**Day 1:**

**Project:** **Simple Network Scanner WebApp (nmap)**

The main motivation for selecting the Simple Network Scanner WebApp (nmap) project is familiarity with the technologies involved. Working on this project aims to improve skills and upgrade knowledge in these areas. Additionally, this project will facilitate professional growth and bring immediate value to the company by leveraging existing skills to deliver a high-quality product.

* **What area of work?**

A Network Scanner WebApp using nmap integrates network scanning capabilities with web technology to deliver a streamlined method for monitoring and managing network resources. It combines the power of nmap's scanning capabilities with the accessibility and convenience of web-based interfaces.

* **Purpose of your work?**

The primary objective of this project is to create a web application that allows users to monitor the network status in real-time through a browser interface. By using nmap and PHP scripting, the application performs periodic scans of specified network ranges, aggregates the results into a readable format, and presents them dynamically on a web page.

**Software required:**

* Linux
* Appache2
* php
* nmap
* Cron

**Objective:** The goal of this project is to create a simple network scanner accessible via a web browser. This scanner periodically scans the local network using nmap, formats the scan results, and displays them on a web page.

**Components:**

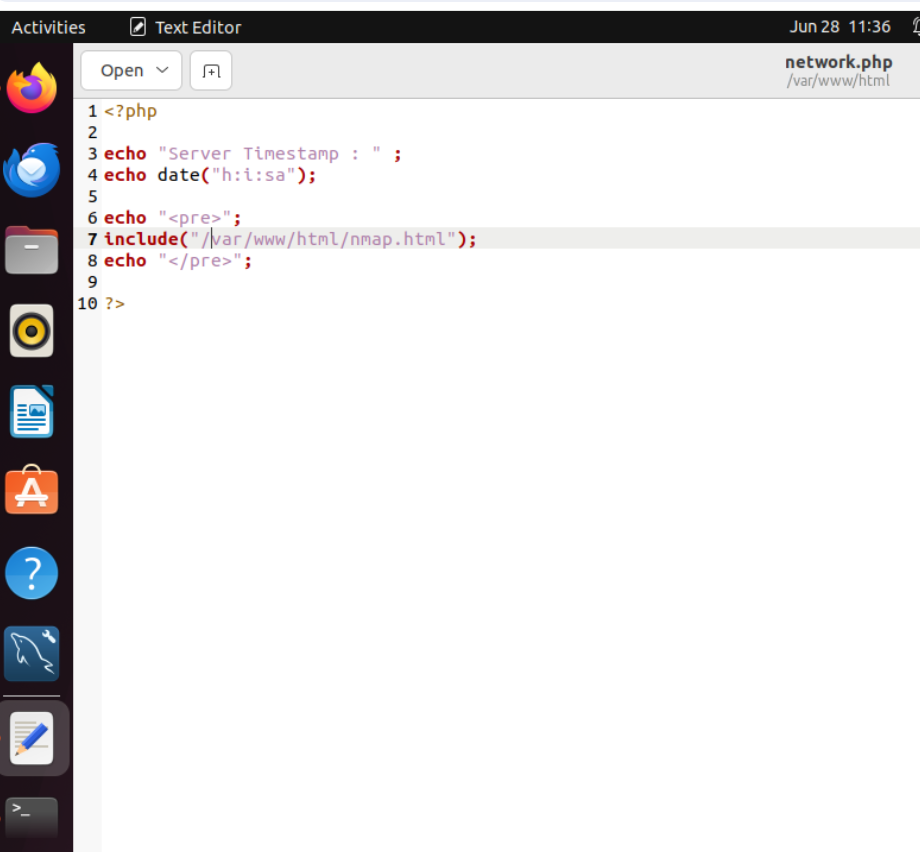
1. **Setup:**

* Linux Environment: Ubuntu
* Installed Packages: Apache2 (web server), PHP, nmap
* Permissions and Ownership: Adjusted to allow web server access to necessary files (/var/www/html)

1. **Implementation Steps:**
   * **Installing Required Software:**
     + Apache2: sudo apt-get install apache2
     + PHP: sudo apt-get install php
     + nmap: sudo apt-get install nmap
   * **Configuring Permissions:**
     + Changed ownership of web directory: sudo chown ubuntu /var/www/html
     + Changed permissions of web directory (for simplicity, used chmod 777 which is generally not recommended for production environments due to security risks).
   * **Setting up Cron Job:**
     + Configured a Cron Job to run every 10 minutes (\*/10 \* \* \* \*) to execute nmap and save results to /var/www/html/nmap.html

\*/10 \* \* \* \* nmap 172.20.0.25/24 -oN /var/www/html/nmap.html

* + **Creating PHP Script (network.php):**
    - PHP script (network.php) fetches and displays the nmap results formatted for web:



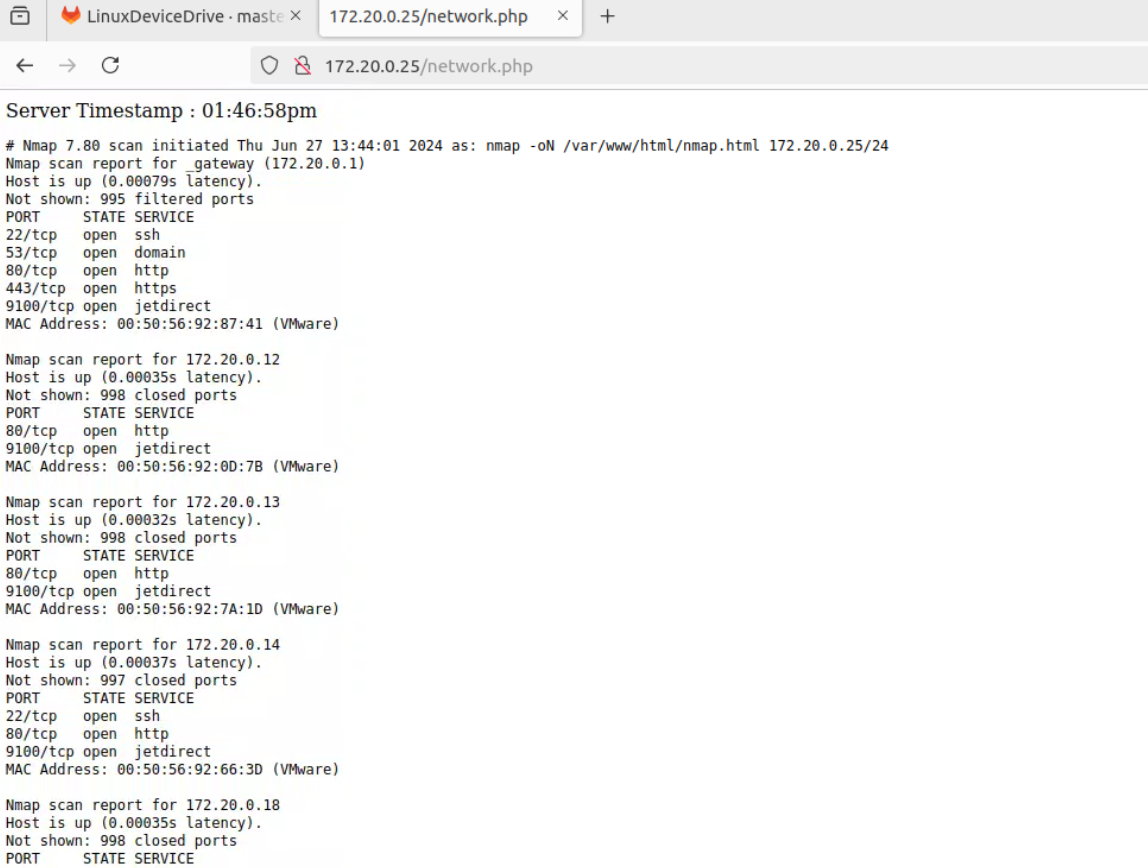
* + **Web Interface:**
    - Users access the network scan results via a web browser by visiting http://172.20.0.25/network.php.
    - The webpage displays the timestamp of the server when the page was loaded and the formatted nmap results.

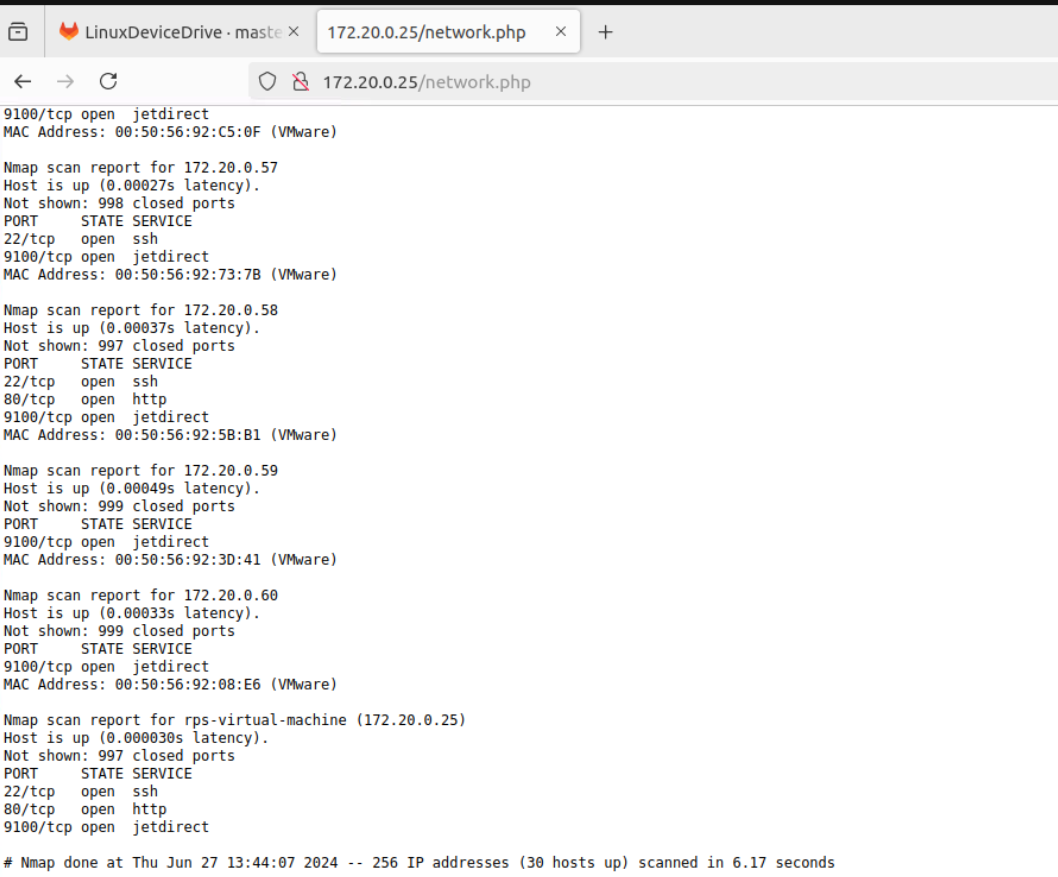
**What We Learned:**

* **Network Scanning:** Utilizing nmap for network scanning and saving results.
* **Automation:** Using Cron Jobs for scheduling repetitive tasks
* **Web Development:** Basic PHP scripting for reading files and displaying content.
* **Server Configuration:** Setting up Apache2, configuring file permissions, and understanding potential security risks (chmod 777).

**Conclusion:** This project demonstrates the integration of network scanning with web technologies, providing a simple yet effective way to monitor network activity and status through a web interface. It also highlights the importance of security and proper permissions management in web server environments.

**Output:**

****

****