2.1.6 Linux Shell Command Facts

The Linux shell is a text user interface that provides a command line interface (CLI). Users employ the shell to interact with the Linux kernel by typing commands at the shell prompt.

This lesson covers the following topics:

- Bash shell command line prompt
- $PATH Linux shell environment variable
- Running an executable file
- Tab complete feature
- Command history feature
- Common Linux shell commands

Bash Shell Command Line Prompt

Each Linux shell type has a customizable command line prompt. For the bash shell, the default command line prompt varies with each Linux distribution, but is generally displayed in the following format:

- The username of the current user
- The @ symbol
- The hostname
- A space
- The base name of the current working directory

  > If the current working directory is the home directory (the default directory when the user first logs in), the tilde symbol (**~**) is displayed instead.

- A character that indicates the type of user.
  - The # character indicates that the current user is the root user.
  - The $ character indicates that the current user is a normal user.

$PATH Linux Shell Environment Variable

The $PATH environment variable contains the set of directories that are searched when you type an executable file at the shell prompt.

- The shell does not look in the current working directory for the executable file.
- To see the list of directories, type **echo $PATH** at the shell prompt.

- To add a directory to a path, type **PATH=$PATH:[directory_path]** and then type **export PATH**.

Running an Executable File

The following table describes how to run executable files.

File Location	Run Action
Resides in a directory that is included in the $PATH environment variable	Type the filename at the shell prompt.
Resides in the current working directory and the current directory is not included within the path environment variable	Type **./** followed by the filename.
Does not reside in the current working directory and its directory is not included within the path environment variable	Type the full path to the executable file.

File names and paths are case sensitive.

Command History Feature

The Linux shell keeps a history of commands you type at the shell prompt.

- The commands are stored as a history queue within the hidden .bash_history file in your home directory.
- Press the Up and Down keys to scroll through your previously typed commands.
    - You can edit the command or re-run it as is by pressing Enter.
- Type **history** to display the commands stored in the history queue.
- Type **history -c** to clear the history queue.

Tab Complete Feature

The Linux shell keeps a history of commands you type at the shell prompt.

- The commands are stored in the .bash_history file in your home directory.
- After typing the beginning of a command, file, or directory, press Tab to complete it.
- If pressing Tab does not fully complete the command, file, or directory, press Tab again.

- If there is no matching command, file, or directory name, your command entry will not change.
- If there is more than one matching command, file, or directory name, a list of all matches will be displayed to help you resolve your command entry.

Common Linux Shell Commands

The following table describes several common commands used at the shell prompt.

Command	Function
pwd	Shows the present working directory.
whoami	Displays the current username.
uname	Prints system information. The **uname** command has the following options: • **-a** prints all system information. • **-o** prints the operating system. • **-p** prints the processor's architecture type.

su	Switches users in the shell prompt. The **su** command has the following options:
	• **su -l [username]** switches to the specified user and creates a new login shell.
	• **su [username]** (without the dash, but with the username) switches to the user in the current shell.
	• **su - [username]** (with the dash and the username) switches to the user and loads that user's environmental variables.
	• **su -** (with the dash, but without the username) switches to the root user and loads the root user's environmental variables.
	  ◦ The root user account is the Linux system superuser.
	  ◦ The root user can perform any task; some utilities do not work if the administrator is not logged in as the root user.
	• **su** (no dash or username) switches to the root user, but does not load the root user's environmental variables.
	su requires the password of the user except when switching from root to a normal user.

exit	Exits the current shell (which may close the login shell) or to go back to the original user after using the **su** command.
exec	Executes an executable to replace the shell process with the new process created by the executable file.

cd	Changes directories. For example, when the **/usr** directory is the current directory: • **cd bin** changes to the **bin** directory in the current directory. • **cd /usr/bin** changes to the **/usr/bin** directory from anywhere in the file system.
ls	Shows names of files and directories in the current directory. The **ls** command has the following options: • **-a** shows all files and directories, including hidden files. • **-l** shows extended information about files, including size, permissions, owner, and modified date. • **-d** displays only directories. • **-s** sorts files by size. • **-X** sorts by extension. Many distributions use a color scheme to identify different file types as follows: • Directories are blue. • Text files are white. • Links are cyan. • Executable files are green. • Compressed files are red.
history	Shows all the commands in the history queue. The **-c** option clears the history list. History command queues are separate for each user. For example, a command typed as one user cannot be used after using the **su** command to switch to another user.
clear	Clears the shell screen.
chsh	Changes the default shell. The **chsh** command has the following options: • **-s** changes to a different installed shell. The command prompts for a password. • **-l** lists all installed shells. For example, **chsh -s /bin/ksh [username]** changes the default shell for the user to the Korn shell if it is installed on the computer.