Wireshark Project Report — TCP 3-Way Handshake & Stealth Scan Analysis

Author: Oghenetega T. Gold

Role: Cybersecurity Analyst

Date: October 2025

Lab Host: kali (privileged)

Executive Summary

This report documents a Wireshark analysis performed to observe and validate a TCP 3-way handshake (SYN → SYN-ACK → ACK) between a host (10.0.2.15) and a destination server (147.79.120.186) on destination port 443. The capture demonstrates how to filter and inspect handshake packets, highlights common port-scan evasion techniques (SYN/stealth scans, decoy scans, time-fragmentation), and provides practical detection and remediation recommendations to reduce exposure from scanning and handshake-evasion techniques. Wireshark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development, and education.

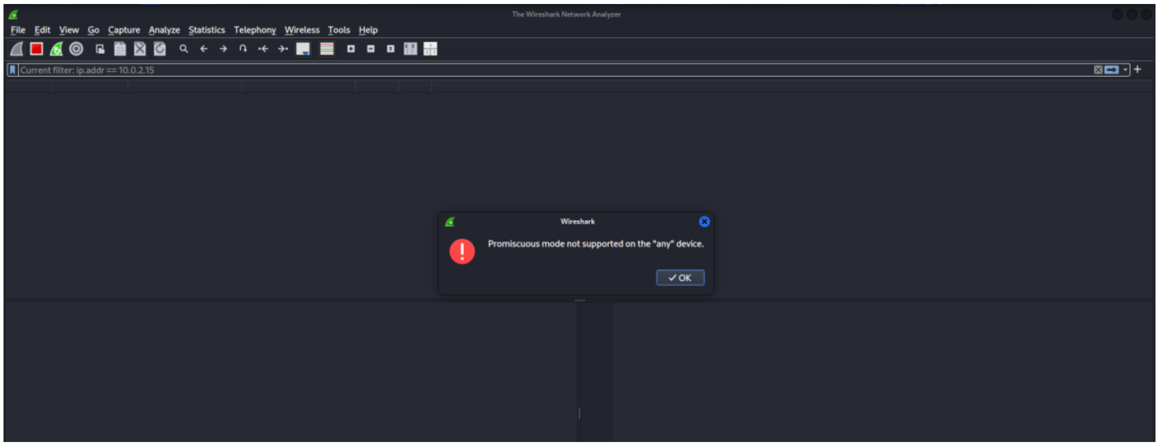
Objectives

- Capture and identify the complete TCP 3-way handshake (SYN, SYN-ACK, ACK) between 10.0.2.15 and 147.79.120.186 on port 443.
- Demonstrate Wireshark capture and filtering methodologies to isolate handshake and scan traffic.
- Explain common port-scan evasion techniques and how they interfere with handshake observation.
- Provide mitigation and remediation guidance (firewall rules and IDS tuning).

Environment & Capture Details

- Operating system / Capture host: Kali (running Wireshark with privileged capture).
- Wireshark: launched from Kali GUI/terminal (Wireshark interface).

- Target: officialnyscportal.com → 147.79.120.186 on TCP port 443 (HTTPS).
- Scanning tool used in lab**:** Nmap. Example command used for TCP connect scan on port 443:
- Capture mode: promiscuous mode on the interface used by the Kali host.


The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Current filter: ip.addr == 10.0.2.15

Wireshark

Promiscuous mode not supported on the "any" device.

Methodology — how the test was performed

- Start Wireshark on Kali to capture traffic on the network interface that will see the scan.
- Launch the Nmap scan against the target's TCP port 443:
- Observe the stream of packets in Wireshark. Thousands of packets may be visible depending on network activity. Narrow the view using display filters.
- Filter the capture to show only packets relevant to port 443 by using the filter :tcp.port == 443. Alternatively, we could filter by the target ip address using the filter: ip.addr == 147.79.120.186

Identifying the 3-Way Handshake

In a 3-way handshake, there are 4 TCP flags used in network communication to manage connection states. This includes;

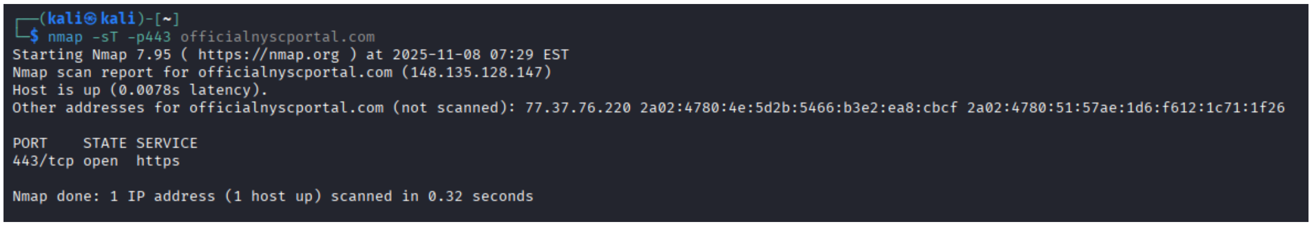
SYN (Synchronize): the initial packet sent by a client to request a connection.

SYN-ACK (Synchronize acknowledged): the server's response to acknowledge the request and send its own synchronization request.

ACK (Acknowledged)confirms a received packet

RST (Reset) is a packet used to immediately and abnormally terminate a connection when something is wrong.

To identify the TCP 3-way handshake, a full TCP scan was conducted using the -sT flag.



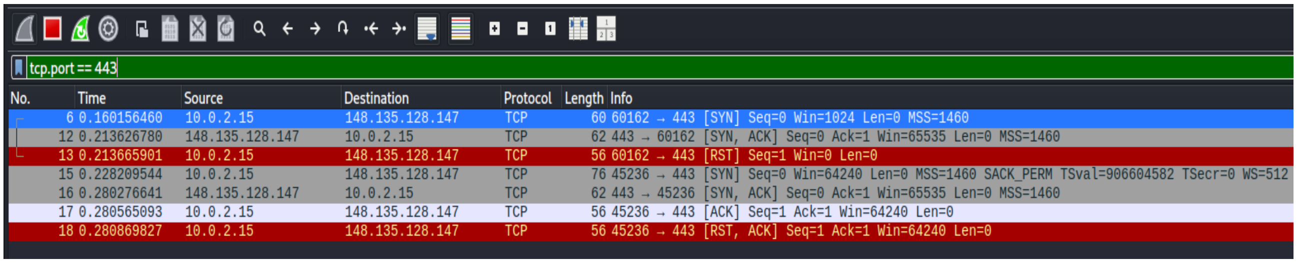
```
┌──(kali㉿kali)-[~]
└─$ nmap -sT -p443 officialnyscportal.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-08 07:29 EST
Nmap scan report for officialnyscportal.com (148.135.128.147)
Host is up (0.0078s latency).
Other addresses for officialnyscportal.com (not scanned): 77.37.76.220 2a02:4780:4e:5d2b:5466:b3e2:ea8:cbcf 2a02:4780:51:57ae:1d6:f612:1c71:1f26

PORT    STATE SERVICE
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 0.32 seconds
```

While the above command is prompted, Wireshark begins network analysis, which enables us identify the TCP 3 -way handshake. The TCP 3-way handshake can be identified by locating three packets in sequence:

1. SYN — Client (10.0.2.15) sends TCP segment with SYN bit set (tcp.flags.syn==1, tcp.flags.ack==0).

2. SYN-ACK — Server (147.79.120.186) replies with SYN+ACK (tcp.flags.syn==1, tcp.flags.ack==1).

3. ACK — Client (10.0.2.15) sends ACK (tcp.flags.ack==1, tcp.flags.syn==0) to complete the handshake.



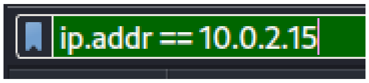
tcp.port == 443

No.	Time	Source	Destination	Protocol	Length	Info
6	0.160156460	10.0.2.15	148.135.128.147	TCP	60	60162 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
12	0.213626780	148.135.128.147	10.0.2.15	TCP	62	443 → 60162 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
13	0.213665901	10.0.2.15	148.135.128.147	TCP	56	60162 → 443 [RST] Seq=1 Win=0 Len=0
15	0.228209544	10.0.2.15	148.135.128.147	TCP	76	45236 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=906604582 TSecr=0 WS=512
16	0.280276641	148.135.128.147	10.0.2.15	TCP	62	443 → 45236 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
17	0.280565093	10.0.2.15	148.135.128.147	TCP	56	45236 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
18	0.280869827	10.0.2.15	148.135.128.147	TCP	56	45236 → 443 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0

Port Scanning & Filters Used

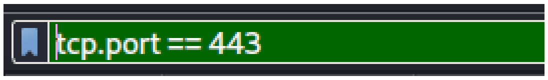
Port scanning was performed targeting port 443 on 147.79.120.186. To isolate scan traffic, the following Wireshark filters were used:

- To filter by Host Ip address, enter: ip.addr= =host ip address (in this case, 10.0.2.15)



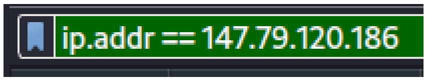
ip.addr == 10.0.2.15

- To filter by TCP ports, enter: tcp.port= =target port (in this case, 443)



tcp.port == 443

- To filter by target Ip address, enter: ip.addr= =target ip address (in this case, 147.79.120.186)

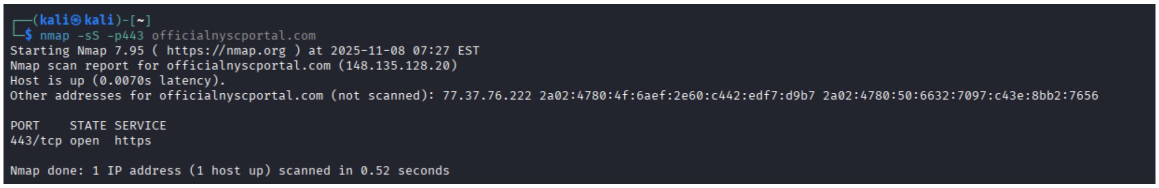


ip.addr == 147.79.120.186

Evasion Techniques Observed / Discussed

While inspecting handshake packets, be aware attackers commonly use methods to avoid easy detection. These techniques can make detection harder for simple signature-based IDS/IPS or manual review.

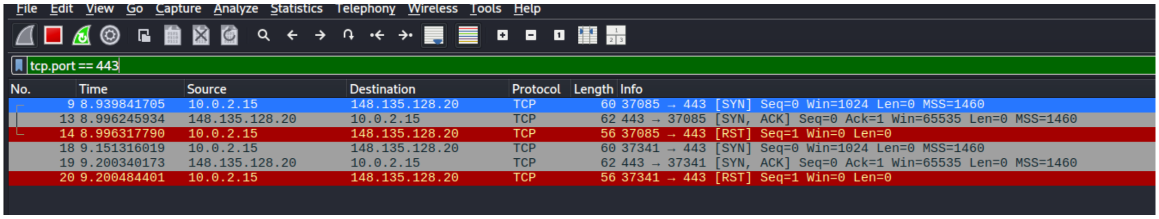
1) Stealth (SYN) Scans — using the -sS flag



```
┌──(kali㉿kali)-[~]
└─$ nmap -sS -p443 officialnyscportal.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-08 07:27 EST
Nmap scan report for officialnyscportal.com (148.135.128.20)
Host is up (0.0070s latency).
Other addresses for officialnyscportal.com (not scanned): 77.37.76.222 2a02:4780:4f:6aef:2e60:c442:edf7:d9b7 2a02:4780:50:6632:7097:c43e:8bb2:7656

PORT    STATE SERVICE
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 0.52 seconds
```

- Behavior: The attacker sends a SYN and analyzes the response. If SYN-ACK is received the port is open; attacker sends RST instead of completing the handshake (no final ACK), avoiding full connection establishment.
- Why it can bypass detection: Some naive detection rules look for completed handshakes or payloads; dropping the handshake completion can avoid certain logging. However, modern IDS/IPS and connection tracking usually detect large volumes of SYNs and incomplete handshakes (SYN floods or unusual SYN/SYN-ACK ratios).

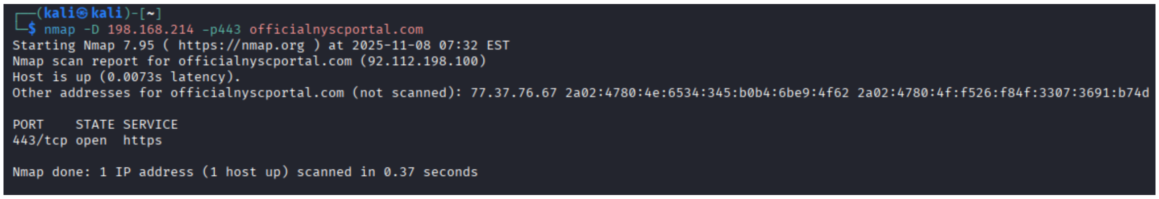
  See result below:



tcp.port == 443

No.	Time	Source	Destination	Protocol	Length	Info
9	8.939841705	10.0.2.15	148.135.128.20	TCP	60	37085 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
13	8.996245934	148.135.128.20	10.0.2.15	TCP	62	443 → 37085 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
14	8.996317790	10.0.2.15	148.135.128.20	TCP	56	37085 → 443 [RST] Seq=1 Win=0 Len=0
18	9.151316019	10.0.2.15	148.135.128.20	TCP	60	37341 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
19	9.200340173	148.135.128.20	10.0.2.15	TCP	62	443 → 37341 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
20	9.200484401	10.0.2.15	148.135.128.20	TCP	56	37341 → 443 [RST] Seq=1 Win=0 Len=0

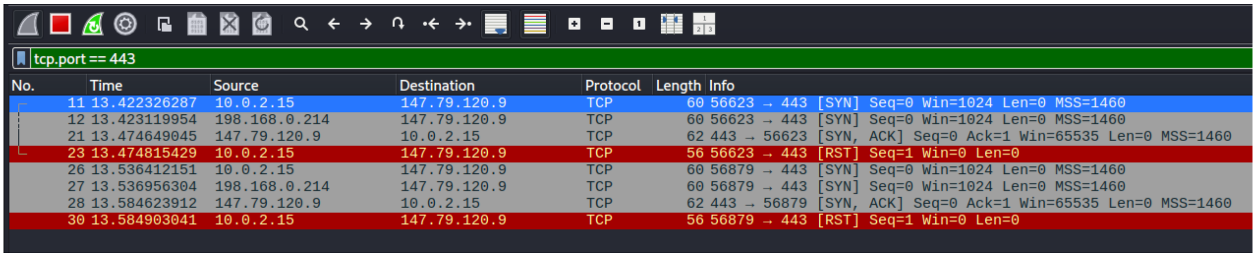
2) Decoy Scans – using the -D flag <ip address>



```
┌──(kali㉿kali)-[~]
└─$ nmap -D 198.168.214 -p443 officialnyscportal.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-08 07:32 EST
Nmap scan report for officialnyscportal.com (92.112.198.100)
Host is up (0.0073s latency).
Other addresses for officialnyscportal.com (not scanned): 77.37.76.67 2a02:4780:4e:6534:345:b0b4:6be9:4f62 2a02:4780:4f:f526:f84f:3307:3691:b74d

PORT    STATE SERVICE
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 0.37 seconds
```

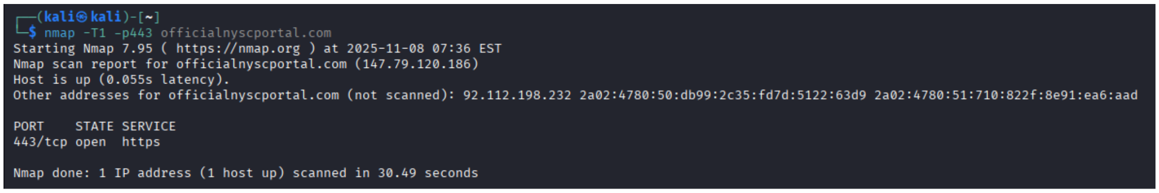
- Behavior: The attacker uses multiple spoofed source IPs (decoys) along with the true source to mix legitimate-looking traffic with malicious probes. This blends probe traffic with benign-looking flows, making attribution and detection harder.
- Why it can bypass detection: Alerts generated per-source may be diluted; threshold-based systems may not detect low-rate probes spread across many decoy IPs. Detection strategies: Correlate destination-side logs, look for identical probe patterns (same TTL, window size, TCP options), and use anomaly detection across multiple sources.

  See result below:



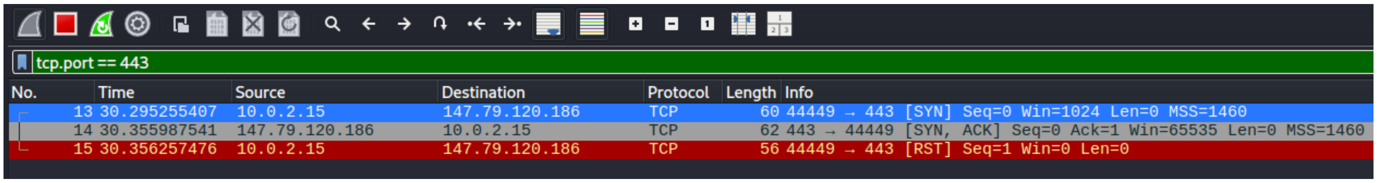
```
tcp.port == 443
No.   Time          Source         Destination    Protocol Length Info
   11 13.422326287  10.0.2.15      147.79.120.9   TCP          60 56623 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   12 13.423119954  198.168.0.214  147.79.120.9   TCP          60 56623 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   21 13.474649045  147.79.120.9   10.0.2.15      TCP          62 443 → 56623 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
   23 13.474815429  10.0.2.15      147.79.120.9   TCP          56 56623 → 443 [RST] Seq=1 Win=0 Len=0
   26 13.536412151  10.0.2.15      147.79.120.9   TCP          60 56879 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   27 13.536956304  198.168.0.214  147.79.120.9   TCP          60 56879 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
   28 13.584623912  147.79.120.9   10.0.2.15      TCP          62 443 → 56879 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
   30 13.584903041  10.0.2.15      147.79.120.9   TCP          56 56879 → 443 [RST] Seq=1 Win=0 Len=0
```

3) Time Fragmentation / Fragmented Scans- using the -T1 flag (also known as the sluggish scan)



```
┌──(kali㉿kali)-[~]
└─$ nmap -T1 -p443 officialnyscportal.com
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-08 07:36 EST
Nmap scan report for officialnyscportal.com (147.79.120.186)
Host is up (0.055s latency).
Other addresses for officialnyscportal.com (not scanned): 92.112.198.232 2a02:4780:50:db99:2c35:fd7d:5122:63d9 2a02:4780:51:710:822f:8e91:ea6:aad

PORT    STATE SERVICE
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 30.49 seconds
```

- Behavior: Attackers fragment packets or spread probe payloads across multiple small fragments and/or time the fragments to arrive slowly. Fragments may evade signature-based detection that inspects single packets.
- Why it can bypass detection: If IDS lacks full IP fragment reassembly or has limits on reassembly buffers/timeouts, the signature won't match. Time-based fragmentation spaces probes to avoid threshold-triggered alarms. Detection strategies: Enable full IP reassembly in IDS/Wireshark, tune reassembly timeouts, monitor unusual fragmentation patterns, and correlate with flow/session metrics. See result below:



tcp.port == 443

No.	Time	Source	Destination	Protocol	Length	Info
13	30.295255407	10.0.2.15	147.79.120.186	TCP	60	44449 → 443 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
14	30.355987541	147.79.120.186	10.0.2.15	TCP	62	443 → 44449 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
15	30.356257476	10.0.2.15	147.79.120.186	TCP	56	44449 → 443 [RST] Seq=1 Win=0 Len=0

Detection & Mitigation Recommendations

1. Use stateful network devices and enable connection tracking—this helps detect incomplete handshakes.

2. Enable IP fragment reassembly in IDS/IPS (and ensure adequate buffers/timeouts).

3. Correlate network flow telemetry (NetFlow/sFlow) with packet captures to detect distributed low-rate scans.

4. Implement rate limiting and SYN cookies to mitigate SYN-based evasions and floods.

5. Use behavioral detection (anomaly-based IDS) to spot patterns across decoys or time-sliced probes.

6. Log and centralize alerts; enrich with context (TCP options, TTL, packet sizes) for better

triage.

7. Deploy honeypots to attract scans and analyze attacker tools and techniques safely.

Remediation (simple actionable step)

As an immediate and simple remediation step for Windows endpoints, configure Windows Defender Firewall rules to explicitly allow only required outbound and inbound connections and log blocked attempts. Example high-level steps:

- Create inbound rule: block unsolicited TCP connections to sensitive ports except from trusted IP ranges.
- Create outbound rule: restrict which applications/ports can initiate outbound TCP connections.
- Enable and forward firewall logs to a central collector for analysis.

Conclusion

Using Wireshark to observe traffic to officialnyscportal.com (147.79.120.186) on TCP port 443 demonstrates how the TCP 3-way handshake (SYN, SYN-ACK, ACK) is recorded and inspected. Attackers can use stealthy scanning and fragmentation to hinder handshake observation, but combining packet captures, flow telemetry, stateful devices, and IDS/IPS tuning provides strong detection and mitigation capabilities. Implementing strict firewall rules and centralized logging are high-impact target mitigations.