Risoluzione dei problemi MWI di Unity Express

Sommario

Introduzione

Questo documento offre una panoramica della funzionalità MWI (Message Waiting Indication) di Cisco Unity Express.

Prerequisiti

Requisiti

I lettori di questo documento devono conoscere l'interfaccia della riga di comando (CLI) di Cisco Unity Express.

Componenti usati

Il riferimento delle informazioni contenute in questo documento è Cisco Unity Express versione 1.0/2.3.x/8.x o successive. Tutte le configurazioni di esempio e l'output dello schermo sono tratti da Cisco Unity Express versione 1.1.1.

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

Convenzioni

Per ulteriori informazioni sulle convenzioni usate, consultare il documento Cisco sulle convenzioni nei suggerimenti tecnici.

Panoramica di MWI

L'operazione MWI fornisce agli utenti registrati con Cisco CallManager Express o CallManager un'indicazione visiva della presenza di nuovi messaggi di posta vocale. Il protocollo MWI non funziona se Cisco Unity Express è integrato con Cisco CallManager e il sistema è in modalità SRST (Survivable Remote Site Telephony) a causa di un'interruzione della rete WAN.

Se Cisco Unity Express è integrato con Cisco CallManager Express, viene effettuata una chiamata SIP (Session Initiation Protocol) all'*indirizzo extension_MWI_on/off_number@CallManager_Express_IP_address* quando un nuovo messaggio di posta vocale arriva in una cassetta postale dell'utente. La chiamata SIP viene avviata anche quando l'utente recupera tutti i nuovi messaggi. Corrisponde a un numero phone-dn sul router Cisco CallManager Express. Il numero di telefono-dn ha il numero MWI più un numero di cifre jolly uguale al numero di cifre nell'estensione degli abbonati a Cisco Unity Express. Si supponga, ad esempio, che il numero MWI-on per la cassetta postale 12345 sia 420. L'indirizzo IP di Cisco CallManager Express è 10.2.3.6. In questo esempio, il messaggio viene inviato a 42012345@10.2.3.6. Il numero di telefono-dn con il parametro di configurazione "mwi on" impostato è "420...".

Per le integrazioni di Cisco CallManager, il protocollo JTAPI (Java Telephony Application Programming Interface) accende direttamente una lampada. Non è necessario chiamare un numero specifico. Il protocollo JTAPI supporta un comando **setMessageWaiting** per la gestione degli eventi MWI. Pertanto, il funzionamento delle MWI deve essere indipendente dalla configurazione delle estensioni MWI in Cisco CallManager. Tenere presente che le MWI non funzionano quando Cisco Unity Express è in modalità SRST. L'aggiornamento MWI completo viene eseguito solo dopo la nuova registrazione di Cisco Unity Express con Cisco CallManager e dopo che i telefoni IP non sono più in modalità di fallback di CallManager.

La maggior parte dei problemi si verifica con l'integrazione tra Cisco CallManager Express/CallManager e Cisco Unity Express. Tenere presente che MWI potrebbe non essere correlato a una lampada fisica. Se il numero che riceve il messaggio non è una linea primaria in un telefono, può solo ricevere una notifica sulla busta sul display del telefono. In Cisco CallManager è possibile configurare la modalità di gestione di MWI per ciascuna linea. Se solo uno o due utenti hanno un problema, è possibile iniziare a cercare il problema qui.

Per ricevere un MWI, un numero di directory deve disporre di una cassetta postale valida sul sistema Cisco Unity Express. Il numero deve essere associato a un utente che deve disporre di una cassetta postale. Prima di iniziare il debug e di adottare misure avanzate per la risoluzione dei problemi, è possibile eseguire una semplice operazione per risolvere i problemi: verificare che l'utente abbia eseguito l'accesso alla cassetta postale e che sia in grado di inviare e recuperare messaggi di posta vocale.

Dalla GUI o dalla CLI, è possibile trovare un utente con cui eseguire il test. In questo caso, si tratta dell'utente 3. È possibile individuare l'estensione configurata per l'utente, determinare lo stato della cassetta postale dell'utente (abilitata o non abilitata, tra le altre informazioni) e determinare se l'utente ha messaggi nuovi o vecchi. In questo esempio, viene usata la CLI per risolvere i problemi:

```
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>
```

Verificare che l'utente esista, che disponga di un numero associato e che non disponga di messaggi. Se questi elementi sono veri, lo stato MWI deve essere disattivato.

Nota: l'indirizzo E.164 (ITU-T) non viene utilizzato per MWI. È possibile utilizzare solo il numero di telefono principale.

Problemi di integrazione con Cisco Unity Express

MWI con Cisco CallManager Express

Ènecessario verificare la configurazione prima di eseguire qualsiasi altra operazione. Su Cisco CallManager Express, visualizzare la configurazione con il comando **show running-config**. Più direttamente, è possibile usare il comando **show telephony-service telephone-dn**. Viene

visualizzato un output simile al seguente:

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

Questo output illustra alcune informazioni importanti. Il numero per MWI on è 11099. Il numero per MWI off è 11098. Il numero di cifre nel dial plan è cinque. (I cinque punti [. . . . .] che seguono il codice MWI on o off lo mostrano.) In altre parole, MWI funziona solo per un numero di directory (DN) che contiene esattamente cinque cifre.

Sul lato Cisco Unity Express, è possibile verificare la configurazione e anche la licenza. Molto spesso viene caricata una licenza Cisco CallManager anziché una licenza per CallManager Express. Per verificare questa condizione, eseguire il comando **show software licenses** da Cisco Unity Express:

```
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
```

Se invece la modalità dell'applicazione è `CCM`, Cisco CallManager, tutto funziona *tranne* MWI. Sfortunatamente, se la licenza è sbagliata, l'unica opzione è ricreare l'immagine del software e riapplicare la licenza. Non è possibile salvare o ripristinare messaggi o configurazioni.

Verificare quindi la configurazione. Per visualizzare la configurazione, usare il comando **show run** o il comando **show cn application**:

```
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
```

Nota: l'applicazione è abilitata e i numeri `MWI_OFF` e `MWI_ON` sono rispettivamente 11098 e 11099. Il sistema non ha un concetto del numero di cifre nelle estensioni; è sufficiente chiamare il numero MWI appropriato e aggiungere l'estensione della cassetta postale. Per indirizzare correttamente la chiamata, il sistema Cisco CallManager Express deve avere un peer di composizione con il numero di punti appropriato nel modello di destinazione.

Infine, verificare che l'indirizzo IP del gateway SIP di Cisco Unity Express punti all'indirizzo IP corretto di Cisco CallManager Express.

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

Se l'operazione non è corretta, le chiamate non vengono inviate al Cisco CallManager Express corretto. Le chiamate hanno esito negativo.

Esistono due modi per iniziare a risolvere i problemi relativi alla segnalazione. Dal lato Cisco Unity Express, in genere è più facile disabilitare prima le tracce predefinite; quindi, riattivarle in base alle esigenze. A tal fine, usare il comando **no trace all**. Il comando trace con cui iniziare è **trace ccn stacksip debug**.

Nota: per ulteriori informazioni sull'analisi, consultare il documento Impostazione e raccolta dei dati di traccia in CUE.

Prima di inviare un messaggio MWI, cancellare il buffer di traccia. Tutti i messaggi di analisi vengono scritti in questo buffer di memoria. Si desidera cancellarlo in modo che non sia necessario visualizzare tutti i messaggi precedenti quando si controlla dopo la chiamata di prova. A tale scopo, è possibile usare un semplice comando **clear trace**.

Inviare quindi il messaggio MWI. A tale scopo, utilizzare il comando **mwi refresh telephonenumber *xxxx***. È possibile aggiornare anche la GUI.

Infine, visualizzare il buffer di traccia e visualizzare l'output con il comando **show trace buffer long**. In questo esempio vengono evidenziati alcuni elementi importanti:

```
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"




      ;tag=f0a4ab8e-488
To:




      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
```

Come mostrato in questo output, l'utente invia un messaggio **INVITE** e Cisco CallManager Express risponde con un messaggio **Trying**. Non appena Cisco CallManager Express invia un messaggio di **chiamata**, viene inviato un messaggio di **annullamento**. Il numero MWI non risponde a una chiamata. La posizione di una chiamata al numero stesso è sufficiente per accendere o spegnere la lampada. In questo caso, è necessario sapere se la porta 11098 è attivata o disattivata. Inoltre, 11043 deve essere un'estensione valida in Cisco CallManager Express.Dopo aver raccolto tutte le tracce di Cisco Unity Express necessarie, la cosa migliore da fare è disabilitare tutte le tracce e quindi abilitare di nuovo le tracce predefinite. Utilizzare il comando clear trace all per disabilitare le tracce. Quindi, incollare il codice mostrato qui nella CLI di Cisco Unity Express per riabilitare tutte le tracce predefinite:Nota: in alternativa, è possibile ripristinare le tracce predefinite se si riavvia 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
```

Inoltre, è possibile diagnosticare facilmente tutti i messaggi SIP sul router Cisco CallManager Express. In genere, i messaggi debug ccsip e i supporti debug ccsip sono i comandi più utili. Quando è necessaria solo la segnalazione SIP, questa diagnosi è molto più rapida e Cisco Unity Express rileva informazioni meno necessarie. Se Cisco Unity Express invia la segnalazione all'indirizzo IP corretto di CallManager Express, la segnalazione SIP viene riflessa su ciascun server.Le chiamate da o verso Cisco Unity Express richiedono il protocollo G.711, che presenta un altro problema comune. Ad esempio, i debug possono mostrare questo pacchetto SIP del modulo Cisco CallManager Express:

```
Mar 11 10:09:13.767 EST: //-1/xxxxxxxxxxxx/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
```

Questo output indica che Cisco CallManager Express ha rifiutato la chiamata perché il messaggio `SIP INVITE` da Cisco Unity Express non corrisponde a un peer di composizione configurato per G.711. Per correggere il rifiuto della chiamata, è possibile aggiungere un peer di composizione specifico per il traffico MWI. L'esempio in questa sezione ha `11099.....` per `MWI on` e `11098.....` per `MWI disattivato`. È possibile aggiungere:

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

L'ultimo problema comune è che il traffico MWI corrisponde a un modello di conversione applicato a un peer di composizione, a una regola in ingresso VoIP o a un altro percorso. In alternativa, le regole COR (Class of Restriction) possono bloccare la chiamata. Tenere presente che, anche se si compone il numero MWI on/off e l'estensione per accendere il MWI, la chiamata non si comporta necessariamente nello stesso modo quando arriva tramite SIP. Per ulteriori informazioni sul CDR, consultare il documento relativo alla configurazione della classe di limitazioni (COR).In sintesi, verificare sempre i seguenti elementi:

- Èpresente una licenza Cisco CallManager Express. Eseguire il comando show software licenses. Con una licenza Cisco CallManager, tutto funziona *tranne* MWI.
- I numeri MWI on e off sono configurati in Cisco CallManager Express. Il numero di punti indica la lunghezza delle estensioni. Eseguire il comando show telephony-service telephone-dn.
- In Cisco Unity Express, i numeri MWI on e off sono configurati in modo da corrispondere ai numeri on e off di Cisco CallManager Express senza i punti. Il comando show ccn application visualizza questa condizione.

- Cisco Unity Express punta all'indirizzo IP corretto del server Cisco CallManager Express. Il comando show ccn subsystem sip visualizza questa condizione.
- Verificare che mwi sip outcall sia configurato con il comando ccnsubsystem sip.

Quindi, se il problema persiste, iniziare a risolvere il problema con il comando trace cn stacksip debug.Indicatori MWI (Message Waiting Indicators) (solo Cisco Unified CallManager Express)Sintomo: Dopo l'aggiornamento a una nuova versione di Cisco Unity Express, le MWI non si illuminano anche quando i messaggi vengono lasciati nelle cassette postali.

- Spiegazione - La procedura di aggiornamento ha rimosso l'indirizzo IP del sottosistema SIP (Session Initiation Protocol).
- Azione consigliata: riconfigurare l'indirizzo IP SIP in modo che punti al router Cisco Unified CME.

Errore: È stato rilevato un errore durante la ricerca durante la visualizzazione del messaggio

Quando si tenta di recuperare i messaggi, viene visualizzato il messaggio di errore `Ricerca in corso.`Completare la procedura descritta in Per abilitare la visualizzazione del telefono per un sistema telefonico per risolvere il problema.

Come risolvere i problemi relativi a un sistema Cisco CallManager Express

Per risolvere i problemi relativi al sistema Cisco CallManager Express, attenersi alla seguente procedura:

1. Immettere il comando show telephone per visualizzare tutti i telefoni registrati. Se non è registrato alcun telefono, eseguire le seguenti attività:Controllare la configurazione DHCP, che include il router predefinito e l'indirizzo del server TFTP (opzione 150).Per controllare che i file richiesti si trovino nella memoria flash del router, usare il comando dir.Verificare che il comando tftp-server sia impostato per i file richiesti.Usare il comando debug phone register mac-address per visualizzare l'attività di registrazione del telefono IP Cisco.Usare il comando debug ip dhcp per confermare l'operazione DHCP.
2. Immettere il comando show telephone per visualizzare tutti i telefoni registrati. Se i telefoni sono registrati e visualizzati, procedere come segue:Verificare che il collegamento del pulsante del telefono al numero di directory sia corretto.Verificare che i telefoni IP Cisco mostrino come registrati.Usare il display Settings (Impostazioni) sul telefono per verificare le impostazioni dei parametri IP sul telefono IP Cisco.Verificare che il numero di keepalive venga aggiornato quando si immette il comando show phone.Immettere il comando debug phone register mac-address per ripristinare il telefono e osservare la nuova registrazione per visualizzare i telefoni IP Cisco.Immettere il comando show telephone-dn summary per controllare lo stato delle linee telefoniche IP Cisco.Controllare l'indirizzo IP del telefono e provare a eseguire il ping dell'indirizzo.
3. Usare il comando debug phone keepalive per impostare il debug keepalive per i telefoni IP Cisco.
4. Usare il comando debug phone state per impostare il debug dello stato dei telefoni IP Cisco.

MWI con Cisco CallManager

Per le integrazioni di Cisco Unity Express con Cisco CallManager, è molto importante assicurarsi che Unity Express sia registrato e che disponga di tutte le informazioni di accesso corrette.Il primo passaggio consiste nel determinare se un telefono è in modalità SRST, se disponibile, per risolvere il problema. Accedere al router in cui è installato il modulo Cisco Unity Express. Quindi, usare il comando show telephone registered. I telefoni registrati non ricevono MWI, anche se Cisco Unity Express è stato correttamente registrato su 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
```

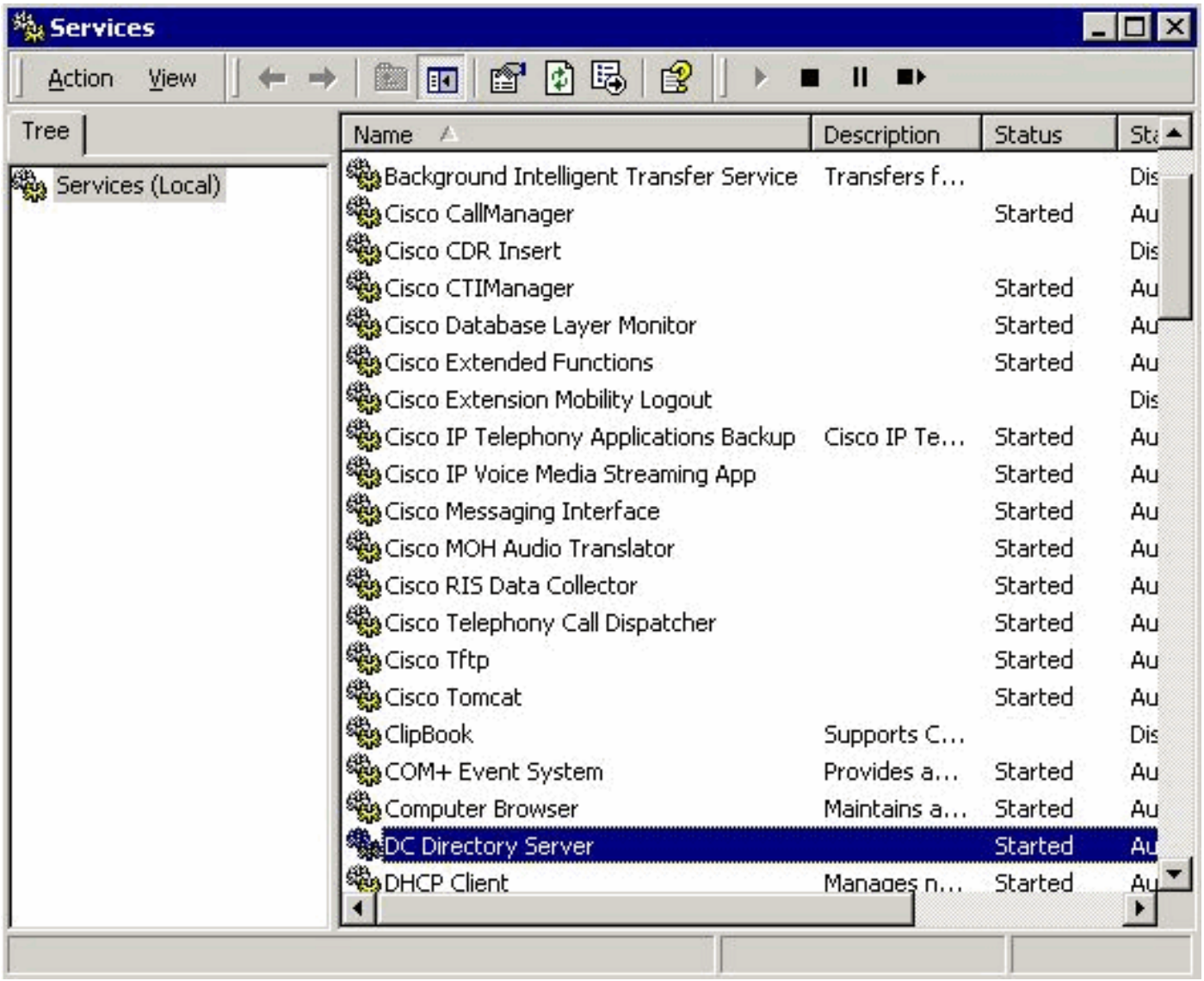
```
button 1: dn 1  number 2103  CM Fallback CH1   IDLE
button 2: dn 2  number 2199  CM Fallback CH1   IDLE


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
```

Se non ci sono telefoni nello stato di fallback di Cisco CallManager, indicato dallo stato `REGISTERED` (REGISTRATO), come mostrato in precedenza, SRST non è attivo per questi dispositivi. Il passaggio successivo consiste quindi nel verificare le configurazioni di Cisco Unity Express e Cisco CallManager per essere certi che Unity Express sia registrato in 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
```

In questo output vengono elencati tutti i numeri di directory dei punti di routing CTI (Computer Telephony Integration) e l'account JTAPI utilizzato da Cisco Unity Express per accedere a Cisco CallManager.Ènecessario verificare che Cisco Unity Express si registri correttamente su Cisco CallManager. In primo luogo, confermare che le porte CTI siano effettivamente registrate. Il modo più semplice per effettuare questa operazione è andare alla pagina Web di amministrazione di Cisco CallManager. Quindi, scegliere Periferica > Telefono e cercare le porte CTI elencate nell'output di cui sopra. I campi Stato e Indirizzo IP devono essere compilati completamente.

Se le porte non sono registrate, Cisco Unity Express non è in grado di comunicare con Cisco CallManager. Un'altra possibilità è che l'accesso non sia corretto. Per risolvere il problema, eseguire ping semplici tra il modulo Cisco Unity Express e Cisco CallManager. Se l'operazione riesce, verificare che Cisco CTIManager e i servizi directory, in questo caso il server delle directory DC, siano stati avviati. Dal server Cisco CallManager, scegliere Start > Programmi > Strumenti di amministrazione > Servizi per
verificare:



Èinoltre necessario verificare che l'account utente JTAPI, in questo esempio site1cue, esista. Per le porte CTI, i punti di instradamento e l'opzione Abilita uso applicazione CTI (Enable CTI Application Use) selezionata, Verificare inoltre la password.Un altro problema comune è lo spazio di ricerca delle chiamate delle porte CTI. Questo spazio di ricerca chiamate deve contenere le partizioni dei numeri di directory per cui si cerca di accendere la luce MWI. Ad esempio, lo spazio di ricerca delle chiamate per le porte CTI, non i punti di instradamento, deve contenere la partizione Line1 per impostare un MWI per l'estensione 1234 nella linea di partizione 1. Se lo spazio di ricerca delle chiamate per le porte CTI è Nessuno, solo le estensioni nella partizione Nessuno funzionano per MWI.Se la configurazione risulta corretta, è possibile abilitare la diagnostica JTAPI sul modulo Cisco Unity Express. Tuttavia, le opzioni enable e disable richiedono un riavvio. Questo livello di diagnostica va oltre le normali impostazioni di debug della traccia. Non lasciare questa opzione attivata, in particolare per il modulo AIM (Advanced Integration Module), in quanto scritture eccessive sulla scheda flash interna possono ridurre la durata della memoria flash.Usare il comando show ccn trace jtapi per visualizzare le tracce JTAPI correnti abilitate:Nota: per impostazione predefinita, tutte le tracce JTAPI sono disabilitate.

```
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
```

Per abilitare tutte le tracce, usare questi comandi:

```
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
```

Ora è necessario ricaricare il sistema. Eseguire gli stessi comandi ccn trace indicati sopra, ma anteporre a ciascun comando la parola chiave no per disabilitarlo in seguito. Ad esempio, no ccn trace jtapi debug all. Si tratta di un passo importante da ricordare, soprattutto per quanto riguarda l'AIM. La mancata esecuzione di questo passaggio influisce sulle prestazioni potenziali e riduce la durata della scheda Compact Flash sull'AIM.Dopo il riavvio, il sistema inizia a scrivere i file CiscoJtapi1.log e CiscoJtapi2.log, quando il primo è pieno.Per visualizzare questi log, usare il comando show log name CiscoJap1.log. È inoltre possibile copiare il file di registro in un server FTP e quindi visualizzare le informazioni non in linea. Il comando è copy log CiscoJap1.log url ftp://user:passwd@ftpservipaddr/.Con entrambi i metodi, vengono visualizzate tutte le informazioni JTAPI. Nell'esempio, il modulo Cisco Unity Express tenta di eseguire la registrazione, ma l'operazione non riesce a causa di un errore della WAN:

```
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
```

La traccia successiva mostra una registrazione completa di Cisco Unity Express su Cisco CallManager. In questo esempio, sono presenti otto porte CTI associate all'utente JTAPI. Tuttavia, poiché Cisco Unity Express è concesso in licenza solo per quattro porte, vengono utilizzate solo quattro porte. Inoltre, il sistema esegue automaticamente una risincronizzazione MWI completa dopo la registrazione di nuovo in 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-esponse:
 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 {
  }
```

Tracce generali MWI e segreteria telefonica

Oltre ai problemi di integrazione menzionati nella sezione Panoramica MWI, è possibile risolvere i problemi relativi alla consegna e agli eventi MWI nel sistema con la funzione di traccia. In genere questo tipo di risoluzione dei problemi rientra nella categoria di risoluzione dei problemi generali della casella vocale. Tuttavia, dato che questi problemi spesso si sovrappongono, è bene evidenziare alcuni elementi fondamentali.In questa sezione viene illustrato un esempio del comando trace voicemail all. Viene effettuata una chiamata all'utente 11044 che viene inoltrata alla segreteria telefonica. Usare almeno il comando trace voicemail vmxl all e il comando trace voicemail mwi all.Nota: l'utente preme 2 per contrassegnare il messaggio come urgente. L'evento MWI indicato in questo esempio si verifica *dopo* la segnalazione. Si verifica il segnale SIP/JTAPI, quindi il messaggio viene stampato per notificare l'avvenuta elaborazione.Nota: esiste un ID chiamata che consente di tenere traccia di una chiamata specifica se sono presenti più chiamate contemporaneamente. In questo caso, l'ID chiamata è 0x000000037e11d669. Se il sistema è integrato con Cisco CallManager Express, usare anche il comando trace ccn stacksip debug. Questo comando mostra in modo più chiaro quando vengono immesse le cifre, nonché quando si verificano disconnessione e altri eventi.

```
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_000000000000000000003
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_000000000000000000003
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_00000000000000000003';
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_00000000000000000003,1089912843407_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_00000000000000000003',
 '1089912843407_0',1,1089912843407);
18885 07/15 13:34:03.564 voicemail database "SQL: " 0x000000037e11d669 select
 totalmessagetime from vm_mailbox where mailboxid='PERSONAL_00000000000000000003'
 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_00000000000000000003';
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
```

Informazioni correlate

- Configurazione della visualizzazione telefono in Cisco Unity Connection 8.x
- Guida per l'amministratore di sistema di Cisco CallManager Express 3.1
- Guida all'installazione e all'aggiornamento di Cisco Unity Express 2.3
- Guida per l'amministratore dell'interfaccia utente grafica di Cisco Unity Express per Cisco CallManager, versione 2.1
- Supporto alla tecnologia vocale
- Supporto ai prodotti voce e Unified Communications
- Documentazione e supporto tecnico – Cisco Systems