Experiment–07

7.1 Aim:

Vulnerability Scanning with OpenVAS/Nessus Essentials: Use an automated scanner to identify potential vulnerabilities on target systems to enhance system security and patch management.

7.2 Course Outcome:

Apply vulnerability scanning techniques using automated tools to detect system weaknesses, assess potential risks, and recommend appropriate mitigations.

7.3 Lab Objective:

To scan a target system using OpenVAS or Nessus Essentials and analyze the discovered vulnerabilities for severity, type, and remediation strategies.

7.4 Requirements:

Operating System: Linux (Kali Linux, Ubuntu) / Windows / macOS
Tools:

- OpenVAS (Greenbone Vulnerability Management) or Nessus Essentials (Tenable)
- Web browser interface for managing scans
- Target system (could be localhost or a test VM)
- Internet connection for vulnerability database updates

7.5 Theory:

Vulnerability scanning is an automated process that detects known weaknesses in systems, applications, and configurations. It is a critical component of a cybersecurity defense strategy, helping organizations proactively fix exploitable flaws before attackers can leverage them.

OpenVAS (now part of Greenbone) and **Nessus Essentials** are two widely used vulnerability scanners that offer detailed insights, including:

- CVE (Common Vulnerabilities and Exposures) identification
- CVSS (Common Vulnerability Scoring System) severity ratings
- Affected ports/services
- Suggested patches or mitigations

Vulnerability scanners perform tasks such as:

- Host discovery
- Port scanning
- Service enumeration
- Vulnerability database matching
- Report generation

7.6 Tasks:

1. Install and configure OpenVAS or Nessus Essentials on your machine or VM.
2. Update the vulnerability database (important for accurate detection).
3. Define the target system IP (e.g., localhost, internal server, or test VM).
4. Initiate a **Full System Scan** or **Basic Network Scan**.
5. Monitor the progress and wait for the scan to complete.
6. After scanning, open the report and review:
    - Vulnerability names and severity (Low/Medium/High/Critical)
    - CVE IDs and references
    - Description and risk level
    - Affected services and ports
    - Suggested remediation steps
7. Export or save the vulnerability report in PDF or HTML format.
8. Optionally, rescan after patching to verify if vulnerabilities were mitigated.

7.7 Output Screenshots:

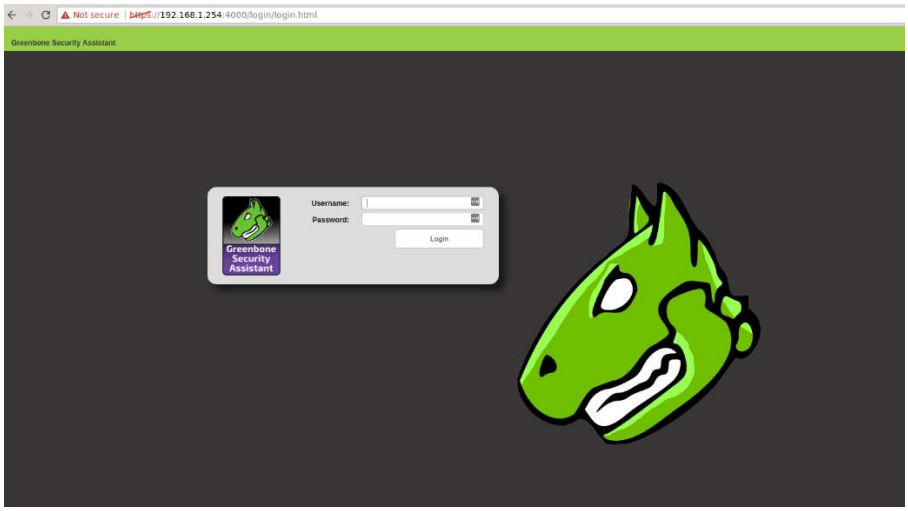
- OpenVAS/Nessus dashboard
- Scan configuration setup
- Scan results summary (severity breakdown)
- Detailed view of a critical vulnerability
- Exported report snippet
- Before and after scan comparison (optional)

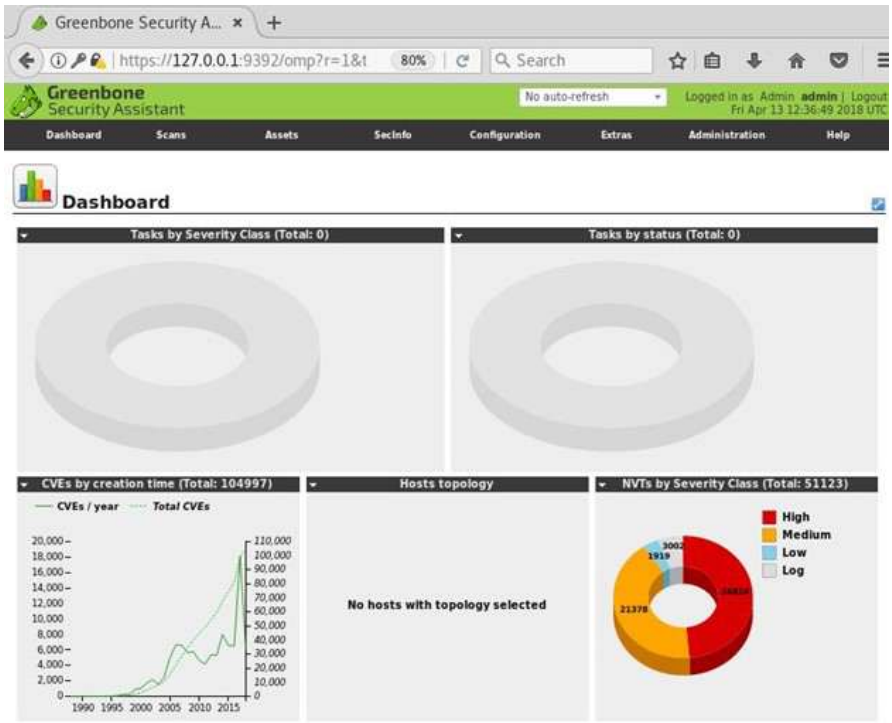
```
┌──(hassen㉿hannachi)-[~]
└─$ sudo apt-get update
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [19.9 MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [47.3 MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [121 kB]
Get:5 http://kali.download/kali kali-rolling/contrib amd64 Contents (deb) [261 kB]
Get:6 http://kali.download/kali kali-rolling/non-free amd64 Packages [194 kB]
Get:7 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [885 kB]
Get:8 http://kali.download/kali kali-rolling/non-free-firmware amd64 Packages [33.0 kB]
Get:9 http://kali.download/kali kali-rolling/non-free-firmware amd64 Contents (deb) [16.8 kB]
Fetched 68.8 MB in 33s (2,115 kB/s)
Reading package lists ... Done
```

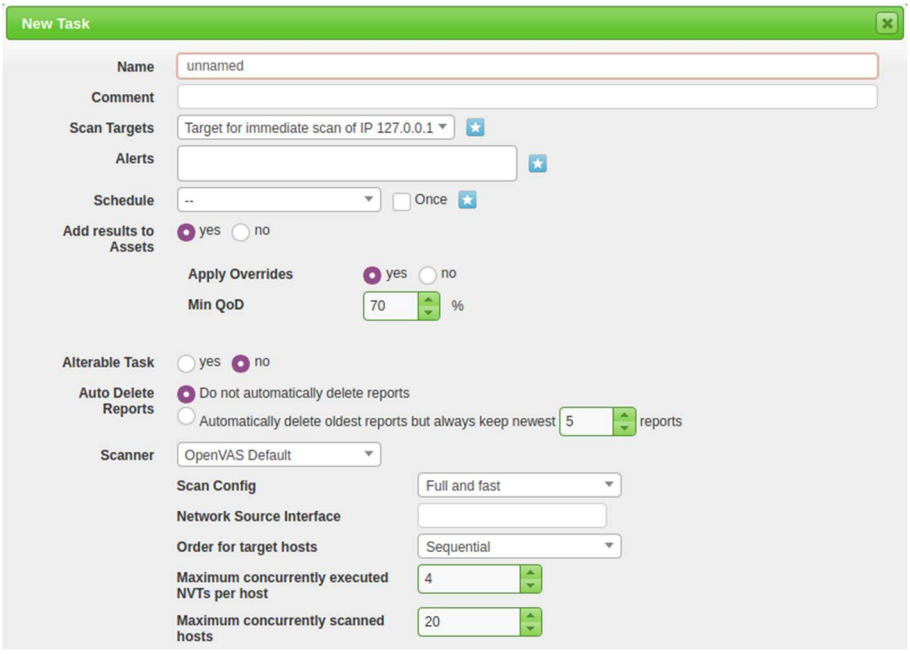
```
┌──(hassen㉿hannachi)-[~]
└─$ sudo apt upgrade
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
Calculating upgrade ... Done
The following packages were automatically installed and are no longer required:
  libboost-dev libboost1.83-dev libopenblas-dev libopenblas-pthread-dev libopenblas0 libpython3-all-dev
  libpython3.12 libpython3.12-dev libxsimd-dev python3-all-dev python3-beniget python3-gast
  python3-pythran python3.12-dev xtl-dev
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  libgck-2-2 libgcr-4-4 libjq1 libonig5 python3-pyasyncore
The following packages will be upgraded:
  adwaita-icon-theme autopsy bind9-dnsutils bind9-host bind9-libs binutils binutils-common
  binutils-x86-64-linux-gnu binwalk colord colord-data console-setup console-setup-linux debconf
  debconf-i18n dnsmasq-base exploitdb firefox-esr firmware-linux-free fontconfig fontconfig-config
  fonts-lyx gdal-data gdal-plugins geoip-database gir1.2-gstreamer-1.0 gir1.2-nm-1.0 gir1.2-vte-2.91
  glib-networking glib-networking-common glib-networking-services go-l2tp gsettings-desktop-schemas
  gstreamer1.0-gl gstreamer1.0-libav gstreamer1.0-plugins-bad gstreamer1.0-plugins-base
  gstreamer1.0-plugins-good gstreamer1.0-x gvfs gvfs-backends gvfs-common gvfs-daemons gvfs-fuse
  gvfs-libs iputils-ping isc-dhcp-client isc-dhcp-common iso-codes keyboard-configuration kmod
  libadwaita-1-0 libappstream5 libatkmm-1.6-1v5 libaudio2 libavif16 libbinutils libboost-dev
  libbson-1.0-0 libc-ares2 libcapstone-dev libcapstone4 libcolord2 libcolorhug2 libcompress-raw-lzma-perl
  libctf-nobfd0 libctf0 libdav1d7 libdaxctl1 libeac3 libencode-perl libfontconfig1 libgdal34
  libgdata-common libgdata22 libgprofng0 libgstreamer-gl1.0-0 libgstreamer-plugins-bad1.0-0
  libgstreamer-plugins-base1.0-0 libgstreamer1.0-0 libgtk-layer-shell0 libgtop-2.0-11 libgtop2-common
  libhogweed6 libkmod2 libldb2 liblua5.4-0 libmanette-0.2-0 libmd4c0 libmjpegutils-2.1-0 libmongoc-1.0-0
  libmosquitto1 libmousepad0 libmpeg2encpp-2.1-0 libmpfr6 libmplex2-2.1-0 libmtdev1 libmujs3 libndctl6
  libnet-dns-perl libnetcdf19 libnettle8 libnghttp3-3 libnm0 libnpth0 libnspr4 libnss3 libnvme1
```

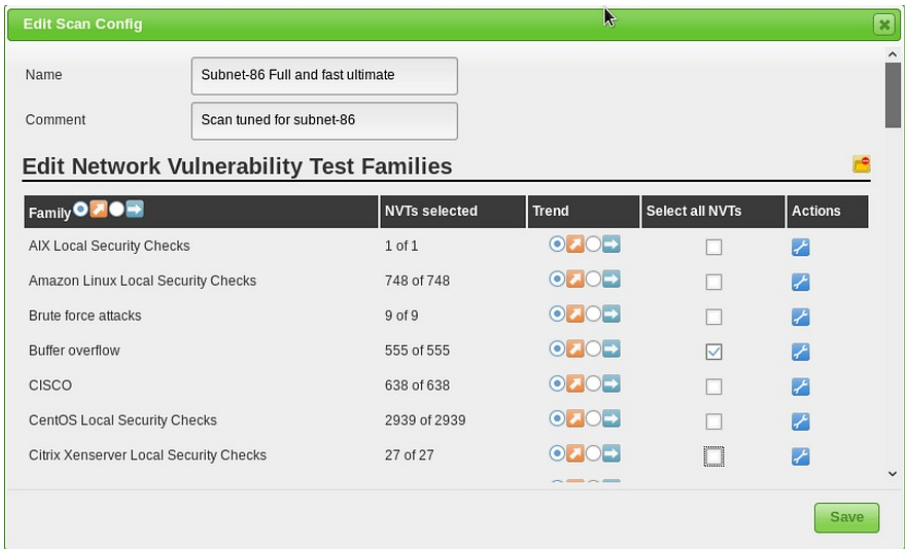
```
┌──(hassen㉿hannachi)-[~]
└─$ sudo apt dist-upgrade -y
[sudo] password for hassen:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
Calculating upgrade ... Done
The following packages were automatically installed and are no longer required:
  libboost-dev libboost1.83-dev libopenblas-dev libopenblas-pthread-dev libopenblas0 libpython3-all-dev
  libpython3.12 libpython3.12-dev libxsimd-dev python3-all-dev python3-beniget python3-gast
  python3-pythran python3.12-dev xtl-dev
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.

┌──(hassen㉿hannachi)-[~]
└─$
```

```
┌──(hassen㉿hannachi)-[~]
└─$ sudo apt install openvas
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
Note, selecting 'gvm' instead of 'openvas'
The following packages were automatically installed and are no longer required:
  libboost-dev libboost1.83-dev libopenblas-dev libopenblas-pthread-dev libopenblas0 libpython3-all-dev
  libpython3.12 libpython3.12-dev libxsimd-dev python3-all-dev python3-beniget python3-gast
  python3-pythran python3.12-dev xtl-dev
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  greenbone-security-assistant gsad gvm-tools libmicrohttpd12
The following NEW packages will be installed:
  greenbone-security-assistant gsad gvm gvm-tools libmicrohttpd12
0 upgraded, 5 newly installed, 0 to remove and 0 not upgraded.
Need to get 5,153 kB of archives.
After this operation, 20.9 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```




Greenbone Security A...
https://127.0.0.1:9392/omp?r=1&t
80%
Search
Greenbone
Security Assistant
No auto-refresh
Logged in as Admin admin | Logout
Fri Apr 13 12:36:49 2018 UTC
Dashboard
Scans
Assets
SecInfo
Configuration
Extras
Administration
Help
Dashboard
Tasks by Severity Class (Total: 0)
Tasks by status (Total: 0)
CVEs by creation time (Total: 104997)
CVEs / year
Total CVEs
Hosts topology
No hosts with topology selected
NVTs by Severity Class (Total: 51123)
High
Medium
Low
Log
1919
21378


New Task
Name
unnamed
Comment
Scan Targets
Target for immediate scan of IP 127.0.0.1
Alerts
Schedule
--
Once
Add results to Assets
yes
no
Apply Overrides
yes
no
Min QoD
70
%
Alterable Task
yes
no
Auto Delete Reports
Do not automatically delete reports
Automatically delete oldest reports but always keep newest
5
reports
Scanner
OpenVAS Default
Scan Config
Full and fast
Network Source Interface
Order for target hosts
Sequential
Maximum concurrently executed NVTs per host
4
Maximum concurrently scanned hosts
20



#	Vulnerability / CVE (example)	Severity	Affected Service / Port	Brief description	Suggested remediation
1	CVE-2021-3449 (OpenSSL)	Critical	HTTPS / 443	Remote code execution / TLS vulnerability	Update OpenSSL to patched version; apply vendor patch; restart service
2	CVE-2020-25712 (SSH)	High	SSH / 22	Weak SSH configuration / outdated SSH server	Upgrade SSH server; disable weak ciphers; enforce key-based auth
3	CVE-2019- (Apache)	High	HTTP / 80	Directory traversal / outdated module	Apply vendor patch; update Apache; review modules
4	Missing MS Patch KB	Medium	SMB / 445	Known privilege escalation	Apply Windows update KB

5	Outdated PHP library	Medium	HTTP (PHP-FPM)	Known vulnerabilities in library versions	Update PHP libraries / composer packages
6	Information disclosure	Low	FTP / 21	Banner reveals version info	Hide version banners; minimize exposed info
7	Weak TLS cipher supported	Low	HTTPS / 443	Use of TLS 1.0 / weak ciphers	Disable TLS1.0/1.1; enable TLS1.2+; update cipher suite

7.8 Conclusion:

In this experiment we used an automated vulnerability scanner (Nessus Essentials / OpenVAS) to identify and analyze security weaknesses on a target system. The scan produced a prioritized list of findings categorized by severity (Critical → Low) and provided CVE identifiers, affected services, and recommended remediation steps. The most significant finding was a critical OpenSSL-related vulnerability that could allow remote compromise; immediate patching of the affected library was recommended and verified via a rescan.