2.9.5 File Management Facts

Several Linux commands are used with files.

This lesson covers the following topics:

- File viewing commands
- File management commands

File Viewing Commands

The following table describes Linux commands that can be used to view the content of files:

Command	Function	Examples
cat	Displays the contents of a file in the shell. This command can display multiple files at once.	• **cat myfile** displays the contents of **myfile**. • **cat myfile yourfile** displays the contents of **myfile** and **yourfile** together.
less	Displays the contents of a file, pausing one screen at a time. • Use the Spacebar to scroll to the next screen. • Use the Up arrow and Down arrow to scroll up and down. • Press q to exit.	• **less bigfile** displays the contents of **bigfile** one screen at a time so it can be read.
head	Lists the first 10 lines (the default) of a specified file. Use the **-n** option to specify a specific number of lines to display.	• **head /home/user/myfile** lists the first 10 lines of **myfile**. • **head -n 20 /home/user/myfile** lists the first 20 lines of **myfile**. • **head -n -35 /home/user/myfile** displays all lines in **myfile**, omitting the last 35 lines.
tail	Lists the last 10 lines (the default) of a specified file. • **-n** specifies a specific number of lines. • **-f** monitors the file.	• **tail /home/user/myfile** lists the last 10 lines of **myfile**. • **tail -n 20 /home/user/myfile** lists the last 20 lines of **myfile**. • **tail -n -15 /home/user/myfile** displays all lines in **myfile**, omitting the first 15 lines.

		• **tail -f /var/firewalld** displays the last 10 lines of **/var/firewalld** and then dynamically displays new lines in the file as they are added.

File Management Commands

The following table describes Linux commands that can be used to manage files.

Command	Function	Examples
touch	If the file does not exist, **touch** creates a blank version of the file. If the file does exist, this command updates the file's modification and last accessed times.	• **touch myfile** makes a blank file named **myfile**.
file	Shows the file type. The **file** command is useful because Linux does not require file extensions. The **file** command uses file signatures in: • **/usr/share/misc/magic** • **/usr/share/misc/magic.mgc** • **/etc/magic**	• **file myfile** shows whether **myfile** is a text, data, xml, or other type of file.
cp	Copies files. Copying leaves the source file intact. • **-f** overwrites files that already exist in the destination directory. • **-i** prompts before overwriting a file in the destination directory.	• **cp /temp/document_ab.txt ~/doc/document.txt** copies **document_ab.txt** from the **/temp** directory to the **~/doc** directory and renames the file to **document.txt**. • **cp /temp/*.txt ~/doc** copies all text files from the **/temp** directory to the **~/doc** directory.
mv	Moves or renames files (and directories). Moving files erases the source file and moves it to the destination. • **-f** overwrites files that already exist in the destination directory. • **-i** prompts before overwriting a file in the destination directory.	• **mv /temp/document.txt ~/doc/document.txt** moves **document.txt** from the **/temp** directory to the **~/doc** directory. • **mv /temp/*.txt ~/doc/*.txt** copies all text files from the **/temp**

	• **-n** never overwrites files in the destination directory.	directory to the **~/doc** directory.
rm	Removes a file or directory. Use the **-f** option to delete without a prompt. The **rm** command deletes a file or directory's inode, but it does not actually delete its data. To permanently remove data, use the **shred** command.	• **rm myfile** deletes a file in the current directory named **myfile**. • **rm /home/user/myfile** deletes **myfile** from the **/home/user** directory regardless of the current directory. • **rm -f /home/user/temp/*** deletes all files in the temp directory without prompts.
shred	Deletes the file and overwrites the file's data on the hard disk. The **shred** command is useful when deleting files that contain proprietary information or other sensitive data. • **-n** specifies the times to overwrite. The default is 25 times. • **-u** deletes the inode. • **-v** display the progress of the file deletion. • **-z** overwrites the filename with zeros.	• **shred -u -z companysecrets.txt** deletes **companysecrets.txt**, overwrites the file with random information, then leaves zeros in place of the file.
chattr	Modifies file attributes. A + or - is used to add or remove attributes respectively. For example, to make a file immutable (cannot be modified), using the +i flag sets the immutable file attribute and -i removes the immutable file attribute. • **-R** recursively change attributes of directories and their contents. • **-V** displays verbose output and the program version. • **-f** suppresses error messages. • **-p** sets the file's project number. • **-v** sets the file's version/generation number.	• **sudo chattr +i annualreport.txt** sets the immutable file attribute so the file cannot be modified. • sudo chattr +a feedback.txt sets the append attribute to allow users to append data to the file without changing or modifying any existing data.
lsattr	Lists file attributes. • **-R** recursively list attributes of directories and their contents. • **-V** displays the program version. • **-a** lists all files in directories.	• **lsattr /etc/grub/grub.conf** lists the attributes of the **grub.conf** file.

	• **-d** lists directories like other files, rather than listing their contents. • **-v** lists the file's version/generation number.	