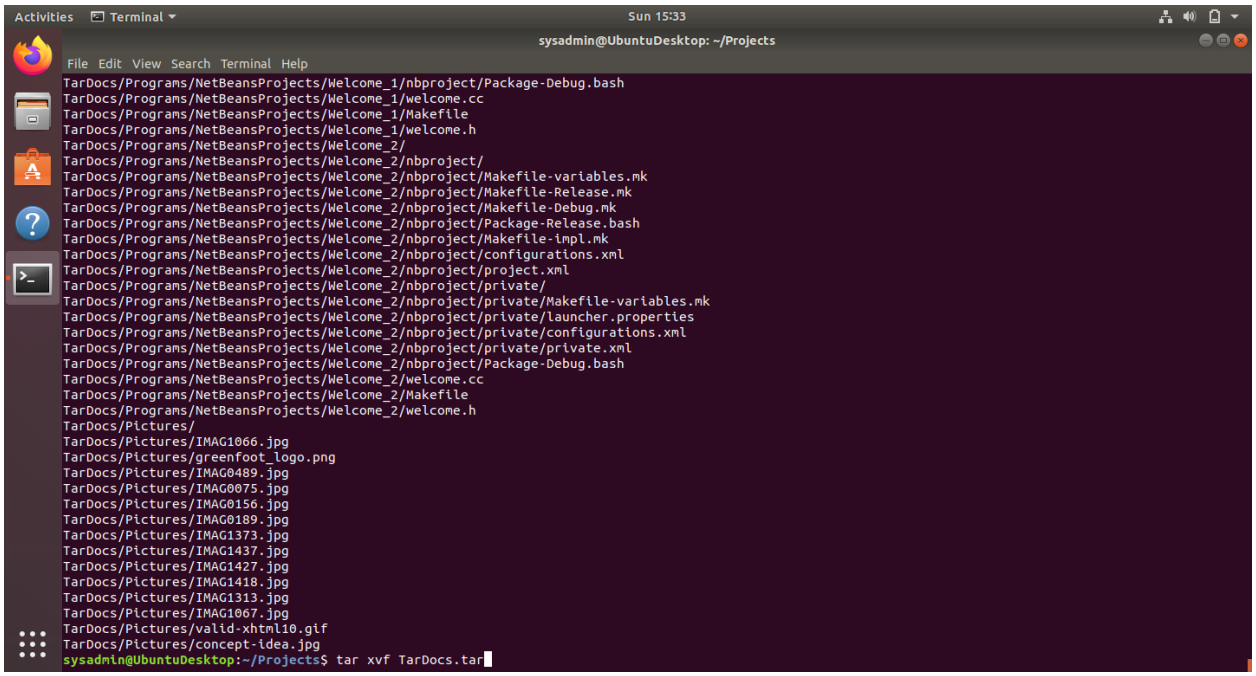
Week 5 Homework Submission File: Archiving and Logging Data

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

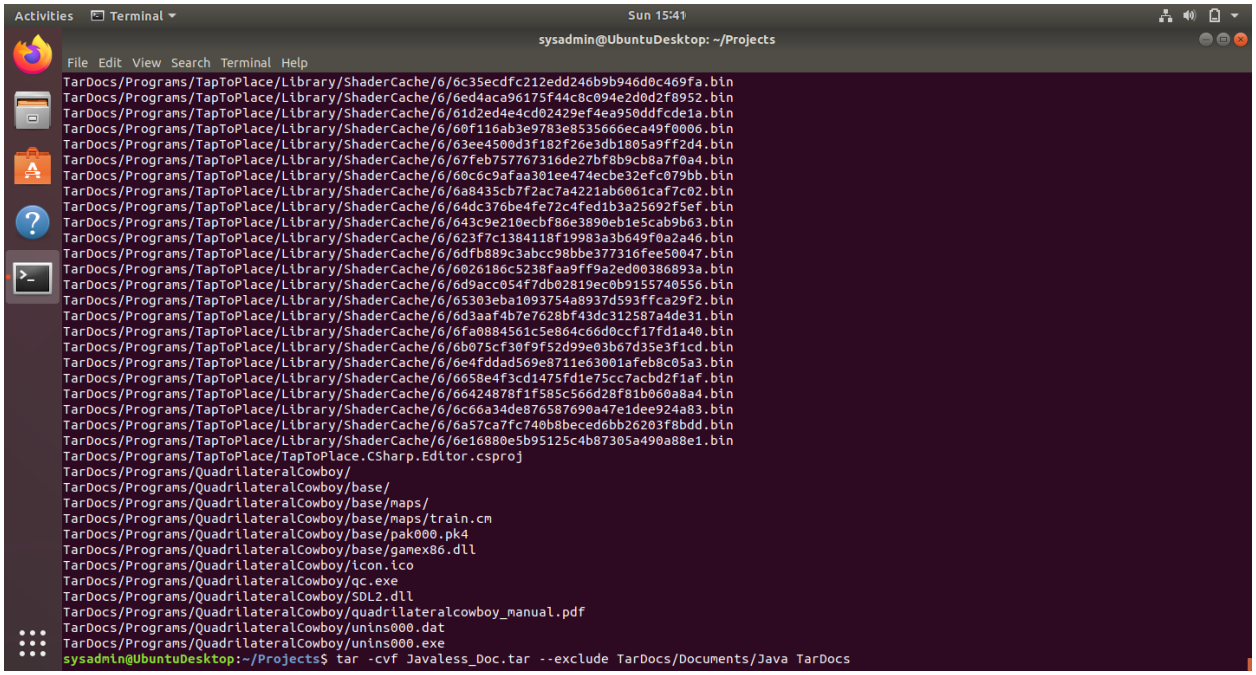
1. Command to **extract** the TarDocs.tar archive to the current directory:

   tar xvf TarDocs.tar



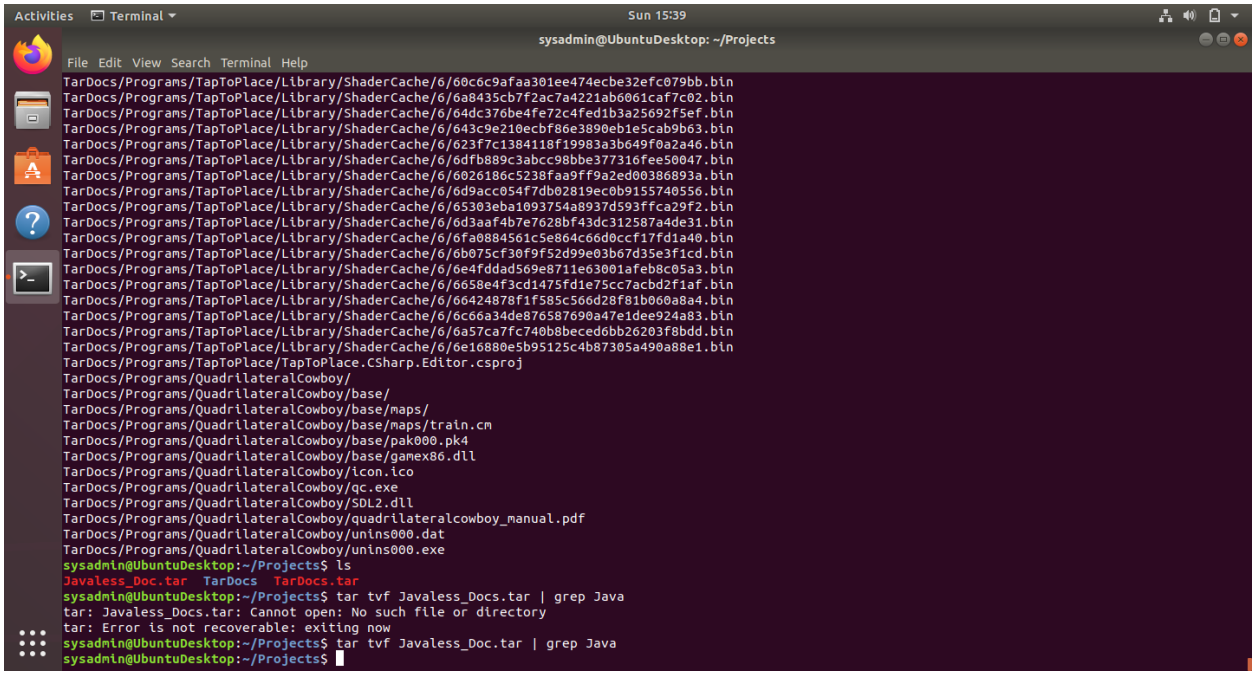
2. Command to **create** the Javaless_Doc.tar archive from the TarDocs/ directory, while excluding the TarDocs/Documents/Java directory:

tar -cvf Javaless_Doc.tar - -exclude TarDocs/Documents/Java TarDocs



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

tar tvf Javaless_Doc.tar | grep Java



Bonus

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

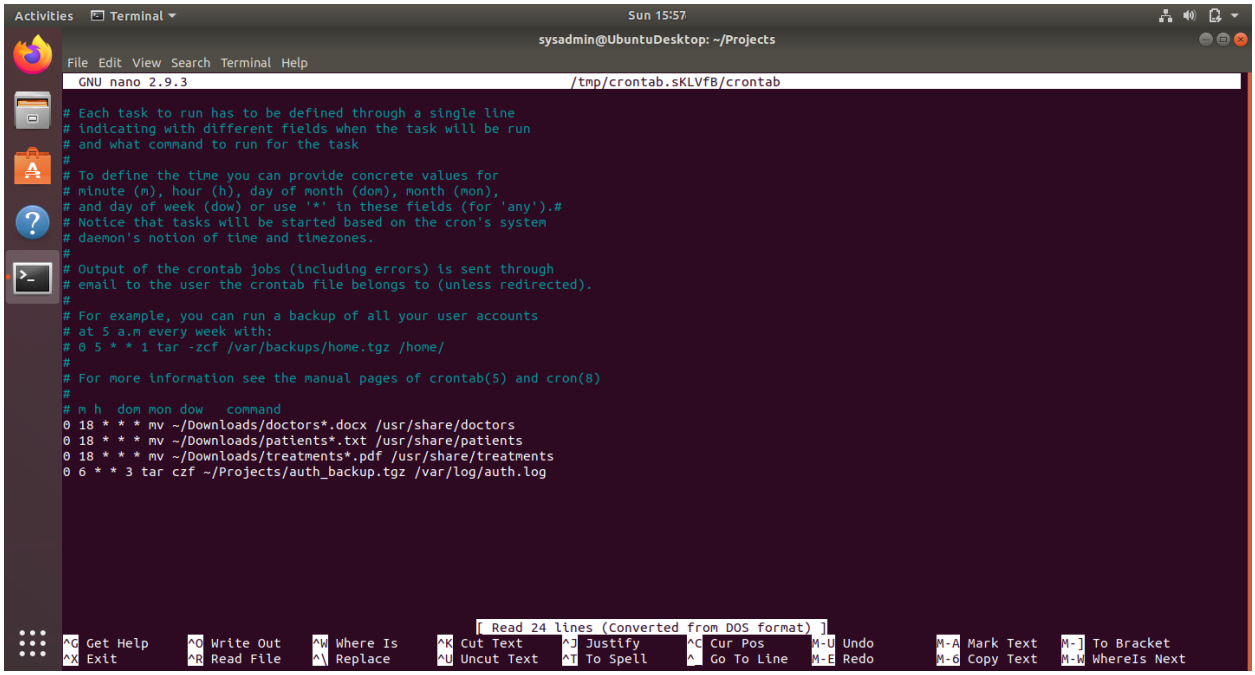
Critical Analysis Question

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

Step 2: Create, Manage, and Automate Cron Jobs

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

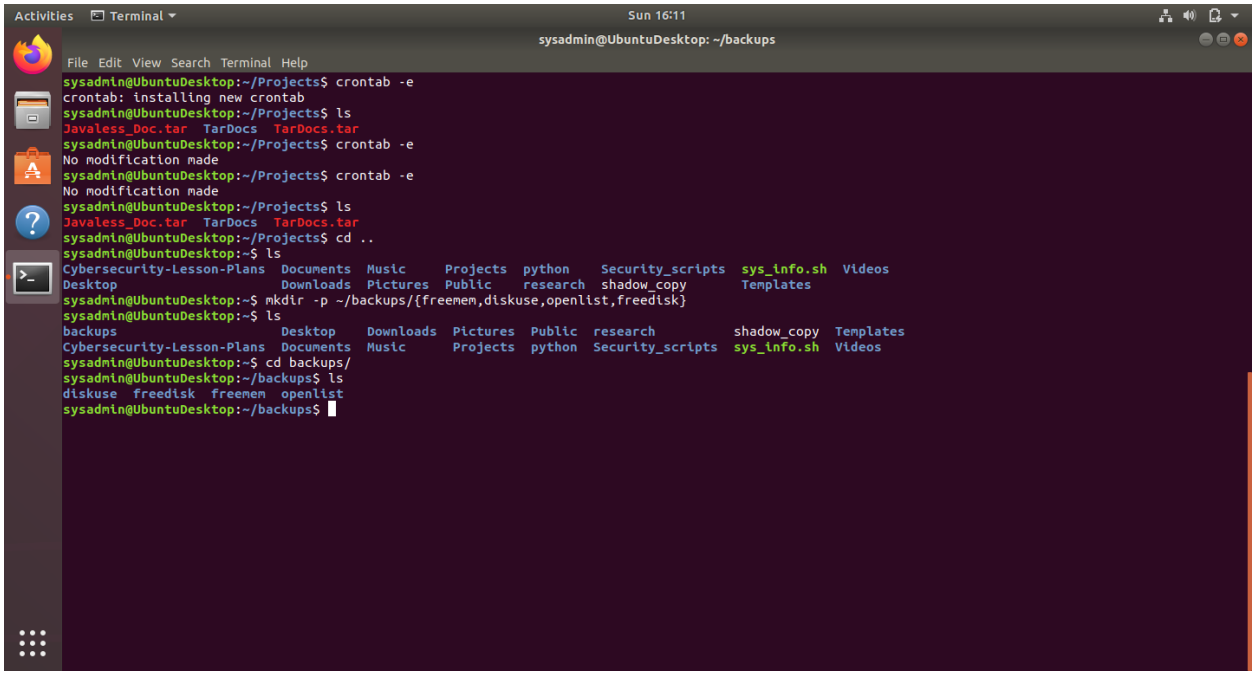
   0 6 * * 3 tar czf ~/Projects/auth_backup.tgz /var/log/auth.log



Step 3: Write Basic Bash Scripts

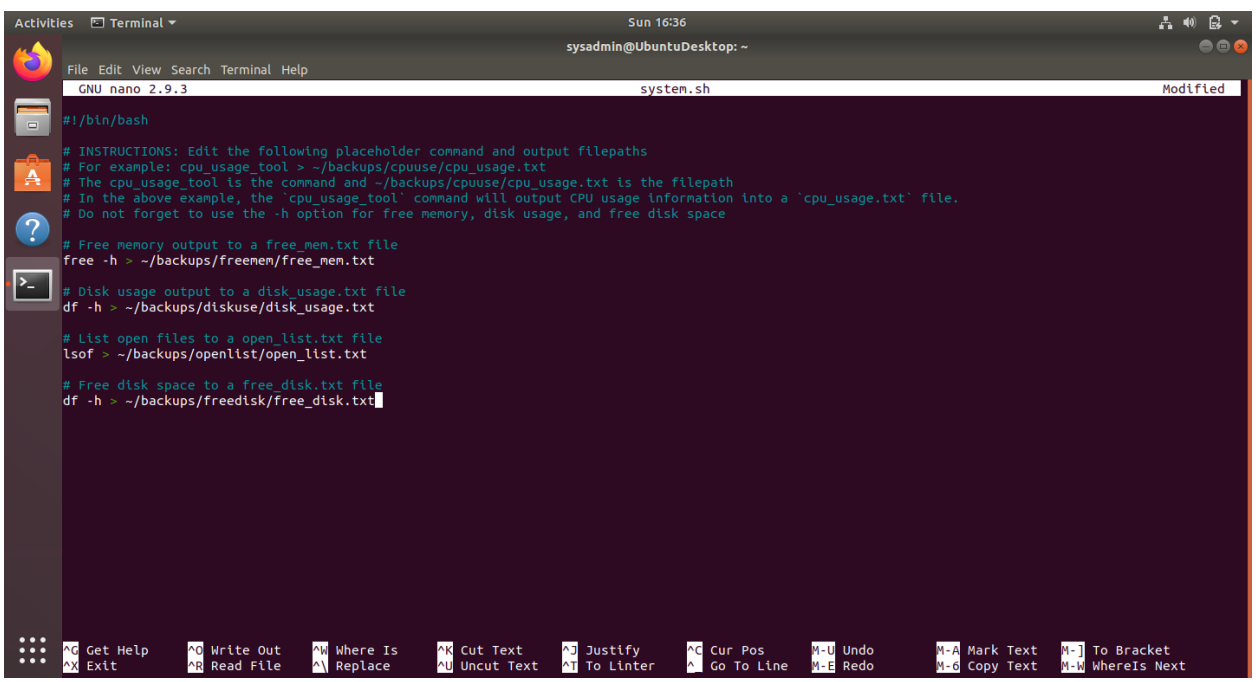
1. Brace expansion command to create the four subdirectories:

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



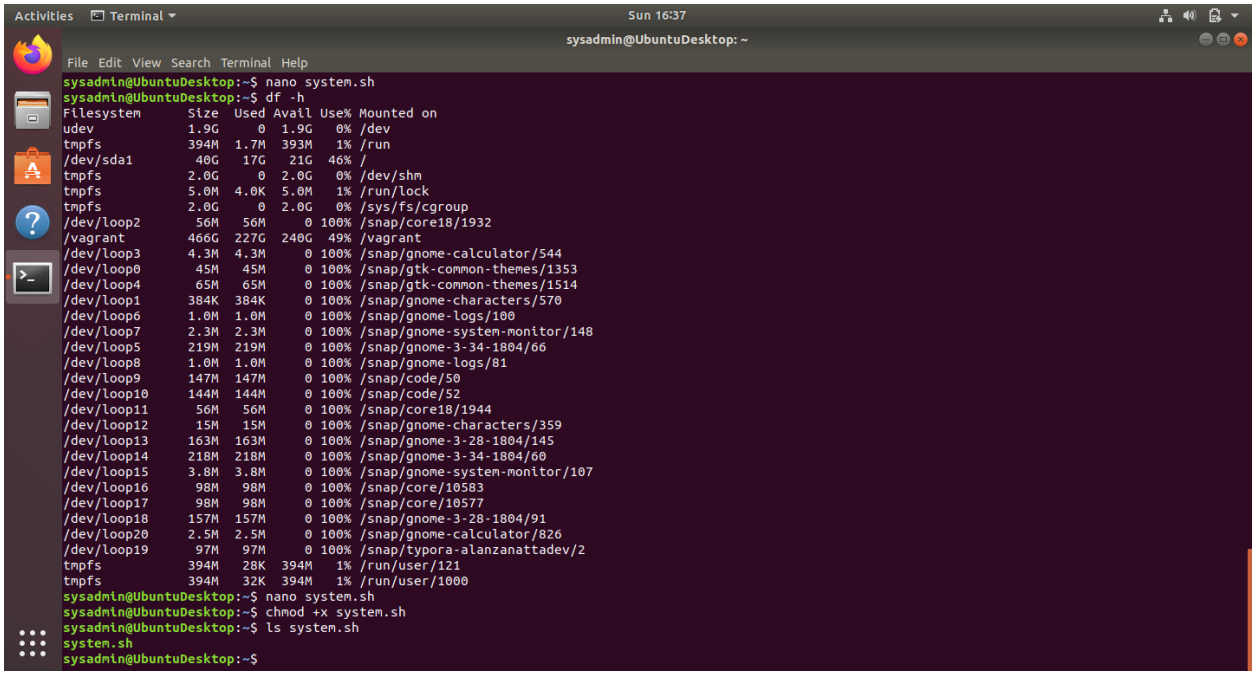
Paste your system.sh script edits below:
#!/bin/bash

2. [Your solution script contents here]



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

chmod +x system.sh



Optional

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

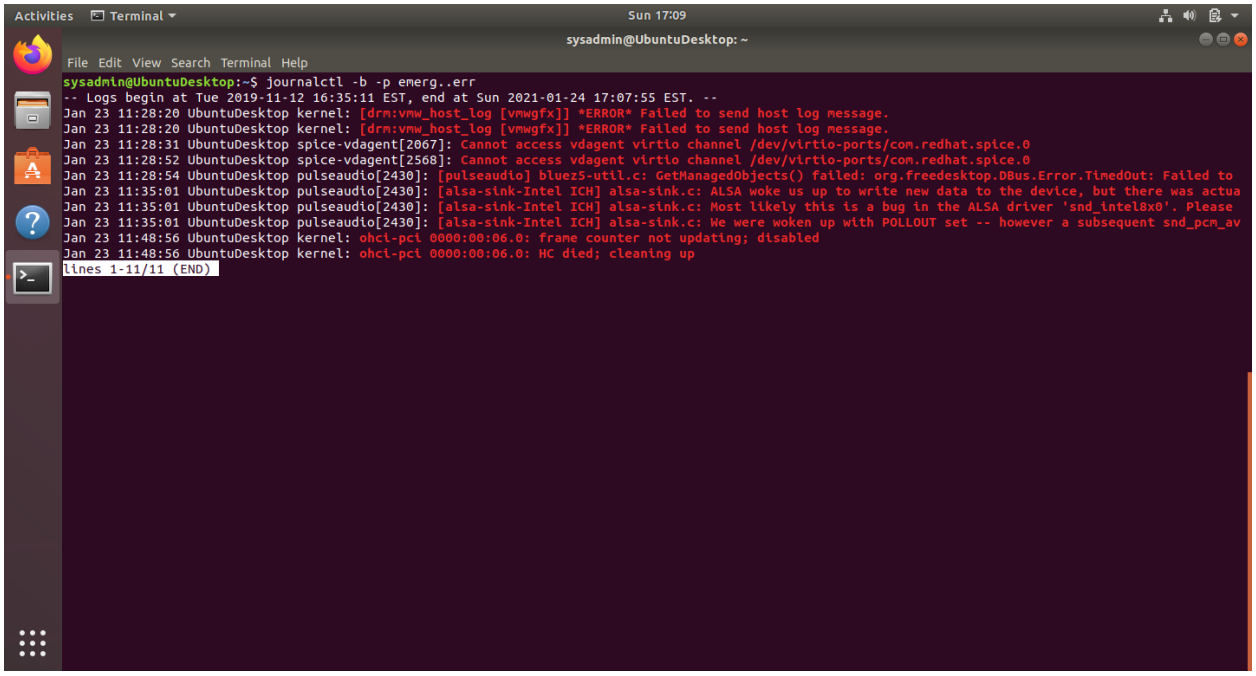
Bonus

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

Step 4: Perform Various Log Filtering Techniques

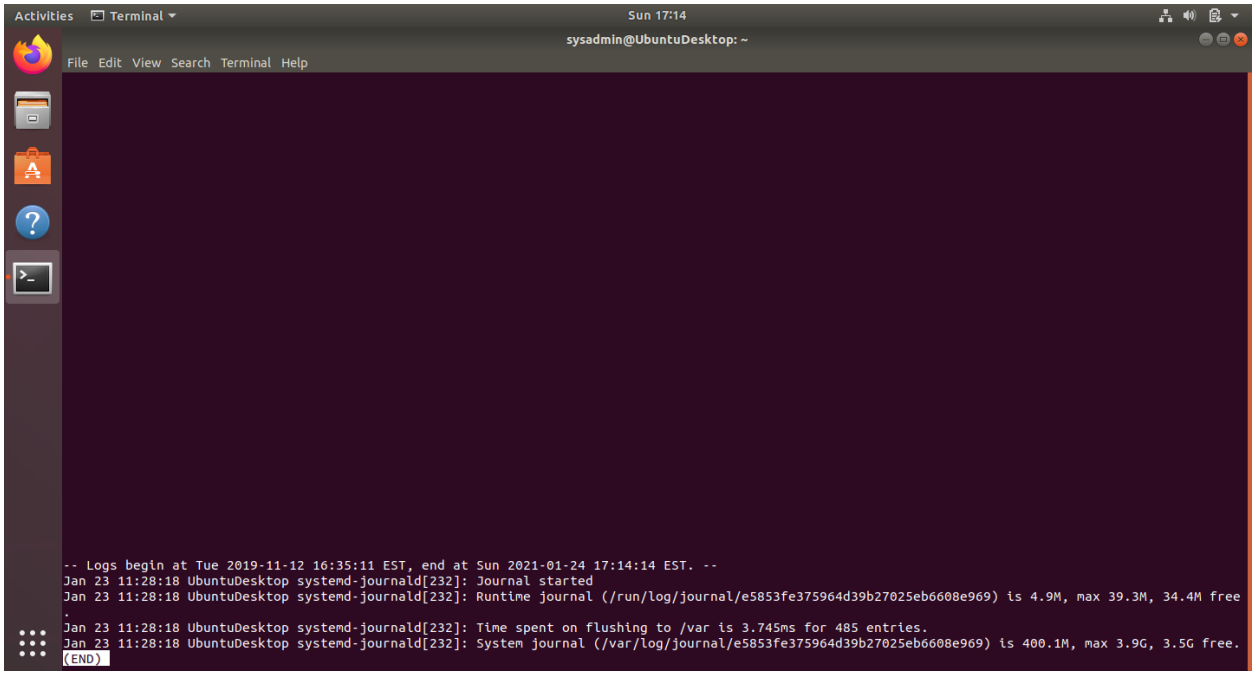
1. Command to return journalctl messages with priorities from emergency to error:

journalctl -b -p emerg..err



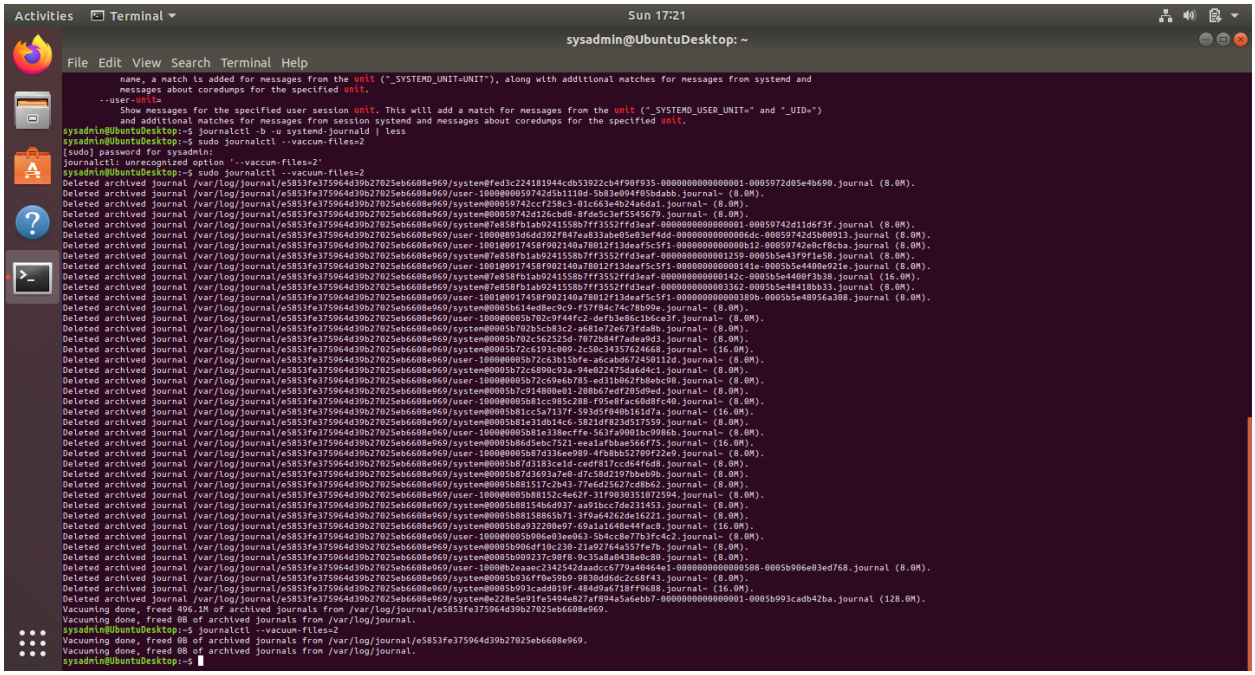
2. Command to check the disk usage of the system journal unit since the most recent boot:

journalctl -b -u systemd-journald | less



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

sudo journalctl --vacuum-files=2



Bonus

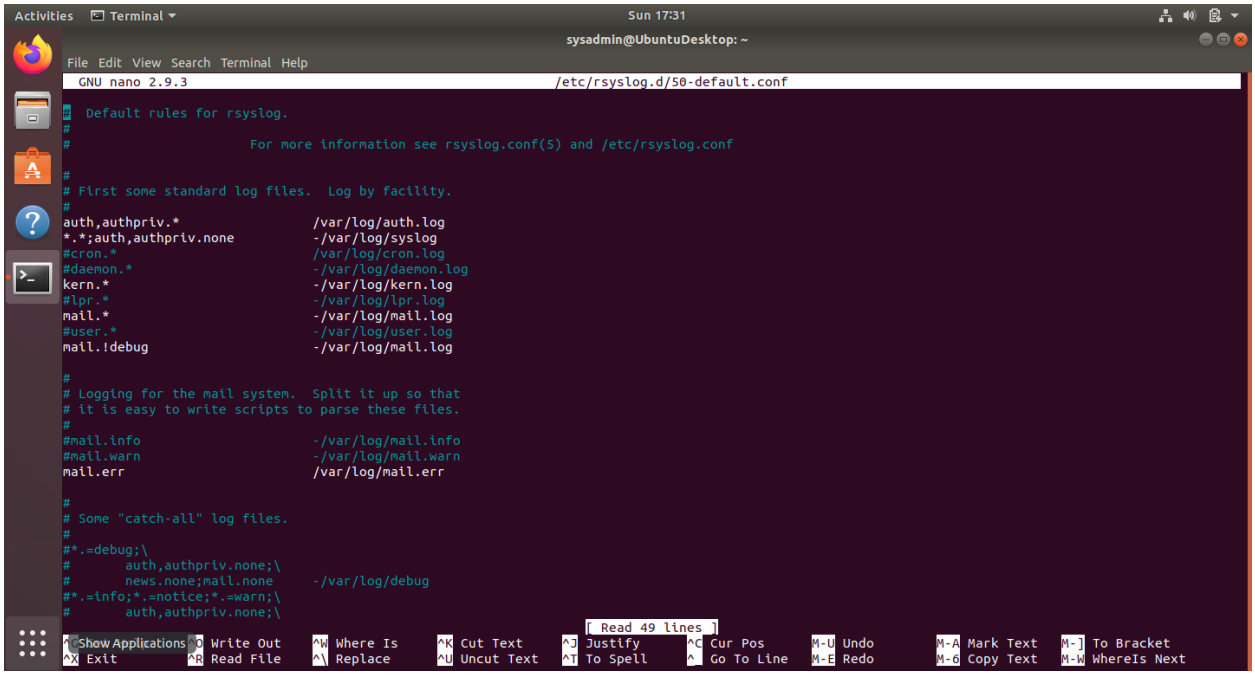
- Command to filter all log messages with priority levels between zero and two, and save output to /home/sysadmin/Priority_High.txt:
- Command to automate the last command in a daily cronjob:
- Add the edits made to the crontab file below:
  [Your solution cron edits here]

Step 5. Create Priority-Based Log Files

1. Command to record all mail log messages, except for debug, to /var/log/mail.log:
   - Add the edits made to the configuration file below:

     mail.!debug -/var/log/mail.log

2. [Your solution edits here]



Bonus

- Command to record all boot log messages, except for info and debug, to /var/log/boot.log:
    - Add the edits made to the configuration file below:
- [Your solution edits here]

Step 6. 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

      rotate 7

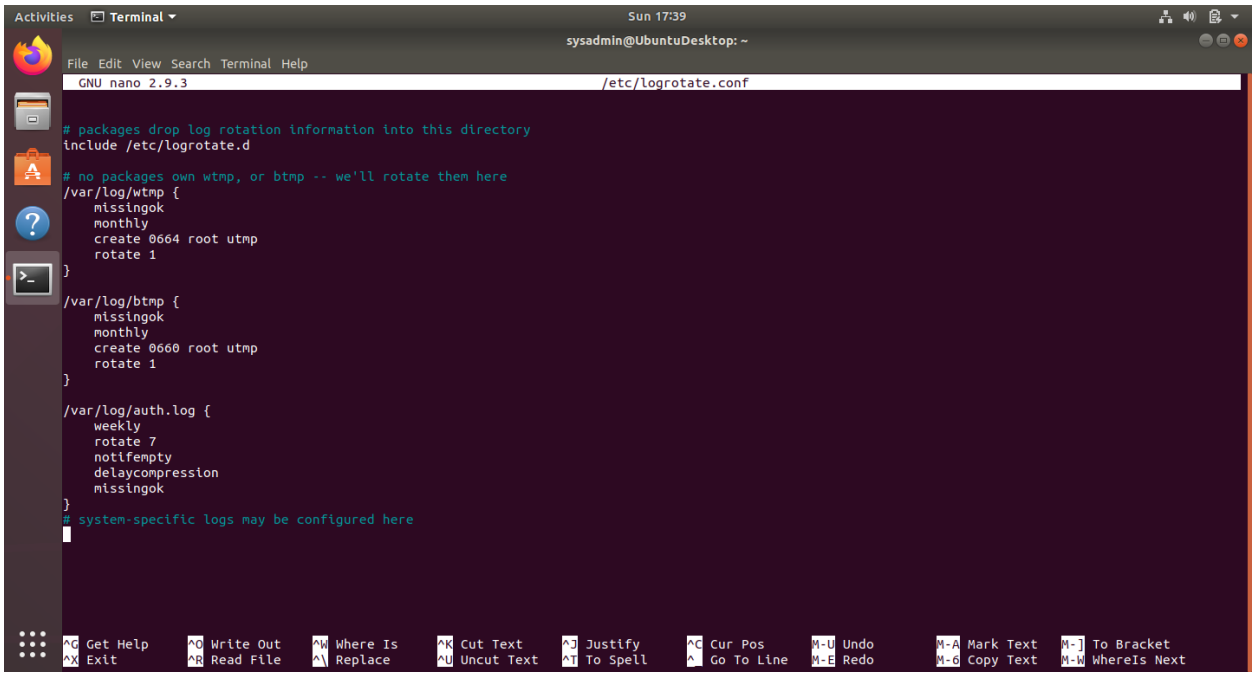
      notifempty

      delaycompression

Missingok

}

2. [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:
3. [Your solution edits here]
4. 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:
5. [Your solution edits here]
6. Command to restart auditd:
7. Command to list all auditd rules:
8. Command to produce an audit report:
9. Create a user with sudo useradd attacker and produce an audit report that lists account modifications:
10. Command to use auditd to watch /var/log/cron:
11. Command to verify auditd rules: