| 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 xvvf TarDocs.tar  x = extract  vv = displays the full file specifications  f = filename |
| --- |

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

| #Command an from /home/sysadmin/Projects/  tar --exclude="TarDocs/Documents/Java" -cvf Javaless\_Docs.tar TarDocs/ |
| --- |

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

| tar tvvf 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 --listed-incremental=snapshot.file -cvzf logs\_backup.tar.gz /var/log |
| --- |

#### Critical Analysis Question

1. Why wouldn't you use the options -x and -c at the same time with tar?
   1. The -x option is used to extract contents from an already existing .tar file, whereas the -c option is used to create a .tar archive of a directory. Simply, the -x option can only be used once an archive has been created using the -c option.

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

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

| Ensure to run crontab -e in sudo mode for escalated system-wide privileges.  Command:  #Creating an archive every wednesday at 6AM of /var/log/auth.log  0 6 \* \* 3 tar zcvf /auth\_backup.tgz /var/log/auth.log |
| --- |

### Step 3: Write Basic Bash Scripts

1. Brace expansion command to create the four subdirectories:

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

1. Paste your system.sh script edits:

| #!/bin/bash  #Prints the amount of free memory on the system and saves it to disk\_usage.txt  free -h >> /home/sysadmin/backups/freemem/free\_mem.txt  #Prints disk usage and saves it to disk\_usage.txt  du -h >> /home/sysadmin/backups/diskuse/disk\_usage.txt  # Lists all open files and saves it to open\_list.txt  lsof >> /home/sysadmin/backups/openlist/open\_list.txt  #Prints file system disk space statistics and saves it to free\_disk.txt  df -h >> /home/sysadmin/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:

| sudo ./system.sh  #Lists all Directories/Subdirectories, and all files created from system.sh.  #Will need to be completed within /home/sysadmin/backups/  **OUTPUT**:  diskuse:  disk\_usage.txt  freedisk:  free\_disk.txt  freemem:  free\_mem.txt  openlist:  Open\_list.txt  **Cat Command:**  #Dispalys all contents in output files in subdirectories.  cat /home/sysadmin/backups/openlist/open\_list.txt /home/sysadmin/backups/freedisk/free\_disk.txt /home/sysadmin/backups/freemem/free\_mem.txt /home/sysadmin/backups/diskuse/disk\_usage.txt |
| --- |

#### 

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

| sudo cp /home/sysadmin/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:

| /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:

| sudo systemctl status auditd |
| --- |

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

| sudo nano /etc/audit/auditd.conf |
| --- |

Add the edits made to the configuration file:

| [Enter answer here]# This file controls the configuration of the audit daemon  #  local\_events = yes  write\_logs = yes  log\_file = /var/log/audit/audit.log  log\_group = adm  log\_format = RAW  flush = INCREMENTAL\_ASYNC  freq = 50  **max\_log\_file = 35**  **num\_logs = 7**  priority\_boost = 4  disp\_qos = lossy  dispatcher = /sbin/audispd  name\_format = NONE |
| --- |

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:

| GNU nano 4.8 /etc/audit/rules.d/audit.rules    ## -w = path to file,  ## -p = permissions  ## -k = keyname  ## wra = Write, Read, and Change file attribute.  ## setting the permissions to monitor /etc/shadow & set keynote to hashpass\_aud>  -w /etc/shadow -p wra -k hashpass\_audit  ## Setting the permissions to monitor /etc/passwd & set keynote to userpass\_aud>  -w /etc/passwd -p wra -k userpass\_audit  ## setting the permissions to monitor /var/log/auth.log & set keynote to authlo>  -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 -au |
| --- |

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 |
| --- |

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:

| journalctl -p emerg..err |
| --- |

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

| sudo journalctl -b -u systemd-journald |
| --- |

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

| sudo journalctl --vacuum-time=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 >> /home/sysadmin/Priority\_High.txt |
| --- |

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

| Emerg-Crit.sh  !#/bin/bash  ##Shell script to utilise journactl to filter logs by 0-2 priority and output to Priority\_High.txt file in home folder.  Command to add as a daily cron job.  @daily journalctl -p 0..2 >> /home/sysadmin/Priority\_High.txt  sudo cp /home/sysadmin/Emerg-Crit.sh /etc/cron.daily |
| --- |

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