NETWORK SECURITY - Milestone 1

Network Packet capture and understanding:
Basic understanding of OSI layer TCP/IP protocol and packets using Wireshark.

Port and Service Scanning:

1. Netdiscover
2. Nmap Scan
    a. sT scan
    b. sS scan
    c. sF scan
    d. sN scan
    e. sU scan
    f. sV and O scan (service based and operating system scan)
3. Nmap script-based scanning - for all ports

Metasploit server ip: **192.168.56.105**

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:fd:5b:aa
          inet addr:192.168.56.105  Bcast:192.168.56.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fefd:5baa/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:29 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1188 (1.1 KB)  TX bytes:3638 (3.5 KB)
          Base address:0xd020 Memory:f1200000-f1220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:97 errors:0 dropped:0 overruns:0 frame:0
          TX packets:97 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:21529 (21.0 KB)  TX bytes:21529 (21.0 KB)
```

Kali linux ip: **192.168.56.101**

```
root@kali:/home/kali# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:1f:30:76 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.101/24 brd 192.168.56.255 scope global dynamic noprefixroute eth0
       valid_lft 377sec preferred_lft 377sec
    inet6 fe80::a00:27ff:fe1f:3076/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

1. **Netdiscover:** To identify the IP and MAC addresses that a business organization using netdiscover

Before scanning, we must do information gathering about the company/ business whom we are going to exploit knowing their vulnerabilities. For example, we can perform an analysis to know what are the actual servers running in a particular range in a business organization.

Netdiscover is an initial recon tool where you can scan for live hosts in a network within an organization.

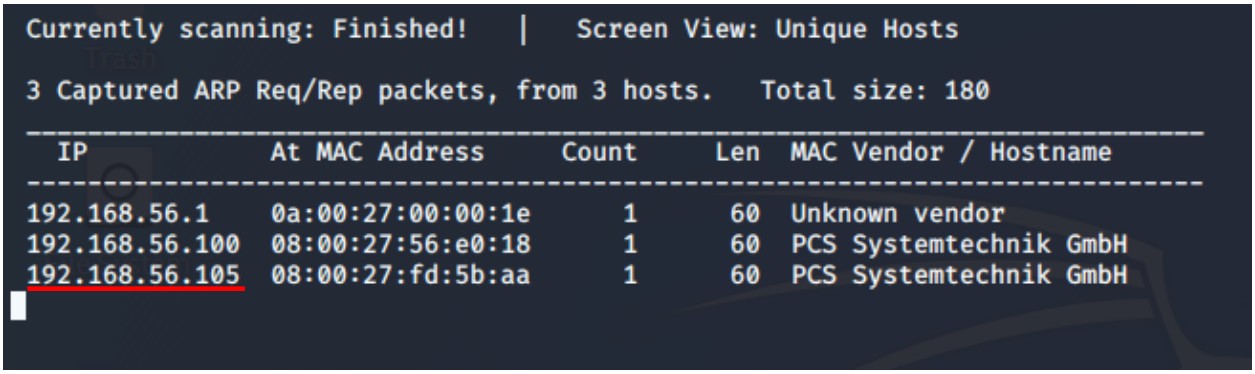
- About netdiscover command with options listed: **netdiscover --help**

```
root@kali:/home/kali# netdiscover --help
netdiscover: invalid option -- '-'

Netdiscover 0.5.1 [Active/passive ARP reconnaissance tool]
Written by: Jaime Penalba <jpenalbae@gmail.com>

Usage: netdiscover [-i device] [-r range | -l file | -p] [-m file] [-F filter] [-s time] [-c count] [-n node] [-dfPLNS]
  -i device: your network device
  -r range: scan a given range instead of auto scan. 192.168.6.0/24,/16,/8
  -l file: scan the list of ranges contained into the given file
  -p passive mode: do not send anything, only sniff
  -m file: scan a list of known MACs and host names
  -F filter: customize pcap filter expression (default: "arp")
  -s time: time to sleep between each ARP request (milliseconds)
  -c count: number of times to send each ARP request (for nets with packet loss)
  -n node: last source IP octet used for scanning (from 2 to 253)
  -d ignore home config files for autoscan and fast mode
  -f enable fastmode scan, saves a lot of time, recommended for auto
  -P print results in a format suitable for parsing by another program and stop after active scan
  -L similar to -P but continue listening after the active scan is completed
  -N Do not print header. Only valid when -P or -L is enabled.
  -S enable sleep time suppression between each request (hardcore mode)

If -r, -l or -p are not enabled, netdiscover will scan for common LAN addresses.
```

The server address is **192.168.56.105** and the netmask value is 24: **192.168.56.0/ 24**
With the help of range '**-r**' option we can scan for specific range.

Syntax: **netdiscover -r <ip range>**

- After executing **netdiscover -r 192.168.56.0/24**,



Output: The output generates a table with live IP addresses with their corresponding MAC addresses, the number of responses, the length of responses and MAC vendor.

Here, my server IP address (192.168.56.105) is also listed with its corresponding MAC address, and response is one from the server with length 60 and vendor - "PCS Systemtechnik GmbH"

- Also we scan scan for multiple ranges. In real time, all organizations own a large network with multiple subnets and network. We can specify all ranges we want to scan.

Syntax: **netdiscover -l ranges**

- Another unique feature that sets this tool apart from the others is the capability to perform passive discovery. Broadcasting ARP requests for every IP address in an entire subnet can sometimes trigger alerts or responses from security devices such as Intrusion Detection Systems (IDS) or Intrusion Prevention Systems (IPS). A stealthier approach is to listen for the ARP traffic, as the scanning system naturally interacts with other systems on the network, and then record the data collected from ARP responses. This passive scanning technique can be performed using the -p option.

Syntax: **netdiscover -p**

This technique is much more effective if it is run on a wireless network, as a promiscuous wireless adapter will receive ARP replies intended for other devices.

```
Currently scanning: (passive)   |   Screen View: Unique Hosts

3 Captured ARP Req/Rep packets, from 2 hosts.   Total size: 180

  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.56.100  08:00:27:56:e0:18      2     120  PCS Systemtechnik GmbH
192.168.56.105  08:00:27:fd:5b:aa      1      60  PCS Systemtechnik GmbH
```

2. **Nmap**: Scanning of running services and ports using Nmap

   Basically, ports are a sort of endpoint connection for two devices to transfer information between each other. For all services, specific port numbers are assigned to identify them. For eg, the common port numbers are: web traffic HTTP uses port 80, for FTP (file transfer) port 21, for secure HTTPS port 443, SSH port 22 etc,

   Nmap is used to discover ports and the services that run on these ports.

   Syntax: **nmap -sS -p- IPaddress**

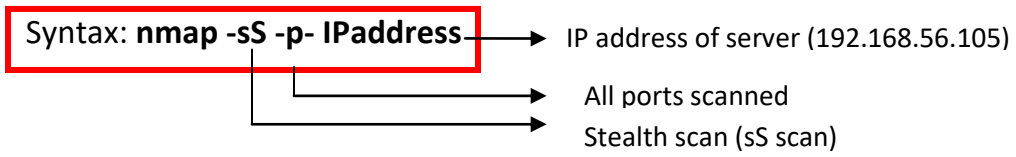
```
root@kali:/home/kali# nmap -sS -p- 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 05:56 EDT
```

```
Nmap scan report for 192.168.56.105
Host is up (0.00034s latency).
Not shown: 65505 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
3632/tcp  open  distccd
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
6697/tcp  open  ircs-u
8009/tcp  open  ajp13
8180/tcp  open  unknown
8787/tcp  open  msgsrvr
38835/tcp open  unknown
40085/tcp open  unknown
42076/tcp open  unknown
48088/tcp open  unknown
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 16.88 seconds
```

Different scanning techniques

a) **Stealth Scan - sS (TCP SYN Scan):** Stealth scan is the default scan. It can be performed quickly than a normal standard scan. Scanning without mentioning port number shows all **open** ports (not al 65000) along with the services running on each port.

The below syntax is used to scan all ports,



```
root@kali:/home/kali# nmap -sS -p- 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 07:36 EDT
```

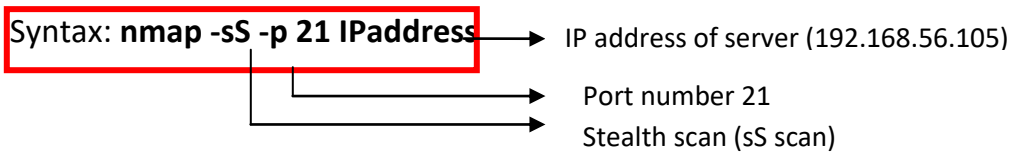
```
Nmap scan report for 192.168.56.105
Host is up (0.00071s latency).
Not shown: 65505 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
3632/tcp  open  distccd
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
6697/tcp  open  ircs-u
8009/tcp  open  ajp13
8180/tcp  open  unknown
8787/tcp  open  msgsrvr
38835/tcp open  unknown
40085/tcp open  unknown
42076/tcp open  unknown
48088/tcp open  unknown
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)
```

When port is open

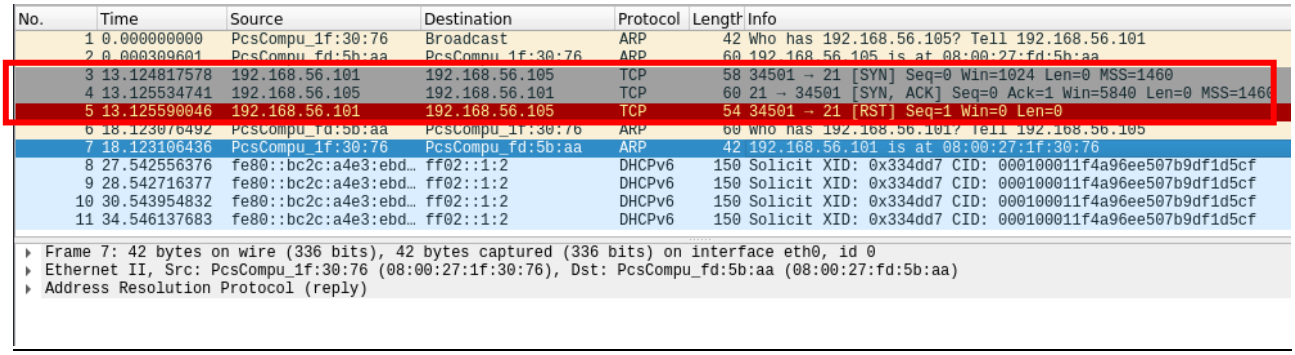
When port is open, it works as:

- Client sends the TCP SYN packet to the server
- Server responds to the request with an acknowledge SYN/ACK
- Client must send the RST to break off communication. Else server keeps on sending ACK which causes DoS attack.

Wireshark tool (Wireshark is an open source tool used as a **packet analyser**, they can see all traffic visible on the interface) is opened at the background and the below syntax is run for an open port **21**. The work flow defined above is depicted in the highlighted area.




```
root@kali:/home/kali# nmap -sS -p 21 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 07:24 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00038s latency).

PORT   STATE SERVICE
21/tcp open  ftp
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.39 seconds
```



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	PcsCompu_1f:30:76	Broadcast	ARP	42	Who has 192.168.56.105? Tell 192.168.56.101
2	0.000309601	PcsCompu_fd:5b:aa	PcsCompu_1f:30:76	ARP	60	192.168.56.105 is at 08:00:27:fd:5b:aa
3	13.124817578	192.168.56.101	192.168.56.105	TCP	58	34501 → 21 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
4	13.125534741	192.168.56.105	192.168.56.101	TCP	60	21 → 34501 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
5	13.125590046	192.168.56.101	192.168.56.105	TCP	54	34501 → 21 [RST] Seq=1 Win=0 Len=0
6	18.123076492	PcsCompu_fd:5b:aa	PcsCompu_1f:30:76	ARP	60	Who has 192.168.56.101? Tell 192.168.56.105
7	18.123106436	PcsCompu_1f:30:76	PcsCompu_fd:5b:aa	ARP	42	192.168.56.101 is at 08:00:27:1f:30:76
8	27.542556376	fe80::bc2c:a4e3:ebd…	ff02::1:2	DHCPv6	150	Solicit XID: 0x334dd7 CID: 000100011f4a96ee507b9df1d5cf
9	28.542716377	fe80::bc2c:a4e3:ebd…	ff02::1:2	DHCPv6	150	Solicit XID: 0x334dd7 CID: 000100011f4a96ee507b9df1d5cf
10	30.543954832	fe80::bc2c:a4e3:ebd…	ff02::1:2	DHCPv6	150	Solicit XID: 0x334dd7 CID: 000100011f4a96ee507b9df1d5cf
11	34.546137683	fe80::bc2c:a4e3:ebd…	ff02::1:2	DHCPv6	150	Solicit XID: 0x334dd7 CID: 000100011f4a96ee507b9df1d5cf

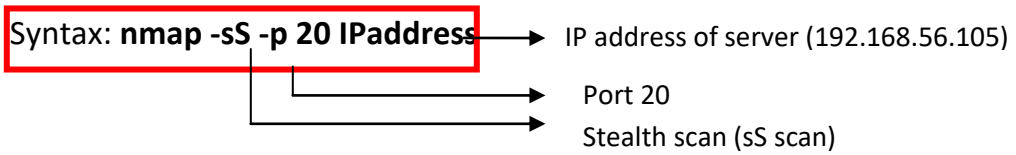
```
▸ Frame 7: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
▸ Ethernet II, Src: PcsCompu_1f:30:76 (08:00:27:1f:30:76), Dst: PcsCompu_fd:5b:aa (08:00:27:fd:5b:aa)
▸ Address Resolution Protocol (reply)
```

When port is closed

When port is closed, it works as:

- Client sends the TCP SYN packet to the server
- But server abruptly closes the communication by sending RST

The below syntax is run for a closed port 20. Wireshark depicts the above scenario for a closed port in the highlighted area.



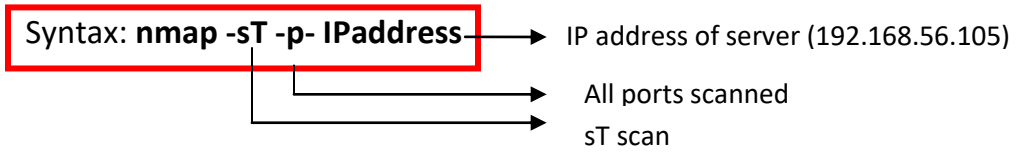
```
root@kali:/home/kali# nmap -sS -p 20 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 07:56 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00043s latency).

PORT   STATE  SERVICE
20/tcp closed ftp-data
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.25 seconds
```

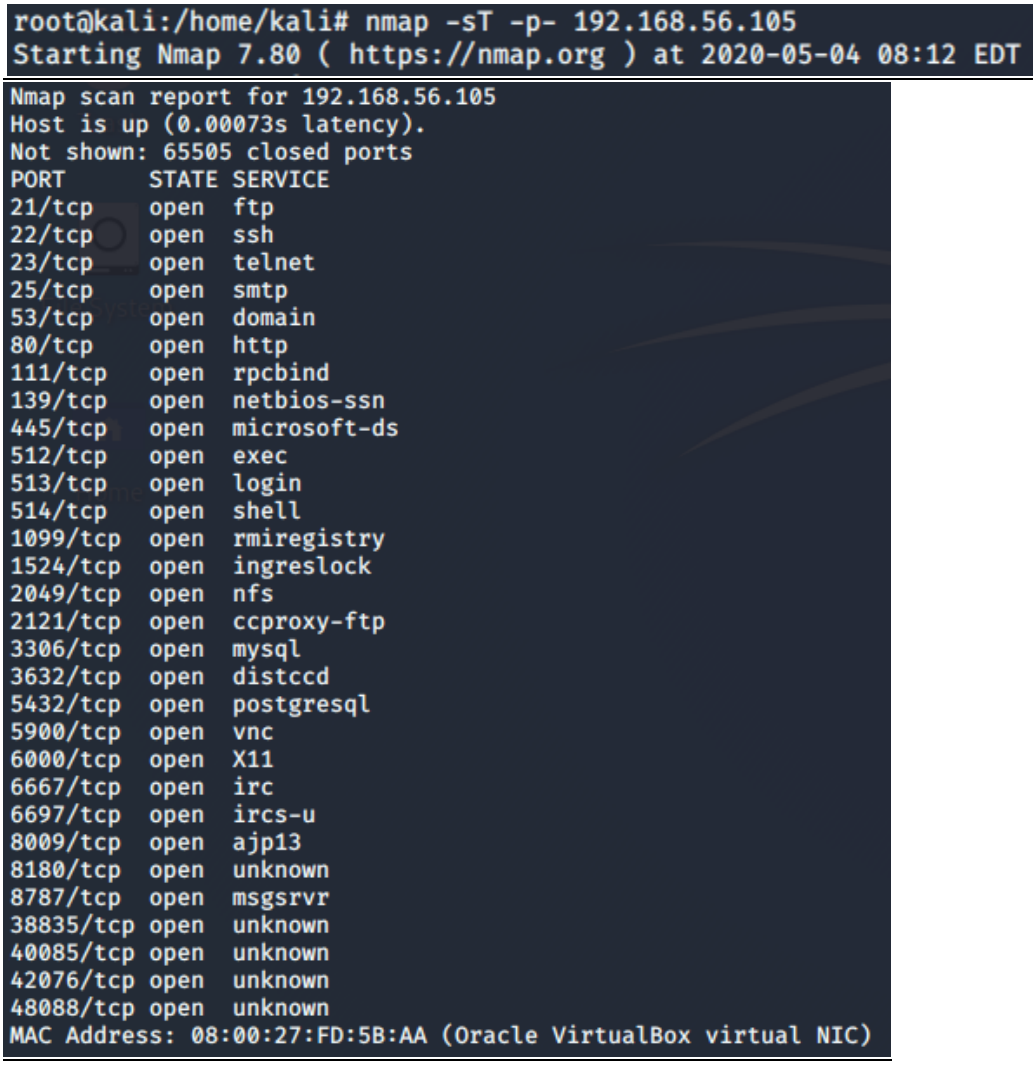
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	PcsCompu_1f:30:76	Broadcast	ARP	42	Who has 192.168.56.105? Tell 192.168.56.101
2	0.000397453	PcsCompu_fd:5b:aa	PcsCompu_1f:30:76	ARP	60	192.168.56.105 is at 08:00:27:fd:5b:aa
4	13.068097766	192.168.56.101	192.168.56.105	TCP	58	63338 → 20 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
5	13.068629403	192.168.56.105	192.168.56.101	TCP	60	20 → 63338 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
7	18.063511895	PcsCompu_1f:30:76	PcsCompu_fd:5b:aa	ARP	42	192.168.56.101 is at 08:00:27:1f:30:76

```
▸ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
▸ Ethernet II, Src: PcsCompu_1f:30:76 (08:00:27:1f:30:76), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▸ Address Resolution Protocol (request)
```

b) **- sT (TCP Connect Scan):** The TCP connect scan is a normal standard scan. This scan doesn't need root privilege and executed same as Stealth scan. It shows a complete TCP handshake connection and it takes a very long time to perform port scanning. So, we can mention port number to make this work faster. When SYN scan is available, it is usually a better choice. Nmap has less control over the high level connect call than with raw packets, making it less efficient.

The below syntax is exceuted without mentioning port number:



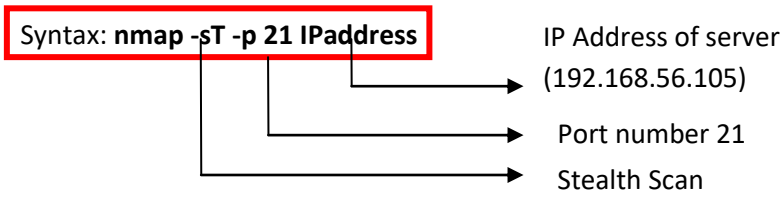


When port is open

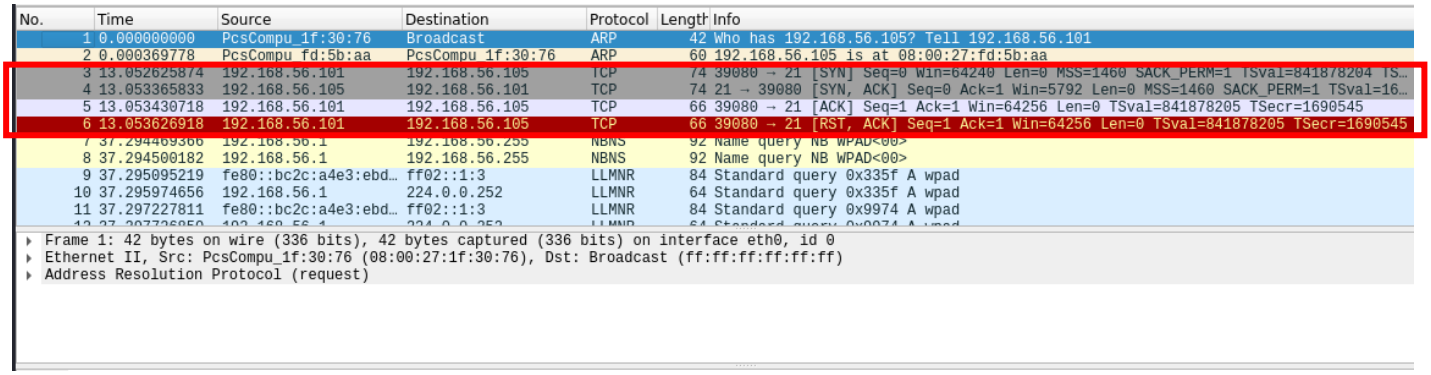
When port is open, it works as:

- Client sends the TCP SYN packet to the server
- Server responds to the request with an acknowledge SYN/ACK
- Client must send the RST to break off communication. Else server keeps on sending ACK which causes DoS attack.

Wireshark tool is opened at the background and the below syntax is run for an open port **21**. The work flow defined above is depicted in the highlighted area.




```
root@kali:/home/kali# nmap -sT -p 21 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 08:37 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00046s latency).

PORT   STATE SERVICE
21/tcp open  ftp
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.20 seconds
```

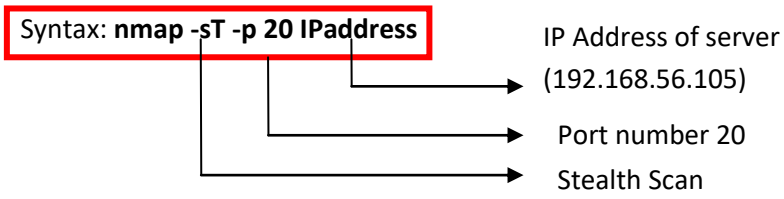


When port is closed

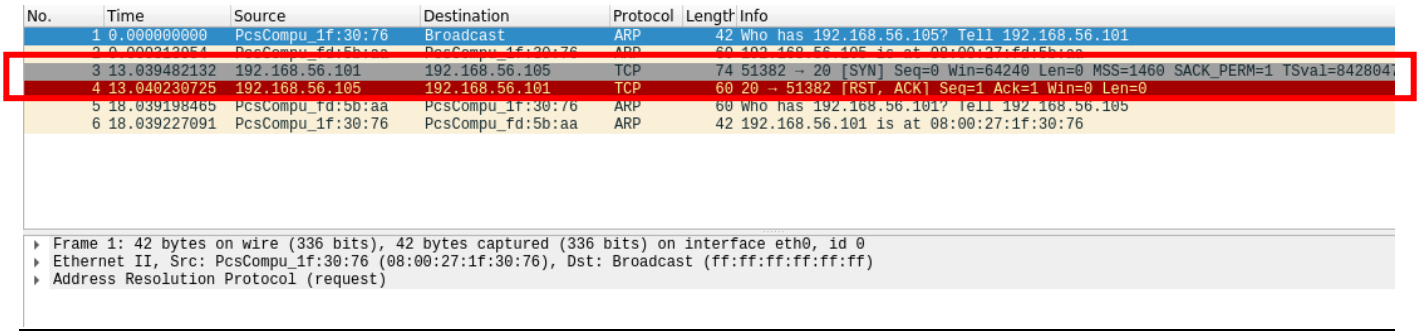
When port is closed, it works as:

- Client sends the TCP SYN packet to the server
- But server abruptly closes the communication by sending RST

The below syntax is run for a closed port 20. Wireshark depicts the above scenario for a closed port in the highlighted area.



```
root@kali:/home/kali# nmap -sT -p 20 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 08:53 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00041s latency).

PORT   STATE  SERVICE
20/tcp closed ftp-data
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.16 seconds
```



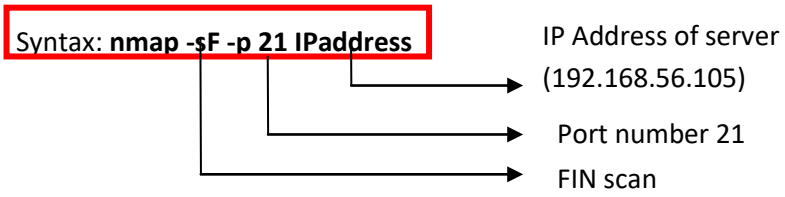
c) **<u>- sF (FIN Scan):</u>** A TCP FIN scan to determine if ports are closed on the target machine. In TCP FIN scan, server sends RST packet and tries abruptly close the communication between server and client. Not most of the computers send a RST packet if they receive input and make communication.

When port is open

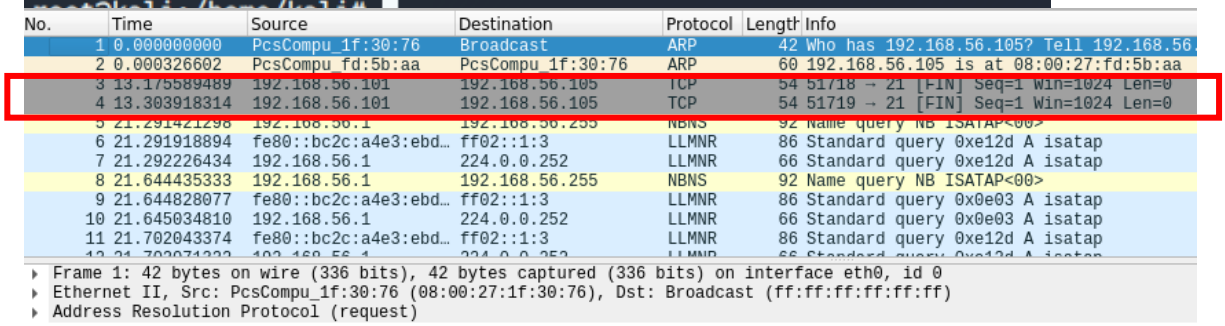
When port is open, it works as:

- Client will send the FIN packet to server
- Server won't respond while the port is open. Any TCP segment with an FIN Flag sent to an open port is discarded, whereas segments with FIN flags sent to closed ports should be handled with a RST in response.

The below syntax is executed, with an open port **21**. The Wireshark depicts the above mentioned scenario.



```
root@kali:/home/kali# nmap -sF -p 21 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 09:25 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00034s latency).

PORT   STATE         SERVICE
21/tcp open|filtered ftp
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.75 seconds
```

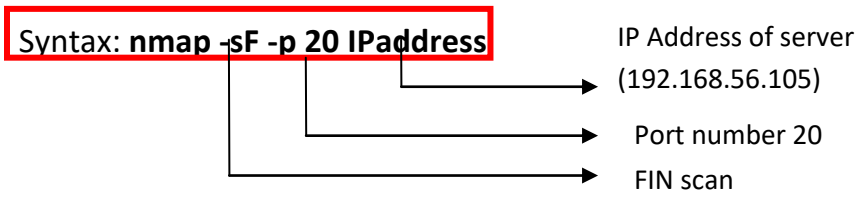


When port is closed

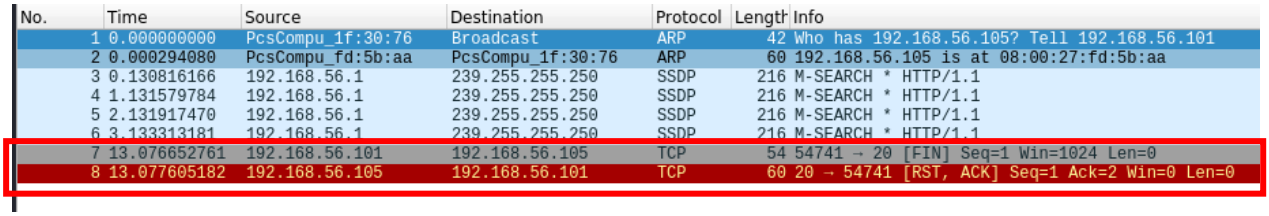
When port is closed, it works as:

- Client sends FIN packet to the server
- Since port is closed, server sends RST packet as response

The below syntax is executed with a closed port **20**. Wireshark depicts the above scenario.




```
root@kali:/home/kali# nmap -sF -p 20 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 09:34 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00040s latency).

PORT   STATE  SERVICE
20/tcp closed ftp-data
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.23 seconds
```



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	PcsCompu_1f:30:76	Broadcast	ARP	42	Who has 192.168.56.105? Tell 192.168.56.101
2	0.000294080	PcsCompu_fd:5b:aa	PcsCompu_1f:30:76	ARP	60	192.168.56.105 is at 08:00:27:fd:5b:aa
3	0.130816166	192.168.56.1	239.255.255.250	SSDP	216	M-SEARCH * HTTP/1.1
4	1.131579784	192.168.56.1	239.255.255.250	SSDP	216	M-SEARCH * HTTP/1.1
5	2.131917470	192.168.56.1	239.255.255.250	SSDP	216	M-SEARCH * HTTP/1.1
6	3.133313181	192.168.56.1	239.255.255.250	SSDP	216	M-SEARCH * HTTP/1.1
7	13.076652761	192.168.56.101	192.168.56.105	TCP	54	54741 → 20 [FIN] Seq=1 Win=1024 Len=0
8	13.077605182	192.168.56.105	192.168.56.101	TCP	60	20 → 54741 [RST, ACK] Seq=1 Ack=2 Win=0 Len=0

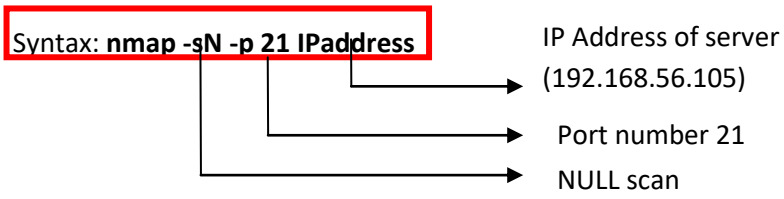
d) **-sN (NULL Scan):** For both FIN and NULL scan the work flow for both open and closed port are similar. But the FIN scan sends the packets containing only the FIN flag, where as the Null scan **does not send any bit on the packet**, and the xmas sends FIN, PSH, and URG flags.

When port is open

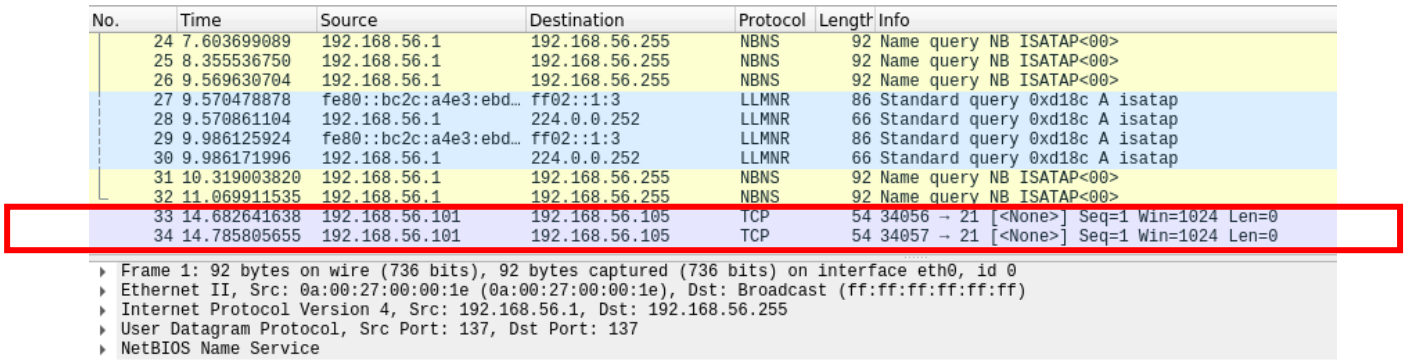
When port is open, it works as:

- Client will send the NULL packet to server. The packet does not contain any bit on the packet. They are TCP packets that contain a sequence number **of 0 and no set flags**.
- Server won't respond while the port is open. The expected result of a Null Scan on an open port is no response. Since there are no flags set, the target will not know how to handle the request. It will discard the packet and no reply will be sent.

The below syntax is executed, with an open port **21**. The Wireshark depicts the above mentioned scenario.




```
root@kali:/home/kali# nmap -sN -p 21 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 09:51 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00039s latency).

PORT   STATE         SERVICE
21/tcp open|filtered ftp
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.48 seconds
```



No.	Time	Source	Destination	Protocol	Length	Info
24	7.603699089	192.168.56.1	192.168.56.255	NBNS	92	Name query NB ISATAP<00>
25	8.355536750	192.168.56.1	192.168.56.255	NBNS	92	Name query NB ISATAP<00>
26	9.569630704	192.168.56.1	192.168.56.255	NBNS	92	Name query NB ISATAP<00>
27	9.570478878	fe80::bc2c:a4e3:ebd…	ff02::1:3	LLMNR	86	Standard query 0xd18c A isatap
28	9.570861104	192.168.56.1	224.0.0.252	LLMNR	66	Standard query 0xd18c A isatap
29	9.986125924	fe80::bc2c:a4e3:ebd…	ff02::1:3	LLMNR	86	Standard query 0xd18c A isatap
30	9.986171996	192.168.56.1	224.0.0.252	LLMNR	66	Standard query 0xd18c A isatap
31	10.319003820	192.168.56.1	192.168.56.255	NBNS	92	Name query NB ISATAP<00>
32	11.069911535	192.168.56.1	192.168.56.255	NBNS	92	Name query NB ISATAP<00>
33	14.682641638	192.168.56.101	192.168.56.105	TCP	54	34056 → 21 [<None>] Seq=1 Win=1024 Len=0
34	14.785805655	192.168.56.101	192.168.56.105	TCP	54	34057 → 21 [<None>] Seq=1 Win=1024 Len=0

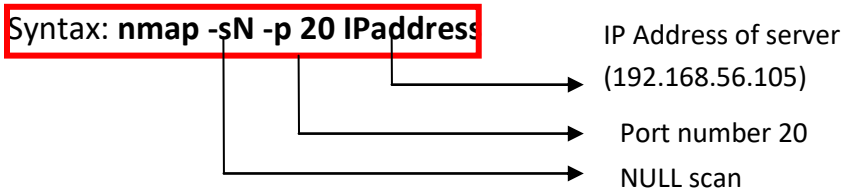
```
▸ Frame 1: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interface eth0, id 0
▸ Ethernet II, Src: 0a:00:27:00:00:1e (0a:00:27:00:00:1e), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▸ Internet Protocol Version 4, Src: 192.168.56.1, Dst: 192.168.56.255
▸ User Datagram Protocol, Src Port: 137, Dst Port: 137
▸ NetBIOS Name Service
```

When port is closed

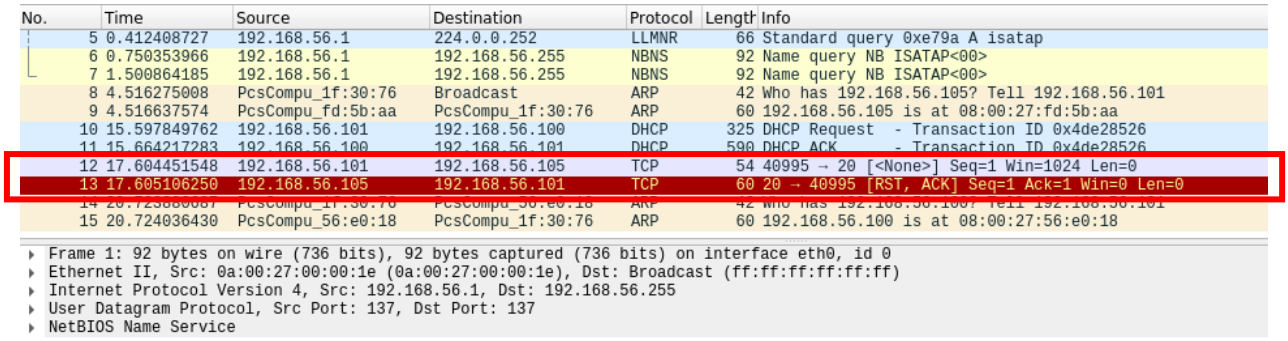
When port is closed, it works as:

- Client sends NULL packet to the server
- Since port is closed, server sends RST packet as response. If the port is closed, the target will send an RST packet in response.

The below syntax is executed with a closed port **20**. Wireshark depicts the above scenario.




```
root@kali:/home/kali# nmap -sN -p 20 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 09:55 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00042s latency).

PORT   STATE  SERVICE
20/tcp closed ftp-data
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.27 seconds
```



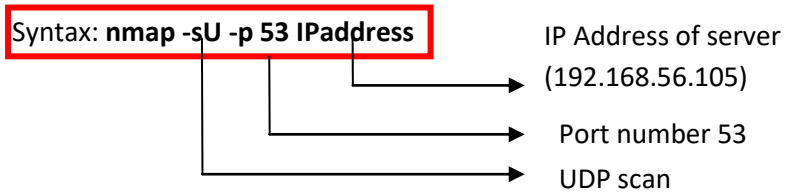
e) **-sU (UDP Scan):** For transport layer, we use two protocols namely TCP and UDP. In the above scenarios, we saw examples for TCP packet scan. Similarly, this scan checks if any UDP packets are listening. It may be quite slow, since some machines intentionally slow down responses to this kind of traffic to avoid being overwhelmed.

When port is open

- Client sends UDP packet to the server to start the conversation
- Since UDP does not respond with a positive acknowledgment like TCP and only responds to an incoming UDP packet when the port is closed. Thus, server does not respond even after several retransmissions.

I received **ICMP Port Unreachable** error. If no response is received after retransmissions, the port is classified as **open|filtered**. This means that the port could be open, or perhaps packet filters are blocking the communication.

The below syntax for an open port **53** is used and Wireshark depicts the above scenario.



```
root@kali:/home/kali# nmap -sU -p 53 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 10:19 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00044s latency).

PORT   STATE SERVICE
53/udp open  domain
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.26 seconds
```

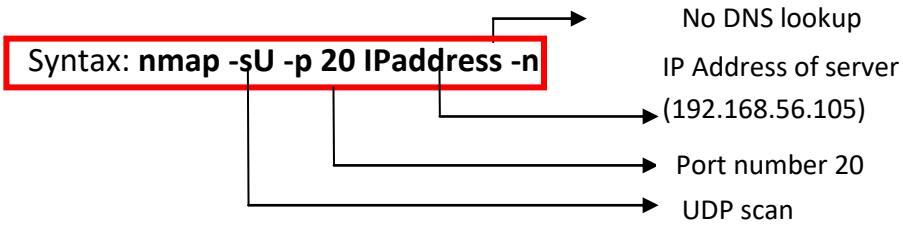
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	PcsCompu_1f:30:76	Broadcast	ARP	42	Who has 192.168.56.105? Tell 192.168.56.101
2	0.000365306	PcsCompu_fd:5b:aa	PcsCompu_1f:30:76	ARP	60	192.168.56.105 is at 08:00:27:fd:5b:aa
3	13.104114149	192.168.56.101	192.168.56.105	DNS	54	Server status request 0x0000
4	13.104982168	192.168.56.105	192.168.56.101	DNS	60	Server status request response 0x0000 Not implemented
5	13.105046206	192.168.56.101	192.168.56.105	ICMP	82	Destination unreachable (Port unreachable)

```
▸ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface eth0, id 0
▸ Ethernet II, Src: PcsCompu_1f:30:76 (08:00:27:1f:30:76), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▸ Address Resolution Protocol (request)
```

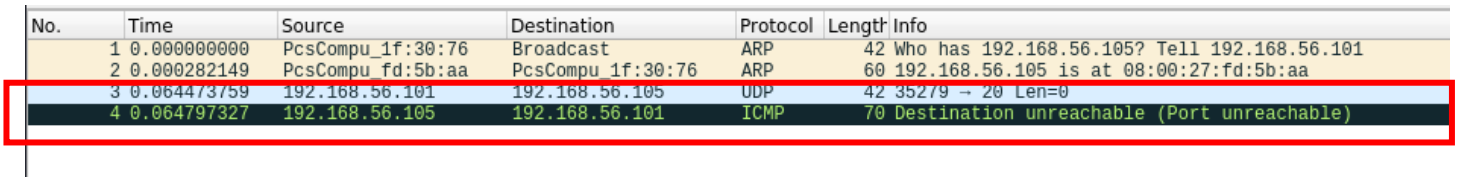
When port is closed

- Client sends a UDP packet to the server and waits for the response
- Closed ports usually receives **ICMP Port Unreachable** error.

The below syntax with a closed port **20** was used and Wireshark depicted the above scenario.



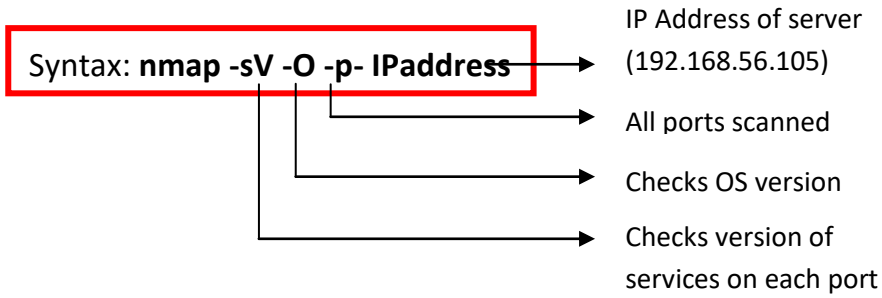

```
root@kali:/home/kali# nmap -sU -p 20 192.168.56.105 -n
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 10:26 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00030s latency).

PORT   STATE  SERVICE
20/udp closed ftp-data
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.18 seconds
```



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	PcsCompu_1f:30:76	Broadcast	ARP	42	Who has 192.168.56.105? Tell 192.168.56.101
2	0.000282149	PcsCompu_fd:5b:aa	PcsCompu_1f:30:76	ARP	60	192.168.56.105 is at 08:00:27:fd:5b:aa
3	0.064473759	192.168.56.101	192.168.56.105	UDP	42	35279 → 20 Len=0
4	0.064797327	192.168.56.105	192.168.56.101	ICMP	70	Destination unreachable (Port unreachable)

f) Service-based and Operating system scanning

To check the open ports and also the version of services on each port along with OS version, the following syntax is used,



Nmap to ask the server for the versions of services it is running and to guess the operating system based on that. Say for example we run Stealth Scan on ports with ip 192.168.56.105, we get their Version of service as for ftd we get **vsftpd 2.3.4**, for ssh we get **OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)** and OS details, hosts etc., are also mentioned.

```
root@kali:/home/kali# nmap -sS -sV -O -p- 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 10:56 EDT
Stats: 0:02:03 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 96.67% done; ETC: 10:58 (0:00:04 remaining)
Nmap scan report for 192.168.56.105
Host is up (0.00052s latency).
Not shown: 65505 closed ports
PORT      STATE SERVICE     VERSION
21/tcp    open  ftp         vsftpd 2.3.4
22/tcp    open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet      Linux telnetd
25/tcp    open  smtp        Postfix smtpd
53/tcp    open  domain      ISC BIND 9.4.2
80/tcp    open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind     2 (RPC #100000)
139/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp   open  exec        netkit-rsh rexecd
513/tcp   open  login       OpenBSD or Solaris rlogind
514/tcp   open  shell       Netkit rshd
1099/tcp  open  java-rmi    GNU Classpath grmiregistry
1524/tcp  open  bindshell   Metasploitable root shell
2049/tcp  open  nfs         2-4 (RPC #100003)
2121/tcp  open  ftp         ProFTPD 1.3.1
3306/tcp  open  mysql       MySQL 5.0.51a-3ubuntu5
3632/tcp  open  distccd     distccd v1 ((GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4))
5432/tcp  open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc         VNC (protocol 3.3)
6000/tcp  open  X11         (access denied)
6667/tcp  open  irc         UnrealIRCd
6697/tcp  open  irc         UnrealIRCd
8009/tcp  open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
8787/tcp  open  drb         Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drb)
38835/tcp open  nlockmgr    1-4 (RPC #100021)
40085/tcp open  status      1 (RPC #100024)
42076/tcp open  java-rmi    GNU Classpath grmiregistry
```

OS and host details: Linux version 2.6.9-2.6.33

```
48088/tcp open  mountd      1-3 (RPC #100005)
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 145.33 seconds
```

3. **Nmap script-based scanning**

To check if exploits work or not, we use **Nmap scripts.** First, find the open ports available using nmap and their service versions on each port. The following syntax is used to get first 10,000 ports (not all ports present),

Syntax: **nmap -sS -sV -O 192.168.56.105**

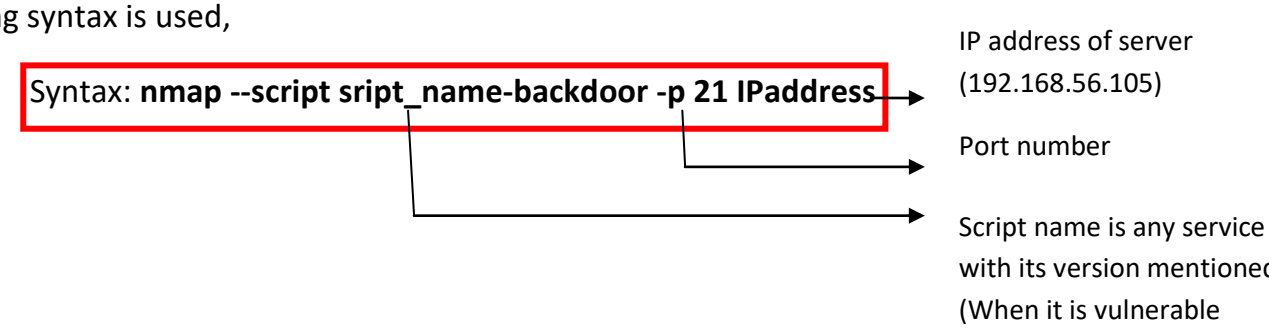
```
root@kali:/home/kali# nmap -sS -sV -O 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 12:02 EDT
```

```
Nmap scan report for 192.168.56.105
Host is up (0.00069s latency).
Not shown: 977 closed ports
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         vsftpd 2.3.4
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open  rpcbind     2 (RPC #100000)
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login
514/tcp  open  shell       Netkit rshd
1099/tcp open  java-rmi    GNU Classpath grmiregistry
1524/tcp open  bindshell   Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 26.60 seconds
```

We get the above mentioned **23 ports**, services running in each port, their version and OS version.

Now to check the vulnerability associated to all 23 ports.

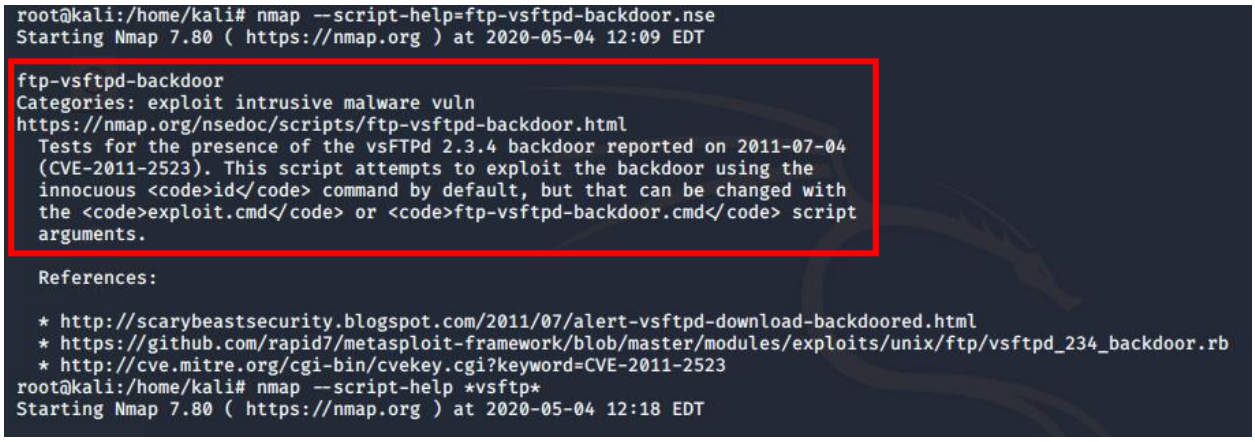
The following syntax is used,



Sno	Service	Description
1	ftp	An FTP server is a computer which has a file transfer protocol (FTP) address and is dedicated to receiving an FTP connection. The File Transfer Protocol (FTP) is a standard network protocol used for the transfer of computer files between a client and server on a computer network.
2	ssh	SSH, also known as Secure Shell or Secure Socket Shell, is a network protocol that gives users, particularly system administrators, a secure way to access a computer over an unsecured network.
3	telnet	*Telnet* is a network protocol used to virtually access a computer and to provide a two-way, collaborative and text-based communication channel between two machines.
4	Smtp	Simple Mail Transfer Protocol, a protocol for sending e-mail messages between servers. The messages can then be retrieved with an e-mail client using either POP or IMAP.
5	domain	A domain service exposes a set of related operations in the form of a service layer. When you define a domain service, you specify the data operations that are permitted through the domain service.
6	http	HTTP are web services, a collection of open protocols and standards used for exchanging data between applications or systems.
7	Rpcbind	The rpcbind utility is a server that converts RPC program numbers into universal addresses. It must be running on the host to be able to make RPC calls on a server on that machine.
8	Smbd	smbd is the server daemon that provides filesharing and printing services to Windows clients. The server provides filespace and printer services to clients using the SMB (or CIFS) protocol.
9	exec	exec will replace the contents of the currently running process with the information from a program binary. After forking process, address space of the child process is overwritten with the new process data.
10	Login	Rlogin port is opened and it's enable the attacker to establish a remote shell with root privilege. This service allows users to login to host remotely.
11	Shell	A shell is special user program which provide an interface to user to use operating system services. Shell accept human readable commands from user and convert them into something which kernel can understand.
12	Java rmi	Java Remote Method with port 1099 , allow attacker establish remote session via vulnerability found in this version, Using metasploit module to scan and exploit.

13	Bindshell	We discovered opened port 1524 with service bindshell. This service allow attacker to establish remote shell using bind shell connection
14	Nfs	The Network File System (NFS) is a client/server application that lets a computer user view and optionally store and update files on a remote computer as though they were on the user's own computer.
15	Proftpd	ProFTPD (short for Pro FTP daemon) is an FTP server. ProFTPD is Free and open-source software, compatible with Unix-like systems and Microsoft Windows (via Cygwin).
16	mysql	MySQL is an Oracle-backed open source relational database management system (RDBMS) based on Structured Query Language (SQL)
17	postgresql	PostgreSQL is a general purpose and object-relational database management system, the most advanced open source database system.
18	Vnc	Virtual network computing (VNC) is a type of remote-control software that makes it possible to control another computer over a network connection. It also interprets commands coming from the viewer and carries them out on the remote computer.
19	x11	X11 service enabled. This service let the attacker take remote screen shot without victim known,this service should be turned off to avoid malicious usage.
20	Irc	IRC stands for "Internet Relay Chat." IRC is a service that allows people to chat with each other online. It operates on a client/server model where individuals use a client program to connect to an IRC server.
21	ajp13	AJP (Apache Jserv Protocol) is basically a binary protocol that allows to reverse proxying requests from a FE Web Server to a BE Application Server, effectively propagating all the needed information to make the Req-Res flow continuing successfully.
22	Apache Tomcat/ Coyote JSP Engine	Apache Tomcat provides software to run Java applets in th browser. Coyote is a stand-alone web server that provides servlets to Tomacat applets.

A. **PORT 21:** if nmap has any scripts that attempt to check for the **VSftpd** vulnerability. With the help of **--script-help**, we can get information about ftp



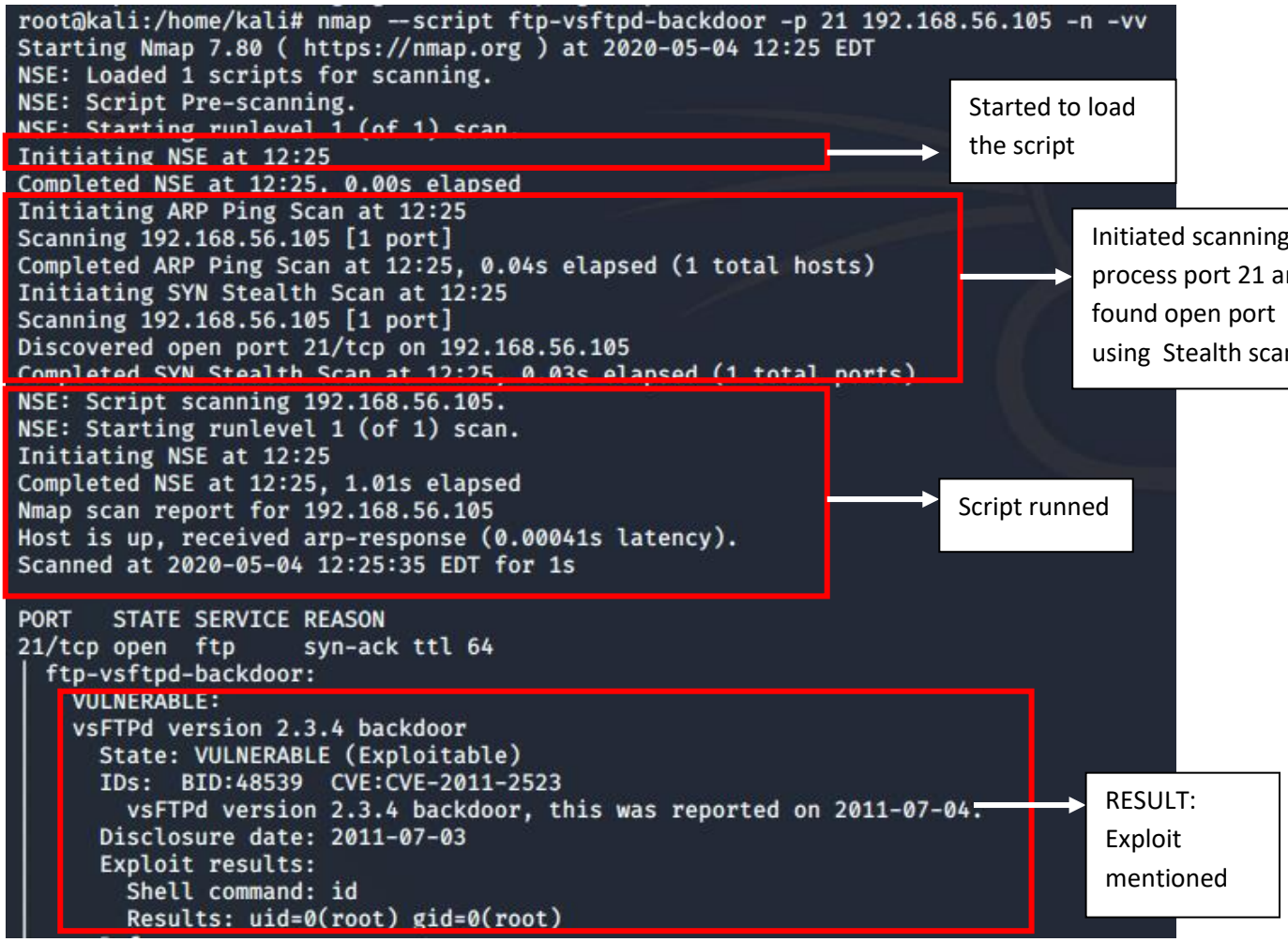
- We got an exploit, so to know if the above highlighted exploit is vulnerable or not, use the syntax described using port number for ftp - port 21.

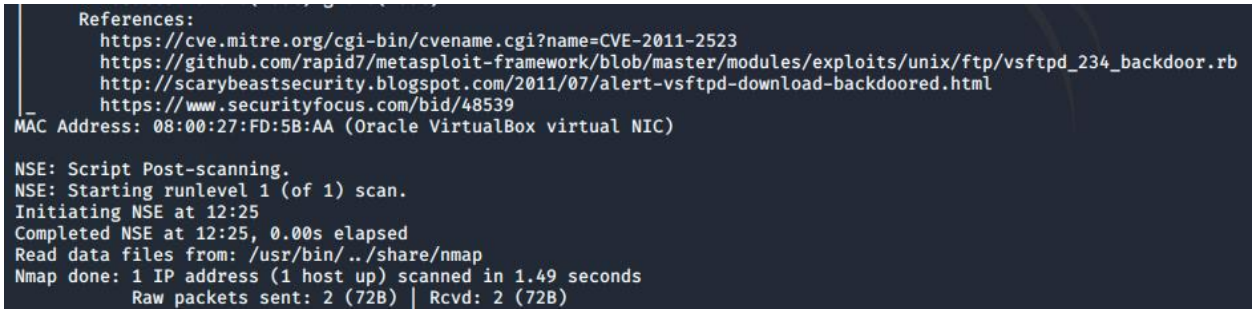
Syntax: **nmap --script ftd-vsftpd-backdoor -p 21 192.168.56.105 -n -vv**

→ script is run for **ftp** with port **21**

→ **-n:** No DNS Lookup

→ **-vv:** verbose flag. This gives information about things happening over script execution.





Output: The last highlighted box gives the result:

- the exploit mentioned, with given Shell command ID, we can obtain the given results and can gain root access.
- Root access is known to attacker (privilege accee disclosed)
- Script executed: **ftp-vsftpd-backdoor**

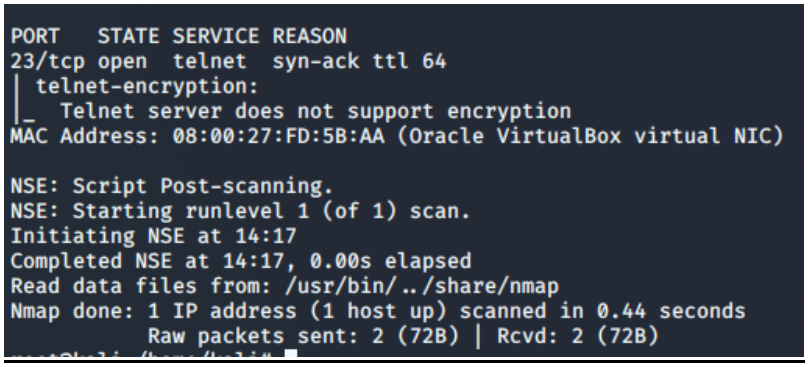
B. **PORT 22:** To check if nmap has any scripts that attempt to check for exploits related to "SSH" service and discover vulnerabilities

```
PORT   STATE SERVICE REASON
22/tcp open  ssh     syn-ack ttl 64
| ssh-hostkey:
|   1024 60:0f:cf:e1:c0:5f:6a:74:d6:90:24:fa:c4:d5:6c:cd (DSA)
| ssh-dss AAAAB3NzaC1kc3MAAACBALz4hsc8a2Srq4nlW960qV8xwBG0JC+jI7fWxm5METIJH4tKr/xUTwsTYEYnaZLzcOiy21D3ZvOwYb6AA3765zdgCd2Tgand7F0YD5UtXG7b7fbz99chReivL0SIWEG/
E96Ai+pqYMP2WD5KaOJwSIXSUajnU5oWmY5×85sBw+XDAAAAFQDFkMpmdFQTF+oRqaoSNVU7Z+hjSwAAAIBCQxNKzi1TyP+QJIFa3M0oLqCVWI0We/ARtXrzpBOJ/dt0hTJXCeYisKqcdwdtyIn8OUCOyrIjqN
uA2QW217oQ6wXpbFh+5AQm8Hl3b6C6o8lX3Ptw+Y4dp0lzfWHwZ/jzHwtuaDQaok7u1f971lEazeJLqfiWrAzoklqSWyDQJAAAAIA1lAD3xWYkeIeHv/R3P9i+XaoI7imFkMuYXCDTq843YU6Td+0mWpllCqAW
UV/CQamGgQLtYy5S0ueoks01MoKdOMMhKVwqdr08nvCBdNKjIEd3gH6oBk/YRnjzxlEAYBsvCmM4a0jmhz0oNiRWlc/F+bkUeFKrBx/D2fdfZmhrGg=
|   2048 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3 (RSA)
|_ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEAstqnuFMBOZvO3WTEjP4TUdjgWkIVNdTq6kboEDjteOfc65TlI7sRvQBwqAhQjeeyyIk8T55gMDkOD0akSlSXvLDcmcdYfxeIF0ZSuT+nkRhij7XSSA/Oc5QS
k3sJ/SInfb78e3anbRHpmkJcVgETJ5WhKObUNf1AKZW++4Xlc63M4KI5cjvMMIPEVOyR3AKmI78Fo3HJjYucg87JjLeC66I7+dlEYX6zT8i1XYwa/L1vZ3qSJISGVu8kRPikMv/cNSvki4j+qDYyZ2E5497W87
+Ed46/8P42LNGoOV8OcX/ro6pAcbEPUdUEfkJrqi2YXbhvwIJ0gFMb6wfe5cnQew=
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 16:02
Completed NSE at 16:02, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
```

```
PORT   STATE SERVICE REASON
22/tcp open  ssh     syn-ack ttl 64
| ssh2-enum-algos:
|   kex_algorithms: (4)
|       diffie-hellman-group-exchange-sha256
|       diffie-hellman-group-exchange-sha1
|       diffie-hellman-group14-sha1
|       diffie-hellman-group1-sha1
|   server_host_key_algorithms: (2)
|       ssh-rsa
|       ssh-dss
|   encryption_algorithms: (13)
|       aes128-cbc
|       3des-cbc
|       blowfish-cbc
|       cast128-cbc
|       arcfour128
|       arcfour256
|       arcfour
|       aes192-cbc
|       aes256-cbc
|       rijndael-cbc@lysator.liu.se
|       aes128-ctr
|       aes192-ctr
|       aes256-ctr
|   mac_algorithms: (7)
|       hmac-md5
|       hmac-sha1
|       umac-64@openssh.com
|       hmac-ripemd160
|       hmac-ripemd160@openssh.com
|       hmac-sha1-96
|       hmac-md5-96
|   compression_algorithms: (2)
|       none
|_      zlib@openssh.com
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)
```

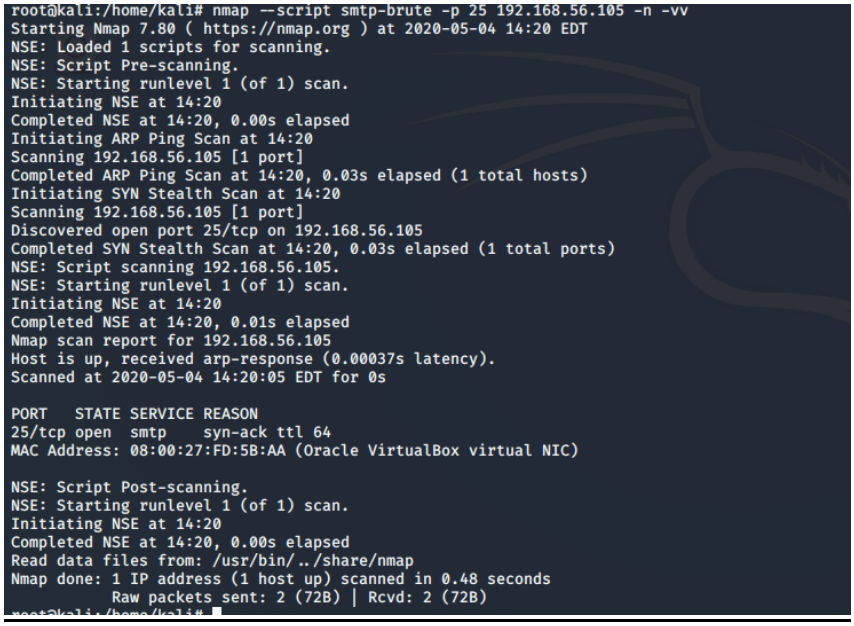
Output: **Exploits** found for the script **ssh-hostkey** displays entire key when executed. NSE script in nmap that can retrieve the ssh-hostkey (RSA or DSA) from the target host. And **ssh2-enum-algos** exposes all cryptographic algorithms used in ssh.

C. **PORT 23:** To check if nmap has any scripts that attempt to check for exploits related to "telnet" service and discover vulnerabilities



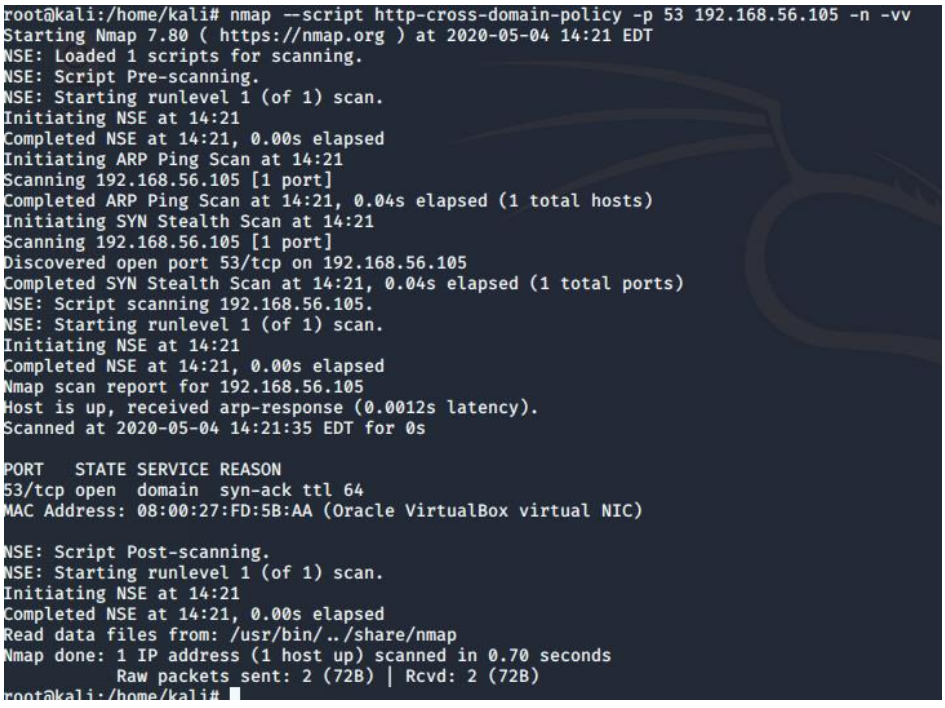
Output: Exploits found for this script **telnet-encryption** saying no encryption given, which means data are seen in clear. Telnet, by default, does not encrypt any data sent over the connection (including passwords), and so it is often feasible to eavesdrop on the communications and use the password later for malicious purposes; anybody who has access the network between the two hosts where Telnet is being used can intercept the packets passing between source and destination and obtain login, password and data information.

D. **PORT 25:** To check if nmap has any scripts that attempt to check for exploits related to "smtp" service and discover vulnerabilities



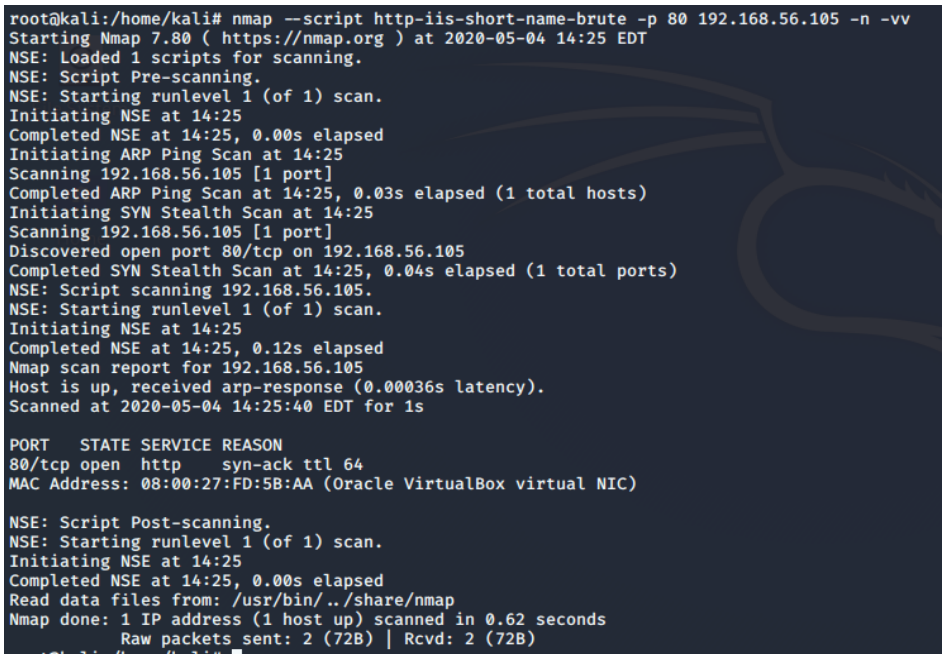
Output: **No exploits** found for this script **smtp-brute** and anything following this script

E. **PORT 53:** To check if nmap has any scripts that attempt to check for exploits related to "domain" service and discover vulnerabilities



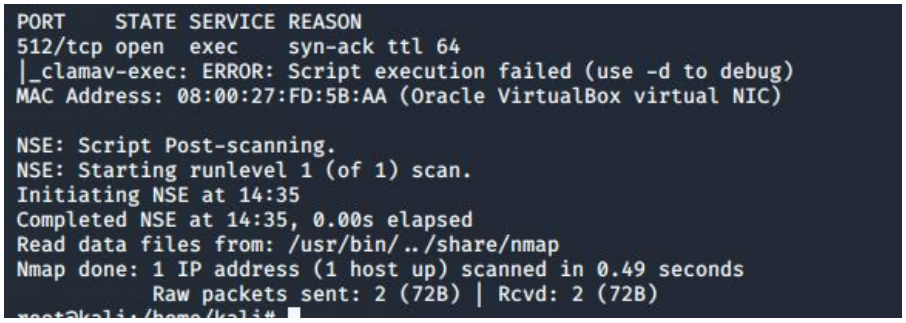
Output: **No exploits** found for this script **http-cross-domain-policy** and anything following this script

F. **PORT 80:** To check if nmap has any scripts that attempt to check for exploits related to "HTTP" service and discover vulnerabilities



Output: **No exploits** found for this script **http-iss-short-name-bruce** and anything following this script

G. **PORT 512:** To check if nmap has any scripts that attempt to check for exploits related to "exec" service and discover vulnerabilities



Output: Exploits found for this script **clamav-exec**

H. PORT 513: To check if nmap has any scripts that attempt to check for exploits related to "login" service and discover vulnerabilities

```
PORT    STATE SERVICE REASON
513/tcp open  login   syn-ack ttl 64
| rlogin-brute:
|   Accounts: No valid accounts found
|   Statistics: Performed 26 guesses in 3 seconds, average tps: 8.7
|_  ERROR: The service seems to have failed or is heavily firewalled...
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:42
Completed NSE at 14:42, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 4.37 seconds
```

Output: Exploits found for this script **rlogin-brute,** may password attempts executed

I. **PORT 514:** To check if nmap has any scripts that attempt to check for exploits related to "SHELL" service and discover vulnerabilities

```
root@kali:/home/kali# nmap --script http-shellshock -p 514 192.168.56.105 -n -vv
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 14:44 EDT
NSE: Loaded 1 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:44
Completed NSE at 14:44, 0.00s elapsed
Initiating ARP Ping Scan at 14:44
Scanning 192.168.56.105 [1 port]
Completed ARP Ping Scan at 14:44, 0.03s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 14:44
Scanning 192.168.56.105 [1 port]
Discovered open port 514/tcp on 192.168.56.105
Completed SYN Stealth Scan at 14:44, 0.03s elapsed (1 total ports)
NSE: Script scanning 192.168.56.105.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:44
Completed NSE at 14:44, 0.00s elapsed
Nmap scan report for 192.168.56.105
Host is up, received arp-response (0.00029s latency).
Scanned at 2020-05-04 14:44:51 EDT for 0s

PORT    STATE SERVICE REASON
514/tcp open  shell   syn-ack ttl 64
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:44
Completed NSE at 14:44, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.49 seconds
           Raw packets sent: 2 (72B) | Rcvd: 2 (72B)
```

Output: **No exploits** found for this script **http-shellshock** and anything following this script

J. **PORT 2049:** To check if nmap has any scripts that attempt to check for exploits related to "nfs" service and discover vulnerabilities

```
root@kali:/home/kali# nmap --script nfs-ls -p 2049 192.168.56.105 -n -vv
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 14:50 EDT
NSE: Loaded 1 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:50
Completed NSE at 14:50, 0.00s elapsed
Initiating ARP Ping Scan at 14:50
Scanning 192.168.56.105 [1 port]
Completed ARP Ping Scan at 14:50, 0.03s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 14:50
Scanning 192.168.56.105 [1 port]
Discovered open port 2049/tcp on 192.168.56.105
Completed SYN Stealth Scan at 14:50, 0.03s elapsed (1 total ports)
NSE: Script scanning 192.168.56.105.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:50
Completed NSE at 14:50, 0.00s elapsed
Nmap scan report for 192.168.56.105
Host is up, received arp-response (0.00035s latency).
Scanned at 2020-05-04 14:50:43 EDT for 0s

PORT     STATE SERVICE REASON
2049/tcp open  nfs     syn-ack ttl 64
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:50
Completed NSE at 14:50, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.51 seconds
           Raw packets sent: 2 (72B) | Rcvd: 2 (72B)
```

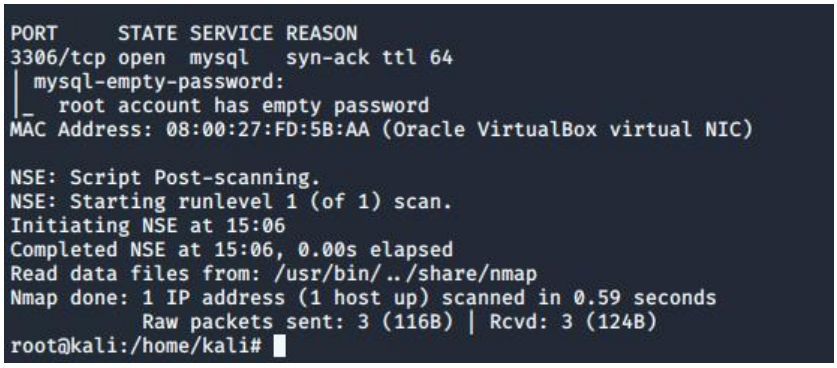
Output: **No exploits** found for this script **nfs-ls** and anything following this script

K. **PORT 2121:** To check if nmap has any scripts that attempt to check for exploits related to "ftp" service - "port 2121" and discover vulnerabilities

```
root@kali:/home/kali# nmap --script ftp-vsftpd-backdoor -p 2121 192.168.56.105 -n -vv
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 14:57 EDT
NSE: Loaded 1 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:57
Completed NSE at 14:57, 0.00s elapsed
Initiating ARP Ping Scan at 14:57
Scanning 192.168.56.105 [1 port]
Completed ARP Ping Scan at 14:57, 0.04s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 14:57
Scanning 192.168.56.105 [1 port]
Discovered open port 2121/tcp on 192.168.56.105
Completed SYN Stealth Scan at 14:57, 0.03s elapsed (1 total ports)
NSE: Script scanning 192.168.56.105.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:57
Completed NSE at 14:57, 0.00s elapsed
Nmap scan report for 192.168.56.105
Host is up, received arp-response (0.00032s latency).
Scanned at 2020-05-04 14:57:12 EDT for 0s

PORT     STATE SERVICE     REASON
2121/tcp open  ccproxy-ftp syn-ack ttl 64
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 14:57
Completed NSE at 14:57, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
           Raw packets sent: 2 (72B) | Rcvd: 2 (72B)
```

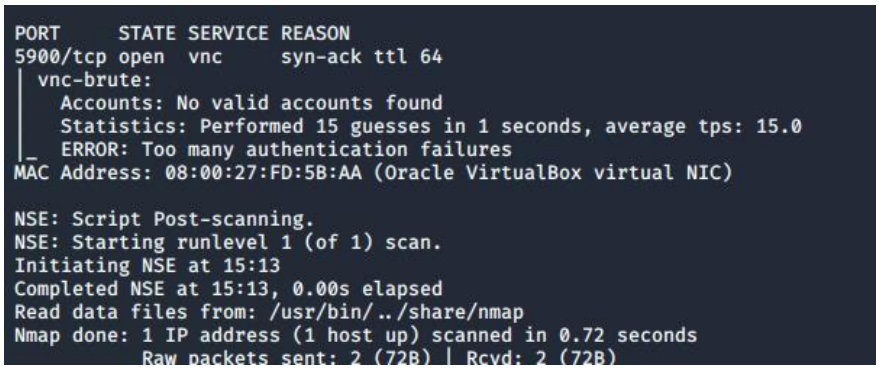
Output: **No exploits** found for this script **ftp-vsftpd-backdoor** and anything following this script using port **2121**

L. **PORT 3306:** To check if nmap has any scripts that attempt to check for exploits related to "mysql" service and discover vulnerabilities



OUTPUT: Exploits found for script **mysql-empty-password,** saying root account is provided no password. No authentication is given and anyone can login as root user.

M. **PORT 5900:** To check if nmap has any scripts that attempt to check for exploits related to "vnc" service and discover vulnerabilities



OUTPUT: Exploits found for script **vnc-brute,** saying brute force attack has been performed trying too many password attempts.

N. **PORT 6000:** To check if nmap has any scripts that attempt to check for exploits related to "x11" service and discover vulnerabilities

```
root@kali:/home/kali# nmap --script x11-access -p 6000 192.168.56.105 -n -vv
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 15:17 EDT
NSE: Loaded 1 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 15:17
Completed NSE at 15:17, 0.00s elapsed
Initiating ARP Ping Scan at 15:17
Scanning 192.168.56.105 [1 port]
Completed ARP Ping Scan at 15:17, 0.08s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 15:17
Scanning 192.168.56.105 [1 port]
Discovered open port 6000/tcp on 192.168.56.105
Completed SYN Stealth Scan at 15:17, 0.03s elapsed (1 total ports)
NSE: Script scanning 192.168.56.105.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 15:17
Completed NSE at 15:17, 0.00s elapsed
Nmap scan report for 192.168.56.105
Host is up, received arp-response (0.00028s latency).
Scanned at 2020-05-04 15:17:01 EDT for 0s

PORT     STATE SERVICE REASON
6000/tcp open  X11     syn-ack ttl 64
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 15:17
Completed NSE at 15:17, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.51 seconds
           Raw packets sent: 2 (72B) | Rcvd: 2 (72B)
```

OUTPUT: **No exploits** found for this script **X11-access** and it has only that script and executed no exploit.

O. **PORT 6667:** To check if nmap has any scripts that attempt to check for exploits related to "irc" service and discover vulnerabilities

```
PORT     STATE SERVICE REASON
6667/tcp open  irc     syn-ack ttl 64
|_irc-unrealircd-backdoor: Looks like trojaned version of unrealircd. See http://seclists.org/fulldisclosure/2010/Jun/277
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 15:29
Completed NSE at 15:29, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 40.47 seconds
```

OUTPUT: Exploit found for script **irc-unrealircd-backdoor.**Some virus (Trojhan) version noticed.

P. **PORT 8180:** To check if nmap has any scripts that attempt to check for exploits related to "http" service for "port 8180" and discover vulnerabilities

```
root@kali:/home/kali# nmap --script http-iis-short-name-brute -p 8180 192.168.56.105 -n -vv
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 15:58 EDT
NSE: Loaded 1 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 15:58
Completed NSE at 15:58, 0.00s elapsed
Initiating ARP Ping Scan at 15:58
Scanning 192.168.56.105 [1 port]
Completed ARP Ping Scan at 15:58, 0.04s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 15:58
Scanning 192.168.56.105 [1 port]
Discovered open port 8180/tcp on 192.168.56.105
Completed SYN Stealth Scan at 15:58, 0.03s elapsed (1 total ports)
NSE: Script scanning 192.168.56.105.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 15:58
Completed NSE at 15:58, 0.13s elapsed
Nmap scan report for 192.168.56.105
Host is up, received arp-response (0.00037s latency).
Scanned at 2020-05-04 15:58:19 EDT for 0s

PORT     STATE SERVICE REASON
8180/tcp open  unknown syn-ack ttl 64
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 1) scan.
Initiating NSE at 15:58
Completed NSE at 15:58, 0.00s elapsed
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.63 seconds
           Raw packets sent: 2 (72B) | Rcvd: 2 (72B)
root@kali:/home/kali#
```

OUTPUT: **No Exploit** found for script **http-iis-short-name-brute** and any scripts following that also has no exploits.

Q. **PORT 1099:** To check if nmap has any scripts that attempt to check for exploits related to "java-rmi" service for "port 8180" and discover vulnerabilities

Syntax: **nmap --script=rmi-vuln-classloader -p 1099 192.168.56.105**

```
PORT     STATE SERVICE
1099/tcp open  rmiregistry
| rmi-vuln-classloader:
|   VULNERABLE:
|   RMI registry default configuration remote code execution vulnerability
|     State: VULNERABLE
|       Default configuration of RMI registry allows loading classes from remote URLs which can lead to remote code execution.
|
|     References:
|_      https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/misc/java_rmi_server.rb
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.67 seconds
```

OUTPUT: Exploit found in script **rmi-vuln-classloader.** Tests whether Java rmiregistry allows class loading. The default configuration of rmiregistry allows loading classes from remote URLs, which can lead to remote code execution.

R. **PORT 8009:** To check if nmap has any scripts that attempt to check for exploits related to "ajp13" service for "port 8180" and discover vulnerabilities

Syntax: **nmap -p 8009 192.168.56.105 --script ajp-methods**

```
PORT     STATE SERVICE
8009/tcp open  ajp13
|_ajp-methods: Failed to get a valid response for the OPTION request
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 13.67 seconds
root@kali:/home/kali#
```

OUTPUT: **No exploits** found in script **ajp-methods.**

S. **PORT 5432:** To check if nmap has any scripts that attempt to check for exploits related to "postgresql" service for "port 8180" and discover vulnerabilities

Syntax: **nmap -p 5432 --script pgsql-brute 192.168.56.105**

```
root@kali:/home/kali# nmap -p 5432 --script pgsql-brute 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 17:00 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00029s latency).

PORT     STATE SERVICE
5432/tcp open  postgresql
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 213.15 seconds
```

OUTPUT: **No exploits** found in script **pgsql-brute.**

T. **PORT 139:** To check if nmap has any scripts that attempt to check for exploits related to "netbios-smb" service for "port 8180" and discover vulnerabilities

Syntax: **nmap -p 139 --script smb-protocols 192.168.56.105**

nmap -sU -sS --script smb-enum-shares.nse -p 139 192.168.56.105

nmap -sU -sS --script smb-os-discovery.nse -p 139 192.168.56.105

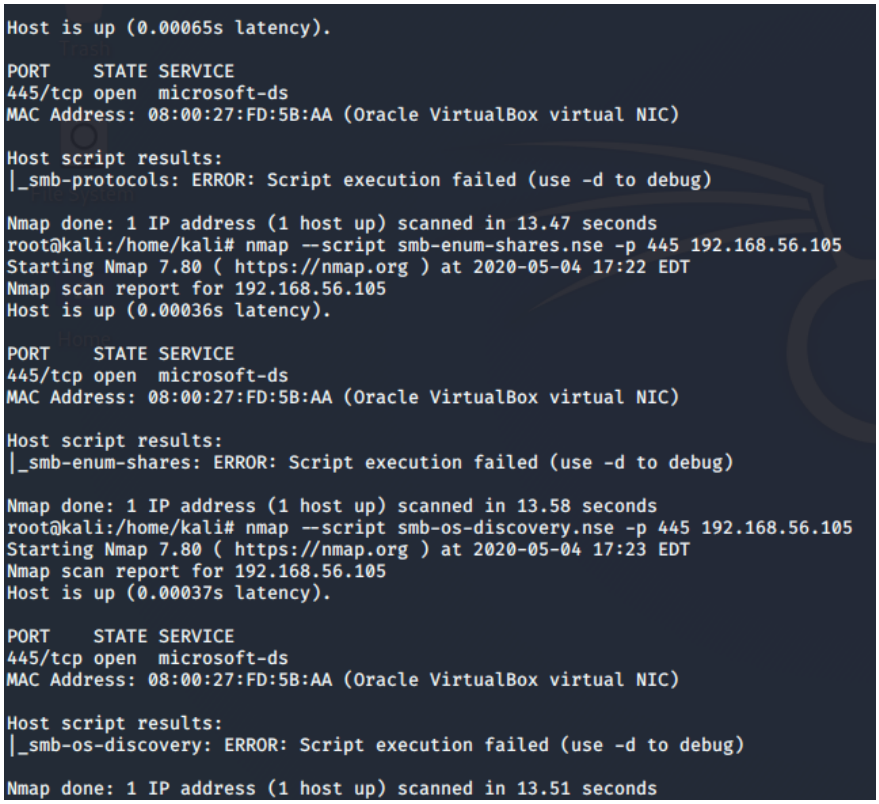
```
root@kali:/home/kali# nmap -p 139 --script smb-protocols 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 17:07 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00070s latency).

PORT    STATE SERVICE
139/tcp open  netbios-ssn
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Host script results:
|_smb-protocols: ERROR: Script execution failed (use -d to debug)

Nmap done: 1 IP address (1 host up) scanned in 13.79 seconds
```

```
root@kali:/home/kali# nmap -sU -sS --script smb-enum-shares.nse -p 139 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 17:10 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00064s latency).

PORT    STATE  SERVICE
139/tcp open   netbios-ssn
139/udp closed netbios-ssn
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Host script results:
|_smb-enum-shares: ERROR: Script execution failed (use -d to debug)

Nmap done: 1 IP address (1 host up) scanned in 13.77 seconds
root@kali:/home/kali# nmap -sU -sS --script smb-os-discovery.nse -p 139 192.168.56.105
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-04 17:11 EDT
Nmap scan report for 192.168.56.105
Host is up (0.00073s latency).

PORT    STATE  SERVICE
139/tcp open   netbios-ssn
139/udp closed netbios-ssn
MAC Address: 08:00:27:FD:5B:AA (Oracle VirtualBox virtual NIC)

Host script results:
|_smb-os-discovery: ERROR: Script execution failed (use -d to debug)

Nmap done: 1 IP address (1 host up) scanned in 13.68 seconds
```

OUTPUT: For all 3 syntax, **no exploits** were found.

U. **PORT 445:** To check if nmap has any scripts that attempt to check for exploits related to "netbios-smb" service for "port 8180" and discover vulnerabilities

Syntax: **nmap -p 445 --script smb-protocols 192.168.56.105**
nmap --script smb-enum-shares.nse -p 445 192.168.56.105
nmap --script smb-os-discovery.nse -p 445 192.168.56.105



OUTPUT: For all 3 syntax, **no exploits** were found.