Experiment 7: Shell Scripting Part III (Automation)

1. Aim:

      a. To automate system admin tasks using shell scripts.

2. Requirements: Linux OS

3. Related Theory:

Shell script is a text file with a list of commands. These scripts are used to instruct the operating system or system-level process to execute and obtain the desired result. All commands will be executed line by line by checking conditional statements, executing built-in functions, performing some logic operations, changing the system-level variables, checking the processes, memory usage by each process, etc.

Shell Scripting is used for multiple system level operations like 'system administration's tasks', 'for creating, maintaining and implementing system boot scripts', 'automating tedious repetitive tasks', 'scheduling and executing system tasks', 'for automating the installation process for new software or for new software updates across the organization', 'for scheduling data backup process', 'etc.

Shell Scripts can be bifurcated into four major areas:

1. Used by System Administrator
System administrator uses shell scripts to automate the repetitive tasks, account creation for different users, monitoring the system status, usage of memory, running process on a system, kill or terminate multiple processes etc.

2. Data Backup
Shell scripts can be used to automate the data backup process by creating a cron job and schedule the shell script to execute data backup commands and perform the task in an easy way. When we want to use the backup of a particular day, data archiving, restoring the system to a particular date can be done by using shell scripts, we can monitor the system status while doing backup operation, restore operation.

3. Programming
Shell scripts can be used by programmers to develop programs, scripts to perform system-level tasks, finding patterns in a file, processing the data in a file based on the user inputs, etc. can be done using shell scripts.

4. Automation
Shell scripts can be used to automate several daily tasks, repetitive tasks, etc. We can automate the process of installing the pre-requisite packages on a system before running an application or set up a fresh system with the new operating system and then install all required packages so a user can use it for his tasks.

4. Laboratory Exercise:

Write shell scripts to perform following tasks.

    i.   Delete files
   ii.   Check machine status
  iii.   Display formatting
  iv.   User management
   v.   Package management

5. Post-Experiment Exercise:

A. Conclusion:
#Summarize your experience about the skills acquired from this experiment.

B. Tasks:

1. Write a shell script to perform the following user management tasks.

    a.   Create a menu as shown below

           i.   Create a user (Your name)
          ii.   Change the group for a user
         iii.   Create a group
         iv.   Check password status
          v.   Change Password
         vi.   Lock user account
        vii.   Unlock user account
       viii.   Delete user

Script must execute the task based on the choice selected from menu.

2. Write a shell script to perform the following package management tasks.

   a. Create a menu as shown below

      i. yum commands/options

      ii. Available packages

      iii. Installed packages

      iv. Package Information

      v. Find package

      vi. Check yum history

      vii. Install package

      viii. Update package

      ix. Upgrade package

      x. Clean cache

    Script must execute the task based on the choice selected from menu.

Submission Instructions:

For each question, students need to submit the question, screenshot of the scripts and outputs.

The screenshots must cover all possible options in the menu. In case the output is big, take partial screenshot.