Introduction

In this project, I set up and configured **Nessus Essentials**, one of the most widely used vulnerability assessment tools, on an **Ubuntu Virtual Machine (VM)**. The objective of this project was to gain hands-on experience in installing, configuring, and using Nessus to perform vulnerability scans within a controlled environment.

Nessus is a powerful tool developed by Tenable that helps security professionals and system administrators identify misconfigurations, vulnerabilities, and potential threats within a network or system. By scanning systems for open ports, running services, and known vulnerabilities, Nessus provides valuable insights into an organization's security posture and helps in mitigating risks.

In this lab, I focused on:

- **Setting up the environment:** Launching an Ubuntu VM and preparing it for the Nessus installation.
- **Installing Nessus Essentials:** Downloading, installing, and configuring Nessus on the VM.
- **Configuring Nessus for first use:** Creating an admin account, activating Nessus Essentials, and accessing the web interface.
- **Performing a vulnerability scan:** Running a scan on the **localhost (127.0.0.1)** to test the installation and review identified vulnerabilities.
- **Analyzing scan results:** Reviewing detected vulnerabilities, understanding their severity, and exploring recommended remediations.

This project provided me with practical experience in **system administration**, **security tool deployment**, and **vulnerability assessment techniques**, which are essential skills for both IT and cybersecurity professionals.

Project Objectives and Approach

To achieve the objectives of this project, I carried out a structured, step-by-step process that included the following:

Step 1: Install and Launch Your Ubuntu Instance

Step 2: Updated the System

Step 3: Installed cURL

Step 4: Downloaded and Installed Nessus

Step 4: Start and Enable Nessus Service

Step 5: Access the Nessus Web Interface

Step 6: Log In to Nessus Web Interface

Step 7: Create a New Scan

Step 8: Run and Monitor the Scan

Step 9: Review Scan Results

Step 10: Analyzed Vulnerabilities

Step 1: Install and Launch Your Ubuntu Instance

The first step of the project was to set up the working environment. For this project, I used an **Ubuntu Virtual Machine (VM)** as the host system where Nessus Essentials would be installed.

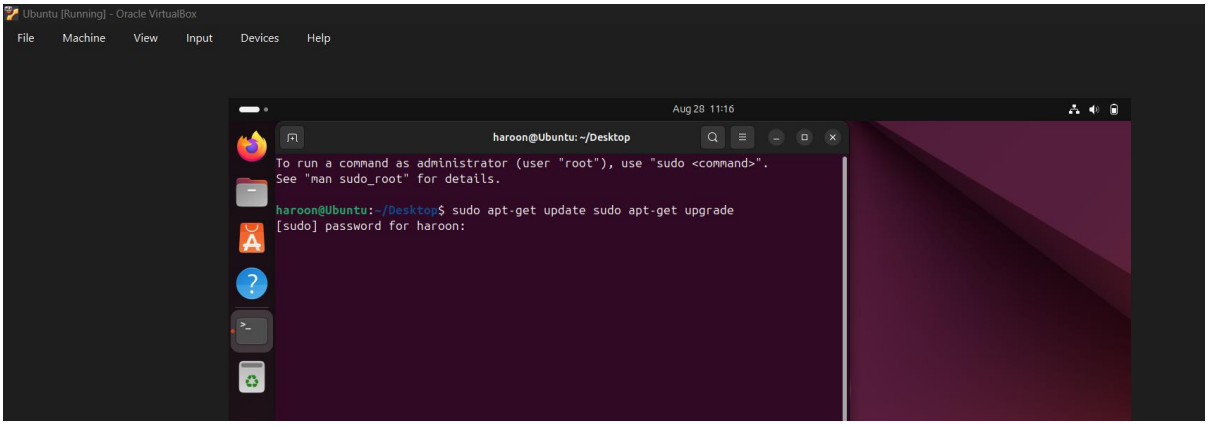
There are multiple ways to launch an Ubuntu instance:

- **On a Virtual Machine (VM):** Using virtualization platforms such as **VirtualBox**, **VMware Workstation**, or **Hyper-V**.
- **On Cloud Infrastructure:** Using cloud providers such as **AWS EC2**, **Microsoft Azure**, or **Google Cloud Platform**.

For this project, I chose the **VM approach**, which provided a controlled environment for learning and testing.

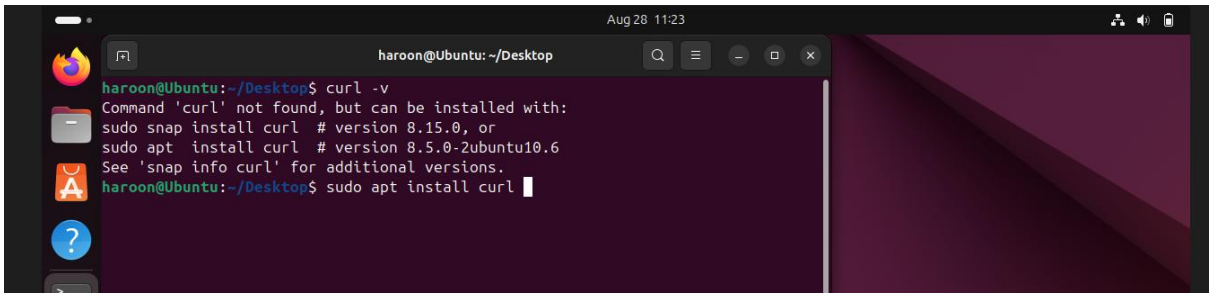
Key Actions Performed:

1. Downloaded and installed an Ubuntu ISO image (20.04 LTS was used in my case).
2. Created a new VM with sufficient resources (2 CPUs, 4GB RAM, 20GB Disk).
3. Booted into Ubuntu and completed the OS installation.
4. Verified network connectivity and system readiness before proceeding to the Nessus installation.



Step 2: Updated the System

Ran `sudo apt-get update && sudo apt-get upgrade -y` to ensure all system packages are up to date.

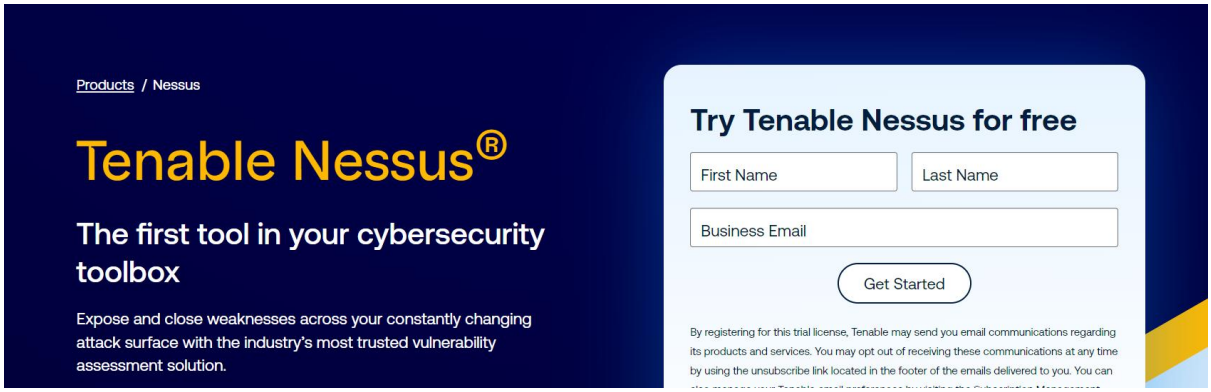


Step 3: Installed cURL

Ran `sudo apt-get install curl -y` to install cURL, which is required to download the Nessus package.

Step 4: Downloaded and Installed Nessus

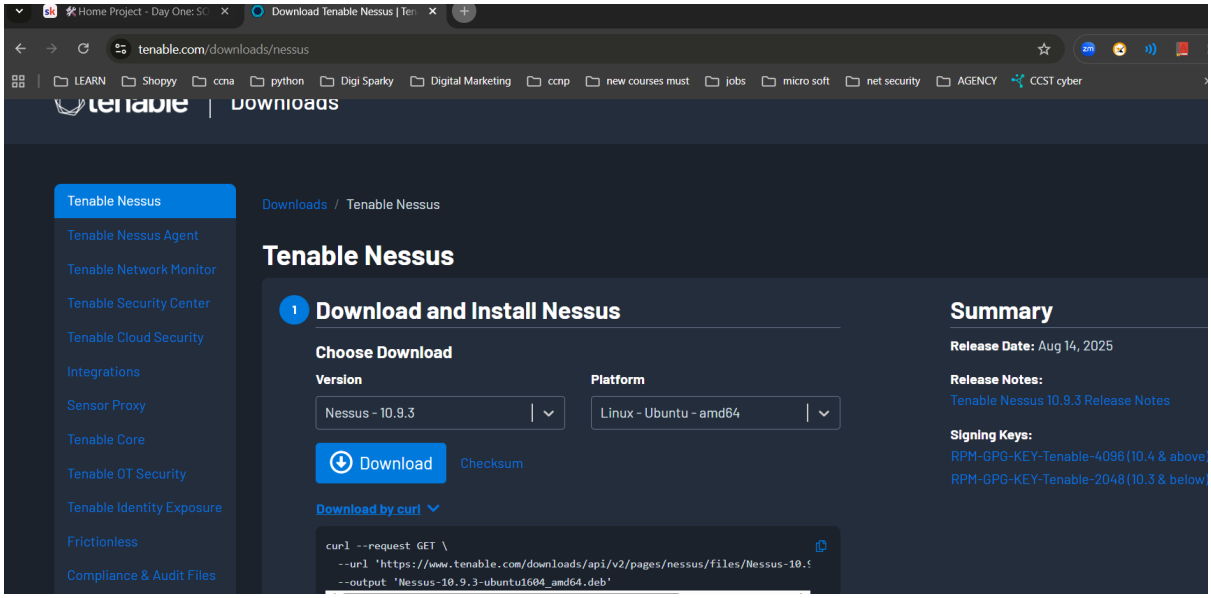
- I went to the **Tenable website → Products → Nessus →** click **Try for Free**.
- Filled out the registration form to receive a free **Nessus Essentials activation code** via email.



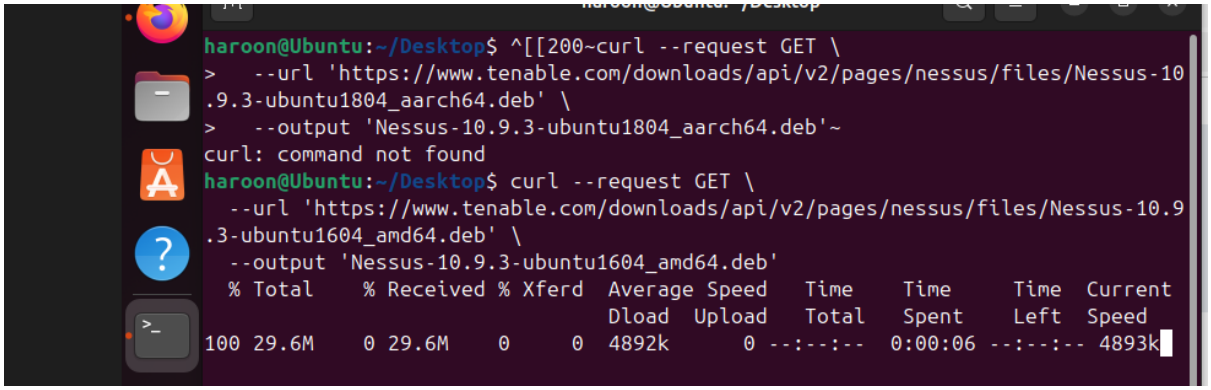
- On the redirected page, I selected the **Ubuntu Linux** distribution and copy the provided **cURL command**.
- Ran the cURL command in my Ubuntu terminal to download the Nessus package:

```
curl --request GET \
  --url
'https://www.tenable.com/downloads/api/v2/pages/nessus/files/Nessus-10.9.3-
ubuntu1604_amd64.deb' \
  --output 'Nessus-10.9.3-ubuntu1604_amd64.deb'
```

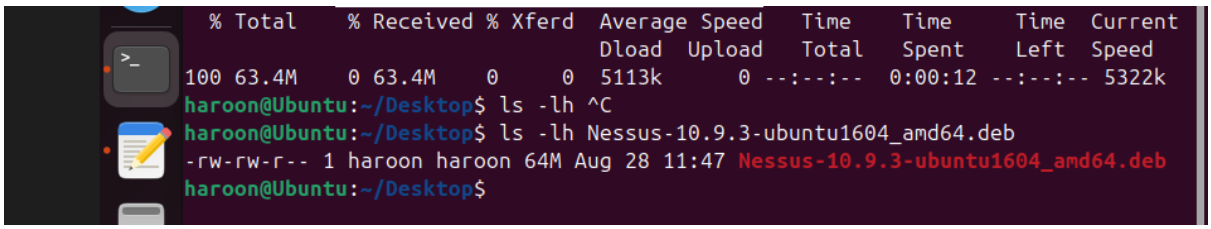
This started the download for Nessus installer to my VM.



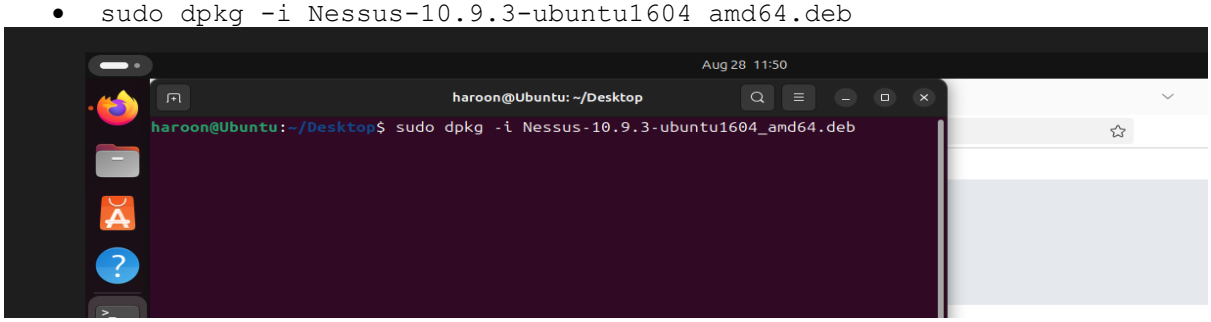
- The download took few seconds to complete.



- After the completion I had to verify by giving the command.
- `ls -lh`



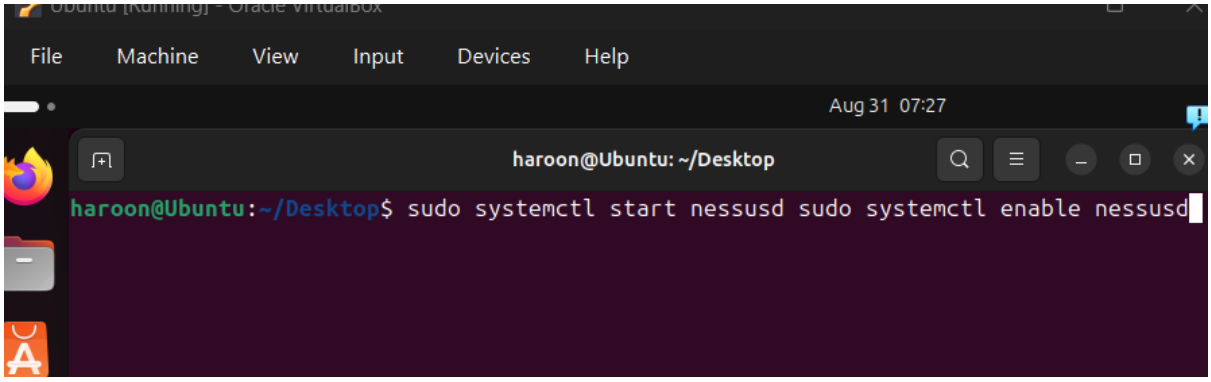
- After that, I installed Nessus by running the following command in the terminal:
- `sudo dpkg -i Nessus-10.9.3-ubuntu1604_amd64.deb`



Step 5: Start and Enable Nessus Service

I started the Nessus service and enabled it to run automatically on system boot by using the following commands:

```
sudo systemctl start nessusd
sudo systemctl enable nessusd
```



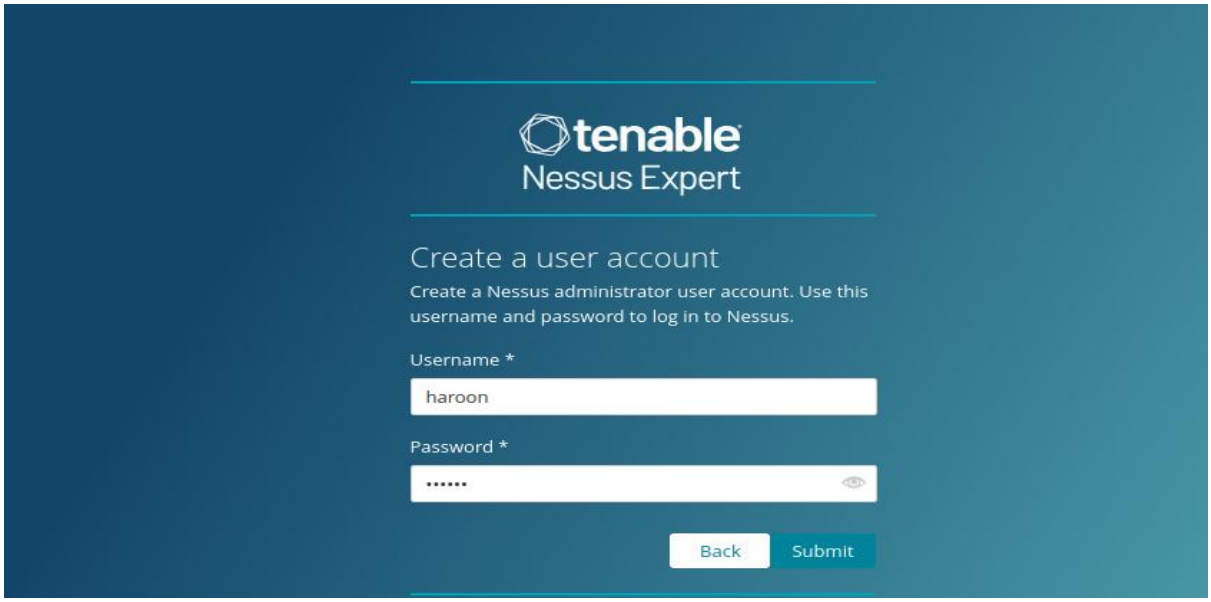
This ensured that the Nessus service was active and persistent across reboots.

Step 6: Access the Nessus Web Interface

I accessed the Nessus web interface by opening a browser and navigating to my **private VM IP** at:

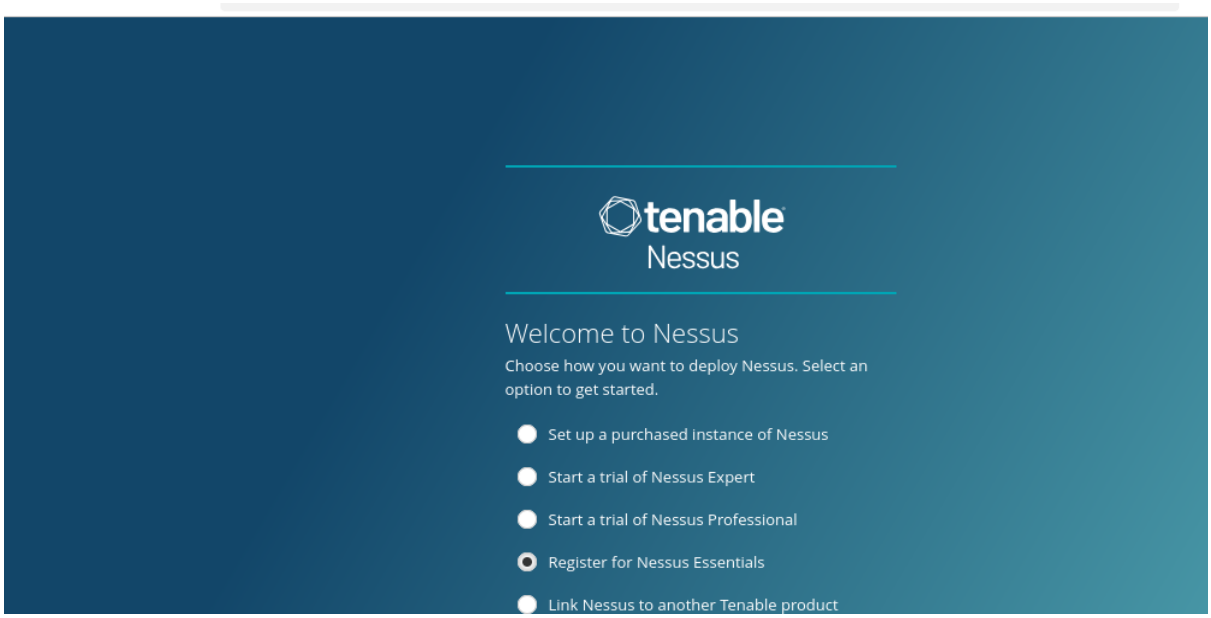
`https://<private_vm_ip>:8834`

Upon first access, I was prompted to select a Nessus version. Initially, I selected the **Expert version** by mistake, so I restarted the process and then correctly chose **Nessus Essentials**.

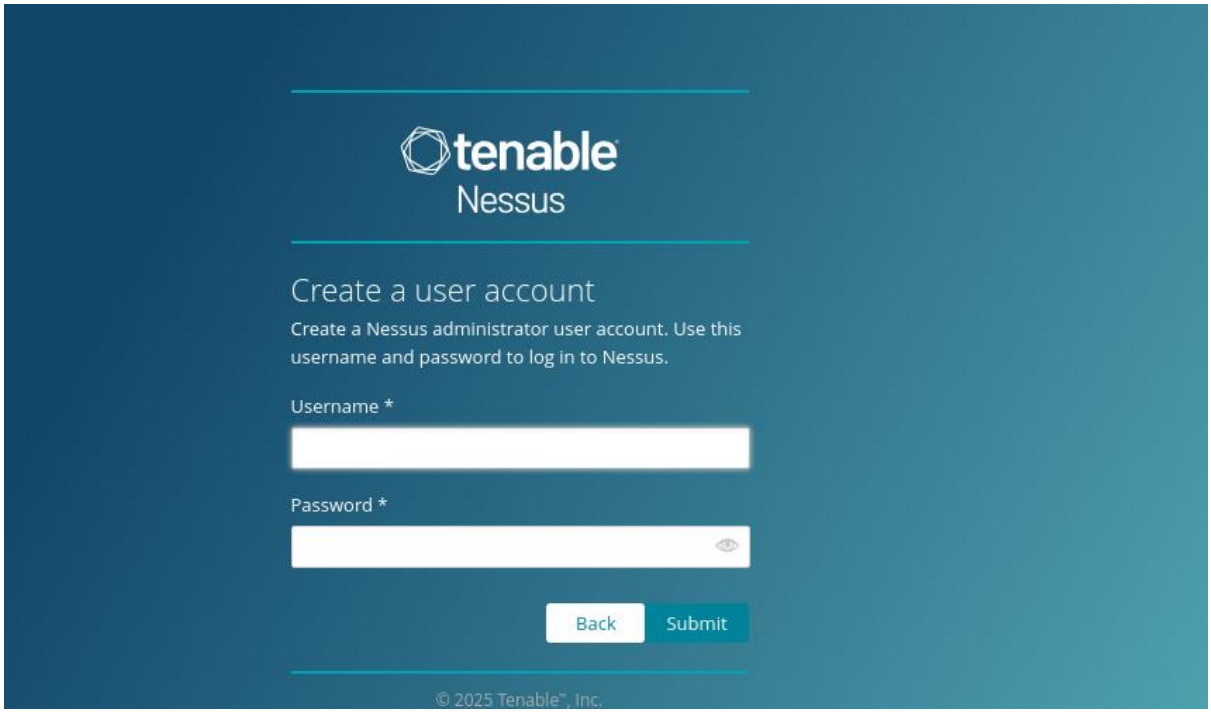


Next, I completed the Nessus setup by:

- **Selecting "Nessus Essentials"** as the version.

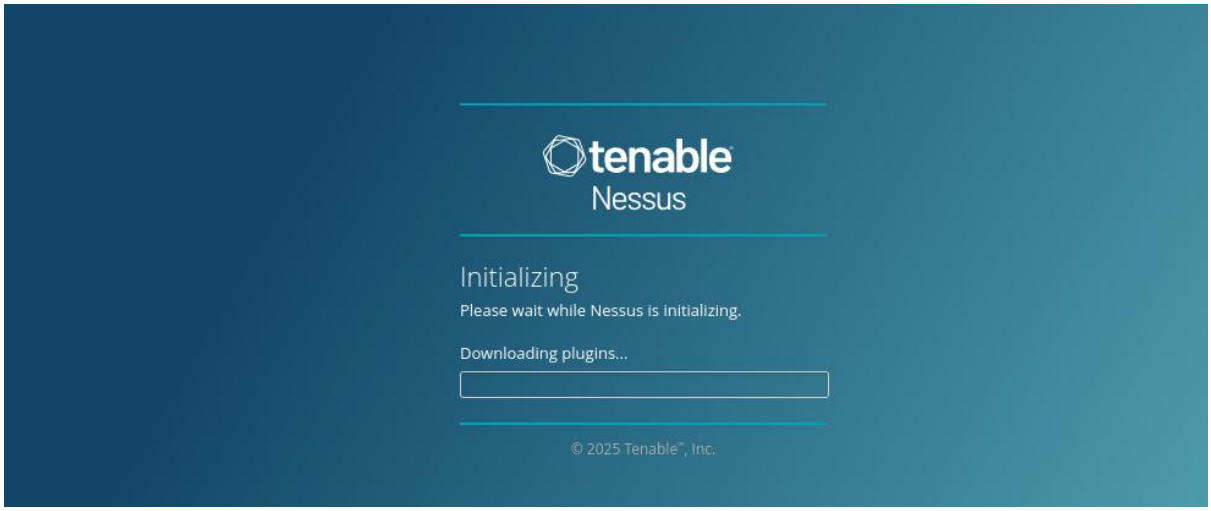


- **Creating an user account** for login.



- After that I was prompted to the **activation code** which I received via email after registration.

After completing the setup, Nessus automatically **downloads and compiles the required plugins**, which are essential for detecting vulnerabilities. This process typically takes **5–10 minutes** to complete, depending on system performance and network speed.



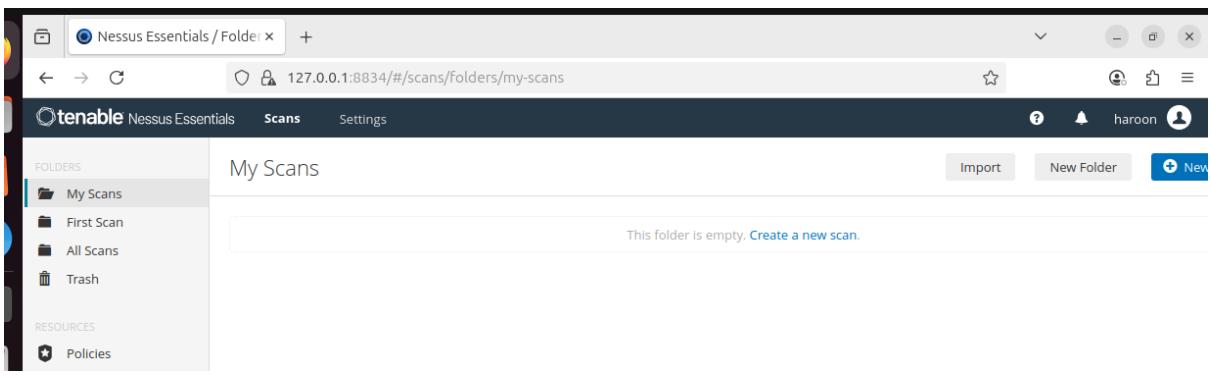
Step 7: Log In to Nessus Web Interface

After the plugin download and compilation completed, I navigated back to the Nessus login page at:

`https://<your_private_vm_ip>:8834`

I logged in using the **admin credentials** that I created during the Nessus setup. Once logged in, the Nessus dashboard was ready for creating and running vulnerability scans.

- **Navigate to Scans:** On the Nessus dashboard, I clicked the **"Scans"** tab on the right-hand side.



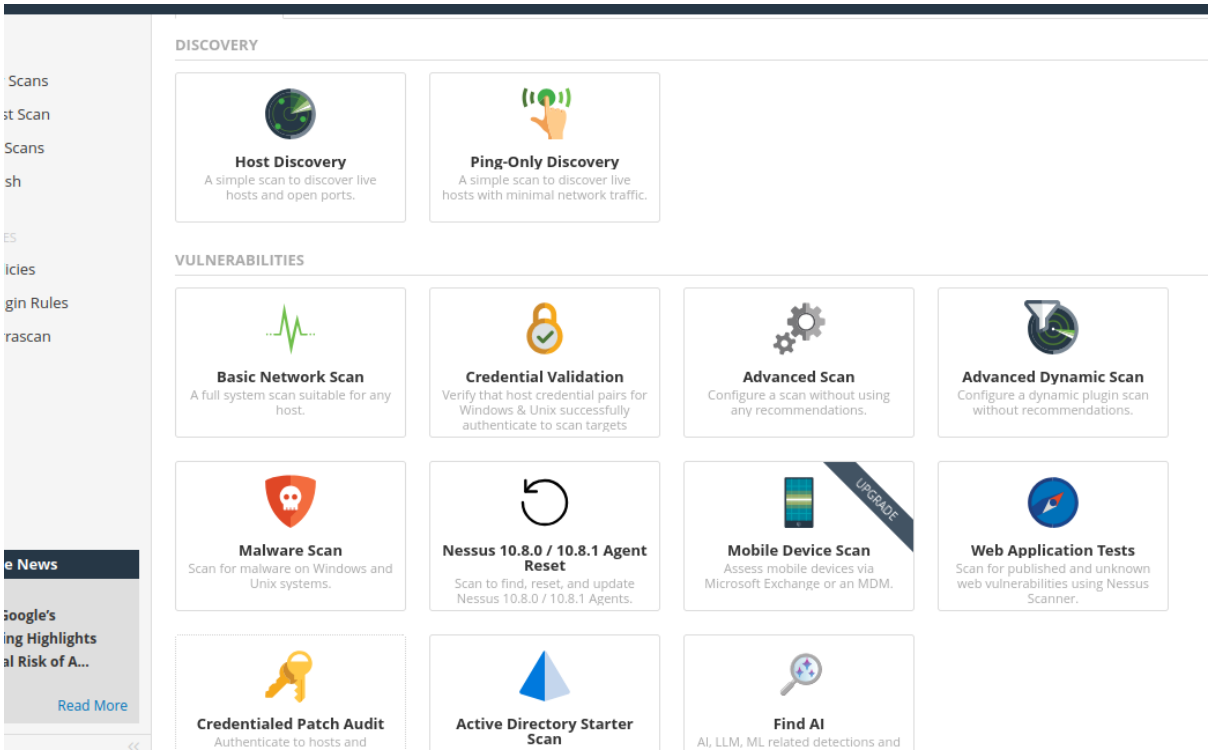
Step 8: Create a New Scan

I created a new vulnerability scan in Nessus by following these steps:

- **Create a New Scan:** I clicked the **"+ New Scan"** button.
- **Choose a Scan Template:** For a basic network scan, I selected the **"Basic Network Scan"** template.

You can also choose other options that are available

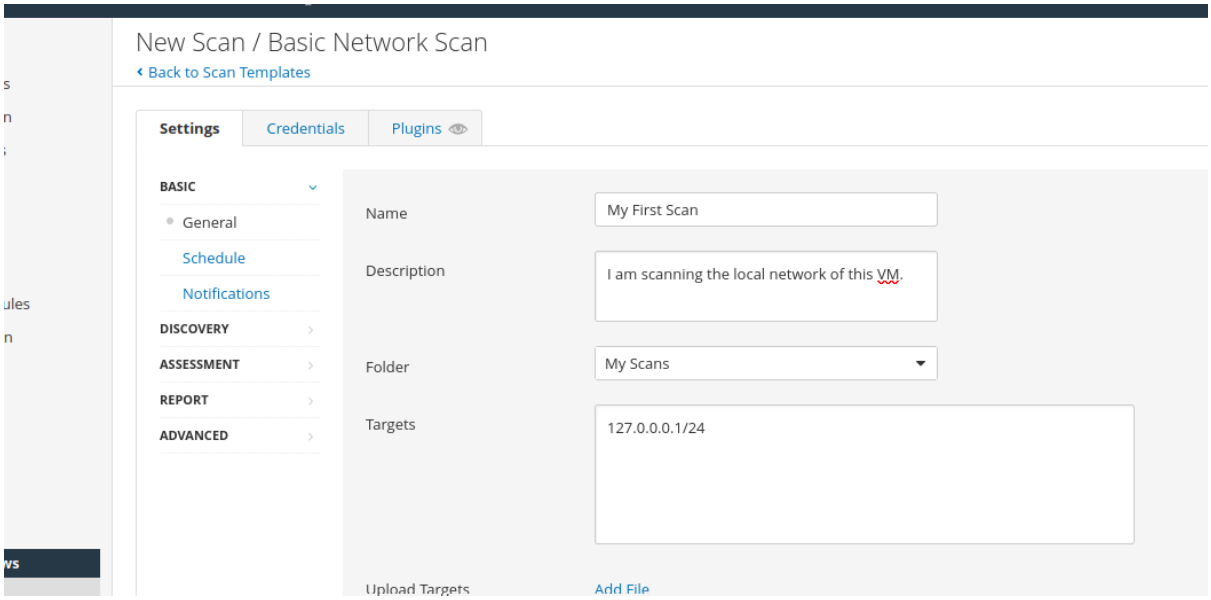
- Host Discovery
- Ping-Only Discovery
- Basic Network Scan
- Credential Validation
- Advanced Scan
- Advanced Dynamic Scan
- Malware Scan
- Nessus Agent Reset
- Mobile Device Scan
- Web Application Tests
- Credentialed Patch Audit Scan
- Active Directory Starter Scan
- Find AI



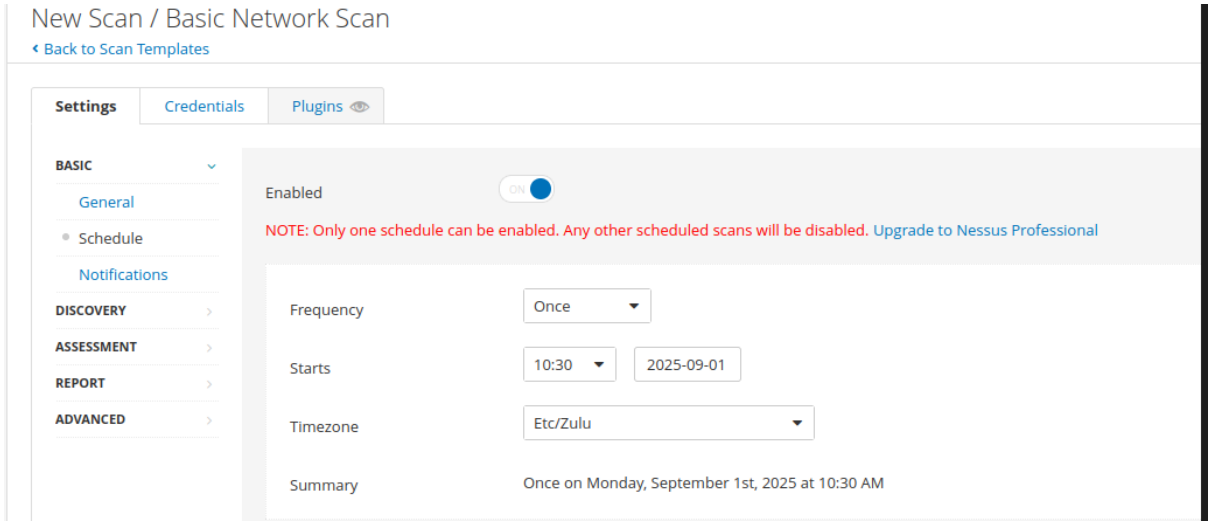
Configure the Scan

After selecting the scan template, I configured the scan settings as follows:

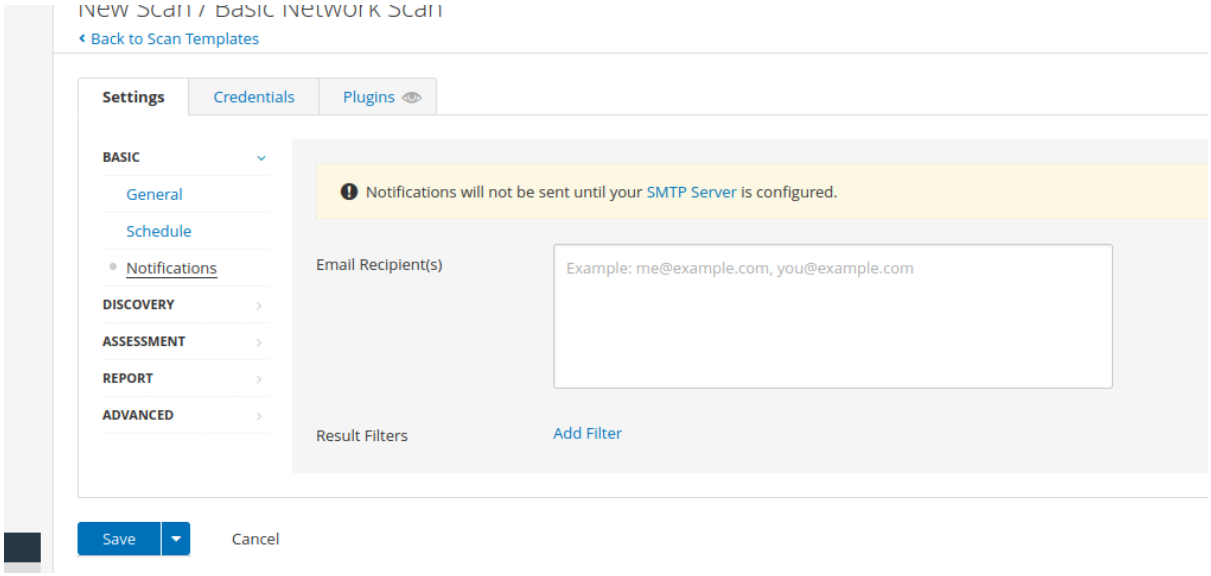
- **Name:** I named the scan **"My First Scan"**.
- **Description:** Optionally added a brief description for reference.
- **Targets:** Entered the **private IP of my VM** which was **127.0.0.1/24** as the target for the scan.
- At first I entered the wrong IP which was **127.0.0.0.1/24** which I corrected it.



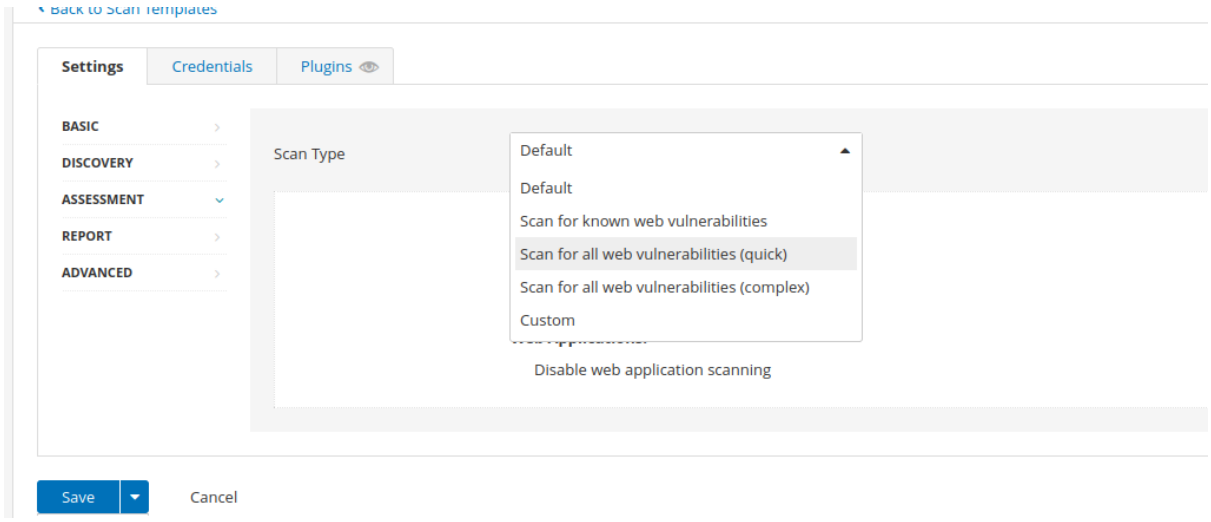
- **Schedule:** Set the scan to **run immediately**.
- Or you can schedule it on specific time.



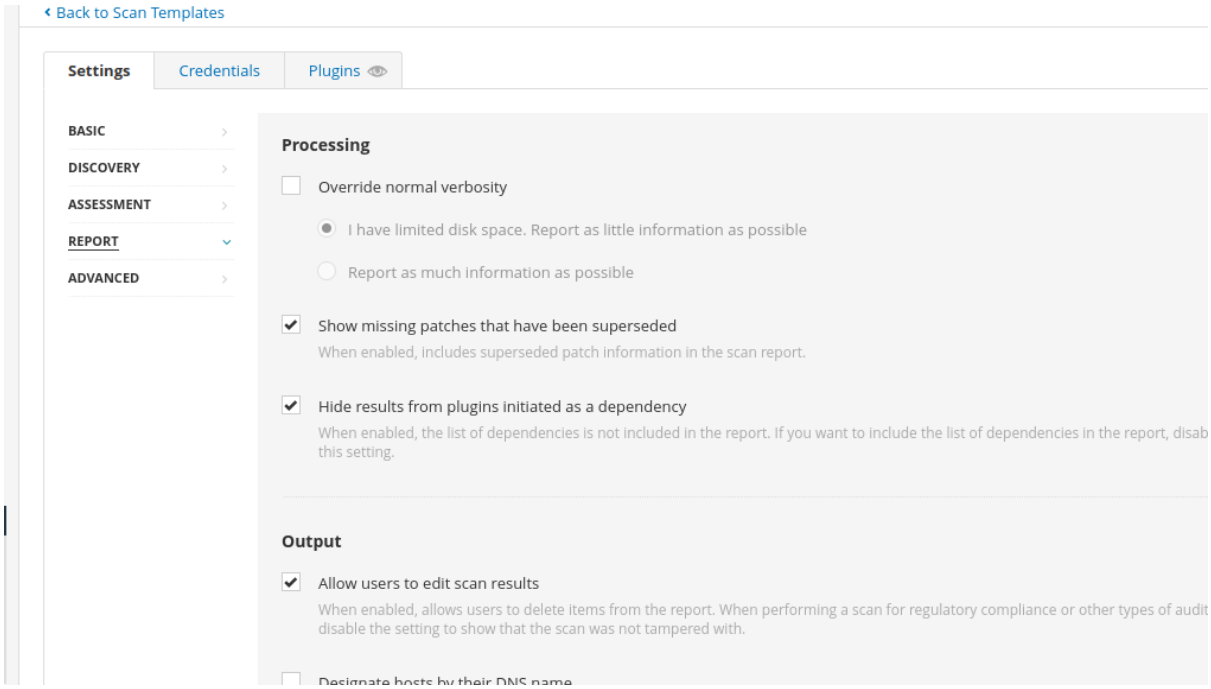
- You can also send notification of the scan to a specific client but you have set **SMTP** server.



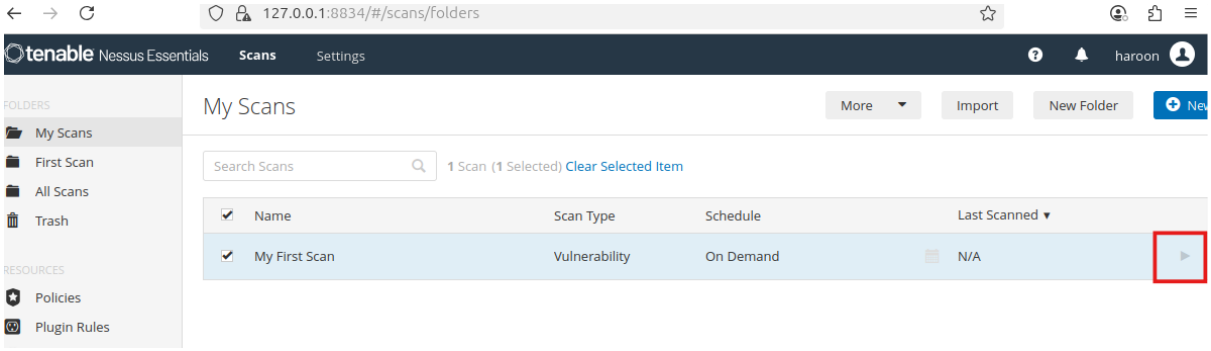
- You can also choose for what kind of **assessment** you need the option are shown in the given screenshot.



- Before running the scan, Nessus provides options to control how the scan report is generated and what information is processed



- Once all settings were configured, I clicked **"Save"** to create the scan.
- Then I lunched the scan by clicking the play button in the red square.



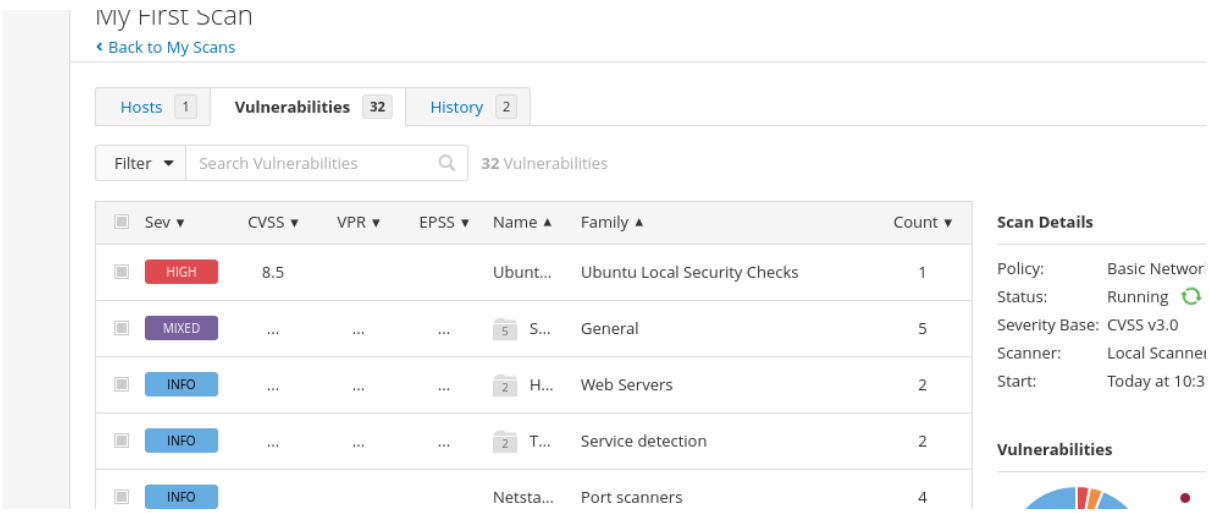
Step 9: Run and Monitor the Scan

After saving the scan, I launched it by clicking the **"Launch"** button. The scan began running, and I could **monitor its progress in real-time** on the dashboard.

Nessus scanned the specified **private IP address**, checking for:

- Open ports
- Running services
- Potential vulnerabilities

The scan continues until it is either completed or manually stopped, providing live updates on progress and findings.



Step 10: Review Scan Results

After the scan completed, I clicked on the **scan name** in the **"My First Scan"** dashboard to view the results.

The results displayed all detected vulnerabilities, categorized by severity:

- **Critical**
- **High**
- **Medium**
- **Low**
- **info**

This categorization allowed me to quickly identify the most serious issues that required immediate attention.

After running the scan, the results showed vulnerabilities categorized as:

- **High** – Critical security issues that pose a significant risk and require immediate remediation.
- **Mixed** – Findings that include multiple severity levels grouped together (for example, a host that has both medium and low vulnerabilities). This indicates a combination of risks that should be reviewed in detail.
- **Info** – Informational findings that are not direct vulnerabilities but provide useful context, such as open ports, detected services, or system configuration details.

Precedence:
High-severity vulnerabilities take precedence, as they represent the greatest security risk. Mixed categories should be analyzed to break down the underlying issues, while informational results can be used to support further investigation and hardening.

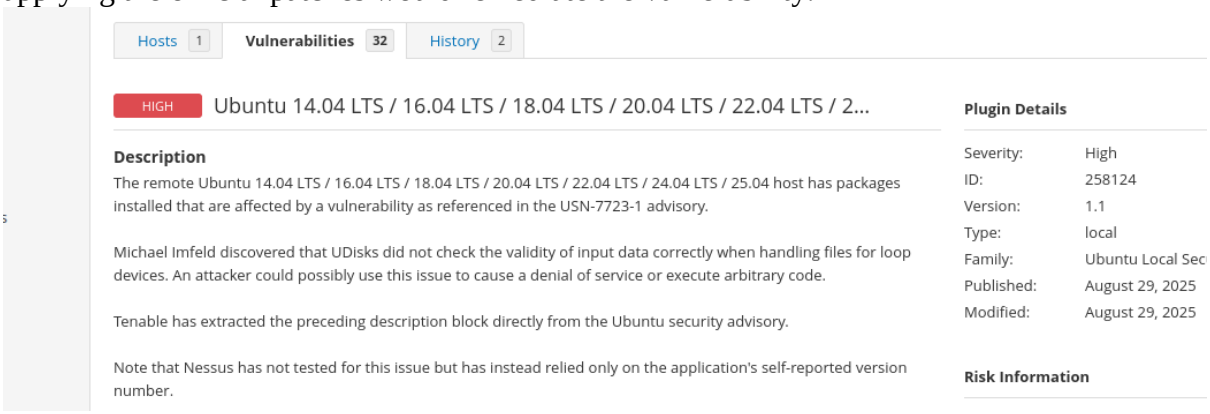
Step 11: Analyzed Vulnerabilities

After clicking on an individual vulnerability from the scan results, I was able to view detailed information, including the affected systems, description, severity level, and remediation guidance.

In my case, Nessus reported a **High severity vulnerability** affecting multiple versions of Ubuntu (14.04 LTS through 25.04 LTS). The issue was related to **UDisk not properly validating input data when handling files for loop devices**. This misconfiguration could allow an attacker to trigger a denial of service or even execute arbitrary code.

Nessus referenced the official Ubuntu security advisory **USN-7723-1** for this issue. Although Nessus did not directly exploit the vulnerability, it flagged it based on the package versions installed on my system.

The scan result also provided the **solution**, which was to **update the affected packages** as outlined in the Ubuntu security advisory (USN-7723-1). Based on this, I confirmed that applying the official patches would remediate the vulnerability.



Conclusion

Through this project, I gained **hands-on experience in system administration, cybersecurity, and vulnerability assessment**. By setting up an Ubuntu VM and installing **Nessus Essentials**, I learned how to properly configure and manage a security tool in a controlled environment.

Performing a vulnerability scan on the local host allowed me to understand how Nessus identifies **open ports, running services, and potential security threats**. I also learned how to **analyze scan results**, interpret the severity of vulnerabilities, and explore recommended remediation steps.

Overall, this project strengthened my practical skills in:

- **System Administration:** Updating and maintaining Linux systems.
- **Security Tool Deployment:** Installing, configuring, and managing Nessus Essentials.
- **Vulnerability Assessment:** Detecting, categorizing, and analyzing system vulnerabilities.
- **Network Security Awareness:** Recognizing potential threats in networked systems.

- **Incident Detection and Response:** Monitoring scan progress and interpreting findings in real-time.

This hands-on experience has provided a solid foundation for further learning in **cybersecurity, ethical hacking, and network protection techniques**.