

# LFCS Mock Exam

Generated by ChatGPT

This mock exam is designed to test your knowledge across all LFCS topics: Operations, Networking, Storage, Essential Commands, and Users & Groups.

---

## Instructions

- Complete the exercises without looking at the answers.
  - The answers are provided at the end of the document.
- 

## Part 1: Operations & Deployment

1. Configure a system to automatically load a kernel module `dummy_module` at boot.
  2. Identify a device connected via PCI and check if a kernel module is loaded for it.
  3. Create a cron job to run `/usr/local/bin/backup.sh` every day at 2:30 AM.
  4. Check if the system is vulnerable to the Meltdown CPU flaw.
- 

## Part 2: Networking

1. Assign a static IP address `192.168.50.10/24` with gateway `192.168.50.1` to interface `eth0`.
  2. Configure the system to synchronize time with `pool.ntp.org`.
  3. Allow SSH connections on port 2222 through the firewall.
  4. Add a static route to reach network `10.10.0.0/16` via gateway `192.168.50.254`.
- 

## Part 3: Storage

1. Create a 10 GB LVM logical volume named `lv_data` in volume group `vg01`.

2. Format it with `ext4` and mount it on `/mnt/data`.
  3. Create a 2 GB swap file and enable it.
  4. Mount an NFS share `server:/share` on `/mnt/nfs` automatically at boot.
- 

## Part 4: Essential Commands

1. Check which process is consuming the most CPU and kill it.
  2. Find the 10 largest files in `/var/log`.
  3. Generate a self-signed SSL certificate valid for 365 days.
  4. Clone a Git repository and push a file named `README.md` after adding a new line.
- 

## Part 5: Users & Groups

1. Create a new user `developer1` with bash shell and home directory.
  2. Add this user to a group `developers`.
  3. Set the maximum number of open files for `developer1` to 4096.
  4. Set ACL so that `developer1` has read/write access to `/var/www/html/index.html` without changing the file's group ownership.
- 

# Answers (Do not read until done!)

## Part 1: Operations & Deployment

1. `echo "dummy_module" | sudo tee /etc/modules-load.d/dummy_module.conf`
2. `lspci -v` and look for `kernel driver in use`
3. `crontab -e` → `30 2 * * * /usr/local/bin/backup.sh`
4. `grep bugs /proc/cpuinfo` or `cat /proc/cpuinfo | grep -i meltdown`

# Part 2: Networking

1. `sudo nmcli con mod eth0 ipv4.addresses 192.168.50.10/24` `sudo nmcli con mod eth0 ipv4.gateway 192.168.50.1` `sudo nmcli con mod eth0 ipv4.method manual` `sudo nmcli con up eth0`
2. `sudo apt install chrony` `sudo nano /etc/chrony/chrony.conf` → add `server pool.ntp.org iburst` `sudo systemctl restart chrony`
3. `sudo firewall-cmd --permanent --add-port=2222/tcp` `sudo firewall-cmd --reload`
4. `sudo ip route add 10.10.0.0/16 via 192.168.50.254`

# Part 3: Storage

1. `sudo lvcreate -L 10G -n lv_data vg01`
2. `sudo mkfs.ext4 /dev/vg01/lv_data` `sudo mkdir -p /mnt/data` `sudo mount /dev/vg01/lv_data /mnt/data`
3. `sudo fallocate -l 2G /swapfile` `sudo chmod 600 /swapfile` `sudo mkswap /swapfile` `sudo swapon /swapfile`
4. `/etc/fstab` entry: `server:/share /mnt/nfs nfs defaults 0 0`

# Part 4: Essential Commands

1. `top` or `ps aux --sort=-%cpu | head` → `kill <PID>`
2. `du -ah /var/log | sort -rh | head -10`
3. `openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout server.key -out server.crt`
4. `git clone <url>` `echo "new line" >> README.md` `git add README.md` `git commit -m "Add new line"` `git push origin main`

# Part 5: Users & Groups

1. `sudo useradd -m -s /bin/bash developer1` `sudo passwd developer1`
2. `sudo groupadd developers` `sudo usermod -aG developers developer1`
3. `ulimit -n 4096` (temporary) or add to `/etc/security/limits.conf`: `developer1 hard nofile 4096`
4. `setfacl -m u:developer1:rw /var/www/html/index.html`

---

Revision #3

Created 2025-11-07 10:40:54 UTC by Loïc

Updated 2025-11-07 10:43:53 UTC by Loïc