Troubleshooting Unity Express Message Waiting Indication (MWI) Problems

Document ID: 60081

Contents

Introduction

This document provides an overview of Message Waiting Indication (MWI) functionality in Cisco Unity Express.

Prerequisites

Requirements

Readers of this document should have knowledge of Cisco Unity Express command–line interface (CLI).

Components Used

The information in this document is based on Cisco Unity Express version 1.0/2.3.x/8.x or later. All sample configurations and screen output are taken from Cisco Unity Express version 1.1.1.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

MWI Overview

The MWI operation provides users who are registered with Cisco CallManager Express or CallManager with a visual indication that there are new voice–mail messages present. MWI does not work when you have Cisco Unity Express integrated with Cisco CallManager and the system is in Survivable Remote Site Telephony

(SRST) mode because of a WAN outage.

If you have Cisco Unity Express integrated with Cisco CallManager Express, a session initiation protocol (SIP) call is placed to the *extension_MWI_on/off_number@CallManager_Express_IP_address* when a new voice mail message arrives into a user mailbox. The SIP call also initiates when the user retrieves all new messages. This matches an ephone–dn number on the Cisco CallManager Express router. The ephone–dn number has the MWI number plus a number of wildcard digits equal to the number of digits in the extension of the Cisco Unity Express subscribers. As an example, suppose that the MWI–on number for mailbox 12345 is 420. The Cisco CallManager Express IP address is 10.2.3.6. In this example, the message is sent to 42012345@10.2.3.6. The ephone–dn number with the "mwi on" configuration parameter set is "420...".

For Cisco CallManager integrations, the Java Telephony Application Programming Interface (JTAPI) protocol lights a lamp directly. There is no need to place a call to a specific number. The JTAPI protocol itself supports a **setMessageWaiting** command, which handles MWI events. Therefore, MWIs should work regardless of whether MWI extensions are configured in the Cisco CallManager. Remember that MWIs do not work when Cisco Unity Express is in SRST mode. A complete MWI refresh only occurs after Cisco Unity Express reregisters with Cisco CallManager and the IP phones are no longer in CallManager fallback mode.

The vast majority of problems occur with integration between Cisco CallManager Express/CallManager and Cisco Unity Express. Keep in mind that MWI can possibly not correlate to a physical lamp. If the number that receives the message is not a primary line on a phone, it can only receive an envelope notification on the phone display. In Cisco CallManager, you can configure how each line handles MWI. If only one or two users have a problem, you can begin to look for the problem here.

A directory number must have a valid mailbox on the Cisco Unity Express system in order to receive an MWI. The number must be associated to a user, and that user must have a mailbox. Before you begin to debug and take advanced measures in order to troubleshoot, you can perform one simple task in order to troubleshoot: ensure that the user has logged into the mailbox and can send and retrieve voice mail messages.

From the GUI or CLI, you can find a user with which to test. In this case, it is user3. You can locate the configured extension for the user, determine the user mailbox status (enabled or not enabled, among other information), and determine if the user has any new or old messages. In this example, you utilize the CLI in order to troubleshoot:

```
cue-3660-41a>show users
administrator
operator
user1
user2
user3
user4
user6
user7
user8
cue-3660-41a>show user detail username user3
Full Name:        user
First Name:
Last Name:        user
Nickname:         user
Phone:            11044
Phone(E.164):
Language:         en_US
cue-3660-41a>show voicemail mailboxes
OWNER                          MSGS NEW   SAVED   MSGTIME MBXSIZE   USED
"operator"                     0    0     0       0       3000      0 %
"user1"                        0    0     0       0       3000      0 %
"user2"                        0    0     0       0       3000      0 %
"user3"                        0    0     0       0       3000      0 %
"user4"                        0    0     0       0       3000      0 %
```

```
"user6"                                           0    0    0    0       3000      0 %
"user7"                                           0    0    0    0       3000      0 %
"user8"                                           0    0    0    0       3000      0 %
cue-3660-41a>show voicemail detail mailbox user3
Owner:                                /sw/local/users/user3
Type:                                 Personal
Description:
Busy state:                           idle
Enabled:                              true
Mailbox Size (seconds):               3000
Message Size (seconds):               60
Play Tutorial:                        true
Space Used (seconds):                 0
Total Message Count:                  0
New Message Count:                    0
Saved Message Count:                  0
Expiration (days):                    30
Greeting:                             standard
Zero Out Number:
Created/Last Accessed:                Jun 17 2004 09:54:39 EDT
cue-3660-41a>
```

Verify that this user exists, has a number associated, and does not have any messages. If these items are true, the MWI status should be off.

Note: The E.164 (ITU–T) address is not used for MWI purposes. Only the primary phone number can be used.

Cisco Unity Express Integration Problems

MWI with Cisco CallManager Express

You must verify the configuration before you do anything else. On the Cisco CallManager Express, view the configuration with the issue of the **show running–config** command. More direct, you can issue the **show telephony–service ephone–dn** command. An output similar to this appears:

```
ephone-dn  44
 number 11099.....
 mwi on
!
!
ephone-dn  45
 number 11098.....
 mwi off
!
```

This output illustrates some important information. The number for MWI on is 11099. The number for MWI off is 11098. The number of digits in the dial plan is five. (The five dots [. . . . .] that follow the MWI on or off code show this.) In other words, MWI only works for a directory number (DN) that contains exactly five digits.

On the Cisco Unity Express side, you can verify the configuration and also the license. A common problem is that a Cisco CallManager license is loaded instead of a license for CallManager Express. Issue the **show software licenses** command from the Cisco Unity Express in order to verify this:

```
cue-3660-41a>show software licenses
Core:e
 - application mode: CCME

!--- CCME represents Cisco CallManager Express.
```

```
      - total usable system ports: 8
     Voicemail/Auto Attendant:
      - max system mailbox capacity time: 6000
      - max general delivery mailboxes: 20
      - max personal mailboxes: 100
     Languages:
      - max installed languages: 1
      - max enabled languages: 1
```

If you find, instead, that the application mode is CCM, Cisco CallManager, everything works *except* MWI. Unfortunately, if the license is wrong, the only option is to reimage the software and reapply the license. You cannot save or restore any messages or configuration.

Next, verify the configuration. You can view the configuration itself with the **show run** command, or you can use the **show ccn application** command:

```
cue-3660-41a> show ccn application
Name:                           ciscomwiapplication
Description:                    ciscomwiapplication
Script:                         setmwi.aef
ID number:                      0
Enabled:                        yes
Maximum number of sessions:     4
strMWI_OFF_DN:                  11098
strMWI_ON_DN:                   11099
CallControlGroupID:             0
```

Note: The application is enabled and the MWI_OFF and MWI_ON numbers are 11098 and 11099, respectively. The system does not have a concept of the number of digits in the extensions; it simply places a call to the appropriate MWI on or off number and appends the mailbox extension. The Cisco CallManager Express system must have a dial peer with the appropriate number of dots in the destination pattern in order to route the call properly.

Finally, be sure that the Cisco Unity Express SIP gateway IP address points to the correct Cisco CallManager Express IP address.

```
cue-3660-41a>show ccn subsystem sip
SIP Gateway:                    14.80.227.125
SIP Port Number:                5060
```

If this is incorrect, the calls are not sent to the correct Cisco CallManager Express. The calls fail.

There are two ways to begin to troubleshoot signaling problems. From the Cisco Unity Express side, it is usually easiest to disable the default traces first; then, reenable them as needed. Issue the **no trace all** command in order to do this. The trace command to start with is **trace ccn stacksip dbug**.

Note: Refer to the document Set Up and Gather Trace Data in CUE for more information about tracing.

Before you send an MWI message, clear the trace buffer. All trace messages write to this memory buffer. You want to clear it so that there is no need to display all previous messages when you look at it after the test call. A simple **clear trace** command accomplishes this.

Next, send the MWI message. Use the **mwi refresh telephonenumber** *xxxx* command to do this. You can issue refreshes from the GUI as well.

Finally, display the trace buffer and view the output with the **show trace buffer long** command. This example highlights some important items:

```
cue-3660-41a>trace ccn stacksip dbug
cue-3660-41a>clear trace
cue-3660-41a>mwi refresh telephonenumber 11043
cue-3660-41a>show trace buffer long
Press <CTRL-C> to exit...
2106 07/14 14:28:27.263 ACCN SIPL 0 --- send message --- to 14.80.227.125:5060
INVITE sip:1109811043@14.80.227.125;user=phone SIP/2.0
Via: SIP/2.0/UDP 14.80.227.145:5060
From: "Cisco SIP Channel3" <sip:outbound-0@14.80.227.125>;tag=f0a4ab8e-488
To: <sip:1109811043@14.80.227.125;user=phone>
Call-ID: a1c0ece2-486@14.80.227.145:5060
CSeq: 51 INVITE
Contact: sip:outbound-0@14.80.227.145:5060
User-Agent: Jasmin UA / ver 1.1
Accept: application/sdp
Content-Type: application/sdp
Content-Length: 224

v=0
o=CiscoSystemsSIP-Workflow-App-UserAgent 3582 3582 IN IP4 14.80.227.145
s=SIP Call
c=IN IP4 14.80.227.145
t=0 0
m=audio 16902 RTP/AVP 0 111
a=rtpmap:0 pcmu/8000
a=rtpmap:111 telephone-event/8000
a=fmtp:111 0-11

2069 07/14 14:28:27.275 ACCN SIPL 0 receive 379 from 14.80.227.125:51955
2070 07/14 14:28:27.275 ACCN SIPL 0 not found header for Date
2070 07/14 14:28:27.275 ACCN SIPL 0 not found header for Allow-Events
2070 07/14 14:28:27.276 ACCN SIPL 0 -------
SIP/2.0 100 Trying
Via: SIP/2.0/UDP 14.80.227.145:5060
From: "Cisco SIP Channel3" <sip:outbound-0@14.80.227.125>;tag=f0a4ab8e-488
To: <sip:1109811043@14.80.227.125;user=phone>;tag=5FF5244-43A
Date: Sat, 15 Jun 2002 13:33:41 GMT
Call-ID: a1c0ece2-486@14.80.227.145:5060
Server: Cisco-SIPGateway/IOS-12.x
CSeq: 51 INVITE
Allow-Events: telephone-event
Content-Length: 0

2069 07/14 14:28:27.276 ACCN SIPL 0 receive 441 from 14.80.227.125:51955
2070 07/14 14:28:27.294 ACCN SIPL 0 not found header for Date
2070 07/14 14:28:27.294 ACCN SIPL 0 not found header for Allow-Events
2070 07/14 14:28:27.294 ACCN SIPL 0 -------
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP 14.80.227.145:5060
From: "Cisco SIP Channel3" <sip:outbound-0@14.80.227.125>;tag=f0a4ab8e-488
To: <sip:1109811043@14.80.227.125;user=phone>;tag=5FF5244-43A
Date: Sat, 15 Jun 2002 13:33:41 GMT
Call-ID: a1c0ece2-486@14.80.227.145:5060
Server: Cisco-SIPGateway/IOS-12.x
CSeq: 51 INVITE
Allow: UPDATE
Allow-Events: telephone-event
Contact: <sip:1109811043@14.80.227.125:5060>
Content-Length: 0

2072 07/14 14:28:27.294 ACCN SIPL 0 ignore null remote tag for Dialog1610: callid=
 a1c0ece2-486@14.80.227.145:5060, localTag=f0a4ab8e-488, remoteTag=5FF5244-43A
2072 07/14 14:28:27.294 ACCN SIPL 0 ltp95: ContactingState processResponse 100 Trying
2072 07/14 14:28:27.294 ACCN SIPL 0 ignore null remote tag for Dialog1611: callid=
 a1c0ece2-486@14.80.227.145:5060, localTag=f0a4ab8e-488, remoteTag=5FF5244-43A
2072 07/14 14:28:27.294 ACCN SIPL 0 ltp95: ContactingState processResponse 180 Ringing
```

```
2106 07/14 14:28:32.274 ACCN SIPL 0 ltp95: ContactingState close terminate cause=20
2106 07/14 14:28:32.275 ACCN SIPL 0 addHeadersAndBody: branch = null
2106 07/14 14:28:32.276 ACCN SIPL 0 --- send message --- to 14.80.227.125:5060
CANCEL sip:1109811043@14.80.227.125;user=phone SIP/2.0
Via: SIP/2.0/UDP 14.80.227.145:5060
From: "Cisco SIP Channel3" <sip:outbound-0@14.80.227.125>;tag=f0a4ab8e-488
To: <sip:1109811043@14.80.227.125;user=phone>
Call-ID: a1c0ece2-486@14.80.227.145:5060
CSeq: 51 CANCEL
Max-Forwards: 50
Content-Length: 0

2069 07/14 14:28:32.282 ACCN SIPL 0 receive 293 from 14.80.227.125:51955
2070 07/14 14:28:32.283 ACCN SIPL 0 not found header for Date
2070 07/14 14:28:32.283 ACCN SIPL 0 -------
SIP/2.0 200 OK
Via: SIP/2.0/UDP 14.80.227.145:5060
From: "Cisco SIP Channel3" <sip:outbound-0@14.80.227.125>;tag=f0a4ab8e-488
To: <sip:1109811043@14.80.227.125;user=phone>
Date: Sat, 15 Jun 2002 13:33:46 GMT
Call-ID: a1c0ece2-486@14.80.227.145:5060
Content-Length: 0
CSeq: 51 CANCEL

2072 07/14 14:28:32.283 ACCN SIPL 0 ignore null remote tag for Dialog1612: callid=
 a1c0ece2-486@14.80.227.145:5060, localTag=f0a4ab8e-488, remoteTag=null
2072 07/14 14:28:32.283 ACCN SIPL 0 ltp95: TerminatedState process response to CANCEL,
 unregister
2072 07/14 14:28:32.284 ACCN SIPL 0 ignore null remote tag for Dialog1609: callid=
 a1c0ece2-486@14.80.227.145:5060, localTag=f0a4ab8e-488, remoteTag=null
2072 07/14 14:28:32.284 ACCN SIPL 0 com.cisco.jasmin.impl.sip.MessageDispatcherImpl
 unregister Dialog1609: callid=a1c0ece2-486@14.80.227.145:5060, localTag=f0a4ab8e-488,
 remoteTag=null
2069 07/14 14:28:32.284 ACCN SIPL 0 receive 390 from 14.80.227.125:51955
2070 07/14 14:28:32.284 ACCN SIPL 0 not found header for Date
2070 07/14 14:28:32.284 ACCN SIPL 0 not found header for Allow-Events
2070 07/14 14:28:32.284 ACCN SIPL 0 -------
SIP/2.0 487 Request Cancelled
Via: SIP/2.0/UDP 14.80.227.145:5060
From: "Cisco SIP Channel3" <sip:outbound-0@14.80.227.125>;tag=f0a4ab8e-488
To: <sip:1109811043@14.80.227.125;user=phone>;tag=5FF5244-43A
Date: Sat, 15 Jun 2002 13:33:46 GMT
Call-ID: a1c0ece2-486@14.80.227.145:5060
Server: Cisco-SIPGateway/IOS-12.x
CSeq: 51 INVITE
Allow-Events: telephone-event
Content-Length: 0

2072 07/14 14:28:32.285 ACCN SIPL 0 LocalLineImpl outbound-0 send ACK to INVITE 487
2072 07/14 14:28:32.285 ACCN SIPL 0 can not extract contact address from null
2072 07/14 14:28:32.285 ACCN SIPL 0 --- send message --- to 14.80.227.125:5060
ACK sip:1109811043@14.80.227.125;user=phone SIP/2.0
Via: SIP/2.0/UDP 14.80.227.145:5060
From: "Cisco SIP Channel3" <sip:outbound-0@14.80.227.125>;tag=f0a4ab8e-488
To: <sip:1109811043@14.80.227.125;user=phone>;tag=5FF5244-43A
Call-ID: a1c0ece2-486@14.80.227.145:5060
CSeq: 51 ACK
Max-Forwards: 50
Content-Length: 0
```

As seen in this output, you send an INVITE message, and Cisco CallManager Express responds with a
Trying message. As soon as Cisco CallManager Express sends a Ringing message, you send a CANCEL
message. The MWI number does not actually pick up and take a call. The placement of a call to the number
itself is enough to light the lamp on or off. In this case, you need to know if 11098 has MWI on or off. Also,
11043 needs to be a valid extension in Cisco CallManager Express.

After you collect all the necessary Cisco Unity Express traces, the best thing to do is to disable all traces and then enable the default traces again. Issue the **clear trace all** command to disable traces. Then paste the code shown here into the Cisco Unity Express CLI in order to reenable all of the default traces:

Note: Alternatively, you can restore default traces if you restart the Cisco Unity Express.

```
trace ccn engine dbug
trace ccn libldap dbug
trace ccn subsystemappl dbug
trace ccn managerappl dbug
trace ccn managerchannel dbug
trace ccn subsystemjtapi dbug
trace ccn subsystemsip dbug
trace ccn stacksip dbug
trace ccn subsystemhttp dbug
trace ccn vbrowsercore dbug
trace ccn subsystemcmt dbug
trace ccn libmedia dbug
trace ccn managercontact dbug
trace ccn stepcall dbug
trace ccn stepmedia dbug
trace config-ccn sip-subsystem debug
trace config-ccn jtapi-subsystem debug
trace config-ccn sip-trigger debug
trace config-ccn jtapi-trigger debug
trace config-ccn http-trigger debug
trace config-ccn group debug
trace config-ccn application debug
trace config-ccn script debug
trace config-ccn prompt debug
trace config-ccn miscellaneous debug
trace voicemail database query
trace voicemail database results
trace voicemail database transaction
trace voicemail database connection
trace voicemail database execute
trace voicemail mailbox login
trace voicemail mailbox logout
trace voicemail mailbox send
trace voicemail mailbox save
trace voicemail mailbox receive
trace voicemail mailbox delete
trace voicemail message create
trace voicemail message dec
trace voicemail message delete
trace voicemail message get
trace voicemail message inc
trace webinterface initwizard init
```

You can also easily diagnose all the SIP messaging on the Cisco CallManager Express router itself. Usually, **debug ccsip messages** and **debug ccsip media** are the most useful commands. When only SIP signaling is necessary, this diagnosis is much quicker, and the Cisco Unity Express traces less unnecessary information. If the Cisco Unity Express sends the signaling to the correct CallManager Express IP address, the SIP signaling is mirrored on each server.

Calls to or from Cisco Unity Express require G.711, which presents another common issue. For example, the debugs can show this SIP packet from the Cisco CallManager Express module:

```
Mar 11 10:09:13.767 EST: //-1/xxxxxxxxxxx/SIP/Msg/ccsipDisplayMsg:
Sent:
SIP/2.0 488 Not Acceptable Media
Via: SIP/2.0/UDP 172.18.106.88:5060
From: "Cisco SIP Channel1" <sip:outbound-0@172.18.106.66>;tag=75b5194d-133
```

```
To: <sip:1109811043@172.18.106.66;user=phone>;tag=23F1578C-252
Date: Fri, 11 Mar 2005 15:09:13 GMT
Call-ID: e34bafcc-131@172.18.106.88:5060
Server: Cisco-SIPGateway/IOS-12.x
CSeq: 51 INVITE
Allow-Events: telephone-event
Content-Length: 0
```

This output indicates that Cisco CallManager Express rejected the call because the SIP `INVITE` message from Cisco Unity Express did not match a dial peer that G.711 configured. You can add a dial peer specifically for the MWI traffic in order to correct this call rejection. The example in this section has `11099.....` for `MWI on` and `11098.....` for `MWI off`. You can add:

```
dial-peer voice 123 voip
incoming called-number 1109[8,9].....
codec g711ulaw
no vad
!
```

The last common issue is that the MWI traffic matches a translation pattern that is applied on a dial peer, VoIP incoming rule, or elsewhere. Or, Class of Restriction (COR) rules can block the call. Keep in mind that, even though you dial the MWI on/off number and the extension to light the MWI, the call does not necessarily behave the same when a call arrives via SIP. Refer to the document Configuring Class of Restrictions (COR) for more information about COR.

In summary, always verify these items:

- A Cisco CallManager Express license is present. Issue the **show software licenses** command. With a Cisco CallManager license, everything works *except* MWI.
- MWI on and off numbers are configured in Cisco CallManager Express. The number of dots indicate the length of the extensions. Issue the **show telephony–service ephone–dn** command.
- In Cisco Unity Express, the MWI on and off numbers are configured in order to match the on and off numbers in Cisco CallManager Express without the dots. The **show ccn application** command shows this.
- Cisco Unity Express points to the correct Cisco CallManager Express server IP address. The **show ccn subsystem sip** command shows this.
- Make sure that **mwi sip outcall** is configured under the **ccnsubsystem sip** command.

Then, if all else fails, begin to troubleshoot with the issue of the **trace ccn stacksip dbug** command.

Message Waiting Indicators (MWIs) (Cisco Unified CallManager Express Only)

Symptom: After you upgrade to a new version of Cisco Unity Express, the MWIs do not light up even when messages are left in the mailboxes.

- **Explanation** The upgrade procedure removed the IP address of the Session Initiation Protocol (SIP) subsystem.
- **Recommended Action** Reconfigure the SIP IP address to point to the Cisco Unified CME router.

Error: Searching, there was an error displaying your message

When you try to retrieve messages, the `Searching, there was an error displaying your message` error message appears.

Complete the steps described in To Enable Phone View for a Phone System in order to resolve the issue.

How to Troubleshoot a Cisco CallManager Express System

Perform these steps in order to troubleshoot the Cisco CallManager Express System:

1. Enter the **show ephone** command in order to display all registered phones. If no phones are registered, then perform these tasks:

    a. Check DHCP configuration, which includes the default router and the TFTP server address (option 150).
    b. Use the **dir** command in order to check that the required files are in the Flash memory of the router.
    c. Check that the **tftp−server** command is set for the required files.
    d. Use the **debug ephone register** mac−address command in order to display Cisco IP phone registration activity.
    e. Use the **debug ip dhcp** command in order to confirm DHCP operation.

2. Enter the **show ephone** command in order to display all registered phones. If phones are registered and are displayed, then perform these steps:

    a. Check that the phone button binding to the directory number is correct.
    b. Check that the Cisco IP phones show as registered.
    c. Use the Settings display on the phone in order to verify the IP parameter settings on the Cisco IP phone.
    d. Check that the keepalive count is updated when you enter the **show phone** command.
    e. Enter the **debug ephone register** mac−address command in order to reset the phone and observe the re−registration, in order to display the Cisco IP phones.
    f. Enter the **show ephone−dn summary** command in order to check the state of the Cisco IP phone lines.
    g. Check the IP address of the phone and attempt in order to ping the address.

3. Use the **debug ephone keepalive** command in order to set keepalive debugging for the Cisco IP phones.
4. Use the **debug ephone state** command in order to set state debugging for the Cisco IP phones.

MWI with Cisco CallManager

For Cisco Unity Express integrations with Cisco CallManager, it is most important to be sure that Unity Express is registered and has all the correct logon information.

The first step is to determine if a phone is in SRST mode, if available, in order to troubleshoot. Log in to the router in which the Cisco Unity Express module is installed. Then, issue the **show ephone registered** command. Any phones that are registered do not receive any MWI, even if Cisco Unity Express is correctly registered to Cisco CallManager.

```
vnt-2651-44a#show ephone registered

ephone-3 Mac:0008.E31B.7AFC TCP socket:[2] activeLine:0 REGISTERED
mediaActive:0 offhook:0 ringing:0 reset:0 reset_sent:0 paging 0 debug:0
IP:14.80.119.206 51984 Telecaster 7960  keepalive 2697 max_line 6
button 1: dn 1  number 2103  CM Fallback CH1    IDLE
button 2: dn 2  number 2199  CM Fallback CH1    IDLE


ephone-4 Mac:0008.E37F.A119 TCP socket:[4] activeLine:0 REGISTERED
mediaActive:0 offhook:0 ringing:0 reset:0 reset_sent:0 paging 0 debug:0
IP:14.80.119.207 50963 Telecaster 7960  keepalive 2696 max_line 6
button 1: dn 3  number 2104  CM Fallback CH1    IDLE
```

If no phones are in the Cisco CallManager fallback state, indicated by the `REGISTERED` status, as previously shown, SRST is not active for those devices. The next step, then, is to verify the Cisco Unity Express and Cisco CallManager configurations in order to be sure that Unity Express is registered to CallManager.

```
VNT-AIM-CUE1>show ccn subsystem jtapi
Cisco Call Manager:                 14.80.227.127
CCM JTAPI Username:                 site1cue
CCM JTAPI Password:                 *****
Call Control Group 1 CTI ports:     28001,28002,28003,28004
```

This output lists all the computer telephony integration (CTI) route point directory numbers and the JTAPI account Cisco Unity Express uses to log in to the Cisco CallManager.

You need to be sure that the Cisco Unity Express properly registers to the Cisco CallManager. First, confirm that the CTI ports are actually registered. The easiest way to do this is to go to the Cisco CallManager administration web page. Then, choose **Device > Phone** and search for the CTI ports listed in the output above. The Status and IP Address fields should be filled out completely.

If you find that the ports are unregistered, the Cisco Unity Express is unable to communicate with the Cisco CallManager. Another possibility is that the login is incorrect. Issue simple pings from the Cisco Unity Express module to the Cisco CallManager in order to troubleshoot this. If this works, verify that the Cisco CTIManager and the directory services, which is DC Directory Server in this case, have started. From the Cisco CallManager server, choose **Start > Programs > Administrative Tools > Services** in order to verify:
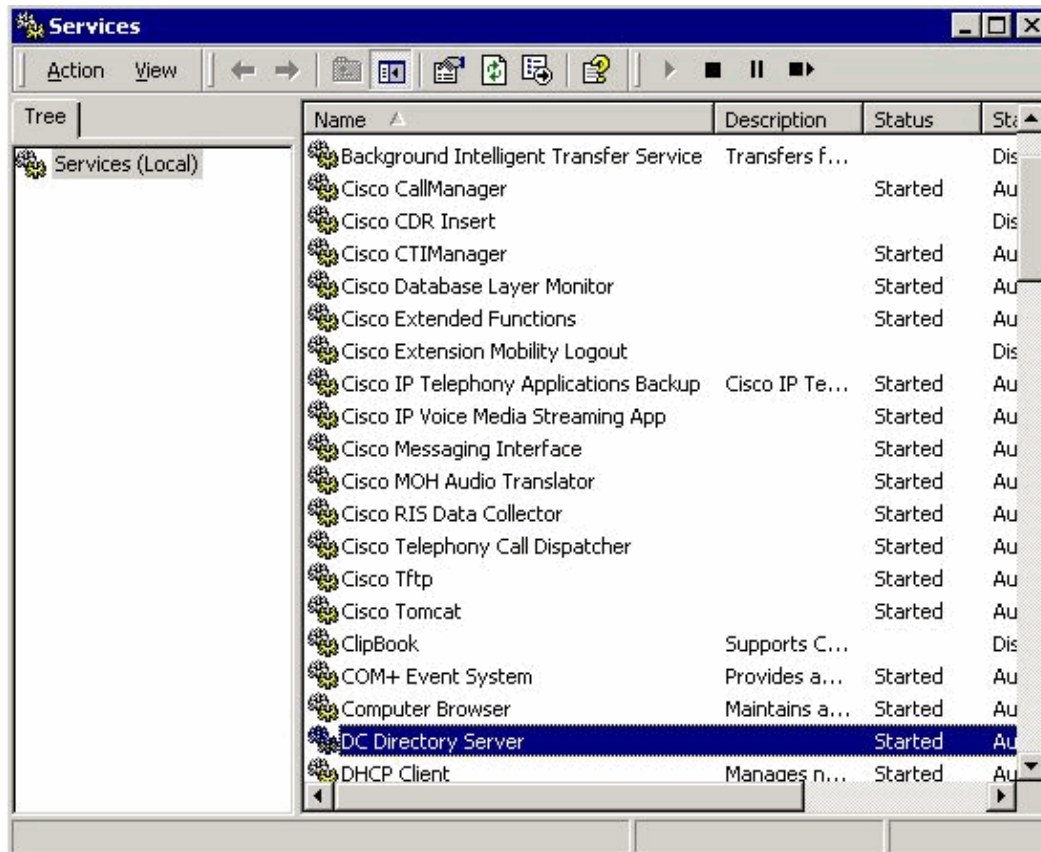
You should also verify that the JTAPI user account, which is site1cue in this example, exists. You should find the CTI ports, route points, and the Enable CTI Application Use checked. Also, verify the password.

Another common problem is the Calling Search Space of the CTI ports. This Calling Search Space must contain the Partitions of the directory numbers for which you' try to light the MWI light. For example, the Calling Search Space for the CTI ports, not the Route Points, must contain the Line1 partition in order to set an MWI for extension 1234 in Partition Line1. If the Calling Search Space for the CTI ports is None, then only extensions in the None partition work for MWI.

If the configuration appears to be correct, JTAPI diagnostics can be enabled on the Cisco Unity Express module. But, the enable and disable require a reboot. This level of diagnostics is beyond the regular **trace** debug settings. Do not leave this enabled, especially for the advanced integration module (AIM), because excessive writes to the internal flash card can reduce the life of the flash.

Issue a **show ccn trace jtapi** command in order to view the current, enabled JTAPI traces:

Note: By default, all JTAPI traces disabled.

```
VNT-AIM-CUE1>show ccn trace jtapi
Warning:                        0
Informational:                  0
Jtapi Debugging:                0
Jtapi Implementation:           0
CTI Debugging:                  0
CTI Implementation:             0
Protocol Debugging:             0
Misc Debugging:                 0
```

Issue these commands in order to enable all traces:

```
VNT-AIM-CUE1>ccn trace jtapi debug all
```

```
You will have to reload the system for your changes to take effect
VNT-AIM-CUE1>ccn trace jtapi informational all
You will have to reload the system for your changes to take effect
VNT-AIM-CUE1>ccn trace jtapi warning all
You will have to reload the system for your changes to take effect
VNT-AIM-CUE1>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
```

Now, you need to reload the system. Issue the same **ccn trace** commands shown above, but precede each command with the **no** keyword in order to disable this later on. For example, issue **no ccn trace jtapi debug all**. This is an important step to remember, especially on the AIM. Failure to perform this step affects potential performance, and it reduces the life of the compact flash card on the AIM.

After the reload, the system begins to write the files CiscoJtapi1.log and CiscoJtapi2.log, when the first one is full.

You can view these logs if you issue the **show log name CiscoJtapi1.log** command. You can also copy the log file to an FTP server, and then view the information offline. The command is **copy log CiscoJtapi1.log url ftp://user:passwd@ftpservipaddr/**.

With either method, all JTAPI information appears. In this example, the Cisco Unity Express module attempts to register, but is unsuccessful because of a WAN failure:

```
15252: Jul 14 03:58:24.412 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Trying
 connection to server: 14.80.227.127
15253: Jul 14 03:58:24.416 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Provider.tryOpen
 () Failure java.net.NoRouteToHostException: No route to host
15254: Jul 14 03:58:24.417 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) ProviderRetryThread
 waiting for 30000 msecsCCNException = com.cisco.cti.client.CCNException: No route to host
15255: Jul 14 03:58:54.803 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Trying connection
 to server: 14.80.227.127
15256: Jul 14 03:58:54.808 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Provider.tryOpen
 () Failure java.net.NoRouteToHostException: No route to host
15257: Jul 14 03:58:54.809 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) ProviderRetryThread
 waiting for 30000 msecsCCNException = com.cisco.cti.client.CCNException: No route to host
15258: Jul 14 03:59:24.817 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Trying connection
 to server: 14.80.227.127
15259: Jul 14 03:59:24.820 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Provider.tryOpen
 () Failure java.net.NoRouteToHostException: No route to host
15260: Jul 14 03:59:24.821 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) ProviderRetryThread
 waiting for 30000 msecsCCNException = com.cisco.cti.client.CCNException: No route to host
15261: Jul 14 03:59:55.210 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Trying connection
 to server: 14.80.227.127
```

The next trace shows a full registration of Cisco Unity Express to a Cisco CallManager. In this example, you see that there are eight CTI ports associated with the JTAPI user. But, because Cisco Unity Express is only licensed for four ports, only four ports are used. Also, notice that the system automatically does a full MWI resynchronization after reregistering to the Cisco CallManager:

```
17937: Jul 14 11:28:56.037 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Trying
 connection to server: 14.80.227.127
17938: Jul 14 11:28:56.042 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) connected
17939: Jul 14 11:28:56.043 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread: created
17940: Jul 14 11:28:56.045 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread
```

```
 starting up...
17941: Jul 14 11:28:56.056 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.ProviderOpenRequest {
sequenceNumber = 238
provider = 14.80.227.127
qbeClientVersion = Cisco JTAPI 1.4(3.12) Release
login = site1cue
password = 0c0a000a2c
filter = com.cisco.cti.protocol.ProviderEventFilter {
deviceRegistered = true
deviceUnregistered = true
directoryChangeNotify = true
}
applicationID = Cisco IP IVR
desiredServerHeartbeatTime = 30
cmAssignedApplicationID = 0
}
17942: Jul 14 11:28:56.072 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) ReceiveThread
 starting up...
17943: Jul 14 11:28:56.114 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.ProviderOpenResponse {
sequenceNumber = 238
providerInfoString = 3.3(3)
clientHeartbeat = 30
serverHeartbeat = 30
}
17944: Jul 14 11:28:56.131 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Server response:
 will send server heartbeat every 30 seconds
17945: Jul 14 11:28:56.131 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) Server response:
 expecting client heartbeat every 30 seconds
17946: Jul 14 11:28:56.133 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) HeartbeatSendThread
 starting up
17947: Jul 14 11:28:56.135 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 DeviceLineUpdateThread: created
17948: Jul 14 11:28:56.136 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 DeviceLineUpdateThread starting up...
17949: Jul 14 11:28:56.671 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.ProviderOpenCompletedEvent {
eventSequence = 279
reason = 0
sequenceNumber = 238
providerInfoString = 3.3(3)
clientHeartbeat = 30
serverHeartbeat = 30
failureDescription = null
bMonitorCallParkDNs = false
}
1ISC-7-UNK:(P1-14.80.227.127) EventThread: queuing
 com.cisco.cti.protocol.ProviderOpenCompletedEvent
17951: Jul 14 11:28:56.674 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.ProviderOpenCompletedEvent[279]
17952: Jul 14 11:28:56.674 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) connected to
 CTIManager version 3.3(3)
17953: Jul 14 11:28:56.676 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.ProviderGetCapabilitiesRequest {
sequenceNumber = 239
}
17954: Jul 14 11:28:56.679 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.ProviderGetCapabilitiesResponse {
sequenceNumber = 239
providerCapabilitiesInfo = com.cisco.cti.protocol.ProviderCapabilitiesInfo {
controlAnyDevice = false
maxNumberOfDevicesOpen = 0
}
}
17955: Jul 14 11:28:56.680 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) can control any
```

```
 device = false
17956: Jul 14 11:28:56.681 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.ProviderGetDeviceInfoRequest {
sequenceNumber = 240
deviceGroup = 1
enumerateRegisterableDevices = true
}
17957: Jul 14 11:28:56.685 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.ProviderGetDeviceInfoResponse {
sequenceNumber = 240
enumerationHandle = 3
}
17958: Jul 14 11:28:56.686 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.80.227.127) received Response:
 com.cisco.cti.protocol.GetDeviceInfoFetchResponse {
sequenceNumber = 241
info = 11@[
com.cisco.cti.protocol.DeviceInfo {
name = CUE_SIte1_GMS
type = 73
allowsRegistration = true
},
com.cisco.cti.protocol.DeviceInfo {
name = CUE_Site1_AA
type = 73
allowsRegistration = true
},
com.cisco.cti.protocol.DeviceInfo {
name = CUE_Site1_VM
type = 73
allowsRegistration = true
},
com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p01
type = 72
allowsRegistration = true
},
com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p03
type = 72
allowsRegistration = true
},
com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p02
type = 72
allowsRegistration = true
},
com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p05
type = 72
allowsRegistration = true
},
com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p04
type = 72
allowsRegistration = true
},
com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p07
type = 72
allowsRegistration = true
},
com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p06
type = 72
allowsRegistration = true
```

```
},
com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p08
type = 72
allowsRegistration = true
}]
more = false
}
17960: Jul 14 11:28:56.706 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetDeviceInfoCloseRequest {
sequenceNumber = 242
enumerationHandle = 3
}
17961: Jul 14 11:28:56.709 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetDeviceInfoCloseResponse {
sequenceNumber = 242
}
17962: Jul 14 11:28:56.710 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) creating controlled
 devices
17963: Jul 14 11:28:56.712 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p08(0,0)
 updating lines
17964: Jul 14 11:28:56.713 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 243
deviceName = cue_site1_p08
}
17965: Jul 14 11:28:56.716 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 243
enumerationHandle = 1
}
17966: Jul 14 11:28:56.718 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 244
enumerationHandle = 1
count = 10
}
17967: Jul 14 11:28:56.754 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = ol.LineInfo {
name = 28008
permanentLineID = 1936802189
}]
more = false
}
17968: Jul 14 11:28:56.761 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 245
enumerationHandle = 1
}
17969: Jul 14 11:28:56.967 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 245
}
17970: Jul 14 11:28:56.968 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p08(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
17971: Jul 14 11:28:56.969 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p07(0,0)
 updating lines
17972: Jul 14 11:28:56.970 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 246
deviceName = cue_site1_p07
}
17973: Jul 14 11:28:56.973 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 246
```

```
enumerationHandle = 2
}
17974: Jul 14 11:28:56.975 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 247
enumerationHandle = 2
count = 10
}
17975: Jul 14 11:28:57.007 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 247
info = 1@[
com.cisconeID = 829100962
}]
more = false
}
17976: Jul 14 11:28:57.009 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 248
enumerationHandle = 2
}
17977: Jul 14 11:28:57.227 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 248
}
17978: Jul 14 11:28:57.229 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p07(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
17979: Jul 14 11:28:57.229 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p06(0,0)
 updating lines
17980: Jul 14 11:28:57.230 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 249
deviceName = cue_site1_p06
}
17981: Jul 14 11:28:57.233 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 249
enumerationHandle = 3
}
17982: Jul 14 11:28:57.235 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 250
enumerationHandle = 3
count = 10
}
17983: Jul 14 11:28:57.260 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 250
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28006
permanentLineID = 294850253
}]
more = false
}
17984: Jul 14 11:28:57.262 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 251
enumerationHandle = 3
}
17985: Jul 14 11:28:57.265 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 251
}
17986: Jul 14 11:28:57.267 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p06(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
```

```
17987: Jul 14 11:28:57.268 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p05(0,0)
 updating lines
17988: Jul 14 11:28:57.268 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 252
deviceName = cue_site1_p05
}
17989: Jul 14 11:28:57.271 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 252
enumerationHandle = 4
}
17990: Jul 14 11:28:57.273 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 253
enumerationHandle = 4
count = 10
}
17991: Jul 14 11:28:57.309 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 253
info = 1@[
com.cisco.cti.protocol.LineInfo {7.311 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 254
enumerationHandle = 4
}
17993: Jul 14 11:28:57.314 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 254
}
17994: Jul 14 11:28:57.316 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p05(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
17995: Jul 14 11:28:57.317 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p04(0,0)
 updating lines
17996: Jul 14 11:28:57.318 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 255
deviceName = cue_site1_p04
}
17997: Jul 14 11:28:57.322 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 255
enumerationHandle = 5
}
17998: Jul 14 11:28:57.324 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 256
enumerationHandle = 5
count = 10
}
17999: Jul 14 11:28:57.358 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 256
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28004
permanentLineID = 1897211172
}]
more = false
}
18000: Jul
enumerationHandle = 5
}
18001: Jul 14 11:28:57.363 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
```

```
sequenceNumber = 257
}
18002: Jul 14 11:28:57.364 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p04(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
18003: Jul 14 11:28:57.365 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p03(0,0)
 updating lines
18004: Jul 14 11:28:57.366 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 258
deviceName = cue_site1_p03
}
18005: Jul 14 11:28:57.587 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 258
enumerationHandle = 6
}
18006: Jul 14 11:28:57.589 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 259
enumerationHandle = 6
count = 10
}
18007: Jul 14 11:28:57.632 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 259
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28003
permanentLineID = 2109152574
}]
more = false
}
18008: Jul 14 11:28:57.634 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 260
enumerationHandle = 6
}
18009: Jul 14 11:28:57.637 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 260
}
18010: Jul 14 11:28:57.638 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p03(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
18011: Jul 14 11:28:57.639 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p02(0,0)
 updating lines
18012: Jul 14 11:28:57.640 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 261
deviceName = cue_site1_p02
}
18013: Jul 14 11:28:57.645 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 261
enumerationHandle = 7
}
18014: Jul 14 11:28:57.646 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 262
enumerationHandle = 7
count = 10
}
18015: Jul 14 11:28:57.681 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 262
info = 1@[
com.cisco.cti.protocol.LineInfo {
```

```
name = 28002
permanentLineID = 1035863534
}]
more = false
}
18016: Jul 14 11:28:57.683 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLUNK:(P1-14.80.227.127)
 received Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 263
}
18018: Jul 14 11:28:57.687 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p02(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
18019: Jul 14 11:28:57.688 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p01(0,0)
 updating lines
18020: Jul 14 11:28:57.689 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 264
deviceName = cue_site1_p01
}
18021: Jul 14 11:28:57.692 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 264
enumerationHandle = 8
}
18022: Jul 14 11:28:57.694 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 265
enumerationHandle = 8
count = 10
}
18023: Jul 14 11:28:57.708 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 265
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28001
permanentLineID = 1084634008
}]
more = false
}
18024: Jul 14 11:28:57.710 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 266
enumerationHandle = 8
}
18025: Jul 14 11:28:57.713 EDT %JTAPI-espose:
 com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 266
}
18026: Jul 14 11:28:57.716 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p01(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
18027: Jul 14 11:28:57.717 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_SIte1_GMS(0,0)
 updating lines
18028: Jul 14 11:28:57.718 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 267
deviceName = CUE_SIte1_GMS
}
18029: Jul 14 11:28:57.725 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 267
enumerationHandle = 9
}
18030: Jul 14 11:28:57.727 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 268
```

```
enumerationHandle = 9
count = 10
}
18031: Jul 14 11:28:57.961 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 268
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28111
permanentLineID = 632514620
}]
more = false
}
18032: Jul 14 11:28:57.963 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 269
enumerationHandle = 9
}
18033: Jul 14 11:28:57.966 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 269
}
18034: Jul 14 11:28:57.967 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_SIte1_GMS(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
18035: Jul 14 11:28:57.968 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_Site1_AA(0,0)
 updating lines
18036: Jul 14 11:28:57.969 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 270
deviceName = CUE_Site1_AA
}
18037: Jul 14 11:28:57.972 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 270
enumerationHandle = 10
}
18038: Jul 14 11:28:57.974 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 271
enumerationHandle = 10
count = 10
}
18039: Jul 14 11:28:58.011 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 271
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28100
permanentLineID = 117519949
}]
more = false
}
18040: Jul 14 11:28:58.013 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 272
enumerationHandle = 10
}
18041: Jul 14 11:28:58.018 EDT %JTAved Response:
 com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 272
}
18042: Jul 14 11:28:58.019 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_Site1_AA(0,0)
 refreshing lines: previous=1 current=1 created=0 removed=0
18043: Jul 14 11:28:58.020 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_Site1_VM(0,0)
 updating lines
18044: Jul 14 11:28:58.021 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
```

[ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 273
deviceName = CUE_Site1_VM
}
18045: Jul 14 11:28:58.025 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 273
enumerationHandle = 11
}
18046: Jul 14 11:28:58.035 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 274
enumerationHandle = 11
count = 10
}
18047: Jul 14 11:28:58.060 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 274
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28000
permanentLineID = 1978608865
}]
more = false
}
18048: Jul 14 11:28:58.061 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 275
enumerationHandle = 11
}
18049: Jul 14 11:28:58.277 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227s=1 current=1
 created=0 removed=0
18051: Jul 14 11:28:58.279 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) refreshing device
 map: previous=11 current=11 created=0 removed=0
18052: Jul 14 11:28:58.280 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.ProviderGetDeviceInfoRequest {
sequenceNumber = 276
deviceGroup = 3
enumerateRegisterableDevices = true
}
18053: Jul 14 11:28:58.283 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.ProviderGetDeviceInfoResponse {
sequenceNumber = 276
enumerationHandle = 4
}
18054: Jul 14 11:28:58.285 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetDeviceInfoFetchRequest {
sequenceNumber = 277
enumerationHandle = 4
count = 100
type = 2
}
18055: Jul 14 11:28:58.296 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetDeviceInfoFetchResponse {
sequenceNumber = 277
info = null
more = false
}
18056: Jul 14 11:28:58.298 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.GetDeviceInfoCloseRequest {
sequenceNumber = 278
enumerationHandle = 4
}
18057: Jul 14 11:28:58.507 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetDeviceInfoCloseResponse {
sequenceNumber = 278

```
}
18058: Jul 14 11:28:58.508 EDT %JTAPI-MISC-7-UNK:Provider "(P1-site1cue)" changing
 state to IN_SERVICE
18059: Jul 14 11:28:58.509 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue)[ProviderRetryThread]
 (P1-site1cue) Request: getObservers
18060: Jul 14 11:28:58.510 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) ProvInServiceEv [#684]
18061: Jul 14 11:28:58.511 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@107836e4]
 ObserverProxy.queueEvents: queuing asynchronously
18062: Jul 14 11:28:58.511 EDT %JTAPI-MISC-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@107836e4):
 queuing com.cisco.jtapi.JtapiProviderEventSet
18063: Jul 14 11:28:58.512 EDT %JTAPI-JTAPIIMPL-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@107836e4):
 delivering JPES[1]
18064: Jul 14 11:28:58.513 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@107836e4]
 ObserverProxy.deliverEvents()
18065: Jul 14 11:28:58.517 EDT %JTAPI-JTAPIIMPL-7-UNK:
  [com.cisco.wf.subsystems.jtapi.SubsystemJTAPI$ProviderObserver@107836e4]
 ObserverProxy.deliverEvents() completed
18066: Jul 14 11:28:58.522 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening device
 (P1-site1cue) CUE_SIte1_GMS(0,0)
18067: Jul 14 11:28:58.525 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceOpenRequest {
sequenceNumber = 279
deviceName = CUE_SIte1_GMS
filter = com.cisco.cti.protocol.DeviceEventFilter {
deviceModeChanged = false
keyPressed = false
displayChanged = false
startTransmission = true
stopTransmission = true
startReception = true
stopReception = true
softKeyPressed = false
deviceData = true
}
disableAutoRecovery = false
}
18068: Jul 14 11:28:58.544 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 received Event: com.cisco.cti.protocol.DeviceRegisteredEvent {
eventSequence = 280
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
name = CUE_SIte1_GMS
type = 73
allowsRegistration = true
}
loginAllowed = false
loginUserID =
controllable = true
reason = 0
}
18069: Jul 14 11:28:58.545 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceRegisteredEvent
18070: Jul 14 11:28:58.546 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceRegisteredEvent[280]
18071: Jul 14 11:28:58.546 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) Received
 DeviceRegisteredEvent
18072: Jul 14 11:28:59.303 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceOpenResponse {
sequenceNumber = 279
callManagerID = 16777227
deviceID = 33
}
18073: Jul 14 11:28:59.306 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) DeviceMap:
```

```
 opening device "CUE_SIte1_GMS"
18074: Jul 14 11:28:59.314 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 DeviceLineUpdateThread: queuing com.cisco.cti.client.implementation.Device
18075: Jul 14 11:28:59.315 EDT %JTAPI-CTi.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 280
deviceName = CUE_SIte1_GMS
}
18077: Jul 14 11:28:59.325 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_SIte1_GMS(16777227,33)
 reopening line 28111(0,0)
18078: Jul 14 11:28:59.328 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.LineOpenRequest {
sequenceNumber = 281
deviceName = CUE_SIte1_GMS
lineName = 28111
filter = com.cisco.cti.protocol.LineEventFilter {
callStateChanged = true
dtmf = true
ring = false
toneChanged = false
globalCallHandleChanged = true
openReceiveChannel = false
partyInfoChanged = true
bExistingCallEvent = true
bNewCallEvent = true
bLineCfwdAllStatus = true
}
disableAutoRecovery = false
}
18079: Jul 14 11:28:59.305 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.DeviceInServiceEvent {
eventSequence = 281
deviceCallManagerID = 16777227
deviceID = 33
}
18080: Jul 14 11:28:59.330 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceInServiceEvent
18081: Jul 14 11:28:59.331 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceInServiceEvent[281]
18082: Jul 14 11:28:59.332 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Terminal
 "CUE_SIte1_GMS" in service
18083: Jul 14 11:28:59.333 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [CUE_SIte1_GMS]
 CiscoTermInServiceEv [#685]
18084: Jul 14 11:28:59.334 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 280
enumerationHandle = 12
}
18085: Jul 14 11:28:59.336 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 282
enumerationHandle = 12
count = 10
}
18086: Jul 14 11:28:59.362 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineOpenResponse {
sequenceNumber = 281
callManagerID = 16777227
lineID = 33
}
18087: Jul 14 11:28:59.364 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening device
 (P1-site1cue) CUE_Site1_AA(0,0)
18088: Jul 14 11:28:59.367 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceOpenRequest {
sequenceNumber = 283
deviceName = CUE_Site1_AA
```

```
filter = com.cisco.cti.protocol.DeviceEventFilter {
deviceModeChanged = false
keyPressed = false
featureButtonPressed = false
lampModeChanged = false
ringModeChanged = false
displayChanged = false
startTransmission = true
stopTransmission = true
startReception = true
stopReception = true
softKeyPressed = false
deviceData = true
}
dilse
}
18089: Jul 14 11:28:59.371 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.LineInServiceEvent {
eventSequence = 282
lineCallManagerID = 16777227
lineID = 33
}
18090: Jul 14 11:28:59.371 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.LineInServiceEvent
18091: Jul 14 11:28:59.372 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.LineInServiceEvent[282]
18092: Jul 14 11:28:59.373 EDT %JTAPI-CTI-7-UNK:(P1-site1cue){Line:28111(16777227,33)}
 LineInServiceEvent
18093: Jul 14 11:28:59.374 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Address "28111"
 in service
18094: Jul 14 11:28:59.374 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [28111]
 CiscoAddrInServiceEv [#686]
18095: Jul 14 11:28:59.375 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@6d8576e6]
 ObserverProxy.queueEvents: queuing asynchronously
18096: Jul 14 11:28:59.376 EDT %JTAPI-MISC-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@6d8576e6):
 queuing com.cisco.jtapi.JtapiAddressEventSet
18097: Jul 14 11:28:59.377 EDT %JTAPI-JTAPIIMPL-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@6d8576e6):
 delivering JAES[1]
18098: Jul 14 11:28:59.378 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@6d8576e6]
 ObserverProxy.deliverEvents()
18099: Jul 14 11:28:59.391 EDT %JTAPI-JTAPIIMPL-7-UNK:[com.cisco.wf.subsyscompleted
18100: Jul 14 11:28:59.403 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 282
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28111
permanentLineID = 632514620
}]
more = false
}
18101: Jul 14 11:28:59.405 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 284
enumerationHandle = 12
}
18102: Jul 14 11:28:59.408 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.DeviceRegisteredEvent {
eventSequence = 283
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
name = CUE_Site1_AA
```

```
type = 73
allowsRegistration = true
}
loginAllowed = false
loginUserID =
controllable = true
reason = 0
}
}
18103: Jul 14 11:28:59.409 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceRegisteredEvent
18104: Jul 14 11:28:59.410 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceRegisteredEvent[283]
18105: Jul 14 11:28:59.411 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) Received
 DeviceRegisteredEvent
18106: Jul 14 11:28:59.412 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceOpenResponse {
sequenceNumber = 283
callManagerID = 16777227
deviceID = 34
}
18107: Jul 14 11:28:59.414 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.DeviceInServiceEvent {
eventSequence = 284
deviceCallManagerID = 16777227
deviceID = 34
}
18108: Jul 14 11:28:59.416 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) DeviceMap: opening
 device "CUE_Site1_AA"
18109: Jul 14 11:28:59.417 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 DeviceLineUpdateThread: queuing com.cisco.cti.client.implementation.Device
18110: Jul 14 11:28:59.418 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_Site1_AA(16777227,34)
 reopening line 28100(0,0)
18111: Jul 14 11:28:59.420 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.LineOpenRequest {
sequenceNumber = 285
deviceName = CUE_Site1_AA
lineName = 28100
filter = com.cisco.cti.protocol.LineEventFilter {
callStateChanged = true
dtmf = true
ring = false
toneChanged = false
globalCallHandleChanged = true
openReceiveChannel = false
partyInfoChanged = true
bExistingCallEvent = true
bNewCallEvent = true
bLineCfwdAllStatus = true
}
disableAutoRecovery = false
}
18112: Jul 14 11:28:59.422 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceInServiceEvent
18113: Jul 14 11:28:59.423 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.proto
18115: Jul 14 11:28:59.425 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [CUE_Site1_AA]
 CiscoTermInServiceEv [#687]
18116: Jul 14 11:28:59.428 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 284
}
18117: Jul 14 11:28:59.429 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_SIte1_GMS(16777227,33)
 refreshing lines: previous=1 current=1 created=0 removed=0
18118: Jul 14 11:28:59.430 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_Site1_AA(16777227,34)
 updating lines
18119: Jul 14 11:28:59.431 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
```

```
[(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 286
deviceName = CUE_Site1_AA
}
18120: Jul 14 11:28:59.434 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineOpenResponse {
sequenceNumber = 285
callManagerID = 16777227
lineID = 34
}
18121: Jul 14 11:28:59.436 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening device
 (P1-site1cue) cue_site1_p08(0,0)
18122: Jul 14 11:28:59.436 EDT %JTAPI-CTIIMPL-7-UNK:(P1-site1cue) cue_site1_p08(0,0)
 Device is not Opened previously, not attempting to open
18123: Jul 14 11:28:59.437 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening device
 (P1-site1cue) CUE_Site1_VM(0,0)
18124: Jul 14 11:28:59.439 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceOpenRequest {
sequenceNumber = 287
deviceName = CUE_Site1_VM
filter ssed = false
lampModeChanged = false
ringModeChanged = false
displayChanged = false
startTransmission = true
stopTransmission = true
startReception = true
stopReception = true
softKeyPressed = false
deviceData = true
}
disableAutoRecovery = false
}
18125: Jul 14 11:28:59.442 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.LineInServiceEvent {
eventSequence = 285
lineCallManagerID = 16777227
lineID = 34
}
18126: Jul 14 11:28:59.443 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.LineInServiceEvent
18127: Jul 14 11:28:59.444 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.LineInServiceEvent[285]
18128: Jul 14 11:28:59.445 EDT %JTAPI-CTI-7-UNK:(P1-site1cue){Line:28100(16777227,34)}
 LineInServiceEvent
18129: Jul 14 11:28:59.446 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Address "28100"
 in service
18130: Jul 14 11:28:59.447 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [28100]
 CiscoAddrInServiceEv [#688]
18131: Jul 14 11:28:59.448 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3f0ab6e7]
 ObserverProxy.queueEvents: queuing asynchronously
18132: Jul 14 11:28:59.448 EDT %JTAPI-MISC-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3f0ab6e7):
 queuing com.cisco.jtapi.JtapiAddressEventSet
18133: Jul 14 11:28:59.449 EDT %JTAPI-JTAPIIMPL-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3f0ab6e7):
 delivering JAES[1]
18134: Jul 14 11:28:59.450 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3f0ab6e7]
 ObserverProxy.deliverEvents()
18135: Jul 14 11:28:59.468 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3f0ab6e7]
 ObserverProxy.deliverEvents() completed
18136: Jul 14 11:28:59.475 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
```

```
  Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 286
enumerationHandle = 13
}
18137: Jul 14 11:28:59.476 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 288
enumerationHandle = 13
count = 10
}
18138: Jul 14 11:28:59.481 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.DeviceRegisteredEvent {
eventSequence = 286
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
name = CUE_Site1_VM
type = 73
allowsRegistration = true
}
loginAllowed = false
loginUserID =
controllable = true
reason = 0
}
18139: Jul 14 11:28:59.482 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceRegisteredEvent
18140: Jul 14 11:28:59.483 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceRegisteredEvent[286]
18141: Jul 14 11:28:59.484 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) Received
 DeviceRegisteredEvent
18142: Jul 14 11:28:59.705 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceOpenResponse {
sequenceNumber = 287
callManagerID = 16777227
deviceID = 35
}
18143: Jul 14 11:28:59.707 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) DeviceMap: opening
 device "CUE_Site1_VM"
18144: Jul 14 11:28:59.708 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 DeviceLineUpdateThread: queuing com.cisco.cti.client.implementation.Device
18145: Jul 14 11:28:59.709 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_Site1_VM(16777227,35)
 reopening line 28000(0,0)
18146: Jul 14 11:28:59.711 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.LineOpenRequest {
sequenceNumber = 289
deviceName = CUE_Site1_VM
lineName = 28000
filter = com.cisco.cti.protocol.LineEventFilter {
callStateChanged = true
dtmf = true
ring = false
toneChanged = false
globalCallHandleChanged = true
openReceiveChannel = false
partyInfoChanged = true
bExistingCallEvent = true
bNewCallEvent = true
bLineCfwdAllStatus = true
}
disableAutoRecovery = false
}
18147: Jul 14 11:28:59.714 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.DeviceInServiceEvent {
eventSequ
18149: Jul 14 11:28:59.716 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceInServiceEvent[287]
```

```
18150: Jul 14 11:28:59.718 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Terminal
 "CUE_Site1_VM" in service
18151: Jul 14 11:28:59.718 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [CUE_Site1_VM]
 CiscoTermInServiceEv [#689]
18152: Jul 14 11:28:59.720 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 288
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28100
permanentLineID = 117519949
}]
more = false
}
18153: Jul 14 11:28:59.722 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 290
enumerationHandle = 13
}
18154: Jul 14 11:28:59.724 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineOpenResponse {
sequenceNumber = 289
callManagerID = 16777227
lineID = 35
}
18155: Jul 14 11:28:59.726 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening device
 (P1-site1cue) cue_site1_p07(0,0)
18156: Jul 14 11:28:59.726 EDT %JTAPI-CTIIMPL-7-UNK:(P1-site1cue) cue_site1_p07(0,0)
 Device is not Opened previously, not attempting to open
18157: Jul 14 11:28:59.727 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening device
 (P1-site1cue) cue_site1_p06(0,0)
18158: Jul 14 11:28:59.728 EDT %JTAPI-CTIIMPL-7-UNK:(P1-site1cue) cue_site1_p06(0,0)
 Device is not Opened previously, not attempting to open
18159: Jul 14 11:28:59.728 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening device
 (P1-site1cue) cue_site1_p05(0,0)
18160: Jul 14 11:28:59.729 EDT %JTAPI-CTIIMPL-7-UNK:(P1-site1cue) cue_site1_p05(0,0)
 Device is not Opened previously, not attempting to open
18161: Jul 14 11:28:59.729 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening device
 (P1-site1cue) cue_site1_p04(0,0)
18162: Jul 14 11:28:59.733 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceRegisterDeviceRequest {
sequenceNumber = 291
deviceName = cue_site1_p04
ipAddr = 1802113708
rtpPortNumber = 16384
mediaSpecificationTimeout = 0
mediaCaps = 2@[
com.cisco.cti.protocol.MediaCapability {
payloadCapability = 4
maxFramesPerPacket = 30
bitRate = 1
},
com.cisco.cti.protocol.MediaCapability {
payloadCapability = 2
maxFramesPerPacket = 30
bitRate = 1
}]
filter = com.cisco.cti.protocol.DeviceEventFilter {
deviceModeChanged = false
keyPressed = false
featureButtonPressed = false
lampModeChanged = false
ringModeChanged = false
displayChanged = false
startTransmission = true
```

```
stopTransmission = true
startReception = true
stopReception = true
softKeyPressed = false
deviceData 163: Jul 14 11:28:59.737 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.LineInServiceEvent {
eventSequence = 288
lineCallManagerID = 16777227
lineID = 35
}
18164: Jul 14 11:28:59.737 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.LineInServiceEvent
18165: Jul 14 11:28:59.739 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.LineInServiceEvent[288]
18166: Jul 14 11:28:59.739 EDT %JTAPI-CTI-7-UNK:(P1-site1cue){Line:28000(16777227,35)}
 LineInServiceEvent
18167: Jul 14 11:28:59.740 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Address "28000" in
 service
18168: Jul 14 11:28:59.741 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [28000]
 CiscoAddrInServiceEv [#690]
18169: Jul 14 11:28:59.741 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@40b3b6e1]
 ObserverProxy.queueEvents: queuing asynchronously
18170: Jul 14 11:28:59.742 EDT %JTAPI-MISC-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@40b3b6e1):
 queuing com.cisco.jtapi.JtapiAddressEventSet
18171: Jul 14 11:28:59.744 EDT %JTAPI-JTAPIIMPL-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@40b3b6e1):
 delivering JAES[1]
18172: Jul 14 11:28:59.744 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@40b3b6e1]
 ObserverProxy.deliverEvents()
18173: Jul 14 11:28:59.760 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.T
18174: Jul 14 11:28:59.768 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 290
}
18175: Jul 14 11:28:59.769 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_Site1_AA(16777227,34)
 refreshing lines: previous=1 current=1 created=0 removed=0
18176: Jul 14 11:28:59.770 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_Site1_VM(16777227,35)
 updating lines
18177: Jul 14 11:28:59.771 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 292
deviceName = CUE_Site1_VM
}
18178: Jul 14 11:28:59.775 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received Event:
 com.cisco.cti.protocol.DeviceRegisteredEvent {
eventSequence = 289
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p04
type = 72
allowsRegistration = true
}
loginAllowed = false
loginUserID =
controllable = true
reason = 0
}
18179: Jul 14 11:28:59.776 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceRegisteredEvent
18180: Jul 14 11:28:59.777 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceRegisteredEvent[289]
18181: Jul 14 11:28:59.778 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) Received
```

```
 DeviceRegisteredEvent
18182: Jul 14 11:28:59.780 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceRegisterDeviceResponse {
sequenceNumber = 291
callManagerID = 16777227
deviceID = 36
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p04
type = 72
allowsRegistration = true
}
}
18183: Jul 14 11:28:59.781 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) DeviceMap: opening
 device "cue_site1_p04"
18184: Jul 14 11:28:59.782 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 DeviceLineUpdateThread: queuing com.cisco.cti.client.implementation.Device
18185: Jul 14 11:28:59.783 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p04(16777227,36)
 reopening line 28004(0,0)
18186: Jul 14 11:28:59.785 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.LineOpenRequest {
sequenceNumber = 293
deviceName = cue_site1_p04
lineName = 28004
filter = com.cisco.cti.protocol.LineEventFilter {
callStateChanged = true
dtmf = true
ring = false
toneChanged = false
globalCallHandleChanged = true
openReceiveChannel = false
partyInfoChanged = true
bExistingCallEvent = true
bNewCallEvent = true
bLineCfwdAllStatus = true
}
disableAutoRecovery = false
}
18187: Jul 14 11:28:59.789 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.DeviceInServiceEvent {
eventSequence = 290
deviceCallManagerID = 16777227
deviceID cti.protocol.DeviceInServiceEvent
18189: Jul 14 11:28:59.790 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceInServiceEvent[290]
18190: Jul 14 11:28:59.791 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Terminal
 "cue_site1_p04" in service
18191: Jul 14 11:28:59.792 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [cue_site1_p04]
 CiscoTermInServiceEv [#691]
18192: Jul 14 11:28:59.794 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 292
enumerationHandle = 14
}
18193: Jul 14 11:28:59.796 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 294
enumerationHandle = 14
count = 10
}
18194: Jul 14 11:28:59.799 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineOpenResponse {
sequenceNumber = 293
callManagerID = 16777227
lineID = 36
}
```

```
18195: Jul 14 11:28:59.800 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening
 device (P1-site1cue) cue_site1_p03(0,0)
18196: Jul 14 11:28:59.803 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceRegisterDeviceRequest {
sequenceNumber = 295
deviceName = cue_site1_p03
ipAddr = 1802113708
rtpPortNumber = 16386
mediaSpecificationTimeout = 0
mediaCaps = 2@[
com.cisco.cti.ability {
payloadCapability = 2
maxFramesPerPacket = 30
bitRate = 1
}]
filter = com.cisco.cti.protocol.DeviceEventFilter {
deviceModeChanged = false
keyPressed = false
featureButtonPressed = false
lampModeChanged = false
ringModeChanged = false
displayChanged = false
startTransmission = true
stopTransmission = true
startReception = true
stopReception = true
softKeyPressed = false
deviceData = true
}
disableAutoRecovery = false
}
18197: Jul 14 11:28:59.807 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.LineInServiceEvent {
eventSequence = 291
lineCallManagerID = 16777227
lineID = 36
}
18198: Jul 14 11:28:59.808 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.LineInServiceEvent
18199: Jul 14 11:28:59.809 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.LineInServiceEvent[291]
18200: Jul 14 11:28:59.810 EDT %JTAPI-CTI-7-UNK:(P1-site1cue){Line:28004(16777227,36)}
 LineInServiceEvent
18201: Jul 14 11:28:59.810 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Address "28004"
 in service
18202: Jul 14 11:28:59.811 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [28004]
 CiscoAddrInServiceEv [#692]
18203: Jul 14 11:28:59.812 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3928f6e1]
 ObserverProxy.queueEvents: queuing asynchronously
18204: Jul 14 11:28:59.812 EDT %JTAPI-MISC-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3928f6e1):
 queuing com.cisco.jtapi.JtapiAddressEventSet
18205: Jul 14 11:28:59.813 EDT %JTAPI-JTAPIIMPL-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3928f6e1):
 delivering JAES[1]
18206: Jul 14 11:28:59.814 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3928f6e1]
 ObserverProxy.deliverEvents()
18207: Jul 14 11:28:59.948 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@3928f6e1]
 ObserverProxy.deliverEvents() completed
18208: Jul 14 11:29:00.057 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 294
info = 1@[
```

```
             com.cisco.cti.protocol.LineInfo {
             name = 28000
             permanentLineID = 1978608865
             }]
             more = false
             }
             18209: Jul 14 11:29:00.059 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
              [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
              com.cisco.cti.protocol.GetLineInfoCloseRequest {
             sequenceNumber = 296
             enumerationHandle = 14
             }
             18210: Jul 14 11:29:00.062 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
              Event: com.cisco.cti.protocol.DeviceRegisteredEvent {
             eventSequence = 292
             deviceInfo = com.cisco.cti.protocol.DeviceInfo {
             name = cue_site1_p03
             type = 72
             owsRegistration = true
             }
             loginAllowed = false
             loginUserID =
             controllable = true
             reason = 0
             }
             18211: Jul 14 11:29:00.063 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
              queuing com.cisco.cti.protocol.DeviceRegisteredEvent
             18212: Jul 14 11:29:00.064 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
              handling event com.cisco.cti.protocol.DeviceRegisteredEvent[292]
             18213: Jul 14 11:29:00.065 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) Received
              DeviceRegisteredEvent
             18214: Jul 14 11:29:00.067 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
              Response: com.cisco.cti.protocol.DeviceRegisterDeviceResponse {
             sequenceNumber = 295
             callManagerID = 16777227
             deviceID = 37
             deviceInfo = com.cisco.cti.protocol.DeviceInfo {
             name = cue_site1_p03
             type = 72
             allowsRegistration = true
             }
             }
             18215: Jul 14 11:29:00.068 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) DeviceMap: opening
              device "cue_site1_p03"
             18216: Jul 14 11:29:00.069 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
              DeviceLineUpdateThread: queuing com.cisco.cti.client.implementation.Device
             18217: Jul 14 11:29:00.070 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p03
              (16777227,37) reopening line 28003(0,0)
             18218: Jul 14 11:29:00.072 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
              [ProviderRetryThread] sending: com.cisco.cti.protocol.LineOpenRequest {
             sequenceNumber = 297
             deviceName = cue_site1_p03
             lineName = 28003
             filter = com.cisco.cti.protocol.LineEventFilter {
             callS
             partyInfoChanged = true
             bExistingCallEvent = true
             bNewCallEvent = true
             bLineCfwdAllStatus = true
             }
             disableAutoRecovery = false
             }
             18219: Jul 14 11:29:00.096 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
              Event: com.cisco.cti.protocol.DeviceInServiceEvent {
             eventSequence = 293
             deviceCallManagerID = 16777227
```

```
deviceID = 37
}
18220: Jul 14 11:29:00.097 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceInServiceEvent
18221: Jul 14 11:29:00.098 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceInServiceEvent[293]
18222: Jul 14 11:29:00.098 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Terminal
 "cue_site1_p03" in service
18223: Jul 14 11:29:00.099 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [cue_site1_p03]
 CiscoTermInServiceEv [#693]
18224: Jul 14 11:29:00.101 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 296
}
18225: Jul 14 11:29:00.102 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) CUE_Site1_VM(16777227,35)
 refreshing lines: previous=1 current=1 created=0 removed=0
18226: Jul 14 11:29:00.103 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p04(16777227,36)
 updating lines
18227: Jul 14 11:29:00.104 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 298
deviceName = cue_site1_p04
}
18228: Jul 14 11:29:00.107 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineOpenResponse {
sequenceNumber = 297
callManagerID = 16777227
lineID = 37
}
18229: Jul 14 11:29:00.108 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening device
 (P1-site1cue) cue_site1_p02(0,0)
18230: Jul 14 11:29:00.112 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceRegisterDeviceRequest {
sequenceNumber = 299
deviceName = cue_site1_p02
ipAddr = 1802113708
rtpPortNumber = 16388
mediaSpecificationTimeout = 0
mediaCaps = 2@[
com.cisco.cti.protocol.MediaCapability {
payloadCapability = 4
maxFramesPerPacket = 30
bitRate = 1
},
com.cisco.cti.protocol.MediaCapability {
payloadCapability = 2
maxFramesPerPacket = 30
bitRate = 1
}]
filter = com.cisco.cti.protocol.DeviceEventFilter {
deviceModeChanged = false
keyPressed = false
featureButtonPressed = false
lampModeChanged = false
ringModeChanged = false
displayChanged = false
startTransmission = true
stopTransmission = true
startReception = true
stopReception = true
softKeyPressed = false
deviceData = true
}
disableAutoRecovery = false
}
```

```
18231: Jul 14 11:29:00.116 EDT %JTAPI-PROTOCOL-7-UNK:(P1-1 294
lineCallManagerID = 16777227
lineID = 37
}
18232: Jul 14 11:29:00.117 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.LineInServiceEvent
18233: Jul 14 11:29:00.118 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.LineInServiceEvent[294]
18234: Jul 14 11:29:00.119 EDT %JTAPI-CTI-7-UNK:(P1-site1cue){Line:28003(16777227,37)}
 LineInServiceEvent
18235: Jul 14 11:29:00.120 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Address "28003"
 in service
18236: Jul 14 11:29:00.120 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [28003]
 CiscoAddrInServiceEv [#694]
18237: Jul 14 11:29:00.121 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@2f3a76e1]
 ObserverProxy.queueEvents: queuing asynchronously
18238: Jul 14 11:29:00.122 EDT %JTAPI-MISC-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@2f3a76e1):
 queuing com.cisco.jtapi.JtapiAddressEventSet
18239: Jul 14 11:29:00.123 EDT %JTAPI-JTAPIIMPL-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@2f3a76e1):
 delivering JAES[1]
18240: Jul 14 11:29:00.123 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@2f3a76e1]
 ObserverProxy.deliverEvents()
18241: Jul 14 11:29:00.139 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@2f3a76e1]
 ObserverProxy.deliverEvents() completed
18242: Jul 14 11:29:00.141 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227ceNumber = 298
enumerationHandle = 15
}
18243: Jul 14 11:29:00.142 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 300
enumerationHandle = 15
count = 10
}
18244: Jul 14 11:29:00.147 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.DeviceRegisteredEvent {
eventSequence = 295
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p02
type = 72
allowsRegistration = true
}
loginAllowed = false
loginUserID =
controllable = true
reason = 0
}
18245: Jul 14 11:29:00.147 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceRegisteredEvent
18246: Jul 14 11:29:00.148 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceRegisteredEvent[295]
18247: Jul 14 11:29:00.149 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) Received
 DeviceRegisteredEvent
18248: Jul 14 11:29:00.151 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceRegisterDeviceResponse {
sequenceNumber = 299
callManagerID = 16777227
deviceID = 38
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p02
type = 72
```

```
allowsRegistration = true
}
}
18249: Jul 14 11:29:00.152 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) DeviceMap: opening
 device "cue_site1_p02"
18250: Jul 14 11:29:00.154 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 DeviceLineUpdateThread: queuing com.cisco.cti.client.implementation.Device
18251: Jul 14 11:29:00.155 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p02(16777227,38)
 reopening line 28002(0,0)
18252: Jul 14 11:29:00.157 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.LineOpenRequest {
sequenceNumber = 301
deviceName = cue_site1_p02
lineName = 28002
filter = com.cisco.cti.protocol.LineEventFilter {
callStateChanged = true
dtmf = true
ring = false
toneChanged = false
globalCallHandleChanged = true
openReceiveChannel = false
partyInfoChanged = true
bExistingCallEvent = true
bNewCallEvent = true
bLineCfwdAllStatus = true
}
disableAutoRecovery = false
}
18253: Jul 14 11:29:00.161 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.DeviceInServiceEvent {
eventSequence = 296
deviceCallManagerID = 16777227
deviceID = 38
}
18254: Jul 14 11:29:00.161 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceInServiceEvent
18255: Jul 14 11:29:00.162 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceInServiceEvent[296]
18256: Jul 14 11:29:00.163 EDT %JTAPI-JTAPIIMPL-7-UNKscoTermInServiceEv [#695]
18258: Jul 14 11:29:00.166 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 300
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28004
permanentLineID = 1897211172
}]
more = false
}
18259: Jul 14 11:29:00.188 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 302
enumerationHandle = 15
}
18260: Jul 14 11:29:00.192 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineOpenResponse {
sequenceNumber = 301
callManagerID = 16777227
lineID = 38
}
18261: Jul 14 11:29:00.193 EDT %JTAPI-CTI-7-UNK:(P1-14.80.227.127) reopening
 device (P1-site1cue) cue_site1_p01(0,0)
18262: Jul 14 11:29:00.197 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [ProviderRetryThread] sending: com.cisco.cti.protocol.DeviceRegisterDeviceRequest {
sequenceNumber = 303
```

```
                    deviceName = cue_site1_p01
                    ipAddr = 1802113708
                    rtpPortNumber = 16390
                    mediaSpecificationTimeout = 0
                    mediaCaps = 2@[
                    com.cisco.cti.protocol.MediaCapability {
                    payloadCapability = 4
                    maxFramesPerPacket = 30
                    bitRate = 1
                    },
                    com.cisco.cti.protocol.MediaCapability {
                    payloadCapability = 2
                    maxFramesPerPacket = 30
                    bitRate = 1
                    }]
                    filter false
                    featureButtonPressed = false
                    lampModeChanged = false
                    ringModeChanged = false
                    displayChanged = false
                    startTransmission = true
                    stopTransmission = true
                    startReception = true
                    stopReception = true
                    softKeyPressed = false
                    deviceData = true
                    }
                    disableAutoRecovery = false
                    }
18263: Jul 14 11:29:00.202 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.LineInServiceEvent {
eventSequence = 297
lineCallManagerID = 16777227
lineID = 38
}
18264: Jul 14 11:29:00.202 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.LineInServiceEvent
18265: Jul 14 11:29:00.204 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.LineInServiceEvent[297]
18266: Jul 14 11:29:00.204 EDT %JTAPI-CTI-7-UNK:(P1-site1cue){Line:28002(16777227,38)}
 LineInServiceEvent
18267: Jul 14 11:29:00.205 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Address "28002"
 in service
18268: Jul 14 11:29:00.206 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [28002]
 CiscoAddrInServiceEv [#696]
18269: Jul 14 11:29:00.207 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@6d4a36e0]
 ObserverProxy.queueEvents: queuing asynchronously
18270: Jul 14 11:29:00.207 EDT %JTAPI-MISC-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@6d4a36e0):
 queuing com.cisco.jtapi.JtapiAddressEventSet
18271: Jul 14 11:29:00.208 EDT %JTAPI-JTAPIIMPL-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@6d4a36e0):
 delivering JAES[1]
18272: Jul 14 11:29:00.209 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@6d4a36e0]
 ObserverProxy.deliverEvents()
18273: Jul 14 11:29:00.218 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@6d4a36e0]
 ObserverProxy.deliverEvents() completed
18274: Jul 14 11:29:00.220 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 302
}
18275: Jul 14 11:29:00.222 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p04(16777227,36)
 refreshing lines: previous=1 current=1 created=0 removed=0
```

```
18276: Jul 14 11:29:00.223 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p03(16777227,37)
 updating lines
18277: Jul 14 11:29:00.224 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 304
deviceName = cue_site1_p03
}
18278: Jul 14 11:29:00.231 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue)[Thread-37][28002]Request:
 setMessageWaiting ( 2104,true )
18279: Jul 14 11:29:00.232 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) [Thread-37]
 sending: com.cisco.cti.protocol.LineSetMessageWaitingRequest {
sequenceNumber = 305
lineCallManagerID = 16777227
lineID = 38
lineName = 2104
lampMode = 2
}
1828PROTOCOL-7-UNK:(P1-14.80.227.127) received Event:
 com.cisco.cti.protocol.DeviceRegisteredEvent {
eventSequence = 298
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p01
type = 72
allowsRegistration = true
}
loginAllowed = false
loginUserID =
controllable = true
reason = 0
}
18281: Jul 14 11:29:00.237 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceRegisteredEvent
18282: Jul 14 11:29:00.238 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceRegisteredEvent[298]
18283: Jul 14 11:29:00.238 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) Received
 DeviceRegisteredEvent
18284: Jul 14 11:29:00.240 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceRegisterDeviceResponse {
sequenceNumber = 303
callManagerID = 16777227
deviceID = 39
deviceInfo = com.cisco.cti.protocol.DeviceInfo {
name = cue_site1_p01
type = 72
allowsRegistration = true
}
}
18285: Jul 14 11:29:00.242 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) DeviceMap: opening
 device "cue_site1_p01"
18286: Jul 14 11:29:00.242 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 DeviceLineUpdateThread: queuing com.cisco.cti.client.implementation.Device
18287: Jul 14 11:29:00.244 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p01(16777227,39)
 reopening line 28001(0,0)
18288: Jul 14 11:29:00.246 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.
sequenceNumber = 306
deviceName = cue_site1_p01
lineName = 28001
filter = com.cisco.cti.protocol.LineEventFilter {
callStateChanged = true
dtmf = true
ring = false
toneChanged = false
globalCallHandleChanged = true
openReceiveChannel = false
partyInfoChanged = true
```

```
bExistingCallEvent = true
bNewCallEvent = true
bLineCfwdAllStatus = true
}
disableAutoRecovery = false
}
18289: Jul 14 11:29:00.249 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received Event:
 com.cisco.cti.protocol.DeviceInServiceEvent {
eventSequence = 299
deviceCallManagerID = 16777227
deviceID = 39
}
18290: Jul 14 11:29:00.250 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.DeviceInServiceEvent
18291: Jul 14 11:29:00.251 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.DeviceInServiceEvent[299]
18292: Jul 14 11:29:00.252 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Terminal
 "cue_site1_p01" in service
18293: Jul 14 11:29:00.253 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [cue_site1_p01]
 CiscoTermInServiceEv [#697]
18294: Jul 14 11:29:00.255 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 304
enumerationHandle = 16
}
18295: Jul 14 11:29:00.268 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 307
enumerationHandle = 16
count = 10
}
18296: Jul 14 11:29:00.271 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineSetMessageWaitingResponse {
sequenceNumber = 305
}
18297: Jul 14 11:29:00.290 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineOpenResponse {
sequenceNumber = 306
callManagerID = 16777227
lineID = 39
}
18298: Jul 14 11:29:00.291 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 ProviderRetryThread stopping retries
18299: Jul 14 11:29:00.292 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127)
 ProviderRetryThread waiting until notified
18300: Jul 14 11:29:00.294 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Event: com.cisco.cti.protocol.LineInServiceEvent {
eventSequence = 300
lineCallManagerID = 16777227
lineID = 39
}
18301: Jul 14 11:29:00.294 EDT %JTAPI-MISC-7-UNK:(P1-14.80.227.127) EventThread:
 queuing com.cisco.cti.protocol.LineInServiceEvent
18302: Jul 14 11:29:00.295 EDT %JTAPI-CTIIMPL-7-UNK:(P1-14.80.227.127) EventThread
 handling event com.cisco.cti.protocol.LineInServiceEvent[300]
18303: Jul 14 11:29:00.296 EDT %JTAPI-CTI-7-UNK:(P1-site1cue){Line:28001(16777227,39)}
 LineInServiceEvent
18304: Jul 14 11:29:00.297 EDT %JTAPI-JTAPIIMPL-7-UNK:(P1-site1cue) Address "28001"
 in service
18305: Jul 14 11:29:00.298 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue) [28001]
 CiscoDT %JTAPI-MISC-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@324e36e0):
 queuing com.cisco.jtapi.JtapiAddressEventSet
18308: Jul 14 11:29:00.300 EDT %JTAPI-JTAPIIMPL-7-UNK:ObserverThread
 (com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@324e36e0):
```

```
 delivering JAES[1]
18309: Jul 14 11:29:00.301 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@324e36e0]
 ObserverProxy.deliverEvents()
18310: Jul 14 11:29:00.327 EDT %JTAPI-JTAPIIMPL-7-UNK:
 [com.cisco.wf.subsystems.jtapi.TAPIPortGroup$ServiceAddressObserver@324e36e0]
 ObserverProxy.deliverEvents() completed
18311: Jul 14 11:29:00.376 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 307
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28003
permanentLineID = 2109152574
}]
more = false
}
18312: Jul 14 11:29:00.377 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 308
enumerationHandle = 16
}
18313: Jul 14 11:29:00.381 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 308
}
18314: Jul 14 11:29:00.382 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p03(16777227,37)
 refreshing lines: previous=1 current=1 created=0 removed=0
18315: Jul 14 11:29:00.383 EDT %JTAPI-CTI-7-UNK EDT %JTAPI-PROTOCOL-7-UNK:
 (P1-14.80.227.127) [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 309
deviceName = cue_site1_p02
}
18317: Jul 14 11:29:00.387 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 309
enumerationHandle = 17
}
18318: Jul 14 11:29:00.389 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 310
enumerationHandle = 17
count = 10
}
18319: Jul 14 11:29:00.397 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 310
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28002
permanentLineID = 1035863534
}]
more = false
}
18320: Jul 14 11:29:00.398 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 311
enumerationHandle = 17
}
18321: Jul 14 11:29:00.403 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 311
```

```
}
18322: Jul 14 11:29:00.405 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p02(16777227,38)
 refreshing lines: previous=1 current=1 created=0 removed=0
18323: Jul 14 11:29:00.405 EDT %JTAPI-CTI-7-UNK:(P1-site1cue) cue_site1_p01(16777227,39)
 updating lines
18324: Jul 14 11:29:00.406 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.DeviceGetLineInfoRequest {
sequenceNumber = 312
deviceName = cue_site1_p01
}
18325: Jul 14 11:29:00.409 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.DeviceGetLineInfoResponse {
sequenceNumber = 312
enumerationHandle = 18
}
18326: Jul 14 11:29:00.411 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoFetchRequest {
sequenceNumber = 313
enumerationHandle = 18
count = 10
}
18327: Jul 14 11:29:00.419 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoFetchResponse {
sequenceNumber = 313
info = 1@[
com.cisco.cti.protocol.LineInfo {
name = 28001
permanentLineID = 1084634008
}]
more = false
}
18328: Jul 14 11:29:00.476 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
 [(P1-14.80.227.127) DeviceLineUpdateThread] sending:
 com.cisco.cti.protocol.GetLineInfoCloseRequest {
sequenceNumber = 314
enumerationHandle = 18
}
18329: Jul 14 11:29:00.480 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.GetLineInfoCloseResponse {
sequenceNumber = 314
}
18330: Jul 14 11:29:00.521 EDT %JTAPI-CTI-7-UNK:(P1-site1cue)
18331: Jul 14 11:29:01.514 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue)[Thread-36][28001]
 Request: setMessageWaiting ( 2104,true )
18332: Jul 14 11:29:01.516 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) [Thread-36]
 sending: com.cisco.cti.protocol.LineSetMessageWaitingRequest {
  sequenceNumber    = 315
  lineCallManagerID = 16777227
  lineID            = 39
  lineName          = 2104
  lampMode          = 2
  }
18333: Jul 14 11:29:01.520 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineSetMessageWaitingResponse {
  sequenceNumber = 315
  }
18334: Jul 14 11:29:02.807 EDT %JTAPI-JTAPI-7-UNK:(P1-site1cue)[Thread-37][28001]
 Request: setMessageWaiting ( 2103,false )
18335: Jul 14 11:29:02.808 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) [Thread-37]
 sending: com.cisco.cti.protocol.LineSetMessageWaitingRequest {
  sequenceNumber    = 316
  lineCallManagerID = 16777227
  lineID            = 39
  lineName          = 2103
```

```
    lampMode          = 1
    }
18336: Jul 14 11:29:02.815 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
 Response: com.cisco.cti.protocol.LineSetMessageWaitingResponse {
   sequenceNumber = 316
   }
18337: Jul 14 11:29:26.129 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
  server Heartbeat: com.cisco.cti.protocol.Heartbeat {
   }
18338: Jul 14 11:29:41.158 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
  [HeartbeatSendThread] sending: com.cisco.cti.protocol.Heartbeat {
   }
18339: Jul 14 11:29:56.473 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
  server Heartbeat: com.cisco.cti.protocol
   }
18340: Jul 14 11:30:11.480 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
  [HeartbeatSendThread] sending: com.cisco.cti.protocol.Heartbeat {
   }
18341: Jul 14 11:30:26.172 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127) received
  server Heartbeat: com.cisco.cti.protocol.Heartbeat {
   }
18342: Jul 14 11:30:41.503 EDT %JTAPI-PROTOCOL-7-UNK:(P1-14.80.227.127)
  [HeartbeatSendThread] sending: com.cisco.cti.protocol.Heartbeat {
   }
```

General MWI and Voice Mail Traces

Aside from the integration issues mentioned in the section MWI Overview, it is possible to troubleshoot the delivery and MWI events in the system with the **trace** facility. This usually falls within the category of general voice mail troubleshooting. But, since these issues often overlap, it is good to point out some basics.

This section provides an example of the **trace voicemail all** command. A call is made to user 11044, and it is forwarded to voice mail. At a minimum, you should issue the **trace voicemail vmxl all** command and the **trace voicemail mwi all** command.

Note: The user presses **2** in order to mark the message as urgent. The MWI event that this example indicates is actually *after* the signaling occurs. The SIP/JTAPI signal occurs, and then this message prints in order to notify you that it was successful.

Note: There is a call ID that helps track a particular call if there are multiple calls simultaneously. In this case, the call ID is 0x000000037e11d669. If this were a Cisco CallManager Express−integrated system, you should also issue the **trace ccn stacksip dbug** command. This command shows more clearly when digits are entered, as well as when disconnect and other events occur.

```
cue-3660-41a>show trace buffer long
Press <CTRL-C> to exit...
5047 07/15 13:33:44.198 voicemail ldap "getUserByPhoneNo" 11044
5047 07/15 13:33:44.200 voicemail ldap "getUserByPhoneNo: userDn."
 /sw/local/users/user3
5047 07/15 13:33:44.200 voicemail ldap 0 getAttributeValue:
 /sw/local/users/user3/Language/preferredLanguage
5047 07/15 13:33:44.201 voicemail ldap 0 getAttributeValue:
 /sw/local/users/user3/TelephoneNumbers/primaryExtension
5047 07/15 13:33:44.202 voicemail database 0 Got connection: 1, inUse: 1, active: 3
5047 07/15 13:33:44.202 voicemail database "SQL: " select mailboxid from vm_mbxusers
 where owner=true and userdn='/sw/local/users/user3';
5047 07/15 13:33:44.204 voicemail database "Database query results"
 PERSONAL_0000000000000000000000003
5047 07/15 13:33:44.204 voicemail database 0 Freed connection: 1, inUse: 0, active: 3
5047 07/15 13:33:44.255 voicemail database 0 Got connection: 2, inUse: 1, active: 3
5047 07/15 13:33:44.255 voicemail database "SQL: " 0x000000037e11d669 select mailboxid
```

```
 from vm_mbxusers where owner=true and userdn='/sw/local/users/user3';
5047 07/15 13:33:44.257 voicemail database "Database query results" 0x000000037e11d669
 PERSONAL_0000000000000000000000003
5047 07/15 13:33:44.258 voicemail database "SQL: " 0x000000037e11d669 select distinct
 vm_mbxusers.mailboxid, orphanedtime from vm_mbxusers, vm_mailbox where
 vm_mailbox.mailboxid=vm_mbxusers.mailboxid and (userdn='/sw/local/users/user3') and
 orphanedtime=0 and owner=false;
5047 07/15 13:33:44.265 voicemail database 0 Freed connection: 2, inUse: 0, active: 3
18885 07/15 13:33:44.279 **voicemail ldap "getSpokenNameByName: userDn."
 /sw/local/users/user3**
18885 07/15 13:33:44.279 voicemail ldap "normalizeDN" /sw/local/users/user3
18885 07/15 13:33:44.279 voicemail ldap "getSpokenName: dn." uid=user3,ou=users,
 ou=branch123,o=cisco.com
18885 07/15 13:33:44.292 voicemail database 0 Got connection: 0, inUse: 1, active: 3
18885 07/15 13:33:44.293 voicemail database "SQL: " 0x000000037e11d669 select
 greetingid,greetingtype,messagelength,messagesize,greetingoid from vm_greeting
 where greetingtype=10 and mailboxid='PERSONAL_0000000000000000000000003';
18885 07/15 13:33:44.296 voicemail database 0 Freed connection: 0, inUse: 0, active: 3
1989 07/15 13:33:44.324 **voicemail vxml "Sorry. Extension" 0x000000037e11d669
 AvPHGreetENU021.wav**
1989 07/15 13:33:44.334 voicemail vxml 0 0x000000037e11d669 11044
1989 07/15 13:33:44.334 **voicemail vxml "is not available." 0x000000037e11d669
 AvSubGreetingsENU018.wav**
1989 07/15 13:33:44.348 **voicemail vxml "You may record your message at the tone.
 When you are finished, press #" 0x000000037e11d669** AvSubSendMsgENU050.wav
2043 07/15 13:33:51.757 voicemail agc "AGC processing buffer" 8160 0
2043 07/15 13:33:52.777 voicemail agc "AGC processing buffer" 8160 0
2043 07/15 13:33:53.797 voicemail agc "AGC processing buffer" 8160 0
2043 07/15 13:33:54.817 voicemail agc "AGC processing buffer" 8160 0
2043 07/15 13:33:55.837 voicemail agc "AGC processing buffer" 8160 0
2043 07/15 13:33:56.257 voicemail agc "AGC processing buffer" 8160 0
1989 07/15 13:33:56.627 **voicemail vxml "To send this message with normal
 priority, press 1. To send this message with urgent priority, press 2."
 0x000000037e11d669** AvPHGreetENU002.wav
1989 07/15 13:33:56.627 **voicemail vxml "To listen to your message, press 3.
 To re-record it, press 4." 0x000000037e11d669** AvAesopCustomENU004.wav
1989 07/15 13:33:56.632 **voicemail vxml "To cancel press 6"
 0x000000037e11d669** AvPHGreetENU403.wav
1989 07/15 13:34:03.395 **voicemail vxml "callerMsgRecord.record_message.action"
 0x000000037e11d669  2**
18885 07/15 13:34:03.402 voicemail ldap "getUserByPhoneNo" undefined
18885 07/15 13:34:03.407 voicemail ldap "getUserByPhoneNo: No entry found."
18885 07/15 13:34:03.407 voicemail message "Creating Message" 1089912843407_0
18885 07/15 13:34:03.407 voicemail message "Message Length" 5398, Message Size: 44218
18885 07/15 13:34:03.407 voicemail mailbox "Sending message(s) from"
 0x000000037e11d669 /sw/local/users/user3
18885 07/15 13:34:03.407 voicemail mailbox "Sending message to"
 0x000000037e11d669 11044
18885 07/15 13:34:03.408 voicemail database 0 Got connection: 1, inUse: 1, active: 3
18885 07/15 13:34:03.408 voicemail mailbox "Message received" 0x000000037e11d669
 PERSONAL_0000000000000000000000003,1089912843407_0
18885 07/15 13:34:03.408 voicemail database "SQL: " 0x000000037e11d669 select count
 (messageid) from vm_message where messageid='1089912843407_0';
18885 07/15 13:34:03.413 voicemail database "Database query results"
 0x000000037e11d669 0
18885 07/15 13:34:03.413 voicemail database "SQL: " 0x000000037e11d669 update
 vm_message set messageid='1089912843407_0',messagetype=1,sender='Unknown',
 urgent=true,private=false,attachedmsgid=null where messageId='OID_16650';
18885 07/15 13:34:03.559 voicemail database "SQL: " 0x000000037e11d669 insert
 into vm_usermsg values('PERSONAL_0000000000000000000000003',
 '1089912843407_0',1,1089912843407);
18885 07/15 13:34:03.564 voicemail database "SQL: " 0x000000037e11d669 select
 totalmessagetime from vm_mailbox where mailboxid='PERSONAL_0000000000000000000000003'
 for update;
18885 07/15 13:34:03.566 voicemail database "Database query results"
 0x000000037e11d669 28061
```

```
18885 07/15 13:34:03.567 voicemail database "SQL: " 0x000000037e11d669 update
 vm_mailbox set totalmessagetime=33459 where
 mailboxid='PERSONAL_0000000000000000000000003';
18885 07/15 13:34:03.570 voicemail database "Commiting transaction"
 0x000000037e11d669
18885 07/15 13:34:03.601 voicemail ldap 0 getAttributeValue:
 /sw/local/users/user3/TelephoneNumbers/primaryExtension
18885 07/15 13:34:03.601 voicemail mwi "setMessageWaiting"
 0x000000037e11d669 11044,true
18885 07/15 13:34:03.602 voicemail mwi " job state" adding job
1677 07/15 13:34:03.602 voicemail mwi " job state"
 http://localhost:8080/mwiapp?extn=11044&state=1
18885 07/15 13:34:03.677 voicemail database 0 Freed connection: 1, inUse: 0,
 active: 3
1989 07/15 13:34:03.688 voicemail vxml "Thank you. Your message has been sent."
 0x000000037e11d669 AvPHGreetENU008.wav
1989 07/15 13:34:03.700 voicemail  "Hello, Unity-lite messaging system. If you
 have a mailbox in this system press '*', Otherwise please hold for an operator."
 0x000000037e11d669 AvAesopCustomENU001.wav
1989 07/15 13:34:07.756 voicemail vxml 0 0x000000037e11d669 TIMEOUT
1989 07/15 13:34:07.757 voicemail vxml 0 0x000000037e11d669 TIMEOUT
```

Related Information

- **Setting Up Phone View in Cisco Unity Connection 8.x**
- **Cisco CallManager Express 3.1 System Administrator Guide**
- **Cisco Unity Express 2.3 Installation and Upgrade Guide**
- **Cisco Unity Express GUI Administrator Guide for Cisco CallManager, Release 2.1**
- **Voice Technology Support**
- **Voice and Unified Communications Product Support**
- **Technical Support & Documentation – Cisco Systems**

Updated: Feb 08, 2012   Document ID: 60081