

# CUE JTAPI問題和案例研究

## 目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[CUE JTAPI與CUCM整合概述](#)

[高級呼叫流程示例](#)

[跟蹤的啟用和收集](#)

[即時JTAPI CCN跟蹤](#)

[JTAPI CCN跟蹤日誌](#)

[收集跟蹤日誌檔案](#)

[檢查日誌前必須擁有詳細資訊](#)

[基本CTI概念](#)

[常見CTI呼叫狀態](#)

[跟蹤日誌應該是什麼樣子](#)

[CTI RP和埠註冊](#)

[基本呼叫已轉接到語音信箱](#)

[新呼叫和重定向到可用埠](#)

[新呼叫和重定向到可用埠](#)

[對CTI埠的新呼叫](#)

[CTI埠接受重定向呼叫](#)

[媒體交涉](#)

[呼叫斷開](#)

[MWI開/關信令](#)

[CUE為3001線路開啟MWI燈](#)

[已撥打DTMF號碼「3」以從郵箱中刪除郵件](#)

[CUE關閉3001線路的MWI燈](#)

[即時CCN日誌](#)

[呼叫設定](#)

[呼叫斷開](#)

[疑難排解個案研究](#)

[連線問題](#)

[使用者未啟用CTI](#)

[CUCM CTI管理器服務已關閉](#)

[組態不相符](#)

[CUCM呼叫路由問題](#)

[埠註冊故障排除檢查表](#)

[相關資訊](#)

## 簡介

本文提供如何對Cisco Unity Express(CUE)Java電話應用程式設計介面(JTAPI)進行故障排除的資訊。此外，本文檔還提供了有關如何啟用、收集和檢視不同跟蹤和日誌的資訊和命令，以及故障排除案例研究示例。

## 必要條件

### 需求

思科建議您瞭解以下主題：

- 有關如何通過Web管理介面配置和使用思科統一通訊管理器(CUCM)的基本知識。
- 基本熟悉CUCM中的電腦電話介面(CTI)埠和路由點(RP)。
- 基本熟悉Cisco Unity Express命令列介面。

### 採用元件

本文中的資訊係根據以下軟體和硬體版本：

- Cisco Unity Express版本3.x或更高版本。
- Cisco Unified Communications Manager 7.x或更高版本。

使用的整合方法僅適用於帶Cisco Unified Communications Manager的Cisco Unity Express;不適用於Cisco Unified Communications Manager Express(CUCME)。

Cisco Unity Express必須獲得CUCM而不是CUCME的許可。CUE可隨時與CUCM或CUCME整合，並相應地獲得許可。

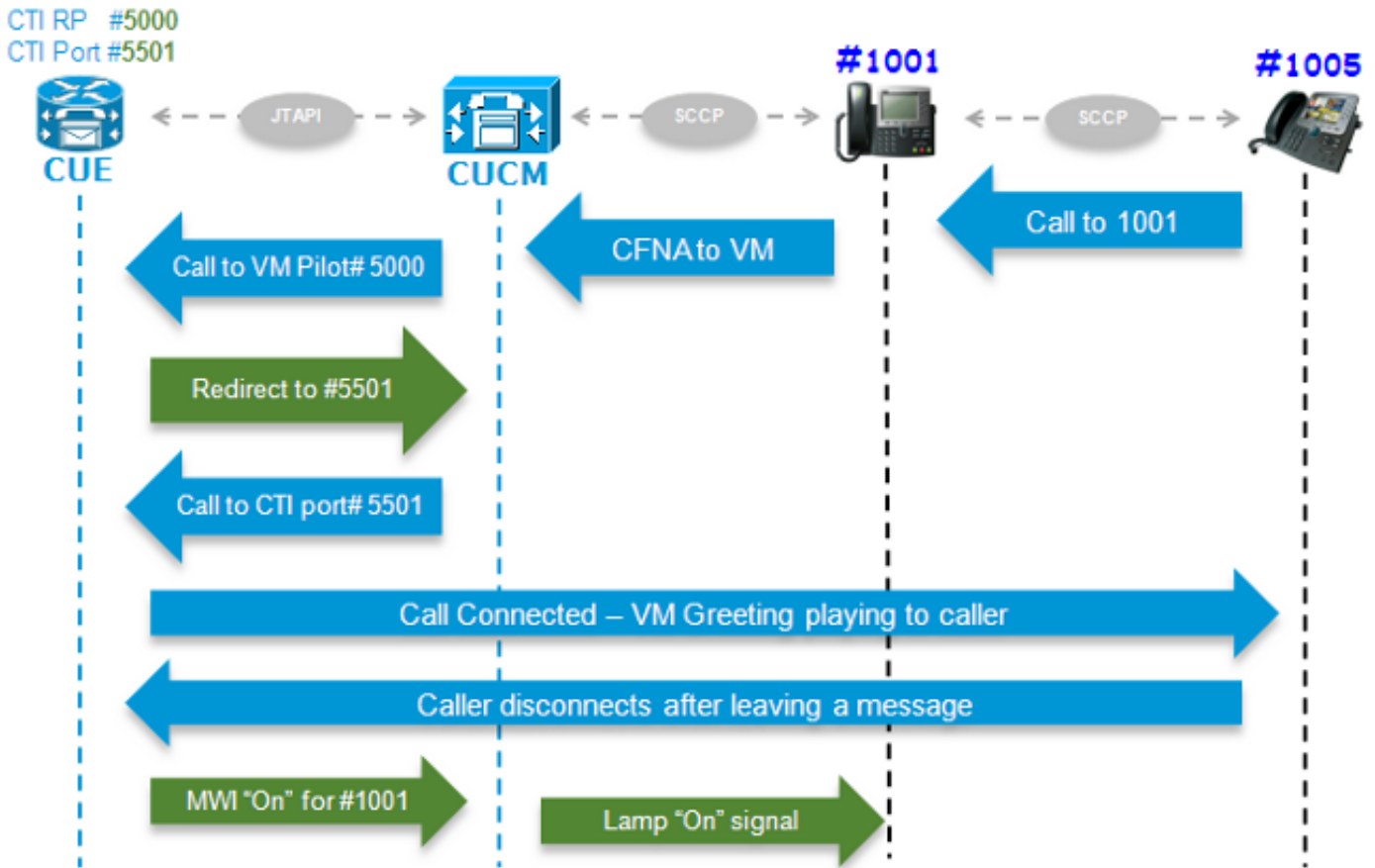
本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路正在作用，請確保您已瞭解任何指令可能造成的影響。

## CUE JTAPI與CUCM整合概述

通過JTAPI語音郵件協定(VM)和自動話務員(AA)功能，可以將CUE與CUCM整合。如果要為向CUCM伺服器註冊的使用者數量較少的一個或多個分支站點調配虛擬機器功能和/或基本AA呼叫處理，建議使用此解決方案。這不需要成熟的Cisco Unity語音郵件伺服器，而是更經濟實惠的實施。同時，CUE還提供其分支的可生存性選項，並在與CUCM的連線丟失時故障切換到會話發起協定(SIP)。

CUE可以通過JTAPI向CUCM註冊，並控制CTI路由點和CTI埠。這允許您通過CUCM作為附加終端控制和管理CUE，以及方便配置和與集群中其他終端的互動。

### 高級呼叫流程示例



電話號碼為(DN)1005的終端使用者呼叫號碼為DN 1001的使用者。如果呼叫未應答（呼叫前轉無應答[CFNA]），則呼叫會在幾秒鐘後轉接到在使用者1001 VM配置檔案中配置的VM號碼。然後CUCM將呼叫傳送到已配置的VM引導5000，後者指向具有DN 5000的CTI RP，由CUE控制。CUE VM應用被觸發，並且呼叫通過JTAPI重定向到可用的CTI埠(DN 5501)以進行媒體建立。播放音訊問候語，使用者可以通過雙音多頻(DTMF)音離開消息或與系統互動。當呼叫者結束呼叫時，CUE向CUCM發出訊號，通過JTAPI將分機1001的消息等待指示器(MWI)燈設定為「On」。然後，CUCM傳送瘦客戶端控制協定(SCCP)消息以開啟電話上的指示燈，並在顯示屏上顯示信封指示，以便使用者1001知道郵箱中有新的VM消息。

## 跟蹤的啟用和收集

有兩種型別的跟蹤：

- 即時JTAPI思科通訊網路(CCN)跟蹤
- JTAPI CCN跟蹤日誌

### 即時JTAPI CCN跟蹤

- 即時JTAPI CCN跟蹤。（啟用這些跟蹤不需要重新載入CUE模組。）
- 輸出沒有使用CCN跟蹤日誌那麼廣泛，但是它們也沒有提供多少資訊。

輸入以下命令以啟用追蹤：

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

輸入以下命令可驗證是否已啟用：

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

輸入以下命令可收集輸出：

```
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 !!
```

輸入CTRL-C以停止即時記錄到控制檯。

## JTAPI CCN跟蹤日誌

啟用JTAPI CCN跟蹤日誌後，需要重新載入CUE模組才能填充日誌。這些日誌、**messages.log**和**atrace.log**可能非常詳細或含義模糊，也可能提供更多資訊和詳細資訊。有四個不同的日誌：

- **atrace.log**  
在網路模組(NM)上預設啟用，但高級整合模組(AIM)預設禁用。輸入**log trace local enable**命令以啟用。它在本地或FTP伺服器中最多寫入10 Mb。若要重新啟動日誌，請輸入**log trace local disable**命令或**no log trace local enable**命令；然後輸入**log trace local enable**命令。輸入**clear trace file**命令以清除**atrace.log**。必須由技術援助中心(TAC)對資料進行解碼。
- **messages.log**  
這些日誌包含系統日誌消息，例如Info ( 資訊 )、Warning ( 警告 )、Error ( 錯誤 ) 和 Fatal ( 致命 )。
- **CiscoJtapi1.log**和**CiscoJtapi2.log**  
它們記錄所有與JTAPI相關的信令和事件。這些日誌更易於理解，資訊量也非常大。  
CiscoJtapi2.log在CiscoJtapi1.log變滿時開始填充，反之亦然。

無論設定哪些跟蹤，系統都會在重新載入後恢復為預設跟蹤級別。若要更改這些預設設定，使其在重新引導後繼續運行，必須輸入**log trace boot**命令。以下是啟用這些功能的命令：

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

輸入以下命令可驗證是否已啟用：

```
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
```

以下是檢視日誌的步驟：

1. 輸入**show logs**命令以檢視CUE中儲存的日誌檔案的清單。
2. **.prev**副檔名表示這是舊跟蹤檔案的備份，而不是當前活動日誌檔案的備份。
3. 您可以將其解壓到外部FTP伺服器。
4. 您還可以從CUE的終端監視器檢視即時記錄到這些檔案的消息的輸出。

## 收集跟蹤日誌檔案

使用以下命令將日誌提取到外部FTP:

```
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/
```

使用**show log name <logname>**命令將日誌顯示到CUE終端監視器中。以下是範例：

```
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>
```

**atrace.log**已編碼；因此，不能僅使用**show log name**命令即時顯示它。

## 檢查日誌前必須擁有詳細資訊

您必須至少從您正在疑難排解的問題的呼叫中獲得此處概述的所有詳細資訊，以便可以輕鬆地跟蹤和瞭解跟蹤：

- 主叫號碼
- 被叫號碼
- 重新導向號碼
- CTI RP DN和裝置名稱
- CTI埠號和裝置名稱
- JTAPI使用者
- 發生呼叫的時間範圍

## 基本CTI概念

**提供商**：CTI服務提供商。該應用程式通過開啟提供程式建立CTI會話。

**使用者**：應用程式與使用者相關聯。

**裝置**：註冊到CUCM的裝置。

**線路**：CTI支援的裝置上的DN外觀。

**呼叫ID(callLegID)**：與呼叫中的一個呼叫段關聯。

**全域性呼叫 ( 呼叫ID )**：標識單個呼叫的所有呼叫段。

## 常見CTI呼叫狀態

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

state = 8 CONNECTED

## 跟蹤日誌應該是什麼樣子

在能夠找到不正確的信令之前，首先需要知道該信令在正常操作下將是什麼樣子；因此，本節介紹當訊號輸出正常工作時，在不同情況下會看到的訊號輸出片段。

另請注意，這些日誌中的所有信令已彙總，只顯示相關詳細資訊，因為它們包含非常詳細的資訊，非常乏味且重複。

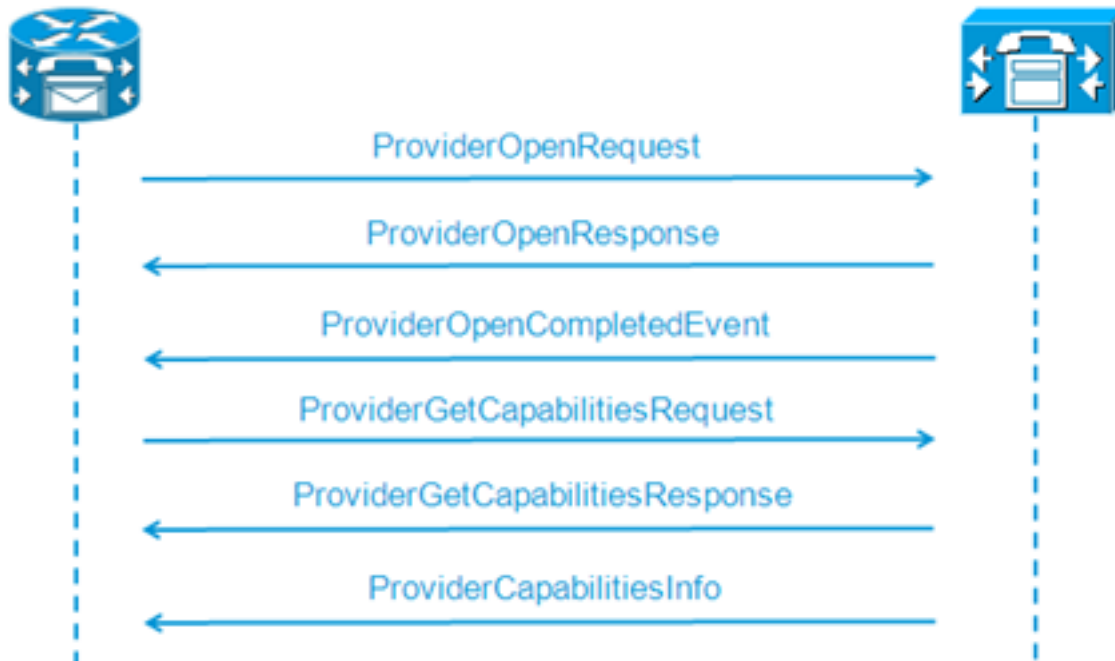
以下是已用組態的詳細資訊：

```
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和埠註冊

( Cisco Jtapi1 / Cisco Jtapi2 日誌的輸出 )

### 1. 開啟提供程式連線



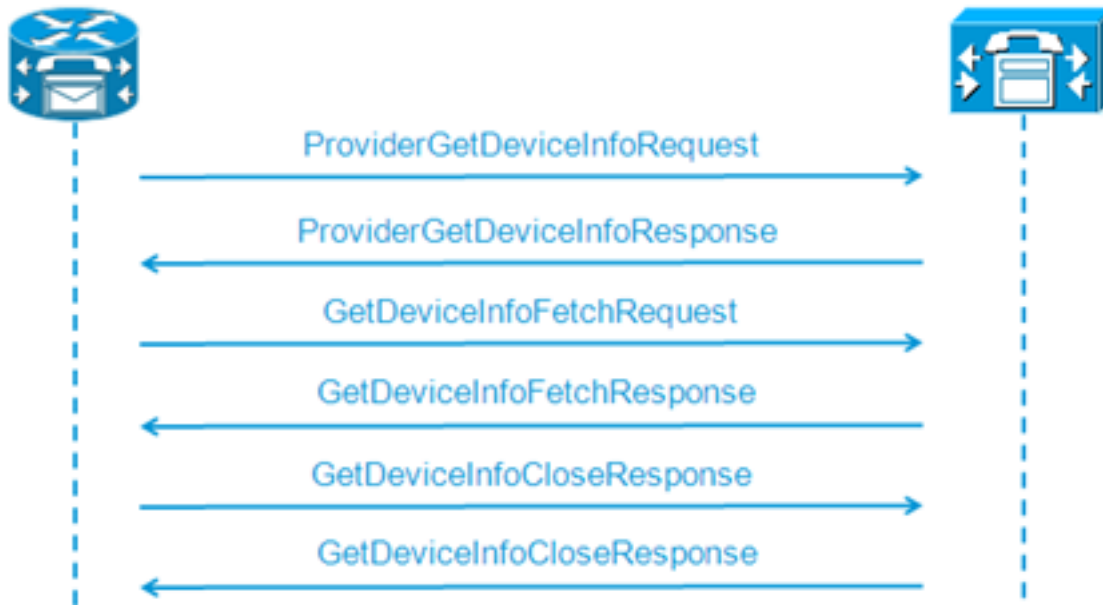
```
21: 12:05:23.686 CST %JTAPI-CTIIMPL-7-UNK.(P1-tacjtapiuser) ProviderID =
P1-tacjtapiuser
22: 12:05:23.739 CST %JTAPI-CTIIMPL-7-UNK.(P1-tacjtapiuser) Trying to
create normal socket connection to 192.168.100.10
23: 12:05:23.747 CST %JTAPI-CTIIMPL-7-UNK.(P1-tacjtapiuser) connected
26: 12:05:24.112 CST %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. 查詢可控裝置



```

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. 獲取CTI埠線路資訊



```

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. 獲取CTI RP線路資訊

```

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應用接收的配置

```

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. 獲取對CTI裝置和線路的控制



```

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]

```

```

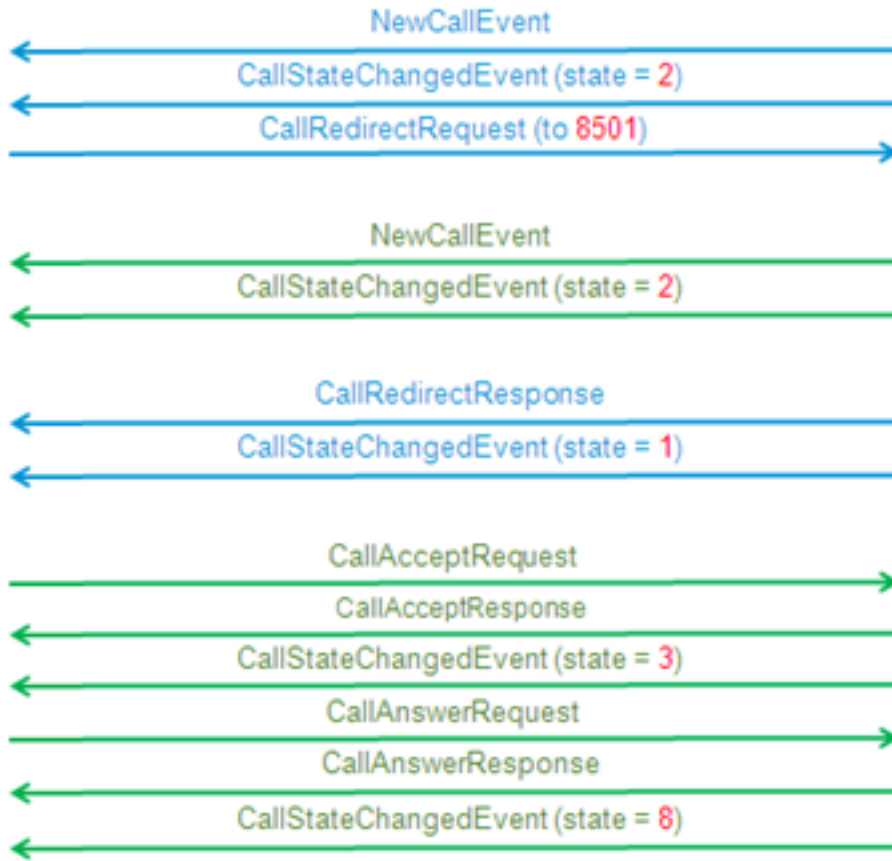
sending: com.cisco.cti.protocol.DeviceCloseRequest {
139: 12:05:25.681 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received
Response: com.cisco.cti.protocol.DeviceCloseResponse {
141: 12:05:25.683 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceRegisterDeviceRequest {
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 {
}

```

## 基本呼叫已轉接到語音信箱

( Cisco Jtapi1 / Cisco Jtapi2 日誌的輸出 )

## 新呼叫和重定向到可用埠



## 新呼叫和重定向到可用埠

```

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
}

```

## 對CTI埠的新呼叫

```

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.

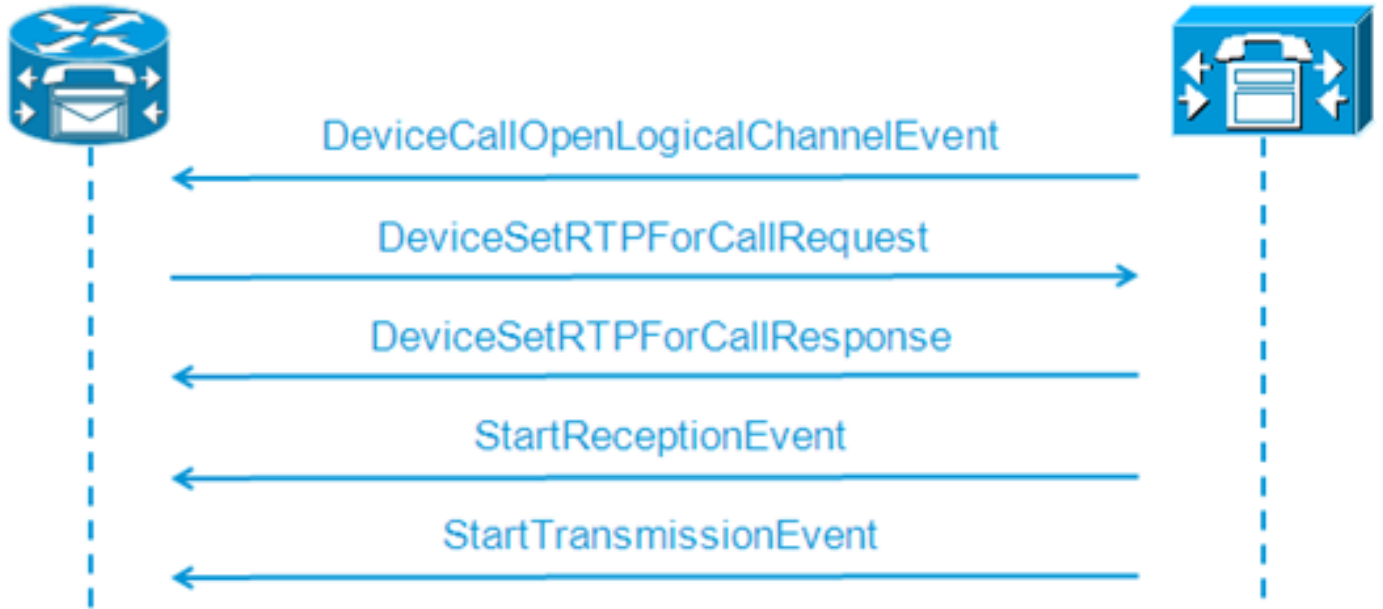
```

```
(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]
```

## CTI埠接受重定向呼叫

```
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
```

## 媒體交涉

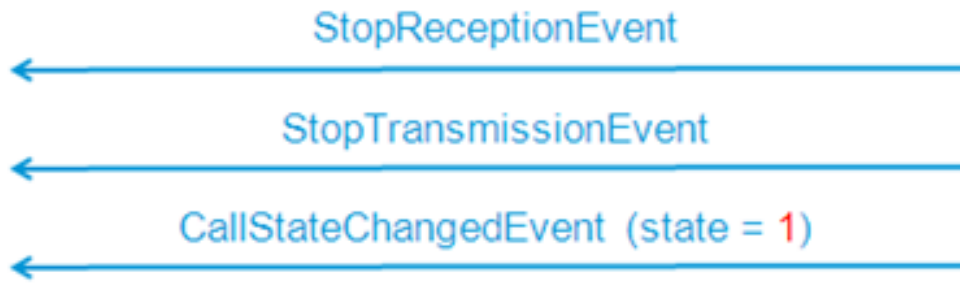


```

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
}
  
```

呼叫斷開





```
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]
```

## MWI開/關信令

### CUE為3001線路開啟MWI燈

```
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
}
```

### 已撥打DTMF號碼「3」以從郵箱中刪除郵件

```
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關閉3001線路的MWI燈

```
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
}
```

## 即時CCN日誌

( 即時CCN日誌的輸出 )

這是當收集即時CCN日誌時，來自此處的上一個示例的相同呼叫出現的方式。

## 呼叫設定

```
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
```

## 呼叫斷開

```
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
```

## 疑難排解個案研究

### 連線問題

在此場景中，由於CUE和CUCM之間缺少連線，CUE埠和觸發器不向CUCM註冊。

```
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 --

```

附註：時間戳秒從13:20:28到13:20:38;因此，在確認無法建立提供程式之前，我們可以判斷CUE無法開啟TCP套接字10秒。

## 驗證問題

在這種情況下，CUE埠和觸發器無法在CUCM中註冊，因為CUE和CUCM之間配置的密碼不匹配。

## CCN日誌

```

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
```

## 使用者未啟用CTI

在此場景中，CUE埠和觸發器無法在CUCM中註冊，因為JTAPI應用程式使用者尚未新增到CUCM端的**Standard CTI Enabled**許可權組中。因此，即使使用者憑證相應地進行驗證，JTAPI使用者(在此案例中為**tacjtapiuser**)無法透過CTI和JTAPI控制任何裝置。

## 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
qbcClientVersion = 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
}
}
```

## CUCM CTI管理器服務已關閉

在這種情況下，CUE埠和觸發器無法註冊，因為CUCM CTI管理器服務已關閉或處於異常狀態。它收到CUE嘗試連線JTAPI TCP埠2748的「連線被拒絕」錯誤。

```
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 msecs
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

```

## 組態不相符

在此場景中，CUE無法註冊編號為9999的JTAPI觸發器，因為沒有配置它應該匹配的CTI RP，或者它尚未新增到CUCM端使用者的「可控裝置」中。CUE在從CUCM收到**GetDeviceInfoFetchResponse**並發現提供商域中沒有裝置（指該使用者的所有可控裝置）後，就實現了這一點，該裝置將匹配其本地配置的觸發器編號。然後CUE不會嘗試為該特定觸發器傳送**DeviceOpenRequest**，而是僅報告跟蹤中的異常。CUE仍嘗試註冊CUCM傳送的提供商域內的所有其他裝置。

```

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.

```

**附註：**即使在CUE中本地配置觸發器9999，它也不是從CUCM接收的提供者域的一部分，因此，它不註冊。

**CUE繼續開啟8000線；包含在提供商的域中**

```

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

```

## CUCM呼叫路由問題

在此場景中，具有DN 3001的使用者呼叫CUE檢查他的VM。該呼叫將呈現給具有DN 8000的CUE的VM引導(CTI RP)。然後CUE請求將呼叫重定向到其帶DN 8501的媒體CTI埠，但由於為DN 3001配置的CSS無權訪問分配了CTI埠的DN的PT，因此呼叫無法重定向到其媒體CTI埠。

```

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

```

## 許可證問題

在此場景中，CUE無法註冊其埠和觸發器，因為VM埠的許可證尚未啟用。由於相同的原因，在捕獲中看不到任何註冊嘗試。

## 已解碼的atrace.log中的摘要：

```

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

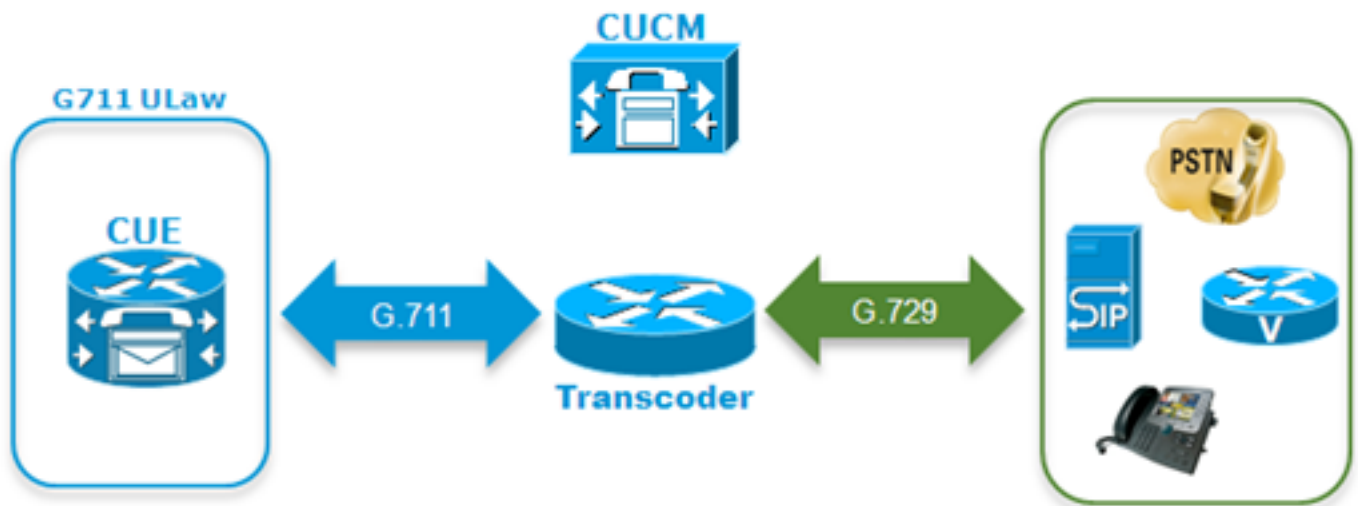
```

```
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
```

### 最佳實踐

CUE僅支援G711ulaw編解碼器；因此，在幾乎每個部署中，CUE都需要一個轉碼器來與使用不同編解碼器（包括G711Alaw）的其他裝置或中繼通訊。這同樣適用於DTMF與僅支援帶內DTMF的裝置(其中還需要介質終端點(MTP)資源)的互通。由於這些限制，思科建議：

- 建立一個單獨的裝置池，僅用於CUE的CTI RP和CTI埠。如果多個CUE與CUCM整合，則每個CUE建立一個裝置池。
- 僅為CUE的RP和埠建立一個單獨區域，並將其應用於該隔離裝置池。
- 確保將區域配置為僅允許G711與所有其他區域進行連線。
- 確保將具有可用轉碼資源的媒體資源組清單(MRGL)應用到CUE的CTI RP和CTI埠的裝置池，以便在需要時這些埠可以訪問轉碼器資源。



- 如果使用者無法使用DTMF音調瀏覽語音選單，則可能必須將MTP資源新增到CUE裝置的MRGL。

### 為CUCM中的CUE建立單獨的VM配置檔案

為了避免最近在CUCM CTI管理器上觀察到一些問題，建議將所有電話關聯到CUCM端的CUE的JTAPI使用者，而不是僅關聯到CTI RP和埠。

如果需要Survivable Remote Site Telephony(SRST)功能：

- 確保為CUE上的每個JTAPI觸發器配置了相應的SIP觸發器。
- 確保將撥號對等體新增到分支路由器，以允許在SRST模式下通過SIP將呼叫路由到CUE模組。
- 配置每個CTI路由點的外部號碼掩碼，以及CUCM中的CFU（未註冊的來電轉駁）欄位的掩碼，以確保CUCM在CUCM和CUE之間的連線丟失或呼叫自動替代路由(AAR)時，通過本地公共交換電話網路(PSTN)網關路由發往分支機構模組的呼叫。分支路由器可能還需要其他轉換規則，才能將入站呼叫從PSTN路由到CUE模組。
- 如果CUCM端存在Direct Transfer to VM配置方法，並且使用者希望在CME-SRST中保持此功能，則必須使用舊的dummy-DN和TransferToVM軟鍵可用之前用於CME的Call Forward All(CFA)配置方法。有關詳細資訊，請參閱[將呼叫者直接轉接到Unity Express郵箱](#)。下面是一個示例。請記住，只有使用CME-SRST，而不是使用call-manager-fallback的舊版SRST才能完

成此操作。假定DN在200-299範圍內。X201的呼叫到達。使用以下命令設定ephone-dn:

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

在指向CUE的撥號對等體中：使用出站轉換規則和配置檔案去除字首("\*")，並將**重定向撥出號碼資訊服務(RDNIS)**替換回原始3位數(例如201)，或者用完整的E.164號碼替換，以備在CUE中使用完整的DID配置電話號。確保傳送到CUE的INVITE的Distribution標頭與CUE端為使用者配置的phonenumber匹配。

## 埠註冊故障排除檢查表

- 檢查CUCM端的配置：** 是否已啟用和啟動CTI Manager、CallManager和管理XML(AXL)Web服務？是否已配置CTI埠和路由點並為其分配唯一的DN？JTAPI使用者CTI是否已啟用，是否具有AXL API訪問許可權？JTAPI使用者是否控制所有CTI路由點和埠？有時，最好在新增配置後在所有伺服器上重新啟動CTI Manager服務。但是，這可能會影響服務，建議安排維護視窗，因為這影響與CUCM一起使用CTI和JTAPI的所有其他裝置，例如Unified Contact Center Express(UCCX)、IP Manager Assistant(IPMA)、話務台、第三方AA或ACD應用程式等。
- 檢查CUE端的配置：** 是否將呼叫代理定義為CUCM？是否已啟用埠許可證？初始配置可接受評估許可證。是否能ping通CUCM？是否正確新增了JTAPI使用者憑據，以及是否定義了呼叫代理？是否已重新載入模組以應用配置更改？如果未自動從CUCM匯入CTI RP和埠，則嘗試在ccn子系統jtapi下手動新增埠DN，以及每個CTI-RP的jtapi觸發器並重新載入模組。

如果所有這些專案都得到確認，則下一步是獲取CUE上的JTAPI跟蹤，並可能獲取CUCM CTI跟蹤，以進一步隔離問題。

## 相關資訊

- [在CUE中設定和收集跟蹤資料](#)
- [Cisco Unity Express系統功能故障排除](#)
- [Cisco Unity Express的CallManager配置示例](#)
- [CUCM語音郵件配置檔案配置](#)
- [Cisco Unity Express部署模式](#)
- [部署Cisco Unity Express的最佳實踐](#)
- [CUCM轉碼](#)
- [CUCM MTP](#)
- [技術支援與文件 - Cisco Systems](#)