1. Q1. Compress all `.log` files older than 7 days from `/var/log/app/` and store them in `/backup/logs/`.

* Solution:

find /var/log/app/ -type f -name "\*.log" -mtime +7 -exec tar -rvf /backup/logs/old\_logs.tar {} \; && gzip /backup/logs/old\_logs.tar

Explanation: Finds .log files older than 7 days, archives them, and compresses the output.

1. Q2. Restore `/etc/config.conf` from `/backup/etc/config.conf` with ownership as `root` and permissions `644`.

* Solution:

cp /backup/etc/config.conf /etc/config.conf && chown root:root /etc/config.conf && chmod 644 /etc/config.conf

Explanation: Restores the file with correct ownership and permissions.

1. Q3. In `/etc/app/config.txt`, change `ENV=dev` to `ENV=prod` without opening the file (use `sed` or `awk`).

* Solution:

sed -i 's/ENV=dev/ENV=prod/' /etc/app/config.txt

Explanation: Replaces the value without opening the file.

1. Q4. Convert `web.conf` from DOS to UNIX format.

* Solution:

dos2unix web.conf # or sed -i 's/\r$//' web.conf

Explanation: Converts DOS line endings to UNIX format.

1. Q5. Display the kernel version and OS release details.

* Solution:

uname -r && cat /etc/os-release

Explanation: Displays kernel and OS version details.

1. Q6. List three commands to monitor CPU, memory, and top-consuming processes in real-time.

* Solution:

top  
vmstat 1  
free -h

Explanation: Monitors CPU, memory, and processes in real-time.

1. Q7. Create a user `murali` with home directory `/home/murali`, group `developers`, and password never expires.

* Solution:

useradd -m -d /home/murali -G developers murali && passwd murali && chage -M 99999 murali

Explanation: Creates a user, sets home and group, and disables password expiry.

1. Q8. Find all users with a valid login shell (not `/sbin/nologin`).

* Solution:

grep -v '/sbin/nologin' /etc/passwd | awk -F: '{print $1, $7}'

Explanation: Lists users with valid login shells.

1. Q9. Your app can’t connect to `10.0.0.5:3306`. Check connectivity, verify port, and trace the route.

* Solution:

ping -c 4 10.0.0.5 && nc -zv 10.0.0.5 3306 && traceroute 10.0.0.5

Explanation: Checks connectivity, verifies port, and traces the network route.

1. Q10. Write a bash script to ping servers from `servers.txt` and log reachable servers in `success.log` and unreachable ones in `failed.log`.

* Solution:

#!/bin/bash  
while read host; do  
 if ping -c 2 $host &>/dev/null; then  
 echo "$host is reachable" >> success.log  
 else  
 echo "$host is unreachable" >> failed.log  
 fi  
done < servers.txt

Explanation: Script pings servers from a list and logs results.

1. Q11. Find all files owned by `root` in `/etc` with `777` permissions.

* Solution:

find /etc -type f -user root -perm 777

Explanation: Lists insecure root-owned files.

1. Q12. Extract all failed login attempts from system logs.

* Solution:

grep 'Failed password' /var/log/auth.log # or journalctl -u sshd | grep 'Failed'

Explanation: Displays failed SSH login attempts.

1. Q13. A Java process consumes high memory. Find it, check memory usage, and kill it gracefully.

* Solution:

ps aux | grep java && top -p <PID> && kill -15 <PID>

Explanation: Finds and gracefully kills high-memory Java process.

1. Q14. Check disk utilization and find which directory under `/var` consumes the most space.

* Solution:

df -h && du -h /var | sort -hr | head -n 10

Explanation: Shows disk utilization and top space-consuming directories.

1. Q15. Write a shell script to back up `/var/www/html` to `/backup/html`, keep only last 5 backups.

* Solution:

#!/bin/bash  
SRC="/var/www/html"  
DEST="/backup/html"  
DATE=$(date +%F\_%H-%M-%S)  
mkdir -p $DEST  
tar -czf $DEST/html\_backup\_$DATE.tar.gz $SRC  
ls -tp $DEST/html\_backup\_\*.tar.gz | tail -n +6 | xargs -I {} rm -- {}

Explanation: Creates timestamped backups and keeps only last 5.