Module I: Introduction to Linux Operating System

Part IV

Basic Commands running in RHEL

There are many commands available in Red Hat Enterprise Linux (RHEL) that can be used to perform various tasks. Here are some basic commands that can be used to manage and navigate the system:

1. **ls**: Lists the files and directories in a directory.

   Syntax: ls [options] [file or directory]

   Example: ls /home/user/Documents

2. **cd**: Changes the current working directory.

   Syntax: cd [directory]

   Example: cd /home/user/Documents

3. **pwd**: Prints the current working directory.

   Syntax: pwd

   Example: pwd

4. **mkdir**: Creates a new directory.

   Syntax: mkdir [directory]

   Example: mkdir /home/user/newfolder

5. **rmdir**: Removes an empty directory.

   Syntax: rmdir [directory]

   Example: rmdir /home/user/emptydir

6. **touch**: Creates a new empty file or updates the timestamp of an existing file.

   Syntax: touch [file]

   Example: touch /home/user/newfile.txt

7. **cp**: Copies a file or directory.

   Syntax: cp [source file or directory] [destination]

   Example: cp /home/user/oldfile.txt /home/user/backup/oldfile.txt

8. **mv**: Moves or renames a file or directory.

   Syntax: mv [source file or directory] [destination]

   Example: mv /home/user/oldfile.txt /home/user/newfile.txt

9. **rm**: Removes a file or directory.

   Syntax: rm [file or directory]

   Example: rm /home/user/oldfile.txt

10. **cat**: Displays the contents of a file.

    Syntax: cat [file]

    Example: cat /home/user/file.txt

11. **more**: Displays the contents of a file one page at a time.

    Syntax: more [file]

    Example: more /home/user/file.txt

12. **less**: Displays the contents of a file one page at a time, with more functionality than more.

    Syntax: less [file]

    Example: less /home/user/file.txt

13. **grep**: Searches for a specified pattern in a file or a group of files.

    Syntax: grep [pattern] [file or directory]

    Example: grep "error" /var/log/messages

14. **find**: Finds files and directories based on specified criteria.

    Syntax: find [directory] [options] [expression]

    Example: find /home/user/ -name "*.txt"

15. **ps**: Lists the processes currently running on the system.

    Syntax: ps [options]

    Example: ps aux

16. **top**: Displays real-time information about the system, including a list of processes and their resource usage.

17. **kill**: Sends a signal to a process to terminate or stop it.

18. **shutdown**: Shut down or reboot the system

These are just a few examples of the many commands available in RHEL. There are many more commands available to perform various tasks, and you can also use options and arguments with these commands to customize their behavior.

VI editors in Linux

The vi editor (short for visual editor) is a text editor that is built into most Linux distributions. It is a command-line based editor, meaning that it is run from the command prompt and does not have a graphical user interface.

Here are some of the key features of the vi editor:

1. Modal editing: The vi editor operates in two modes: command mode and insert mode. In command mode, the user can navigate the document and perform actions such as deleting text, copying text, and saving the document. In insert mode, the user can insert new text into the document.
2. Keyboard shortcuts: The vi editor uses a variety of keyboard shortcuts to perform different actions. For example, the "i" key enters insert mode, the "Esc" key exits insert mode and enters command mode, and the ":wq" command saves the document and exits the editor.
3. Search and replace: The vi editor allows the user to search for text within the document and replace it with other text.
4. Syntax highlighting: The vi editor supports syntax highlighting for different programming languages, making it easier to read and write code.
5. Customizable: The vi editor can be customized with different settings and plugins to suit the user's needs.
6. Works on different platforms: The vi editor is available on most Unix-like operating systems, including Linux, macOS, and BSD.
7. Lightweight: The vi editor is a lightweight text editor that does not require a lot of resources to run, making it a good option for use on older or low-powered computers.
8. High-performance: Because it does not have a GUI, the vi editor can open and save large files very quickly.
9. Vim: Vim is an improved version of vi editor which is widely used in most of the Linux and Unix systems.
10. Nano: Nano is also another text editor which is relatively simpler than vi and vim, widely used for basic editing tasks.

Few examples of using vi editor in Linux

Here are a few examples of how to use the vi editor in Linux:

1. Opening a file: To open a file in the vi editor, simply enter the command "vi <filename>" at the command prompt, replacing <filename> with the name of the file you want to open.

   Example: vi test.txt
2. Inserting text: To insert text into a file, press the "i" key to enter insert mode. Then, type in the text you want to add. Press the "Esc" key to exit insert mode and return to command mode.
3. Saving the file: To save the changes you've made to a file, press the "Esc" key to enter command mode, then type ":wq" and press Enter. This will save the file and exit the editor.
4. Finding and replacing text: To find and replace text in a file, press the "Esc" key to enter command mode, then type ":%s/oldtext/newtext/g" and press Enter. This will replace all occurrences of "oldtext" with "newtext" in the entire file.
5. Moving the cursor: To move the cursor around in a file, use the arrow keys to navigate, or use "h" to move left, "j" to move down, "k" to move up, and "l" to move right.

6. Deleting text: To delete text, press the "Esc" key to enter command mode, then press "d" followed by "w" to delete a word, or "d" followed by "d" to delete a line.
7. Copy and paste text: Press the "Esc" key to enter command mode, then press "y" to copy a line and "p" to paste the copied text.
8. Search: Press the "Esc" key to enter command mode, then type "/text" to search for the text.
9. Undo: Press the "Esc" key to enter command mode, then type "u" to undo the last action.
10. Repeat command: Press the "." key to repeat the last command.

Note that the above examples are just a few of the basic capabilities of the vi editor, and there are many more commands and options available.