Linux System Administration Project

Shell Scripting Project Instructions

Project Overview: You are tasked with creating shell scripts to accomplish 5 specific tasks from the allocated project list. This project is designed to enhance your skills in shell scripting by applying practical solutions to real-world scenarios.

Tasks Allocation:

1. **Select 5 Tasks**: Choose any 5 tasks from the project list provided. Each task should be distinct and demonstrate your understanding and ability to use shell scripting effectively.

Project Requirements:

- **Script Functionality**: Each script should perform a specific task as outlined in the project list.
- **Documentation**: Include comments within your script to explain the purpose of each major step or function.
- **Error Handling**: Implement error handling where appropriate to manage unexpected scenarios.
- **Output**: Scripts should provide clear and readable output to indicate successful execution of tasks.
- **Testing**: Test each script thoroughly on a Linux environment to ensure functionality and correctness.

Project Submission:

- **Deadline**: Scripts must be completed within 5 days from the assignment date.
- **Submission Format**: Submit your scripts using the provided Google Form. Ensure you provide your name, email, and script files in a zip archive or individually.
- **Naming Convention**: Name your scripts descriptively, indicating the task number and a brief title (e.g., `task1_file_count.sh`, `task2_process_monitor.sh`).

Evaluation Criteria:

- **Functionality**: Does the script achieve the intended task effectively?
- **Script Clarity**: Are scripts well-structured and easy to follow?
- **Documentation**: Are comments included to explain complex sections of code?
- **Error Handling**: How well does the script handle unexpected errors or inputs?
- **Creativity**: Did you approach the tasks with creative and efficient solutions?

Additional Notes:

- Use resources such as online documentation, tutorials, and practice exercises to enhance your understanding of shell scripting concepts.
- Reach out for assistance or clarification if you encounter challenges during the project.

Good luck with your project!

Task 1:

Task Description

A batch of 100 fresh engineers is joining the company "MyComp". As a system administrator, you need to create 100 user accounts with the following specifications:

1. **Username and Home Directory**:
    - Each username should be in the format `mycompusr1`, `mycompusr2`, `mycompusr3`, ..., up to `mycompusr100`.
    - Each user should have a home directory created in the format `/home/mycompusr1`, `/home/mycompusr2`, ..., `/home/mycompusr100`.
2. **Password**:
    - The password for each account should be the same as the username. For example, the password for `mycompusr1` should be `mycompusr1`, and so on.
    - The passwords should be set to expire every month.
3. **Permissions**:
    - Each home directory should have permissions set to `700`, allowing only the respective user to read, write, and execute.
4. **Group Membership**:
    - Each user should be added to the `wheel` group to grant them administrative privileges.
5. **Password Expiration Notification**:
    - Implement a mechanism to notify users of their password expiration.

Task 2:

Task Description

As a system administrator at "MyComp," you are responsible for monitoring and maintaining the health of the company's servers. To ensure that disk usage does not exceed critical levels and to keep an eye on storage consumption trends, you need to

generate regular disk usage reports. The reports should cover all mounted filesystems and be emailed to you daily.

Requirements

1. **Generate Disk Usage Report:**
    a. **The script should collect disk usage information for all mounted filesystems.**
    b. **The report should include details such as filesystem name, total size, used space, available space, and usage percentage.**
2. **Email the Report:**
    a. **The report should be emailed to the system administrator (e.g., `admin@mycomp.com`) every day.**
    b. **The email should have a clear subject line and body, making it easy to identify and read the report.**
3. **Automate the Process:**
    a. **The script should be set up to run daily using a cron job.**

Task 3:

Task Description

As a system administrator at "MyComp," you are tasked with ensuring the integrity and safety of critical data on the company's servers. To safeguard this data, you need to develop a backup strategy that includes regular backups of specified directories. These backups should be compressed and stored in a designated backup location. Additionally, the process should be automated using cron jobs to ensure regular, unattended execution.

Requirements

4. **Directories to Back Up:**
    a. **Specify a list of directories that need to be backed up regularly. For example, `/home`, `/etc`, and `/var/www`.**
5. **Backup Location:**
    a. **Choose a designated location where the backups will be stored. For example, `/backup`.**
6. **Backup Method:**
    a. **Use the `tar` command to create compressed archives of the specified directories.**
7. **Scheduling:**
    a. **Set up a cron job to automate the backup process to run at specified intervals (e.g., daily, weekly).**
8. **Retention Policy:**

    a. **Implement a retention policy to manage the storage of backups, such as keeping backups for the last 7 days.**
9. **Notification:**
    a. **Send a notification email to the system administrator once the backup is complete.**

TASK 4:

Task Description

As a system administrator at "MyComp," you are responsible for monitoring the disk usage across all servers to ensure efficient use of storage resources and to anticipate any potential issues that may arise. To facilitate this, you need to develop a script that generates a comprehensive disk usage report for all mounted filesystems. This report should be automatically emailed to you on a regular basis.

Requirements

1. **Generate Disk Usage Report**:
    - The script should gather disk usage information for all mounted filesystems on the server.
    - Include details such as filesystem name, total size, used space, available space, and usage percentage in the report.
2. **Email the Report**:
    - Email the generated report to the system administrator (e.g., `admin@mycomp.com`).
    - The email should have a clear subject line and body, making it easy to identify and read the report.
3. **Automation**:
    - Set up the script to run automatically at regular intervals using cron jobs (e.g., daily, weekly).

10.

Task Description

As a system administrator at "MyComp," you are responsible for ensuring data integrity and disaster recovery preparedness by implementing regular backups of critical directories on company servers. To accomplish this, you need to develop a backup script that automates the process of backing up specified directories to a designated backup location. The script should utilize `tar` for creating compressed archives and should be scheduled using cron jobs for regular execution.

Requirements

1. **Directories to Back Up**:
    - Specify a list of directories that need to be backed up regularly. For example, `/home`, `/etc`, and `/var/www`.
2. **Backup Location**:
    - Choose a designated location where the backups will be stored. For example, `/backup`.
3. **Backup Method**:
    - Use the `tar` command to create compressed archives (`tar.gz`) of the specified directories.
4. **Scheduling**:
    - Set up a cron job to automate the backup process to run at specified intervals (e.g., daily, weekly).
5. **Retention Policy**:
    - Implement a retention policy to manage the storage of backups, such as keeping backups for the last 7 days.
6. **Notification**:
    - Optional: Send a notification email to the system administrator once the backup is complete.

Task 6:

Task Description

As a system administrator at "MyComp," you are responsible for ensuring that critical services on company servers are always running. Services such as `httpd` (web server) and `sshd` (SSH daemon) are essential for the company's operations. To ensure these services are running, you need to develop a monitoring script. This script should check if a particular service is running, and if it is not, the script should start the service and send a notification email to the system administrator.

Requirements

1. **Service Monitoring**:
    - The script should check if a specified service (e.g., `httpd`, `sshd`) is running.
2. **Restart Service**:
    - If the service is not running, the script should start the service.
3. **Notification**:
    - The script should send a notification email to the system administrator if the service was found stopped and restarted.
4. **Automation**:
    - The script should be set up to run at regular intervals using cron jobs (e.g., every 5 minutes).

Task 7:

Task Description

As a system administrator at "MyComp," you are responsible for keeping the servers up to date with the latest security patches and software updates. To streamline this process, you need to develop a script that automatically checks for and installs updates using `yum` or `dnf`, depending on the Linux distribution in use. The script should also log the results of each update operation, including details such as the date and time of the update, packages that were updated, and any errors encountered.

Requirements

1. **Check for and Install Updates**:
    - The script should check for available updates using `yum` or `dnf`.
    - It should automatically install all available updates.
2. **Logging**:
    - The script should log the results of each update operation to a log file.
    - The log should include the date and time of the update, a list of packages that were updated, and any errors that occurred during the update process.
3. **Automation**:
    - The script should be set up to run at regular intervals using cron jobs (e.g., daily).

Task 8:

Task Description

As a system administrator at "MyComp," you are responsible for managing the log files generated by various applications running on the company's servers. To prevent log files from consuming excessive disk space and to maintain an organized log directory, you need to develop a script for log rotation. This script should rotate the log files for a specified application by compressing old logs and deleting logs older than a specified number of days.

Requirements

1. **Log Rotation**:
    - The script should rotate the current log file by renaming it with a timestamp or incremented number.
    - The current log file should be truncated or replaced with a new empty log file.
2. **Compression**:
    - The rotated log files should be compressed to save disk space.
3. **Deletion of Old Logs**:
    - The script should delete log files older than a specified number of days.
4. **Automation**:

- The script should be set up to run at regular intervals using cron jobs (e.g., daily or weekly).

Task 9:

Task Description

As a system administrator at "MyComp," you need to frequently gather and review detailed system information to monitor server health, diagnose issues, and perform routine checks. To streamline this process, you need to develop a script that displays comprehensive system information. This script should include details about the CPU, memory, disk usage, network configuration, and running processes.

Requirements

1. **CPU Information**:
   - Display detailed information about the CPU, including model, speed, and usage statistics.
2. **Memory Information**:
   - Show total, used, and available memory.
3. **Disk Usage**:
   - Display disk usage for all mounted filesystems.
4. **Network Configuration**:
   - Show network interfaces and their configurations, including IP addresses, subnet masks, and MAC addresses.
5. **Running Processes**:
   - List the currently running processes along with their CPU and memory usage.
6. **Readable Output**:
   - Present the information in a clear and readable format.

Task 10:

Task Description

As a system administrator at "MyComp," you are tasked with ensuring the security and integrity of critical system files and directories. Unauthorized changes to file permissions can pose significant security risks. To mitigate this, you need to develop a script that audits file permissions on critical system files and directories, compares them against expected permissions, and generates a report highlighting any deviations. This report will help in identifying and addressing potential security issues promptly.

Requirements

1. **Specify Critical Files and Directories**:

    - Identify a list of critical system files and directories whose permissions need to be audited.
2. **Expected Permissions**:
    - Define the expected permissions for each of these files and directories.
3. **Audit Permissions**:
    - Compare the current permissions of the specified files and directories with the expected permissions.
4. **Generate Report**:
    - Generate a report highlighting any deviations from the expected permissions.
5. **Readable Output**:
    - Present the audit results in a clear and readable format.
6. **Automation**:
    - Optionally, set up the script to run at regular intervals using cron jobs (e.g., daily or weekly).

Task 11:

Task Description

As a system administrator at "MyComp," you are responsible for managing cron jobs for various users on the server. To simplify and streamline the management of these scheduled tasks, you need to develop a script that allows you to list, add, or remove cron jobs for specified users. The script should ensure that cron jobs are properly formatted and that permissions are correctly set to prevent unauthorized modifications.

Requirements

1. **List Cron Jobs**:
    - Display all cron jobs for a specified user.
2. **Add Cron Jobs**:
    - Add a new cron job for a specified user, ensuring proper formatting.
3. **Remove Cron Jobs**:
    - Remove a specific cron job for a specified user, ensuring correct permissions.
4. **Permissions and Formatting**:
    - Ensure that cron jobs are added or removed with proper formatting and appropriate permissions.
5. **User Input and Validation**:
    - Provide user-friendly prompts for input and validate the input to prevent errors.

Task 12:

Task Description

As a system administrator at "MyComp," you need to configure the network interfaces on various servers. This involves setting static IP addresses, configuring network interfaces, and

applying DNS settings to ensure stable and secure network connectivity. To streamline this process, you need to develop a script that automates the configuration of network interfaces, sets static IP addresses, and applies DNS settings.

Requirements

1. **Configure Network Interfaces**:
    - Allow the configuration of network interfaces (e.g., `eth0`, `eth1`).
2. **Set Static IP Addresses**:
    - Set static IP addresses, subnet masks, and gateways for specified network interfaces.
3. **Apply DNS Settings**:
    - Configure DNS servers by updating the `/etc/resolv.conf` file or other relevant configuration files.
4. **Input Validation**:
    - Validate user inputs to ensure correct network configuration.
5. **Apply Configuration**:
    - Apply the network configuration changes.

Task 13:

Task Description

As a system administrator at "MyComp," you are responsible for managing the firewall rules to ensure the security of your servers. This involves configuring and managing firewall rules using `firewalld` or `iptables`, including opening and closing specific ports. To facilitate this process, you need to develop a script that automates the configuration of firewall rules, allowing you to easily open or close ports and apply these changes effectively.

Requirements

1. **Open Ports**:
    - Allow the script to open specific ports for both `firewalld` and `iptables`.
2. **Close Ports**:
    - Allow the script to close specific ports for both `firewalld` and `iptables`.
3. **Display Current Rules**:
    - Provide an option to list current firewall rules.
4. **Input Validation**:
    - Validate user inputs to ensure correct port numbers and protocols.
5. **Apply Configuration**:
    - Apply the firewall rule changes effectively.
6. **User-Friendly Interface**:
    - Provide clear prompts and feedback to the user.

Task 14:

Task Description

As a system administrator at "MyComp," you are responsible for keeping track of user activity on the system, including login and logout times. To effectively monitor this activity and generate useful insights, you need to develop a script that records user activity and generates daily and weekly reports. These reports should include details about user login/logout times and durations of sessions.

Requirements

1. **Monitor User Activity**:
    - Track user login and logout times.
    - Record the duration of user sessions.
2. **Generate Daily Reports**:
    - Create a report summarizing user activity for the current day.
3. **Generate Weekly Reports**:
    - Create a report summarizing user activity for the current week.
4. **Log Management**:
    - Maintain logs of user activity.
    - Archive logs older than a certain period.
5. **User-Friendly Output**:
    - Provide clear and readable reports.

Task 15:

Task Description

As a system administrator at "MyComp," you are responsible for managing storage on the servers. This involves creating, mounting, and managing logical volumes and filesystems. You also need to handle resizing of volumes and managing snapshots for backups. To streamline these tasks, you need to develop a script that automates the creation, mounting, resizing, and snapshot management of logical volumes and filesystems.

Requirements

1. **Create Logical Volumes**:
    - Allow the script to create logical volumes from specified volume groups.
2. **Create Filesystems**:
    - Allow the script to create filesystems on logical volumes.
3. **Mount Filesystems**:
    - Mount the created filesystems at specified mount points.
4. **Resize Logical Volumes and Filesystems**:
    - Allow the script to resize logical volumes and their filesystems.
5. **Snapshot Management**:
    - Create, list, and remove snapshots of logical volumes for backup purposes.
6. **Input Validation**:

- Validate user inputs to ensure correct volume groups, logical volumes, and filesystems.

Task 16:

Task Description

As a system administrator at "MyComp," you are responsible for monitoring the status of critical services running on your servers. This includes services such as `httpd`, `sshd`, `mysql`, and others. To efficiently track and display the status of these services, you need to develop a script that retrieves the current status of each service and presents it on a web-based dashboard. This dashboard will provide real-time updates on whether each service is running or stopped.

Requirements

1. **Monitor Service Status**:
   - Check the status of critical services (`httpd`, `sshd`, `mysql`, etc.).
   - Determine if each service is running or stopped.
2. **Web-Based Dashboard**:
   - Display the service statuses in a web interface.
   - Provide real-time updates without manual refresh.
3. **User-Friendly Interface**:
   - Present the dashboard in a clear and organized manner.
   - Include meaningful labels and indicators for service statuses.
4. **Automatic Updates**:
   - Implement automatic updates or periodic checks to refresh the dashboard.

Task 17:

Task Description

As a system administrator at "MyComp," you are responsible for managing user accounts on the system. To streamline administrative tasks and ensure security compliance, you need to develop a script that automatically generates reports on user accounts. These reports should include details such as the last login time, password expiration status, and account status (active or inactive).

Requirements

1. **User Account Details**:
   - Retrieve and display user account details, including username, last login time, password expiration, and account status.
2. **Password Expiration**:
   - Check and report if a user's password is set to expire soon.
3. **Account Status**:
   - Determine if user accounts are active or inactive.

4. **Automated Report Generation**:
    - Automatically generate reports on user accounts at scheduled intervals (e.g., daily, weekly).
5. **User-Friendly Output**:
    - Present the reports in a clear and readable format for administrative review.

Task 18:

Task Description

As a system administrator at "MyComp," you are responsible for maintaining and configuring multiple servers. This includes ensuring that specific software packages are installed on these systems. To streamline this process, you need to develop a script that automates the installation of a list of specified packages. The script should handle dependencies, verify successful installation, and provide a report on the installation status of each package.

Requirements

1. **Package List**:
    - Specify a list of packages to be installed.
2. **Dependency Management**:
    - Automatically check for and install any dependencies required by the specified packages.
3. **Installation Verification**:
    - Verify that each package has been successfully installed.
4. **Error Handling**:
    - Handle errors gracefully and provide meaningful error messages if a package fails to install.
5. **Installation Report**:
    - Generate a report indicating the installation status of each package (successful or failed).

Task 19:

Task Description

As a system administrator at "MyComp," ensuring the security of your servers is crucial. To proactively identify and mitigate common security issues, you need to develop a script that performs security scans. This script should check for vulnerabilities such as open ports, weak passwords, and outdated software versions. It should then generate a detailed report highlighting these issues along with recommendations for remediation.

Requirements

1. **Security Checks**:
    - Perform scans for open ports, weak passwords, and outdated software versions.

    - Implement checks using appropriate tools and commands.
2. **Remediation Recommendations**:
    - Provide actionable recommendations for each identified security issue.
    - Include steps or commands to remediate the issues where possible.
3. **Detailed Report**:
    - Generate a comprehensive report detailing each security issue found.
    - Format the report for clarity and ease of understanding.
4. **Automation and Schedule**:
    - Automate the script to run periodically (e.g., daily, weekly).
    - Ensure that the script can be scheduled using cron or a similar tool.

Task 20:

Task Description

As a system administrator at "MyComp," ensuring optimal resource usage across your servers is critical for maintaining performance and stability. To proactively monitor system resource usage (CPU, memory, disk), you need to develop a script that continuously monitors these metrics. The script should send alerts to notify you when resource usage exceeds predefined thresholds. This will help you identify potential issues early and take appropriate actions to prevent system downtime or performance degradation.

Requirements

1. **Resource Monitoring**:
    - Continuously monitor CPU usage, memory usage, and disk usage.
2. **Threshold Definition**:
    - Define thresholds for each resource (e.g., CPU > 90%, memory > 80%, disk > 85%).
3. **Alert Mechanism**:
    - Send email alerts to notify the system administrator when any resource usage exceeds its threshold.
    - Include details such as the current resource usage level and timestamp in the alert.
4. **Automation**:
    - Automate the script to run at regular intervals (e.g., every 5 minutes).
    - Ensure reliability and efficiency in resource monitoring and alerting.

Task 21:

Task Description

As a system administrator at "MyComp," you are tasked with automating the deployment of a web application to streamline the deployment process and ensure consistency across multiple servers. The script you develop should automate the steps involved in deploying a web application, including downloading the source code from a repository, configuring the web server

(e.g., Apache or Nginx), and setting up any necessary databases. This automation will help reduce deployment time and minimize human errors.

Requirements

1. **Source Code Management**:
    - Download the web application source code from a Git repository or a specified URL.
2. **Web Server Configuration**:
    - Configure the web server to serve the deployed application.
    - Ensure that the necessary server modules or configurations are set up.
3. **Database Setup**:
    - If required, set up and configure databases needed by the web application.
    - Handle database schema initialization and user permissions as necessary.
4. **Logging and Verification**:
    - Log the deployment process and verify successful deployment of the application.

Task 22:

Task Description

As a system administrator at "MyComp," maintaining system cleanliness and optimizing storage usage is essential for efficient system operation. To automate routine cleanup tasks and ensure optimal performance, you need to develop a script that cleans up unnecessary files, clears caches, and removes unused packages from the system. This script should help free up disk space, improve system responsiveness, and maintain overall system health.

Requirements

1. **Temporary File Cleanup**:
    - Delete temporary files and directories that are no longer needed.
2. **Cache Clearing**:
    - Clear caches for applications such as web browsers, package managers, and system logs.
3. **Unused Package Removal**:
    - Remove unused or orphaned packages that are no longer required by the system.
4. **Logging and Verification**:
    - Log the cleanup activities performed by the script.
    - Verify successful execution of cleanup tasks and report any errors encountered.

Task 23:

Task Description

As a system administrator at "MyComp," keeping systems up-to-date with the latest security patches and software updates is crucial for maintaining system security and stability. To automate the process of checking for available system updates and notifying relevant stakeholders, you need to develop a script that runs daily, checks for available updates using `yum` or `dnf`, and sends an email notification containing the list of available updates. This automation will help ensure timely application of updates and proactive management of system vulnerabilities.

Requirements

1. **Automated Update Check**:
    - Script should automatically check for available updates using `yum` or `dnf`.
2. **Email Notification**:
    - Send an email notification containing the list of available updates to the system administrator or designated email recipient.
3. **Daily Execution**:
    - Schedule the script to run daily using cron or a similar scheduling mechanism.
4. **Logging and Verification**:
    - Log the update checking activities performed by the script.
    - Verify successful execution of update checks and report any errors encountered.

Task 24:

Task Description

As a system administrator at "MyComp," scheduling system reboots is necessary for applying updates and maintaining system health. To minimize disruption and inform users about scheduled reboots, you need to develop a script that sends email notifications to users and creates a broadcast message that appears at regular intervals on user terminals before the scheduled reboot time. This ensures users are aware of the upcoming reboot and can prepare accordingly.

Requirements

1. **Email Notification**:
    - Script should send an email notification to all users about the scheduled reboot, specifying the date and time of the reboot.
2. **Broadcast Message**:
    - Create a broadcast message that appears at regular intervals on user terminals before the reboot, reminding users of the scheduled downtime.
3. **Scheduled Execution**:
    - Schedule the script to run automatically at a specified time before the reboot using cron or a similar scheduling mechanism.
4. **Logging and Verification**:
    - Log the notification activities performed by the script.

- Verify successful execution of notification tasks and report any errors encountered.

Task 25:

Task Description

As a system administrator at "MyComp," you are tasked with automating the setup of custom user environments for new users joining the organization. This involves creating and configuring essential dotfiles such as `.bashrc`, `.bash_profile`, and others to set predefined aliases, functions, and environment variables. This script should ensure consistency and productivity by providing a standardized environment for all new user accounts.

Requirements

1. **Dotfile Configuration**:
   - Create and configure `.bashrc`, `.bash_profile`, and other dotfiles as needed for new user accounts.
   - Set predefined aliases, functions, and environment variables to customize the user environment.
2. **User-Specific Customization**:
   - Customize the environment based on user roles or preferences, if applicable.
3. **Logging and Verification**:
   - Log the setup activities performed by the script.
   - Verify successful execution of setup tasks and report any errors encountered.

Task 26:

Task Description

As a system administrator at "MyComp," you need to automate the analysis of `httpd` (Apache HTTP Server) logs to generate a summary report of key events, errors, and usage statistics. This script will help monitor server performance, identify potential issues, and provide insights into usage patterns.

Requirements

1. **Log Analysis**:
   - Parse `httpd` log files (`access_log` and `error_log`).
   - Extract key events, errors, and usage statistics such as top URLs accessed, status codes, and error counts.
2. **Summary Report**:
   - Generate a summary report with sections for key events, error details, and usage statistics.
   - Include counts, percentages, and trends where applicable.

3. **Automation and Scheduling**:
    - Automate the script to run periodically using cron or a similar scheduling mechanism.
    - Store or email the generated report for review and analysis by system administrators.

Task 27:

Task Description

As a system administrator at "MyComp," ensuring the integrity of critical system files is essential for maintaining system security and stability. You need to automate the process of verifying file integrity by comparing checksums of critical files against pre-defined known good values. This script will detect any unauthorized modifications or corruption in critical system files and alert the administrator for immediate action.

Requirements

1. **Checksum Calculation**:
    - Calculate checksums (SHA-256, MD5, etc.) for critical system files.
    - Store pre-defined checksum values for comparison.
2. **File Integrity Check**:
    - Compare current checksums of files with pre-defined values.
    - Alert if discrepancies or changes are detected.
3. **Alert Mechanism**:
    - Send email notification to the system administrator or alert through system logs if discrepancies are found.
    - Include details of affected files and nature of changes detected.
4. **Automation and Scheduling**:
    - Schedule the script to run periodically using cron or a similar scheduling mechanism.
    - Log activities and results for auditing and troubleshooting purposes.

Task 28:

Task Description

As a system administrator at "MyComp," you need to automate the testing of network connectivity to critical servers or IP addresses. This script will periodically check the reachability of these targets, measure response times, and log any issues such as packet loss or unreachable hosts. This ensures proactive monitoring of network connectivity and facilitates quick resolution of network issues.

Requirements

1. **Predefined Targets**:

    - Define a list of servers or IP addresses to test network connectivity.
2. **Connectivity Test**:
    - Perform connectivity tests (ping or similar) to each target.
    - Measure response time (latency) and detect packet loss.
3. **Logging and Reporting**:
    - Log test results including timestamp, target hostname/IP, response time, and packet loss percentage.
    - Maintain logs for historical analysis and troubleshooting.
4. **Automation and Scheduling**:
    - Schedule the script to run periodically using cron or a similar scheduling mechanism.
    - Send alerts or notifications if connectivity issues are detected.

Task 29:

Task Description

As a system administrator at "MyComp," you need to automate the process of checking user accounts with passwords set to expire soon. This script will identify users whose passwords are expiring within a specified number of days and send them notification emails, reminding them to change their passwords. This ensures users maintain secure access to systems without experiencing unexpected password expiration.

Requirements

1. **Password Expiry Check**:
    - Query user account details to determine passwords that are expiring soon.
    - Calculate days until password expiration based on current date and user account settings.
2. **Notification Email**:
    - Send an email notification to users whose passwords are expiring soon.
    - Include details such as username, days until expiration, and instructions to change password.
3. **Automation and Scheduling**:
    - Schedule the script to run periodically using cron or a similar scheduling mechanism.
    - Customize notification frequency and content based on organizational policies.

Task 30:

Task Description

As a system administrator at "MyComp," you need to automate the process of checking the health of your systems. This script will perform various checks to ensure the system's stability and reliability. It includes checking for disk errors, verifying file system integrity, and ensuring

critical services necessary for operation are running. The script will log any issues found and send alerts or notifications to administrators for prompt resolution.

Requirements

1. **Disk Error Checking**:
    - Scan all disks for errors using `fsck` or appropriate disk checking utilities.
    - Log and report any disk errors detected.
2. **File System Integrity Verification**:
    - Check the integrity of file systems using `fsck` or equivalent tools.
    - Log and report any file system integrity issues.
3. **Critical Services Status**:
    - Verify the status of critical services (e.g., `httpd`, `sshd`, `mysql`) to ensure they are running.
    - Restart services if they are not running and log actions taken.
4. **Automation and Scheduling**:
    - Schedule the script to run periodically using cron or a similar scheduling mechanism.
    - Implement alert mechanisms (e.g., email notifications) for critical issues detected during health checks.

Task 31:

Task Description

As a system administrator at "MyComp," you need to automate the backup process for databases running on your servers. This script will connect to databases (MySQL, PostgreSQL), dump their contents to backup files, compress the backups to save storage space, and implement rotation to manage old backups efficiently. This ensures data integrity and provides a mechanism to recover data in case of failures or data loss incidents.

Requirements

1. **Database Backup**:
    - Connect to MySQL and/or PostgreSQL databases to dump their contents.
    - Store backup files in a designated directory with timestamps for identification.
2. **Compression**:
    - Compress backup files to reduce storage space usage and improve transfer efficiency.
3. **Rotation**:
    - Implement backup rotation to manage storage space and keep a defined number of recent backups.
    - Delete older backups based on a retention policy (e.g., keep last 7 days' backups).
4. **Automation and Scheduling**:

    - Schedule the script to run periodically using cron or a similar scheduling mechanism.
    - Log actions and results for monitoring and audit purposes.

Task 32:

Task Description

As a system administrator at "MyComp," you manage multiple remote servers and need to perform routine management tasks remotely. This script will connect to each remote server using SSH, execute commands to update packages, check system status, and restart services as needed. This automation ensures consistent server management across your infrastructure without manual intervention on each server individually.

Requirements

1. **Remote Server Access**:
    - Use SSH to connect securely to remote servers.
    - Authenticate using SSH keys or passwords as appropriate.
2. **Management Tasks**:
    - Update packages using package manager commands (`yum`, `apt`, etc.).
    - Check system status (e.g., CPU usage, memory usage, disk space).
    - Restart specified services (e.g., `httpd`, `sshd`, `mysql`) if they are not running or need to be restarted.
3. **Automation and Logging**:
    - Schedule the script to run periodically using cron or a similar scheduling mechanism.
    - Log actions and results for monitoring and audit purposes.

Task 33:

Task Description

As a system administrator at "MyComp," you need to monitor critical processes on your servers. This script will continuously check the status of specified processes, alert administrators via email or other notification mechanisms if any process is not running or has crashed, and attempt to restart the processes to maintain system stability and uptime.

Requirements

1. **Process Monitoring**:
    - Specify a list of critical processes (e.g., `httpd`, `sshd`, `mysql`) to monitor.
    - Check the status of these processes periodically.
2. **Alerting Mechanism**:

    - Send alerts to administrators via email or other means if a process is not running or has crashed.
3. **Automated Recovery**:
    - Attempt to restart the processes automatically if they are not running.
4. **Automation and Logging**:
    - Schedule the script to run periodically using cron or a similar scheduling mechanism.
    - Log actions and results for monitoring and audit purposes.

Task 34:

Task Description

As a system administrator at "MyComp," you need to synchronize files between two directories, which could be on the same server or across the network. This script will use `rsync` to perform efficient file synchronization, ensuring that changes in one directory are reflected in the other. Logging will track the synchronization process, providing visibility into file changes and ensuring data integrity.

Requirements

1. **File Synchronization**:
    - Use `rsync` to synchronize files between a source directory (`SOURCE_DIR`) and a destination directory (`DEST_DIR`).
    - Handle synchronization both locally (same server) and across the network (different servers).
2. **Logging**:
    - Log synchronization actions and results to a log file (`SYNC_LOG_FILE`) for monitoring and audit purposes.
    - Include timestamps and details of synchronized files in the log.
3. **Automation and Scheduling**:
    - Schedule the script to run periodically using cron or a similar scheduling mechanism to maintain synchronization.
    - Ensure robust error handling and reporting to manage synchronization failures.

Task 35:

As a system administrator at "MyComp," you need to monitor and log system performance metrics regularly to analyze system health and performance trends. This script will collect CPU usage, memory usage, and disk I/O statistics at specified intervals, store the data in a log file, and optionally provide alerts or notifications based on predefined thresholds.

Requirements

1. **Performance Metrics**:

    - Monitor CPU usage (%), memory usage (%), and disk I/O (read/write operations) over time.
    - Collect metrics at regular intervals (e.g., every 5 minutes).
2. **Data Logging**:
    - Store collected metrics in a log file (`PERF_LOG_FILE`) with timestamps for each data point.
3. **Automation and Scheduling**:
    - Schedule the script to run periodically using cron or a similar scheduling mechanism to capture performance metrics continuously.
    - Optionally include alerting mechanisms based on predefined thresholds for CPU, memory, or disk usage.

Task 36:

Task Description

As a system administrator at "MyComp," you need to automate user management tasks in an LDAP directory. This script will facilitate operations such as adding new users, deleting existing users, and modifying user attributes like names, email addresses, and group memberships. This ensures efficient management of LDAP user entries across the organization.

Requirements

1. **LDAP Utilities**:
    - Utilize command-line LDAP utilities (`ldapadd`, `ldapmodify`, `ldapdelete`) to perform user management operations.
    - Authenticate with LDAP server credentials for administrative privileges (`LDAP_ADMIN_USER`, `LDAP_ADMIN_PASS`).
2. **User Management Operations**:
    - **Adding Users**: Add new user entries with specified attributes (`uid`, `cn`, `mail`, `userPassword`, etc.).
    - **Deleting Users**: Remove user entries based on their `uid`.
    - **Modifying Users**: Update user attributes such as `mail`, `cn`, and group memberships (`memberOf`).
3. **Error Handling and Logging**:
    - Implement error handling to manage exceptions during LDAP operations.
    - Log actions and results to a log file (`LDAP_MANAGEMENT_LOG`) for auditing and troubleshooting purposes.

Task 37:

Task Description

As a system administrator at "MyComp," you are tasked with automating system hardening procedures to enhance the security posture of servers. This script will automate tasks such as disabling unnecessary services, applying security updates, configuring firewall rules, and adjusting system settings to reduce vulnerabilities and enhance overall system security.

Requirements

1. **System Hardening Tasks**:
    - **Disable Unused Services**: Identify and disable unnecessary services that are not required for system operation.
    - **Apply Security Patches**: Automate the process of checking for and applying security updates using package management tools (`yum` or `dnf` for CentOS/RHEL, `apt` for Debian/Ubuntu).
    - **Configure Firewall Rules**: Implement firewall rules to restrict access to necessary ports and services.
    - **Adjust System Settings**: Configure system settings (e.g., password policies, file permissions) to align with security best practices.
2. **Error Handling and Logging**:
    - Implement error handling mechanisms to manage exceptions during hardening operations.
    - Log actions and results (`HARDENING_LOG_FILE`) for auditing and troubleshooting purposes.