**Week 5 Homework Submission File**

# **Linux 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 xf TarDocs.tar**

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

**tar cvf Javaless\_Docs.tar ~/Projects --exclude ~/Projects/TarDocs/Java**

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

**tar tvvf Javaless\_Docs.tar | grep Java \* Javaless\_Docs.tar**

**Bonus**

* Command to create an incremental archive called logs\_backup\_tar.gz with only changed files to snapshot.filefor 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 time with tar?

**You cannot extract something while creating it at the same time, the data isn’t there, you would have to use the && command**

**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:

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

1. Paste your system.sh script edits below:

**#!/bin/bash**

**﻿free -h > ~/backups/freemem/free\_mem.txt**

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

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

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

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

﻿**chmod +x system.sh**

**Optional**

* Commands to test the script and confirm its execution:

﻿**cat disk\_usage.txt**

**﻿cat free\_disk.txt**

**﻿cat free\_disk.txt**

**﻿cat open\_list.txt**

**Bonus**

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

﻿**sysadmin@UbuntuDesktop:/etc/cron.weekly$ sudo cp ~/system.sh .**

**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:

﻿/**var/log/auth.log {**

**weekly**

**rotate 7**

**notifempty**

**delaycompress**

**missingok**

**}**

**Bonus: 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:
   * Add the edits made to the configuration file below:

﻿**GNU nano 2.9.3 /etc/audit/auditd.conf**

**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**

**##name = mydomain**

**max\_log\_file\_action = ROTATE**

**space\_left = 75**

**space\_left\_action = SYSLOG**

**verify\_email = yes**

**action\_mail\_acct = root**

**admin\_space\_left = 50**

**admin\_space\_left\_action = SUSPEND**

**disk\_full\_action = SUSPEND**

**disk\_error\_action = SUSPEND**

**use\_libwrap = yes**

**##tcp\_listen\_port = 60**

**tcp\_listen\_queue = 5**

**tcp\_max\_per\_addr = 1**

**##tcp\_client\_ports = 1024-65535**

**tcp\_client\_max\_idle = 0**

**enable\_krb5 = no**

**krb5\_principal = auditd**

**##krb5\_key\_file = /etc/audit/audit.key**

**distribute\_network = no**

1. 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:

﻿#**watch the following paths:**

**-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:

**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 -w**

1. 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:

**journalctl -b -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. Comand to remove all archived journal files except the most recent two:

**sudo journalctl -p 0..2 > /home/sysadmin/Priority\_High.txt**

1. Command to filter all log messages with priority levels between zero and two, and save output to /home/sysadmin/Priority\_High.txt:

**sudo cp ~/Priority\_High.sh /etc/cron.daily**

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

**#!/bin/bash @daily journalctl -p 0..2 >> ~/home/sysadmin/Priority\_High.txt**