

CUE JTAPI 문제 및 사례 연구

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[CUCM과의 CUE JTAPI 통합 개요](#)

[상위 레벨 통화 흐름 예](#)

[추적 지원 및 수집](#)

[실시간 JTAPI CCN 추적](#)

[JTAPI CCN 추적 로그](#)

[추적 로그 파일 수집](#)

[로그를 확인하기 전에 세부 정보가 있어야 함](#)

[기본 CTI 개념](#)

[일반 CTI 통화 상태](#)

[추적 로그의 모양](#)

[CTI RP 및 포트 등록](#)

[기본 통화가 음성 메일로 착신 전환됨](#)

[새 통화 및 사용 가능한 포트에 리디렉션](#)

[새 통화 및 사용 가능한 포트에 리디렉션](#)

[CTI 포트에 대한 새 통화](#)

[CTI 포트에서 리디렉션된 통화 수락](#)

[미디어 협상](#)

[통화 연결 해제](#)

[MWI 켜기/끄기 신호](#)

[CUE가 회선 3001에 대해 MWI 램프를 켜게 함](#)

[사서함에서 메시지를 삭제하기 위해 전화 건 DTMF 번호 '3'](#)

[CUE 회선 3001에 대해 MWI 램프를 끕니다.](#)

[실시간 CCN 로그](#)

[통화 설정](#)

[통화 연결 해제](#)

[문제 해결 사례 연구](#)

[연결 문제](#)

[사용자가 CTI를 사용할 수 없음](#)

[CUCM CTI Manager 서비스가 다운되었습니다.](#)

[구성 불일치](#)

[CUCM 통화 라우팅 문제](#)

[포트 등록 문제 해결을 위한 체크리스트](#)

[관련 정보](#)

소개

이 문서에서는 Cisco CUE(Unity Express) JTAPI(Java Telephony Application Programming Interface)의 문제 해결 방법에 대해 설명합니다. 또한 이 문서에서는 문제 해결 사례 연구 예와 함께 다양한 추적 및 로그를 활성화, 수집 및 보는 방법에 대한 정보와 명령을 제공합니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- 웹 관리 인터페이스를 통해 Cisco CUCM(Unified Communications Manager)을 구성하고 사용하는 방법에 대한 기본적인 지식
- CUCM의 CTI(Computer Telephony Interface) 포트 및 RP(Route-Points)에 대한 기본적인 지식
- Cisco Unity Express 명령줄 인터페이스에 대한 기본적인 지식

사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- Cisco Unity Express 버전 3.x 이상
- Cisco Unified Communications Manager 버전 7.x 이상

사용되는 통합 방법은 Cisco Unified Communications Manager를 사용하는 Cisco Unity Express에만 적용됩니다.CUCM(Cisco Unified Communications Manager Express)이 아닙니다.

Cisco Unity Express는 CUCM이 아니라 CUCM에 대한 라이선스가 있어야 합니다.CUE는 언제든지 CUCM 또는 CUCME와 통합될 수 있으며 그에 따라 라이선스가 부여됩니다.

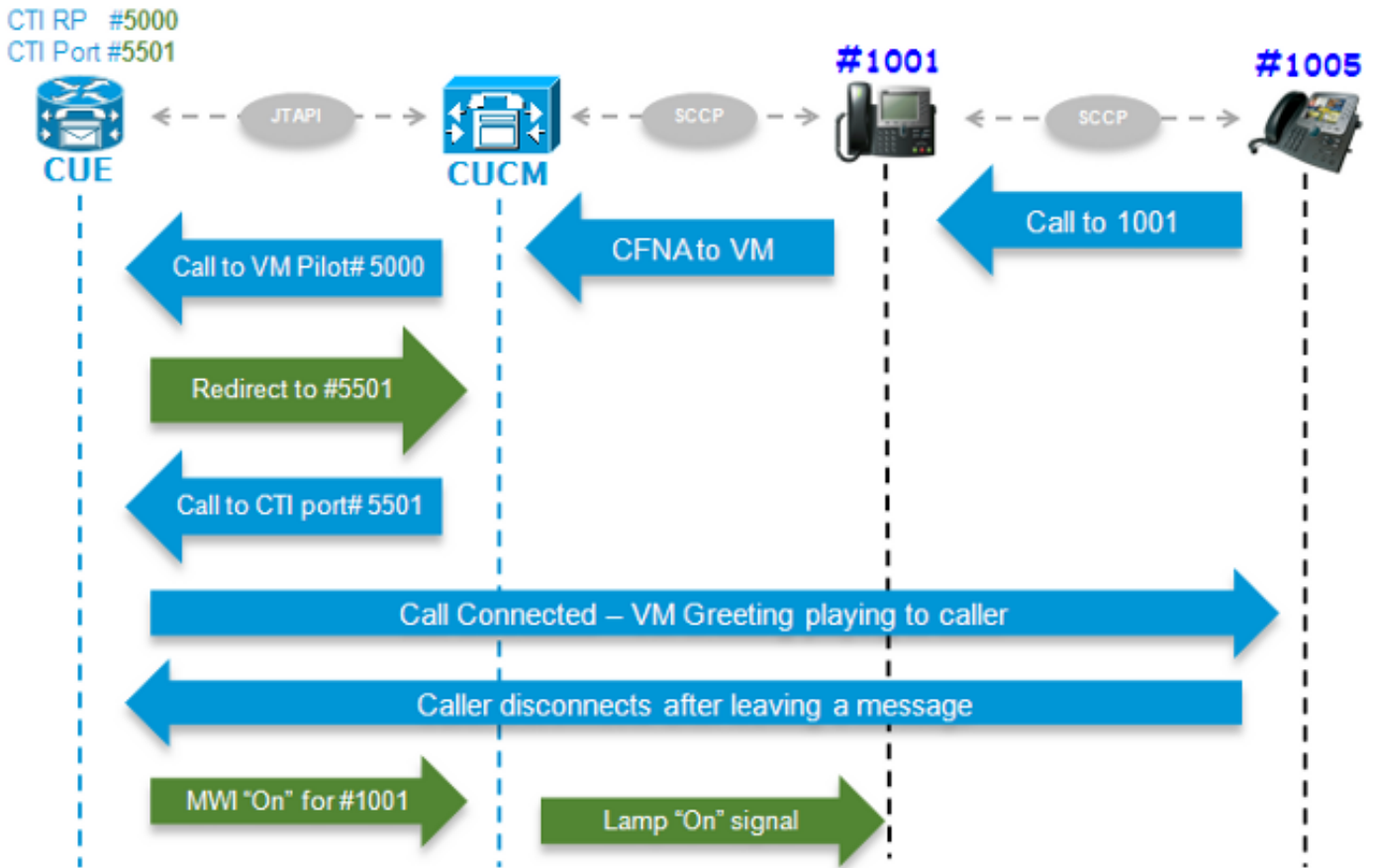
이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다.이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다.현재 네트워크가 작동 중인 경우, 모든 명령어의 잠재적인 영향을 미리 숙지하시기 바랍니다.

CUCM과의 CUE JTAPI 통합 개요

VM(Voice Mail) 및 AA(Automated Attendant) 기능을 위한 JTAPI 프로토콜을 통해 CUE를 CUCM과 통합할 수 있습니다.CUCM 서버에 등록된 사용자 수가 적은 하나 이상의 지사 사이트에 대해 VM 기능 및/또는 기본 AA 통화 처리를 프로비저닝하려는 경우 이 솔루션을 사용하는 것이 좋습니다.이를 위해 Cisco Unity Voicemail Server를 완전히 도입할 필요는 없지만 훨씬 더 경제적인 구현 방법이 필요합니다.동시에 CUE는 해당 브랜치에 대해 존속성 옵션을 제공하며 CUCM에 대한 연결이 끊기면 SIP(Session Initiated Protocol)로 장애 조치됩니다.

CUE는 JTAPI를 통해 CUCM에 등록할 수 있으며 CTI 경로 포인트 및 CTI 포트를 제어합니다.이렇게 하면 CUCM을 통해 추가 엔드포인트로 CUE를 제어하고 관리할 수 있을 뿐만 아니라 클러스터의 다른 엔드포인트와의 컨피그레이션 및 상호 작용을 촉진할 수 있습니다.

상위 레벨 통화 흐름 예



DN(Directory Number) 1005의 최종 사용자는 DN 1001을 사용하여 사용자를 호출합니다.통화가 응답하지 않으면 몇 초 후 통화가 CFNA(Call Forward No Answer)가 사용자 1001 VM 프로필에 구성된 VM 번호로 전달됩니다.그런 다음 CUCM은 구성된 VM Pilot 5000으로 통화를 전송합니다. CUE에서 제어하는 DN 5000이 있는 CTI RP를 가리킵니다.CUE VM 애플리케이션이 트리거되며, 이 통화는 JTAPI를 통해 미디어 설정을 위해 사용 가능한 CTI 포트(DN 5501)로 리디렉션됩니다.오디오 인사말이 재생되며 사용자는 DTMF(Dual Tone Multi-frequency) 신호음을 통해 메시지를 남기거나 시스템과 상호 작용할 수 있습니다.발신자가 통화를 종료하면 CUE는 JTAPI를 통해 확장 1001에 대한 MWI(Message Waiting Indicator) 램프를 "On"으로 설정하도록 CUCM에 신호를 보냅니다.그런 다음 CUCM은 전화기의 조명을 켜는 SCCP(Skinny Client Control Protocol) 메시지를 전송하고, 사용자 1001이 사서함에 새 VM 메시지가 있음을 인식하도록 디스플레이에 봉투 표시를 표시합니다.

추적 지원 및 수집

두 가지 유형의 추적이 있습니다.

- 실시간 JTAPI CCN(Cisco Communications Network) 추적
- JTAPI CCN 추적 로그

실시간 JTAPI CCN 추적

- 실시간 JTAPI CCN 추적.이러한 추적을 활성화해도 CUE 모듈을 다시 로드할 필요가 없습니다.
 - 이 출력은 CCN 추적 로그만큼 광범위하지는 않지만 매우 유익하지는 않습니다.
- 추적을 활성화하려면 다음 명령을 입력합니다.

```
no trace all
```

`trace ccn SubsystemJtapi all`

이 명령이 활성화되었는지 확인하려면 다음 명령을 입력합니다.

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

출력을 수집하려면 다음 명령을 입력합니다.

```
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 !!
```

CTRL-C를 입력하여 콘솔에 대한 실시간 로깅을 중지합니다.

JTAPI CCN 추적 로그

로그가 채워지도록 JTAPI CCN 추적 로그를 활성화한 후 CUE 모듈을 다시 로드해야 합니다. 이러한 로그, **messages.log** 및 **atrace.log**는 매우 상세하거나 암호화할 수 있으며 훨씬 더 유익하고 자세한 정보를 제공합니다. 다음과 같은 네 가지 로그가 있습니다.

- **atrace.log**

NM(Network Module)에서 기본적으로 활성화되지만 AIM(Advanced Integration Module)에서는 기본적으로 비활성화되어 있습니다. 활성화하려면 **log trace local enable** 명령을 입력합니다. 로컬 또는 FTP 서버에 최대 10Mb의 데이터를 기록합니다. 로그를 다시 시작하려면 **log trace local disable** 명령 또는 **no log trace local enable** 명령을 입력합니다. 그런 다음 **log trace local enable** 명령을 입력합니다. **atrace.log**를 지우려면 **clear trace file** 명령을 입력합니다. 데이터는 TAC(Technical Assistance Center)에서 디코딩해야 합니다.

- **메시지.로그**

Info(정보), Warning(경고), Error(오류), Fatal(치명적) 등의 Syslog 메시지가 포함된 로그입니다.

- **CiscoJtapi1.log** 및 **CiscoJtapi2.log**

모든 JTAPI 관련 신호 및 이벤트를 기록합니다. 이러한 로그는 훨씬 더 쉽게 이해할 수 있으며 매우 유익합니다. **CiscoJtapi2.log**는 **CiscoJtapi1.log**가 꽉 차거나 그 반대로 채워지면 채워집니다.

어떤 추적이 설정되었는지에 관계없이 시스템은 다시 로드 후 기본 추적 레벨로 돌아갑니다. 이러한 기본 설정을 변경하여 리부팅을 유지하려면 **log trace boot** 명령을 입력해야 합니다. 다음 명령을 사용하여 활성화합니다.

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

이 명령이 활성화되었는지 확인하려면 다음 명령을 입력합니다.

```
CUE# show ccn trace jtapi
Warning:
```

Informational:	1
Jtapi Debugging:	1
Jtapi Implementation:	1
CTI Debugging:	1
CTI Implementation:	1
Protocol Debugging:	1
Misc Debugging:	1

로그를 보는 단계는 다음과 같습니다.

1. CUE에 저장된 로그 파일 목록을 보려면 **show logs** 명령을 입력합니다.
2. **.prev** 파일 확장자는 이 파일이 현재 활성 로그 파일이 아니라 이전 추적 파일의 백업임을 의미합니다.
3. 외부 FTP 서버에 추출할 수 있습니다.
4. CUE의 터미널 모니터에서 실시간으로 이러한 파일에 로깅되는 메시지의 출력을 볼 수도 있습니다.

추적 로그 파일 수집

다음 명령을 사용하여 외부 FTP에 로그를 추출합니다.

```
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/
```

show log name <logname> 명령을 사용하여 CUE 터미널 모니터에 로그를 표시합니다. 예를 들면 다음과 같습니다.

```
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.log는 인코딩됩니다. 따라서 **show log name** 명령을 사용하여 실시간으로 표시할 수 없습니다.

로그를 확인하기 전에 세부 정보가 있어야 함

최소한 문제 해결 중인 문제와 함께 이 전화에 설명된 모든 세부 정보를 확보하여 추적을 쉽게 추적하고 이해할 수 있어야 합니다.

- 발신 번호
- 전화 번호
- 리디렉션 번호
- CTI RP DN 및 디바이스 이름
- CTI 포트 번호 및 디바이스 이름
- JTAPI 사용자
- 통화가 발생한 시간 범위

기본 CTI 개념

공급자:CTI 서비스 제공업체.응용 프로그램은 공급자를 열어 CTI 세션을 설정합니다.

사용자:애플리케이션은 사용자와 연결됩니다.

장치:CUCM에 등록하는 장치입니다.

줄:CTI 지원 디바이스의 DN 표시

통화 ID(callLegID):통화의 한 통화 레그와 연결되었습니다.

글로벌 통화(callID):단일 통화에 대한 모든 통화 레그를 식별합니다.

일반 CTI 통화 상태

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

추적 로그의 모양

잘못된 신호 처리를 찾기 전에 먼저 정상적인 작동 상태에서 이 신호 처리가 어떻게 표시되는지 알아야 합니다.따라서 이 섹션에서는 여러 시나리오에서 정상적으로 작동하는 신호 출력 일부를 보여 줍니다.

또한 이러한 로그에서의 모든 신호 처리는 관련 세부 정보만 표시하도록 요약되어 있습니다. 매우 지루하고 반복적인 세부 정보가 포함되어 있기 때문입니다.

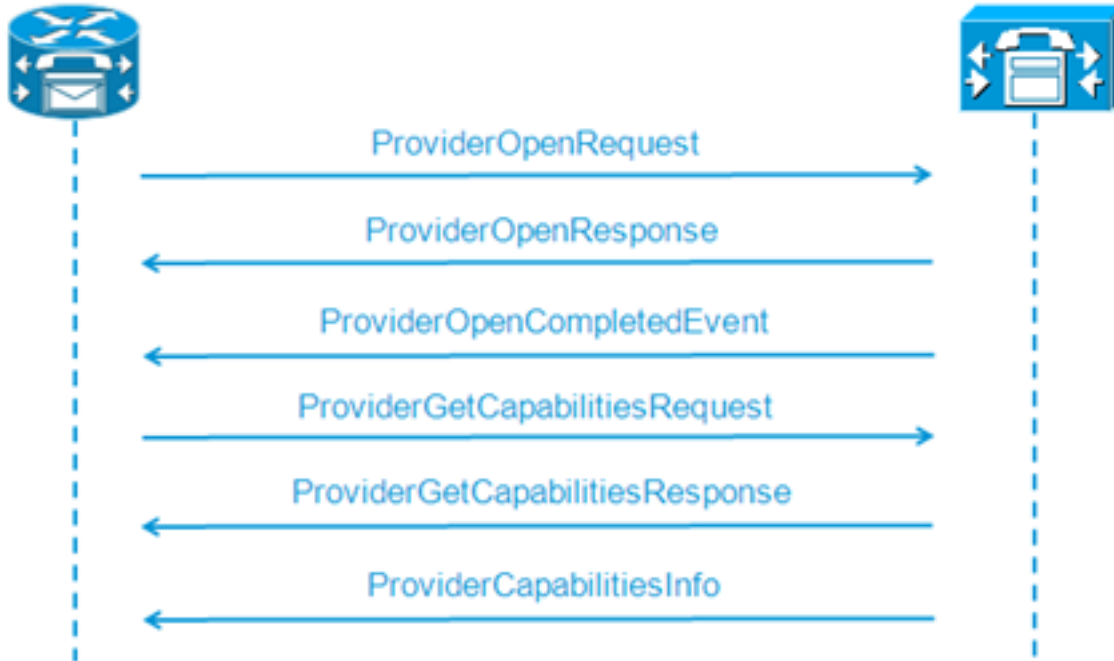
사용된 컨피그레이션의 세부 정보는 다음과 같습니다.

```
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 및 포트 등록

(CiscoJtapi1/Cisco JTAPI2 로그의 출력)

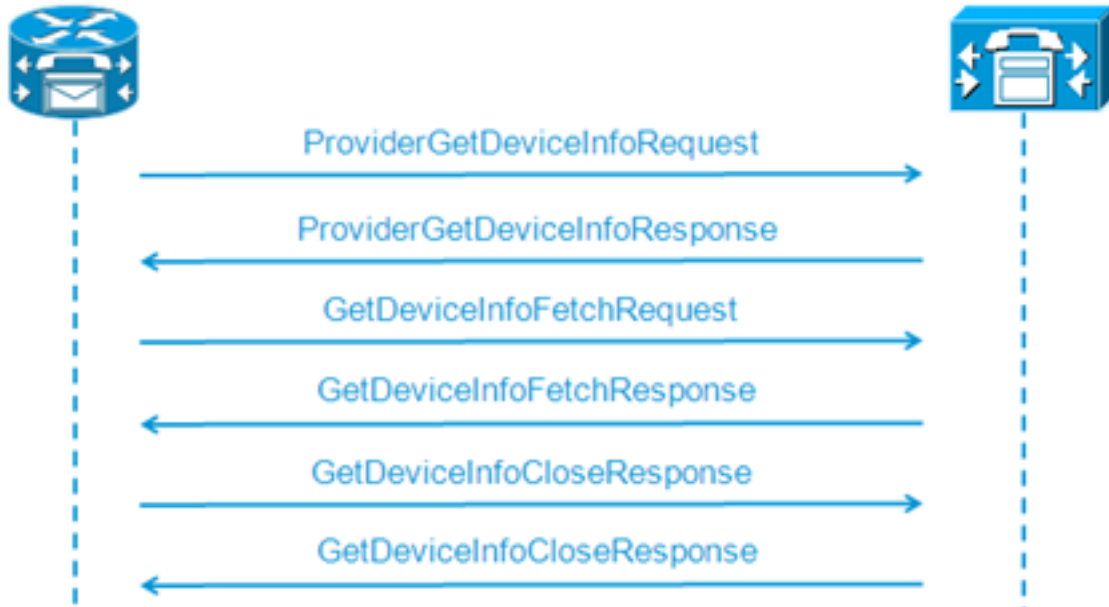
1. 공급자 연결 열기



```

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
}
  
```

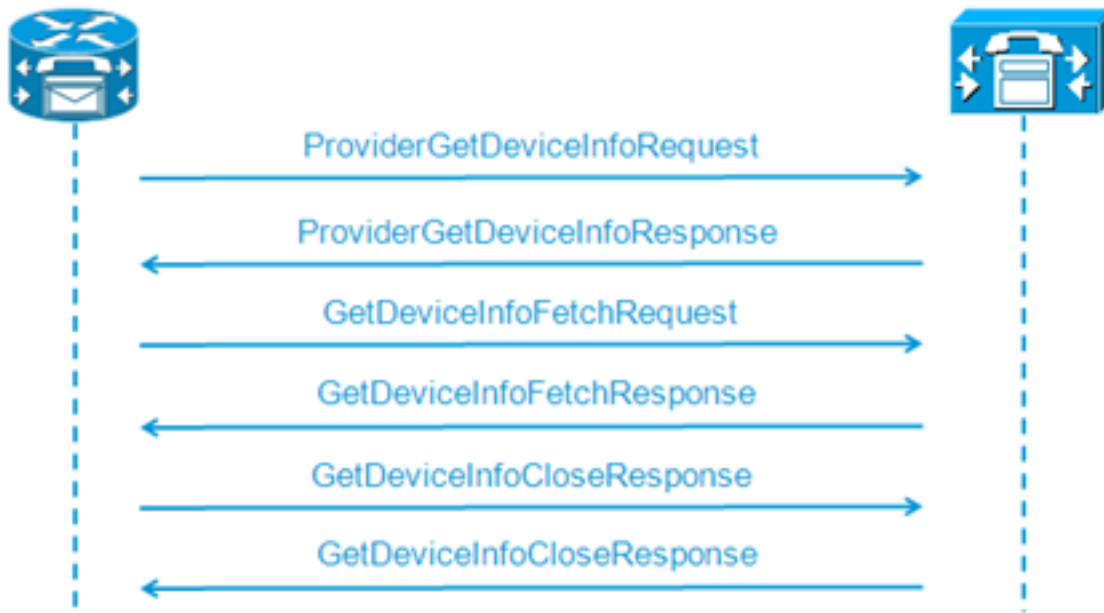
2. 제어 가능한 디바이스에 대한 쿼리



```

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


3. CTI 포트 라인 정보 가져오기



```

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. CTI RP 라인 정보 얻기

```
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. CUE가 수신된 컨피그레이션을 적용합니다.

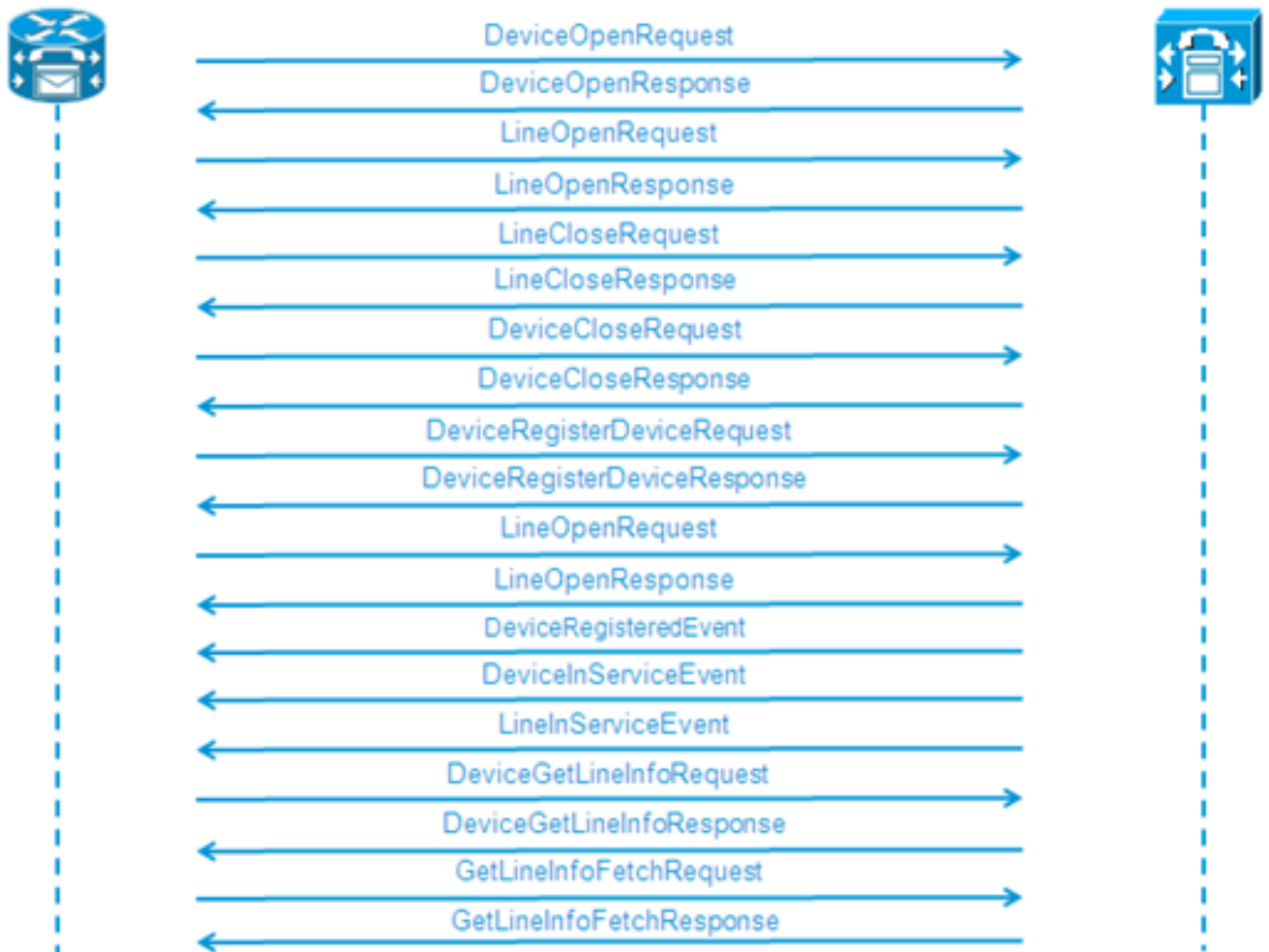
```
76: 12:05:25.064 CST %JTAPI-MISC-7-UNK.Provider 192.168.100.10
open, beginning device
initialization
77: 12:05:25.071 CST %JTAPI-JTAPI-7-UNK.(P1-tacjtapiuser)[SS_TEL_INIT]
(P1-tacjtapiuser) Request: addObserver
79: 12:05:25.073 CST %JTAPI-MISC-7-UNK.ObserverThread
(com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@3d823d82):created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



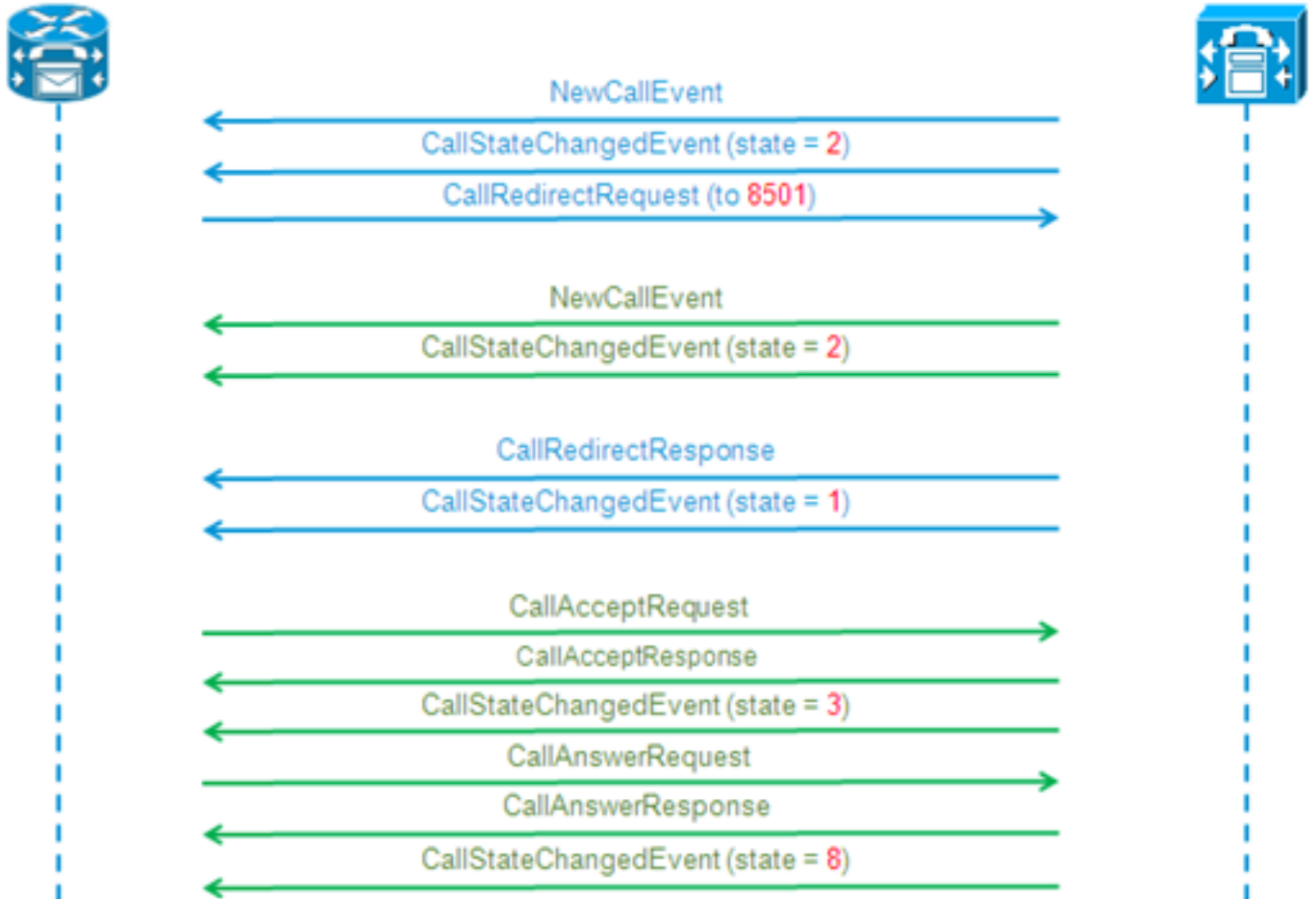
109: 12:05:25.528 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10) [SS_TEL_INIT]
sending:
com.cisco.cti.protocol.**DeviceOpenRequest** {
deviceName = **cue_ctiport1**
}
110: 12:05:25.533 CST %JTAPI-PROTOCOL-7-UNK.(P1-192.168.100.10)
received Response:
com.cisco.cti.protocol.**DeviceOpenResponse** {
result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

기본 통화가 음성 메일로 착신 전환됨

(CiscoJtapi1/Cisco JTAPI2 로그의 출력)

새 통화 및 사용 가능한 포트로 리디렉션



새 통화 및 사용 가능한 포트로 리디렉션

```
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
}

```

CTI 포트에 대한 새 통화

```

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
}

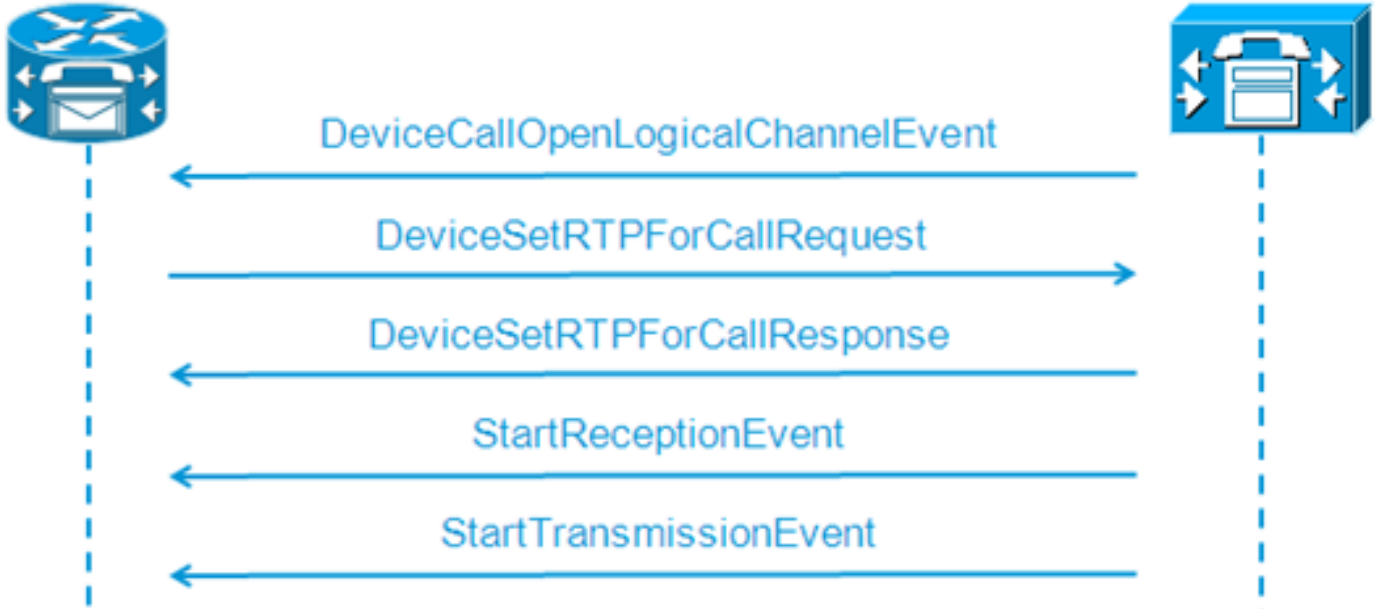
```

```
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 포트에서 리디렉션된 통화 수락

```
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
```

미디어 협상

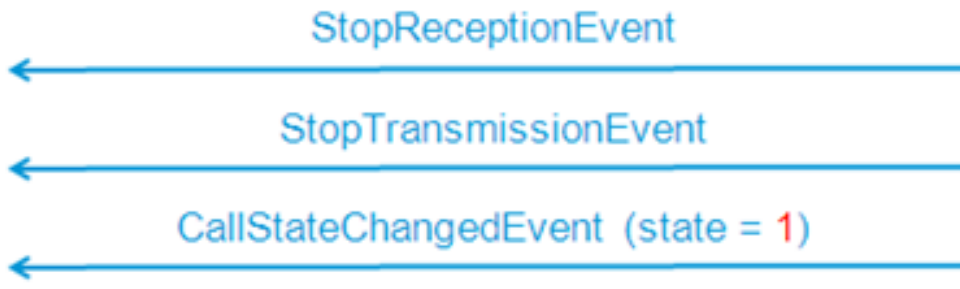


```

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
}

```

통화 연결 해제



```

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 켜기/끄기 신호

CUE가 회선 3001에 대해 MWI 램프를 켜게 함

```

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
}
  
```

사서함에서 메시지를 삭제하기 위해 전화 건 DTMF 번호 '3'

```

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 회선 3001에 대해 MWI 램프를 끕니다.

```
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
}
```

실시간 CCN 로그

(실시간 CCN 로그의 출력)

이것은 실시간 CCN 로그가 대신 수집될 때 이전 예제의 동일한 통화가 나타나는 방법입니다.

통화 설정

```
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
```

통화 연결 해제

```
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
```

문제 해결 사례 연구

연결 문제

이 시나리오에서는 CUE와 CUCM 간의 연결이 부족하여 CUE 포트와 트리거가 CUCM에 등록되지 않습니다.

```
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 --

```

참고:타임스탬프 초는 13:20:28에서 13:20:38로 이동합니다.따라서 CUE가 TCP 소켓을 10초 동안 열 수 없었음을 알 수 있습니다.

인증 문제

이 시나리오에서는 CUE와 CUCM 간에 구성된 비밀번호가 일치하지 않으므로 CUE 포트 및 트리거가 CUCM에 등록되지 않습니다.

CCN 로그

```

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
```

사용자가 CTI를 사용할 수 없음

이 시나리오에서는 JTAPI 응용 프로그램 사용자가 CUCM 측의 표준 CTI 사용 권한 그룹에 추가되지 않았으므로 CUE 포트 및 트리거가 CUCM에 등록되지 않습니다. 따라서 사용자 자격 증명이 그에 따라 인증될 경우에도 JTAPI 사용자, **tacjtapiuser**는 CTI 및 JTAPI를 통해 어떤 디바이스도 제어할 수 없습니다.

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 Manager 서비스가 다운되었습니다.

이 시나리오에서는 CUCM CTI Manager 서비스가 다운되었거나 비정상적인 상태이므로 CUE 포트

및 트리거를 등록할 수 없습니다.JTAPI TCP 포트 2748에 대한 CUE의 연결 시도에 대해 "연결 거부" 오류가 수신됩니다.

```
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
```

구성 불일치

이 시나리오에서 CTI RP가 일치해야 하는 CTI RP가 구성되지 않았거나 CUCM 측의 사용자에 대해 "제어 가능한 디바이스"에 추가되지 않았으므로 CUE는 번호 9999로 JTAPI 트리거를 등록할 수 없습니다.CUE는 CUCM에서 GetDeviceInfoFetchResponse를 수신하고 공급자 도메인에 장치가 없음을 인식합니다. 이 도메인은 해당 사용자가 제어하는 모든 디바이스를 참조하며, 이는 로컬에서 구성한 트리거 번호와 일치합니다.그런 다음 CUE는 해당 특정 트리거에 대해 **DeviceOpenRequest**를 전송하려고 시도하지 않고 대신 추적의 예외만 보고합니다.CUE는 여전히 CUCM에서 전송하는 공급자의 도메인 내에 있는 다른 모든 디바이스를 등록하려고 시도합니다.

```
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.
```

참고:트리거 9999가 CUE에서 로컬로 구성된 경우에도 CUCM에서 받은 공급자 도메인의 일부가 아니므로 등록되지 않습니다.

CUE는 계속 8000선을 엽니다.제공 기관의 도메인에 포함된

```
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 통화 라우팅 문제

이 시나리오에서는 DN 3001을 사용하는 사용자가 CUE를 호출하여 VM을 확인합니다.이 통화는 DN 8000을 사용하는 CTI RP(CUE VM 파일럿)에 표시됩니다.그런 다음 CUE는 DN 8501을 사용하여 해당 미디어 CTI 포트에 리디렉션되도록 통화를 요청하지만 DN 3001에 대해 구성된 CSS가 CTI 포트의 DN이 할당된 PT에 액세스할 수 없기 때문에 통화가 리디렉션되지 않습니다.

```
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

```

라이선스 문제

이 시나리오에서는 VM 포트에 대한 라이선스가 활성화되지 않았기 때문에 CUE가 포트와 트리거를 등록할 수 없습니다. 동일한 이유로 인해 캡처에 등록 시도가 표시되지 않습니다.

디코딩된 atrace.log의 요약:

```

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:

```

```

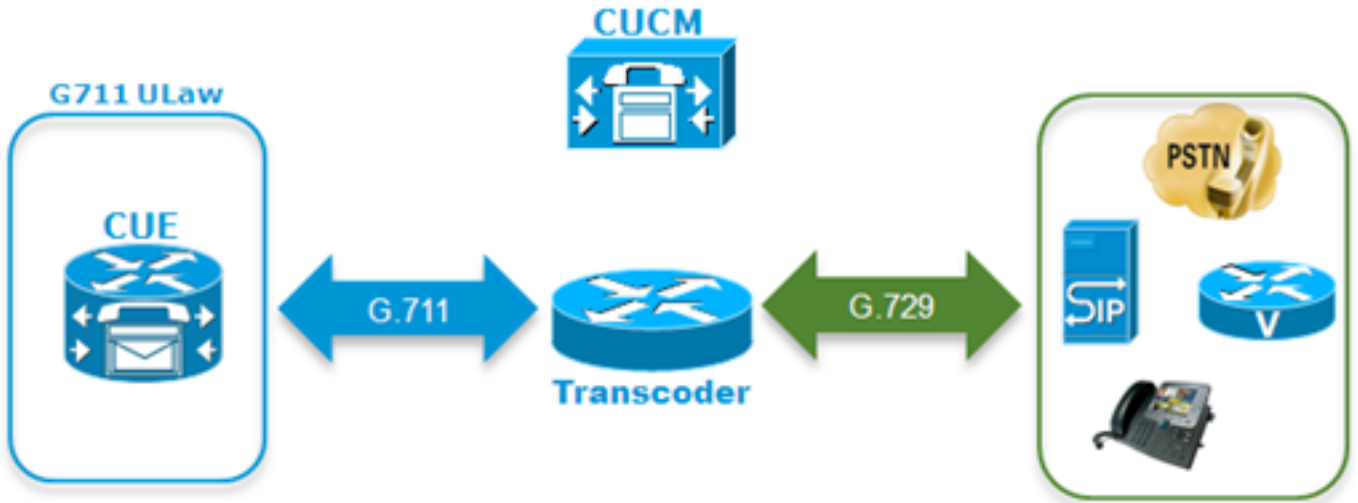
192.168.100.10;login=tacjtapiuser ;passwd=cisco;appinfo=Cisco IP IVR
3477 11:45:31.330 ACCN TELS 0 Got JTAPI provider: Cisco Jtapi version
7.0(1.1000)-1 Release
3621 11:45:31.338 ACCN TELS 0 JTAPI_PROVIDER_EVENT:JTAPI Provider
state is changed: JTAPI provider name=192.168.100.10,Event=
ProvOutOfServiceEv 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

```

모범 사례

CUE는 G711ulaw 코덱만 지원합니다.따라서 거의 모든 구축에서 CUE가 다른 코덱을 사용하는 다른 디바이스 또는 트렁크와 통신하려면 트랜스코더가 필요합니다(G711Alaw 포함). MTP(Media Termination Point) 리소스도 필요한 대역 내 DTMF만 지원하는 디바이스와의 DTMF 상호 작용에도 동일하게 적용됩니다.이러한 제한 때문에 Cisco는 다음을 권장합니다.

- CUE의 CTI RP 및 CTI 포트만 사용하도록 격리된 디바이스 풀을 생성합니다.둘 이상의 CUE가 CUCM과 통합된 경우 CUE당 디바이스 풀을 생성합니다.
- CUE의 RP 및 포트에 대해서만 별도의 영역을 생성하고 격리된 디바이스 풀에 적용합니다.
- 해당 지역이 다른 모든 지역에서 G711만 허용하도록 구성되었는지 확인합니다.
- 트랜스코딩 리소스를 사용할 수 있는 MRGL(Media Resource Group List)이 CUE의 CTI RP 및 CTI 포트의 디바이스 풀에 적용되어 필요한 경우 트랜스코더 리소스에 액세스할 수 있도록 해야 합니다.



- 사용자가 DTMF 신호음이 있는 음성 메뉴를 탐색할 수 없는 경우 MTP 리소스를 CUE 장치의 MRGL에 추가해야 할 수 있습니다.

CUCM에서 CUE에 대한 별도의 VM 프로파일 생성

CUCM CTI Manager에서 관찰된 몇 가지 최근 문제를 방지하려면 CTI RP 및 포트만 사용하는 대신 모든 전화기를 CUCM 측의 CUE JTAPI 사용자에게 연결하는 것이 좋습니다.

SRST(Survivable Remote Site Telephony) 기능이 필요한 경우:

- CUE의 각 JTAPI 트리거에 대해 해당 SIP 트리거가 구성되었는지 확인합니다.
- SRST 모드에서 SIP를 통해 통화를 CUE 모듈로 라우팅할 수 있도록 다이얼 피어가 브랜치 라우터에 추가되었는지 확인합니다.
- CUCM에서 CUCM과 CUE 간의 연결이 끊겼거나 AAR(Automated Alternate Routing)이 호출될 때 CUCM이 로컬 PSTN(Public Switched Telephone Network) 게이트웨이를 통해 지사 모듈에 대한 통화를 라우팅하도록 하려면 CUCM의 CTI Route Points(Call Forward Unregistered) 필드의 마스크와 외부 번호 마스크를 구성합니다. 브랜치 라우터에서 PSTN에서 CUE 모듈로 인바운드 통화를 라우팅하려면 추가 변환 규칙이 필요할 수 있습니다.
- CUCM 측에 VM으로 직접 전송 컨피그레이션 접근 방식이 있고 사용자가 CME-SRST에서 이 기능을 유지하려는 경우 TransferToVM 소프트웨어를 사용하기 전에 CFA(Call Forward All) 컨피그레이션과 함께 이전 더미-DN을 사용해야 합니다. 자세한 내용은 [발신자를 Unity Express 사서함으로 직접](#) 호전환을 참조하십시오. 이것이 어떻게 보일 수 있는지를 보여주는 예가 여기에 있다. 이 작업은 CME-SRST가 사용되고 call-manager-fallback이 있는 레거시 SRST가 아닌 경우에만 수행할 수 있습니다. DN이 200~299 범위에 있다고 가정합니다. x201에 대한 통화가 수신됩니다. 다음 명령을 사용하여 ephone-dn을 구성합니다.

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

CUE를 가리키는 다이얼 피어: 아웃바운드 변환 규칙 및 프로필을 사용하여 미리 고정된 별표 ("*")를 제거하고 RDNIS(Redirect Dialed Number Information Service)를 원래 3자리 번호(예: 201)로 다시 바꾸거나, CUE 내에서 전화 번호가 전체 DID로 구성된 경우 전체 E.164 번호로 다시 바꿉니다. CUE로 전송된 INVITE의 Diversion 헤더가 CUE 측에 있는 사용자에게 대해 구성된 전화 번호와 일치하는지 확인합니다.

포트 등록 문제 해결을 위한 체크리스트

1. CUCM 측에서 컨피그레이션을 확인합니다. CTI Manager, CallManager 및 AXL(Administrative XML) 웹 서비스가 활성화되고 시작됩니까? CTI 포트 및 경로 포인트가 구성되어 고유한 DN을 할당했습니까? JTAPI 사용자 CTI가 활성화되고 AXL API 액세스가 가능합니까? JTAPI 사용자는 모든 CTI 경로 포인트 및 포트를 제어합니까? 구성을 추가한 후 모든 서버에서 CTI Manager 서비스를 다시 시작하는 것이 좋습니다. 그러나 UCCX(Unified Contact Center Express), IP Manager Assistant(IPMA), Attendant Console, 타사 AA 또는 ACD 애플리케이션 등 CUCM과 함께 CTI 및 JTAPI를 사용하는 다른 모든 장치에 영향을 주므로 이는 서비스에 영향을 주고 유지 관리 기간을 예약하는 것이 좋습니다.
2. CUE 측에서 컨피그레이션을 확인합니다. 통화 에이전트가 CUCM으로 정의됩니까? 포트 라이선스가 활성화되었습니까? 평가판 라이선스는 초기 컨피그레이션에 사용할 수 있습니다. CUCM을 ping할 수 있습니까? JTAPI 사용자 자격 증명이 올바르게 추가되었으며 통화 에이전트가 정의되었습니까? 컨피그레이션 변경 사항이 적용되도록 모듈을 다시 로드했습니까? CTI RP 및 포트를 CUCM에서 자동으로 가져오지 않는 경우 ccn 하위 시스템 jtapi에 포트 DN을 수동으로 추가하고, 각 CTI-RP에 대한 jtapi 트리거를 추가하고 모듈을 다시 로드합니다.

이러한 모든 항목이 확인되면 다음 단계는 CUE에서 JTAPI 추적을 얻고, CUCM CTI 추적을 통해 문제를 추가로 격리하는 것입니다.

관련 정보

- [CUE에서 추적 데이터 설정 및 수집](#)
- [Cisco Unity Express System 기능 문제 해결](#)
- [CallManager for Cisco Unity Express 컨피그레이션 예](#)
- [CUCM Voice-Mail 프로파일 컨피그레이션](#)
- [Cisco Unity Express 구축 모델](#)

- [Cisco Unity Express 구축을 위한 모범 사례](#)
- [CUCM 트랜스코딩](#)
- [CUCM MTP](#)
- [기술 지원 및 문서 - Cisco Systems](#)