



# Troubleshooting Cisco Unity Express System Features

---

This application note describes system-level troubleshooting techniques relating to the following topics:

- [General Troubleshooting Techniques, page 147](#)
- [Troubleshooting Installation Problems, page 154](#)
- [Troubleshooting Cisco Unity Express Startup, page 156](#)
- [Troubleshooting the Initialization Wizard, page 160](#)
- [Troubleshooting Backup and Restore, page 163](#)

## General Troubleshooting Techniques

This section summarizes several general types of tools that can help you isolate problems on a Cisco Unity Express system. These tools are described in the following section:

- [Viewing Generic System Information, page 147](#)
- [Logging in Cisco Unity Express, page 149](#)
- [Tracing Techniques, page 150](#)
- [Troubleshooting System Time Issues and NTP, page 153](#)

## Viewing Generic System Information

This application note presents information about Cisco Unity Express to help you troubleshoot and fix most common problems you might encounter. If you are unable to isolate and resolve issues, contact the Cisco Technical Assistance Center (TAC) for additional support and provide the Cisco TAC with the basic information about your system. This basic information includes the software version, the software license, the current configuration of the Cisco CME and Cisco Unity Express components, and some information about the current usage of the system. This section describes how to view that basic information.

The following example uses the **show version** command to provide information about the hardware form factor of the Cisco Unity Express module, the model type of the router that is hosting the Cisco Unity Express, the speed with which CPU is running, and additional hardware-specific information. One thing to note is that the AIM-CUE runs at 150 MHz if it is hosted on Cisco 2600XM or Cisco 2691 series access routers. In other routers, it runs at 300 MHz.

```
CUE# show version
CPU Model:                Celeron (Coppermine)
CPU Speed (MHz):          154.926
CPU Cache (KByte):        128
Chassis Type:             C2600
Chassis Serial:           FSJC0400673
Module Type:              AIM
Module Serial:            FHH07330001
```

The following example uses the **show software version** command to display the versions of the Cisco Unity Express software components installed on the system. Identifying the installed software versions is important when troubleshooting interworking issues. For example, some Cisco Unity Express features might be supported only with a certain version of Cisco CME or Cisco Unity Express.

```
Cue# show software version
Installed Packages:
- Bootloader (Primary)  1.0.17
- Global  2.0.0.12
- Voice Mail  2.0.0.12
- Bootloader (Secondary)  2.0.0
- Core  2.0.0.12
- Auto Attendant  2.0.0.12
```

```
Installed Languages:
- US English  2.0.0.0
```

The following example shows the license installed on the system by using the **show software licenses** command.

```
Cue# show software licenses
Core:
- application mode: CCME
- total usable system ports: 4
Voicemail/Auto Attendant:
- max system mailbox capacity time: 480
- max general delivery mailboxes: 15
- max personal mailboxes: 50
Languages:
- max installed languages: 1
- max enabled languages: 1
```

The following example uses **show voicemail usage** to show the current usage of the different resources in the system, including the number of personal mailboxes, general delivery mailboxes (GDMs), and the voice mail capacity allocation.

```
Cue# show voicemail usage
personal mailboxes:                50
general delivery mailboxes:        0
orphaned mailboxes:                0
capacity of voicemail (minutes):   480
allocated capacity (minutes):      350.0
message time used (seconds):       15227
message count:                     856
average message length (seconds):  17.78855140186916
greeting time used (seconds):      0
greeting count:                    0
average greeting length (seconds):  0.0
total time used (seconds):          15227
total time used (minutes):          253.78334045410156
percentage used time (%):           53
```

The show voicemail limits command, shown in the following example, shows the limits currently enforced on the voice mail system (based on the system’s license). These limits include the default mailbox size, default recording limit for callers and subscribers, and message expiration duration.

```
Cue# show voicemail limits
Default Mailbox Size (seconds):      420
Default Caller Message Size (seconds): 60
Maximum Recording Size (seconds):    900
Default Message Age (days):         30
System Capacity (minutes):           480
Default Prompt Language:              en_US
Operator Telephone:                   0
```

## Logging in Cisco Unity Express

Cisco Unity Express provides a full-fledged logging facility. Messages are logged to the console, to a Syslog server, or to the messages.log file on the disk or flash. Cisco Unity Express has four levels of logging. Table 10 shows the Cisco Unity Express logging levels mapped to Syslog logging levels.

**Table 10** Cisco Unity Express Logging Levels

Cisco Unity Express Logging Level	Syslog Logging Level
INFO	Debug, Info, and Notice
WARNING	Warning
ERROR	Error
FATAL	Critical, Alert, and Emergency

If you are monitoring straightforward issues, such as whether the Cisco Unity Express system reloaded, you might want to look at the Syslog messages before you start troubleshooting by turning on traces. These messages provide a brief problem description, a possible reason for the event, and other relevant information about a problem. These messages might be displayed on the console, written to a Syslog server, or stored in the Cisco Unity Express messages.log file, depending on the system’s configuration. The following example shows different ways of turning on logging in Cisco Unity Express. Logging to the console displays the log messages on the console when they occur.

```
Cue(config)# log console ?
  errors      Error messages
  filter      Filter events from syslog
  info        Information messages
  warning     Warning messages
Cue(config)# log console info
```

```
Logging to a server
Cue(config)# log server a.4.53.1
```

If you are using a Network Module Cisco Unity Express (NM-CUE), the log messages are also written to a file called messages.log on the hard disk. If you want offline access to the file, you can use command-line interface (CLI) commands to copy the messages.log file to an FTP server. The following example shows the log files on the system.

```
Cue# show logs
linux_session.log
dmesg
syslog.log
```

```
shutdown.log
atrace.log
debug_server.log
memmon.log
klog.log
messages.log
```

The following steps show you how to copy the log file to an external server:

- 
- Step 1** Have an FTP server ready that the Cisco Unity Express system can reach. Ensure that the FTP server is active.
  - Step 2** Set up a session to the Cisco Unity Express CLI from the router using the **service-module service-engine <slot/port> session** command. Enter the **show logs** command to make sure that the desired log files are available.
  - Step 3** Enter the following command for each file you need to transfer:

```
Cue# copy log log_file url ftp://ftp_ip_address/
```

where `log_file` is the filename of the desired log file and `ftp_ip_address` is the IP address of the FTP server. An example of a command to copy `messages.log` to an FTP server is

```
Cue# copy log messages.log url ftp://user:password@a.3.61.16/messages.log
```

---

## Tracing Techniques

Cisco Unity Express provides an extensive CLI-based tracing and debugging facility. It provides tracing commands for most of the system's important components, such as voice mail, the Customer Response Solutions (CRS) component, the web GUI interface, the Session Initiation Protocol (SIP) call processing component, and voice mail networking. Tracing uses the concepts of module, entity, and activity. For example, consider the following command:

```
Cue# trace ccn managerappl dbug
```

In this example, **trace** is the command, **ccn** is the module that represents the CRS component, and **managerappl** (which stands for application manager) is the entity within the `ccn` module. The **dbug** parameter is the activity to be traced.

By default, trace output is stored in a memory buffer. You can use commands to view the trace buffer contents on the console. You also have the option of configuring the trace output to write to a file called `atrace.log`. If you are using an NM-CUE, writing trace output to the disk is enabled by default. On the Advanced Integration Module (AIM-CUE) hardware, tracing is disabled by default. Enable it with care, because the AIM-CUE flash card has space limitations and a finite lifetime in terms of the number of write cycles before you must replace it. Doing indiscriminate tracing to your flash card shortens its life unnecessarily. Instead, if you are using an AIM-CUE, it is recommended that you use an external FTP server as the destination device for writing trace files.

## trace Command Summary

[Table 11](#) lists the most important trace commands. You can find the complete list of commands for tracing, modules, entities, and activities in the [Cisco Unity Express documentation](#) available on Cisco.com.

**Table 11** Cisco Unity Express trace Command

Command	Description
<b>clear trace</b>	Clears the trace buffer. Usually used when troubleshooting problems in real time and when you don't need the previous contents of the trace buffer.
<b>show trace</b>	Displays the current trace settings.
<b>show trace buffer</b>	Shows the current contents of the trace buffer. This output can be voluminous, depending on what traces were turned on, so use it carefully. This command displays the complete contents of the trace buffer from top to bottom.
<b>show trace buffer containing <i>regexp pattern</i></b>	Displays the lines in the trace buffer that contain a particular pattern.
<b>show trace buffer tail <i>N</i></b>	Shows the latest trace entries to the trace buffer in real time as and when they happen. If <i>N</i> is given, it shows the last <i>n</i> most recent trace entries and then continues displaying subsequent trace entries as and when they happen. If <i>N</i> is omitted, only new events are shown as they occur.
<b>show trace buffer tail <i>N</i> long</b>	Prints the traces in long format. The long format expands on the internal codes for different events and messages so that the trace output is more readable. For example, when you use <b>trace voicemail vxml all</b> , the output prints the text equivalent of the prompt file instead of just giving the prompt names.
<b>trace module entity activity</b>	Turns on the trace for a desired module, entity, and activity.

## Enabling Traces

Depending on the problem area you are facing, you turn on traces for specific modules and entities. You might often be required to turn on traces for several different modules and entities simultaneously to see how each module handles a particular data item or event. You might need to clear the trace buffer between different trace activities to make the troubleshooting output more readable and manageable, especially because the console connection speed is hard-coded to 9600 baud.

Each trace message in the output contains a time stamp of when the trace was generated. It also contains the module and entity that issued the trace output line so that when you read the trace, it is easy to understand which trace line was originated by which module. For example, in the following trace line, ACCN is the module—meaning the Cisco Communications Network (CCN)/CRS component—and VBRO is the entity (meaning the voice browser).

```
2699 10/10 12:23:30.115 ACCN VBRO 0 Task:18000000079 Invoke:
http://localhost/voicemail/vxmlscripts/login.vxml
```

For most problems, turning on trace for a couple of modules and entities is sufficient, and the trace output is a manageable volume. More complex problems might require tracing to be turned on for many modules and entities, and the trace output can become cumbersome.

If a trace is turned on for only a couple of modules, it is easy to view the trace in real time. Thus, you turn on the trace, clear the trace buffer, issue the **show trace buffer tail** command, and make the call to trigger the system events you want to trace. As the call enters the Cisco Unity Express application, you see trace output printed on the console. You can stop the trace output by pressing Ctrl-C, as shown in the following command. The trace is cut off the moment you press Ctrl-C.

```

Cue# trace ccn VbrowserCore debug
Cue# clear trace
Cue# show trace buffer tail
Press <CTRL-C> to exit...
DefaultCompilationConfig is now set to JVM en_US
2043 09/25 15:56:22.305 ACCN VBRW 0 Task:20000000089 enterLevel: SESSION_LEVEL
2043 09/25 15:56:22.306 ACCN VBRW 0 callContact.getSessionID() = 6000000059
2043 09/25 15:56:22.306 ACCN VBRW 0 Task:20000000089 VoiceBrowser sessionid:
    6000000059, taskid: 20000000089
2043 09/25 15:56:22.306 ACCN VBRW 0 callContact.getCallingNumber() = 5702
2043 09/25 15:56:22.306 ACCN VBRW 0 callContact.getCallingNumber() = 5702
2043 09/25 15:56:22.306 ACCN VBRW 0 callContact.getDNIS() = null
2043 09/25 15:56:22.306 ACCN VBRW 0 callContact.getCalledNumber() = 5800
2043 09/25 15:56:22.307 ACCN VBRW 0 callContact.getDNIS() = null
2043 09/25 15:56:22.307 ACCN VBRW 0 callContact.getCalledNumber() = 5800
2043 09/25 15:56:22.307 ACCN VBRW 0 callContact.getANIIDigits() = null
2043 09/25 15:56:22.307 ACCN VBRW 0 callContact.getUserToUserInfo() = null
2043 09/25 15:56:22.307 ACCN VBRW 0 callContact.getRDNIS() = null
2043 09/25 15:56:22.307 ACCN VBRW 0 callContact.getLastRedirectedNumber() = null
2043 09/25 15:56:22.307 ACCN VBRW 0 callContact.getSessionID() = 6000000059
2043 09/25 15:56:22
Cue#

```

If you run into a situation where you must turn on tracing for many modules and entities, or even **trace all**, it is better not to attempt to watch the trace output in real time, because it becomes very difficult to follow. All the trace output is written to a file called *atrace.log*. On the NM-CUE, this file is written by default. On an AIM-CUE, use the **log trace local enable** command to allow the trace buffer to be written to flash. You can also use **log trace server** so that trace output is written to an FTP server instead of a flash on your Cisco Unity Express, as shown in the following example.

```

Cue(config)# log trace server url ftp://userid:password@b.168.0.1/trace
Cue(config)# log trace server enable

```

After you make the preceding configuration, the trace file is created on the specified server.

## Copying a Trace File to an FTP Server

After you make the test call to trigger the events you want to capture to a trace file, you must copy the *atrace.log* file to an FTP server, as shown in the following example.

```

Cue# show logs
linux_session.log
dmesg
syslog.log
shutdown.log
atrace.log
debug_server.log
memmon.log
klog.log
messages.log
Cue# copy log atrace.log url ftp://user:password@a.3.61.16/atrace.log

```

The *atrace.log* file is a binary file that you must decode to view in a text editor. You can get the utility to decode the trace file from the Cisco TAC. The utility consists of three files:

- **atrace\_decode** (an executable)
- **trace.tcnd**
- **trace.tmsg**

Decode the *atrace.log* file as shown in the following example. This creates an output file (*dtmf\_issue.log*, in this example) that you can now view in a text editor. Currently, *atrace\_decode* runs only on a Linux platform.

```
[test@CUE-core ftp]$ ./atrace_decode
Usage: ./atrace_decode <file_to_decode> <format> <out_file> <trace_command_file>
<trace_message_file>
format = 0 for short, 1 for long
[test@CUE-core ftp]$ ./atrace_decode atrace.log 1 dtmf_issue.log ./trace.tcmd
./trace.tmsg
```

## Troubleshooting System Time Issues and NTP

Many of the Cisco Unity Express features depend on a reliable clock. The voice message envelope information is one important example. The Cisco Unity Express clock is managed entirely by Network Time Protocol (NTP) and cannot be set manually.

Another Cisco Unity Express feature that requires clock information is the trace and log messages that carry time stamps. If the clock on the Cisco Unity Express system is not set correctly, troubleshooting might become very difficult, because log and trace messages cannot be correlated with real time.

Configure an NTP source for Cisco Unity Express that is the clock's source. An NTP source can be the host Cisco CME router (although this is not recommended because the low-end routers do not have a very reliable onboard clock source) or any other clock source in your network. Sometimes, for many reasons, the clock on Cisco Unity Express might not be synchronized to its configured source. You can verify the status of NTP and the location of the clock's source with CLI commands. The following example shows these commands.

```
NTP not working due to wrong configuration:
Cue# show ntp status
NTP reference server :      a.3.6.190
Status:                    reject
Time difference (secs):    0.0
Time jitter (secs):       4.0

NTP in Sync:
Cue#show ntp status
NTP reference server :      a.4.13.190
Status:                    sys.peer
Time difference (secs):    0.02110639284364879
Time jitter (secs):       0.019472182961180806
```

The following commands are available to debug NTP synchronization issues.

```
Cue# trace ntp ?
  all      Every entity and activity
  ntp      Entity
Cue# trace ntp ntp ?
  all      Every activity
  clkadj   Activity
  clkselect Activity
  clkvalidity Activity
  clockstats Activity
  event    Activity
  loopfilter Activity
  loopstats Activity
  packets  Activity
  peerstats Activity
```

# Troubleshooting Installation Problems

When you install a Cisco Unity Express module, problems might occur in downloading the software package. These problems might be caused by network connectivity or package issues. The following sections describe some common problems that might occur during software installation of Cisco Unity Express and ways to troubleshoot them.

- [Network Connectivity Issues, page 154](#)
- [Software Package Issues, page 155](#)

## Network Connectivity Issues

If the Cisco Unity Express module is unable to establish contact with the FTP server where the software load resides, the error shown in the following example occurs when you attempt to install the software.

```
Cueinstaller#> software install package url
ftp://username:password@a.3.61.61/cue-vm.1.1.0.6.pkg
RAMDisk mounted
Connecting to host...
curl: (7) Connect failed
ERROR: Host did not respond.
Please check the host ip and try again.
RAMDisk unmounted
```

First ensure that the IP address of the FTP server is correct. Verify all the parameters given in the **install** command. If you are sure these are all correct, verify the IP connectivity from the Cisco Unity Express module to the router. Reboot the Cisco Unity Express module, as shown in the following example, and press **\*\*\*** at the first prompt. This action takes you to the bootloader prompt.

```
Cueinstaller#> reboot
WARNING: This will reboot the Service Engine!
Do you wish to continue (y,n) [n] y
```

The bootloader has a **ping** command, as shown in the following example.

```
ServicesEngine boot-loader> ping a.3.61.61
Sending 5, 32 byte ICMP Echos to a.3.61.61:
.....
Success rate is 0% (0/5)

ServicesEngine boot-loader> ping a.3.61.16
Sending 5, 32 byte ICMP Echos to a.3.61.16:
!!!!
Success rate is 100% (5/5)
```

If the Cisco Unity Express system cannot ping the FTP server, you might have the wrong configuration of IP parameters in the bootloader. The following example shows how to check the bootloader configuration. If you see anything wrong, you can use the bootloader **config** command to make modifications.

```
ServicesEngine boot-loader> show config
IP addr:                a.4.13.90
Netmask:                255.255.0.0
TFTP server:           a.4.53.1
GW IP addr:            a.4.13.190
Default boot:         disk
Bootloader Version:   1.0.17
Default Helper-file:  cue-installer.1.1.1
Default BIOS:         primary
```

```
Default bootloader:    primary
Default cpu throttle: 50%
```

Another reason why the **ping** command might not be successful is the routing configuration on the Cisco IOS router. With an **ip unnumbered** configuration for the service-engine interface, you can verify the routing as follows:

- Ping the FTP host from the Cisco IOS router to ensure that the host can be reached. If this fails, examine the Cisco IOS routing configuration.
- If the FTP host can be reached from the router, verify the Cisco Unity Express module’s connectivity, as shown in the following example.

```
Router# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
Gateway of last resort is a.3.0.1 to network 0.0.0.0
 a.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C     a.3.0.0/16 is directly connected, FastEthernet0/0
S     a.3.6.129/32 is directly connected, Service-Engine1/0
 a.0.0.0/24 is subnetted, 1 subnets
S     a.20.20.0 [1/0] via a.3.6.26
 a.0.0.0/24 is subnetted, 1 subnets
C     a.10.10.0 is directly connected, FastEthernet0/1
 a.0.0.0/24 is subnetted, 1 subnets
S     a.41.41.0 [1/0] via a.3.6.28
S*   0.0.0.0/0 [1/0] via a.3.0.1
```

When the **show ip route** command is executed, a host route similar to the one highlighted in the following example will be displayed (where a.3.6.129 is the IP address of your Cisco Unity Express module and Service-Engine1/0 is the Cisco Unity Express module seated in NM slot 1 of the router). If such a route does not show in your routing table, use the following command to add it:

```
router(config)#ip route a.3.6.129 255.255.255.255 Service-Engine1/0
```

## Software Package Issues

Sometimes a problem might occur in downloading the software, such as if binary mode was not used in the FTP command. This causes a problem when installing the software, such as the error shown in the following example.

```
cueinstaller#> software install package
ftp://username:password@a.3.61.16/pkg/cue-vm.1.1.0.6.pkg
RAMDisk mounted
Connecting to host...
  % Total      % Received % Xferd  Average Speed          Time          Curr.
                Dload  Upload  Total      Current  Left      Speed
100 92785  100 92785    0     0  221k      0  0:00:00  0:00:00  0:00:00 2623k
RAMDisk unmounted
ERROR:: Security Header Validation Failed.
```

The error shown in the following example indicates that not all the files needed for installation are present on the FTP server.

```
cueinstaller#> software install package ftp ://username
:password@a.3.61.16/cue-vm.1.1.1.pkg
```

```

Created RAMDisk.
File System Type = reiser
RAMDisk mounted
Connecting to host...
  % Total    % Received % Xferd  Average Speed          Time      Curr.
100 92811  100 92811    0      361k      0  0:00:00  0:00:00  0:00:00 8822k
Core File List is Signed
Retrieving cue-vm.1.1.1.manifest from a.3.61.16
Retrieve manifest from server failed.
RAMDisk unmounted
Connecting to host...
curl: (19) cue-vm.1.1.1.manifest: No such file or directory
ERROR: The requested file does not exist on host.
Please check the package name and try again.

```

To recover from these software package installation errors, again download the software to the FTP server. Ensure that three types of files are present on the FTP server:

- .pkg
- .prt1
- .manifest (this file no longer exists separately in Cisco Unity Express releases after 2.0)

Along with the available software image packages, a bootloader package is available. All Cisco Unity Express software versions have a minimum bootloader version requirement. The manifest does not check that the minimum bootloader version is present in the system. Refer to the Cisco Unity Express documentation to identify the bootloader version that is compatible with the Cisco Unity Express software version you're trying to install. The exact files that comprise the installation might also vary from release to release. The examples in this section pertain to Cisco Unity Express 1.1.1.

## Troubleshooting Cisco Unity Express Startup

You might experience situations in which the basic IP communication between the host router and the Cisco Unity Express module fails even after proper configuration of the host router. You can identify such a problem by looking for waiting events such as those shown in an extract of the installation output in the following example.

```

==> only eth0 exists, we must be running on an AIM
==> only eth0 exists, we must be running on an AIM

Router communications servers initializing...complete.
Waiting for IOS to register IP address.
- waited 10 seconds...
Waiting for IOS to register IP address.
- waited 20 seconds...

```

Cisco Unity Express is waiting for commands from the Cisco IOS router to configure Cisco Unity Express with its IP address and default gateway parameters so that Cisco Unity Express can communicate with the rest of the network. However, it is not receiving any response from the router. This process of configuring the Cisco Unity Express module through the host Cisco IOS router uses the RBCP. There might be some situations in which you must troubleshoot this protocol exchange between the host router and Cisco Unity Express.

When Cisco Unity Express successfully communicates with the router using RBCP and receives its IP parameters, you see the message shown in the following example on the Cisco Unity Express console during application bootup.

```
==> only eth0 exists, we must be running on an AIM
==> only eth0 exists, we must be running on an AIM

Router communications servers initializing...complete.
IOS IP Address Registration complete.
```

The following sections discuss troubleshooting problems during application bootup and communication issues between Cisco Unity Express and the host router:

- [Checking Console Output on the Cisco Unity Express Hardware, page 157](#)
- [Checking the RBCP Communication, page 157](#)
- [Debugging the RBCP Communication, page 157](#)

## Checking Console Output on the Cisco Unity Express Hardware

If you are having a problem opening a session to the Cisco Unity Express module, or you don't see any output on the console, you can use the following command to check the console messages on the Cisco Unity Express module without needing to open a session to the Cisco Unity Express module:

```
Router# test service-module service-engine slot/unit console
```

By default, this command displays the most recent 80 lines stored in the console buffer. However, it is possible to specify an offset of greater or less than 80, or to view all the messages stored in the console buffer.

## Checking the RBCP Communication

To check the RBCP status on the Cisco Unity Express module from the router, you can use the following test command:

```
Router# test scp ping slot
```

This command sends a ping to the Cisco Unity Express module as an RBCP message using operational code (opcode) 0x11. If the RBCP process on the Cisco Unity Express module is up and running, the ping succeeds, and the output of the **test** command will look as shown in the following example.

```
Router# test scp ping 3
pinging addr 3 (0x3)
assigned sap 0x4
addr 3 (0x3) is alive
```

## Debugging the RBCP Communication

There are situations in which you must troubleshoot the RBCP messages between the Cisco Unity Express module and the router. One situation is when the interface configuration was verified, but you still cannot **ping** the Cisco Unity Express module.

First, check the interface’s status and ensure that the interface and line protocol are up, as shown in the following example.

```
Router# show interfaces service-engine 1/0
Service-Engine1/0 is up, line protocol is up
  Hardware is I82559FE, address is 0001.b912.xxxx (bia 0001.b912.xxxx)
  Interface is unnumbered. Using address of FastEthernet0/0 (a.3.6.29)
```

Next, verify the RBCP state machine status on the router, as shown in the following example. The Cisco Unity Express module will be in steady state for proper operation. The example shows the status of service-engine 1/0.

```
Ccme# service-module service-Engine 1/0 status
Service Module is Cisco Service-Engine1/0
Service Module supports session via TTY line 33
Service Module is in Steady state
cisco service engine 1.0
```

If you are still unable to ping the Cisco Unity Express module IP address, troubleshoot the RBCP messages exchanged between the Cisco Unity Express module and the host router. Table 12 describes the flags and how they are used for the scp-tx message. scp-tx stands for Switch Communication Protocol transmission, meaning that the router transmits this message to the Cisco Unity Express module.

**Table 12**      *Flags for the scp-tx RBCP Message*

Opcode	Action	Action Description	Type	Interface Description
0054	01	Configure	01	Internal IP address
0054	00	Unconfigure	01	Internal IP address
0059	01	Configure	—	—
0059	00	Unconfigure	—	—

Table 13 describes the flags and how they are used for the scp-rx message, which is a *Switch Communication Protocol* message the router receives from the Cisco Unity Express module.

**Table 13**      *Flags for the scp-rx RBCP Message*

Opcode	Type	Type Description
0054	01	Internal IP address rejected
0054	02	Internal IP address OK
0054	03	External IP address rejected (not used for NM-CUE)
0054	04	External IP address OK (not used for NM-CUE)
0059	00	Default IP gateway OK
0059	01	Default IP gateway rejected

The output of **debug scp all** is shown in the following example. An IP address (b.168.0.0 255.255.0.0) is being configured on the Ethernet interface of the Cisco Unity Express module.

```
Router# debug scp all
router(config-if)#service-module ip address b.168.0.2 255.255.0.0
router(config-if)#
*Mar  2 18:07:24.673:  scp-tx: SA:0F/01 DA:01/01 Op:0054 Sq:13C7 Ln:000A I:00
```

```
*Mar 2 18:07:24.673: 000: 01 01 C0 A8 00 02 FF FF 00 00 .....L....
*Mar 2 18:07:24.681: scp-rx: SA:0E/01 DA:0F/01 Op:0054 Sq:13C7 Ln:000A I:01
*Mar 2 18:07:24.681: 000: 02 00 00 00 00 00 00 00 00 00
```

The output shows that the scp-tx message transmitted has the Source Address (SA) field set to 0F/01, which indicates that the message originated from the router. The Destination Address (DA) field is set to 01/01, which indicates that the Cisco Unity Express module is present in slot 1. The opcode of 0054 indicates that this is an IP address configuration. The sequence number (Sq) field is 0B26, and the length of the payload is 10 bytes.

The first parameter on the second line is the type, and the second parameter is the action. In the message, the type is 01 and the action is 01, indicating that the Cisco Unity Express module interface is being configured. The next 8 bytes are the IP address and subnet mask.

In the output shown for the scp-rx message, the SA field is set to 0E/01, indicating that it originated from the Cisco Unity Express module in slot 1. The DA field is set to 0F/01, indicating that the message is destined for the router. The Opcode and Sq fields are the same as in the scp-tx message. The Type field in the second line is set to 02, which means that the Cisco Unity Express module IP address was set properly. The rest of the parameters have no significance.

The following example shows the Cisco Unity Express module's **default-gateway** parameter being set.

```
Router# debug scp all
router(config)#int content-engine 1/0
router(config-if)#service-module ip default-gateway b.168.0.1
1d23h: scp-tx: SA:0F/01 DA:01/01 Op:0059 Sq:0B28 Ln:0005 I:00
1d23h: 000: 01 C0 A8 00 01 .....
1d23h: scp-rx: SA:01/01 DA:0F/01 Op:0059 Sq:0B28 Ln:0005 I:01
1d23h: 000: 00 FF FF FF E0
```

The debug output of the scp-tx message shows that the opcode is different. The value 0059 indicates that this message pertains to the IP default-gateway configuration parameter. The length of the payload is 5 bytes. The payload is shorter than the scp-tx message shown in preceding example **debug scp all** output (5 bytes versus 10 bytes), because no subnet mask is associated with the default gateway IP address. The action flag is set to 01, which indicates that the default gateway is being configured. In the output of scp-rx message, the action flag is set to 00, confirming that the configuration of the IP default gateway address was successful.

[Table 14](#) summarizes some of the common problems encountered when installing and configuring a Cisco Unity Express module, and possible solutions to those problems.

**Table 14 Common Installation Problems and Resolutions**

Problem	Possible Reason	Possible Solution
You can't open a session into the NM-CUE.	The TTY line associated with the NM-CUE is already occupied.	Use the <b>service-module service-engine slot/port session clear</b> command, or clear line xx to clear the TTY line.
The session, when invoked, results in a connection refused error message.	The TTY line associated with the NM-CUE is occupied.	Configure <b>no exec</b> under the TTY line associated with the NM-CUE. This prevents the line from being unavailable because of a rogue EXEC process.
Service module commands do not seem to take effect.	The service module status might not be in steady state. RBCP configuration messages go through only when the service module is in steady state.	It is possible that the service module is not responding. Try reloading the service module. If that does not work, use the <b>reset</b> command.
You can't ping the internal address when using the IP unnumbered scheme.	The IP route table is incorrect.	When using <b>ip unnumbered</b> , always remember to add a static route pointing toward the service-engine interface.
IOS doesn't let you change or remove the IP address of the CE NM interface.	The default gateway of the CE NM must be pointing to the same IP subnet as the interface being changed or removed.	First remove the IP default gateway from under the service-engine interface. Then change the interface's IP address and add back the IP default gateway.
You can set the speed of the terminal line from the router side, but you can't see any CLI for doing the same on the CE side.	There is no CLI to change the speed. The speed is hard-set to 9600, 8-N-1 on both the Cisco IOS and CE sides. Even though Cisco IOS allows you to change the speed settings, this doesn't take effect.	—

## Troubleshooting the Initialization Wizard

The Cisco Unity Express Initialization Wizard is a GUI tool provided to complete the installation of the system. After new software is loaded on the system, you must run the Initialization Wizard. You invoke a browser on a PC with a URL pointing to Cisco Unity Express. After logging in, you go through a number of steps, including the following:

- Importing users from Cisco CME
- Defining the various pilot numbers

The following sections describe how to identify both issues:

- [Unable to Log in to the Initialization Wizard, page 161](#)
- [Initialization Wizard Reports Failures, page 161](#)

## Unable to Log in to the Initialization Wizard

At the login window, you must enter the administrator username and password assigned during the post-installation procedures in order to access the initialization wizard. If you are unable to log in, the username and password might not have been entered properly (both are case sensitive and permit special characters). To verify that the administrator username is legitimate, access the CLI and check the Cisco Unity Express configuration. The username entered should be the one administratively defined on your system. You can verify that the username `admin` belongs to the Administrators group, as shown in the following example.

```
Cue# show group detail groupname Administrators
Full Name:      Administrators
Description:
Phone:
Phone(E.164):
Language:      systemDefault(ga_IE)
Owners:
Members:       admin
Privileges:    superuser ManagePrompts ManagePublicList ViewPrivateList
```

If you have verified that the username is an administrator, but you still cannot log in, you can bypass this process by assigning a new password for the login—as shown in the following example.

```
Cue# username temp password temp
Cue# config terminal
Cue(config)# group Administrators member temp
```

The variable `temp` is any name you choose for this purpose.



### Note

Note that the group is Administrators; take care with both the case of the letters and the spelling. If you enter it wrong, it creates the wrong group and you still cannot access the system.

## Initialization Wizard Reports Failures

During the process of running the Initialization Wizard, the most common problem experienced is that the Cisco Unity Express GUI cannot log in to Cisco CME to read the user and extension information. This might be because the configuration on Cisco CME is missing some mandatory items. Cisco Unity Express asks you to enter the username and password for Cisco CME, so ensure that you are using the correct Cisco CME username and password in these fields. The following example shows debugs on Cisco CME where a Cisco Unity Express login fails because of wrong authentication information.

```
cme# debug ip http authentication
Dec 14 01:35:49.915: its_urlhook url: /telephony_service.html, method 1
Dec 14 01:35:49.915: Tue, 14 Dec 2004 01:35:49 GMT a.4.13.153
/telephony_service.html auth_required Protocol = HTTP/1.1 Method = GET
Dec 14 01:35:49.915:
Dec 14 01:35:49.923: its_urlhook url: /ITSMmain, method 1
Dec 14 01:35:49.923: validate_username_password (asfd, sdf)validate_
admin_user (asfd: sdf)
validate_admin_user: validate admin [0] locally
validate_admin_user (asfd: sdf)
validate_username_password: check password for phone[-1]

Dec 14 01:35:49.923: HTTP: Authentication failed for realm its_access
Dec 14 01:35:49.923: HTTP: Authentication failed for level 15
Dec 14 01:35:51.923: Tue, 14 Dec 2004 01:35:51 GMT 1.4.13.153
/ITSMmain auth_failed
```

```

        Protocol = HTTP/1.1 Method = GET
Dec 14 01:35:51.923:
Dec 14 01:35:51.927: its_urlhook url: /ipkeyswitch, method 1
Dec 14 01:35:51.927: lds_urlhook, url=/ipkeyswitch

```

You can see the current Cisco CME administration account details using the **show telephony-service** command on Cisco CME, as shown in the following example.

```

CME# show telephony-service
CONFIG (Version=3.2)
=====
Version 3.2
Cisco CallManager Express
For on-line documentation please see:
www.cisco.com/univercd/cc/td/doc/product/access/ip_ph/ip_ks/index.htm
ip source-address a.4.13.53 port 2000
max-ephones 48
max-dn 192
.....
web admin system name admin password admin
web admin customer name Customer
edit DN through Web: disabled.
edit TIME through web: disabled.

```

The rest of the Initialization Wizard usually proceeds smoothly. If you encounter issues, turn on the traces shown in the following example. The following example shows Cisco Unity Express trace output when an administrator uses the wrong username and password for Cisco CME.

```

Cue# trace webinterface all
Cue# show trace buffer tail
Press <CTRL-C> to exit...
3662 12/15 15:06:10.904 webI init 5
3715 12/15 15:07:09.898 webI ctrl 0 Enter /Web/IW/InitWizard.do
3715 12/15 15:07:09.898 webI sydb 0 /sw/apps/monitor//ctrl/offline
3715 12/15 15:07:09.899 webI sydb 2 0
.....
3715 12/15 15:07:09.903 webI ctrl 2 null
3715 12/15 15:07:09.903 webI ctrl 0 Center /Web/IW/InitWizard.do
3715 12/15 15:07:09.905 webI sydb 0 /sw/limits/global/applicationMode
3715 12/15 15:07:09.905 webI sydb 2 ITS
3715 12/15 15:07:09.905 webI sydb 0 /sw/limits/global/applicationMode
3715 12/15 15:07:09.906 webI sydb 2 ITS
3715 12/15 15:07:09.906 webI prxy 2 Call to connection.getInputStream()
3715 12/15 15:07:09.911 webI prxy 2 Response code 401. Receiving data from Router
3715 12/15 15:07:09.912 webI prxy 2 Receiving data from Router...done
3715 12/15 15:07:09.912 webI prxy 2 Detected Lambretta 1+ Image
3715 12/15 15:07:09.912 webI prxy 2 ITSL2Connection initiated
3715 12/15 15:07:09.912 webI prxy 2 Enter ITSL2 Login Center
3715 12/15 15:07:09.913 webI prxy 2 Call to connection.getInputStream()
3715 12/15 15:07:11.917 webI prxy 2 UNREACHABLE: Unable to contact Router
3715 12/15 15:07:11.917 webI prxy 2 Enter ITSL2 Login Center
3715 12/15 15:07:11.918 webI prxy 2 Call to connection.getInputStream()
3715 12/15 15:07:11.925 webI prxy 2 UNREACHABLE: Unable to contact Router
3715 12/15 15:07:11.925 webI init 5
3715 12/15 15:07:11.929 webI sydb 0 /sw/limits/global/applicationMode
3715 12/15 15:07:11.930 webI sydb 2 ITS
3715 12/15 15:07:11.932 webI sydb 0 /sw/limits/global/applicationMode
3715 12/15 15:07:11.933 webI sydb 2 ITS
3715 12/15 15:07:11.945 webI ctrl 0 Exit /Web/WEB-INF/screens/iw/InitWizard.jsp

```

# Troubleshooting Backup and Restore

The following sections describe problem scenarios you might encounter when performing a backup or restore operation of the Cisco Unity Express configuration or voice message data:

- [Cannot Find or Connect to the Backup/Restore Server, page 163](#)
- [Backup or Restore Cannot Start, page 163](#)
- [Backup or Restore is Incomplete, page 164](#)
- [Restore Does Not Restore the Correct or Complete Configuration or Data, page 164.](#)

The next sections describe techniques for troubleshooting each of these issues.

## Cannot Find or Connect to the Backup/Restore Server

To start a backup or restore operation, you must first configure the FTP server URL and corresponding directory. Backup and restore are done to the same server location. The CLI to configure the backup server is as follows:

```
Cue(config)# backup server url ftp://server_ip_address/directory [username name password password]
```

Use the following processes to troubleshoot server connectivity problems:

1. Check the Cisco Unity Express configuration to ensure that all the backup and restore configuration parameters are correct. The critical parameters are the server's IP address and the appropriate directory path.
2. Use the commands **show backup server** and **show backup history** to ensure that the configuration is set up properly.
3. Ping the FTP server to ensure that it can be reached.
4. The backup path specified in the Cisco Unity Express configuration shows the default FTP path. Check the path set on the server to ensure that the proper directory was selected.

## Backup or Restore Cannot Start

The FTP server for backup and restore might require a user login and password (this is an optional but recommended configuration). Also, some FTP servers limit the transferable block of data to a certain size. If a login is configured on your FTP server, verify that you have permission to read and write on the FTP server.

If the backup server was configured but cannot be reached when the backup process starts, the system shows an error message, as shown in the following example.

```
Cue# show backup server  
Backup Restore: Unable to connect to backup server
```

The following Cisco Unity Express **trace** command is useful in capturing the activities while a backup or restore is progressing:

```
Cue# trace BackupRestore all
```

The following example shows traces indicating that the backup server has not been configured.

```
996 02/28 19:59:59.448 bare bare 0 BackupRestore: Backupsysdbnode get  
996 02/28 19:59:59.448 bare bare 0 BackupRestore: done Backupsysdbnode get, val:
```

```

997 02/28 19:59:59.449 bare bare 0 BackupRestore: Backupsysdbnode get
997 02/28 19:59:59.449 bare bare 0 BackupRestore: done Backupsysdbnode get, val:
996 02/28 19:59:59.452 bare bare 0 in isServerConfigured
996 02/28 19:59:59.454 bare bare 0 in isServerConfigured, url:ftp://a.0.0.1/ftp
user:backupadmin password:backupadminpassword
996 02/28 19:59:59.454 bare bare 129
996 02/28 19:59:59.454 bare bare 130
996 02/28 19:59:59.454 bare bare 0 out isServerConfigured, backup server not configured
996 02/28 19:59:59.454 bare bare 0 history sysdbprovider: backup server not configured
997 02/28 19:59:59.457 bare bare 0 BackupRestore: Backupsysdbnode get
997 02/28 19:59:59.457 bare bare -1 BackupRestore: done Backupsysdbnode get, val:
998 02/28 19:59:59.457 bare bare 0 BackupRestore: Backupsysdbnode get
998 02/28 19:59:59.457 bare bare 0 BackupRestore: done Backupsysdbnode get, val:
1082 02/28 19:59:59.458 bare bare 0 BackupRestore: Backupsysdbnode get
1082 02/28 19:59:59.458 bare bare 0 BackupRestore: done Backupsysdbnode get, val
    
```

On the other hand, if the FTP server is not present or reachable, the trace includes the following outputs.

```

1186 03/01 01:40:56.823 bare bare 130
1186 03/01 01:40:56.825 bare bare 0 in isServerReachable, Unable to reach backup server
1186 03/01 01:40:56.825 bare bare 0 backup sysdbprovider: backup server unreachable
    
```

## Backup or Restore is Incomplete

When the backup or restore is started, but is then aborted or incomplete, first check the integrity of the network between the Cisco Unity Express system and the server. If available, use a sniffer to see if there are excessive packet drops. The other possibility is that the FTP server might not be configured properly to transfer large blocks of data. Some FTP servers might have the block size default set to smaller than 16 MB. Configure the FTP server to transfer and accept larger blocks of data. If the FTP server cannot be configured, use a different one. In addition, verify that the FTP server supports passive FTP.

## Restore Does Not Restore the Correct or Complete Configuration or Data

When the restore operation cannot restore the proper configuration or data, users notice wrong parameters and missing stored messages or greetings. Use the **trace** command shown in the following example to troubleshoot this situation.

```

Cue# trace backrestore backrestore ?
CONF          Activity
HISTORY       Activity
INIT          Activity
OPERATION     Activity
SERVER        Activity
all           Every activity
    
```

Each of these traces records the detailed activities of the operation. Turn on the appropriate trace and repeat the failed activity. The activities are recorded accordingly and can then be examined for the cause of the failure.