***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 -xvf TarDocs.tar***

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

*excluding the `TarDocs/Documents/Java` directory:*

*$* ***sudo tar --exclude='./Documents/J\*' -cvvWf Javaless\_Docs.tar ./Documents***

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

*$* ***tar tvvf Javaless\_Docs.tar | grep -i 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 with `tar`?*

***The -x extracts an archive, -c creates an archive. You may want to extract the***

***contents of the archive to a specific directory by appending -c switch to the end of the***

***command. It is not applicable to use them together. For instance, tar -xzvf archive.tar.gz***

***-C /tmp will extract the contents of the tar.gz file to the /tmp directory.***

*---*

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

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

***0 6 7 \* 3 tar -czf /var/backups/auth\_backup.tgz /var/log/auth.log >/dev/null 2>&1***

*---*

*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 | awk-F" " '{print $1,$4}' | grep -i Mem > ~/backups/freemem/freemem.txt***

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

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

***du -h > ~/backups/freedisk/free\_disk.txt***

*```*

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

*$* ***chmod 755 system.sh***

*\*\*Optional\*\**

*- Commands to test the script and confirm its execution:*

*$* ***sudo bash system.sh && echo Success || echo Fail***

*or*

*$* ***sudo bash system.sh***

*$* ***cd backups/freemem***

*$* ***ls***

*$* ***cat freemem.txt***

*$* ***cd*** *(to the remaining 3 directories and repeat the steps)*

*\*\*Bonus\*\**

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

*$* ***sudo 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***

***}***

*```*

*---*

*### 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:*

*```bash*

***num\_logs =7***

***max\_log\_file = 35***

*```*

*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*

***-w /etc/shadow -p wra -k hashpass\_audit***

***-w /etc/passwd -p wra -k userpass\_audit***

***-w /var/log/auth.log -p wra -k authlog\_audit***

*4sud. 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*** *(for Summary)*

*$* ***sudo aureport -au*** *(for Authentication)*

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

*modifications:*

*$* ***sudo adduser attacker***

*$* ***sudo aureport -m***

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

***-w /var/log/cron -p rwxa***

*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 -p 0..3***

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

*$* ***sudo journalctl -b --disk-usage***

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

*$* ***sudo journalctl --vacuum-file=2***

*1. 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 > Priority\_High.txt***

*1. Command to automate the last command in a daily cronjob. Add the edits made to the*

*crontab file below:*

*```bash*

***0 5 \* \* \* journalctl -p 0..2 > /home/sysadmin/Priority\_High.txt > /dev/null 2>&1***