

Problemas y casos prácticos de la SEÑAL JTAPI

TAC

ID del Documento: 116060

Actualizado: De sept el 06 de 2013

Contribuido por Michael Mendoza y Bakthavatsal Muralidharan, ingenieros de Cisco TAC.



[Descarga PDF](#)



[Imprimir](#)

[Comentarios](#)

Productos Relacionados

- [Cisco Unity Express](#)

Contenido

[Introducción](#)

[prerrequisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Integración de la SEÑAL JTAPI con la descripción CUCM](#)

[Ejemplo de alto nivel del flujo de llamada](#)

[Habilitación y colección de trazas](#)

[Trazas en tiempo real del CCN JTAPI](#)

[Registros de la traza del CCN JTAPI](#)

[Recoja los archivos del registro de la traza](#)

[Detalles imprescindibles antes de que usted marque los registros](#)

[Conceptos básicos CTI](#)

[Estados de la llamada comunes CTI](#)

[Qué traza registra debe parecer](#)

[CTI RP y registro del puerto](#)

[Llamadas básico remitidas al voicemail](#)

[Nuevos llamada y cambio de dirección al puerto disponible](#)

[Nuevos llamada y cambio de dirección al puerto disponible](#)

[Nueva llamada al puerto CTI](#)

[El puerto CTI valida la llamada reorientada](#)

[Negociación de medios](#)

[Desconexión de la llamada](#)

[Señalización con./desc. del MWI](#)

[La SEÑAL enciende la lámpara MWI para la línea 3001](#)
[Marcado '3' del número DTMF para borrar el mensaje del buzón](#)
[La SEÑAL apaga la lámpara MWI para la línea 3001](#)
[Registros en tiempo real del CCN](#)
[Configuración de la llamada](#)
[Desconexión de la llamada](#)
[Resolver problemas los casos prácticos](#)
[Problemas de conectividad](#)
[Usuario CTI-no habilitado](#)
[El servicio del Administrador CTI. CUCM está abajo](#)
[Discrepancia de configuración](#)
[Problema del ruteo de llamadas CUCM](#)
[Lista de verificación para el troubleshooting del registro del puerto](#)
[Información Relacionada](#)
[Discusiones relacionadas de la comunidad del soporte de Cisco](#)

Introducción

Este documento proporciona la información sobre cómo resolver problemas el Java Telephony Application Programming Interface del Cisco Unity Express (SEÑAL) (JTAPI). Además, este documento proporciona la información y los comandos en cómo habilitar, recoger, y ver las diversas trazas y registros con los ejemplos del caso práctico del troubleshooting.

Prerrequisitos

Requisitos

Cisco recomienda que tenga conocimiento sobre estos temas:

- Conocimiento básico de cómo configurar y utilizar al administrador de las Comunicaciones unificadas de Cisco (CUCM) a través de la interfaz administrativa de la red.
- Familiaridad básica con los puertos y los puntos de ruta (RP) del Computer Telephony Interface (CTI) en CUCM.
- Familiaridad básica con la interfaz de la línea de comandos del Cisco Unity Express.

Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Versión 3.x o posterior del Cisco Unity Express.
- Versión del administrador 7.x de las Comunicaciones unificadas de Cisco o más adelante.

El método de integración usado solicita solamente el Cisco Unity Express con el administrador de las Comunicaciones unificadas de Cisco; no con el Cisco Unified Communications Manager

Express (CUCME).

El Cisco Unity Express se debe autorizar para CUCM, no CUCME. La SEÑAL se puede integrar con CUCM o CUCME en cualquier momento y autorizar por consiguiente.

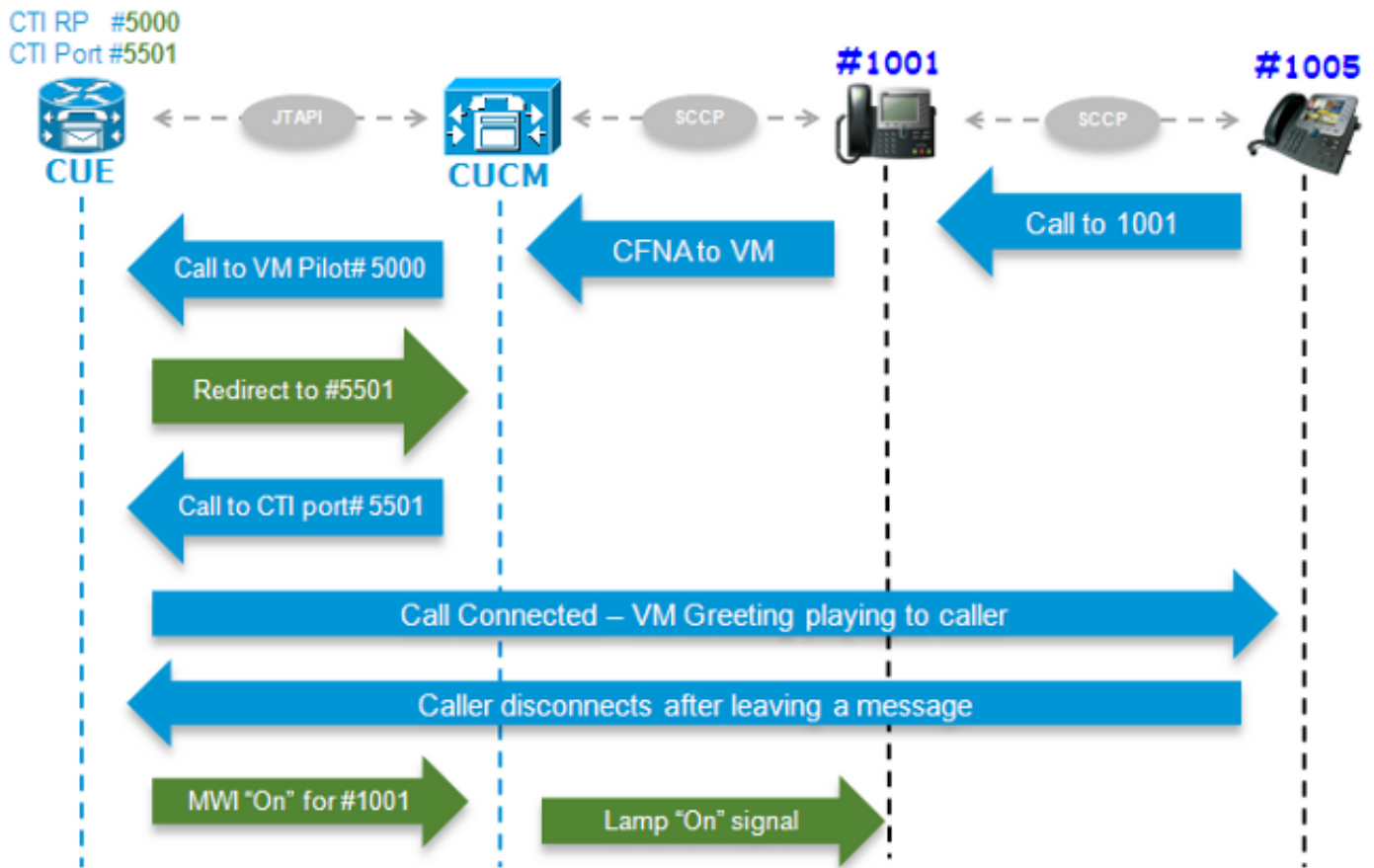
La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si la red está funcionando, asegúrese de haber comprendido el impacto que puede tener cualquier comando.

CUENTE la integración JTAPI con la descripción CUCM

Es posible integrar la SEÑAL con CUCM con el protocolo JTAPI para el correo de voz (VM) y las funciones del Automated Attendant (AA). Se recomienda esta solución cuando usted quiere provision las características VM y/o el manejo de llamadas básico AA para uno o los sitios con varias sucursales con una pequeña cantidad de usuarios se registra a un servidor CUCM. Esto no requiere un servidor de correo de voz hecho y derecho del Cisco Unity, sino una implementación mucho más asequible. Al mismo tiempo, la SEÑAL también proporciona las opciones de la supervivencia para sus bifurcaciones y falla encima al protocolo iniciado sesión (SORBO) cuando la Conectividad al CUCM se pierde.

La SEÑAL puede registrarse con el CUCM con el JTAPI y controla los puntos de ruta CTI y los puertos CTI. Esto permite que usted controle y que maneje la SEÑAL como punto final adicional con el CUCM, así como facilita las configuraciones y las interacciones con otros puntos finales en el cluster.

Ejemplo de alto nivel del flujo de llamada



El usuario final con el número de directorio (DN) 1005 llama al usuario con DN 1001. La llamada se remite después de algunos segundos si la llamada no se contesta, el Call Forward No Answer (CFNA), al número VM configurado en el perfil del usuario 1001 VM. El CUCM entonces envía la llamada al piloto configurado 5000 VM, que señala a un CTI RP con el DN 5000 que es controlado por la SEÑAL. Se acciona la aplicación de la SEÑAL VM, y la llamada se reorienta con el JTAPI a un puerto disponible CTI (DN 5501) para el establecimiento de media. Los juegos del saludo del audio, y el usuario pueden salir de un mensaje u obrar recíprocamente con el sistema con los tonos del Dual Tone Multi-frequency (DTMF). Cuando el llamador termina la llamada, la SEÑAL señala el CUCM para fijar la lámpara del indicador de mensaje en espera (MWI) para la extensión 1001 a "ON" con el JTAPI. El CUCM entonces envía el mensaje del protocolo skinny client control (SCCP) para encender la luz en el teléfono, así como muestra la indicación del sobre en la visualización así que el usuario 1001 es consciente que hay un nuevo mensaje VM en el buzón.

Habilitación y colección de trazas

Hay dos tipos de trazas:

- Trazas en tiempo real de la red de comunicaciones JTAPI Cisco (CCN)
- Registros de la traza del CCN JTAPI

Trazas en tiempo real del CCN JTAPI

- Trazas en tiempo real del CCN JTAPI. (Habilitar estas trazas no requiere una recarga del módulo de la SEÑAL.)
- La salida no es tan extensa como los registros de la traza del CCN, sino que no son muy

informativos tampoco.

Ingrese estos comandos para habilitar las trazas:

```
no trace all
trace ccn SubsystemJtapi all
```

Ingrese este comando para verificar que están habilitadas:

```
CUE# show trace
MODULE ENTITY SETTING
ccn SubsystemJtapi ffffffff
```

Ingrese este comando para recoger la salida:

```
CUE# show trace buffer ?
containing Only display events matching a regex pattern
long Show long format
short Show short format
tail Wait for events and print them as they occur !!
```

Ingrese el **CTRL-C** para parar la registración en tiempo real a la consola.

Registros de la traza del CCN JTAPI

Una recarga del módulo de la SEÑAL se requiere después de que los registros de la traza del CCN JTAPI se habiliten para que los registros se pueblen. Estos registros, **messages.log** y **atrace.log**, pueden ser muy detallados o secretos así como mucho más informativo y detallado. Hay cuatro diversos registros:

- **atrace.log**

Habilitado por abandono en los módulos de red (NM), pero inhabilitado por abandono para los módulos advanced integration (objetivos). Ingrese el **comando log trace local enable** para habilitar. Prepara al 10 Mb localmente o a un servidor FTP. Para recomenzar el registro, ingrese el **comando log trace local disable** o el **comando no log trace local enable**; entonces ingrese el **comando log trace local enable**. Ingrese el comando claro del archivo de traza para borrar **atrace.log**. Los datos se deben decodificar por el Centro de Asistencia Técnica (TAC).

- **messages.log**

Éstos son los registros que contienen los mensajes de Syslog, tales como información, advertencia, error, y fatal.

- **CiscoJtapi1.log** y **CiscoJtapi2.log**

Registran toda la señalización y eventos JTAPI-relacionados. Estos registros son mucho más fáciles de entender y muy informativo. CiscoJtapi2.log comienza a poblar cuando se convierte CiscoJtapi1.log por completo y viceversa.

Sin importar se fijan qué trazas, el sistema invierte a los niveles de traza predeterminados después de una recarga. Para cambiar estas configuraciones predeterminadas de modo que sobrevivan una reinicialización, usted debe ingresar el **comando log trace boot**. Aquí está el comando de habilitarlos:

```
CUE#(CONFIG)> log console info !!
ccn trace jtapi deb all
ccn trace jtapi info all
ccn trace jtapi warn all
log trace boot
reload
```

Ingrese este comando para verificar que están habilitados:

```
CUE# show ccn trace jtapi
Warning: 1
Informational: 1
Jtapi Debugging: 1
Jtapi Implementation: 1
CTI Debugging: 1
CTI Implementation: 1
Protocol Debugging: 1
Misc Debugging: 1
```

Aquí están los pasos para ver los registros:

1. Ingrese el **comando show logs** para ver una lista de los archivos del registro salvados en la SEÑAL.
2. La extensión de archivo del **.prev** significa que éste es un respaldo de un más viejo archivo de traza y no del archivo de registro activo actual.
3. Usted puede extraerlos a un servidor FTP externo.
4. Usted puede también ver la salida de los mensajes que se están registrando a estos archivos en tiempo real del monitor terminal de la SEÑAL.

Recoja los archivos del registro de la traza

Extraiga los registros a un externo FTP con estos comandos:

```
copy log CiscoJtapi2.log url ftp://username:password@192.168.105.1/
copy log CiscoJtapi1.log url ftp://username:password@192.168.105.1/
copy log messages.log url ftp://username:password@192.168.105.1/
copy log atrace.log url ftp://username:password@192.168.105.1/
```

Visualice los registros al monitor terminal de la SEÑAL con el **<logname >** el comando del **nombre del registro de la demostración**. Aquí tiene un ejemplo:

```
CUE# show log name messages.log ?
containing Only display events matching a regex pattern
paged Display in page mode
tail Wait for events and print them as they occur
<cr>
```

Se codifica **atrace.log**; por lo tanto usted puede no sólo visualizarlo en tiempo real con el **comando show log name**.

Detalles imprescindibles antes de que usted marque los registros

Usted debe obtener, por lo menos, todos los detalles delineados adjunto de las llamadas con el problema que usted está resolviendo problemas de modo que usted pueda seguir y entender fácilmente las trazas:

- Número que llama
- Número al que se llamó
- Reoriente el número
- CTI RP DN y Nombre del dispositivo
- Número del puerto y Nombre del dispositivo CTI
- Usuario de JTAPI
- El rango de tiempo las llamadas ocurrió

Conceptos básicos CTI

Proveedor: Un proveedor de los servicios CTI. La aplicación establece una sesión CTI **abriendo** un proveedor.

Usuario: Las aplicaciones se asocian a un usuario.

Dispositivo: Un dispositivo que se registra al CUCM.

Línea: Aspecto DN en un dispositivo admitido CTI.

ID de llamada (callLegID): Asociado a un tramo de llamada en una llamada.

Llamada global (callID): Identifica todos los tramos de llamada para una sola llamada.

Estados de la llamada comunes CTI

```
state = 1                IDLE
state = 2 OFFERING
state = 3 ACCEPTED
state = 8 CONNECTED
```

Qué traza registra debe parecer

Antes de que usted pueda encontrar la señalización incorrecta, usted primero necesita conocer lo que parecería esta señalización el funcionamiento normal inferior; esta sección muestra tanto como el snippets de la señalización le hace salir vería en diversos escenarios cuando trabajan normalmente.

También satisfaga sea consciente que toda la señalización de estos registros se ha resumido para mostrar solamente los detalles relevantes porque contienen mismo la información detallada que es muy aburrida y repetidor.

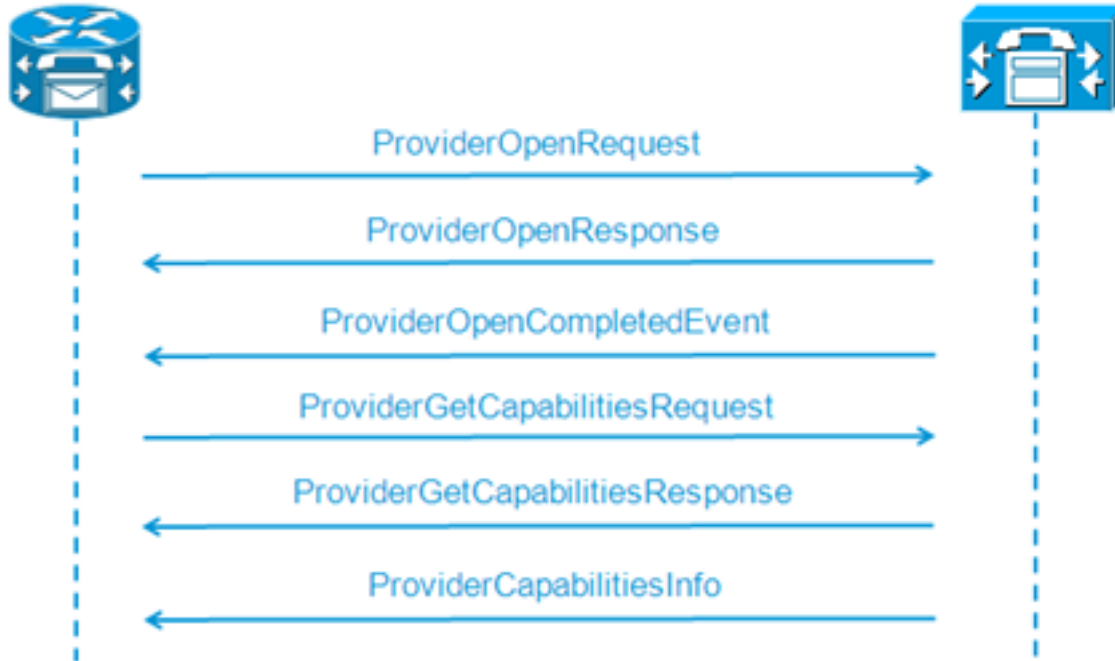
Aquí están los detalles de las configuraciones usadas:

```
Jtapi User:                tacjtapiuser
CUCM IP Address: 192.168.100.10
CUE CTI Route Point: cue_vm_ctirp
CUE CTI Port: cue_ctiport1
CUE and Phone Partition: cue_pt
IP Phone MAC: SEP0023331C29EC
CTI Route Point DN: 8000
CTI Port DN: 8501
IP Phone DN: 3001
```

CTI RP y registro del puerto

(Salidas de los registros CiscoJtapi1/de Cisco Jtapi2)

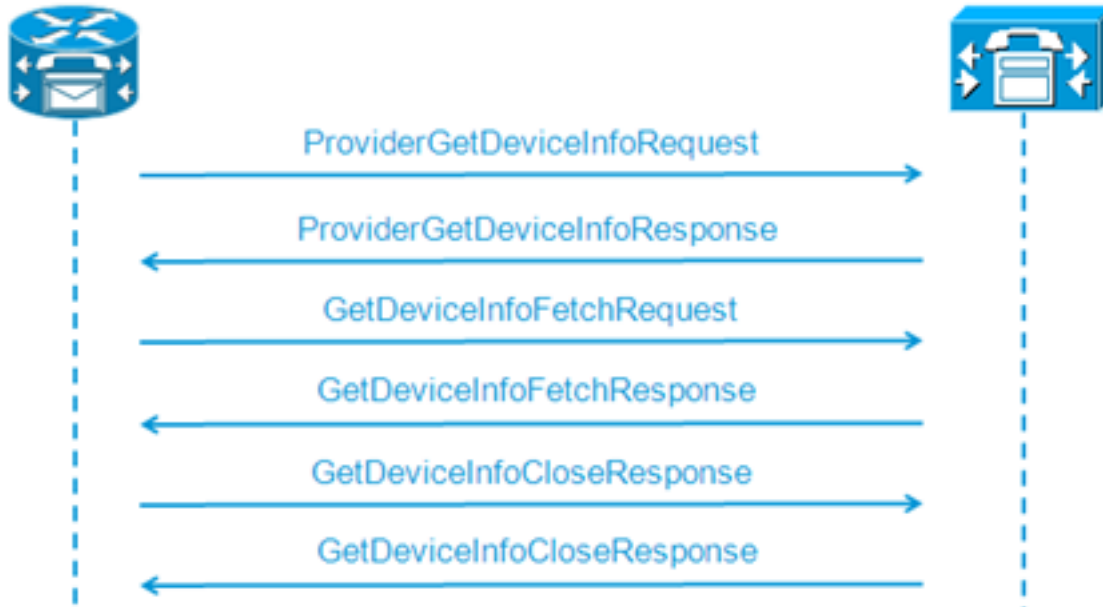
1. Abra una conexión del proveedor



```

21: 12:05:23.686 CST %JTAPI-CTIIMPL-7-UNK.(P1-tacjtapiuser) ProviderID =
P1-tacjtapiuser
22: 12:05:23.739 CST %JTAPI-CTIIMPL-7-UNK.(P1-tacjtapiuser) Trying to
create normal socket connection to 192.168.100.10
23: 12:05:23.747 CST %JTAPI-CTIIMPL-7-UNK.(P1-tacjtapiuser) connected
26: 12:05:24.112 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.ProviderOpenRequest {
provider = 192.168.100.10
qbeClientVersion = Cisco JTAPI 7.0(1.1000)-1 Release
login = com.cisco.cti.protocol.UnicodeString {
unicodedisplayName = tacjtapiuser
}
applicationID = Cisco IP IVR
desiredServerHeartbeatTime = 30
pluginName = CiscoJTAPI
}
28: 12:05:24.131 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.ProviderOpenResponse {
sequenceNumber = 0
result = 0
providerInfoString = 7.1.5.10000-12
clientHeartbeat = 30
serverHeartbeat = 30
pluginVersion = 7.1.5.10000-2
pluginLocation = http://192.168.100.10/plugins/
providerId = 16777236
}
35: 12:05:24.858 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.ProviderOpenCompletedEvent {
eventSequence = 0
reason = 0
providerInfoString = 7.1.5.10000-12
clientHeartbeat = 30
serverHeartbeat = 30
failureDescription = null
providerId = 16777236
}
  
```

2. Interrogación para los dispositivos controlables



```

48: 12:05:24.864 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.ProviderGetDeviceInfoRequest {
sequenceNumber = 2
deviceGroup = 1
}
49: 12:05:24.865 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.ProviderGetDeviceInfoResponse {
sequenceNumber = 2
result = 0
}
50: 12:05:24.865 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.GetDeviceInfoFetchRequest {
sequenceNumber = 3
}
51: 12:05:25.011 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.GetDeviceInfoFetchResponse {
sequenceNumber = 3
result = 0
info = 2@[
com.cisco.cti.protocol.DeviceInfo {
name = cue_ctiport1
type = 72
allowsRegistration = true
deviceID = 62
devTypeName = CTI Port
},
com.cisco.cti.protocol.DeviceInfo {
name = cue_vm_ctirp
type = 73
allowsRegistration = true
deviceID = 61
devTypeName = CTI Route Point
}]
}
52: 12:05:25.012 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.GetDeviceInfoCloseRequest {
sequenceNumber = 4
}
53: 12:05:25.013 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.GetDeviceInfoCloseResponse {
sequenceNumber = 4
}
54: 12:05:25.013 CST %JTAPI-MISC-7-UNK.(P1-192.168.100.10)

```

creating controlled devices

3. Obtenga la línea información del puerto CTI



```
55: 12:05:25.024 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 5
deviceName = cue_ctiport1
}
56: 12:05:25.026 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 5
result = 0
}
57: 12:05:25.026 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 6
}
58: 12:05:25.029 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 6
result = 0
com.cisco.cti.protocol.LineInfo {
name = 8501
displayName =
maxNumberOfCalls = 4
lineInstance = 1
unicodeDisplayName = com.cisco.cti.protocol.UnicodeString {
}
partition = cue_pt
defaultIntercomTargetInfo = com.cisco.cti.protocol.LineIntercomSpeedDialInfo {
}
}
59: 12:05:25.029 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 7
}
60: 12:05:25.031 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 7
result = 0
}
61: 12:05:25.042 CST %JTAPI-CTI-7-UNK.(P1-tacjtapiuser)
DeviceMap: adding device "cue_ctiport1"
```

4. Obtenga la línea información CTI RP

```
62: 12:05:25.043 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
```

```

sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 8
deviceName = cue_vm_ctirp
}
63: 12:05:25.044 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 8
result = 0
}
64: 12:05:25.045 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 9
}
65: 12:05:25.047 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 9
result = 0
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 8000
displayName =
permanentLineID = 52
partition = cue_pt
defaultIntercomTargetInfo = com.cisco.cti.protocol.LineIntercomSpeedDialInfo {
unicodeLabel = com.cisco.cti.protocol.UnicodeString {
}
}
}
66: 12:05:25.048 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 10
}
67: 12:05:25.058 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 10
result = 0
}
68: 12:05:25.059 CST %JTAPI-CTI-7-UNK.(P1-tacjtapiuser)
DeviceMap: adding device "cue_vm_ctirp"
69: 12:05:25.059 CST %JTAPI-CTI-7-UNK.(P1-192.168.100.10)
refreshing device map: previous=0 current=2 created=2 removed=0

```

5. La SEÑAL aplica la configuración recibida

```

76: 12:05:25.064 CST %JTAPI-MISC-7-UNK.Provider
192.168.100.10
open, beginning device
initialization
77: 12:05:25.071 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[SS_TEL_INIT]
(P1-tacjtapiuser) Request: addObserver
79: 12:05:25.073 CST %JTAPI-MISC-7-UNK.ObserverThread
(com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@3d823d82):created
80:12:05:25.074 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser) ProvOutOfServiceEv [#0]
Cause:100 CallCtlCause:0 CiscoFeatureReason:12
82: 12:05:25.085 CST %JTAPI-MISC-7-
UNK.ObserverThread
(com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@3d823d82):
queuing com.cisco.jtapi.JtapiProviderEventSet
83: 12:05:25.084 CST %JTAPI-MISC-7-UNK.(P1-192.168.100.10)
ProviderRetryThread starting up
85: 12:05:25.084 CST %JTAPI-MISC-7-UNK.ObserverThread
(com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@3d823d82)
starting up...
90: 12:05:25.102 CST %JTAPI-JTAPIIMPL-7-UNK.Partition Support 8000 in
partitioncue_pt
91: 12:05:25.102 CST %JTAPI-JTAPIIMPL-7-UNK.(P1-tacjtapiuser) cue_vm_ctirp:
Address: 8000 in partitioncue_pt created

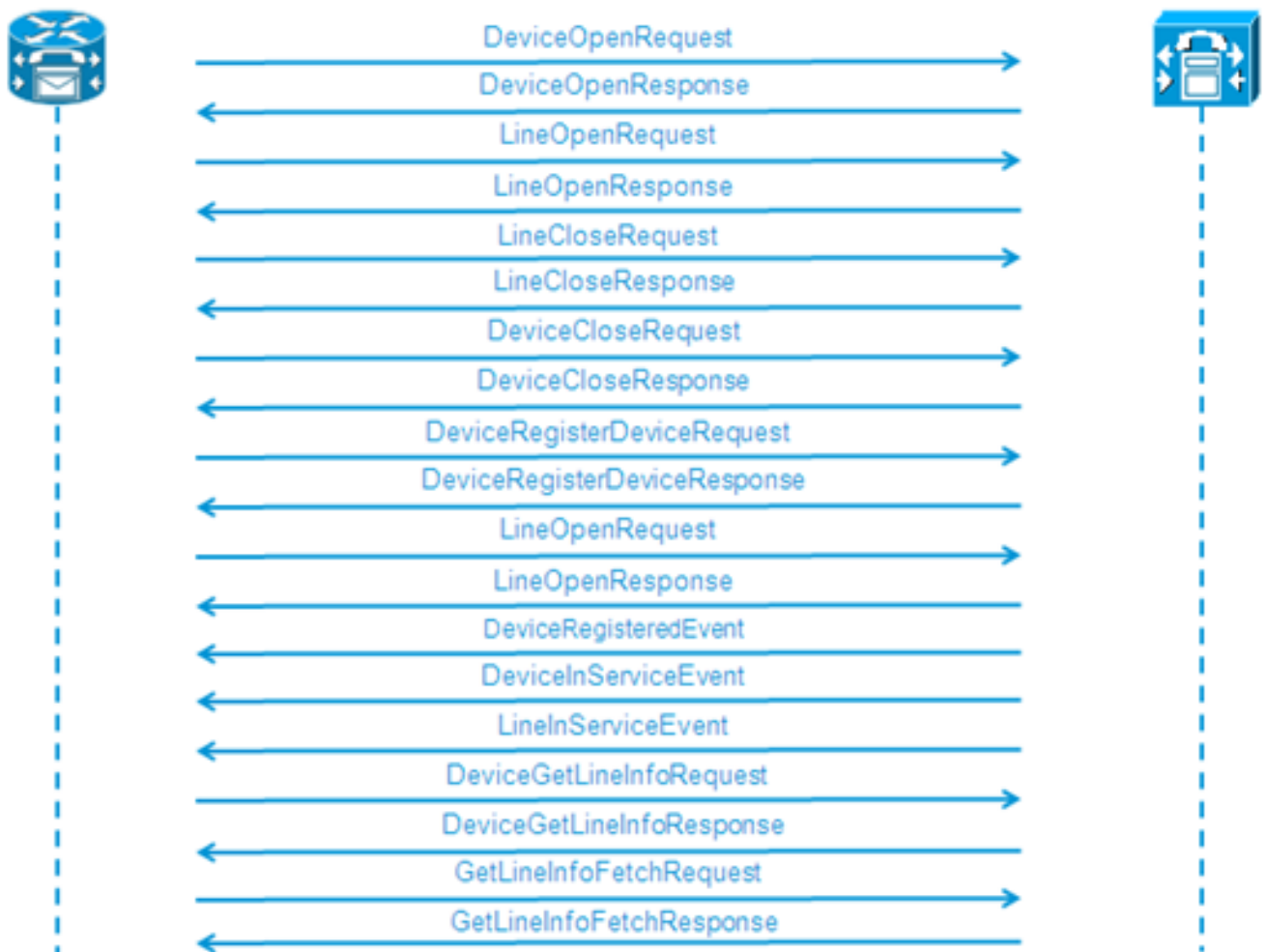
```

```

92: 12:05:25.102 CST %JTAPI-JTAPIIMPL-7-UNK.Partition Internal Address Added
8000 in Partition cue_pt
93: 12:05:25.102 CST %JTAPI-JTAPIIMPL-7-UNK.Partition Support 8501 in
partitioncue_pt
94: 12:05:25.103 CST %JTAPI-JTAPIIMPL-7-UNK.(P1-tacjtapiuser) cue_ctiport1:
Address: 8501 in partitioncue_pt created
95: 12:05:25.103 CST %JTAPI-JTAPIIMPL-7-UNK.Partition Internal Address Added
8501 in Partition cue_pt
96: 12:05:25.103 CST %JTAPI-MISC-7-UNK.Provider "(P1-tacjtapiuser)" changing
state to IN_SERVICE
97: 12:05:25.103 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[Thread-76]
(P1-tacjtapiuser) Request: getObservers
98: 12:05:25.103 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser) ProvInServiceEv [#1]
Cause:100 CallCtlCause:0 CiscoFeatureReason:12
100: 12:05:25.103 CST %JTAPI-MISC-7-UNK.ObserverThread
(com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@3d823d82):
queuing com.cisco.jtapi.JtapiProviderEventSet
101: 12:05:25.103 CST %JTAPI-JTAPIIMPL-7-UNK.Provider 192.168.100.10
initialized 2 devices
104: 12:05:25.104 CST %JTAPI-JTAPIIMPL-7-UNK:
[com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@3d823d82]
delivering to providerChangedEvent
106: 12:05:25.523 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[SS_TEL_INIT]
(P1-tacjtapiuser) Request: getAddress( 8501 )Partition = cue_pt
107: 12:05:25.526 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[SS_TEL_INIT]
[cue_ctiport1]Request: addObserver
(com.cisco.wf.subsystems.jtapi.TAPIPortGroup$Port$AddressCallObserver@5d085d08)

```

6. Obtenga el control de los dispositivos CTI y de las líneas



```

109: 12:05:25.528 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending:

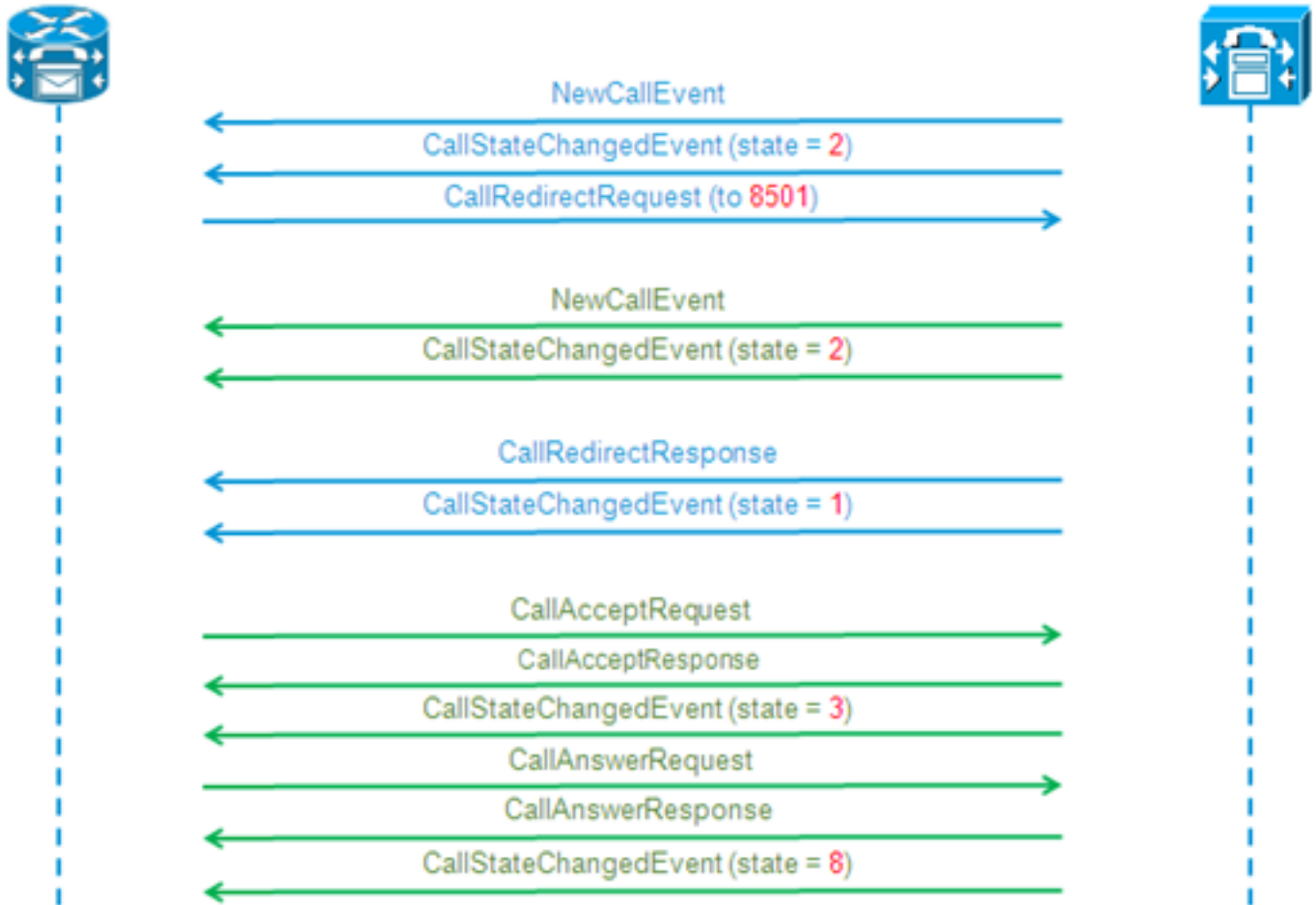
```

```
com.cisco.cti.protocol.DeviceOpenRequest {
deviceName = cue_ctiport1
}
110: 12:05:25.533 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response:
com.cisco.cti.protocol.DeviceOpenResponse {
result = 0
}
111: 12:05:25.533 CST %JTAPI-CTI-7-UNK.(P1-tacjtapiuser) DeviceMap: opening
device "cue_ctiport1"
112: 12:05:25.533 CST %JTAPI-JTAPIIMPL-7-UNK.(P1-tacjtapiuser) Terminal
"cue_ctiport1" out of service
113: 12:05:25.534 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser) [cue_ctiport1]
CiscoTermOutOfServiceEv [#2] Cause:100 CallCtlCause:0 CiscoFeatureReason:12
119: 12:05:25.544 CST %JTAPI-JTAPIIMPL-7-UNK:Address [cue_ctiport1:8501:
cue_pt.(0,0)] out of service
120: 12:05:25.544 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser) [8501:cue_pt]
CiscoAddrOutOfServiceEv [#3] Cause:100 CallCtlCause:0 CiscoFeatureReason:12
121: 12:05:25.546 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.LineOpenRequest {
deviceName = cue_ctiport1
lineName = 8501
}
122: 12:05:25.582 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.LineOpenResponse {
134: 12:05:25.670 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.LineCloseRequest {
135: 12:05:25.673 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.LineCloseResponse {
138: 12:05:25.674 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceCloseRequest {
139: 12:05:25.681 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.DeviceCloseResponse {
141: 12:05:25.683 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceRegisterDeviceRequest {
deviceName = cue_ctiport1
142: 12:05:25.687 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.DeviceRegisterDeviceResponse {
result = 0
name = cue_ctiport1
allowsRegistration = true
}
143: 12:05:25.687 CST %JTAPI-CTI-7-UNK.(P1-tacjtapiuser) DeviceMap: opening
device "cue_ctiport1"
150: 12:05:25.688 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.LineOpenRequest {
deviceName = cue_ctiport1
lineName = 8501
151: 12:05:25.690 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.LineOpenResponse {
152: 12:05:25.691 CST %JTAPI-JTAPIIMPL-7-UNK:cue_ctiport1: Lines opened
153: 12:05:25.739 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.DeviceRegisteredEvent {
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
allowsRegistration = true
controllable = true
}
}
156: 12:05:25.739 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) Received
DeviceRegisteredEvent
160: 12:05:25.740 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.DeviceInServiceEvent {
162: 12:05:25.741 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.LineInServiceEvent {
}
```

Llamadas básico remitidas al voicemail

(Salidas de los registros CiscoJtapi1/de Cisco Jtapi2)

Nuevos llamada y cambio de dirección al puerto disponible



Nuevos llamada y cambio de dirección al puerto disponible

```
12:46:00.396 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.NewCallEvent {
deviceName = cue_vm_ctirp
callLegID = 25626132
callID = 9040
callingParty = 3001
calledParty = 8000
callingPartyName = Ext 3001 - Phone
callingPartyDeviceName = SEP0023331C29EC
unModifiedCalledParty = 8000
unModifiedOriginalCalledParty = 8000
unModifiedLastRedirectingParty =
}
12:46:00.400 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.CallStateChangedEvent {
callLegID = 25626132
state = 2
reason = 1
}
```

```

12:46:00.402 CST %JTAPI-CTI-7-UNK.(P1-tacjtapiuser){Line:cue_vm_ctirp:8000:
cue_pt.(1,28)
|Call:[GCID=
(9040/1),CID=25626132]} NewCall [ state=OFFERING auxData=1 destCM=1 destType=
IN_CLUSTER unModifiedCg=3001
unModifiedCd=8000 unModifiedOriginalCd=8000 unModifiedLastRedirected= calling=3001
callingName=Ext 3001 -
Phone called=8000 calledName= origParty=8000 origName= lastRedirected=
lastRedirectedName= origin=INBOUNDINTERNAL reason=DIRECTCALL activeTone=0
deviceName=cue_vm_ctirp bRemoteInUse=false bPrivacy=false CallSelectStatus=0
CallingPartyPI=True CallingPartyDisplayNamePI=True CalledPartyPI=True
CalledPartyDisplayNamePI=True OriginalCalledPartyPI=True]
12:46:00.424 CST %JTAPI-JTAPIIMPL-7-UNK:((P1-tacjtapiuser) GCID=(1,9040)->ACTIVE)
Initializing to OFFERING for 8000:cue_pt Cause=CAUSE_NORMAL Reason= 1
12:46:00.424 CST %JTAPI-JTAPI-7-UNK:[3001:cue_pt/(P1-tacjtapiuser) GCID=
(1,9040)->ACTIVE]->IDLE]creating external connection for 3001:cue_pt
12:46:00.424 CST %JTAPI-JTAPI-7-UNK:{ CcnCall=Call:[GCID=(9040/1),CID=25626132]
Connection=[3001:cue_pt/(P1-tacjtapiuser) GCID=(1,9040)->ACTIVE]->IDLE: creating
new Connection for CCNCall }
12:46:00.425 CST %JTAPI-JTAPI-7-UNK:[9040/1]CallImpl.deliverEvents(): for all
1 observers
12:46:00.430 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[SS_TEL_ROUTE_CALL_EV][[
8000:cue_pt/(P1-tacjtapiuser) GCID=(1,9040)->ACTIVE]->OFFERED]Request: redirect
(8501, REDIRECT_NORMAL, DEFAULT_SEARCH_SPACE, CALLED_ADDRESS_UNCHANGED,
REDIRECT,8501,null,REDIRECT_WITHOUT_MODIFIED_CALLING_PARTY,1)
12:46:00.430 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
[SS_TEL_ROUTE_CALL_EV] sending: com.cisco.cti.protocol.CallRedirectRequest {
callLegID = 25626132
redirectAddress = 8501
unconditional = false
redirectReason = 0
preferredOriginalCalledParty = 8501
}

```

Nueva llamada al puerto CTI

```

12:46:00.460 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.NewCallEvent {
deviceName = cue_ctiport1
callLegID = 25626133
callID = 9040
callingParty = 3001
calledParty = 8501
originalCalledParty = 8000
reason = 6
lastRedirectingParty = 8000
callingPartyDeviceName = SEP0023331C29EC
}
12:46:00.463 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.CallStateChangedEvent {
callLegID = 25626133
state = 2
}
12:46:00.464 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.CallRedirectResponse {
result = 0
}
12:46:00.468 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.CallStateChangedEvent {
callLegID = 25626132
state = 1
farEndpointSpecified = true

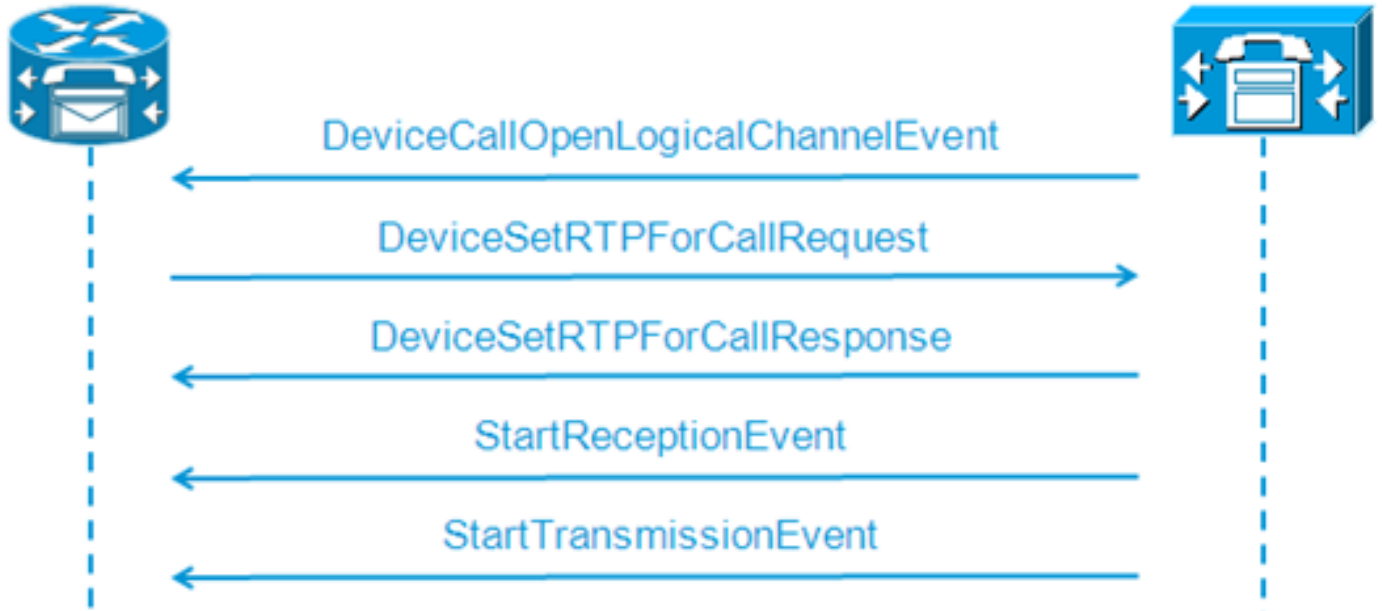
```

```
fwdDestinationAddress =
reason = 68501
callingParty = 3001
callingPartyName = Ext 3001 - Phone
calledParty = 8000 }
12:46:00.481 %JTAPI-JTAPIIMPL-7-UNK: {(P1-tacjtapiuser) GCID=(1,9040)->ACTIVE}
Initializing to OFFERING for 8501:cue_pt Cause=CAUSE_REDIRECTED Reason= 6
12:46:00.481 %JTAPI-JTAPIIMPL-7-UNK: {(P1-tacjtapiuser) GCID=(1,9040)->ACTIVE}
Received a redirected call -- lastRedAddress is 8000
12:46:00.487 %JTAPI-CTI-7-UNK.(P1-tacjtapiuser){Line:cue_ctiport1:8501:cue_pt.
(1,24)|Call:[GCID=(9040/1),CID=25626133]} CallStateChanged [ state=OFFERING
cause=NOERROR]
12:46:00.489 %JTAPI-CTI-7-UNK.(P1-tacjtapiuser){Line:cue_vm_ctirp:8000:cue_pt.
(1,28)|Call:[GCID=(9040/1),CID=25626132]} CallStateChanged [ state=IDLE cause=
NOERROR destType=IN_CLUSTER destCM=1 fwdDestination=8501]
```

El puerto CTI valida la llamada reorientada

```
12:46:00.490 %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[SS_TEL_CALL_CONN_OFFERED:8501]
[[8501:cue_pt/(P1-tacjtapiuser) GCID=(1,9040)->ACTIVE]->OFFERED]Request: accept()
12:46:00.491 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_CALL_
CONN_OFFERED:8501] sending: com.cisco.cti.protocol.CallAcceptRequest {
callLegID = 25626133
}
12:46:00.495 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Response:
com.cisco.cti.protocol.CallAcceptResponse {
result = 0
}
12:46:00.498 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.CallStateChangedEvent {
callLegID = 25626133
state = 3
12:46:00.499 %JTAPI-CTI-7-UNK.(P1-tacjtapiuser){Line:cue_ctiport1:8501:cue_pt.
(1,24)|Call:[GCID=(9040/1),CID=25626133]} CallStateChanged [ state=ACCEPTED
cause=NOERROR]
12:46:00.502 %JTAPI-JTAPIIMPL-7-UNK.(P1-tacjtapiuser) Terminal "cue_ctiport1"
in service
12:46:00.503 %JTAPI-JTAPIIMPL-7-UNK: {(P1-tacjtapiuser) GCID=(1,9040)->ACTIVE}
Handling
External STATE_RINGBACK for 3001:cue_pt
12:46:00.517 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
[ENG_TASK:0x98bca5a08_voicebrowser.aef] sending:
com.cisco.cti.protocol.CallAnswerRequest {
callLegID = 25626133
}
12:46:00.522 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Response:
com.cisco.cti.protocol.CallAnswerResponse {
result = 0
}
12:46:00.530 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.CallStateChangedEvent {
callLegID = 25626133
state = 8
```

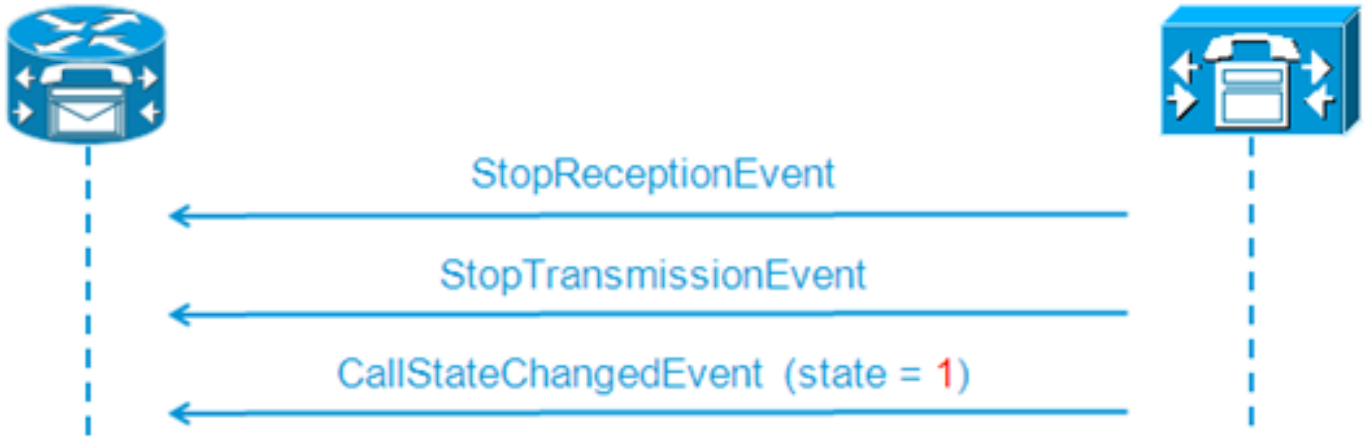
Negociación de medios



```

12:46:00.531 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.DeviceCallOpenLogicalChannelEvent {
callLegID = 25626133
compressionType = 4
}
12:46:00.531 %JTAPI-CTI-7-UNK.(P1-tacjtapiuser){Line:cue_ctiport1:8501:
cue_pt.(1,24)|Call:[GCID=(9040/1),CID=25626133]} CallStateChanged
[ state=CONNECTED cause=NOERROR]
12:46:00.537 %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[SS_TEL_OPEN_LOGICAL_CHANNEL:
8501][cue_ctiport1]
Request: setRTPParams(CiscoRTPParams192.168.105.224/16384)
12:46:00.537 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_OPEN_
LOGICAL_CHANNEL:8501] sending:
com.cisco.cti.protocol.DeviceSetRTPForCallRequest {
callLegID = 25626133
ipAddress = -529946432
rtpPortNumber = 16384
}
12:46:00.540 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Response:
com.cisco.cti.protocol.DeviceSetRTPForCallResponse {
result = 0
}
12:46:00.591 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.StartReceptionEvent {
callLegID = 25626133
ipAddr = -529946432
rtpPortNumber = 16384
compressionType = 4
}
12:46:00.596 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.StartTransmissionEvent {
callLegID = 25626133
ipAddr = -1167415104
rtpPortNumber = 22668
compressionType = 4
}
  
```

Desconexión de la llamada



```

12:46:09.438 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.StopReceptionEvent {
callLegID = 25626133
}
12:46:09.438 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.StopTransmissionEvent {
callLegID = 25626133
}
12:46:09.441 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.CallStateChangedEvent {
callLegID = 25626133
state = 1
cause = 16
12:46:09.443 %JTAPI-CTI-7-UNK.(P1-tacjtapiuser){Line:cue_ctiport1:8501:
cue_pt.(1,24)|Call:[GCID=(9040/1),CID=25626133]} CallStateChanged
[ state=IDLE cause=NORMALCALLCLEARING]

```

Señalización con./desc. del MWI

La SEÑAL enciende la lámpara MWI para la línea 3001

```

12:46:02.714 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[Thread-88][8501:cue_pt]
Request:
setMessageWaiting ( 3001,true )
12:46:02.714 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [Thread-88]
sending: com.cisco.cti.protocol.LineSetMessageWaitingRequest {
sequenceNumber = 57
lineName = 3001
lampMode = 2
}
12:46:02.718 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.LineSetMessageWaitingResponse {
sequenceNumber = 57
result = 0
}

```

Marcado '3' del número DTMF para borrar el mensaje del buzón

```

12:55:52.145 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.DtmfEvent {
eventSequence = 70
callLegID = 25626160
digit = 3
}

```

```
12:55:52.145 CST %JTAPI-CTIIMPL-7-UNK.(P1-192.168.100.10) EventThread handling
event com.cisco.cti.protocol.DtmfEvent[70]
12:55:52.146 CST %JTAPI-CTI-7-UNK.(){Line:cue_ctiport1:8501:cue_pt.(1,64)|Call:
[GCID=(9047/1),CID=25626160]}
DTMF [digit=3]
```

La SEÑAL apaga la lámpara MWI para la línea 3001

```
12:55:52.209 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[Thread-86][8501:cue_pt]
Request: setMessageWaiting ( 3001,false )
12:55:52.209 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [Thread-86] sending:
com.cisco.cti.protocol.LineSetMessageWaitingRequest {
sequenceNumber = 62
lineName = 3001
lampMode = 1
}
12:55:52.212 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Response:
com.cisco.cti.protocol.LineSetMessageWaitingResponse {
sequenceNumber = 62
result = 0
}
```

Registros en tiempo real del CCN

(Salidas de los registros en tiempo real del CCN)

Éste es cómo aparece la misma llamada del ejemplo anterior adjunto cuando los registros en tiempo real del CCN se recogen en lugar de otro.

Configuración de la llamada

```
12:46:00.425 ACCN TELS 0 assigned STANDARD-worker-8
12:46:00.425 ACCN TELS 0 Route Connection=[8000:cue_pt/(P1-tacjtapiuser) GCID=
(1,9040)->ACTIVE]->OFFERED, reason=1...
12:46:00.426 ACCN TELS 0 Call.received() JTAPICallContact[id=7,type=Cisco JTAPI
Call,implId=9040/1,active=true,state=CALL_RECEIVED,inbound=true...
12:46:00.429 ACCN TELS 0 Route Connection: [8000:cue_pt/(P1-tacjtapiuser)
GCID=(1,9040)->ACTIVE]->OFFERED, CTI Port selected: TP[id=0,implId=8501,
state=IN_USE]
12:46:00.429 ACCN TELS 0 RouteCallObserver.callChangedEvent: redirecting to
8501, css=default
12:46:00.480 ACCN TELS 0 Call.associated() JTAPICallContact[id=7,type=Cisco
JTAPI Call,implId=9040/1,active=true,state=CALL_RECEIVED,
12:46:00.480 ACCN TELS 0 Route Connection: [8000:cue_pt/(P1-tacjtapiuser)
GCID=(1,9040)->ACTIVE]->OFFERED has 1 current sessions active.
12:46:00.484 ACCN TELS 0 CallID: 7, MediaID: 9040/1 CallCtlConnOfferedEv
received for CTI Port: 8501, lastRedirectedAddress: 8000
12:46:00.490 ACCN TELS 0 assigned STANDARD-worker-9
12:46:00.490 ACCN TELS 0 Route TR[num=8000], event=(P1-tacjtapiuser) 9040/1
CallCtlConnDisconnectedEv 8000:cue_pt [#108] Cause:100 CallCtlCause:0
CiscoCause:0 CiscoFeatureReason:6, cause=CAUSE_NORMAL[100],
meta=META_CALL_REMOVING_PARTY[131]
12:46:00.499 ACCN TELS 0 CallID: 7, MediaID: 9040/1 Accepting call for CTI
Route Point: 8000 on CTI Port: 8501, ciscoCause=31
12:46:00.501 ACCN TELS 0 Call.accepted() JTAPICallContact[id=7,type=Cisco
JTAPI Call,implId=9040/1,active=true,state=CALL_RECEIVED...
12:46:00.501 ACCN TELS 0 CallID:7 MediaId:9040/1, TerminalConnection to
Terminal: cue_ctiport1 is RINGING, [8501:cue_pt/(P1-tacjtapiuser)
```

```
GCID=(1,9040)->ACTIVE]->ALERTING
12:46:00.504 ACCN TELS 0 CallID:7 MediaId:9040/1 com.cisco.jtapi.
CiscoTermInServiceEvImpl received
12:46:00.504 ACCN TELS 0 TR[num=8000] Get TriggerMap[] return:
{secondaryDialogGroup=0, primaryDialogGroup=0}
12:46:00.513 ACCN TELS 0 Call.attributed() JTAPICallContact[id=7,type=Cisco
JTAPI Call,implId=9040/1,active=true,state=CALL_RECEIVED,...
12:46:00.513 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008 associated
with Task ID: 41000000008
12:46:00.533 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008,
TerminalConnection to Terminal:cue_ctiport1 is ACTIVE
12:46:00.534 ACCN TELS 0 Call.answered() JTAPICallContact[id=7,type=
Cisco JTAPI Call,implId=9040/1,active=true,state=CALL_ANSWERED,...
12:46:00.536 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008
com.cisco.jtapi.CiscoMediaOpenLogicalChannelEvImpl received
12:46:00.593 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008
com.cisco.jtapi.CiscoRTPInputStartedEvImpl received
12:46:00.597 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008
com.cisco.jtapi.CiscoRTPOutputStartedEvImpl received
```

Desconexión de la llamada

```
12:46:09.442 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008
com.cisco.jtapi.CiscoRTPInputStoppedEvImpl received
12:46:09.443 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008
com.cisco.jtapi.CiscoRTPOutputStoppedEvImpl received
12:46:09.447 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008
gets TermConnDroppedEv, meta code:132, cause code:100
12:46:09.447 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008,
TerminalConnection to Terminal: cue_ctiport1 is DROPPED, 9040/1
12:46:09.448 ACCN TELS 0 CallID:7 MediaId:9040/1 is removed from call session
mapping in Session[id=0x60db88402,parent=null,active=true,state=SESSION_IN_USE,
time=1354733160426], result:true
12:46:09.466 ACCN TELS 0 Call.abandoned() JTAPICallContact[id=7,type=Cisco
JTAPI Call,implId=9040/1,active=false,state=CALL_DISCONNECTED,...
12:46:09.466 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008, released TP
[type=Cisco CTI Port,id=0,implId=8501,active=false,state=IDLE] from 8000, and
releasing udpPort 16384
12:46:09.467 ACCN TELS 0 CallID:7 MediaId:9040/1 Task:41000000008
com.cisco.jtapi.TermObservationEndedEvImpl received
```

Resolver problemas los casos prácticos

Problemas de conectividad

En este escenario, los puertos y los activadores de la SEÑAL no se registran con el CUCM debido de la falta de Conectividad entre la SEÑAL y el CUCM.

```
CUE# show log name CiscoJtapi1.log tail
!! or show log name CiscoJtapi2.log tail
456: 13:20:28.331 CDT %JTAPI-MISC-7-UNK.(P20-) started preloading classes
457: 13:20:28.331 CDT %JTAPI-MISC-7-UNK.(P20-) finished preloading classes
461: 13:20:28.331 CDT %JTAPI-CTI-7-UNK.(P20-) EventThread queue size
threshold is 25
462: 13:20:28.331 CDT %JTAPI-CTI-7-UNK.(P20-) Provider retry interval is set
to 30 seconds
463: 13:20:28.331 CDT %JTAPI-CTI-7-UNK.(P20-) Client desired server heartbeat
time is set to 30 seconds
```

```

464: 13:20:28.331 CDT %JTAPI-CTI-7-UNK.(P20-) CTI request timeout is is set to
30 seconds
465: 13:20:28.331 CDT %JTAPI-CTI-7-UNK.(P20-) Provider open request timeout
is set to 200 seconds
467: 13:20:28.331 CDT %JTAPI-CTI-7-UNK.(P20-) Provider Reconnect attempts is
set to 0
468: 13:20:28.331 CDT %JTAPI-CTI-7-UNK.(P20-) JAVA Socket Connect Timeout is
set to 15 seconds
469: 13:20:28.332 CDT %JTAPI-CTIIMPL-7-UNK.(P20-) Provider.info(CCMEncryption:
:encryptPassword was successful)
471: 13:20:28.334 CDT %JTAPI-JTAPIIMPL-7-UNK.ProviderImpl(): calling
jtapiProperties.getSecurityPropertyForInstance()
472: 13:20:28.334 CDT %JTAPI-JTAPIIMPL-7-UNK.(P20-tacjtapiuser ) TraceModule:
JTAPI version Cisco Jtapi version 7.0(1.1000)-1 Release
473: 13:20:28.334 CDT %JTAPI-JTAPIIMPL-7-UNK.(P20-tacjtapiuser ) Route Select
Timeout is 5000 msec
474: 13:20:28.335 CDT %JTAPI-JTAPIIMPL-7-UNK.(P20-tacjtapiuser ) Jtapi post
condition timeout is set to 15 seconds
476: 13:20:28.335 CDT %JTAPI-CTIIMPL-7-UNK.(P20-tacjtapiuser ) Opening server
"192.168.100.10" login "tacjtapiuser "
477: 13:20:28.335 CDT %JTAPI-CTIIMPL-7-UNK.(P20-tacjtapiuser ) ProviderID =
P20-tacjtapiuser 478: 13:20:28.337 CDT %JTAPI-CTIIMPL-7-UNK.(P20-tacjtapiuser )
Trying to create normal socket connection to 192.168.100.10
479: 13:20:38.338 CDT %JTAPI-JTAPI-7-UNK:[DefaultJtapiPeer]PlatformExceptionImpl
caught:
Unable to create provider --

```

Nota: Los segundos del grupo fecha/hora van a partir la 13:20:28 a 13:20:38; por lo tanto, podemos decir la SEÑAL no podíamos abrir el socket TCP por 10 segundos antes del acuse de recibo de la incapacidad para crear el proveedor.

Problemas de la autenticación

En este escenario, los puertos y los activadores de la SEÑAL no pueden registrarse con el CUCM porque las contraseñas configuradas entre la SEÑAL y el CUCM no hacen juego.

Registro del CCN

```

CUE# show trace buffer tail
Press CTRL-C to exit...
140053.173 ACCN TELS 0 TAPIPortGroup Leaving getActiveCCM(), retvalnull
140123.184 ACCN TELS 0 TAPIPortGroup Enter getActiveCCM()
140123.184 ACCN TELS 0 TAPIPortGroup getActiveCCM() subsystemstate3
140123.184 ACCN TELS 0 TAPIPortGroup getActiveCCM() subsystemJTAPI is not
in service or partial service
140123.184 ACCN TELS 0 TAPIPortGroup Leaving getActiveCCM(), retvalnull

```

atrace.log

```

14:12:18.681 ACCN TELS 0 JTAPI_PROVIDER_EVENT:JTAPI Provider state is changed:
JTAPI provider name=192.168.100.10,Event=ProvShutdownEv received
14:12:18.682 ACCN TELS 0 SS_LOGIN:JTAPI Login String: Module=JTAPI Subsystem,
JTAPI login string=192.168.100.10;login=tacjtapiuser ;passwd=****;appinfo=
Cisco IP IVR
14:12:18.682 ACCN TELS 0 PROVIDER_CLEANUP:Cleaning up JTAPI provider:
Module=JTAPI Subsystem,JTAPI provider name=192.168.100.10
14:12:18.682 ACCN TELS 0 TAPIPortGroup 1 getNumPorts() for Cisco CTI Port = 2
14:12:18.682 ACCN TELS 0 TPG[id=1,state=PARTIAL_SERVICE] removeRoute() -
TR[num=9500]
14:12:18.682 ACCN TELS 0 TPG[id=1,state=PARTIAL_SERVICE] removeRoute() -
TR[num=9000]
14:12:18.682 ACCN TELS 0 MwiAddress.clear: [addrStr=, addr=null, inService=false,

```

```

isRegistered=false]
14:12:18.682 ACCN TELS 0 MwiAddress.unregister: [addrStr=, addr=null,
inService=false, isRegistered=false]
14:12:18.682 ACCN TELS 0 TAPIPortGroup 1 getNumPorts() for Cisco CTI Port = 0
14:12:18.682 ACCN TELS 0 Number of CTI ports = 0
14:12:18.682 ACCN TELS 0 calculateSubsystemState
14:12:18.682 ACCN TELS 0 TPG[id=1,state=PARTIAL_SERVICE] Triggers: ISV = 0,
OOS = 0, PARTIAL = 0
14:12:18.682 ACCN TELS 0 TAPIPortGroup 1 getNumPorts() for Cisco CTI Port = 0
14:12:18.682 ACCN TELS 0 calculateSubsystemState -> Groups: ISV = 0, OOS = 0,
PARTIAL/OTHERS = 1
14:12:18.682 ACCN TELS 0 calculateSubsystemState -> Triggers: ENABLED = 0,
DISABLED = 2, CONFIG ERR = 0
14:12:18.682 ACCN TELS 0 calculateSubsystemState -> subsystem partial in
service, unchanged cause:
A number of route points are OOS - TR[num=9000], TR[num=9500]; A number of
CTI ports are OOS - TPG[id=1,state=PARTIAL_SERVICE].Ports[9590]
14:12:18.689 ACCN TELS 0 SS_PARTIAL_SERVICE:JTAPI subsystem in partial service:
Failure reason=A number of route points are OOS - TR[num=9000], TR[num=9500];
A number of CTI ports are OOS - TPG[id=1,state=PARTIAL_SERVICE].Ports[9590]
14:12:18.689 ACCN TELS 0 GET_NEW_PROVIDER:Attempt to get JTAPI provider
14:12:18.693 ACCN TELS 0 Calling updateJTAPIPackage: 192.168.100.10
Module=JTAPI_PROVIDER_INIT,Exception=com.cisco.jtapi.PlatformExceptionImpl:
Unable to create provider
-- bad login or password.
14:12:18.828 ACCN TELS 0 EXCEPTION:com.cisco.jtapi.PlatformExceptionImpl:
Unable to create provider
-- bad login or password.

```

CiscoJtapi1.log/CiscoJtapi2.log

```

6318: 14:22:26.653 CDT %JTAPI-CTIIMPL-7-UNK.(P62-tacjtapiuser ) Trying to
create normal socket connection to 192.168.100.10
6319: 14:22:26.654 CDT %JTAPI-CTIIMPL-7-UNK.(P62-tacjtapiuser ) connected
6321: 14:22:26.654 CDT %JTAPI-PROTOCOL-7-UNK.(P62-192.168.100.10)
[SS_TEL_REINIT] sending: com.cisco.cti.protocol.ProviderOpenRequest {
provider = 192.168.100.10
qbcClientVersion = Cisco JTAPI 7.0(1.1000)-1 Release
login = com.cisco.cti.protocol.UnicodeString {
unicodedisplayName = tacjtapiuser
}
filter = com.cisco.cti.protocol.ProviderEventFilter {
deviceRegistered = true
deviceUnregistered = true
desiredServerHeartbeatTime = 30
}
6331: 14:22:26.781 CDT %JTAPI-PROTOCOL-7-UNK(P62-192.168.100.10)
received Event: com.cisco.cti.protocol.ProviderOpenCompletedEvent {
eventSequence = 251
reason = -1932787616
providerInfoString = 7.1.2.21900-5
failureDescription = Directory login failed - authentication failed.
providerId = 16777255
}
6333: 14:22:26.781 CDT %JTAPI-PROTOCOL-7-UNK.(P62-192.168.100.10)
received Event: com.cisco.cti.protocol.ProviderClosedEvent {
eventSequence = 252
reason = 4
}
6338: 14:22:26.781 CDT %JTAPI-PROTOCOL-7-UNK.(P62-192.168.100.10)
Received ProviderClosedEvent
6339: 14:22:26.781 CDT %JTAPI-PROTOCOL-7-UNK.(P62-192.168.100.10)
received Event: com.cisco.cti.protocol.ProviderOutOfServiceEvent {
eventSequence = 253

```

```
PROVIDER_OUT_OF_SERVICE_EVENT = 200
}
6343: 14:22:26.782 CDT %JTAPI-JTAPI-7-UNK:[DefaultJtapiPeer]
PlatformExceptionImpl caught: Unable to create provider -- bad login or password.
6344: 14:22:26.881 CDT %JTAPI-CTIIMPL-7-UNK.(P62-192.168.100.10) ReceiveThread:
caught java.net.SocketException: The socket was closed
```

Usuario CTI-no habilitado

En este escenario, los puertos y los activadores de la SEÑAL no pueden registrarse con el CUCM porque no han agregado al usuario de la aplicación JTAPI al grupo **habilitado CTI estándar** del permiso en el lado CUCM. Por lo tanto, incluso cuando los credenciales de usuario autentican por consiguiente, el usuario de JTAPI, **tacjtapiuser** en este caso, no puede controlar ninguna dispositivos con el CTI y el JTAPI.

CiscoJtapi1.log/CiscoJtapi2.log

```
11590:14:41:08.768 CDT %JTAPI-PROTOCOL-7-UNK.(P115-192.168.100.10)
[ProviderRetryThread] sending:
com.cisco.cti.protocol.ProviderOpenRequest {
provider = 192.168.100.10
qbeClientVersion = Cisco JTAPI 7.0(1.1000)-1 Release
login = com.cisco.cti.protocol.UnicodeString {
unicodedisplayName = tacjtapiuser
}
applicationID = Cisco IP IVR
desiredServerHeartbeatTime = 30
requestTimer = 0
cmAssignedApplicationID = 0
pluginName = CiscoJTAPI
}
11593:14:41:08.770 CDT %JTAPI-PROTOCOL-7-UNK.(P115-192.168.100.10)
received Response: com.cisco.cti.protocol.ProviderOpenResponse {
sequenceNumber = 117
result = 0
providerInfoString = 7.1.2.21900-5
clientHeartbeat = 30
serverHeartbeat = 30
requestTimer = 5
pluginVersion = 7.1.2.10000-5
pluginLocation = http://192.168.100.10/plugins/
providerId = 16777220
}
11600: 14:41:08.899 CDT %JTAPI-PROTOCOL-7-UNK.(P115-192.168.100.10)
received Event: com.cisco.cti.protocol.ProviderOpenCompletedEvent {
eventSequence = 461
reason = -1932787617
sequenceNumber = 117
failureDescription = Directory login failed - User not present in Standard
CTI Users group.
providerId = 16777220
}
11608:14:41:08.900 CDT %JTAPI-PROTOCOL-7-UNK.(P115-192.168.100.10)
received Event:
com.cisco.cti.protocol.ProviderOutOfServiceEvent {
eventSequence = 463
PROVIDER_OUT_OF_SERVICE_EVENT = 200
}
```

El servicio del Administrador CTI. CUCM está abajo

En este escenario, los puertos y los activadores de la SEÑAL no pueden registrarse porque el servicio del Administrador CTI. CUCM está abajo o en de un estatus anormal. Recibe un error rechazado "conexión" para el intento de conexión de la SEÑAL al puerto TCP 2748 JTAPI.

```
18956: 16:25:45.516 CDT %JTAPI-CTIIMPL-7-UNK.(P200-) Provider.
info(CCMEncryption::encryptPassword was successful)
18957: 16:25:45.516 CDT %JTAPI-CTIIMPL-7-UNK.(P200-) application did
not set appinfo, creating default
18958: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.ProviderImpl(): calling
jtapiProperties.getSecurityPropertyForInstance()
18959: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.(P200-tacjtapiuser )
TraceModule: JTAPI version Cisco Jtapi version 7.0(1.1000)-1 Release
18960: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.(P200-tacjtapiuser )
Route Select Timeout is 5000 msec
18961: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.(P200-tacjtapiuser )
Jtapi post condition timeout is set
to 15 seconds
18962: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.(P200-tacjtapiuser )
IgnoreFwdDestination
set to false
18963: 16:25:45.516 CDT %JTAPI-CTIIMPL-7-UNK.(P200-tacjtapiuser )
Opening server "192.168.100.10" login "tacjtapiuser "
18964: 16:25:45.516 CDT %JTAPI-CTIIMPL-7-UNK.(P200-tacjtapiuser )
ProviderID = P200-tacjtapiuser
18965: 16:25:45.517 CDT %JTAPI-CTIIMPL-7-UNK.(P200-tacjtapiuser )
Trying to create normal socket connection to 192.168.100.10
18966: 16:25:45.518 CDT %JTAPI-JTAPI-7-UNK:[DefaultJtapiPeer]
PlatformExceptionImpl caught:
Unable to create provider -- 192.168.100.10/192.168.100.10:2748 -
Connection refused
```

Discrepancia de configuración

¿En este escenario, la SEÑAL no puede registrar el activador JTAPI con el número 9999 porque el CTI RP que debe hacer juego no se configura, o no se ha agregado al? ¿dispositivos controlables? para el usuario en el lado CUCM. La SEÑAL realiza esto después de que reciba el **GetDeviceInfoFetchResponse** del CUCM y note que no hay un dispositivo en el dominio del proveedor, que refiere a todos los dispositivos controlables de ese usuario, que hiciera juego el número del activador ha configurado localmente. La SEÑAL entonces no intenta enviar un **DeviceOpenRequest** para ese activador específico y en lugar de otro señala solamente la excepción en las trazas. La SEÑAL todavía intenta registrar todos los otros dispositivos que estén dentro del dominio del proveedor enviado por el CUCM.

```
13:27:58.864 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Response:
com.cisco.cti.protocol.GetDeviceInfoFetchResponse {
com.cisco.cti.protocol.DeviceInfo {
name = cue_vm_ctirp
}
}
13:27:58.960 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
deviceName = cue_vm_ctirp
}
13:27:58.962 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.GetLineInfoFetchRequest
13:27:58.964 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Response:
com.cisco.cti.protocol.GetLineInfoFetchResponse{
name = 8000
}
13:27:58.966 CST %JTAPI-CTI-7-UNK(P1-tacjtapiuser) DeviceMap: adding device
"cue_vm_ctirp"
```



```
13:27:59.427 CST %JTAPI-JTAPI-7-UNK: InvalidArgumentExceptionImpl caught:
Address 9999 is not in provider's domain.
```

Nota: Incluso cuando el activador 9999 se configura localmente en la SEÑAL, no es parte del dominio del proveedor recibido del CUCM, y por lo tanto, no se registra.

La SEÑAL continúa abriendo la línea 8000; ¿cuál se incluye en el proveedor? dominio s

```
13:28:00.953 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceOpenRequest {
deviceName = cue_vm_ctirp
13:28:00.979 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.LineOpenRequest {
deviceName = cue_vm_ctirp
lineName = 8000
13:28:00.983 CST %JTAPI-JTAPIIMPL-7-UNK:cue_vm_ctirp: Lines opened
13:28:00.997 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceRegisterDeviceRequest
deviceName = cue_vm_ctirp
13:28:01.000 CST %JTAPI-CTI-7-UNK.(P1-tacjtapiuser) DeviceMap: opening device
"cue_vm_ctirp"
13:28:01.001 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.LineOpenRequest {
lineName = 8000
13:28:01.012 CST %JTAPI-JTAPIIMPL-7-UNK:cue_vm_ctirp: Lines opened
13:28:01.164 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.DeviceRegisteredEvent {
13:28:01.165 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.DeviceInServiceEvent {
13:28:01.166 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.LineInServiceEvent {
13:28:01.168 CST %JTAPI-JTAPIIMPL-7-UNK.(P1-tacjtapiuser) Terminal
"cue_vm_ctirp" in service
```

Problema del ruteo de llamadas CUCM

En este escenario, el usuario con DN 3001 llama la SEÑAL para marcar su VM. La llamada se presenta al piloto VM de la SEÑAL (CTI RP) con DN 8000. La SEÑAL entonces solicita la llamada para conseguir reorientada a su puerto de los media CTI con DN 8501, pero la llamada no puede conseguir reorientada porque el CSS configurado para DN 3001 no tiene acceso al PT donde se asigna el DN del puerto CTI.

```
12:56:01.392 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.NewCallEvent {
deviceName = cue_vm_ctirp
callLegID = 25626135
callID = 9041
callingParty = 3001
calledParty = 8000
originalCalledParty state = 2
}
12:56:01.404 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
[SS_TEL_ROUTE_CALL_EV] sending: com.cisco.cti.protocol.CallRedirectRequest {
callLegID = 25626135
redirectAddress = 8501
}
12:56:01.397 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.CallStateChangedEvent {
callLegID = 25626135
state = 2
```

```

}
12:56:01.450 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.FailureResponse {
result = -1932787660
description = redirect failure
}
12:56:01.450 CST %JTAPI-JTAPI-7-UNK:[[8000:cue_pt/(P1-tacjtapiuser)
GCID=(1,9041)->ACTIVE]->OFFERED]InvalidPartyExceptionImpl caught:
Request failed because of an invalid destination.
12:56:05.456 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Event: com.cisco.cti.protocol.CallStateChangedEvent {
callLegID = 25626135
state = 1
cause = 17
}
12:56:05.456 CST %JTAPI-CTI-7-UNK.(P1-tacjtapiuser){Line:cue_vm_ctirp:
8000:cue_pt.(1,28)|Call:[GCID=(9041/1),CID=25626135]}CallStateChanged
[ state=IDLE cause=USERBUSY]
12:56:05.457 CST %JTAPI-CTI-7-UNK:{ALL EXTERNAL ADDRESSES|Call(P1-tacjtapiuser)
GCID=(1,9041)->ACTIVE} ExternalCallStateChanged
[ state=IDLE cause=17 processEvent= reason =1 ]
12:56:05.457 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser) 9041/1 ConnDisconnectedEv
3001:cue_pt [#160]
Cause:17 CallCtlCause:0 CiscoCause:17 CiscoFeatureReason:12
12:56:05.457 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[(P1-192.168.100.10)
EventThread][SEP0023331C29EC] Request: getCallingTerminal()
12:56:05.457 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser) 9041/1
CallCtlConnDisconnectedEv 3001:cue_pt [#161] Cause:17 CallCtlCause:0
CiscoCause:17 CiscoFeatureReason:12= 8000

```

Problemas de la licencia

En este escenario, la SEÑAL no puede registrar sus puertos y activadores porque las licencias para los puertos VM no se han activado. No se considera ningunas tentativas del registro en las capturas debido a la misma razón.

Resumen de atrace.log decodificado:

```

2551 11:45:17.178 LLMA LAPI 0 Llama: getMaxVmPortCount():
2547 11:45:17.178 LLMA LVMP 0 LlamaVmPortQuery: get(): maxCount
2551 11:45:17.178 LLMA LSDB 0 Llama: getMaxVmPortCount(): LlamaSysdbUser():
getInt(): Getting int /sw/apps/limitsManager/vmPort/query/maxCount returns 0
2551 11:45:17.178 LLMA LAPI 0 Llama: getMaxVmPortCount(): count: 0
2551 11:45:17.178 WFSP JTRG 0 WFSysdbNdJtapiTrg::getMaxSessions for trigger
for app: voicemail 0
2551 11:45:17.178 WFSP JTRG 0 WFSysdbNdJtapiTrg::commit warning session
value exceeded license max
2551 11:45:17.181 WFSP JTRG 0 com.cisco.aesop.sysdb.xactSysdbException:
Session value exceeds license limit
2551 11:45:19.654 LLMA LVMM 0 LlamaVmMbxQuery: get(): licenseStatus
2575 11:45:19.654 LLMA LSDB 0 Llama: showVoicemail(): LlamaSysdbUser():
getInt(): Getting int /sw/apps/limitsManager/vmMbx/query/licenseStatus returns 2
2575 11:45:19.657 LLMA LLMT 0 voicemail disabled, voicemail mailbox
activation count has been set to zero
3456 11:45:23.114 LLMA LAPI 0 Llama: getMaxPortCount():
2555 11:45:23.114 LLMA LPRT 0 LlamaPortQuery: get(): maxCount
3456 11:45:23.115 LLMA LSDB 0 Llama: getMaxPortCount(): LlamaSysdbUser():
getInt(): Getting int/sw/apps/limitsManager/port/query/maxCount returns 0
3456 11:45:23.115 LLMA LAPI 0 Llama: getMaxPortCount(): count: 0
3456 11:45:28.727 ACCN TELS 0 CueCiscoCall:getMajorVer() jtapi version=
7.0(1.1000)-1 majorVer=7
3456 11:45:28.785 ACCN TELS 0 JTAPI Login Str:
192.168.100.10;login=tacjtapiuser ;passwd=****;appinfo=Cisco IP IVR

```

```

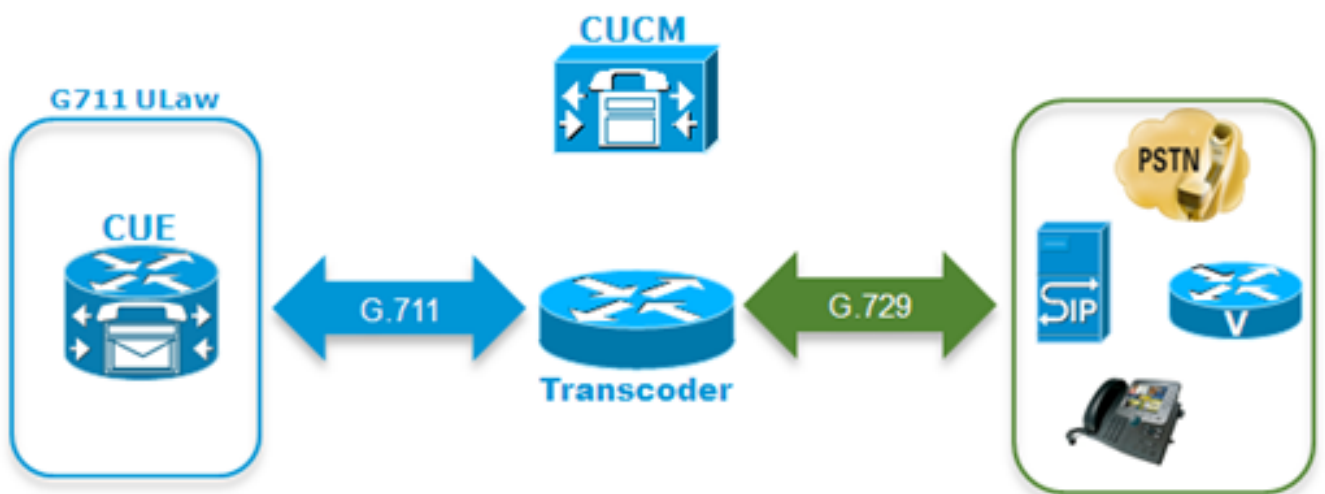
3456 11:45:28.785 ACCN TELS 0 Actual Login Str:
192.168.100.10;login=tacjtapiuser ;passwd=cisco;appinfo=Cisco IP IVR
3477 11:45:31.330 ACCN TELS 0 Got JTAPI provider: Cisco Jtapi version
7.0(1.1000)-1 Release
3621 11:45:31.338 ACCN TELS 0 JTAPI_PROVIDER_EVENT:JTAPI Provider
state is changed: JTAPI provider name=192.168.100.10,Event=
ProvOutOfServiceEv received
3621 11:45:31.352 ACCN TELS 0 JTAPI_PROVIDER_EVENT:JTAPI Provider state
is changed: JTAPI provider name=192.168.100.10,Event=ProvInServiceEv received
3621 11:45:31.353 ACCN ATJT 0 checkConnectivity:
urlString:http://192.168.100.10/CCMpluginsServer/CiscoJTAPIClient.exe
3477 11:45:34.130 ACCN TELS 0 SS_OUT_OF_SERVICE:JTAPI subsystem in
out of service: Failure reason=A number of route points are OOS; A number of
CTI ports are OOS - all ports in TPG
3751 11:45:48.558 ACCN TELS 0 TAPIPortGroup: getActiveCCM() subsystemJTAPI
is not in service or partial service

```

Mejores medidas

La SEÑAL soporta solamente el codificador-decodificador G711ulaw; por lo tanto en casi cada despliegue un transcoder se requiere para que la SEÑAL comunique con los otros dispositivos o los trunks que utilizan diverso codecs (incluye G711Alaw). Lo mismo solicita el DTMF que intertrabaja con los dispositivos que soportan solamente la en-banda DTMF donde un recurso del Media Termination Point (MTP) también se requiere. Debido a estas limitaciones, Cisco recomienda a:

- ¿Cree a una agrupación de dispositivos aislada para el uso solamente de la SEÑAL? s CTI RP y puertos CTI. En caso que más de una SEÑAL se integre con CUCM, después cree a una agrupación de dispositivos por la SEÑAL.
- Cree una región separada solamente para el RP y los puertos de la SEÑAL y aplíquela a esa agrupación de dispositivos aislada.
- Asegúrese que la región esté configurada para permitir solamente el G711 con el resto de las regiones.
- Asegúrese que una lista del grupo de los recursos del medio (MRGL) con los recursos de transcodificación disponibles esté aplicada a la agrupación de dispositivos de los puertos CTI RP y CTI de la SEÑAL de modo que éstas tengan acceso a un recurso transcodificador, cuando esté necesitado.



- Si el usuario no puede navegar a través de los menús de la Voz con los tonos DTMF, después es posible que un recurso MTP debe ser agregado al MRGL de los dispositivos de la SEÑAL.

Cree un perfil separado VM para la SEÑAL en el CUCM

Para evitar algunos problemas recientes observados con el Administrador CTI. CUCM, se recomienda para asociar todos los teléfonos al usuario de JTAPI de la SEÑAL en el lado CUCM, en vez solamente del CTI RP y puertos.

Si se desean las funciones del Survivable Remote Site Telephony (SRST):

- Asegúrese que el activador correspondiente del SORBO esté configurado para cada activador JTAPI en la SEÑAL.
- Asegúrese agreguen al dial-peers al router de rama para permitir que las llamadas sean ruteadas al módulo de la SEÑAL a través del SORBO mientras que en el modo SRST.
- Configure la máscara del Número externo de cada uno de los puntos de ruta CTI, así como la máscara para el campo CFU (delantero de la llamada desregistrado) en el CUCM para asegurar el CUCM rutea las llamadas destinado para el módulo de sucursal a través del gateway local del Public Switched Telephone Network (PSTN) cuando la Conectividad entre el CUCM y la SEÑAL se ha perdido o si se invoca el ruteo alternativo automatizado (AAR). Las Reglas de traducción adicionales se pudieron requerir para que el router de rama pueda rutear las llamadas entrantes del PSTN al módulo de la SEÑAL también.
- Si la transferencia directa al acercamiento de la configuración VM está presente en el lado CUCM, y el usuario quiere mantener estas funciones mientras que en CME-SRST, después usted debe utilizar el simulado-DN viejo con el acercamiento de la configuración del call forward all (CFA) que fue utilizado para el CME antes de que la clave suave de **TransferToVM** estuviera disponible. Refiera a la [transferencia un llamador directamente en una casilla de correo de Unity Express](#) para más información. Aquí está un ejemplo de cómo éste puede mirar. Tenga por favor presente que esto puede ser hecha solamente si se utiliza **CME-SRST** y no llamada-administrador-retraso de **SRSTwith de la** herencia. Asuma que los DN están en el 200-299 del rango. La llamada viene adentro para x201. Configure un **ephone abajo** con este comando:

```
ephone-dn 99
number 2..
call-forward all <VM Pilot>
```

En el dial-peer que señala PARA CONTAR: Utilice una regla y un perfil de la traducción saliente para eliminar hacia fuera el asterisco prefijado (el "***") y substituir el **Dialed Number Information Service de la reorientación (RDNIS)** de nuevo a la original 3 números de dígito, por ejemplo, 201, o por el número completo E.164 en caso de que el **phonenumber** fuera configurado con el lleno HIZO dentro de la SEÑAL. Asegúrese que el encabezamiento de distracción de la INVITACIÓN que se envía a la SEÑAL hace juego el **phonenumber** configurado para el usuario en el lado de la SEÑAL.

Lista de verificación para el troubleshooting del registro del puerto

1. **Marque la configuración en el lado CUCM:** ¿Habilitan y se comienzan al Administrador CTI., el CallManager, y a los servicios web administrativos XML (AXL)? ¿Los puertos y los puntos de ruta CTI se han configurado y asignaron un DN único? ¿Habilitan al usuario de JTAPI CTI, y tiene acceso AXL API? ¿El usuario de JTAPI tiene control de todos los puntos de ruta y de los puertos CTI? Es a veces una buena idea recomenzar el servicio del Administrador CTI. en todos los servidores después de que se agregue la configuración. Sin embargo, éste

podría ser afectación del servicio y su recomendable para programar una ventana de mantenimiento, porque éste afecta a todos los otros dispositivos que utilicen el CTI y el JTAPI con el CUCM, tal como Centro de contacto unificado expreso (UCCX), Asistente del administrador de IP (IPMA), Consola de Attendant, de otras compañías AA o aplicaciones ACD, y así sucesivamente.

2. **Marque la configuración en el lado de la SEÑAL:** ¿El llamada-agente se define como CUCM? ¿Se han habilitado las licencias del puerto? Las licencias de evaluación son aceptables para la configuración inicial. ¿Puede usted hacer ping el CUCM? ¿Las credenciales del usuario de JTAPI se han agregado correctamente, y se han definido los llamada-agentes? ¿Se ha recargado el módulo de modo que los cambios de configuración sean aplicados? Si el CTI RP y puerto no se importa del CUCM automáticamente, después intente agregar manualmente el puerto DN bajo el **jtapi del subsistema del ccn**, así como los activadores del **jtapi** para cada CTI-RP y recargar el módulo.

Si todos estos elementos se confirman, después su siguiente paso es obtener las trazas JTAPI en la SEÑAL y las trazas CUCM CTI para aislar posiblemente más lejos el problema.

Información Relacionada

- [Datos de la configuración y de la traza del frunce en la SEÑAL](#)
- [Resolver problemas las funciones del sistema del Cisco Unity Express](#)
- [CallManager para el ejemplo de configuración del Cisco Unity Express](#)
- [Configuración del perfil del correo de voz CUCM](#)
- [Modelos de despliegue del Cisco Unity Express](#)
- [Mejores prácticas para el Cisco Unity Express que despliega](#)
- [CUCM que transcodifica](#)
- [CUCM MTP](#)
- [Soporte Técnico y Documentación - Cisco Systems](#)

¿Era este documento útil? [Sí](#) [ningún](#)

Gracias por su feedback.

[Abra un caso de soporte](#) (requiere un [contrato de servicios con Cisco](#).)

Discusiones relacionadas de la comunidad del soporte de Cisco

[La comunidad del soporte de Cisco](#) es un foro para que usted haga y conteste a las preguntas, las sugerencias de la parte, y colabora con sus pares.

Refiera a los [convenios de los consejos técnicos de Cisco](#) para la información sobre los convenios usados en este documento.

Actualizado: De sept el 06 de 2013

ID del Documento: 116060