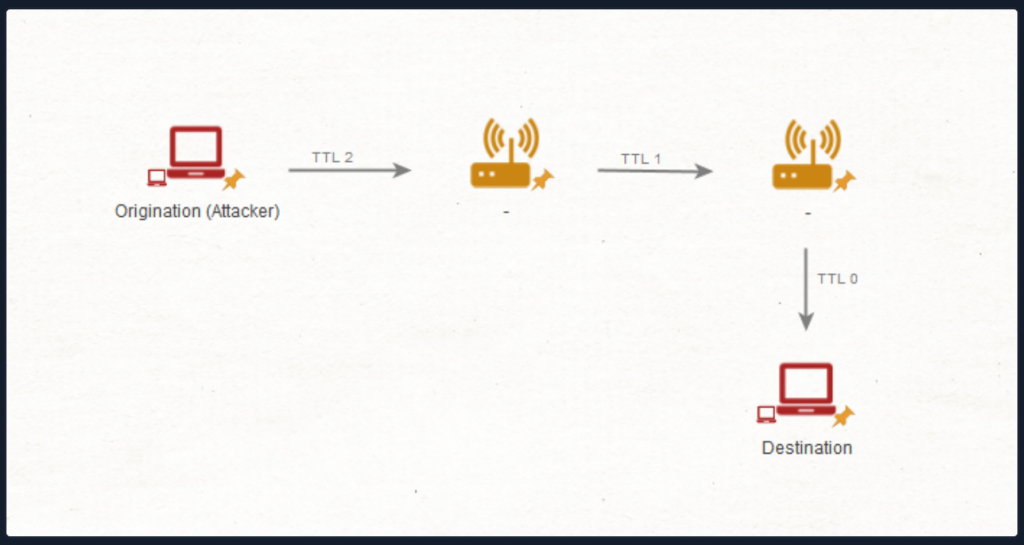
IP Time-to-Live Attacks

Related PCAP File(s):

- `ip_ttl.pcapng`

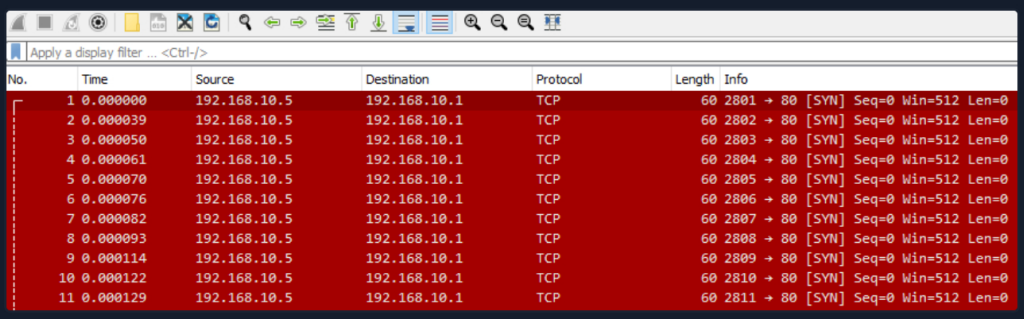
Time-to-Live attacks are primarily utilized as a means of evasion by attackers. Basically speaking the attacker will intentionally set a very low TTL on their IP packets in order to attempt to evade firewalls, IDS, and IPS systems. These work like the following.



1. The attacker will craft an IP packet with an intentionally low TTL value (1, 2, 3 and so on).
2. Through each host that this packet passes through this TTL value will be decremented by one until it reaches zero.
3. Upon reaching zero this packet will be discarded. The attacker will try to get this packet discarded before it reaches a firewall or filtering system to avoid detection/controls.
4. When the packets expire, the routers along the path generate ICMP Time Exceeded messages and send them back to the source IP address.

Finding Irregularities in IP TTL

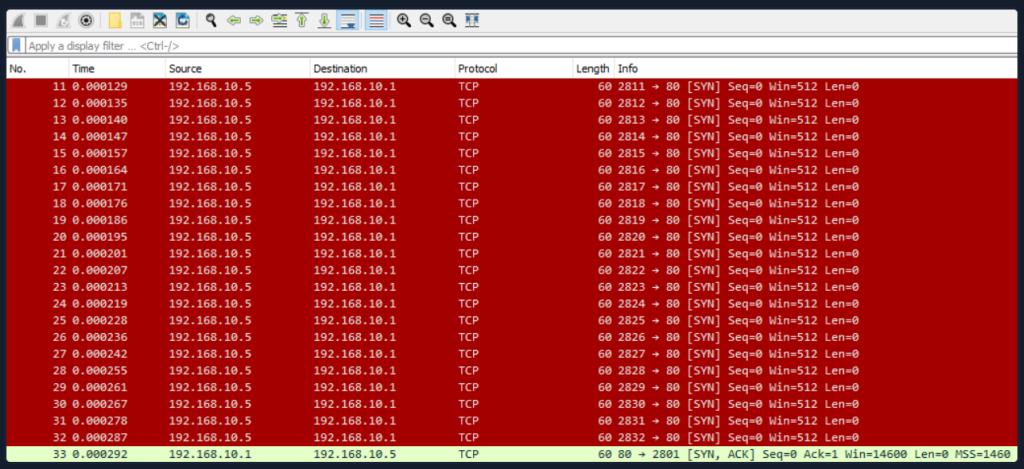
For starters, we can begin to dump our traffic and open it in Wireshark. Detecting this in small amounts can be difficult, but fortunately for us attackers will most times utilize ttl manipulation in port scanning efforts. Right away we might notice something like the following.


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

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.10.5	192.168.10.1	TCP	60	2801 → 80 [SYN] Seq=0 Win=512 Len=0
2	0.000039	192.168.10.5	192.168.10.1	TCP	60	2802 → 80 [SYN] Seq=0 Win=512 Len=0
3	0.000050	192.168.10.5	192.168.10.1	TCP	60	2803 → 80 [SYN] Seq=0 Win=512 Len=0
4	0.000061	192.168.10.5	192.168.10.1	TCP	60	2804 → 80 [SYN] Seq=0 Win=512 Len=0
5	0.000070	192.168.10.5	192.168.10.1	TCP	60	2805 → 80 [SYN] Seq=0 Win=512 Len=0
6	0.000076	192.168.10.5	192.168.10.1	TCP	60	2806 → 80 [SYN] Seq=0 Win=512 Len=0
7	0.000082	192.168.10.5	192.168.10.1	TCP	60	2807 → 80 [SYN] Seq=0 Win=512 Len=0
8	0.000093	192.168.10.5	192.168.10.1	TCP	60	2808 → 80 [SYN] Seq=0 Win=512 Len=0
9	0.000114	192.168.10.5	192.168.10.1	TCP	60	2809 → 80 [SYN] Seq=0 Win=512 Len=0
10	0.000122	192.168.10.5	192.168.10.1	TCP	60	2810 → 80 [SYN] Seq=0 Win=512 Len=0
11	0.000129	192.168.10.5	192.168.10.1	TCP	60	2811 → 80 [SYN] Seq=0 Win=512 Len=0

However, we might also notice a returned SYN, ACK message from one of our legitimate service ports on our affected host. In doing so, the attacker might have successfully evaded one of our firewall controls

host. In doing so, the attacker might have successfully evaded one of our firewall controls.



So, if we were to open one of these packets, we could realistically see why this is. Suppose we opened the IPv4 tab in Wireshark for any of these packets. We might notice a very low TTL like the following.

```
Internet Protocol Version 4, Src: 192.168.10.5, Dst: 192.168.10.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 40
    Identification: 0x7312 (29458)
  > 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
  v Time to Live: 3
      > [Expert Info (Note/Sequence): "Time To Live" only 3]
    Protocol: TCP (6)
    Header Checksum: 0xaf67 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.10.5
    Destination Address: 192.168.10.1
```

As such, we can implement a control which discards or filters packets that do not have a high enough TTL. In doing so, we can prevent these forms of IP packet crafting attacks.

← Previous Next → Mark Complete & Next