Week 5 Homework Submission File: Archiving and Logging Data

Please edit this file by adding the solution commands on the line below the prompt.

Save and submit the completed file for your homework submission.

Step 1: Create, Extract, Compress, and Manage tar Backup Archives

1. Command to **extract** the `TarDocs.tar` archive to the current directory:
2. Command to **create** the `Javaless_Doc.tar` archive from the `TarDocs/` directory, while excluding the `TarDocs/Documents/Java` directory:
3. Command to ensure `Java/` is not in the new `Javaless_Docs.tar` archive:

Bonus - Command to create an incremental archive called `logs_backup_tar.gz` with only changed files to `snapshot.file` for the `/var/log` directory:

Critical Analysis Question

- Why wouldn't you use the options `-x` and `-c` at the same time with `tar`?

Step 2: Create, Manage, and Automate Cron Jobs

1. Cron job for backing up the `/var/log/auth.log` file:

Step 3: Write Basic Bash Scripts

1. Brace expansion command to create the four subdirectories:
2. Paste your `system.sh` script edits below:

```
#!/bin/bash
[Your solution script contents here]
```

3. Command to make the `system.sh` script executable:

Optional - Commands to test the script and confirm its execution:

Bonus - Command to copy `system` to system-wide cron directory:

Step 4. Manage Log File Sizes

1. Run `sudo nano /etc/logrotate.conf` to edit the `logrotate` configuration file.

   Configure a log rotation scheme that backs up authentication messages to the `/var/log/auth.log`.

   - Add your config file edits below:

```
[Your logrotate scheme edits here]
```

Bonus: Check for Policy and File Violations

1. Command to verify `auditd` is active:
2. Command to set number of retained logs and maximum log file size:

   - Add the edits made to the configuration file below:

```
[Your solution edits here]
```

3. Command using `auditd` to set rules for `/etc/shadow`, `/etc/passwd` and `/var/log/auth.log`:

   - Add the edits made to the `rules` file below:

```
[Your solution edits here]
```

4. Command to restart `auditd`:
5. Command to list all `auditd` rules:

6. Command to produce an audit report:

7. Create a user with `sudo useradd attacker` and produce an audit report that lists account modifications:

8. Command to use `auditd` to watch `/var/log/cron`:

9. Command to verify `auditd` rules:

Bonus (Research Activity): Perform Various Log Filtering Techniques

1. Command to return `journalctl` messages with priorities from emergency to error:

2. Command to check the disk usage of the system journal unit since the most recent boot:

3. Comand to remove all archived journal files except the most recent two:

4. Command to filter all log messages with priority levels between zero and two, and save output to `/home/sysadmin/Priority_High.txt`:

5. Command to automate the last command in a daily cronjob. Add the edits made to the crontab file below:

```
[Your solution cron edits here]
```
