

패킷 분석 보고서

Ver. 1.0

백 정 이

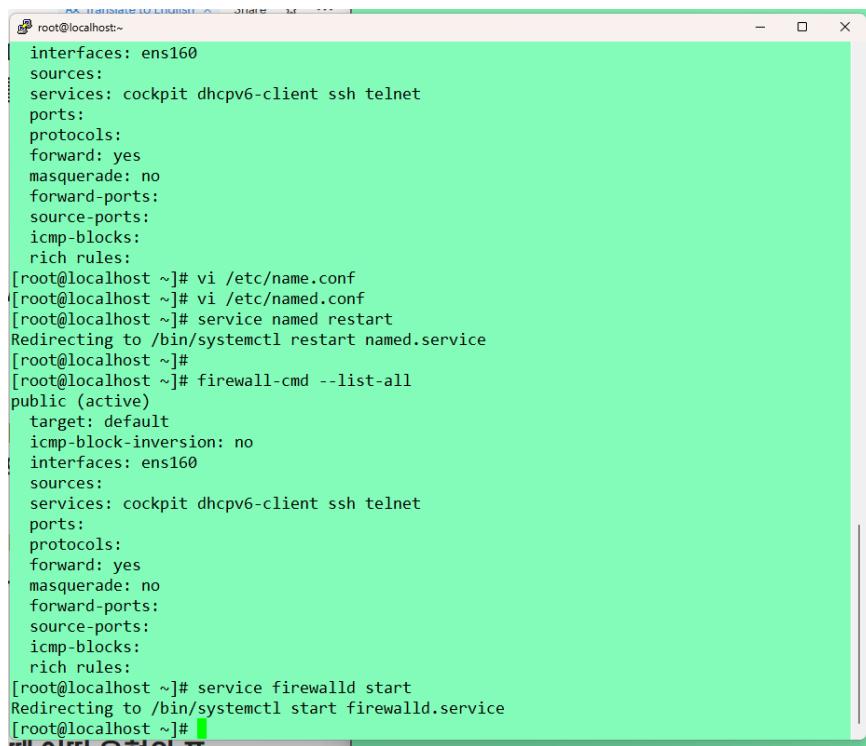
1. 포트스캔 결과 분석

1.1. 패킷 분석 목적

- 타겟의 포트를 nmap을 이용하여 TCP, UDP 스캔한 뒤 방화벽이 열렸을 때와 닫혔을 때의 wireshark 결과 차이 확인
- SYN 스캔과 FIN 스캔 실행 시 패킷 흐름 조사

1.2. 타겟 IP(10.10.10.10)의 1~100번 포트를 TCP SYN 스캔한 뒤 Wireshark 패킷 분석

- 방화벽 작동 시작



```
root@localhost ~]# vi /etc/hostname
root@localhost ~]# vi /etc/named.conf
root@localhost ~]# service named restart
Redirecting to /bin/systemctl restart named.service
[root@localhost ~]#
[root@localhost ~]# firewall-cmd --list-all
public (active)
  target: default
  icmp-block-inversion: no
  interfaces: ens160
  sources:
  services: cockpit dhcpcv6-client ssh telnet
  ports:
  protocols:
  forward: yes
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
[root@localhost ~]# service firewalld start
Redirecting to /bin/systemctl start firewalld.service
[root@localhost ~]#
```

● nmap으로 포트 스캔 시작

```
Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

[root@kali]-[~]
# nmap -sS -p 1-100 10.10.10.10
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-10-24 02:09 KST
Nmap scan report for 10.10.10.10
Host is up (0.023s latency).

Not shown: 92 filtered tcp ports (no-response), 6 filtered tcp ports (admin-prohibited)
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet

Nmap done: 1 IP address (1 host up) scanned in 2.42 seconds
```

```
[root@kali]-[~]
#
```

● 방화벽이 커져있을 때 패킷 확인

No.	Time	Source	Destination	Protocol	Length	Info
15	0.182277	10.10.10.10	20.20.20.20	ICMP	64	Standard query
17	0.184427	20.20.20.20	192.168.10.2	DNS	84	Standard query response
18	0.174661	192.168.10.2	20.20.20.20	DNS	84	Standard query response
19	0.186684	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
20	0.187799	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
21	0.188894	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
22	0.189989	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
23	0.190784	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
24	0.191712	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
25	0.191738	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
26	0.191781	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
27	0.191733	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
28	0.191749	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
29	0.192519	10.10.10.10	20.20.20.20	TCP	58	22 > 55983 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1468
30	0.205240	10.10.10.10	20.20.20.20	ICMP	86	Destination unreachable (Communication administratively filtered)
31	0.205240	10.10.10.10	20.20.20.20	TCP	58	23 > 55983 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1468
32	0.205254	10.10.10.10	20.20.20.20	ICMP	86	Destination unreachable (Communication administratively filtered)
33	0.205262	10.10.10.10	20.20.20.20	ICMP	86	Destination unreachable (Communication administratively filtered)
34	0.205271	10.10.10.10	20.20.20.20	ICMP	86	Destination unreachable (Communication administratively filtered)
35	0.205279	10.10.10.10	20.20.20.20	ICMP	86	Destination unreachable (Communication administratively filtered)
36	0.205634	10.10.10.10	20.20.20.20	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
37	0.206060	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
38	0.208147	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
39	0.209151	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
40	0.209159	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
41	0.209279	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
42	0.209332	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
43	0.209355	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
44	0.209412	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
45	0.209463	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
46	0.209553	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
47	0.209559	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
48	0.209584	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
49	0.209783	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
50	0.209832	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468
51	0.209846	20.20.20.20	10.10.10.10	TCP	60	Seq=0 Win=1024 Len=0 MSS=1468

> Frame 1: 118 bytes on wire (944 bits), 118 bytes captured (944 bits) on interface 1dev(eth0) [id=0x0000000000000000] at 2025-10-24 02:09:20.000000000 (0.000000000 seconds ago) (ethernet II, Src: VMware (00:0c:29:30:12:00), Dst: VMware (00:0c:29:30:12:00))

No.	Time	Source	Destination	Protocol	Length	Info
29	0.285199	10.10.10.10	20.20.20.20	TCP	58	22 > 55983 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1468
31	0.285249	10.10.10.10	20.20.20.20	TCP	58	23 > 55983 [SYN, ACK] Seq=0 Ack=1 Win=1024 Len=0 MSS=1468

칼리 리눅스(20.20.20.20)에서 록키 리눅스(10.10.10.10)로 1부터 100 번 포트에 syn 패킷을 보내고 반대로 syn + ack 응답은 방화벽에서 허용한 22, 23 번 포트에서 옴

● 방화벽이 꺼져있을 때 패킷 확인

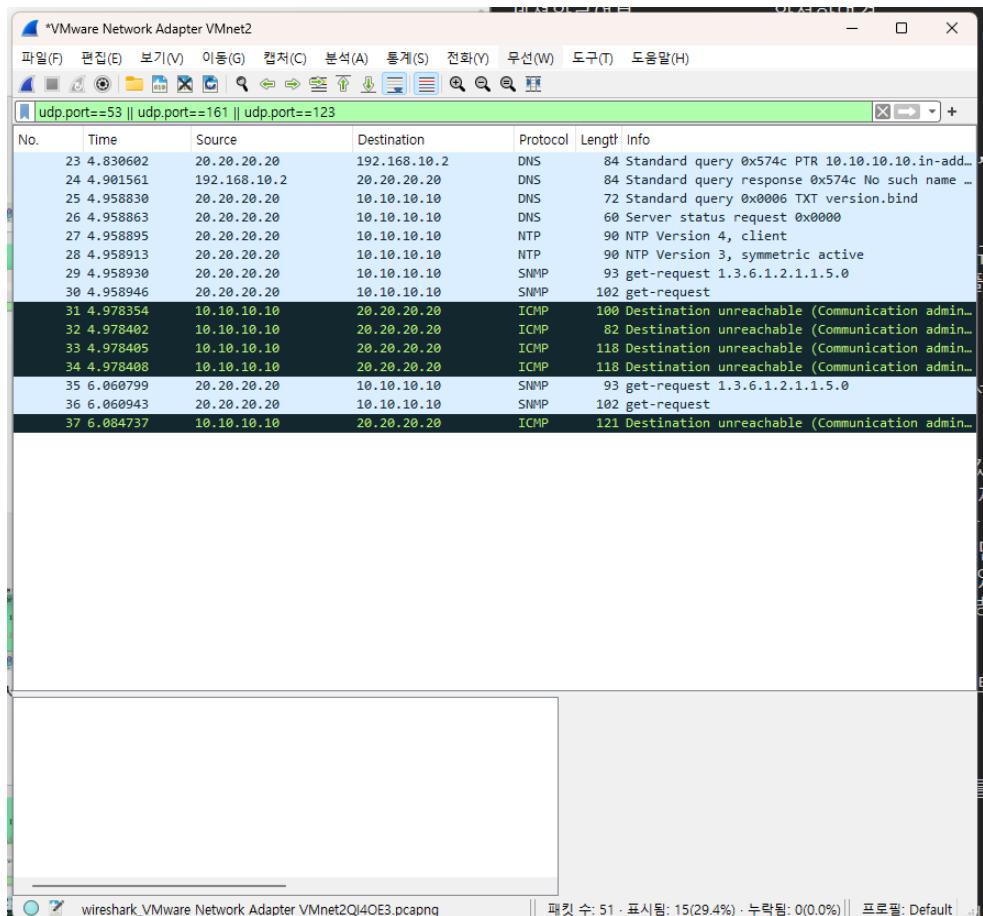
No.	Time	Source	Destination	Protocol	Length	Info
30	0.781262	10.10.10.10	20.20.20.20	TCP	58	53 → 55602 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460
31	0.781414	10.10.10.10	20.20.20.20	TCP	58	23 → 55602 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460
32	0.781453	10.10.10.10	20.20.20.20	TCP	58	80 → 55602 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460
33	0.781478	10.10.10.10	20.20.20.20	TCP	58	22 → 55602 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460
35	0.781519	10.10.10.10	20.20.20.20	TCP	58	21 → 55602 [SYN, ACK] Seq=0 Ack=1 Win=32120 Len=0 MSS=1460

서비스 가동중인 포트인 21, 22, 23, 53, 80 번 포트에서 syn+ack 응답이 옴

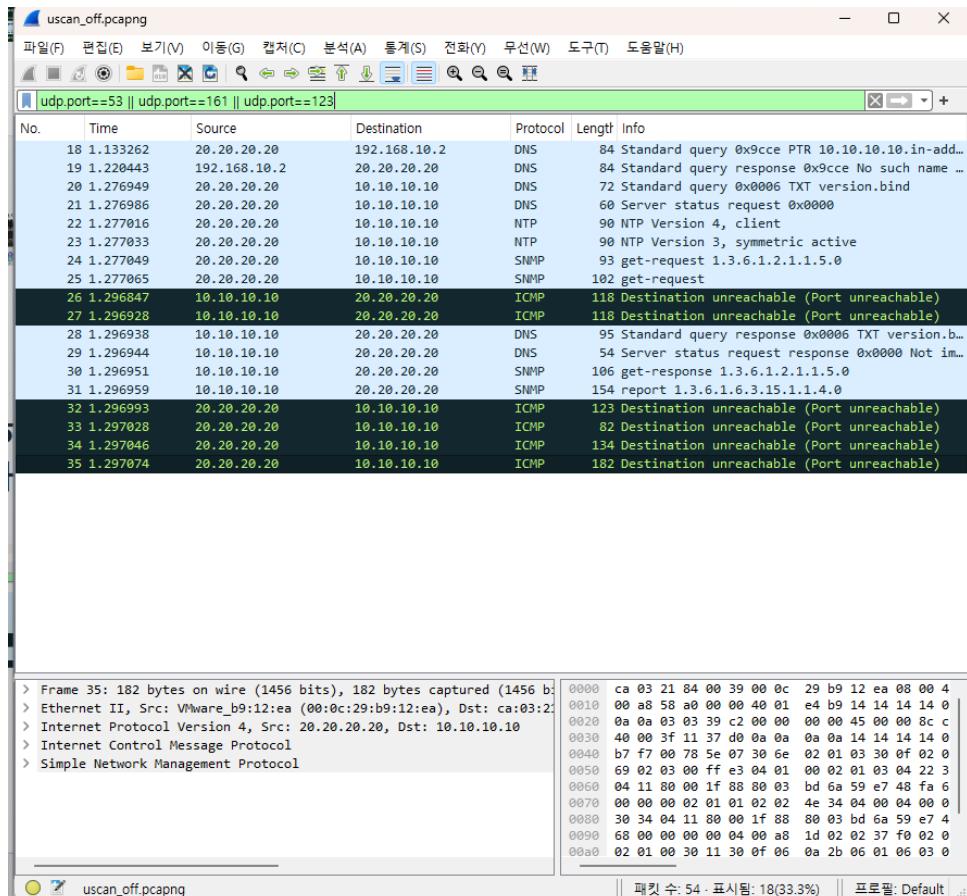
⇒ 열려 있는 포트에 syn 을 보내면 syn + ack 로 응답하고 닫혀있으면 응답하지 않음

1.3. UDP 포트 53, 161, 123을 스캔한 뒤 Wireshark 패킷 분석

● 방화벽이 꺼져있을 때 패킷 확인



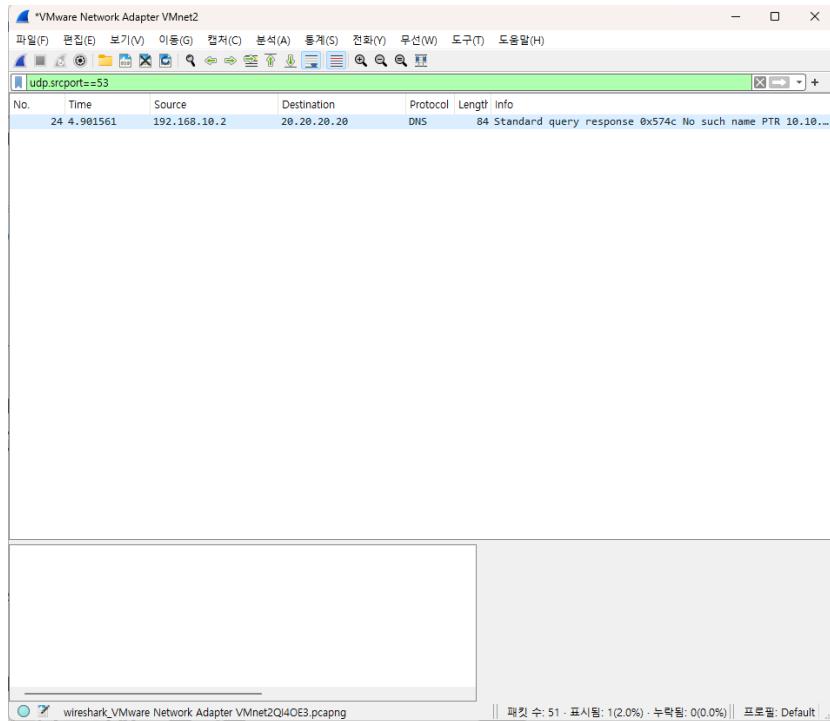
● 방화벽이 꺼져있을 때 패킷 확인



⇒ 방화벽이 켜져있을 때 Destination Unreachable 메시지가 오지만 꺼져있을 때 10.10.10.10 으로부터 DNS(53), SNMP(161) 패킷이 전송됨

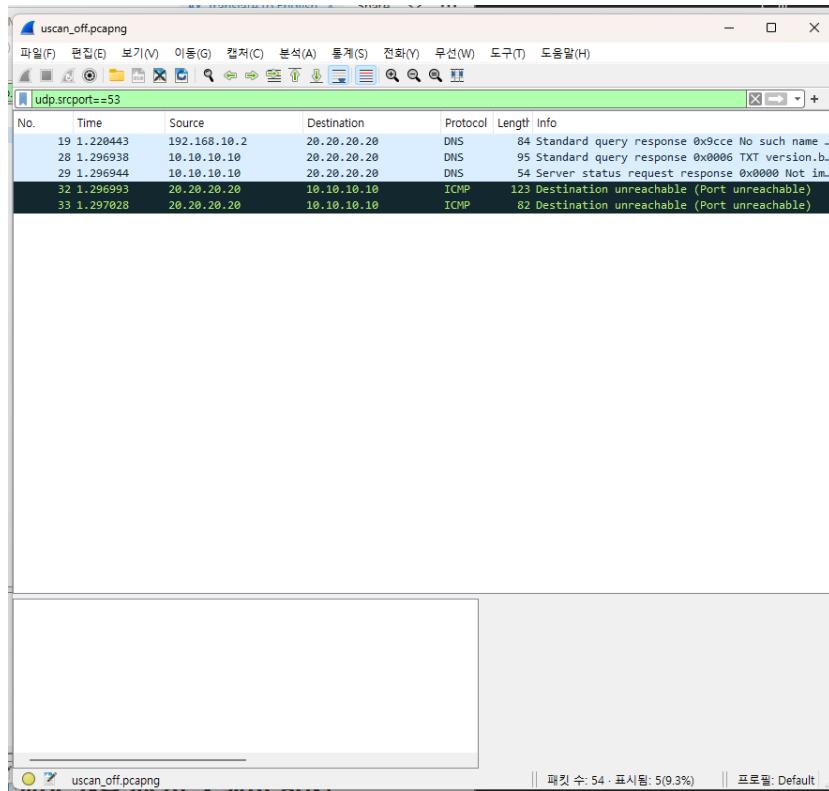
1.4. 출발지 포트를 53번으로 설정하여 DNS 요청으로 보이는 패킷을 Wireshark에서 분석

● 방화벽이 켜져있을 때 패킷 확인



dns(53) 포트에서 패킷이 전송되지 않음

● 방화벽이 꺼져있을 때 패킷 확인



DNS(53) 포트에서 패킷 전송됨

⇒ DNS 가 서비스되고 있지만 방화벽에 의해 차단됨

1.5. SYN 스캔의 wireshark 패킷 흐름과 특징

[공격자] → [서버] SYN

[서버] → [공격자] SYN+ACK

[공격자] → [서버] RST

- 연결 수립하지 않고 RST 를 보내므로 SYN SCAN 임
- 연결 수립하지 않으므로 탐지 회피가 쉬움

1.6. FIN 스캔의 wireshark 패킷 흐름과 wireshark 탐지 필터

[공격자] → [서버] FIN

[서버] → [공격자] (응답 없음)

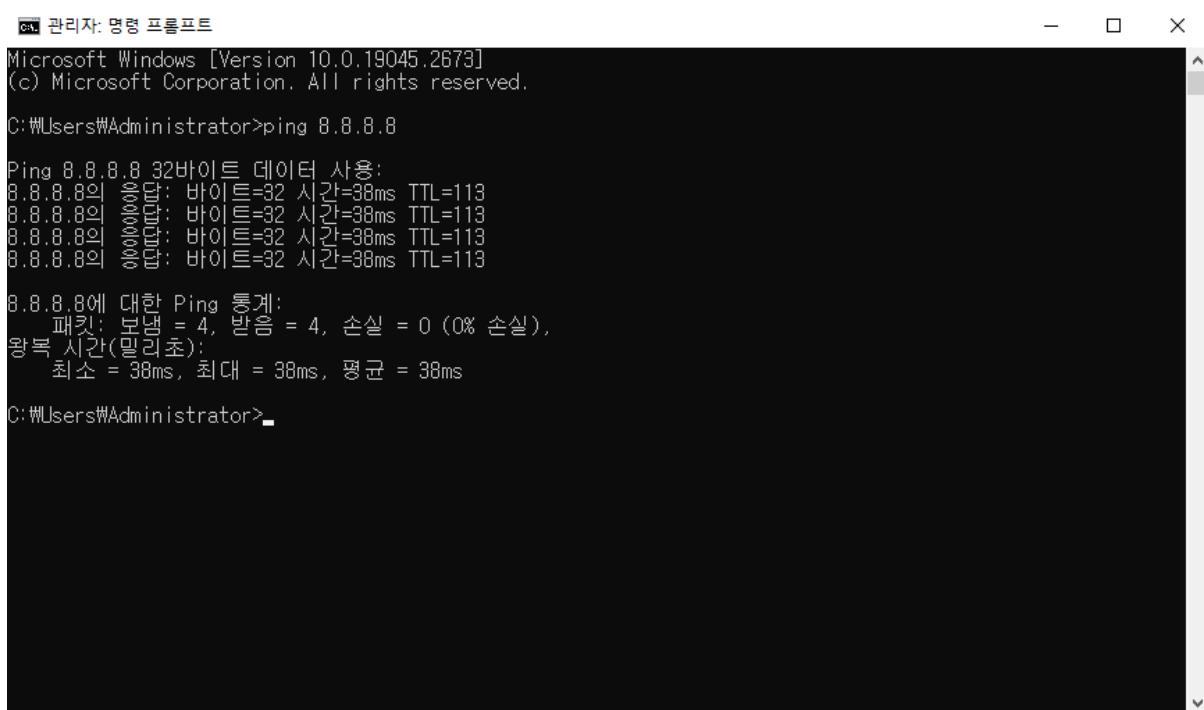
[공격자] → [서버] FIN

[서버] → [공격자] RST

- FIN 을 보냈을 때 포트가 열려있으면 응답이 없고, 닫혀있으면 RST + ACK 를 보내는 것을 통해 열린 포트를 확인할 수 있음
- 와이어샤크 탐지 필터 : tcp.flags.fin

2. ICMP 패킷 분석

- 8.8.8로 ping 보내기



```
관리자: 명령 프롬프트
Microsoft Windows [Version 10.0.19045.2673]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ping 8.8.8.8

Ping 8.8.8.8 32바이트 데이터 사용:
8.8.8.8의 응답: 바이트=32 시간=38ms TTL=113

8.8.8.8에 대한 Ping 통계:
    패킷: 보냄 = 4, 받음 = 4, 손실 = 0 (0% 손실),
    왕복 시간(밀리초):
        최소 = 38ms, 최대 = 38ms, 평균 = 38ms

C:\Users\Administrator>
```

● ICMP 필터링 결과 첫번째 패킷

icmp						
No.	Time	Source	Destination	Protocol	Length	Info
→ 62 2.579059		192.168.20.4	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=1/256, ttl=128 (reply in 63)
← 63 2.617890		8.8.8.8	192.168.20.4	ICMP	74	Echo (ping) reply id=0x0001, seq=1/256, ttl=113 (request in 62)
115 3.584459		192.168.20.4	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=2/512, ttl=128 (reply in 116)
116 3.622846		8.8.8.8	192.168.20.4	ICMP	74	Echo (ping) reply id=0x0001, seq=2/512, ttl=113 (request in 115)
142 4.594881		192.168.20.4	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=3/768, ttl=128 (reply in 143)
143 4.633784		8.8.8.8	192.168.20.4	ICMP	74	Echo (ping) reply id=0x0001, seq=3/768, ttl=113 (request in 142)
161 5.604064		192.168.20.4	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=4/1024, ttl=128 (reply in 163)
163 5.642773		8.8.8.8	192.168.20.4	ICMP	74	Echo (ping) reply id=0x0001, seq=4/1024, ttl=113 (request in 161)


```

> Frame 62: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{92708C19-2217-4AE2-8544-098ACC90D322}, id 0
> Ethernet II, Src: MicroStarINT_cb:4b:e9 (d8:bb:c1:cb:4b:e9), Dst: Secuicom_23:30:19 (00:05:66:23:30:19)
> Internet Protocol Version 4, Src: 192.168.20.4, Dst: 8.8.8.8
└ Internet Control Message Protocol
  Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x4d5a [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 1 (0x0001)
    Sequence Number (LE): 256 (0x0100)
    [Response frame: 63]
  Data (32 bytes)

```

- ICMP 헤더에 type : 8 이 표시되고, IPv4 헤더에 Src : 192.168.20.4, Dst : 8.8.8.8 가 표시됨
- 목적지를 google DNS(8.8.8.8)으로 하여 ping 요청(ICMP Echo Request)을 보내는 것을 알 수 있음

● ICMP 필터링 결과 두번째 패킷

icmp						
No.	Time	Source	Destination	Protocol	Length	Info
→ 62 2.579059		192.168.20.4	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=1/256, ttl=128 (reply in 63)
← 63 2.617890		8.8.8.8	192.168.20.4	ICMP	74	Echo (ping) reply id=0x0001, seq=1/256, ttl=113 (request in 62)
115 3.584459		192.168.20.4	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=2/512, ttl=128 (reply in 116)
116 3.622846		8.8.8.8	192.168.20.4	ICMP	74	Echo (ping) reply id=0x0001, seq=2/512, ttl=113 (request in 115)
142 4.594881		192.168.20.4	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=3/768, ttl=128 (reply in 143)
143 4.633784		8.8.8.8	192.168.20.4	ICMP	74	Echo (ping) reply id=0x0001, seq=3/768, ttl=113 (request in 142)
161 5.604064		192.168.20.4	8.8.8.8	ICMP	74	Echo (ping) request id=0x0001, seq=4/1024, ttl=128 (reply in 163)
163 5.642773		8.8.8.8	192.168.20.4	ICMP	74	Echo (ping) reply id=0x0001, seq=4/1024, ttl=113 (request in 161)


```

> Frame 63: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{92708C19-2217-4AE2-8544-098ACC90D322}, id 0
> Ethernet II, Src: Secuicom_23:30:19 (00:05:66:23:30:19), Dst: MicroStarINT_cb:4b:e9 (d8:bb:c1:cb:4b:e9)
> Internet Protocol Version 4, Src: 8.8.8.8, Dst: 192.168.20.4
└ Internet Control Message Protocol
  Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x555a [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 1 (0x0001)
    Sequence Number (LE): 256 (0x0100)
    [Request frame: 62]
    [Response time: 38.831 ms]
  Data (32 bytes)

```

- ICMP 헤더에 typ : 0, IPv4 헤더에 Src : 8.8.8.8, Dst : 192.168.20.4 가 표시됨
- google DNS(8.8.8.8)으로부터 목적지 호스트(192.168.20.4)로 ping 응답(ICMP Echo Reply)을 보내는 것을 알 수 있음

3. TCP 패킷 분석

- google chrome 브라우저에서 “라우팅” 검색

라우팅(Routing)은 네트워크 내에서 데이터 패킷을 가장 최적의 경로로 전송하기 위해 경로를 선택하고 결정하는 과정입니다. 이는 인터넷과 같은 컴퓨터 네트워크나 공중전화망(PSTN)과 같은 다양한 네트워크에서 수행되며, 라우터라는 특수한 하드웨어 장비가 라우팅 테이블을 참조하여 최적의 경로를 찾아 패킷을 전달합니다.

라우팅의 핵심 요소

- **라우터(Router)**: 라우팅 결정을 내리고 패킷을 특정 경로로 전달하는 네트워크 장치입니다.
- **라우팅 테이블(Routing Table)**: 라우터가 패킷을 전달하기 위해 목적

[더보기 ▾](#)

비전공자도 이해할 수 있는 쉬운 라우팅 개념

2021. 3. 20. — 사전적인 의미로 라우팅, 즉 Routing은 Route를 하는 것을 의미합니다. Route는 명사로 한 곳에서 다른 곳으로 가기 위한 길, 경로라고 적혀 있죠? 여기서 ...

라우팅이란 무엇인가요? - 네트워크 라우팅 설명 - AWS

라우팅은 네트워크에서 경로를 선택하는 프로세스입니다. 컴퓨터 네트워크는 노드라고 하는 여러 시스템과 이러한 노드를 연결하는 경로 또는 링크로 구성됩니다.

Cloudflare

라우팅

Network 1, Network 2, Network 3

Static Routing / Default Routing

192.168.20.4, 172.16.21.00

172.16.2.0/24

● google chrome 브라우저에서 “라우팅” 검색 시 캡처한 TCP 패킷

No.	Time	Source	Destination	Protocol	Lengt	Info
29	1.847486	192.168.20.4	216.239.34.157	TCP	60	53850 + 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
33	1.908048	216.239.34.157	192.168.20.4	TCP	60	443 + 53850 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
34	1.908086	192.168.20.4	216.239.34.157	TCP	54	53850 + 443 [ACK] Seq=1 Ack=1 Win=262400 Len=0
35	1.908415	192.168.20.4	216.239.34.157	TLSv1.3	1848	Client Hello (SNI=tunnel.googlezip.net)
59	1.968029	216.239.34.157	192.168.20.4	TCP	60	443 + 53850 [ACK] Seq=1 Ack=1795 Win=268032 Len=0
62	1.995106	216.239.34.157	192.168.20.4	TLSv1.3	1466	Server Hello, Change Cipher Spec
63	1.995106	216.239.34.157	192.168.20.4	TCP	1466	443 + 53850 [PSH, ACK] Seq=1413 Ack=1795 Win=268032 Len=1412 [TCP PDU reassembled in 66]
64	1.995106	216.239.34.157	192.168.20.4	TCP	1466	443 + 53850 [ACK] Seq=2825 Ack=1795 Win=268032 Len=1412 [TCP PDU reassembled in 66]
65	1.995106	216.239.34.157	192.168.20.4	TCP	1466	443 + 53850 [PSH, ACK] Seq=4237 Ack=1795 Win=268032 Len=1412 [TCP PDU reassembled in 66]
66	1.995106	216.239.34.157	192.168.20.4	TLSv1.3	192	Application Data
67	1.995132	192.168.20.4	216.239.34.157	TCP	54	53850 + 443 [ACK] Seq=1795 Ack=5787 Win=262400 Len=0
68	1.995987	192.168.20.4	216.239.34.157	TLSv1.3	128	Change Cipher Spec, Application Data
69	1.996088	192.168.20.4	216.239.34.157	TLSv1.3	148	Application Data
70	1.996103	192.168.20.4	216.239.34.157	TLSv1.3	249	Application Data
130	2.057059	216.239.34.157	192.168.20.4	TCP	60	443 + 53850 [ACK] Seq=5787 Ack=2156 Win=267776 Len=0
188	2.082555	216.239.34.157	192.168.20.4	TLSv1.3	995	Application Data, Application Data
189	2.082703	192.168.20.4	216.239.34.157	TLSv1.3	85	Application Data
190	2.083076	216.239.34.157	192.168.20.4	TLSv1.3	98	Application Data
228	2.129371	192.168.20.4	216.239.34.157	TCP	54	53850 + 443 [ACK] Seq=2187 Ack=6772 Win=261632 Len=0
236	2.147043	216.239.34.157	192.168.20.4	TCP	60	443 + 53850 [ACK] Seq=6772 Ack=2187 Win=267776 Len=0
240	2.175174	216.239.34.157	192.168.20.4	TLSv1.3	133	Application Data
241	2.175556	192.168.20.4	216.239.34.157	TLSv1.3	1809	Application Data
260	2.235464	216.239.34.157	192.168.20.4	TCP	60	443 + 53850 [ACK] Seq=6851 Ack=3942 Win=266240 Len=0
388	2.350809	216.239.34.157	192.168.20.4	TLSv1.3	1466	Application Data

```

> Frame 35: 1848 bytes on wire (14784 bits), 1848 bytes captured (14784 bits) on interface \Device\NPF_{92708C19-2217-4AE2-8544-098ACC900322}, id 0
> Ethernet II, Src: MicroStarINT_c8:4b:e9 (d8:bb:c1:cb:4b:e9), Dst: Secuicom_23:30:19 (00:05:66:23:30:19)
> Internet Control Message Protocol Version 4, Src Port: 53850, Dst Port: 443, Seq: 1, Ack: 1, Len: 1794
> Transmission Control Protocol, Src Port: 53850, Dst Port: 443, Seq: 1, Ack: 1, Len: 1794
> Transport Layer Security
  > TLSv1.3 Record Layer: Handshake Protocol: Client Hello
    Content Type: Handshake (22)
    Version: TLS 1.0 (0x0301)
    Length: 1789
    > Handshake Protocol: Client Hello
      Handshake Type: Client Hello (1)
      Length: 1785
      > Version: TLS 1.2 (0x0303)
        Random: 7d081bb0d9522f467d3d52e1f06ca5590cfcaed79880264d64d38ccafeb6180
        Session ID Length: 32
        Session ID: e3996c34fc9a344294fc26b2f4effb929491d030b8b14fb6f64650d3db8c6641
        Cipher Suites Length: 32
        Cipher Suites (16 suites)
        Compression Methods Length: 1
        Compression Methods (1 method)
        Extensions Length: 1680
        Extension: Reserved (GREASE) (len=0)
        Extension: server_name (len=25) name=tunnel.googlezip.net
        Extension: renegotiation_info (len=1)
        Extension: signed_certificate_timestamp (len=0)
        Extension: ec_point_formats (len=2)
        Extension: supported_versions (len=7) TLS 1.3, TLS 1.2

```

- chrome 사용시 TLS1.3 버전 프로토콜을 사용
- 목적지 서버의 포트번호는 443 번, 클라이언트의 포트번호는 53850 번임
- 처음 연결 수립시 TCP 3 way handshake 과정을 거친 후 client hello, server hello 를 주고받음
- 처음 클라이언트는 Client Hello, Change Cipher Spec 패킷(seq : 1, ack : 1)을 보내고, 이에 대한 응답으로 서버는 Server Hello(seq : 1, ack : 1795)를 전송
- 서버가 클라이언트에게 Application Data(seq : 5649, ack : 1795)를 한 번 보낸 뒤 클라이언트가 서버에게 Change Cipher Spec 패킷(seq : 1795, ack : 5787) 전송(Change Cipher Spec : 클라이언트와 서버가 협상한 암호 명세를 이후부터 적용 혹은 변경함을 알림)
- 그 이후 Application Data 를 서버와 클라이언트가 주고받음

● edge 브라우저에서 네이버에서 “라우팅” 검색

The screenshot shows the Naver search results page for the query "라우팅".

AI 브리핑: AI 브리핑 실험 단계로 정확하지 않을 수 있어요. ①

라우팅은 네트워크에서 데이터 패킷이 최적의 경로를 통해 목적지까지 전달되도록 경로를 선택하는 과정을 의미합니다. ① ②

주요 개념:

- 1. 라우팅의 정의와 역할**
 - 라우팅은 패킷 스위칭 네트워크에서 패킷이 출발지에서 최종 목적지까지 효율적으로 전달되도록 중간 노드(라우터 등)가 경로를 결정하는 과정입니다. ① ②
 - 네트워크 주소와 라우팅 테이블을 기반으로 경로를 선택하며, 최적의 경로는 보통 최단 거리나 최소 시간 기준으로 결정됩니다. ① ③
- 2. 라우팅 방식**
 - 정적 라우팅: 네트워크 관리자가 수동으로 경로를 설정하는 방식입니다. ② ④
 - 동적 라우팅: 라우터가 네트워크 상태 변화에 따라 자동으로 경로를 설정하고 업데이트 하는 방식입니다. ② ④
 - 라우팅 프로토콜: RIP, OSPF, BGP 등 다양한 프로토콜이 있으며, 내부망(RIP, OSPF 등)과 외부망(BGP 등)에 따라 구분됩니다. ③ ⑤ ⑥
- 3. 라우팅 테이블**
 - 라우터는 목적지 주소, 서브넷 마스크, 다음 흙(다음 전달 노드) 정보를 저장하는 라우팅 테이블을 사용해 패킷을 전달합니다. ④ ⑦

4. 라우팅 알고리즘

펼쳐서 더보기 ▾

ko.wordow.com > english

routing 뜻 - 영어 사전 | routing 의미 해석

에서 한국어 내부, 우리는 어떻게 설명 할**routing**영어 단어 그것은? **routing**영어 단어는 다음과 같은 의미를 한국어 :라우팅 라우팅 라우팅(영어: **routing**)은 어떤 네트워크 안에서

● edge 브라우저에서 네이버에서 “라우팅” 검색 시 캡처한 TCP 패킷

No.	Time	Source	Destination	Protocol	Lengt	Info
172	1.752021	192.168.20.4	223.130.195.167	TCP	60	54093 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
173	1.755037	110.93.159.43	192.168.20.4	TCP	60	443 → 54075 [ACK] Seq=1 Ack=2 Win=16 Len=0
174	1.755037	223.130.192.205	192.168.20.4	TCP	60	443 → 54084 [ACK] Seq=1 Ack=2 Win=319 Len=0
175	1.756057	223.130.195.167	192.168.20.4	TCP	60	443 → 54093 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1440 SACK_PERM WS=128
176	1.756096	192.168.20.4	223.130.195.167	TCP	54	54093 → 443 [ACK] Seq=1 Ack=1 Win=263424 Len=0
177	1.756359	192.168.20.4	223.130.195.167	TLSv1.2	1976	[Client Hello (SNI=static.nid.naver.com)]
178	1.758771	223.130.195.167	192.168.20.4	TCP	60	443 → 54093 [ACK] Seq=1 Ack=1441 Win=32128 Len=0
179	1.759053	223.130.195.167	192.168.20.4	TCP	60	443 → 54093 [ACK] Seq=1 Ack=1923 Win=35672 Len=0
180	1.759992	223.130.195.167	192.168.20.4	TLSv1.2	1514	Server Hello
181	1.760191	223.130.195.167	192.168.20.4	TCP	1514	443 → 54093 [ACK] Seq=1461 Ack=1923 Win=358072 Len=1460 [TCP PDU reassembled in 183]
182	1.760191	223.130.195.167	192.168.20.4	TCP	1514	443 → 54093 [ACK] Seq=2921 Ack=1923 Win=358072 Len=1460 [TCP PDU reassembled in 183]
183	1.760191	223.130.195.167	192.168.20.4	TLSv1.2	955	Certificate, Server Key Exchange, Server Hello Done
184	1.760214	192.168.20.4	223.130.195.167	TCP	54	54093 → 443 [ACK] Seq=1923 Ack=5282 Win=263424 Len=0
185	1.760502	192.168.20.4	223.130.195.167	TLSv1.2	186	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
186	1.760637	192.168.20.4	223.130.195.167	TLSv1.2	153	Application Data
187	1.760751	192.168.20.4	223.130.195.167	TLSv1.2	688	Application Data
188	1.762806	223.130.195.167	192.168.20.4	TLSv1.2	312	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
189	1.762888	223.130.195.167	192.168.20.4	TLSv1.2	123	Application Data
190	1.762900	192.168.20.4	223.130.195.167	TCP	54	54093 → 443 [ACK] Seq=2782 Ack=5609 Win=263168 Len=0
191	1.762998	192.168.20.4	223.130.195.167	TLSv1.2	92	Application Data
192	1.763044	223.130.195.167	192.168.20.4	TCP	60	443 → 54093 [ACK] Seq=5609 Ack=2782 Win=37888 Len=0
193	1.763044	223.130.195.167	192.168.20.4	TLSv1.2	92	Application Data
194	1.764069	223.130.195.167	192.168.20.4	TLSv1.2	502	Application Data
195	1.764082	192.168.20.4	223.130.195.167	TCP	54	54093 → 443 [ACK] Seq=2820 Ack=6095 Win=262656 Len=0

> Frame 177: 1976 bytes on wire (15808 bits), 1976 bytes captured (15808 bits) on interface \Device\WPF_{92708C19-2217-4AE2-8544-098ACC90D322}, id 0
> Ethernet II, Src: MicroStarINT_c8:e9 (d8:bb:c1:cb:4b:e9), Dst: Secuicom_23:30:19 (00:05:66:23:30:19)
> Internet Protocol Version 4, Src: 192.168.20.4, Dst: 223.130.195.167
> Transmission Control Protocol, Src Port: 54093, Dst Port: 443, Seq: 1, Ack: 1, Len: 1922
`- Transport Layer Security
 `- TLSv1.2 Record Layer: Handshake Protocol: Client Hello
 Content Type: Handshake (22)
 Version: TLS 1.0 (0x0301)
 Length: 1917
 `- Handshake Protocol: Client Hello
 Handshake Type: Client Hello (1)
 Length: 1913
 `- Version: TLS 1.2 (0x0303)
 `- Random: 1153601f9c01b0097b97ac7dd7352b3d47d0c136784f188629a86a1a40b653eb
 Session ID Length: 32
 Session ID: bb63c61470178ac2a81966ad05399ea7a80fcfb7a6612d337fa5786bc62ec013
 Cipher Suites Length: 32
 Cipher Suites (16 suites)
 Compression Methods Length: 1
 Compression Methods (1 method)
 Extensions Length: 1808
 Extension: Reserved (GREASE) (len=0)
 Extension: key_share (len=1263) Unknown (4588), x25519
 Extension: extended_master_secret (len=0)
 Extension: psk_key_exchange_modes (len=2)
 Extension: encrypted_client_hello (len=186)
 Extensions: supported_ciphersuites (len=17)

- 네이버에서는 TLS1.2 버전 프로토콜을 사용
- 클라이언트(192.168.20.4)와 서버(223.130.195.167)가 TCP 3-way handshake 수행 후 데이터를 주고받음
- 목적지 서버의 포트번호는 443 번, 클라이언트의 포트번호는 54093 번임
- 3 way handshake 과정에서 처음 SYN 을 보낼 때는 seq 번호가 0 임
- 이에 대한 응답으로 서버가 클라이언트에게 syn + ack 를 보낼 때는 seq 번호가 0, ack 번호가 1 임
- 다시 클라이언트가 서버에게 ack 를 보낼 때는 seq 번호가 1, ack 번호가 1 임
- 구글에서 검색했을 때와 마찬가지로 3 way handshake 과정을 거친 후 TLS handshake 과정을 거침
- client hello 패킷에 접속하려는 도메인 이름(static.nid.naver.com)이 표시됨
- client hello, server hello 패킷을 주고받은 후 서버가 클라이언트에게 Certificate, Server Key Exchange, Server Hello Done(seq : 4381, ack : 1923) 전송
 - Certificate: 서버의 인증서와 서버 인증서에 서명한 인증기관들의 인증서 목록을 클라이언트에게 전달
 - Server Key Exchange: 키 교환에 필요한 정보를 전달, ex) Diffie-Hellman 매개변수

- Server Hello Done : Server Hello 과정을 종료함을 알림
- 클라이언트는 서버에게 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message(seq : 1923, ack : 5282) 전송
 - Client Key Exchange : 서버의 인증서를 검증한 후 앞으로 사용할 세션키 생성을 위한 pre-master secret 을 서버 인증서의 공개키로 암호화하여 전송, 혹은 Diffie-Hellman 공개키를 생성하고 서버에 전송한 후 클라이언트와 서버가 각각 Diffie-Hellman 연산을 통해 공통의 pre-master secret 생성
 - Encrypted Handshake Message : 클라이언트 측의 협상을 종료

4. UDP 패킷 분석

- edge 브라우저에서 네이버에서 “라우팅” 검색 시 캡처한 UDP 패킷

No.	Time	Source	Destination	Protocol	Lengt	Info
2	0.464992	192.168.12.142	239.255.255.250	SSDP	143	M-SEARCH * HTTP/1.1
15	0.756590	192.168.3.253	255.255.255.255	UDP	215	37084 > 7421 [HTTP/1.1]
20	0.756593	192.168.3.22	239.255.255.250	SSDP	173	M-SEARCH * HTTP/1.1
21	1.254254	192.168.20.4	168.126.63.1	DNS	76	Standard query 0x00da A search.naver.com
22	1.254369	192.168.20.4	168.126.63.1	DNS	76	Standard query 0xac44 A search.naver.com
23	1.254896	192.168.20.4	110.93.159.35	UDP	1231	54332 + 443 Len=1189
24	1.256915	168.126.63.1	192.168.20.4	DNS	212	Standard query response 0x00da A search.naver.com CNAME search.naver.com.nheos.com CNAME Gece37f3.ndash.net A 12?
25	1.256915	168.126.63.1	192.168.20.4	DNS	212	Standard query response 0xac44 A search.naver.com CNAME search.naver.com.nheos.com CNAME Gece37f3.ndash.net A 12?
26	1.273038	110.93.159.35	192.168.20.4	UDP	867	443 + 54332 Len=825
27	1.273405	110.93.159.35	192.168.20.4	UDP	284	443 + 54332 Len=252
28	1.273568	110.93.159.35	192.168.20.4	UDP	284	443 + 54332 Len=242
29	1.273584	110.93.159.35	192.168.20.4	UDP	1294	443 + 54332 Len=1252
30	1.273584	110.93.159.35	192.168.20.4	UDP	288	443 + 54332 Len=246
31	1.273584	110.93.159.35	192.168.20.4	UDP	1294	443 + 54332 Len=1252
32	1.273584	110.93.159.35	192.168.20.4	UDP	288	443 + 54332 Len=241
33	1.273733	110.93.159.35	192.168.20.4	UDP	1294	443 + 54332 Len=1252
34	1.273733	110.93.159.35	192.168.20.4	UDP	1294	443 + 54332 Len=1252
35	1.273733	110.93.159.35	192.168.20.4	UDP	1294	443 + 54332 Len=1252
36	1.273733	110.93.159.35	192.168.20.4	UDP	1294	443 + 54332 Len=1252
37	1.273733	110.93.159.35	192.168.20.4	UDP	1294	443 + 54332 Len=1252
38	1.273733	110.93.159.35	192.168.20.4	UDP	1050	443 + 54332 Len=1008
39	1.273751	192.168.20.4	110.93.159.35	UDP	74	54332 + 443 Len=32
40	1.273792	192.168.20.4	110.93.159.35	UDP	74	54332 + 443 Len=32
41	1.273806	192.168.20.4	110.93.159.35	UDP	74	54332 + 443 Len=32
42	1.273806	192.168.20.4	110.93.159.35	UDP	74	54332 + 443 Len=32
43	1.273835	192.168.20.4	110.93.159.35	UDP	74	54332 + 443 Len=32
44	1.273849	192.168.20.4	110.93.159.35	UDP	74	54332 + 443 Len=32
45	1.275824	192.168.20.4	110.93.159.35	UDP	75	54332 + 443 Len=33

> Frame 21: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface `Device\WPF_{92708C19-2217-4AE2-8544-99BACC980322}`, id 0
> Ethernet II, Src: MicroStar INT_cba4be9 (d8:b1:c1:b4:b5:e9), Dst: Seulicom_23:30:19 (00:05:66:23:30:19)
> Internet Protocol Version 4, Src: 192.168.20.4, Dst: 168.126.63.1
> User Datagram Protocol, Src Port: 63138, Dst Port: 53
 Domain Name System (query)
 Transaction ID: 0x00da
 Flags: 0x0100 Standard query
 Questions: 1
 Answer RRs: 0
 Authority RRs: 0
 Additional RRs: 0
 < Questions >
 Question: 110.93.159.35.in-addr.arpa. type A, class IN
 [Response In: 24]
 [Response In: 24]

- DNS 서비스 이용 시 UDP 통신이 사용됨
- 초기에 클라이언트(192.168.20.4)가 DNS 서버(168.126.63.1)에게 DNS 요청을 보내고 서버로부터 수신받음
- 서버로 전송하는 4 개의 패킷에서 목적지 포트번호는 53 번이고, 출발지 포트번호는 각각 다른 번호임(63138, 62106)
- search.naver.com 에 대해 질의하였음
- UDP 헤더의 필드 크기는 출발지 포트번호와 목적지 포트번호가 각각 2 바이트, 데이터그램 크기가 2 바이트, checksum 이 2 바이트임
- UDP 데이터그램의 총 크기는 42 바이트이며 payload(헤더를 제외한 데이터)의 크기는 34 바이트임

```

> Frame 22: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface \Device\NPF_{92708C19-2217-4AE2-8544-098ACC900322}, id 0
> Ethernet II, Src: MicroStarINT_cb:4b:e9 (d8:bb:c1:cb:4b:e9), Dst: Secuicom_23:30:19 (00:05:66:23:30:19)
  Internet Protocol Version 4, Src: 192.168.20.4, Dst: 168.126.63.1
    Version: 4
    .... 0100 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      0000 00.. = Differentiated Services Codepoint: Default (0)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 62
    Identification: 0x8f3f (36671)
    .... 0000 ..00 = Flags: 0x0
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..0. .... = More fragments: Not set
      ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 128
    Protocol: UDP (17)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.20.4
    Destination Address: 168.126.63.1
    [Stream index: 4]
  User Datagram Protocol, Src Port: 62106, Dst Port: 53
    Source Port: 62106
    Destination Port: 53
    Length: 42
    Checksum: 0xbc67 [unverified]
    [Checksum Status: Unverified]

> Frame 22: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface \Device\NPF_{92708C19-2217-4AE2-8544-098ACC900322}, id 0
> Ethernet II, Src: MicroStarINT_cb:4b:e9 (d8:bb:c1:cb:4b:e9), Dst: Secuicom_23:30:19 (00:05:66:23:30:19)
  Internet Protocol Version 4, Src: 192.168.20.4, Dst: 168.126.63.1
    Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      0000 00.. = Differentiated Services Codepoint: Default (0)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 62
    Identification: 0x8f3f (36671)
    .... 0000 ..00 = Flags: 0x0
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..0. .... = More fragments: Not set
      ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 128
    Protocol: UDP (17)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.20.4
    Destination Address: 168.126.63.1
    [Stream index: 4]
  User Datagram Protocol, Src Port: 62106, Dst Port: 53
    Source Port: 62106
    Destination Port: 53
    Length: 42
    Checksum: 0xbc67 [unverified]
    [Checksum Status: Unverified]

```

```

> Frame 25: 212 bytes on wire (1696 bits), 212 bytes captured (1696 bits) on interface \Device\NPF_{92708C19-2217-4AE2-8544-098ACC90D322}, id 0
> Ethernet II, Src: Secuicom_23:30:19 (00:05:66:23:30:19), Dst: MicroStarINT_cb:4b:e9 (d8:bb:c1:cb:4b:e9)
  Internet Protocol Version 4, Src: 168.126.63.1, Dst: 192.168.20.4
    Version: 4
      .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      0000 00.. = Differentiated Services Codepoint: Default (0)
        .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 198
    Identification: 0x26a0 (9888)
    Flags: 0x0
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..0. .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 56
    Protocol: UDP (17)
    Header Checksum: 0x9f5b [validation disabled]
      [Header checksum status: Unverified]
    Source Address: 168.126.63.1
    Destination Address: 192.168.20.4
      [Stream index: 4]
  User Datagram Protocol, Src Port: 53, Dst Port: 62106
    Source Port: 53
    Destination Port: 62106
    Length: 178
    Checksum: 0xe988 [unverified]
      [Checksum Status: Unverified]

```

- 클라이언트가 서버에게 요청을 보내는 경우 TTL 값은 128(윈도우 운영체제의 기본값)임
- 서버에서 클라이언트에게 보낸 패킷의 TTL 값은 56 임
- 서버에서 클라이언트에게 보낸 패킷 중 63138 번 포트로 보낸 패킷은 don't fragment 가 설정되어 있으며 62106 번 포트로 보낸 패킷은 설정되어 있지 않음