Certified Ethical Hacking With Penetration Testing

CEHWPT

LABS Course

LAB5 Installing Openvas 9 on Kali Linux

Prepared by Eng. Khaled Gamo

15-6-2019

LAB5 Installing Openvas 9 on Kali Linux

Step1: To install Openvas 9 and its dependencies on our Kali Linux system we simply have to run the following command:

apt-get update && apt-get install openvas

```
root@kali:~# apt-get update && apt-get install openvas
Get:1 http://kali.download/kali kali-rolling InRelease [30.5 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [17.1 MB]
Fetched 17.1 MB in 17s (1,006 kB/s)
```

Step2: The next step to run the setup procedure that will setup OpenVAS and download a large number of Network Vulnerability Tests (NVTs) or signatures. Due to the large number of NVTs (50.000+) the setup procedure might take a while to complete and consume a considerable amount of data. On the test setup we've used for this tutorial the total setup procedure took 10 minutes to complete which is not bad at all.

Run the following command to start the setup process:

openvas-setup

```
root@kali:~# openvas-setup

[>] Updating OpenVAS feeds
[*] [1/3] Updating: NVT
--2019-06-11 15:16:05--  http://dl.greenbone.net/community-nvt-feed-current.tar.
bz2
Resolving dl.greenbone.net (dl.greenbone.net)... 89.146.224.58, 2a01:130:2000:12
7::d1
Connecting to dl.greenbone.net (dl.greenbone.net)|89.146.224.58|:80... connected
.
HTTP request sent, awaiting response... 200 OK
Length: 21940407 (21M) [application/octet-stream]
Saving to: '/tmp/greenbone-nvt-sync.3sEfXmodSX/openvas-feed-2019-06-11-6378.tar.
bz2'

enbone-nvt-sync.3sE  48%[=========>           ]  10.15M  3.08MB/s    eta 5s
```

When the setup process is finished, all required OpenVAS processes are started and the web interface will be opened automatically. The web interface is running locally on port 9392 and can accessed through: https://localhost:9392. OpenVAS will also setup an admin account and automatically generate a password for this account which is displayed in the last section of the setup output:

```
[*] Opening Web UI (https://127.0.0.1:9392) in: 5... 4... 3... 2... 1...

[>] Checking for admin user
[*] Creating admin user
User created with password '04a5df77-30f2-4a01-bc25-b3585c38db86'.

[+] Done
```

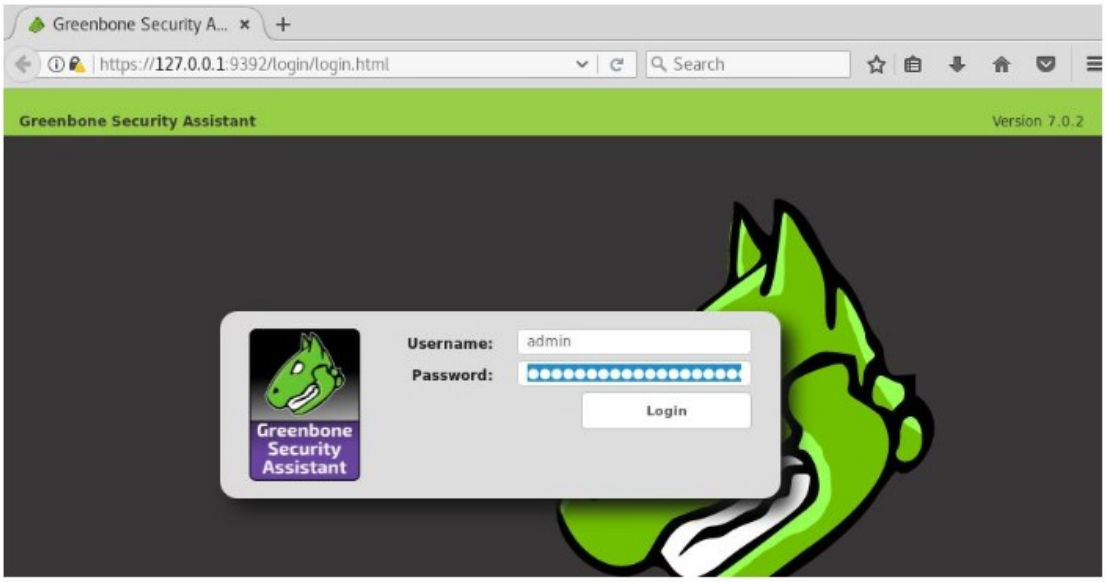
Step3: reset the password

Password reset

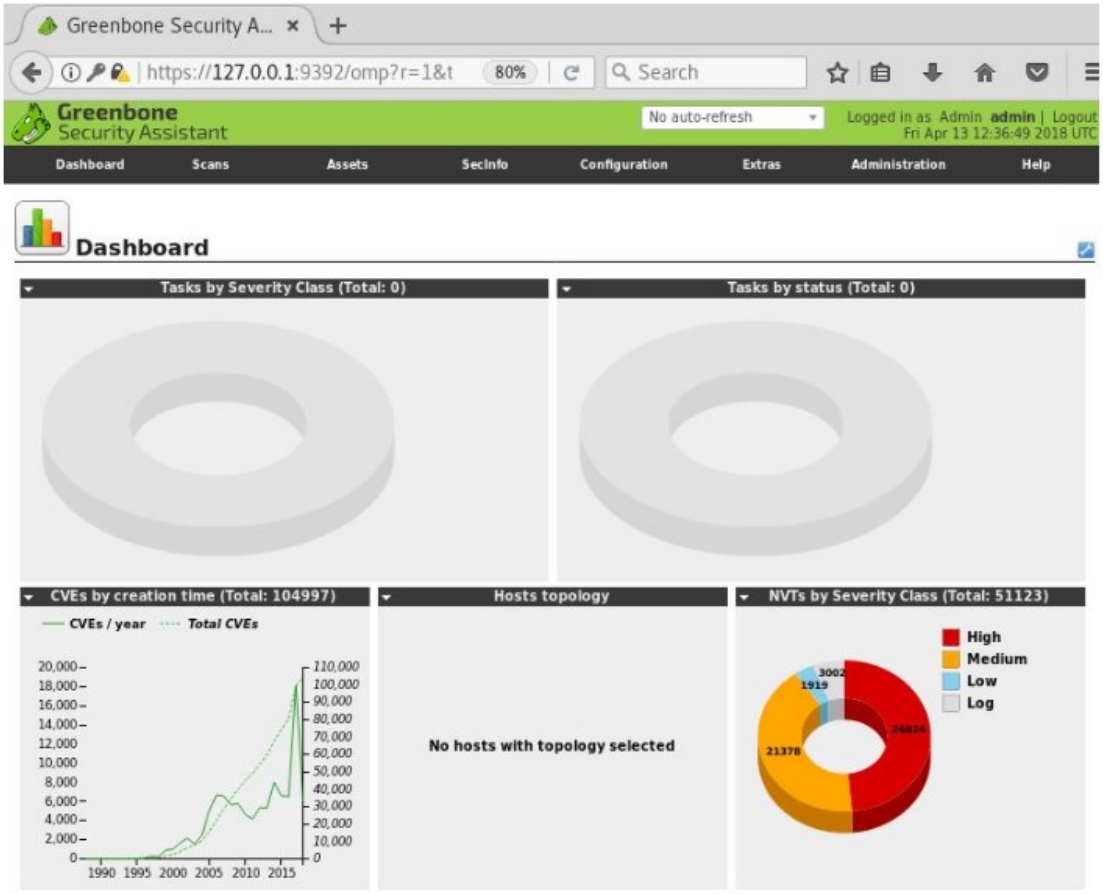
Did you forgot to note down the password? You can change the admin password using the following commands:

openvasmd –user=[username]–new-password=[password]
openvasmd –user=admin –new-password=[password]

Step4: The next step is to accept the self-signed certificate warning and use the automatically generated admin credentials to login on the web interface:



Step5: After logging in on the web interface we're redirected to the Greenbone Security Assistant dashboard. From this point on we can start to configure and run vulnerability scans.



Step 6: Starting and stopping OpenVAS

The last step I want to point out before we head on with the installation of the virtual appliance is how to start and stop OpenVAS services. OpenVAS services may consume a lot of unnecessary resources and therefore it is advised to terminate these services when you're not using OpenVAS.

Run the following command to stop the services:

```
openvas-stop
```

To start the OpenVAS services again, run:

```
openvas-start
```

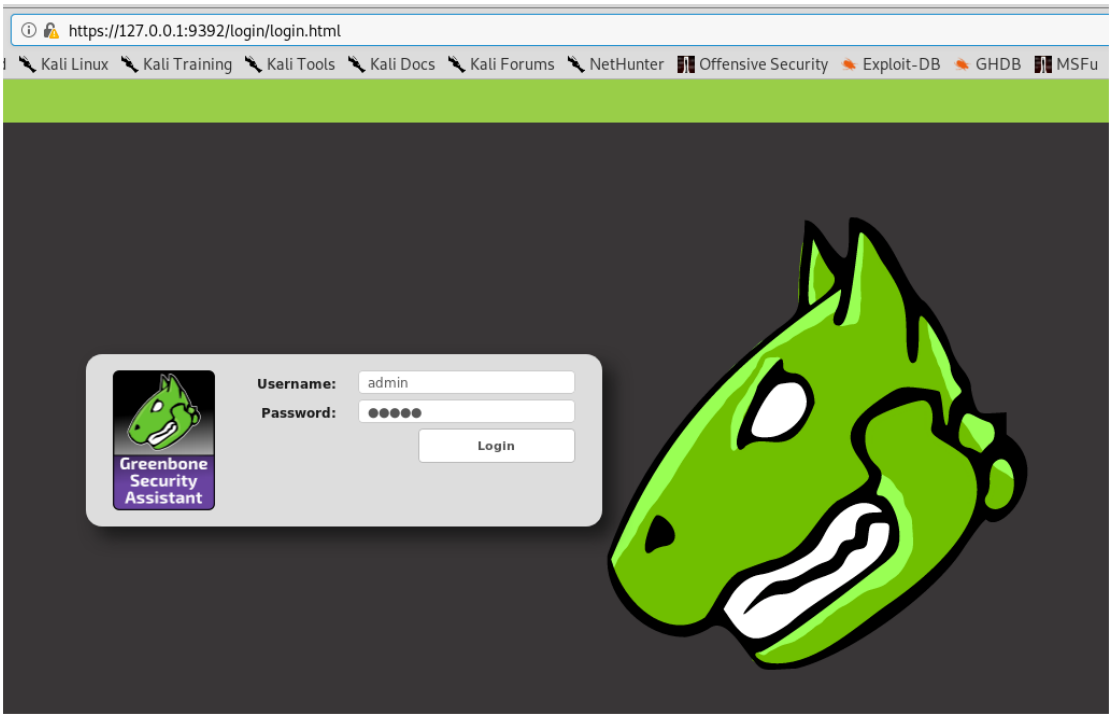
Discovering Vulnerabilities Using OpenVAS

Step1: starting OpenVAS using openvas-start command

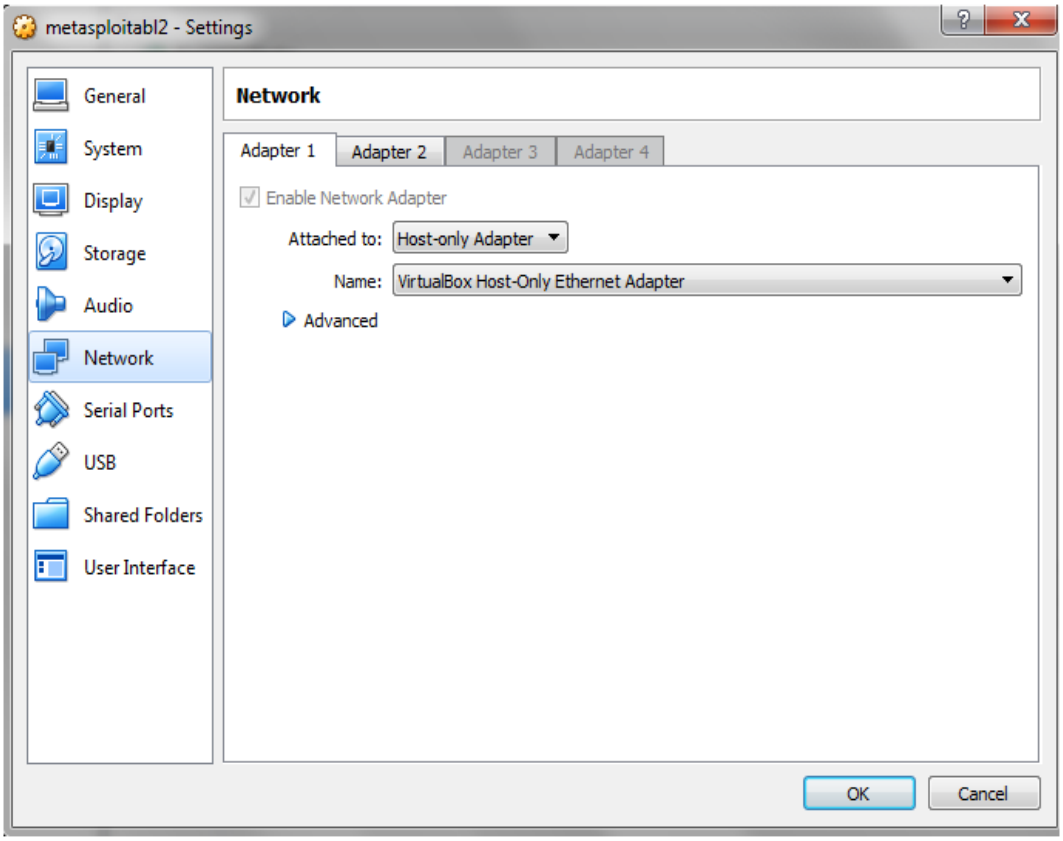
```
root@kali:~# openvas-start
[*] Please wait for the OpenVAS services to start.
[*]
[*] You might need to refresh your browser once it opens.
[*]
[*]  Web UI (Greenbone Security Assistant): https://127.0.0.1:9392

● greenbone-security-assistant.service - Greenbone Security Assistant
   Loaded: loaded (/lib/systemd/system/greenbone-security-assistant.service; dis
abled; vendor preset: disabled)
   Active: active (running) since Tue 2019-06-11 21:16:52 EET; 1min 6s ago
     Docs: man:gsad(8)
           http://www.openvas.org/
 Main PID: 1375 (gsad)
    Tasks: 8 (limit: 2341)
   Memory: 45.3M
   CGroup: /system.slice/greenbone-security-assistant.service
           ├─1375 /usr/sbin/gsad --foreground --listen=127.0.0.1 --port=9392 --m
listen=127.0.0.1 --mport=9390
           └─1378 /usr/sbin/gsad --foreground --listen=127.0.0.1 --port=9392 --m
listen=127.0.0.1 --mport=9390

Jun 11 21:16:52 kali systemd[1]: Started Greenbone Security Assistant.
```

Step2: login to OpenVAS GUI browsing http://127.0.0.1:9392

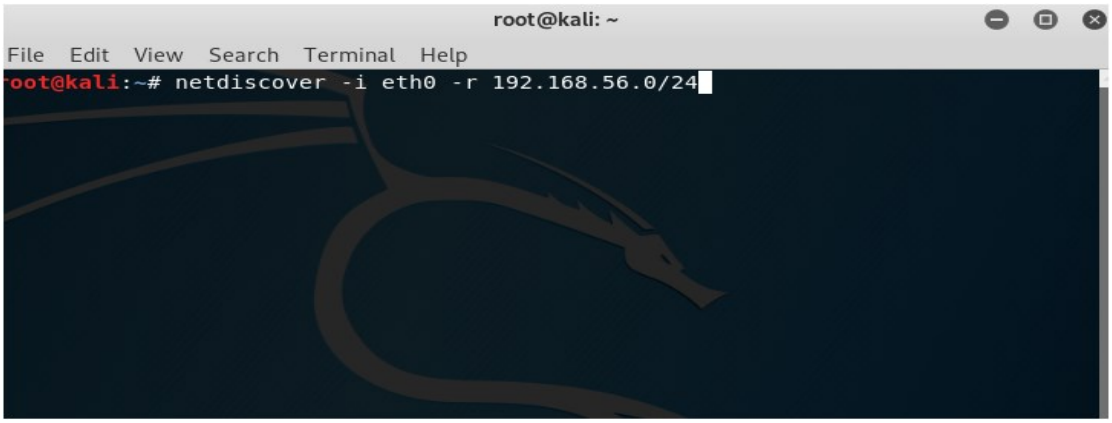
And login by username admin and your configured password

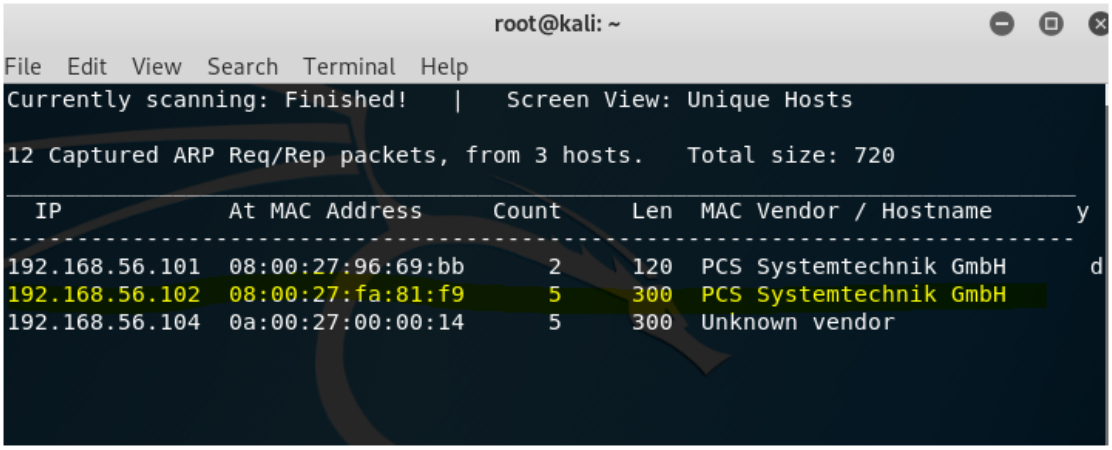


Step3: startup the metasploitable2 machine in Virtual box and make sure that the network setting of Kali Linux and metasplitable2 are configured as Hostonly adapter



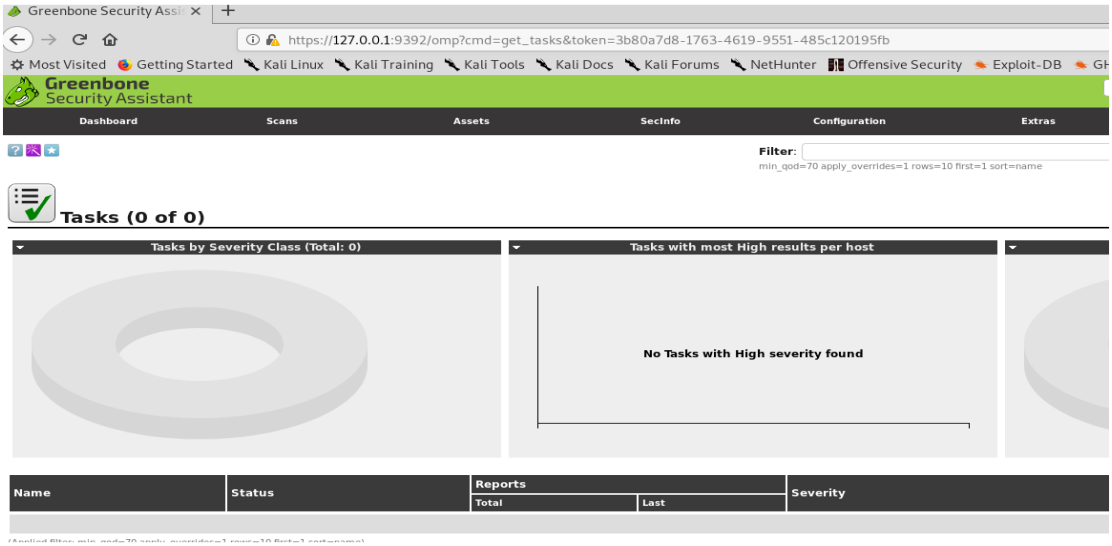
Step4: by using command netdiscover –I eth0 –r 192.168.56.0/24 we can discover the IP address of our target host metasploitable2



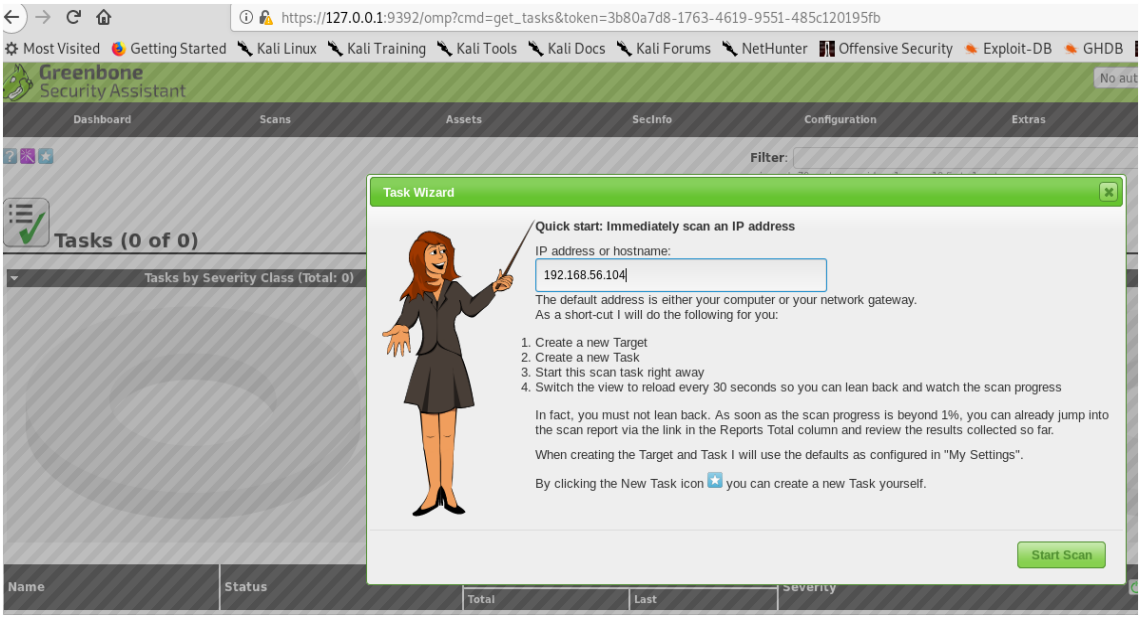


So our target IP address is 192.168.56.102

Step5: go to scan menu and choose scan

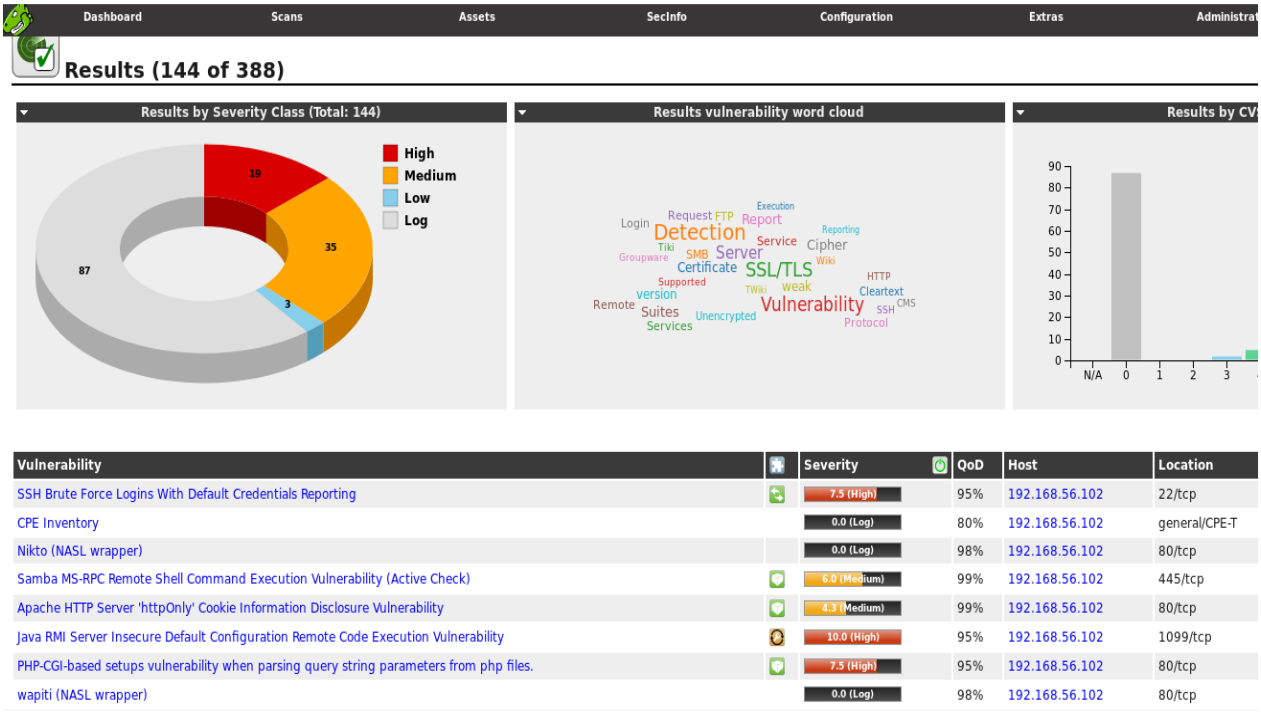


Step6: choose task wizard and put the IP address of your target



Step7: start the scan

Step8: wait until the scan finished and find the results



Vulnerability		Severity	QoD	Host	Location
SSH Brute Force Logins With Default Credentials Reporting		7.5 (High)	95%	192.168.56.102	22/tcp
CPE Inventory		0.0 (Log)	80%	192.168.56.102	general/CPE-T
Nikto (NASL wrapper)		0.0 (Log)	98%	192.168.56.102	80/tcp
Samba MS-RPC Remote Shell Command Execution Vulnerability (Active Check)		6.0 (Medium)	99%	192.168.56.102	445/tcp
Apache HTTP Server 'httpOnly' Cookie Information Disclosure Vulnerability		4.3 (Medium)	99%	192.168.56.102	80/tcp
Java RMI Server Insecure Default Configuration Remote Code Execution Vulnerability		10.0 (High)	95%	192.168.56.102	1099/tcp
PHP-CGI-based setups vulnerability when parsing query string parameters from php files.		7.5 (High)	95%	192.168.56.102	80/tcp
wapiti (NASL wrapper)		0.0 (Log)	98%	192.168.56.102	80/tcp

Step8: explore the vulnerabilities and find the interesting vulnerabilities.

Vulnerability		Severity	QoD
rexec Passwordless / Unencrypted Cleartext Login		10.0 (High)	80%
OS End Of Life Detection		10.0 (High)	80%
TWiki XSS and Command Execution Vulnerabilities		10.0 (High)	80%
Distributed Ruby (dRuby/DRb) Multiple Remote Code Execution Vulnerabilities		10.0 (High)	99%
Possible Backdoor: Ingreslock		10.0 (High)	99%
Java RMI Server Insecure Default Configuration Remote Code Execution Vulnerability		10.0 (High)	95%
DistCC Remote Code Execution Vulnerability		9.3 (High)	99%
VNC Brute Force Login		9.0 (High)	95%
PostgreSQL weak password		9.0 (High)	99%
MySQL / MariaDB weak password		9.0 (High)	95%