Problemas de Fax Relay H.323 T.38

Contenido

Introducción

Los problemas de Fax Relay T.38 suelen estar asociados a problemas de interoperabilidad entre Cisco y gateways de fax T.38 de otras compañías. Este documento proporciona ejemplos detallados de comandos de debug de llamadas correctas y fallidas de Fax Relay T.38. Estos resultados de comandos de debug contienen comentarios para proporcionar puntos de referencia, de modo que pueda identificar y resolver esos problemas de interoperabilidad. En este documento también se proporcionan los comandos de troubleshooting y verificación pertinentes.

Prerequisites

Requirements

Los lectores de este documento deben tener conocimiento de los conceptos básicos de relé de fax. Consulte [Guía de Troubleshooting de Fax Relay](#) para obtener más información sobre los conceptos de fax relay y los pasos básicos de troubleshooting.

Componentes Utilizados

Este documento no tiene restricciones específicas en cuanto a versiones de software y de hardware.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Convenciones

Consulte Convenciones de Consejos TécnicosCisco para obtener más información sobre las convenciones del documento.

Conceptos básicos de T.38

Un síntoma común de los problemas de relé de fax T.38 es una llamada de voz que se establece donde se escucha un tono de fax, pero la negociación de fax no se completa y la llamada finalmente se interrumpe. A menudo, este problema se asocia a problemas de interoperabilidad de gateway T.38 de Cisco y T.38 de terceros.

El relé de fax T.38 es transmisión de fax en tiempo real; es decir, dos máquinas de fax que se comunican entre sí como si hubiera una línea telefónica directa entre las dos. El relé de fax se configura con algunos comandos adicionales en los pares de marcado de gateway que ya se han definido y configurado para llamadas de voz.

Cisco proporciona dos métodos para la retransmisión de fax: un método propiedad de Cisco y un método basado en el estándar ITU-T T.38. En la mayoría de las plataformas, Cisco fax relay es el valor predeterminado si un método de fax no está configurado explícitamente. Cisco Fax Relay se describe en **Configuración de Cisco Fax Relay** .

Restricciones de T.38

En este momento, el relé de fax Cisco T.38 tiene estas restricciones:

- La interoperabilidad T.38 requiere la versión 2 de Cisco H.323.
- T.38 no es compatible con los Cisco MC3810 Series Concentrators con un Módulo de compresión de voz (VCM).
- El proxy H.323 del administrador de conferencias multimedia (MCM) no admite T.38.
- Sólo se implementa el protocolo de datagramas de usuario (UDP) para H.323 T.38.
- Algunos gateways y gatekeepers de terceros no son compatibles con los gateways de voz de Cisco para relé de fax T.38 porque diferentes fabricantes pueden elegir ciertas partes de H.323 y T.38 para implementarlas en sus gateways y gatekeepers. Las pruebas de interoperabilidad de voz con estos gateways y gatekeepers de terceros deben realizarse para asegurarse de que el relé de fax T.38 pueda tener éxito.

Negociación T.38

Esta sección proporciona un breve resumen paso a paso de cómo se maneja la negociación T.38 dentro de los gateways de Cisco. Refiérase a Guía de Troubleshooting de Fax Relay para obtener más información sobre los conceptos básicos de fax relay.

1. En el mensaje de configuración inicial, la puerta de enlace de origen (OGW) anuncia la capacidad de datos T.38.Si la puerta de enlace de terminación (TGW) admite la capacidad de datos T.38, puede retransmitir esa información en los mensajes posteriores enviados al OGW.

2. Una vez que se establece una llamada de voz y el Procesador de señal digital (DSP) del TGW detecta un tono de fax, la máquina de estado Proveedor de servicios de telefonía de voz (VTSP) informa al tramo de llamada H.323, que negocia el modo T.38 con el OGW.
3. Tras la confirmación del modo T.38, el canal de audio se cierra y el canal lógico T.38 se abre en ambos extremos.
4. A nivel de código VTSP, se realiza la descarga del codificador-decodificador (códec) de fax.
5. Tras una descarga exitosa de T.38 Open Logical Channel (OLC) y códec, VTSP entra en modo de fax.
6. Una vez finalizada la transmisión del fax, la llamada se devuelve a una llamada de voz.**Nota:** Durante la negociación del modo T.38, si el otro extremo no reconoce el modo T.38, la llamada se devuelve a una llamada de voz y se desconecta. Si se recibe un reconocimiento negativo del otro extremo en relación con el OLC T.38, la llamada también se vuelve a una llamada de voz y se desconecta.

Resolución de problemas de T.38

Consejos para la resolución de problemas de relé de fax H.323 o SIP T.38

Para resolver el problema del relé de fax T.38, realice estos pasos:

- **Asegúrese de que puede realizar una llamada de voz.** Confirme que las llamadas de voz normales se pueden completar antes de investigar la conectividad del fax. Si no hay ningún teléfono conectado, desconecte la máquina de fax y conecte un teléfono común. Si las llamadas de voz normales no se conectan, el problema puede estar relacionado con VoX, y usted puede resolver el problema como un problema de conectividad de voz normal antes de continuar con la resolución de problemas de fax.
- Asegúrese de que el protocolo de fax deseado se configuró con el comando fax protocol en las gateways de origen y de terminación.
- Asegúrese de que el protocolo de fax esté configurado como T.38 en el nivel de configuración global o en el nivel de configuración de par de marcado para las gateways de origen y de terminación.

Comandos debug y show

Los comandos **debug** y **show** utilizados para la resolución de problemas de relé de fax T.38 son:

- **debug voip ccapi inout**: este comando rastrea la ruta de ejecución a través de la interfaz de programa de aplicación de control de llamadas (API), que actúa como interfaz entre la aplicación de sesión de llamada y el software específico de la red subyacente. Puede utilizar el resultado de este comando para comprender cómo el gateway de voz maneja las llamadas.
- **debug vtsp all**: Este comando habilita estos comandos **debug** VTSP: **debug vtsp session**, debug vtsp error y debug vtsp dsp.
- **debug h245 asn1**: Este comando muestra el contenido de Notación de Sintaxis Abstracta Uno (ASN.1) de los mensajes H.245. Para inhabilitar el resultado de la depuración, utilice la forma no de este comando.
- **debug cch323 h245**: Este comando proporciona el seguimiento de la transición de estado de la máquina de estado H.245 basada en los eventos procesados. Para inhabilitar el resultado

de la depuración, utilice la forma no de este comando.

- show call active fax brief —Este comando muestra información de llamada para las transmisiones de fax de almacenamiento y reenvío en curso.
- show call history fax —Este comando muestra el historial de llamadas recientes para faxes.

Salida de una llamada T.38 exitosa

Esta sección detalla la anatomía de una configuración de fax T.38 exitosa entre un router serie AS5300 y un router de acceso modular Cisco 3640. Los resultados de los comandos **debug** y **show** fueron capturados en el Cisco AS5300 Universal Gateway como el TGW IOS 12.2:

debug vtsp all command output

```
!---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)csize(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)
```

Ejemplo de una llamada T.38 fallida

Este es un ejemplo de la salida del comando **debug** para una llamada T.38 fallida:

salida del 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 whil

e 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 sección detalla la anatomía de una configuración de fax T.38 exitosa entre un router serie AS5300 y un router de acceso modular Cisco 3640. Los resultados de los comandos **debug** y **show** se capturaron en el comando **debug vtsp all** en un router de acceso modular Cisco 3640 como el IOS 12.4 de TGW:

salida del 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 xxxxxxxxxx. •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/xxxxxxxxxxxx/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/xxxxxxxxxxxx/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/xxxxxxxxxxxx/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/xxxxxxxxxxxx/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/xxxxxxxxxxxx/VTSP:():-1:-1:-
1/vtsp_do_call_setup_ind: . *Mar 1 08:23:10.873: //-
1/xxxxxxxxxxxx/VTSP:():-1:-1:-1/vtsp_allocate_cdb: ,cdb
0x635FC480 *Mar 1 08:23:10.873: //-
1/xxxxxxxxxxxx/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/xxxxxxxxxxxx/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
```

Información Relacionada

- Configuración de Fax Relay con VoIP (T.38)
- Guía de resolución de problemas de Fax Relay
- Ayuda de gateway AVVID de Cisco para Fax Relay y Fax Pass-Through
- Depuración del proveedor de servicio de telefonía de voz
- Resolución de problemas de fax
- Soporte de tecnología de voz
- Soporte de Productos de Voice and Unified Communications
- Troubleshooting de Cisco IP Telephony
- Soporte Técnico y Documentación - Cisco Systems