📄 Network & System Tools + Compression & Archiving + Text Process Tools Assignment

---

🎯 Assignment Objective

You will simulate a real-world Linux system administrator or DevOps engineer role by:
✅ Setting up user and group permissions
✅ Running real-time network & system diagnostics
✅ Performing data compression and decompression
✅ Using powerful text processing tools like `grep` and `awk` in combination.

You must **capture and document** command outputs as if working in a production or staging server environment.

🏢 Industry Scenario

You are a junior system administrator in a company called **CloudOps Ltd.**.
Your manager has asked you to:

1. Set up proper **user and group permissions** for the network team.
2. Run **network and system diagnostics** to check connectivity and performance.
3. Archive and compress **log files** for backup.
4. Use `grep` and `awk` to **extract meaningful data** from logs.

📌 Assignment Tasks

Part 1: User & Group Permissions

✅ Task 1.1 — Create users & groups

- Create a group `network_team`.
- Create two users `alice` and `bob`.
- Add both users to the `network_team` group.

```
sudo groupadd network_team
sudo useradd -m -G network_team alice
sudo useradd -m -G network_team bob
```

✅ Task 1.2 — Set directory permissions

- Create a shared directory `/opt/network_data`.
- Set group ownership to `network_team`.
- Give group members **read/write/execute** access.

```
sudo mkdir /opt/network_data
sudo chown root:network_team /opt/network_data
sudo chmod 770 /opt/network_data
```

✅ Expected Output:

Run `ls -ld /opt/network_data` and capture permissions:

```
drwxrwx--- 2 root network_team 4096 May 10 10:00 /opt/network_data
```

Part 2: Network Tools & Real-Time Checks

✅ Task 2.1 — Check connectivity to google.com

- Use `ping`, `traceroute`, and `mtr`:

```
ping -c 4 google.com
traceroute google.com
mtr --report google.com
```

✅ Expected Output Example:

```
PING google.com (142.250.72.238): 56 data bytes
64 bytes from 142.250.72.238: icmp_seq=0 ttl=117 time=14.2 ms
...
```

✅ Task 2.2 — Check open ports & listening services

- Use `netstat` and `ss`:

```
sudo netstat -tuln
sudo ss -tulwn
```

✅ Expected Output Example:

```
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:22              0.0.0.0:*
LISTEN
```

✅ Task 2.3 — Test remote port connectivity

- Use `telnet` or `nc` to check if port 443 is open:

```
telnet google.com 443
nc -zv google.com 443
```

✅ Expected Output Example:

```
Connection to google.com 443 port [tcp/https] succeeded!
```

✅ Task 2.4 — Check network interfaces

- Use `ifconfig` or `ip addr`:

```
ifconfig
ip addr
```

✅ Task 2.5 — DNS lookup

- Use `nslookup` and `dig`:

```
nslookup google.com
dig google.com
```

✅ Task 2.6 — Download test file

- Use `wget` and `curl`:

```
wget https://example.com/testfile.txt
curl -O https://example.com/testfile.txt
```

✅ Task 2.7 — Monitor bandwidth in real time

- Use `iftop` or `nload` (requires sudo):

```
sudo iftop -i eth0
sudo nload eth0
```

Part 3: Compression & Decompression

✅ Task 3.1 — Archive directory

- Create a `.tar` archive:

```
tar cvf network_data.tar /opt/network_data
```

✅ Task 3.2 — Compress archive

- Use `gzip`:

```
gzip network_data.tar
```

✅ Task 3.3 — Decompress

- Use `gunzip`:

```
gunzip network_data.tar.gz
```

✅ Task 3.4 — Use bzip2 compression

```
bzip2 network_data.tar
bunzip2 network_data.tar.bz2
```

✅ Expected Output Example:

- Run `ls -lh` to show compressed file sizes.

Part 4: Text Processing with `grep` & `awk`

✅ Task 4.1 — Search for “error” in log files

```
grep "error" /var/log/syslog
```

✅ Task 4.2 — Count how many errors found

```
grep -c "error" /var/log/syslog
```

✅ Task 4.3 — Extract specific fields (timestamps, messages)

```
grep "error" /var/log/syslog | awk '{print $1, $2, $3, $5}'
```

✅ Task 4.4 — Combine commands to filter and summarize

- Example: Find unique error sources:

```
grep "error" /var/log/syslog | awk '{print $5}' | sort | uniq -c | sort -nr
```

✅ Expected Output Example:

```
15 kernel:
7 NetworkManager:
3 systemd:
```

📤 What to Submit

✅ A **single document** (Markdown or Word) with:

- Commands you ran (copied)
- Screenshots or copied **real-time outputs**
- A **brief explanation** (1-2 sentences) for each step

✅ Upload on GitHub and Submit via: Google Form

⏱ Deadline

Submission due date: 5 days from assignment date.