

**컴퓨터보안 (AWS4016-02)**

---

**Week 10 – Network Security**

---

**컴퓨터공학과**  
**2020112119 강동희**  
**2023. 05. 16**

---

# 목 차

---

## 1. 서론

### 1.1 문제 정의

## 2. 본론

### 2.1 Snort 윈도우 시스템 구축하기

### 2.2 WiFi 탐색툴 Vistumbler 사용하기

## 3. 결론

### 3.1 느낀 점

---

# 서론

---

## 1.1 문제 정의

### 1. 스노트(Snort) 윈도우 시스템 구축하기

#### [실습 목표]

해당 실습을 통해 Snort 시스템을 윈도우 기반 PC에 구축하는 것이 목표이다. Snort는 네트워크 침입 차단 시스템이자, 네트워크 침입 탐지 시스템으로 오픈 소스이다.

[네트워크 침입 차단 시스템(NIPS; Network Intrusion Prevention System)과 네트워크 침입 탐지 시스템(NIDS; Network Intrusion Detection System)]<sup>1</sup>

NIPS는 네트워크 침입을 차단하는 것을 목표로 하며, 악성 행위를 식별하고 해당 행위를 차단하거나 경고를 발생시켜 네트워크에 대응하는 시스템이다.

NIDS는 악성 행위를 탐지하고 식별하는 데에 중점을 둔다. 이를 위해 네트워크 트래픽을 모니터링하고 이상 징후를 감지하여 보안 관리자에게 경고를 제공하는 시스템이다.

NIPS는 일반적으로 네트워크의 진입점에 위치하여 해당 네트워크로 들어오는 트래픽을 실시간으로 모니터링하고 필요한 조치를 취할 수 있도록 한다.

반면에 NIDS는 네트워크 내의 여러 위치에 분산되어 배치될 수 있어 다양한 지점에서 트래픽을 모니터링하여 네트워크 전체의 보안을 강화한다는 차이점이 있다.

대응 방식에 따른 차이점을 정리하자면, NIPS는 침입을 식별한 후 즉각적으로 대응 조치를 취할 수 있어 악성 행위를 차단하거나 차단된 행위에 대한 경고를 발생시켜 네트워크를 보호하지만 NIDS는 주로 악성 행위를 식별하고 경고를 제공하지만 차단 기능은 갖추고 있지 않는다.

#### [스노트(Snort)]

네트워크 트래픽을 실시간으로 모니터링하고 악성행위를 탐지하는 기능을 제공한다. 주로 침입 탐지 시스템(Intrusion Detection System, IDS)으로 사용되며, 네트워크 보안 및 모니터링에 활용한다. 패킷 스니핑(Packet Sniffing) 기술을 사용하여 네트워크 상에서 전송되는 패킷을 캡처하고 분석한다. 다양한 탐지 방식을 활용하여 악성 행위를 식별하는 데 사용한다.

---

<sup>1</sup> Cisco Systems. (2020). Intrusion Prevention Systems (IPS) vs. Intrusion Detection Systems (IDS): What's the Difference?

## 2. 무선 네트워크 분석을 위한 WiFi 탐색툴 Vistumbler 사용하기

### [실습 목표]

Vistumbler 를 이용하여 활성화된 AP 를 스캔하며 사용법과 해당 데이터 및 기능을 알아본다.

### [Vistumbler]

Windows 운영체제에서 사용할 수 있는 무선 네트워크 스캐너 프로그램이다. 주변의 무선 액세스 포인트(AP)를 탐지하고, 신호 강도, 채널, 암호화 설정 등과 같은 관련 정보를 제공한다.

주요 기능은 다음과 같다.

- (1) 무선 AP 탐지: 주변에 있는 무선 액세스 포인트를 검색하여 시각적으로 표시해준다. 이를 통해 사용자가 주변에 어떤 무선 네트워크가 있는지 쉽게 확인할 수 있다.
- (2) 신호 강도 및 품질 정보: 각 AP 의 신호 강도와 품질정보를 제공한다.
- (3) 채널 및 암호화 설정: 각 AP 의 사용하는 채널 및 암호화 설정 정보를 제공한다.
- (4) GPS 지원 : GPS 기기와 연동하여 AP 의 위치를 지도 위에 표시할 수 있다.

### [무선랜(Wireless-LAN)]

WLAN 은 컴퓨터나 기타 장치들을 로컬 네트워크로 연결하는 기술이다. 유선 네트워크의 제약을 없애고 이동성을 제공하며, 무선 통신을 통해 데이터를 주고받을 수 있게 된다.

WLAN 은 일반적으로 무선 액세스 포인트(AP) 장치를 통해 작동한다. AP 는 무선 신호를 발생시켜 클라이언트 장치들이 네트워크에 접속할 수 있도록 한다.

또한, 다양한 무선 통신 기술을 사용하며, 널리 사용되는 표준은 IEEE 802.11 series 이며, 세부적으로 다른 주파수 대역, 전송 속도, 범위 등을 제공한다.

### [워드라이빙(War-Driving)]

WLAN 의 보안 취약점을 탐지하기 위해 자동차나 이동식 장비를 이용하여 도로를 주행하면서 무선 네트워크를 탐색하는 행위를 말한다. 일반적으로 무선 네트워크의 신호를 탐색하고 해당 네트워크의 위치, 신호 강도, 암호화 설정 등을 수집하게 된다. 이 정보를 통해 공격자나 보안 전문가가 무선 네트워크의 보안 취약점을 파악하고 악의적인 공격을 시도할 수 있게 된다.

---

## 본 론

---

### 2.1 스노트(Snort) 윈도우 시스템 구축하기

[1] winpcap, Npcap, Snort 설치하기

아래 그림 1, 그림 2, 그림 3 과 같이 Snort 사용을 위한 관련 프로그램을 설치한다.

**WinPcap** 은 Windows 운영체제에서 사용되는 패킷 캡처 라이브러리로 네트워크 트래픽을 모니터링하고 패킷을 캡처하는 기능을 제공하여 네트워크 분석, 보안 검사, 네트워크 모니터링 등에 활용된다.

**Npcap** 은 Windows 운영체제에서 사용되는 패킷 캡처 라이브러리 및 드라이버로 WinPcap 의 개선된 버전이기도 하다.

설치한 버전 같은 경우 WinPcap 은 4.1.3, Npcap 은 1.10, Snort 는 2.9.20 버전을 설치하였다.



그림 1. WinPcap 4.1.3 설치

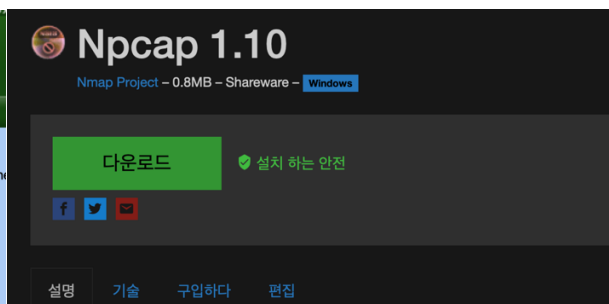


그림 2. Npcap 1.10 설치

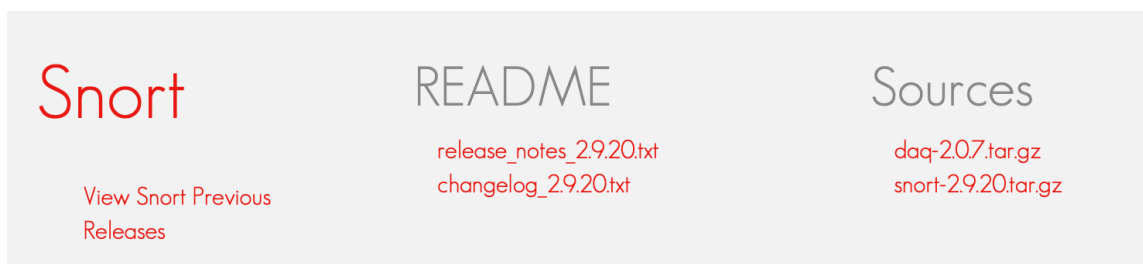
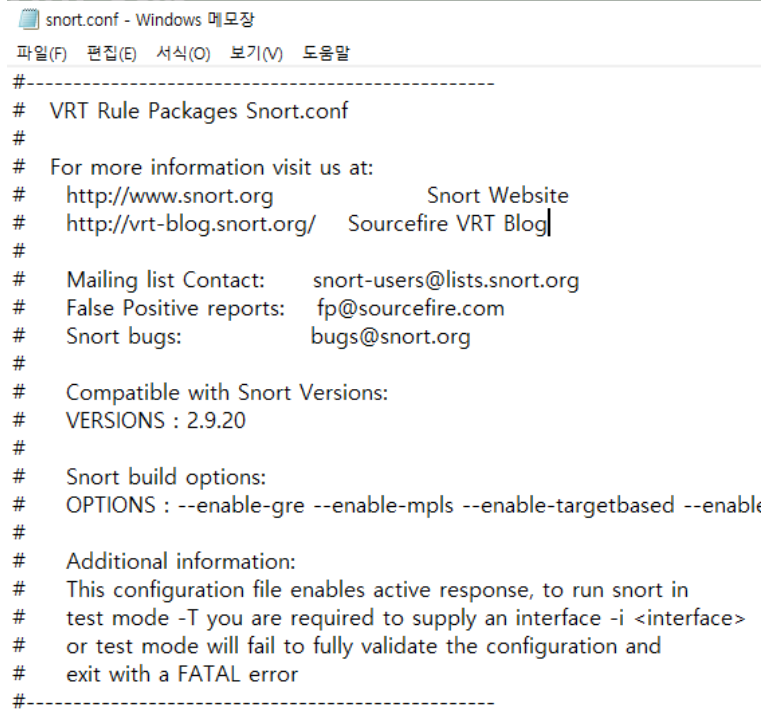


그림 2. Snort 2.9.20 설치

## [2] 설치한 Snort 의 경로로 이동하여 snort.conf 오픈

snort.conf 파일은 Snort Intrusion Detection System 의 주요 구성 파일이다. 해당 파일을 통해 네트워크 설정, 감지 규칙 설정, 로깅 및 출력 설정, 전처리기 설정, 기타 설정을 직접 사용자가 조정할 수 있다.

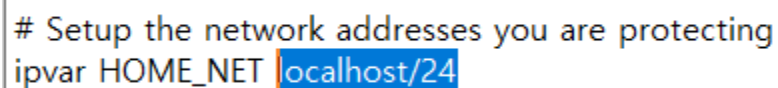


```
#-----  
# VRT Rule Packages Snort.conf  
#  
# For more information visit us at:  
# http://www.snort.org           Snort Website  
# http://vrt-blog.snort.org/     Sourcefire VRT Blog  
#  
# Mailing list Contact:   snort-users@lists.snort.org  
# False Positive reports: fp@sourcefire.com  
# Snort bugs:            bugs@snort.org  
#  
# Compatible with Snort Versions:  
# VERSIONS : 2.9.20  
#  
# Snort build options:  
# OPTIONS : --enable-gre --enable-mpls --enable-targetbased --enable-...  
#  
# Additional information:  
# This configuration file enables active response, to run snort in  
# test mode -T you are required to supply an interface -i <interface>  
# or test mode will fail to fully validate the configuration and  
# exit with a FATAL error  
#-----
```

그림 3. snort.conf 화면

## [3] ipvar HOME\_NET any 를 local host로 변경

Ipvar HOME\_NET은 네트워크 환경의 내부 네트워크 대역을 지정하는 데 사용되는 snort 변수이다. 네트워크 트래픽을 분석하고 감지할 때 snort에게 내부 네트워크 대역을 알려 주게 된다. 실습에서는 localhost/24로 변경하였다.



```
# Setup the network addresses you are protecting  
ipvar HOME_NET localhost/24
```

그림 4. HOME\_NET 로컬 호스트 설정

#### [4] include classification.conf 변경

classification.conf은 Snort 감지 규칙에 대한 분류 체계에 대해 이벤트를 구조화하고 분류할 수 있게 한다. 효과적인 이벤트 관리와 분석을 위해 사용한다.

```
# metadata reference data. do not modify these lines
include c:\snort\etc\classification.conf
include reference.conf
```

그림 5. classification.conf 설정

#### [5] include reference.conf 변경

snort.conf 파일 내에서 다른 설정 파일인 reference.conf 파일을 포함시키는 지시문으로 감지 이벤트에 대한 참고 자료, 보안 공격에 대한 설명, 취약성 정보 등을 포함시켜 사용자가 감지된 이벤트에 대한 추가적인 정보를 확인하고 분석하는 데 도움을 준다.

```
# metadata reference data. do not modify these lines
include c:\snort\etc\classification.conf
include c:\snort\etc\reference.conf
```

그림 6. reference.conf 설정

#### [6] snort -W 를 통해 어댑터 목록 확인

해당 명령어는 사용가능한 네트워크 인터페이스와 관련된 정보를 출력하는 명령어로 수집된 장치의 Index, 물리 주소, IP 주소, 장치 이름과 세부 정보를 얻을 수 있다.

실습에서는 Realtek Ethernet Controller 를 선택하여 진행하므로 앞으로 NIC 카드 번호는 1 로 진행한다.

```
c:\snort\bin>snort -W

->> Snort! <*-
o'')~ Version 2.9.20-WIN64 GRE (Build 82)
.... By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
      Copyright (C) 2014-2022 Cisco and/or its affiliates. All rights reserved.
      Copyright (C) 1998-2018 Sourcefire, Inc., et al.
      Using PCRE version: 8.10 2010-06-25
      Using ZLIB version: 1.2.11

Index  Physical Address      IP Address      Device Name      Description
-----
1      00:00:00:00:00:00      0000:0000:fe80:0000:0000:0000:bd93:b14b #Device#NPF_{19ED8090-1200-4A3D-818C-84E0243312}
EB) Realtek Ethernet Controller
2      00:00:00:00:00:00      disabled      #Device#NPF_{BA4133BA-A83F-46C8-B895-C8E1A177ECB1}
      NdisWan Adapter
3      00:00:00:00:00:00      disabled      #Device#NPF_{FE305F27-179C-4C18-AC1B-AC4081276AF5}
      NdisWan Adapter
4      00:00:00:00:00:00      disabled      #Device#NPF_{55DC7BB0-B037-46C1-95DA-D44C1FC71670}
      NdisWan Adapter
5      00:00:00:00:00:00      disabled      #Device#NPF_{Loopback} Adapter for loopback traffic capture
```

그림 7. snort -W : 어댑터 목록 확인

[7] snort -v -i[NIC 카드 번호] / snort -v i1

Snort 를 실행하는 명령어로 -v 옵션을 통해 상세한(verbose) 출력을 표시하게 된다.  
수행하는 동작과 감지된 이벤트에 대한 자세한 정보를 출력해준다.

또한 -i 옵션으로 해당 네트워크 인터페이스에서 트래픽을 수신하도록 설정해주었다.

```
WARNING: No preprocessors configured for policy 0.  
05/10-10:25:16.402526 210.94.222.128:59503 -> 224.0.0.252:5355  
UDP TTL:1 TOS:0x0 ID:1186 IpLen:20 DgmLen:52  
Len: 24  
++++++  
WARNING: No preprocessors configured for policy 0.  
05/10-10:25:16.402526 fe80::0000:0000:0000:d444:5d79:b7d7:3cc2:5353 -> ff02::00  
UDP TTL:1 TOS:0x0 ID:0 IpLen:40 DgmLen:78  
Len: 30  
++++++  
WARNING: No preprocessors configured for policy 0.  
05/10-10:25:16.402526 210.94.222.128:5353 -> 224.0.0.251:5353  
UDP TTL:1 TOS:0x0 ID:42260 IpLen:20 DgmLen:58  
Len: 30  
++++++
```

그림 8. snort -v i1 실행 화면

[8] ipconfig 이더넷의 기본 게이트웨이 아이피 주소 확인

실습을 진행한 컴퓨터의 게이트웨이 주소를 확인하였다.(210.94.222.2)

```
C:\Users\W3183-27>ipconfig  
  
Windows IP 구성  
  
이더넷 어댑터 이더넷:  
  
연결별 DNS 접미사. . . . . :  
링크-로컬 IPv6 주소 . . . . : fe80::bd33:b14b:e6a1:706d%3  
IPv4 주소 . . . . . : 210.94.222.37  
서브넷 마스크 . . . . . : 255.255.255.0  
기본 게이트웨이 . . . . . : 210.94.222.2
```

그림 9. ipconfig 기본 게이트웨이 확인

[9] 기본 게이트웨이로 ping 전송

실습 [8]에서 진행한 IP 주소로 ping 을 전송한다.

```
C:\Users\W3183-27>ping 210.94.222.2  
  
Ping 210.94.222.2 32바이트 데이터 사용:  
210.94.222.2의 응답: 바이트=32 시간<1ms TTL=254  
210.94.222.2의 응답: 바이트=32 시간<1ms TTL=254  
210.94.222.2의 응답: 바이트=32 시간<1ms TTL=254  
210.94.222.2의 응답: 바이트=32 시간<1ms TTL=254  
  
210.94.222.2에 대한 Ping 통계:  
패킷: 보낸 = 4, 받음 = 4, 손실 = 0 (0% 손실),  
왕복 시간(밀리초):  
최소 = 0ms, 최대 = 0ms, 평균 = 0ms
```

그림 10. ping 210.94.222.2 ping 전송



## [10] 네이버 ip 검색 및 ping 전송

컴퓨터 IP 주소 뿐만 아니라 네이버 서버의 IP 주소를 알아내 해당 주소로도 같이 ping 을 전송하여 트래픽을 snort 에서 캡처하도록 한다.

```
C:\Users\#3183-27>nslookup naver.com
서버: ns.dgu.ac.kr
Address: 210.94.190.7

권한 없는 응답:
이름: naver.com
Addresses: 223.130.200.107
          223.130.195.95
          223.130.195.200
          223.130.200.104

C:\Users\#3183-27>ping 223.130.195.200

Ping 223.130.195.200 32바이트 데이터 사용:
요청 시간이 만료되었습니다.
요청 시간이 만료되었습니다.
요청 시간이 만료되었습니다.
요청 시간이 만료되었습니다.

223.130.195.200에 대한 Ping 통계:
패킷: 보냄 = 4, 받음 = 0, 손실 = 4 (100% 손실),
```

그림 11. Naver.com 서버 및 ping 전송

## [11] snort 종료 및 결과 확인

위 실습 컴퓨터와 네이버 서버로 PING 을 전송하면서 패킷 ICMP 가 12 번 탐지된 것을 확인할 수 있다. 실습 컴퓨터로 총 4 번의 ping 을 전송하였고 수신도 모두 받았기 때문에 8 번, 그리고 네이버 서버로 4 번의 ping 을 전송하였고, 수신을 모두 실패하였기 때문에 8 + 4 로 12 번 탐지된 것을 확인할 수 있다.

```
Run time for packet processing was 200.102000 sec
Snort processed 41434 packets.
Snort ran for 0 days 0 hours 3 minutes 20 seconds
Pkts/min: 13811
Pkts/sec: 207

=====
Packet I/O Totals:
Received: 41581
Analyzed: 41434 ( 99.646%)
Dropped: 0 ( 0.000%)
Filtered: 0 ( 0.000%)
Outstanding: 147 ( 0.354%)
Injected: 0

=====
Breakdown by protocol (includes rebuilt packets):
Eth: 41434 (100.000%)
VLAN: 0 ( 0.000%)
IP4: 23729 ( 57.269%)
Frag: 0 ( 0.000%)
ICMP: 12 ( 0.029%)
UDP: 21082 ( 50.881%)
TCP: 2629 ( 6.345%)
IP6: 13108 ( 31.636%)
IP6 Ext: 13131 ( 31.691%)
IP6 Opts: 23 ( 0.056%)
Frag6: 0 ( 0.000%)
ICMP6: 30 ( 0.072%)
UDP6: 13078 ( 31.563%)
```

그림 12. snort 종료 및 결과 확인

[11] snort 로그 파일 생성 `snort -ix -dev -l WsnortWlog`

스노트를 실행하는 명령어로 위 [10]의 결과로는 짧은 시간내에 많은 패킷을 수집하면서도 내가 필요한 정보의 패킷을 확인하기 어려워 로그 기록을 통해 분석하고자 로그 파일을 생성한다. 따라서 WsnortWlog 경로에 수집 결과를 저장하도록 실행하였다.

```
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.  
WARNING: No preprocessors configured for policy 0.
```

그림 13. Snort -i1 -dev -l WsnortWlog

[12] 게이트웨이와 네이버 서버 핑 전송

이전 실습과 마찬가지로 게이트웨이와 네이버 서버로 핑을 전송한다.

```
C:\Users\W3183-27>ping 210.94.222.2  
  
Ping 210.94.222.2 32바이트 데이터 사용:  
210.94.222.2의 응답: 바이트=32 시간=1ms TTL=254  
210.94.222.2의 응답: 바이트=32 시간<1ms TTL=254  
210.94.222.2의 응답: 바이트=32 시간<1ms TTL=254  
210.94.222.2의 응답: 바이트=32 시간<1ms TTL=254  
  
210.94.222.2에 대한 Ping 통계:  
패킷: 보냄 = 4, 받음 = 4, 손실 = 0 (0% 손실),  
왕복 시간(밀리초):  
최소 = 0ms, 최대 = 1ms, 평균 = 0ms  
  
C:\Users\W3183-27>ping 223.130.195.200  
  
Ping 223.130.195.200 32바이트 데이터 사용:  
요청 시간이 만료되었습니다.  
요청 시간이 만료되었습니다.  
요청 시간이 만료되었습니다.  
요청 시간이 만료되었습니다.  
  
223.130.195.200에 대한 Ping 통계:  
패킷: 보냄 = 4, 받음 = 0, 손실 = 4 (100% 손실),
```

그림 14. 기본 게이트웨이 및 네이버 서버 ping 전송

### [13] snort 종료 및 결과 확인

앞선 실습과 같은 사유로 ICMP 패킷은 총 12회 탐지된 것을 확인할 수 있다. 다만 #snort#log 경로에 수집한 패킷의 기록이 저장된 것이 차이점이다.

```
Run time for packet processing was 66.692000 seconds
Snort processed 12976 packets.
Snort ran for 0 days 0 hours 1 minutes 6 seconds
  Pkts/min:    12976
  Pkts/sec:     196
=====
Packet I/O Totals:
  Received:    13171
  Analyzed:    12976 ( 98.519%)
  Dropped:      0 ( 0.000%)
  Filtered:     0 ( 0.000%)
  Outstanding: 195 ( 1.481%)
  Injected:     0
=====
Breakdown by protocol (includes rebuilt packets):
  Eth:         12976 (100.000%)
  VLAN:        0 ( 0.000%)
  IP4:         7248 ( 55.857%)
  Frag:        0 ( 0.000%)
  ICMP:        12 ( 0.092%)
  UDP:        6966 ( 53.684%)
  TCP:         270 ( 2.081%)
  IP6:         4298 ( 33.123%)
  IP6 Ext:     4298 ( 33.123%)
  IP6 Opts:    0 ( 0.000%)
  Frag6:      0 ( 0.000%)
  IPv6:      0 ( 0.000%)
=====
```

그림 15. snort 종료 및 결과 확인

### [14] log 확인 (메모장, wireshark)

로그 파일이 저장된 경로로 이동하여 파일을 메모장으로 열었을 때 그림 16과 같이 알아보기 힘들었다.

```
HOST: 239.255.255.250:1900
MAN: "ssdp:discover"
MX: 1
ST: urn:dial-multiscreen-org:service:dial:1
USER-AGENT: Microsoft Edge/113.0.1774.35 Windows

04ZdH? H H r ^ ?젠 ISQ E :8 r 4G ??? &등 r -iot_db|local r r04Zd]? W W 3
r -iot_db|local r r04ZdF? H H r ^ ?젠 ISQ E :8 r 4G|??? ??? &등 r -iot_db|loca
셋 4r? ?y관<? r 4등? ?-? r -iot_db - r04ZdF? B B r ^ ?젠 ISQ E 4O:
??? 9弄 ? r 0SEC842519229D41|local r r -?04Zd>? R R $精總? E D'
HOST: 239.255.255.250:1900
MAN: "ssdp:discover"
MX: 1
ST: urn:dial-multiscreen-org:service:dial:1
USER-AGENT: Microsoft Edge/113.0.1774.35 Windows

04Zd면 < < ?AG wQ- rQ -r?AG w?? ?? 04ZdQ? [ [ r ^ ?
?y관<? ??? &=? r -iot_db|local r r04Zd?Q H H r ^ ?젠 ISQ E :8 r 4F
4r? ?y관<? r 4? ? :lp? r -iot_db - r04Zd? B B r ^ ?젠 ISQ E 4C
HOST: 239.255.255.250:1900
MAN: "ssdp:discover"
MX: 1
ST: urn:dial-multiscreen-org:service:dial:1
USER-AGENT: Microsoft Edge/102.0.1245.44 Windows

04Zd? 6 6 ?AG w$精總 E (관@ □-관??!!면?관?L&|肝P+L?; 04Zd? 6 6 ?AG w$
< < r□? ?AG - 'BB' r< ??? 4'관?AG □-r 9 r 4 04Zd器
< < $精總?AG wQ E ( @ 6-[?!!면^?r? ???!!?+r?) 04Zd:
< < ?-rQ -r r? ?-? 04ZdE? < < ?
< < ?AG wQ- rQ -r?AG w?? ?? 04ZdH? H H r ^ ?젠 ISQ
W 33 ?젠 ISQ E? &4r? ?y관<? ??? &=? r -iot_db|local r r04ZdH? H
2 4r? ?y관<? r 4? ? r -iot_db - r04Zd0v? W W 33 ?젠 ISQ E? &4r
r? ?y관<? r 4? ? 精總 r -iot_db - r04Zd? B B r ^ ?젠 ISQ E 4O? r 4?
4'관? ISQ E?
? 4r? ?y관<? r 4? ? 精總 r -iot_db - r04Zd? B B r ^ ?젠 ISQ E 4O? r 4?
```

그림 16. snort log 파일 열기(메모장)

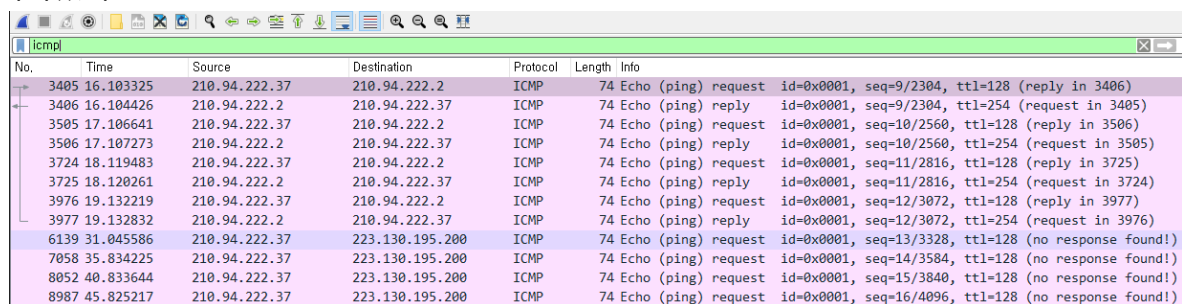
Wireshark 프로그램으로 로그 파일을 열어서 확인하니 시각적으로 한 눈에 알아보기 쉽게 분석할 수 있다. 필터링을 통해 ICMP를 검색하여 해당 패킷만 볼 수 있었다.

먼저, 컴퓨터의 기본 게이트웨이로 보낸 ICMP를 확인할 수 있다.

210.94.222.37은 ping을 전송한 컴퓨터의 IP주소이며 해당 ping이 도착한 게이트웨이 주소인 210.94.222.2로 도착한 것을 알 수 있다. 1, 3, 5, 7번 로그가 이에 해당하며, 관련 정보로 Echo (ping) request로 ping 전송을 통해 ICMP 요청한 정보인 것을 알 수 있다. 게이트웨이에서 응답한 패킷으로 2,4,6,8번 로그에 해당된 것으로 알 수 있다.

다음으로, 네이버 서버로 ping을 보낸 패킷 정보를 확인할 수 있다.

Ping을 보낸 ip주소인 210.94.222.37을 볼 수 있고, 네이버의 서버 주소로 223.130.195.200으로 request 요청을 보낸 것을 확인할 수 있다. 이때 info에 (no response found!)라는 메시지를 확인할 수 있었다. 해당 에러 메시지에 대해 아래에서 정리하였다.

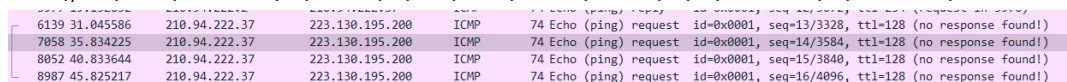


No.	Time	Source	Destination	Protocol	Length	Info
3405	16.103325	210.94.222.37	210.94.222.2	ICMP	74	Echo (ping) request id=0x0001, seq=9/2304, ttl=128 (reply in 3406)
3406	16.104426	210.94.222.2	210.94.222.37	ICMP	74	Echo (ping) reply id=0x0001, seq=9/2304, ttl=254 (request in 3405)
3505	17.106641	210.94.222.37	210.94.222.2	ICMP	74	Echo (ping) request id=0x0001, seq=10/2560, ttl=128 (reply in 3506)
3506	17.107273	210.94.222.2	210.94.222.37	ICMP	74	Echo (ping) reply id=0x0001, seq=10/2560, ttl=254 (request in 3505)
3724	18.119483	210.94.222.37	210.94.222.2	ICMP	74	Echo (ping) request id=0x0001, seq=11/2816, ttl=128 (reply in 3725)
3725	18.120261	210.94.222.2	210.94.222.37	ICMP	74	Echo (ping) reply id=0x0001, seq=11/2816, ttl=254 (request in 3724)
3976	19.132219	210.94.222.37	210.94.222.2	ICMP	74	Echo (ping) request id=0x0001, seq=12/3072, ttl=128 (reply in 3977)
3977	19.132832	210.94.222.2	210.94.222.37	ICMP	74	Echo (ping) reply id=0x0001, seq=12/3072, ttl=254 (request in 3976)
6139	31.045586	210.94.222.37	223.130.195.200	ICMP	74	Echo (ping) request id=0x0001, seq=13/3328, ttl=128 (no response found!)
7058	35.834225	210.94.222.37	223.130.195.200	ICMP	74	Echo (ping) request id=0x0001, seq=14/3584, ttl=128 (no response found!)
8052	40.833644	210.94.222.37	223.130.195.200	ICMP	74	Echo (ping) request id=0x0001, seq=15/3840, ttl=128 (no response found!)
8987	45.825217	210.94.222.37	223.130.195.200	ICMP	74	Echo (ping) request id=0x0001, seq=16/4096, ttl=128 (no response found!)

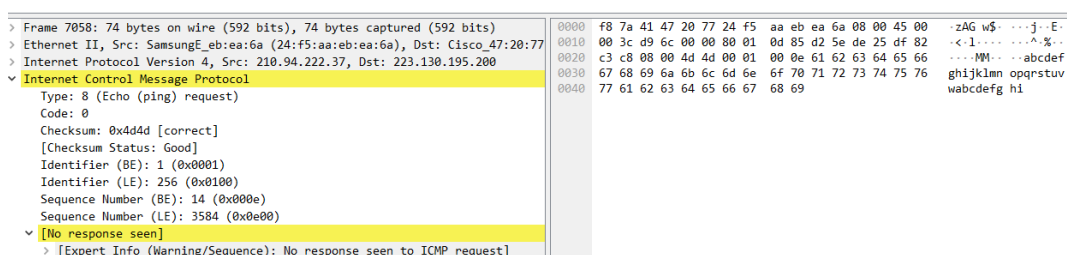
그림 17. snort log 파일 열기(Wireshark)

## [15] 비교 분석

네이버 서버로 보낸 ping 패킷의 응답이 없는 이유에 대해서 wireshark를 통해 확인할 수 있다. 해당 로그를 클릭하여 ICMP 정보를 확인해보면 Type 8, Code 0의 에러를 볼 수 있다. 이는 Echo Request 메시지가 수신자에게 전송되었지만, 수신자로부터(네이버 서버) Echo Reply 메시지가 제대로 도착하지 않은 상태를 나타낸다. 이에 대한 원인은 다양하게 예측할 수 있는데 대표적으로 네트워크 연결 문제 중 네트워크 장애, 방화벽 설정, 라우팅 문제, 네트워크 장치의 문제가 대표적인 해당 에러의 원인이라고 한다.



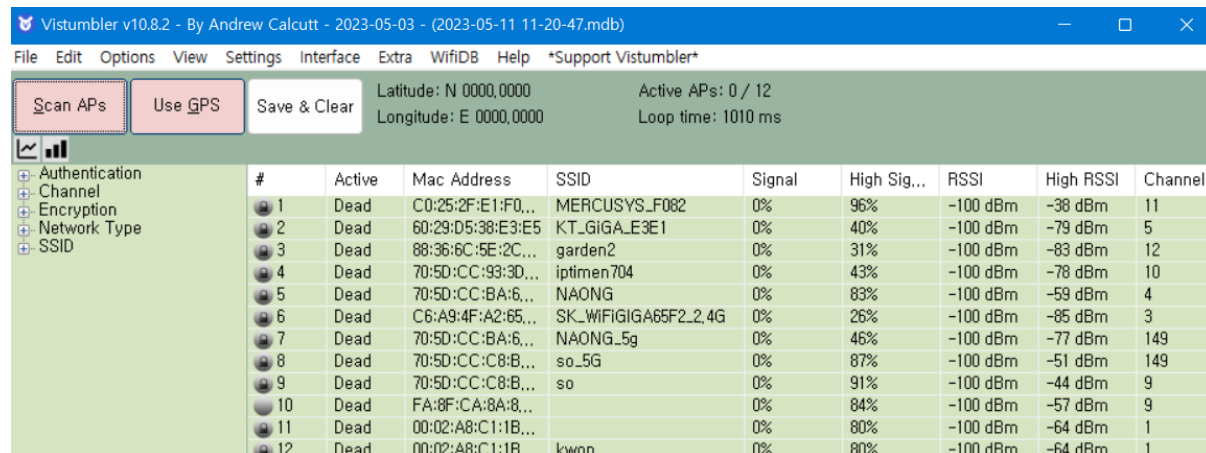
6139	31.045586	210.94.222.37	223.130.195.200	ICMP	74	Echo (ping) request id=0x0001, seq=13/3328, ttl=128 (no response found!)
7058	35.834225	210.94.222.37	223.130.195.200	ICMP	74	Echo (ping) request id=0x0001, seq=14/3584, ttl=128 (no response found!)
8052	40.833644	210.94.222.37	223.130.195.200	ICMP	74	Echo (ping) request id=0x0001, seq=15/3840, ttl=128 (no response found!)
8987	45.825217	210.94.222.37	223.130.195.200	ICMP	74	Echo (ping) request id=0x0001, seq=16/4096, ttl=128 (no response found!)



> Frame 7058: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) > Ethernet II, Src: SamsungE_eb:ea:6a (24:f5:aa:eb:ea:6a), Dst: Cisco_47:20:77 > Internet Protocol Version 4, Src: 210.94.222.37, Dst: 223.130.195.200 > Internet Control Message Protocol Type: 8 (Echo (ping) request) Code: 0 Checksum: 0x4d4d [correct] [Checksum Status: Good] Identifier (BE): 1 (0x0001) Identifier (LE): 256 (0x0100) Sequence Number (BE): 14 (0x000e) Sequence Number (LE): 3584 (0x0e00) [No response seen] > [Expert Info (Warning/Sequence): No response seen to ICMP request]		0000 f8 7a 41 47 20 77 24 f5 aa eb ea 6a 08 00 45 00 ..zAG w\$ . . . j . E 0010 00 3c d9 6c 00 00 00 01 0d 85 d2 5e de 25 df 82 ..< . 1 . . . ^ % . 0020 c3 c8 00 00 4d 4d 00 01 00 0e 61 62 63 64 65 66 ....NM... abcdef 0030 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 ghijklmn opqrstuv 0040 77 61 62 63 64 65 66 67 68 69 wabcedfg hi
--	--	---

## 2.2 무선 네트워크 분석을 위한 WiFi 탐색툴 Vistumbler 사용하기

[1] Vistumbler 설치하여 실행한다.



The screenshot shows the Vistumbler v10.8.2 application window. The title bar reads 'Vistumbler v10.8.2 - By Andrew Calcutt - 2023-05-03 - (2023-05-11 11:20:47.mdb)'. The menu bar includes File, Edit, Options, View, Settings, Interface, Extra, WifiDB, and Help. Below the menu bar are three buttons: 'Scan APs', 'Use GPS', and 'Save & Clear'. To the right of these buttons, it displays 'Latitude: N 0000,0000', 'Longitude: E 0000,0000', 'Active APs: 0 / 12', and 'Loop time: 1010 ms'. The main area contains a table of detected WiFi networks. On the left, there is a sidebar with expandable categories: Authentication, Channel, Encryption, Network Type, and SSID. The table has columns for #, Active, Mac Address, SSID, Signal, High Sig..., RSSI, High RSSI, and Channel.

#	Active	Mac Address	SSID	Signal	High Sig...	RSSI	High RSSI	Channel
1	Dead	C0:25:2F:E1:F0...	MERCUSYS_F082	0%	96%	-100 dBm	-38 dBm	11
2	Dead	60:29:D5:38:E3:E5	KT_GiGA_E3E1	0%	40%	-100 dBm	-79 dBm	5
3	Dead	88:36:6C:5E:2C...	garden2	0%	31%	-100 dBm	-83 dBm	12
4	Dead	70:5D:CC:93:3D...	iptimen704	0%	43%	-100 dBm	-78 dBm	10
5	Dead	70:5D:CC:BA:6...	NAONG	0%	83%	-100 dBm	-59 dBm	4
6	Dead	C6:A9:4F:A2:65...	SK_WiFiGiGA65F2_2.4G	0%	26%	-100 dBm	-85 dBm	3
7	Dead	70:5D:CC:BA:6...	NAONG_5g	0%	46%	-100 dBm	-77 dBm	149
8	Dead	70:5D:CC:C8:B...	so_5G	0%	87%	-100 dBm	-51 dBm	149
9	Dead	70:5D:CC:C8:B...	so	0%	91%	-100 dBm	-44 dBm	9
10	Dead	FA:8F:CA:8A:8...		0%	84%	-100 dBm	-57 dBm	9
11	Dead	00:02:A8:C1:1B...		0%	80%	-100 dBm	-64 dBm	1
12	Dead	00:02:A8:C1:1B...	kwon	0%	80%	-100 dBm	-64 dBm	1

[2] 설치한 WiFi 탐색툴을 실행하여 현재 내 주변에 탐지되는 무선랜 SSID 는 몇가지나 있으며 접속하여 사용하기에 가장 신호상태가 양호한 무선랜은 어떤 것인가?

총 12 개의 무선랜이 탐지되며, High Signal 이 96%로 잡히는 1 번 MERCUSYS\_F082 가 신호상태가 양호한 것으로 확인된다. 해당 와이파이가 실습 환경에서 가장 가까운 공유기이며 정확하게 탐지된 것으로 판단할 수 있다.

[3] 주로 어떤 암호화 기법이 사용되는가? 또 그 기법은 어떠한 기법인가?

WPA2, WPA 는 Temporal Key Integrity Protocol(TKIP)라는 암호화 프로토콜을 사용하여 보안을 제공하는데 WPA2 는 WPA 의 개선된 버전으로 AES(Advanced Encryption Standard) 암호화를 사용하며, 현재까지 가장 널리 사용되는 암호화 방식이라고 한다.

AES 는 대칭키 암호화 알고리즘으로 128 비트 키 길이를 가장 많이 사용한다. 입력 데이터를 연속적인 block 으로 나누고, 각 block 에 대해 복잡한 연산을 수행하여 암호화 또는 복호화하는 방식이다. 암호화 과정에서 입력 데이터와 키를 혼합하는 '라운드 기법'을 사용하여 안정성을 제공하며, 라운드 기법은 비트 연산, 치환, 치환-치환 네트워크 등의 다양한 연산을 통해 데이터를 혼돈시키고 암호화 효과를 증대시킨다.

[4] 해당 Wifi 탐색툴이 제공하는 데이터와 기능은 무엇이며 각각 무엇을 의미하는가?

Vistumbler 가 제공하는 데이터로 무선 네트워크 이름(SSID), 신호 강도, 채널, 암호화 방식의 정보를 제공한다. 또한, 다양한 기능을 제공하는데 (1) 무선 신호 강도 그래프, (2) GPS 지원, (3) 네트워크 세부 정보, (4) 그래프 및 통계 분석 (5) CSV 파일 제공을 한다.

[5] 탐지된 무선랜 SSID 중 가장 취약하다고 판단되는 SSID 와 이유는?

None 으로 뜨는 SSID 가 가장 취약하다. 암호화 방식이 설정되지 않은 무선 네트워크로 해당 네트워크의 트래픽이 암호화되지 않고 노출될 수 있음을 시사한다. 공격자가 무선 네트워크에 접속하게 된다면 네트워크 트래픽을 도청하거나 개인 정보를 도용할 수 있는 취약점이 있다. 또한, 인가되지 않은 사용자가 네트워크에 접속하여 중간자 공격과 같은 악의적인 목적의 행위를 시도할 수 있게 된다.

## 3 결 론

### 3.1 느낀 점

이번 실습을 통해 네트워크의 보안 중 ICMP 프로토콜 전송을 통한 네트워크 패킷 분석과 무선 네트워크 수집 및 분석하는 다양한 프로그램, 툴을 다뤄볼 수 있었다. 특정 서버로 PING을 전송하여 그 동안 스노트를 통해 기록하여 해당 패킷이 어디서 어디로 전송되었는지, 만약 전송 실패했다면 그 원인이 무엇인지 지난 번 실습에서 사용한 Wireshark를 활용하여 분석까지 진행해볼 수 있었다. 에러 Type과 code를 통해 원인 분석과 해결 방법까지 짐작할 수 있게 되었다. 또한, Vistumbler를 통해 실행자 근처의 모든 무선 네트워크 공유기의 현황을 파악할 수 있는 프로그램도 다룰 수 있게 되었다. 해당 프로그램을 통해서 SSID의 암호화 방식부터 신호 상태 등 다양한 정보를 확인이 가능했는데, 와이파이 공유기의 보안 설정의 중요성을 알게 되었다. 또한 Vistumbler의 작동 원리에 대해서 세부적으로 궁금하여 기회가 된다면 직접 클론 코딩도 구현해보고 싶었다.