iGNITE
Technologies

Nmap for Pentester

Hex Value of Flags

WWW.HACKINGARTICLES.IN

Contents

Introduction

Today we are going to scan the target machine by sending TCP flags through their hexadecimal value, and the actual flag name can be confirmed by analysis of Nmap traffic through Wireshark.

Let’s have a look at the hex value of the TCP Flag in the given below table, which we are going to use in Nmap for port enumeration.

Flags	Decimal Value	Hexadecimal Value
NULL	0	0x00
FIN	1	0x01
SYN	2	0x02
RST	4	0x04
PSH	8	0x08
ACK	16	0x10
URG	32	0x20
ECE	64	0x40
CWR	128	0x80
NS	256	0x100

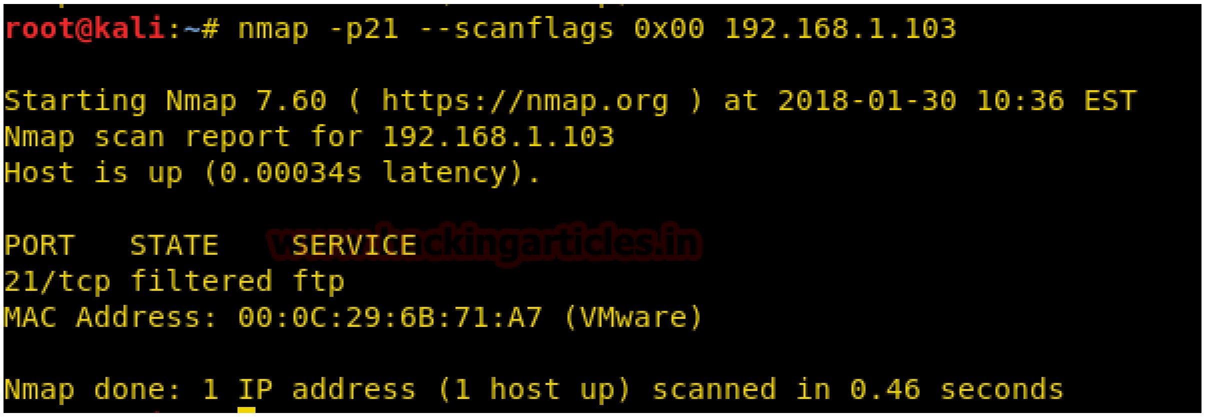
NULL Scan

In this scan, we are sending the NONE flag of the TCP by using its hexadecimal value on the target machine to enumerate the state of ports that are open, closed, or filtered.

Now execute the given below command for enumerating the state of any port. Here we want to identify the state of port 21.

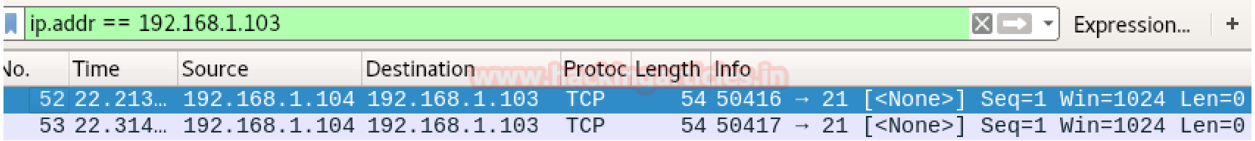
```
nmap -p21 --scanflags 0x00 192.168.1.103
```

From the given below image, you can observe that we have found port 21 filtered.



```
root@kali:~# nmap -p21 --scanflags 0x00 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 10:36 EST
Nmap scan report for 192.168.1.103
Host is up (0.00034s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
```

When a network admin captures the incoming traffic, he will get a packet with the TCP-NONE flag. Here we have used Wireshark for network packet analysis and found that it is showing a **TCP-NONE packet** for hex value **0x00** coming from 192.168.1.104 on port 21 as shown in the given below image.



```
ip.addr == 192.168.1.103                                                     Expression...  +
No.  Time      Source        Destination   Protoc Length Info
 52 22.213…  192.168.1.104 192.168.1.103  TCP       54 50416 → 21 [<None>] Seq=1 Win=1024 Len=0
 53 22.314…  192.168.1.104 192.168.1.103  TCP       54 50417 → 21 [<None>] Seq=1 Win=1024 Len=0
```

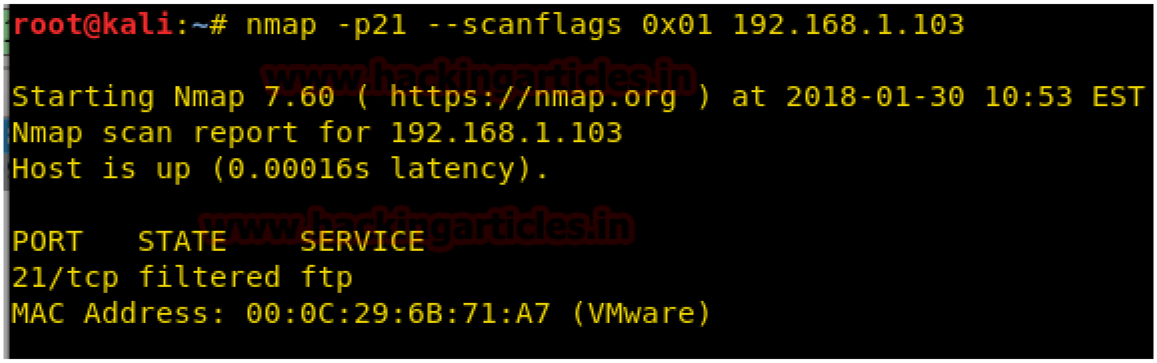
FIN Scan

The TCP-FIN flag is always used to finish the communication with the target network. In this scan, we are sending the FIN flag of the TCP by using its hexadecimal value on the target machine to enumerate the state of ports that are open, closed, or filtered.

Now execute the given below command for enumerating the state of any port. Here we want to identify the state of port 21.

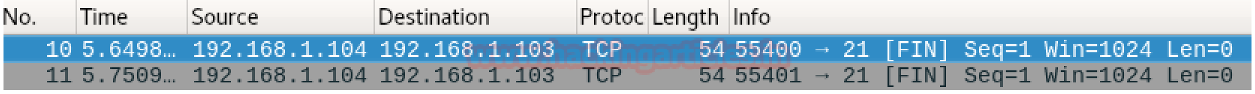
```
nmap -p21 --scanflags 0x01 192.168.1.103
```

From given below image you can observe we have found port 21 filtered.



```
root@kali:~# nmap -p21 --scanflags 0x01 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 10:53 EST
Nmap scan report for 192.168.1.103
Host is up (0.00016s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)
```

When a network admin captures the incoming traffic, he will get a packet for the TCP-FIN flag. Here we have used Wireshark for network **packet** analysis and found that it is showing a **TCP-FIN** packet for hex value **0x01** coming from 192.168.1.104 on port 21 as shown in the given below image.



```
No.  Time      Source        Destination   Protoc Length Info
 10 5.6498…  192.168.1.104 192.168.1.103  TCP       54 55400 → 21 [FIN] Seq=1 Win=1024 Len=0
 11 5.7509…  192.168.1.104 192.168.1.103  TCP       54 55401 → 21 [FIN] Seq=1 Win=1024 Len=0
```

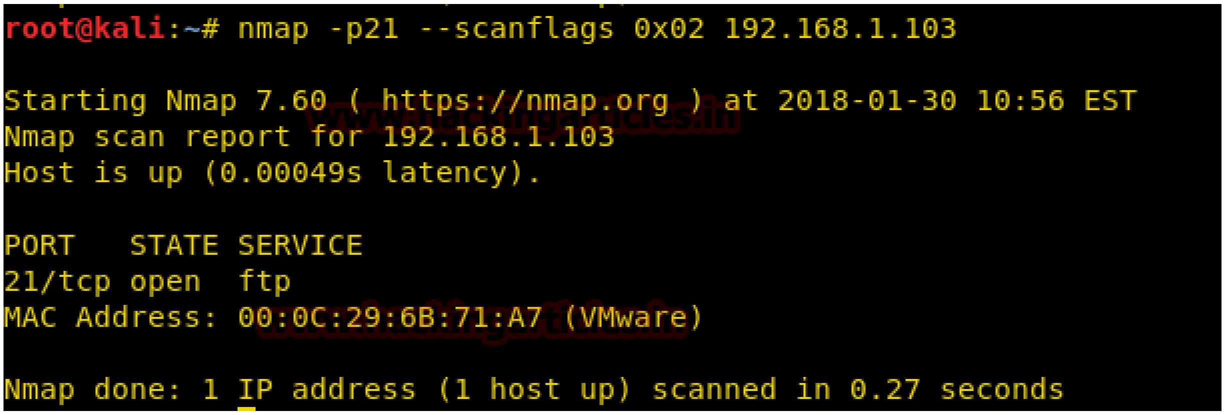
SYN Scan

The TCP-SYN flag always initiates communication to establish a connection with the target network. In this scan, we are sending the SYN flag of the TCP by using its hexadecimal value on the target machine to enumerate the state of ports that are open, closed, or filtered.

Now execute the given below command for enumerating the state of any port. Here we want to identify the state of port 21.

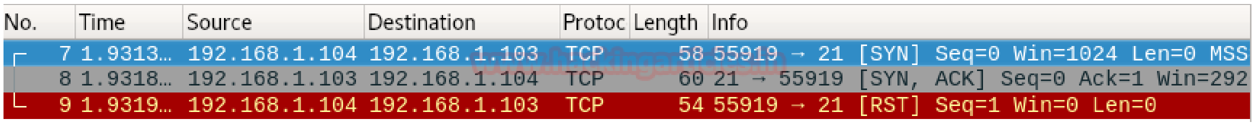
```
nmap -p21 --scanflags 0x02 192.168.1.103
```

From given below image you can observe we have successfully found port 21 open**.**



```
root@kali:~# nmap -p21 --scanflags 0x02 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 10:56 EST
Nmap scan report for 192.168.1.103
Host is up (0.00049s latency).

PORT   STATE SERVICE
21/tcp open  ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.27 seconds
```

When a network admin captures the incoming traffic, he will get a packet with the TCP-SYN flag. Here we have used Wireshark for network packet analysis and found that it is showing a **TCP-SYN packet** with hex value **0x02** coming from 192.168.1.104 on port 21 as shown in the given below image.



No.	Time	Source	Destination	Protoc	Length	Info
7	1.9313…	192.168.1.104	192.168.1.103	TCP	58	55919 → 21 [SYN] Seq=0 Win=1024 Len=0 MSS
8	1.9318…	192.168.1.103	192.168.1.104	TCP	60	21 → 55919 [SYN, ACK] Seq=0 Ack=1 Win=292
9	1.9319…	192.168.1.104	192.168.1.103	TCP	54	55919 → 21 [RST] Seq=1 Win=0 Len=0

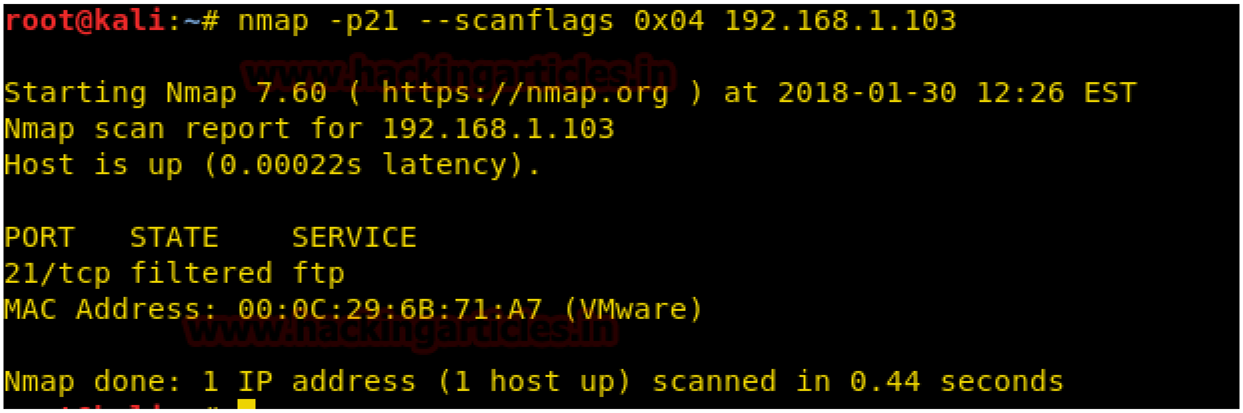
Reset Scan

The RST flag is used to reset the connection between the sender machine and the target machine. In this scan, we are sending the RST flag of the TCP by using its hexadecimal value on the target machine to enumerate the state of ports that are open, closed, or filtered.

Now execute the given below command for enumerating the state of any port. Here we want to identify the state of port 21.

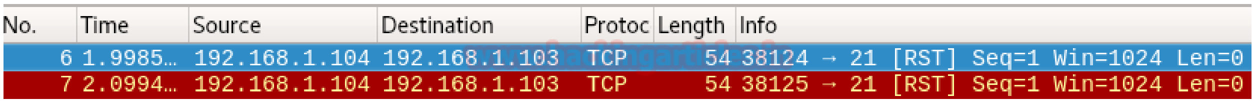
```
nmap -p21 --scanflags 0x04 192.168.1.103
```

From the given below image, you can observe that we have found port 21 filtered.



```
root@kali:~# nmap -p21 --scanflags 0x04 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 12:26 EST
Nmap scan report for 192.168.1.103
Host is up (0.00022s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds
```

When a network admin captures the incoming traffic, he will get a packet for the TCP-RST flag. Here we have used Wireshark for network packet analysis and we found that it is showing a **TCP-RST packet** for hex value **0x04** coming from 192.168.1.104 on port 21 as shown in the given below image.



No.	Time	Source	Destination	Protoc	Length	Info
6	1.9985…	192.168.1.104	192.168.1.103	TCP	54	38124 → 21 [RST] Seq=1 Win=1024 Len=0
7	2.0994…	192.168.1.104	192.168.1.103	TCP	54	38125 → 21 [RST] Seq=1 Win=1024 Len=0

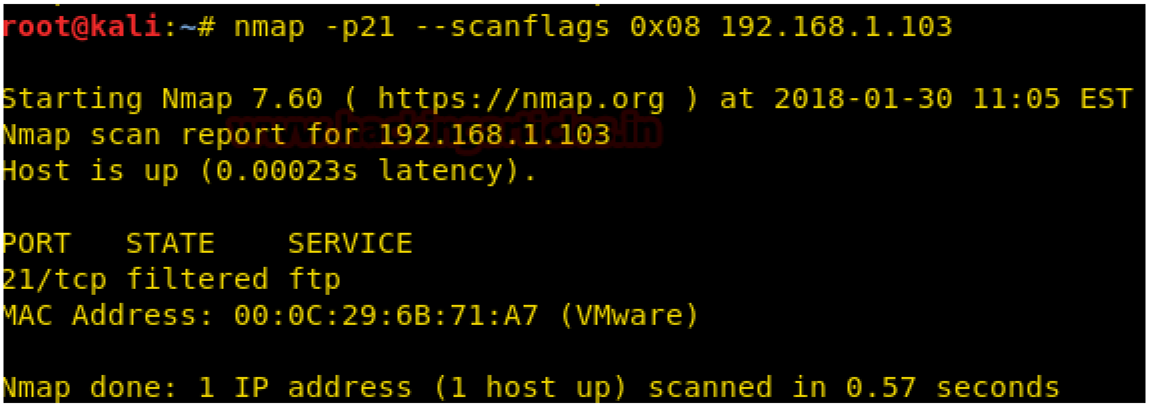
PUSH Scan

In this scan, we are sending the PSH flag of the TCP by using its hexadecimal value on the target machine to enumerate the state of ports that are open, closed, or filtered.

Now execute the given below command for enumerating the state of any port. Here we want to identify the state of port 21.

```
nmap -p21 --scanflags 0x08 192.168.1.103
```

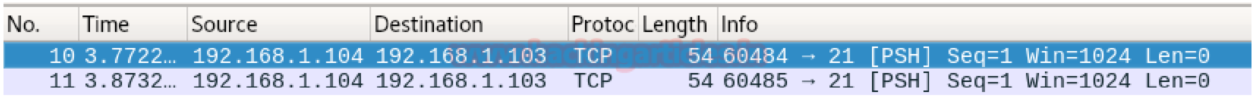
From the given below image, you can observe that we have found port 21 filtered.



```
root@kali:~# nmap -p21 --scanflags 0x08 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 11:05 EST
Nmap scan report for 192.168.1.103
Host is up (0.00023s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.57 seconds
```

When a network admin captures the incoming traffic, he will get a packet for the TCP-PSH flag. Here we have used Wireshark for network packet analysis and found that it is showing a **TCP-PSH packet** for hex value **0x08** coming from 192.168.1.104 on port 21 as shown in the given below image.

The PUSH flag is used to push the process priority of the packet to the target machine.



No.	Time	Source	Destination	Protoc	Length	Info
10	3.7722…	192.168.1.104	192.168.1.103	TCP	54	60484 → 21 [PSH] Seq=1 Win=1024 Len=0
11	3.8732…	192.168.1.104	192.168.1.103	TCP	54	60485 → 21 [PSH] Seq=1 Win=1024 Len=0

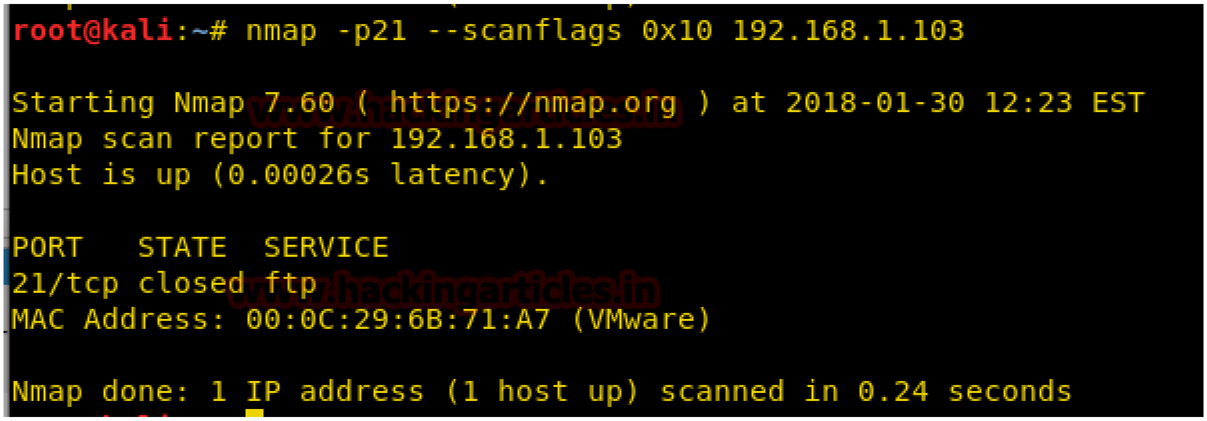
ACK Scan

An Ack flag is used to acknowledge the sender machine whether the packet was received or dropped by the target. So, the sender again sends the lost or dropped packet to the target network to complete the communication process. Here we are sending the ACK flag of the TCP by using its hexadecimal value on the target machine to enumerate the state of ports that are open, closed, or filtered.

Now execute the given below command to enumerate the state of any port. Here we want to identify the state of port 21.

```
nmap -p21 --scanflags 0x10 192.168.1.103
```

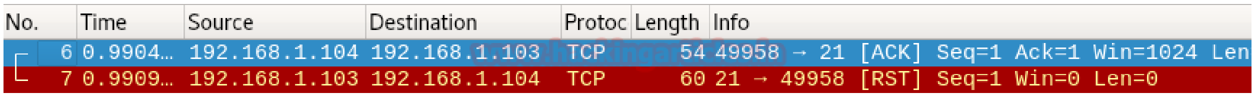
From the given below image, you can observe that we have found port 21 closed.



```
root@kali:~# nmap -p21 --scanflags 0x10 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 12:23 EST
Nmap scan report for 192.168.1.103
Host is up (0.00026s latency).

PORT   STATE  SERVICE
21/tcp closed ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.24 seconds
```

When network admin will capture the incoming traffic he will get a packet for TCP-ACK flag, here we have used Wireshark for network packet analysis and we found that it is showing **TCP-ACK packet** for hex value **0x10** coming from 192.168.1.104 on port 21 as shown in given below image.

Open and closed ports will both return an RST packet. Nmap then labels them as unfiltered, meaning that they are reachable by the ACK packet, but whether they are open or closed is undetermined. Ports that don’t respond, or send certain ICMP error messages back (type 3, code 0, 1, 2, 3, 9, 10, or 13), are labeled filtered. (**From Nmap.org**)



No.	Time	Source	Destination	Protoc	Length	Info
6	0.9904…	192.168.1.104	192.168.1.103	TCP	54	49958 → 21 [ACK] Seq=1 Ack=1 Win=1024 Len
7	0.9909…	192.168.1.103	192.168.1.104	TCP	60	21 → 49958 [RST] Seq=1 Win=0 Len=0

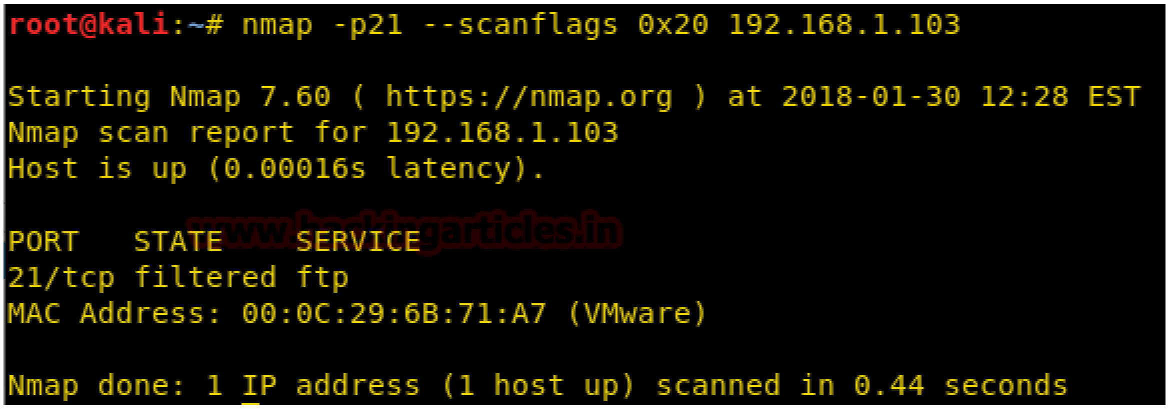
Urgent Scan

The URG flag is used to set the high process priority of the packet to the target. So that target machine stops processing the current packet and starts processing the URG Flag packet. In this scan, we are sending the Urg flag of the TCP by using its hexadecimal value on the target machine to enumerate the state of ports that are open, closed, or filtered.

Now execute the given below command for enumerating the state of any port. Here we want to identify the state of port 21.

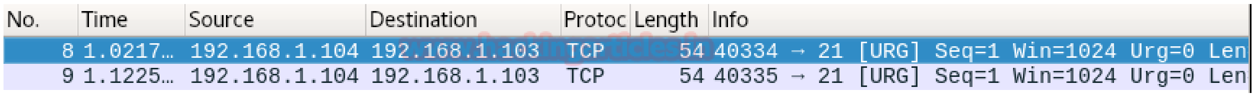
```
nmap -p21 --scanflags 0x20 192.168.1.103
```

From the given below image, you can observe that we have found port 21 filtered.



```
root@kali:~# nmap -p21 --scanflags 0x20 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 12:28 EST
Nmap scan report for 192.168.1.103
Host is up (0.00016s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds
```

When a network administrator captures incoming traffic, he will receive a packet with the TCP-URG flag. In this case, we used Wireshark for network packet analysis and discovered a **TCP-URG packet** with hex value **0x20** coming from 192.168.1.104 on port 21, as shown in the image below.

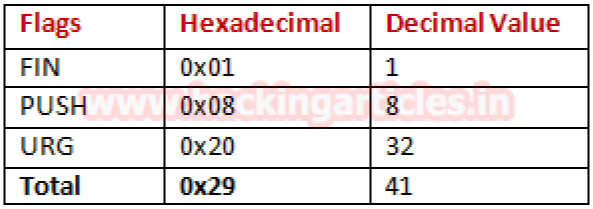


```
No.  Time      Source        Destination   Protoc Length Info
   8 1.0217…  192.168.1.104 192.168.1.103 TCP        54 40334 → 21 [URG] Seq=1 Win=1024 Urg=0 Len
   9 1.1225…  192.168.1.104 192.168.1.103 TCP        54 40335 → 21 [URG] Seq=1 Win=1024 Urg=0 Len
```

XMAS Scan

In this scan, we are sending the combination of the hexadecimal values of the different flags on the target machine. As we know, in the Xmas scan, a combination of three TCP-flags [FIN, PSH, URG] is used to enumerate the state of the port.

By adding the value of the flag, which is equal to the hexadecimal value of the sender's hexadecimal value, as described in the table below,

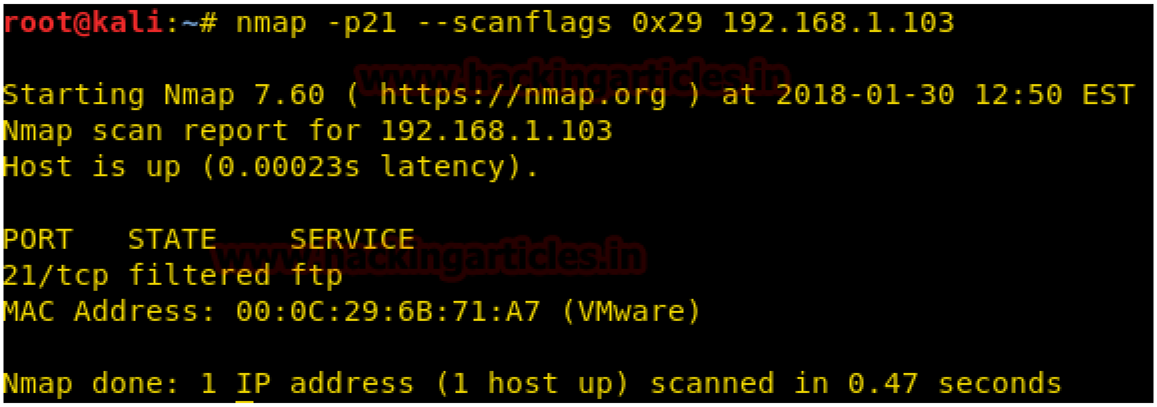


Flags	Hexadecimal	Decimal Value
FIN	0x01	1
PUSH	0x08	8
URG	0x20	32
Total	**0x29**	41

Now execute the given below command to enumerate the state of any port. Here we want to identify the state of port 21.

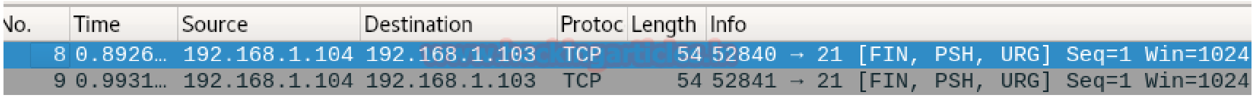
```
nmap -p21 --scanflags 0x29 192.168.1.103
```

From the given below image, you can observe that we have found port 21 filtered.



```
root@kali:~# nmap -p21 --scanflags 0x29 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 12:50 EST
Nmap scan report for 192.168.1.103
Host is up (0.00023s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.47 seconds
```

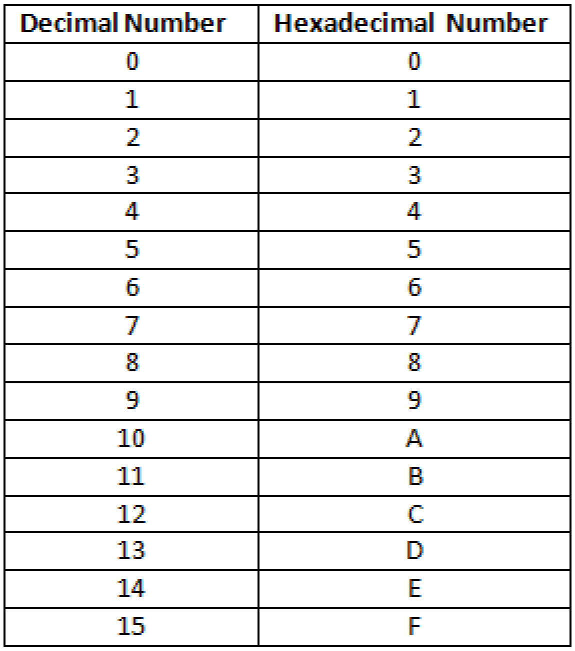
When a network admin captures the incoming traffic, he will get packets with TCP flags [FIN, PSH, URG] Here we have used Wireshark for network packet analysis and found that it is showing **TCP-packets** with **FIN, PSH, and URG** for hex value **0x29** coming from 192.168.1.104 on port 21 as shown in the given below image.



```
No. Time      Source        Destination   Protoc Length Info
  8 0.8926…  192.168.1.104 192.168.1.103 TCP        54 52840 → 21 [FIN, PSH, URG] Seq=1 Win=1024
  9 0.9931…  192.168.1.104 192.168.1.103 TCP        54 52841 → 21 [FIN, PSH, URG] Seq=1 Win=1024
```

Manual Combination of Flags [FIN, SYN, PSH]

Let have a quick review over decimal to hexadecimal conversion with the help of the following table:



Decimal Number	Hexadecimal Number
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

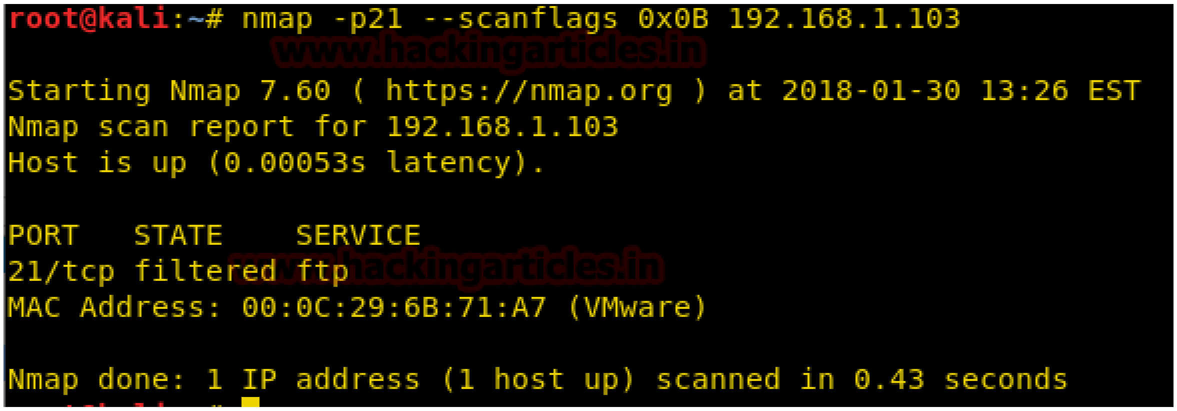
Now repeat the same methodology by changing the combination of the flag to enumerate the state of any port. For example, we want to scan any port by sending a combination of three flags [FIN, SYN, and PSH] so let identify hex value for the sum of three flags.

Flags	Hexadecimal	Decimal Value
FIN	0x01	1
SYN	0x02	2
PUSH	0x08	8
Total	**0x0B**	11

Now execute the given below command for enumerating the state of any port. Here we want to identify the state of port 21.

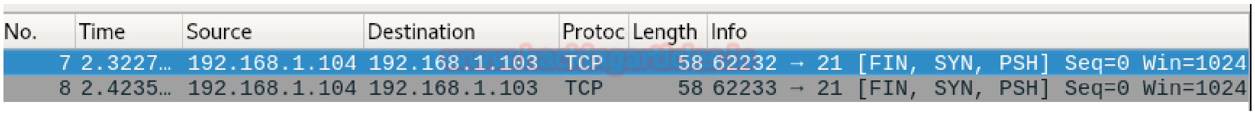
```
nmap -p21 --scanflags 0x0B 192.168.1.103
```

From the given below image, you can observe that we have found port 21 filtered.



```
root@kali:~# nmap -p21 --scanflags 0x0B 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 13:26 EST
Nmap scan report for 192.168.1.103
Host is up (0.00053s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.43 seconds
```

When a network admin captures the incoming traffic, he will get packets for TCP flags [FIN, SYN and PSH] Here we have used Wireshark for network packet analysis and found that it is showing **TCP-packets** for **FIN, SYN, and PSH** for hex value **0x0B** coming from 192.168.1.104 on port 21 as shown in the given below image.



```
No.  Time       Source         Destination    Protoc Length Info
   7 2.3227…  192.168.1.104 192.168.1.103  TCP      58 62232 → 21 [FIN, SYN, PSH] Seq=0 Win=1024
   8 2.4235…  192.168.1.104 192.168.1.103  TCP      58 62233 → 21 [FIN, SYN, PSH] Seq=0 Win=1024
```

Manual Combination of Flags [FIN, RST, PSH]

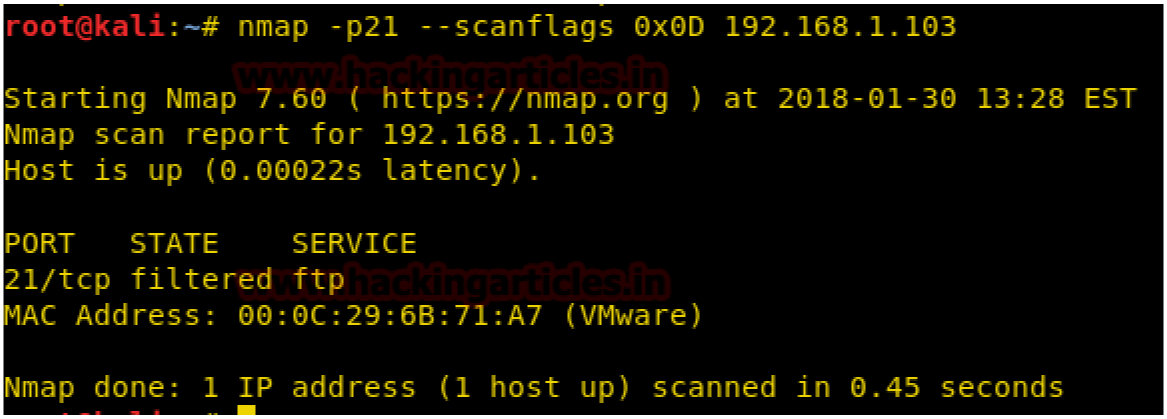
Now repeat the same methodology by changing the combination of the flag to enumerate the state of any port. For example, we want to scan any port by sending a combination of three flags [FIN, RST, and PSH] so let identify the hex value for the sum of three flags.

Flags	Hexadecimal	Decimal Value
FIN	0x01	1
RST	0x04	4
PUSH	0x08	8
Total	**0x0D**	13

Now execute the given below command for enumerating the state of any port. Here we want to identify the state of port 21.

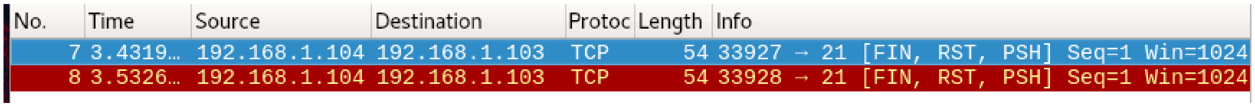
```
nmap -p21 --scanflags 0x0D 192.168.1.103
```

From the given below image, you can observe that we have found port 21 filtered.



```
root@kali:~# nmap -p21 --scanflags 0x0D 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 13:28 EST
Nmap scan report for 192.168.1.103
Host is up (0.00022s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.45 seconds
```

When network admin will capture the incoming traffic he will get packet for TCP flags [FIN, RST, and PSH] here we have used Wireshark for network packet analysis and we found that it is showing **TCP-packet** of **FIN, RST and PSH** for hex value **0x0D** coming from 192.168.1.104 on port 21 as shown in given below image.



```
No.  Time      Source        Destination   Protoc Length Info
   7 3.4319…   192.168.1.104 192.168.1.103 TCP        54 33927 → 21 [FIN, RST, PSH] Seq=1 Win=1024
   8 3.5326…   192.168.1.104 192.168.1.103 TCP        54 33928 → 21 [FIN, RST, PSH] Seq=1 Win=1024
```

Manual Combination of Flags [FIN, SYN, RST, PSH]

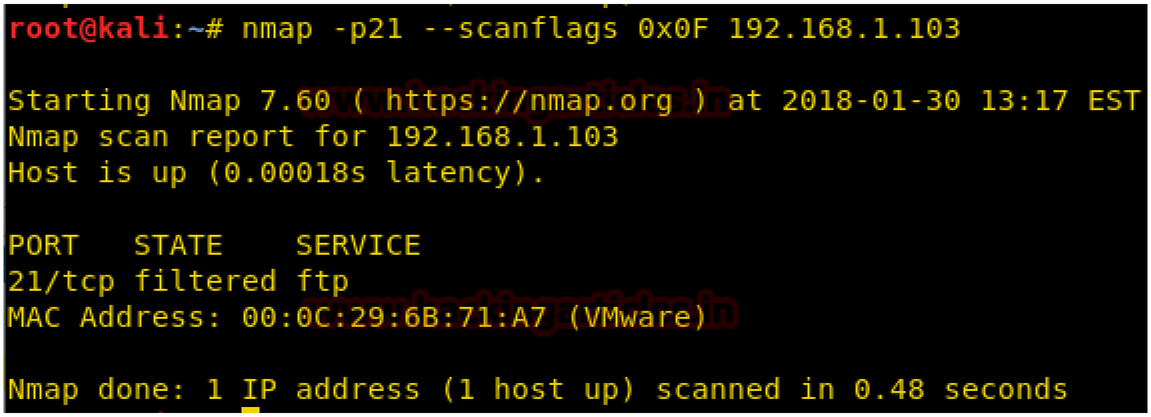
Now repeat the same methodology by changing the combination of the flag to enumerate the state of any port. For example, we want to scan any port by sending a combination of four flags [FIN, SYN, RST, and PSH] so let us identify the hex value for the sum of four flags.

Flags	Hexadecimal	Decimal Value
FIN	0x01	1
SYN	0x02	2
RST	0x04	4
PUSH	0x08	8
Total	**0x0F**	15

Now execute the given below command for enumerating the state of any port. Here we want to identify the state of port 21.

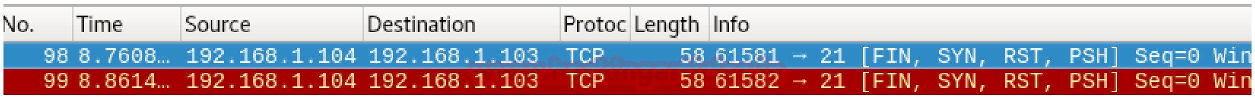
```
nmap -p21 --scanflags 0x0F 192.168.1.103
```

From the given below image, you can observe that we have found port 21 filtered.



```
root@kali:~# nmap -p21 --scanflags 0x0F 192.168.1.103

Starting Nmap 7.60 ( https://nmap.org ) at 2018-01-30 13:17 EST
Nmap scan report for 192.168.1.103
Host is up (0.00018s latency).

PORT   STATE    SERVICE
21/tcp filtered ftp
MAC Address: 00:0C:29:6B:71:A7 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.48 seconds
```

When a network admin captures the incoming traffic, he will get packets for TCP flags [FIN, SYN, RST, and PSH] Here we have used Wireshark for network packet analysis and we found that it is showing **TCP-packets** of **FIN, SYN, RST, and PSH** for hex value **0x0F** coming from 192.168.1.104 on port 21 as shown in the given below image.

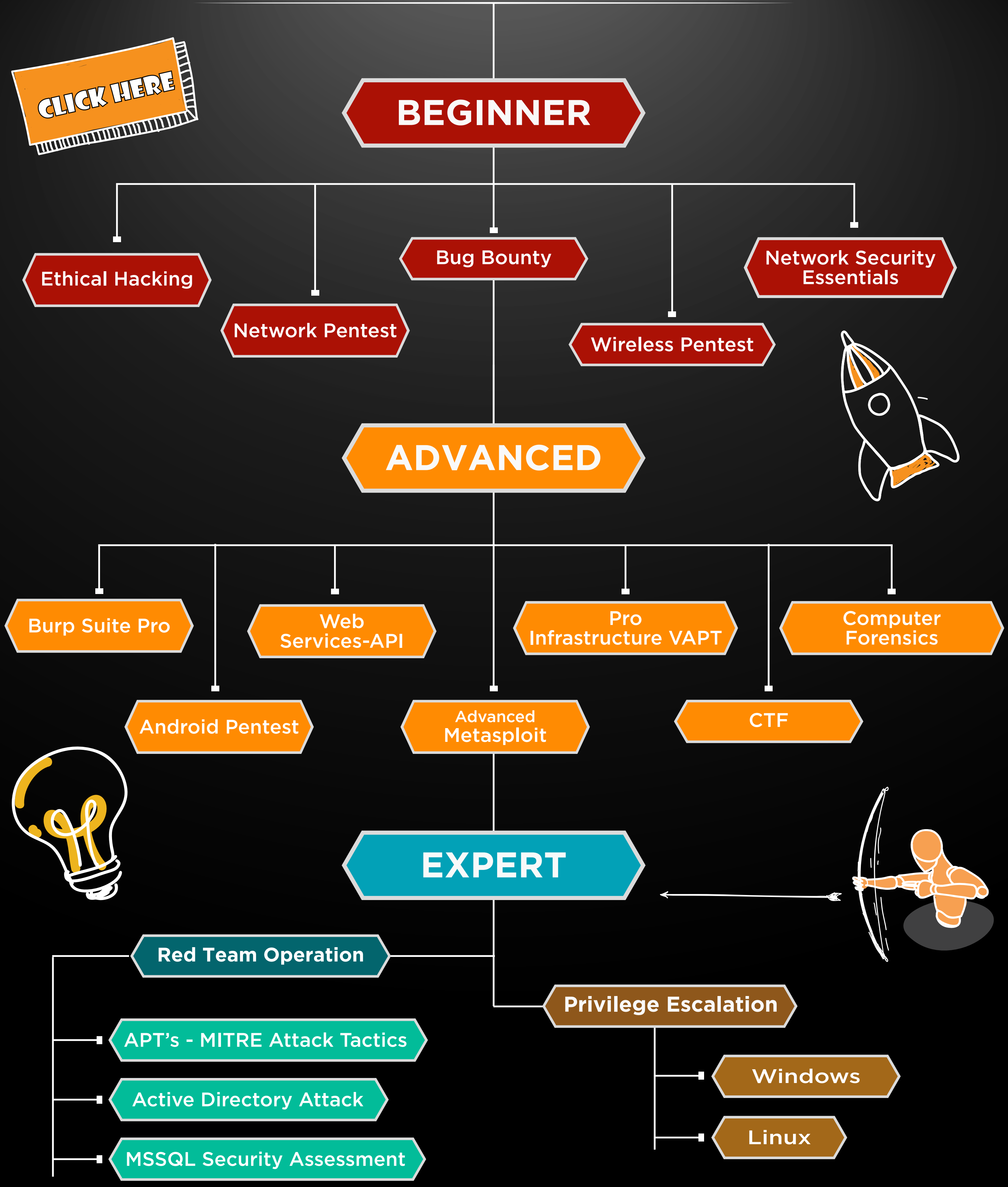


No.	Time	Source	Destination	Protoc	Length	Info
98	8.7608…	192.168.1.104	192.168.1.103	TCP	58	61581 → 21 [FIN, SYN, RST, PSH] Seq=0 Win
99	8.8614…	192.168.1.104	192.168.1.103	TCP	58	61582 → 21 [FIN, SYN, RST, PSH] Seq=0 Win

iGNITE Technologies

JOIN OUR TRAINING PROGRAMS

CLICK HERE



BEGINNER

- Ethical Hacking
- Network Pentest
- Bug Bounty
- Wireless Pentest
- Network Security Essentials

ADVANCED

- Burp Suite Pro
- Android Pentest
- Web Services-API
- Advanced Metasploit
- Pro Infrastructure VAPT
- CTF
- Computer Forensics

EXPERT

- Red Team Operation
  - APT’s - MITRE Attack Tactics
  - Active Directory Attack
  - MSSQL Security Assessment
- Privilege Escalation
  - Windows
  - Linux

www.ignitetechnologies.in