UNIX Final Project Documentation

# Project Overview

Our final project that we developed for our UNIX course is designed to provide system administrators with an efficient and user-friendly tool for system monitoring and management. From display detailed memory usage information to displaying the 10 oldest files in the user’s home directory, our shell script provides an easy to navigate, clear and menu-drive user interface allowing the user to easily go from one task to another depending on their needs. The overall objective is to facilitate the management of your system through the terminal.

# Project Requirements and Deliverables

## System Status

For the first section titled “System Status” the script has many designed outputs depending on the user’s selection.

For the first, second and third tasks, if the user wishes to display detailed memory usage information of their machine, CPU temperature and all active system processes, the expected output will be a visual representation of their memory usage, CPU temperature in Celsius degrees (this will also trigger an alert if above 70 degrees Celsius), and finally each active system process along with its start time, PID, CPU percentage and more.

The fourth and final task of this session was to stop or terminate a specific process. The expected output is a message asking the user to enter the PID of the process they wish to stop. Once entered it will display a confirmation message.

## Backup Management

Our second section of our application titled “Backup Management” is a smaller, but strong and efficient part of our script.

The first task of this section was to create a backup schedule based on the date, file name and destination that the user wishes to input. So, the expected output is for there to be several messages prompting you to enter each of these details along with a confirmation message.

The final task was to display the date and time of the last completed backup, which will display only if you have scheduled a back up in the past. The expected output is the date and time displayed of the last backup.

## Network Management

The third section of our application titled “Network Management” has five main designed outputs depending on the user’s selection which are also very important to our application.

The first task was an option to display all network interfaces which will display a list of each network interface along with its status, IP address and at the bottom of the page, the default gateways.

The second and third tasks consisted of enabling and disabling one of these interfaces where the expected output is a message prompting the user to enter the interface they wish to enable or disable. Once completed it will display a message confirming the result.

The fourth task was to be able to assign an IP address to a network card. The designed output consists of a message prompting the user to enter an interface and then once that is completed the user will be asked to enter the IP address (along with the CDIR) to assign to said interface.

The fifth and final task was to display all available Wi-Fi networks, allowing the user to connect to one. The output is a list of all available networks along with a message prompting the user to enter the SSID of the network they wish to connect to. After that they will be asked to enter the password for said Wi-Fi network in order to connect to it.

## Service Management

The fourth section of our application titled “Service Management” has fewer tasks and designed outputs, but is still an essential part of our application.

The first task was to display all running services on your machine. When selected the user will see the output of the name of the service, its description and status (which will always be “running”)

The second and third tasks consist of the user being able to start or stop a selected service. The user will be prompted with a message asking for the name of the service. Once given the script will print out a confirmation message of the users desired selection.

## User Management

The fifth section of our application titled “User Management” is one of the most important sections of our application with many designed tasks and outputs.

The first task was to be able to create a user, this task prompts the user with a message to enter the desired username and then the desired password.

The second task which is similar in output was to grant root privileges to a user. The output being a message which asks the user to specify which user they wish to grant root privileges.

The third task was to delete a user account, which again is very similar output as it prompts a message asking you which user to remove. After this it will display a confirmation message confirming the user has been removed.

Task number 4 was to display a list of all currently connected users. The output is very simple which is just a list of each user connected and the time they connected on.

The fifth task was to disconnect a remote user. The expected output is a message asking you for the user you wish to disconnect, and then a confirmation message.

The sixth task was to show all groups of a specified user which asks you to input the user and then displays a list of every single group they are apart of.

The seventh and eighth tasks consist of adding and removing a user from a specified group. The output is a message asking you for the user, then another message asking for the group you wish to add them to. Once both entered you will receive a confirmation message.

## File Management

The sixth and final section is a smaller one compared to the last, but is still a robust and required addition to our application.

The first task consists of finding a specified file based on a username and file name as input. This task verifies that the username exists by prompting you to enter the username and file name. Then searches the user’s home directory for the specified file and if found it display its full path, but if not, it shows an error message.

The second and third tasks consist of displaying either the 10 oldest or 10 largest files in a user’s home directory. The designed output first asks you to enter a username, and then displays a list of the files (oldest or largest) that includes its permissions, timestamps and the full path.

# Task Assignment

|  |  |  |  |
| --- | --- | --- | --- |
| Student Name | Mohammad Adyan Khan | James Luciano | Nicholas Hu |
| Tasks Done | 1. System Status 2. Backup Management | 1. Network Management 2. Service Management | 1. User Management 2. File Management |

# Project Components and Solutions

## System Status

### Implemented Features and Solutions

1. Menu System
   1. Uses case to offer options
2. Memory Usage Monitoring
   1. Command used: free -h
   2. This displays memory consumption in a human-readable format
3. CPU Temperature Check
   1. Command used: command -v sensors (from lm-sensors package)
   2. Captures temperature using awk: sensors | awk ‘/Tctl/ {print $2}’
   3. If main field is missing it falls back to sensors | awk ‘/Core 0:/ {print $3}’
   4. Numeric filtering: cpu\_num=${cpu\_temp//[^0-9.]/}
4. Display System Processes
   1. Command used: ps aux
   2. Shows all running processes with user
5. Terminate a Process
   1. Commands used: ps aux | grep “$pid” and sudo kill “$pid”
   2. User is prompted for the PID they want to terminate

## Backup Management

### Implemented Features and Solutions

1. Menu System
   1. Uses case to offer options
2. User Input for Backup Parameters
   1. Scripts asks user for filename, day of week (1-7), time in HH:MM format, and destination path
3. Parsing the Time
   1. Command used: minute=$(echo “$time” | cut -d”:” -f2)
   2. Command used: hour=$(echo “$time” | cut -d”:” -f1)
4. Creating Cron Job Entry
   1. Command: cron\_tab=”$minute $hour \* \* $day cp \”file\” \”$dest\””
   2. Appends it: (crontab -1 2>/dev/null; echo “$cron\_tab”) | crontab –
5. Schedule Backup Execution
   1. The schedule cron job will run weekly at the specified time
   2. Uses cp command to copy the selected file to user specified destination

## Network Management

### Implemented Features and Solutions

1. Menu System
   1. Uses select and case to offer options
2. Display All Network Interfaces
   1. Uses ip commands to retrieve network information
   2. Command used: ip -br addr: displays all interfaces and their IP addresses
   3. Command used; ip route show default
3. Enable/Disable Network Interface
   1. Command: sudo ip link set “iface” up
   2. Command: sudo ip link set “iface” down
4. Assign IP to Interface
   1. Command: sudo ip addr flush dev “$iface”: Removes old Ips
   2. Command: sudo ip addr add “$ipaddr” dev “$iface”: Assigns new IP
   3. Sudo ip link set “$iface” up: Sets interface up
5. Display and Connect to Wi-Fi Networks
   1. Command: sudo nmcli device wifi list: Lists available Wi-Fi networks
   2. Sudo nmcli device wifi connect “$ssid” password “$password”: Connects

## Service Management

### Implemented Features and Solutions

1. Menu System
   1. Uses select and case to offer options
2. Service Monitoring
   1. Command: systemctl –no-page list-units --type=service –state=running
   2. Lists all currently running services and doesn’t close off terminal as page
3. Service Control
   1. Command: sudo systemctl start “$servicename”: Starts the service
   2. Command: sudo systemctl stop “$servicename”: Stops the service

## User Management

### Implemented Features and Solutions

1. Menu System
   1. Use select and case to offer options
2. User Management Functions
   1. Creating user: useradd, chpasswd to create username and password
   2. Grant root privileges using usermod -aG “$username”
   3. Delete a user: deluser --remove-home
3. Group Management Functions
   1. Adding a user: usermond -aG “$group” “$username”
   2. Remove a user: gpasswd -d “$username” “$group”
   3. Show groups of a specific user: groups “$username”
4. Session and Process Management
   1. Display connected users using the “who” command
   2. Disconnecting a user: pkill -kill -u “$username”

## File Management

### Implemented Features and Solutions

1. Menu System
   1. Uses select and case to offer options
2. File Search Function
   1. Searches in user’s home directory recursively for a filename specified
   2. Uses id “$username” which verifies if the user exists
3. File Size Analysis
   1. find “$home\_dir” -type f -exec ls -lh {} + 2>/dev/null | sort -k5 -h | tail -n 10
   2. Command used to find 10 largest files ^
4. File Age Analysis
   1. find “$home\_dir” -type f -printf “%T+ %p\n” | sort | head -n 10
   2. Command used to find 10 oldest files ^

# Detailed Descriptions

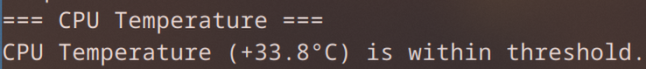
## System Status

The purpose of this section is to monitor the current system state which includes memory usage, CPU, temperature and active processes. Also allowing you to terminate processes. By checking memory and your CPU load, you can detect potential overheating or blockages which would lead to potential system failures and instabilities. As system admins your job is to ensure that the system remains responsive and healthy with this section.

Details:

Memory Monitoring: The command free -h shows memory usage in human-readable format.

CPU Temperature: sensors monitors CPU temperature and alerts if exceeding threshold.



Process Monitoring: ps aux lists all running processes.

Terminate Process: sudo kill allows you to terminate unwanted processes.

## Backup Management

The purpose of this section it to automate file backups using cron to prevent data loss. Backing up your files is an essential part of keeping your data and important/sensitive files securely stored on your machine. As a system administrator, it is very important to ensure that you are able to schedule and manage backups for your files/data.

Creating a cron job: (crontab -l 2>/dev/null; echo "MIN HOUR \* \* DAY cp 'FILE' 'DEST'") | crontab

This is what automates scheduled backups.

A computer screen shot of a computer screen

AI-generated content may be incorrect.

## Network Management

The purpose of the network management section is to provide a bundle of tools which help you manage your network interfaces, IP addresses and Wif-Fi networks. No matter what you need from this section whether it be to display, configure or enable/disable, everything is organized and tailored to help manage your network settings. Keeping track of your network and connection status is essential to ensuring your system and data is secure and not vulnerable to attacks.

Wi-Fi Management: nmcli device wifi list

A screenshot of a computer

AI-generated content may be incorrect.

Example of enabling/disabling an interface (sudo ip link set “$iface” up/down

A screenshot of a computer

AI-generated content may be incorrect.

## Service Management

The purpose of this section is allowing the users to monitor and control their system services through an interactive menu system. Some features include listing all currently running services, stopping one of these services, or even reenabling one of them. This section is essential for maintenance, securing and troubleshooting your system.

Example of listing all services using systemctl --no-pager list-units -- type=service --state=running

A screenshot of a computer

AI-generated content may be incorrect.

## User Management

Section user management’s purpose is to allow the user to be able to perform all the necessary tasks related to the users and groups on their machine. Some of these functions include adding a u ser, deleting a user, granting root privileges, monitoring logged-in users and so much more. By using commands such as useradd and pkill, this section ensures that user management on your system is secure and consistent.

Example of viewing all groups of a user using: groups $”username”

A screenshot of a computer

AI-generated content may be incorrect.

## File Management

The purpose of the section titled “file management” is to focus on helping the user manage and locate files on their system. Some features include searching for specific files by name, listing the oldest files and listing the largest on your machine. This is incredibly useful for cleaning up unnecessary files to free up storage and identifying important directories and files on your computer.

An example of finding a file using the command: find “$home\_dir” -name “$filename” 2>/dev/null

A screenshot of a computer

AI-generated content may be incorrect.

Listing 10 largest files: find “$home\_dir” -type f ls -lh {} + 2?/dev/null | sort -k5 -h | tail -n

A screenshot of a computer

AI-generated content may be incorrect.