TCP Connection Resets & Hijacking

Unfortunately, TCP does not provide the level of protection to prevent our hosts from having their connections terminated or hijacked by an attacker. As such, we might notice that a connection gets terminated by an RST packet, or hijacked through connection hijacking.

TCP Connection Termination

Related PCAP File(s):

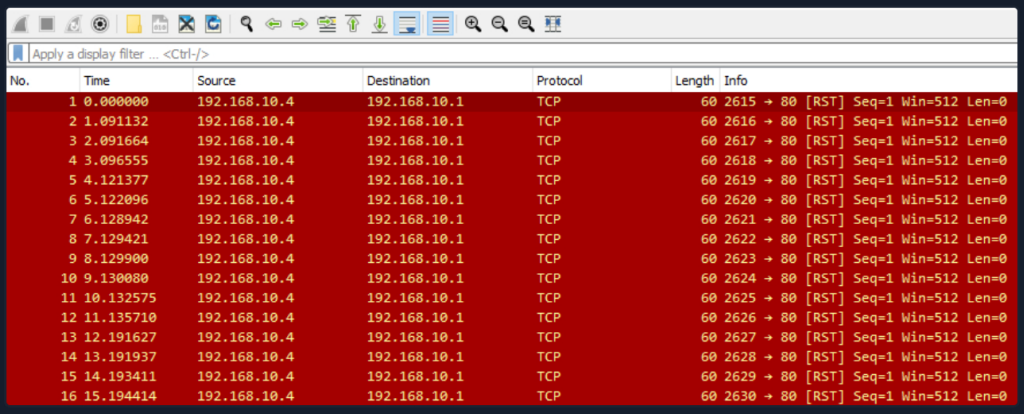
- RST_Attack.pcapng

Suppose an adversary wanted to cause denial-of-service conditions within our network. They might employ a simple TCP RST Packet injection attack, or TCP connection termination in simple terms.

This attack is a combination of a few conditions:

1. The attacker will spoof the source address to be the affected machine's
2. The attacker will modify the TCP packet to contain the RST flag to terminate the connection
3. The attacker will specify the destination port to be the same as one currently in use by one of our machines.

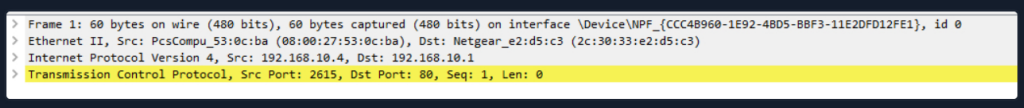
As such, we might notice an excessive amount of packets going to one port.



Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.10.4	192.168.10.1	TCP	60	2615 → 80 [RST] Seq=1 Win=512 Len=0
2	1.091132	192.168.10.4	192.168.10.1	TCP	60	2616 → 80 [RST] Seq=1 Win=512 Len=0
3	2.091664	192.168.10.4	192.168.10.1	TCP	60	2617 → 80 [RST] Seq=1 Win=512 Len=0
4	3.096555	192.168.10.4	192.168.10.1	TCP	60	2618 → 80 [RST] Seq=1 Win=512 Len=0
5	4.121377	192.168.10.4	192.168.10.1	TCP	60	2619 → 80 [RST] Seq=1 Win=512 Len=0
6	5.122096	192.168.10.4	192.168.10.1	TCP	60	2620 → 80 [RST] Seq=1 Win=512 Len=0
7	6.128942	192.168.10.4	192.168.10.1	TCP	60	2621 → 80 [RST] Seq=1 Win=512 Len=0
8	7.129421	192.168.10.4	192.168.10.1	TCP	60	2622 → 80 [RST] Seq=1 Win=512 Len=0
9	8.129900	192.168.10.4	192.168.10.1	TCP	60	2623 → 80 [RST] Seq=1 Win=512 Len=0
10	9.130080	192.168.10.4	192.168.10.1	TCP	60	2624 → 80 [RST] Seq=1 Win=512 Len=0
11	10.132575	192.168.10.4	192.168.10.1	TCP	60	2625 → 80 [RST] Seq=1 Win=512 Len=0
12	11.135710	192.168.10.4	192.168.10.1	TCP	60	2626 → 80 [RST] Seq=1 Win=512 Len=0
13	12.191627	192.168.10.4	192.168.10.1	TCP	60	2627 → 80 [RST] Seq=1 Win=512 Len=0
14	13.191937	192.168.10.4	192.168.10.1	TCP	60	2628 → 80 [RST] Seq=1 Win=512 Len=0
15	14.193411	192.168.10.4	192.168.10.1	TCP	60	2629 → 80 [RST] Seq=1 Win=512 Len=0
16	15.194414	192.168.10.4	192.168.10.1	TCP	60	2630 → 80 [RST] Seq=1 Win=512 Len=0

One way we can verify that this is indeed a TCP RST attack is through the physical address of the transmitter of these TCP RST packets. Suppose, the IP address 192.168.10.4 is registered to aa:aa:aa:aa:aa:aa in our network device list, and we notice an entirely different MAC sending these like the following.



```
> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{CCC4B960-1E92-4BD5-BBF3-11E2DFD12FE1}, id 0
> Ethernet II, Src: PcsCompu_53:0c:ba (08:00:27:53:0c:ba), Dst: Netgear_e2:d5:c3 (2c:30:33:e2:d5:c3)
> Internet Protocol Version 4, Src: 192.168.10.4, Dst: 192.168.10.1
> Transmission Control Protocol, Src Port: 2615, Dst Port: 80, Seq: 1, Len: 0
```

This would indicate malicious activity within our network, and we could conclude that this is likely a TCP RST Attack. However, it is worth noting that an attacker might spoof their MAC address in order to further evade detection. In this case, we could notice retransmissions and other issues as we saw in the ARP poisoning section.

TCP Connection Hijacking

Related PCAP File(s):

- TCP-hijacking.pcap

For more advanced actors, they might employ TCP connection hijacking. In this case the attacker will actively monitor the target connection they want to hijack.

The attacker will then conduct sequence number prediction in order to inject their malicious packets in the correct order. During this injection they will spoof the source address to be the same as our affected machine.

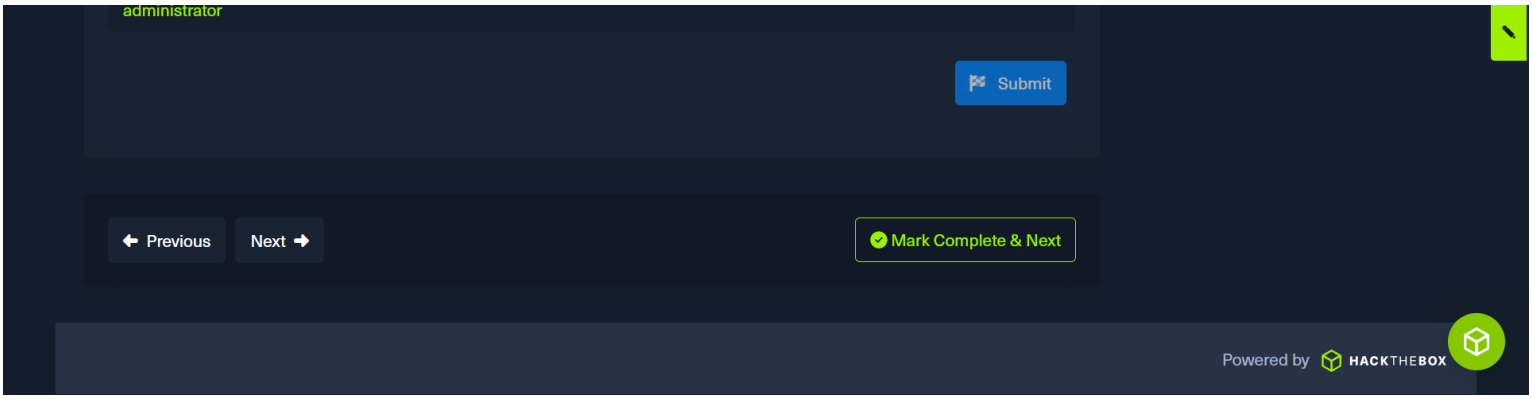
The attacker will need to block ACKs from reaching the affected machine in order to continue the hijacking. They do this either through delaying or blocking the ACK packets. As such, this attack is very commonly employed with ARP poisoning, and we might notice the following in our traffic analysis.

```
[TCP Retransmission] 23 → 36212 [PSH, ACK]
[TCP Retransmission] 23 → 36212 [PSH, ACK]
[TCP Retransmission] 23 → 36212 [PSH, ACK]
```


Questions

Answer the question(s) below to complete this Section and earn cubes!

+ 1 Inspect the TCP-hijacking.pcap file, part of this module's resources, and enter the username that has been used through the telnet protocol as your answer.


administrator
Submit
Previous
Next
Mark Complete & Next
Powered by HACKTHEBOX