105 – BlueShark

Team Information

Team Name: kimbabasaksaksak

Team Member: Jaeheon Kim, Donghyun Kim, Soyoung Yoo, Minhee Lee

Email Address: uaaoong@gmail.com

Instructions

Description Analyze the following evidence to identify the message

Target	Hash (MD5)
evidence.zip	B4345B48C5FCE8205762A856DB98D03C

Questions

1) What is the message from evidence1? (40 points)
2) What is the message from evidence2? (40 points)
3) What is the message from evidence3? (20 points)

Teams must:

- Develop and document the step-by-step approach used to solve this problem to allow another examiner to replicate team actions and results.
- Specify all tools used in deriving the conclusion(s).

Tools used:

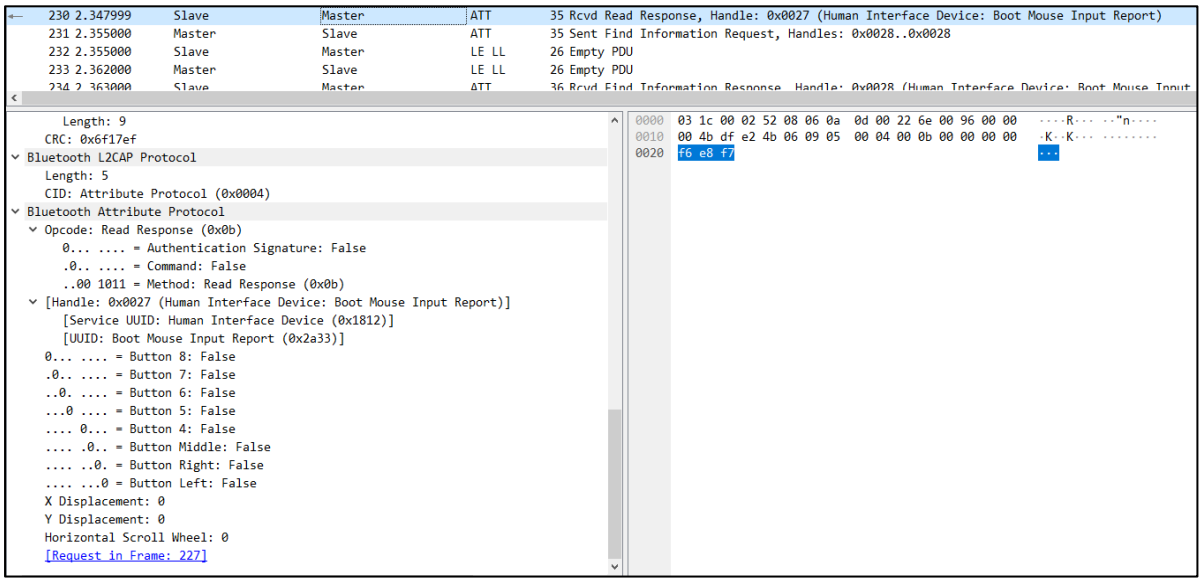
Name:	Wireshark	Publisher:	Wireshark
Version:	4.0.6		
URL:	https://www.wireshark.org/		

Step-by-step methodology:

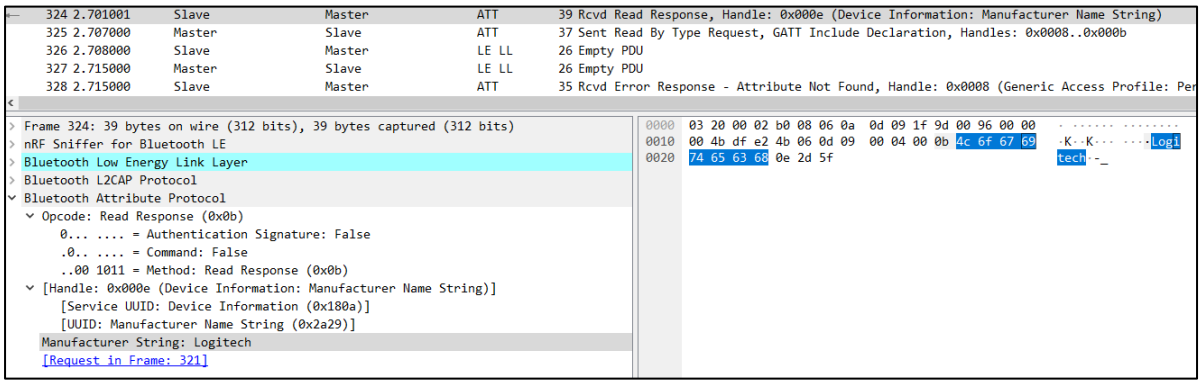
Q1. What is the message from evidence1? (40 points)

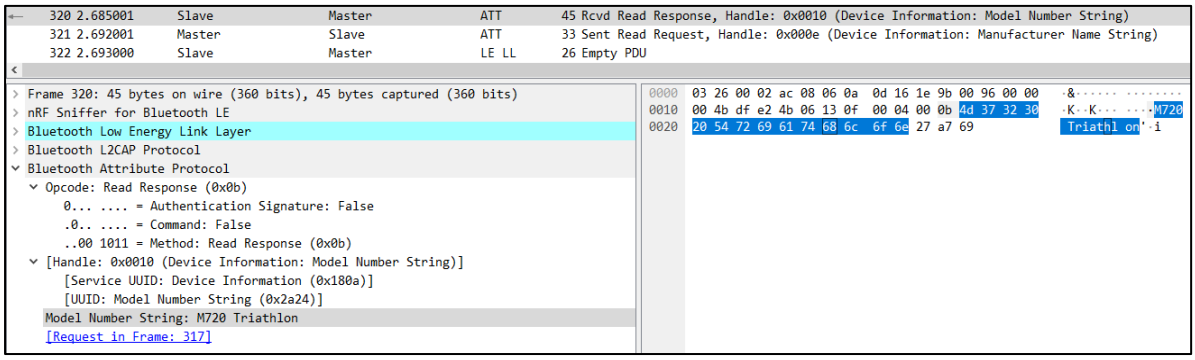
- 2023.11.23 13:30 KST

evidence1.pcap 파일은 블루투스 마우스 패킷을 캡쳐한 파일이다. 블루투스 Low Energy(BLE) 프로토콜을 통해 전송된 마우스 입력 데이터를 나타내며, 마우스의 이동 정보가 담겨있다.



ATT 프로토콜 Device Information의 Manufacturer String, Model Number String 값을 통해 Logitech M720 Triathlon 블루투스 마우스 장비에서 발생한 패킷을 캡처한 것을 알 수 있다.





파이썬으로 btatt(Bluetooth Attribute Protocol) 패킷을 필터링하고 마우스 이동 데이터를 추출하여 그래프로 표시하는 코드를 작성하였다.

Q1.py

```
import matplotlib.pyplot as plt
import pyshark

mousePositionX = 0
mousePositionY = 0

X = []
Y = []

f = pyshark.FileCapture('evidence1.pcap', display_filter="btatt")

for p in f:
    data = p['btatt'].get_field_value('value')
    if data == None:
        continue

    Bytes = data.split(":")
    if len(Bytes) == 7:
        horizontal = 2
        vertical = 3
    else:
        continue

    offsetX = int(Bytes[horizontal], 16)
    offsetY = int(Bytes[vertical], 16)
```

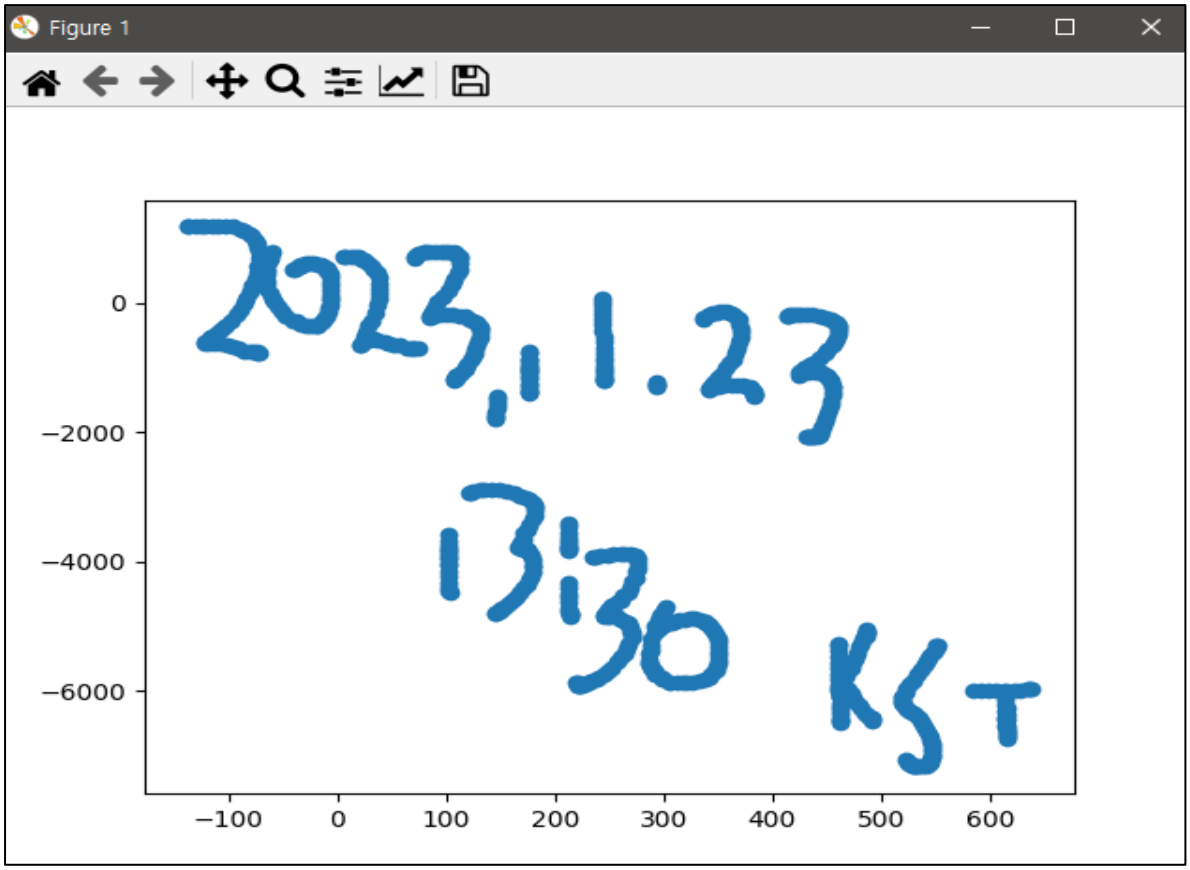
```
    if offsetX > 127:
        offsetX -= 256
    if offsetY > 127:
        offsetY -= 256

    mousePositionX += offsetX
    mousePositionY += offsetY

    if Bytes[0] == "01":
        #print("[+] Left.")
        X.append(mousePositionX)
        Y.append(-mousePositionY)
    elif Bytes[0] == "02":
        #print("[+] Right.")
        X.append(mousePositionX)
        Y.append(-mousePositionY)
    else:
        print("[-] Known operate.")
        pass

fig = plt.figure()
ax1 = fig.add_subplot()
ax1.scatter(X, Y)
plt.show()
```

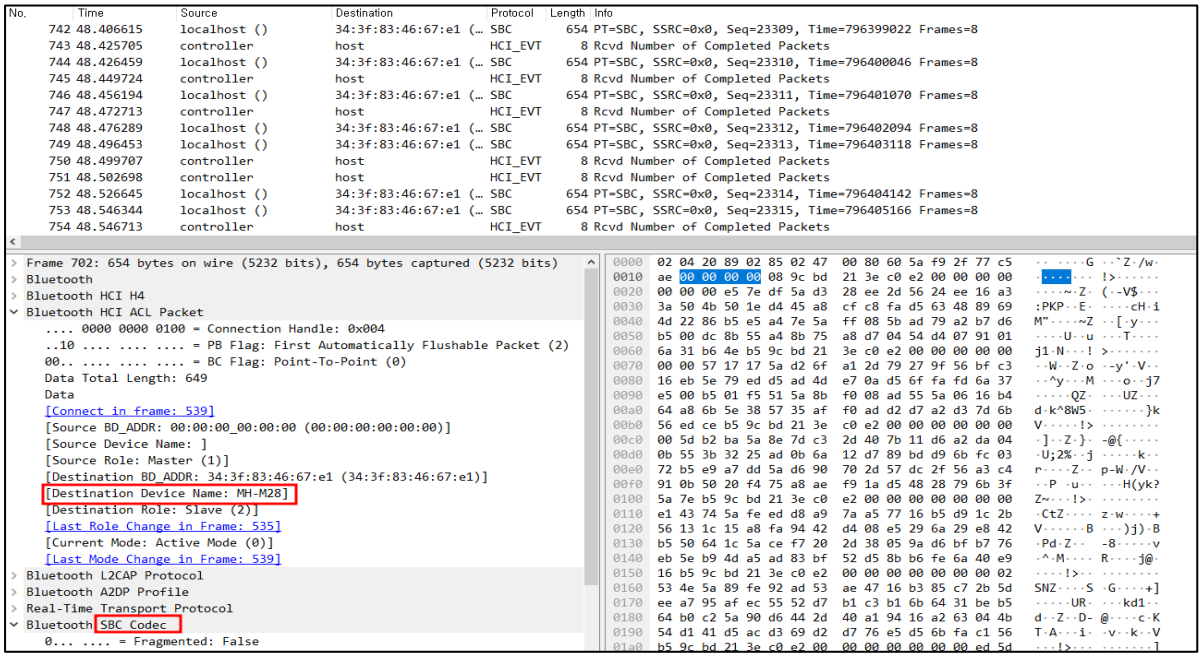
파이썬 코드 실행 결과는 아래와 같으며, "2023.11.23 13:30 KST" 메시지를 확인할 수 있다.



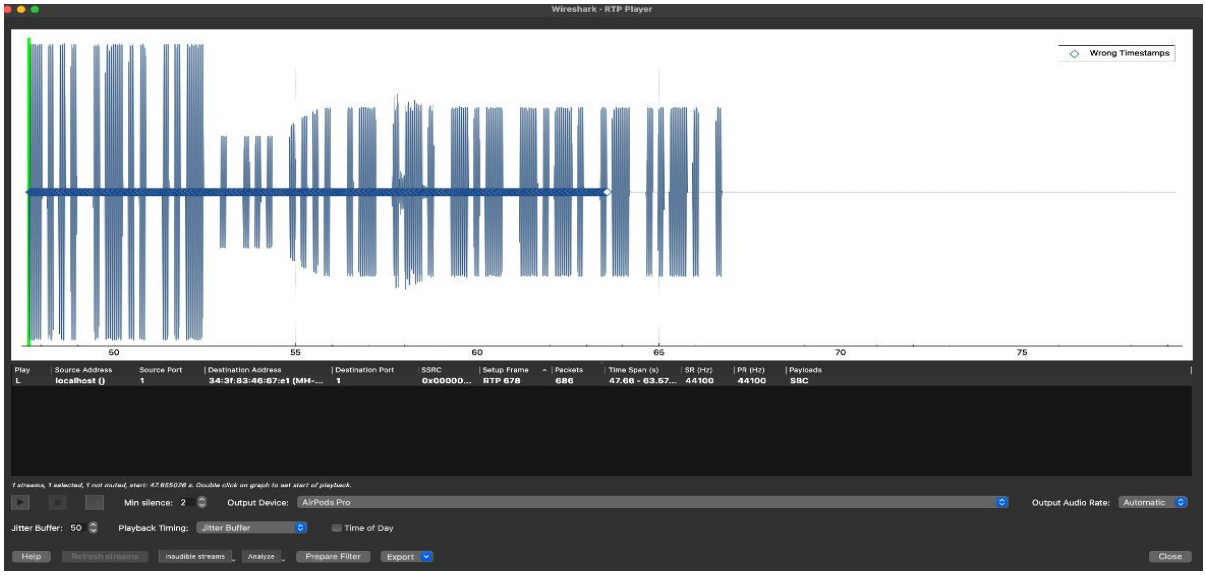
Q2. What is the message from evidence2? (40 points)

- BLUESHARKCAFE

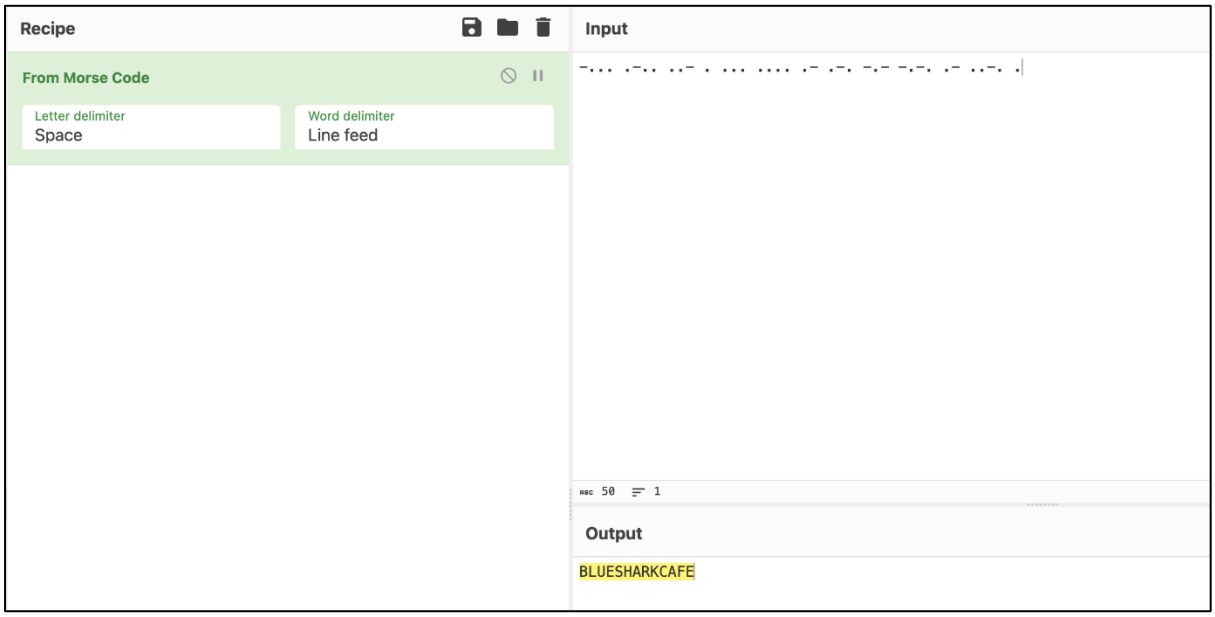
Evidence2.pcap 파일은 블루투스 오디오 패킷을 캡쳐한 파일이다. 패킷은 블루투스 A2DP(Advanced Audio Distribution Profile)를 통해 오디오 스트림을 전송한다. A2DP 프로파일을 통해 오디오 스트림 데이터를 RTP 프로토콜을 사용하여 전송하고 있으며, SBC 오디오 코덱을 사용하고 "MH-M28" 모델의 블루투스 오디오 모듈을 사용하는 것을 알 수 있다.



Wireshark에서 제공하는 RTP 스트림 재생 기능 RTP Player를 통해 음성 데이터를 재생할 수 있다.



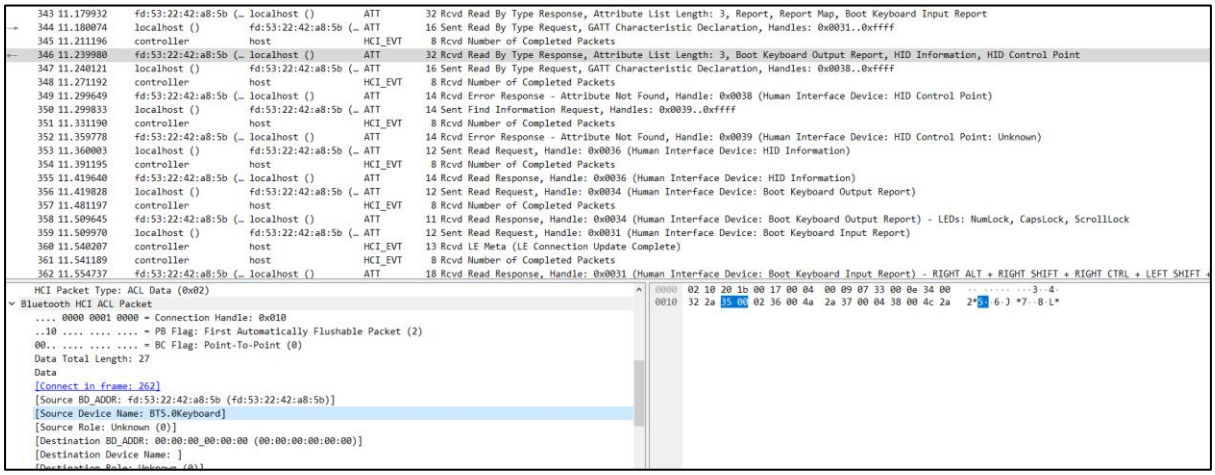
음성 신호의 패턴 분석 결과 모스부호로 이루어진 음성임을 파악하였고, 이를 Cyberchef를 통해 문자열로 변환하면 "BLUESHARKCAFE"임을 확인할 수 있다.



Q3. What is the message from evidence3? (20 points)

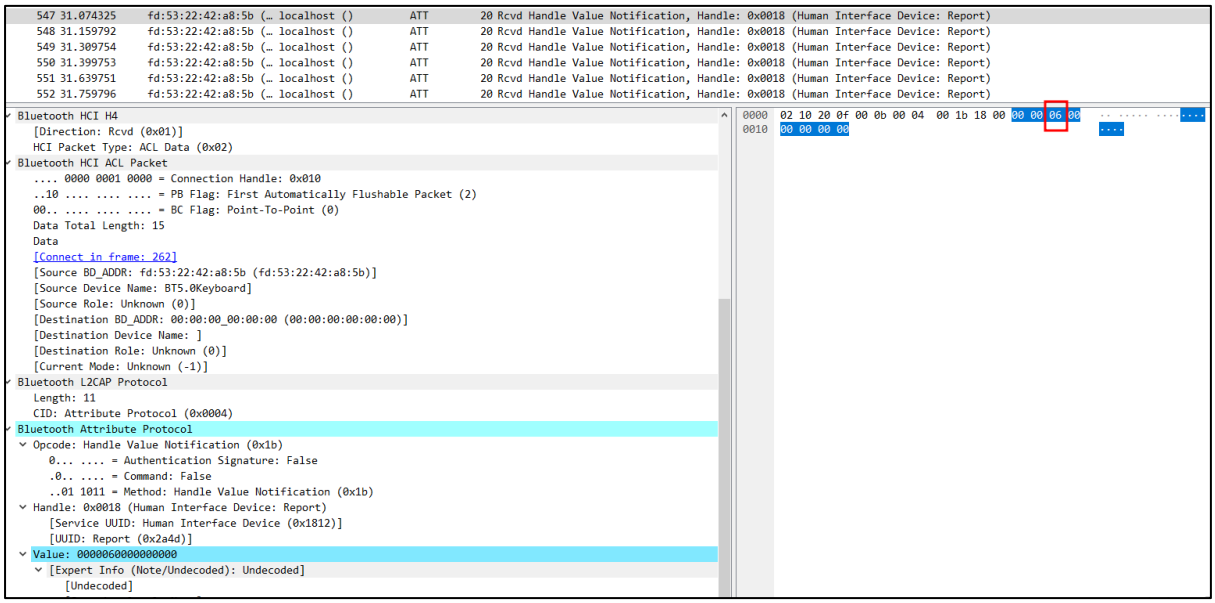
- contact the person in the black hat

Evidence3.pcap 파일은 블루투스 키보드 패킷을 캡쳐한 파일이다. 블루투스 키보드는 HID(Human Interface Device) 프로파일을 사용하여 데이터를 전송하며, GATT(Generic Attribute Profile)를 기반으로 하여 프로토콜 스택의 계층 구조를 따른다. 문제에서 주어진 패킷 파일을 살펴보면, 블루투스 키보드의 패킷 구조를 나타내고 있음을 확인 가능하다.



GATT는 ATT 프로토콜을 기반으로 동작하기 때문에 키보드의 입력 데이터는 ATT의 Attribute에 포함되어 GATT 프로파일을 통해 전송된다. 입력 데이터는 Value 값에서 확인 가능하며, 입력된 문자의 대소문자 여부를 판단하는 modifier byte*가 "00"으로 입력된 문자는 모두 소문자인 것을 알 수 있다. (대문자의 경우 "20")

- https://cdn.sparkfun.com/datasheets/Wireless/Bluetooth/RN-HID-User-Guide-v1.0r.pdf



HID Keyboard Usage Table(https://www.usb.org/sites/default/files/documents/hut1_12v2.pdf) 에서 hex 값에 따른 문자열 테이블을 찾은 후 파이썬으로 pcap 파일 내부 패턴을 파싱하여 문자열로 변환하는 코드를 작성하였다.

Q3.py

```
import binascii

urb = {
    '04': 'a',
    '05': 'b',
    '06': 'c',
    '07': 'd',
    '08': 'e',
    '09': 'f',
    '0a': 'g',
    '0b': 'h',
    '0c': 'i',
    '0d': 'j',
    '0e': 'k',
    '0f': 'l',
    '10': 'm',
    '11': 'n',
    '12': 'o',
    '13': 'p',
```

```
    '14': 'q',
    '15': 'r',
    '16': 's',
    '17': 't',
    '18': 'u',
    '19': 'v',
    '1a': 'w',
    '1b': 'x',
    '1c': 'y',
    '1d': 'z',
    '1e': '1',
    '1f': '2',
    '20': '3',
    '21': '4',
    '22': '5',
    '23': '6',
    '24': '7',
    '25': '8',
    '26': '9',
    '27': '0',
    '28': '₩n',
    '29': '[ESC]',
    '2a': '[BACKSPACE]',
    '2b': '[TAB]',
    '2c': ' ',
    '2d': '_',
    '2e': '=',
    '2f': '{',
    '30': '}',
    '31': '[CAPSLOCK]',
    '32': '[F1]',
    '33': '[F2]',
    '34': '[F3]',
    '35': '[F4]',
    '36': '[F5]',
    '37': '[F6]',
    '38': '[F7]',
    '39': '[F8]',
    '3a': '[F9]',
```

```
    '3b': '[F10]',
    '3c': '[F11]',
    '3d': '[F12]',
    '4c': '[DELETE]',
    '4f': '[RIGHT]',
    '50': '[LEFT]',
    '51': '[DOWN]',
    '52': '[UP]'
}

def find_hex_values(pcap_file):
    pattern = '02 10 20 0F 00 0B 00 04 00 1B 18 00 00 00'
    hex_values = []

    with open(pcap_file, 'rb') as file:
        pcap_data = file.read()

    pattern_bytes = bytes.fromhex(pattern)

    indices = [i for i in range(len(pcap_data)) if pcap_data[i:i+len(pattern_bytes)] == pattern_bytes]

    for index in indices:
        hex_byte = pcap_data[index + len(pattern_bytes):index + len(pattern_bytes) + 1]
        hex_value = hex_byte.hex().lower()

        if hex_value == '00':
            continue

        if hex_value in urb:
            hex_values.append(urb[hex_value])

    return hex_values

pcap_file = 'evidence3.pcap'
result = find_hex_values(pcap_file)
print(''.join(result))
```

파이썬 코드 실행 결과는 아래와 같으며, "contact the person in the black hat" 메시지를 확인할 수 있다.

```
D:₩Users₩mini₩Desktop₩DFC2023₩105 - BlueShark₩evidence>python Q3.py evidence3.pcap
contact the person in the black hat
```