**LAB 2: Package & Service Management**

This lab sheet is designed to help you practice using Ubuntu commands related to package management (APT) and service management (systemctl). You will perform hands-on tasks, solve scenarios, and answer questions using the provided commands.

# Task 1: Package Management (APT)

Scenario: You are managing a server and need to ensure that software packages are up-to-date and properly maintained.

1. Update the package database of your system.

sudo apt update

2. List the packages.

apt list unzip

apt list --installed

3. Search for packages containing the keyword 'ssh'.

apt search ssh

4. View detailed information about the package 'unzip'.

apt show unzip

5. Install the package 'unzip'.

apt install unzip

6. Install a specific version of unzip, for example version 6.0-28ubuntu4.

apt install unzip=6.0-28ubuntu4

7. Check if 'unzip' has an upgrade available.

apt list --upgradable | grep unzip

8. Upgrade 'unzip' to the latest version.

apt install unzip

9. Remove the package 'unzip' but keep configuration files.

apt remove unzip

10. Completely remove 'unzip' including its configuration files.

apt purge unzip

# Task 2: Service Manager (systemctl)

Scenario: You are the system administrator and need to check and manage system services to ensure they are running properly.

1. Check which program is running as PID 1 (the init system).

ps -p 1 -o comm=

2. View the overall system status.

Command. systemctl status

3. Check the status of the 'ssh' service.

Command. systemctl status ssh

4. Verify if the 'cron' service is enabled on boot.

Command. systemctl is-enabled cron

5. Check if the 'cron' service is currently running.

Command. systemctl is-active cron

6. Restart the 'cron' service.

Command. systemctl restart cron

7. Disable the 'cron' service and verify the status.

Command. systemctl disable cron && systemctl is-enabled cron

8. Reboot the system.

Command. reboot

9. Enable the 'cron' service and start it immediately.

Command. systemctl enable cron && systemctl start cron

## Conclusion

By completing this lab, you have practiced using APT commands to manage packages and systemctl commands to manage services on Ubuntu. These skills are essential for system administrators to maintain secure and stable systems.

**Exercise**

Define Your Own Technology Stack and Prepare the Server Environment. You are required to **design and select a technology stack** for developing a web-based application of your choice. The application should include at least the following components:

1. **Web server or reverse proxy** (e.g., Nginx, Apache).
2. **Frontend framework or library** (e.g., React, Angular, Vue.js, Svelte).
3. **Backend framework** (e.g., FastAPI, Django, Express.js, Spring Boot, Ruby on Rails).
4. **Database system** (e.g., MySQL, PostgreSQL, MongoDB, Redis).

**Instructions:**

1. Define your technology stack clearly. Write down the reasons for choosing each component (frontend, backend, database, web server).

Web Server : Nginx เพราะ ใช้ resource น้อย

Frontend framework : React เพราะ มีความยืดหยุ่นสูง มี library เยอะ

Backend framework : Express.js เพราะ เรียนรู้ง่าย และมีความยืดหยุ่นสูง ใช้ javascript

Database system : MySQL เพราะ ใช้งานง่าย และฟร

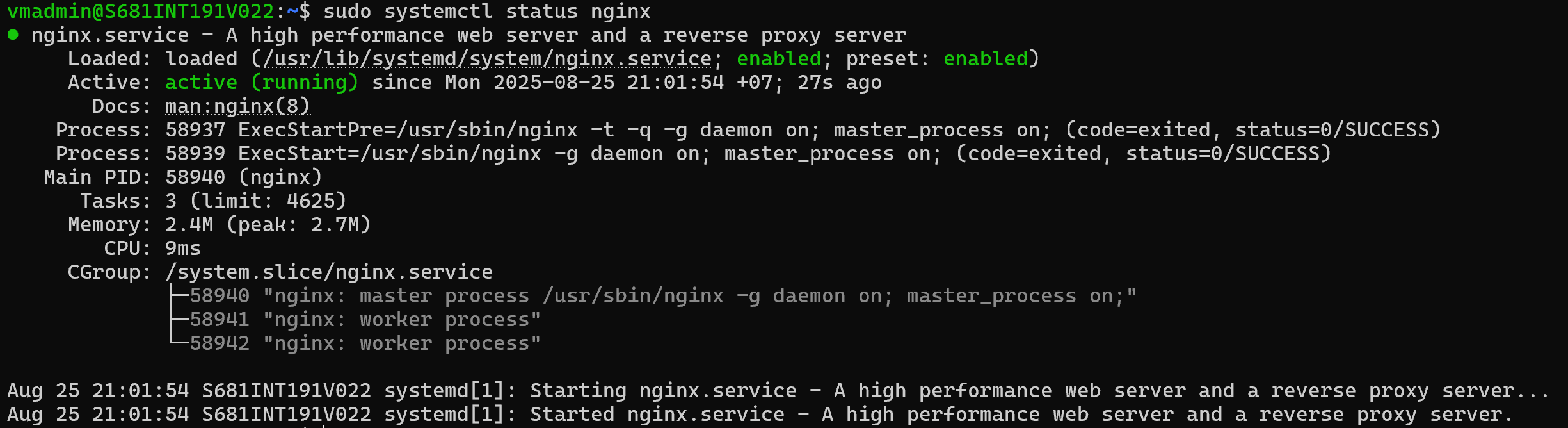
1. On your assigned Linux server, **prepare the environment** by installing all necessary services and packages for your chosen stack.
   * Use apt install to install the required packages.
   * Use systemctl to check, start, and enable services.
2. Verify that each service is correctly installed and running.
   * Check with systemctl status <service\_name>.
   * Confirm using systemctl is-active and systemctl is-enabled.
3. Submit a short report that includes:
   * The chosen technology stack.
   * All commands used for setting up each component.
   * Screenshots or logs showing the installation and running status of each service.
   * A reflection (3–5 sentences) explaining what you learned about managing server services.

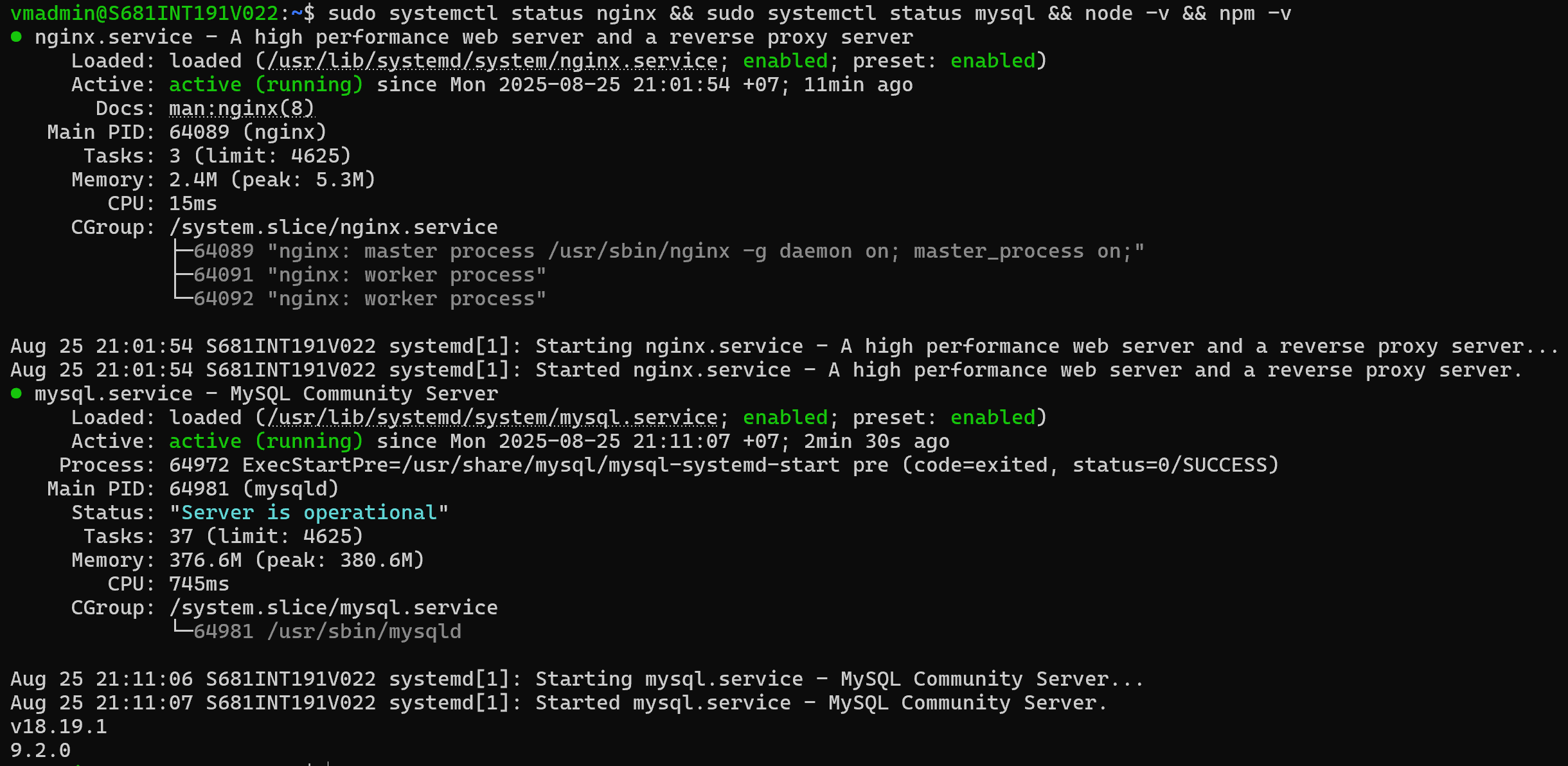
Install Nginx command : sudo apt install nginx

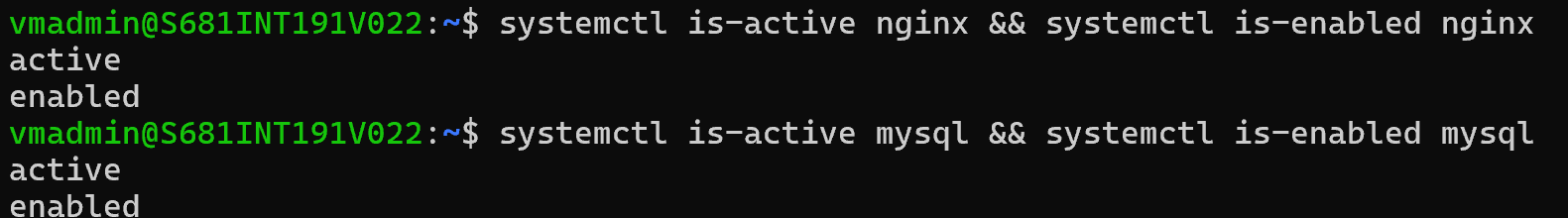
Install react and express.js : sudo apt install nodejs npm

Install MySQL : sudo apt install mysql-server

Reflex : หลังจากที่ได้ทำlab นี้ก็มีความรู้ความเข้าใจเรื่องการจัการแพคเกจและเซร์วิสในฐานะแอดมินมากขึ้น ได้ฝึกแก้ปัญหาต่างๆที่ได้เจอขณะติดตั้ง มีปัญหาเรื่องที่vm ใช้พอร์ต 80 ไม่ได้ จึงต้องมีการแก้ไขใน config ของ niginxอีกด้วย

****

****

****