## 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:

**Tar xvvf TarDocs.tar ~/Projects**

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

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

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

**tar tvvf 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:

#### Critical Analysis Question

- Why wouldn't you use the options `-x` and `-c` at the same time with `tar`? **The option -x is used to extract files from a tarball, while the option -c is used to create a tarball. You can not do both at once.**

---

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

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

**0 6 \* \* 4 tar -cfz /auth\_backup.tgz /var/log/auth.log**

---

### Step 3: Write Basic Bash Scripts

1. Brace expansion command to create the four subdirectories:

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

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

```bash

**#!/bin/bash**

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

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

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

**df -h > ~/backups/freedisk/free\_disc.txt**

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

**Chmod +x system.sh**

\*\*Optional\*\*

- Commands to test the script and confirm its execution:

**Sudo ./system.sh**

**Cat ~/backups/freemem/free\_mem.txt**

\*\*Bonus\*\*

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

**Mv ~/system.sh /etc/cront.weekly**

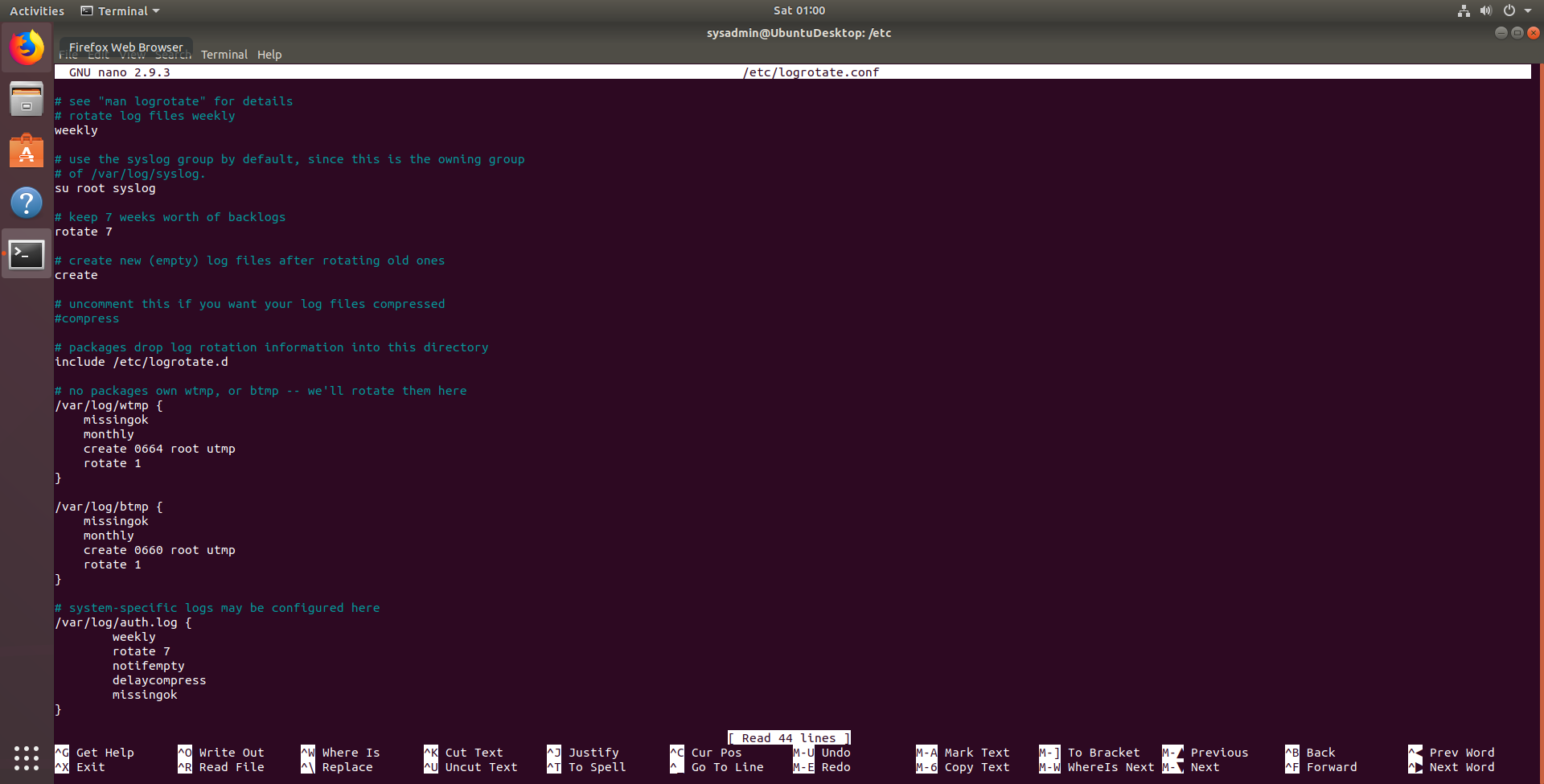
---

### 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:



---

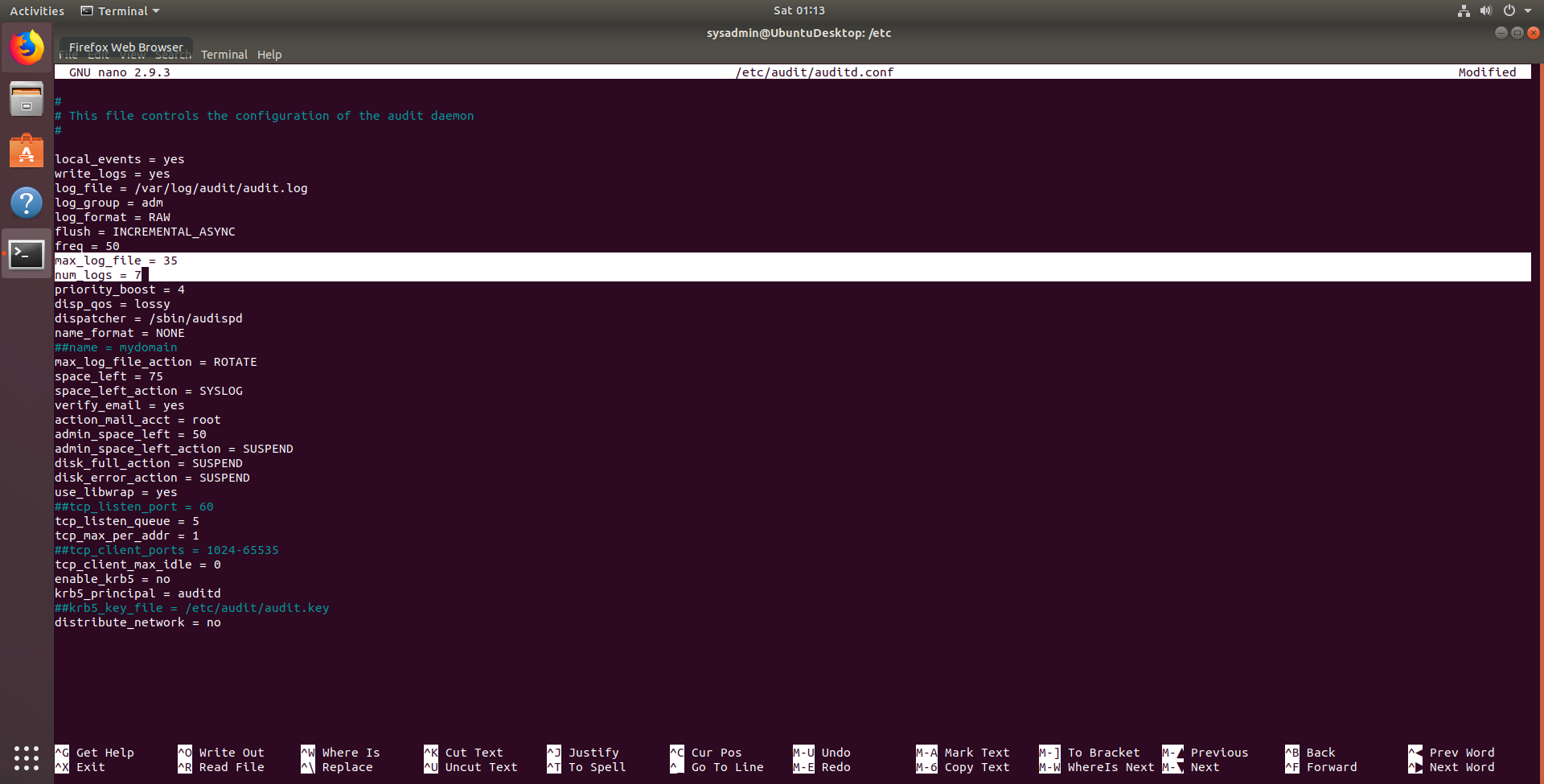
### Bonus: Check for Policy and File Violations

1. Command to verify `auditd` is active:

**Sudo systemctl status auditd**

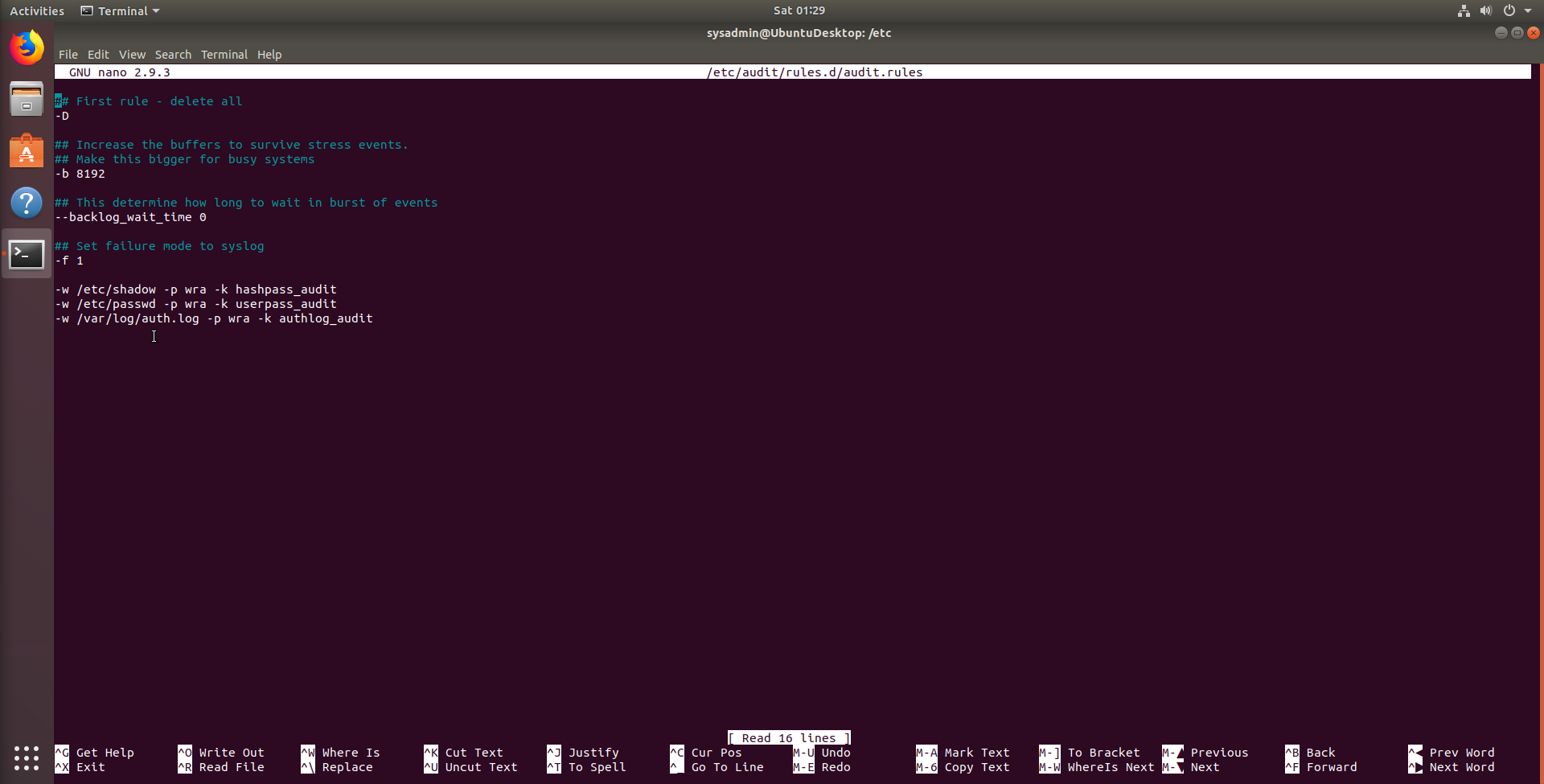
2. Command to set number of retained logs and maximum log file size:

- Add the edits made to the configuration file below:



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:



4. Command to restart `auditd`:

**Sudo systemctl restart auditd**

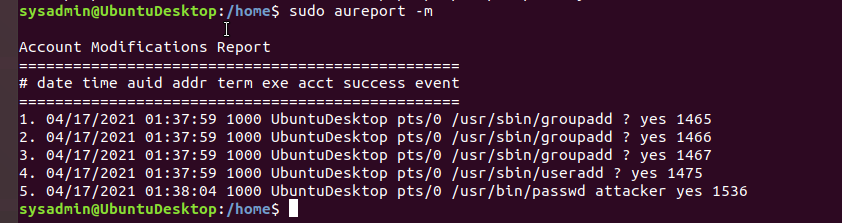
5. Command to list all `auditd` rules:

**Sudo auditctl -l**

6. Command to produce an audit report:

**Sudo aureport -au**

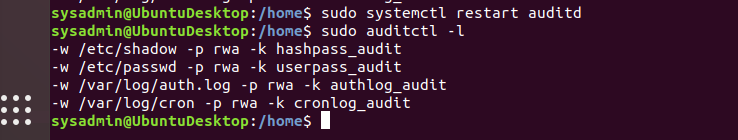
7. Create a user with `sudo useradd attacker` and produce an audit report that lists account modifications:



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

**-w /var/log/cron -p wra -k cronlog\_audit**

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:

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]

```

---

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