Cybersecurity

Module 5 Challenge Submission File

Archiving and Logging Data

Make a copy of this document to work in, and then for each step, add the solution command below the prompt. Save and submit this completed file as your Challenge deliverable.

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

1. Command to **extract** the `TarDocs.tar` archive to the current directory:

```
tar xvf TarDocs.tar
```

2. Command to **create** the Javaless_Doc.tar archive from the TarDocs/ directory, while excluding the `TarDocs/Documents/Java` directory:

```
tar -cvf Javaless_Docs.tar --exclude=TarDocs/Documents/Java
TarDocs/Documents
```

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

```
ls -l Javaless_Docs.tar | grep Java
```

Optional

4. Command to create an incremental archive called `logs_backup.tar.gz` with only changed files to `snapshot.file` for the `/var/log` directory:

© 2020 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.

```
sudo tar cvvWf logs_backup.tar --listed-incremental=snapshot.file /var/log

gzip logs_backup.tar
```

Critical Analysis Question

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

```
Using the -c option with tar creates a new archive, while the -x option
extracts files from an existing archive. These options are not used together
because they represent different actions (creating an archive versus
extracting files from one). They are mutually exclusive and would create a
conflict if we tried to use them together.
```

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:

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

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

```
#!/bin/bash

free -h > ~/backups/freemem/free_mem.txt

du -h > ~/backups/diskuse/disk_usage.txt
```

© 2020 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.

```
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

4. Commands to test the script and confirm its execution:

```
bash system.sh
./system.sh
```

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

```
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`.

   a. Add your config file edits:

© 2020 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.

```
/var/log/auth.log {
weekly
rotate 7
notifempty
delaycompress
missingok
}
```

Optional Additional Challenge: 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:

```
sudo nano /etc/audit/auditd.conf
```

Add the edits made to the configuration file:

```
max_log_file = 35
num_logs = 7
```

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

```
sudo nano /etc/audit/rules.d/audit.rules
```

Add the edits made to the `rules` file below:

```
-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
```

© 2020 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.

4. Command to restart `auditd`:

```
sudo systemctl restart auditd
```

5. Command to list all `auditd` rules:

```
sudo auditctl -l
```

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:

```
sudo aureport -m
```

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

```
sudo auditctl -w /var/log/cron
```

9. Command to verify `auditd` rules:

```
sudo auditctl -l
```

Optional (Research Activity): Perform Various Log Filtering Techniques

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

```
sudo journalctl -b -p "emerg".."err"
```

© 2020 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.

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

```
journalctl --disk-usage
```

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

```
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`:

```
From /home/sysadmin run the following:
sudo journalctl -p 0..2 > Priority_High.txt
```

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

```
crontab -e
0 0 * * * journalctl -p 0..2 > /home/sysadmin/Priority_High.txt
```

© 2022 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.

© 2020 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.