## Archiving and Logging Data

### Step 1: Create, Extract, Compress, and Manage tar Backup Archives

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:

# see extended history

sysadmin@UbuntuDesktop:~/Projects/TarDocs/Documents$ tar --exclude='./Java' -cvf ~/Projects/Javaless\_Docs.tar ./

./

./Music-Sheets/

./Music-Sheets/Stairway-to-heaven-guitar.pdf

./Music-Sheets/Stairway-to-heaven-bass-tab.pdf

./Music-Sheets/Thumbs.db

./Music-Sheets/Stairway-to-heaven-piano-guitar-A-minor.pdf

./Google-Maps-Hacks/

./Google-Maps-Hacks/googlemapshks-CHP-6.PDF

./Google-Maps-Hacks/googlemapshks-CHP-5.PDF

./Google-Maps-Hacks/googlemapshks-CHP-1.PDF

./Google-Maps-Hacks/googlemapshks-CHP-7.PDF

./Google-Maps-Hacks/googlemapshks-CHP-4.PDF

./Google-Maps-Hacks/googlemapshks-CHP-2.PDF

./Google-Maps-Hacks/googlemapshks-CHP-3.PDF

./IntelliJIDEA\_ReferenceCard.pdf

./Design-Patterns/

./Design-Patterns/Head\_First\_Design\_Patterns\_\_2008\_.pdf

./Design-Patterns/DesignPatterns.pdf

./c++interviewquestions.pdf

./Javaless\_Docs.tar

sysadmin@UbuntuDesktop:~/Projects/TarDocs/Documents$ cd ..

sysadmin@UbuntuDesktop:~/Projects/TarDocs$ ls

Documents Financials Movies Pictures Programs

sysadmin@UbuntuDesktop:~/Projects/TarDocs$ cd ..

sysadmin@UbuntuDesktop:~/Projects$ ls

Javaless\_Docs.tar TarDocs TarDocs.tar

sysadmin@UbuntuDesktop:~/Projects$ tar ft ~/Projects/Javaless\_Docs.tar | grep Java

./Javaless\_Docs.tar

# Java only appears in filename

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

tar ft Javaless\_Docs.tar | grep Java

# supplemental

- Command to create an incremental archive called `logs\_backup\_tar.gz` with only changed files to `snapshot.file` for the `/var/log` directory:

\*\*\* excerpt from screen \*\*\*

sysadmin@UbuntuDesktop:/var/log$ ls lo\*

logs\_backup.snar logs\_backup.tar.gz

\*\*\* (create 1st incremental-full backup w/.snar snapshot and compress the result using gzip) \*\*\*

sysadmin@UbuntuDesktop:/var/log$ sudo tar cvvWf logs\_backup.tar --listed-incremental=logs\_backup.snar --level=0 ./ && sudo gzip logs\_backup.tar

#### Critical Analysis Question

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

x command is to extract contents of existing tar, c option is to create a new tar 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 cvf /auth\_backup.tar /var/log/auth.log | gzip > /auth\_backup.tgz

\*\*\* NOTES \*\*\*

cron jobs run as root. in sudo -s shell test, .tar archive file can be output using gzip to a .tgz file - see below -

root@UbuntuDesktop:/# tar cvf /auth\_backup.tar /var/log/auth.log | gzip > /auth\_backup.tgz

tar: Removing leading `/' from member names

You have new mail in /var/mail/root

root@UbuntuDesktop:/# ls

03-instructor boot git-ctf lib32 opt snap tmp vmlinuz.old

03-student cdrom home lib64 proc splunk usr

auth\_backup.tar cmdCTF initrd.img lost+found root srv vagrant

auth\_backup.tgz dev initrd.img.old media run swapfile var

### Step 3: Write Basic Bash Scripts

1. Brace expansion command to create the four subdirectories:

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

2. Paste your `system.sh` script edits below:

#!/bin/bash

#[Your solution script contents here]

# Free memory output to a free\_mem.txt file

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

#

# Disk usage output to a disk\_usage.txt file

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

#

# List open files to a open\_list.txt file

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

#

# Free disk space to a free\_disk.txt file

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

#

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

chmod +x system.sh

- Commands to test the script and confirm its execution:

\*\*\* from terminal history \*\*\*

796 sh system.sh

797 pwd

798 ls

799 cd backups

800 ls

801 cd diskuse/

802 ls

803 cat disk\_usage.txt

804 cd ..

805 ls

806 cd freedisk

807 ls

808 cat free\_disk.txt

809 cd ..

810 ls

811 cd freemem

812 ls

813 cat free\_mem.txt

814 cd ..

815 ls

816 cd openlist

817 ls

818 cat open\_list.txt

819 history

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

(for whichever is needed) \* from history \*

825 sudo cp system.sh /etc/cron.weekly

(or) 826 sudo cp system.sh /etc/cron.daily

### 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

endscript

}

### Check for Policy and File Violations

1. Command to verify `auditd` is active:

systemctl status auditd

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

- Add the edits made to the configuration file below:

# This file controls the configuration of the audit daemon

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

3. Command using `auditd` to set rules for `/etc/shadow`, `/etc/passwd` and `/var/log/auth.log`:

-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

4. Command to restart `auditd`:

sysadmin@UbuntuDesktop:/var/log$ sudo systemctl restart auditd

5. Command to list all `auditd` rules:

sysadmin@UbuntuDesktop:/var/log$ sudo auditctl -l

-w /etc/shadow -p rwa -k hashpass\_audit

-w /etc/passwd -p rwa -k userpass\_audit

-w /var/log/auth.log -p rwa -k authlog\_audit

6. Command to produce an audit report:

sudo aureport -au

7. Create a user with `sudo useradd attacker` and produce an audit report that lists account modifications:

sysadmin@UbuntuDesktop:~$ sudo useradd attacker

sysadmin@UbuntuDesktop:~$ sudo aureport -m

Account Modifications Report

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

# date time auid addr term exe acct success event

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

1. 07/27/2021 18:59:52 1000 UbuntuDesktop pts/1 /usr/sbin/groupadd ? yes 383

2. 07/27/2021 18:59:52 1000 UbuntuDesktop pts/1 /usr/sbin/groupadd ? yes 384

3. 07/27/2021 18:59:52 1000 UbuntuDesktop pts/1 /usr/sbin/groupadd ? yes 385

4. 07/27/2021 18:59:52 1000 UbuntuDesktop pts/1 /usr/sbin/useradd ? yes 386

5. 07/27/2021 18:59:56 1000 UbuntuDesktop pts/1 /usr/bin/passwd hacker yes 388

6. 07/27/2021 19:01:05 1000 UbuntuDesktop pts/1 /usr/sbin/usermod hacker yes 426

7. 07/27/2021 19:01:05 1000 UbuntuDesktop pts/1 /usr/sbin/usermod hacker yes 427

8. 07/27/2021 19:09:35 1000 UbuntuDesktop pts/1 /usr/sbin/groupadd ? yes 514

9. 07/27/2021 19:09:35 1000 UbuntuDesktop pts/1 /usr/sbin/groupadd ? yes 515

10. 07/27/2021 19:09:35 1000 UbuntuDesktop pts/1 /usr/sbin/groupadd ? yes 516

11. 07/27/2021 19:09:35 1000 UbuntuDesktop pts/1 /usr/sbin/useradd ? yes 517

12. 07/27/2021 19:09:39 1000 UbuntuDesktop pts/1 /usr/bin/passwd badguy yes 519

13. 07/27/2021 20:45:37 1000 UbuntuDesktop pts/1 /usr/sbin/useradd criminal yes 1534

14. 07/27/2021 20:45:37 1000 UbuntuDesktop pts/1 /usr/sbin/useradd ? yes 1535

15. 07/29/2021 08:48:47 -1 ? ? /usr/sbin/useradd vboxadd no 314

16. 07/29/2021 08:48:47 -1 ? ? /usr/sbin/useradd vboxadd no 315

17. 07/29/2021 08:48:47 -1 ? ? /usr/sbin/useradd vboxadd no 316

18. 07/29/2021 08:48:47 -1 ? ? /usr/sbin/useradd vboxadd no 317

19. 07/29/2021 15:36:48 1000 UbuntuDesktop pts/0 /usr/sbin/useradd attacker yes 150812

20. 07/29/2021 15:36:48 1000 UbuntuDesktop pts/0 /usr/sbin/useradd ? yes 150816

8. Command to use `auditd` to watch `/var/log/cron`:

sysadmin@UbuntuDesktop:~$ auditctl -w /var/log/cron

You must be root to run this program.

sysadmin@UbuntuDesktop:~$ sudo auditctl -w /var/log/cron

9. Command to verify `auditd` rules:

sysadmin@UbuntuDesktop:~$ auditctl -l

You must be root to run this program.

sysadmin@UbuntuDesktop:~$ sudo auditctl -l

-w /etc/shadow -p rwa -k hashpass\_audit

-w /etc/passwd -p rwa -k userpass\_audit

-w /var/log/auth.log -p rwa -k authlog\_audit

-w /var/log/cron -p rwxa

### Research Activity: Perform Various Log Filtering Techniques

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

sysadmin@UbuntuDesktop:~$ sudo journalctl --priority=3

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

sysadmin@UbuntuDesktop:/var/log/journal$ journalctl --disk-usage

Archived and active journals take up 896.0M in the file system.

# or alternatively

sysadmin@UbuntuDesktop:~$ journalctl -b | grep systemd-journald

Jul 29 08:48:26 UbuntuDesktop systemd-journald[229]: Journal started

Jul 29 08:48:26 UbuntuDesktop systemd-journald[229]: Runtime journal (/run/log/journal/e5853fe375964d39b27025eb6608e969) is 4.9M, max 39.3M, 34.4M free.

Jul 29 08:48:26 UbuntuDesktop systemd-journald[229]: Time spent on flushing to /var is 159.235ms for 492 entries.

Jul 29 08:48:26 UbuntuDesktop systemd-journald[229]: System journal (/var/log/journal/e5853fe375964d39b27025eb6608e969) is 176.0M, max 3.9G, 3.7G free.

Jul 29 16:03:21 UbuntuDesktop sudo[12029]: sysadmin : TTY=pts/0 ; PWD=/home/sysadmin ; USER=root ; COMMAND=/bin/journalctl systemd-journald

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

\*\*\* from https://unix.stackexchange.com/questions/139513/how-to-clear-journalctl - "You don't typically clear the journal yourself. That is managed by systemd itself and old logs are rotated out as new data comes in. The correct thing to do would be to schedule journald to only keep as much data as you are interested in. The most usual thing to adjust is the total disk space it is allowed to take up. Once it crosses this boundry it will start pitching old entries to stay near this value."

# GNU nano 2.9.3 /etc/systemd/journald.conf # Modified

# This file is part of systemd.

# systemd is free software; you can redistribute it and/or modify it

# under the terms of the GNU Lesser General Public License as published by

# the Free Software Foundation; either version 2.1 of the License, or

# (at your option) any later version.

# Entries in this file show the compile time defaults.

# You can change settings by editing this file.

# Defaults can be restored by simply deleting this file.

# See journald.conf(5) for details.

[Journal]

Storage=auto

#Compress=yes

#Seal=yes

#SplitMode=uid

#SyncIntervalSec=5m

#RateLimitIntervalSec=30s

#RateLimitBurst=1000

#SystemMaxUse=

#SystemKeepFree=

#SystemMaxFileSize=

SystemMaxFiles=2

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

(works for system user - root)

journalctl -p 0..2 > /home/student/Priority\_High.txt

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

sudo crontab -e

@daily journalctl -p 0..2 > /home/student/Priority\_High.txt