---

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

1. Command to \*\*extract\*\* the `TarDocs.tar` archive to the current directory: **sudo tar xvvf TarDocs.tar**
2. Command to \*\*create\*\* the `Javaless\_Doc.tar` archive from the `TarDocs/` directory, while excluding the `TarDocs/Documents/Java` directory: **sudo tar cvvf Javaless\_Docs.tar -–exclude=’TarDocs/Documents/Java’ ~/Projects**
3. Command to ensure `Java/` is not in the new `Javaless\_Docs.tar` archive: **sudo tar -tvf Javaless\_Docs.tar | grep ‘Java’**

### Step 2: Create, Manage, and Automate Cron Jobs

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

**crontab -e**

**set to run at 6am every Wednesday: 0 6 \* \* 3**

**sudo tar zcf ~/auth\_backup.tgz /var/log/auth.log**

**gzip /auth\_backup.tgz**

### Step 3: Write Basic Bash Scripts

1. Brace expansion command to create the four subdirectories:

**mkdir -p ~/sysadmin/backups/{freemem,diskuse,openlist,freedisk}**

2. Paste your `system.sh` script edits below:

```bash

**#!/bin/bash**

**#Instructions**

**#Free memory output to a free\_mem.txt file**

**free -h > ~/backups/freemem/free\_mem.txt**

**#Disk usage output to a disk\_usage.txt file**

**du -h > ~/backups/diskuse/disk\_usage.txt**

**#List open files to an open\_list.txt file**

**lsof > ~/backups/openlist/open\_list.txt**

```

1. Command to make the `system.sh` script executable: **chmod +x system.sh**

\*\*Optional\*\*

- Commands to test the script and confirm its execution: **cd ~/backups cat system.sh**

\*\*Bonus\*\*

- Command to copy `system` to system-wide cron directory: **cp system.sh >**

### 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:

**/var/log/auth.log {**

**weekly**

**rotate7**

**notifempty**

**delaycompress**

**missingok**

**}**

### Bonus: Check for Policy and File Violations

1. Command to verify `auditd` is active: **systemctl status auditd**

2. Command to set number of retained logs and maximum log file size:

- Add the edits made to the configuration file below:

```bash

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:

```bash

[Your solution edits here]

```

4. Command to restart `auditd`: **sudo systemctl restart auditd**

5. Command to list all `auditd` rules: **sudo** **auditctl -l**

6. Command to produce an audit report: **sudo aureport -au**

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: **sudo auditctl -l**

---

### Bonus (Research Activity): Perform Various Log Filtering Techniques

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

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

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

1. Command to filter all log messages with priority levels between zero and two, and save output to `/home/sysadmin/Priority\_High.txt`:

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

```bash

[Your solution cron edits here]

```