**Nmap User Training Guide**

**Group Members:**

Scott Duff

Table of Contents

3. **Introduction**

**Module 1**

**Scanning Basics**

4. Set Up and Configurations

**5.** VM Startup and Commands

**Single Host Scan**

6. IP Address

7. Hostname

8. URL

**Multi Host Scan**

9. IP Address

10. Hostname/URL

Range Scan

11. IP Address

**Port Scans**

12. Important Note

13. Single Port Scan

14. Multiple Port Scan

15. Port Range Scan

16. Scan Results

17. Common Ports

**Export Scan Results**

18. Information

19. Exporting Scan Results

**Introduction**

Nmap (“Network Mapper”) is an open-source tool for network exploration and security auditing. It was designed to rapidly scan large networks. (nmap manual pages, nmap 7.93). In addition to being able to scan a range of targets to be scanned a user can control the depth of each scan. Light scans might determine which ports are open and closed by firewall settings. A more detailed scan might tell what kind of devices are using the ports, the operating system they are running, and the active services on them.

The core reason why nmap was created was for host discovery. Its purpose was to identify the types of devices actively using scanned ports. These included servers, routers, switches and other devices (Breeden II, 2022). Nmap is useful for seeing how these devices are hooked together to form a network map.

**Port Rules Discovery**: Nmap can tell if a port is open or closed by a firewall. Nmap can be used by professionals to see if their policies have the desired effect and if the firewalls are working correctly (Breeden II, 2022).

**Shadow IT Hunting:** Can be used to identify things that shouldn’t be there at all. If, for example, someone has placed an Xbox on the network, not only could it drain the network, it could be a soft spot for an attack. Nmap would be especially helpful in detecting such a rogue device on the network (Breeden II, 2022).

**Operating System Detection**: Can provide the types of operating systems running on discovered devices, also known as OS fingerprinting.

**Service Discovery**: Nmap can be used to discover what role discovered devices are playing. For example, nmap can see if they are acting as a mail server, a web server, a database repository or a storage device (Breeden II, 2022).

**Vulnerability Scanning**: Nmap can be used to identify vulnerabilities to specific threats.

**Zenmap** is a free open-source GUI for Nmap. It is available on many operating systems such as Linux, Windows, and Mac. Zenmap is not meant to replace Nmap, but rather to make it more useful. Zenmap will also be discussed in this manual. Now, to discuss scanning basics.

**Scanning Basics**

**Set Up and Configurations**

Table 1. VM Operating System Networking Information

|  |  |  |
| --- | --- | --- |
| Operating System | IP Address | Hostname |
| Kali Linux | 192.168.32.128 | kali.local |
| Ubuntu Linux | 192.168.32.129 | ubuntu.local |
| Debian Linux | 192.168.32.130 | debian.local |

Table 2. NMAP Version

|  |  |
| --- | --- |
| Operating System | Version |
| Kali Linux | nmap 7.93 |

Table 1 lists the configurations used in this portion of the guide.

Table 2 shows the version of nmap used in this section of the guide.

**Scanning Basics**

**VM Startup and Commands**

1. Start your Kali Linux VM, log in as a regular user and open the terminal, Figure 1.

The example in Figure 1, shows that nmap has scanned a host and detected that all 1000 ports are in an ignored state. This may seem like an error or that nmap has failed to scan the target, but modern Linux distributions are secure by design. Perhaps the hosts on your network have disabled ports or services?

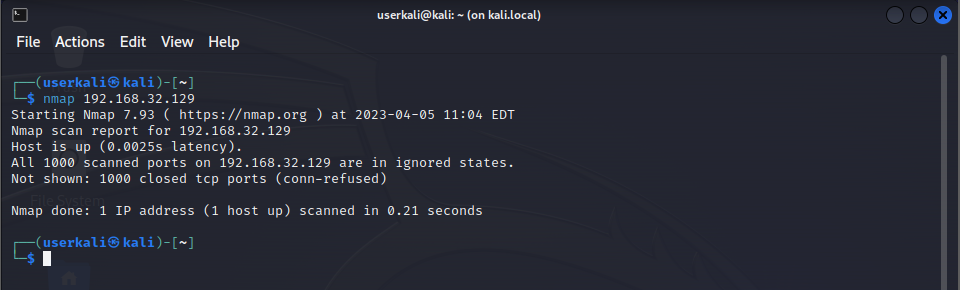


Figure 1. Example output from scanning a secure host.

To prepare for the following tutorial we are going to start some services. This will allow us to generate some useful output in our NMAP scans.

2. Start your Ubuntu Linux VM, log in as a regular user or as root.

3. On your Ubuntu Linux VM, type the following command (\*):

**service apache2 start**

4. Start your Debian Linux VM, log in as a regular user or as root.

5. On your Debian Linux VM, run the following command (\*):

**systemctl start vsftpd**

*(\*) Not all Linux distributions use the same set of commands to start and stop services. The commands you use could vary. Your goal is to start some services on your own Linux VM installations.*

**Single Host Scan**

**IP Address Scan**

Let’s start scanning.

1. On your Kali Linux VM, select the terminal window, as seen in Figure 1.

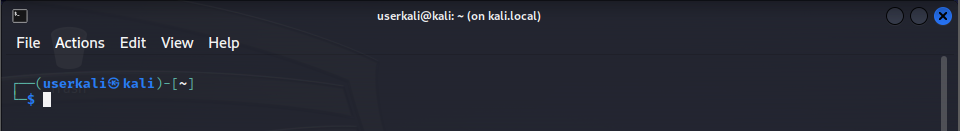


Figure 1. Open Kali Linux terminal, logged in as a regular user.

2. On your Kali Linux VM, in the terminal, type the following command (\*):

**nmap 192.168.32.129**

The output in Figure 2, shows that the host at 192.168.32.129 has an open port of 80/tcp is open and that the service is HTTP.

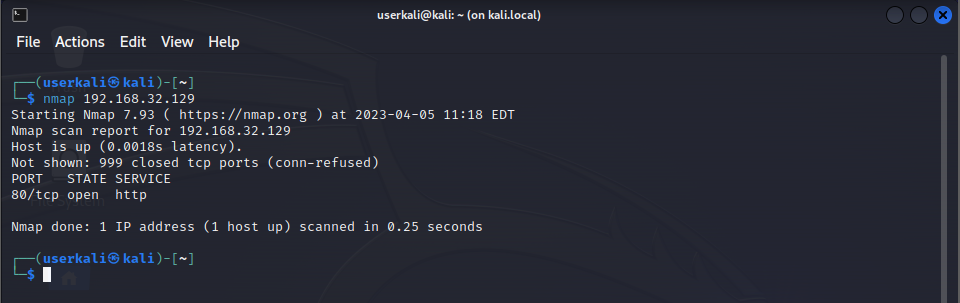


Figure 2. Kali Linux VM, output from Single Host Scan - IP Address Scan - Step 2.

*(\*) Do not scan IP addresses that you do not have permission to scan.*

**Single Host Scan**

**Hostname Scan**

Let’s start scanning by hostname.

1. On your Kali Linux VM, type the following command (\*) (\*\*) (\*\*\*):

**nmap debian.local**

The output in Figure 1, shows that the host with the hostname debian.local has an open port of 21/tcp - FTP.

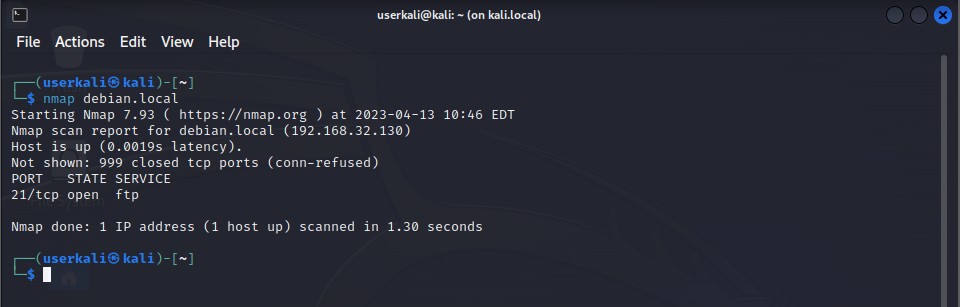


Figure 1. Kali Linux VM output from Single Host Scan Hostname, Step 1.

*(\*) Not all Linux distributions use the same commands to configure or give the output that includes your computer’s hostname. Your goal is to configure and/or find your computer’s hostname.*

*(\*\*) Depending on your home network and what services you have running, there may be no name resolution available. So NMAP will not be able to resolve the IP address of the hostname.*

*(\*\*\*) This is an important skill to remember. The IP Address is not the only method to scan a host. This is included in various IT certification exams.*

**Single Host Scan**

**URL Scan**

**WARNING WARNING WARNING**

Do not scan URLs that you do not have permission to scan. This can be seen as aggressive action. You could be treated as an illegal hacker.

**WARNING WARNING WARNING**

Let’s start scanning by URL.

1. On your Kali Linux VM, type the following command (\*):

**nmap scanme.nmap.org**

The output in Figure 1, shows that the host at scanme.nmap.org has an open port of 22/tcp, Port 80/tcp, and Port 31337/tcp and those services are SSH, HTTP, and Elite.

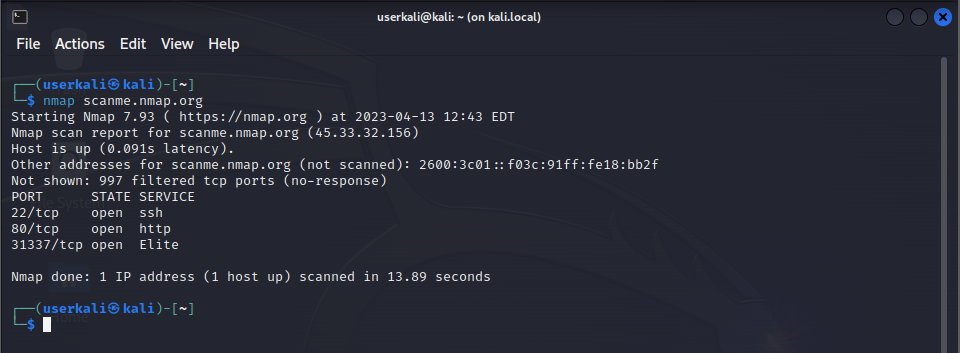
****

Figure 1. Kali Linux VM, output from Single Host Scan - URL Scan, Step 1.

*(\*) NMAP’s creator has generously provided a scannable URL for learning how to use NMAP.*

**Multi Host Scans**

**IP Address Scans**

Let’s start scanning multiple IP addresses.

1. On your Kali Linux VM, run the following command (\*):

**nmap 192.168.32.129 192.168.32.130**

The output in Figure 1, shows that the host at IP address 192.168.32.129 has Port 80/tcp - HTTP is open and that the host at IP address 192.168.32.130 has Port 21/tcp - FTP is open.

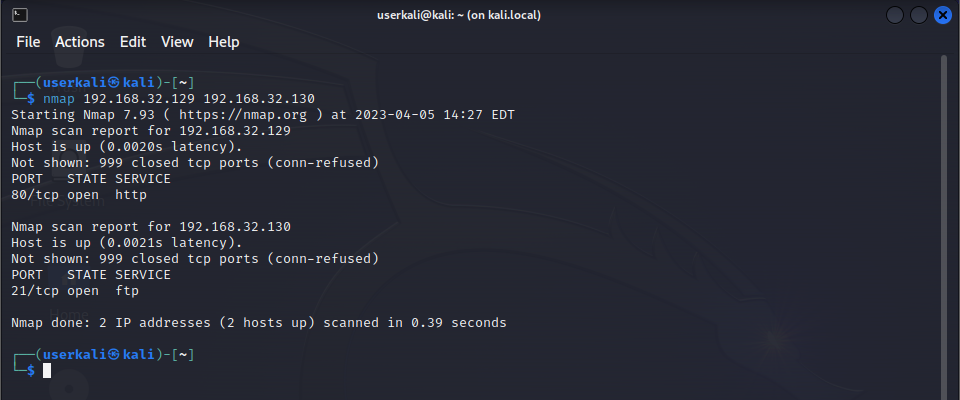


Figure 1. Kali Linux VM, output from Multi Host Scans - IP Address Scans, Step 1.

*(\*) By listing each IP address that you want to scan you can reduce the overhead of manual typing each command once for each IP address. This is a silly example. There really isn’t much overhead. What this really does is it allows for easier automation. Passing one command to the system and returning multiple results.*

**Multi Host Scans**

**Hostname and URL/Hostnames Address Scans**

Let’s start scanning multiple IP addresses.

1. On your Kali Linux VM, run the following command (\*):

**nmap scanme.nmap.org debian.local**

The output in Figure 1, shows that the host at URL scanme.nmap.org has open ports on Port 22/tcp - SSH, Port 80/tcp - HTTP, and Port 31337/tcp - Elite and the host at hostname debian.local has an open port at Port 21/tcp - FTP.

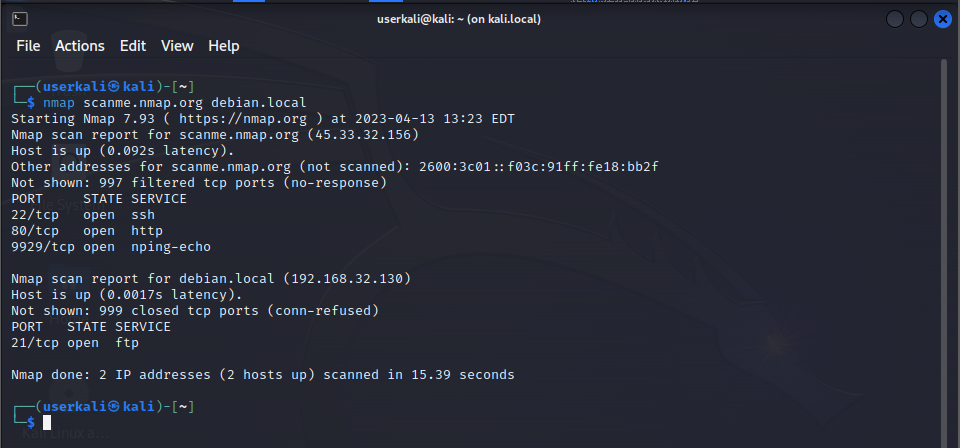


Figure 1. Kali Linux VM, output from Multi Host Scans - Hostname and URL/Hostname Address Scans, Step 1.

*(\*) Hostname and URL scanning can be combined when scanning multiple hosts. As in the Multi Host Scan IP Address Scan module this allows for easier automation in your scanning procedures.*

**Range Scans**

**IP Address Scans**

Let’s start scanning ranges of IP addresses.

1. On your Kali Linux VM, run the following command (\*):

**nmap 192.168.32.128-130**

The output in Figure 1, shows that the host at IP address 192.168.32.128 has no open ports, the host at IP address 192.168.32.129 has an open port of 80/tcp - HTTP, and the host at IP address 192.168.32.30 has an open port of 21/tcp - FTP.

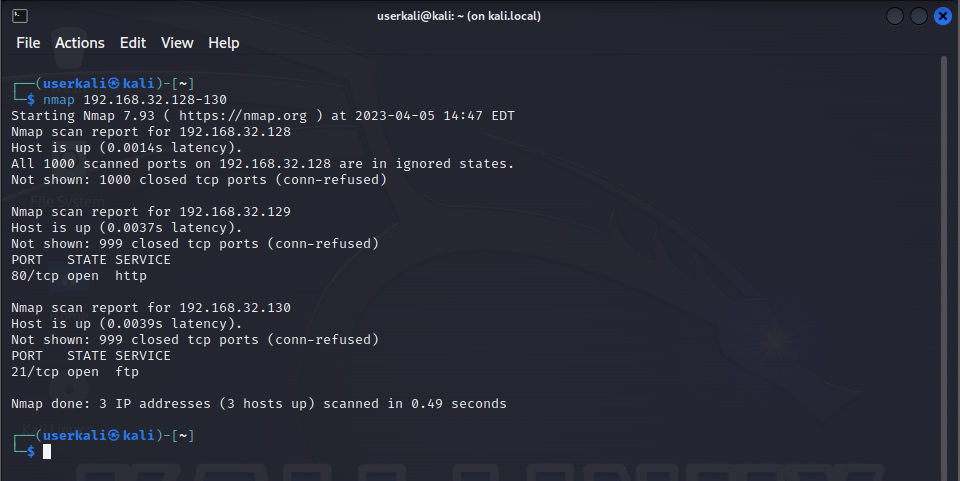


Figure 1. Kali Linux VM, output from Range Scans - IP Address Scans, Step 1.

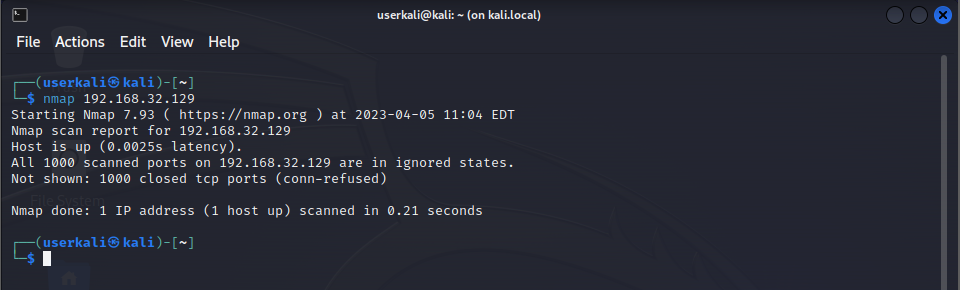
*(\*) There are many reasons you could scan ranges of IP addresses; you can automate your scanning procedures, you can use a range scan to discover hosts, and you can use range scans for vulnerability assessments.*

**Port Scans**

**Important Note**

If we look back to the first section **Scanning Basics VM Startup and Commands** and the first example figure provided. Figure 1. Example output from scanning a secure host.

We see that the output states that, “All 1000 scanned ports on 192.168.32.129 are in an ignored state.” These 1000 ports are not ports 1 through 1000, they are the most common ports for each protocol. NMAP has used their own methods to determine the 1000 ports programmed into NMAP. It is beyond the scope of the **Scanning Basic Module** to get into the details and theory behind the 1000 ports.



**Scanning Basics VM Startup and Commands**

Figure 1. Example output from scanning a secure host.

**Port Scans**

**Single Port Scan**

Let’s start a scanning port.

1. On your Kali Linux VM, run the following command (\*) (\*\*):

**nmap -p 80 192.168.32.129**

Previously we scanned the Ubuntu Linux VM using the command **nmap 192.168.32.128**, nmap returned that it scanned the default 1000 ports and found that port 80 was open.

Now we scan the same host at 192.168.32.129 using the nmap switch “**-p**” and specify the port 80. We see in Figure 1, the output from the command shows that port 80 is open. You will notice that the output doesn’t display that NMAP scanned all 1000 ports. It only scanned port 80 and found that it was open.

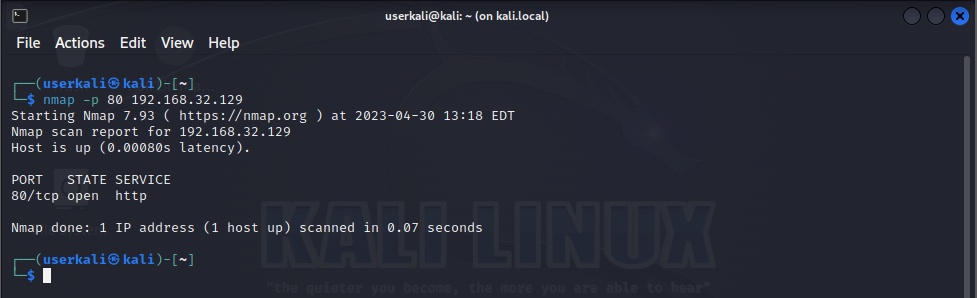


Figure 1. Kali Linux VM, Port Scans, Step 1.

*(\*) NMAP, like many command line programs, uses switches. Switches provide arguments that modify the command. In this case we use the “****-p****” switch to specify port numbers.*

*(\*\*) Specifying ports reduces network traffic and it guarantees that the ports that need to be scanned get scanned.*

**Port Scans**

**Multiple Port Scan**

Let’s start scanning multiple ports.

1. On your Kali Linux VM, run the following command (\*):

**nmap -p 23,80,53,11235 192.168.32.129**

The output in Figure 1, shows that ports 23, 53, and 11235 are closed and that port 80 is open on IP address 192.168.32.129. You can use the switch “**-p**” and port numbers separated by commas to scan multiple specific ports.

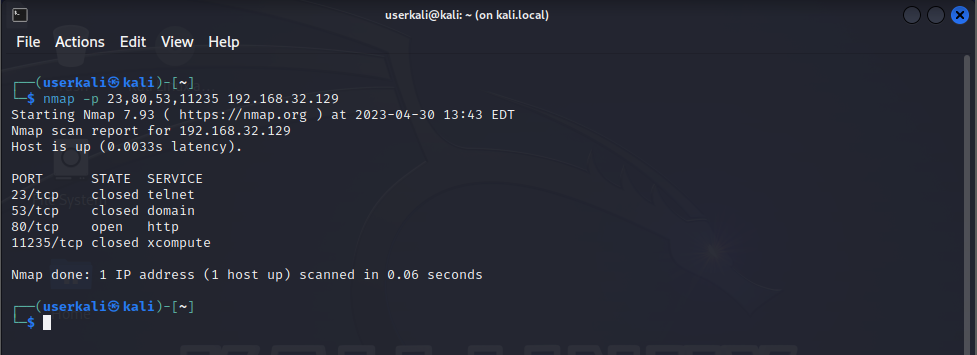


Figure 1. Kali Linux VM, output from Port Scans Multiple Port Scans, Step 1.

*(\*) Specifying multiple ports in one command allows for easier automation and reduces command redundancy.*

**Port Scans**

**Port Range Scan**

Let’s start scanning ranges of ports.

1. On your Kali Linux VM, run the following command (\*) (\*\*):

**nmap -p 1-255 192.168.32.129**

The output from Figure 1, shows that the command scanned ports 1 through 255, but only ports 1 through 255. The command did not scan all 1000 default ports. This command is set up using the switch “**-p**” and the range is setup using numbers separated by a hyphen, “**1-255**”.

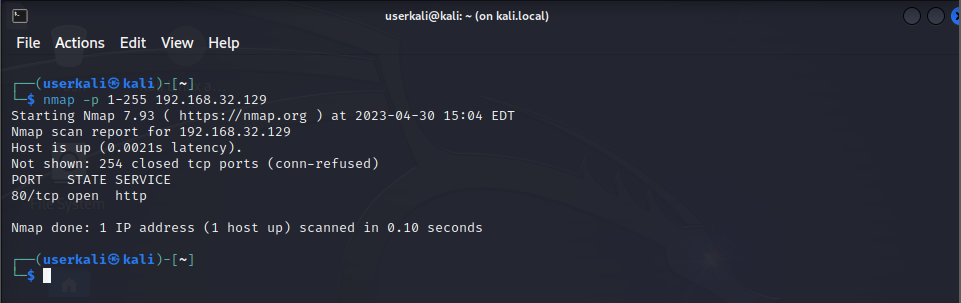


Figure 1. Kali Linux VM, output from Port Scans Port Range Scan, Step 1.

*(\*) Specifying port ranges in one command allows for easier automation and reduces command redundancy.*

*(\*\*) Using a range of ports allow you to scan for ports that may or may not be available.*

**Port Scans**

**Scan Results**

We have seen that ports are open or closed. The scan results have additional port states. These definitions are taken directly from NMAP, (*Port Scanning Basics | Nmap Network Scanning*, n.d.)

**Open**

An application is actively accepting TCP connections, UDP datagrams or SCTP associations on this port.

**Closed**

A closed port is accessible (it receives and responds to Nmap probe packets), but there is no application listening on it.

**Filtered**

Nmap cannot determine whether the port is open because packet filtering prevents its probes from reaching the port. The filtering could be from a dedicated firewall device, router rules, or host-based firewall software.

**Unfiltered**

The unfiltered state means that a port is accessible, but Nmap is unable to determine whether it is open or closed. Only the ACK scan, which is used to map firewall rulesets, classifies ports into this state.

**Open/Filtered**

Nmap places ports in this state when it is unable to determine whether a port is open or filtered. This occurs for scan types in which open ports give no response. The lack of response could also mean that a packet filter dropped the probe or any response it elicited. So Nmap does not know for sure whether the port is open or being filtered.

**Closed/Filtered**

This state is used when Nmap is unable to determine whether a port is closed or filtered

**Port Scans**

**Common Ports**

Listed in Table 1, are common ports you should know.

Table 1. Common Ports

|  |  |  |
| --- | --- | --- |
| **Port Number** | **Service** | **Description** |
| **20** | **FTP-Data** |  |
| **21** | **FTP** | **File Transfer Protocol data transfer** |
| **22** | **SSH-SCP** | **File Transfer Protocol (FTP) control connection** |
| **23** | **Telnet** | **Telnet** |
| **25** | **SMTP** | **Simple Mail Transfer Protocol** |
| **53** | **DNS** | **Domain Name System** |
| **69** | **TFTP** | **Trivial File Transfer Protocol** |
| **80** | **HTTP** | **Hypertext Transfer Protocol (HTTP)** |
| **88** | **Kerberos** | **Network authentication system** |
| **110** | **POP3** | **Post Office Protocol, version 3 (POP3)** |
| **137** | **NetBios-ns** | **NetBIOS Name Service, used for name registration and resolution** |
| **139** | **NetBios-ssn** | **NetBIOS Session Service** |
| **143** | **IMAP4** | **Internet Message Access Protocol (IMAP)** |
| **443** | **HTTP over SSL** | **Hypertext Transfer Protocol Secure (HTTPS)** |
| **636** | **LDAP over TLS/SSL** | **Lightweight Directory Access Protocol over TLS/SSL** |
| **853** | **Secure DNS** | **Secure Domain Name System** |
| **989** | **FTP over SSL** | **FTPS Protocol (data), FTP over TLS/SSL** |
| **990** | **FTP over SSL** | **FTPS Protocol (control), FTP over TLS/SSL** |
| **993** | **IMAP4 over SSL** | **Internet Message Access Protocol over TLS/SSL (IMAPS)** |
| **995** | **POP3 over SSL** | **Post Office Protocol 3 over TLS/SSL** |

**Export Scan Results**

**Information**

Using NMAP to access the status of ports and services immediately is useful. Often, we are going to need to analyze the results at a later point in time. Exporting the results to a file is how we do this.

Command line programs display their results to the terminal. To export the results to a file we need some additional steps.

To export the results of a program or command to a file we use the following syntax.

**command > file**

The important element in this example is the greater than “**>**” symbol. This directs the output from the command or program into the file. The greater than “**>**” symbol overwrites the contents of the file that the output has been directed to.

To append the results of a command or program, use the double greater than “**>>**” symbol. This will not destroy the original contents of the file. It will add the results from the current command to the existing information.

**command >> file**

**Export Scan Results**

**Exporting Results**

Let’s export the results of a nmap scan to a file.

1. On your Kali Linux VM, run the following command:

**touch output.txt**

The touch command creates an empty file in Kali Linux. We need a destination for our output in the next command.

2. On your Kali Linux VM, run the following command (\*):

**nmap -p 1-255 192.168.32.129 >> output.txt**

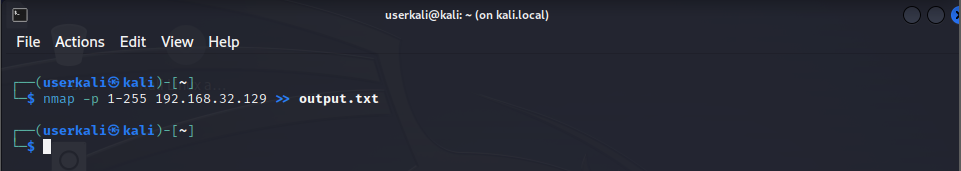


Figure 1. Kali Linux VM, output from Export Scan Results Exporting Results, Step 2.

The output in Figure 1, shows the output from the command in Step 2. We see that there is no output. If we redirect the output from a command or program to a file, then the results will not be displayed to the terminal.

To view the output from nmap that was directed to the output file “**output.txt**” we can use several commands and programs. A core linux command line program is “**cat**”, this command outputs the contents of a file to the terminal.

*(\*) With the results of a command or program output to a file you are to use and edit the results in other interesting ways. Such as processing them in aggregation programs like a SEIM.*

3. On your Kali Linux VM, run the following command:

**cat output.txt**

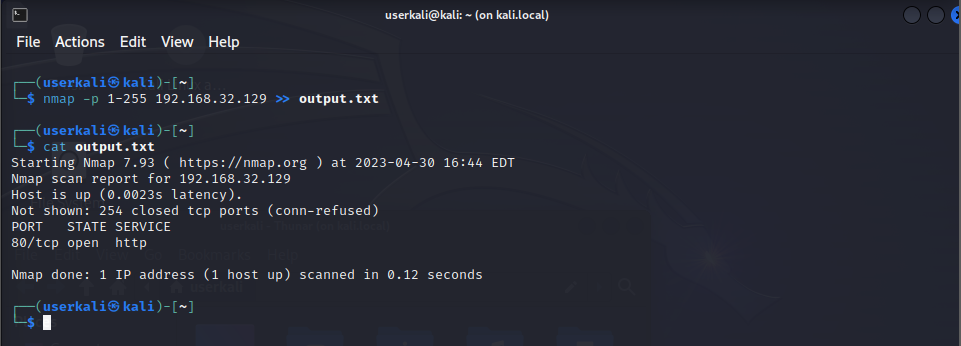


Figure 2. Kali Linux VM, output from Export Scan Results Exporting Results, Step 3.

The output in Figure 2, shows the results from the command in Step 3. This is redundant in our examples. Displaying the same results from the exported results from the nmap command in Step 2 to the terminal.

The program “**cat**” is a foundational command that is useful in the operation of the Linux terminal.

4. On your Kali Linux VM, run the following command (\*):

**vi output.txt**

The screenshot in Figure 3, shows the file output.txt open in the vi program. With the vi program you are able to edit the file.

5. On your Kali Linux VM, use the following keys to exit vi.

1. **press, ESC.**
2. **press, : (colon).**
3. **press, q (lowercase Q).**



Figure 3. Kali Linux VM, output from Export Scan Results Exporting Results, Step 4.

(\*) Linux has a variety of text editing programs like vi. Depending on the distribution you use. The commands presented here are for vi.