WIRESHARK ANALYSIS OF VARIOUS NMAP SCANS



Author Name: Aditya Srivastava

Twitter: <u>https://twitter.com/0xLegendKiller</u> LinkedIn: <u>https://www.linkedin.com/in/adityasrivastava0703/</u>

Basic Scan Types (-sT and -sS)

The two basic scan types used most in Nmap are TCP connect() scanning [-sT] and SYN scanning (also known as half-open, or stealth scanning) [-sS]

TCP connect() SCAN (-sT)

TCP connect scan will scan for TCP port like 21,22,23,445 etc. and ensures that ports are available for connection through a 3-way handshake connection between the source and destination.

This scan is very efficient, but the drawback is that it is very easy to detect due to modern firewalls or Intrusion detection system being present on server/victims' side.

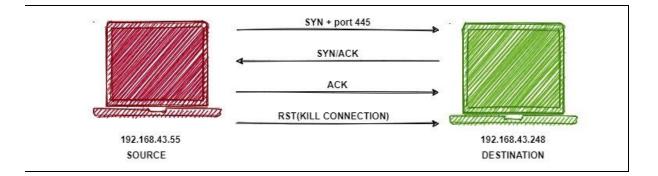
Advantage:

- 1) Results are highly accurate.
- 2) Handshake is complete, that ensures secure communication.

Disadvantage:

1) Very noisy, easy to detect.

For open ports



This involves mainly 4 steps - SYN, SYN/ACK, ACK and RST.

First 3 steps complete the handshake while the fourth one resets the connection. We will try to understand what happens at the data packet level and see how this scan works. Let's scan a port which is open and at the same time capture the packets in Wireshark for analysis.

Syntax

nmap -sT -p <port number> <destination IP>

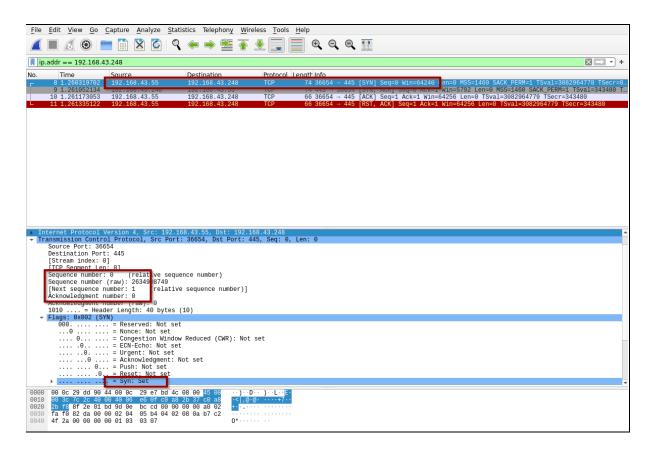
Scan command

```
nmap -sT -p 445 192.168.43.248
```

kali@kali:~\$ sudo nmap -sT -p 445 192.168.43.248
[sudo] password for kali:
Starting Nmap 7.80 (https://nmap.org) at 2020-08-19 19:46 IST
Nmap scan report for 192.168.43.248
Host is up (0.00088s latency).
PORT STATE SERVICE
445/tcp open microsoft-ds
MAC Address: 00:0C:29:DD:90:44 (VMware)
Nmap done: 1 IP address (1 host up) scanned in 0.41 seconds
kali@kali:~\$

Nmap tries the 3-way handshake and once the handshake is successful it declares the port is open.

Let us see what happened in the background when we ran TCP connect scan using nmap.



We added a Wireshark filter ip.addr == 192.168.43.248 to filter packets which deals with 192.168.43.248.

So total of 4 packets are captured, that's what we expected.

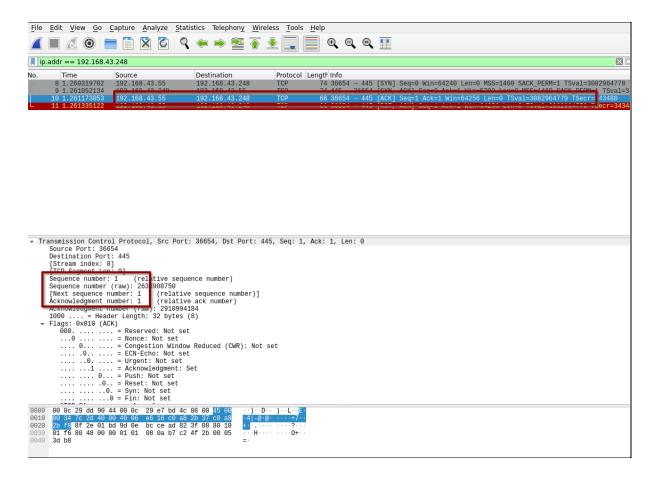
Let's analyze the first packet, we see that SYN packet is sent to the destination, in the flag section we see SYN flag is set.

Let's see if the server responds.

<u>F</u> ile	<u>E</u> dit <u>V</u> iew <u>G</u> o	<u>Capture Analyze Stat</u>	istics Telephony <u>W</u> in	eless <u>T</u> ools	<u>H</u> elp				
		= 🗋 🖹 🎑 🍳	🔶 🔿 🖉 🍝	₹.	• •	Q !!			
📕 ip.a	addr == 192.168.43	3.248							×⊃ •
		Source 192.168.43.55 192.168.43.248 192.108.43.55 192.168.43.55	Destination 192 168 43 248 192 168 43 55 192 108 43 248 192 168 43 248	Protocol I TCP TCP TCP TCP	74 36654 - 74 445 → 3 00 30054 -	6654 [SYN, AC 445 [ACK] Se	K] Seq=0 Ack=1 Win q=1 Ack=1 Win=6425	=5792 Len=0 MSS=14 6 Len=0 TSval=308	ERM=1 TSval=3082964778 TSecr=0 460 SACK PERM=1 TSval=343480 T 2964779 TSecr=343480 1=3082964779 TSecr=343480
			20. 40. 0.40. D. I. 400.	100 10 55					
👻 Tra	nsmission Contro	ersion 4, Src: 192.1 1 Protocol, Src Port			Ack: 1, Len:	Θ			
1	Source Port: 445 Destination Port [Stream index: 0 [TCP Segment Len	: 36654]							
	Sequence number [Next sequence n Acknowledgment n	0 (relaive sequ (raw): 2910)94183 umber: 1 (relativ umber: 1 (relativ umber: 1 2634988	e sequence number)] e ack number)						
	1010 = Head Flags: 0x012 (SY	er Length: 40 bytes N, ACK)	(10)						
	0 0 0 1	. = Reserved: Not set . = Nonce: Not set . = Congestion Window . = ECN-Echo: Not set . = Acknowledgment: S . = Push: Not set	Reduced (CWR): Not	set					
		. = Reset: Not set							
	00 3c 00 00 40 0 2b 37 01 bd 8f 2	Ic 00 0c 29 dd 90 44 00 40 06 62 3c c0 a8 2e ad 82 3f 07 9d 0e 00 02 04 05 b4 04 02	2b f8 c0 a8 ·<··@· bc ce a0 12 +7···	···)··D··E· @·b<··+··· ?····					

In the second packet the server sends an ACK (responds for SYN packet 1) and send SYN packet to the source for his validation. We can see the sequence number is 0 and Acknowledgement number is 1.

Now in packet 3 source should ACK the server.



In this step the source sends an ACK packet to the server and thus completes the 3-way handshake.

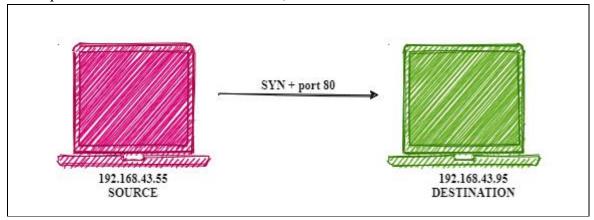
The last packet completes the connection and then resets it.

<u>F</u> ile <u>E</u>	dit <u>V</u> iew <u>G</u> o <u>C</u>	Capture <u>A</u> nalyze <u>S</u> tati	stics Telephony <u>W</u> i	reless <u>T</u> ools	<u>H</u> elp							
		🛯 🗋 🖹 🎑 🔍	< 🔿 🖉	₹.	€ €							
📕 ip.ad	ldr == 192.168.43	3.248										
1	Time 8 1.260319702 9 1.261052134 0 1.261173053 1 1.261335122	Source 192.168.43.55 192.168.43.248 192.168.43.55 192.168.43.55	Destination 192.168.43.248 192.168.43.55 192.168.43.248 192.168.43.248	Protocol TCP TCP TCP TCP	74 445 → 66 26654	36654	SYN, AC	eq=0 Win=64240 CK] Seq=0 Ack= CK] Seq=1 Ack=1 Win CK] Seq=1 Ack=	1 Win=5792 Ler	=0 MSS=1460 S/	ACK_PERM=1 TS	Sva1=343480
	<mark>smission Contro</mark> Durce Port: 3665 estination Port:	: 445			Ack: 1, Ler	n:0						
Se Se A	Next sequence nu cknowledgment nu cknowledgment nu	1 (relative seque (raw): 2634588750 umber: 1 (relative	sequence number)] ack number) 84									
		<pre>= Reserved: Not set = Nonce: Not set = Congestion Window = ECN-Echo: Not set = Arknowledgment' So = Push: Not set</pre>		: set								
0000	00 0c 29 dd 90 4 00 34 7c 2e 40 0	= Resel: Sel = Syn: Not set 4 00 0c 29 e7 bd 4c 0 40 06 e6 15 c0 a8 d 9d 0e bc ce ad 82	2b 37 c0 a8 ·4[.@ 3f 08 80 14 +···	D···)··L··E· @····+7··								
0030		0 01 01 08 0a b7 c2		···· 0+··								

Since, the 3-way handshake is completed therefore we know the particular port is open on the destination server.

For Filtered Port

A filtered port indicates the presence of firewall, or any other network issue/security device that is blocking the port, thus nmap cannot identify if it's open or closed. (The machine does not respond at all if firewall is enabled in it).

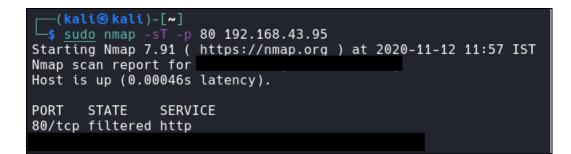


Syntax

nmap -sT -p <port number> <destination IP>

Nmap command:

nmap -sT -p 80 192.168.43.95

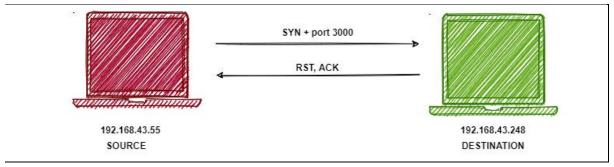


The scan shows port as filtered which signifies firewall is on. Let's analyses what comes up in Wireshark.

in a	ddr == 192.168.43											X
		-									_	
No.	Time 7 0.657314468	192.168.43.55	192.168.43.95	тср	74 44350 →	80 [SYN]	Seg=0 Win=	64240 Len=0	MSS=1460 SAC	K PERM=1 TSva	=3541816887	TSeci
	8 0.758876174	192.168.43.55	192.168.43.95	TCP	74 44352 →	80 [SYN]	Seq=0 Win=	64240 Len=0	MSS=1460 SAC MSS=1460 SAC	K_PERM=1 TSva	=3541816989	TSecr
	estination Port											
	Stream index: 0 TCP Segment Len											
	Sequence number:	0 (relative seque	nce number)									
1	Sequence number	(raw): 3170103020 umber: 1 (relative	soguence number)]									
1	cknowledgment n	umber: 0	sequence number)]									
	cknowledgment n		40)									
-	lags: 0x002 (SY	er Length: 40 bytes (N)	10)									
	000	= Reserved: Not set										
		. = Nonce: Not set . = Congestion Window	Reduced (CWR): Not s	et								
		. = ECN-Echo: Not set										
		. = Urgent: Not set . = Acknowledgment: No	nt set									
	0	_ = Push: Not set	500									
		- Resel: Not set										
		= Syn: Set										
	[TCP Flags: ··	s.]										

Only SYN packets were sent from source (our machine), the destination doesn't respond at all. Firewall simply drops the packets.

For Closed Port



Let's scan a closed port. **Syntax**

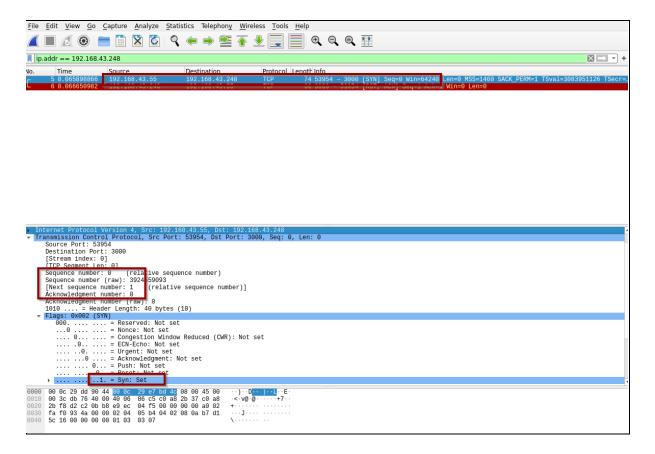
nmap -sT -p <port number> <destination IP>

Nmap scan command

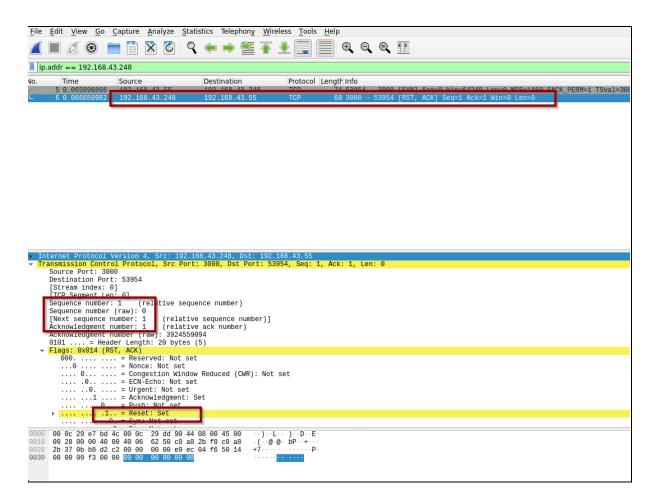
nmap -sT -p 3000 192.168.43.248

kali@kali:~\$ sudo nmap -sT -p 3000 192.168.43.248
[sudo] password for kali:
Starting Nmap 7.80 (https://nmap.org) at 2020-08-19 20:02 IST
Nmap scan report for 192.168.43.248
Host is up (0.00092s latency).
PORT STATE SERVICE
3000/tcp closed ppp
MAC Address: 00:0C:29:DD:90:44 (VMware)
Nmap done: 1 IP address (1 host up) scanned in 0.29 seconds
kali@kali:~\$

Since the port is closed 3-way handshake would not be completed. Let's analyze it in Wireshark.



First the source sends a SYN packet to the server.



And the server/destination sends an RST packet and resets the connections and thus 3-way handshake doesn't complete and therefore Nmap declares the port as closed.

TCP SYN (STEALTH) SCAN (-sS)

Stealth scan is one of the most popular scanning techniques. This technique is often referred as half-open scanning as this does not open full TCP connection.

In this we (source) send SYN packet to the destination and the server responds with SYN,

ACK packet, then we immediately terminate or kill the connection.

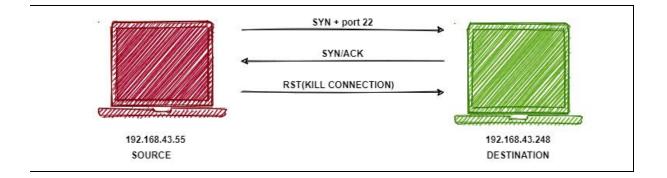
Advantage:

- 1) As the name suggest the scanning is stealth thus is not detected easily.
- 2) Faster than -sT because it doesn't complete the three-way handshake.

Disadvantage:

- 1) Can be used for malicious intent.
- 2) Handshake is not completed thus not authorized/secured.

For open port



Let's scan a port which is open and at the same time capture the packets in Wireshark for analysis.

Syntax

nmap -sS -p <port number> <destination IP>

Nmap command is

nmap -sS -p 22 192.168.43.248

kali@kali:~\$ sudo nmap -sS -p 22 192.168.43.248 Starting Nmap 7.80 (https://nmap.org) at 2020-08-20 07:51 IST Nmap scan report for 192.168.43.248 Host is up (0.00091s latency). PORT STATE SERVICE 22/tcp open ssh MAC Address: 00:0C:29:DD:90:44 (VMware) Nmap done: 1 IP address (1 host up) scanned in 0.53 seconds kali@kali:~\$

Port is open, lets analyze it in Wireshark.

	<u>E</u> dit	<u>V</u> iew <u>G</u> o	<u>C</u> apture	<u>A</u> nalyze	<u>S</u> tatistics	Telephony	Wireless	s <u>T</u> ools	<u>H</u> elp						
			03 03 03 10 03 13	🗙 🏠	۹ 🔶	۵	₹		Ð						
📕 ip	addr =	= 192.168	.43.248												
No.	Tir	ne	Course		Dect	ination		Drotocol	Longth Inf	•			_		
Г		109584439		8.43.55		168.43.24		ТСР						=0 MSS=1460	
		110470458 110532275		8.43.248		168.43.55		TCP TCP	60 22	→ 41149	[SYN, A	ACK] Seq=0 Seq=1 Win=0	Ack=1 Wi	in=5840 Len	=0 MSS=1460
-	70.	110032270	192.10	0.45.00	192.	100.43.24	ŧo	TUP	34 41	149 → ZZ	[Kol] :	sed-t will-d	0 Len-0		
▶ In	ternet	Protocol	Version	4 Src: 1	92 168 43	55 Det:	100 100								
T							192.168.4	43.248							
Ψ II			rol Proto	col, Src	Port: 4114	9, Dst Po	192.168.4 rt: 22, S	43.248 Seq: 0,	Len: 0						
+ II	Sourc	e Port: 4:	rol Proto 1149	col, Src	Port: 4114	9, Dst Po	192.168.4 ort: 22, 9	43.248 Seq: 0,	Len: 0						
+ II	Sourc Desti [Stre	e Port: 4: nation Po am index:	rol Proto 1149 rt: 22 0]	col, Src	Port: 4114	9, Dst Po	192.168.4 ort: 22, 8	43.248 Seq: 0,	Len: 0						
+ II	Sourc Desti [Stre [TCP	e Port: 4: nation Po am index: Segment Le	rol Proto 1149 rt: 22 0] en: 0]	col, Src	Port: 4114	9, Dst Po	192.168.4 ort: 22, S	43.248 Seq: 0,	Len: 0						
• II	Sourc Desti [Stre [TCP Seque	e Port: 4: nation Po am index: Segment Lo nce numbe	rol Proto 1149 rt: 22 0] en: 0] r: 0 (1	col, Src	Port: 4114 sequence n	9, Dst Po	192.168.4	43.248 Seq: 0,	Len: 0						
₹ <mark>II</mark>	Sourc Desti [Stre [TCP Seque Seque [Next	e Port: 4: nation Por am index: Segment L nce numbe nce numbe sequence	rol Proto 1149 rt: 22 0] en: 0] r: 0 (1 r (raw): 3 number: 3	col, Src relative 338876405 1 (rel	Port: 4114 sequence n	9, Dst Po umber)	rt: 22, S	43.248 Seq: 0,	Len: 0						
* <u>11</u>	Sourc Desti [Stre [TCP Seque Seque [Next Ackno	e Port: 4 nation Po am index: Segment L nce numbe nce numbe sequence wledgment	rol Proto 1149 rt: 22 0] r: 0 (1 r (raw): 3 number: 1 number: 0	col, Src relative 338876405 1 (rel	Port: 4114 sequence n 0	9, Dst Po umber)	rt: 22, S	43.248 Seq: 0,	Len: O						
* <u>11</u>	Sourc Desti [Stre [TCP Seque Seque [Next Ackno	e Port: 4 nation Po am index: Segment L nce numbe nce numbe sequence wledgment	rol Proto 1149 rt: 22 0] en: 0] r: 0 (1 r (raw): 3 number: 1 number: (col, Src relative 338876405 1 (rel aw): 0	Port: 4114 sequence n 0 ative sequ	9, Dst Po umber)	rt: 22, S	43.248 Seq: 0,	Len: 0						
	Sourc Desti [Stre ITCP Seque Seque [Next Ackno 0110 Flags	e Port: 4: nation Por am index: Segment Lu nce number sequence wledgment wledgment = Her : 0x002 (3	rol Proto 1149 rt: 22 0] en: 0] r: 0 (1 r (raw): 3 number: 1 number: (number: (ader Lengt SYN)	col, Src relative 338876405 1 (rel aw): 0 th: 24 by	Port: 4114 sequence n @ ative sequ tes (6)	9, Dst Po umber)	rt: 22, S	43.248 Seq: 0,	Len: 0						
	Sourc Desti [Stre [TCP Seque [Next Ackno 0110 Flags 000	e Port: 4: nation Poi am index:: Segment 14 nce numbel nce numbel sequence wledgment wledgment = Hea : 0x002 (: 0	rol Proto 1149 rt: 22 0] en: 0] r: 0 (1 r (raw): 3 number: 0 number: 0 number: 0 number: 0 syn) Syn) = Rese	col, Src relative 3388 76405 1 (rel aw): 0 th: 24 by erved: No	Port: 4114 sequence n @ ative sequ tes (6) t set	9, Dst Po umber)	rt: 22, S	43.248 Seq: 0,	Len: 0						
	Sourc Desti [Stre [TCP Seque Seque [Next Ackno 0110 Flags 000	e Port: 4: nation Poi am index: Sequent 14 nce number sequence wledgment wledgment = He: : 0x002 (0.	rol Proto 1149 rt: 22 0] ep: 0] r: 0 ((r r(raw): 3 number: 0 number:	relative 3388 76405 1 awy: 0 th: 24 by prved: Not se	Port: 4114 sequence n 0 ative sequ tes (6) t set et	9, Dst Po umber) ence numb	er)]	Seq: 0,	Len: 0						
	Sourc Desti [Stre ITCP Seque [Next Ackno 0110 Flags 000 	e Port: 4: nation Poi sequent L nce numbe sequence wledgment sequence wledgment : 0x002 (: 	rol Proto 1149 rt: 22 0] en: 0 (1 r (raw): 3 number: 1 number: 1 number: 2 number: 2 numb	col, Src elative 338876405 (rel aw): 0 th: 24 by erved: Not se: Not s Jestion W. Echo: No	Port: 4114 sequence n @ ative sequ tes (6) t set et indow Reduk t set	9, Dst Po umber) ence numb	er)]	Seq: 0,	Len: 0						
	Sourc Desti [Stre ITCP Seque Seque [Next Ackno 0110 Flags 000 	e Port: 4: nation Poi am index: Segment 10 nce number sequence wledgment wledgment : 0x002 (: .0	rol Proto 1149 rt: 22 0] r: 0 (1 r (raw): 3 number: 0 number: 1 number: 1 ader Lengt SYN) = Rese = Nonc = Cong = ECN-	relative 338 76405 4 (rel aw): 0 th: 24 by erved: Not se: Not s gestion W Echo: Not	Port: 4114 sequence n 0 ative sequ tes (6) t set et indow Reduc set	9, Dst Po umber) ence numb ced (CWR)	er)]	Seq: 0,	Len: 0						
	Sourc Desti [Stre ITCP Seque [Next Ackno Ackno 0110 Flags 000 	e Port: 4: nation Poi mindex: Segment L nce numbe sequence wledgment : 0x002 (3 0	rol Proto 1149 rt: 22 0] r: 0 (1 r (raw): 3 number: 1 number: 1 number: 1 number: 2 Number: 2 numbe	relative 338876405 1 (rel awy: 0 th: 24 by reved: Not se: Not s pestion W Echo: No ent: Not : Not : Not : Not	Port: 4114 sequence n 0 ative sequ tes (6) t set et set set nt: Not se t	9, Dst Po umber) ence numb ced (CWR)	er)]	Seq: 0,	Len: 0						
	Sourc Desti [Stre ITCP Seque [Next Ackno Ackno 0110 Flags 000 	e Port: 4: nation Por am index: Segment L sequence sequence sequence wledgment wledgment wledgment i 0.	rol Proto 1149 2149 0] 	relative 3388 76405 3388 76405 awy: 0 th: 24 by perved: No re: Not se isouledgme :: Not se t: Not se	Port: 4114 sequence n 0 ative sequ tes (6) t set et set set nt: Not se t	9, Dst Po umber) ence numb ced (CWR)	er)]	Seq: 0,	Len: 0						
	Sourc Desti [Stre ITCP Seque [Next Ackno Ackno 0110 Flags 000 	e Port: 4: nation Por am index: Segment L sequence sequence sequence wledgment wledgment wledgment i 0.	rol Proto 1149 rt: 22 0] r: 0 (1 r (raw): 3 number: 1 number: 1 number: 1 number: 2 Number: 2 numbe	relative 3388 76405 3388 76405 awy: 0 th: 24 by perved: No re: Not se isouledgme :: Not se t: Not se	Port: 4114 sequence n 0 ative sequ tes (6) t set et set set nt: Not se t	9, Dst Po umber) ence numb ced (CWR)	er)]	Seq: 0,	Len: 0						
•	Sourc Desti [Stre Seque [Next Ackno 0110 Flags 000 	e Port: 4: nation Poor Segment Lines Segment Lines sequence wledgment Mergiment Nergim	rol Proto 1149 1149 0 1149 0 1149 0 1149	relative 338376405 1 (rel awy: 0 th: 24 by rrved: Not se: Not s pestion W Echo: Not set: Not set: Not set: Not set 29 e7 b	Port: 4114 sequence n 0 ative sequ tes (6) t set et indow Reduc t set set ntos Reduc t set d 4c 08 00	19, Dst Po umber) ence numb ced (CWR) t	rt: 22, \$ er)] : Not set) LE .	Len: 0						
0000	Sourc Desti [Stre Seque [Next Ackno Ackno 0110 Flags 000 	e Port: 4: nation Por am index: Segment Line sequence wiedgment wiedgment : 0x002 (1 3. 	rol Proton 1149 1149 0]	relative 3388 76405 3388 76405 405 awy: 0 th: 24 by reved: Not s set Not s Set 29 e7 b 3a 29 c7 b	Port: 4114 sequence n 0 ative sequ tes (6) t set et indow Redut t set set nt: Not se t d 4c 08 00 0 a8 2b 37	9, Dst Po umber) ence numb ced (CWR) t t	rrt: 22, \$ er)] : Not set .,n#;) •• L •• E • ; ;) •• + 7 •	Len: 0						
•	Sourc Desti [Stre Seque [Next Ackno 0110 Flags 000 00 00 00 00 00 00 00 00 00 00 00 0	e Port: 4: nation Por am index: Segment Lince number sequence Wiedgment wiedgment 0.0	rol Proton 1149 1149 0]	col, Src elative 33876405 1 (rel 2 awy: 0 th: 24 by erved: Not setion: Not estion: W Echo: Not setion: Not setion: Not 29 e7 b 3a 29 c 6 f 92 0 6 f 92 0	Port: 4114 sequence n 0 ative sequ tes (6) t set et indow Reduc t set set ntos Reduc t set d 4c 08 00	45 00 45 000 45 0000 45 000000000000000000000000000000000000	rt: 22, \$ er)] : Not set))))))))))	Len: 0						

Let's analyze the first packet, we see that SYN packet is sent to the destination, in the flag section we see SYN is in Set mode.

Let's see if the server responds.

<u>F</u> ile	<u>E</u> dit <u>V</u> iew <u>G</u> o	<u>Capture</u> <u>A</u> nalyze	Statistics Teleph	non <u>y W</u> ireless	<u>T</u> ools <u>H</u> elp			
		📄 🔝 🛣 🍯	۹ 🗭 🏓	2 🛉 👱		ର୍ବ୍	11	
📕 ip.	addr == 192.168.4	43.248						
No.	Time	Source	Destination		tocol Length Info			
4	5 0.109584439	192 168 43 55	192 168 4				[SYN] Seg=0 Win=1024 Len=0 MSS=14	
L	6 0.110470458 7 0.110532275	192.168.43.248	192.168.4				[SYN, ACK] Seq=0 Ack=1 Win=5840 L [Noi] seq=1 Win=0 Len=0	en=0 155=1460
		Version 4, Src: :				0.01		
↓ 112	Source Port: 22	ol Protocol, Src	POIL: 22, DSL P	ort: 41149, Set	[: 0, ACK: ⊥, L	len: U		
	Destination Por							
_	[Stream index: [TCP Segment Le							
- F	Sequence number	: 0 (relitive)				
- 1	Sequence number [Next sequence	(raw): 251.26289 number: 1 (rel	06 Lative sequence r	umber)]				
- L	Åcknowledgment	number: 1 🛛 (re]	ative ack number					
		number (raw):`338 der Length: 24 by						
-	Flags: 0x012 (S	YN, ACK)						
		= Reserved: No = Nonce: Not s						
		= Congestion W		WR): Not set				
		= ECN-Echo: No						
		<pre> = Urgent: Not = Acknowledgme</pre>						
		= Push: Not se						
		= Reset: Not s 1. = Svn: Set	et					
0000	•		0 44 08 00 45 00	···)··L··)··	D. E			
0010	00 2c 00 00 40	00 40 06 62 4c d	:0 a8 2b f8 c0 a8	·,··@·@· bL·	· + · · ·			
0010 0020	00 2c 00 00 40 2b 37 00 16 a 0	00 40 06 62 4c d bd 95 be 16 f0 d	:0 a8 2b f8 c0 a8 :9 fc 6f 93 60 12	·,··@·@· bL· +7 <mark>·····</mark>	· · + · · ·			
0010	00 2c 00 00 40 2b 37 00 16 a 0	00 40 06 62 4c d	:0 a8 2b f8 c0 a8 :9 fc 6f 93 60 12	·,··@·@· bL·	· · + · · ·			

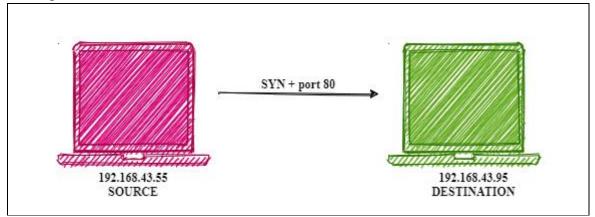
In the second packet the server sends an ACK (responds for SYN packet 1) and send SYN packet to the source for his validation. We can see the sequence number is 0 and Acknowledgement number is 1.

Next step is important, this step makes this scanning Stealthy, instead of sending ACK packets to destination, we send RST (reset) packet and the connection is killed. Since we received SYN and ACK packet from the destination, this means that the port is open.

File	Edit View Go	Capture Analyze	Statistics Telephony	Wireless Tools	Help			
					~ ~	~ #		
📕 ip.	addr == 192.168.4	3.248						
No.	Time 5 0.109584439	Source 192.168.43.55	Destination 192.168.43.2	Protocol L 48 TCP		22 FOVN1	Seq=0 Win=1024 Len=0 M	100-1460
<u> </u>	6 0.110470458	402 400 40 240	402.400.40.5	TOP	00 00 144	10 ECV/0	AGK1 Con O Aok 4 Min 5	40 Len=0 MSS=1460
L	7 0.110532275	192.168.43.55	192.168.43.2	48 TCP	54 41149 →	22 [RST]	Seq=1 Win=0 Len=0	
								_
→ In	ternet Protocol '	Version 4, Src: 1	.92.168.43.55, Dst:	192.168.43.248				
+ Ir	Source Port: 41:		Port: 41149, Dst Po	ort: 22, Seq: 1, L	en:⊍			
	Destination Port [Stream index: (
	TCD Sogmont Lo	ا م آه						
- 1	Sequence number Sequence number	: 1 (relative (raw): 338876405	sequence number) 1					
- 1	[Next sequence i	number:1 (rel	alive sequence numb	er)]				
- 1	Acknowledgment Acknowledgment	number (raw): 0						
	Flags: 0x004 (R	ter Length. 20 by	es (5)					
•	000	. = Reserved: No						
		. = Nonce: Not s = Congestion W	et indow Reduced (CWR)	: Not set				
		. = ECN-Echo: No	t set `´´	i not oot				
		. = Urgent: Not . = Acknowledgme						
		. = Push: Not se						
	···· ··· ···	Syn: Not set						
	00 0c 29 dd 90	44 00 0c 29 e7 b	d 4c 08 00 <mark>45 00</mark>	··)··D··)··L·· <mark>E·</mark>				
0010		00 40 06 62 50 c 16 c9 fc 6f 93 0		·(··@·@· bP··+7·· +······ 0·····P·				

For Filtered Port

A filtered port indicates the presence of firewall, or any other network issue/security device that is blocking the port, thus nmap cannot identify if it's open or closed. (The machine does not respond at all if firewall is enabled in it).



Syntax

nmap -sS -p <port number> <destination IP>

Nmap command:

nmap -sS -p 80 192.168.43.95

(kali⊛ kali)-[~] _ \$ <u>sudo</u> nmap -sS -p 80 192.168.43.95
Starting Nmap 7.91 (https://nmap.org) at 2020-11-12 11:53 IST
Nmap scan report for
Host is up (0.00043s latency).
PORT STATE SERVICE
80/tcp filtered http
MAC Address:
Nmap done: 1 IP address (1 host up) scanned in 0.53 seconds

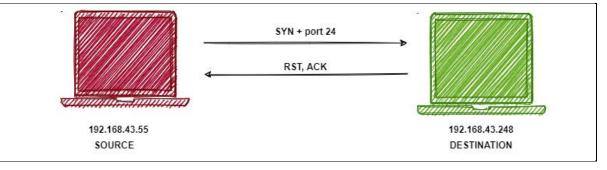
The scan shows port as filtered which signifies firewall is on. Let's analyses what comes up in Wireshark.

ip.	addr == 192.168.4	13.55						
	Time							
	10 2.102235058	192.168.43.55	192.168.43.95	TCP	58 48355 → 80 [S	VN1 Seg-0	Win-1024	Lon-0 MSS-1/
	11 2.204690829	192.168.43.55	192.168.43.95	TCP	58 48356 → 80 [S			
				~ ~ ~				
	Insmission Contr Source Port: 483		rt: 48355, Dst Port:	80, Seq: 0,	Len: 0			
	Destination Port							
	[Stream index: (
	[TCP Segment Ler							
		: 0 (relative sec	quence number)					
		(raw): 3474478607	ive sequence number)]					
	Acknowledgment i		rve sequence number)]					
	Acknowledgment i							
		der Length: 24 bytes	5 (6)					
,	Flags: 0x002 (S		()					
		. = Reserved: Not s	et					
		. = Nonce: Not set						
	0	= Congestion Wind	low Reduced (CWR): Not	t set				
		= ECN-Echo: Not s = Urgent: Not set						
		= Orgent: Not set = Acknowledgment:						
		= Push: Not set	NOL SEL					
		Pusit. Not set						
	I							

Only SYN packets were sent from source (our machine), the destination doesn't respond at all. Firewall simply drops the packets.

For closed ports

If the port is closed, then it means that we should receive an RST packet from the server in response to our SYN packet.



Let's scan a closed port.

Syntax

nmap -sS -p <port number> <destination IP>

Nmap scan command

nmap -sS -p 24 192.168.43.248

kali@kali:~\$ sudo nmap -sS -p 24 192.168.43.248 Starting Nmap 7.80 (https://nmap.org) at 2020-08-20 07:57 IST Nmap scan report for 192.168.43.248 Host is up (0.0015s latency). PORT STATE SERVICE 24/tcp closed priv-mail MAC Address: 00:0C:29:DD:90:44 (VMware) Nmap done: 1 IP address (1 host up) scanned in 0.52 seconds kali@kali:~\$

Let's analyze this scan in Wireshark.



🕶 Transmission Control Protocol, Src Port: 43286, Dst Port: 24, Seq: 0, Len: 0
Source Port: 43286
Destination Port: 24
[Stream index: 0]
TCP_Segment_Len [*] _0]
Sequence number: 0 (relative sequence number)
Sequence number (raw): 34224937
[Next sequence number: 1 (relative sequence number)]
Acknowledgment number: 0
Acknowledgment number (raw): 0
0110 = Header Length: 24 bytes (6)
✓ Flags: 0x002 (SYN)
000 = Reserved: Not set
0 = Nonce: Not set
0 = Congestion Window Reduced (CWR): Not set
0 = ECN-Echo: Not set
0 = Acknowledgment: Not set
•1 = Syn: Set
0030 04 00 3D 2a 00 00 02 04 05 D4 ···;*····
000 00 0c 29 d4 00 0c 29 d5 0d 4c 08 00 45 00 (1)

We sent a SYN packet to the server and we can see in flag that SYN is Set. Let's see the response from the server.

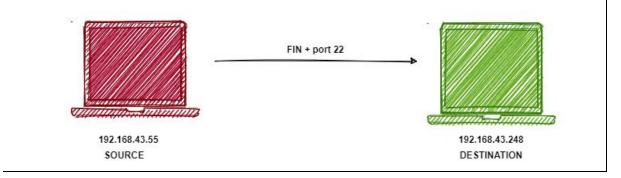
<u>File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help</u>
📶 🔳 🖉 🕲 🚞 🛅 🖹 🙆 I 🍳 🖛 🌩 警 春 速 📃 📃 🔍 🍳 🍳 🏗
ip.addr == 192.168.43.248
No. Time Source Destination Protocol Length Info 5 0.094050461 102.168.42.55 102.168.42.248 TCP 58.42286 24.[SYM] Sog=0. Win=1024. Lon=0. WSS=1460 L 6 0.094955727 192.168.43.248 192.168.43.55 TCP 60.24 → 43286 [RST, ACK] Seg=1. Ack=1. Win=0. Len=0.
 Internet Protocol Version 4, Src: 192.168.43.248, Dst: 192.168.43.55 Transmission Control Protocol, Src Port: 24, Dst Port: 43286, Seq: 1, Ack: 1, Len: θ
Source Port: 24 Destination Port: 43286
[Stream index: 0] ITCP Segment Len: 0]
Sequence number: 1 (rela ive sequence number) Sequence number (raw): 0
[Next sequence number: 1 (relative sequence number)] Acknowledgment number: 1 (relative ack number)
Acknowledgment number (raw): 344244938 0101 = Header Length: 20 bytes (5) - Flags: 0x014 (RST, ACK)
000 = Reserved: Not set 0
0 = COngestion Window Reduced (CWR): Not set 0 = ECN-Echo: Not set
0000 00 0c 29 e7 bd 4c 00 0c 29 dd 90 44 08 00 45 00 ······························
0020 20 31 00 18 19 10 0

We get an RST and ACK packet, as the port is closed.

TCP FIN SCAN (-sF)

This scan is simple. We send a FIN packet and if the port is open then there is no response from the server, if port is closed then the server responds with a RST, ACK packet. **Drawback**: Fin Scans only work for Linux machines and cannot be run against latest Windows machines.

For open port



Let's perform a FIN scan through nmap **Syntax**

```
nmap -sF -p <port number> <destination IP>
```

Nmap scan command

nmap -sF -p 22 192.168.43.248

kali@kali:~\$ sudo nmap -sF -p 22 192.168.43.248 Starting Nmap 7.80 (https://nmap.org) at 2020-08-20 08:43 IST Nmap scan report for 192.168.43.248 Host is up (0.00074s latency). PORT STATE SERVICE 22/tcp open|filtered ssh MAC Address: 00:0C:29:DD:90:44 (VMware) Nmap done: 1 IP address (1 host up) scanned in 1.05 seconds kali@kali:~\$

Let's analyze this scan through Wireshark.

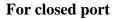
<u>File E</u> dit <u>V</u> iew <u>Go</u> <u>C</u> apture <u>A</u> nalyze <u>S</u> tatistics Telephony <u>W</u> ireless <u>T</u> ools <u>H</u> elp	
📶 🔳 🖉 🐵 🚞 🖺 🕅 🗳 I 🔍 👄 🔶 警 春 👱 🔜 🗐 🔍 Q, Q, 🏢	
ip.addr == 192.168.43.248	
No. Time Source Destination Protocol Length Info	
5 0.130019275 192.168.43.55 192.168.43.248 TCP 54 51004 → 22 [FIN] Seq=1 Win=1024 Len=0 6 0.234064789 192.108.43.55 192.108.43.248 ICP 54 51005 → 22 [FIN] Seq=1 Win=1024 Len=0	
6 0.234064789 192.108.43.55 192.108.43.248 ICP 54 51005 → 22 [FIN] Seq=1 W1N=1024 Len=0	
▶ Internet Protocol Version 4, Src: 192.168.43.55, Dst: 192.168.43.248	
 Transmission Control Protocol, Src Port: 51004, Dst Port: 22, Seq: 1, Len: 0 Source Port: 51004 	
Destination Port: 22	
Destination Port: 22 [Stream index: 0] [TCD Seguence Lon: 0] Sequence number: 1 (relative sequence number)	
Destination Port: 22 [Stream index: 0] [TCD Segment Len: 0] Sequence number: 1 (relative sequence number) Sequence number (raw): 4164097312	
Destination Port: 22 [Stream index: 0] [TCD Segmence number: 1 (relative sequence number) Sequence number: 1 (relative sequence number) Sequence number: 416407312 [Next sequence number: 2 (relative sequence number)] Acknowledgment number: 0	
Destination Port: 22 [Stream index: 0] (TCD Sequence number: 0] Sequence number: 1 (relative sequence number) Sequence number (raw): 4164097312 [Next sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number (raw): 0	
Destination Port: 22 [Stream index: 0] [TCD Segmence number: 1 (relative sequence number) Sequence number: 1 (relative sequence number) Sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number: 0 Acknowledgment number: 0 Flags: 0x001 (FIN)	
Destination Port: 22 [Stream index: 0] [TCD Sequence number: 0] Sequence number: 1 (relative sequence number) Sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number: 0 0101 = Header Length: 20 bytes (5) Flags: 0x001 (FIN) 000 = Reserved: Not set	
Destination Port: 22 [Stream index: 0] [TCD Segment Len: 0] Sequence number: 1 (relative sequence number) Sequence number: 1 (relative sequence number) Sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number (raw): 0 0101 = Header Length: 20 bytes (5) Flags: 0x001 (FIN) 000 = Reserved: Not set 0 = Nonce: Not set 0 = Congestion Window Reduced (CWR): Not set	
Destination Port: 22 [Stream index: 0] ITCD Seguence number: 1 (relative sequence number) Sequence number: 1 (relative sequence number) Sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number (raw): 0 0101 = Header Length: 20 bytes (5) Flags: 0x001 (FIN) 000 = Reserved: Not set 0 = Nonce: Not set 0 = COngestion Window Reduced (CWR): Not set 0 = ECN-Echo: Not set	
Destination Port: 22 [Stream index: 0] [TCD Segment Len: 0] Sequence number: 1 (relative sequence number) Sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number: 0 Acknowledgment number (raw): 0 0101 = Header Length: 20 bytes (5) Flags: 0x001 (FIN) 000 = Reserved: Not set 0 = Congestion Window Reduced (CWR): Not set 0 = ECN-Echo: Not set 0 = Urgent: Not set 0 = Urgent: Not set 0 = Urgent: Not set	
Destination Port: 22 [Stream index: 0] TCD Segment Len: 0] Sequence number: 1 (relative sequence number) Sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number: 0 Acknowledgment number (raw): 0 0101 = Header Length: 20 bytes (5) Flags: 0x001 (FIN) 000 = Reserved: Not set 0 = Reserved: Not set 0 = COngestion Window Reduced (CWR): Not set 0 = ECW-Echo: Not set 0 = Urgent: Not set 0 = Acknowledgment: Not set 0 = Push: Not set	
Destination Port: 22 [Stream index: 0] [TCD Segment Len: 0] Sequence number: 1 (relative sequence number) Sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number: 0 Acknowledgment number (raw): 0 0101 = Header Length: 20 bytes (5) Flags: 0x001 (FIN) 000 = Reserved: Not set 0 = Congestion Window Reduced (CWR): Not set 0 = ECN-Echo: Not set 0 = Urgent: Not set 0 = Urgent: Not set 0 = Urgent: Not set	
Destination Port: 22 [Stream index: 0] [TCD Sequence number: 1 (relative sequence number) Sequence number: 1 (relative sequence number)] Sequence number: 2 arelative sequence number)] Acknowledgment number: 0 Acknowledgment number (raw): 0 0101 = Header Length: 20 bytes (5) Flags: 0x001 (FIN) 0000 = Reserved: Not set 0 = Congestion Window Reduced (CWR): Not set 0 = Congestion Window Reduced (CWR): Not set 0 = ECN-Echo: Not set 0 = Length: Not set 0 = Acknowledgment: Not set 0 = Acknowledgment: Not set 0 = Syn: Not set 0 = Syn: Not set 0 = Syn: Not set 0 = Syn: Not set	
Destination Port: 22 [Stream index: 0] [TCD Segment Len; 0] Sequence number: 1 (relative sequence number) Sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number: 0 Acknowledgment number: 0 000 = Reserved: Not set = Reserved: Not set = Congestion Window Reduced (CWR): Not set 0 = Set 	
Destination Port: 22 [Stream index: 0] ITCD Seguence number: 1 (relative sequence number) Sequence number: 2 (relative sequence number)] Acknowledgment number: 0 Acknowledgment number (raw): 0 0101 = Header Length: 20 bytes (5) Flags: 0x001 (FIN) 0000 = Reserved: Not set 0 = Reserved: Not set 0 = Congestion Window Reduced (CWR): Not set 0 = Congestion Window Reduced (CWR): Not set 0 = ECN-Echo: Not set 0 = Acknowledgment: Not set 0 = Acknowledgment: Not set 0 = Acknowledgment: Not set 0 = Syn: Not set 0 = Syn: Not set 0 = Syn: Not set 0 = Syn: Not set	

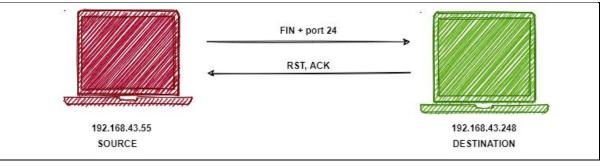
FIN packet is sent from the source , you can see the flag section(there's no SYN, ACK or RST packet SET)

The next packet is also sent from the source.

<u>F</u> ile	<u>E</u> dit	<u>V</u> iew <u>G</u> o	<u>C</u> apture	<u>A</u> nalyze	<u>S</u> tatistic	5 Telephor	<u>y W</u> ir	reless <u>T</u> o	ols <u>H</u> elp				
			0303	8	٩ <	• 🔿 🗳	1	₹.		Θ			
📕 ip	.addr	== 192.168	3.43.248										
No.		ime	Source			stination		Proto	ol Length In	fo			
		.13001927 .23406478		68.43.55		2.168.43.		TCP	54 51	L004 L005 →	22 [FIN] Seg=1 Win=1024 Len=0 FIN] Seg=1 Win=1024 Len=0	
		t Protoco											
+ Ir		ssion Cont ce Port: 5		ocol, Src	Port: 51	005, Dst I	ort: 2	22, Seq:	1, Len: 0				
		ination Po eam index:											
	ŤТСР	Segment	en 01										
- 1	Sequ	ence numbe ence numbe	er (raw):`		7								
- 1		t sequence owledgment			ative se	quence num	ber)]						
	ACKI	owreagment	number (raw): 0	tee (E)								
*	Flag	s: 0x001 (FIN)										
		00 0											
		0	= Con	gestion W	indow Re	duced (CWR): Not	: set					
		0 . 0				set							
			0 = Res	et: Not s	et								
	_			- · ·				.	-				
0000	00	0c 29 dd 9 28 c5 62 0	0 00 35 06	6 e7 ed o	0 a8 2b	37 c0 a8	·(·b·	D··)··L· ·5· ···+	7				
0020		f8 c7 3d 0 00 fe bb 0		2 15 21 0	0 00 00	00 50 01	+ • • = •	· · 2 · ! · · · ·	· P ·				
0000	04												

Again, it's SYN packet sent. We did not receive any packet from destination or server, but that's what this scan is about. If we don't receive any response from the server that means the port is open for connection.





Let's scan a closed port.

Syntax

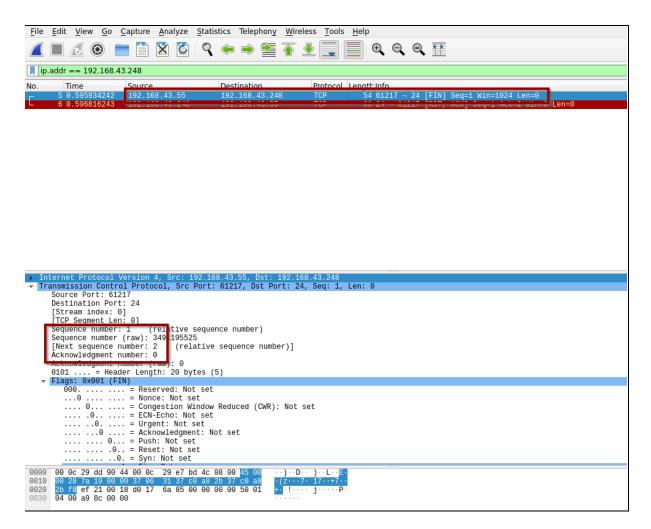
```
nmap -sF -p <port number> <destination IP>
```

Nmap scan command

```
nmap -sF -p 24 192.168.43.248
```

```
kali@kali:~$ sudo nmap -sF -p 24 192.168.43.248
[sudo] password for kali:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-08-20 08:32 IST
Nmap scan report for 192.168.43.248
Host is up (0.00091s latency).
PORT STATE SERVICE
24/tcp closed priv-mail
MAC Address: 00:0C:29:DD:90:44 (VMware)
Nmap done: 1 IP address (1 host up) scanned in 1.24 seconds
kali@kali:~$
```

Port is closed that means we should see a response from the server, let's verify it in Wireshark.



First packet is sent from us and it's a FIN packet.

<u>File Edit View Go Capture Analyze Statistics Telephony Wireless Tools H</u> elp
/ E 🖉 🐵 🚞 🛅 🖄 🙆 I 🔍 🗢 🔿 墅 春 速 📃 🗐 🍳 Q Q 🏢
ip.addr == 192.168.43.248
No. Time Source Destination Protocol LengthInfo
5 0.595934242 192 168 43 248 TCP 54 61217 → 24 [ETN] Seg=1 Win=1024 Len=0 L 6 0.596816243 192.168.43.248 192.168.43.55 TCP 60 24 → 61217 [RST, ACK] Seg=1 Ack=2 Win=0 Len=0
▶ Internet Protocol Version 4, Src: 192.168.43.248, Dst: 192.168.43.55 ▼ Transmission Control Protocol, Src Port: 24, Dst Port: 61217, Seq: 1, Ack: 2, Len: 0
Source Port: 24
Destination Port: 61217 [Stream index: 0]
Seguence number: 1 (relative sequence number)
Sequence number (raw): 0
[Next sequence number: 1 (felative sequence number)] Acknowledgment number: 2 (felative ack number)
Acknowledgment number (raw): 3491195526
0101 = Header Length: 20 bytes (5) - Flags: 0x014 (RST, ACK)
000 = Reserved: Not set 0 = Nonce: Not set
0 = Congestion Window Reduced (CWR): Not set
1 = Acknowledgment: Set
0000 00 0c 29 e7 bd 4c 00 0c 29 dd 90 44 08 00 45 0c ·······················
0010 00 28 00 00 40 00 40 06 62 50 c0 a8 2b f8 c0 a8 (···⊕·⊕· bP··+··· 0020 2b 37 00 18 ef 21 00 00 00 00 d0 17 6a 86 50 14 +7····!·····j·P·
0030 00 00 ad 78 00 00 00 00 00 00 00 00 00 00 00 00 00

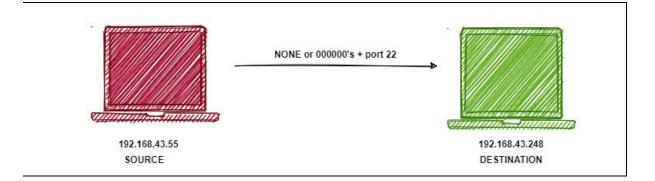
Since the port is closed the server sends the responds as RST (reset), ACK packet.

NULL SCAN (-sN)

In this scan the source sends TCP packets that contain a series of zeros "00000000", and since no flags are set , the destination doesn't know how to process the request and thus discards the packets. This means that port is open. If the server responds to the packet, then it means those ports are closed.

Drawback: Null Scans only work for Linux machines and cannot be run against latest Windows machines.

For open ports (we will see two packets sent when state of port is open as well as filtered)



Let's scan an open port and capture the packets in Wireshark. **Syntax**

nmap -sN -p <port number> <destination IP>

Nmap scan command

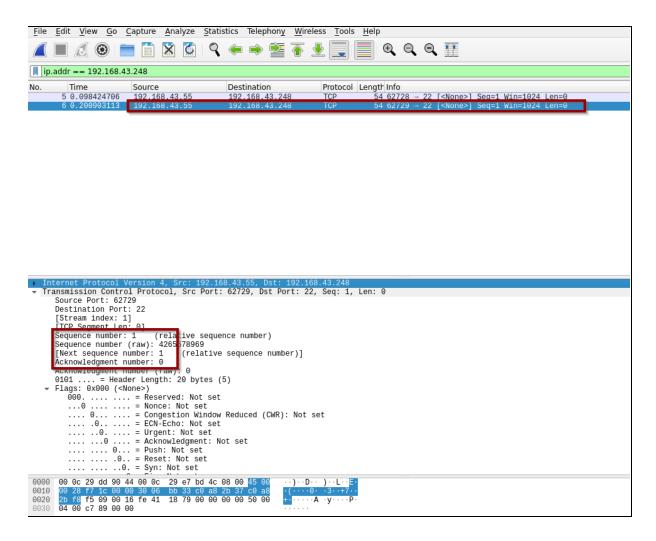
nmap -sN -p 22 192.168.43.248

kali@kali:~\$ sudo nmap -sN -p 22 192.168.43.248 Starting Nmap 7.80 (https://nmap.org) at 2020-08-20 09:17 IST Nmap scan report for 192.168.43.248 Host is up (0.00093s latency). PORT STATE SERVICE 22/tcp open|filtered ssh MAC Address: 00:0C:29:DD:90:44 (VMware) Nmap done: 1 IP address (1 host up) scanned in 0.73 seconds kali@kali:~\$

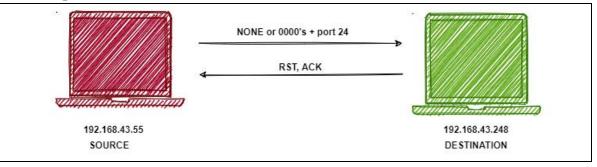
Port is open, this means we didn't get any response from the destination.

<u>File Edit View Go Capture Analyze Stat</u>	istics Telephon <u>y W</u> ire	eless <u>T</u> ools <u>H</u>	lelp	
🦲 🔳 🖉 🎯 📄 🛅 🕱 🎑 ۹	. 🗢 🔿 警 🖣) e , q	**
ip.addr == 192.168.43.248				
No. Time Source	Destination	Protocol Le		
5 0.098424706 192.168.43.55 6 0.200903113 402.169.43.55	192.168.43.248 102.168.43.248	TCP	54 62728	<pre>[<none>] Seq=1 Win=1024 Len=0 [<none>] Seq=1 Win=1024 Len=0</none></none></pre>
▶ Internet Protocol Version 4, Src: 192.1	68 43 55 Det: 102 1	68 43 248		
 Transmission Control Protocol, Src Port 			en: 0	
Source Port: 62728 Destination Port: 22				
[Stream index: 0] 				
Sequence number: 1 (relative sequence number (raw): 4265613432	ence number)			
[Next sequence number: 1 (relative	e sequence number)]			
Acknowledgment number: 0 Acknowledgment number (raw). 0				
0101 = Header Length: 20 bytes + Flags: 0x000 (<none>)</none>	(5)			
000 = Reserved: Not set 0 = Nonce: Not set	t			
0 = Congestion Window		set		
0 = ECN-Echo: Not set 0 = Urgent: Not set				
0 = Acknowledgment: M 0 = Push: Not set	Not set			
0 = Reset: Not set 0. = Syn: Not set				
0000 00 0c 29 dd 90 44 00 0c 29 e7 bd 4c	08 00 45 00 ···)··D)···)··L·· <mark>E·</mark>		
0010 00 28 ae f1 00 00 35 06 fe 5e c0 a8 0020 2b f8 f5 08 00 16 fe 40 18 78 00 00	2b 37 c0 a8 ·(····	5^+7 @ .xP.		
0030 04 00 c7 8c 00 00		8 A B		

Packet one is sent with no flags Set and we did not receive the response in second packet as that packet is also sent from the source.



For closed port



Let's scan a closed port and capture the packets in Wireshark.

Syntax

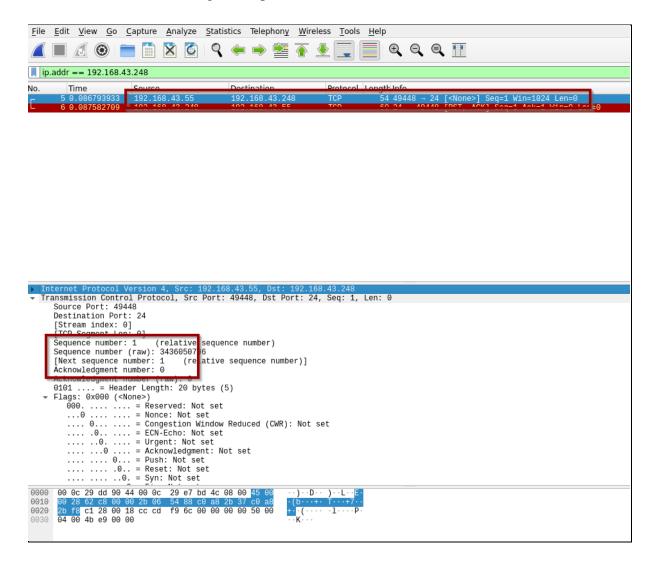
nmap -sN -p <port number> <destination IP>

Nmap scan command

```
nmap -sN -p 24 192.168.43.248
```

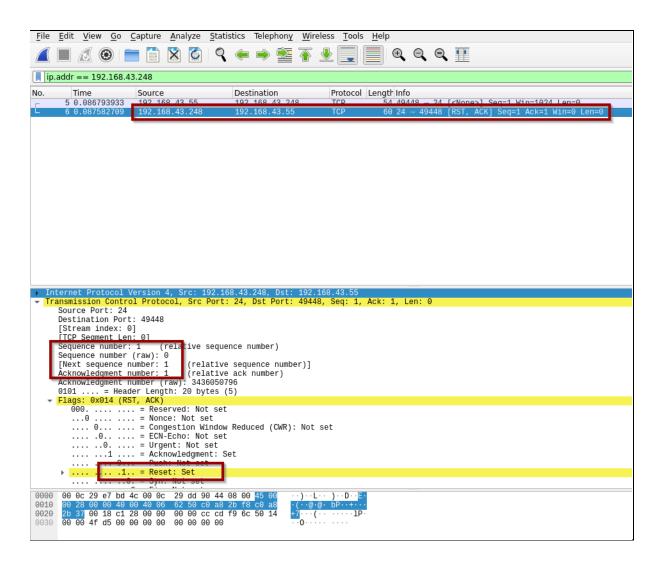
kali@kali:~\$ sudo nmap -sN -p 24 192.168.43.248 Starting Nmap 7.80 (https://nmap.org) at 2020-08-20 09:13 IST Nmap scan report for 192.168.43.248 Host is up (0.00085s latency). PORT STATE SERVICE 24/tcp closed priv-mail MAC Address: 00:0C:29:DD:90:44 (VMware) Nmap done: 1 IP address (1 host up) scanned in 0.57 seconds kali@kali:~\$

Port is closed, it means we got a response from the destination.



First packet is sent from the source with no flags.

In second packet we get a response from the destination as RST(reset), ACK packet.



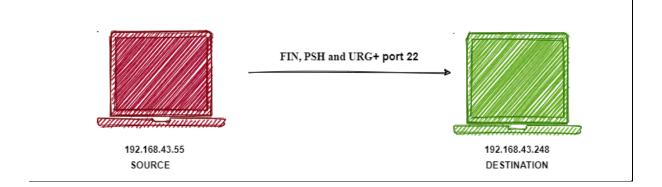
We got a reset flag from the destination which means port is closed.

TCP XMAS SCAN (-sX)

In this source send packets with flags like FIN, PSH and URG, this lights up the packet like a Christmas tree thus names XMAS scan, if the port is open then the destination doesn't respond and discards the packets. If the port is closed, then the destination responds with RSR (reset) and ACK packet.

Drawback: XMAS Scans only work for Linux machines and cannot be run against latest Windows machines.

For open ports



Let's scan an open port.

Syntax

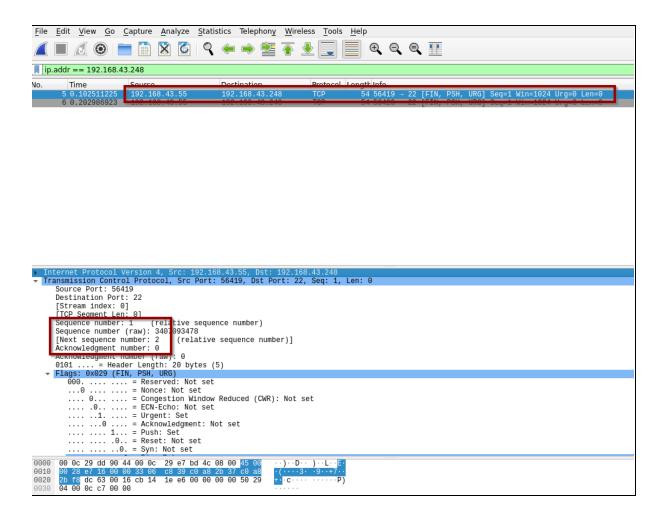
nmap -sX -p <port number> <destination IP>

Nmap scan command

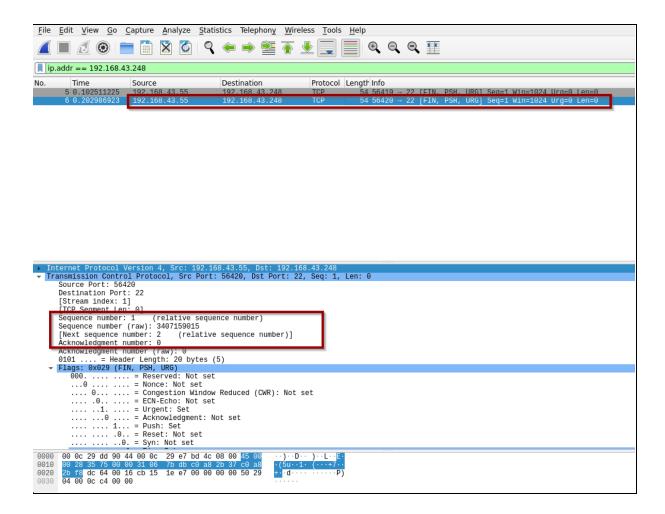
nmap -sX -p 22 192.168.43.248

kali@kali:~\$ sudo nmap -sX -p 22 192.168.43.248 Starting Nmap 7.80 (https://nmap.org) at 2020-08-20 08:59 IST Nmap scan report for 192.168.43.248 Host is up (0.00067s latency).
PORT STATE SERVICE 22/tcp open filtered ssh MAC Address: 00:0C:29:DD:90:44 (VMware)
Nmap done: 1 IP address (1 host up) scanned in 0.80 seconds kali@kali:~\$

Port is open, it means we do not get any response from the destination.

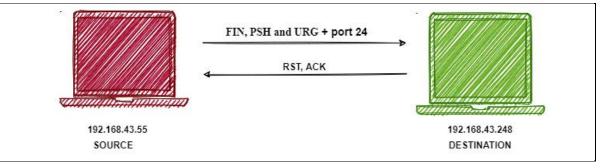


Source also send a second packet (as part of filtered port)



And still there is no response from the destination. This means that port is open for connection.

For closed port



Let's scan a closed port.

Syntax

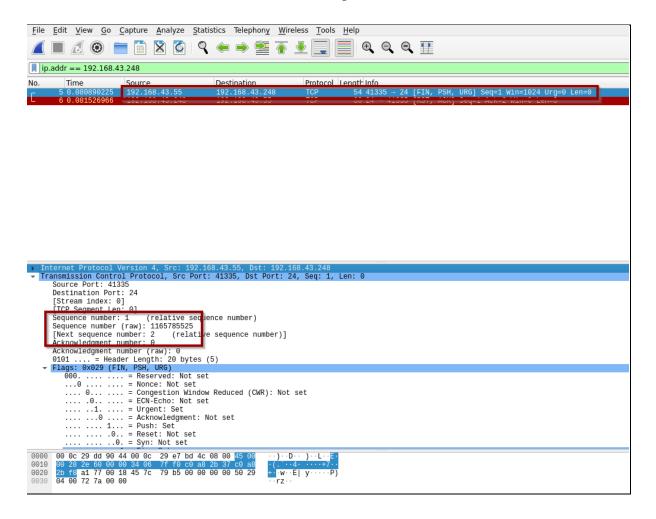
nmap -sX -p <port number> <destination IP>

Nmap scan command

nmap -sX -p 24 192.168.43.248

kali@kali:~\$ sudo nmap -sX -p 24 192.168.43.248 Starting Nmap 7.80 (https://nmap.org) at 2020-08-20 09:02 IST Nmap scan report for 192.168.43.248 Host is up (0.00100s latency). PORT STATE SERVICE 24/tcp closed priv-mail MAC Address: 00:0C:29:DD:90:44 (VMware) Nmap done: 1 IP address (1 host up) scanned in 0.29 seconds kali@kali:~\$

Port is closed which means destination send a response.



First packet contains FIN, PSH and URG flags. Since the port is closed the destination responds.

<u>F</u> ile <u>E</u> dit <u>V</u>	iew <u>G</u> o <u>C</u> apture <u>A</u> n	alyze <u>S</u> tatistics Telephony	<u>W</u> ireless <u>T</u> ools <u>H</u> elp			
	0 📄 🗋 🗙	🚳 🍳 🔶 🚔		Q Q 🗉		
ip.addr ==	192.168.43.248					
	e Source 80890225 <u>192 168 4</u> 81526966 <u>192.168.4</u>			1335 → 24 FETN	PSH_URG] Sen=1 Win=1024 Urn= ACK] Seq=1 Ack=2 Win=0 Len=0	=0_Len=0
		Src: 192.168.43.248, Dst: Src Port: 24, Dst Port:		len: 0		
Source	Port: 24 Ation Port: 41335		10007 0001 17 7000 27	2011 0		
[Stream	n index: 0]					
	eament Len: 01 ce number: 1 (rela	tive sequence number)				
Sequen	e number (raw): 0					
	equence number: 1	relative sequence number relative ack number)	r)]			
ACKNOW.	ledgment number (raw)	: 1165785526				
0101 .	<pre> = Header Length: 0x014 (RST, ACK)</pre>	20 bytes (5)				
000.	= Reserve					
	0 = Congest	Not set ion Window Reduced (CWR):	Not set			
	.0 = ECN-Ech	o: Not set	NOT SET			
	0 = Urgent: 1 = Acknowl					
	0 Puch: N	ot oot				
) · · L · ·) · · D · · E ·			
0010 00 28 0020 2b 37	00 00 40 00 40 06 62	2 50 c0 a8 2b f8 c0 a8 ·(0 00 45 7c 79 b6 50 14 +7	···@·@·bP··+···			

UDP SCAN (-sU)

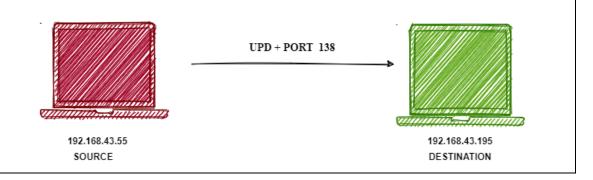
This scan works by sending UDP packets to the destination port, if the target port is open then there is no response, if port is closed then destination sends an ICMP packet saying unreachable.

This scan is also called connection less protocol.

Advantage:

- 1) More control over data sent out.
- 2) UPD header is 20 bytes while TCP header is 80 bytes.

For open port



Scan an open port and capture the packets in Wireshark. **Syntax**

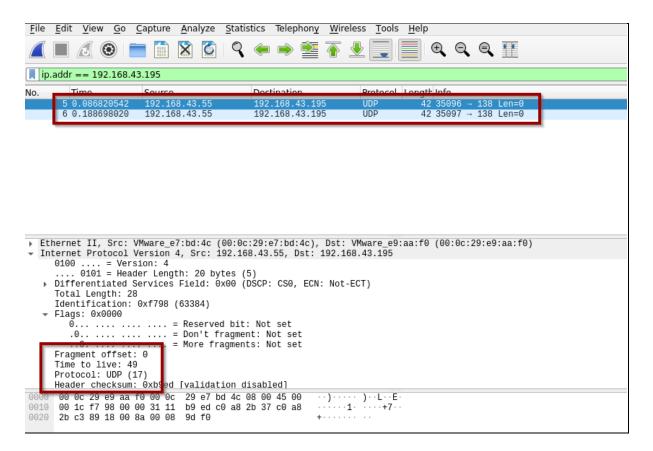
nmap -sU -p <port number> <destination IP>

Nmap scan command

nmap -sU -p 138 192.168.43.195

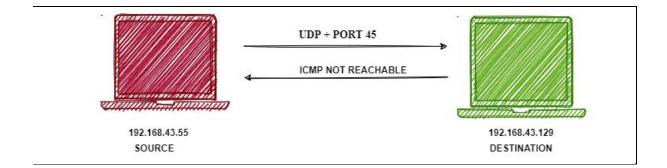
kali@kali:~\$ sudo nmap -sU -p 138 192.168.43.195
Starting Nmap 7.80 (https://nmap.org) at 2020-08-27 15:35 IST
Nmap scan report for IRONMAN (192.168.43.195)
Host is up (0.00047s latency).
PORT STATE SERVICE
138/udp open|filtered netbios-dgm
MAC Address: 00:0C:29:E9:AA:F0 (VMware)
Nmap done: 1 IP address (1 host up) scanned in 0.59 seconds
kali@kali:~\$

Let's see the Wireshark.



We can see that the server is not responding to our UDP packet, in the flag section we can see that protocol used is UDP.

For closed port



Let's scan a closed port through nmap and analyze the packets through Wireshark. **Syntax**

nmap -sU -p <port number> <destination IP>

Nmap command

nmap -sU -p 45 192.168.43.129

kaligkali:~\$ sudo nmap -sU -p 45 192.168.43.129 Starting Nmap 7.80 (https://nmap.org) at 2020-08-27 16:12 IST Nmap scan report for owaspbwa (192.168.43.129) Host is up (0.00088s latency). PORT STATE SERVICE 45/udp closed mpm MAC Address: 00:0C:29:DE:AD:CC (VMware) Nmap done: 1 IP address (1 host up) scanned in 0.32 seconds kaligkali:~\$

Port is closed, let's see Wireshark.

	t <u>V</u> iew <u>G</u> o	<u>C</u> apture <u>A</u> na	alyze <u>S</u> tatis	tics Telepho	n <u>y W</u> ireless	s <u>T</u> ools <u>H</u> elp	1				
	60	i	(أ 🗢 🗢	è 🖣 🛓		€€	€	3 0		
📕 ip.add	r == 192.168.	43.129									
	Time	Source		Destination		Protocol Lenat					
	0.001637015	192.168.43		192.168.43			0 51318	_		. Fort unre	achable)
	0.840191258	192.168.43		192.168.43			2 56825		en=0		achabie)
14	0.841275405	192.168.43	3.129	192.168.43	. 55	ICMP 7	0 Destir	ation	unreachabl	e (Port unre	achable)
Eramo	7: 70 bytos	on wire (560	hits) 70	bytes capti	unod (560 bi	its) on intor	faco ot	hQ id	0		
Ethern	net II, Śrc:	IntelCor_63:	61:9f (1c:	1b:b5:63:61:	:9f), Dst: \	/Mwáre_de:ad:					
		Version 4, S	rc: 192.16	8.43.95, Dst	:: 192.168.4	13.129					
	0 = Ver	ader Length:	20 bytes (S	5)							
▶ Dif	ferentiated	Services Fie			CN: Not-ECT)					
Tot	al Length: §	56 : 0x4219 (169	21)								
Tdo			21)								
	gs: 0x0000	0x4219 (109									
. . Fla	gs: 0x0000 0	= Re									
Fla	gs: 0x0000 0	= Re = Do	n't fragmer	nt: Not set							
✓ Fla	gs: 0x0000 0	= Re = Do = Mo	n't fragmer								
✓ Fla Fra Tim	gs: 0x0000 0	= Re = Do = Mo 	n't fragmer	nt: Not set							
✓ Fla Fra Tim Pro	gs: 0x0000 0 .0 gment offset e to live: f tocol: UDP	= Re = Do = Mo 	n't fragmer re fragment	nt: Not set s: Not set							
 ✓ Fla Fra Fra Tim Pro 0000 00 00 00 	gs: 0x0000 0 gment offset e to live: 2 tocol: UDP 0C 29 de ad 38 42 19 00	= Re = Do = Mo = Mo = Mo 17) = Mo 17) = Mo 17) = Mo 128 17) = Re = Do = Mo	n't fragmen re fragment 6 63 61 9f 0 6 6b c0 a8 2	nt: Not set ss: Not set 08 00 45 00 2b 5f c0 a8	· ·) · · · · · · · · · · · · · · · · ·	k··+_··					
 ✓ Fla Fra Tim Pro 0000 0010 0000 0020 2b 	gs: 0x0000 0 gment offset tocol: UDP (0C 29 de ad 38 42 19 00 81 c8 76 08	= Re = Do = Mo 128 17) CC 1C 1b b5 00 80 11 20 06 00 24 3d	n't fragmen re fragment 6 63 61 9f 0 6 60 c0 a8 2 1 a2 00 01 0	nt: Not set ss: Not set 08 00 45 00 2b 5f c0 a8 08 00 06 04	·8É···· +··v··\$ ≕	k··+_·· =·					
 Fla Fra Fra Fra Pro 0000 000 0000 0000	gs: 0x0000 0 gment offset tocol: UDP (0C 29 de ad 38 42 19 00 81 c8 76 08	= Re = Do = Mo = Mo = Mo = Mo = Mo = Mo = Mo = Mo = 0 = Mo = 0 = 0 = 0 	n't fragmen re fragment 6 63 61 9f 0 6 60 c0 a8 2 1 a2 00 01 0	nt: Not set ss: Not set 08 00 45 00 2b 5f c0 a8 08 00 06 04	·8É····	k··+_·· =·					

First UDP packet is sent by us, since the port is closed, we should receive ICMP packet from the server.

<u>F</u> ile <u>E</u> di	t <u>V</u> iew	<u>G</u> o	<u>C</u> apture	<u>A</u> nalyze	<u>S</u> tatisti	cs Tele	phon <u>y</u>	Wire	eless <u>T</u> oo	ols <u>H</u> elp)						
			0103 0310 0313	8	٩.	(ب) 🗳				Ð	Q					
📕 ip.add	r == 192	.168.4	3.129														
No.	Time	7045	Source	0 40 05	[Destinati	ion		Protoc	ol Leng	th Info)	005				
	0.00163			8.43.129 8.43.55		192.168			ICMP		8 Des		tion	unreachabl	.e (Por	t unread	chable)
	0.84127			8.43.129		192.168			ICMP		-		tion	unreachab]	.e (Por	t unread	chable)
▶ Frame	8 · 98 h	vtes r	n wire	(784 bits	s) 98 h	ovtes ca	anture	d (78	4 hits)	on inte	rface	eth	hi (0			
Etherr	et II,	Śrc: V	/Mware_d		(00:0c:2	29:de:a	d:cc),	Dst:	IntelĆo	r_63:61				63:61:9f)			
010	0 =	= Vers	ion: 4	th: 20 by			, 530.	1921	100.40.5	,							
▶ Dif		ated S	Services	Field: 0			3, ECN	: Not	-ECT)								
Ide		tion:	0x86b1	(34481)													
	ŏ			Reserve													
				= Don't f = More fr													
Tim	gment of	ve: 64															
Pro	tocol: 1	LCWH (1)														

We can see the server responds with ICMP packet, and tells us that destination is unreachable.

Idle Scan /Zombie Scan (-sI)

This scan is far more complex as compared to other scans. Idle scan allows complete blind port scanning. The scan is carried out using another host known as "zombie host". This is somewhat similar to identity theft. The idea is to prevent the attackers IP address to be logged in victims Intrusion Detection System (IDS). This scan involves an attacking machine, a target, a zombie.

Basic understanding: -

- 1) TCP scan involves attacker sending SYN scan to determine if open is open or not. If port is open the target machine sends SYN/ACK packet, else sends an RST packet.
- 2) Machine that receives an unsolicited SYN/ACK packet will responds with RST packet.
- 3) Each IP packet has a fragment Identity Number (IP ID), each packet sent results in increment if IP ID, probing/inspecting these IP ID the attacker can tell how many packets have been sent since last probe. This also allows attacker to forge his identity and perform blind scan.

This scan is carried out in three steps: -

- 1) Probe the zombie IPID and note it.
- 2) Forge a SYN packet from the zombie and send it to the desired port to the target. The target will respond on basis of open or close port which may or may not cause the zombie IPID to be incremented (Increase in IPID is because of SYN/ACK packet received by zombie and encouraging the zombie to send RST packet).
- 3) Probe the zombie IPID again. The port state is determined by comparing the current IPID with the probe IPID record in step 1.

Finding zombie/idle machine:

Nmap is a very power full tool. It becomes even more power full when you use nmap scripts (.nse) effectively. This script detects vulnerable zombie devices. Idle scan uses port 80 as default port but you can also manually set the port too, we also need to provide range/number of machines we want to scan (high range means greater scan time).

Nmap scan

Syntax:

nmap -p<port scan> --script ipidseq -iR <range>

Nmap Scan Command

nmap -p80 --script ipidseq -iR 100

(-iR \rightarrow Random IP address scans)

```
-(kali�kali)-[~]
$ sudo nmap
                      script ipidseq -iR 1000
Starting Nmap 7.91 ( https://nmap.org ) at 2020-11-01 11:51 IST
Nmap scan report for 177-22-12-48.alfanetwork.com.br (177.22.12.48)
Host is up (0.58s latency).
PORT
       STATE
                SERVICE
80/tcp filtered http
Nmap scan report for softbank126125080030.bbtec.net (126.125.80.30)
Host is up (0.079s latency).
PORT
       STATE
                SERVICE
80/tcp filtered http
Nmap scan report for 100.55.62.203
Host is up (0.066s latency).
PORT
       STATE
                SERVICE
80/tcp filtered http
Nmap scan report for 149.204.140.35
Host is up (0.071s latency).
PORT
       STATE
                SERVICE
80/tcp filtered http
Nmap scan report for c-67-188-37-175.hsd1.ca.comcast.net (67.188.37.175)
Host is up (0.32s latency).
```

The scan gives out a lot of result but what we are looking for is results in which ipidseq was detected as incremental and port is open.

```
PORT
       STATE
                 SERVICE
80/tcp filtered http
Nmap scan report for 40.69.66.10
Host is up (0.30s latency).
     STATE SERVICE
PORT
80/tcp <mark>open</mark> http
Host script results:
_ipidseq: Incremental!
Nmap scan report for 16.63.149.98
Host is up (0.11s latency).
                 SERVICE
PORT
       STATE
80/tcp filtered http
```

Note down the corresponding IP address and use the as zombie.

The downside is

- 1) Most of the time the ipidseq shows Random, All zeros..etc, this maybe be due to reasons like the firewall or not vulnerable..etc.
- 2) Most of Operating Systems assign IP ID randomly.

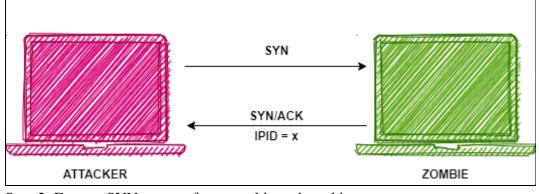
3) Well configured firewalls and honeypots may return false positive.

For explanation I used the following IP address

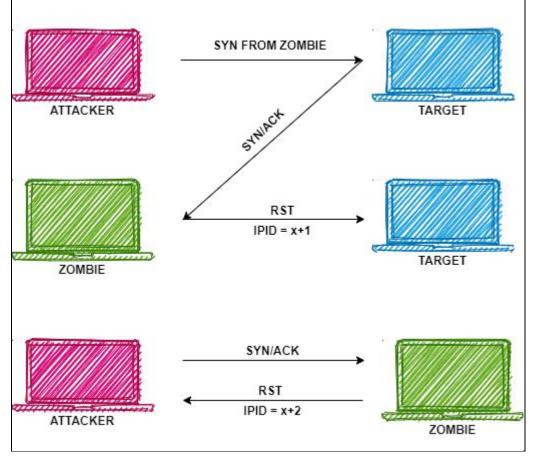
Attacker IP 192.168.43.55 Zombie IP 192.168.43.195 Target IP 192.168.43.248

For open ports

Step 1: Probe zombie IPID



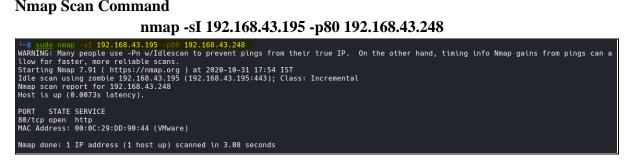
Step 2: Forge a SYN request from zombie and send it to target.



<u>Step 3</u>: Probe the zombie IPID again. If the IPID got increased by 2 from the previous IPID, then it suggests that the port was open. **Syntax:**



Nmap Scan Command



Wireshark analysis

The screenshot given below shows an open port scan. I filtered the scan to show that the attacker never directly communicates with the target.

📕 ip	.addr == 192.168.43	3.55			
No.	Time	Source	Destination	Protocol	Length Info
_	97 3.412141504	192.168.43.55	192.168.43.195	TCP	58 42515 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
L	98 3.412798933	192.168.43.195	192.168.43.55	TCP	60 443 → 42515 [RST] Seq=1 Win=0 Len=0
	99 3.449039786	192.168.43.55	192.168.43.195	TCP	58 42516 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	100 3.449930784	192.168.43.195	192.168.43.55	TCP	60 443 → 42516 [RST] Seq=1 Win=0 Len=0
	102 3.482576885	192.168.43.55	192.168.43.195	TCP	58 42517 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	103 3.483159560	192.168.43.195	192.168.43.55	TCP	60 443 → 42517 [RST] Seq=1 Win=0 Len=0
	104 3.516494245	192.168.43.55	192.168.43.195	TCP	58 42518 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	105 3.517090346	192.168.43.195	192.168.43.55	TCP	60 443 → 42518 [RST] Seq=1 Win=0 Len=0
	106 3.550300971	192.168.43.55	192.168.43.195	TCP	58 42519 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	107 3.551105355	192.168.43.195	192.168.43.55	тср	60 443 → 42519 [RST] Seq=1 Win=0 Len=0
	108 3.581400804	192.168.43.55	192.168.43.195	TCP	58 42520 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	109 3.581864315	192.168.43.195	192.168.43.55	TCP	60 443 → 42520 [RST] Seq=1 Win=0 Len=0
	132 4.040644956	192.168.43.55	192.168.43.195	TCP	58 42729 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	133 4.041192899	192.168.43.195	192.168.43.55	TCP	60 443 → 42729 [RST] Seq=1 Win=0 Len=0
	137 4.093074752	192.168.43.55	192.168.43.195	TCP	58 42554 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	138 4.093986805	192.168.43.195	192.168.43.55	TCP	60 443 → 42554 [RST] Seq=1 Win=0 Len=0
	142 4.145781092	192.168.43.55	192.168.43.195	TCP	58 42578 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	143 4.146609934	192.168.43.195	192.168.43.55	TCP	60 443 → 42578 [RST] Seq=1 Win=0 Len=0
	147 4.199235319	192.168.43.55	192.168.43.195	TCP	58 42641 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	148 4.199803556	192.168.43.195	192.168.43.55	TCP	60 443 → 42641 [RST] Seq=1 Win=0 Len=0
	149 4.251867863	192.168.43.55	192.168.43.195	TCP	58 42725 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	150 4.252603208	192.168.43.195	192.168.43.55	TCP	60 443 → 42725 [RST] Seq=1 Win=0 Len=0

The screenshot below shows the phases in which scan happens.

The first slot show communication between attacker and zombie.

The second slot is where the actual scan takes place. The zombie communicates with the target.

The third slot is communication between zombie and attacker to verify port status.

li	p.addr == 192.168.4	3.195			
No.	Time	Source	Destination	Protocol	Length Info
	105 3.782795529	192.168.43.55	192.168.43.195	TCP	58 42984 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	106 3.783839045	192.168.43.195	192.168.43.55	TCP	60 443 → 42984 [RST] Seq=1 Win=0 Len=0
	107 3.902077511		192.168.43.195	TCP	58 42985 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	108 3.902695737	192.168.43.195	192.168.43.55	TCP	60 443 → 42985 [RST] Seq=1 Win=0 Len=0
	109 3.934094919	192.168.43.55	192.168.43.195	TCP	58 42986 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	110 3.934648824	192.168.43.195	192.168.43.55	TCP	60 443 → 42986 [RST] Seq=1 Win=0 Len=0
	111 3.965342439	192.168.43.55	192.168.43.195	TCP	58 42987 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	112 3.966267591	192.168.43.195	192.168.43.55	TCP	60 443 → 42987 [RST] Seq=1 Win=0 Len=0
	113 3.997686645		192.168.43.195	TCP	58 42988 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	114 3.998431093	192.168.43.195	192.168.43.55	TCP	60 443 → 42988 [RST] Seq=1 Win=0 Len=0
	115 4.030192307		192.168.43.195	TCP	58 42989 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	116 4.031011507	192.168.43.195	192.168.43.55	TCP	60 443 → 42989 [RST] Seq=1 Win=0 Len=0
	117 4.031680822		192.168.43.195	TCP	58 42983 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	118 4.032327576	192.168.43.195	192.168.43.248	TCP	60 443 → 42983 [RST] Seq=1 Win=0 Len=0
	119 4.083465876	192.168.43.248	192.168.43.195	TCP	58 [TCP Port numbers reused] 42983 → 443 [SYN, ACK] Seq=1 Ack=1 Win=1024 Len=0 MSS=1
	120 4.084294249	192.168.43.195	192.168.43.248	TCP	60 443 → 42983 [RST] Seq=1 Win=0 Len=0
	121 4.134443083	192.168.43.248	192.168.43.195	TCP	58 [TCP Port numbers reused] 42983 → 443 [SYN, ACK] Seq=2 Ack=1 Win=1024 Len=0 MSS=1
	122 4.135381509	192.168.43.195	192.168.43.248	TCP	60 443 → 42983 [RST] Seq=1 Win=0 Len=0
	123 4.185150605	192.168.43.248	192.168.43.195	TCP	58 [TCP Port numbers reused] 42983 → 443 [SYN, ACK] Seq=3 Ack=1 Win=1024 Len=0 MSS=1…
	124 4.185798777	192.168.43.195	192.168.43.248	TCP	60 443 → 42983 [RST] Seq=1 Win=0 Len=0
	125 4.485665204		192.168.43.195	TCP	58 43176 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	126 4.486596194	192.168.43.195	192.168.43.55	TCP	60 443 → 43176 [RST] Seq=1 Win=0 Len=0
	127 4.486820591	192.168.43.195	192.168.43.248	TCP	58 443 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
	128 4.487645431	192.168.43.248	192.168.43.195	TCP	60 80 → 443 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
	129 4.487888172		192.168.43.248	TCP	60 443 → 80 [RST] Seq=1 Win=0 Len=0
	130 4.538607805	192.168.43.55	192.168.43.195	TCP	58 43200 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	131 4.539231033	192.168.43.195	192.168.43.55	TCP	60 443 → 43200 [RST] Seq=1 Win=0 Len=0
	132 4.590046037		192.168.43.195	TCP	58 43232 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	133 4.590675394	192.168.43.195	192.168.43.55	TCP	60 443 → 43232 [RST] Seq=1 Win=0 Len=0
	134 4.590776563	192.168.43.195	192.168.43.248	TCP	58 [TCP Retransmission] 443 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
	135 4.591476428	192.168.43.248 192.168.43.195	192.168.43.195	TCP TCP	60 [TCP Previous segment not captured] [TCP Port numbers reused] 80 → 443 [SYN, ACK]
	136 4.591502995		192.168.43.248		60 443 → 80 [RST] Seq=1 Win=0 Len=0
	137 4.643513377		192.168.43.195	TCP	58 43216 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	138 4.644481870	192.168.43.195	192.168.43.55	TCP	60 443 → 43216 [RST] Seg=1 Win=0 Len=0

<u>Step 1</u>: Record the Identification number present in Internet Protocol Version Section. Pay close attention to which Request you should analyze. The marked one is the last request that communicates with the attacker and sends the RST packet and this should be used for analysis.

		-	९ 🗢 🔿 警 有	₹	📃 @, @, @, 🎹
	o.addr == 192.168.4	3.55 or ip.addr == 192	2.168.43.248 and ip.addr	== 192.16	8.43.195 A
No.	▼ Time	Source	Destination	Protoc	ol Lengt ^I Info
	125 4.485665204	192.168.43.55	192.168.43.195	TCP	58 43176 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	126 4.486596194		192.168.43.55	TCP	60 443 → 43176 [RST] Seq=1 Win=0 Len=0
	127 4.486820591	192.168.43.195	192.168.43.248	TCP	58 443 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
	128 4.487645431	192.168.43.248	192.168.43.195	TCP TCP	60 80 → 443 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
	129 4.487888172 130 4.538607805		192.168.43.248 192.168.43.195	TCP	60 443 → 80 [RST] Seq=1 Win=0 Len=0 58 43200 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	131 4.539231033		192.168.43.195	TCP	60 443 → 443200 [RST] Seg=1 Win=0 Len=0
	132 4.590046037	132.100.43.133	132.100.43.33	TCP	58 43232 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
L	133 4.590675394	192.168.43.195	192.168.43.55	TCP	60 443 → 43232 [RST] Seg=1 Win=0 Len=0
	134 4.590776563		192.168.43.248	TCP	58 [TCP Retransmission] 443 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
	135 4.591476428		192.168.43.195	ТСР	60 [TCP Previous segment not captured] [TCP Port numbers reused] 80 → 443 [SYN,
	136 4.591502995		192.168.43.248	TCP	60 443 → 80 [RST] Seq=1 Win=0 Len=0
	137 4.643513377		192.168.43.195	TCP	58 43216 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	138 4.644481870		192.168.43.55	TCP	60 443 → 43216 [RST] Seq=1 Win=0 Len=0
	142 4.696588075		192.168.43.195	TCP	58 43177 → 443 [SYN, ACK] Seg=0 Ack=1 Win=1024 Len=0 MSS=1460
. ▶ E	thernet II, Src: \ nternet Protocol \	/Mware_e9:aa:f0 (00 /ersion 4, Src: 192		VMware_) on interface eth0, id 0 97:bd:4c (00:0c:29:e7:bd:4c) 5
	0100 = Vers	ion: 4 ler Length: 20 byte:	5 (5)		
)) (DSCP: CS0, ECN: Not	-ECT)	
	Total Length: 40			,	
	Identification:				
)	Flags: 0x4000, D				
	Fragment offset:				
	Time to live: 12 Protocol: TCP (6				
		0x21cb [validation]	[beldesib		
		status: Unverified			
	Source: 192.168.		.1		
	Destination: 192				
- T			rt: 443, Dst Port: 432	32, Seq:	1, Len: 0
	Source Port: 443				
	Destination Port				
	[Stream index: 1				
	[TCP Segment Ler		worker humber)		
	Sequence number:	T (LETUTING SEC	quence number)		

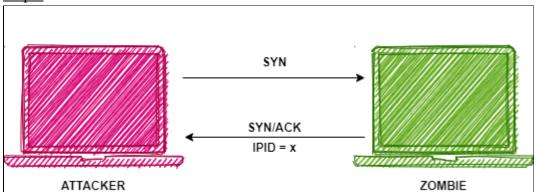
<u>Step 2 and 3</u>: After the RST packet to attacker the zombie scans the target (Retransmission step). After successful scan the zombie returns to communicate with attacker. Note the Identification number again. See the increase from previous number, its 188-186=2

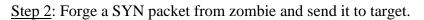
0.	 Time 	Source	Destination	Protocol	Length Info
	125 4.485665204	192.168.43.55	192.168.43.195	TCP	58 43176 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	126 4.486596194		192.168.43.55	TCP	60 443 → 43176 [RST] Seq=1 Win=0 Len=0
	127 4.486820591		192.168.43.248	TCP	58 443 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
	128 4.487645431 129 4.487888172		192.168.43.195 192.168.43.248	TCP TCP	60 80 → 443 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460
	130 4.538607805		192.168.43.248	TCP	60 443 → 80 [RST] Seq=1 Win=0 Len=0 58 43200 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	131 4.539231033		192.168.43.55	TCP	60 443 → 43200 [RST] Seg=1 Win=0 Len=0
	132 4.590046037		192.168.43.195	TCP	58 43232 → 443 [SYN, ACK] Seg=0 Ack=1 Win=1024 Len=0 MSS=1460
	133 4.590675394	192.168.43.195	192.168.43.55	TCP	60 443 → 43232 [RST] Seg=1 Win=0 Len=0
	134 4.590776563		192.168.43.248		58 [TCP Retransmission] 443 → 80 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
	135 4.591476428	192.168.43.248	192.168.43.195	TCP	60 [TCP Previous segment not captured] [TCP Port numbers reused] 80 → 443 [SYN, ACK
	136 4.591502995	192.168.43.195	192.168.43.248	TCP	60 443 → 80 [RST] Seq=1 Win=0 Len=0
	137 4.643513377 138 4.644481870	192.168.43.195	192.168.43.55	TCP	60 443 - 43216 [RST] Sed=1 Win=0 Len=0
	138 4.644481870	192.168.43.195	192.168.43.55	TCP	60 443 → 43216 [RSI] Seq=1 W1n=0 Len=0 20 90117 → 990 1010, AUXT 200-0 AUXT 1011024 Len=0 MSS=1460
Et In	thernet II, Src: V ternet Protocol V 0100 = Vers 0101 = Head Differentiated S	/Mware_e9:àa:f0 (00: /ersion 4, Src: 192. sion: 4 Wer Length: 20 bytes Services Field: 0x00	0c:29:e9:aa:f0), Dst: 168.43.195, Dst: 192.	VMware_e7 168.43.55	n interface eth0, id 0 :bd:4c (00:0c:29:e7:bd:4c)
Et In	hernet II, Src: 1, tternet Protocol N 0100 = Vers 0101 = Head Differentiated S Total Length: 44 Identification: Fragment offset: Protocol: TCP (6 Header checksum [Header checksum	Mware_e9:aa:F0 (00 Version 4, Src: 192. ion: 4 Src: 192. ior Length: 20 bytes ervices Field: 0x00 0x000 (108) 0 (108) 0 (108) 0 0 (108) 0 0 0 (108) 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	:0c:29:69:aa:f0), Dst: 168.43.195, Dst: 192. : (5) (DSCP: CS0, ECN: Not	VMware_e7 168.43.55	
Et	hernet II, Src: 1 ternet Protocol X 0100 e Vers 0101 - Head Differentiated S Total Length: 40 Fragment Orfset: Time to live: 12 Protocol: TCP (6 Header checksum Source: 192.168 Destination: 192	MWware_e9:aa:F0 (00) Version 4, Src: 192. ion: 4 Src: 192. ion: 4 Canter 192. ion: 4 Canter 192. ion: 1 Cannent 0 Cooper (188) ion: 1 Cannent 0 Cooper (188) ion: 1 Cannent 0 Canter 192 ion: 1 Cannent 0	:0c:20:40:aa:r6), Dst: 168.43.195, Dst: 192. ; (5)) (DSCP: CS0, ECN: Not disabled]]	VMware_07: 168.43.55	:bd:4c (00:0c:29:e7:bd:4c)
Et	hernet II, Src: 1 ternet Protocol X 0100 e Vers 0101 - Head Differentiated S Total Length: 40 Fragment Orfset: Time to live: 12 Protocol: TCP (6 Header checksum Source: 192.168 Destination: 192	<pre>//Wware_e9:aa:F0 (00) /ersion 4, Src: 192. ion: 4 /er Length: 20 bytes /ervices Field: 0x00 /ox000 (188) /oon't ragment 0 /ox000 (188) /oon't ragment 0 /ox2109 [validation 0 status: Unverified 43.195 /168.43.55 /or Portocol, Src Portoc</pre>	:0c:29:69:aa:f0), Dst: 168.43.195, Dst: 192. : (5) (DSCP: CS0, ECN: Not	VMware_07: 168.43.55	:bd:4c (00:0c:29:e7:bd:4c)

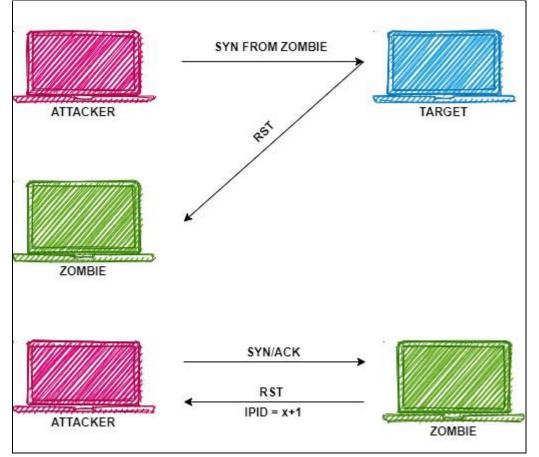
This difference signifies that the port is open for connection.

For closed port

<u>Step 1</u>: Probe the zombie IPID.



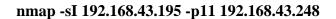




<u>Step 3</u>: Probe the zombie IPID again. If the IPID got increased by 1 from the previous IPID, then it suggests that the port was closed. **Syntax:**

nmap -sI <zombie IP> -p<port number> <target IP>

Nmap scan:





Wireshark analysis:

<u>Step 1</u>: Similar to the previous one, analysis the request in which the zombie sends the last RST packet to attacker. Note down the Identification number.

D.	Time	Source	Destination	Protocol	Length Info
	88 5.687442855	192.168.43.55	192.168.43.195	TCP	58 42149 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	89 5.710505694	192.168.43.195	192.168.43.55	тср	60 443 → 42149 [RST] Seg=1 Win=0 Len=0
	90 5.714890952	192.168.43.55	192.168.43.195	TCP	58 42078 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	91 5.761447883	192.168.43.195	192.168.43.55	TCP	60 443 → 42078 [RST] Seq=1 Win=0 Len=0
	92 5.812391615	100 160 10 EE	100 100 10 105	TCD	E0 43100 442 [SVN ACK] Sog=0 Ack=1 Win=1024 Len=0 MSS=1460
	93 5.822172243	192.168.43.195	192.168.43.55	тср	60 443 → 42190 [RST] Seq=1 Win=0 Len=0
		192.108.43.195	192.108.43.248	IUP	So [TOP Recramsmission] 443 - 11 [STW] Sey=0 Win=1024 Len=0 MSS=140
	95 5.874895807	192.168.43.55	192.168.43.195	TCP	58 42148 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	96 5.878887608	192.168.43.248	192.168.43.195	TCP	60 11 → 443 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
	97 5.878894395	192.168.43.195	192.168.43.55	TCP	60 443 → 42148 [RST] Šeq=1 Win=0 Len=0
	98 5.898270172	192.168.43.55	192.168.43.195	TCP	58 42155 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	99 5.918429733	192.168.43.195	192.168.43.55	TCP	60 443 → 42155 [RST] Seq=1 Win=0 Len=0
	100 5.972712764	192.168.43.55	192.168.43.195	TCP	58 42154 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
	101 5.975253896	192.168.43.195	192.168.43.55	TCP	60 443 - 42154 [RST] Seg=1 Win=0 Len=0
	0100 = Ver	Version 4, Src: 192.	:0c:29:e9:aa:f0), Dst .168.43.195, Dst: 192	: VMware_e7:	n interface eth0, id 0 bd:4c (00:0c:29:e7:bd:4c)
I	0100 = Ver 0100 = Ver 0101 = Head 0ifferentiated S Total Length: 4	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x00 0	:0c:29:e9:aa:f0), Dst .168.43.195, Dst: 192	: VMware_e7: .168.43.55	
11 •	nternet Protocol 0100 = Vers 0101 = Head Differentiated S Total Length: 4 Identification:	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x00 0 0x0051 (88)	:0c:29:e9:aa:f0), Dst .168.43.195, Dst: 192 s (5)	: VMware_e7: .168.43.55	
1r	nternet Protocol 0100 = Vers 0101 = Heau Differentiated 3 Total Length: 4 Identification: Flags: 0x4000, 1	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x00 0 0x005; (88) Don't i casmon	:0c:29:e9:aa:f0), Dst .168.43.195, Dst: 192 s (5)	: VMware_e7: .168.43.55	
11 •	nternet Protocol 0100 = Ver: 0101 = Hear Differentiated Total Length: 44 Identification: Flags: 0x4000, Fragment offset	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 9 0x0051 (88) 0001t 1 common : 0	:0c:29:e9:aa:f0), Dst .168.43.195, Dst: 192 s (5)	: VMware_e7: .168.43.55	
Ir ▶	hternet Protocol 0100 = Ver: 0101 = Heau Differentiated 3 Total Length: 44 Identification: Flags: 0x4000, 1 Fragment offset Time to live: 13	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 0 0x0051 (88) Don't 1 common : 0 28	:0c:29:e9:aa:f0), Dst .168.43.195, Dst: 192 s (5)	: VMware_e7: .168.43.55	
Ir ▶	hternet Protocol 0100 = Ver: 0101 = Head Differentiated 3 Total Length: 44 Identification: Flags: 0x4000, 1 Fragment offset Time to live: 12 Protocol: TCP (0	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x00 0x005: (88) 0x005: (88) 0x005: 28 5)	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 5 (5) 0 (DSCP: CS0, ECN: Not	: VMware_e7: .168.43.55	
11 •	tternet Protocol 0100 = Vers 0101 = Head Differentiated 3 Total Length: 44 Identification: Flags: 0x4000, 1 Fragment offset Time to live: 12 Protocol: TCP (i Header checksum	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 0x0055 (88) 0x0055 (:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 5 (5) 0 (DSCP: CS0, ECN: Not 0 disabled]	: VMware_e7: .168.43.55	
11 •	<pre>hternet Protocol 0100 = Ver: 0101 = Head Differentiated : Total Length: 44 Tlaentification: Flags: 0x4000, 1 Fragment offset Time to live: 1 Protocol: TCP (Header checksum [Header checksum]</pre>	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 0x005 (88) Don't : 6 28 5 0x222d [validation status: Unverified	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 5 (5) 0 (DSCP: CS0, ECN: Not 0 disabled]	: VMware_e7: .168.43.55	
1r	tternet Protocol 0100 = Ver 0101 = Hea Differentiated 3 Total Length: 4 Identification: Flags: 0x4000, Fragment offset Time to live: 12 Protocol: TCP (Header checksum [Header checksum Source: 192.168	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 0x005 (88) 0x005 (88) 0x0	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 5 (5) 0 (DSCP: CS0, ECN: Not 0 disabled]	: VMware_e7: .168.43.55	
11 +	ternet Protocol 0100 9 Ver 0101 = Hear Differentiated 3 Total Length: 44 Identification: Flags: 0x4000, 1 Fragment offset Time to live: 1: Protocol: TCP (0 Header checksum Source: 192.168 Destination: 19:	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 0x0051 (88) 0x0051 (:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 5 (5) 0 (DSCP: CS0, ECN: Not 0 (DSCP: disabled] 1]	: VMware_e7: .168.43.55 :-ECT)	bd:4c (00:0c:29:e7:bd:4c)
1	tternet Protocol 0100 = Ver 0101 = Hea Differentiated 3 Total Length: 44 Total Length: 44 Totaltication: Fragment offset Time to live: 13 Protocol: TCP (Header checksus Source: 192.168 Destination: 19 ansmission Contr	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 0x0055 (88) 00n't 1 20000 : 0 28 b) : 0x222d [validation m status: Unverified .43.195 2.108.43.55 ol Protocol, Src Pon	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 5 (5) 0 (DSCP: CS0, ECN: Not 0 disabled]	: VMware_e7: .168.43.55 :-ECT)	bd:4c (00:0c:29:e7:bd:4c)
1	ternet Protocol 0100 e Ver 0101 = Hea Differentiated : Total Length: 44 Identification: Flags: 0x4000, 1 Fragment offset Time to live: 1 Protocol: TCP (1 Header checksum [Header checksum [Header checksum [Bestination: 19: ansmission Comt: 44:	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 0x0055 (88) 0x0055 (:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 5 (5) 0 (DSCP: CS0, ECN: Not 0 (DSCP: disabled] 1]	: VMware_e7: .168.43.55 :-ECT)	bd:4c (00:0c:29:e7:bd:4c)
11 +	tternet Protocol 0100 = Ver 0101 = Hea Differentiated 3 Total Length: 44 Total Length: 44 Total Length: 44 Total Length: 44 Fragment offset Time to live: 12 Protocol: TCP (1 Header checksus Source: 192.168 Destination: 19 ramsmission Contr Source Port: 44.	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 0x005 (88) 0x005 (88) 0x005 (88) 0x022 (validation status: Unverified .43.195 2.168.43.55 ol Protocol, Src Por 3 : 42190	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 5 (5) 0 (DSCP: CS0, ECN: Not 0 (DSCP: disabled] 1]	: VMware_e7: .168.43.55 :-ECT)	bd:4c (00:0c:29:e7:bd:4c)
1	ternet Protocol 0100 e Ver 0101 = Hea Differentiated : Total Length: 44 Identification: Flags: 0x4000, 1 Fragment offset Time to live: 1 Protocol: TCP (1 Header checksum [Header checksum [Header checksum [Bestination: 19: ansmission Comt: 44:	Version 4, Src: 192. sion: 4 der Length: 20 bytes Services Field: 0x06 0x005 (88) 0x015 (88) 0x015 (88) 0x025 (88) 0x0	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 5 (5) 0 (DSCP: CS0, ECN: Not 0 (DSCP: disabled] 1]	: VMware_e7: .168.43.55 :-ECT)	bd:4c (00:0c:29:e7:bd:4c)

<u>Step 2 and 3</u>: The next request is communication between zombie and target. After the scan is performed the zombie communicates with attacker and we note the Identification Number again. Calculate the difference between both the numbers i.e. 89-88=1

88 5.88744285 192.168.43.105		Time	Source	Destination	Protocol L	Length Info
99 5.714896952 192.168.43.55 192.168.43.195 TCP 58 42078 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1624 Len=0 MSS=1460 91 5.761447883 192.168.43.55 192.168.43.55 TCP 60 443 - 42078 [RST] Seq=1 Win=0 Len=0 92 5.612391615 192.168.43.55 192.168.43.195 TCP 58 42109 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460 93 5.822172243 192.168.43.195 192.168.43.195 TCP 58 42109 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460 94 5.82280705 192.168.43.195 192.168.43.195 TCP 58 42109 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460 95 5.8788976897 192.168.43.195 192.168.43.195 TCP 58 42148 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460 95 5.8788976895 192.168.43.195 192.168.43.55 TCP 60 443 - 42148 [KST] Seq=1 Win=0 Len=0 97 5.8788976895 192.168.43.195 192.168.43.55 TCP 60 443 - 42148 [KST] Seq=1 Win=0 Len=0 98 5.898276172 192.168.43.195 192.168.43.55 TCP 60 443 - 42148 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460 99 5.918429733 192.168.43.195 192.168.43.55 TCP 60 443 - 42145 [KST] Seq=1 Win=0 Len=0 101 5.975253896 192.168.43.195 19		88 5.687442855	192.168.43.55	192.168.43.195	TCP	58 42149 → 443 [SYN, ACK] Seg=0 Ack=1 Win=1024 Len=0 MSS=1460
915.761447883 192.168.43.195 192.168.43.195 TCP 60 443 - 42078 [RST] Seq=1 Win=0 Len=0 925.812391615 192.168.43.55 192.168.43.195 TCP 58 42190 - 443 [SYN, ACK] Seq=0 AUXH WIN=1624 Len=0 MSS=1460 935.822172243 192.168.43.195 192.168.43.195 TCP 58 42190 - 443 [SYN, ACK] Seq=0 AUXH = 143 [SYN] Seq=0 Win=1024 Len=0 MSS=1460 945.822289705 192.168.43.195 192.168.43.195 TCP 58 42148 - 423 [SYN, ACK] Seq=0 Win=1024 Len=0 MSS=1460 955.874895807 192.168.43.195 192.168.43.195 TCP 58 42148 - 443 [SYN, ACK] Seq=0 AUXH = 143 [SYN] Seq=0 Win=1024 Len=0 MSS=1460 955.878894395 192.168.43.195 192.168.43.195 TCP 50 4145 4145 (SYN, ACK] Seq=0 AUXH = 143 [SYN, AUX] Seq=0 AUXH = 143		89 5.710505694	192.168.43.195	192.168.43.55	TCP	60 443 → 42149 [RST] Seg=1 Win=0 Len=0
92 5.812391615 192.168.43.55 192.168.43.195 192.16		90 5.714890952	192.168.43.55	192.168.43.195	TCP	58 42078 → 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460
98 5.822172243 192.168.43.195 192.168.43.55 TCP 60 443 - 42190 [RST] Seq=1 Win=0 Len=0 94 5.822280755 192.168.43.195 192.168.43.195 TCP 58 [TCP Retransission] 443 - 11 [SVN] Seq=0 Win=1024 Len=0 NSS=1460 95 5.874895807 192.168.43.195 192.168.43.195 TCP 58 [TCP Retransission] 443 - 11 [SVN] Seq=0 Win=1024 Len=0 NSS=1460 96 5.878897608 102.168.43.195 192.168.43.55 TCP 60 443 - 42148 [SVN] Seq=1 Win=0 Len=0 97 5.878897608 192.168.43.195 192.168.43.55 TCP 60 443 - 4215 [SVN] Seq=1 Win=0 Len=0 98 5.898276172 192.168.43.195 192.168.43.55 TCP 60 443 - 4215 [SVN] Seq=1 Win=0 Len=0 98 5.9782762 192.168.43.195 192.168.43.55 TCP 60 443 - 4215 [SVN] Seq=1 Win=0 Len=0 190 5.97212764 192.168.43.195 TCP 58 42154 - 443 [SVN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460 101 5.975253896 192.168.43.195 192.168.43.55 TCP 60 443 - 42154 [SVN] Seq=1 Win=0 Len=0 101 5.975253896 192.168.43.195 192.168.43.55 TCP 60 443 - 42154 [SVN] Seq=1 Win=0 Len=0 101 S.975253896 192.168.43.195 192.168.43.55 TCP 60 443 - 42154 [SVN] Seq=1 Win=0 Len=0		91 5.761447883	192.168.43.195	192.168.43.55	TCP	60 443 → 42078 [RST] Seq=1 Win=0 Len=0
94 5.922289705 192.168.43.195 192.1						
95 5.874895807 192.168.43.55 192.168.43.195 TCP 58 42148 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1624 Len=0 MSS=1460 96 5.878897608 192.168.43.195 192.168.43.55 TCP 60 443 - 42148 [RST] Seq=1 Win=0 Len=0 98 5.898276172 192.168.43.195 192.168.43.55 TCP 60 443 - 42155 [RST] Seq=1 Win=0 Len=0 98 5.898276172 192.168.43.195 192.168.43.55 TCP 60 443 - 42155 [RST] Seq=1 Win=0 Len=0 99 5.912429733 192.168.43.195 192.168.43.55 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 106 5.972712764 192.168.43.195 192.168.43.55 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 101 5.972712764 192.168.43.195 192.168.43.55 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 Frame 97: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface ethe, id 0 Ethernet II, Src: VMware_e9:aa:f0 (00:6c:29:e9:aa:f0), Dst: 192.168.43.55 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 Frame 97: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface ethe, id 0 10 Ethernet II, Src: VMware_e9:aa:f0 (00:6c:29:e9:aa:f0), Dst: 192.168.43.55 10 Internet Protocol Version 4 cost is 12.168.43.195.05 12.168.43.55 10 10 Identification: 0x00051 (49)		93 5.822172243	192.168.43.195	192.168.43.55		
96 5.97887608 402 402 40 40 40 40 404 40 40 40 40 404 40 40 40 40 40 40 97 5.87887608 192.168.43.195 192.168.43.55 TCP 60 443 - 42148 [RST] Seq=1 Win=0 Len=0 98 5.898270172 192.168.43.195 192.168.43.55 TCP 60 443 - 42155 [KST] Seq=1 Win=0 Len=0 100 5.97712764 192.168.43.195 192.168.43.195 TCP 60 443 - 42155 [KST] Seq=1 Win=0 Len=0 100 5.97721764 192.168.43.195 192.168.43.195 TCP 58 42154 - 443 [SYN, ACK] Seq=0 Ack.4 Win=10 Len=0 101 5.975253896 192.168.43.195 192.168.43.195 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 Frame 97: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0 Ethernet II, Src: VMware_e9:aa:F0 (00:6c:29:e9:aa;F0), Dst: 192.168.43.55 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 Frame 97: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0 Ethernet II, Src: VMware_e9:aa:F0 (00:6c:29:e9:ae;F0), Dst: 192.168.43.55 010 = Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 010 0 = Version: 4 0101 = Header Length: 20 bytes (5) 0101 = Header Length: 20 bytes (5) 0101 = Header Length: 40 Identification: 0xc055 [69] Flass: 0x4000, Don't interface interface interface interface interface interface interface interface interf						
397 5.878894395 192.108.43.195 192.108.43.55 TCP 60 443 - 42148 [RST] Seq=1 Win=0 Len=0 924 Len=0 MSS=1460 99 5.988270172 100 450 47.455 TCP 60 443 - 42158 [RST] Seq=1 Win=0 Len=0 924 Len=0 MSS=1460 100 5.972712764 192.108.43.195 192.108.43.195 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 924 Len=0 MSS=1460 100 5.972712764 192.108.43.195 192.108.43.195 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 92.108.43.195 192						
98 5.998270172 100.40.47 100.47 <						
99 5.91842973 192.168.43.195 192.168.43.55 TCP 60 443 - 42155 [RST] Seq=1 Win=0 Len=0 109 5.972712764 192.168.43.195 192.168.43.195 TCP 58 42154 -4415 101 5.975253896 192.168.43.195 192.168.43.195 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 Frame 97: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, 1d 0 Ethernet II, Src: VMware_e9:aa:F0 (00:0c:29:e9:aa;F0), Dst: VMware e7:bd:4c (00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 00:0c:29:e7:bd:4c) Identification: 0x0051 (89) 00 trees Elevel: 0x005 (89) 00:0c:29:e7:bd:4c) Fragment offset: 0 0 0 0 Identification: 0x0055 (89) 0 0 0 Fragment offset: 0 0 <					тср	60 443 → 42148 [RST] Seq=1 Win=0 Len=0
100 5.972712764 192.168.43.55 192.168.43.195 TCP 58 42154 - 443 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1460 101 5.975253896 192.168.43.195 192.168.43.195 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 Frame 97: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0 Ethernet II, Src: WWare_e9:aa:f0 (00:0c:29:e9:aa:f0), Dst: WWare_e7:bd:4c (00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 0 0 0100 = Version: 4 0101 = Header Length: 20 bytes (5) 0 Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT) Total Length: 40 Identification: 0x0051 (09) 0101 Fragment offset: 0 0101 Time to live: 128 04400, Don't 1 Protocol: TCP (6) Header checksum status: Unverified] Source: 192.168.43.195 0102					TCD	ED 491EE 442 FSVI ACK3 Seg=0 Ack=1 Wip=1024 Len=0 MSS=1460
101 5.975253896 192.168.43.195 192.168.43.55 TCP 60 443 - 42154 [RST] Seq=1 Win=0 Len=0 Frame 97: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0 Ethernet II, Src: VMware_e9:aa:f0 (00:0c:29:e9:aa:f0), Dst: VMware_e7:bd:4c (00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 VMware_e7:bd:4c (00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 0100 = Version: 4						
<pre>Frame 97: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface eth0, id 0 Ethernet II, Src: VMware_e9:aa:f0 (00:0c:29:e9:aa:f0), Dst: VMware_e7:bd:4c (00:0c:29:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 0100 = Version: 4 0101 = Header Length: 20 bytes (5) Differentiated Services Field: 0x000 (DSCP: CS0, ECN: Not-ECT) Total Length: 40 Identification: 0x0055 (89) Flags: 0x4000, Don't Fragment offset: 0 Time to live: 128 Protocol: TCP (6) Header checksum status: Unverified] Source: 192.168.43.195</pre>						
<pre>Ethernet II, Src: VMware_e0:aa:F0 (00:0c:20:e0:aa:f0), Dst: VMware_e7:bd:4c (00:0c:20:e7:bd:4c) Internet Protocol Version 4, Src: 192.168.43.195, Dst: 192.168.43.55 0100 = Version: 4 0101 = Header Length: 20 bytes (5) Differentiated Services Field 0x00 (DSCP: CS0, ECN: Not-ECT) Total Length: 40 Identification: 0x0051 (89) Flags: 0x4000, Don't 1 Fragment offset: 0 Time to live: 128 Protocol: TCP (6) Header checksum status: Unverified] Source: 192.168.43.195</pre>						
Fragment offset: 0 Time to live: 128 Protocol: TCP (6) Header checksum: 0x222c [validation disabled] [Header checksum status: Unverified] Source: 192.168.43.195	Et	rame 97: 60 bytes thernet II, Src: ' iternet Protocol '	on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192	0c:29:e9:aa:f0), Dst	480 bits) on : VMware_e7:b	interface eth0, id 0
Fragment offset: 0 Time to live: 128 Protocol: TCP (6) Header checksum: 0x222c [validation disabled] [Header checksum status: Unverified] Source: 192.168.43.195	Et In	rame 97: 60 bytes chernet II, Src: ' 0100 = Vers 0101 = Hea Differentiated S	on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192 sion: 4 der Length: 20 bytes Services Field: 0x00	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 : (5)	480 bits) on : VMware_e7:t .168.43.55	interface eth0, id 0
Fragment offset: 0 Time to live: 128 Protocol: TCP (6) Header checksum: 0x222c [validation disabled] [Header checksum status: Unverified] Source: 192.168.43.195	Et In	ame 97: 60 bytes thernet II, Src: ' 0100 = Vers 0101 = Heas Differentiated S Total Length: 40	on wire (480 bits) VMware_e9:aa:f0 (00 Version 4, Src: 192 sion: 4 der Length: 20 bytes Services Field: 0x00 9	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 : (5)	480 bits) on : VMware_e7:t .168.43.55	interface eth0, id 0
Protocol: TCP (6) Header checksum: 0x222c [validation disabled] [Header checksum status: Unverified] Source: 102.168.43.195	Et In	rame 97: 60 bytes thernet II, Src: ' 0100 = Vers 0101 = Head Differentiated S Total Length: 44 Identification: Flags: 0x4000, f	on wire (480 bits) VMware_09:aa:f0 (00 Version 4, Src: 192 sion: 4 der Length: 20 bytes Services Field: 0x06 0x005: (89) Don't 1	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 : (5)	480 bits) on : VMware_e7:t .168.43.55	interface eth0, id 0
Header checksum:'0x222c [validation disabled] [Header checksum status: Unverified] Source: 192.168.43.195	Et In	rame 97: 60 bytes thernet II, Src: ' thernet Protocol ' 0100 = Vers 0101 = Head Differentiated S Total Length: 44 Identification: Flags: 0x4000, I Fragment offset	on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192 sion: 4 der Length: 20 bytes Services Field: 0x00 0x005: (89) cn t	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 : (5)	480 bits) on : VMware_e7:t .168.43.55	interface eth0, id 0
[Header checksum status: Unverified] Source: 192.168.43.195	Et In	rame 97: 60 bytes thernet II, Src: ' 0100 = Vers 0101 = Head Differentiated S Total Length: 40 Tentification: Flags: 0x4000, I Fragment offset Time to live: 12	on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192 Sion: 4 der Length: 20 bytes Services Field: 0x06 0x0051 (89) 0x0051 : 0 : 0 28	:0c:29:e9:aa:f0), Dst 168.43.195, Dst: 192 : (5)	480 bits) on : VMware_e7:t .168.43.55	interface eth0, id 0
Source: 192.168.43.195	Et In	rame 97: 60 bytes thernet II, Src: ' ternet Protocol ' 6100 e Ver 0101 = Heat Differentiated ' Total Length: 44 Tdentification: Fragment offset Time to live: 12 Protocol: TCP ((on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192 sion: 4 der Length: 20 bytes Services Field: 0x063 0x0055 (89) 0on't 1 : 0 28 5)	(0:29:09:aa:f0), Dst 168.43.195, Dst: 192 € (5) ⊖ (DSCP: CS0, ECN: Not	480 bits) on : VMware_e7:t .168.43.55	interface eth0, id 0
	Et In	rame 97: 60 bytes thernet II, Src: 1 ternet Protocol 0100 2 Ver 0101 = Hea Differentiated 3 Total Length: 41 Genification: Flags: 0x4000, I Fragment offset Time to live: 12 Protocol: TCP (U Header checksum	on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192 Sion: 4 der Length: 20 bytes Services Field: 0x06 0 0x005: (89) Don't 1 : 0 28 5) : 0x222c [validatior	.00:29:69:aa:f0), Dst 168.43.195, Dst: 192 : (5) (DSCP: CS0, ECN: Not disabled]	480 bits) on : VMware_e7:t .168.43.55	interface eth0, id 0
	Et In	rame 97: 60 bytes thernet II, Src: ' ternet Protocol' 0100 e Ver 0101 = Hea Differentiated 3 Total Length: 44 Identifications Fragment offset Fragment offset Time to live: 12 Protocol: TCP (I Header checksum	on wire (480 bits), VMware_e9:aa:F0 (00 Version 4, Src: 192 sion: 4 der Length: 20 bytes Services Field: 0x00 0x005 (09) 0x005 (09) 0x005 (09) 0x005 (09) 0x005 (09) 0x005 (00) 0x005 (00)	.00:29:69:aa:f0), Dst 168.43.195, Dst: 192 : (5) (DSCP: CS0, ECN: Not disabled]	480 bits) on : VMware_e7:t .168.43.55	interface eth0, id 0
Destination: 192.168.43.55	Et In	rame 97: 60 bytes thernet II, Src: 1 ternet Protocol 1 0100 2 Vers 0101 = Heas Differentiated 3 Total Length: 44 Cientification: Flags: 0x4000, 1 Fragment offset Time to live: 12 Protocol: TCP (0 Header checksum [Header checksum]	on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192 Sion: 4 der Length: 20 bytes Services Field: 0x06 0 0x005; (89) 20 t 20 t : 0x222c [validation n status: Unverifier .43.195	.00:29:69:aa:f0), Dst 168.43.195, Dst: 192 : (5) (DSCP: CS0, ECN: Not disabled]	480 bits) on : VMware_e7:t .168.43.55	interface eth0, id 0
Transmission Control Protocol, Src Port: 443, Dst Port: 42148, Seq: 1, Len: 0	Et In	rame 97: 60 bytes thernet II, Src: ' tternet Protocol' 0100 e Ver: 0101 = Hea Differentiated 3 Total Length: 40 Identification: Fragment offset Fragment offset Time to live: 11 Protocol: TCP (U Header checksum Source: 192.168 Destination: 192	on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192 sion: 4 der Length: 20 bytes Services Field: 0x00 0x0051 (89) 0x01 t : 0 28 5) : 0x222c [validation m status: Unverified .43.195 2.168.43.55	:00:29:09:aa:f0), Dst 168.43.195, Dst: 192 : (5) (DSCP: CS0, ECN: Not disabled]]	480 bits) on : VMware_e7:t .168.43.55 t-ECT)	interface eth0, id 0 d:4c (00:0c:29:e7:bd:4c)
Source PUIC: 443 Destination PUIC: 42148	Et In	ame 97: 60 bytes thernet II, Src: 1 ternet Protocol 1 0100 e Vers 0101 = Hea Differentiated 3 Total Length: 44 <u>Jentafication:</u> Flags: 0x4000, 1 Fragment offset Time to live: 11 Protocol: TCP ((Header checksum [Header checksum Extination: 192.168 Destination: 192.168	on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192 Sion: 4 der Length: 20 bytes Services Field: 0x06 0x0055 (89) 200n t 1 0 0 20 5) : 0x222c [validation m status: Unverifier 43.195 2.168.43.55 01 Protocol, Src Poi	:00:29:09:aa:f0), Dst 168.43.195, Dst: 192 : (5) (DSCP: CS0, ECN: Not disabled]]	480 bits) on : VMware_e7:t .168.43.55 t-ECT)	interface eth0, id 0 d:4c (00:0c:29:e7:bd:4c)
USSTINATION FOR 151	Et In	rame 97: 60 bytes thernet II, Src: ' ternet Protocol' 0100 e Vers 0101 = Heas Differentiated 3 Total Length: 44 Identification: Fragment offset Fragment offset Time to live: 11 Protocol: TCP (U Header checksum (Header checksum Source: 192.168 Destination: 19; ansmission Contr: Source Port: 44;	on wire (480 bits), VMware_e9:aa:f0 (00 Version 4, Src: 192 sion: 4 der Length: 20 bytes Services Field: 0x06 0x0051 (89) 0x0051 (89) 0x01 t : 0 28 5) : 0x222c [validation m status: Unverified .43.195 2.108.43.55 ol Protocol, Src Pon 3	:00:29:09:aa:f0), Dst 168.43.195, Dst: 192 : (5) (DSCP: CS0, ECN: Not disabled]]	480 bits) on : VMware_e7:t .168.43.55 t-ECT)	interface eth0, id 0 d:4c (00:0c:29:e7:bd:4c)

This difference of 1signifies that the port is closed.

Advantage:

- 1) The attacker IP never gets logged in victims IDS (Intrusion Detection System).
- 2) This is very stealthy scan.

Disadvantage:

- 1) Complex scan.
- 2) The attacker IP is still logged in zombie so there's always a chance of traceback.
- 3) Not easy to find zombie/idle machines.