**ASSIGNMENT 1. DUE DATE 22/FEB/2024**

**Question 1**

1. Discuss the Unix commands and tools available for controlling and monitoring processes. Explain how to start a process in the background, stop a running process, and check the resource utilization of a specific process.
2. Provide examples of how you might use commands like ps, top, and kill to manage and monitor processes

**Question 2**

You are tasked with automating a file management system using shell scripting. (20 Marks)

1. Write a shell script that scans a specified directory for files older than 30 days and moves them to an "archive" directory. Ensure the script logs the moved files and their timestamps.
2. Extend the script to encrypt all files in the "archive" directory using the gpg command. Allow users to decrypt files on demand. Include error handling and user prompts for password input during encryption and decryption processes.

**Question 3**

You have recently joined a team responsible for maintaining a critical Unix cluster used by a financial institution. One of the nodes in this cluster has experienced intermittent failures, causing disruptions to the organization's operations. Answer the following questions:

1. Which Unix tools would you utilize to identify the root cause of the node failure? Explain how you would use these tools to pinpoint the issue within the complex environment of the Unix cluster.
2. How would you configure High Availability (HA) settings to ensure minimal downtime during maintenance or unexpected failures? Provide examples of HA configurations using relevant Unix tools and concepts.
3. Describe the steps involved in setting up automated backups for the entire Unix cluster. Include details about backup storage options and data recovery procedures.
4. Design a strategy for load balancing across multiple nodes in the Unix cluster. Discuss the advantages and disadvantages of various load balancing algorithms and explain your chosen approach.

**ASSIGNMENT 2. DUE DATE 1ST MARCH 2024**

**Question 1**

Write a shell script that automates the process of regular backups for critical system files. The script should create a compressed archive of specified directories and store them with a timestamp in a designated backup directory.

**Question 2**

Consider a scenario where you are tasked with managing a Unix-based server for a web hosting company. The server hosts multiple websites, each with its own directory and user account.

**1. File System Organization**

1. Explain the importance of a well-organized file system on a Unix server. Discuss how the hierarchical file system structure contributes to efficient management. Provide examples of directories you would create for the websites on the server.
2. You are tasked with listing the contents of a directory that contains various file types, including text files, images, and scripts. Describe the ls command options you would use to display detailed information about each file, including permissions, ownership, and file size. Provide an example command.

**2. Shell Scripting and User Management**

1. Write a simple shell script that accomplishes the following tasks:

Creates a new user account named "webuser" with the home directory set to "/var/www/webuser."

Creates a new directory named "website" within the user's home directory.

Copies a sample HTML file, "index.html," into the "website" directory.

1. Explain how you would modify the script to ensure that the "webuser" account has read and execute permissions on the "website" directory, while other users have no access. Provide the necessary command or commands to achieve this.

**Question 3**

XYZ Corporation operates a large Unix-based server environment that hosts critical applications. The system administration team is seeking to streamline routine tasks and improve system efficiency through the implementation of custom shell scripts. As a Unix OS and Programming specialist in the team, your task is to design and implement scripts for various system administration activities.

Part A: System Monitoring and Maintenance

1. *Disk Space Monitoring:* a. Develop a shell script named disk\_monitor.sh that checks disk space usage on all mounted filesystems. The script should generate an alert if any filesystem has less than 15% free space.
2. *Log Rotation:* a. Create a shell script named log\_rotation.sh to automatically rotate and compress log files in the /var/log directory. Ensure that log files older than 30 days are archived, and the script should run as a scheduled task.

Part B: User Management

1. *User Account Provisioning:* a. Design a shell script named user\_provisioning.sh that takes user details as input (username, full name, and password) and creates user accounts with appropriate permissions.
2. *Account Locking:* a. Develop a script named lock\_accounts.sh that reads a list of usernames from a file named locked\_users.txt and locks those user accounts, preventing login access.

Part C: Backup and Recovery

1. *Automated Backup:* a. Create a shell script named automated\_backup.sh that performs a full system backup, including critical configuration files and user data. The script should use a rotating backup strategy and retain backups for the last two weeks.
2. *Disaster Recovery Plan:* a. Outline a basic disaster recovery plan that includes steps to restore the system using the backups created by the automated\_backup.sh script in the event of a system failure.