12.6.2 Manipulating Files Facts

This lesson covers the following topics:

- File viewing commands
- File management commands

File Viewing Commands

The following commands can be used to control how Linux files are viewed:

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.
head	Lists the first 10 lines (the default) of a specified file. Use the **-n** option to specify the 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

Linux files can also be managed using the following commands:

Command	Description	Examples
chmod	Assigns a special permission. Be aware of the following syntax options: *[decimal_value]* - Sets the permissions for the file according to the numbers represented for each mode category. The special permission precedes the standard octal representation of a set of permissions.	**chmod 4** *xxx* - Sets the set owner userid (SUID). **chmod u+s** - Sets the SUID. **chmod u-s** - Removes the SUID. **chmod 2** *xxx* - Sets the set group ID (SGID). **chmod g+s** - Sets the SGID. **chmod 1** *xxx* - Sets the sticky bit. **chmod u+t** - Sets the sticky bit. **chmod u-t** - Removes the sticky bit.

	▪ Only the first number changes to identify the special permission group settings. ▪ *[category]* + *[permission]* - Adds a special permission for a user, group, or other (category) to a file. ▪ *[category]* - *[permission]* - Removes a special permission for a user, group, or other from a file.	▪ **chmod 6** *xxx* - Sets both the SUID and SGID. ▪ **chmod 7** *xxx* - Sets the SUID, GUID, and sticky bit.
grep	Searches through files for a specified character string. By default, **grep** is context sensitive and displays the string in the context of the line containing the string. ▪ **-A** *[number]* - Prints a specified number of lines following the matching lines. ▪ **-a** - Searches binary (executable) files as though they were text files. ▪ **-B** *[number]* - Prints a specified number of lines before the matching lines. ▪ **-C** *[number]* - Prints a specified number of lines of context around the matching lines. ▪ **-c** - Shows the number of matches of the string for the file. ▪ **-E** - Uses regular expressions for the text pattern. ▪ **-e** *[pattern]* - Specifies a literal pattern. ▪ **-f** - Searches for multiple strings using a file that lists the string patterns. ▪ **-l** - Lists the names of the files with a match. This is used to search multiple files. ▪ **-m** *[number]* - Shows the specified number of matches for a file. ▪ **-n** - Displays the line number of the lines containing the term. ▪ **-r** - Searches the directory and all subdirectories for files containing the term. ▪ **-v** - Displays non-matching lines. ▪ **--include=** *[file_name]* - Searches in files with names that match a specified string. ▪ **--exclude=** *[file_name]* - Searches in files with names that do not match a specified string. ▪ **-w** - Searches for whole words only.	▪ **grep -A 3 Midway ~/docs/WWII-report** - Searches WWII-report for the pattern Midway and prints the line and the next three lines. ▪ **grep -a var11 /bin** - Searches all files, including binary files, in the /bin directory for the pattern var11. ▪ **grep -c Midway ~/docs/WWII-report** - Shows a number representing the number of times the pattern Midway was found in the WWII-report file. ▪ **grep -C -3 Midway ~/docs/WWII-report** - Shows the specified number of lines preceding and following the matching lines. ▪ **grep -e '--count' ~/docs/doc1** - Looks for the pattern --count in the doc1 file rather than interpreting it as an option. ▪ **grep -l -r Midway ~/docs** - Shows the name of all files in the /home/user/docs directory that contain the term Midway. ▪ **grep -m 2 battle ~/docs/WWII-report** - Shows the first two times the term battle is found in the file. ▪ **grep -n -i customVariable1 ~/java/program1.java** - Shows the line numbers of lines that have the term customVariable1 in the program1.java file. This ignores the case. ▪ **grep -r battle ~/docs/** - Searches the directory and all subdirectories for the term battle. ▪ **grep -w tank ~/docs/WWII-report** - Searches for the whole word tank in the file.

logger	Lets you add entries in the system log file. The syntax is **logger <message>**. The message portion of the command can also be entered from the output of another command.	▪ **logger Here is my message** - Adds the line Here is my message to the log file. ▪ **logger 'who'** - Uses the output of the **who** command as the message to be added to the log file. ▪ **logger -f msg** - Adds the contents of the msg file to the log file. ▪ **logger --size 5 1234567890123467890** - Limits the input to the first 5 Kib characters.

Copyright © 2022 TestOut Corporation All rights reserved.