COURSE TECHNOLOGY
CENGAGE Learning

Linux Operations and Administration

Ahmad AlRjoub
ahmadrj@ksu.edu.sa

Chapter Six

Managing Data: Backup and Recovery Processes

Objectives

- Perform system backups at the command line
- Compress and decompress files at the command line
- Explain how to schedule backups with a cron table

System Backups

- System backup
  - Copies files and directories to an archive
  - Use to restore data in case of a system failure or data loss and corruption
- Archive
  - File containing many other files, each of which is still identified by its filename, owner, permissions, and timestamp
- Most common backup medium is tape

System Backups (cont'd.)

- Directories to include in regular backups
  - `/etc`—contains core configuration files, security files, network configuration files, user and group information, etc
  - `/home`—each user has a `/home` directory
  - `/opt`—software and packages added after the default installation
  - `/root`—root user's home directory
  - `/var`—system-specific information that changes while the system is running normally

The Tar Utility

- Tar (tape archive) utility
  - Creates archives by combining files and directories into a single file
  - Used to work with tar archives
  - Use it to store files on a hard disk, CD/DVD, or even on a network
- Table 6-1
  - Most common tar options

The Tar Utility (cont'd.)

Operation	Description
`-c` or `--create`	Creates an archive file
`-t` or `--list`	Lists an archive's contents
`-x` or `--extract`	Extracts an archive's contents
`-f` or `--file`	Specifies the archive file's name and location
`-v` or `--verbose`	Displays details about copying files to and extracting files from archives
`-z` or `--gzip, --ungzip`	Filters an archive through gzip

Table 6-1 Common tar options

Advanced Tar Options

- Example: `tar -cf files.tar file1 file2`
- Activity 6-1: Using the Tar Utility
  - Create, view, and extract tar archives members
- Members
  - Different from files
  - Can be viewed only with the `tar` command's `-t` option
  - Can extract only a few members from an archive
    - When a user deletes a file accidentally

Advanced Tar Options (cont'd.)

- Example of viewing members:

```
1. tar -tvf fruit.tar
2. -rw-r--r-- martha/users 0
2012-05-01 11:22 apple
3. -rw-r--r-- martha/users 0
2012-05-01 11:22 banana
4. -rw-r--r-- martha/users 0
2012-05-01 11:22 orange
5. tar -xvf fruit.tar apple
6. apple
```

Using Tar to Perform Full Backups

- Table 6-2
  - Advanced options for tar
- Activity 6-2: Using the Advanced Tar Options
  - Use advanced tar options
- Table 6-3
  - Modifying tar options
- Activity 6-3: Using Tar Options to Modify Archive Members' File Information
  - Use advanced tar options to modify file information

Using Tar to Perform Full Backups (cont'd.)

Option	Description
`-r` or `--append`	Adds files to an existing archive.
`-u` or `--update`	Compares the date and time of a member with the date and time of the file with the same name. If the file was modified after the archive was created, its newer version is added to the archive.
`-A` or `--concatenate`	Similar to the append option but adds one archive to another archive.
`--delete`	Removes specific members from a tar archive. This option doesn't have a short name.
`-d` or `--compare`	Compares specified members with files in the file system having the same name and reports differences in file size, mode, owner, modification date, and contents.

Table 6-2 Advanced tar options

Using Tar to Perform Full Backups (cont'd.)

Option	Description
`--mode`	Changes a member's permissions in an archive with numeric or symbolic notation. For instance, you could use `tar -cvf backup.tar --mode=777 /home` or `tar -cvf backup.tar --mode='a=rwx' /home`.
`--mtime`	Changes a member's modification date in the tar archive. For instance, `tar -cf backup.tar --mtime='2012-05-02' /home` archives everything in `/home`, and the members in the archive have 2012-05-02 as the modification date. You can also use terms such as `today` or `yesterday` instead of a specific date.
`--owner`	Changes the member's owner in the tar archive. For instance, `tar -cf backup.tar --owner=root /home` creates a tar archive of your `/home` directory and makes root the new owner.
`--group`	Similar to `--owner`, except it's used to change the member's group instead of owner.

Table 6-3 Modifying tar options

Using Tar to Perform Incremental Backups

- Full backup
  - Archive of all files on the file system
  - Never perform a full backup while users are accessing the system
    - Tar file could be corrupted if files are modified during the backup process
- Command to perform full backup with tar:
  - `tar -cvf backup0.tar -V "This is a full backup of the /home directory" --listed-incremental=/home/backup.snap *`

Using Tar to Perform Incremental Backups (cont'd.)

- Creates a full backup of the `/home` directory
- `-cvf` options
  - Create an archive called backup0.tar
- `-V` option
  - Label the archive with the text between the quotation marks
- `--listed-incremental` option
  - Creates a snapshot file named `backup.snap`
  - Used during incremental backups to determine which files have changed since the last backup

Using Tar to Perform Incremental Backups (cont'd.)

- Incremental backup
  - Archive containing only files modified since the last backup
- Organizations have different backup strategies, depending on their needs
- Syntax for creating an incremental backup
  - Same as the command for full backups
  - If the snapshot file already exists, the tar utility examines it to determine whether any files have changed

Using Tar to Perform Incremental Backups (cont'd.)

- Table 6-4
  - Important options used with the tar command when creating full or incremental backups
- Activity 6-4: Performing Full and Incremental Backups with the Tar Utility
  - Use the tar utility to perform full and incremental backups

Option	Description
`-V` or `--label`	Adds a volume header to a tar file. A volume header is simply a digital label used to describe the file. Physically labeling tapes, CDs, and DVDs is important, but you should also digitally label tar files with the `--label` option. You can view the volume header with the `-t` option.
`-W` or `--verify`	Verifies that files have been included in the archive correctly.
`-g` or `--listed-incremental`	Forces tar to archive only files that have been modified since the last backup (full or incremental) by analyzing the **snapshot file**. A snapshot file is created during a full backup and contains the condition of the backed-up directory to determine which files were modified since the last backup. This is a good practice because there's no need to back up files that have already been backed up and haven't changed. The snapshot file is given as an argument to the `--listed-incremental` option so that the tar utility knows which files have been changed, added, or deleted since the last backup.
`--no-check-device`	When a device number changes (which can happen when upgrading a kernel version, for example), an incremental backup might back up files that haven't been changed. This option prevents this from happening by forcing the tar utility to not rely on device numbers when preparing for an incremental backup.
`--check-device`	The default option; necessary only when you want to undo the `--no-check-device` option.

Table 6-4 Full and incremental backup options

The Cpio Utility

- Cpio (copy in/out) utility
  - Uses the results of the `ls` or `find` command to generate files to be archived
- Operates in three modes:
  - Copy-out
    - Cpio creates an archive from the output of the `ls` or `find` commands
  - Copy-in
    - Extracts files from an archive
  - Copy-pass
    - Copies files from one directory to another

Standard Input, Standard Output, and Redirection

- Default standard input
  - Keyboard
- < symbol
  - Redirect input from the keyboard
- Default standard output
  - Screen
- Output redirection operators redirect a command's output to a file
  - > (the "greater than" symbol)
  - | (the "pipe" symbol)

Standard Input, Standard Output, and Redirection (cont'd.)

- `ls > home_listing`
  - Redirect `ls` command's output to the `home_listing` file instead of displaying the output onscreen
- `ls | more`
  - Redirect `ls` output to `more` command

Copy-Out Mode

- `-o` or `--create` option
  - Create archives by accepting the output of `ls` or `find` as the input for an archive
- Example: `ls | cpio -o > files.cpio`
  - Result of `ls` command used to determine which files to archive in `files.cpio`
- `-v` option
  - Display which files are being archived

Copy-In Mode

- Extract an archive
- Use standard input redirection symbol (<) to extract the archive members

```
cpio -iv < demo.cpio
/home/student1/Demo/file1
/home/student1/Demo/file2
1 block
```

- `-i` option
  - Extract files from a cpio archive
- `-v` option
  - Lists files as they're being extracted

Copy-Pass Mode

- Copy files and directories from one directory and paste them in another directory
  - Without actually creating an archive
- Not a practical option for backups
- Cpio utility
  - Preserves modification times and ownership
- `-p` option
  - Copy files from one directory tree to another

Copy-Pass Mode (cont'd.)

```
find . | cpio -pv
/home/dustin/dir2
/home/dustin/dir2/./juice
/home/dustin/dir2/./water
/home/dustin/dir2/./soda
0 blocks
```

- Activity 6-5: Using the Cpio Utility
  - Use the cpio utility to create and extract an archive

Compression

- Standard DVD can hold 4.7 GB of data
- Compress data to:
  - Save storage space
  - Make it fit on removable media
  - Transfer it across the network faster
- Compression
  - Reduces the size of data to store information in less space
- Two most common compression utilities:
  - Gzip and bzip2

The Gzip Utility

- File compressed with gzip has:
  - Extension .gz
  - Same file permissions, ownership, and modification time as the original file
  - Much smaller file size
- Compression ratio
  - Defines by how much a file is reduced after compressing it
  - Text files are typically reduced by 60% to 70% with compression
  - `-v` option: view compression ratio

The Gzip Utility (cont'd.)

- Table 6-5
  - Options used with the gzip utility
- Combine tar and gzip commands
  - Compress an archive as you create it

```
1. tar -czf commands.tar.gz cd.man ls.man pwd.man
2. tar -cf commands.tar cd.man ls.man pwd.man
3. ~/dir1> ls -l commands.tar
4. -rw-r--r-- 1 linda users 20480 2012-05-08 21:09
commands.tar
5. ~/dir1> ls -l commands.tar.gz
6. -rw-r--r-- 1 linda users 3802 2012-05-08 21:09
commands.tar.gz
```

The Gzip Utility (cont'd.)

Option	Description
`-c`	Keeps the original files unchanged. For instance, if you type `gzip -c file1 > file1.gz`, you have two files: `file1` and `file1.gz`. The `file1.gz` file is the compressed file. The `-c` option is also used to concatenate files. For instance, if you type `gzip -c file1 file2 > file3.gz`, `file3.gz` is one compressed file, which contains the contents of `file1` and `file2`.
`-h`	Displays an information page describing the options used with the gzip utility.
`-v`	Displays information about a file as it's being compressed, such as the compression ratio.

Table 6-5 Gzip options

The Gzip Utility (cont'd.)

- Activity 6-6: Using the Gzip Utility
  - Compress and uncompress files with command-line utilities

The Bzip2 Utility

- Bzip2 utility
  - Compresses files and adds the .bz2 extension
- Bunzip2 utility
  - Uncompress files
- Table 6-6
  - Options for bzip2
- Activity 6-7: Using the Bzip2 Utility
  - Compress and uncompress files with the bzip2 utility

The Bzip2 Utility (cont'd.)

Option	Description
`-c`	Use this option to keep the original files unchanged. For instance, if you type `bzip2 -c file1 > file1.bz2`, you have two files: `file1` and `file1.bz2` (which is the compressed file). Another use of the `-c` option is to concatenate files. For example, the command `bzip2 -c file1 file2 > file3.bz2` concatenates `file1` and `file2` into the compressed file `file3.bz2`.
`-v`	Displays information, such as compression ratio and filename, about the file as it's being compressed.

Table 6-6 Bzip2 options

Scheduling Backups

- Scheduling system backups during off hours is best
  - Fewer people are logged in to the system
  - Less disruption in service
- Cron daemon (crond)
  - System daemon
  - Uses a configuration file called a cron table to schedule commands that run at specified times
- Cron table has six fields
  - Table 6-7 describes the first five fields
  - Sixth field contains the command to run

Scheduling Backups (cont'd.)

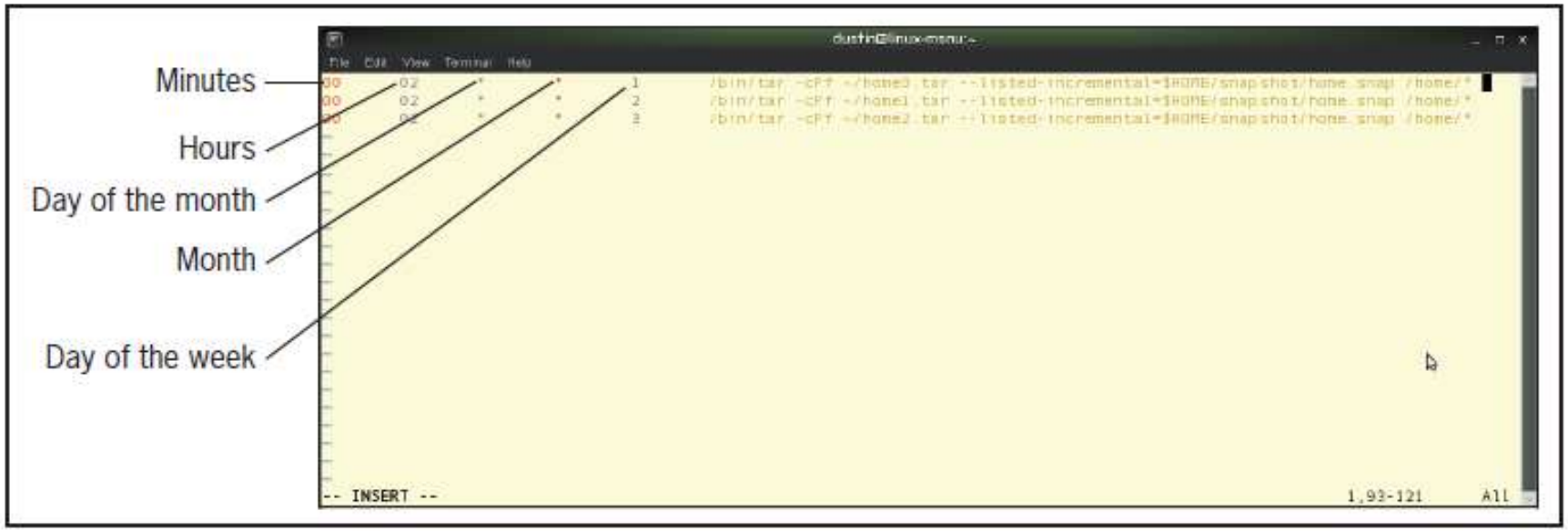

Figure 6-1 A cron table

Scheduling Backups (cont'd.)

Field	Allowed values
1: minutes	0–59
2: hours	0–23 (uses a 24-hour clock)
3: day of the month	1–31
4: month	1–12
5: day of the week	0–6, with 0 representing Sunday and 6 representing Saturday

Table 6-7 The cron table format

Scheduling Backups (cont'd.)

- Cron daemon uses two types of cron tables:
  - User cron table in `/var/spool/cron/tabs`
  - System cron table in `/etc/crontab`
- Users use the user cron table to schedule tasks
- System uses the system cron table to schedule system tasks

User Cron Tables

- Every user (including root) on an openSUSE system has a cron table
- `crontab` command
  - Create, delete, and list cron tables
  - Using the options listed in Table 6-8
- Activity 6-8: Editing a User Cron Table
  - Edit a user cron table

User Cron Tables (cont'd.)

Options	Description
-e	Opens the vim editor to edit the current user's cron table. If the cron table doesn't exist, a blank table is created for you to edit.
-u	Specifies the name of the user whose cron table is to be edited.
-l	Displays the current user's cron table.
-r	Removes the current user's cron table.

Table 6-8 Options used with the crontab command

System Cron Table

- System cron table
  - Schedule tasks
  - Sixth field is used to specify the user account for issuing commands
- Only the root user can edit the system cron table
- Commands scheduled via the system cron table are for backups and system maintenance
- `vim /etc/crontab`
  - Start the vim editor so that to view and edit the system cron table

System Cron Table (cont'd.)

- Example of the system cron table (see next slide for explanation):

```
SHELL=/bin/sh
PATH=/usr/bin:/usr/sbin:/sbin:/bin:/usr/lib/news/bin
MAILTO=root
20 * * * * root mail student1 < /var/log/messages
```

System Cron Table (cont'd.)

- `SHELL=/bin/sh`—defines the shell the cron table uses to run the scheduled task
- `PATH=/usr/bin:/usr/sbin:/sbin:/bin:/usr/lib/news/bin`—defines the PATH variable
- `MAILTO=root`—defines the MAILTO variable
- `20 * * * * root mail student1 < /var/log/messages`—scheduled task to e-mail student1 system log messages every 20 minutes

Summary

- Several directories should be backed up regularly
  - Particularly users' home directories
- Linux includes backup utilities:
  - Tar and cpio
- Archives
  - Stored on many different types of media, such as tapes, CD-RWs/DVD-RWs, removable media, and hard disks

Summary (cont’d.)

- Compression utilities reduce the size of files so that they can fit on backup media or be sent across the network faster
  - Gzip and bzip2
- Cron table
  - Configuration file used to specify tasks to run at a certain time
  - Common task is scheduling backups