



ID de document : 116060

Mis à jour : Sept 06, 2013

Contribué par Michael Mendoza et Bakthavatsal Muralidharan, ingénieurs TAC Cisco.



[PDF de téléchargement](#)



[Copie](#)

[\[+\] Feedback](#)

## Produits connexes

- [Cisco Unity Express](#)

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Intégration du CUE JTAPI avec l'aperçu CUCM](#)

[Exemple de haut niveau d'écoulement d'appel](#)

[Activation et collecte de suivis](#)

[Suivis du temps réel JTAPI CCN](#)

[Logs de suivi JTAPI CCN](#)

[Collectez les fichiers journal de suivi](#)

[Détails incontournables avant que vous vérifiez les logs](#)

[Concepts de base CTI](#)

[États d'appel communs CTI](#)

[Quel suivi se connecte devrait ressembler à](#)

[CTI RP et enregistrement des ports](#)

[Appel de base expédié à la messagerie vocale](#)

[Nouveaux appel et redirection au port disponible](#)

[Nouveaux appel et redirection au port disponible](#)

[Nouvel appel au port CTI](#)

[Le port CTI reçoit l'appel réorienté](#)

[Négociation de support](#)

[Déconnexion d'appel](#)

[Signalisation "Marche/Arrêt" de MWI](#)

[Le CUE allume la lampe de MWI pour la ligne 3001](#)

[Composé '3' de nombre DTMF pour supprimer le message de la boîte aux lettres](#)

[Le CUE arrête la lampe de MWI pour la ligne 3001](#)

[Logs du temps réel CCN](#)

[Établissement d'appel](#)

[Déconnexion d'appel](#)

[Dépannage des études de cas](#)

[Problèmes de connectivité](#)

[Utilisateur CTI-non activé](#)

[Le service de CTI Manager CUCM est en baisse](#)

[Disparité de configuration](#)

[Question de routage d'appels CUCM](#)

[Liste de contrôle pour le dépannage d'enregistrement des ports](#)

[Informations connexes](#)

[Cisco relatif prennent en charge des discussions de la Communauté](#)

## Introduction

Ce document fournit des informations sur la façon dont dépanner l'interface de programmation de téléphonie de Javas de Cisco Unity Express (CUE) (JTAPI). Supplémentaire, ce document fournit les informations et les commandes sur la façon dont activer, collecte, et visualise les différents suivis et logs avec des exemples d'étude de cas de dépannage.

## Conditions préalables

### Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Connaissance de base de la façon configurer et utiliser Cisco Unified Communications Manager (CUCM) par l'interface d'administration de Web.
- Connaissance de base des ports de l'interface de couplage de la téléphonie et de l'informatique (CTI) et des points d'acheminement (RPS) dans CUCM.
- Connaissance de base de l'interface de ligne de commande de Cisco Unity Express.

### Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Version 3.x ou ultérieures de Cisco Unity Express.
- Version 7.x ou ultérieures de Cisco Unified Communications Manager.

La méthode d'intégration utilisée s'applique seulement pour le Cisco Unity Express avec Cisco Unified Communications Manager ; pas avec le Cisco Unified Communications Manager Express (CUCME).

Le Cisco Unity Express doit être autorisé pour CUCM, pas CUCME. Le CUE peut être intégré avec CUCM ou CUCME à tout moment et être autorisé en conséquence.

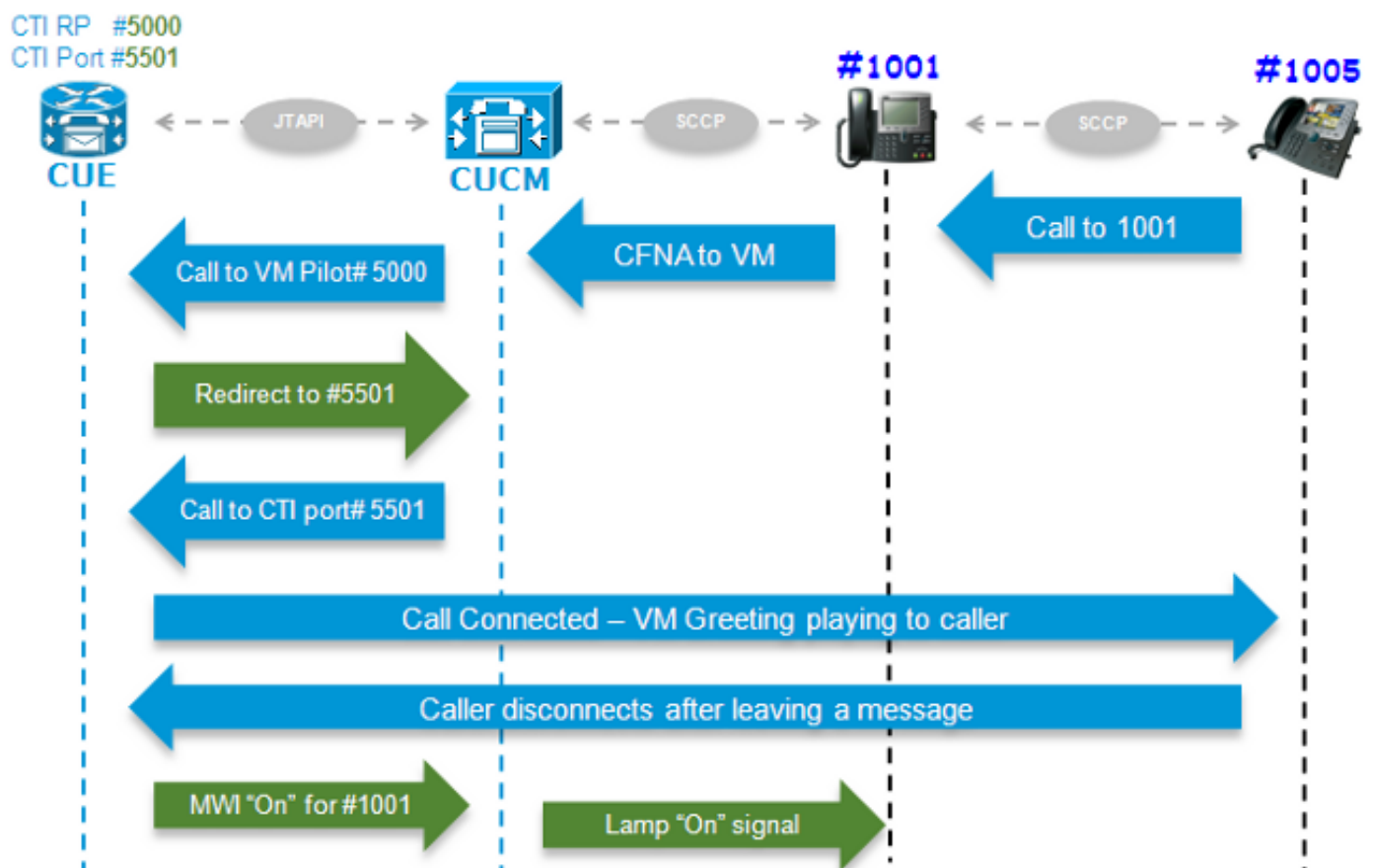
Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

## Intégration du CUE JTAPI avec l'aperçu CUCM

Il est possible d'intégrer le CUE avec CUCM par le protocole JTAPI pour la messagerie vocale (VM) et la fonctionnalité (aa) propre automatisée. Cette solution est recommandée quand vous voulez provision des caractéristiques VM et/ou la gestion des appels de base aa pour un ou de plusieurs filiales avec un nombre restreint d'utilisateurs est enregistrée à un serveur CUCM. Ceci n'exige pas un véritable serveur de messagerie vocale de Cisco Unity, mais une implémentation beaucoup plus abordable. En même temps, le CUE également fournit des options de capacité de survie pour ses branchements et bascule à Protocol initié par session (SIP) quand la Connectivité au CUCM est perdue.

Le CUE peut s'inscrire au CUCM par JTAPI et contrôle des points de routage CTI et des ports CTI. Ceci te permet pour contrôler et gérer le CUE comme point final supplémentaire par le CUCM, aussi bien que facilite les configurations et les interactions avec d'autres points finaux dans la batterie.

### Exemple de haut niveau d'écoulement d'appel



L'utilisateur final avec le répertoire nombre (DN) 1005 appelle l'utilisateur avec le DN 1001. L'appel est expédié après quelques secondes si l'appel n'est pas répondu, d'appel le pas de réponse en

avant (CFNA), au nombre VM a configuré sur le profil VM de l'utilisateur 1001. Le CUCM envoie alors l'appel au pilote configuré 5000 VM, qui indique un CTI RP avec le DN 5000 qui est contrôlé par CUE. L'application VM de CUE est déclenchée, et l'appel est réorienté par JTAPI à un port CTI disponible (DN 5501) pour l'établissement de medias. Les jeux sonores de message d'accueil, et l'utilisateur peuvent partir d'un message ou interagir avec le système par des tonalités multifréquences de double tonalité (DTMF). Quand l'appelant finit l'appel, le CUE signale le CUCM pour placer la lampe d'indicateur de message en attente (MWI) pour l'extension 1001 à "ON" par JTAPI. Le CUCM envoie alors le message de Skinny Client Control Protocol (SCCP) pour activer la lumière au téléphone, aussi bien qu'affiche l'indication d'enveloppe sur l'affichage ainsi l'utilisateur 1001 se rend compte qu'il y ait un nouveau message VM dans la boîte aux lettres.

## Activation et collecte de suivis

Il y a deux types de suivis :

- Suivis du réseau de communications du temps réel JTAPI Cisco (CCN)
- Logs de suivi JTAPI CCN

### Suivis du temps réel JTAPI CCN

- Suivis du temps réel JTAPI CCN. (L'activation de ces suivis n'exige pas une recharge du module de CUE.)
- La sortie n'est pas aussi étendue que le suivi CCN se connecte, mais ils ne sont pas très instructifs non plus.

Sélectionnez ces commandes afin d'activer les suivis :

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

Sélectionnez cette commande afin de vérifier qu'ils sont activés :

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

Sélectionnez cette commande afin de collecter la sortie :

```
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 !!
```

Écrivez **CTRL-C** pour cesser le temps réel se connecter à la console.

### Logs de suivi JTAPI CCN

Une recharge du module de CUE est exigée après que les logs de suivi JTAPI CCN soient activés pour que les logs deviennent remplis. Ces logs, **messages.log** et **atrace.log**, peuvent être très détaillés ou cryptiques aussi bien que beaucoup plus instructif et détaillé. Il y a quatre logs différents :

- **atrace.log**

Activé par défaut sur les modules réseau (NM), mais désactivé par défaut pour les modules d'intégration avancés (objectifs). Sélectionnez la commande **locale d'enable de suivi de log** afin d'activer. Il prépare à la mi-bande 10 localement ou à un ftp server. Afin de redémarrer le log, ne sélectionnez la commande **locale de débranchement de suivi de log** ou l'**aucune** commande **locale d'enable de suivi de log** ; sélectionnez alors la commande **locale d'enable de suivi de log**. Sélectionnez la commande **claire de fichier de suivi** afin d'effacer **atrace.log**. Les données doivent être décodées par le centre d'assistance technique (TAC).

- **messages.log**

Ce sont des logs qui contiennent des messages de Syslog, tels que les informations, l'avertissement, l'erreur, et mortel.

- **CiscoJtapi1.log** et **CiscoJtapi2.log**

Ils se connectent tous les signalisation et événements liés JTAPI. Il est beaucoup plus facile comprendre ces logs et très instructif. CiscoJtapi2.log commence au remplir quand CiscoJtapi1.log devient complètement et vice-versa.

Indépendamment quels suivis sont placés, le système retourne aux niveaux de suivi par défaut après une recharge. Afin de changer ces valeurs par défaut de sorte qu'ils survivent à une réinitialisation, vous devez sélectionner la commande de **démarrage de suivi de log**. Voici la commande de les activer :

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

Sélectionnez cette commande afin de vérifier qu'ils sont activés :

```
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
```

Voici les étapes pour visualiser les logs :

1. Écrivez les **shows log** commandent afin de visualiser une liste des fichiers journal enregistrés dans le CUE.
2. L'extension de fichier **.prev** signifie que c'est une sauvegarde d'un fichier de suivi plus ancien et pas du fichier journal actif en cours.
3. Vous pouvez les extraire à un ftp server externe.
4. Vous pouvez également visualiser la sortie des messages qui sont enregistré à ces fichiers en temps réel du terminal monitor du CUE.

## Collectez les fichiers journal de suivi

Extrayez les logs à un FTP externe avec ces commandes :

```
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

Les logs d'affichage au terminal monitor de CUE avec le **show log** nomment le **<logname >** la commande. Voici un exemple :

```
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** est encodé ; donc vous pouvez non seulement l'afficher en temps réel avec la commande de nom de **show log**.

## Détails incontournables avant que vous vérifiez les logs

Vous devez obtenir, pour le moins, tous les détails tracés les grandes lignes dans le présent des appels avec la question que vous dépannez de sorte que vous puissiez facilement dépister et comprendre les suivis :

- Numéro d'appel
- Numéro appelé
- Réorientez le nombre
- DN et nom du périphérique CTI RP
- Nombre et nom du périphérique de port CTI
- Utilisateur JTAPI
- La plage de temps les appels a eu lieu

## Concepts de base CTI

**Fournisseur** : Un fournisseur des services CTI. L'application établit une session CTI **en ouvrant un** fournisseur.

**Utilisateur** : Des applications sont associées avec un utilisateur.

**Périphérique** : Un périphérique qui s'enregistre au CUCM.

**Ligne** : Apparence de DN sur un périphérique pris en charge CTI.

**ID d'appel (callLegID)** : Associé avec un tronçon d'appel dans un appel.

**Appel global (callID)** : Identifie tous les tronçons d'appel pour un appel unique.

## États d'appel communs CTI

```
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>
```

## Quel suivi se connecte devrait ressembler à

Avant que vous puissiez trouver la signalisation incorrecte, vous le premier besoin de savoir ce que ressemblerait à cette signalisation sous le fonctionnement normal ; ainsi cette section affiche que les extraits de la signalisation vous sort verraient dans différents scénarios quand ils fonctionnent normalement.

Veillez également se rendre compte que toute la signalisation de ces derniers se connecte a été récapitulée pour afficher seulement les détails appropriés parce qu'ils contiennent très les informations détaillées qui sont tout à fait pénibles et répétitives.

Voici les détails des configurations utilisées :

```
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>
```

## CTI RP et enregistrement des ports

*(Les sorties du CiscoJtapi1/de Cisco Jtapi2 se connecte)*

### 1. Ouvrez une connexion de fournisseur



```
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. Requête pour les périphériques contrôlables



```

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. Obtenez la ligne les informations de port 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. Obtenez la ligne les informations CTI RP

```

62: 12:05:25.043 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 8
deviceName = cue_vm_ctirp
}
63: 12:05:25.044 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 8
result = 0
}
64: 12:05:25.045 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 9
}
65: 12:05:25.047 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 9
result = 0
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 8000
displayName =
permanentLineID = 52
partition = cue_pt
defaultIntercomTargetInfo = com.cisco.cti.protocol.LineIntercomSpeedDialInfo {
unicodeLabel = com.cisco.cti.protocol.UnicodeString {
}
}
}
66: 12:05:25.048 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 10
}
67: 12:05:25.058 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 10
result = 0
}
68: 12:05:25.059 CST %JTAPI-CTI-7-UNK.(P1-tacjtapiuser)

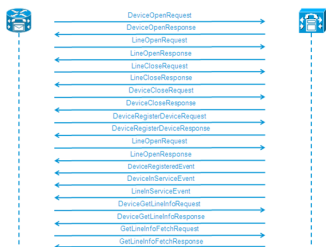
```

**DeviceMap: adding device "cue\_vm\_ctirp"**  
 69: 12:05:25.059 CST %JTAPI-CTI-7-UNK.(P1-192.168.100.10)  
 refreshing device map: previous=0 current=2 created=2 removed=0

**5. Le CUE applique la configuration reçue** 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. Obtenez le contrôle des périphériques et des lignes 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

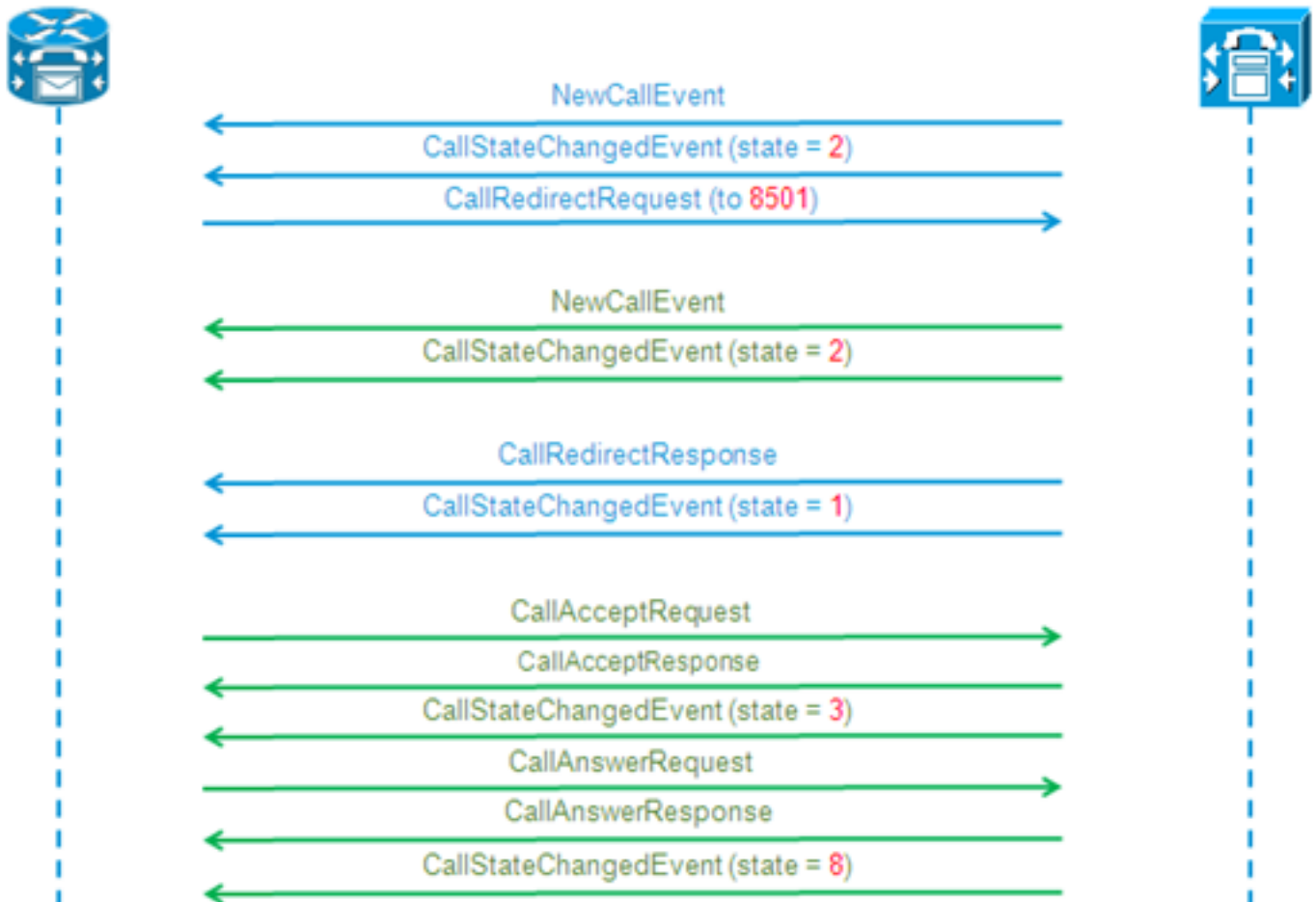
```

```
Event: com.cisco.cti.protocol.LineInServiceEvent {  
}
```

## Appel de base expédié à la messagerie vocale

(Les sorties du CiscoJtapi1/de Cisco Jtapi2 se connecte)

### Nouveaux appel et redirection au port disponible



### Nouveaux appel et redirection au port disponible

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



```

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]

```

## Le port CTI reçoit l'appel réorienté

```

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

```

## Négociation de support



```

12:46:00.531 %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:

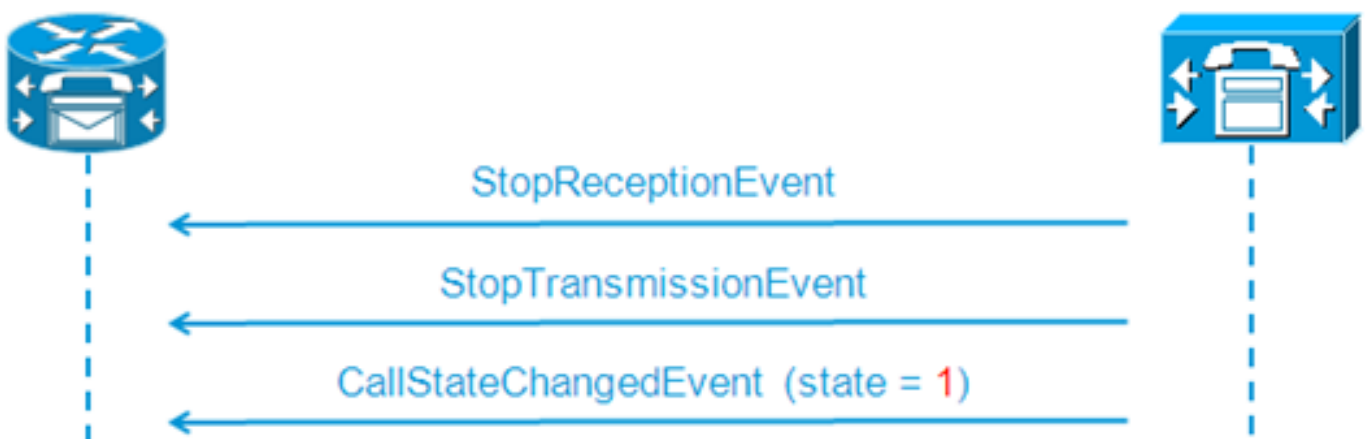
```

```

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
}

```

## Déconnexion d'appel



```

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]
```

## Signalisation "Marche/Arrêt" de MWI

### Le CUE allume la lampe de MWI pour la ligne 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
}
```

### Composé '3' de nombre DTMF pour supprimer le message de la boîte aux lettres

```
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]
```

### Le CUE arrête la lampe de MWI pour la ligne 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
}
```

## Logs du temps réel CCN

*(Sorties des logs de temps réel CCN)*

C'est comment le même appel de l'exemple précédent dans le présent apparaît quand les logs du temps réel CCN sont collectés à la place.

## Établissement d'appel

```
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
```

## Déconnexion d'appel

```
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
```

## Dépannage des études de cas

### Problèmes de connectivité

Dans ce scénario, les ports et les déclencheurs de CUE ne s'inscrivent pas au CUCM dû à du manque de Connectivité entre le CUE et le 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 msecs
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 --

Remarque: Les secondes d'horodateur disparaissent de 13:20:28 à 13:20:38 ; donc, nous pouvons dire le CUE ne pouvions pas ouvrir le socket de TCP pendant 10 secondes avant accusé de réception d'incapacité de créer le fournisseur.

## Questions d'authentification

Dans ce scénario, les ports et les déclencheurs de CUE ne s'inscrivent pas au CUCM parce que les mots de passe configurés entre le CUE et le CUCM ne s'assortissent pas.

## Log 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
```

## Utilisateur CTI-non activé

Dans ce scénario, les ports et les déclencheurs de CUE ne s'inscrivent pas au CUCM parce que l'utilisateur d'application JTAPI n'a pas été ajouté au groupe **standard d'autorisation activé par CTI** dans le côté CUCM. Par conséquent, même lorsque les identifiants utilisateurs authentifient en conséquence, l'utilisateur JTAPI, **tacjtapiuser** dans ce cas, ne peut contrôler aucun périphériques par CTI et 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
}

```

## Le service de CTI Manager CUCM est en baisse

Dans ce scénario, les ports et les déclencheurs de CUE ne peuvent pas s'enregistrer parce que le service de CTI Manager CUCM est en baisse ou dans un état anormal. Il reçoit une erreur refusée « par connexion » pour la tentative de la connexion du CUE au port TCP 2748 JTAPI.

```

18956: 16:25:45.516 CDT %JTAPI-CTIIMPL-7-UNK.(P200-) Provider.
info(CCMEncryption::encryptPassword was successful)
18957: 16:25:45.516 CDT %JTAPI-CTIIMPL-7-UNK.(P200-) application did
not set appinfo, creating default
18958: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.ProviderImpl(): calling
jtapiProperties.getSecurityPropertyForInstance()
18959: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.(P200-tacjtapiuser )
TraceModule: JTAPI version Cisco Jtapi version 7.0(1.1000)-1 Release
18960: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.(P200-tacjtapiuser )
Route Select Timeout is 5000 msec
18961: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.(P200-tacjtapiuser )
Jtapi post condition timeout is set
to 15 seconds
18962: 16:25:45.516 CDT %JTAPI-JTAPIIMPL-7-UNK.(P200-tacjtapiuser )
IgnoreFwdDestination
set to false
18963: 16:25:45.516 CDT %JTAPI-CTIIMPL-7-UNK.(P200-tacjtapiuser )
Opening server "192.168.100.10" login "tacjtapiuser "

```

```
18964: 16:25:45.516 CDT %JTAPI-CTIIMPL-7-UNK.(P200-tacjtapiuser )
ProviderID = P200-tacjtapiuser
18965: 16:25:45.517 CDT %JTAPI-CTIIMPL-7-UNK.(P200-tacjtapiuser )
Trying to create normal socket connection to 192.168.100.10
18966: 16:25:45.518 CDT %JTAPI-JTAPI-7-UNK:[DefaultJtapiPeer]
PlatformExceptionImpl caught:
Unable to create provider -- 192.168.100.10/192.168.100.10:2748 -
Connection refused
```

## Disparité de configuration

Dans ce scénario, le CUE ne peut pas enregistrer le déclencheur JTAPI avec le numéro 9999 parce que le CTI RP qu'il devrait apparier n'est pas configuré, ou on ne l'a pas ajouté au ? périphériques contrôlables ? pour l'utilisateur du côté CUCM. Le CUE réalise ceci après qu'il reçoive le **GetDeviceInfoFetchRequest** du CUCM et note qu'il n'y a pas un périphérique dans le domaine de fournisseur, qui se rapporte à tous les périphériques contrôlables par cet utilisateur, qu'apparie le nombre de déclencheur il a configuré localement. Le CUE alors n'essaye pas d'envoyer un **DeviceOpenRequest** pour ce déclencheur spécifique et signale à la place seulement l'exception dans les suivis. De CUE toujours les essais pour enregistrer tous autres périphériques qui sont dans le domaine du fournisseur ont envoyé par le 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.
```

Remarque: Même lorsque le déclencheur 9999 est configuré localement dans le CUE, ce n'est pas une partie du domaine de fournisseur reçu du CUCM, et donc, ne s'enregistre pas.

## Le CUE continue à ouvrir la ligne 8000 ; qu'est inclus dans le fournisseur ? domaine s

```
13:28:00.953 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceOpenRequest {
deviceName = cue_vm_ctirp
}
13:28:00.979 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.LineOpenRequest {
deviceName = cue_vm_ctirp
lineName = 8000
}
13:28:00.983 CST %JTAPI-JTAPIIMPL-7-UNK:cue_vm_ctirp: Lines opened
13:28:00.997 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.DeviceRegisterDeviceRequest
deviceName = cue_vm_ctirp
13:28:01.000 CST %JTAPI-CTI-7-UNK.(P1-tacjtapiuser) DeviceMap: opening device
"cue_vm_ctirp"
```

```

13:28:01.001 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending: com.cisco.cti.protocol.LineOpenRequest {
lineName = 8000
13:28:01.012 CST %JTAPI-JTAPIIMPL-7-UNK:cue_vm_ctirp: Lines opened
13:28:01.164 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.DeviceRegisteredEvent {
13:28:01.165 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.DeviceInServiceEvent {
13:28:01.166 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) received Event:
com.cisco.cti.protocol.LineInServiceEvent {
13:28:01.168 CST %JTAPI-JTAPIIMPL-7-UNK.(P1-tacjtapiuser) Terminal
"cue_vm_ctirp" in service

```

## Question de routage d'appels CUCM

Dans ce scénario, l'utilisateur avec le DN 3001 appelle le CUE pour vérifier sa VM. L'appel est présenté au pilote VM du CUE (CTI RP) avec le DN 8000. Le CUE invite alors l'appel à obtenir réorienté à son port CTI de médias avec le DN 8501, mais l'appel n'obtient pas réorienté parce que le CSS configuré pour le DN 3001 n'a pas accès à la pinte où le DN du port CTI est assigné.

```

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
```

## Questions de permis

Dans ce scénario, le CUE ne peut pas enregistrer ses ports et déclencheurs parce que les permis pour les ports VM n'ont pas été lancés. Aucune tentative d'enregistrement n'est vue dans les captures pour la même raison.

## Résumé d'atrace.log décodé :

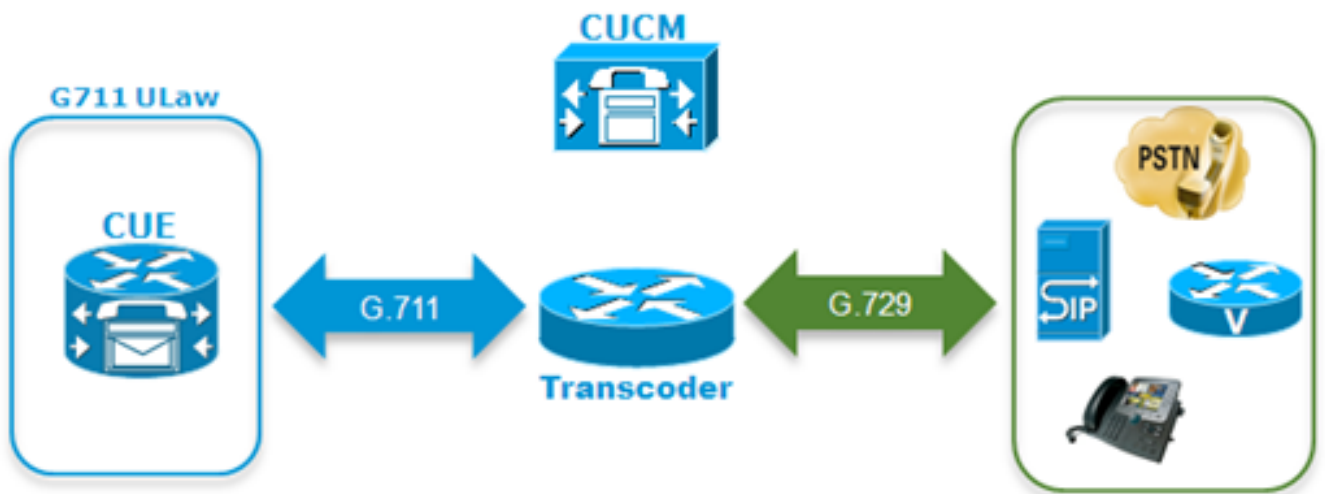
```
2551 11:45:17.178 LLMA LAPI 0 Llama: getMaxVmPortCount():
2547 11:45:17.178 LLMA LVMP 0 LlamaVmPortQuery: get(): maxCount
2551 11:45:17.178 LLMA LSDB 0 Llama: getMaxVmPortCount(): LlamaSysdbUser():
getInt(): Getting int /sw/apps/limitsManager/vmPort/query/maxCount returns 0
2551 11:45:17.178 LLMA 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
```

## [Meilleures pratiques](#)

Le CUE prend en charge seulement les codecs G711ulaw ; donc dans presque chaque déploiement un transcodeur est exigé pour que le CUE communique avec d'autres périphériques ou joncteurs réseau qui utilisent différents codecs (inclut G711Alaw). Le même s'applique pour l'interworking DTMF avec les périphériques qui prennent en charge seulement l'intrabande DTMF

où une ressource en Media Termination Point (MTP) est également exigée. En raison de ces limites, Cisco recommande à :

- Créez un Pool d'appareils d'isolement pour l'utilisation du CUE seulement ? RPS s CTI et ports CTI. Au cas où plus d'un CUE serait intégré avec CUCM, puis créez un Pool d'appareils par CUE.
- Créez une région distincte seulement pour le RP et les ports du CUE et appliquez-vous l'à celle Pool d'appareils d'isolement.
- Assurez que la région est configurée pour permettre seulement G711 avec toutes autres régions.
- Assurez qu'une liste de groupe de ressources de medias (MRGL) avec des ressources en transcodage disponibles est appliquée au Pool d'appareils du CTI RP et ports CTI du CUE de sorte que ceux-ci aient accès à une ressource en transcodeur, une fois nécessaire.



- Si l'utilisateur ne peut pas naviguer par les menus de Voix avec des tonalités DTMF, alors il est possible qu'une ressource MTP doive être ajoutée au MRGL des périphériques de CUE.

### Créez un profil distinct VM pour le CUE dans le CUCM

Afin d'éviter quelques questions récentes observées avec le CTI Manager CUCM, il est recommandé pour associer tous les téléphones à l'utilisateur JTAPI du CUE du côté CUCM, au lieu seulement de la RPS CTI et des ports.

Si la fonctionnalité de Survivable Remote Site Telephony (SRST) est désirée :

- Assurez que le déclencheur correspondant de SIP est configuré pour chaque déclencheur JTAPI sur le CUE.
- Assurez que des cadran-pairs sont ajoutés au routeur secondaire afin de permettre des appels à conduire au module de CUE par le SIP tandis qu'en mode SRST.
- Configurez le masque externe de nombre de chacun des points de routage CTI, aussi bien que le masque pour le champ CFU (non enregistrés en avant d'appel) dans le CUCM pour assurer le CUCM conduit les appels destiné pour le module de succursale par la passerelle locale du réseau téléphonique public commuté (PSTN) quand la Connectivité entre le CUCM et le CUE a été perdue ou si la voie de déroutement automatisée (AAR) est appelée. Des Règles de traduction supplémentaires pourraient être exigées pour que le routeur secondaire puisse conduire des appels d'arrivée du PSTN au module de CUE aussi bien.

- Si le transfert direct à l'approche de configuration VM est présent dans le côté CUCM, et l'utilisateur veut mettre à jour cette fonctionnalité tandis que dans CME-SRST, alors vous devez utiliser le vieux factice-DN avec l'approche call forward all de la configuration (CFA) qui a été utilisée pour CME avant que la clé douce de **TransferToVM** soit devenue disponible. Référez-vous au [transfert un appelant directement dans un](#) pour en savoir plus de [boîte aux lettres d'Unity Express](#). Voici un exemple de la façon dont ceci peut regarder. Maintenez s'il vous plaît dans l'esprit que ceci peut seulement être fait si **CME-SRST** est utilisé et appel-gestionnaire-retour non existant de **SRSTwith**. Supposez que les dn sont dans la plage 200-299. L'appel entre pour x201. Configurez un **ephone-dn** avec cette commande : 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

```

Dans le cadran-pair se dirigeant POUR INTERCALER : Utilisez une règle de conversion et un profil sortants d'éliminer le ("\*")préfixé d'astérisque et de remplacer le **service d'informations de nombre composé par réorientation (RDNIS)** de nouveau au nombre de chiffre de l'original 3, par exemple, 201, ou avec le plein nombre E.164 au cas où le **phonenumber** était configuré avec le plein ONT FAIT à l'intérieur du CUE. Assurez que l'en-tête de transfert de l'INVITATION qui est envoyée au CUE apparie le **phonenumber** configuré pour l'utilisateur du côté de CUE.

# Liste de contrôle pour le dépannage d'enregistrement des ports

1. **Vérifiez la configuration du côté CUCM** : Est-ce que CTI Manager, le CallManager, et les services Web administratifs XML (AXL) sont activés et mis sur pied ? Les ports CTI et les points d'acheminement ont-ils été configurés et ont-ils assigné un seul DN ? L'utilisateur CTI JTAPI est-il activé, et a-t-il accès AXL API ? L'utilisateur JTAPI a-t-il le contrôle de tous les points de routage CTI et ports ? Parfois c'est une bonne idée de redémarrer le service de CTI Manager sur tous les serveurs après que la configuration soit ajoutée. Cependant, ceci pourrait être affecter de service et son recommandé pour programmer une fenêtre de maintenance, parce que ceci affecte tous autres périphériques qui utilisent CTI et JTAPI avec le CUCM, tel qu'Unified Contact Center Express (UCCX), l'IP Manager Assistant (IPMA), la console de réception, le tiers aa ou les applications ACD, et ainsi de suite.
2. **Vérifiez la configuration du côté de CUE** : Le **call-agent** est-il défini comme CUCM ? Les permis de port ont-ils été activés ? Les permis d'évaluation sont acceptables pour la configuration initiale. Pouvez-vous cingler le CUCM ? Les identifiants utilisateurs JTAPI ont-ils été ajoutés correctement, et les **calls-agent** ont-ils été définis ? Le module a-t-il été rechargé de sorte que les modifications de configuration soient appliquées ? Si les CTI RP et port ne sont pas importés du CUCM automatiquement, alors essayez pour ajouter manuellement les dn de port sous le **jtapi de sous-système de ccn**, aussi bien que les déclencheurs de **jtapi** pour chaque CTI-RP et pour recharger le module.

Si tous ces éléments sont confirmés, alors votre étape suivante est d'obtenir des suivis JTAPI sur le CUE et probablement des suivis CUCM CTI afin d'isoler plus loin la question.

## Informations connexes

- [Configurer et collecter des données de suivi dans CUE](#)
- [Dépannage des fonctionnalités du système de Cisco Unity Express](#)
- [Exemple de configuration de CallManager pour Cisco Unity Express](#)
- [Configuration de profil d'audio-messagerie CUCM](#)
- [Modèles de déploiement de Cisco Unity Express](#)
- [Pratiques recommandées pour déployer le Cisco Unity Express](#)
- [Transcodage CUCM](#)
- [CUCM MTP](#)
- [Support et documentation techniques - Cisco Systems](#)

Ce document était-il utile ? [Oui](#) [aucun](#)

Merci de votre feedback.

[Ouvrez une valise de support](#) (exige un [contrat de service Cisco](#).)

# Cisco relatif prennent en charge des discussions de la Communauté

[Cisco prennent en charge la Communauté](#) est un forum pour que vous posiez et pour répondez à des questions, des suggestions de partage, et collabore avec vos pairs.

Référez-vous au [Conventions relatives aux conseils techniques Cisco](#) pour les informations sur des conventions utilisées dans ce document.

Mis à jour : Sept 06, 2013

ID de document : 116060