

Task 3

In this task we are going to perform a basic vulnerability scan on our PC, to do that first let's install an automation tool that will do the scan for us. The most common tools for this case are OpenVAS and nessus, let's go with OpenVAS for our task.

OpenVAS Installation process (Kali linux):

- In your terminal run “ `sudo apt install openvas -y` ”
- This will install all the necessary components including the scanner web UI etc.
- Now to initialise OpenVAS run “ `sudo gvm-setup` ”.
- During this process you will see the auto generated user and password make sure to copy it. It will look something like in fig1

```
[+] Done
[*] Please note the password for the admin user
[*] User created with password '44bdd57b-71a0-41ab-b452-6306ddd0aff7'.

[>] You can now run gvm-check-setup to make sure everything is correctly configured

(kali㉿kali)-[~/ELI/day3]
$ _
```

fig1

- After this is done run “ `sudo gvm-start` ” to start OpenVAS
- Then open “ `https://127.0.0.1:9392` ” in your browser and insert the credentials you got when OpenVAS was initializing like in fig2.

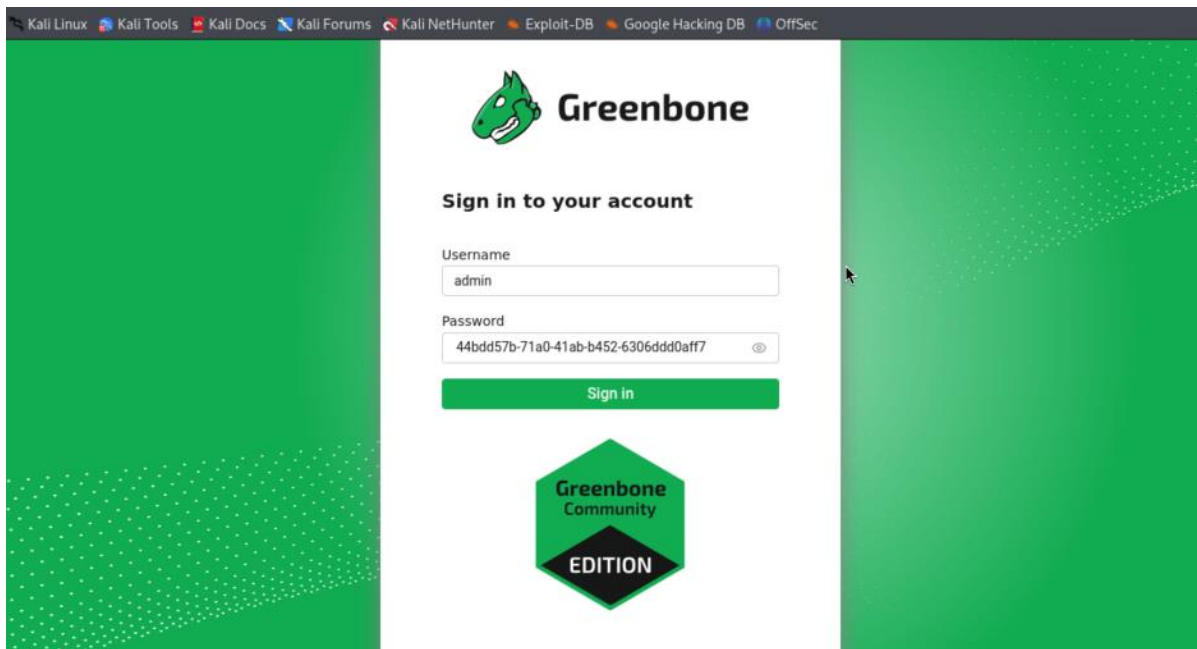


fig2

Pro Tip: run “ sudo systemctl enable nessusd ” to start nessus on boot

: also you can check the status of whether your nessus is working or not by this command:
“ sudo systemctl status nessusd ”

Vulnerability Scan Process:

1. After you are logged in you will see the Dashboard, in there go to configurations tab in the left options and choose targets under, then add your machine using the new target button and put “ 127.0.0.1 ” as target host and name the machine something like in fig3

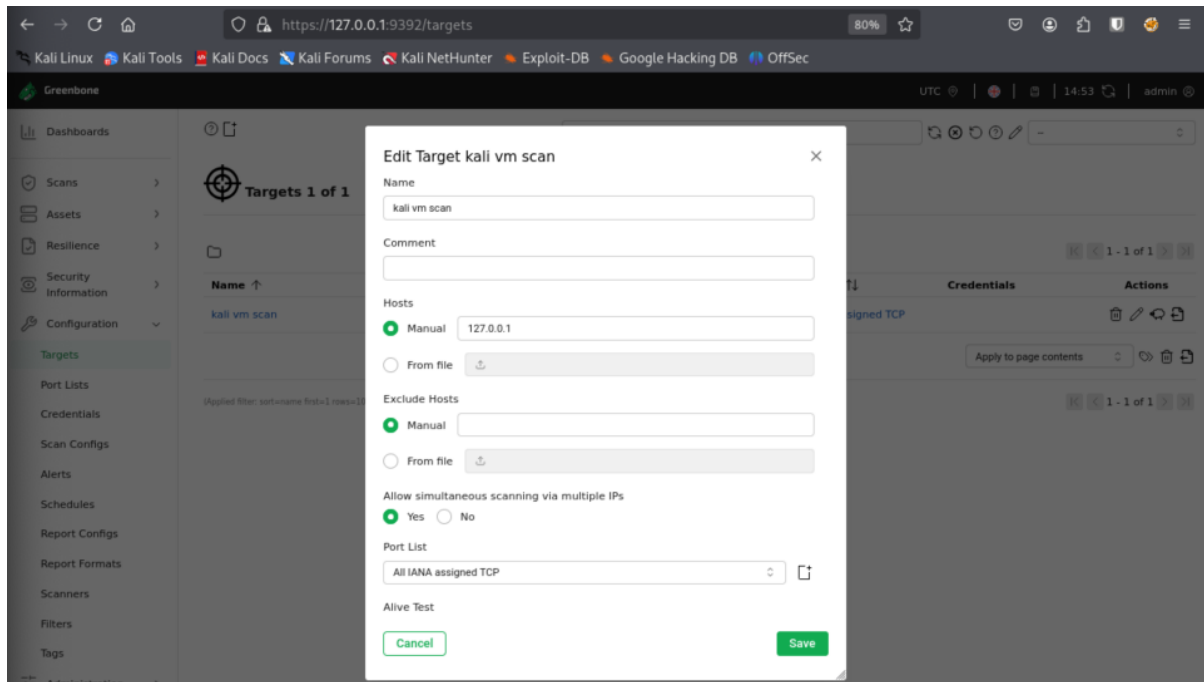


fig3

2. Leave everything as default since we are doing a basic scan in this task and click save.
3. Then after you have created your target go to scan in the left menu and then tasks in the sub section of it, click create new task.
4. Then name the task and choose the target you just created and leave everything else as default and click save just like in fig 4

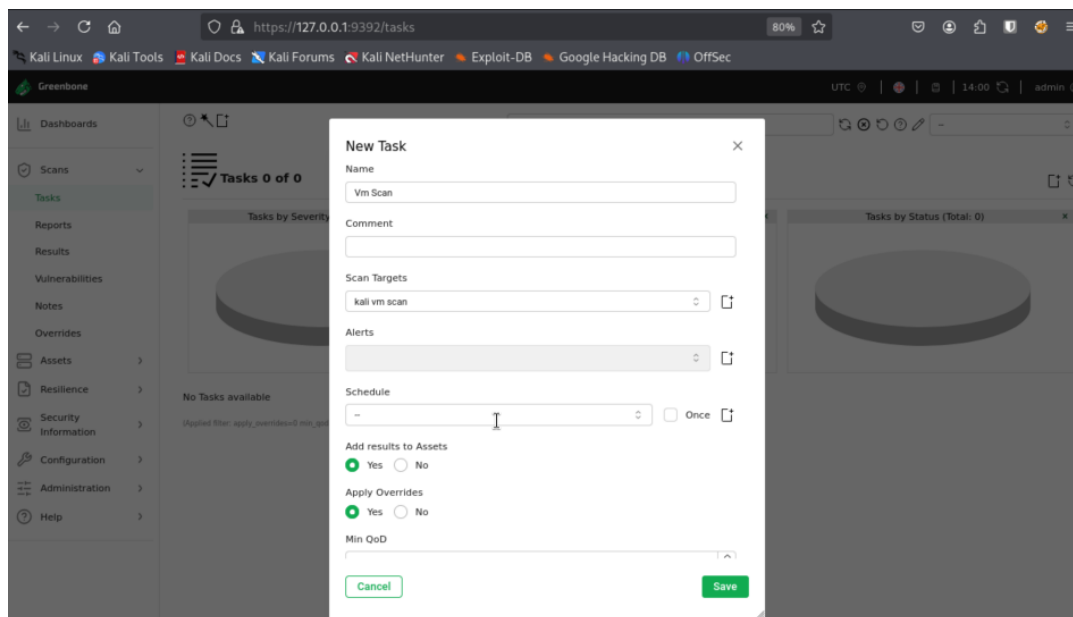


fig 4

5. After this is done start the scan using the play button that's mostly to the right side of where your task is listed, wait for some time (this could take a long time depending on what machine you scan and what type of scan you run).
6. The final output after the scan is complete should look something like fig 5.

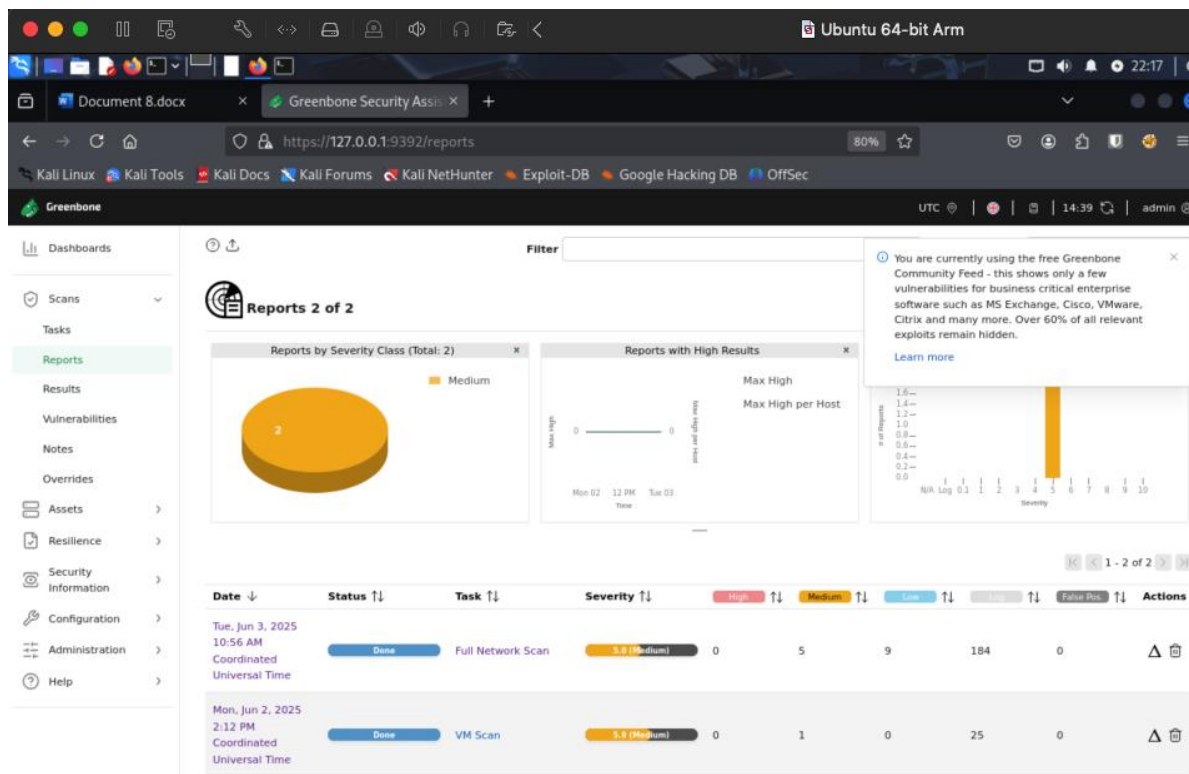


Fig5

7. Then comes the most important part of this task the vulnerability assessment which we will do next.

Vulnerability Assessment process:

1. After the scan is complete click on the done option under the status section to view the report which shows us what scan were run and what vulnerabilities our system has.
2. It should look something like fig6

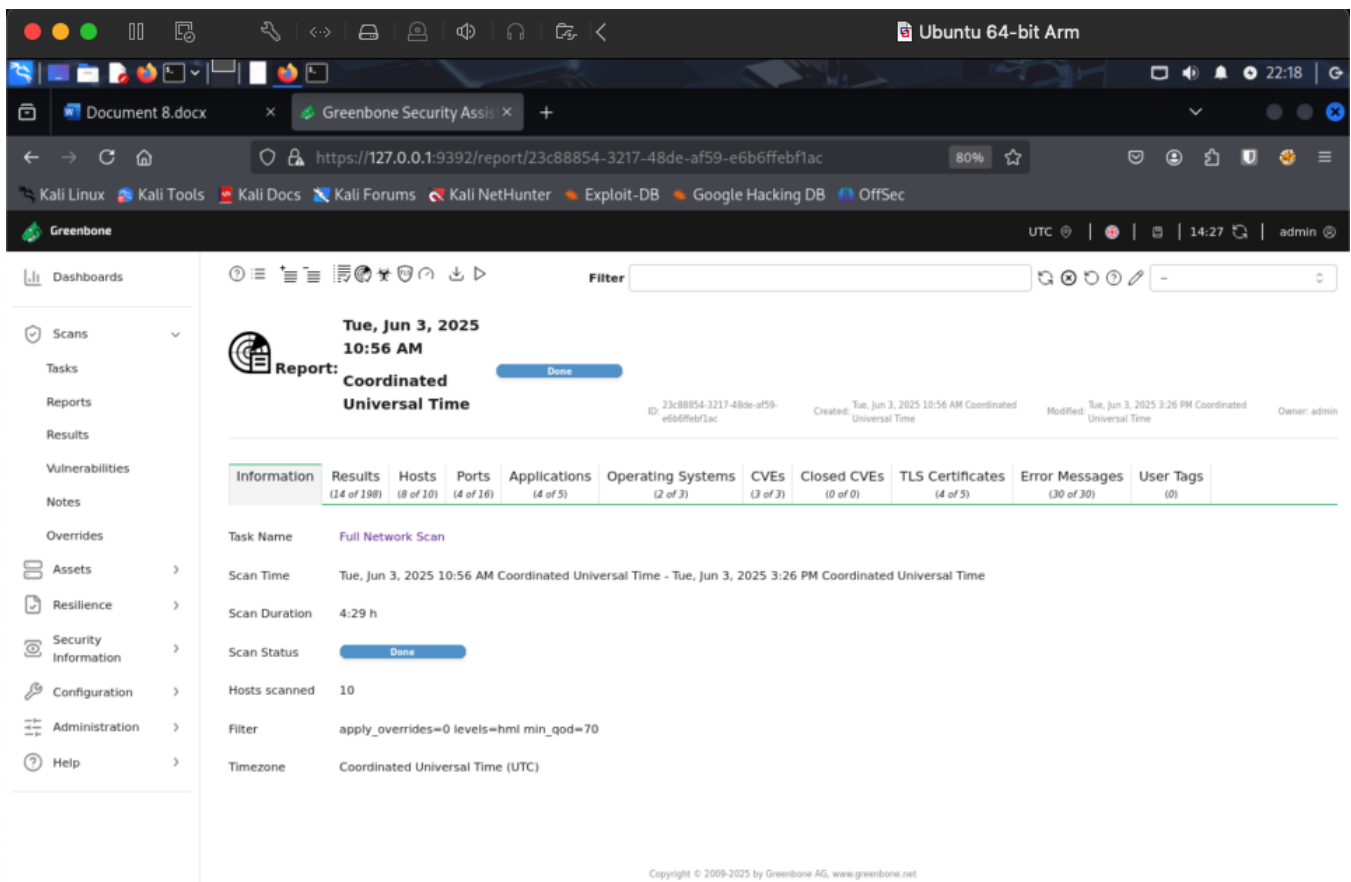


Fig6

3. You can download a simple and easy to understand part of the report in a xml or pdf format the pdf report of my vulnerability assesment report is attached in the github repository for you to refer and see what a report looks like.
4. Now let's analyze the key vulnerabilities on my network:
 - The first most vulnerability that catches my eyes is for the host (192.168.31.1) which is my wifi router which is vulnerable to SSL/TLS renegotiation Dos Vulnerability on ports 8443, 443, 74443, 5068 The CVE given for them is: (CVE-2011-1473, CVE-2011-5094)

- The mitigation to it (process to fix) is: Disabling SSL renegotiation on 192.168.31.1 which will prevent my host from someone trying to overwhelm my host with multiple requests.
- This Page gives the entire summary and is very important for you to learn how to navigate to it; when on a report click on the specific vulnerability you want to have details for and then scroll down until you find the details section like in fig7

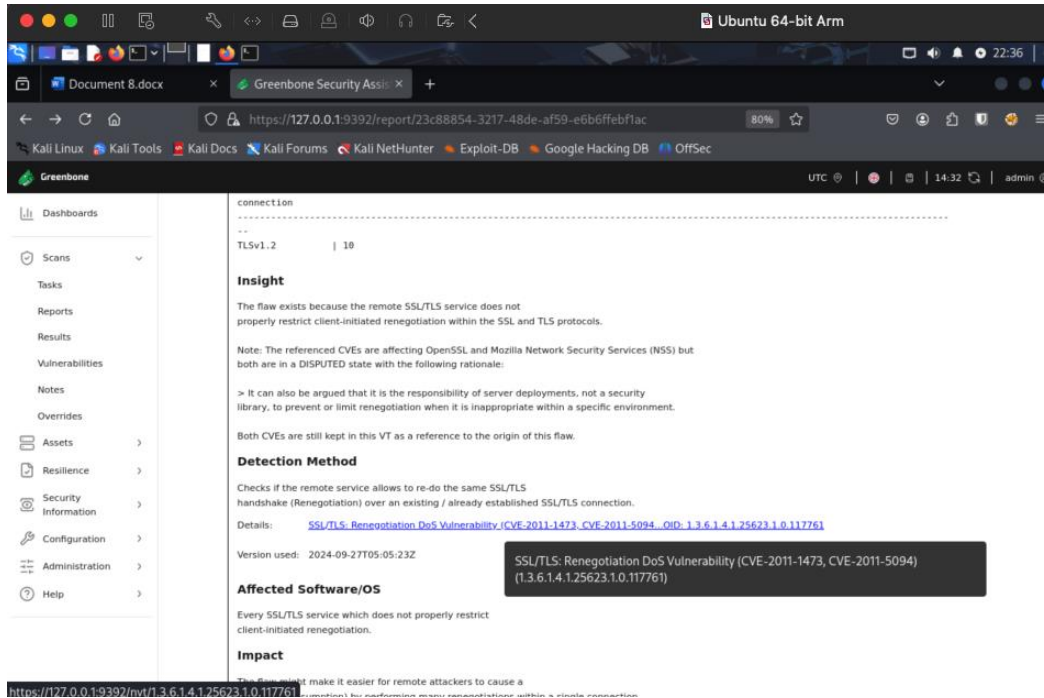


Fig7

- Now after you click on the details tab you will see the full details about the vulnerability and the CVE which was detected for it like in fig 8

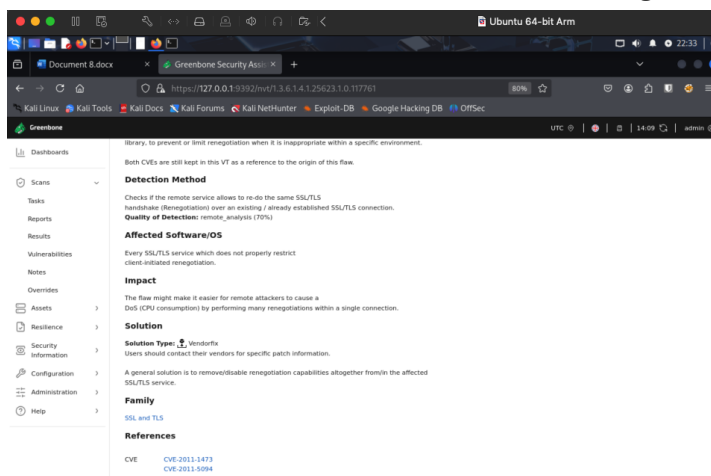
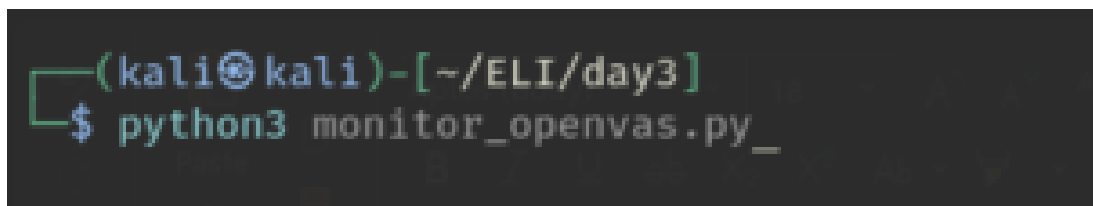


Fig 8

- Like this you can view any vulnerability and its impact on your network which is really helpful for security assessment purposes and use case scenarios.

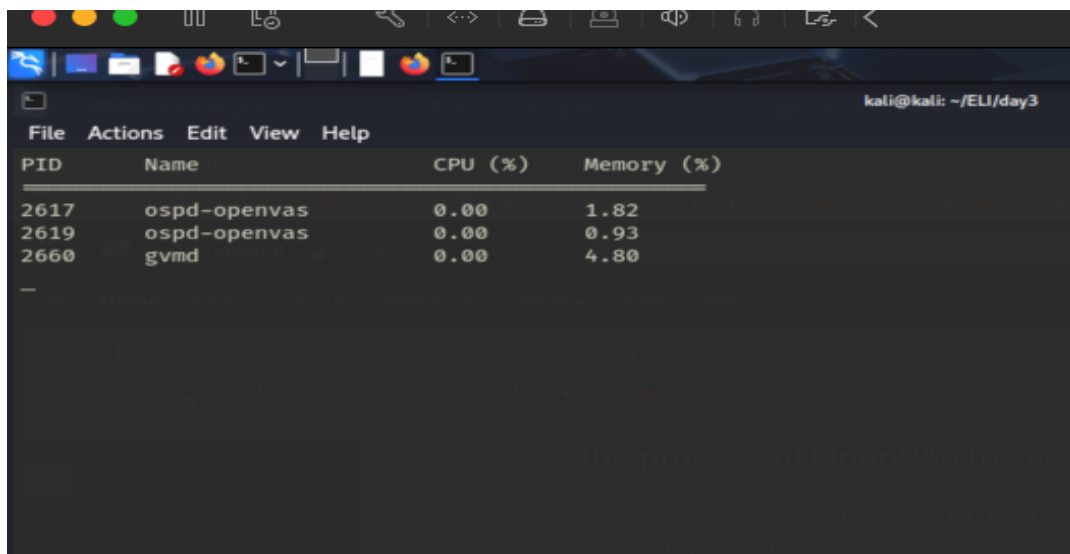
Also I created a python script that automatically helps monitor when your OpenVAS Scan is running to identify whether OpenVAS is running or not or maybe its stuck. The tool is provided in the github repository which you can install and the method to run it very simple just run “python3 Script_name.py” and it will keep cheching for the process of OpenVAS to see whether its running or not



```
(kali㉿kali)-[~/ELI/day3]
$ python3 monitor_openvas.py_
```

Fig9

- After you run it, it will look something like this:



PID	Name	CPU (%)	Memory (%)
2617	ospd-openvas	0.00	1.82
2619	ospd-openvas	0.00	0.93
2660	gvmd	0.00	4.80
—			

Fig 10