



Module 5 Challenge Submission File

Archiving and Logging Data

Make a copy of this document to work in, and then for each step, add the solution command below the prompt. Save and submit this completed file as your Challenge deliverable.

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

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

```
cd ~/Projects  
sudo tar xvf TarDocs.tar
```

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

```
sudo tar -cvvf Javaless_Docs.tar --exclude='TarDocs/Documents/Java' TarDocs/
```

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

```
tar tvvf Javaless_Docs.tar | grep Java
```

Optional

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

[Enter answer here]

Critical Analysis Question

5. 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:

```
0 6 * * */3 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 {freemem,diskuse,openlist,freedisk}
```

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

```
#!/bin/bash

echo -e "CPU_memory: $(free -h) \n" > ~/backups/freemem/free_mem.txt
echo -e "disk_usage: $(du -h) \n" > ~/backups/diskuse/disk_usage.txt
echo -e "open_file_list: $(lsof) \n" > ~/backups/openlist/open_list.txt
echo -e "free_disk: $(df -h) \n" > ~/backups/freedisk/free_disk.txt
```

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

```
chmod +x system.sh
```

Optional

4. Commands to test the script and confirm its execution:

```
sysadmin@UbuntuDesktop:~$ ./system.sh
sysadmin@UbuntuDesktop:~$ cd backups
sysadmin@UbuntuDesktop:~/backups$ ls
diskuse freedisk freemem openlist
sysadmin@UbuntuDesktop:~/backups$ cd freemem/
sysadmin@UbuntuDesktop:~/backups/freemem$ ls
free_mem.txt
sysadmin@UbuntuDesktop:~/backups/freemem$ cat free_mem.txt
CPU_memory:                total          used          free          shared
buff/cache    available
Mem:           3.9G          1.2G          1.5G          48M          1.1G
2.3G
Swap:           1.9G           0B          1.9G
```

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

[Enter answer here]

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`.

- a. Add your config file edits:

```
/var/log/auth.log {  
rotate 7  
weekly  
notifempty  
delaycompress  
missingok  
endscript  
}
```

Optional Additional Challenge: Check for Policy and File Violations

1. Command to verify `auditd` is active:

[Enter answer here]

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

[Enter answer here]

Add the edits made to the configuration file:

[Enter answer here]

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

[Enter answer here]

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

[Enter answer here]

4. Command to restart `auditd`:

[Enter answer here]

5. Command to list all `auditd` rules:

[Enter answer here]

6. Command to produce an audit report:

[Enter answer here]

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

[Enter answer here]

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

[Enter answer here]

9. Command to verify `auditd` rules:

[Enter answer here]

Optional (Research Activity): Perform Various Log Filtering Techniques

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

[Enter answer here]

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

[Enter answer here]

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

[Enter answer here]

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

[Enter answer here]

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

[Your solution cron edits here]