## Lab 02: Linux Review

### Case Study

NetOpsTech, a mid-sized managed services provider, experienced increasing operational delays due to their IT staff's inconsistent Linux command-line skills. Technicians often relied heavily on graphical interfaces or struggled to perform basic system administration tasks efficiently. As a result, routine operations like file management, script editing, and process monitoring were prone to errors and took longer than necessary. This limited the team’s ability to automate tasks, quickly respond to incidents, and manage server environments effectively.

To address these inefficiencies, the company initiated a Linux proficiency review for all support engineers. Through structured hands-on labs, they revisited core Linux concepts such as file system navigation, command syntax, regular expressions, and basic system administration. This upskilling effort led to noticeable improvements in productivity, reduced human error, and better preparedness for working with DevOps tools and environments.

### Business Challenge

NetOpsTech struggled with slow incident response and inefficiencies in IT operations due to limited command-line proficiency among their junior staff. Tasks such as file manipulation, system monitoring, or process management were time-consuming and error-prone, making it difficult to scale or adopt automation tools effectively. Without a strong foundation in Linux fundamentals, the team was unprepared to handle the demands of managing cloud-native and containerized infrastructure.

To overcome these gaps, the organization onboarded you, a DevNet-certified specialist, to assess team capabilities and conduct a Linux fundamentals review to strengthen operational readiness.

### Solution

In this lab, you will revisit and reinforce your core Linux skills to ensure you're ready for advanced DevOps environments. The lab covers the following areas:

1. Review Command Syntax Navigation
2. Review File Management
3. Review Regular Expressions
4. Review System Administration

|  |
| --- |
| // Review Command Syntax Navigation  1. In this part, you will use the ls, pwd, cd, and sudo commands to review basic command syntax navigation.  2. First launch the DEVASC VM. Open a Terminal in the DEVASC-LABVM. Double-click the Terminal Emulator icon on the desktop to open a terminal window. To navigate directories, use the **ls** command to display a listing of the current directory. Remember that commands are case-sensitive.    3. Use the **ls** command with the **labs** argument to display the contents of the labs folder.    4. Use the **ls** command with the **-l** option to display a long display of the contents of the current directory.    5.Use the **ls** command with the **-r** option to display the contents of the current directory in reverse alphabetical order. Multiple options can be used at the same time. Use the ls command with both the **-l** and **-r** options to display the contents of the current directory both in long and reverse order.    6. There are many more options that can be used with the ls command. Use the **man** command with the argument **ls** to see all of the possibilities in the manual. The **man** command can be used to look up any command within the system. Use the **space bar** to advance to subsequent screens. Press **q** to quit.    7. You can also use **--help** argument after most commands to see a shorter summary of all the available command options.    8. You can use the **pwd** command to display the current working directory, **cd** command to change the directory to **/home/devasc/Documents**. **cd** command with the **/** symbol to change directories to the root directory. Use **pwd** again to see that you are now in the root directory.    9. Return to the **/home/devasc/Documents** directory. You can move one directory at a time or all the way to a destination. To quickly enter the command, type the first few letters of the directory name and press Tab for the system to automatically enter the rest of the name. Remember that names are case-sensitive.  Use the **cd** **..** characters to move up a single directory. Use **pwd** again to see you are back in the user’s home directory.    **// Use Super User Commands for Administrative Access**  1. Use the **sudo** command to issue a single command as the root user. A new terminal will not be created. Use the **sudo apt-get update** command to update to refresh the list of available packages installed on the VM. This command will not work without using the **sudo** command.  **Note:** Your output will most likely be different.    **// Review File Management**  1. To review file permissions, Use the **ls Desktop -l** to display the contents of the Desktop folder.    2. Use the command**cd** to change to the **Documents** directory. Use the command **echo** to create a shell script file, that will have the command **ls ../Desktop** inside the file. Remember that the greater than **(>)** character redirects command output to a file. The **myfile.sh** script is stored in the **/Documents** directory.    3. The **myfile.sh** script is stored in the **/Documents** directory. Use the **cat** command to view the only command in the script. This file will be used as an example to modify permissions and ownership.    4. Use the command **./myfile.sh** to run the script. Access is denied because you must set the permission of executable on the file.    5. Use the command **ls -l** myfile.sh to view the current file permissions and then use the command **chmod +x myfile.sh** to allow you to execute the file.    6. Use the command **./myfile.sh** to run the script.    7. Use the command **sudo chown root** **myfile.sh** to change the ownership of the file to **root**. Display the permissions of the **myfile.sh** file.    8. Use the command **mv** to move the **myfile.sh** file to the desktop. Use **ls ../Desktop/** to display the contents of the Desktop folder.    9. Return the file to the Documents folder. Use the command **mv** to rename **myfile.sh** to **myfile\_renamed.sh.**    10. You can use the command **cp** to make a copy of the **myfile\_renamed.sh** file to **myfile\_renamed\_and\_copied.sh** .    10. Use the **rm** command to remove the **myfile\_renamed\_and\_copied.sh** file.    11. Use the redirect **(>)** to place text into a new file called **linux.txt**. Use the command cat to redirect the contents of **linux.txt**to another file. Use the command cat to view the contents of **linux2.txt**.    12. Use the echo command to append text to the **linux2.txt**. Use the **cat** command to view the contents of the **linux2.txt** file. Use the echo command to overwrite the contents of a file using the single angle bracket.    13. Use the **echo** command to add the contents in a file using the double angle bracket.  Use the **cat** command to view the contents of the **linux.txt** file.    14. Now, to start the **vi** text editor and open a text file use command vi **linux2.txt.**    15. The text editor window opens.    **// Review Regular Expressions**  1. In this part, you use the **grep** command to review how you can use regular expressions for filtering.  **Note:** Your output may differ than the output shown below as the state of the VM is based on the most recent iteration that you downloaded as well as any changes you may have made. However, you should get some output from the passwd file but your highlighted output will differ.  2. Use the **grep** command to filter the contents of the passwd file to display the line from the passwd file containing **devasc**. Notice that the two instances of **devasc** are highlighted. Also notice that the grep command is **case-sensitive**.    2. Use the **grep** command to show how many times root appears in the passwd file. Notice that all three instances of **root** are highlighted.    3. Use the **grep** command with the anchor character **^** to find the word, but only at the beginning of the line. Notice that only the word at the beginning of the line is highlighted.    4.  Use the **grep** command with the anchor character **$** to find a word at the end of a line.    5. Use the **grep** command with the anchor character. to match specific length words with different letters in them. Notice that not only is **daem** highlighted, but also **dnsm** is highlighted.    6. Use the **grep** command to find lines where only the numbers 8 or 9 are present. Notice that only the lines containing an 8, a 9, or both are returned.    7.  Use the **grep** command to find literal characters. Notice that only the lines containing a comma are returned.    8. Use the **grep** command to find occurrences of zero or more of the pattern preceding it. Notice that only the lines with either **new** and **ne** are returned.    // **View and Test the Network Configuration**  1. You can use the command **shutdown now** to initiate a shutdown of the OS (and the VM) immediately. You do not have to perform this action as the VM will shut down and you will need to restart it manually. You can also use the command **shutdown +1 "Come back soon!"** to shut down the OS in 1 minute and display the message “Come back soon!”. Be sure to cancel or your VM will shut down.  2. Use the **ip address** command to display the network configuration. The output is a bit more detailed. For example, notice that five IPv4 addresses are shown for the **dummy0** interface.    3. To ping a computer on your local network four times use command **ping -c 4** . You must use a valid IP address of a device on your local network. The following example is using 192.168.1.1, but your network will most likely have different IPv4 addresses.    4. You can also ping a name and Domain Name System (DNS) will resolve the name to an IP address. For example, ping Cisco’s website. Your VM will send out a DNS request first to get the IP address and then send the ping packets. The DNS process is not shown in the ping output.    5. Use the ps command to display the processes that are running in the current terminal. Use the ps with the -e option to display all the processes that are running on the computer.    6. You can pipe any command output to one screen at a time by adding **| more**. One screen of output displays with the --more-- shown at the bottom. You can now use the **Enter** key to display one line at a time, the **space bar** to display one screen at a time, or Ctrl+C to exit and return to the command prompt.    7.  Use the **ps** with the **-ef** option to display all the processes that are running on the computer with more detail.    8. Use the command **apt-get update** to refresh the list of available packages in the OS, as shown previously in Part 1 of this lab. You must use administrative level permissions to use this command. Use the command **apt-cache search** to find a specific package. Use the command **apt-get install** to install a package. You can also use the **speedtest-cli** command to test your current Internet connection speed. Use the command **apt-get upgrade** to update all packages and dependencies on the computer. Use the command **apt-get purge** to completely remove a package from the computer.  9. Use the command **passwd** to update your password. Use the command **passwd** with the option **-S** to view the status of your password. |