

Edições do fax relay de H.323 T.38

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenções](#)

[Princípios de T.38](#)

[Limitações de T.38](#)

[Negociação de T.38](#)

[Pesquisa de defeitos de T.38](#)

[Dicas de Troubleshooting para o fax relay de H.323 ou de T.38 do SORVO](#)

[comandos debug e show](#)

[Saída de um atendimento bem sucedido de T.38](#)

[Um exemplo de um atendimento falhado de T.38](#)

[Informações Relacionadas](#)

[Introdução](#)

Os problemas de retransmissão de fax T.38 são associados geralmente a problemas de interoperabilidade entre gateways de fax Cisco T.38 de terceiros. Este documento contém exemplos detalhados do comando de depuração de erros de chamadas de retransmissão de fax T.38 bem e malsucedidas. Estes comandos de depuração de erros contêm comentários para fornecer pontos de referência para que seja possível resolver problemas tais como questões de interoperabilidade. Os comandos pertinentes de troubleshooting e verificação também são fornecidos neste documento.

[Pré-requisitos](#)

[Requisitos](#)

Os leitores deste documento devem ser conhecedores dos conceitos básicos do fax relay. Refira o [guia de Troubleshooting do fax relay](#) para obter mais informações sobre dos conceitos e das etapas de Troubleshooting básicas do fax relay.

[Componentes Utilizados](#)

Este documento não se restringe a versões de software e hardware específicas.

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](#)

Consulte as [Convenções de Dicas Técnicas da Cisco](#) para obter mais informações sobre convenções de documentos.

[Princípios de T.38](#)

Um sintoma comum de problemas do fax relay de T.38 é uma chamada de voz que seja estabelecida onde um tom do fax é ouvido, mas a negociação do fax não é terminada e o atendimento é deixado cair eventualmente. Esta edição é associada frequentemente com o gateway de Cisco T.38 e os problemas de interoperabilidade de gateway da terceira de T.38.

O fax relay de T.38 é transmissão de fax do tempo real; isto é, duas máquinas de fax que se comunicam um com o outro como se havia uma linha telefônica direta entre os dois. O fax relay é configurado com alguns comandos adicionais nos gateway dial peer que já têm sido definidos e configurados para chamadas de voz.

Cisco fornece dois métodos para o fax relay: um método proprietário Cisco e um método baseado no padrão de T.38 do ITU-T. Na maioria de Plataformas, o fax relay de Cisco é o padrão se um método do fax não é configurado explicitamente. O fax relay de Cisco é descrito em [configurar o fax relay de Cisco](#).

[Limitações de T.38](#)

Neste momento, o fax relay de Cisco T.38 tem estas limitações:

- A Interoperabilidade de T.38 exige a versão 2 de Cisco H.323.
- T.38 não é apoiado em concentradores da Cisco MC3810 Series com uma voice compression module (VCM).
- T.38 não é apoiado pelo proxy de H.323 do Multimedia Conference Manager (MCM).
- Somente o User Datagram Protocol (UDP) é executado para H.323 T.38.
- Alguns gateways e roteiros da terceira não são compatíveis com os ciscos voices gateways para o fax relay de T.38 porque os fabricantes diferentes podem escolher determinadas partes de H.323 e de T.38 executar em seus gateways e roteiros. Os testes da Interoperabilidade da Voz com estes gateways e roteiros da terceira devem ser executados para certificar-se de que o fax relay de T.38 pode ser bem sucedido.

[Negociação de T.38](#)

Esta seção fornece um breve sumário passo a passo de como a negociação de T.38 é segurada dentro dos Cisco gateway. Refira o [guia de Troubleshooting do fax relay](#) para obter mais informações sobre dos princípios do fax relay.

1. Na mensagem de instalação inicial, a capacidade dos dados de T.38 é anunciada pelo gateway de origem (OGW). Se o gateway de terminação (TGW) apoia a capacidade dos dados de T.38, pode retransmitir que informação nos mensagens subseqüente enviados ao

OGW.

2. Uma vez que uma chamada de voz está estabelecida e o processador do sinal digital (DSP) no TGW detecta um tom do fax, a máquina de estado do provedor de serviços de telefonia por voz (VTSP) informa o trecho de chamada de H.323, que negocia o modo de T.38 com o OGW.
3. Em cima do reconhecimento do modo de T.38, o canal de áudio é fechado, e o canal lógico de T.38 é aberto no ambas as extremidades.
4. A nível do código VTSP, a transferência do codificador-decodificador do fax (codec) ocorre.
5. Em cima de uma transferência aberta bem sucedida do canal lógico (OLC) e do codec de T.38, o VTSP participa no fax - modo.
6. Após a conclusão da transmissão de fax, o atendimento é revertido de volta a uma chamada de voz. **Nota:** Durante a negociação do modo de T.38, se a outra extremidade não reconhece o modo de T.38, o atendimento é revertido de volta a uma chamada de voz e desligado. Se o reconhecimento negativo é recebido da outra extremidade em relação a T.38 OLC, a seguir o atendimento igualmente está revertido de volta a uma chamada de voz e desligado.

[Pesquisa de defeitos de T.38](#)

[Dicas de Troubleshooting para o fax relay de H.323 ou de T.38 do SORVO](#)

A fim de pesquisar defeitos o fax relay de T.38, execute estas etapas:

- **Certifique-se de que você pode fazer uma chamada de voz.** Confirme que as chamadas de voz normais podem ser terminadas antes que você investigue a Conectividade do fax. Se não houver nenhum telefone conectado, desligue o fax e conecte um telefone regular. Se as chamadas de voz normais não conectam, a edição pode ser VoX-relacionada, e você pode pesquisar defeitos o problema enquanto uma edição normal da conectividade de voz antes que você continue com Troubleshooting de Fax.
- Certifique-se de que o protocolo desejado do fax esteve ajustado com o comando fax protocol em ambos os gateways de origem e finalização.
- Certifique-se de que o protocolo do fax está configurado como T.38 a nível da configuração global ou na configuração de dial peer em nível para ambos os gateways de origem e finalização.

[comandos debug e show](#)

Os comandos **debug and show** usados pesquisando defeitos o fax relay de T.38 são:

- **debug o inout do ccapi do voip** — Este comando segue o trajeto da execução com o Application Program Interface do Controle de chamadas (API), que serve como a relação entre o aplicativo da sessão de chamada e o software rede-específico subjacente. Você pode usar a saída deste comando compreender como os atendimentos estão sendo segurados pelo gateway de voz.
- **debug o vtsp todo** — Este comando permite estes **debug** comandos VTSP: **o debug vtsp session**, **debug o erro do vtsp**, e **debug o dsp do vtsp**.
- **debug o asn1 h245** — Este comando indica os índices do Abstract Syntax Notation One (ASN.1) das mensagens H.245. Para desabilitar o resultado do debug, não use nenhum

formulário deste comando.

- **debug cch323 h245** — Este comando fornece o traço da transição de estado da máquina de estado H.245 baseada nos eventos processados. Para desabilitar o resultado do debug, não use nenhum formulário deste comando.
- [mostre o resumo do fax ativo do atendimento](#) — Este comando indica a informação de chamada para as transmissões de fax de armazenamento e de encaminhamento em andamento.
- [mostre o fax do histórico da chamada](#) — Este comando indica o histórico da chamada recente para fax.

Saída de um atendimento bem sucedido de T.38

Esta seção detalha a anatomia de um fax bem sucedido de T.38 setup entre um roteador do AS5300 Series e um Modular Access Router do Cisco 3640. As saídas do comando **debug and show** foram capturadas no Universal Gateway do Cisco AS5300 como o TGW IO 12.2:

saída do comando debug vtsp all

```
!---After the voice call setup: !--- Usually, after the
call is connected, the ccCallConnect debug !--- message
is seen as follows: May 3 21:41:21.424: ccCallConnect
(callID=0x9), prog_ind = 0 May? 3 21:41:21.424:
ssaFlushPeerTagQueue cid(9) peer list: (empty) May 3
21:41:21.424: H.225 SM: process event
H225_EVENT_SETUP_CFM, for callID 9 May 3 21:41:21.424:
cch323_run_h225_sm: received event H225_EVENT_SETUP_CFM
while at state H225_ALERT May 3 21:41:21.424: H.225 SM:
changing from H225_ALERT state to H225_ACTIVE state for
callID 9 May 3 21:41:21.424: ==== PI in
cch323_h225_generic_send_setup_cfm = 0 !---After the
voice call is established, the TGW DSP detected fax
tone: May 3 21:41:26.741: vtsp_process_dsp_message:
MSG_TX_TONE_DETECT: type=0 trigger=1 tone_id=0 May 3
21:41:26.741: vtsp:[1:D (10), S_CONNECT,
E_DSP_TONE_DETECT] May 3 21:41:26.745:
vtsp_modem_proto_from_cdb: cap_modem_proto 0 May 3
21:41:26.745: cc_api_call_feature: (vdbPtr=0x624130C0,
callID=0xA,feature_ind.type=1 !---Switched to fax mode:
May 3 21:41:26.745: act_lfax_switch: cap_modem_proto=16,
fax_relay_on=1, state=19 May 3 21:41:26.745:
vtsp_t38_switchover:2 - data_mode:1 !--- Note that 2
means T.38; 1 means Cisco proprietary. May 3
21:41:26.745: cc_api_t38_fax_start
(dstVdbPtr=0x61B45A90, dstCallId=0x9, srcCallId=0xA,???)
caps={codec=0x10000, fax_rate=0x2, vad=0x2,
modem=0x0codec_bytes=160, signal_type=1}) May 3
21:41:26.745: vtsp_timer: 2016656 May 3 21:41:26.745:
sess_appl: ev(28=CC_EV_CALL_FEATURE), cid(10), disp(0)
May 3 21:41:26.745:
cid(10)st(SSA_CS_ACTIVE)ev(SSA_EV_CALL_FEATURE)
oldst(SSA_CS_CONFERENCED_ALERT)cfid(5)csiz(0)in(0)fDest
(0) May 3 21:41:26.745: -
cid2(9)st2(SSA_CS_ACTIVE)oldst2(SSA_CS_CONFERENCING_ALER
T) !---H245 ModeRequest was sent to the OGW: May 3
21:41:26.745: ccCallFeature (callID=0x9, feature.type=1)
Set new event H245_EVENT_MR, for callID 9 May 3
21:41:26.745: cch323_run_h245_mr_sm: received event
H245_EVENT_MR while at state H245_MR_NONE? !---Above,
```

```
state H245_MR_NONE refers to ModeRequest state. May 3
21:41:26.745: H245 MSC OUTGOING PDU ::= value
MultimediaSystemControlMessage ::= request : requestMode
: ??? { ????? sequenceNumber 1 ????? requestedModes
?????{ ??????? { ?????????? { ????????????? type dataMode :
????????????? { ????????????????? application t38fax :
????????????????? { ????????????????????? t38FaxProtocol udp :
NULL ?????????????????? t38FaxProfile ?????????????? ???{
????????????????????? fillBitRemoval FALSE ?????????????????????
transcodingJBIG FALSE ????????????????????? transcodingMMR
FALSE ????????????????????? version 0 ?????????????????????
t38FaxRateManagement transferredTCF : NULL
????????????????????? t38FaxUdpOptions ??????? ??????????????{
????????????????????????? t38FaxMaxBuffer 200
????????????????????????? t38FaxMaxDatagram 72
????????????????????????? t38FaxUdpEC t38UDPRedundancy : NULL
????????????????????????? } ????????????????????? } ?????????????????? }
????????????????????? bitrate 144 ?????????????? } ?????????? }
????????? } ?????? } ??? } May 3 21:41:26.753: changing
from H245_MR_NONE state to H245_MR_WAIT_FOR_ACK state
May 3 21:41:26.861: vtsp_process_dsp_message:
MSG_TX_TONE_DETECT: type=0 trigger=0 tone_id=0 May 3
21:41:26.861: vtsp:[1:D (10), S_LFAX_WAIT_CAPS_ACK,
E_DSP_TONE_DETECT] May 3 21:41:26.865:
vtsp_process_event(): prev_state = 0.11 , state =
S_LFAX_WAIT_CAPS_ACK, event = E_DSP_TONE_DETECT ?Invalid
FSM? Input on channel 1:D
(10)h323chan_chn_process_read_socket: fd (3) of type
ACCEPTED has data PROCESS_READ: NOT COMPLETE, rc 10,
fd=3 May? 3 21:41:27.001: vtsp_process_dsp_message:
MSG_TX_TONE_DETECT: type=0 trigger=1 tone_id=0 May? 3
21:41:27.001: vtsp:[1:D (10), S_LFAX_WAIT_CAPS_ACK,
E_DSP_TONE_DETECT] May? 3 21:41:27.005:
vtsp_process_event(): prev_state = 0.11 , ?state =
S_LFAX_WAIT_CAPS_ACK, event = E_DSP_TONE_DETECT Invalid
FSM?Input on channel 1:D (10) May 3 21:41:27.101:
vtsp_process_dsp_message: MSG_TX_TONE_DETECT: type=0
trigger=0 tone_id=0 May 3 21:41:27.101: vtsp:[1:D (10),
S_LFAX_WAIT_CAPS_ACK, E_DSP_TONE_DETECT] May 3
21:41:27.105: vtsp_process_event(): prev_state = 0.11 ,
state = S_LFAX_WAIT_CAPS_ACK, event = E_DSP_TONE_DETECT
Invalid FSM Input on channel 1:D
(10)h323chan_chn_process_read_socket: fd (3) of type
ACCEPTED has data Hex representation of the received
TPKT0321000827000100 May 3 21:41:27.173: ? state = 0
bytesLeftToDecode = 4 May 3 21:41:27.173: H245 MSC
INCOMING ENCODE BUFFER ::= 27 000100 !---Received
ModeRequestAck from the OGW: May 3 21:41:27.173: H245
MSC INCOMING PDU ::= value
MultimediaSystemControlMessage ::= response :
requestModeAck : ??? { ????? sequenceNumber 1 ?????
response willTransmitMostPreferredMode : NULL ??? } Set
new event H245_EVENT_MR_CFM, for callID 9 May 3
21:41:27.173: cch323_run_h245_mr_sm: received event
H245_EVENT_MR_CFM while at state H245_MR_WAIT_FOR_ACK !-
--The voice LC is closed and the T.38 fax data LC is
opened: May 3 21:41:27.173: H245 MSC OUTGOING PDU ::=
value MultimediaSystemControlMessage ::= request :
closeLogicalChannel :? !---In the previous line,
LogicalChannel refers to the voice LC. ??? { ?????
forwardLogicalChannelNumber 1 ????? source user : NULL
??? } May 3 21:41:27.173: H245 MSC OUTGOING ENCODE
BUFFER ::= 04 00000000 May 3 21:41:27.173: send result :0
May 3 21:41:27.173: changing from H245_OLC_DONE state to
```

```

H245_OLC_NONE state May 3 21:41:27.173:
cch323_update_new_codec_info: Remote codec 17 May 3
21:41:27.173: cch323_update_new_codec_info:
negotiated_codec set(17)(40 bytes) May 3 21:41:27.173:
Changing to new event H245_EVENT_OLC May 3 21:41:27.177:
cch323_h245_olc_sm: received event H245_EVENT_OLC while
at state H245_OLC_NONE May 3 21:41:27.177: changing from
H245_OLC_NONE state to H245_OLC_WAIT state May 3
21:41:27.177: H245 MSC OUTGOING PDU ::= value
MultimediaSystemControlMessage ::= request :
openLogicalChannel :? !---In the previous line,
LogicalChannel refers to the T.38 or data LC. ??? {
????? forwardLogicalChannelNumber 2 ??????
forwardLogicalChannelParameters ?????? { ???????? dataType
data : ???????? { ?????????? application t38fax :
???????????? { ?????????????? t38FaxProtocol udp : NULL
???????????????? t38FaxProfile ?????????????? { ??????????????????
fillBitRemoval FALSE ?????????????????? transcodingJBIG FALSE
???????????????????? transcodingMMR FALSE ?????????????????? version
0 ?????????????????? t38FaxRateManagement transferredTCF :
NULL ?????????????????? t38FaxUdpOptions ?????????????? ??{
???????????????????? t38FaxMaxBuffer 200 ?????????????????????
t38FaxMaxDatagram 72 ????????????????????? t38FaxUdpEC
t38UDPRedundancy : NULL ?????????????????? } ?????????????? }
????????????? } ?????????????? maxBitRate 144 ?????????? } ??????????
multiplexParameters h2250LogicalChannelParameters :
????????? { ?????????????? sessionID 3? !---The previous line
refers to the data session ID. ??????????
mediaControlChannel unicastAddress : ipAddress :
????????????? { ?????????????????? network 'AB44BA66'H ??????????????????
tsapIdentifier 17517 ?????????????? } ??????????????
silenceSuppression FALSE ?????????? } ?????? } ??? } May 3
21:41:27.181: H245 MSC OUTGOING ENCODE BUFFER::= 03
00000111 04118601 00805C01 00014007 C00200C8 01484000
90800B05 000300AB 44BA6644 6D00 May 3 21:41:27.181: send
result :0 May 3 21:41:27.181: OLC using T38Fax May 3
21:41:27.181: changing from H245_MR_WAIT_FOR_ACK state
to H245_MR_NONE state h323chan_chn_process_read_socket:
fd (3) of type ACCEPTED has data Hex representation of
the received TPKT032100090400000000 May 3 21:41:27.185:
? state = 0 bytesLeftToDecode = 5 May 3 21:41:27.185:
H245 MSC INCOMING ENCODE BUFFER::= 04 00000000 May 3
21:41:27.185: May 3 21:41:27.185: H245 MSC INCOMING PDU
::= value MultimediaSystemControlMessage ::= request :
closeLogicalChannel :?? !---In the previous line,
LogicalChannel refers to the voice LC. ??? { ??????
forwardLogicalChannelNumber 1 ?????? source user : NULL
??? } May? 3 21:41:27.185: H245 MSC OUTGOING PDU ::=
value MultimediaSystemControlMessage ::= response :
closeLogicalChannelAck :??? !---In the previous line,
LogicalChannel refers to the voice LC. ??? { ??????
forwardLogicalChannelNumber 1 ??? } May 3 21:41:27.185:
H245 MSC OUTGOING ENCODE BUFFER::= 23 800000 May 3
21:41:27.185: H245 MSC INCOMING ENCODE BUFFER::= 03
00000111 04118601 00805C01 00014007 C00200C8 01484000
90800B05 000300AC 10AF6941 7100 May 3 21:41:27.189: H245
MSC INCOMING PDU ::= value
MultimediaSystemControlMessage ::= request :
openLogicalChannel :? !---In the previous line,
LogicalChannel refers to the T.38 or data LC. ??? {
?????? forwardLogicalChannelNumber 2 ??????
forwardLogicalChannelParameters ?????? { ?????????? dataType
data : ?????????? { ?????????????? application t38fax :
???????????????? { ?????????????????? t38FaxProtocol udp : NULL

```

```
???????????? t38FaxProfile ????????????? { ?????????????
fillBitRemoval FALSE ????????????? transcodingJBIG FALSE
???????????? transcodingMMR FALSE ????????????? version
0 ????????????? t38FaxRateManagement transferredTCF :
NULL ????????????? t38FaxUdpOptions ????????????? {
???????????? t38FaxMaxBuffer 200 ?????????????
t38FaxMaxDatagram 72 ????????????? t38FaxUdpEC
t38UDPRedundancy : NULL ????????????? } ????????????? }
???????????? } ????????????? maxBitRate 144 ????????????? } ?????????
multiplexParameters h2250LogicalChannelParameters :
???????? { ????????? sessionID 3 ?????????
mediaControlChannel unicastAddress : ipAddress :
???????? { ????????? network 'AC10AF69'H ?????????
tsapIdentifier 16753 ????????? } ?????????
silenceSuppression FALSE ???? ??? } ????? } !---DSP
started T.38 fax codec download: May 3 21:41:27.193:
cc_api_t38_fax_start (dstVdbPtr=0x624130C0,
dstCallId=0xA, srcCallId=0x9, ??? caps={codec=0x10000,
fax_rate=0x2, vad=0x2, modem=0x codec_bytes=160,
signal_type=1}) May 3 21:41:27.193: vtsp:[1:D (10),
S_LFAX_WAIT_CAPS_ACK, E_CC_T38_START] May 3
21:41:27.193: act_caps_ack_lfax_dnld May 3 21:41:27.193:
vtsp_timer_stop: 2016700 May 3 21:41:27.193:
dsp_idle_mode: [1:D (10)] packet_len=8 channel_id=8481
packet_id=68 May 3 21:41:27.193:
cc_api_local_codec_dnld_done (dstVdbPtr=0x61B45A90,
dstCallId=0x9, srcCallId=0xA) May 3 21:41:27.193:
vtsp_timer: 2016700cch323_h245_local_codec_dnld_done:
negotiatedCodec[17] May 3 21:41:27.197: Changing to new
event H245_EVENT_OLC_IND May 3 21:41:27.197:
cch323_h245_olc_sm: received event H245_EVENT_OLC_IND
while at state H245_OLC_WAIT May 3 21:41:27.197: H245
MSC OUTGOING PDU ::= value
MultimediaSystemControlMessage ::= response :
openLogicalChannelAck : ??? { ?????
forwardLogicalChannelNumber 2 ?????
forwardMultiplexAckParameters
h2250LogicalChannelAckParameters : ????? { ?????
sessionID 1 ????? mediaChannel unicastAddress :
ipAddress : ????? { ????? network 'AB44BA66'H
???? tsapIdentifier 17516 ????? } ?????
??mediaControlChannel unicastAddress : ipAddress :
???? { ????? network 'AB44BA66'H ?????
tsapIdentifier 17517 ????? } ????? flowControlToZero
FALSE ????? } ??? } May 3 21:41:27.197: H245 MSC
OUTGOING ENCODE BUFFER: := 22 C0000104 80145C00 00AB44BA
66446C00 AB44BA66 446D0300 0100 May 3 21:41:27.589: ?
state = 0 bytesLeftToDecode = 4 May 3 21:41:27.589: H245
MSC INCOMING ENCODE BUFFER:= 23 800000 May 3
21:41:27.589: May 3 21:41:27.589: H245 MSC INCOMING PDU
::= value MultimediaSystemControlMessage ::= response :
closeLogicalChannelAck : ??? { ?????
forwardLogicalChannelNumber 1 ??? } May 3 21:41:27.789:
H245 MSC INCOMING ENCODE BUFFER: := 22 C0000104 80145C00
00AC10AF 69417000 AC10AF69 41710300 0100 May 3
21:41:27.789: H245 MSC INCOMING PDU ::= value
MultimediaSystemControlMessage ::= response :
openLogicalChannelAck : ??? { ?????
forwardLogicalChannelNumber 2 ?????
forwardMultiplexAckParameters
h2250LogicalChannelAckParameters : ????? { ?????
sessionID 3 ????? mediaChannel unicastAddress :
ipAddress : ????? { ????? network 'AC10AF69'H
???? tsapIdentifier 16752 ????? } ?????
```

```

mediaControlChannel unicastAddress : ipAddress : ???????
{ ????????? network 'AC10AF69'H ????????? tsapIdentifier
16753 ??????? } ??????? flowControlToZero FALSE ?????? }
??? } May 3 21:41:27.793: Changing to new event
H245_EVENT_OLC_CFM May 3 21:41:27.793:
cch323_h245_olc_sm: received event H245_EVENT_OLC_CFM
while at state H245_OLC_WAIT May 3 21:41:27.793:
changing from H245_OLC_WAIT state to H245_OLC_DONE state
May 3 21:41:27.793: cc_api_t38_fax_start
(dstVdbPtr=0x624130C0, dstCallId=0xA, srcCallId=0x9,
??? caps={codec=0x10000, fax_rate=0x2, vad=0x2,
modem=0x0 codec_bytes=160, signal_type=1}) May 3
21:41:27.793: H.225 SM: process event
H225_EVENT_H245_SUCCESS, for callID 9 May 3
21:41:27.793: cch323_run_h225_sm: received event
H225_EVENT_H245_SUCCESS while at state H225_ACTIVE May 3
21:41:27.793: cc_api_remote_codec_dnld_done
(dstVdbPtr=0x624130C0, dstCallId=0xA, srcCallId=0x9) May
3 21:41:27.793: vtsp:[1:D (10), S_LFAX_WAIT_FAX,
E_CC_T38_START] May 3 21:41:27.793: vtsp:[1:D (10),
S_LFAX_WAIT_FAX, E_CC_T30_CAP_ACK] May 3 21:41:27.793:
act_t38_lfax_mode May 3 21:41:27.793: vtsp_timer_stop:
2016760 May 3 21:41:27.793: cc_api_set_fax_mode
(dstVdbPtr=0x61B45A90, dstCallId=0x9, srcCallId=0xA) May
3 21:41:27.793: dsp_idle_mode: [1:D (10)] packet_len=8
channel_id=8481 packet_id=68 May 3 21:41:27.793:
dsp_encap_config: T38 May 3 21:41:27.793: dsp_fax_mode:
[1:D (10)] FaxRate 0x2, Codec 0x10000? dsp_fax_mode()
ECM_DISABLE not set, debug_info not requested May 3
21:41:27.793: dsp_fax_mode:[1:D (10)] packet_len=28
channel_id=8481 packet_id=69 max_trans=6 info_size=20,
fax_protocol_type=3,hs_data_len=40, ls_data_red=0,
hs_data_red=0, tcf_handling=2, fax_relay_cntl=0x0
nsf_country = 0xAD, nsf_mfg = 0x0051 May 3 21:41:29.621:
ccGetCallActive (next=1, setup_time=0x0, index=0x0,
p=0x6293A8C0) May 3 21:41:29.621: ccGetCallActive
(next=1, setup_time=0x1EC241, index=0x1, p=0x6293A8C0)

```

[Um exemplo de um atendimento falhado de T.38](#)

Este é um exemplo do comando debug para um atendimento falhado de T.38:

saída do comando debug vtsp all

```

!---When the ModeRequest was sent, T35 nonStandard was
sent instead of T38: *Jun 14 15:35:01.743: PDU DATA =
61593960 value MultimediaSystemControlMessage ::=
request : requestMode : ??? { ????? sequenceNumber 12
????? requestedModes ?????? { ??????? { ?????????? {
???????????? type dataMode : ?????????????? { ??????????????
application nonStandard : ?????????????? {
???????????????? nonStandardIdentifier h221NonStandard :
????????????????? { ?????????????????? t35CountryCode 181 ?
?????????????????t35Extension 0 ??????????????????
manufacturerCode 20 ?????????????????? } ??????????????????
data '543338466178554450'H ?????????????????? } ??????????????????
bitRate 144 ?????????????? } ?????????? } ??????? } ?????? }
??? } Set new event H245_EVENT_MR_IND, for callID C *Jun
14 15:35:01.751: cch323_run_h245_mr_sm: received event
H245_EVENT_MR_IND wh ile at state H245_MR_NONE *Jun 14
15:35:01.751: Scan Preferred List for g729r8PDU DATA =

```



```
61593960 value MultimediaSystemControlMessage ::=
response : requestModeAck : ??? { ????? sequenceNumber
12 ????? response willTransmitMostPreferredMode : NULL
??? } RAW_BUFFER ::= 27 000C00 *Jun 14 15:35:01.751: PDU
DATA = 61593960 value MultimediaSystemControlMessage ::=
request : closeLogicalChannel : ??? { ??
????forwardLogicalChannelNumber 2 ????? source user :
NULL ??? } RAW_BUFFER ::= 04 00000100 *Jun 14
15:35:01.751: *Jun 14 15:35:01.751: changing from
H245_OLC_DONE state to H245_OLC_NONE state *Jun 14
15:35:01.751: cch323_update_new_codec_info: Remote codec
17 *Jun 14 15:35:01.751: cch323_update_new_codec_info:
negotiated_codec set(17)(40 bytes) *Jun 14 15:35:01.751:
Changing to new event H245_EVENT_OLC *Jun 14
15:35:01.751: cch323_h245_olc_sm: received event
H245_EVENT_OLC while atstate H245_OLC_NONE *Jun 14
15:35:01.751: changing from H245_OLC_NONE state to
H245_OLC_WAIT state PDU DATA = 61593960 value
MultimediaSystemControlMessage ::= request :
openLogicalChannel : ??? { ?????
forwardLogicalChannelNumber 3 ?????
forwardLogicalChannelParameters ????? { ?????? dataType
data : ?????? { ?????? application nonStandard :
???????? { ????????? nonStandardIdentifier
h221nonStandard : ????????? { ?????????
t35CountryCode 181 ????????? t35Extension 0
???????? manufacturerCode 18 ? ?????????}
???????? data '543338466178554450'H ????????? }
???????? maxBitRate 144 ?????? } ??????
multiplexParameters h2250LogicalChannelParameters :
???????? { ????????? sessionID 3 ?????????
mediaControlChannel unicastAddress : ipAddress : ??????
??? { ????????? network 'C95C381E'H ?????????
tsapIdentifier 18101 ????????? } ?????? } ?????? } ??? }
RAW_BUFFER ::= 03 00000210 08B50000 12095433 38466178
55445000 90800A04 000300C9 5C381E46 B5 *Jun 14
15:35:01.759: *Jun 14 15:35:01.759: OLC using T38Fax
*Jun 14 15:35:01.783: Changing to new event
H245_PROCESS_H245CONTROL *Jun 14 15:35:01.783:
cch323_h245_connection_sm:H245_CONNECT: received event
H24 5_PROCESS_H245CONTROL while at H245_CONNECTED state
RAW_BUFFER ::= 04 80000100 800100 *Jun 14 15:35:01.783:
PDU DATA = 61593960 value MultimediaSystemControlMessage
::= request : closeLogicalChannel : ??? { ?????
forwardLogicalChannelNumber 2 ????? source user : NULL
???? reason unknown : NULL ??? } PDU DATA = 61593960
value MultimediaSystemControlMessage ::= response :
closeLogicalChannelAck : ??? { ?????
forwardLogicalChannelNumber 2 ??? } RAW_BUFFER ::= 23
800001 *Jun 14 15:35:01.787: *Jun 14 15:35:01.787:
Changing to new event H245_PROCESS_H245CONTROL *Jun 14
15:35:01.787: cch323_h245_connection_sm:H245_CONNECT:
received event H24 5_PROCESS_H245CONTROL while at
H245_CONNECTED state RAW_BUFFER ::= 03 00000310 08B50000
14095433 38466178 55445000 90800300 0003 *Jun 14
15:35:01.787: PDU DATA = 61593960 value
MultimediaSystemControlMessage ::= request :
openLogicalChannel : ??? { ?????
forwardLogicalChannelNumber 4 ?????
forwardLogicalChannelParameters ????? { ?????? dataType
data : ?????? { ?????? application nonStandard :
???????? { ????????? nonStandardIdentifier
h221NonStandard : ?? ????????? { ?????????
t35CountryCode 181 ????????? t35Extension 0
```

```

???????????? manufacturerCode 20 ?????????? }
???????????? data '543338466178554450'H ?????????? }
???????????? maxBitRate 144 ?????????? } ??????????
multiplexParameters h2250LogicalChannelParameters :
???????? { ?????????? sessionID 3 ?????????? } ?????? } ??? }
*Jun 14 15:35:01.831: Changing to new event
H245_PROCESS_H245CONTROL *Jun 14 15:35:01.831:
cch323_h245_connection_sm:H245_CONNECT: received event
H24 5_PROCESS_H245CONTROL while at H245_CONNECTED state
RAW_BUFFER::= 23 800001 *Jun 14 15:35:01.831: PDU DATA =
61593960 value MultimediaSystemControlMessage ::=
response : closeLogicalChannelAck : ??? { ??????
forwardLogicalChannelNumber 2 ??? } *Jun 14
15:35:01.883: Changing to new event
H245_PROCESS_H245CONTROL *Jun 14 15:35:01.883:
cch323_h245_connection_sm:H245_CONNECT: received event
H24 5_PROCESS_H245CONTROL while at H245_CONNECTED state
RAW_BUFFER::= 22 C0000204 800C5804 00875C34 CB1B4801
0100 *Jun 14 15:35:01.883: PDU DATA = 61593960 value
MultimediaSystemControlMessage ::= response :
openLogicalChannelAck : ??? { ??????
forwardLogicalChannelNumber 3 ??????
forwardMultiplexAckParameters
h2250LogicalChannelAckParameters : ?????? { ??????????
sessionID 3 ?????????? mediaChannel unicastAddress :
IPAddress : ?????????? { ?????????? network '875C34CB'H
???????????? tsapIdentifier 6984 ?????????? } ??????????
flowControlToZero FALSE ?????? } ??? } *Jun 14
15:35:01.887: Changing to new event H245_EVENT_OLC_CFM
*Jun 14 15:35:01.887: cch323_h245_olc_sm: received event
H245_EVENT_OLC_CFM while at state H245_OLC_WAIT *Jun 14
15:35:01.887: changing from H245_OLC_WAIT state to
H245_OLC_DONE state cch323_h245_local_codec_dnld_done:
negotiatedCodec[17] *Jun 14 15:35:01.979: Changing to
new event H245_EVENT_OLC_IND *Jun 14 15:35:01.979:
cch323_h245_olc_sm: received event H245_EVENT_OLC_IND
while at state H245_OLC_DONE !---Session ID was sent as
voice session ID, fallback to voice and the call
disconnected: PDU DATA = 61593960 value
MultimediaSystemControlMessage ::= response :
openLogicalChannelAck : ??? { ??????
forwardLogicalChannelNumber 4 ??????
forwardMultiplexAckParameters
h2250LogicalChannelAckParameters : ?????? { ??????????
sessionID 1 ?????????? mediaChannel unicastAddress :
IPAddress : ?????????? { ??? ??????????network 'C95C381E'H
???????????? tsapIdentifier 18100 ?????????? } ??????????
mediaControlChannel unicastAddress : IPAddress : ??????????
{ ????????????? network 'C95C381E'H ????????????? tsapIdentifier
18101 ?????????? } ?????????? flowControlToZero FALSE ?????? }
??? } RAW_BUFFER::= 22 C0000304 80145C00 00C95C38
1E46B400 C95C381E 46B50300 0100 *Jun 14 15:35:01.983:

```

Esta seção detalha a anatomia de um fax bem sucedido de T.38 setup entre um roteador do AS5300 Series e um Modular Access Router do Cisco 3640. As saídas do comando **debug and show** foram capturadas no comando **debug vtsp all** em um Modular Access Router do Cisco 3640 como o TGW IO 12.4:

saída do comando debug vtsp all

```

Router# debug vtsp all

Voice telephony call control all debugging is on

```

```
!--- At this point, the VTSP is not aware of anything.
The format of this message is //callid/GUID/VTSP:(voice-
port):T1-channel_number:DSP_number:DSP_channel_number:
•CallEntry ID is -1. •GUID is xxxxxxxxxxxx. •The voice
port is blank. •Channel ID is -1. •DSP ID is -1. •DSP
channel ID is -1. *Mar 1 08:23:10.869: //-
1/xxxxxxxxxxxxx/VTSP():-1:-1:-
1/vtsp_do_regxrule_translate: !--- The original and the
translated calling number are the same (55555) and the
original and the translated called number are the same
(888545). These numbers are often the same because if a
translation rule is applied, it will be on the dial
peers or the ports, both of which comes later than these
VTSP messages in the Cisco IOS code execution. *Mar 1
08:23:10.869: //-1/xxxxxxxxxxxxx/VTSP():-1:-1:-1/vtsp
_do_regxrule_translate: calling_number(original)=
calling_number(xlated)=55555 called_number(original)=
called_number(xlated)=888545 redirectNumber(original)=
redirectNumber(xlated)= !--- The VTSP got a call setup
indicator from the TSP layer with called number 888545
and calling number 55555. There is no awareness of the
CallEntry ID (-1) or the GUID (xxxxxxxxxxxx). *Mar 1
08:23:10.873: //-1/xxxxxxxxxxxxx/VTSP():-1:-1:-
1/vtsp_tsp_call_setup_ind: (sdb=0x634C90EC,
tdm_info=0x0, tsp_info=0x63083950, calling_number=55555
calling_oct3 = 0x80, called_number=888545 called_oct3 =
0x80, oct3a=0x0): peer_tag=10002 *Mar 1 08:23:10.873:
//-1/xxxxxxxxxxxxx/VTSP():-1:-1:-
1/vtsp_tsp_fill_setup_ind : ev.clg.clir is 0
ev.clg.clid_transparent is 0 ev.clg.null_orig_clg is 0
ev.clg.calling_translated is false *Mar 1 08:23:10.873:
//-1/xxxxxxxxxxxxx/VTSP():-1:-1:-
1/vtsp_do_call_setup_ind: . *Mar 1 08:23:10.873: //-
1/xxxxxxxxxxxxx/VTSP():-1:-1:-1/vtsp_allocate_cdb: ,cdb
0x635FC480 *Mar 1 08:23:10.873: //-
1/xxxxxxxxxxxxx/VTSP():-1:-1:-1/vtsp_do_call_setup_ind:
*Mar 1 08:23:10.873: source route label !--- At this
point, the VTSP is not aware of anything. The format of
this message is //callid/GUID/VTSP:(voice-port):T1-
channel_number:DSP_number:DSP_channel_number: •CallEntry
ID is -1. •GUID is D2F6429A8A8A. •The voice port is
1/0:23 where 23 indicates D channel. •The T1 channel is
still unknown at this point (-1). •The digital signal
processor (DSP) is 0. •The DSP channel is 4. *Mar 1
08:23:10.873: //-1/D2F6429A8A8A/VTSP:(1/0:23):-
1:0:4/vtsp_do_call_setup_ind: Call ID=101002,
guid=635FCB08 !--- The VTSP learns about the B channel
(changed from -1 to 22), and the CallEntry ID is still
unknown (-1). *Mar 1 08:23:10.873: //-
1/D2F6429A8A8A/VTSP:
(1/0:23):22:0:4/vtsp_do_call_setup_ind: type=0,
under_spec=1615186336, name=, id0=23, id1=0, id2=0,
calling=55555,called=888545
subscriber=RegularLinevtsp_do_call_setup_ind: redirect
DN = reason = -1 *Mar 1 08:23:10.877: //-
1/xxxxxxxxxxxxx/VTSP():-1:-1:-
1/vtsp_do_normal_call_setup_ind: . !--- The VTSP learns
the CallEntry ID. The format of this message is
//callid/GUID/VTSP:(voice-port):T1-
channel_number:DSP_number:DSP_channel_number: •CallEntry
ID is 899 (changed from -1 to 899) •GUID is D2F6429A8A8A
•The voice port is 1/0:23 where 23 indicates D channel
•The T1 channel is 22 •The DSP is 12 •The DSP channel is
4 *Mar 1 08:23:10.877: //899/D2F6429A8A8A/VTSP:(1/0:23)
```

```
:22:12:4/vtsp_insert_cdb:,cdb 0x635FC480, CallID=899
*Mar 1 08:23:10.877:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/vtsp_open_voice
_and_set_params: . !--- In these outputs, VTSP sets some
of the voice parameters for this call: •Modem capability
•Playout delay •Dial-peer tag 10003 •Digit timeouts *Mar
1 08:23:10.877: //899/D2F6429A8A8A/VTSP:(1/0:23):
22:12:4/vtsp_modem_proto_from_cdb: cap_modem_proto 0
*Mar 1 08:23:10.881:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/set_playout_cdb
:playout default *Mar 1 08:23:10.881:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/vtsp_dsp_echo_c
anceller_control: echo_cancel: 1 *Mar 1 08:23:10.885:
//899/D2F6429A8A8A/VTSP:
(1/0:23):22:12:4/vtsp_save_dialpeer_tag: tag = 10003
*Mar 1 08:23:10.885: //899/D2F6429A8A8A/VTSP:
(1/0:23):22:12:4/vtsp_report_digit_control:
vtsp_report_digit_control: enable=0: *Mar 1
08:23:10.885: //899/D2F6429A8A8A/VTSP:(1/0:23):
22:12:4/vtsp_report_digit_control: digit reporting
disabled *Mar 1 08:23:10.885:
//899/D2F6429A8A8A/VTSP:(1/0:23):
22:12:4/vtsp_get_digit_timeouts: :
vtsp_get_digit_timeouts !--- VTSP sends out a call-
proceeding message to the POTS leg *Mar 1 08:23:10.885:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/vtsp_process_ev
ent:vtsp:[1/0:23:899, S_SETUP_INDICATED,
E_CC_PROCEEDING] *Mar 1 08:23:10.885:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/act_proceeding:
. *Mar 1 08:23:10.941: //899/D2F6429A8A8A/VTSP:
(1/0:23):22:12:4/vtsp_get_dialpeer_tag: tag = 10003 *Mar
1 08:23:10.949: //899/D2F6429A8A8A/VTSP:(1/0:23):
22:12:4/vtsp_get_dialpeer_tag: tag = 10003 !--- VTSP
sends out an alerting to the POTS leg; the phone is
ringing at this time. *Mar 1 08:23:10.949:
//899/D2F6429A8A8A/VTSP:
(1/0:23):22:12:4/vtsp_process_event: vtsp:[1/0:23:899,
S_PROCEEDING, E_CC_ALERT] *Mar 1 08:23:10.949:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/act_alert: .
*Mar 1 08:23:10.949:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/vtsp_timer_stop
:3019095 *Mar 1 08:23:18.769:
//899/D2F6429A8A8A/VTSP:(1/0:23):
22:12:4/vtsp_get_dialpeer_tag: tag = 10003 !--- The
phone gets answered here, a bridge is now set up between
the two call legs. *Mar 1 08:23:10.949:
//899/D2F6429A8A8A/VTSP:
(1/0:23):22:12:4/vtsp_process_event: vtsp:[1/0:23:899,
S_PROCEEDING, E_CC_ALERT] *Mar 1 08:23:10.949:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/act_alert: .
*Mar 1 08:23:10.949:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/vtsp_timer_stop
:3019095 *Mar 1 08:23:18.769:
//899/D2F6429A8A8A/VTSP:(1/0:23):
22:12:4/vtsp_get_dialpeer_tag: tag = 10003 !--- The call
is now connected. Mar 1 08:23:18.769:
//899/D2F6429A8A8A/VTSP:(1/0:23)
:22:12:4/vtsp_process_event: vtsp:[1/0:23:899,
S_ALERTING, E_CC_CONNECT] *Mar 1 08:23:18.769:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/act_alert_conne
ct: . *Mar 1 08:23:18.773:
//899/D2F6429A8A8A/VTSP:(1/0:23):22:12:4/vtsp_ring_noan
timer_stop: 3019877
```

Informações Relacionadas

- [Configurando o fax relay com VoIP \(T.38\)](#)
- [Manual de Troubleshooting de Fax Relay](#)
- [Suporte do Gateway Cisco AVVID para Fax Relay e passagem de fax](#)
- [Depuração de provedor de serviço de telefonia de voz](#)
- [Pesquisando defeitos o fax](#)
- [Suporte à Tecnologia de Voz](#)
- [Suporte ao Produto de Voz e Comunicações Unificadas](#)
- [Troubleshooting da Telefonia IP Cisco](#)
- [Suporte Técnico e Documentação - Cisco Systems](#)