| Cybersecurity |
| --- |
| 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:

| tar -xf TarDocs.tar |
| --- |

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

| tar -cf Javaless\_Docs.tar --exclude=Documents/Java -C ~/Projects TarDocs/Documents/ |
| --- |

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

| tar -tf Javaless\_Docs.tar | grep Java |
| --- |

#### Optional

1. 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 --create --gzip --file=logs\_backup.tar.gz --listed-incremental=snapshot.file /var/log |
| --- |

#### Critical Analysis Question

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

| -x tells tar to extract files from archives and read existing archive files and restore them to their original locations. -c tells tar to collect files and directories and put them into an archive file. They are opposites and can cause tar not to work properly if they are in a single line. |
| --- |

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

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

| crontab -e  Add the following command in the editor:  0 6 \* \* 3 tar czf /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} |
| --- |

1. Paste your system.sh script edits:

| #!/bin/bash  # Free memory output to a free\_mem.txt file  free -h > ~/backups/freemem/free\_mem.txt  # Disk usage output to a disk\_usage.txt file  df -h > ~/backups/diskuse/disk\_usage.txt  # List open files to an open\_list.txt file  lsof > ~/backups/openlist/open\_list.txt  # Free disk space to a free\_disk.txt file  df -h > ~/backups/freedisk/free\_disk.txt |
| --- |

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

| chmod +x system.sh |
| --- |

#### 

#### 

#### Optional

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

| ./system.sh  cat ~/backups/freemem/free\_mem.txt  cat ~/backups/diskuse/disk\_usage.txt  cat ~/backups/openlist/open\_list.txt  cat ~/backups/freedisk/free\_disk.txt |
| --- |

#### 

1. 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.

* 1. Add your config file edits:

| GNU nano 4.8 /etc/logrotate.conf Modified  # see "man logrotate" for details  # rotate log files weekly  weekly  # use the adm group by default, since this is the owning group  # of /var/log/syslog.  su root adm  # keep 4 weeks worth of backlogs  rotate 4  # create new (empty) log files after rotating old ones  create  # use date as a suffix of the rotated file  #dateext  notifempty  # uncomment this if you want your log files compressed  #compress  compress  # packages drop log rotation information into this directory  include /etc/logrotate.d  # system-specific logs may be also be configured here.  **/var/log/auth.log {**  **weekly**  **rotate 7**  **notifempty**  **delaycompress**  **missingok**  **}** |
| --- |

### Optional Additional Challenge: Check for Policy and File Violations

1. Command to verify `auditd` is active:

| systemctl status auditd |
| --- |

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

| sudo nano /etc/audit/auditd.conf  Changed max\_log\_file = 35 and num\_logs = 7 |
| --- |

Add the edits made to the configuration file:

| **max\_log\_file = 35**  **num\_logs = 7** |
| --- |

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

| sudo nano /etc/audit/rules.d/audit.rules |
| --- |

Add the edits made to the rules file below:

| -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 |
| --- |

1. Command to restart auditd:

| sudo systemctl restart auditd |
| --- |

1. Command to list all auditd rules:

| sudo auditctl -l |
| --- |

1. Command to produce an audit report:

| sudo aureport --auth |
| --- |

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

| sudo aureport -m |
| --- |

1. Command to use auditd to watch /var/log/cron:

| sudo auditctl -w /var/log/cron -p wra -k cronlog\_audit |
| --- |

1. Command to verify auditd rules:

| sudo auditctl -l |
| --- |

#### 

### Optional (Research Activity): Perform Various Log Filtering Techniques

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

| [Enter answer here] |
| --- |

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

| sudo journalctl --disk-usage |
| --- |

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

| [Enter answer here] |
| --- |

1. 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] |
| --- |

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

| [Enter answer here] |
| --- |

© 2022 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.