OpenVAS

OpenVAS is a full-featured vulnerability scanner. Its capabilities include unauthenticated and authenticated testing, various high-level and low-level internet and industrial protocols, performance tuning for large-scale scans, and a powerful internal programming language for implementing any type of vulnerability test.

The scanner obtains the tests for detecting vulnerabilities from a feed with a long history and daily updates.

Installation:

Step 1: OpenVas can be installed using following command whereas for purpose of this guide Ubuntu 20.04 OS is used:

- sudo apt-get -y install openvas

```
File  Actions  Edit  View  Help
$ sudo apt-get install openvas
[sudo] password for seed:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following packages were automatically installed and are no longer required:
  libabsl20210324 libarmadillo10 libavfilter7 libavformat58 libcharls2 libdrm-intel1 libev4 libgdal30
  libgeos3.10.2 libgrpc++1 libgrpc10 liblttng-ust-ctl4 liblttng-ust0 libpostproc55
  libpython3.9-minimal libpython3.9-stdlib libsrt1.4-gnutls libswscale5 libwebsockets16
  python3-iptools python3-toml python3.9 python3.9-minimal
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  doc-base dvisvgm greenbone-security-assistant gsad gvm gvm-tools gvmd gvmd-common
  libapache-pom-java libbit-vector-perl libcarp-clan-perl libcommons-logging-java
  libcommons-parent-java libcrypt-rc4-perl libdate-calc-perl libdate-calc-xs-perl
  libdigest-perl-md5-perl libfontbox-java libgvm21 libhiredis0.14 libjcode-pm-perl libmicrohttpd12
  libole-storage-lite-perl libparse-recdescent-perl libpdfbox-java libptexenc1 libradcli4
  libspreadsheet-parseexcel-perl libspreadsheet-writeexcel-perl libteckit0 libtexlua53-5
  libtexluajit2 libunicode-map-perl libuuid-perl libyaml-tiny-perl libzzip-0-13 lmodern
  openvas-scanner ospd-openvas preview-latex-style python3-deprecated python3-gvm python3-psutil
  python3-wrapt t1utils tcl tex-common tex-gyre texlive-base texlive-binaries
  texlive-fonts-recommended texlive-latex-base texlive-latex-extra texlive-latex-recommended
  texlive-pictures texlive-plain-generic tipa tk
Suggested packages:
  dhelp | dwww | dochelp | doc-central | yelp | khelpcenter libavalon-framework-java
  libcommons-logging-java-doc libexcalibur-logkit-java liblog4j1.2-java pnscan strobe python-gvm-doc
  python-psutil-doc debhelper ghostscript perl-tk xzdec texlive-fonts-recommended-doc
```

Step 2: After that following command can be used to configure OpenVAS, download the most recent standards, create an administrator client, and start the various services. This could take some time depending on your data transfer capacity and host assets.

- Sudo gvm-setup

```
File  Actions  Edit  View  Help
$ 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
[>] Migrating database
[>] Checking for GVM admin user
[*] Creating user admin for gvm
[*] Please note the generated admin password
```

```
[*] Creating extension pgcrypto
CREATE EXTENSION
[>] Migrating database
[>] Checking for GVM admin user
[*] Creating user admin for gvm
[*] Please note the generated admin password
[*] User created with password 'b7cc819b-b044-4ad9-9fc0-288159294235'.
[*] Define Feed Import Owner
[>] Updating GVM feeds
[*] Updating NVT (Network Vulnerability Tests feed from Greenbone Security Feed/Community Feed)
Greenbone community feed server - http://feed.community.greenbone.net/
This service is hosted by Greenbone Networks - http://www.greenbone.net/

All transactions are logged.

If you have any questions, please use the Greenbone community portal.
See https://community.greenbone.net for details.

By using this service you agree to our terms and conditions.

Only one sync per time, otherwise the source ip will be temporarily blocked.

receiving incremental file list
./
```

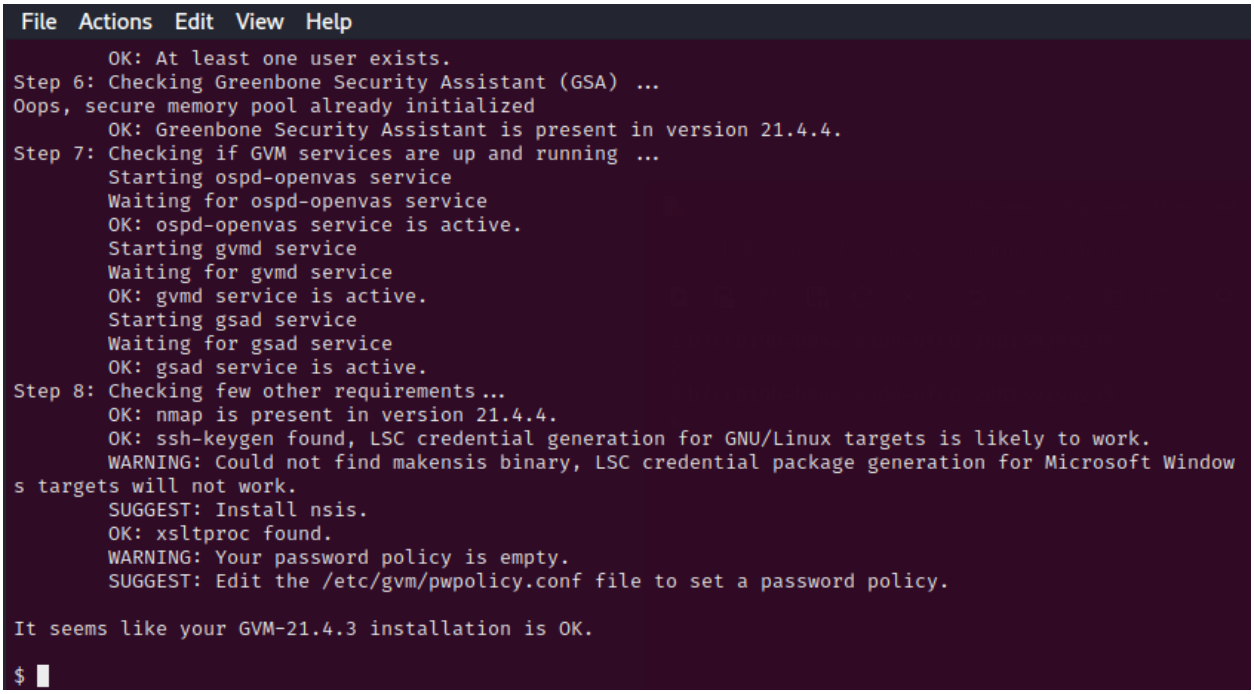
```
File  Actions  Edit  View  Help
dfn-cert-2020.xml
      3,661,718 100%  888.20kB/s    0:00:04 (xfr#24, to-chk=5/30)
dfn-cert-2021.xml
      3,615,074 100%  789.26kB/s    0:00:04 (xfr#25, to-chk=4/30)
dfn-cert-2022.xml
      2,637,898 100%  881.31kB/s    0:00:02 (xfr#26, to-chk=3/30)
sha256sums
          2,180 100%    2.37kB/s    0:00:00 (xfr#27, to-chk=2/30)
sha256sums.asc
            833 100%    0.87kB/s    0:00:00 (xfr#28, to-chk=1/30)
timestamp
             13 100%    0.01kB/s    0:00:00 (xfr#29, to-chk=0/30)

sent 658 bytes  received 85,621,886 bytes  978,543.36 bytes/sec
total size is 85,599,200  speedup is 1.00

[+] GVM feeds updated
[*] Checking Default scanner
[*] Modifying Default Scanner
Scanner modified.

[+] Done
[*] Please note the password for the admin user
[*] User created with password 'b7cc819b-b044-4ad9-9fc0-288159294235'.

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

As seen from above screenshot, it can be observed that once the setup is done following information will be provided to user whereas it is important to note down the password to access the service.

Step 3: Next step is to run following command to check readiness of gvm package:

- sudo gvm-check-setup

```
File  Actions  Edit  View  Help
$ sudo gvm-check-setup
[sudo] password for seed:
gvm-check-setup 21.4.3
  Test completeness and readiness of GVM-21.4.3
Step 1: Checking OpenVAS (Scanner)...
        OK: OpenVAS Scanner is present in version 21.4.4.
        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: redis-server is running and listening on socket: /var/run/redis-openvas/redis-server.sock.
        OK: redis-server configuration is OK and redis-server is running.
        OK: _gvm owns all files in /var/lib/openvas/plugins
        OK: NVT collection in /var/lib/openvas/plugins contains 102868 NVTs.
Checking that the obsolete redis database has been removed
        OK: No old Redis DB
        OK: ospd-OpenVAS is present in version 21.4.4.
Step 2: Checking GVMD Manager ...
        OK: GVM Manager (gvmd) is present in version 21.4.5.
Step 3: Checking Certificates ...
        OK: GVM client certificate is valid and present as /var/lib/gvm/CA/clientcert.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.
```

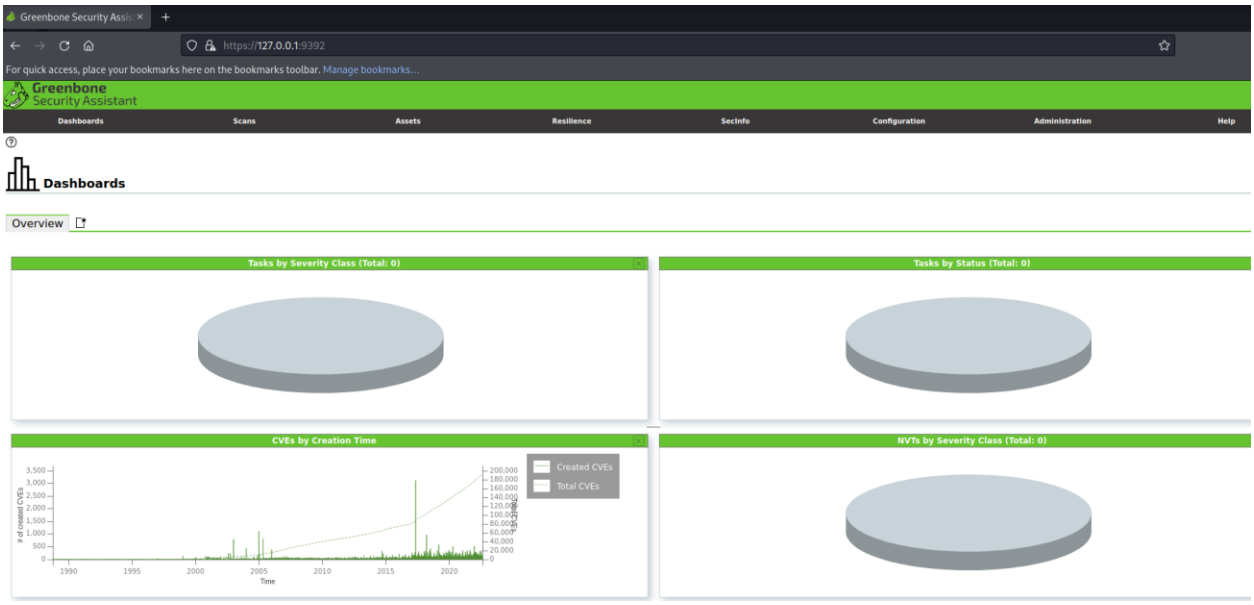


Step 4: Next step is to start the gvm service using following command:

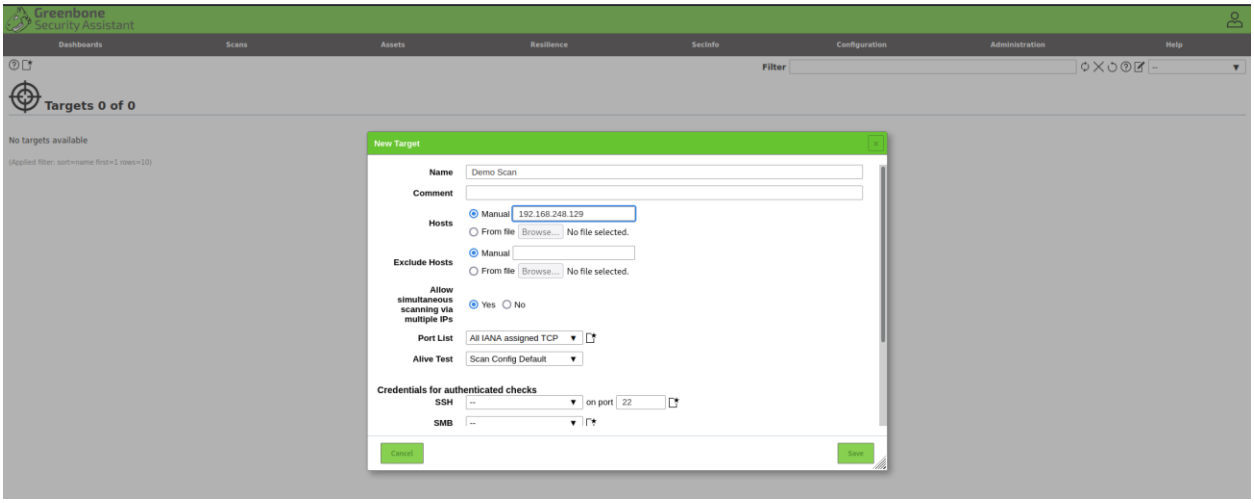
- sudo gvm-start

Step 5: After verifying the service is running, the web UI of Openvas can be accessed using below URL and with username as “admin” and generated password in previous step:

- https://127.0.0.1:9392



Step 6: After logging in, user can navigate to scan's tab and create a "New Target" to run a scan on target domain/host as shown in below screenshot:



Like shown above, multiple scans can be run against the target environment in ethical penetration test to identify and match vulnerabilities along with possible CVEs to which the application is vulnerable. This information can also be exported through the portal which can be used for further getting access into the network.