## **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:
   1. tar -xvf TarDocs.tar
2. Command to **create** the Javaless\_Doc.tar archive from the TarDocs/ directory, while excluding the TarDocs/Documents/Java directory:
   1. tar -cvf Javaless\_Docs.tar --exclude=“TarDocs/Documents/Java” TarDocs/
3. Command to ensure Java/ is not in the new Javaless\_Docs.tar archive:
   1. tar -tvf Javaless\_Docs.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 --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 wouldn’t use the -x -c at the same time because they conflict with each other. -c is to create and -x is to extract

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

1. Cron job for backing up the /var/log/auth.log file:
   1. **0 6 \* \* 3 tar -cvzf /auth\_backup.tgz /var/log/auth.log**

### 

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

1. Brace expansion command to create the four subdirectories:
   1. sudo mkdir -p ~/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\_usage.txt

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

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

1. Command to make the system.sh script executable:
   1. sudo chmod 755 system.sh

**Optional**

* Commands to test the script and confirm its execution:
  + sudo ./system.sh
  + its green?? ls -l
  + I used cat \* in the subdirectories

**Bonus**

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

COPY OVER

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.  
   * Add your config file edits below:
2. [Your logrotate scheme edits here]
   * /var/log/auth.log {
   * weekly
   * rotate 7
   * notifempty
   * delaycompression
   * missingok
   * }

### 

### 

### 

### 

### 

### 

### **Bonus: Check for Policy and File Violations**

1. Command to verify auditd is active:
   * sudo systemctl status auditd
2. Command to set number of retained logs and maximum log file size:

[Your solution edits here]

Add the edits made to the configuration file below:

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:

[Your solution edits here]

* + 1. -w /etc/shadow -p wra -k hashpass\_audit
    2. -w /etc/passwd -p wra -k userpass\_audit
    3. -w /var/log/auth.log -p wra -k authlog\_audit

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

Account Modifications Report

=================================================

# date time auid addr term exe acct success event

=================================================

1. 03/29/2021 18:51:03 -1 ? ? /usr/sbin/useradd vboxadd no 251

2. 03/29/2021 18:51:03 -1 ? ? /usr/sbin/useradd vboxadd no 252

3. 03/29/2021 18:51:03 -1 ? ? /usr/sbin/useradd vboxadd no 253

4. 03/29/2021 18:51:03 -1 ? ? /usr/sbin/useradd vboxadd no 254

5. 04/01/2021 18:51:57 -1 ? ? /usr/sbin/useradd vboxadd no 248

6. 04/01/2021 18:51:57 -1 ? ? /usr/sbin/useradd vboxadd no 249

7. 04/01/2021 18:51:57 -1 ? ? /usr/sbin/useradd vboxadd no 250

8. 04/01/2021 18:51:57 -1 ? ? /usr/sbin/useradd vboxadd no 251

9. 04/01/2021 22:06:13 1000 UbuntuDesktop pts/0 /usr/sbin/groupadd ? yes 5506

10. 04/01/2021 22:06:13 1000 UbuntuDesktop pts/0 /usr/sbin/groupadd ? yes 5507

11. 04/01/2021 22:06:13 1000 UbuntuDesktop pts/0 /usr/sbin/groupadd ? yes 5508

12. 04/01/2021 22:06:13 1000 UbuntuDesktop pts/0 /usr/sbin/useradd ? yes 5516

13. 04/01/2021 22:06:19 1000 UbuntuDesktop pts/0 /usr/bin/passwd attacker yes 5556

1. Command to use auditd to watch /var/log/cron:
   * w /var/log/cron -p wra
2. Command to verify auditd rules:
   * sudo auditd -l

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

1. Command to return journalctl messages with priorities from emergency to error:
   1. journalctl -ef
2. Command to check the disk usage of the system journal unit since the most recent boot:
   1. sudo journalctl -u systemd-journald
   2. sudo journalctl --disk-usage
3. Command to remove all archived journal files except the most recent two:
   1. sudo journalctl --vacuum-files=2
4. Command to filter all log messages with priority levels between zero and two, and save output to /home/sysadmin/Priority\_High.txt:
   1. sudo journalctl -p emerg >> /home/sysadmin/Priority\_High.txt
   2. sudo journalctl -p alert >> /home/sysadmin/Priority\_High.txt
   3. sudo journalctl -p crit >> /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]
   1. 0 18 \* \* \* sudo journalctl -p emerg >> /home/sysadmin/Priority\_High.txt
   2. 0 18 \* \* \* sudo journalctl -p alert >> /home/sysadmin/Priority\_High.txt
   3. 0 18 \* \* \* sudo journalctl -p crit >> /home/sysadmin/Priority\_High.txt