



Backing Up and Restoring Data

Last Updated: July 12, 2007

Cisco Unity Express backup and restore functions use an FTP server to store and retrieve data. The backup function copies the files from the Cisco Unity Express application to the FTP server and the restore function copies the files from the FTP server to the Cisco Unity Express application. The FTP server can reside anywhere in the network if the backup and restore functions can access it with an IP address or hostname.

We recommend that backups be done regularly to preserve voice-mail messages and configuration data.

Backup and restore commands are available in configuration mode and in offline mode.

- In configuration mode, commands are available to set the following parameters:
 - Number of backup files to keep (the oldest file is deleted).
 - URL of the FTP server where the files will be stored.
- In offline mode, perform the backup or restore procedure. Decide the following:
 - Type of files to be backed up: all files (configuration and data), only configuration files, or only data files. Data files consist of voice-mail messages. Configuration files consist of all other system and application parameters.
 - URL of the FTP server where the files will be stored.



Caution

Offline mode terminates all existing voice-mail calls and IMAP and VoiceView Express sessions. No new voice-mail calls are allowed. Calls to auto attendant are allowed. We recommend doing a backup when no calls are active.

This chapter contains the following sections:

- [Restrictions, page 236](#)
- [Setting Backup Parameters, page 236](#)
- [Backing Up Files, page 238](#)
- [Restoring Files, page 241](#)
- [Copying Configurations, page 243](#)
- [Restoring Factory Default Values, page 246](#)
- [Backup and Restore Using SFTP, page 247](#)
- [Backup Server Authentication Using a SSH Host Key, page 248](#)

- [Encrypting and Signing of Backup Content on the Server, page 251](#)
- [Encrypting PINs in Backup Files, page 253](#)

Restrictions

Cisco Unity Express does not support the following backup and restore capabilities:

- Scheduled backup and restore operations. The backup and restore procedures begin when the appropriate command is entered.
- Centralized message storage arrangement. Cisco Unity Express backup files cannot be used or integrated with other message stores.
- Selective backup and restore. Only full backup and restore functions are available. Individual voice-mail messages or other specific data cannot be stored or retrieved.

Setting Backup Parameters

The backup parameters define the FTP server to use for storing Cisco Unity Express backup files and the number of backups that are stored before the system deletes the oldest one.

All Cisco Unity Express backup files are stored on the specified server. You can copy the backup files to other locations or servers, if necessary.

Cisco Unity Express automatically assigns an ID to each successful backup. Use this backup ID to restore the backup.

Prerequisites

- Verify that the backup server is configured.
- Verify that an FTP administrator or other user who can log in to the FTP server has full permission on the FTP server, such as read, write, overwrite, create, and delete permissions for files and directories.

Required Data for This Procedure

- Number of revisions to save before the oldest backup is written over
- FTP server URL
- User ID and password of the FTP server login

SUMMARY STEPS

1. **config t**
2. **backup {revisions *number* | server url *ftp-url* username *ftp-username* password *ftp-password*}**
3. **exit**
4. **show backup**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>config t</pre> <p>Example: se-10-0-0-0# config t</p>	Enters configuration mode.
Step 2	<pre>backup {revisions number server url ftp-url username ftp-username password ftp-password}</pre> <p>Example: se-10-0-0-0(config)# backup server url ftp://main/backups username "admin" password "wxyz" se-10-0-0-0(config)# backup server url ftp://172.168.10.10/backups username "admin" password "wxyz" se-10-0-0-0(config)# backup revisions 5</p>	Sets the backup parameters. <ul style="list-style-type: none"> • server url—The <i>ftp-url</i> value is the URL to the network FTP server where the backup files will be stored. The <i>ftp-username</i> and <i>ftp-password</i> values are the user ID and password for the network FTP server. <p>Note The backup server must be configured before the backup revisions can be configured.</p> <ul style="list-style-type: none"> • revisions—The number of backup files that will be stored. When this number is reached, the system deletes the oldest stored file. <p>In the example, main is the hostname of the FTP server and backups is the directory where backup files are stored.</p>
Step 3	<pre>exit</pre> <p>Example: se-10-0-0-0(config)# exit</p>	Exits configuration mode.
Step 4	<pre>show backup</pre> <p>Example: se-10-0-0-0# show backup</p>	Displays the backup server configuration information, including the FTP server URL and the number of revisions.

Examples

The following example configures a backup server and displays the **show backup** output:

```
se-10-0-0-0# config t
se-10-0-0-0#(config)# backup server url ftp://172.16.0.0/backups username admin password voice
se-10-0-0-0#(config)# backup revisions 10
se-10-0-0-0#(config)# exit
se-10-0-0-0#

se-10-0-0-0# show backup
Server URL: ftp://172.16.0.0/backups
User Account on Server: admin
Number of Backups to Retain: 10
se-10-0-0-0#
```

Backing Up Files

Three types of backup requests are available: data only, configuration only, or all.

- **Data**—Backs up voice-mail greetings and voice-mail messages.
- **Configuration**—Backs up system configuration, including recorded names, custom scripts, and custom prompts. Use the **show run** command to display the current running configuration.
- **All**—Backs up all data and configuration information.

Backups are performed only in offline mode.

Cisco Unity Express automatically numbers and dates the backup files and identifies the revision number in a backupid field.

Performing different backup types at various times causes different backup IDs for data backups and configuration backups. For example, the last data backup ID might be 3, and the last configuration backup might be 4. Performing an “all” backup might result in a backup ID of 5 for both data and configuration.

When restoring the files, refer to the backup ID for the backup file that you want to use. Use the **show backup server** command for a list of backup IDs.



Note

We recommend that you back up your configuration files whenever changes are made to the system or application files. Data files, which contain voice messages, should be backed up regularly to minimize data loss, such as from a hardware failure.



Caution

Offline mode terminates all existing voice-mail calls, and no new voice-mail calls are allowed. Calls to auto attendant are allowed. We recommend doing a backup when telephone subscribers are not active on calls.

SUMMARY STEPS

1. **offline**
2. **backup category {all | configuration | data}**
3. **continue**
4. **show backup history**
5. **show backup server**

DETAILED STEPS

	Command or Action	Purpose
Step 1	offline Example: se-10-0-0-0# offline	Enters offline mode. All active voice-mail calls are terminated.
Step 2	backup category {all configuration data} Example: se-10-0-0-0(offline)# backup category all se-10-0-0-0(offline)# backup category configuration se-10-0-0-0(offline)# backup category data	Specifies the type of data to be backed up and stored.
Step 3	continue Example: se-10-0-0-0(offline)# continue	Exits offline mode and returns to EXEC mode.
Step 4	show backup history Example: se-10-0-0-0# show backup history	Displays the backup and restore procedures and the success or failure of those attempts.
Step 5	show backup server Example: se-10-0-0-0# show backup server	Displays the backup files available on the backup server, the date of each backup, and the backup file ID.

Examples

The following example displays the output from the **show backup** commands:

```
se-10-0-0-0# show backup history

#Start Operation
Category:      Configuration
Backup Server: ftp://10.100.10.215/CUE_backup
Operation:     Backup
Backupid:     2
Restoreid:    -1
Description:   test backup 1
Date:         Sun Jun 13 12:32:48 PDT 1993
Result:       Success
Reason:
#End Operation

#Start Operation
Category:      Data
Backup Server: ftp://10.100.10.215/CUE_backup
Operation:     Backup
Backupid:     2
Restoreid:    -1
Description:   CUE test backup
```

```

Date:          Sun Jun 13 12:32:57 PDT 1993
Result:        Success
Reason:
#End Operation

#Start Operation
Category:      Configuration
Backup Server: ftp://10.100.10.215/CUE_backup
Operation:     Restore
Backupid:      2
Restoreid:     1
Description:
Date:          Sun Jun 13 12:37:52 PDT 1993
Result:        Success
Reason:
#End Operation

#Start Operation
Category:      Data
Backup Server: ftp://10.100.10.215/CUE_backup
Operation:     Restore
Backupid:      2
Restoreid:     1
Description:
Date:          Sun Jun 13 12:38:00 PDT 1993
Result:        Success
Reason:
#End Operation

se-10-0-0-0# show backup server

Category:      Data
Details of last 5 backups
Backupid:      1
Date:          Tue Jul 22 10:55:52 PDT 2003
Description:

Backupid:      2
Date:          Tue Jul 29 18:06:33 PDT 2003
Description:

Backupid:      3
Date:          Tue Jul 29 19:10:32 PDT 2003
Description:

Category:      Configuration
Details of last 5 backups
Backupid:      1
Date:          Tue Jul 22 10:55:48 PDT 2003
Description:

Backupid:      2
Date:          Tue Jul 29 18:06:27 PDT 2003
Description:

Backupid:      3
Date:          Tue Jul 29 19:10:29 PDT 2003
Description:

se-10-0-0-0#

```

Restoring Files

After the backup files are created, you can restore them when needed. Restoring is done in offline mode, which terminates all voice-mail active voice-mail calls and IMAP and VoiceView Express sessions. It does not permit new voice-mail calls (auto attendant calls are permitted) or new IMAP and VoiceView Express sessions. You should consider doing the restore when telephone subscribers are least likely to be on the telephone.

Use the **show backup server** command to locate the backup ID of the file that you want to restore.

SUMMARY STEPS

1. **show backup server**
2. **offline**
3. **restore id *backupid* category {all | configuration | data}**
4. **show backup history**
5. **reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	show backup server Example: se-10-0-0-0# show backup server	Lists the data and configuration backup files. Look at the backup ID field for the revision number of the file that you want to restore.
Step 2	offline Example: se-10-0-0-0# offline	Enters offline mode. All active voice-mail calls are terminated.
Step 3	restore id <i>backupid</i> category {all configuration data} Example: se-10-0-0-0(offline)# restore id 22 category all se-10-0-0-0(offline)# restore id 8 category configuration se-10-0-0-0(offline)# restore id 3 category data	Specifies the backup ID <i>backupid</i> value and the file type to be restored.
Step 4	show backup history Example: se-10-0-0-0# show backup history	Displays the backup and restore procedures and the success or failure of those attempts.
Step 5	reload Example: se-10-0-0-0(offline)# reload	Resets the Cisco Unity Express module so that the restored values take effect.

Example

The following example displays the backup server and backup history:

```

se-10-0-0-0# show backup server

Category:      Data
Details of last 5 backups
Backupid:      1
Date:          Tue Jul 22 10:55:52 PDT 2003
Description:

Backupid:      2
Date:          Tue Jul 29 18:06:33 PDT 2003
Description:

Backupid:      3
Date:          Tue Jul 29 19:10:32 PDT 2003
Description:

Category:      Configuration
Details of last 5 backups
Backupid:      1
Date:          Tue Jul 22 10:55:48 PDT 2003
Description:

Backupid:      2
Date:          Tue Jul 29 18:06:27 PDT 2003
Description:

Backupid:      3
Date:          Tue Jul 29 19:10:29 PDT 2003
Description:

se-10-0-0-0#

se-10-0-0-0# show backup history

Start Operation
Category:      Configuration
Backup Server: ftp://10.100.10.215/CUE_backup
Operation:     Backup
Backupid:      1
Restoreid:     -1
Description:   test backup 1
Date:          Sun Jun 13 12:23:38 PDT 1993
Result:        Failure
Reason:        Script execution failed: /bin/BR_VMConfig_backup.sh: returnvalue:1
               ; Server Url:ftp://10.100.10.215/CUE_backup: returnvalue:9 Unable to authenticate
#End Operation

#Start Operation
Category:      Data
Backup Server: ftp://10.100.10.215/CUE_backup
Operation:     Backup
Backupid:      1
Restoreid:     -1
Description:   test backup 1
Date:          Sun Jun 13 12:23:44 PDT 1993
Result:        Failure
Reason:        Script execution failed: /bin/BR_VMData_backup.sh: returnvalue:1
               Voicemail Backup failed; Server Url:ftp://10.100.10.215/CUE_backup: returnvalue:9

```

```

Unable to authenticate
#End Operation

#Start Operation
Category:      Configuration
Backup Server: ftp://10.100.10.215/CUE_backup
Operation:     Backup
Backupid:     2
Restoreid:    -1
Description:   CUE test backup
Date:         Sun Jun 13 12:32:48 PDT 1993
Result:       Success
Reason:
#End Operation

#Start Operation
Category:      Data
Backup Server: ftp://10.100.10.215/CUE_backup
Operation:     Backup
Backupid:     2
Restoreid:    -1
Description:   CUE test backup
Date:         Sun Jun 13 12:32:57 PDT 1993
Result:       Success
Reason:
#End Operation

```

Copying Configurations

The following Cisco Unity Express EXEC commands are available to copy the startup configuration and running configuration to and from Flash memory, the network FTP server, and the network TFTP server.

Copying from Flash Memory to Another Location

Starting in Cisco Unity Express EXEC mode, use the following command to copy the startup configuration in Flash memory to another location:

```
copy startup-config {ftp: user-id:password@ftp-server-address / [directory] | tftp:tftp-server-address} filename
```

Keyword or Argument	Description
ftp: <i>user-id:password@</i>	User ID and password for the FTP server. Include the colon (:) and the at sign (@) in your entry.
<i>ftp-server-address</i>	IP address of the FTP server.
<i>/directory</i>	(Optional) Directory on the TFTP server where the copied file will reside. If you use it, precede the name with the forward slash (/).
tftp: <i>tftp-server-address</i>	IP address of the TFTP server.
<i>filename</i>	Name of the destination file that will contain the copied startup configuration.

This command is interactive and prompts you for the information. You cannot enter the parameters in one line. The following examples illustrate this process.

In this example, the startup configuration is copied to the FTP server, which requires a user ID and password to transfer files. The IP address of the FTP server is 172.16.231.193. The startup configuration file is saved on the FTP server with the filename start.

```
se-10-0-0-0# copy startup-config ftp
Address or name of remote host? admin:voice@172.16.231.193
Source filename? start
```

The following example shows the startup configuration copied to the TFTP server, which does not require a user ID and password. The IP address of the TFTP server is 172.16.231.190. The startup configuration is saved in the TFTP directory configs as filename temp_start.

```
se-10-0-0-0# copy startup-config tftp
Address or name of remote host? 172.16.231.190/configs
Source filename? temp_start
```

Copying from the Network FTP Server to Another Location

Starting in Cisco Unity Express EXEC mode, use the following command to copy the network FTP server configuration to another location:

```
copy ftp: {running-config | startup-config} user-id:password@ftp-server-address [directory]
filename
```

Keyword or Argument	Description
running-config	Active configuration in Flash memory.
startup-config	Startup configuration in Flash memory.
<i>user-id:password@</i>	User ID and password for the FTP server. Include the colon (:) and the at sign (@) in your entry.
<i>ftp-server-address</i>	IP address of the FTP server.
<i>/directory</i>	(Optional) Directory name for retrieving the file. If you use it, precede the name with the forward slash (/).
<i>filename</i>	Name of the source file to be copied.

This command is interactive and prompts you for the information. You cannot enter the parameters in one line. The following example illustrates this process.

In this example, the FTP server requires a user ID and password. The IP address of the FTP server is 10.3.61.16. The file start in the FTP server configs directory is copied to the startup configuration.

```
se-10-0-0-0# copy ftp: startup-config
!!!WARNING!!! This operation will overwrite your startup configuration.
Do you wish to continue[y]? y
Address or name or remote host? admin:voice@10.3.61.16/configs
Source filename? start
```

Copying the Flash Running Configuration to Another Location

Starting in Cisco Unity Express EXEC mode, use the following command to copy the running configuration in Flash memory to another location:

```
copy running-config {ftp: user-id:password@ftp-server-address [directory] | startup-config | tftp:ftp-server-address} filename
```

Keyword or Argument	Description
ftp: <i>user-id:password@</i>	User ID and password for the FTP server. Include the colon (:) and the at sign (@) in your entry.
<i>ftp-server-address</i>	IP address of the FTP server.
<i>/directory</i>	(Optional) Directory on the FTP server where the copied file will reside. If you use it, precede the name with the forward slash (/).
startup-config	Startup configuration in Flash memory.
tftp: <i>ftp-server-address</i>	IP address of the TFTP server.
<i>filename</i>	Name of the destination file that will contain the copied running configuration.

When you copy the running configuration to the startup configuration, enter the command on one line.

When you copy to the FTP or TFTP server, this command becomes interactive and prompts you for the information. You cannot enter the parameters in one line. The following example illustrates this process.

In the following example, the running configuration is copied to the FTP server, which requires a user ID and password. The IP address of the FTP server is 172.16.231.193. The running configuration is copied to the configs directory as file saved_start.

```
se-10-0-0-0# copy running-config ftp:  
Address or name of remote host? admin:voice@172.16.231.193/configs  
Source filename? saved_start
```

In the following example, the running configuration is copied to the startup configuration as file start. In this instance, enter the command on a single line.

```
se-10-0-0-0# copy running-config startup-config start
```

Copying the Network TFTP Configuration to Another Location

Starting in Cisco Unity Express EXEC mode, use the following command to copy the network TFTP configuration to another location:

```
copy tftp: { running-config | startup-config } tftp-server-address [directory] filename
```

Keyword or Argument	Description
running-config	Active configuration in Flash memory.
startup-config	Startup configuration in Flash memory.
<i>tftp-server-address</i>	IP address of the TFTP server.
<i>/directory</i>	(Optional) Directory on the TFTP server where the copied file will reside. If you use it, precede the name with the forward slash (/).
<i>filename</i>	Name of the source file to be copied.

This command is interactive and prompts you for the information. You cannot enter the parameters in one line. The following example illustrates this process.

In this example, the TFTP server has IP address 10.3.61.16. The file `start` in directory `configs` on the TFTP server is copied to the startup configuration.

```
se-10-0-0-0# copy tftp: startup-config
!!!WARNING!!! This operation will overwrite your startup configuration.
Do you wish to continue[y]? y
Address or name of remote host? 10.3.61.16/configs
Source filename? start
```

Restoring Factory Default Values

Cisco Unity Express provides a command to restore the factory default values for the entire system. Restoring the system to the factory defaults erases the current configuration. This function is available in offline mode.



Caution

This operation is irreversible. All data and configuration files are erased. Use this feature with caution. We recommend that you do a full system backup before proceeding with this feature.

When the system is clean, the administrator sees a message that the system will reload, and the system begins to reload. When the reload is complete, the system prompts the administrator to go through the postinstallation process.

When logging in to the graphical user interface (GUI), the administrator has the option to run the initialization wizard.

Perform the following steps to reset the system to Cisco Unity Express factory default values.

Step 1 se-10-0-0-0# **offline**

This command puts the system into offline mode.

Step 2 (offline)# **restore factory default**

```
This operation will cause all the configuration and data on the system to be erased. This operation is not reversible. Do you wish to continue? (n)
```

Step 3 Do one of the following:

- Enter **n** if you want to retain the system configuration and data.

The operation is cancelled, but the system remains in offline mode. To return to online mode, enter **continue**.

- Enter **y** if you want to erase the system configuration and data.

When the system is clean, a message appears indicating that the system will start to reload. When the reload is complete, a prompt appears to start the postinstallation process.

Backup and Restore Using SFTP

This section discusses the following topics:

- [Overview, page 247](#)
- [Configuring Backup and Restore Using SFTP, page 247](#)

Overview

This feature enables you to transfer files from any Cisco Unity Express application to and from the backup server using Secure File Transfer Protocol (SFTP). SFTP provides data integrity and confidentiality that is not provided by FTP.

Because SFTP is based on Secure Shell tunnel version 2 (SSHv2), only SSHv2 servers are supported for this feature.

To run backup and restore over SFTP, you must configure the URL of the backup server in the form of `sftp://hostname/dir`, and also the username and password to login to the server. The backup server must have an SSH daemon running with the SFTP subsystem enabled. The SSH protocol allows various user authentication schemes. In Version 3.0, however, only password authentication is supported.

Configuring Backup and Restore Using SFTP

Prerequisites

There are no prerequisites.

Required Data for This Procedure

There is no data required.

SUMMARY STEPS

1. `config t`
2. `backup {revisions number | server url sftp-url username sftp-username password sftp-password}`
3. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>config t</code> Example: <code>se-10-0-0-0# config t</code>	Enters configuration mode.
Step 2	<code>backup {revisions number server url sftp-url username sftp-username password sftp-password}</code> Example: <code>se-10-0-0-0(config)# backup server url sftp://branch/vmbackups username admin password mainserver</code>	Performs a backup to the specified SFTP or FTP server. To use SFTP, the URL must be of the form <code>sftp://hostname/directory</code> .
Step 3	<code>end</code> Example: <code>se-10-0-0-0(config)# end</code>	Returns to privileged EXEC mode.

Backup Server Authentication Using a SSH Host Key

This section discusses the following topics:

- [Overview, page 247](#)
- [Configuring Backup Server Authentication Without Using the SSH Host Key, page 249](#)
- [Configuring Backup Server Authentication Using the SSH Host Key, page 250](#)

Overview

This feature enables you to authenticate the backup server using the SSH protocol before starting a backup/restore operation. The SSH protocol uses public key cryptography for server authentication.

This feature provides two methods of authenticating a server:

- Establishing a secure connection based only on the URL of a trusted backup server.
- Obtaining the fingerprint of the backup server and using it to establish a secure connection. This fingerprint is also known as the host key or private key.

The first method is easier than the second method, but it is less secure because it does not depend on you knowing the backup server's private host key. However, if you know the URL of a trusted backup server, it is generally safe. In this case, the backup server securely provides the client with its private host key.

In both cases, when server authentication is enabled, the system validates the SSH server's private host key by comparing the fingerprint of the key received from the server with a preconfigured string. If the two fingerprints do not match, the SSH handshake fails, and the backup/restore operation does not occur.

You cannot use the GUI to configure this feature; you must use the CLI.

Both methods are explained in the following sections.

Configuring Backup Server Authentication Without Using the SSH Host Key

Prerequisites

There is no prerequisites.

Required Data for This Procedure

To enable SSH authentication of a backup server without knowing the server's fingerprint (private host key), you must know the URL of a trusted backup server.

SUMMARY STEPS

1. `config t`
2. `backup server url sftp://url`
3. `backup server authenticate`
4. `end`
5. `show security ssh knowhost`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>config t</code> Example: se-10-0-0-0# config t	Enters configuration mode.
Step 2	<code>backup server url sftp://url</code> Example: se-10-0-0-0(config)# backup server url sftp://company.com/server22	Establishes a initial connection with the backup server.
Step 3	<code>backup server authenticate</code> Example: se-10-0-0-0(config)# backup server authenticate	Retrieves the fingerprint of the backup server's host key and establishes a secure SSH connection.
Step 4	<code>end</code> Example: se-10-0-0-0(config)# end	Returns to privileged EXEC mode.
Step 5	<code>show security ssh knowhost</code> Example: se-10-0-0-0(config)# show security ssh knowhost	Displays a list of configured SSH servers and their fingerprints.

Configuring Backup Server Authentication Using the SSH Host Key

Prerequisites

There is no prerequisites.

Required Data for This Procedure

To use a backup server's fingerprint (private host key) to enable SSH authentication, you must first retrieve the fingerprint "out-of-band" by running the **ssh-keygen** routine on the backup server. This routine is included in the OpenSSH package. The following example shows the command and its output:

```
ssh-keygen -l -f /etc/ssh/ssh_host_dsa_key.pub
```

```
1024 4d:5c:be:1d:93:7b:7c:da:56:83:e0:02:ba:ee:37:c1 /etc/ssh/ssh_host_dsa_key.pub
```

SUMMARY STEPS

1. **config t**
2. **security ssh knownhost** *host* {**ssh-rsa** | **ssh-dsa**} *fingerprint-string*
3. **end**
4. **show security ssh knownhost**

DETAILED STEPS

	Command or Action	Purpose
Step 1	config t Example: se-10-0-0-0# config t	Enters configuration mode.
Step 2	security ssh knownhost <i>host</i> { ssh-rsa ssh-dsa } <i>fingerprint-string</i> Example: se-10-0-0-0(config)# security ssh knownhost server.cisco.com ssh-rsa a5:3a:12:6d:e9:48:a3:34:be:8f:ee:50:30:e5:e6:c3	Configures the MD5 fingerprint of the SSH server's host key using the following arguments and keywords: <i>host</i> — Fully qualified hostname or IP address of the SSH server. ssh-rsa — RSA algorithm was used to create this fingerprint for a SSH server's host key. ssh-dsa — DSA algorithm was used to create this fingerprint for a SSH server's host key. <i>fingerprint-string</i> — MD5 fingerprint string.

	Command or Action	Purpose
Step 3	<code>end</code> Example: <code>se-10-0-0-0(config)# end</code>	Returns to privileged EXEC mode.
Step 4	<code>show security ssh knownhost</code> Example: <code>se-10-0-0-0(config)# show security ssh knownhost</code>	Displays a list of configured SSH servers and their fingerprints.

Encrypting and Signing of Backup Content on the Server

This section discusses the following topics:

- [Overview, page 251](#)
- [Configuring the Encryption and Signing of Backup Content on the Server, page 251](#)

Overview

This feature enables you to protect backed up configuration and data files using signing and encryption before the files are transferred to the backup server.

To enable this feature, you must configure a master key, from which the encryption and signing key (known as the session key) are derived. The backup files are encrypted and signed before they are sent to the backup server. When you restore the files, the master key is used to validate the integrity of the files and decrypt them accordingly. You can also restore the backup files to any other machine running Cisco Unity Express 3.0, if you configure the same master key before you begin the restore process. To make it easier to automate a scheduled backup, the master key is stored securely on the hosting device. It is not included in the backup content.

During the restore process, if the system detects that backup content has been tampered with, the restore process aborts. The system also halts and waits for the administrator to take some action, such as restoring using a different revision.

For backward compatibility, you can allow unsigned backup files to be restored if the risk is acceptable.

Configuring the Encryption and Signing of Backup Content on the Server

Prerequisites

There are no prerequisites.

Required Data for This Procedure

There is no data required.

SUMMARY STEPS

1. `config t`
2. `backup security key generate`
3. `backup security protected`
4. `backup security enforced`
5. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>config t</code> Example: <code>se-10-0-0-0# config t</code>	Enters configuration mode.
Step 2	<code>backup security key generate</code> Example: <code>se-10-0-0-0(config)# backup security key generate</code>	Creates the master key used for encrypting and signing the backup files.
Step 3	<code>backup security protected</code> Example: <code>se-10-0-0-0(config)# backup security protected</code>	Enables secure mode for backups. In secure mode, all backup files are protected using encryption and a signature.
Step 4	<code>backup security enforced</code> Example: <code>se-10-0-0-0(config)# backup security enforced</code>	Specifies that only protected and untampered backup files are restored.
Step 5	<code>end</code> Example: <code>se-10-0-0-0(config)# end</code>	Returns to privileged EXEC mode.

Encrypting PINs in Backup Files

Before 3.0, PINs were stored as clear text in LDAP and were therefore visible in the backup file. This is because user PINs are stored in LDAP, which is backed up in LDIF format. This feature applies SHA-1 hash encryption to PINs before storing them in the LDAP database. As a result, when a user logs in to voice mail, the PIN they submit is hashed and compared to the PIN attribute retrieved from LDAP directory.

To migrate from earlier version, you must convert from a clear PIN to a hashed PIN in the LDAP directory. This conversion is typically done right after a system upgrade from an earlier version or after a restore operation from an old backup. At this point, the clear PIN is removed from the database and replaced with the encrypted PIN.

Because encryption using SHA-1 is not reversible, after the conversion is complete, you cannot disable or turn off this feature to restore the encrypted PIN to its clear form.

**Note**

This feature, does not require any configuration using the GUI or CLI.
