

UNIX Course – Final Project

Project Overview

The purpose of this project is to develop a menu-driven shell script that will combine the necessary tools for a system administrator to monitor and manage their system. The application will contain a user-friendly interface that will allow the user to perform fundamental tasks such as system status, backup scheduling, network configuration, service control, user account management and file management. The script will use dynamic user input to accomplish a wide range of Unix commands from monitoring real-time CPU temperature and switching on/off certain aspects of the operating system to automating backup files. This project will allow us to practice using shell scripting while emphasizing usability and efficiency.

Project Requirement and Deliverables

The project requirement is to create a script-based program that is functional, easy to navigate and user-friendly. The menu must be responsive, meaning the user can run multiple programs without restarting the script. The script mainly uses case structure because it allows the user to select the option far more easily. The program must include the following tasks: system status, backup management, network management, service management and user management.

Task Assignment

| Members | Assigned Tasks |
|-----------------|--|
| Mustafa Hussain | Service Management and System Status |
| Steve Banh | Backup Management and User Management |
| Lucas Coveyduck | Network Management and File Management |

Project Components and Solutions

The program offers an easy and interactive menu that includes six different components utilizing UNIX tools for system administration. The main menu will contain units in the following order:

- 1) The system status unit utilizes commands such as free (for displaying memory), cat (for checking CPU temperature), top (for displaying system status), and kill (for terminating a process).
- 2) The backup management unit implements a dynamic scheduling backup file/folder and displaying previous backup files by reading user input. It uses commands such as crontab (for creating a scheduling), tar (for creating a backup file/folder), and tail (for showing the most recent backup).
- 3) The network management unit relies on the ip command for displaying and configuring interfaces, and assigning IP addresses. This unit also uses nmcli to handle Wi-Fi connections and displays the networks.
- 4) The service management unit utilizes the systemctl command to list, start, and stop system services and uses the flags `--type=services` `--state=running` to filter and display only the services that are running.
- 5) The user management unit uses user input to dynamically execute commands such as useradd, userdel, groups, and usermod, whereas session control relies on the who command for connection status and the pkill command for remote user disconnection.
- 6) The file management unit has a combination of commands like find, du, and sort to locate files and display 10 oldest and largest files from the home directory.

Project Description

1. System Status

The Menu was coded using a while loop and a case statement for making it more flexible and responsive. The Menu gives the user four options to choose .

1) Display detailed memory usage

Free -h command was used to display memory usage in human readable form.

2) Check the CPU temperature

The cat command is used to read the file that is located in /sys/class/thermal/ where the temperature data is stored.

3) List all active system process

The top command was used to display the system status because it provides a more engaging and interactive view of system processes.

4) Terminate a process

The kill command takes the PID as a parameter and stops the corresponding process.

2. Backup Management

The program contains a main menu that asks the user if they want to create a scheduled backup or check the previous backup date. The program collects user input for minutes, hours, day of the month, month, and day of the week while constantly checking the input's validity (accepting only numbers if a custom schedule is required). After, it will ask the user for the path of the file/folder that they wish to backup and for the destination directory of the backup files/folders. Once everything is complete, it will ask for a confirmation before proceeding with the scheduled backup. If the user chooses yes, the tar command will first execute creating a folder containing the backups and printing a message of the backup date into a logging file, and then the crontab command will execute thus

finalizing the action. For checking the date of the last backup, the program prompts the user to enter the path of the destination directory, and it will extract the latest content of the logging file using the `tail -n 1` command.

3. Network Management

The menu for this task was made with a select case and asks the user to select the operation they want from the following options: DisplayNetworkInterfaces, EnableOrDisableNetworkInterface, IPAddressAssignment, DisplayAvailableWiFiNetworks, and Exit. When they select DisplayNetworkInterfaces, they should see all interfaces on their system. After the interfaces, It should show the IP address, if one exists for said interface, and the default gateway if there is an IP address. EnableOrDisableNetworkInterface will give the user a secondary choice whether to enable, disable or return. Enable will ask the user for the name of the interface and will send the name they input to the `sudo ip link set "$interfaceName" up` which will enable the inputted interface. Disable will do the same thing except it will disable it (The up will be switched for down in this case). Finally return will send the user back to the Network Management menu. IPAddressAssignment will ask the user to input the interface name and the ip address they want to assign to said interface and using the command `sudo ip addr add $ipAddress dev $interfaceName` it will assign the IP to the appropriate interface. DisplayAvailableWiFiNetworks will use the NetworkManagement Console-Line interface to show all available networks around and prompt them to say the network name and the password for said network before implementing them into `nmcli device wifi connect $network password $passwordInput`

4. Service Management

The menu was coded using a while loop and a case statement to make it more flexible and responsive. The menu gives the user access to the 3 executable programs and lets the user go to the main menu.

1) Display a list of currently running services

The command `Systemctl list-units` was used to list all the units in the system.

To filter only the services, the flag `-type=service` was added, and for displaying only the running services the flag `-state=running` was used.

2) Start service

The command `sudo systemctl start` was used to start a service

3) Stop Service

The command `sudo systemctl stop` was used to stop the service. However, if the service is critical for the system, it might not terminate the service.

5. User Management

The user management provides a main menu prompting the user to choose the action that they want to perform which includes creating a new user account, grant root privileges, delete a user, display currently connected user, disconnect a user remotely, show a user's groups, and change a user's group membership. The following explains the functionality of each action:

- 1) The program can collect user input for username and password (using `read -s` for hiding input). After, it will ask the user to confirm their action to which `sudo useradd -m` and `sudo chpasswd` commands are used to finalize the account and home directory and password set up.
- 2) The program prompts the user to enter the username of whom they want to grant root privileges and check if the user exists. Then, the user will be asked to confirm their action and `sudo usermod -aG sudo` will be used to finalize the action.
- 3) The program prompts the system administrator to specify the username of the user they wish to delete and checks if they exist. The program will then check if the user is currently connected. If they are still connected, the program will ask the administrator to disconnect the user before proceeding. Then, a confirmation message will show up. If the administrator proceeds, `sudo userdel -r` is used to delete the user and `getent group` is used to check if the user has any group privileges and delete those as well by using `sudo groupdel`.

- 4) The who command is used to show the current connected user.
- 5) The program asks for the username to disconnect. It will start by checking if the user exists. Then, it will ask the administrator to confirm if they want to disconnect the specified user's account. If yes, the program will proceed with the command `sudo pkill -KILL -u` and show a message saying the user has been successfully disconnected.
- 6) The program prompts the system administrator to enter the username as to display which groups the user is currently part of by using the command `groups`
- 7) The program asks for the username that the administrator wishes to change group membership for and checks if the user exists before proceeding. Then, it will ask for the group name and `sudo usermod -g` is used to finalize the group membership modification.

6. File Management

File Management's menu is made with another select case. The choices given from the select case are FindFile, LargestTen, OldestTen, and Exit. When FindFile is selected, it prompts the user to enter the username and the file in order to search for the file within that user's home directory. If the user inputs a user that doesn't exist, it returns that the user does not exist. If the file doesn't exist, then it'll tell the user that, However, if the user and the file exist, then it shall print the full file path to the file. If the user selects LargestTen, the program will print out the 10 largest files in your home directory by taking your home directory (\$HOME), using the `du` command to get the files and making the size human readable. Then, it sorts it in descending order before cutting the first 10 lines and printing them. OldestTen does almost the same thing. It uses `find` to search through all the files in the home directory before format printing the time they were made. After it sorts and cuts the

lines, similar to LargestTen before using awk to take the format printed time to show only date and time.