## Week 5 Homework Submission File: Archiving and Logging Data.

---

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

1. Command to \*\*extract\*\* the `TarDocs.tar` archive to the current directory: tar xvvf TarDocs.tar

2. Command to \*\*create\*\* the `Javaless\_Doc.tar` archive from the `TarDocs/` directory, while excluding the `TarDocs/Documents/Java` directory:

tar -cvvf ~/Projects/Javaless\_Docs.tar --exclude="\*Java" ~/Projects/TarDocs/Documents

3. Command to ensure `Java/` is not in the new `Javaless\_Docs.tar` archive:

tar --list -f Javaless\_Docs.tar | grep Java OR tar -tvf Javaless\_Docs.tar | grep Java

\*\*Bonus\*\*

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

sudo tar --listed-incremental=snapshot.file -cvzf logs\_backup\_tar.gz /var/log

#### Critical Analysis Question

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

--- X extracts the archive and C creates the archive. They would cancel each other out.

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

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

--- 0 6 \* \* wed tar czvf /var/log/auth\_backup.tgz /var/log/auth.log

### Step 3: Write Basic Bash Scripts

1. Brace expansion command to create the four subdirectories:

mkdir ~/Projects/backups/{freemem,diskuse,openlist,freedisk}

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

```bash

#!/bin/bash

# Prints the amount of free memory on the system

cat /proc/meminfo | grep MemFree >> ~/Projects/backups/freemem/free\_mem.txt

# Prints disk usage

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

# Lists all open files

isof >> ~/Projects/backups/openlist/open\_list.txt

# Prints file system disk space statistics

df -h >> ~/Projects/backups/freedisk/free\_disk.txt

```

3. Command to make the `system.sh` script executable: sudo chmod +x system.sh

\*\*Optional\*\*

- Commands to test the script and confirm its execution: sudo ./system.sh

\*\*Bonus\*\*

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

--- cp system.sh /etc/cron.weekly

### 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:

```bash

/var/log/auth.log {

weekly

rotate 7

notifempty

delaycompress

missingok

endscript

}

```

---

### Bonus: Check for Policy and File Violations

1. Command to verify `auditd` is active: systemctl | grep auditd

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

sudo nano /etc/audit/auditd.conf

- Add the edits made to the configuration file below:

```bash

max\_log\_file = 35

num\_logs = 7

```

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: aureport -au

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

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

sudo journalctl -b -p 0..3

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

journalctl --disk-usage

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

journalctl --vacuum-file=2

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

sudo journalctl -p 0..2 >> /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:

```bash

```@daily

---

© 2020 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.