# WEEK 4.- Initial System Configuration & Security Implementation

## Objective

This week's phase focused on deploying the Ubuntu Server and implementing essential system configuration and security controls. This involved setting up SSH key-based authentication, configuring a firewall, managing users and permissions, securing the SSH service, and performing remote administration entirely from the Linux Mint workstation.

## System Setup

- **Host Operating System:** Linux Mint 22.2 (Zara)
- **Target Server:** Ubuntu 22.04.5 LTS
- **Network Mode:** Host-Only Adapter + Internal Network
- **Connection Method:** SSH (Secure Shell) from Linux Mint → Ubuntu Server
- **Administrator User:** richard (non-root user with sudo privileges)

## Task 1.- Configure SSH with Key-Based Authentication Purpose

To enable secure, password-less access to the Ubuntu Server using an RSA key pair generated from the Linux Mint workstation.

## Commands Used

**ssh-keygen -t rsa -b 4096**
**ssh-copy-id richard@192.168.56.2**
**ssh richard@192.168.56.2**

## Explanation

- **ssh-keygen** - Generates a public/private RSA key pair.
- **ssh-copy-id** - Copies the public key to the server for authentication.
- **ssh** - Verifies that login now works via key authentication.

## Result

Connection succeeded without a password, confirming that SSH key-based authentication was correctly configured.

## Difficulties & Solutions

- **Issue:** Initial “Permission denied (publickey)” errors occurred.
- **Cause:** Wrong username used during ssh-copy-id.
- **Fix:** Re-ran command with correct user (admin first, then richard) and ensured public key existed in ~/.ssh/id_rsa.pub.

## Task 2.- Configure Firewall (UFW)

## Purpose

To enable and customise the Ubuntu Server firewall so that only the Mint workstation can connect via SSH.

## Commands Used

**sudo ufw enable**
**sudo ufw allow from 192.168.56.3 to any port 22**
**sudo ufw status verbose**

## Result

Firewall active and restricted.  Only 192.168.56.3 (Linux Mint) was permitted SSH access, and all other connections were denied. 'sudo ufw status' confirmed the rules.

## Difficulties & Solutions

**Issue:** Firewall initially blocked all SSH connections after UFW was enabled.

**Fix:** Accessed the VM console directly, manually allowed SSH, and then restricted access to Mint's IP.

# Task 3.- User Management & Privilege Control

## Purpose

To create a non-root administrative user and assign sudo privileges for safer management.

## Commands Used

**sudo adduser richard**
**sudo usermod -aG sudo richard**
**groups Richard**

## Explanation

adduser creates a new user account.
usermod -aG sudo adds the user to the sudo group.
groups verifies the privilege assignment.

## Result

User **richard** was created and gained sudo rights.  All administrative tasks could be performed without using root directly.

## Difficulties & Solutions

- **Issue:** “Permission denied” when switching users.
- **Fix:** Logged out and back in to refresh sudo group membership.

## Task 4.- SSH Access Verification

## Purpose

To confirm that the Mint workstation can establish an SSH connection to the Ubuntu Server with the new user.

## Commands Used

**ssh** richard@192.168.56.2

## Result

Connection was successful. This verified that the SSH service was running and that the firewall rule was correctly set up.

### Difficulties & Solutions

**Issue:** Connection refused until UFW rule was added.

**Fix:** Allowed port 22 from Mint IP using sudo ufw allow from 192.168.56.3 to any port 22.

## Task 5.- SSH Configuration Enhancement

## Purpose

To harden SSH security by editing the **sshd_config** file and restarting the SSH service.

## Commands Used

**sudo nano /etc/ssh/sshd_config**

**# Updated lines:**
**PubkeyAuthentication yes**
**PasswordAuthentication yes**
**PermitRootLogin prohibit-password**
**AllowUsers richard**
**sudo systemctl restart ssh**
**sudo systemctl status ssh**

## Explanation

These settings ensure:

- Root login is disabled.
- Only key-based and richard logins are allowed.
- SSH service runs securely and actively.

## Result

**systemctl status ssh** showed **active (running)**, confirming the secure configuration was valid.

## Difficulties & Solutions

- **Issue:** SSH failed to start due to wrong directive (AllUsers).
- **Fix:** Reviewed error logs using sudo sshd -t, removed invalid line, and restarted service successfully.

## Task 6.- Firewall Documentation (Ruleset)

## Purpose
To log and verify all active firewall rules for security audits.

## Commands Used

**sudo ufw status verbose**
**sudo ufw app list**
**sudo iptables -L -v -n**
**sudo ufw status verbose > firewall_rules.txt**

## Result
The ruleset was successfully exported to firewall_rules.txt.  SSH access was limited to the Mint workstation, and all other traffic remained blocked.

## Observation
Firewall status shown as:

```
To                         Action  From
--                         ------  ----
22/tcp (OpenSSH)           ALLOW   192.168.56.3
22/tcp (v6)                DENY    Anywhere (v6)
```

## Difficulties & Solutions

- **Issue:** Needed IPv6 rules were also active.
- **Fix:** Explicitly set DENY IN Anywhere (v6) for port 22 in UFW.

# Task 7.- Remote Administration via SSH

## Purpose
To demonstrate comprehensive remote server management via SSH from Linux Mint, including monitoring and updates.

## Commands Used

**ssh richard@192.168.56.2**
**uptime**
**uname -r**
**top -n 1**
**free -h**
**df -h**
**sudo apt update**
**sudo apt upgrade -y**
**sudo systemctl status apache2**
**sudo ufw status verbose**

## Results / Evidence

- The SSH connection was established successfully without a password.

- System monitoring commands displayed uptime, kernel details, memory, and disk usage.

- The 'sudo apt update && upgrade' command was executed correctly remotely. The Apache2 service remains active and running.

- Firewall rules remained intact during the remote session. (Some network repositories were unreachable because of the Host-Only network, but all commands executed successfully.)

## Difficulties & Solutions

This task presented the **most notable issue**:

- **Problem:** Repeated “Permission denied (publickey)” and “Host key verification failed” messages.

- **Cause:** Key mismatch and conflict between old and new users (admin and richard).

**Fix Steps:**

1. Removed old SSH entries (rm -rf ~/.ssh), recreated the directory.
2. Regenerated new RSA key pair (ssh-keygen -t rsa -b 4096).
3. Re-copied the public key to the server (ssh-copy-id richard@192.168.56.2).
4. Restarted SSH service and tested connection until login was successful.

- **Outcome:** SSH connection finally established without passwords, verifying secure remote administration.

# Analysis & Reflection

This week's practical covered core Linux system administration tasks. Through each task, I learned how to:

- Generate and use SSH keys for secure authentication.
- Configure and verify firewall rules (UFW and iptables).
- Create and manage users with appropriate privileges.
- Harden the SSH service by disabling root access.
- Remotely monitor and maintain a Linux server entirely from the terminal.

# Conclusion

All seven tasks were completed successfully. Key-based SSH login, user management, firewall rules, and remote administration were implemented and tested. This phase demonstrated that secure remote system management is achievable through Linux Mint and Ubuntu Server using SSH. Each command enhanced my understanding of how operating systems manage users, permissions, and network security.