



[kali Tool](#)

- [nmap](#) [\[1\]](#) [\[2\]](#) [\[3\]](#)
- [OPENVAS\(GVM\)](#) [\[1\]](#) [\[2\]](#)
- [\[1\]](#) [\[2\]](#) [\[3\]](#) [\[4\]](#)
- [\[1\]](#) [\[2\]](#) [\[3\]](#) [\[4\]](#) [\[5\]](#)

| | | | | | |
|--|--|--|--|--|--|
| | | | | | |
|--|--|--|--|--|--|

[illegible]

```
nmap -A -Pn {IP}
```

| | | | |
|--|--|--|--|
| | | | |
|--|--|--|--|

| | | | | | |
|--|--|--|--|--|--|
| | | | | | |
|--|--|--|--|--|--|

```
nmap -iL hostlist.txt --excludefile excludelist.txt
```

■■■■■■■■

```
nmap -sA scanme.nmap.org
```

■■■■■■■■

```
nmap -PN scanme.nmap.org
```

■■■■■■■(ping scan)

```
nmap -sP 140.115.35.0/24
```

```
nmap -F www.hinet.net
```

```
■■■■ ■■■■-T5■■-T3
```

■■port■■

```
nmap -p 80,443 192.168.1.1
```

```
nmap -p 80-200 192.168.1.1
```

■■tcp■■

```
nmap -p T:80 192.168.1.1
```

■■udp■■

```
nmap -p U:53 192.168.1.1
```

■■■■10port

```
nmap --top-ports 10 192.168.1.1
```

■■■

```
nmap -sS -P0 -sV -O 192.168.56.102
```

param:

-sS: Scan Syn

-sV: Determine OS Version

-P0: Protocol Scan

-O : Operation System

script scan

SCRIPT SCAN:

-sC: equivalent to --script=default

--script=<Lua scripts>: <Lua scripts> is a comma separated list of directories, script-files or script-categories

--script-args=<n1=v1,[n2=v2,...]>: provide arguments to scripts

--script-args-file=filename: provide NSE script args in a file

--script-trace: Show all data sent and received

--script-updatedb: Update the script database.

--script-help=<Lua scripts>: Show help about scripts.

<Lua scripts> is a comma-separated list of script-files or script-categories.

#####TLS

```
nmap --script ssl-enum-ciphers -p 443 {url}
```

#####txt#####10.0.0.0/16tls

```
echo 10.0.{1..254}.{1..254} >> iplist.txt
```

#####sh#####tls1.0-1.#####

```
#!/bin/bash
```

```
IPLIST=$(cat iplist.txt)
```

```
TLS_VER=(TLSv1.0 TLSv1.1 TLSv1.2)
```

```
SCAN_PORT="443"
```

```
NMAP_OPS="--host-timeout 3000ms --max-rtt-timeout 3000ms --script ssl-enum-ciphers -p ${SCAN_PORT}"
```

```
OUTPUT="output.csv"
```

```
for (( i = 0; i < ${#IPLIST[@]}; i++ )); do
```

```
    SUP_TLS=$(nmap $NMAP_OPS ${IPLIST[i]} | egrep "${TLS_VER[0]}${TLS_VER[1]}${TLS_VER[2]}" | cut -c 5-11)
```

```
    if [[ ! -z $SUP_TLS ]]; then
```

```
        echo -ne "\n${IPLIST[i]}" >> $OUTPUT
```

```
        for (( j = 0; j < ${#SUP_TLS[@]}; j++ )); do
```

```
            echo -n ",${SUP_TLS[@]}" >> $OUTPUT
```

done

fi

unset SUP_TLS

done



[nmap](#) [TLS](#) [Ciphers](#) [Mr.](#)

OPENVAS(GVM)



openvas

kali

```
sudo apt install gvm
```

linux

<https://greenbone.github.io/docs/latest/22.4/source-build/index.html>

debian

<https://github.com/Kastervo/OpenVAS-Installation>

gvm

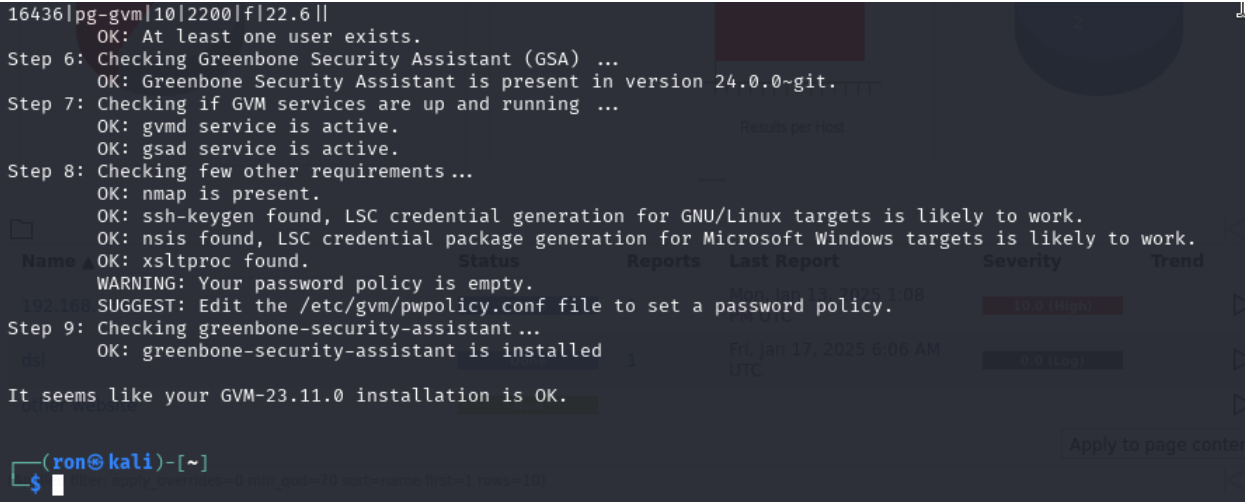
```
sudo gvm-setup
```



```
sudo runuser -u _gvm -- gvmc --user=admin --new-password=password
```



```
sudo gvm-check-setup
```



It seems like your GVM-xx.xx.xx installation is OK. 

gvm

```
sudo gvm-start
```

gsad(GreenBone Security Assistant daemon)gvmd(greenbone Vulnerability M
daemon)opasd-openvas

gvm

https://127.0.0.1:9392

gvm

```
sudo gvm-stop
```

IPPort

```
sudo vi /usr/lib/systemd/system/gsad.service
```

```
[Unit]
Description=Greenbone Security Assistant daemon (gsad)
Documentation=man:gsad(8) https://www.greenbone.net
After=network.target gvmd.service
Wants=gvmd.service

[Service]
Type=exec
User=_gvm
Group=_gvm
RuntimeDirectory=gsad
RuntimeDirectoryMode=2775
PIDFile=/run/gsad/gsad.pid
ExecStart=/usr/sbin/gsad --foreground --listen 0.0.0.0 --port 9392
Restart=always
TimeoutStopSec=10

[Install]
WantedBy=multi-user.target
Alias=greenbone-security-assistant.service
~
~
```

listen 0.0.0.0 -Port port

```
sudo greenbone-feed-sync
```

```
sudo greenbone-feed-sync --type nvt
```

```
sudo greenbone-feed-sync --type scap
```

```
sudo greenbone-feed-sync --type cert
```

crontab -e

gvm

```
sudo -u _gvm gvmc --rebuild-gvmd-data=all
```

gvm

openvas --ip 192.168.1.100 --ip 192.168.1.101

Alive Test **Scan config Default** **Consider Alive**

| | | |
|--|---|--|
| Name | Wlan | |
| Comment | | |
| Hosts | <input checked="" type="radio"/> Manual | |
| Exclude Hosts | | |
| Allow simultaneous scanning via multiple IPs | | |
| Port List | | |
| Alive Test | Consider Alive ▲ | |

Scan Config Default
ICMP Ping
TCP-ACK Service Ping
TCP-SYN Service Ping
ARP Ping
ICMP & TCP-ACK Service Ping
ICMP & ARP Ping
TCP-ACK Service & ARP Ping
ICMP, TCP-ACK Service & ARP Ping
Consider Alive

(Database is wrong Version)

gvm


```

(ron@cc140-87)-[~]
$ sudo systemctl status gvmd.service
● gvmd.service - Greenbone Vulnerability Manager daemon (gvmd)
   Loaded: loaded (/usr/lib/systemd/system/gvmd.service; disabled; preset: disabled)
   Active: activating (start) since Wed 2025-04-09 13:25:28 CST; 38s ago
     Job: 1660
 Invocation: 80b19be4686a4a329bc4609347cb1fd9
    Docs: man:gvmd(8)
 Process: 2595 ExecStart=/usr/sbin/gvmd --osp-vt-update=/run/ospd/ospd.sock --listen-group=_gvm (code=exited, status=0/SUCCESS)
   Tasks: 0 (limit: 19043)
  Memory: 4K (peak: 5.8M)
    CPU: 17ms
   CGroup: /system.slice/gvmd.service

Apr 09 13:25:28 cc140-87 systemd[1]: gvmd.service: Scheduled restart job, restart counter is at 1.
Apr 09 13:25:28 cc140-87 systemd[1]: Starting gvmd.service - Greenbone Vulnerability Manager daemon (gvmd)...
Apr 09 13:25:28 cc140-87 gvmd[2595]: md main:MESSAGE:2025-04-09 05h25.28 utc:2595: Greenbone Vulnerability Manager version 25.1.3 (DB revision 258)
Apr 09 13:25:28 cc140-87 systemd[1]: gvmd.service: Can't open PID file '/run/gvmd/gvmd.pid' (yet?) after start: No such file or directory
Apr 09 13:25:28 cc140-87 gvmd[2597]: md manage:MESSAGE:2025-04-09 05h25.28 utc:2597: check_db_versions: database version of database: 256
Apr 09 13:25:28 cc140-87 gvmd[2597]: md manage:MESSAGE:2025-04-09 05h25.28 utc:2597: check_db_versions: database version supported by manager: 258
Apr 09 13:25:28 cc140-87 gvmd[2597]: md main:CRITICAL:2025-04-09 05h25.28 utc:2597: gvmd: database is wrong version
Apr 09 13:25:28 cc140-87 gvmd[2597]: md main:CRITICAL:2025-04-09 05h25.28 utc:2597: gvmd: Your database is too old for this version of gvmd.
Apr 09 13:25:28 cc140-87 gvmd[2597]: md main:CRITICAL:2025-04-09 05h25.28 utc:2597: gvmd: Please migrate to the current data model.
Apr 09 13:25:28 cc140-87 gvmd[2597]: md main:CRITICAL:2025-04-09 05h25.28 utc:2597: gvmd: Use a command like this: gvmd --migrate

```

□□

gvmd --migrate

role "root" does not exist□

```

(ron@cc140-87)-[~]
$ sudo gvmd --migrate
(gvmd:4844): md manage-WARNING **: 13:29:50.001: sql_open: PQconnectPoll failed
(gvmd:4844): md manage-WARNING **: 13:29:50.001: sql_open: PQerrorMessage (conn): connection to server on socket "/var/run/postgresql/.s.PGSQL.5432" failed: FATAL: role "root" does not exist
(gvmd:4844): md manage-WARNING **: 13:29:50.001: init_manage_open_db: sql_open failed

```

□□

gvm-setup solve

□□□□□□□□



<https://greenbone.github.io/docs/latest/22.4/kali/index.html>

<https://www.greenbone.net/en/documents/>

<https://github.com/greenbone/>

<https://community.greenbone.net/getting-started/greenbone-community-edition-via-linux-distribution-packages/>

<https://forum.greenbone.net/t/your-database-is-too-old-for-this-version-of-gvmd/20812>



nslookup

ip

nslookup 2022.4inlibra.com {dns}

```
ron@ubuntu2:~$ nslookup 2022.4inlibra.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   2022.4inlibra.com
Address: 118.163.77.237
```

whois

Dns

whatweb

ip

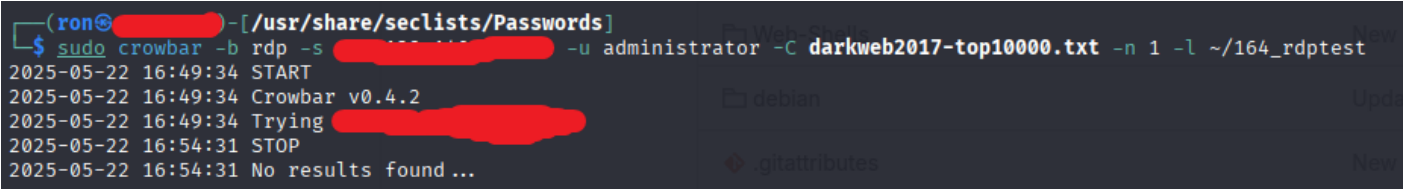


crowbar

kali

```
sudo apt install crowbar
```

```
crowbar -b {ssh,ftp...} -u {username} -c {password} -n {session number} -l {logfile}
#-u[ ]-U [ ]txt[ ]-c[ ]-C [ ]txt[ ]
```



ip/32