
\*Mar 1 00:11:19.911: ssaReportDigitsDone callid=3 Reporting disabled.  
\*Mar 1 00:11:19.947: cc\_api\_call\_disconnected(vdbPtr=0x621D74DC,  
callID=0x4, cause=0x1)  
\*Mar 1 00:11:19.947: sess\_appl: ev(11=CC\_EV\_CALL\_DISCONNECTED), cid(4), disp(0)  
\*Mar 1 00:11:19.947: cid(4)st(SSA\_CS\_CALL\_SETTING)ev(SSA\_EV\_CALL\_DISCONNECTED)  
oldst(SSA\_CS\_MAPPING)cfid(-1)csize(0)in(0)fDest(0)  
\*Mar 1 00:11:19.947: -cid2(3)st2(SSA\_CS\_CALL\_SETTING)oldst2(SSA\_CS\_CALL\_SETTING)  
\*Mar 1 00:11:19.951: ssaDiscSetting  
\*Mar 1 00:11:19.951: ssa: Disconnected cid(4) state(1) cause(0x1)  
\*Mar 1 00:11:19.951: ccCallDisconnect (callID=0x4, cause=0x1 tag=0x0)  
\*Mar 1 00:11:19.951: ccCallDisconnect (callID=0x3, cause=0x1 tag=0x0)  
\*Mar 1 00:11:19.951: cc\_api\_call\_disconnect\_done(vdbPtr=0x620B2DE8, callID=0x3,  
disp=0, tag=0x0)  
\*Mar 1 00:11:19.955: sess\_appl: ev(12=CC\_EV\_CALL\_DISCONNECT\_DONE), cid(3),  
disp(0)  
\*Mar 1 00:11:19.955: cid(3)st(SSA\_CS\_DISCONNECTING)ev  
(SSA\_EV\_CALL\_DISCONNECT\_DONE)oldst(SSA\_CS\_CALL\_SETTING)  
cfid(-1)csize(0)in(1)fDest(1)  
\*Mar 1 00:11:19.955: -cid2(4)st2(SSA\_CS\_DISCONNECTING)oldst2(SSA\_CS\_CALL\_SETTING)  
\*Mar 1 00:11:19.955: ssaDisconnectDone  
\*Mar 1 00:11:19.963: cc\_api\_icpif: expect factor = 0  
\*Mar 1 00:11:19.963: cc\_api\_call\_disconnect\_done(vdbPtr=0x621D74DC,  
callID=0x4, disp=0, tag=0x0)  
\*Mar 1 00:11:19.967: sess\_appl: ev(12=CC\_EV\_CALL\_DISCONNECT\_DONE),



```
cid(4), disp(0)
*Mar 1 00:11:19.967: cid(4)st(SSA_CS_DISCONNECTING)ev
(SSA_EV_CALL_DISCONNECT_DONE)oldst(SSA_CS_CALL_SETTING)
cfid(-1)csize(1)in(0)fDest(0)
*Mar 1 00:11:19.967: ssaDisconnectDone
```

## [Informações Relacionadas](#)

- [Configurando a conexão PLAR para gateways de VoIP](#)
- [Conceitos Básicos de Troubleshooting e Depuração de Chamadas VoIP](#)
- [Comandos de debug VoIP](#)
- [Suporte à Tecnologia de Voz](#)
- [Suporte de Produtos de Comunicação de Voz e de IP](#)
- [Troubleshooting da Telefonia IP Cisco](#)
- [Suporte Técnico - Cisco Systems](#)