|  |
| --- |
| 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 -xvf TarDocs.tar -C ~/projects |

1. 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” ~/Projects/TarDocs/cta |

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

|  |
| --- |
| tar -tvf Javaless\_Doc.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:

|  |
| --- |
| tar -cvf 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?

The -c option creates a new archive, while the -x option extracts files from an existing archive. Using both options together does not make sense because it creates conflicting instructions.

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

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

|  |
| --- |
| 0 6 \* \* 3 tar -zcf /auth\_backup.tgz /var/log/auth.log |

### Step 3: Write Basic Bash Scripts

1. Brace expansion command to create the four subdirectories:

|  |
| --- |
| sudo mkdir -p ~/backups/{freemam,openlist,freedisk,diskuse} |

1. Paste your system.sh script edits:

|  |
| --- |
| #Free memory output to a free\_mem.txt file  Free -h > ~/backups/freemem/free\_mem.txt  #Disk useage output to a disk\_usage.txt file  Du -h > ~/backups/diskuse/disk\_usage.txt  #List open files to open\_list.txt file  lsof > ~/backups/openlist/open\_list.txt  #Free command to disk space to a free\_disk.txt file  Df -h > ~/backups/freedisk/free\_disk.txt |

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

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

#### 

#### 

#### Optional

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

|  |
| --- |
| Sudo ./system.sh && 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:

|  |
| --- |
| /var/log/auth.log {  weekly  missingok  rotate 7  notifempty  delaycompress  compress  postrotate  /usr/bin/killall -HUP syslog-ng > /dev/null 2>&1 || true  endscript  } |

### 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:

|  |
| --- |
| 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 shadow\_changes  -w /etc/passwd -p wra -k passwd\_changes  -w /var/log/auth.log -p wra -k authlog\_changes |

1. Command to restart auditd:

|  |
| --- |
| sudo systemctl restart audit |

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 useradd attacker  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:

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

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

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

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:

|  |
| --- |
| [Your solution cron edits here] |

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