Linux Administration Assignment – Documentation

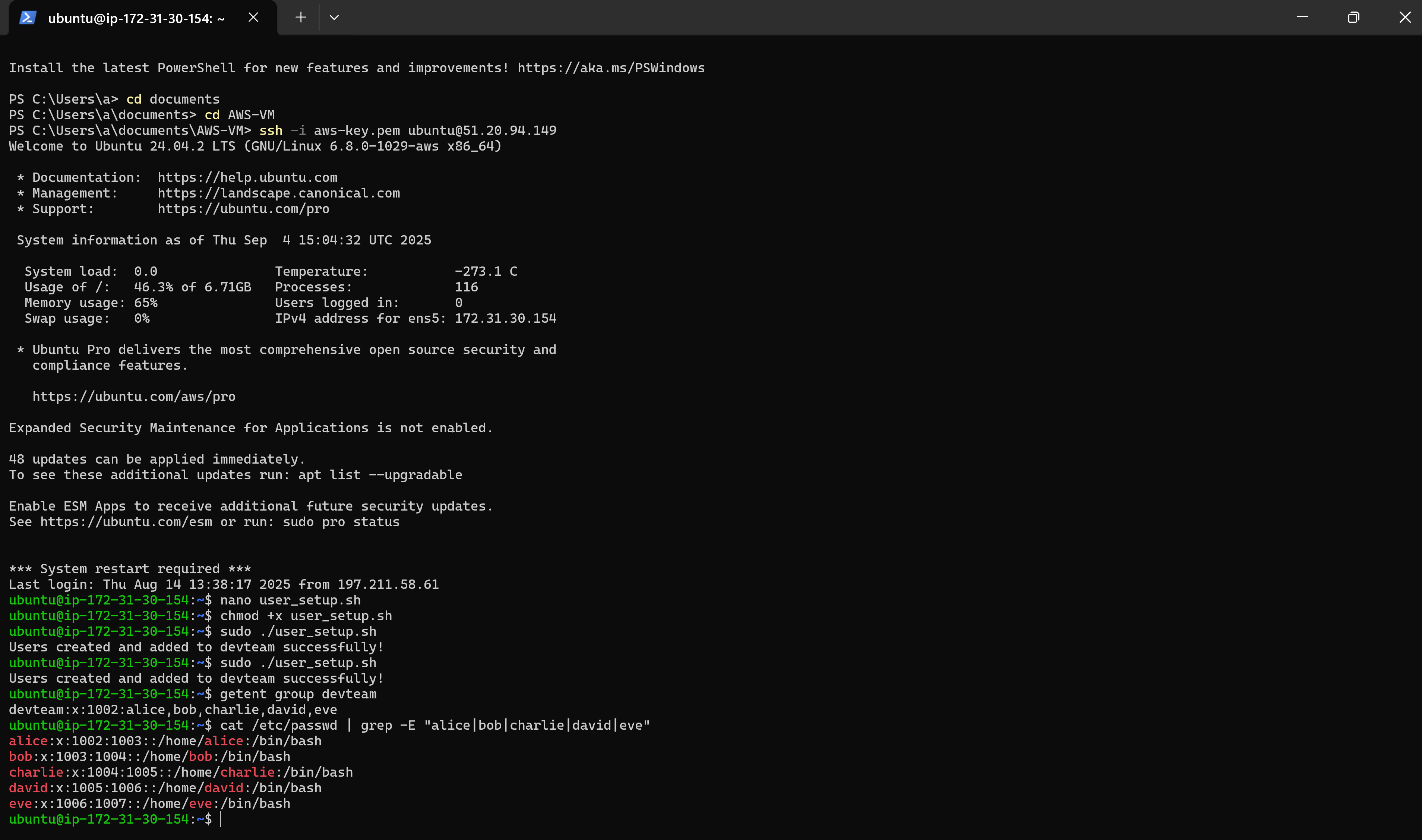
# Task 1: User & Group Management Automation

**Description:**

* Created new users and groups using a shell script.
* Assigned users to groups and set default passwords.
* Configured password expiration policies.

**Steps:**

1. Created a script to automate user and group creation (useradd, groupadd).
2. Assigned users to specific groups with usermod -aG.
3. Applied password policies using chage.

.  


# Task 2: File Permissions & ACLs Project

**Description:**

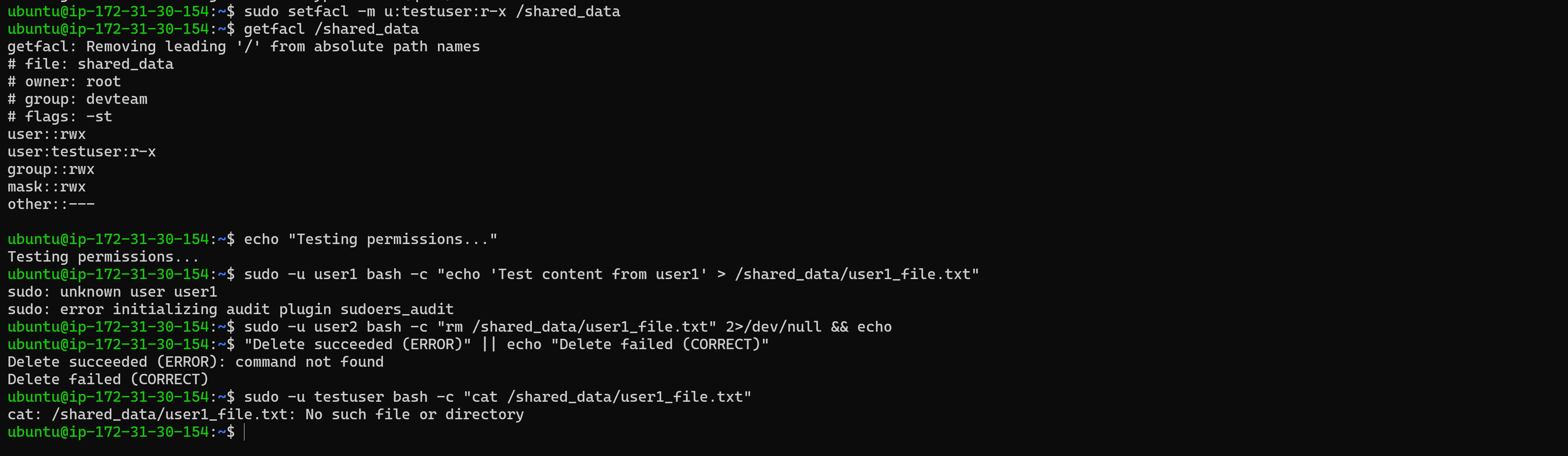
* Configured file permissions using chmod.
* Applied Access Control Lists (ACLs) with setfacl to give fine-grained access.

**Steps:**

1. Created shared directories for groups.
2. Applied group ownership with chown.
3. Used setfacl -m user:username:rwx to assign special permissions.

**Verification:**

* Checked permissions with ls -l and getfacl.
* Confirmed users could access/modify files according to ACLs.

.  


# Task 3: Apache Virtual Hosts Setup

**Description:**

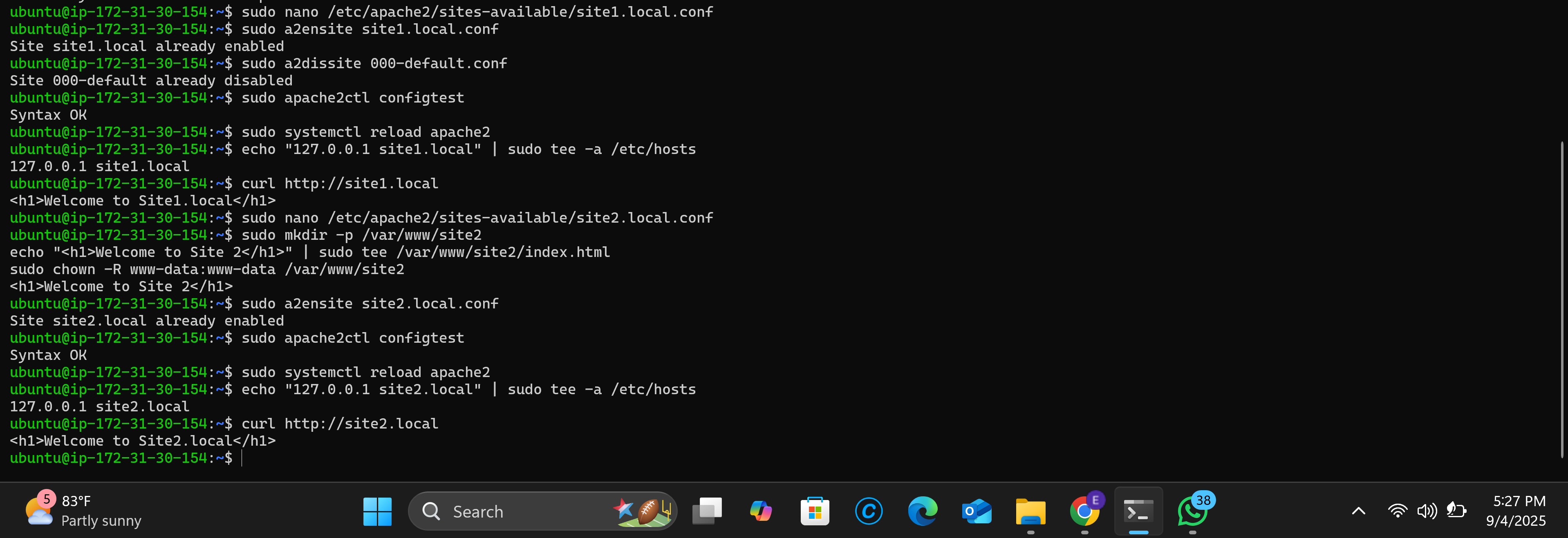
* Configured Apache virtual hosts for site1.local and site2.local.
* Added custom HTML pages for testing.

**Steps:**

1. Created config files in /etc/apache2/sites-available/.
2. Enabled sites with a2ensite and reloaded Apache.
3. Added entries in /etc/hosts for local resolution.

**Verification:**

* Accessed both sites via curl http://site1.local and curl http://site2.local.
* Confirmed correct pages loaded.



# Task 4: SSL/TLS Implementation

**Description:**

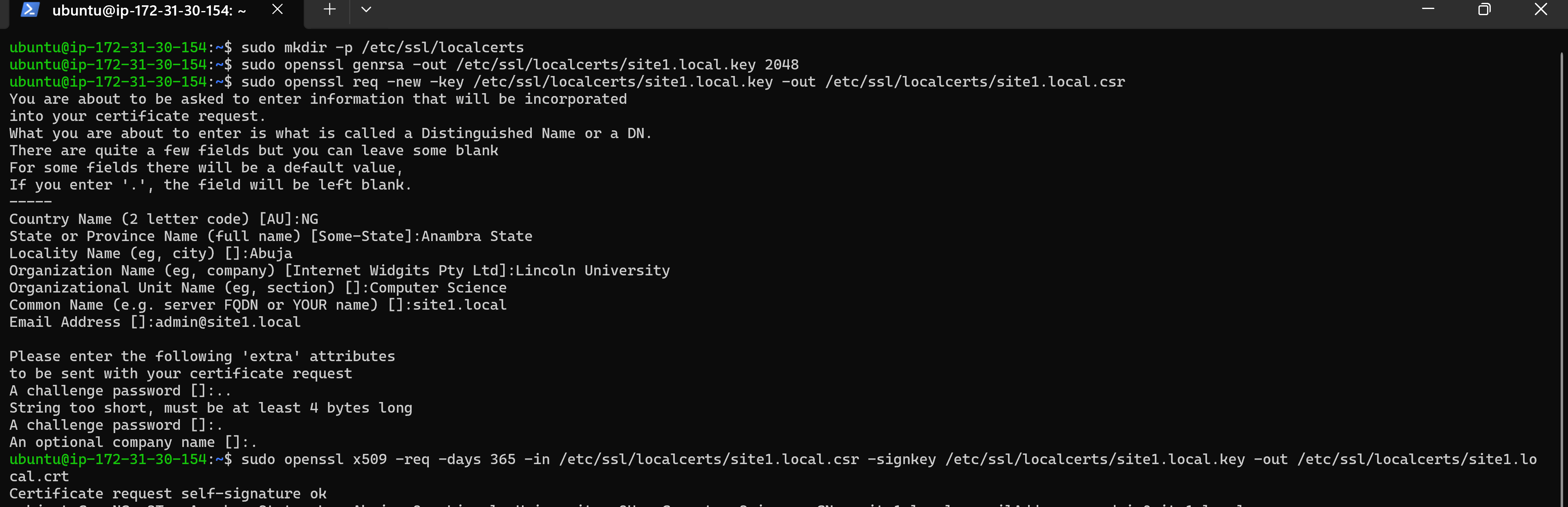
* Enabled HTTPS on Apache using self-signed SSL certificates.

**Steps:**

1. Generated certificates with openssl req -newkey rsa:2048 -nodes -keyout site.key -x509 -days 365 -out site.crt.
2. Updated Apache configuration to use SSL.
3. Enabled ssl module and restarted Apache.

**Verification:**

* Accessed https://site1.local and confirmed SSL lock icon in browser.



# Task 5: MySQL Remote Access & Security

**Description:**

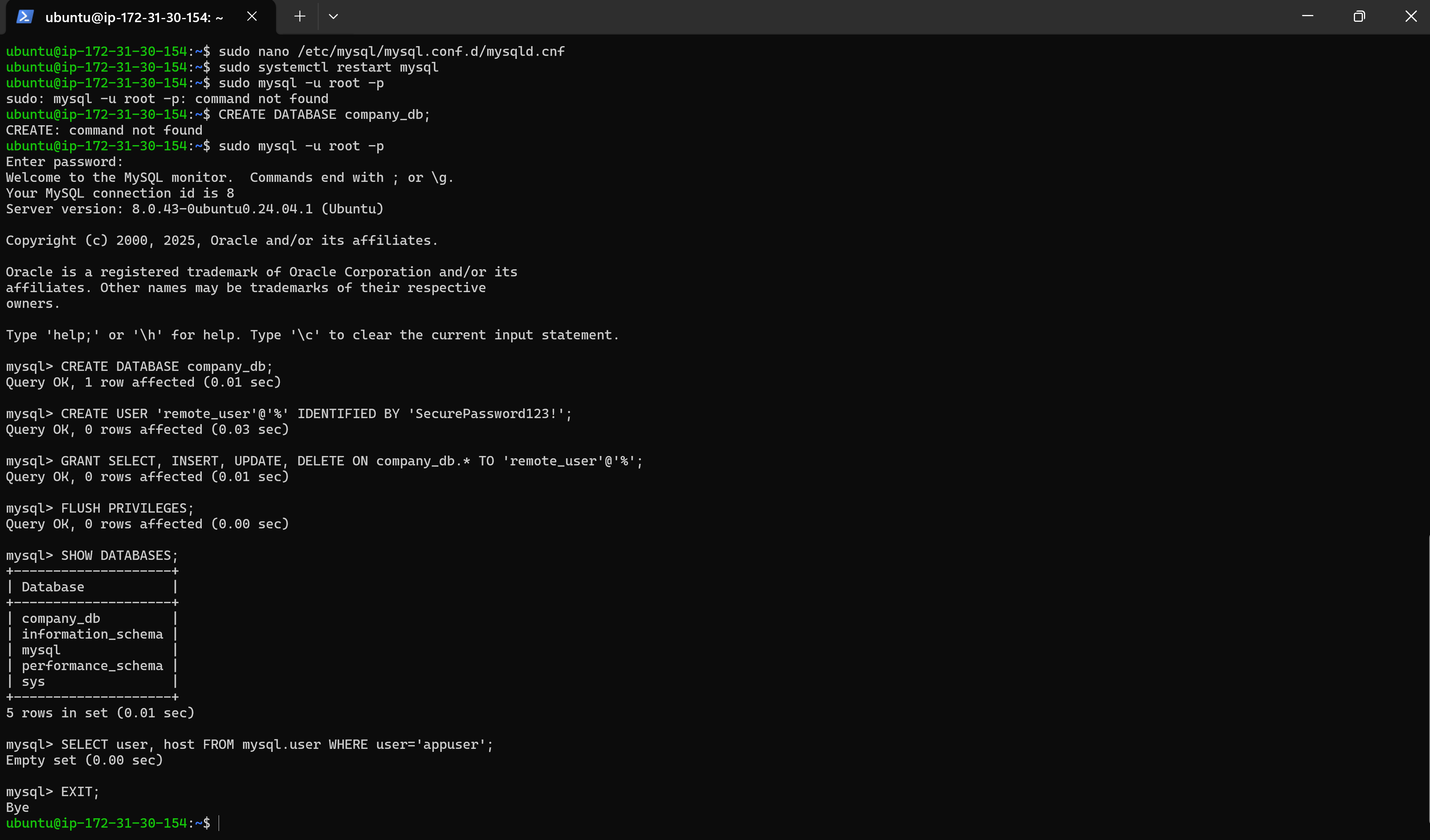
* Configured MySQL to allow remote access securely.

**Steps:**

1. Edited /etc/mysql/mysql.conf.d/mysqld.cnf to allow remote connections.
2. Created a remote user with limited privileges.
3. Applied firewall rules to restrict access.

**Verification:**

* Connected remotely using mysql -h server\_ip -u user -p.
* Confirmed secure access.



# Task 6: Firewall Configuration

**Description:**

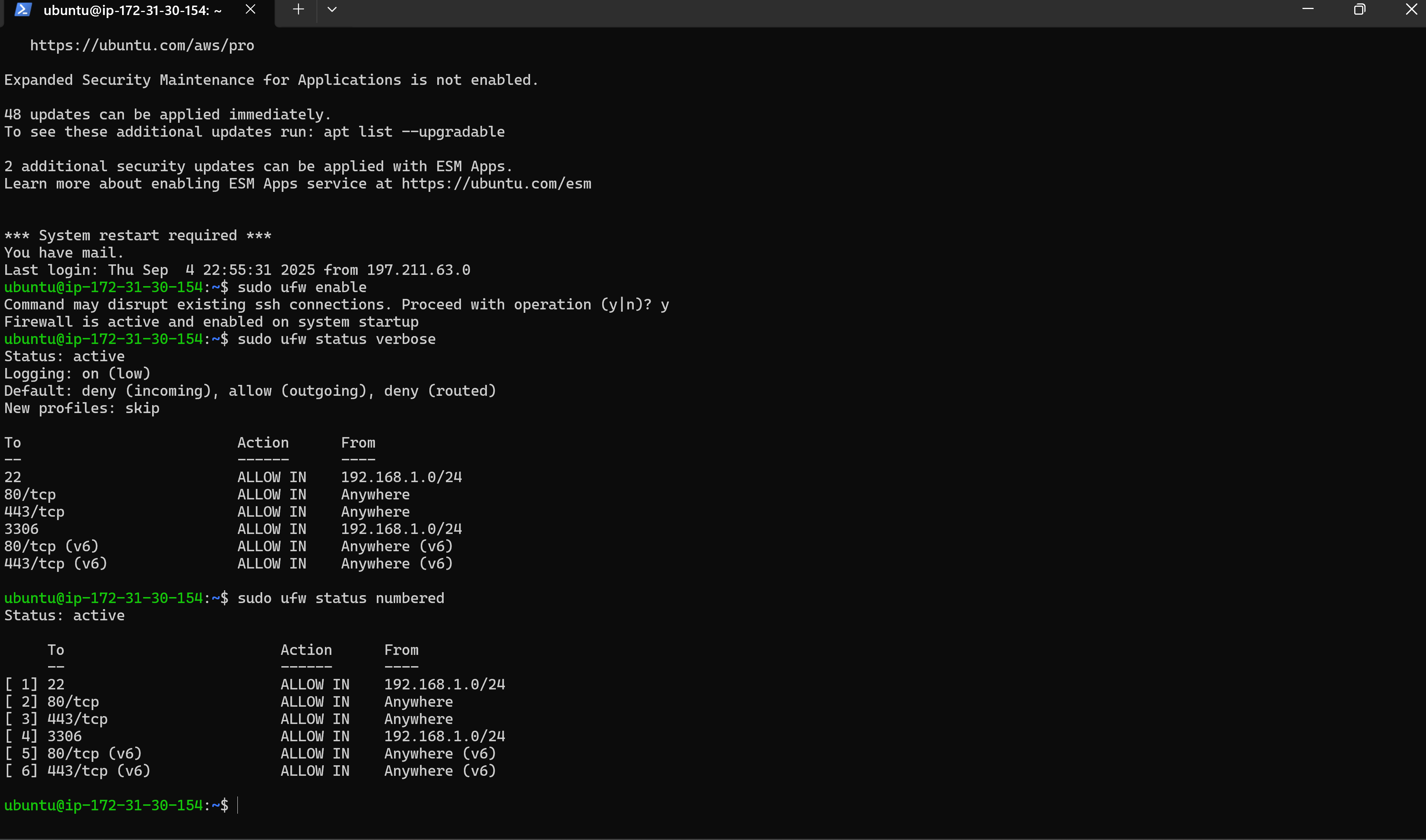
* Configured firewall rules using ufw.

**Steps:**

1. Enabled UFW with sudo ufw enable.
2. Allowed HTTP, HTTPS, and SSH.
3. Denied all other incoming traffic.

**Verification:**

* Checked status with sudo ufw status.
* Tested access to allowed and denied ports.



# Task 7: System Monitoring Script

**Description:**

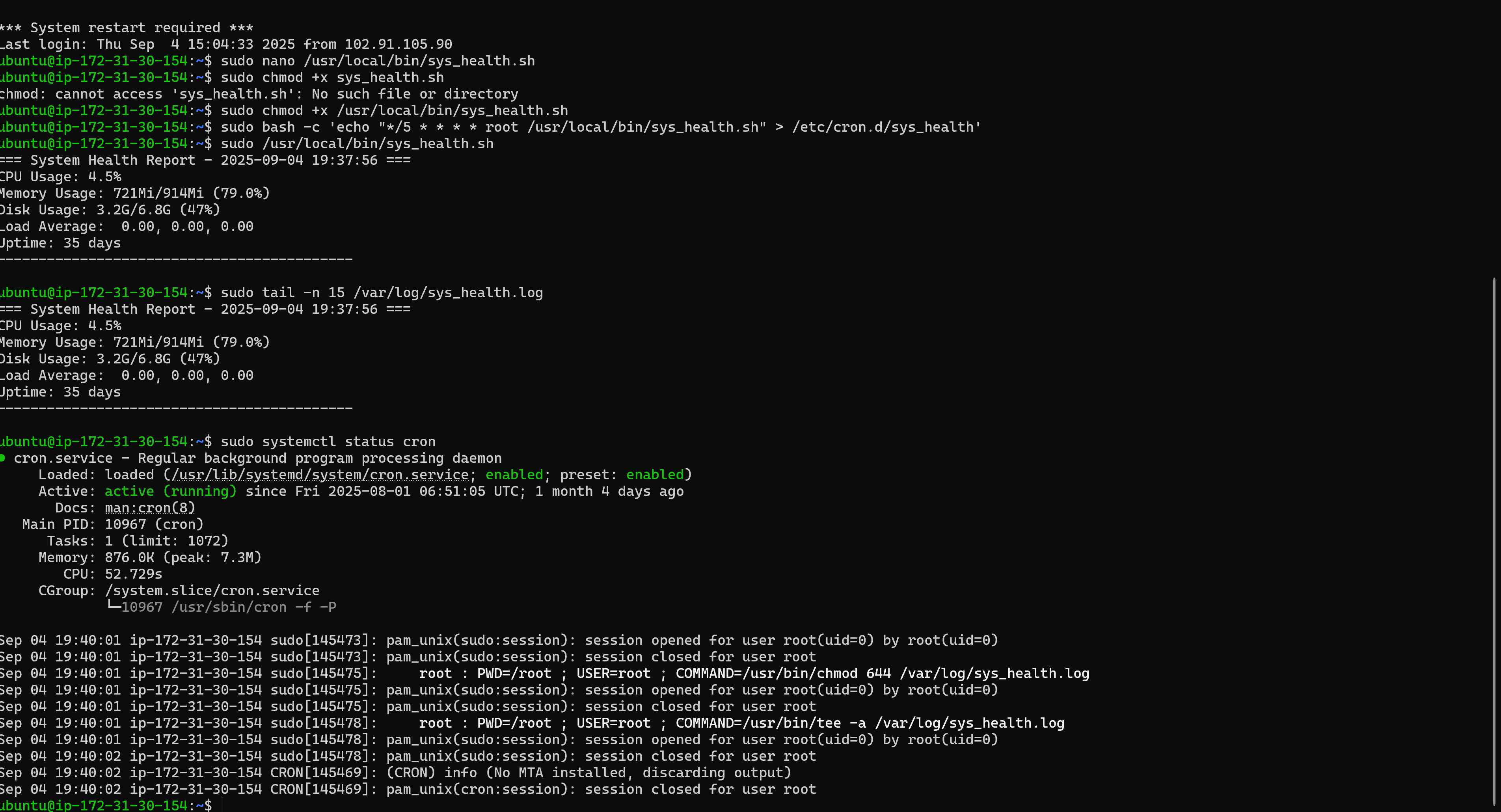
* Created a script to monitor system resources.

**Steps:**

1. Script used top, df -h, and free -m commands.
2. Logged results to /var/log/sysmon.log.
3. Added cron job to run every 5 minutes.

**Verification:**

* Checked log file for updates.
* Verified cron execution with grep CRON /var/log/syslog.



# Task 8: Log Rotation Setup

**Description:**

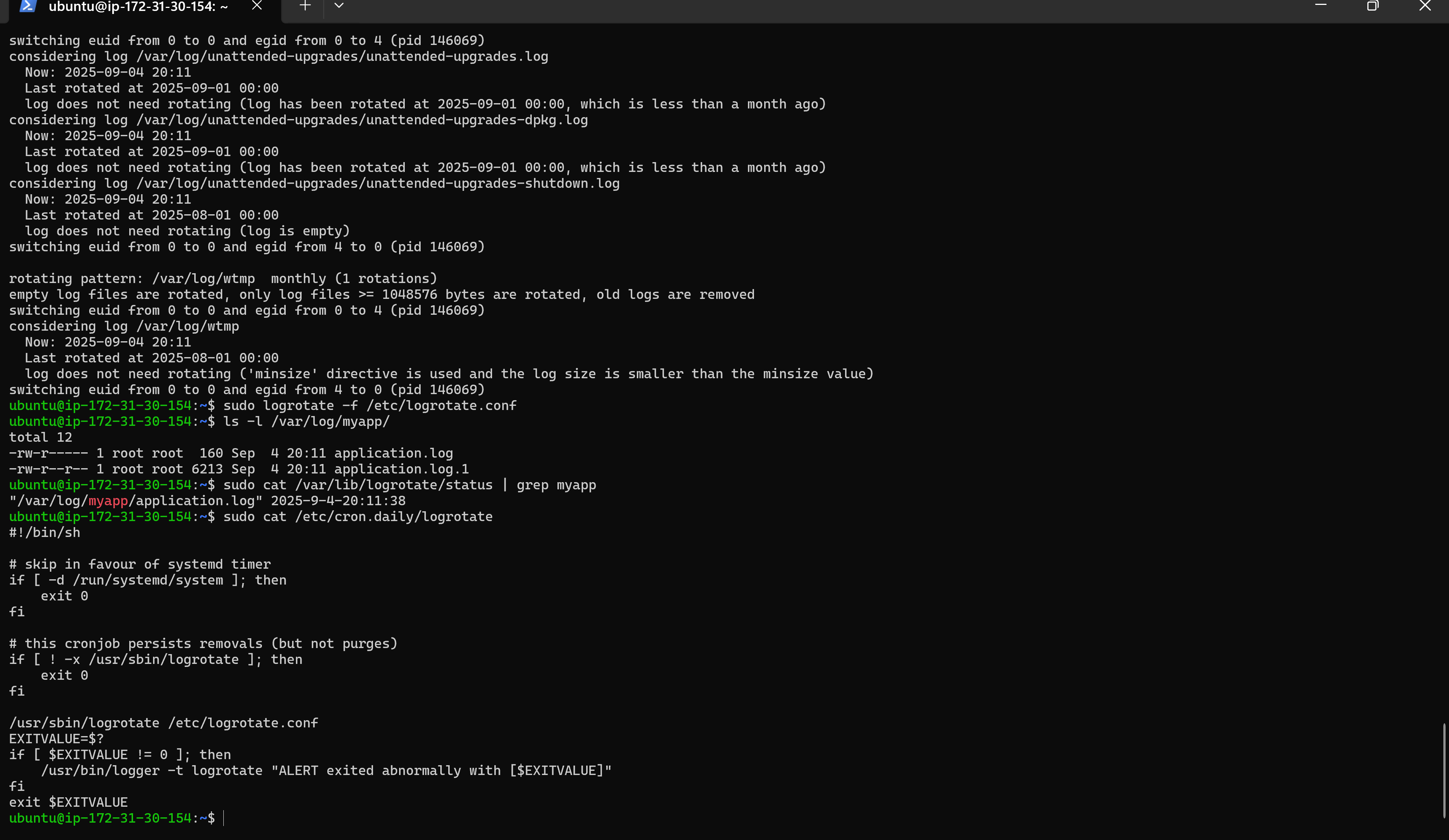
* Configured log rotation using logrotate.

**Steps:**

1. Created custom logrotate config in /etc/logrotate.d/custom.
2. Set rotation to weekly, keep 4 backups, compress old logs.

**Verification:**

* Ran sudo logrotate -f /etc/logrotate.d/custom.
* Verified logs rotated.



# Task 9: DNS Server Setup

**Description:**

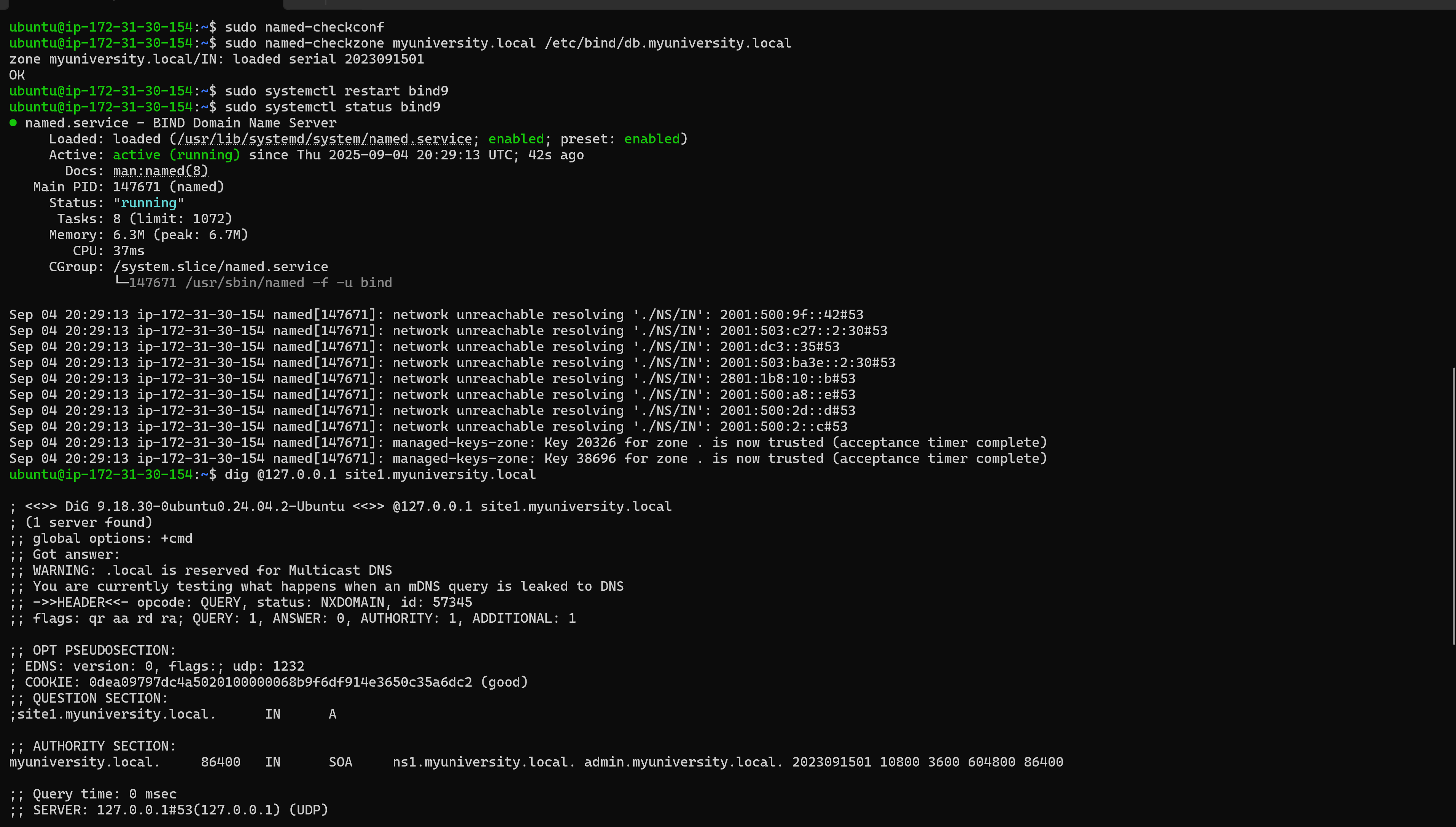
* Configured BIND9 as a DNS server.

**Steps:**

1. Installed BIND9.
2. Configured zone files for site1.local and site2.local.
3. Updated named.conf.local with zone definitions.

**Verification:**

* Tested resolution with dig site1.local and dig site2.local.



# Task 10: SSH Key Authentication + Hardening

**Description:**

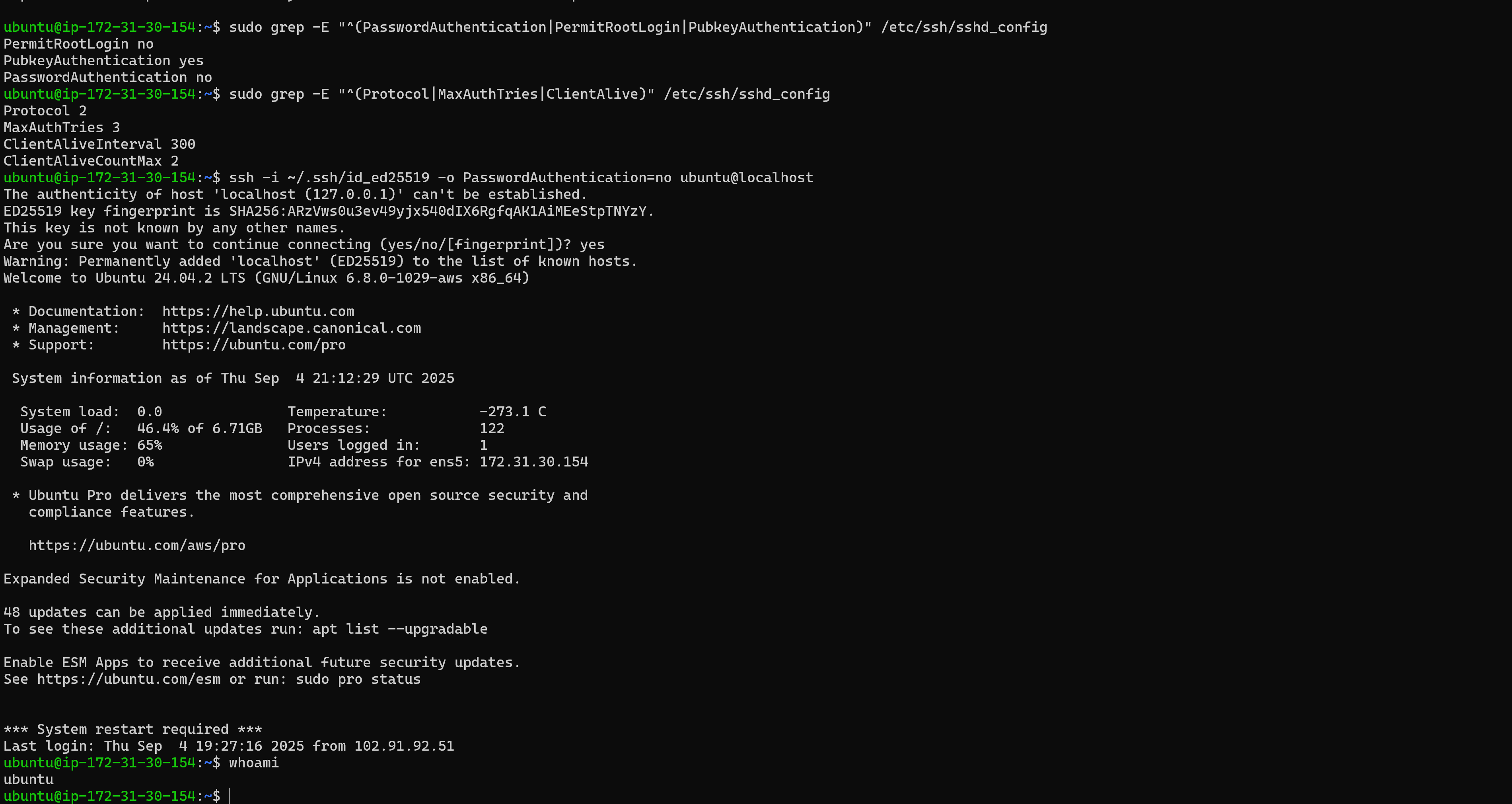
* Set up SSH key authentication and disabled password login.

**Steps:**

1. Generated SSH keys with ssh-keygen -t ed25519.
2. Copied public key to server and added to authorized\_keys.
3. Modified /etc/ssh/sshd\_config to disable password login.

**Verification:**

* Logged in using SSH key only.
* Confirmed password authentication was denied.



# Task 11: Systemd Service Creation

**Description:**

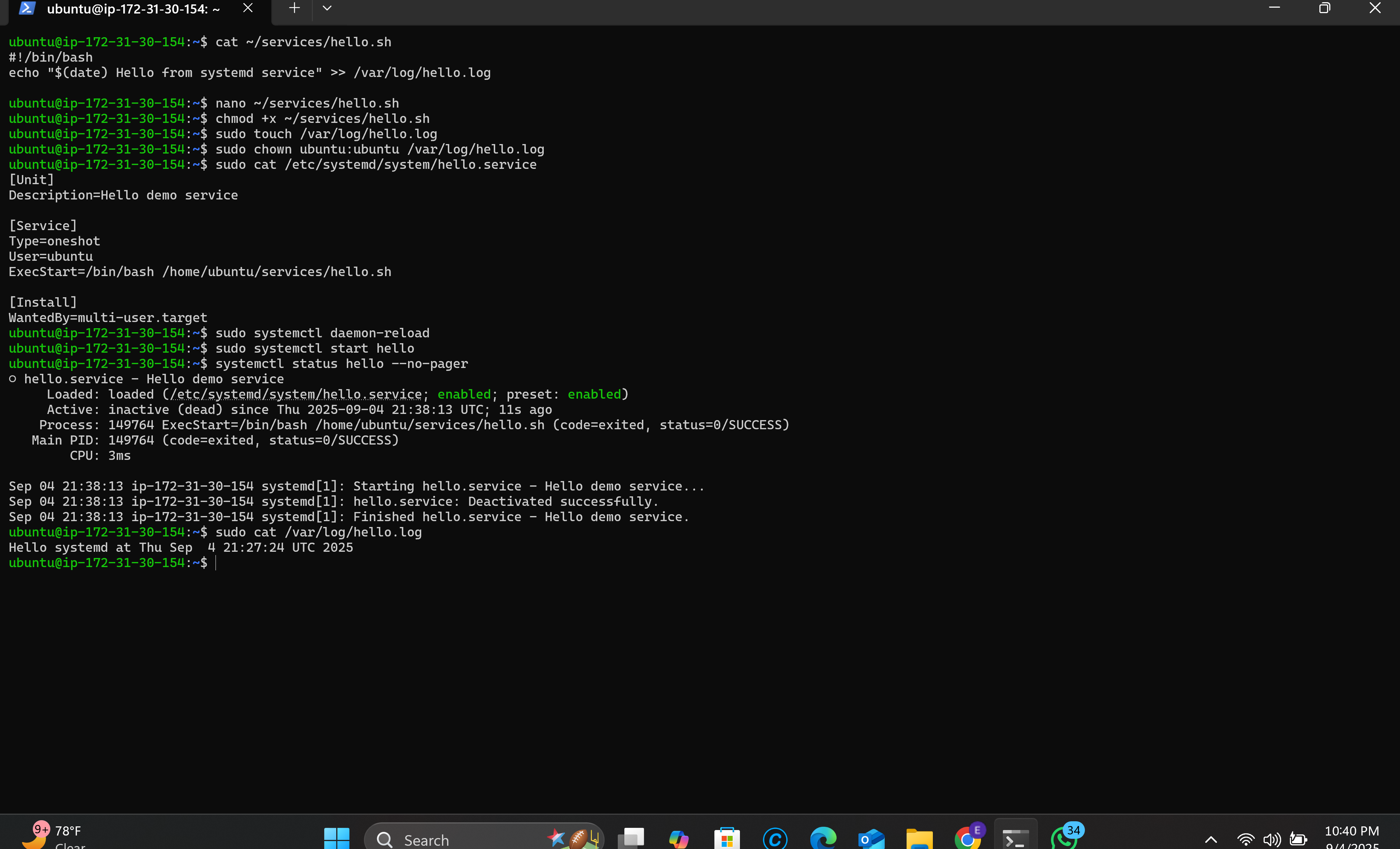
* Created a custom systemd service for a hello-world script.

**Steps:**

1. Wrote a script hello.sh to log a message.
2. Created /etc/systemd/system/hello.service.
3. Enabled and started the service.

**Verification:**

* Checked status with systemctl status hello.
* Verified log entry was created.



# Task 12: Disk Partitioning & Mounting

* Created and mounted a new disk partition.

**Steps:**

1. Used fdisk to create a new partition.
2. Formatted with mkfs.ext4.
3. Mounted under /mnt/data and added to /etc/fstab.

**Verification:**

* Confirmed with df -h.
* Rebooted system to ensure persistence.

# Task 13: Postfix Mail Server (Local Only)

**Description:**

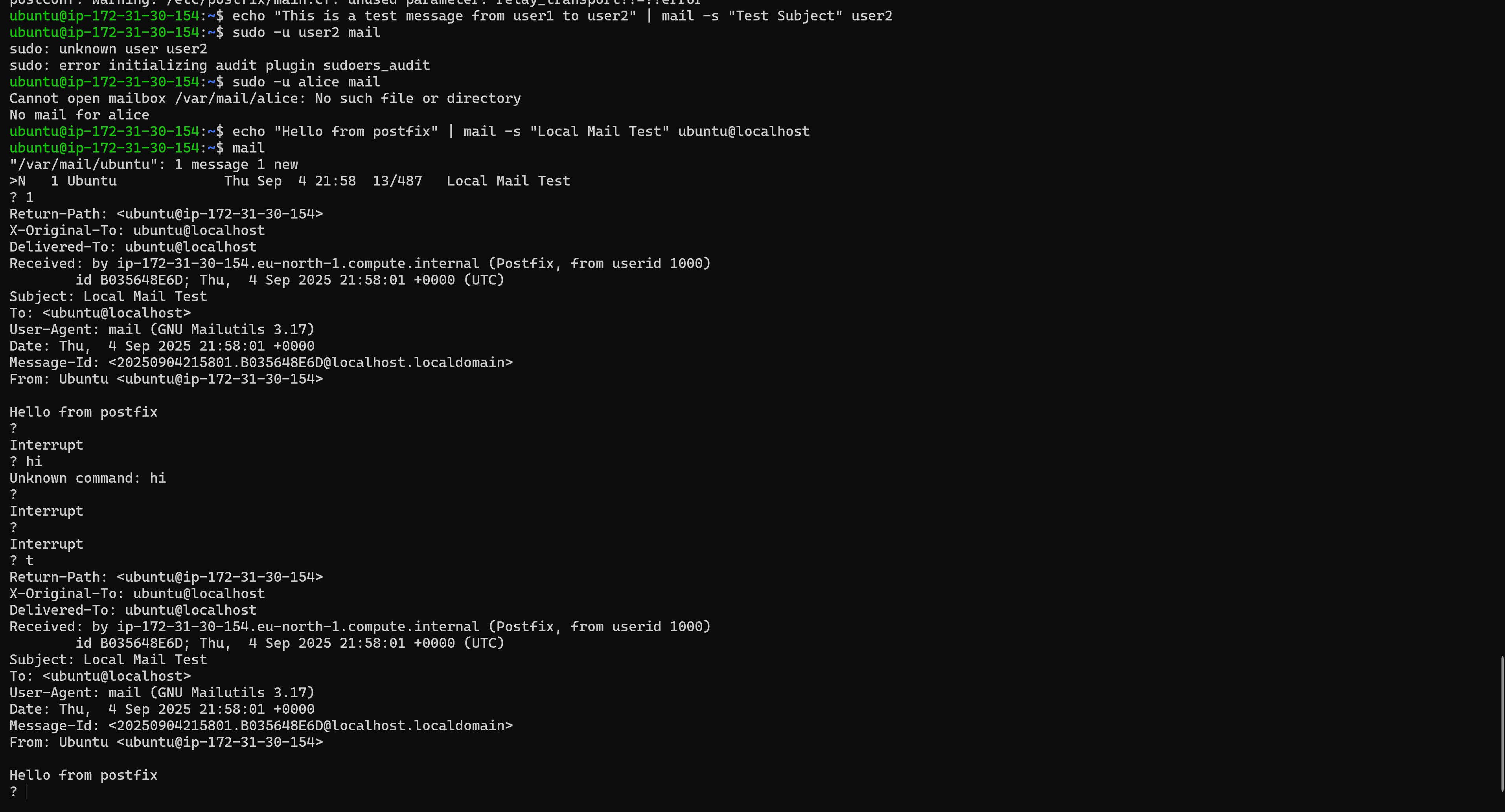
* Installed and configured Postfix for local mail.

**Steps:**

1. Installed Postfix with apt install postfix.
2. Configured for local delivery only.

**Verification:**

* Sent test email with mail command.
* Verified delivery in /var/mail/username.



# Task 14: Backup & Restore Project

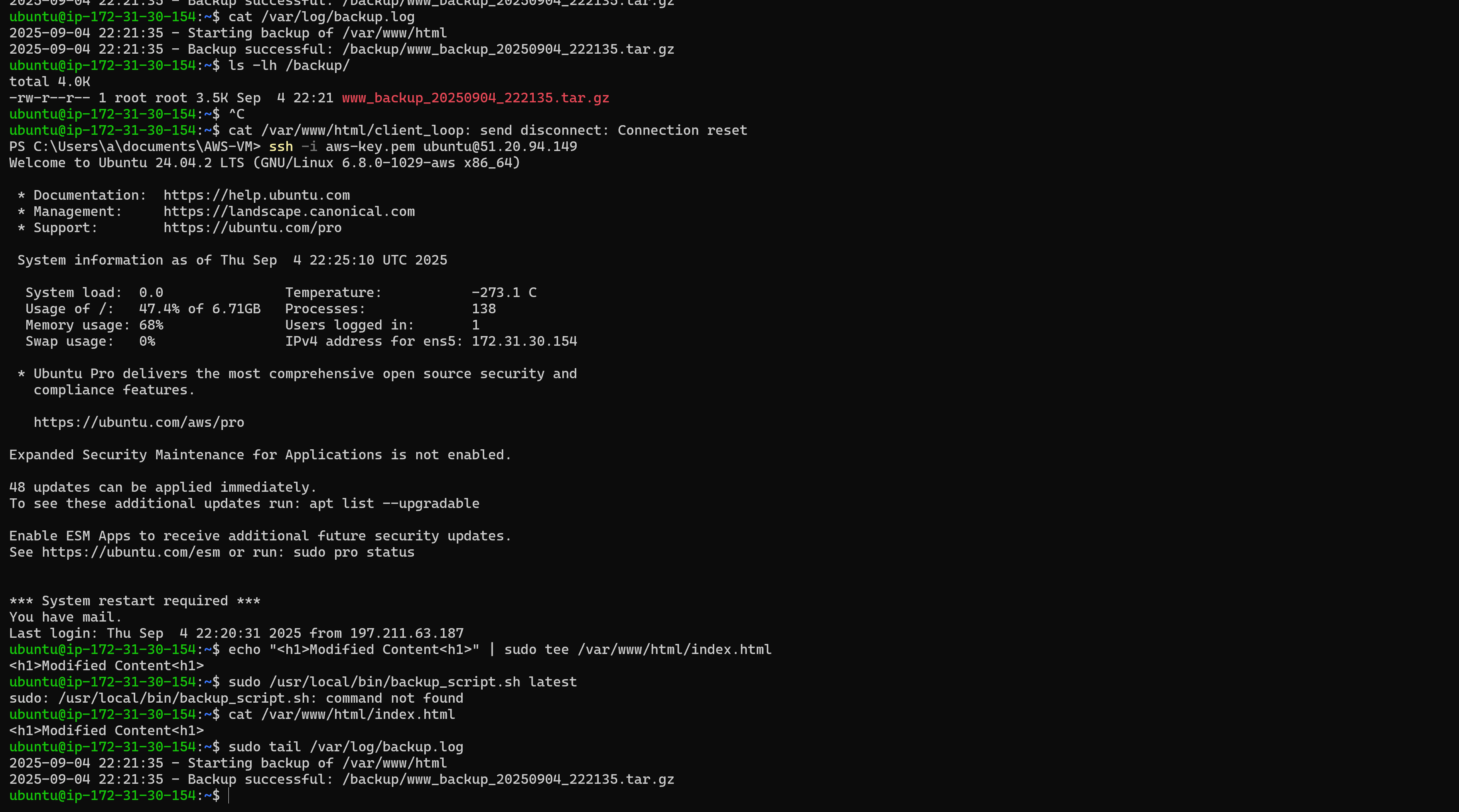
* Automated backup of /var/www/html with timestamped archive.

**Steps:**

1. Created backup\_www.sh script.
2. Configured logs in /var/log/backup.log.
3. Stored backups in /backup.

**Verification:**

* Ran script and confirmed archive creation.
* Checked log entries for backup success.



# Task 15: Containerization Challenge

Description: Deployed a custom Nginx container using Docker.

**Steps:**

1. Installed Docker and Podman.
2. Pulled Nginx image and ran container on port 8080.
3. Mounted custom HTML page into container.

**Verification:**

* Accessed http://localhost:8080 and confirmed custom page.
* Checked container logs with docker logs mynginx.

