Applied Cyber Security Industry Led-Course

Instructor: XYZ

Lab Instructor: Moeez Javed

Lab 3: Vulnerability Scanning

Availability:
Monday to Friday: 9 AM – 5 PM (at CUST)
After 5 PM: Please drop a message instead of calling.

Lab Instructor Contact Details:

Phone: +92 333 8744696
Email: moeezjavedmj@gmail.com

Introduction

With the rise of cyber threats and increasing vulnerabilities in modern IT infrastructures, organizations need robust security measures to protect their systems. **Vulnerability assessment tools** such as **OpenVAS** and **Nessus** play a crucial role in identifying security weaknesses and mitigating potential risks.

This manual serves as a step-by-step guide for **installing, configuring, and utilizing OpenVAS and Nessus on Kali Linux**. These tools allow cybersecurity professionals and students to conduct vulnerability scans, analyze security risks, and strengthen network defenses.

What You Will Learn

By following this manual, users will gain hands-on experience in:

- Setting up **OpenVAS** and **Nessus** on **Kali Linux**
- Performing vulnerability scans on networked systems
- Configuring scan parameters for targeted security assessments
- Analyzing scan results to identify security weaknesses
- Implementing best practices for **network security and risk mitigation**

Who Should Use This Manual?

This guide is designed for:

- **Cybersecurity students** who want to learn vulnerability scanning
- **Ethical hackers and penetration testers** aiming to assess system security
- **IT professionals** responsible for securing network environments
- **Anyone interested in learning cybersecurity tools and techniques**

By completing the exercises and practical tasks included in this manual, readers will be well-equipped to use OpenVAS and Nessus for **real-world vulnerability assessments**, making them valuable assets in the field of cybersecurity.

Prepare Kali Linux for the installation of OpenVAS

Unless you have already done so, make sure that the ***Kali Linux is up to date*** and ***install the latest Kali Linux***. You automatically download the latest rules, create admin users, and start the various services. Depending on bandwidth and computer resources, this may take a while.

- ***sudo apt update*** — or use *sudo apt-get update*

```
┌──(root㉿kali)-[/home/kali]
└─# sudo apt-get update
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [20.5 MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [49.0 MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [112 kB]
Get:5 http://kali.download/kali kali-rolling/contrib amd64 Contents (deb) [261 kB]
Get:6 http://kali.download/kali kali-rolling/non-free amd64 Packages [195 kB]
Get:7 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [875 kB]
Get:8 http://kali.download/kali kali-rolling/non-free-firmware amd64 Packages [10.6 kB]
Get:9 http://kali.download/kali kali-rolling/non-free-firmware amd64 Contents (deb) [24.2 kB]
Fetched 71.0 MB in 27s (2,589 kB/s)
Reading package lists ... Done
```

sudo apt-get update

- ***sudo apt upgrade -y***

```
┌──(root㉿kali)-[/home/kali]
└─# sudo apt upgrade

The following packages were automatically installed and are no longer required:
  firebird3.0-common       libgles-dev                  libpaper1
  firebird3.0-common-doc   libgles1                    libpoppler140
  libbfio1                 libglvnd-core-dev            libsuperlu6
  libc++1-19               libglvnd-dev                 libtag1v5
  libc++abi1-19            libgtksourceview-3.0-1       libtag1v5-vanilla
  libcapstone4             libgtksourceview-3.0-common  libtagc0
  libconfig++9v5           libgtksourceviewmm-3.0-0v5   libunwind-19
  libconfig9               libgumbo2                    libwebrtc-audio-processing1
  libdirectfb-1.7-7t64     libhdf5-103-1t64             libx265-209
  libegl-dev               libhdf5-hl-100t64            openjdk-23-jre
  libfmt9                  libjxl0.9                    openjdk-23-jre-headless
  libgdal35                libmbedcrypto7t64            python3-appdirs
  libgl1-mesa-dev          libmsgraph-0-1
Use 'sudo apt autoremove' to remove them.

Upgrading:
  7zip                                libsdl2-ttf-2.0-0
  alsa-ucm-conf                       libseccomp2
  apache2                             libsecret-1-0
  apache2-bin                         libsecret-common
  apache2-data                        libselinux1
  apache2-utils                       libsemanage-common
  apparmor                            libsemanage2
  apt                                 libserd-0-0
  apt-utils                           libsframe1
  aspell                              libsharpyuv0
```

sudo apt upgrade

sudo apt dist-upgrade -y

```
┌──(root㉿kali)-[/home/kali]
└─# sudo apt dist-upgrade -y
The following packages were automatically installed and are no longer required:
  firebird3.0-common        libglvnd-dev                  libqt5×11extras5
  firebird3.0-common-doc    libgtksourceview-3.0-1        libsuperlu6
  imagemagick-6.q16         libgtksourceview-3.0-common   libtag1v5
  libbfio1                  libgtksourceviewmm-3.0-0v5    libtag1v5-vanilla
  libc++1-19                libgumbo2                     libtagc0
  libc++abi1-19             libhdf5-103-1t64              libunwind-19
  libcapstone4              libhdf5-hl-100t64             libwebrtc-audio-processing1
  libconfig++9v5            libjxl0.9                     libx265-209
  libconfig9                libldap-2.5-0                 openjdk-23-jre
  libdirectfb-1.7-7t64      libmagickcore-6.q16-7-extra   openjdk-23-jre-headless
  libegl-dev                libmagickcore-6.q16-7t64      python3-appdirs
  libfmt9                   libmagickwand-6.q16-7t64      python3.12
  libgdal35                 libmbedcrypto7t64             python3.12-dev
  libgl1-mesa-dev           libmsgraph-0-1                python3.12-minimal
  libgles-dev               libpaper1                     python3.12-venv
  libgles1                  libpoppler140
  libglvnd-core-dev         libpython3.12-dev
Use 'sudo apt autoremove' to remove them.

Upgrading:
  blueman                 libsmbclient0   passing-the-hash  python3-tdb       samba-dsdb-modules
  imagemagick             libtalloc2      python3           python3-tk        samba-libs
  kali-linux-headless     libtdb1         python3-aardwolf  python3-venv      smbclient
  libldb2                 libwbclient0    python3-dev       qterminal         winbind
  libnss-winbind          libzbar0t64     python3-ldb       samba
```

Installing OpenVAS on Kali Linux

To install Openvas and its dependencies on our Kali Linux system run the following command:

sudo apt install openvas

or use

sudo apt install gvm

```
┌──(root㉿kali)-[/home/kali]
└─# sudo apt install openvas
Note, selecting 'gvm' instead of 'openvas'
The following packages were automatically installed and are no longer required:
  firebird3.0-common        libglvnd-dev                  libqt5×11extras5
  firebird3.0-common-doc    libgtksourceview-3.0-1        libsuperlu6
  imagemagick-6.q16         libgtksourceview-3.0-common   libtag1v5
  libbfio1                  libgtksourceviewmm-3.0-0v5    libtag1v5-vanilla
  libc++1-19                libgumbo2                     libtagc0
  libc++abi1-19             libhdf5-103-1t64              libunwind-19
  libcapstone4              libhdf5-hl-100t64             libwebrtc-audio-processing1
  libconfig++9v5            libjxl0.9                     libx265-209
  libconfig9                libldap-2.5-0                 openjdk-23-jre
  libdirectfb-1.7-7t64      libmagickcore-6.q16-7-extra   openjdk-23-jre-headless
  libegl-dev                libmagickcore-6.q16-7t64      python3-appdirs
  libfmt9                   libmagickwand-6.q16-7t64      python3.12
  libgdal35                 libmbedcrypto7t64             python3.12-dev
  libgl1-mesa-dev           libmsgraph-0-1                python3.12-minimal
  libgles-dev               libpaper1                     python3.12-venv
  libgles1                  libpoppler140
  libglvnd-core-dev         libpython3.12-dev
Use 'sudo apt autoremove' to remove them.

Installing:
  gvm

Installing dependencies:
  greenbone-security-assistant  gsad  gvm-tools  libmicrohttpd12t64
```

sudo apt install openvas

The next step is to run the installer, which will configure OpenVAS and download various *network vulnerability tests* (NVT) or signatures. Due to a large number of NVTs (*50.000+*), the setting process may take some time and consume a lot of data.

Run the following command to start the setup process:

sudo gvm-setup

```
┌──(root㉿kali)-[/home/kali]
└─# sudo gvm-setup
[>] Starting PostgreSQL service
[>] Creating GVM's certificate files
[>] Creating PostgreSQL database
[*] Creating database user
[*] Creating database
[*] Creating permissions
CREATE ROLE
[*] Applying permissions
GRANT ROLE
[*] Creating extension uuid-ossp
CREATE EXTENSION
[*] Creating extension pgcrypto
CREATE EXTENSION
[*] Creating extension pg-gvm
CREATE EXTENSION
[>] Migrating database
[>] Checking for GVM admin user
[*] Creating user admin for gvm
```

The gvm-setup command will take a **long time** to download all the vulnerabilty definitions (*Notus files, NASL files, SCAP data, CRET-Bund data, gvmd data*).

Hint: OpenVAS will also set up an ***admin account*** and automatically generate a ***password*** for this account which is displayed in the last section of the setup output.

Password reset

Did you forget to note down the password? You can change the admin password using the following commands:

sudo gvmd --user=admin --new-password=passwd

> ***No***te: *if you don't rest the automatically generated admin credentials [password], make sure to safe a copy as you will need it later for login.*

```
┌──(hassen㉿hannachi)-[~]
└─$ sudo gvmd --user=admin --new-password=password

┌──(hassen㉿hannachi)-[~]
└─$
```

update admin user password

Note: To create a new user

sudo runuser -u _gvm — gvmd — create-user=admin2 — new-password=12345

To change the password of the existing user

sudo runuser -u _gvm — gvmd — user=admin — new-password=new_password

Verify the Installation

You can verify your installation with.

- ***sudo gvm-check-setup***

```
┌──(hassen㉿hannachi)-[~]
└─$ sudo gvm-check-setup
gvm-check-setup 23.11.0
  Test completeness and readiness of GVM-23.11.0
Step 1: Checking OpenVAS (Scanner)...
        OK: OpenVAS Scanner is present in version 22.7.9.
        OK: Notus Scanner is present in version 22.6.2.
        OK: Server CA Certificate is present as /var/lib/gvm/CA/servercert.pem.
Checking permissions of /var/lib/openvas/gnupg/*
        OK: _gvm owns all files in /var/lib/openvas/gnupg
        OK: redis-server is present.
        OK: scanner (db_address setting) is configured properly using the redis-
server socket: /var/run/redis-openvas/redis-server.sock
        OK: the mqtt_server_uri is defined in /etc/openvas/openvas.conf
        OK: _gvm owns all files in /var/lib/openvas/plugins
        OK: NVT collection in /var/lib/openvas/plugins contains 88489 NVTs.
        OK: The notus directory /var/lib/notus/products contains 456 NVTs.
Checking that the obsolete redis database has been removed
        OK: No old Redis DB
        Starting ospd-openvas service
        Waiting for ospd-openvas service
        OK: ospd-openvas service is active.
        OK: ospd-OpenVAS is present in version 22.6.2.
Step 2: Checking GVMD Manager ...
        OK: GVM Manager (gvmd) is present in version 23.1.0.
Step 3: Checking Certificates ...
        OK: GVM client certificate is valid and present as /var/lib/gvm/CA/clien
tcert.pem.
        OK: Your GVM certificate infrastructure passed validation.
Step 4: Checking data ...
        OK: SCAP data found in /var/lib/gvm/scap-data.
        OK: CERT data found in /var/lib/gvm/cert-data.
```

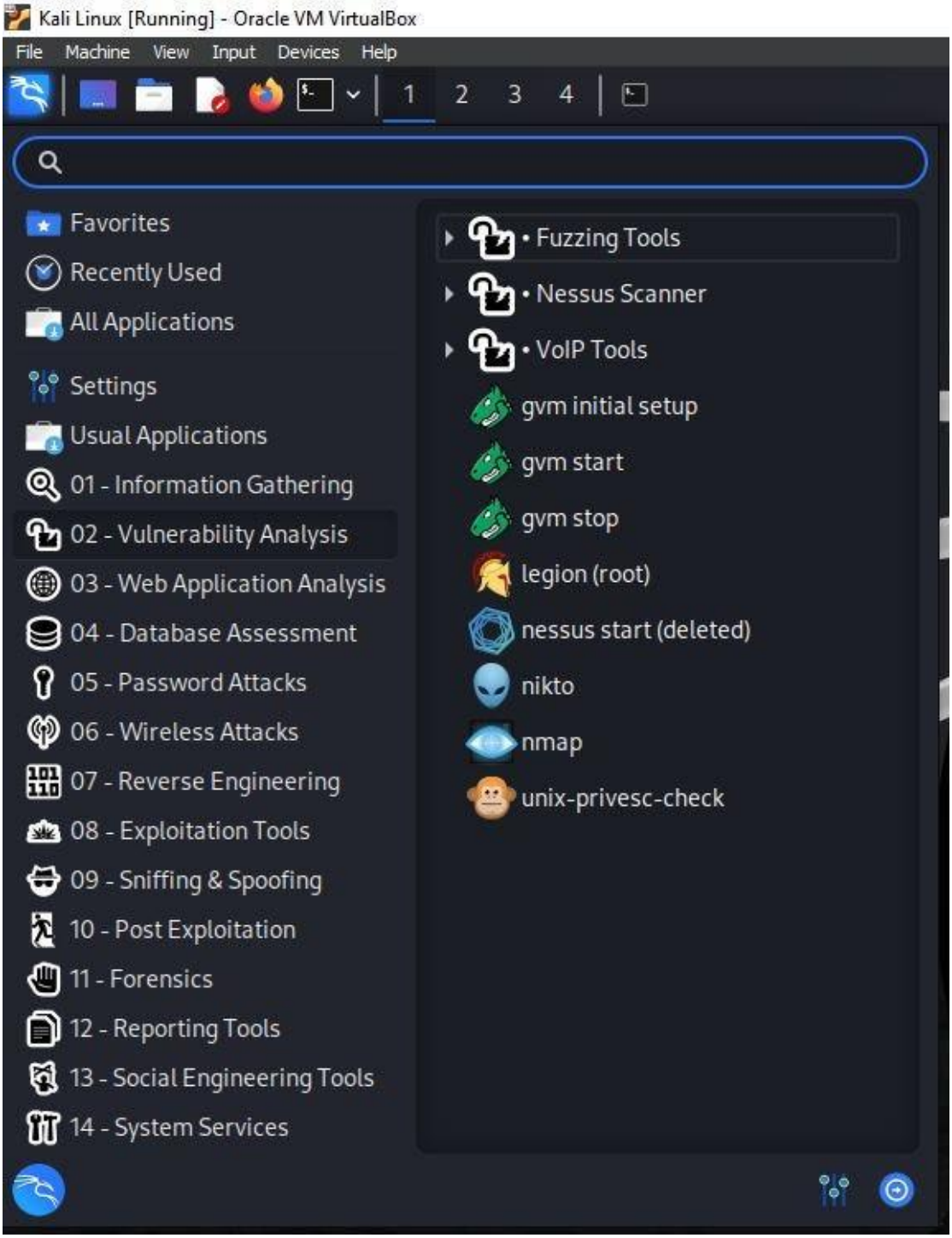
after the process is complete, we should get a confirmation that the installation was completed without error.

```
Step 9: Checking greenbone-security-assistant ...
        OK: greenbone-security-assistant is installed

It seems like your GVM-23.11.0 installation is OK.
```

Starting and stopping OpenVAS

Before starting to install the virtual appliance, the last step I have to consider is to start and stop the OpenVAS service. OpenVAS services consume a lot of unnecessary resources, so it is recommended that you disable these services when you are not using OpenVAS.



Run the following command to start the services:

sudo gvm-start

```
┌──(root㉿kali)-[/home/kali]
└─# sudo gvm-setup
[>] Starting PostgreSQL service
[>] Creating GVM's certificate files
[>] Creating PostgreSQL database
[*] Creating database user
[*] Creating database
[*] Creating permissions
CREATE ROLE
[*] Applying permissions
GRANT ROLE
[*] Creating extension uuid-ossp
CREATE EXTENSION
[*] Creating extension pgcrypto
CREATE EXTENSION
[*] Creating extension pg-gvm
CREATE EXTENSION
[>] Migrating database
[>] Checking for GVM admin user
[*] Creating user admin for gvm
```

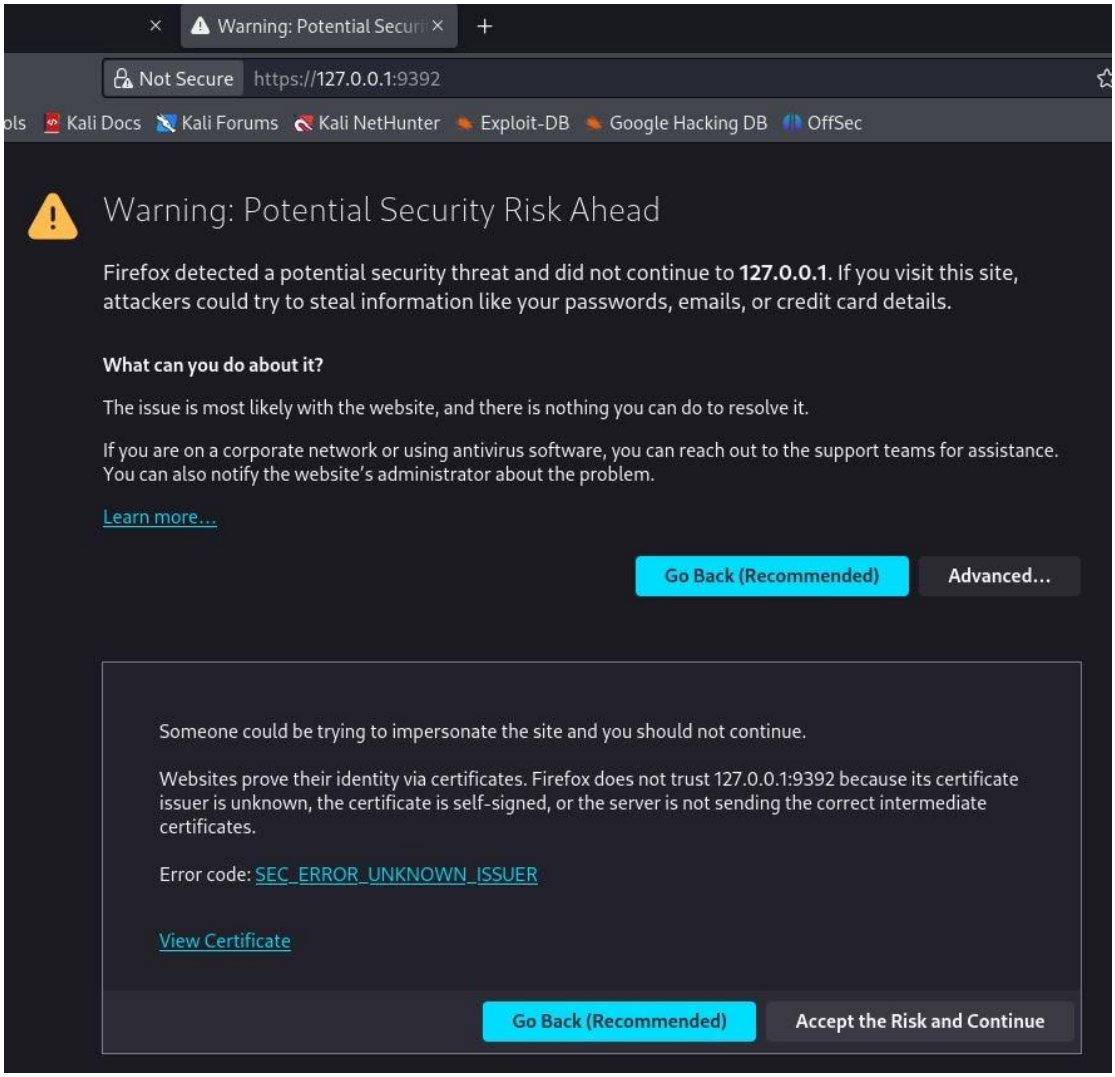
```
┌──(hassen㉿hannachi)-[~]
└─$ sudo gvmd --user=admin --new-password=password

┌──(hassen㉿hannachi)-[~]
└─$
```

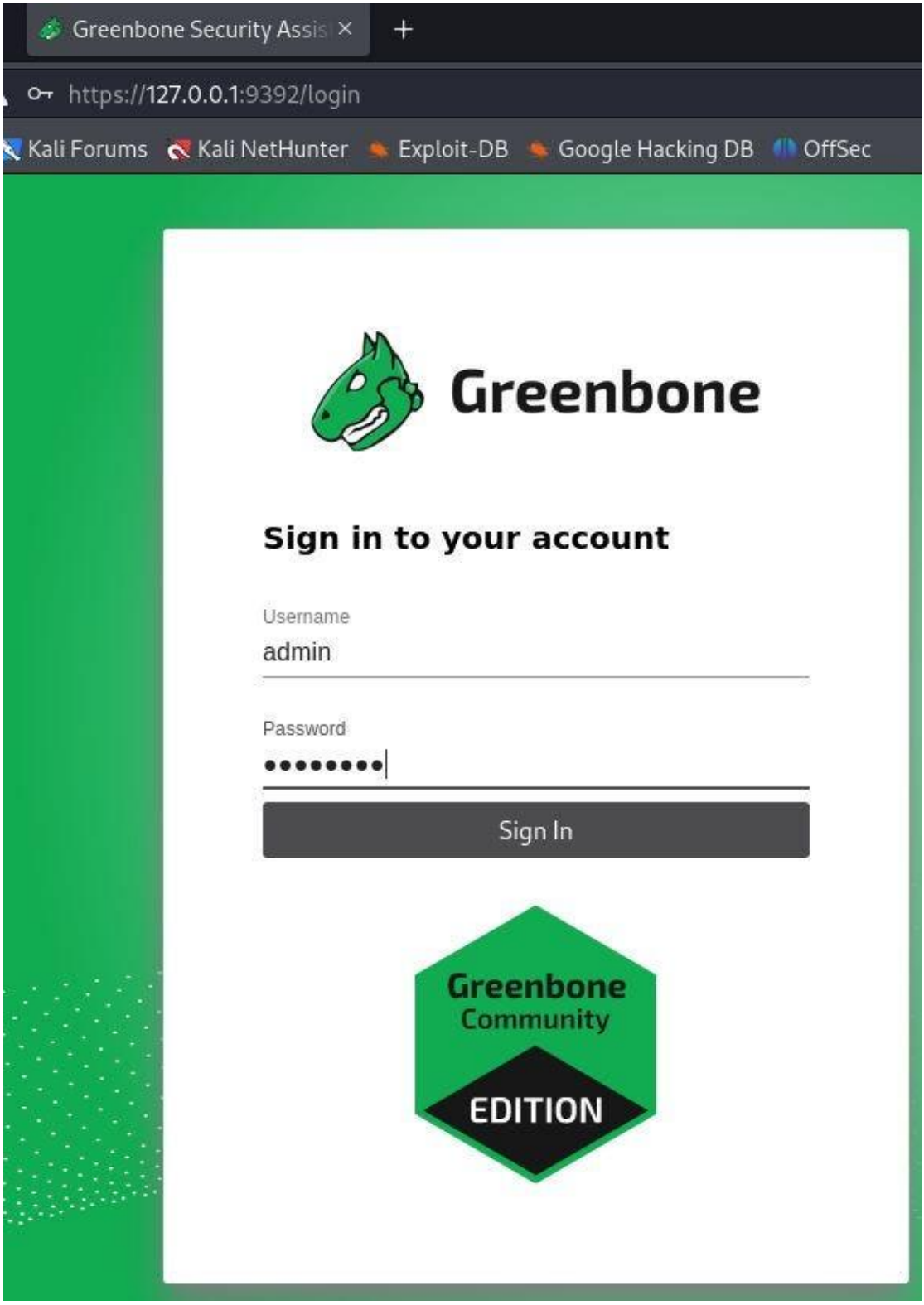
Hint: *To stop the OpenVAS services again, run:* ***sudo gvm-stop***

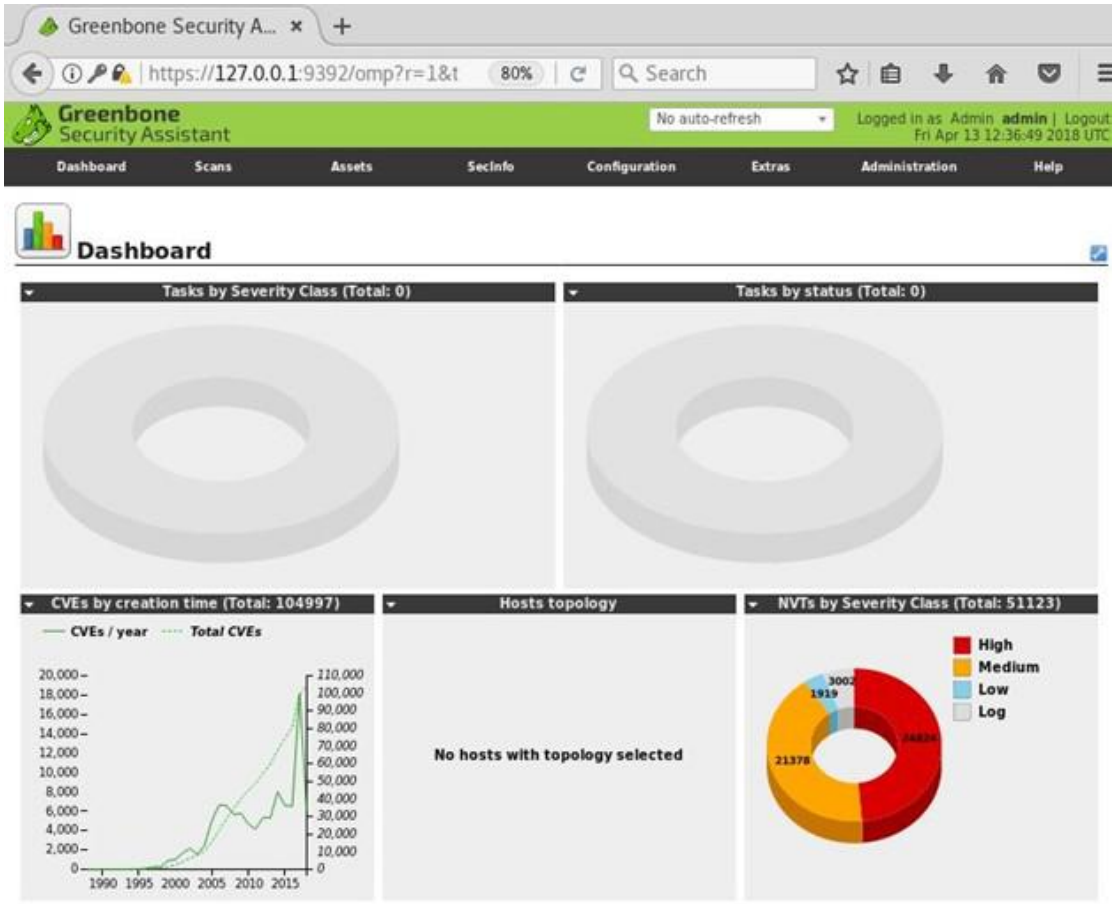
After the configuration process is complete, all the necessary OpenVAS processes will start and the web interface will open automatically (In my case I had to open the browser manually). The web interface is *running locally* on *port 9392* and can be accessed through **https://localhost:9392**

First time you want to open this URL you will get a security warning. Click on **Advanced** and **Accept the Risk and Continue**.



The next step is to accept the self-signed certificate warning and use the automatically generated admin credentials (in my case I rest the admin password) to login on to the web interface:

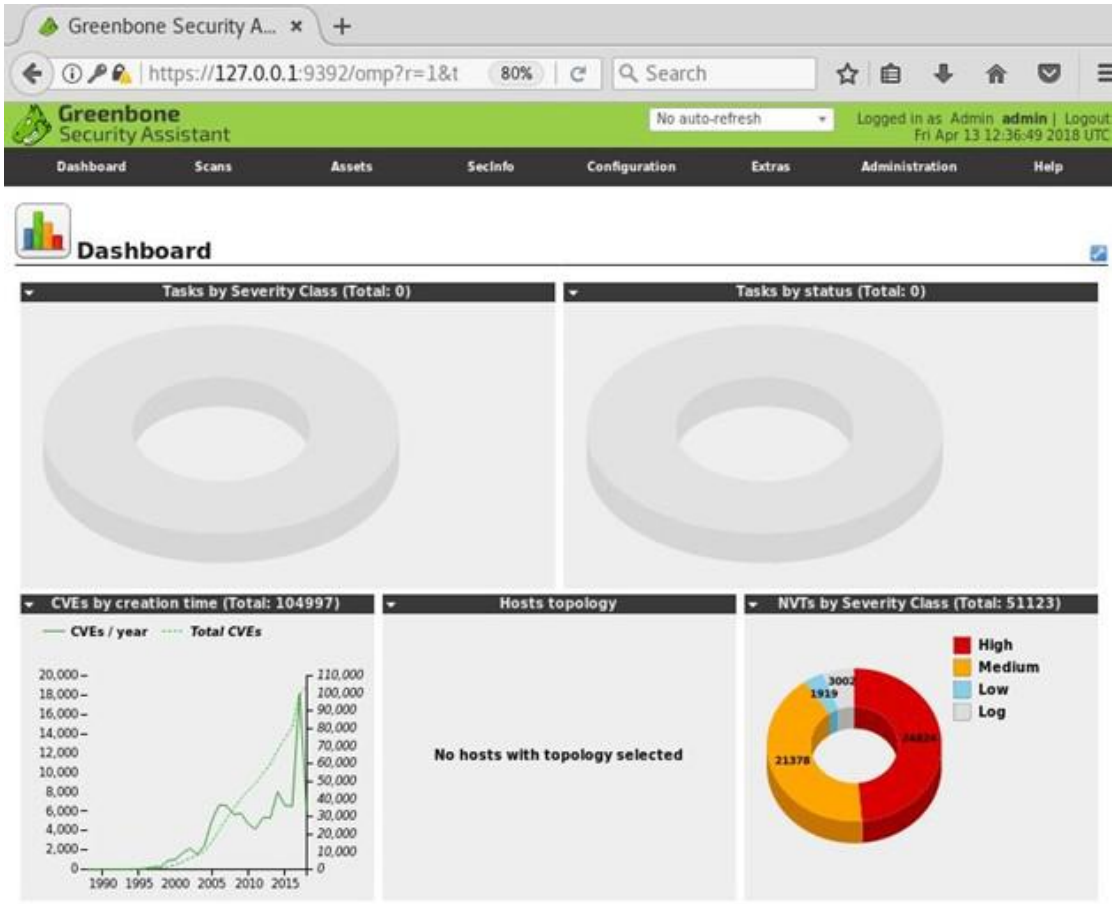

Greenbone Security Assis
https://127.0.0.1:9392/login
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
Greenbone
Sign in to your account
Username
admin
Password
Sign In
Greenbone
Community
EDITION



Configuration for a new target

Begin by navigating to ***Scans > Tasks*** and clicking on the ***purple magic wand icon*** to begin the basic configuration wizard. After successfully navigating to the wizard, you should see a pop-up window similar to the one shown above. You can set up the initial scan of the local host here to make sure everything is set up correctly.

Scanning may take a while. Please allow OpenVAS enough time to complete the scan. You will then see a new dashboard for monitoring and analyzing your completed and ongoing scans, as shown below.

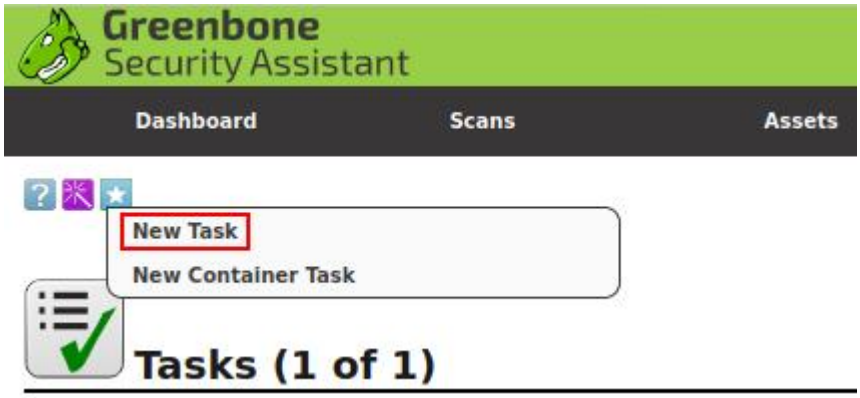


Schedule the scanning process

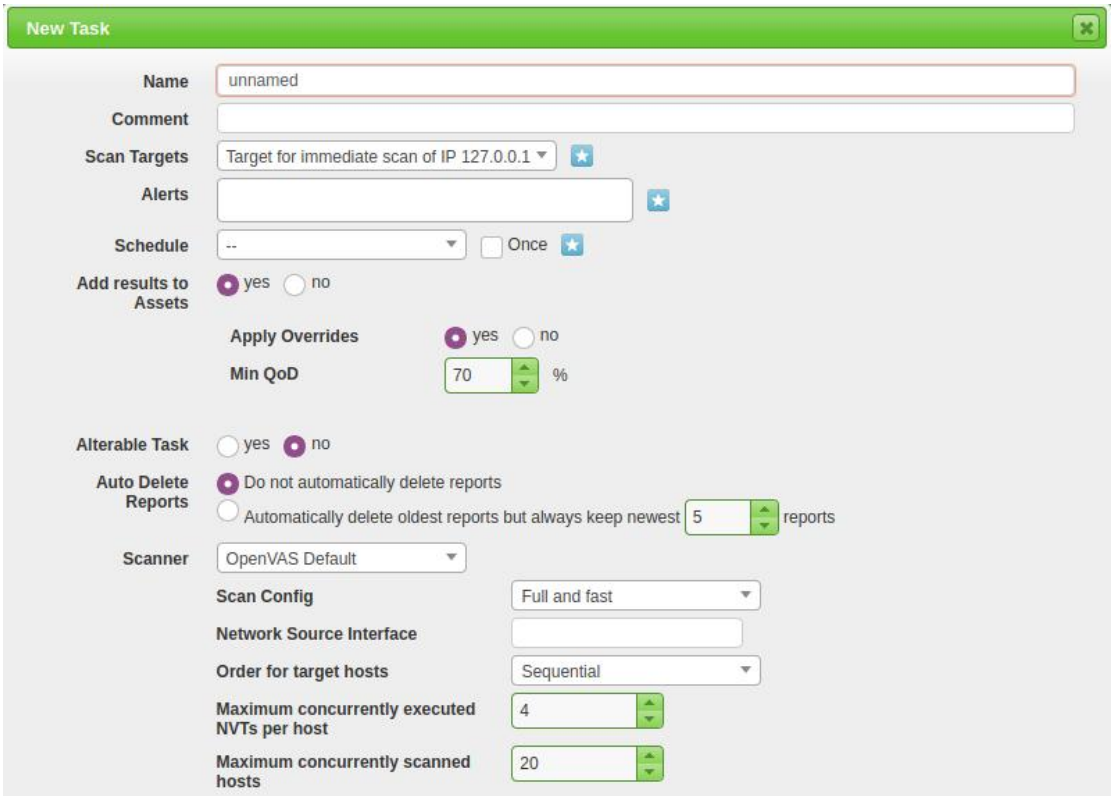
Now that we know everything is normal, we can take a closer look at OpenVAS and how it works. Expand the car to **scan and> start the task** of creating a scan task for the managed computer.

Creating a Task

To create a custom task, navigate to the star icon in the upper right corner of the taskbar and select New task.



After selecting ” New Task” from the drop-down menu, you will see a large pop-up window with many options. We will introduce each option part and its purpose.



For this task, we’ll be specializing only in the Name, Scan Targets, and Scanner Type, and Scan Config. In later tasks, we will be focusing on the opposite choices for additional advanced configuration and implementation/automation.

1. **Name**: permits North American country to line the name the scan are going to be referred to as inside OpenVAS

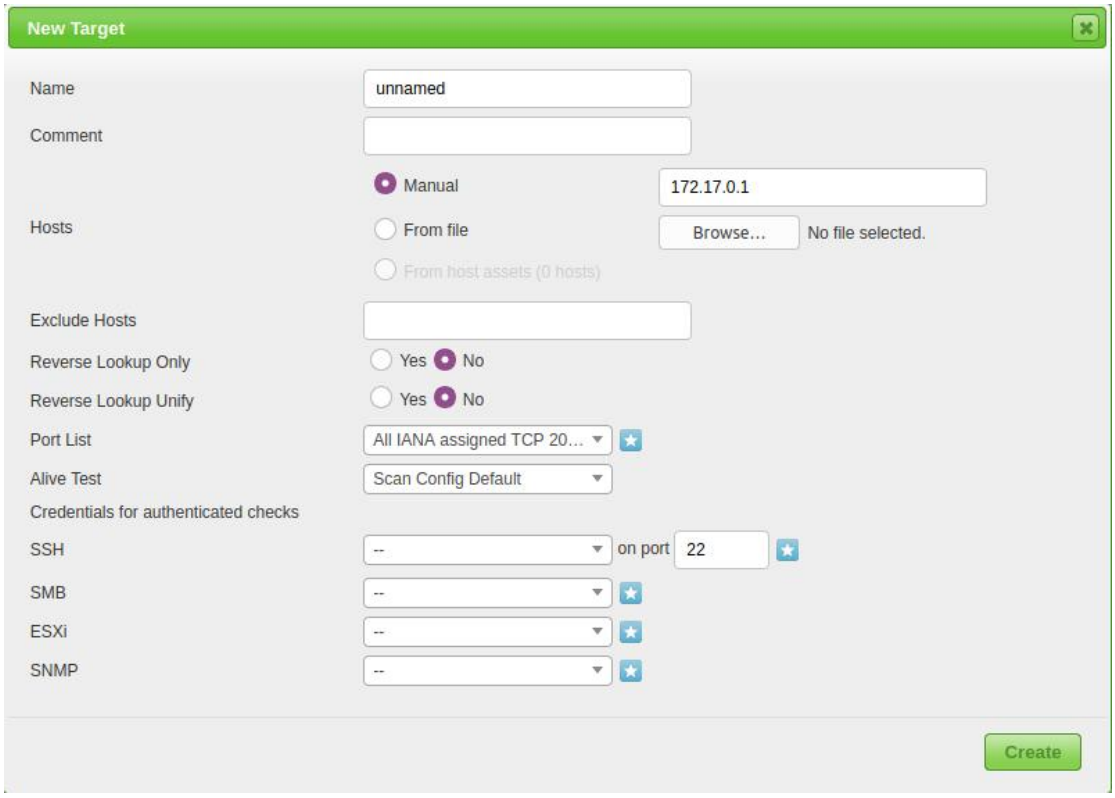
2. **Scan Targets:** The targets to scan, can embrace Hosts, Ports, and Credentials. to make a brand new target you may follow another pop-up, this can be lined later during this task.

3. **Scanner:** The scanner to use by default will use the OpenVAS design but you’ll be able to set this to any scanner of your selecting within the settings menu.

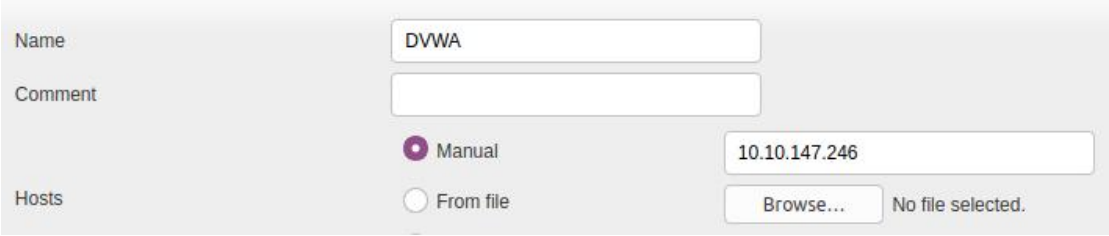
4. **Scan Config**: OpenVAS has seven totally different scan sorts you can choose from and can be used supported however you’re aggressive or what info you wish to gather from your scan.

Scoping a New Target

To scope a new target, navigate to the star icon next to Scan Targets.



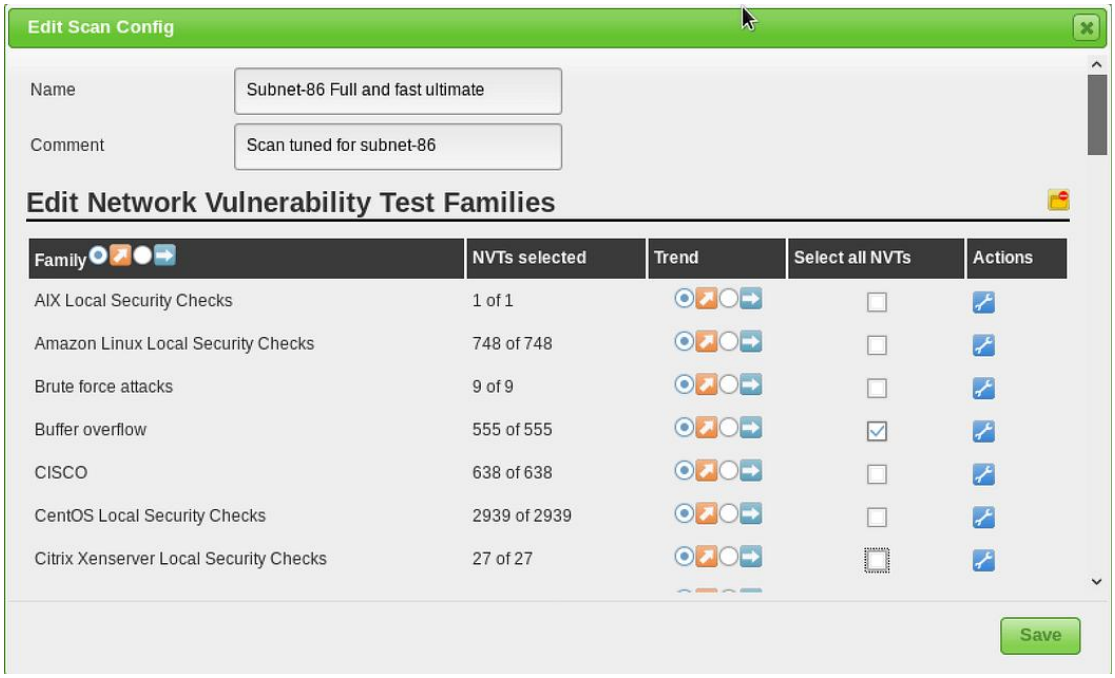
Above is that the menu for configuring a replacement target. the 2 main choices you may have to be compelled to assemble are the Name and therefore the Hosts. This procedure is fairly uncomplicated and different options will solely be employed in advanced vulnerability management solutions. These are going to be lined in later tasks.



Now that we've got our target scoped we are able to still produce our task and start the scan. When the task is created, you'll come to the scanning management panel, wherever you'll track and execute the task. To run the task, navigate to the run icon within the operation.

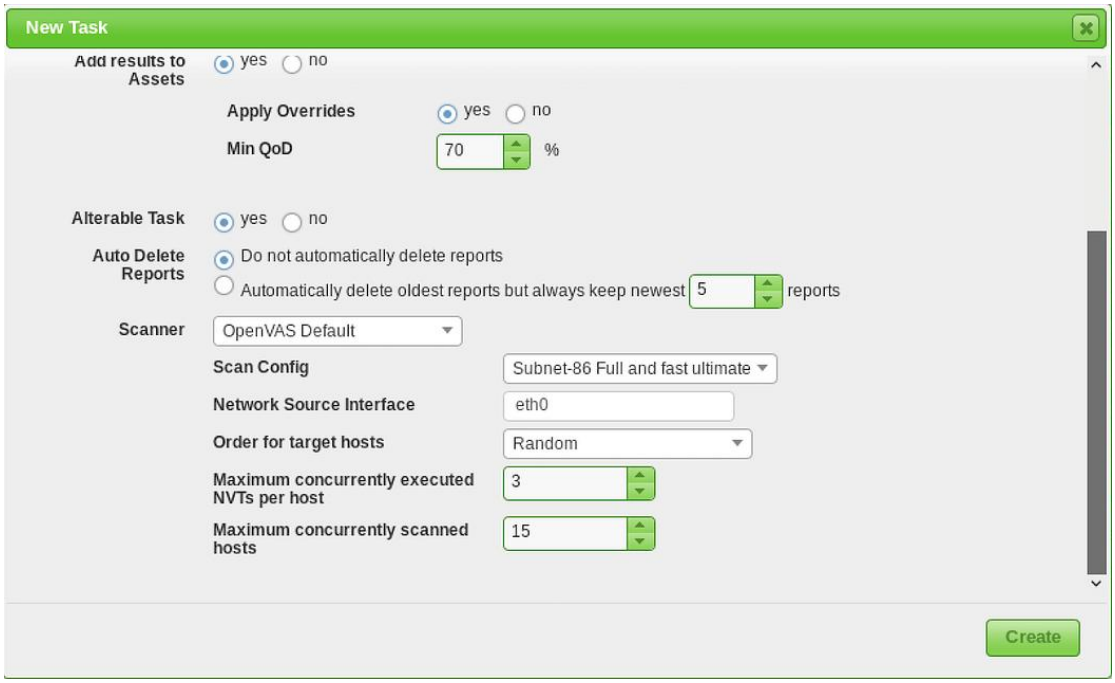
Scan Configuration

Prior to launching a vulnerability scan, you should fine-tune the Scan Config that will be used, which can be done under the "Scan Configs" section of the "Configuration" menu. You can clone any of the default Scan Configs and edit its options, disabling any services or checks that you don't require. If you use Nmap to conduct some prior analysis of your target(s), you can save hours of vulnerability scanning time.

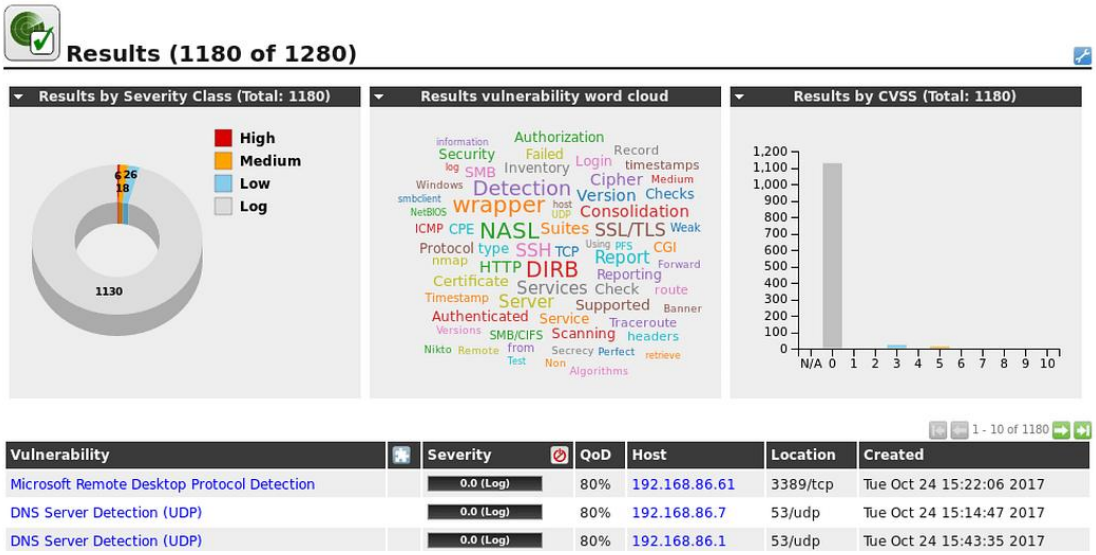


Task Configuration

Your credentials, targets, and scan configurations are setup so now you're ready to put everything together and run a vulnerability scan. In OpenVAS, vulnerability scans are conducted as "Tasks". When you set up a new task, you can further optimize the scan by either increasing or decreasing the concurrent activities that take place. With our system with 3GB of RAM, we adjusted our task settings as shown below.

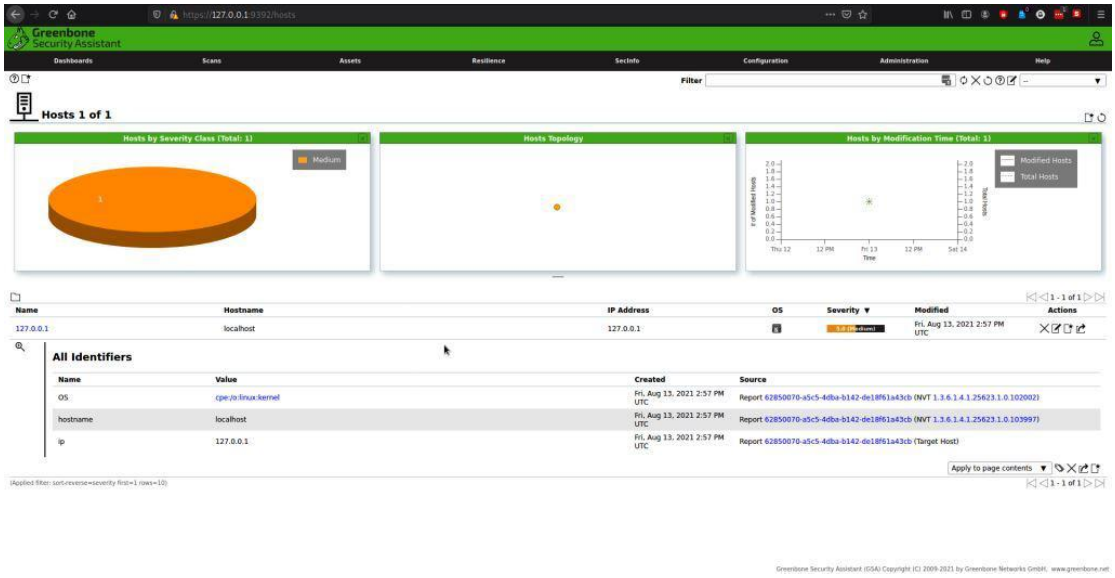


With our more finely-tuned scan settings and target selection, the results of our scan are much more useful.



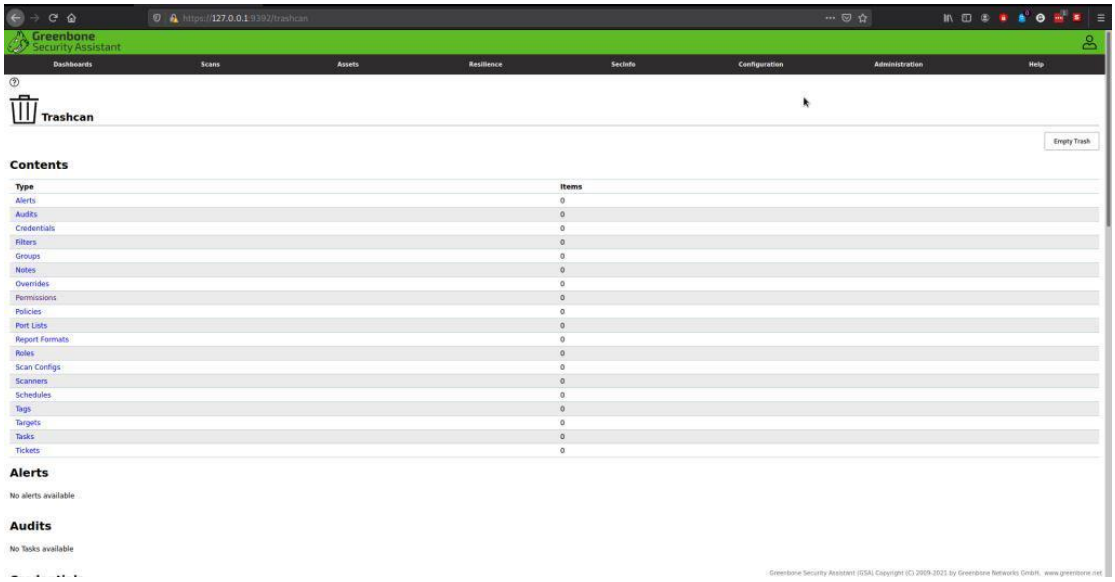
Assets

It permits visualizing the vulnerability of the parts akin to hosts or in operation systems:



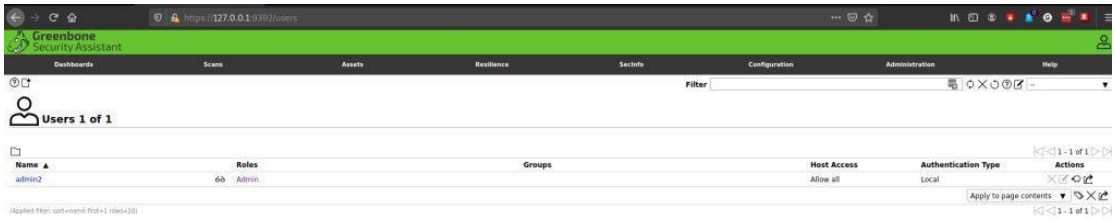
Additional features

Allow adding common parameters to OpenVAS:



Administration

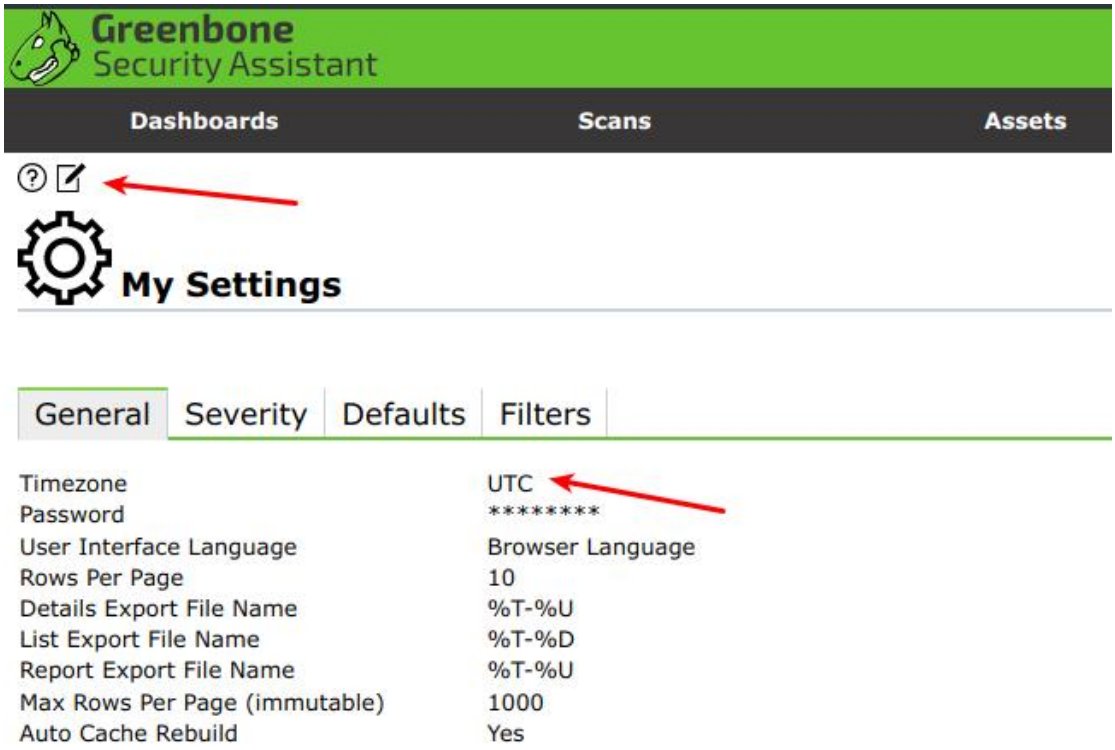
As the name suggests, you can manage passwords, users, etc.:



Change timezone

Note: Recommend setting the timezone as UTC, the report displays UTC time only no matter what timezone you set

Top-Right corner > My Settings



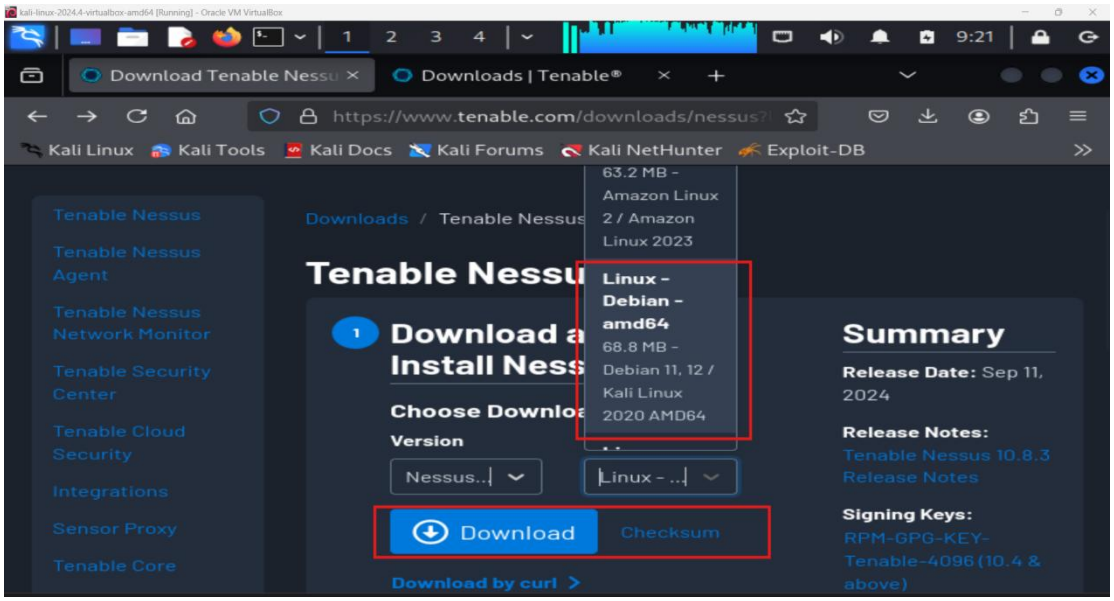
With the wide range of options available in OpenVAS, we were only really able to just scratch the surface in this post but if you take your time and effectively tune your vulnerability scans, you will find that the bad reputation of OpenVAS and other vulnerability scanners is undeserved. The number of connected devices in our homes and workplaces is increasing all the time and managing them becomes more of a challenge. Making effective use of a vulnerability scanner can make that management at least a little bit easier.

Now

How to Install Nessus on Kali Linux

This section will guide you through the process of downloading, installing and running Nessus Essentials on Kali Linux. Nessus does not come pre-installed in Kali and you have to download it from the Nessus website.

Download Nessus

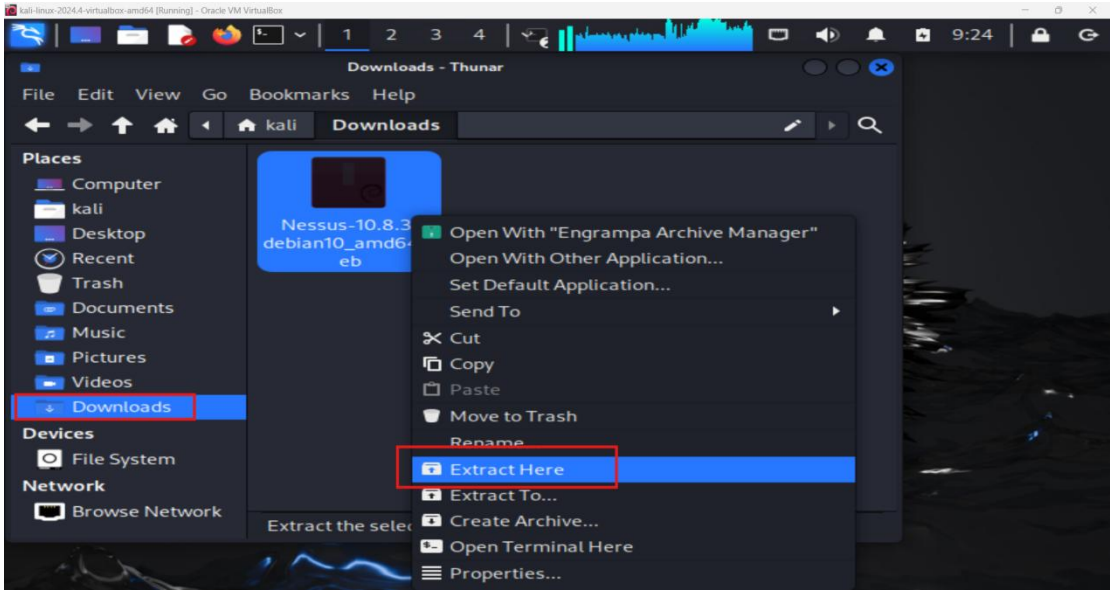


To download Nessus, visit the download page and select the **Linux-Debian-amd64.**

Then select "Download" to download the file to Kali. Alternatively, you can use the command curlto download the file or download and install Nessus as a Docker image.

Installing Nessus

To install Nessus, simply enter the following command in the terminal, making sure you are in the same folder as the downloaded file:



sudo dpkg -i Nessus-10.8.3-debian10_amd64.deb

To start installing the plugins required before using Nessus, enter the following command at the command line:

sudo systemctl start nessusd.service

After starting the service, go to https://kali:8834/ in your browser to access and set up Nessus.

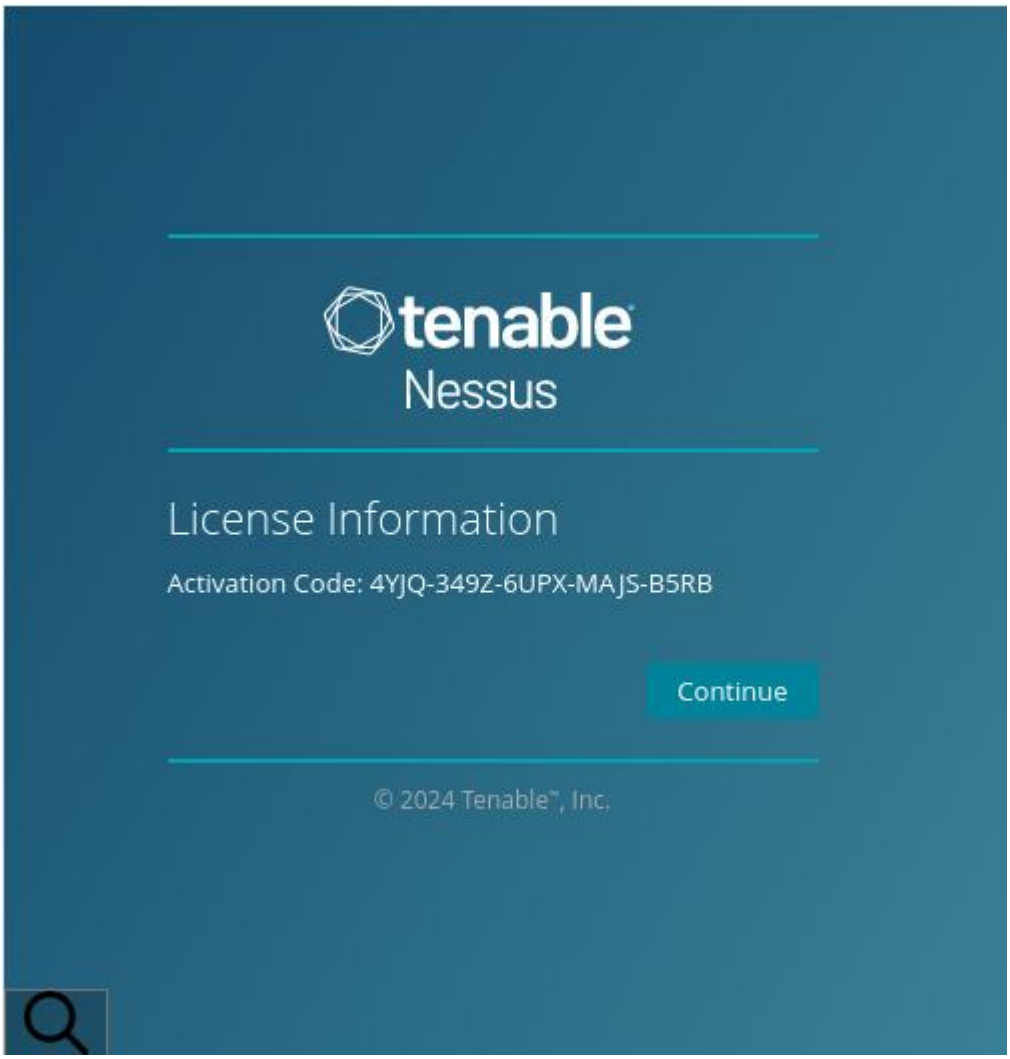
When you try to access the URL, a warning message will appear. Click on "Advanced…" and select "Accept the Risk and Continue."

A Nessus welcome screen will then appear. Click "Continue" to proceed.

Select "Register for Nessus Essentials" on the next screen and click "Continue."

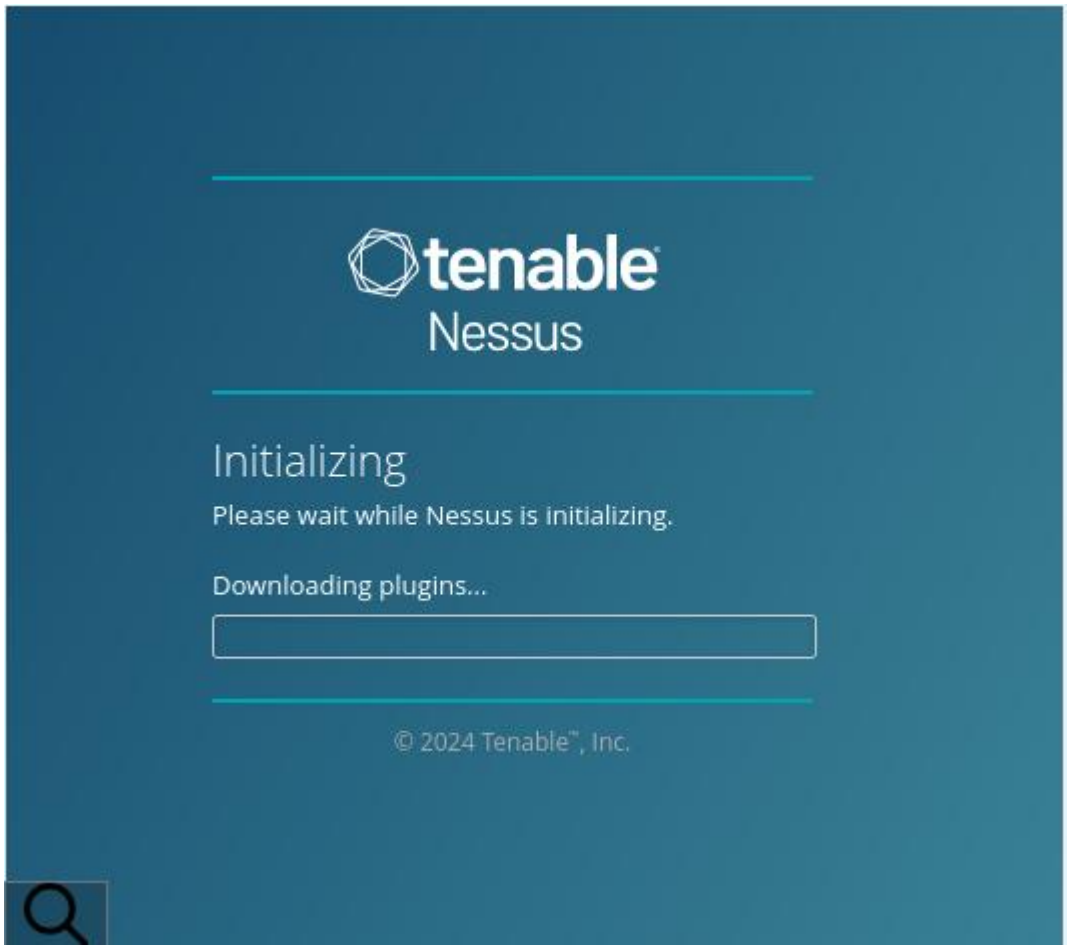
On the next screen, enter your name and email address and click "Register" to continue.

On the next screen, enter your name and email address and click "**Register**" to continue.



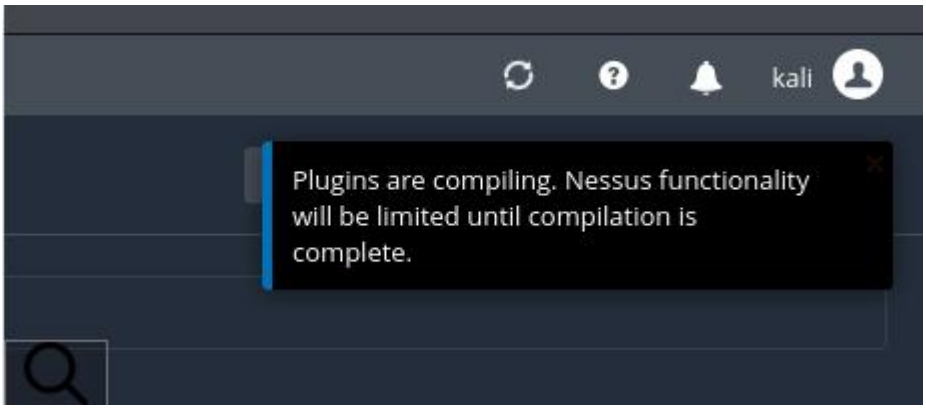
On the next screen, you need to create a Nessus admin account, which will be used to log into Nessus.

Nessus will now start downloading the plugins.



Once the process is complete, you will be taken to the Nessus dashboard.

From here, Nessus will start setting up the plugins, which will take some time to complete. So grab a coffee and relax while Nessus does his thing.



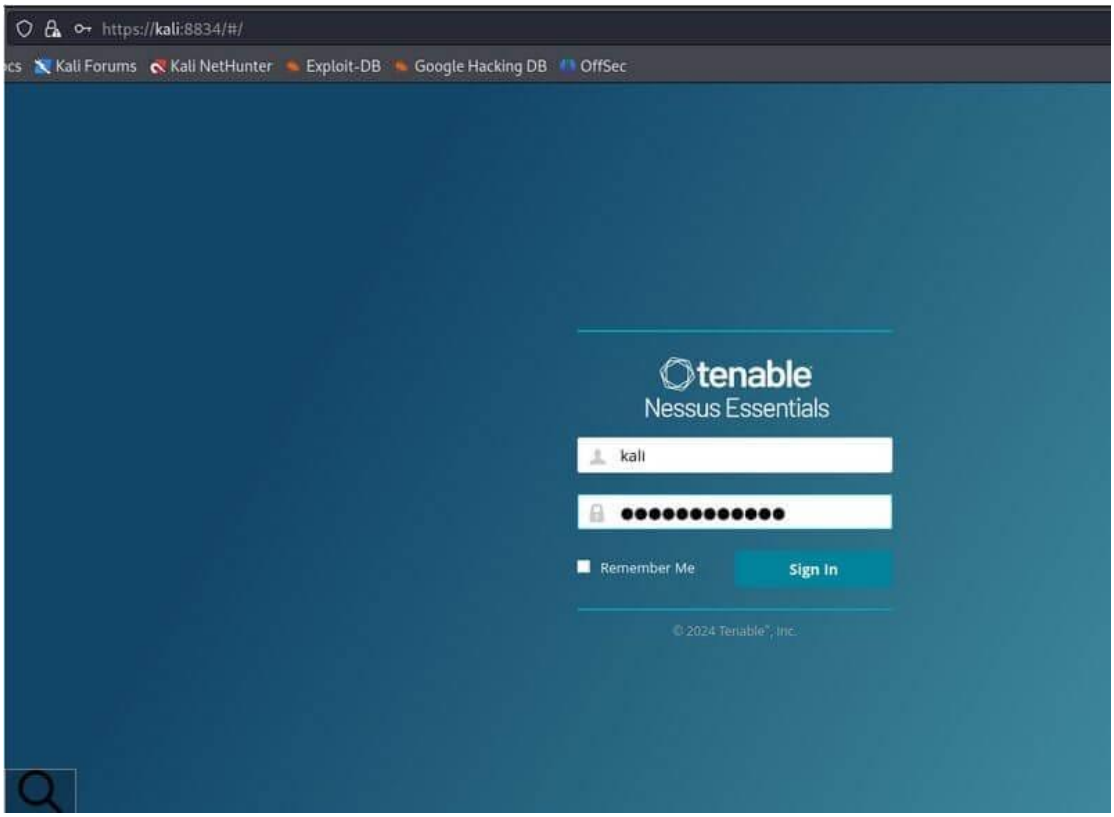
Launch Nessus

To start Nessus, use the command:

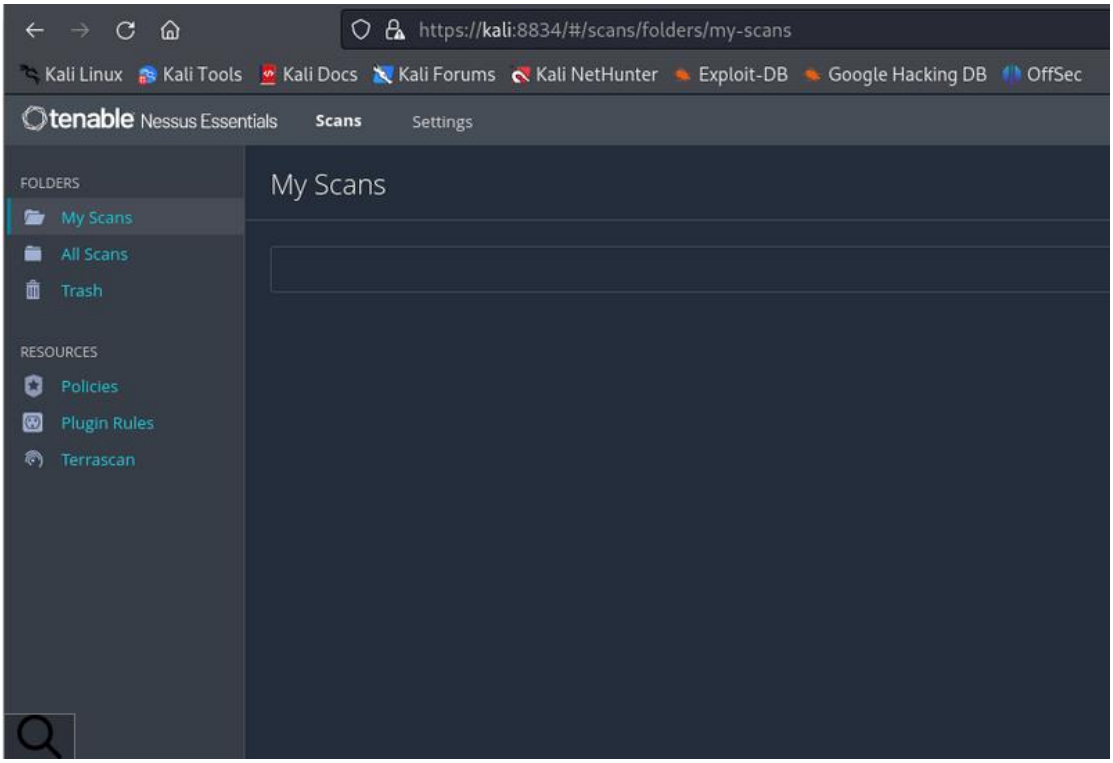
sudo systemctl start nessusd.service

and then open https ://kali:8834/ in your browser.

You will need to log in with the details you set up earlier.



Once you're logged in, you can start using Nessus.



Once you are done using Nessus, you can stop the service with the command:

sudo systemctl stop nessusd.service

Later in this guide, we'll show you how to use Nessus in Kali.

Student Task:

Task 1: Install OpenVAS and Nessus

- Update and upgrade **Kali Linux**.
- Install **OpenVAS** and **Nessus**.
- Set up both scanners and verify their installation.

Task 2: Configure and Start OpenVAS

- Run **OpenVAS setup**.
- Start and stop **OpenVAS services**.
- Log in to the **web interface**.

Task 3: Perform a Basic Vulnerability Scan Using OpenVAS

- Create a **scan task**.
- Configure **scan settings**.

- Run and analyze **scan results**.

Task 4: Configure and Start Nessus

- Start the **Nessus service**.
- Register and activate **Nessus Essentials**.
- Log in and explore the **Nessus interface**.

Task 5: Perform a Basic Vulnerability Scan Using Nessus

- Create a **new scan task**.
- Configure **scan settings and targets**.
- Run and analyze **scan results**.

Task 6: Compare OpenVAS and Nessus Results

- Conduct the **same scan on a local network**.
- Compare the **vulnerabilities detected** by both tools.
- Document **key differences and insights**.

Task 7: Prepare and Submit the Report

- Summarize the **findings from OpenVAS and Nessus**.
- Include **screenshots of scan results**.
- Provide an **analysis of vulnerabilities detected**.
- Write **conclusions and recommendations** based on the scan results.
- Submit the **final report**.