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

From the ~/Projects directory run:

tar cvvW --exclude=Java -f Javaless\_doc.tar TarDocs/Documents

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

tar tvf Javaless\_doc.tar | grep 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 czvvf logs\_backup.tar.gz --listed-incremental=snapshot.file /var/log

#### **Critical Analysis Question**

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

Because -x indicates extracting and -c indicates creating. You would need to do one or the other, but not both at the same time.

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

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

sudo crontab -e

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}

Paste your system.sh script edits below:  
  
 #!/bin/bash

1. 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:

./ system.sh

ls -la backups/\* This will confirm that files were created or appended.

cat free\_mem.txt from ~/backups/freemem directory to view contents of file.

**Bonus**

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

sudo cp system.sh /etc/cron.weekly

### **Step 4: Perform Various Log Filtering Techniques**

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

journalctl -p 3

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

journalctl -b 0 --unit=systemd-journald | less

I got the same result with just: journalctl --unit=systemd-journald

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

sudo journalctl --vacuum-files=2

**Bonus**

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

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

* Command to automate the last command in a daily cronjob:

30 3 \* \* \* journalctl -p 2 > /home/sysadmin/Priority\_High.txt

Or put it in a script and add it to the /etc/cron.daily directory

* Add the edits made to the crontab file below:  
    
   crontab -e

30 3 \* \* \* journalctl -p 2 > /home/sysadmin/Priority\_High.txt

### **Step 5. Create Priority-Based Log Files**

1. Command to record all mail log messages, except for debug, to /var/log/mail.log:

Mail.info -/var/log/mail.log

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

1. Mail.info -/var/log/mail.log

**Bonus**

* Command to record all boot log messages, except for info and debug, to /var/log/boot.log:  
  + Add the edits made to the configuration file below:
* local7.notice /var/log/boot.log

### **Step 6. 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:
2. /var/log/auth.log {

weekly

rotate 7

notifempty

compress

delaycompress

missingok

endscript

}

### **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:

num\_logs = 7

max\_log\_file = 35

1. -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 using auditd to set rules for /etc/shadow, /etc/passwd and /var/log/auth.log:  
   * Add the edits made to the rules file below:

Sudo nano /etc/audit/rules.d/audit.rules

-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. Sudo auditctl -l
2. Command to restart auditd: systemctl restart auditd
3. Command to list all auditd rules: sudo auditctl -l
4. Command to produce an audit report: sudo aureport -au
5. Create a user with sudo useradd attacker and produce an audit report that lists account modifications:

sudo useradd attacker

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