## **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

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

tar cvvWf Javaless\_Docs.tar --exclude='TarDocs/Documents/Java' TarDocs/Documents

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

Tar tvvf Javaless\_Docs | 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:

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?

Because -c creates the archived file whereas -x will extract the listed archived file

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

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

0 6 \* \* 3 tar czf var/backups/auth\_backup.tgz /var/log/auth.log

### **Step 3: Write Basic Bash Scripts**

1. Brace expansion command to create the four subdirectories:

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

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

1. [Your solution script contents here]

#!/bin/bash

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

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

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

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

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

chmod +x system.sh

**Optional**

* Commands to test the script and confirm its execution:

Sudo ./system.sh

**Bonus**

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

sudo mv 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.

# rotate log files weekly

weekly

# use the syslog group by default, since this is the owning group

# of /var/log/syslog.

su root syslog

# keep 4 weeks worth of backlogs

rotate 4

# create new (empty) log files after rotating old ones

create

# uncomment this if you want your log files compressed

compress

# packages drop log rotation information into this directory

include /etc/logrotate.d

# Do not rotate empty logs

notifempty

1. [Your logrotate scheme edits here]

/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 auditid

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

Nano etc/audit/auditd.conf

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

[Your solution edits here]

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:
2. [Your solution edits here]
3. Command to restart auditd:

Systemctl restart auditid

1. Command to list all auditd rules:
2. Command to produce an audit report:
3. Create a user with sudo useradd attacker and produce an audit report that lists account modifications:
4. Command to use auditd to watch /var/log/cron:
5. Command to verify auditd rules:

### **Bonus (Research Activity): Perform Various Log Filtering Techniques**

1. Command to return journalctl messages with priorities from emergency to error:
2. Command to check the disk usage of the system journal unit since the most recent boot:
3. Comand to remove all archived journal files except the most recent two:
4. Command to filter all log messages with priority levels between zero and two, and save output to /home/sysadmin/Priority\_High.txt:
5. Command to automate the last command in a daily cronjob. Add the edits made to the crontab file below:  
     
    [Your solution cron edits here]