

7

Performing Vulnerability Assessments

As you have learned so far, the reconnaissance phase is very important for successfully moving on to the exploitation phase of penetration testing and the Cyber Kill Chain. Discovering security vulnerabilities on a targeted system helps adversaries identify the attack surface, which is the point of entry on a system that can be exploited to gain unauthorized access. As an aspiring ethical hacker and penetration tester, understanding how to efficiently identify the attack surface and profile a targeted system will help you better plan your method of attack and determine which exploits will help you gain a foothold on the target. However, it's important to ensure you obtain written legal permission from the authorities prior to performing any sort of security assessment on a targeted system or network that you do not own. In addition, ensure you adhere to ethics in ethical hacking and penetration testing.

The use of automated tools helps penetration testers reduce the time needed to identify security vulnerabilities on a targeted system, using tools such as Nessus, Nmap NSE, Greenbone Vulnerability Manager, and common web application scanners. However, it's important to note that while automated scanning tools reduce the time it takes to do vulnerability assessments, manual testing ensures that penetration testers are able to validate the findings and remove any false positives from an automated tool.

After the vulnerability verification process, the risk assessment phase focuses on assessing the severity and potential impact of each security vulnerability that's found on a system. The risk assessment phase helps organizations to better understand the ease of compromising each security vulnerability and how it can impact their business operations. In addition, each security vulnerability is assigned a risk rating score, which helps cybersecurity teams to prioritize and allocate resources to remediate security vulnerabilities based on their severity or criticality.

Next, the remediation plan is developed to help the organization address all the security vulnerabilities found within the scope of the penetration testing. This may include applying security patches to systems, performing configuration

changes to harden devices, and implementing countermeasures to safeguard the organization's assets.

The reporting phase focuses on generating a comprehensive technical and executive summary report that contains the details of identified security vulnerabilities, analysis, and recommendations to resolve the security flaws. After applying the recommendations, it's important to re-test to ensure all remediation efforts are successful in improving the security of the posture of the organization.

In this chapter, you will learn how to use Kali Linux with various popular tools to perform a vulnerability assessment on a network. You will start by learning how you can install, perform, and analyze scan results using Nessus, one of the most popular industry-recognized vulnerability scanners within the cybersecurity industry. Then, you will learn how to leverage the hidden secrets and power of Nmap to easily discover security flaws in systems. Finally, you will learn how to perform web vulnerability assessments.

In this chapter, we will cover the following topics:

- Getting started with Nessus
- Vulnerability identification using Nmap
- Working with Greenbone Vulnerability Manager
- Using web application scanners

Let's dive in!

Technical requirements

To follow along with the exercises in this chapter, please ensure that you have met the following hardware and software requirements:

- Kali Linux: <https://www.kali.org/get-kali/>
- Nessus Essentials: <https://www.tenable.com/products/nessus/nessus-essentials>
- Greenbone Vulnerability Manager: <https://github.com/greenbone/gvmd>

Getting started with Nessus

When diving into the field of cybersecurity, there is a very well-known tool everyone needs to know about, and that's Nessus. Nessus is a vulnerability scanner that can detect over 83,000 **Common Vulnerability and Exposure (CVE)** security

flaws on systems. Furthermore, Nessus allows security professionals to deploy Nessus within centralized locations and automate periodic scanning on targeted systems, which allows continuous and automated vulnerability assessment within an organization.

As an aspiring penetration tester, you may need to use Nessus to perform a vulnerability assessment within an organization, determine the risk and severity of each security flaw, and provide recommendations on how to mitigate the risk of possible cyber-attacks based on the security vulnerabilities found. In this section, you will learn how to set up and perform a vulnerability assessment using Nessus on your Kali Linux machine.

Before getting started with installing and setting up Nessus, ensure that your Kali Linux machine meets the following requirements:

- Stable internet connection
- Minimum dual-core processor
- Minimum of 4 GB RAM

Minimum of 30 GB free storage space. To get started working with Nessus Essentials, please use the following instructions.



If you're a Mac user who is running Kali Linux in Parallels on the M1 Mac (ARM64) chip, you may experience some issues when setting up Nessus within Kali Linux. However, the process works fine on a Windows-based system.

Part 1 – installing Nessus

In this part, you will learn how to install and set up Nessus Essentials on the Kali Linux virtual machine to identify security vulnerabilities on targeted systems:

1. Firstly, power on the **Kali Linux** virtual machine and ensure it has internet connectivity.
2. Next, either on Kali Linux or your host machine, open the web browser and go to <https://www.tenable.com/products/nessus/nessus-essentials> to register for a free license to activate Nessus Essentials during the setup process:

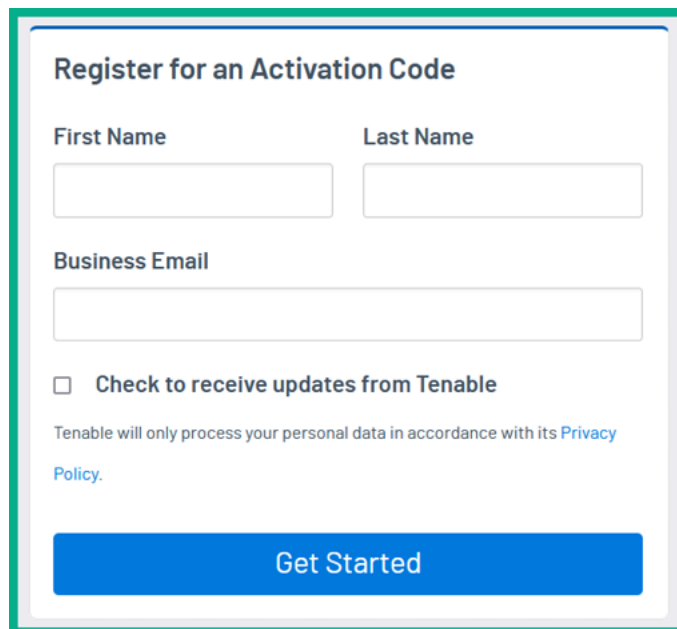


Figure 7.1: Nessus Essentials registration page

As shown in the preceding screenshot, a business email address is required to complete the registration. However, I've used a personal free email address and was able to successfully register and receive a Nessus Essentials activation code.



A business email address is required to register and receive a free Nessus Essentials activation code. However, using a personal email address works.

3. On **Kali Linux**, open the **Terminal** and use the following commands to update the local software packages repository list:

```
kali@kali:~$ sudo apt update
```

4. Next, use the following commands to download the Nessus Essentials package onto the Kali Linux virtual machine:


```
kali@kali:~$ curl -o Nessus-10.7.2-debian10_amd64.deb 'https://www.tenable.com/downloads/api/v2/pages/nessus/files/Nessu
```

The following screenshot shows the execution of the preceding commands:

```
kali@kali:~$ curl -o Nessus-10.7.2-debian10_amd64.deb 'https://www.tenable.com/downloads/api/v2/pages/nessus/files/Nessus-10.7.2-debian10_amd64.deb'
```

| % Total Current | % Received | % Xferd | Average Speed | | Time | Time | Time |
|--------------------|------------|---------|---------------|--------|----------|----------|----------|
| | | | Dload | Upload | Total | Spent | Left |
| Speed | | | | | | | |
| 0 0 | 0 0 | 0 0 | 0 0 | 0 0 | --:--:-- | --:--:-- | --:--:-- |
| 0 0 | 0 0 | 0 0 | 0 0 | 0 0 | --:--:-- | --:--:-- | --:--:-- |
| 100 12.7M | 0 12.7M | 0 0 | 11.7M | 0 | 0:00:01 | --:--:-- | --:--:-- |
| 100 53.0M | 0 53.0M | 0 0 | 25.3M | 0 | 0:00:02 | --:--:-- | --:--:-- |
| 100 65.6M | 0 65.6M | 0 0 | 26.5M | 0 | 0:00:02 | --:--:-- | --:--:-- |
| - 26.5M | | | | | | | |

Figure 7.2: Downloading Debian package



If you're having difficulties running the preceding commands, please go to <https://www.tenable.com/downloads/nessus>, select the latest version of Nessus, and choose **Linux – Debian – amd64** to download the software package onto Kali Linux.

5. Next, install the Nessus software package onto Kali Linux:

```
kali@kali:~$ sudo dpkg -i Nessus-10.7.2-debian10_amd64.deb
```

The following screenshot shows the installation of Nessus:

```
kali@kali:~$ sudo dpkg -i Nessus-10.7.2-debian10_amd64.deb
Selecting previously unselected package nessus.
(Reading database ... 453732 files and directories currently installed.)
Preparing to unpack Nessus-10.7.2-debian10_amd64.deb ...
Unpacking nessus (10.7.2) ...
Setting up nessus (10.7.2) ...
HMAC : (Module_Integrity) : Pass
```

Figure 7.3: Installing the Nessus package

6. Next, use the following command to start and restart the Nessus service:

```
kali@kali:~$ sudo /bin/systemctl start nessusd.service
kali@kali:~$ sudo /bin/systemctl restart nessusd.service
```



The `systemctl status nessusd.service` command can be used to verify whether the Nessus service is active and running on Kali Linux.

7. To continue the Nessus setup process, open the web browser within Kali Linux and go to `https://kali:8834/`, as shown below:

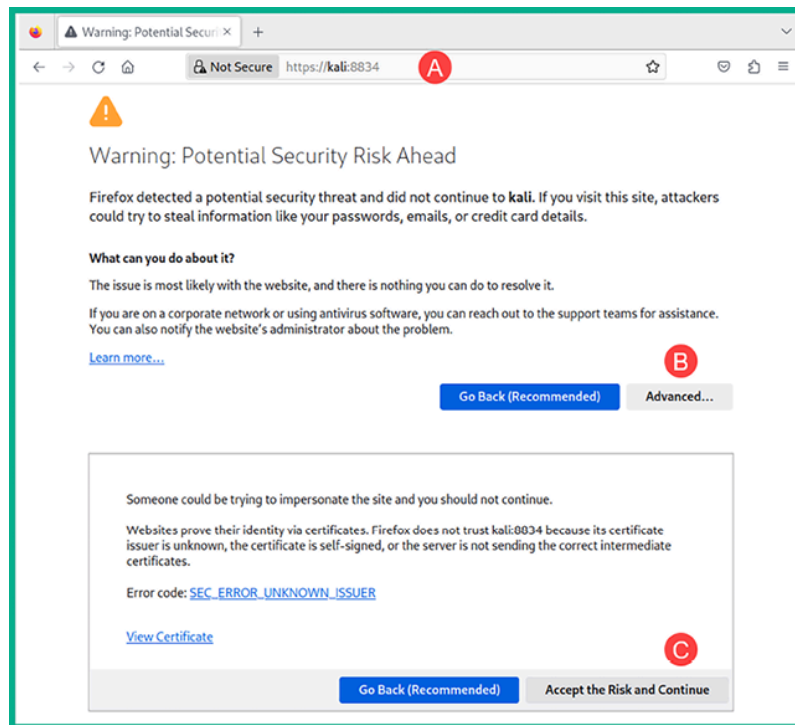


Figure 7.4: Firefox web browser warning

When you first visit `https://kali:8834/`, the web browser will provide a security warning because Nessus uses a self-signed digital certificate. Click on **Advanced**, then on **Accept the Risk and Continue**.

8. Next, the Nessus initialization page will appear. Click on **Continue**, as shown below:

Figure 7.5: Nessus setup welcome page

9. Next, select the **Register for Nessus Essentials** option and click on **Continue**, as shown below:

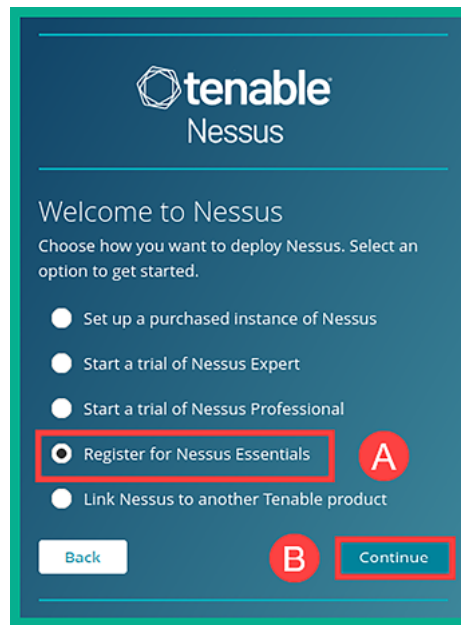


Figure 7.6: Selecting Nessus Essentials

10. Since you registered and received a Nessus Essentials activation code during *step 2*, click on **Skip** on the registration page, as shown below:

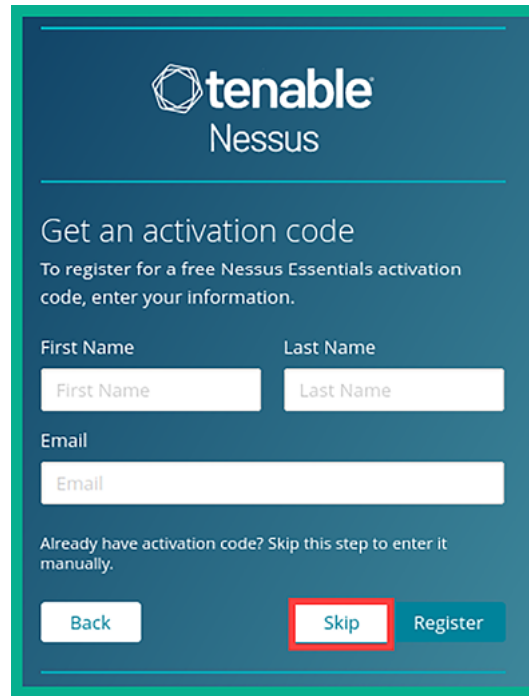
The image shows the Tenable Nessus registration page. At the top is the Tenable Nessus logo. Below it, the heading "Get an activation code" is followed by the text "To register for a free Nessus Essentials activation code, enter your information." There are three input fields: "First Name", "Last Name", and "Email". Below these fields is a link that says "Already have activation code? Skip this step to enter it manually." At the bottom, there are three buttons: "Back", "Skip", and "Register". The "Skip" button is highlighted with a red rectangular border.

Figure 7.7: Skipping registration

11. Next, enter the activation code from your email in the **Activation Code** field, then click on **Continue**, as shown below:



Figure 7.8: Entering activation code

12. Next, Nessus will show the activation code. Click on **Continue**, as shown below:

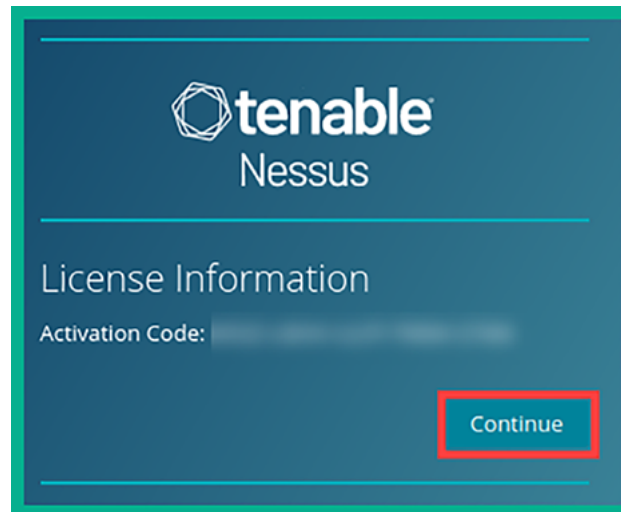


Figure 7.9: Verifying the activation code

13. Next, create a user account and click on **Submit**, as shown below:

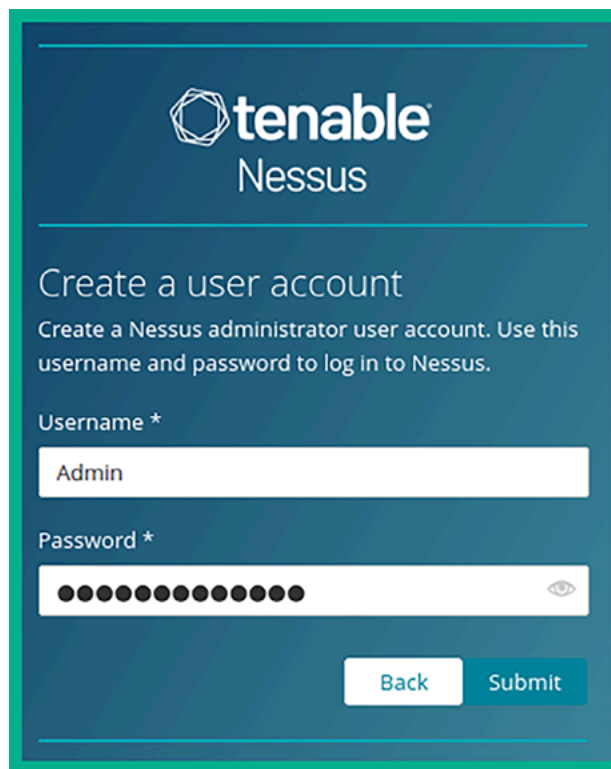
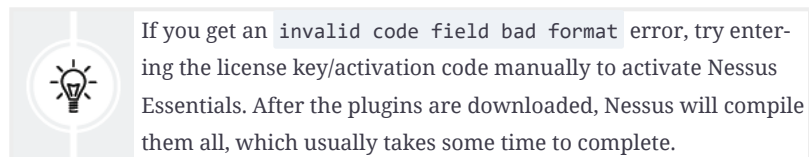
The screenshot shows the Tenable Nessus web interface for creating a user account. At the top, the Tenable Nessus logo is displayed. Below it, the heading 'Create a user account' is followed by instructions: 'Create a Nessus administrator user account. Use this username and password to log in to Nessus.' There are two input fields: 'Username *' with the text 'Admin' entered, and 'Password *' which is masked with dots and has an eye icon for toggling visibility. At the bottom right, there are two buttons: 'Back' and 'Submit'.

Figure 7.10: Creating an account



14. Nessus will automatically log in to the dashboard, then start the initialization process and begin downloading additional updates and plugins for the application. This process usually takes a few minutes to complete. To view the event logs, click on **Settings** | **About** | **Events**, as shown in the following screenshot:

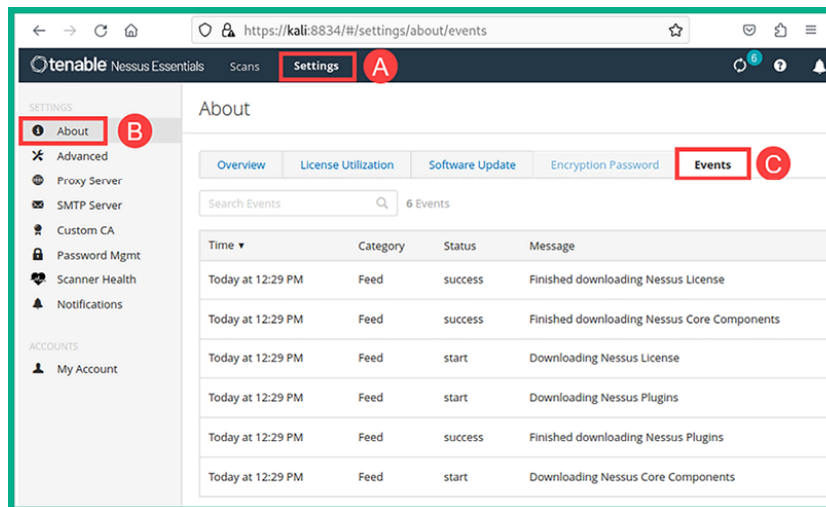


Figure 7.11: Verifying vulnerability database is up to date

- Once the download process is complete, Nessus will compile all the plugins.
Ensure this is completed before proceeding to scan a targeted system.

Part 2 – identifying vulnerabilities

Nessus can detect over 83,000 CVEs on targeted systems to help cybersecurity professionals such as ethical hackers and penetration testers to identify the attack surface of assets owned by organizations and use the collected information to provide recommendations on preventing and mitigating cyber-attacks and threats.

Use the following instructions to get started with scanning for security vulnerabilities using Nessus:

- Power on the **Metasploitable 3 (Windows version)** virtual machine as our targeted system on the network.
- On **Kali Linux**, log in to the Nessus Essentials dashboard at `https://kali:8834/` and click on **New Scan**, as shown below:

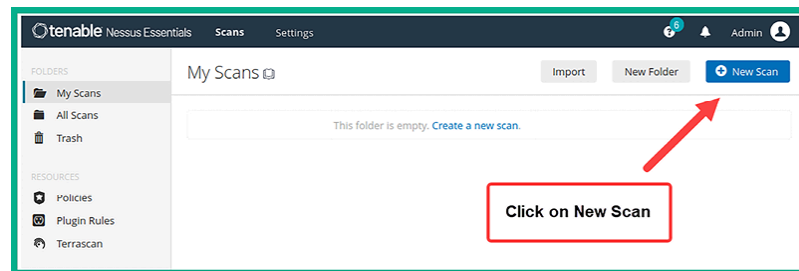


Figure 7.12: Selecting New Scan



3. Next, various vulnerability and compliance scanning templates will be presented, enabling you to easily choose the most suitable template for your scanning objectives. For instance, you can use a pre-defined template to detect whether targeted systems are vulnerable to WannaCry, ZeroLogon, PrintNightmare, and even Log4Shell. For our exercise, click on **Basic Network Scan**, as shown below:

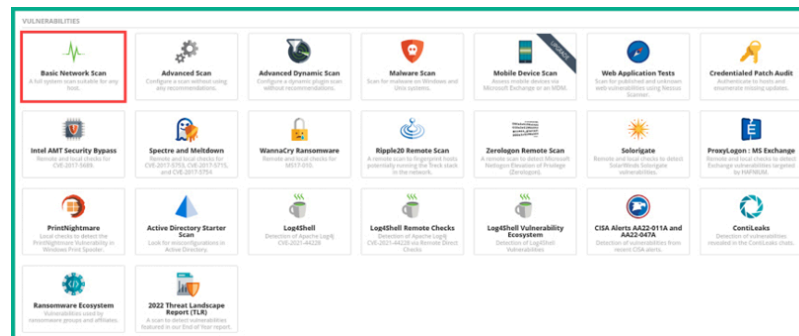
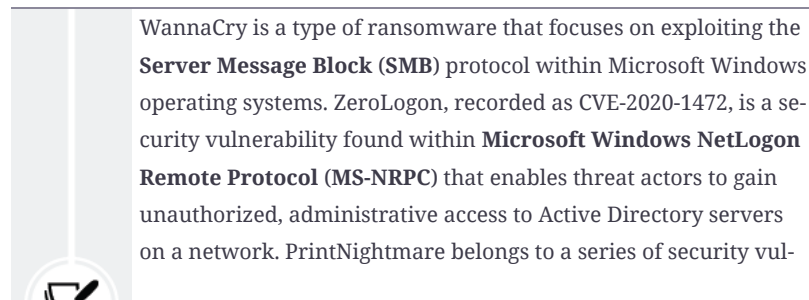
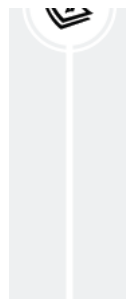


Figure 7.13: Selecting Basic Network Scan





nerabilities associated with the Windows Print Spooler services on Windows-based systems. This vulnerability enables a threat actor to execute arbitrary code remotely to gain system-level privileges on a targeted system. Log4Shell, recorded as CVE-2021-44228, is a security vulnerability found within the Apache Log4j library that enables threat actors to remotely execute arbitrary code.

- Next, the scan **Settings** page will appear. This page allows you to set a name, a description, a folder to easily organize your scans, and targets. Set a name, description, and the IP address of the Metasploitable 3 (Windows version) virtual machine as the target, then click on **Launch**, as shown below:

The screenshot shows the Nessus 'Settings' page for a scan. The 'Name' field is labeled 'Identifying Vulnerability on Target1' (A). The 'Description' field is labeled 'Basic vulnerability scan on Metasploitable 3 (Windows version) virtual machine.' (B). The 'Folder' dropdown is set to 'My Scans' (C). The 'Targets' text area contains the IP address '172.30.1.48' (D). At the bottom, there are 'Save', 'Cancel', and 'Launch' buttons (E).

Figure 7.14: Completing the scan details



To learn more about the various Nessus scanning templates, please visit <https://docs.tenable.com/nessus/Content/ScanAndPolicyTemplates.htm>.

As shown in the preceding screenshot, there are various options and sub-menus, such as the following:

1. The **Credentials** tab enables you to specify login credentials and allows Nessus to log in to the targeted system to retrieve specific information that's not easily available when performing a non-credential scan.
 2. **Scheduling** allows penetration testers to automate their scans over a period of time.
 3. **Notifications** allows Nessus to send email notifications when scans have started and completed.
 4. **DISCOVERY** specifies port scanning options.
 5. **ASSESSMENT** enables you to choose whether Nessus scans for web vulnerabilities.
 6. **REPORT** allows you to specify how Nessus handles the processing of information **that** will be shown in its report.
 7. **ADVANCED** enables you to specify how much traffic Nessus will send on the network, this is useful for low-bandwidth networks.
5. Next, Nessus will begin scanning the target and will display the progress within the **My Scans** summary window, as shown below:

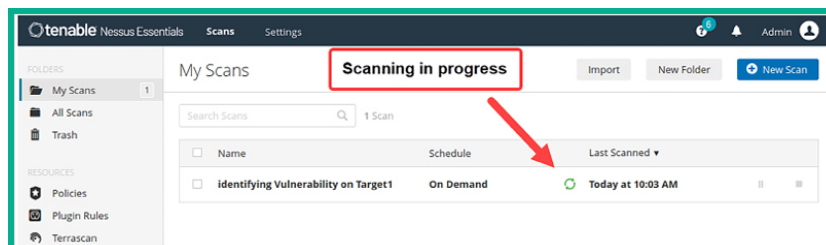


Figure 7.15: Checking scan progress

6. When the scan is complete, Nessus will automatically update the scan status, and the scan will be saved within the **My Scans** section, as shown below:

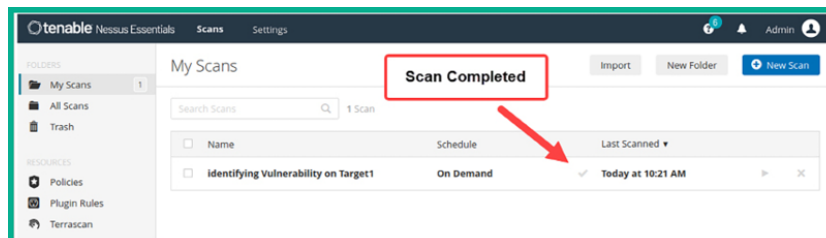


Figure 7.16: Scan completion

Part 3 – vulnerability analysis

Using vulnerability scanners such as Nessus can help us automate our process of vulnerability discovery and classification. As an aspiring ethical hacker and penetration tester, it's essential to understand how to perform vulnerability analysis on reported data. In addition, it's important to ensure that Nessus and any other vulnerability scanners have up-to-date vulnerability databases to ensure the scanner is able to identify the latest vulnerabilities, thus making the tool more effective for ethical hackers and penetration testers.

To get started with vulnerability analysis with Nessus, please use the following instructions:

1. To view the scan results, click on **My Scans** | **Identifying Vulnerability on Target1**, as shown below:

Figure 7.17: Selecting the completed scan

The following screenshot shows a summary of all the security vulnerabilities that were found on the targeted system:

Figure 7.18: Scan summary

As shown in the preceding screenshot, Nessus provides a very nice and easy-to-understand view of all the security vulnerabilities that were discovered. Both the column and doughnut charts provide an overview of how many security vulnerabilities were found based on their severity ratings and scores.

2. To view a list of all discovered security vulnerabilities, click on the **Vulnerabilities** tab, as shown below:

Figure 7.19: Listing of security vulnerabilities

As shown in the preceding screenshot, Nessus has grouped multiple security vulnerabilities together.

3. Next, click on the **CRITICAL** severity group to display all the security vulnerabilities that belong to this group, as shown below:

Figure 7.20: Viewing critical vulnerabilities

As shown in the preceding screenshot, Nessus has listed the security vulnerabilities in order of most to least severe. As a penetration tester, this is an indication of the security vulnerabilities that are most likely to have a large impact on the targeted system.

4. Next, click on any one of the critical vulnerabilities to view more details about it, as shown below:

Figure 7.21: Viewing vulnerability description

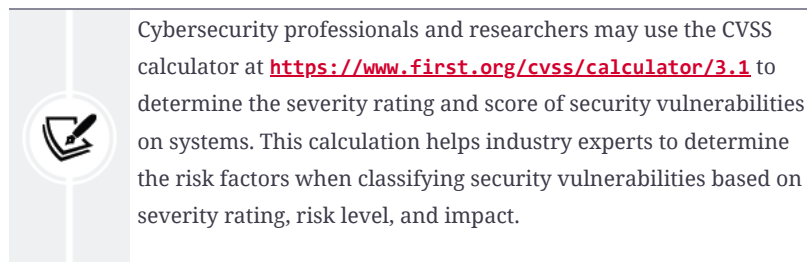
As shown in the preceding screenshot, Nessus provides a description to help cybersecurity professionals better understand the risk of having this security vulnerability on a system and its impact. In addition, Nessus also provides solutions to remediate this security vulnerability and provide the security posture of the targeted system or asset owned by the organization.

5. Furthermore, Nessus provides its **Vulnerability Priority Rating (VPR)** scoring system to help cybersecurity professionals prioritize their resources in resolving this security risk, as shown below:

Figure 7.22: Viewing the VPR key drivers

6. Additionally, Nessus provides the metrics that were used by the **Common Vulnerability Scoring System (CVSS)** to calculate the severity of the vulnerability, as shown below:

Figure 7.23: CVSS scoring



7. Next, let's take the CVSS 3.0 Vector and insert it into the calculator to determine how a threat actor would compromise a system with this vulnerability:

```
CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
```

8. Next, append the CVSS 3.0 Vector to the end of the following URL:

<https://www.first.org/cvss/calculator/3.0#>

The following is the final version of the URL, with the CVSS 3.0 Vector as the suffix:

<https://www.first.org/cvss/calculator/3.0#CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H>

9. Upon visiting the preceding URL, you will see how the vectors were allocated to determine the vulnerability score of **9.8**, as shown below:

Figure 7.24: CVSS metrics

As shown in the preceding screenshot, a threat actor will need to create an exploit that needs to be delivered across a **Network (N)** path with a **Low (L)** attack complexity, which requires **None (N)** privileges to be successful. Furthermore, **None (N)** human user interactions are needed, due to which the scope of the attack will remain **Unchanged (U)**. Once the exploit takes advantage of the security vulnerability in the targeted system, the impact on the confidentiality, integrity, and availability of the system will be **High (H)**.

Part 4 – exporting vulnerability reports

Generating a report from Nessus helps you quickly reference vulnerabilities and their descriptions after a penetration test. In this section, you will learn how to generate various types of reports using Nessus.

To complete this exercise, please use the following instructions:

1. On the Nessus dashboard, click on **Report** as shown below:

Figure 7.25: Finding the Report option

2. Next, a pop-up window will appear and provide you with various report generation options. Choose the **Report Format** and **Report Template** and then click on **Generate Report**, as shown below:

Figure 7.26: Generating a report

3. Once the report is generated, ensure you save it on your desktop and open it using a PDF reader, as shown below:

Figure 7.27: Viewing a report

4. Lastly, use the following commands to stop the Nessus service when you're no longer using the application:

```
kali@kali:~$ sudo /bin/systemctl stop nessusd.service
```

Having completed this section, you have learned how to use Nessus to perform a vulnerability assessment on a target during a penetration test. In the next section, you will learn how to identify security vulnerabilities using Nmap.

Vulnerability identification using Nmap

The **Nmap Scripting Engine (NSE)** is one of the most powerful features of Nmap. It enables penetration testers and security researchers to create, automate, and

perform customized scanning on targeted systems. When working with NSE, the scanning techniques are usually aggressive and have the potential to cause unexpected data loss or even crash the targeted system. However, NSE allows a penetration tester to easily identify security vulnerabilities and determine whether the target is exploitable.




If the organization is sensitive to disruption or includes **Operational Technology (OT)** assets, the penetration tester should get explicit written permission to run aggressive scripts.

There are 600+ pre-built scripts that belong to the following NSE categories:

- **Auth:** This category contains scripts that scan a targeted system to identify whether authentication bypass is possible.
- **Broadcast:** This category contains scripts that are used to discover host systems on a network.
- **Brute:** This category contains scripts that are used to perform some types of brute-force attacks on a remote server with the intention of gaining unauthorized access.
- **Default:** This category contains a set of default scripts within NSE for scanning.
- **Discovery:** This category contains scripts used in active reconnaissance to identify network services on a targeted system.
- **DoS:** This category contains scripts that simulate a **Denial-of-Service (DoS)** attack on a targeted system to check whether it's susceptible to such types of attack.
- **Exploit:** This category contains scripts that are used to actively exploit security vulnerabilities on a target.
- **External:** This category contains scripts that usually send data that's been gathered from a targeted system to an external resource for further processing.
- **Fuzzer:** This category contains scripts that are used to send random data into an application to discover any software bugs and vulnerabilities within applications.
- **Intrusive:** This category contains high-risk scripts that can crash systems and cause data loss.
- **Malware:** This category contains scripts that can determine whether a target is infected with malware.

- **Safe:** This category contains scripts that are not intrusive and are safe to use on a targeted system.
- **Version:** This category contains scripts used to gather the version information of services on a targeted system.
- **Vuln:** This category contains scripts used to check for specific vulnerabilities in a targeted system.



To learn more about NSE, please see <https://nmap.org/book/nse.html>. For a full list of NSE scripts, please see <https://nmap.org/nsedoc/scripts/>.

To get started working with NSE to identify security vulnerabilities, please use the following instructions:

1. Power on **Kali Linux** and **Metasploitable 2** virtual machines.
2. On **Kali Linux**, open the **Terminal** and use the following commands to view a list of locally available NSE scripts:

```
kali@kali:~$ ls -l /usr/share/nmap/scripts
```

The following screenshot shows there are 4,000+ NSE scripts within the `/usr/share/nmap/scripts` directory on Kali Linux:

Figure 7.28: Viewing NSE scripts

3. To filter all **File Transfer Protocol (FTP)** NSE scripts, use the following commands:

```
kali@kali:~$ ls -l /usr/share/nmap/scripts/ftp*
```

As shown in the following screenshot, the `*` works as a wildcard to show all scripts that begin with `ftp`:

Figure 7.29: Filtering FTP scripts

4. Next, let's use Nmap to determine whether the targeted system (Metasploitable 2) is running an FTP service and determine the service version:

```
kali@kali:~$ sudo nmap -sV -p 20,21 172.30.1.49
```

Figure 7.30: Performing scan using Nmap

As shown in the preceding screenshot, port 21 is open and the service is identified as `vsftpd 2.3.4` on the targeted system.

5. Next, let's use one of the NSE scripts to determine whether `vsftpd` is vulnerable on the target:

```
kali@kali:~$ sudo nmap --script ftp-vsftpd-backdoor 172.30.1.49
```

The `--script` command allows you to specify either a single script, multiple scripts, or a category of scripts. The following screenshot shows the results of performing a scan on our victim machine:

Figure 7.31: Identifying security vulnerability

As shown in the preceding screenshot, the `ftp-vsftpd-backdoor` script was used to check whether the target is vulnerable to a backdoor present within the `vsftpd 2.3.4` application. As a result, NSE indicated the targeted system is running a vulnerable service.



The `sudo nmap --script-updatedb` command can be used within Kali Linux to ensure the NSE scripts are up to date.

6. Now that a vulnerability has been found, the next step is to determine whether there are exploits that can leverage this security weakness. The following screenshot shows the results of performing a Google search for known exploits for the **VSFTPD 2.3.4** service:

Figure 7.32: Researching the vulnerability on the internet

As shown in the preceding screenshot, there's a link for an exploit from Rapid7, the creator of Metasploit. Using this Rapid7 URL, you can gather further details on how to exploit the vulnerability using Metasploit on Kali Linux. Additionally, notice the second URL within the Google search result, which is from Exploit-DB. This is a trusted exploit database that is maintained by the creators of Kali Linux and Offensive Security. These are two trusted online resources for gathering exploits during a penetration test.

7. Additionally, within Kali Linux, there is a tool known as **searchsploit** that allows you to perform a query/search for exploits within the offline version of Exploit-DB on Kali Linux.

The following screenshot shows the search results when using the `searchsploit` command:

Figure 7.33: Using searchsploit

As shown in the preceding screenshot, `searchsploit` was able to identify multiple exploits from the local, offline version of the Exploit-DB database. Notice that there is a particular entry that indicates there's already an exploit module within Metasploit.

The following screenshot shows the `vsFTPD exploit` module within Metasploit:

Figure 7.34: Using Metasploit to find the exploit module

As shown in the preceding screenshot, this exploit module can take advantage of security vulnerabilities that are found within any Linux-based system, which is running vsFTPD version 2.3.4. If the exploit is successful, the penetration tester will be able to create a backdoor with **Remote Code Execution (RCE)** on the targeted system.



Many vulnerability scripts can be used within Nmap as part of NSE. Please be sure to check out the complete list at <https://nmap.org/nsedoc/categories/vuln.html>, where you will be able to identify the names and details of each script that can be found within the vulnerability category.

8. If you want to execute an entire category of scripts, you can use the `nmap --script <category-name>` command, as shown here:

```
kali@kali:~$ sudo nmap --script vuln 172.30.1.49
```

When using the `vuln` category, NSE will use all the vulnerability detection scripts to check for security weaknesses on the target. As shown in the following screenshot, additional security flaws were discovered on the Metasploitable 2 victim machine:

Figure 7.35: Identifying security vulnerabilities

As an aspiring ethical hacker and penetration tester, you have learned how to perform various scanning techniques to fingerprint and discover security vulnerabilities on host systems within a network using Nmap. Fingerprinting involves port scanning, banner grabbing of network services, packet analysis, analyzing HTTP responses, and identifying the operating system on the targeted system. Using the information found within this section can help you in researching exploits and payloads that can take advantage of these security vulnerabilities.

In the next section, you will learn how to install and use an open source vulnerability management tool on Kali Linux.

Working with Greenbone Vulnerability Manager

The **Open Vulnerability Assessment Scanner (OpenVAS)** tool is a free vulnerability scanner that allows both ethical hackers and penetration testers to perform a vulnerability assessment on a network. OpenVAS can scan both authenticated and unauthenticated vulnerability assets within an organization.

When using an authenticated scan, the penetration tester provides valid login credentials to the vulnerability scanner, which allows it to authenticate to a system to provide a thorough scan for any misconfigurations on the target system's settings. However, the unauthenticated scan is usually not as thorough since it looks for any security vulnerabilities on the surface of the target and provides a report.



Authenticated scans, by using valid login credentials, can perform checks against internal files, configurations, and more detailed system information, thereby identifying vulnerabilities that unauthenticated scans cannot detect due to their lack of permissions.

Greenbone Vulnerability Manager (GVM) is a centralized management tool that manages the functions and vulnerabilities of OpenVAS. OpenVAS is the engine for the actual vulnerability scanning, whereas GVM serves as the framework that includes OpenVAS for vulnerability management. It's worth noting that GVM was formerly known as OpenVAS before restructuring.

Part 1 – installing GVM

In this exercise, you will learn how to set up GVM on Kali Linux and perform a vulnerability assessment on a target using OpenVAS. To get started with this exercise, please use the following instructions.

1. Power on the **Kali Linux** virtual machine and ensure it has internet connectivity.
2. On **Kali Linux**, open the **Terminal** and use the following commands to update the local software package repository list file and install the GVM package:

```
kali@kali:~$ sudo apt update  
kali@kali:~$ sudo apt install gvm
```

3. During the installation, you may be prompted to restart various services. Ensure you use the *spacebar* on your keyboard to select the services to be restarted, then use the *Tab* key to move between options and hit *Enter* on **Ok**, as shown below:

Figure 7.36: Restarting the network manager prompt

4. Once the installation is complete, reboot the Kali Linux virtual machine and log in to continue.
5. Next, use the following commands to initialize the setup process and generate default user credentials:


```
kali@kali:~$ sudo gvm-setup
```

The setup process usually takes a while to complete as it downloads additional updates and plugins. Once the setup process is completed, the default admin account is created with a randomized password, as shown below:

Figure 7.37: User account created

6. Next, use the `sudo gvm-check-setup` command to verify that GVM is set up correctly.
7. Next, open the web browser within Kali Linux and go to `https://127.0.0.1:9392` to access the web interface for GVM.
8. Use the default `admin` user account that was created at the end of the setup process and log in, as shown below:

Figure 7.38: Login page

9. After logging in, click on **Administration** | **Feed Status** as shown below:

Figure 7.39: Locating the Feed Status menu

GVM will continue to download additional **Cyber Threat Intelligence (CTI)** from multiple trusted online sources to ensure the vulnerability scanning engine within GVM has the latest updates and signatures to identify the latest security flaws on the system, as shown below:

Figure 7.40: Checking the vulnerability feeds

The download process usually takes a long time to complete. Once all content is updated, the feed status will automatically change, as shown below:

Figure 7.41: Verifying vulnerability feeds updated

Ensure all threat feeds are updated before performing any vulnerability scans on targeted systems.

Part 2 – vulnerability identification

To use GVM to identify security vulnerabilities on a targeted system, please use the following instructions:

1. On the GVM dashboard, click on **Configurations** | **Targets** to set our target host, as shown below:

Figure 7.42: Locating the Targets option

2. Next, click on the **New Target** icon that's located in the top-left corner.
3. In the **New Target** window, ensure you set a **Name** and **Hosts** (IP address of Metasploitable 3) and click on **Save**, as shown below:

Figure 7.43: Setting a target details

4. As shown in the preceding screenshot, the **New Target** window provides additional options such as entering user credentials to perform credential scanning to obtain more information. Furthermore, you can specify multiple targeted systems from a list and exclude specific targets if you're scanning a range of addresses.
5. Next, create a new scan task by clicking on **Scans** | **Tasks**, as shown below:

Figure 7.44: Locating the Tasks menu

6. Next, click on the **Magic Paper** icon (top-left corner), then **New Task**, as shown below:

Figure 7.45: Create a new task

7. On the **New Task** window, enter the name of the task and select **Scan Targets** from the drop-down menu, then click on **Save**, as shown below:

Figure 7.45: New Task window

8. Next, the new scan task will appear in the lower section of the same page. Click on the **Play** icon to start the scan on the targeted system, as shown below:

Figure 7.47: Starting a task

9. The task status will change automatically during this process. Once the task is complete, the status will change to **Done** and display its results, as shown below:

Figure 7.48: Task complete view

Part 3 – vulnerability analysis and reporting

To perform vulnerability analysis using GVM, please use the following instructions:

1. To view the results of the report, click on **Scans | Reports**, as shown below:

Figure 7.49: Vulnerabilities shown in graph format

As shown in the preceding screenshot, GVM analyzed and categorized the discovered security vulnerabilities into **High**, **Medium**, **Low**, **Log**, and **False**

Positives. This categorization aids cybersecurity professionals by providing a clear prioritization framework.

High-severity vulnerabilities, which pose immediate and serious threats, are prioritized, whereas medium and low severities indicate lower risks. Log entries, typically informational, do not necessarily indicate security weaknesses, while false positives are incorrectly identified vulnerabilities.

This structured approach supports efficient decision-making and resource allocation, focusing efforts on mitigating critical vulnerabilities first.

Additionally, the dynamic nature of vulnerabilities and the ability to integrate GVM with other security tools underscore its essential role in maintaining a robust cybersecurity posture.

2. To view a detailed list of identified security vulnerabilities, click on the report date, as shown below:

Figure 7.50: Selecting the vulnerability report

Then, click on the **Results** tab to view a list of all security vulnerabilities and their severity levels that were found on the targeted system, as shown below:

Figure 7.51: List of security vulnerabilities

3. To view the description of a vulnerability, click on any one from the results list, as shown below:

Figure 7.52: Vulnerability results

Using the information shown in the preceding screenshot, ethical hackers and penetration testers will gain better insights into the impact a vulnerability has on a system if it's exploited by an adversary. In addition, ethical hackers can use this information to develop or acquire exploits to compromise multiple systems with the same security flaw on the targeted network.

As shown in the preceding screenshot, the following is a breakdown of the results:

- **Summary:** The **Summary** section provides the user with an overview of the results.
- **Detection results:** The detection results provide specific information about each security vulnerability found during the scan.
- **Product detection result:** This section provides specific information about the applications and services that were found during the scan of the targeted system.
- **Insight:** The **Insight** section provides additional details on the detected security vulnerability.
- **Detection method:** The detection method contains the method used by Nessus to identify the security vulnerability in the targeted system.
- **Affected software/OS:** This section specifies the list of software, applications, and operation systems that are affected by the security vulnerability.

4. Lastly, use the `sudo systemctl stop gvmd` command to stop GVM.

In this section, you have learned how to set up and work with GVM to identify security vulnerabilities in a targeted system. In the next section, you will learn how to use common tools to identify security flaws in web applications.

Using web application scanners

As an aspiring penetration tester, you will also be required to perform web application security testing based on the scope of your penetration testing engagements. Web application security testing aims to identify vulnerabilities that could be exploited by attackers, such as SQL injection, **cross-site scripting (XSS)**, and security misconfigurations. In this section, you will learn how to use various types of web application scanners to identify and fingerprint web applications on a target server.

Let's get started!

WhatWeb

WhatWeb enables ethical hackers and penetration testers to identify and fingerprint the type of technologies that are running on web application servers.

WhatWeb is pre-installed on Kali Linux and should be part of your arsenal of tools during your reconnaissance and vulnerability assessment phase.

To profile a targeted web server using WhatWeb, please use the following instructions:

1. Firstly, power on the **Kali Linux** and **Metasploitable 3** (Windows version) virtual machines.
2. On **Kali Linux**, open the **Terminal** and use the following commands to identify whether there's a web application running on the target:

```
kali@kali:~$ nmap -p 80,443,8080 172.30.1.48
```

As shown in the following screenshot, web services were found on port **80** and **8080**:

Figure 7.53: Running an Nmap scan



3. Next, use the following commands to profile the web server:

```
kali@kali:~$ whatweb http://172.30.1.48
```

As shown in the following screenshot, WhatWeb was able to identify the web application and additional web technologies on the targeted system:

Figure 7.54: WhatWeb output

As an aspiring ethical hacker and penetration tester, some tools will help you gather information about the web server, while others will discover security vulnerabilities. It's important to research all the technologies that are found on a targeted web server when using WhatWeb; many security researchers share their findings and disclosure vulnerabilities to help others fight the battle against cyber criminals. WhatWeb is a tool designed for web fingerprinting, which helps in identifying the components that make up a web server.

To put it simply, WhatWeb provides the following details:

- The web application and its versions
- The web technologies and their versions
- The host operating system and its versions

By researching the version numbers of each technology, you will be able to find exploits that could take advantage of the vulnerabilities in the targeted system. In the next section, you will learn how to use Nmap to discover web application vulnerabilities.

Nmap

As you have learned, Nmap has a lot of very cool features and enables penetration testers to perform various types of scanning on targeted systems to discover specific details about them. Within NSE, many scripts are already pre-loaded onto Kali Linux.

Using the following command, you will be able to see an entire list of all the Nmap scripts that begin with `http`:

```
kali@kali:~$ ls /usr/share/nmap/scripts/http*
```

From the list, you can choose to use a particular script to check for HTTP vulnerabilities on a targeted system. Let's imagine you want to identify whether a web application is vulnerable to **Structured Query Language (SQL)** Injection attacks. The `http-sql-injection` NSE script will be able to identify such security flaws.

The following Nmap command shows how to invoke the SQL Injection script and perform a scan on a target that has port `80` open for web services:

```
kali@kali:~$ nmap --script http-sql-injection -p 80 172.30.1.49
```

The following screenshot shows Nmap was able to identify possible SQL Injection at multiple points on the target:

Figure 7.55: Nmap scan results

As shown in the preceding screenshot, the Nmap script was able to automate the process of checking whether various URLs and paths are susceptible to a possible SQL Injection attack.



NSE is a powerful feature of Nmap that allows users to write scripts to automate a wide range of networking tasks, including vulnerability detection, exploitation, and network discovery. While many NSE scripts can be leveraged to identify security vulnerabilities in web applications, it's important to always identify the service version of the web application by simply using the `-A` or `-sV` syntax when performing an initial scan to profile your target.

Once you have identified the web application's service version, use the internet to research known vulnerabilities. As a penetration tester, it's always good to perform additional research on vulnerabilities as you may find more information on how to compromise the target.

Be sure to perform additional scanning on the target to discover any hidden security vulnerabilities, and use the information found at <https://nmap.org/nsedoc/> to gain an in-depth understanding of the purpose of various NSE scripts. In the next section, you will learn how to use Nikto to check for web application vulnerabilities on a target.

Nikto

Nikto is an open source web application scanner that comes pre-installed within Kali Linux. This tool allows penetration testers to easily automate the process of identifying security vulnerabilities that may exist within a web application on a web server. To put it simply, Nikto is designed to test various types of web servers and web applications for outdated server software, potentially dangerous files/scripts, and default files and programs.

To get started using Nikto, please use the following instructions:

1. Power on the **Kali Linux** and **Metasploitable 2** virtual machines.
2. On **Kali Linux**, open the **Terminal** and use the following commands to scan the web application on **Metasploitable 2**:


```
kali@kali:~$ nikto -h 172.30.1.49
```



Using the `-h` syntax allows you to specify the target's hostname or IP address. To learn more about various scanning options, use the `nikto --help` command.

The following screenshot shows some of the scan results from our target system:

Figure 7.56: Nikto scan results

As shown in the preceding screenshot, Nikto can identify various security vulnerabilities within the target web application. They are listed in bullet format, and the `+` icon is used to indicate a new result.

Take some time to read each line thoroughly as Nikto helps security professionals understand the details of the security vulnerabilities. It also provides references to where the flaws were found and how to resolve those weaknesses. Next, you will learn how to identify web application vulnerabilities using Metasploit.

Metasploit

In this section, you will learn how to leverage the power of Metasploit to discover security vulnerabilities on a web application server. For our target, we'll be using the Metasploitable 2 virtual machine. To get started with this exercise, please use the following instructions:

1. Firstly, power on both the **Kali Linux** and **Metasploitable 2** virtual machines.
2. On **Kali Linux**, open the **Terminal** and use the following command to start the PostgreSQL database and initialize Metasploit:

```
kali@kali:~$ sudo service postgresql start  
kali@kali:~$ sudo msfdb init
```

3. Next, use the following commands to access the Metasploit framework:

```
kali@kali:~$ msfconsole
```

4. Then, use the following command to load the WMAP web vulnerability scanner module within Metasploit:

```
msf6 > load wmap
```

The following screenshot shows the execution of the preceding commands and the WMAP plugin loaded successfully:

Figure 7.57: Loading WMAP

5. Next, use the following commands to set the targeted system as Metasploitable 2:

```
msf6 > wmap_sites -a http://172.30.1.49
```

The following screenshot shows how to set the targeted host within the WMAP web vulnerability scanner:

Figure 7.58: Setting a target in WMAP

6. Next, use the following commands to specify the URL of the targeted web application. We'll be targeting the Mutillidae web application within the Metasploitable 2 virtual machine:

```
msf6 > wmap_targets -t http://172.30.1.49/mutillidae/index.php
```

The following screenshot shows the expected results once the target has been set:

Figure 7.59: Setting a URL

As shown in the preceding screenshot, the target web application has been set to Mutillidae within the host system.

7. Next, use the following commands to automatically load various web scanning modules from Metasploit for security testing:

```
msf6 > wmap_run -t
```

The following screenshot shows many Metasploit web scanning modules that are being loaded into the WMAP web vulnerability scanner:

Figure 7.60: Launching the scan

8. Once the web scanning modules have been loaded, use the following commands to perform web security testing on the target web application:

```
msf6 > wmap_run -e
```

9. When the WMAP scan is complete, use the following command to view a list of web security vulnerabilities that have been discovered by the WMAP web scanner within Metasploit:

```
msf6 > wmap_vulns -l
```

10. Lastly, use the `vulns` command to see the overall results of the security assessment from WMAP:

```
msf6 > vulns
```



If Metasploit is able to identify vulnerabilities based on their CVE IDs, it will be shown with the `vulns` command.

Having completed this exercise, you have learned how to use Metasploit to identify web application vulnerabilities. Next, you will learn how to perform a vulnerability scan on a target WordPress web application using WPScan.

WPScan

While there are many web applications within the e-commerce industry, there are many organizations that deploy the WordPress web application as their preferred **Content Management System (CMS)**. While WordPress provides a very stylish and clean presentation of websites, many organizations do not always update their WordPress platforms and plugins, thereby leaving their web server and web application vulnerable to potential cyber-attacks from threat actors on the internet.



WordPress is one of the most popular CMSs, used not only in the e-commerce sector but across a wide range of industries due to its flexibility, extensibility, and ease of use.

Within Kali Linux, you will learn about the WPScan tool, which allows penetration testers to perform vulnerability scanning and enumeration on the WordPress web application on a target server.

To get started with this exercise, please use the following instructions:

1. Firstly, power on both **Kali Linux** and **Metasploitable 3** (Windows version) virtual machines.
2. On **Kali Linux**, open the **Terminal** and use the following commands to update the WPScan database:

```
kali@kali:~$ wpscan --update
```

3. Next, use the following commands to identify security vulnerabilities on the WordPress web application on the Metasploitable 3 (Windows version) virtual machine:

```
kali@kali:~$ wpscan --url http://172.30.1.48:8585/wordpress --no-update
```

The following screenshot shows the vulnerability scan's results:

Figure 7.61: WPScan results

As shown in the preceding screenshot, WPScan will check each component of the WordPress installation and configuration on the remote target and provide details of its findings.

4. Next, use the `-e u` commands to enumerate the username(s) for any logon accounts on the targeted WordPress web application, as shown below:

```
kali@kali:~$ wpscan --url http://172.30.1.48:8585/wordpress --no-update -e u
```

As shown in the following screenshot, WPScan was able to identify the login usernames of the targeted web server:

Figure 7.62: Usernames found

As you have seen, it's quite simple to perform a vulnerability scan on a WordPress server and gather a list of potentially authorized usernames on the target server.



To learn more about WPScan, please see

<https://www.kali.org/tools/wpscan/>.

Having completed this section, you have learned how to perform web scanning using various tools and techniques within Kali Linux. Having gathered a list of web application security vulnerabilities, with some additional research, you will be able to find working exploits to test whether these vulnerabilities are truly exploitable.

Summary

In this chapter, you have learned about the importance of discovering security vulnerabilities within an organization and its assets. You also gained hands-on experience and skills with using various tools, such as Nessus, Nmap, and GVM, to perform security assessments on systems. You also discovered how various tools, such as WhatWeb, Nikto, and WPScan, can be used to easily identify security flaws in web applications.

I trust that the knowledge presented in this chapter has provided you with valuable insights, supporting your path toward becoming an ethical hacker and pene-

tration tester in the dynamic field of cybersecurity. May this newfound understanding empower you on your journey, allowing you to navigate the industry with confidence and make a significant impact. In the next chapter, *Understanding Network Penetration Testing*, you will focus on how to use various techniques and strategies when performing network penetration testing.

Further reading

- Understanding Nessus: <https://www.techtarget.com/searchnetworking/definition/Nessus>
- Nmap Scripting Engine (NSE): <https://nmap.org/book/man-nse.html>
- Nmap NSE scripts: <https://nmap.org/nsedoc/scripts/>
- CVSS scoring system: <https://www.first.org/cvss/>

Join our community on Discord

Join our community's Discord space for discussions with the author and other readers:

<https://packt.link/SecNet>

