

CUE - Problemi JTAPI e casi di studio

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Panoramica sull'integrazione JTAPI CUE con CUCM](#)

[Esempio di flusso di chiamate di alto livello](#)

[Abilitazione e raccolta di tracce](#)

[Tracce CCN JTAPI in tempo reale](#)

[Registri di traccia CCN JTAPI](#)

[Raccogli i file di log di traccia](#)

[Informazioni obbligatorie prima di controllare i registri](#)

[Nozioni di base sulle CTI](#)

[Stati comuni di chiamata CTI](#)

[Aspetto dei registri di traccia](#)

[CTI RP e registrazione delle porte](#)

[Chiamata di base inoltrata a segreteria telefonica](#)

[Nuova chiamata e reindirizzamento alla porta disponibile](#)

[Nuova chiamata e reindirizzamento alla porta disponibile](#)

[Nuova chiamata alla porta CTI](#)

[La porta CTI accetta la chiamata reindirizzata](#)

[Negoziiazione media](#)

[Disconnessione chiamata](#)

[Segnalazione MWI On/Off](#)

[CUE accende la lampada MWI per la linea 3001](#)

[È stato composto il numero DTMF '3' per eliminare il messaggio dalla cassetta postale](#)

[CUE spegne la lampada MWI per la linea 3001](#)

[Log CCN in tempo reale](#)

[Configurazione chiamata](#)

[Disconnessione chiamata](#)

[Risoluzione dei problemi](#)

[Problemi di connettività](#)

[Utente non abilitato per CTI](#)

[Il servizio CUCM CTI Manager è inattivo](#)

[Configurazione non corrispondente](#)

[Problema di routing delle chiamate CUCM](#)

[Elenco di controllo per la risoluzione dei problemi di registrazione delle porte](#)

[Informazioni correlate](#)

Introduzione

In questo documento viene illustrato come risolvere i problemi relativi a Cisco Unity Express (CUE) Java Telephony Application Programming Interface (JTAPI). Inoltre, questo documento fornisce informazioni e comandi su come abilitare, raccogliere e visualizzare le diverse tracce e i diversi log con esempi di case study per la risoluzione dei problemi.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Conoscenze base di come configurare e utilizzare Cisco Unified Communications Manager (CUCM) tramite l'interfaccia di amministrazione Web.
- Conoscenza di base delle porte CTI (Computer Telephony Interface) e dei punti di routing (RP) in CUCM.
- Conoscenza di base dell'interfaccia della riga di comando di Cisco Unity Express.

Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Cisco Unity Express versione 3.x o successive.
- Cisco Unified Communications Manager versione 7.x o successive.

Il metodo di integrazione utilizzato si applica solo a Cisco Unity Express con Cisco Unified Communications Manager; non con Cisco Unified Communications Manager Express (CUCME).

Cisco Unity Express deve essere concesso in licenza per CUCM, non per CUCME. CUE può essere integrato con CUCM o CUCME in qualsiasi momento e concesso in licenza di conseguenza.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

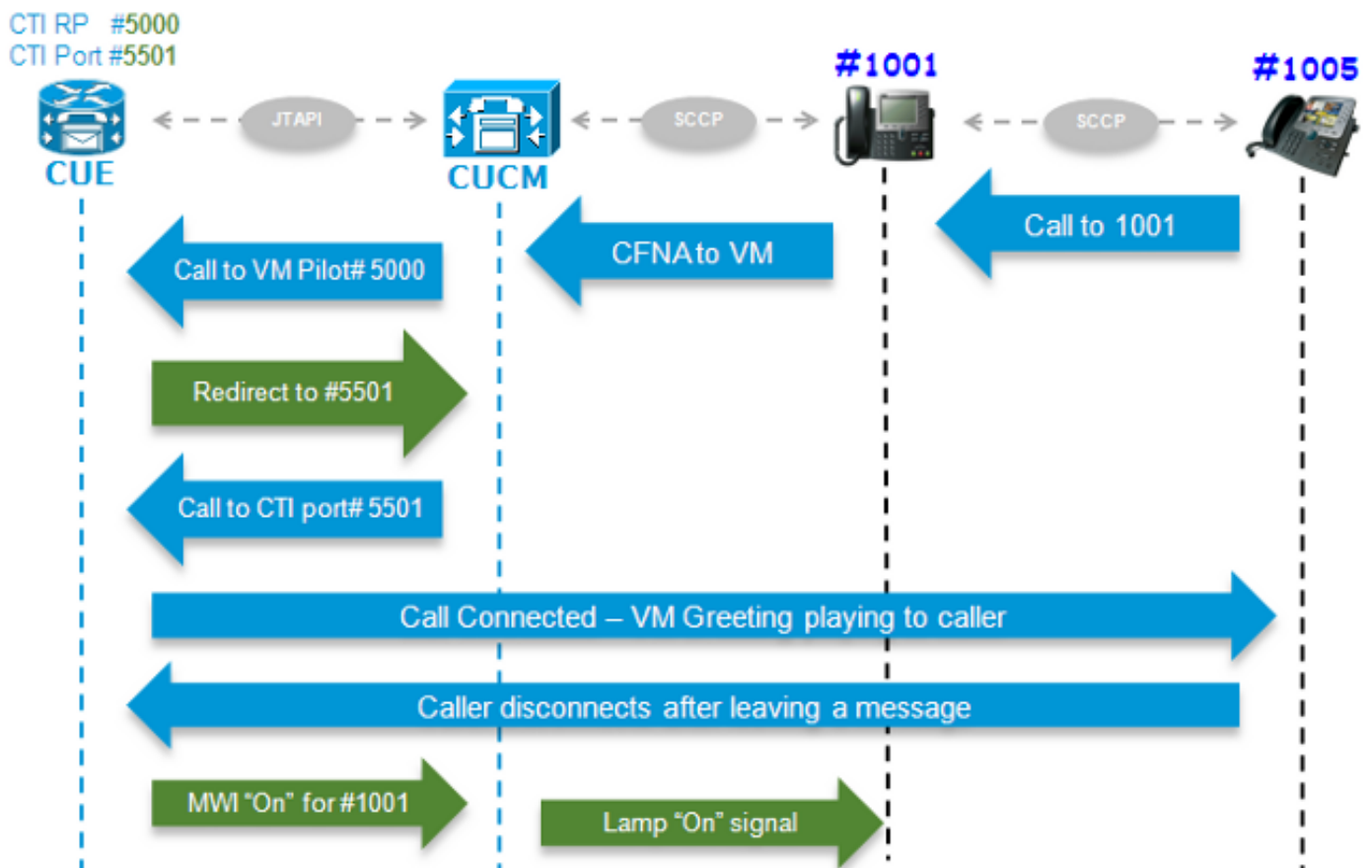
Panoramica sull'integrazione JTAPI CUE con CUCM

È possibile integrare CUE con CUCM tramite il protocollo JTAPI per la posta vocale (VM) e la funzionalità operatore automatico (AA). Questa soluzione è consigliata quando si desidera eseguire il provisioning delle funzionalità VM e/o la gestione delle chiamate ASA di base per uno o più siti di filiali con un numero ridotto di utenti registrati su un server CUCM. Questo non richiede un server Cisco Unity Voicemail completo, ma un'implementazione molto più economica. Allo stesso tempo, CUE fornisce anche opzioni di sopravvivenza per le sue filiali e il failover al Session Initiated Protocol (SIP) quando la connettività al CUCM viene persa.

La CUE è in grado di effettuare la registrazione con CUCM tramite JTAPI e controlla i punti di instradamento e le porte CTI. In questo modo è possibile controllare e gestire CUE come endpoint aggiuntivo tramite CUCM, nonché semplificare le configurazioni e le interazioni con altri

endpoint nel cluster.

Esempio di flusso di chiamate di alto livello



L'utente finale con il numero di directory (DN) 1005 chiama l'utente con DN 1001. La chiamata viene inoltrata dopo alcuni secondi se la chiamata non riceve risposta, Call Forward No Answer (CFNA), al numero di macchina virtuale configurato sul profilo utente VM 1001. Il CUCM invia quindi la chiamata al VM Pilot 5000 configurato, che punta a un CTI RP con DN 5000 controllato da CUE. Viene attivata l'applicazione CUE VM e la chiamata viene reindirizzata tramite JTAPI a una porta CTI disponibile (DN 5501) per la determinazione dei media. Il saluto audio viene riprodotto e l'utente può lasciare un messaggio o interagire con il sistema tramite i toni multi-frequenza (DTMF) dual tone. Quando il chiamante termina la chiamata, CUE segnala al CUCM di impostare la spia MWI (Message Waiting Indicator) per l'estensione 1001 su "On" tramite JTAPI. Il CUCM invia quindi il messaggio SCCP (Skinny Client Control Protocol) per accendere la luce sul telefono, oltre a mostrare l'indicazione della busta sul display in modo che l'utente 1001 sia a conoscenza della presenza di un nuovo messaggio VM nella cassetta postale.

Abilitazione e raccolta di tracce

Esistono due tipi di tracce:

- Tracce JTAPI Cisco Communications Network (CCN) in tempo reale
- Log di traccia CCN JTAPI

Tracce CCN JTAPI in tempo reale

- Tracce CCN JTAPI in tempo reale. L'attivazione di queste tracce non richiede un

ricaricamento del modulo CUE.

- L'output non è così esteso come i log di traccia CCN, ma non sono altrettanto molto informativi.

Per abilitare le tracce, immettere questi comandi:

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

Immettere questo comando per verificare che siano abilitati:

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

Per raccogliere l'output, immettere questo comando:

```
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 !!
```

Immettere **CTRL-C** per interrompere la registrazione in tempo reale sulla console.

Registri di traccia CCN JTAPI

È necessario ricaricare il modulo CUE dopo l'abilitazione dei log di traccia CCN JTAPI per popolare i log. Questi log, **messages.log** e **atrace.log**, possono essere molto dettagliati o criptici, così come molto più informativi e dettagliati. Sono disponibili quattro diversi log:

- **atrace.log**

Abilitato per impostazione predefinita sui moduli di rete (NM), ma disabilitato per impostazione predefinita per i moduli di integrazione avanzata (AIM). Immettere il comando **log trace local enable** per abilitarlo. Scrive fino a 10 MB localmente o su un server FTP. Per riavviare il log, immettere il comando **log trace local disable** o il comando **no log trace local enable**; quindi immettere il comando **log trace local enable**. Immettere il comando **clear trace file** per cancellare il file **atrace.log**. I dati devono essere decodificati dal centro di assistenza tecnica (TAC).

- **messaggi.log**

Si tratta di registri che contengono messaggi Syslog, ad esempio Info, Warning, Error e Fatal.

- **CiscoJapper1.log** e **CiscoJapper2.log**

Registrono tutti i segnali e gli eventi relativi a JTAPI. Questi registri sono molto più facili da comprendere e molto utili. Il file CiscoJtapi2.log inizia a essere compilato quando il file CiscoJtapi1.log è pieno e viceversa.

Indipendentemente dalle tracce impostate, dopo un ricaricamento vengono ripristinati i livelli di traccia predefiniti. Per modificare le impostazioni predefinite in modo che rimangano valide anche dopo il riavvio, è necessario immettere il comando **log trace boot**. Di seguito viene riportato il comando per abilitarli:

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

```
ccn trace jtapi info all
ccn trace jtapi warn all
log trace boot
reload
```

Immettere questo comando per verificare che siano abilitati:

```
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
```

Per visualizzare i registri, eseguire la procedura seguente:

1. Immettere il comando **show logs** per visualizzare un elenco dei file di log memorizzati nella CUE.
2. L'estensione del file **.prev** indica che si tratta di una copia di backup di un file di traccia precedente e non del file di registro attivo corrente.
3. È possibile estrarli in un server FTP esterno.
4. È inoltre possibile visualizzare l'output dei messaggi registrati in questi file in tempo reale dal monitor del terminale della CUE.

Raccogli i file di log di traccia

Estrarre i registri in un FTP esterno con questi comandi:

```
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/
```

Visualizzare i log sul monitor del terminale CUE con il comando **show log name <logname>**. Di seguito è riportato un esempio:

```
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>
```

Il file **atrace.log** è codificato. pertanto, non è possibile visualizzarla solo in tempo reale con il comando **show log name**.

Informazioni obbligatorie prima di controllare i registri

È necessario ottenere almeno tutti i dettagli qui descritti dalle chiamate con il problema che si sta risolvendo, in modo da poter individuare e comprendere facilmente le tracce:

- Numero chiamante
- Numero chiamato

- Numero reindirizzamento
- DN RP CTI e nome dispositivo
- Numero porta CTI e nome dispositivo
- Utente JTAPI
- Intervallo di tempo in cui hanno avuto luogo le chiamate

Nozioni di base sulle CTI

Provider: Fornitore di servizi CTI. L'applicazione stabilisce una sessione CTI **aprendo** un provider.

Utente: Le applicazioni sono associate a un utente.

Sul dispositivo bootflash o slot0:: Periferica che si registra in CUCM.

Riga: Aspetto DN su un dispositivo supportato da CTI.

ID chiamata (callLegID): Associato a una gamba di chiamata in una chiamata.

Chiamata globale (callID): Identifica tutte le parti di chiamata per una singola chiamata.

Stati comuni di chiamata CTI

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

Aspetto dei registri di traccia

Prima di poter trovare una segnalazione errata, è necessario conoscere l'aspetto che avrebbe la segnalazione in condizioni di funzionamento normali; in questa sezione vengono mostrati frammenti degli output di segnalazione che verrebbero visualizzati in scenari diversi quando funzionano normalmente.

Tieni presente che tutte le segnalazioni provenienti da questi log sono state riepilogate per mostrare solo i dettagli pertinenti, perché contengono informazioni molto dettagliate che sono piuttosto noiose e ripetitive.

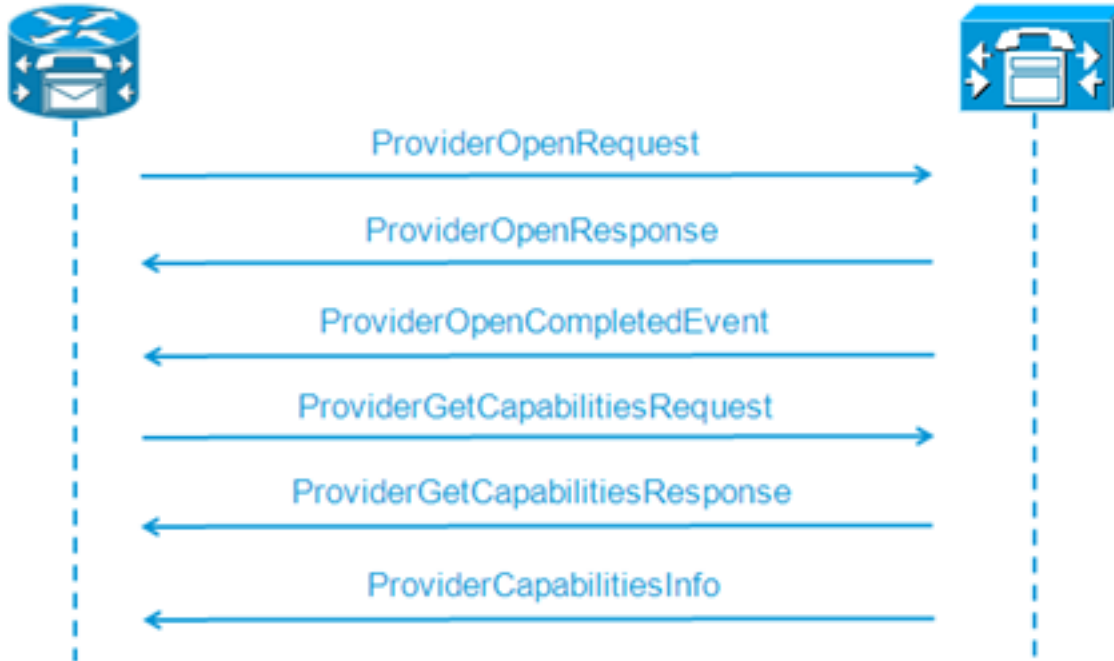
Ecco i dettagli delle configurazioni utilizzate:

```
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 e registrazione delle porte

(output dai registri Cisco Jtapi1 / Cisco Jtapi2)

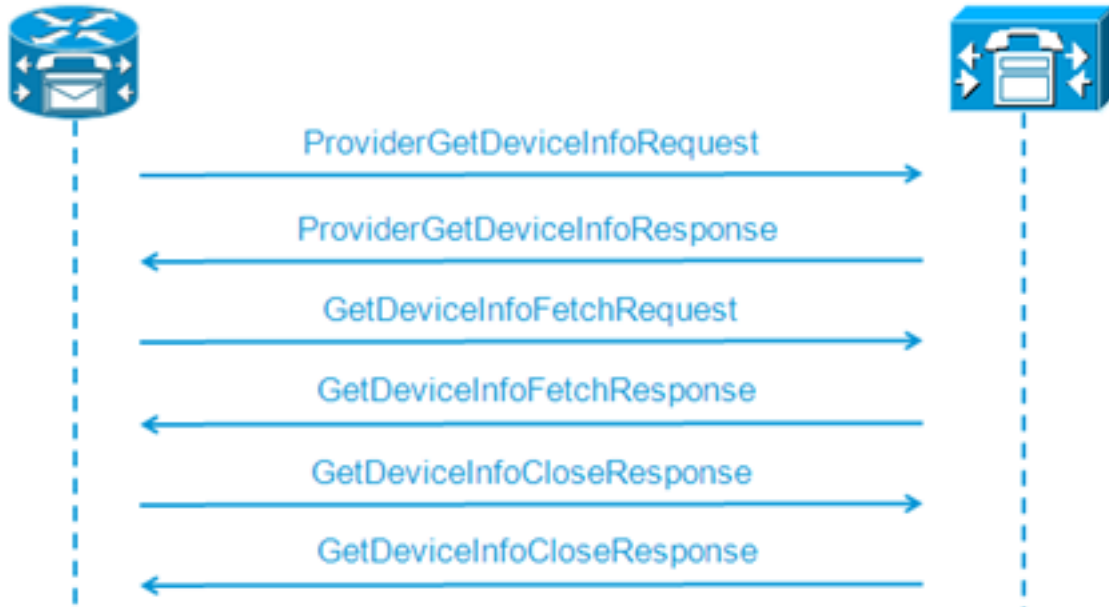
1. Apri connessione provider



```

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
}
  
```

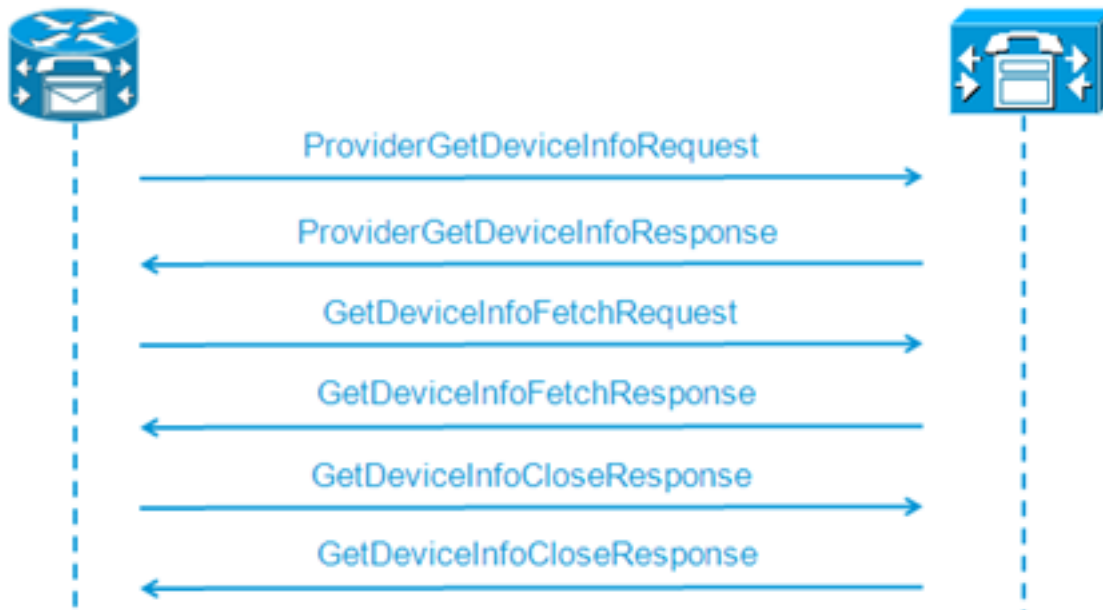
2. Query per dispositivi controllabili



```

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)
  
```


3. Informazioni sulla linea della porta 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. Ottenere informazioni sulla linea RP CTI

```
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. CUE applica la configurazione ricevuta

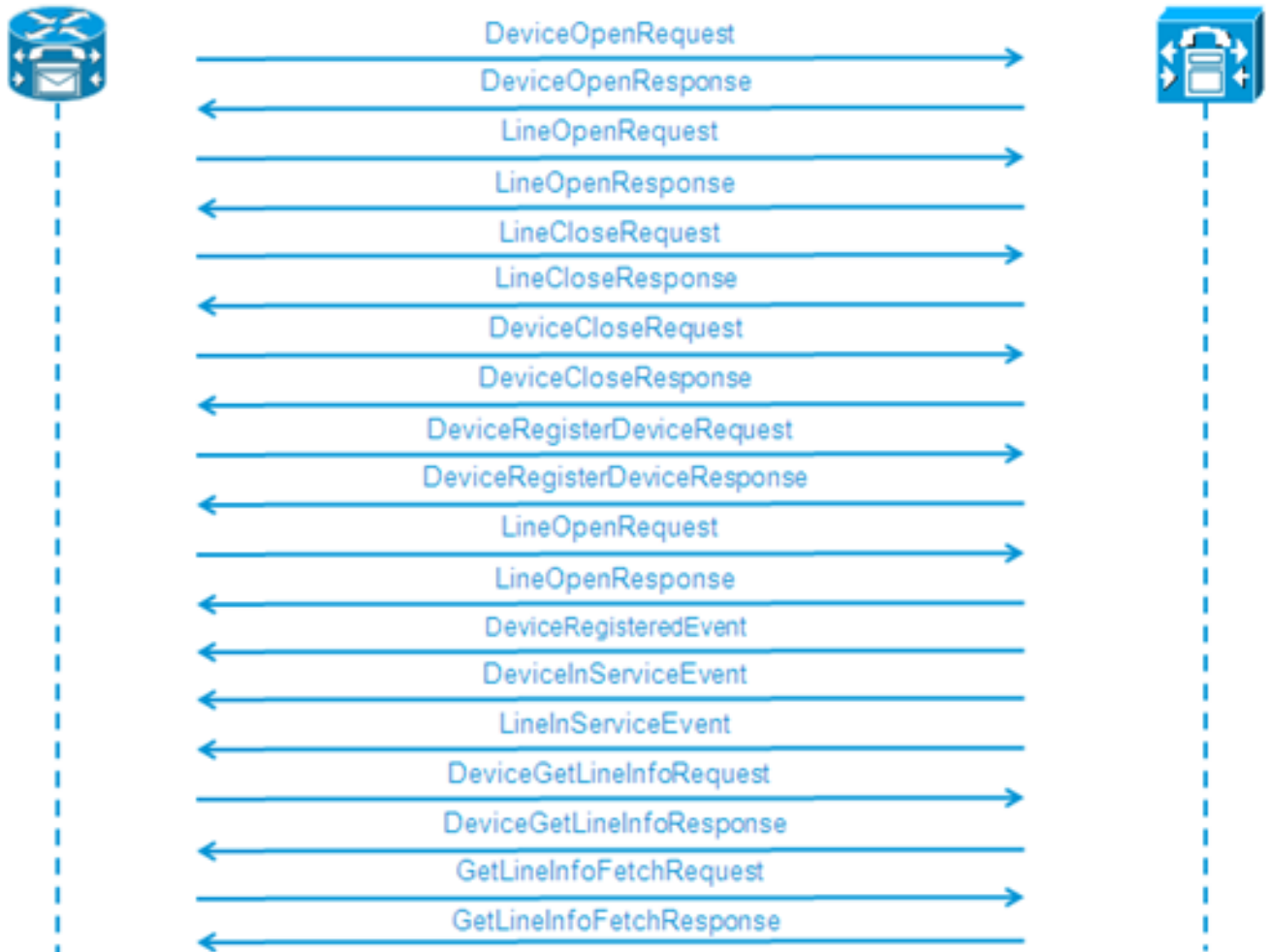
```
76: 12:05:25.064 CST %JTAPI-MISC-7-UNK.Provider 192.168.100.10
open, beginning device
initialization
77: 12:05:25.071 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[SS_TEL_INIT]
(P1-tacjtapiuser) Request: addObserver
79: 12:05:25.073 CST %JTAPI-MISC-7-UNK.ObserverThread
(com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@3d823d82):created
80:12:05:25.074 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser) ProvOutOfServiceEv [#0]
Cause:100 CallCtlCause:0 CiscoFeatureReason:12
82: 12:05:25.085 CST %JTAPI-MISC-7-
UNK.ObserverThread
(com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$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. Ottenere il controllo dei dispositivi e delle linee CTI



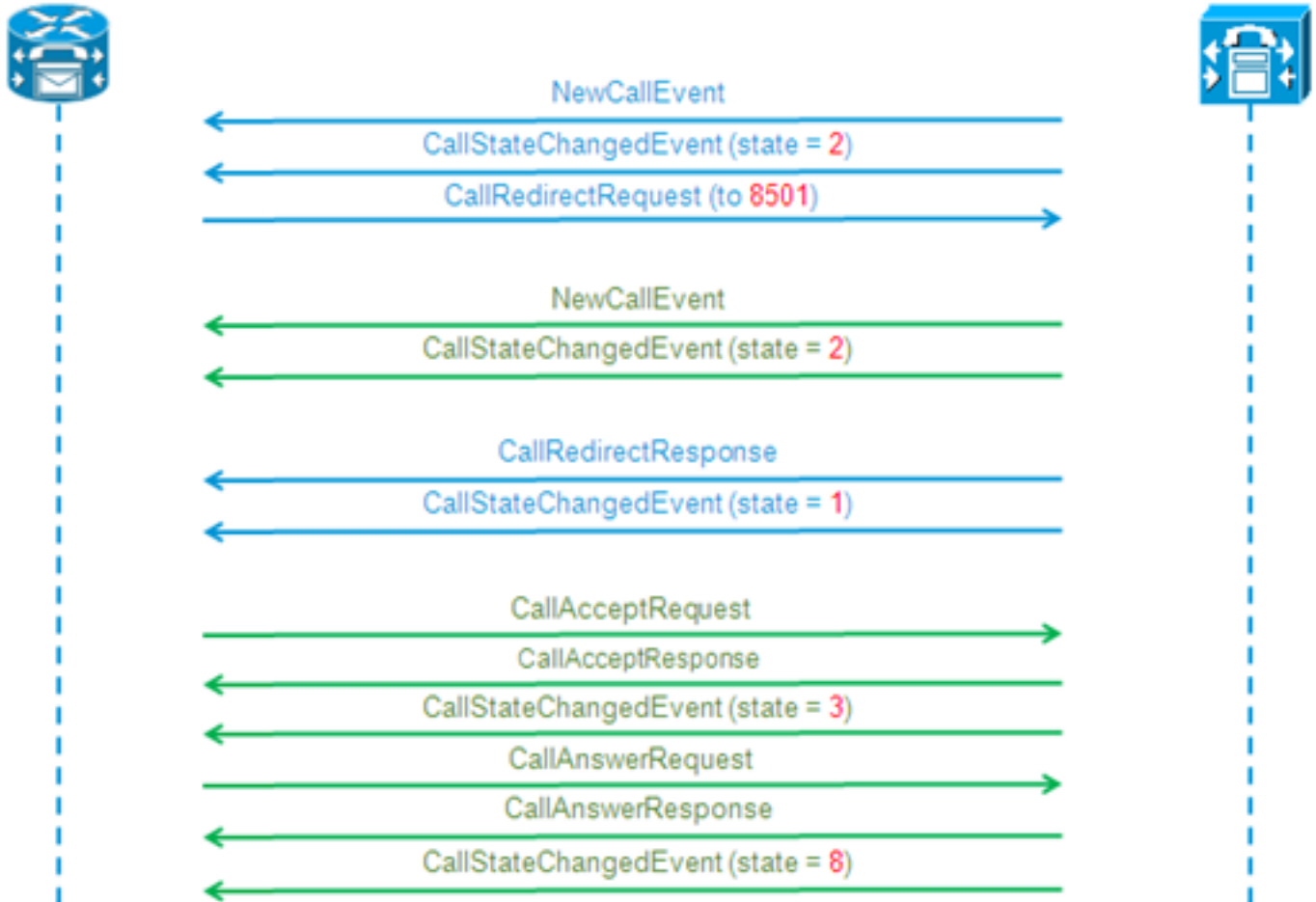
109: 12:05:25.528 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending:
com.cisco.cti.protocol.**DeviceOpenRequest** {
deviceName = **cue_ctiport1**
}
110: 12:05:25.533 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response:
com.cisco.cti.protocol.**DeviceOpenResponse** {
result = 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 {  
}
```

Chiamata di base inoltrata a segreteria telefonica

(output dai registri Cisco Jtapi1 / Cisco Jtapi2)

Nuova chiamata e reindirizzamento alla porta disponibile



Nuova chiamata e reindirizzamento alla porta disponibile

```
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
```

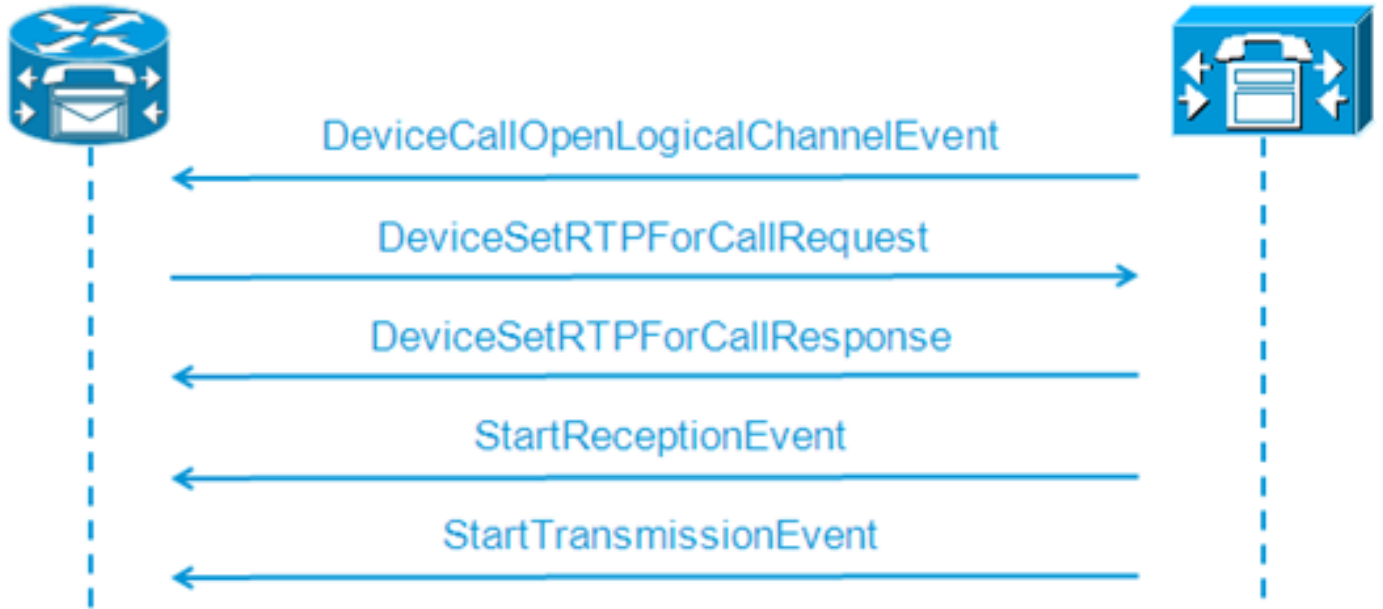


```
farEndpointSpecified = true
fwdDestinationAddress =
reason = 68501
callingParty = 3001
callingPartyName = Ext 3001 - Phone
calledParty = 8000 }
12:46:00.481 %JTAPI-JTAPIIMPL-7-UNK: {(P1-tacjtapiuser) GCID=(1,9040)->ACTIVE}
Initializing to OFFERING for 8501:cue_pt Cause=CAUSE_REDIRECTED Reason= 6
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]
```

La porta CTI accetta la chiamata reindirizzata

```
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
```

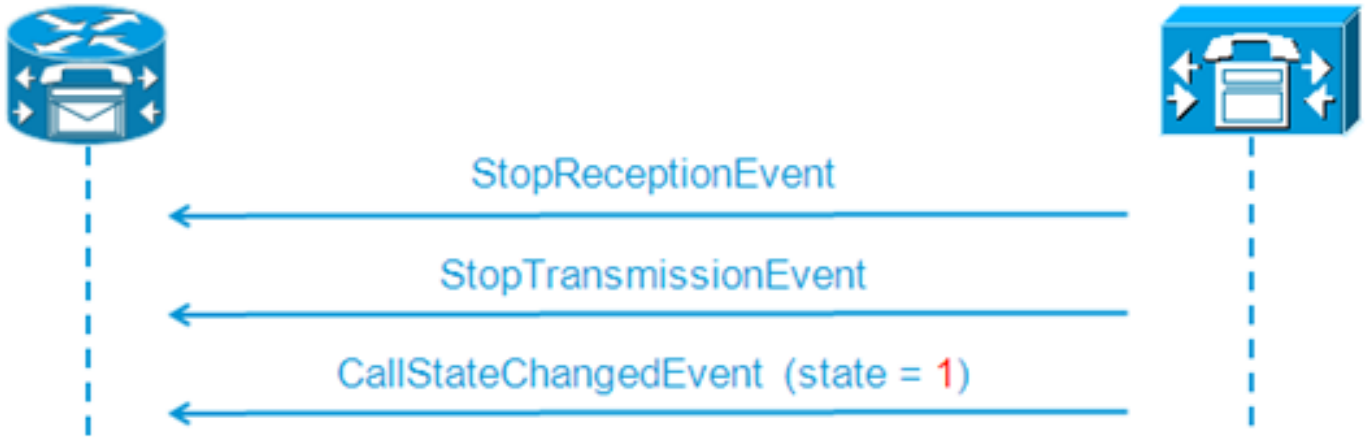
Negoziatore media



```

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
}
  
```

Disconnessione chiamata



```

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]
  
```

Segnalazione MWI On/Off

CUE accende la lampada MWI per la linea 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
}
  
```

È stato composto il numero DTMF '3' per eliminare il messaggio dalla cassetta postale

```

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]
```

CUE spegne la lampada MWI per la linea 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
}
```

Log CCN in tempo reale

(output dai log CCN in tempo reale)

In questo modo viene visualizzata la stessa chiamata dell'esempio precedente quando vengono raccolti i log CCN in tempo reale.

Configurazione chiamata

```
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
```

Disconnessione chiamata

```
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
```

Risoluzione dei problemi

Problemi di connettività

In questo scenario, le porte e i trigger CUE non vengono registrati con CUCM a causa della mancanza di connettività tra CUE e 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: I secondi dell'indicatore orario vanno dalle 13:20:28 alle 13:20:38; pertanto, è possibile stabilire che il CUE non è stato in grado di aprire il socket TCP per 10 secondi prima che venisse riconosciuta l'impossibilità di creare il provider.

Problemi di autenticazione

In questo scenario, le porte e i trigger CUE non vengono registrati con CUCM perché le password configurate tra CUE e CUCM non corrispondono.

Registro 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
```

Utente non abilitato per CTI

In questo scenario, le porte e i trigger CUE non riescono a registrarsi in CUCM perché l'utente dell'applicazione JTAPI non è stato aggiunto al gruppo di autorizzazioni **CTI standard abilitato** sul lato CUCM. Pertanto, anche quando le credenziali utente vengono autenticate di conseguenza, l'utente JTAPI, **tacjtapiuser** in questo caso, non può controllare alcun dispositivo tramite CTI e 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
}
```

Il servizio CUCM CTI Manager è inattivo

In questo scenario, le porte e i trigger CUE non possono essere registrati perché il servizio CTI Manager di CUCM è inattivo o in stato anomalo. Riceve un errore di "connessione rifiutata" per il tentativo di connessione della CUE alla porta TCP JTAPI 2748.

```
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
```

Configurazione non corrispondente

In questo scenario, il CUE non è in grado di registrare il trigger JTAPI con il numero 9999 perché il CTI RP che dovrebbe corrispondere non è configurato o non è stato aggiunto ai "dispositivi controllabili" per l'utente sul lato CUCM. Il CUE lo realizza dopo aver ricevuto il comando **GetDeviceInfoFetchRequest** da CUCM e dopo aver notato che nel dominio del provider non è presente un dispositivo che fa riferimento a tutti i dispositivi controllabili da tale utente e che corrisponderebbe al numero di trigger configurato localmente. La CUE non tenta quindi di inviare una **DeviceOpenRequest** per quel trigger specifico, ma segnala solo l'eccezione nelle tracce. CUE tenta comunque di registrare tutti gli altri dispositivi che si trovano all'interno del dominio del provider inviato da CUCM.

```
13:27:58.864 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Response:
com.cisco.cti.protocol.GetDeviceInfoFetchRequest {
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.GetLineInfoFetchRequest{
```

```
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: Anche quando il trigger 9999 è configurato localmente nella CUE, non fa parte del dominio del provider ricevuto da CUCM e pertanto non viene registrato.

CUE continua ad aprire la linea 8000; incluso nel dominio del provider

```
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 di routing delle chiamate CUCM

In questo scenario, l'utente con DN 3001 chiama il CUE per controllare la sua VM. La chiamata viene presentata al pilota della VM della CUE (CTI RP) con DN 8000. La CUE richiede quindi alla chiamata di essere reindirizzata alla porta CTI del supporto con DN 8501, ma la chiamata non riesce a essere reindirizzata perché il CSS configurato per DN 3001 non ha accesso al PT a cui è assegnato il DN della porta 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

```

Problemi di licenza

In questo scenario, CUE non è in grado di registrare le porte e i trigger perché le licenze per le porte VM non sono state attivate. Per lo stesso motivo, nelle clip non vengono visualizzati tentativi di registrazione.

Riepilogo dal decodificato atrace.log:

```

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():

```

```

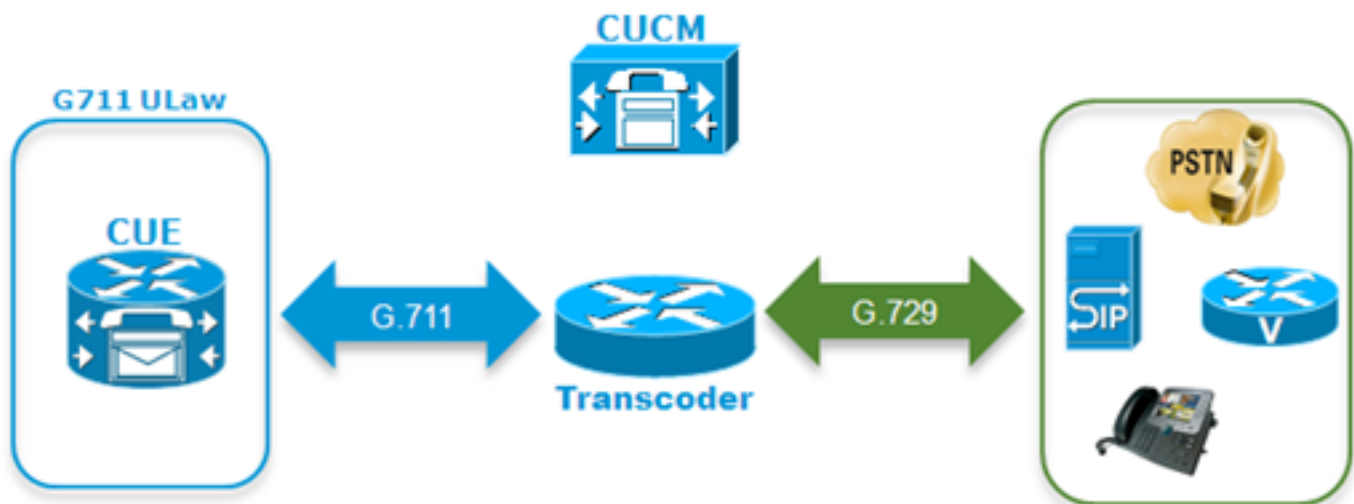
2555 11:45:23.114 LLMA LPRT 0 LlamaPortQuery: get(): maxCount
3456 11:45:23.115 LLMA LSDB 0 Llama: getMaxPortCount(): LlamaSysdbUser():
getInt(): Getting int/sw/apps/limitsManager/port/query/maxCount returns 0
3456 11:45:23.115 LLMA LAPI 0 Llama: getMaxPortCount(): count: 0
3456 11:45:28.727 ACCN TELS 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

```

Procedure ottimali

CUE supporta solo il codec ulaw G711; pertanto, in quasi tutte le implementazioni è necessario un transcodificatore affinché CUE comunichi con altri dispositivi o trunk che utilizzano codec diversi (incluso G711Alaw). Lo stesso vale per l'interazione DTMF con dispositivi che supportano solo DTMF in banda quando è richiesta anche una risorsa Media Termination Point (MTP). A causa di queste limitazioni, Cisco consiglia di:

- Creare un pool di dispositivi isolato per l'uso solo delle porte CTI RP e CTI della CUE. Nel caso in cui più di un CUE sia integrato con CUCM, creare un pool di dispositivi per CUE.
- Create un'area separata solo per le porte RP e CUE e applicatela a tale pool di dispositivi isolato.
- Assicurarsi che la regione sia configurata in modo da consentire solo il G711 con tutte le altre regioni.
- Assicuratevi che venga applicato un elenco di gruppi di risorse multimediali (MRGL) con le risorse di transcodifica disponibili al pool di dispositivi delle porte CTI RP e CTI di CUE in modo che queste abbiano accesso a una risorsa transcodificatore, quando necessario.



- Se l'utente non è in grado di navigare attraverso i menu vocali con toni DTMF, è possibile che sia necessario aggiungere una risorsa MTP al file MRGL dei dispositivi CUE.

Creazione di un profilo VM separato per la PILA in CUCM

Per evitare alcuni problemi recenti osservati con il manager CTI di CUCM, si consiglia di associare tutti i telefoni all'utente JTAPI di CUE sul lato CUCM, anziché solo alle porte e agli RP CTI.

Se si desidera la funzionalità SRST (Survivable Remote Site Telephony):

- Verificare che il trigger SIP corrispondente sia configurato per ogni trigger JTAPI sulla CUE.
- Accertarsi che i dial-peer vengano aggiunti al router della succursale in modo da consentire il routing delle chiamate al modulo CUE tramite SIP in modalità SRST.
- Configurare la maschera del numero esterno di ciascuno dei punti di routing CTI, nonché la maschera per il campo CFU (Call Forward Unregistered) nel CUCM per assicurare che il CUCM instrada le chiamate destinate al modulo succursale ufficio attraverso il gateway PSTN (Public Switched Telephone Network) locale quando la connettività tra CUCM e CUE è stata persa o se viene richiamato il routing alternativo automatico (AAR). Per consentire al router della succursale di instradare le chiamate in entrata dalla PSTN al modulo CUE, potrebbero essere necessarie ulteriori regole di conversione.
- Se l'approccio di trasferimento diretto alla configurazione VM è presente sul lato CUCM e l'utente desidera mantenere questa funzionalità mentre è in CME-SRST, è necessario utilizzare il vecchio modello di DN fittizio con l'approccio di configurazione Call Forward All (CFA) utilizzato per CME prima che il softkey **TransferToVM** diventasse disponibile. Per ulteriori informazioni, fare riferimento a [Trasferimento di un chiamante direttamente in una cassetta postale di Unity Express](#). Ecco un esempio di come può apparire. Tenere presente che questa operazione può essere eseguita solo se si utilizza **CME-SRST** e non SRST legacy con call-manager-fallback. Si supponga che i DN siano compresi nell'intervallo 200-299. Chiamata per x201. Configurare un **dn telefonico** con questo comando:

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

Nel dial-peer che punta a CUE: Utilizzare una regola e un profilo di conversione in uscita per eliminare l'asterisco prefisso ("*") e sostituire il **servizio RDNIS (Redirect Dialed Number Information Service)** con il numero originale a 3 cifre, ad esempio 201, o con il numero completo E.164 nel caso in cui il **numero di telefono** sia stato configurato con il DID completo all'interno della CUE. Assicuratevi che l'intestazione **Diversione di INVITE** inviata alla CUE corrisponda al **numero di telefono** configurato per l'utente sul lato CUE.

Elenco di controllo per la risoluzione dei problemi di registrazione delle porte

1. **Controllare la configurazione sul lato CUCM:** I servizi Web CTI Manager, CallManager e Administrative XML (AXL) sono abilitati e avviati? Le porte CTI e i punti di routing sono stati configurati e assegnati a un DN univoco? L'interfaccia CTI dell'utente JTAPI è abilitata e dispone dell'accesso all'API AXL? L'utente JTAPI controlla tutti i punti di routing e le porte CTI? A volte è consigliabile riavviare il servizio CTI Manager su tutti i server dopo l'aggiunta della configurazione. Tuttavia, questo potrebbe avere un impatto sul servizio e si consiglia di pianificare una finestra di manutenzione, in quanto ciò influisce su tutti gli altri dispositivi che utilizzano CTI e JTAPI con CUCM, ad esempio le applicazioni Unified Contact Center Express (UCCX), IP Manager Assistant (IPMA), Attendant Console, applicazioni ASA o ACD di terze parti e così via.

2. **Controllare la configurazione sul lato CUE:** L'**agente di chiamata** è definito come CUCM?Le licenze per la porta sono state abilitate? Le licenze di valutazione sono accettabili per la configurazione iniziale.È possibile eseguire il ping del CUCM?Le credenziali utente JTAPI sono state aggiunte correttamente e gli **agenti di chiamata** sono stati definiti?Il modulo è stato ricaricato in modo da applicare le modifiche alla configurazione?Se la porta CTI RP e la porta non vengono importate automaticamente da CUCM, provare ad aggiungere manualmente i DN della porta sotto il **sottosistema cn jtapi**, nonché i trigger **jtapi** per ciascun CTI-RP e ricaricare il modulo.

Se tutti questi elementi vengono confermati, il passaggio successivo consiste nell'ottenere le tracce JTAPI sulla CUE ed eventualmente sulle tracce CTI CUCM per isolare ulteriormente il problema.

Informazioni correlate

- [Configurazione e raccolta dei dati di traccia in CUE](#)
- [Risoluzione dei problemi relativi alle funzionalità di sistema di Cisco Unity Express](#)
- [Esempio di configurazione di CallManager per Cisco Unity Express](#)
- [Configurazione profilo casella vocale CUCM](#)
- [Modelli di implementazione di Cisco Unity Express](#)
- [Procedure ottimali per l'implementazione di Cisco Unity Express](#)
- [Transcoding CUCM](#)
- [MTP CUCM](#)
- [Documentazione e supporto tecnico – Cisco Systems](#)