

Lab - Vulnerability Scanning with Kali Tools

Objectives

In this lab, you will explore network vulnerability scanning tools and use them to perform a vulnerability scan on a target host.

- Perform network scans with Nmap.
- Use Greenbone Vulnerability Management to perform a vulnerability scan.

Background / Scenario

In a previous lab, you used Nmap to enumerate a host computer that was creating unusual traffic on the network. In this lab, you will use Nmap and Greenbone Vulnerability Management (GVM) to scan the system to identify potential vulnerabilities.

Required Resources

- Kali VM customized for the Ethical Hacker course
- Internet access

Instructions

Part 1: Run a Nmap Scan on a Target Computer

In this part, you will use Nmap and NSE scripts to uncover potential vulnerabilities in a target host.

Step 1: Start and login to the Kali virtual machine.

- a. Start and log into the Kali virtual machine.
- b. Start a terminal session. Expand the terminal window to a full screen. Use the **ping** command to determine if the computer with the address **10.6.6.23** or **gravemind.vm** is reachable over the network.

```
(kali㉿Kali)-[~]
$ ping -c5 10.6.6.23
PING 10.6.6.23 (10.6.6.23) 56(84) bytes of data.
64 bytes from 10.6.6.23: icmp_seq=1 ttl=64 time=0.038 ms
64 bytes from 10.6.6.23: icmp_seq=2 ttl=64 time=0.031 ms
64 bytes from 10.6.6.23: icmp_seq=3 ttl=64 time=0.053 ms
64 bytes from 10.6.6.23: icmp_seq=4 ttl=64 time=0.040 ms
64 bytes from 10.6.6.23: icmp_seq=5 ttl=64 time=0.038 ms
File System
— 10.6.6.23 ping statistics —
5 packets transmitted, 5 received, 0% packet loss, time 4075ms
rtt min/avg/max/mdev = 0.031/0.040/0.053/0.007 ms

(kali㉿Kali)-[~]
$
```

Home

The **-c5** option tells the ping command to stop after five tries. In Linux, when a **-c** option is not specified the **ping** command will continue indefinitely until **CTRL-C** is issued.

Step 2: Identify open ports and services.

Review the results of a Nmap scan on the host with the IP address 10.6.6.23.

- Execute a ping scan of the target host using the **nmap -sV** command. Note the list of ports and applications that are discovered on the host. All the ports are Open

```
(kali㉿Kali)-[~]
$ nmap -sV 10.6.6.23
Starting Nmap 7.94 ( https://nmap.org ) at 2025-12-30 07:35 UTC
Nmap scan report for gravemind.vm (10.6.6.23)
Host is up (0.00018s latency).
Not shown: 994 closed tcp ports (conn-refused)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 3.0.3
22/tcp    open  ssh          OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
53/tcp    open  domain       ISC BIND 9.11.5-P4-5.1+deb10u5 (Debian Linux)
80/tcp    open  http         nginx 1.14.2
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
Service Info: Host: GRAVEMIND; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kerne
l

Service detection performed. Please report any incorrect results at https://nm
ap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 11.53 seconds

(kali㉿Kali)-[~]
$
```

NetAcad Ethical Hacker Kali VM

Identify the operating system running on the target computer using the **nmap -O** command. The OS is: **Linux 4.15 - 5.8**

```
(kali㉿Kali)-[~]
$ sudo nmap -o 10.6.6.23
[sudo] password for kali:
Starting Nmap 7.94 ( https://nmap.org ) at 2025-12-30 07:37 UTC
Nmap scan report for gravemind.vm (10.6.6.23)
Host is up (0.000052s latency).
Not shown: 994 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
53/tcp    open  domain
80/tcp    open  http
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: 02:42:0A:06:06:17 (Unknown)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.8
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1.66 seconds

(kali㉿Kali)-[~]
$
```

Step 3: Use the Nmap Vulners script to scan for vulnerabilities.

The Vulners script displays known vulnerabilities and the corresponding CVE. The Vulners script uses the open port and software version information to search for common platform enumeration (CPE) names that relate to the identified service. It then makes a request to a remote server to find out if any known vulnerabilities exist for that CPE.

- a. Use the **nmap -script** command to launch the **vulners** script. The syntax for the command is **nmap -sV --script vulners [--script-args mincvss=<arg_val>] <target>** where the script argument **mincvss** restricts the output to only those CVEs that have a higher CVSS score than the one specified in the argument.

The vulnerabilities reported will be those with a CVE score equal to or higher than 4. The output of the command should look similar to what is shown below:

```

(kali㉿Kali)-[~]
$ nmap -sV --script vulners --script-args mincvss=4 10.6.6.23
Starting Nmap 7.94 ( https://nmap.org ) at 2025-12-30 07:41 UTC
Nmap scan report for gravemind.vm (10.6.6.23)
Host is up (0.00018s latency).

PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 3.0.3
|_ vulners:
|   vsftpd 3.0.3:
|     CVE-2021-30047 7.5      https://vulners.com/cve/CVE-2021-30047
|     CVE-2021-3618  7.4      https://vulners.com/cve/CVE-2021-3618
22/tcp    open  ssh          OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
|_ vulners:
|   cpe:/a:openbsd:openssh:7.9p1:
|     PACKETSTORM:173661         9.8      https://vulners.com/packetstorm/PACKETSTORM:173661/*EXPLOIT*
|     F0979183-AE88-53B4-86CF-3AF0523F3807  9.8      https://vulners.com/githubexploit/F0979183-AE88-53B4-86CF-3AF0523F3807/*EXPLOIT*
|     CVE-2023-38408  9.8      https://vulners.com/cve/CVE-2023-38408
|     B8190CDB-3EB9-5631-9828-8064A1575B23  9.8      https://vulners.com/githubexploit/B8190CDB-3EB9-5631-9828-8064A1575B23/*EXPLOIT*
|     8FC9C5AB-3968-5F3C-825E-E8DB5379A623  9.8      https://vulners.com/cve/CVE-2023-38408

```

From the above scan we can identify that **OpenSSH service** has known exploited vulnerabilities

The CVE associated with the known level 5 and above vulnerability is the: **CVE-2023-38408**

Use the National Vulnerability Database at NIST to learn more about the identified vulnerability and how it can be exploited. <https://nvd.nist.gov/vuln/detail/CVE-2023-38408>

What level of severity is assigned to the CVE in the NIST database? **9.8 critical**

CVE-2023-38408 Detail

MODIFIED

This CVE record has been updated after NVD enrichment efforts were completed. Enrichment data supplied by the NVD may require amendment due to these changes.

Description

The PKCS#11 feature in ssh-agent before 9.3p2 has an insufficiently trustworthy search path, leading to remote code execution if an agent is forwarded to an attacker-controlled system. (Code in /usr/lib is not necessarily safe for loading into ssh-agent.) NOTE: This issue exists because of an incomplete fix for CVE-2016-10009.

QUICK INFO

CVE Dictionary Entry: CVE-2023-38408
NVD Published Date: 07/19/2023
NVD Last Modified: 11/21/2024
Source: MITRE

Metrics

NVD	NIST: NVD	Base Score: 9.8 CRITICAL	Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
ADP: CISA-ADP	Base Score: 9.8 CRITICAL	Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H	

References to Advisories, Solutions, and Tools

By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this page. There may be other web sites that are more appropriate for your purpose. NIST does not necessarily endorse the views expressed, or concur with the facts presented on these sites. Further, NIST does not endorse any commercial products that may be mentioned on these sites. Please address comments about this page to nvd@nist.gov.

URL	Source(s)	Tag(s)
http://packetstormsecurity.com/files/173661/OpenSSH-Forwarded-SSH-Agent-Remote-Code-Execution.html	CVE, MITRE	Exploit Third Party Advisory VDB Entry
http://www.openwall.com/lists/oss-security/2023/07/20/1	CVE, MITRE	Exploit Mailing List Third Party Advisory

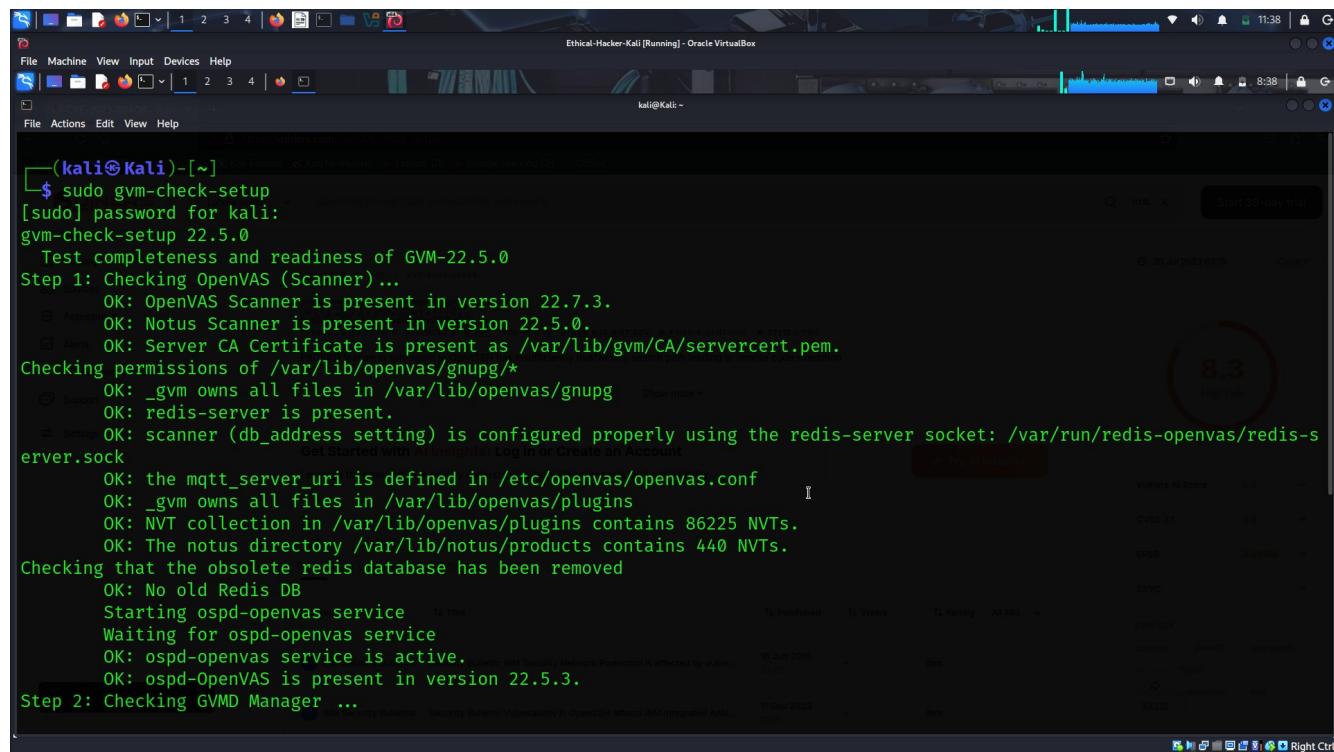
Part 2: Use GVM to Scan for Vulnerabilities

GVM is part of the Open Source Vulnerability Management suite of products produced by Greenbone Networks GmbH. The GVM scanner is one of the most widely used open-source vulnerability scanners. Unlike Nmap, GVM uses a graphical user interface to initiate scans and report vulnerability scan results.

Step 1: Verify the GVM Product Installation.

Before beginning any scan, it is important to verify that GVM is correctly installed and that the files it uses to identify vulnerabilities are up-to-date.

- a. Verify the setup of the GVM service using the **sudo gvm-check-setup** command. This command verifies that the setup completed correctly and the necessary files are available. The verification will flag any issues that need fixing and will provide the commands to use to fix the issues.



```
(kali㉿Kali)-[~] $ sudo gvm-check-setup
[sudo] password for kali:
gvm-check-setup 22.5.0
  Test completeness and readiness of GVM-22.5.0
Step 1: Checking OpenVAS (Scanner) ...
  OK: OpenVAS Scanner is present in version 22.7.3.
  OK: Notus Scanner is present in version 22.5.0.
  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 86225 NVTs.
    OK: The notus directory /var/lib/notus/products contains 440 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 22.5.3.
Step 2: Checking GVMD Manager ...
```

```
OK: ospd-openVAS is present in version 22.5.3.
Step 2: Checking GVM Manager ...
OK: GVM Manager (gvm) is present in version 22.5.5.
Step 3: Checking Certificates ...
C = DE, L = Osnabrueck, O = GVM Users, CN = Ethical-Hacker-Kali.vm
error 10 at 0 depth lookup: certificate has expired
error /var/lib/gvm/CA/clientcert.pem: verification failed
OK: GVM client certificate is valid and present as /var/lib/gvm/CA/clientcert.pem.
ERROR: Your GVM certificate infrastructure did NOT pass validation.
FIX: Run 'sudo runuser -u _gvm -- gvm-manage-certs -a'.

ERROR: Your GVM-22.5.0 installation is not yet complete!

Please follow the instructions marked with FIX above and run this
script again.

(kali㉿Kali)-[~]
$
```

The screenshot shows a terminal window on a Kali Linux system. The terminal output indicates that the GVM certificate has expired and failed verification. A red error message states: "ERROR: Your GVM certificate infrastructure did NOT pass validation." A blue fix command is provided: "FIX: Run 'sudo runuser -u _gvm -- gvm-manage-certs -a'". Above the terminal, there is a circular badge with a score of 8.3 and the text "High risk". Below the terminal, a status bar shows various system metrics like CPU usage, memory, and disk space.

If there are issues, execute the suggested command to fix the problem

```
File Machine View Input Devices Help
File Actions Edit View Help
(kali㉿Kali)-[~]
$ sudo runuser -u _gvm -- gvm-manage-certs -a
Existing certificate infrastructure found, aborting.
Use '-f' parameter to overwrite existing certificates.

(kali㉿Kali)-[~]
$ sudo runuser -u _gvm -- gvm-manage-certs -af
Generated private key in /tmp/tmp.szj2dhMrPG/cakey.pem.
Generated self signed certificate in /tmp/tmp.szj2dhMrPG/cacert.pem.
Installed private key to /var/lib/gvm/private/CA/cakey.pem.
Installed certificate to /var/lib/gvm/CA/cacert.pem.
Generated private key in /tmp/tmp.szj2dhMrPG/serverkey.pem.
Generated certificate request in /tmp/tmp.szj2dhMrPG/serverrequest.pem.
Signed certificate request in /tmp/tmp.szj2dhMrPG/serverrequest.pem with CA certificate in /var/lib/gvm/CA/cacert.pem to generate certificate in /tmp/tmp.szj2dhMrPG/servercert.pem
Installed private key to /var/lib/gvm/private/CA/serverkey.pem.
Installed certificate to /var/lib/gvm/CA/servercert.pem.
Generated private key in /tmp/tmp.szj2dhMrPG/clientkey.pem.
Generated certificate request in /tmp/tmp.szj2dhMrPG/clientrequest.pem.
Signed certificate request in /tmp/tmp.szj2dhMrPG/clientrequest.pem with CA certificate in /var/lib/gvm/CA/cacert.pem to generate certificate in /tmp/tmp.szj2dhMrPG/clientcert.pem
Installed private key to /var/lib/gvm/private/CA/clientkey.pem.
Installed certificate to /var/lib/gvm/CA/clientcert.pem.
Removing temporary directory /tmp/tmp.szj2dhMrPG.

(kali㉿Kali)-[~]
```

The screenshot shows the terminal window after running the fix command. The output shows the process of generating new certificates and keys, signing requests, and installing them. The terminal prompt "(kali㉿Kali)-[~]" is visible at the bottom. The status bar at the top right shows a score of 8.3 and the text "High risk".

and then re-run the **gvm-check-setup** command. When all issues are addressed, the command outputs the string “**It seems like your GVM [version] installation is OK.**”

```

16435|pg-gvm|10|2200|f|22.5|| OK: At least one user exists.
Step 6: Checking Greenbone Security Assistant (GSA) ...
OK: Greenbone Security Assistant is present in version 22.05.1~git.
Step 7: Checking if GVM services are up and running ...
Starting gvmd service
Waiting for gvmd service
OK: gvmd service is active.
Starting gsad service
Waiting for gsad service
OK: gsad service is active.
Step 8: Checking few other requirements ...
OK: nmap is present.
OK: ssh-keygen found, LSC credential generation for GNU/Linux targets is likely to work.
OK: nsis found, LSC credential package generation for Microsoft Windows targets is likely to work.
OK: xsltproc found.
WARNING: Your password policy is empty.
SUGGEST: Edit the /etc/gvm/pwpolicy.conf file to set a password policy.
Step 9: Checking greenbone-security-assistant ...
OK: greenbone-security-assistant is installed

It seems like your GVM-22.5.0 installation is OK.

```

(kali㉿Kali)-[~]

b. Just for this activity, stop the GVM service so you can observe the startup output.

```

(kali㉿Kali)-[~]
$ sudo gvm-stop
[>] Stopping GVM services
o gsad.service - Greenbone Security Assistant daemon (gsad)
  └─ Loaded: loaded (/lib/systemd/system/gsad.service; disabled; preset: disabled)
    └─ Active: inactive (dead)
      Docs: man:gsad(8)
      https://www.greenbone.net
Dec 30 08:43:12 Kali systemd[1]: Starting gsad.service - Greenbone Security Assistant daemon (gsad)...
Dec 30 08:43:12 Kali systemd[1]: Started gsad.service - Greenbone Security Assistant daemon (gsad).
Dec 30 08:45:42 Kali systemd[1]: Stopping gsad.service - Greenbone Security Assistant daemon (gsad)...
Dec 30 08:45:42 Kali systemd[1]: gsad.service: Deactivated successfully.
Dec 30 08:45:42 Kali systemd[1]: Stopped gsad.service - Greenbone Security Assistant daemon (gsad).

o gvmd.service - Greenbone Vulnerability Manager daemon (gvmd)
  └─ Loaded: loaded (/lib/systemd/system/gvmd.service; disabled; preset: disabled)
    └─ Active: inactive (dead)
      Docs: man:gvmd(8)
Dec 30 08:42:59 Kali systemd[1]: Starting gvmd.service - Greenbone Vulnerability Manager daemon (gvmd)...
Dec 30 08:42:59 Kali systemd[1]: gvmd.service: Can't open PID file /run/gvmd/gvmd.pid (yet?) after start: No such file or directory
Dec 30 08:43:02 Kali systemd[1]: Started gvmd.service - Greenbone Vulnerability Manager daemon (gvmd).
Dec 30 08:45:42 Kali systemd[1]: Stopping gvmd.service - Greenbone Vulnerability Manager daemon (gvmd)...

```

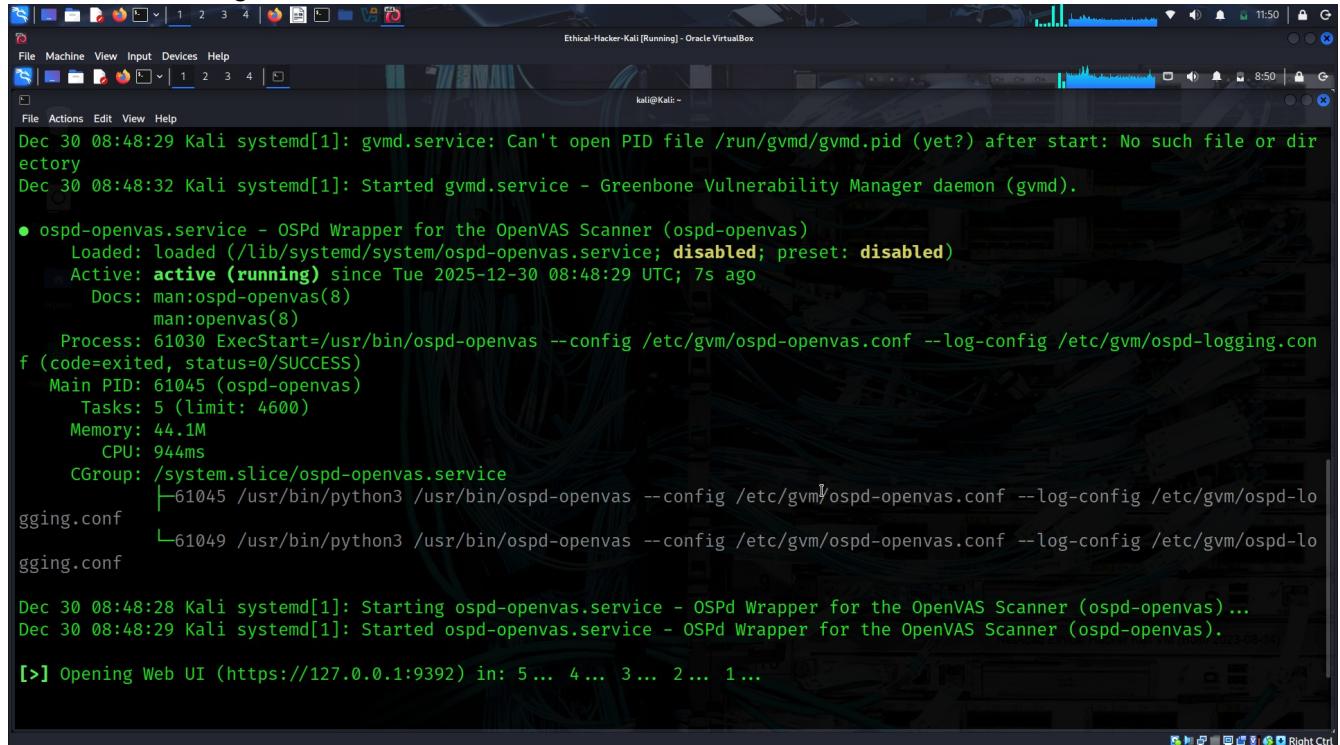
Step 2: Open the GVM Scanner GUI.

a. Start the GVM scanner using the **sudo gvm-start** command. You can also access the **gvm-start** script using the Applications menu on the Kali desktop, **Kali ->02-Vulnerability Analysis ->**

gvm start. It is possible that GVM may already be running as a result of the check setup process.

The output of the command should be similar to what is shown below. At the end of the output, a message that the scanner is loading in Firefox will appear.

Command: sudo gvm-start



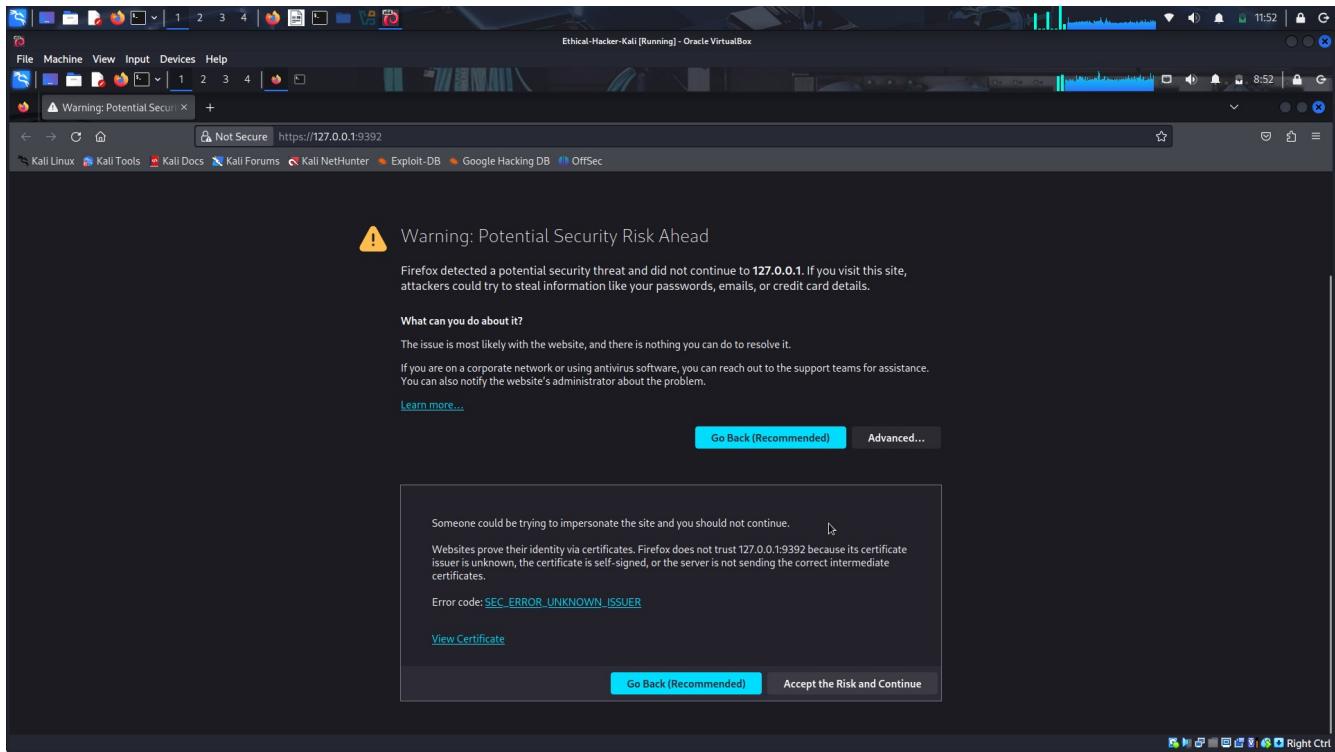
```
Dec 30 08:48:29 Kali systemd[1]: gvmd.service: Can't open PID file /run/gvmd/gvmd.pid (yet?) after start: No such file or directory
Dec 30 08:48:32 Kali systemd[1]: Started gvmd.service - Greenbone Vulnerability Manager daemon (gvmd).

● ospd-openvas.service - OSPD Wrapper for the OpenVAS Scanner (ospd-openvas)
   Loaded: loaded (/lib/systemd/system/ospd-openvas.service; disabled; preset: disabled)
   Active: active (running) since Tue 2025-12-30 08:48:29 UTC; 7s ago
     Docs: man:ospd-openvas(8)
           man:openvas(8)
     Process: 61030 ExecStart=/usr/bin/ospd-openvas --config /etc/gvm/ospd-openvas.conf --log-config /etc/gvm/ospd-logging.conf (code=exited, status=0/SUCCESS)
    Main PID: 61045 (ospd-openvas)
      Tasks: 5 (limit: 4600)
     Memory: 44.1M
        CPU: 944ms
       CGroup: /system.slice/ospd-openvas.service
               ├─61045 /usr/bin/python3 /usr/bin/ospd-openvas --config /etc/gvm/ospd-openvas.conf --log-config /etc/gvm/ospd-logging.conf
               └─61049 /usr/bin/python3 /usr/bin/ospd-openvas --config /etc/gvm/ospd-openvas.conf --log-config /etc/gvm/ospd-logging.conf

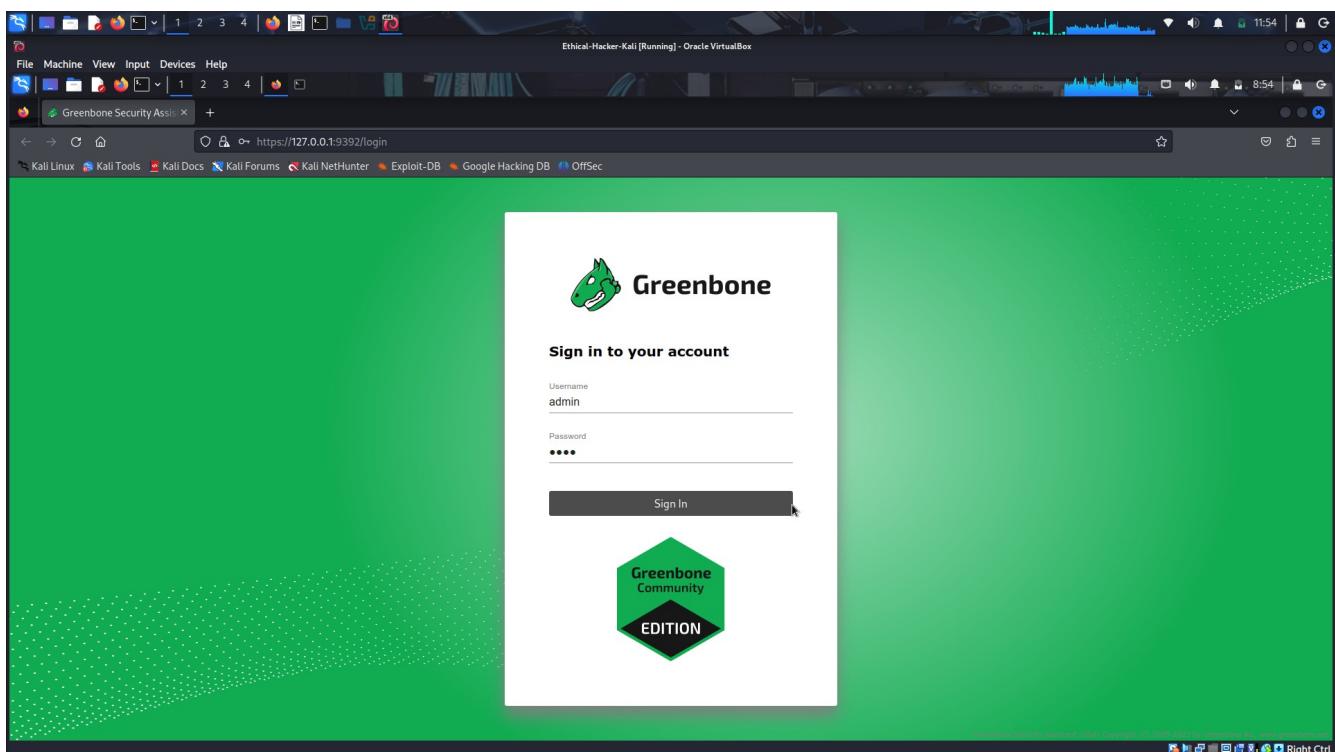
Dec 30 08:48:28 Kali systemd[1]: Starting ospd-openvas.service - OSPD Wrapper for the OpenVAS Scanner (ospd-openvas)...
Dec 30 08:48:29 Kali systemd[1]: Started ospd-openvas.service - OSPD Wrapper for the OpenVAS Scanner (ospd-openvas).

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

A browser window will open with a security warning that can be ignored. If the browser does not automatically open, start your browser manually and navigate to <https://127.0.0.1:9392>. Click the **Advanced** button and scroll down and accept the risk on the warning screen to proceed.



In the Greenbone Security Assistant login box, enter **admin** as the username and **kali** as the password.



The GVM Scanner application GUI should open in the browser. Select **Scans -> Tasks** from the menu bar. At the upper left of the **Tasks** window appear three icons.

The screenshot shows the Greenbone Security Assistant interface running in a browser window. The title bar indicates it's on a Kali Linux machine within an Oracle VirtualBox environment. The main content area displays three large, light-grey circular charts under the heading 'Tasks 0 of 0'. Below these charts, a message states 'No Tasks available' and includes a note about the applied filter: '(Applied filter: apply_overrides=0 min_god=70 sort=name first=1 rows=10)'. The bottom right corner of the interface shows the copyright notice 'Greenbone Security Assistant (GSA) Copyright (C) 2009-2023 by Greenbone AG, www.greenbone.net'.

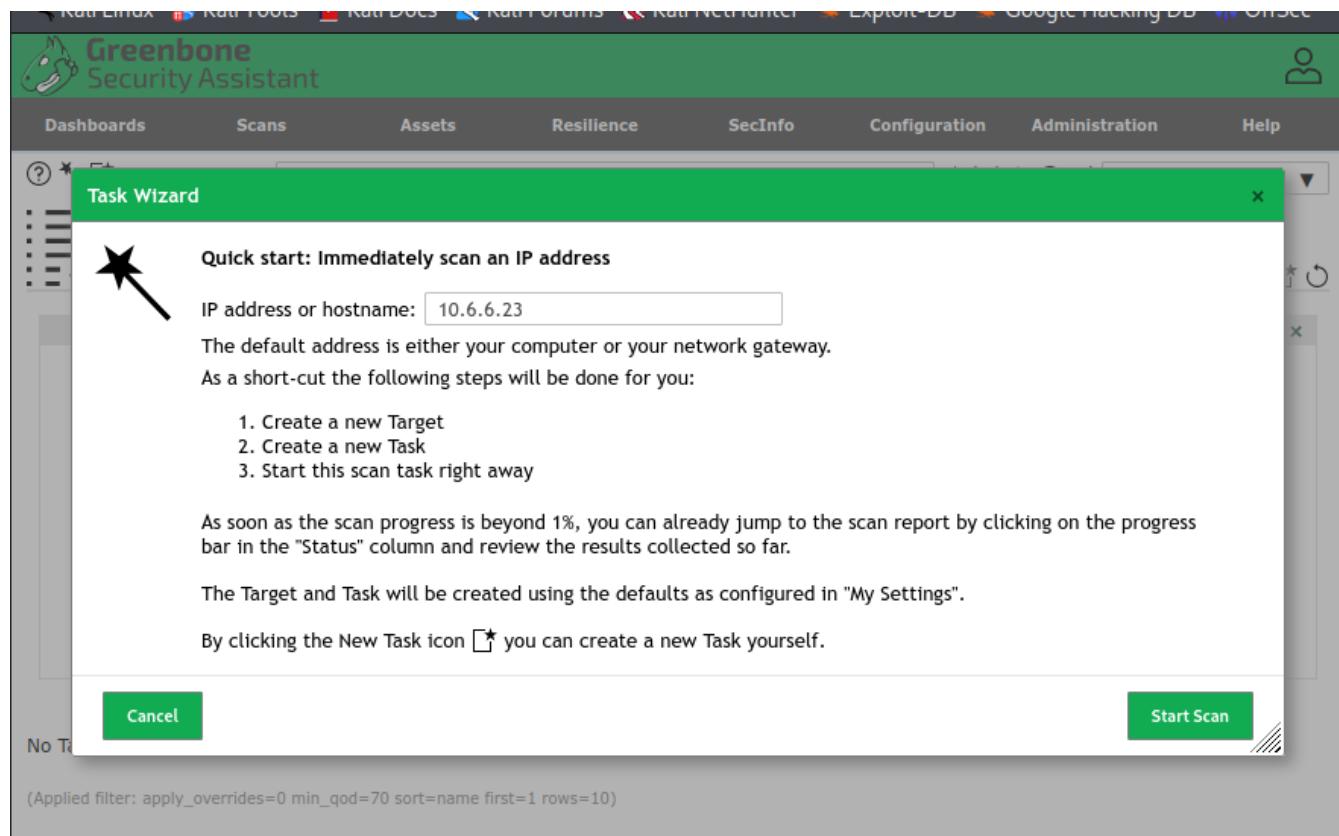
Select the **Task Wizard** icon that looks like a magic wand. Choose **Task Wizard** from the dropdown menu.

The screenshot shows the same Greenbone Security Assistant interface, but now with a 'Task Wizard' dialog box overlaid. The dialog box has a green header bar with the text 'Task Wizard' and a sub-header 'Quick start: Immediately scan an IP address'. It contains a text input field labeled 'IP address or hostname:' with the value '127.0.0.1'. Below the input field, there is explanatory text: 'The default address is either your computer or your network gateway.' and 'As a short-cut the following steps will be done for you:'. A numbered list follows: '1. Create a new Target', '2. Create a new Task', and '3. Start this scan task right away'. At the bottom of the dialog box, there is more text: 'As soon as the scan progress is beyond 1%, you can already jump to the scan report by clicking on the progress bar in the "Status" column and review the results collected so far.' and 'The Target and Task will be created using the defaults as configured in "My Settings". By clicking the New Task icon you can create a new Task yourself.' There are 'Cancel' and 'Start Scan' buttons at the bottom right of the dialog. The background of the interface remains visible, showing the three circular charts and the 'No Tasks available' message.

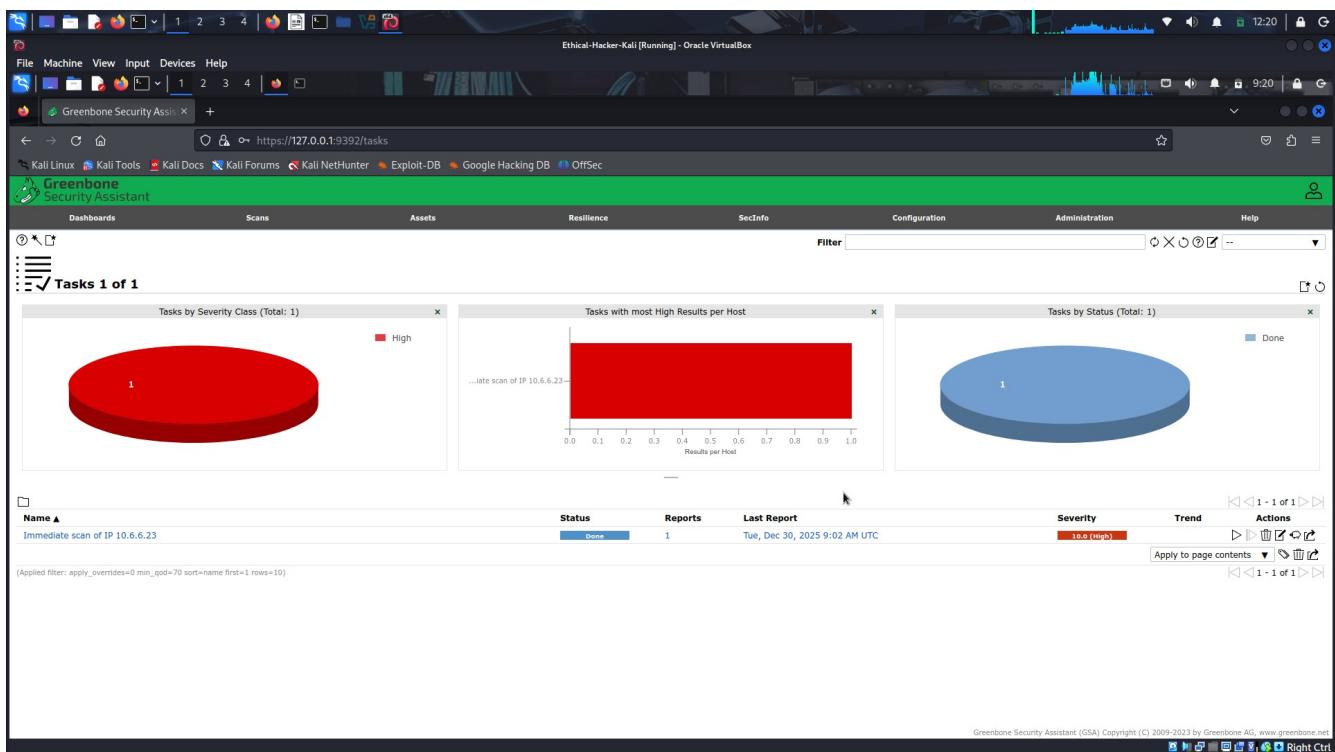
Step 3: Scan the Target Host for Vulnerabilities

In this step you will scan the same target computer for vulnerabilities that you did with the earlier Nmap scan.

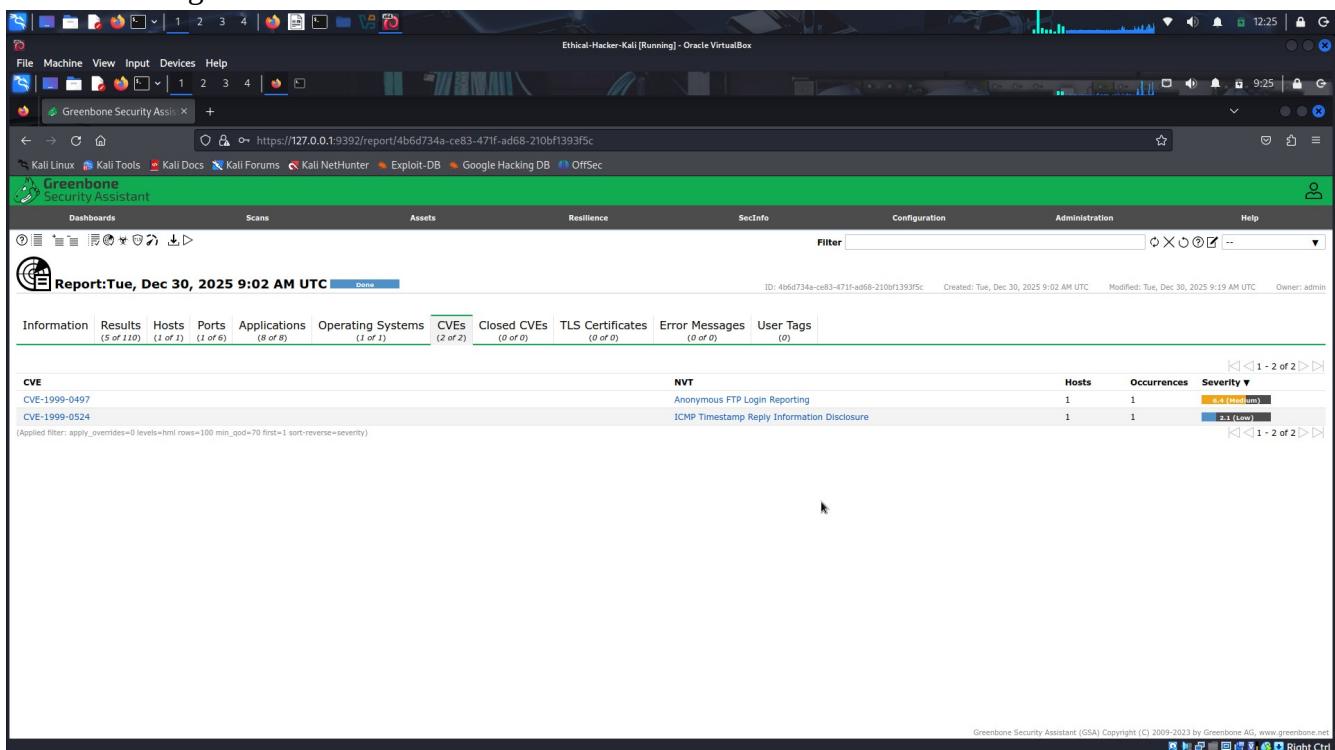
- a. In the **IP address or hostname** box, enter the IP address **10.6.6.23** or **gravemind.vm**. Click the **Start Scan** button at the bottom of the screen.



The scan will take a few minutes, so wait for it to complete. The status and percent complete are displayed on the screen. The scan will be finished when the status changes to **Done**.



- b. Click the number under the **Reports** column while the scanning is running for the associated scan.
- c. When the scan is complete, click the timestamp under the **Date** column to view the report detail.
- d. The CVEs associated with the vulnerabilities that were found on the host can be viewed by clicking the **CVEs** tab.



Explore the other tabs

File Machine View Input Devices Help

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

Greenbone Security Assistant

Report:Tue, Dec 30, 2025 9:02 AM UTC Done

Information Results Hosts Ports Applications Operating Systems CVEs Closed CVEs TLS Certificates Error Messages User Tags

Vulnerability

	Severity ▾	QoD	Host IP	Name	Location	Created
Operating System (OS) End of Life (EOL) Detection	10.0 (High)	80 %	10.6.6.23	gravemind.vm	general/tcp	Tue, Dec 30, 2025 9:08 AM UTC
Anonymous FTP Login Reporting	6.4 (Medium)	80 %	10.6.6.23	gravemind.vm	21/tcp	Tue, Dec 30, 2025 9:08 AM UTC
FTP Unencrypted Cleartext Login	4.8 (Medium)	70 %	10.6.6.23	gravemind.vm	21/tcp	Tue, Dec 30, 2025 9:08 AM UTC
TCP Timestamps Information Disclosure	2.4 (Low)	80 %	10.6.6.23	gravemind.vm	general/tcp	Tue, Dec 30, 2025 9:08 AM UTC
ICMP Timestamp Reply Information Disclosure	2.1 (Low)	80 %	10.6.6.23	gravemind.vm	general/icmp	Tue, Dec 30, 2025 9:08 AM UTC

(Applied filter: apply_overrides=0 levels=hml rows=100 min_qod=70 first=1 sort-reverse+severity)

Greenbone Security Assistant (GSA) Copyright (C) 2009-2023 by Greenbone AG, www.greenbone.net

hosts

Report:Tue, Dec 30, 2025 9:02 AM UTC Done

Information Results Hosts Ports Applications Operating Systems CVEs Closed CVEs TLS Certificates Error Messages User Tags

IP Address	Hostname	OS	Ports	Apps	Distance	Auth	Start	End	High	Medium	Low	Log	False Positive	Total	Severity ▾
10.6.6.23	gravemind.vm	Ubuntu	1	8			Tue, Dec 30, 2025 9:03 AM UTC	Tue, Dec 30, 2025 9:19 AM UTC	1	2	2	0	0	5	10.0 (High)

(Applied filter: apply_overrides=0 levels=hml rows=100 min_qod=70 first=1 sort-reverse+severity)

app:

Report:Tue, Dec 30, 2025 9:02 AM UTC Done

Information Results Hosts Ports Applications Operating Systems CVEs Closed CVEs TLS Certificates Error Messages User Tags

Application CPE	Hosts	Occurrences	Severity ▾
cpe:/a:beasts:vstpd:3.0.3	1	1	N/A
cpe:/a:jquery:jquery	1	1	N/A
cpe:/a:jquery:jquery:3.5.1	1	1	N/A
cpe:/a:nginx:nginx:1.14.2	1	1	N/A
cpe:/a:openbsd:openssl:7.9p1	1	1	N/A
cpe:/a:rs:nginx:1.14.2	1	1	N/A
cpe:/a:samba:samba:4.9.5	1	1	N/A
cpe:/a:isc:bind:9.11.5:p4	1	1	N/A

(Applied filter: apply_overrides=0 levels=hml rows=100 min_qod=70 first=1 sort-reverse+severity)

OS:

The screenshot shows a detailed report from the Greenbone Security Assistant (GSA) interface. At the top, it displays the date and time as 'Report: Tue, Dec 30, 2025 9:02 AM UTC'. Below this is a navigation bar with tabs for 'Information' (5 of 110), 'Results' (1 of 1), 'Hosts' (1 of 6), 'Ports' (8 of 8), 'Applications' (0 of 0), 'Operating Systems' (1 of 1), 'CVEs' (2 of 2), 'Closed CVEs' (0 of 0), 'TLS Certificates' (0 of 0), 'Error Messages' (0 of 0), and 'User Tags' (0). The 'Operating Systems' section shows one host running 'Debian GNU/Linux 10' with a CPE entry of 'cpe:/o:debian:debian_linux:10'. The 'Hosts' section shows 1 host with a severity of 10.0 (High). A status bar at the bottom indicates the report ID is 4b6d734a-ce83-471f-ad68-210bf1393fc, created on Tue, Dec 30, 2025 9:02 AM UTC, modified on Tue, Dec 30, 2025 9:19 AM UTC, and owned by admin.

- Download the report by clicking the **Download Filtered Report** button from the menu in the upper left of the report page. It has a downward-pointing arrow icon. In the settings box, choose to download the report in PDF format. After a brief delay, the PDF file should open in your browser.

The screenshot shows the GSA dashboard with three main visualizations: a pie chart showing 1 task in the 'High' severity class, a bar chart showing the results per host for an immediate scan of IP 10.6.6.23, and another pie chart showing 1 task in the 'Done' status. A download dialog box in the top right corner shows a file named 'task-db4aae9e-0565-436a-b829-3942c5e7cc6.xml' completed at 3.4 KB. The bottom part of the screen shows a table of scan results for the IP 10.6.6.23, with columns for Name, Status, Reports, Last Report, Severity, Trend, and Actions. The status is 'Done', reports are 1, and the last report was on Tue, Dec 30, 2025 9:02 AM UTC. The severity is 10.0 (High). A status bar at the bottom indicates the report ID is 4b6d734a-ce83-471f-ad68-210bf1393fc, created on Tue, Dec 30, 2025 9:02 AM UTC, modified on Tue, Dec 30, 2025 9:19 AM UTC, and owned by admin.

Are the CVEs reported by GVM the same as the CVEs reported by the Nmap scan? **No. The two scanning tools may use different vulnerability databases to make their identifications.**

Step 4: Clean Up

When you are done with GVM services, use the following command to stop GVM. **sudo gym-stop**

The screenshot shows a terminal window titled "Ethical-Hacker-Kali [Running] - Oracle VirtualBox". The terminal is running on a Kali Linux system. The user has run the command `sudo gvm-stop`. The terminal output is as follows:

```
(kali㉿kali)-[~]
$ sudo gvm-stop
[sudo] password for kali:
[>] Stopping GVM services
o gsad.service - Greenbone Security Assistant daemon (gsad)
  Loaded: loaded (/lib/systemd/system/gsad.service; disabled; preset: disabled)
  Active: inactive (dead)
    Docs: man:gsad(8)
          https://www.greenbone.net

Dec 30 08:48:37 Kali systemd[1]: Started gsad.service - Greenbone Security Assistant daemon (gsad).
Dec 30 08:51:03 Kali systemd[1]: Stopping gsad.service - Greenbone Security Assistant daemon (gsad)...
Dec 30 08:51:03 Kali systemd[1]: gsad.service: Deactivated successfully.
Dec 30 08:51:03 Kali systemd[1]: Stopped gsad.service - Greenbone Security Assistant daemon (gsad).
Dec 30 08:51:19 Kali systemd[1]: Starting gsad.service - Greenbone Security Assistant daemon (gsad)...
Dec 30 08:51:19 Kali systemd[1]: Started gsad.service - Greenbone Security Assistant daemon (gsad).
Dec 30 09:40:45 Kali systemd[1]: Stopping gsad.service - Greenbone Security Assistant daemon (gsad)...
Dec 30 09:40:45 Kali systemd[1]: gsad.service: Deactivated successfully.
Dec 30 09:40:45 Kali systemd[1]: Stopped gsad.service - Greenbone Security Assistant daemon (gsad).
Dec 30 09:40:45 Kali systemd[1]: gsad.service: Consumed 3.681s CPU time.

o gvmd.service - Greenbone Vulnerability Manager daemon (gvmd)
  Loaded: loaded (/lib/systemd/system/gvmd.service; disabled; preset: disabled)
  Active: inactive (dead)
    Docs: man:gvmd(8)
```

NOTE: New exploits and vulnerabilities are discovered every day. Not all systems are patched and up-to-date on security. Therefore, it is necessary to keep both new and older CVEs in the database.