Aditya Bidappa M V
ENG24CY0077
3B – 13

Linux Programming Assignment 3

1. **Distinguish between man and whatis commands? Justify with proper example. (CO1)**

The man command and whatis command serve different purposes in the Linux documentation system. The man command displays complete manual pages that contain comprehensive documentation about commands, including detailed descriptions, syntax, options, examples, and related information. When you execute man followed by a command name, it opens the full manual page in a pager where you can scroll through extensive documentation. For example, running man ls will display the complete manual for the ls command, showing all available options, detailed explanations of each flag, usage examples, and related commands. The manual page is typically several screens long and provides thorough documentation that helps users understand every aspect of the command.

In contrast, the whatis command provides only a brief one-line description of what a command does. It searches a database of short descriptions extracted from the manual pages and displays concise summaries. When you run whatis ls, it returns something like "ls - list directory contents" which gives you a quick understanding of the command's primary purpose without overwhelming detail. The whatis command is equivalent to running man -f command_name and is particularly useful when you need a quick reminder of what a command does or when you want to quickly browse through multiple commands without opening full manual pages.

2. **Use the tee command to save the output of ls -l into a file while also displaying it. (CO4)**

The tee command allows you to simultaneously display output on the terminal and save it to a file, making it extremely useful for monitoring command output while creating logs. To save the output of ls -l to a file while also displaying it on screen, you would use the command: ls -l | tee directory_listing.txt. This command pipes the long-format directory listing from ls -l to the tee command, which then writes the output to both the terminal display and the specified file simultaneously.

The tee command acts like a T-junction in plumbing, splitting the data stream so it flows to multiple destinations. When you execute this command, you will see the detailed directory listing displayed on your screen exactly as if you had run ls -l normally, but simultaneously, the exact same output is being written to the file directory_listing.txt. If the file doesn't exist, tee will create it automatically. If you want to append to an existing file instead of overwriting it, you can use the -a flag: ls -l | tee -a directory_listing.txt. This dual output capability makes tee particularly valuable for system administration tasks where you need to monitor command execution in real-time while maintaining permanent records.

3. Explain with an example how the tee command can be used in logging. (CO4)

The tee command is exceptionally valuable for logging purposes because it allows you to monitor processes in real-time while simultaneously creating permanent log files for later analysis. A practical example would be monitoring a system update process where you want to see what's happening immediately while also keeping a complete record of all activities. You could use a command like: sudo apt update && sudo apt upgrade -y 2>&1 | tee -a system_update.log. This command performs system updates while redirecting both standard output and standard error to tee, which displays everything on screen and appends it to the log file.

Another excellent logging example involves monitoring network connectivity over time. You might use: ping -c 100 google.com | tee -a network_connectivity.log to ping a server 100 times while both displaying the results in real-time and logging them for later analysis. The -a flag ensures that each time you run this command, new results are appended to the existing log file rather than overwriting previous data. For script debugging, you could use: ./myscript.sh 2>&1 | tee -a script_debug.log, which captures both the standard output and error messages from your script execution, allowing you to watch the script run while building a comprehensive log file that includes all output and any error messages that occur during execution.

4. List the steps involved in installing Ubuntu 25.04 LTS on Oracle VirtualBox. (CO2)

Installing Ubuntu 25.04 LTS on Oracle VirtualBox involves several systematic steps that ensure proper virtualization setup and operating system installation. First, download the Ubuntu 25.04 LTS ISO file from the official Ubuntu website and ensure you have Oracle

VirtualBox installed on your host system with sufficient system resources available, including at least 4GB of RAM and 25GB of storage space for the virtual machine.

Begin by opening VirtualBox and clicking the "New" button to create a new virtual machine. Provide a descriptive name for your VM, select "Linux" as the type, and choose "Ubuntu (64-bit)" as the version. Allocate appropriate memory, with a minimum of 2048 MB recommended but 4096 MB preferred for better performance. Create a virtual hard disk by selecting "Create a virtual hard disk now," choose VDI as the disk type, select "Dynamically allocated" for storage type, and set the disk size to at least 25GB. After creating the VM, access its settings and navigate to the Storage section where you'll attach the Ubuntu ISO file to the virtual optical drive. In the System settings, ensure that adequate processors are allocated, typically setting it to 2 CPUs to avoid kernel panic issues during installation.

Start the virtual machine, and it will boot from the Ubuntu ISO. Select "Install Ubuntu" from the boot menu, choose your preferred language and keyboard layout, and proceed through the installation wizard. Select "Normal installation" unless you specifically need minimal installation, choose to download updates during installation if you have a stable internet connection, and select "Erase disk and install Ubuntu" when prompted about installation type, remembering this only affects the virtual disk. Set your timezone, create a user account with a strong password, and wait for the installation process to complete. Once finished, remove the ISO from the virtual optical drive, restart the virtual machine, and complete the initial setup including any available system updates.

5. During Ubuntu OS installation, you face a Kernel Panic Error. How would you troubleshoot it? (CO3)

Kernel panic errors during Ubuntu installation in VirtualBox are relatively common issues that typically stem from inadequate virtual machine configuration or compatibility problems between the guest OS and hypervisor settings. The most frequent cause is insufficient CPU allocation in the virtual machine settings, particularly when the VM is configured with only one processor core while modern Ubuntu releases expect at least two cores for proper initialization.

The primary troubleshooting step involves powering off the virtual machine completely and accessing the VM settings through VirtualBox. Navigate to the System section and click on the Processor tab, where you should increase the processor count from one to two cores, as Ubuntu's recent kernels require multiple processor cores for stable boot processes. Additionally, in the same System settings, ensure that sufficient base memory is allocated, typically at least 2048 MB, and verify that the "Enable EFI" option is disabled unless specifically required. In the Display settings, increase the video memory to at least 128 MB and ensure 3D acceleration is disabled initially to avoid graphics-related conflicts during installation.

If the kernel panic persists after adjusting processor and memory settings, try creating a completely new virtual machine with different configuration parameters, ensuring you select the correct Ubuntu version in the OS type dropdown. Some users find success by temporarily disabling hardware virtualization features like VT-x or AMD-V in the VM settings, installing Ubuntu, and then re-enabling these features afterward. Additionally, verify that your host system has adequate resources available and that no other resource-intensive virtual machines are running simultaneously, as resource contention can trigger kernel panics during the installation process.

6. Write the command to display the system's hostname? How to change hostname using sysctl command? (CO1)

To display the current system hostname, you can use several commands depending on the level of detail required. The simplest command is hostname, which displays just the system's hostname. For more comprehensive information, you can use hostnamectl, which provides detailed information about the system's hostname configuration, including static, transient, and pretty hostnames, along with additional system information like machine ID, virtualization status, and operating system details.

Changing the hostname using the sysctl command involves modifying kernel parameters at runtime. To view the current hostname through sysctl, use the command sysctl kernel.hostname, which displays the current kernel hostname parameter. To change the hostname temporarily using sysctl, execute sudo sysctl -w kernel.hostname=newhostname, replacing "newhostname" with your desired hostname. This change affects the transient hostname and will revert after a system reboot.

For permanent hostname changes using sysctl, you need to modify the sysctl configuration file. Edit the /etc/sysctl.conf file using a text editor with sudo privileges and add the line kernel.hostname=newhostname to make the change persistent across reboots. After editing the configuration file, apply the changes immediately by running sudo sysctl -p to reload the sysctl settings. However, it's important to note that while sysctl can modify the kernel hostname parameter, the more conventional and recommended approach for permanent hostname changes in modern Linux distributions is using hostnamectl set-hostname newhostname, which properly updates all hostname-related files and configurations.

7. **Which command is used to show the calendar of the year 1984 with August month? (CO1)**

   The cal command is used to display calendars in Linux and Unix systems, and it provides flexible options for showing specific months and years. To display the calendar for August 1984, you would use the command cal 8 1984, where 8 represents the month number for August and 1984 specifies the year. This command will output a formatted calendar showing August 1984 with proper day-of-week headers and date layout.

   The cal command follows a straightforward syntax where you can specify the month as either a number (1-12) or sometimes by name, followed by the four-digit year. Alternative formats for the same result include cal August 1984 on systems that support month names, though the numeric format cal 8 1984 is more universally supported across different Unix and Linux implementations. The output displays a clean, formatted calendar with Sunday through Saturday columns and properly aligned dates for the specified month and year.

   The cal command is part of the standard Unix utilities and is available on virtually all Linux distributions and Unix systems. It's particularly useful for quick reference when you need to check what day of the week a particular date fell on in the past or will fall on in the future, making it valuable for system administrators, developers, and users who need quick calendar information without opening graphical calendar applications.

8. **Write a command to display system uptime and logged-in users together. (CO3)**

   The w command is the most comprehensive solution for displaying both system uptime and information about currently logged-in users in a single command execution. When you run w, it provides a complete overview that includes the current time, system uptime, number of logged-in users, system load averages, and detailed information about each logged-in user including their login time, idle time, and current processes. The first line of the w command output is identical to what the uptime command displays, showing how long the system has been running and the current load averages.

The w command output format includes a header line showing system uptime information followed by detailed user information in columns. The header displays the current time, uptime duration in a human-readable format like "up 5 days, 3 hours, 45 minutes," the number of users currently logged in, and load averages for the past 1, 5, and 15 minutes. Below this header, you'll see information for each logged-in user including their username, terminal type, source location, login time, idle time, JCPU time representing CPU time of all processes attached to their terminal, PCPU time for their current process, and the command line of their current process.

Alternatively, you could combine multiple commands using semicolons or pipes, such as uptime; who or uptime && who -u, but the w command provides the most elegant single-command solution. The w command essentially combines the functionality of uptime, who, and additional process information, making it the preferred choice for system administrators who need comprehensive system status information including both uptime statistics and user activity details in one convenient command.

9. Use the find command to list all ".c" files in /home/user. (CO1)

The find command is a powerful tool for searching files and directories based on various criteria, and locating C source files is a common use case for developers and system administrators. To list all files with the .c extension in the /home/user directory and its subdirectories, you would use the command find /home/user -name "*.c". This command starts searching from the /home/user directory and recursively examines all subdirectories looking for files whose names match the pattern *.c.

The find command syntax for this operation consists of several components: the starting path /home/user tells find where to begin the search, the -name option specifies that you want to search based on filename patterns, and "*.c" is the pattern that matches any filename ending with .c extension. The asterisk serves as a wildcard that matches any sequence of characters, so this pattern will find files like main.c, utils.c, program.c, and any other files with the .c extension regardless of the filename prefix.

For more specific searches, you can add additional parameters to refine the results. For example, find /home/user -type f -name "*.c" explicitly specifies that you only want regular files, excluding directories that might coincidentally have .c in their names. You can also combine this with other options like find /home/user -name "*.c" -exec ls -l {} ; to display detailed information about each found file, or find /home/user -name "*.c" -print0 | xargs

-0 wc -l to count the total lines of code in all C files. The find command provides extensive flexibility for file searching and can be combined with other commands for powerful file management operations.

10. How do you change file permissions to allow only the owner to read and write? (CO1)

To set file permissions so that only the owner can read and write while denying all access to group members and others, you use the chmod command with the octal permission value 600. The command syntax is chmod 600 filename, where filename is the name of the file you want to modify. The number 600 represents the permission bits in octal notation: 6 for the owner (read + write = 4 + 2 = 6), 0 for the group (no permissions), and 0 for others (no permissions).

The octal permission system uses three digits representing owner, group, and other permissions respectively. Each digit is calculated by adding permission values: 4 for read, 2 for write, and 1 for execute. Therefore, 600 means the owner has read and write permissions (4+2=6), while group and others have no permissions (0+0=0). When you execute chmod 600 on a file, the resulting permission string displayed by ls -l will show -rw-------, clearly indicating that only the owner has read and write access.

You can also achieve the same result using symbolic notation with the command chmod u=rw,g=,o= filename, where u represents the user/owner, g represents the group, and o represents others. The equals sign sets exact permissions, so u=rw grants read and write to the owner, while g= and o= explicitly remove all permissions from group and others respectively. This permission setting is commonly used for sensitive files like private keys, configuration files containing passwords, or personal documents that should remain completely private to the file owner.