204 – SPY's sabotage

Team Information

Team Name : ISEGYE_IDOL

Team Member : Eungchang Lee, Sojeong Kim, Mingyu Seong, Donghyun HA

Email Address : dfc-isegyeidol@googlegroups.com

Instructions

Description Investigators raided the office after receiving an anonymous tip that a spy was targeting someone. Investigators collected some data while the spy was destroying evidence. Analyze the collected image to find the orders and missions the spy received.

Target	Hash (MD5)
trudy_pc.ad1	59b1babd6c2a71acb73e3ddec184e5fc

Questions

1) When is the spy's mission date? (50 points)

2) Where is the spy's mission location? (50 points)

3) Who is the spy targeting? (100 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:	FTK Imager	Publisher:	AccessData
Version:	4.5.0.3		
URL:	https://accessdata.com/		

Name:	DB Browser for SQLite	Publisher:	DigitalOcean
Version:	3.12.2		
URL:	http://sqlitebrowser.org		

Name:	HxD	Publisher:	Maël Hörz
Version:	2.5.0.0		
URL:	https://www.mh-nexus.de		

Name:	Wireshark	Publisher:	The Wireshark team
Version:	3.6.5		
URL:	https://www.wireshark.or		

Name:	UsbMiceDataHacker	Publisher:	WangYihang
Version:	9 Oct 2020		
URL:	https://github.com/WangYihang/UsbMiceDataHacker		

Name:	UsbKeyboardDataHacker	Publisher:	WangYihang
Version:	9 Oct 2020		
URL:	https://github.com/WangYihang/UsbKeyboardDataHacker		

Name:	El Brainfuck	Publisher:	
Version:			
URL:	https://copy.sh/brainfuck/		

Name:	audacity	Publisher:	audacity team
Version:	3.1.3		
URL:	https://www.audacityteam.org/		

Name:	Morse Code Adaptive Audio Decoder	Publisher:	
Version:			
URL:	https://morsecode.world/international/decoder/audio-decoder-adaptive.html		

Step-by-step methodology:

FTK Imager 를 사용하여 아래 아티팩트를 수집 후 분석하였다.

수집한 아티팩트 목록	
Chrome 사용기록	[root]₩Users₩trudy₩Appdata₩Local₩Google₩Chrome₩User Data₩Default₩History

[표 1] 수집한 아티팩트 목록

다음으로 spy가 chrome을 사용한 기록을 시간 순서대로 정리하였다

2022년 5월 27일 금요일 오후 9:46:09 KST ~ 2022년 5월 27일 금요일 오후 9:47:55 KST

분석 아티팩트	아티팩트 분석 프로그램
History	DB Browser for SQLite

[표 2] 분석 아티팩트와 분석 프로그램

Chrome을 사용하여 본인(trudy)의 gmail 받은 메일함과 받은 메일들을 확인, 메일 첨부파일을 다운로드.

1	https://gmail.com/	Gmail	1	1	13298129169810
2	https://www.google.com/gmail/	Gmail	1	0	13298129169810
3	https://mail.google.com/mail/	Gmail	1	0	13298129169810
4	https://mail.google.com/mail/u/0/	받은편지함 (15) - dfc.trudy@gmail.com - Gmail	2	0	13298129186518
5	https://accounts.google.com/ServiceLogin?service=mail&passive=1209600&osid=1&continue=https://mail.google.com/mail/…	Gmail	2	0	13298129170852
6	https://accounts.google.com/signin/v2/identifier?…	Gmail	1	0	13298129171035
7	https://accounts.google.com/signin/v2/challenge/pwd?…	Gmail	1	0	13298129175910
8	https://accounts.google.com/CheckCookie?…	계정 복구 옵션	1	0	13298129181043
9	https://accounts.google.com/ServiceLogin?…	계정 복구 옵션	1	0	13298129181043
10	https://myaccount.google.com/accounts/SetOSID?…	계정 복구 옵션	1	0	13298129181043
11	https://myaccount.google.com/security/signinoptions/recovery-options-collection?…	계정 복구 옵션	1	0	13298129181043
12	https://myaccount.google.com/signinoptions/recovery-options-collection?…	계정 복구 옵션	1	0	13298129181043
13	https://myaccount.google.com/signinoptions/recovery-options-collection?…	계정 복구 옵션	2	0	13298129182535
14	https://accounts.google.com/ServiceLogin?…	받은편지함 (15) - dfc.trudy@gmail.com - Gmail	1	0	13298129186518
15	https://mail.google.com/accounts/SetOSID?…	받은편지함 (15) - dfc.trudy@gmail.com - Gmail	1	0	13298129186518
16	https://accounts.youtube.com/accounts/SetSID?ssdc=1&sidt=ALWU2csQk6lAgU7Rb44mi7O1DaeWnfyz6ee0/…	받은편지함 (15) - dfc.trudy@gmail.com - Gmail	1	0	13298129186518
17	https://accounts.google.co.kr/accounts/SetSID?…	받은편지함 (15) - dfc.trudy@gmail.com - Gmail	1	0	13298129186518
18	https://mail.google.com/mail/u/0/?pli=1	받은편지함 (15) - dfc.trudy@gmail.com - Gmail	1	0	13298129186518
19	https://mail.google.com/mail/u/0/#inbox	받은편지함 (15) - dfc.trudy@gmail.com - Gmail	4	0	13298129227166
20	https://mail.google.com/mail/u/0/#inbox/FMfcgzGpGBFGDpcwVWmnMQTBBbqjpPss	Ah.. Ah... - dfc.trudy@gmail.com - Gmail	1	0	13298129196474
21	https://mail.google.com/mail/u/0/#inbox/FMfcgzGpGBFGNGTpqHprcwqjFXbWtzDv	Trudy, Good luck.. - dfc.trudy@gmail.com - Gmail	4	0	13298129215914
22	https://mail.google.com/mail/u/0/#inbox/FMfcgzGpGBFGNGTpqHprcwqjFXbWtzDv?projector=1&messagePartId=0.1	Trudy, Good luck.. - dfc.trudy@gmail.com - Gmail	3	0	13298129214991
23	https://mail.google.com/mail/u/0/#inbox/FMfcgzGpGBFGNKsHsKGQTVbHvxKGNktC	Trudy, FYI - dfc.trudy@gmail.com - Gmail	5	0	13298129225920
24	https://mail.google.com/mail/u/0/#inbox/FMfcgzGpGBFGNKsHsKGQTVbHvxKGNktC?projector=1&messagePartId=0.1	Trudy, FYI - dfc.trudy@gmail.com - Gmail	4	0	13298129223224
25	https://mail.google.com/mail/u/0/#inbox/FMfcgzGpGBFGWRpHJfNDwtsnBzhBsbzk	Trudy, God Bless You - dfc.trudy@gmail.com - Gmail	2	0	13298129234923
26	https://mail.google.com/mail/u/0/#inbox/FMfcgzGpGBFGWRpHJfNDwtsnBzhBsbzk?projector=1&messagePartId=0.1	Trudy, God Bless You - dfc.trudy@gmail.com - Gmail	2	0	13298129235400
27	https://chrome.google.com/webstore?hl=ko	Chrome 웹 스토어 - 확장 프로그램	1	0	13298129273964
28	https://chrome.google.com/webstore/category/extensions?hl=ko	Chrome 웹 스토어 - 확장 프로그램	1	0	13298129275064

[그림 1] 웹페이지 방문 기록

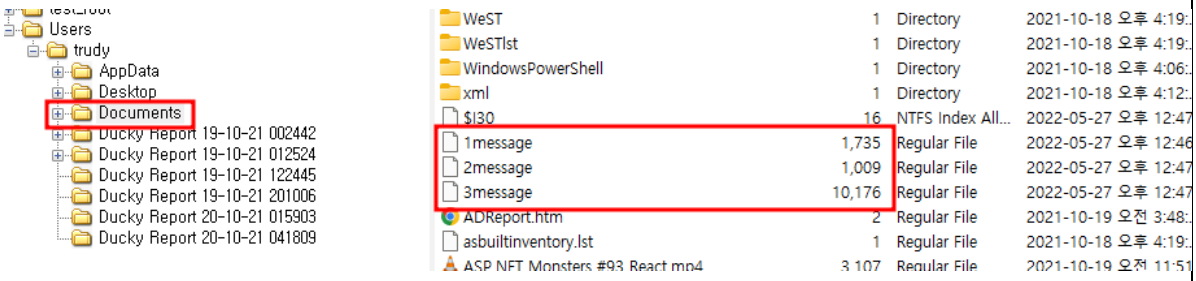
id	guid	current_path	target_path	start_time
…	필터	필터	필터	필터
2	06f0be79-eca7-4001-89ba-1d496719e286	C:₩Users₩trudy₩Downloads₩1message	C:₩Users₩trudy₩Downloads₩1message	13298129209198968
3	d29e6dd1-4543-4b2a-942e-6d80c4f70b11	C:₩Users₩trudy₩Downloads₩2message	C:₩Users₩trudy₩Downloads₩2message	13298129224709374
4	7b90516a-89ff-498c-9517-8bd46104c17b	C:₩Users₩trudy₩Downloads₩3message	C:₩Users₩trudy₩Downloads₩3message	13298129238028355

[그림 2] 파일 다운로드 기록

메일 제목	파일 경로	다운로드 시각
Trudy, Good luck.	C:₩Users₩trudy₩Downloads₩1message	2022년 5월 27일 금요일 오후 9:46:49 KST
Trudy, FYI	C:₩Users₩trudy₩Downloads₩2message	2022년 5월 27일 금요일 오후 9:47:04 KST
Trudy, God Bless You	C:₩Users₩trudy₩Downloads₩3message	2022년 5월 27일 금요일 오후 9:47:18 KST

[표 3] 파일 다운로드 기록 및 메일 제목

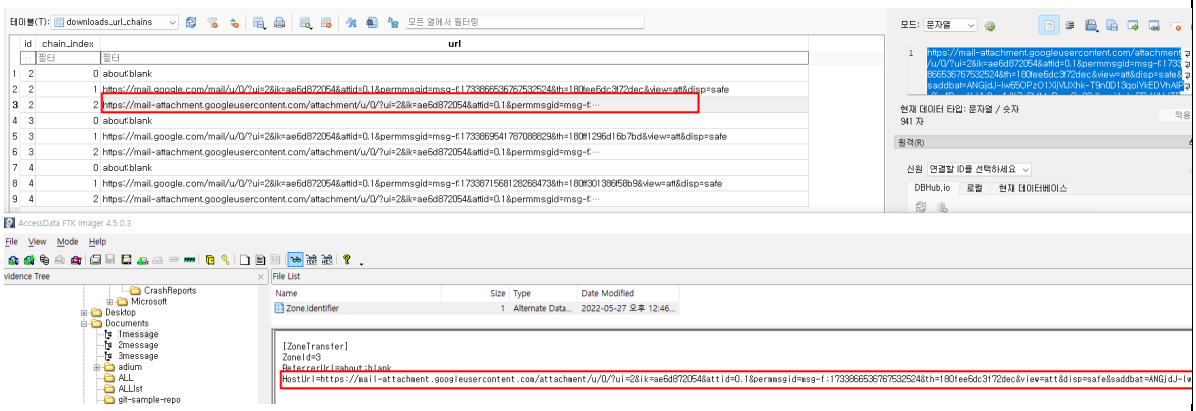
제출된 증거에서는 위 파일 경로가 존재하지 않았으나, 동일한 파일명을 가진 파일을 [root]₩Users₩trudy₩Documents 에서 찾을 수 있었음. 파일 다운로드 기록에 기록된 파일 용량과 일치하며, 크롬 History의 downloads_url_chains에 기록된 url과 각 message파일의 zone.identifier에 기록된 HostUrl이 일치하므로 동일한 파일로 볼 수 있다.



[그림 3] 다운로드 파일

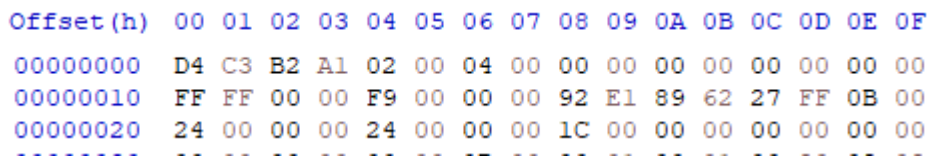
target_path	start_time	received_bytes	total_bytes	s
	필터	필터	필터	필
₩Downloads₩1message	13298129209198968	1776468	1776468	
₩Downloads₩2message	13298129224709374	1032280	1032280	
₩Downloads₩3message	13298129238028355	10419820	10419820	

[그림 4] 파일 다운로드 기록



[그림 5] downloads_url_chains테이블과 zone.identifier

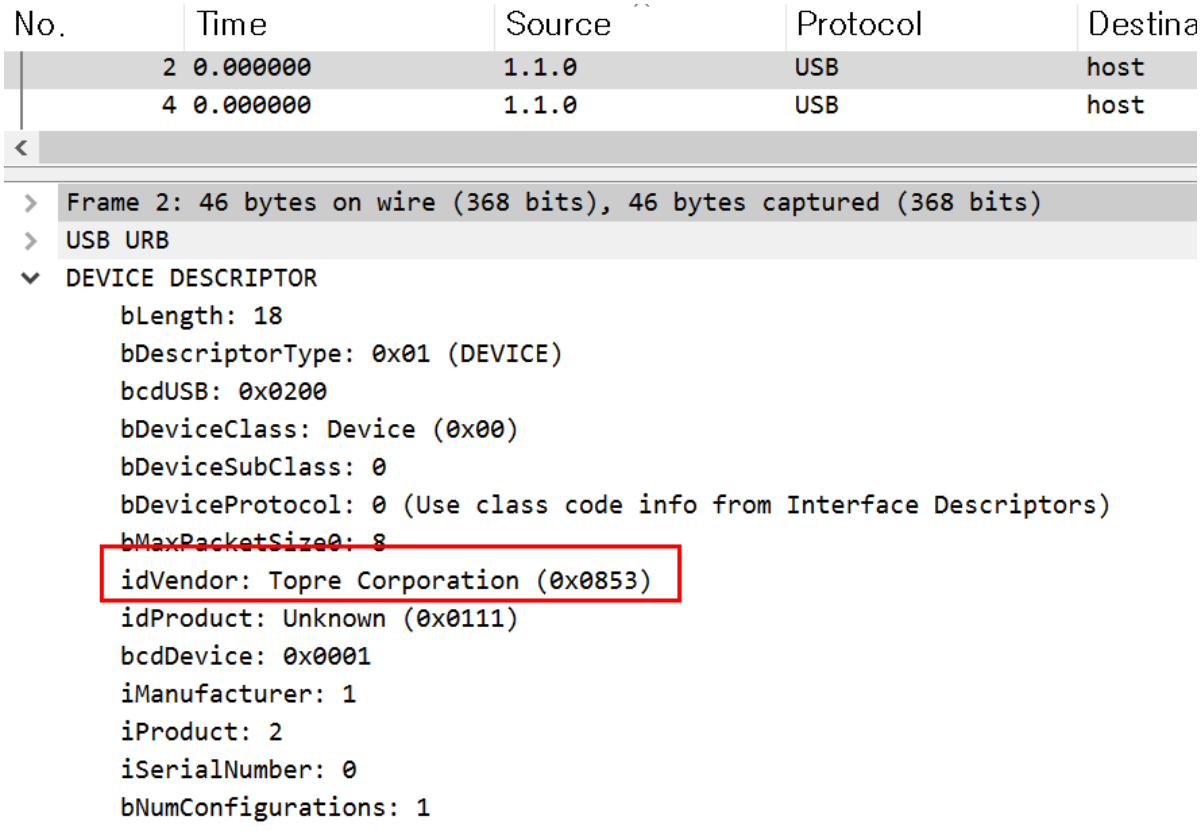
다운로드한 3개 파일은 모두 'D4 B2 C3 D4' 시그니쳐를 가지고 있으므로 pcap 파일로 확인된다.



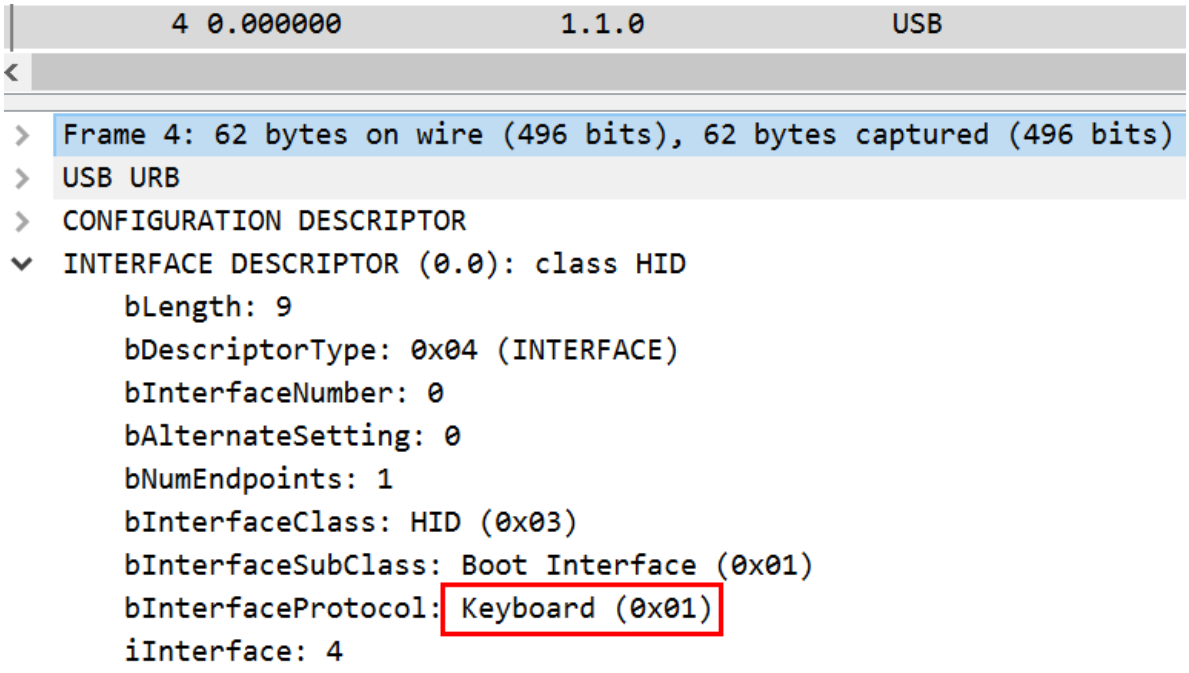
Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000000	D4	C3	B2	A1	02	00	04	00	00	00	00	00	00	00	00	00
00000010	FF	FF	00	00	F9	00	00	00	92	E1	89	62	27	FF	0B	00
00000020	24	00	00	00	24	00	00	00	1C	00	00	00	00	00	00	00

[그림 6] 1message 파일의 헤더부분

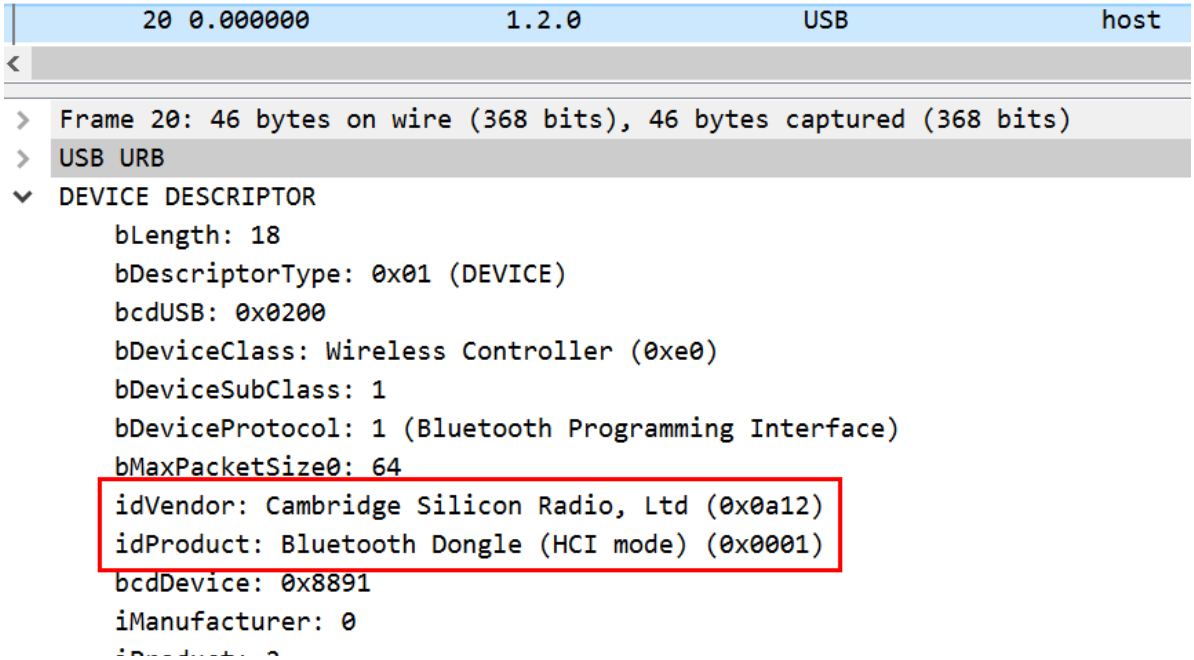
3개 파일을 wireshark로 분석해보면 usb 패킷 파일이 저장되어 있으며, 모두 거의 동일한 환경에서 생성된 것으로 보인다. USB DESCRIPTOR Request/Response DEVICE/CONFIGURATION 패킷을 통해 연결된 장치의 정보를 수집할 수 있으며, 각 장치 별 패킷 내용 및 전체 장치를 모두 정리한 내용은 아래 표와 같다.



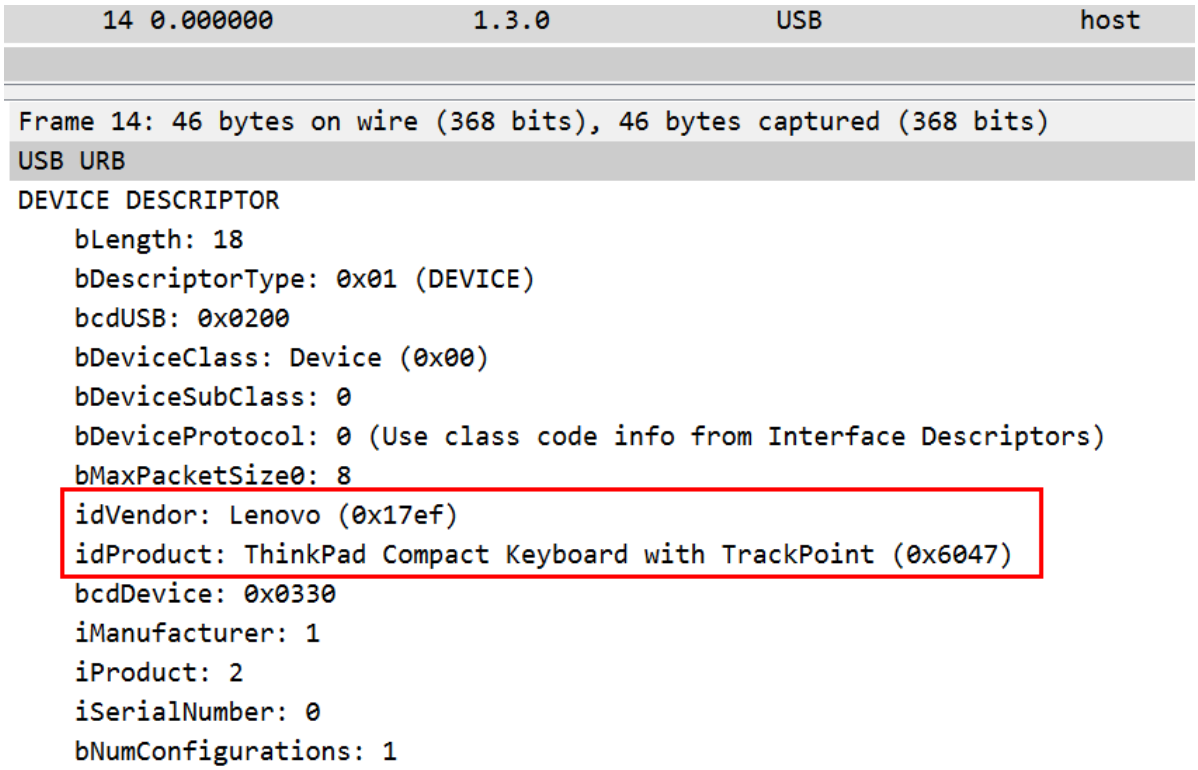
[그림 7] 1.1.0 DESCRIPTOR Response DEVICE



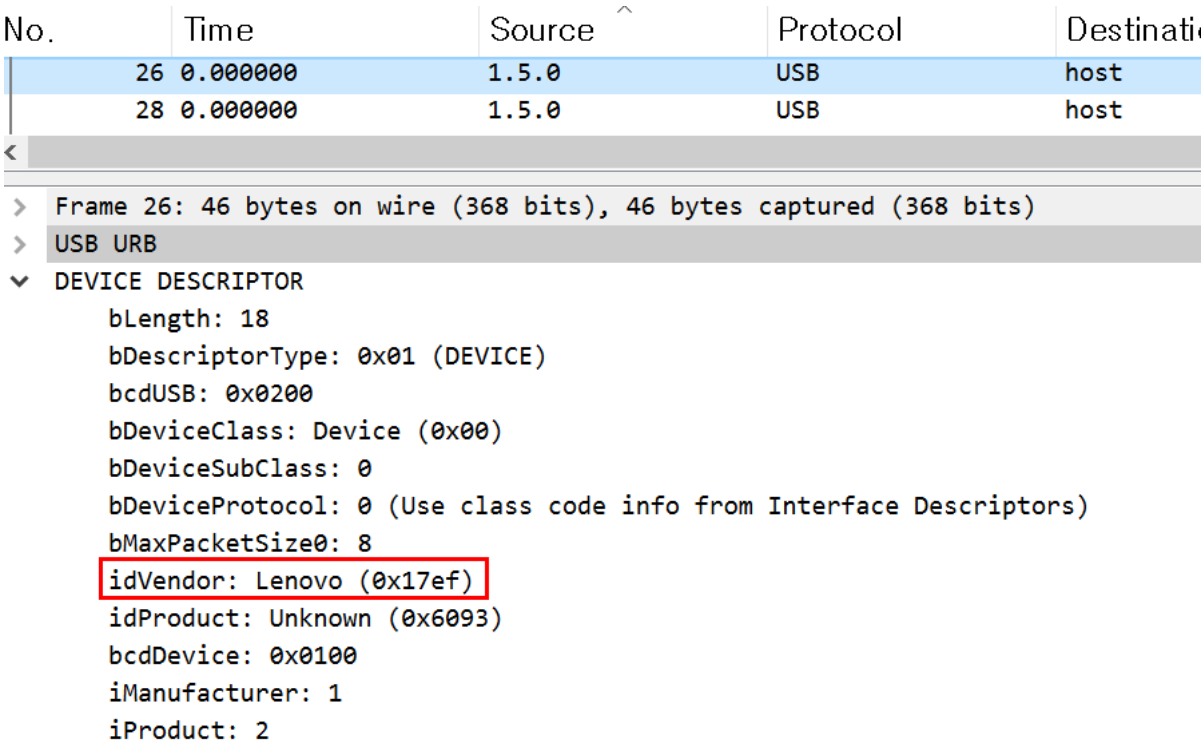
[그림 8] 1.1.0 DESCRIPTOR Response CONFIGURATION



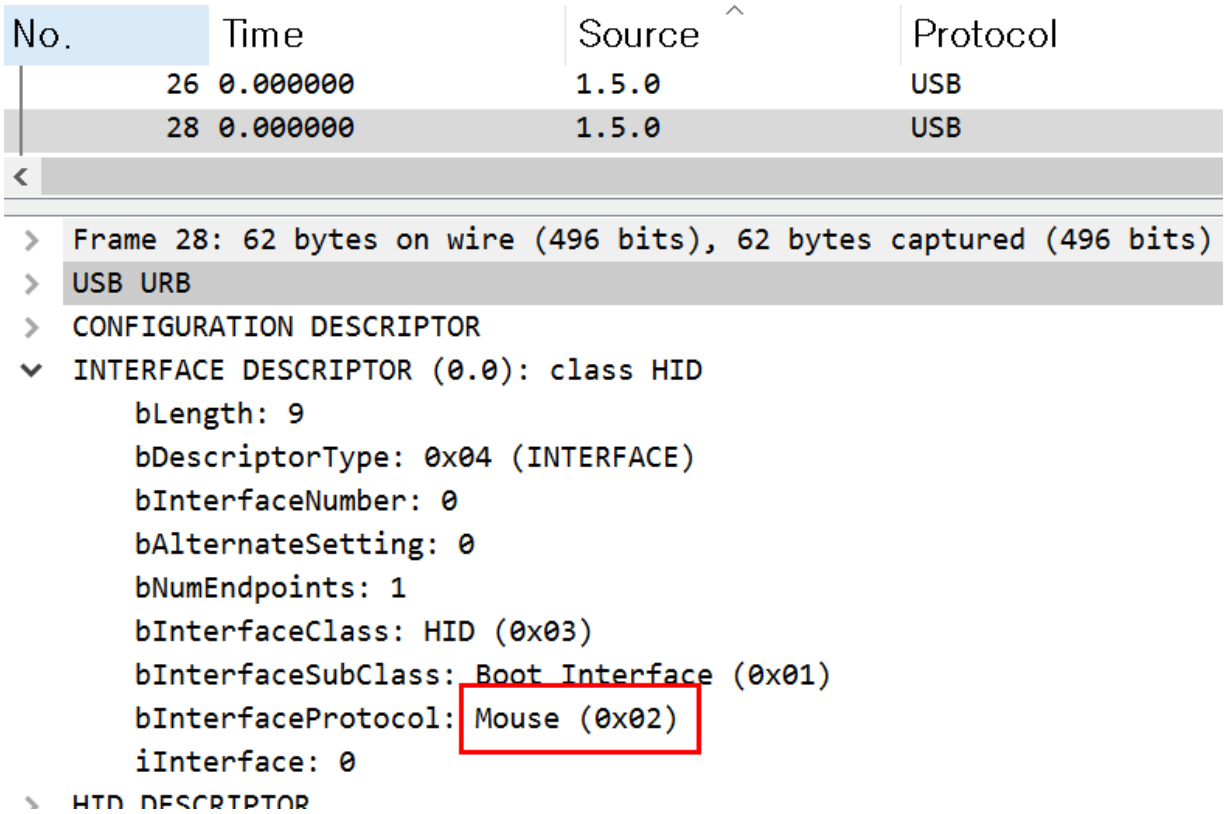
[그림 9] 1.2.0 DESCRIPTOR Response DEVICE



[그림 10] 1.3.0 DESCRIPTOR Response DEVICE



[그림 11] 1.5.0 DESCRIPTOR Response DEVICE



[그림 12] 1.5.0 DESCRIPTOR Response CONFIGURATION

```
     8 0.000000          1.8.0          USB          host
    10 0.000000          1.8.0          USB          host
    12 0.000000          1.8.0          USB          host
   849 8.349324          1.8.2          USB          host

Frame 8: 46 bytes on wire (368 bits), 46 bytes captured (368 bits)
USB URB
DEVICE DESCRIPTOR
    bLength: 18
    bDescriptorType: 0x01 (DEVICE)
    bcdUSB: 0x0110
    bDeviceClass: Device (0x00)
    bDeviceSubClass: 0
    bDeviceProtocol: 0 (Use class code info from Interface Descriptors)
    bMaxPacketSize0: 8
    idVendor: C-Media Electronics, Inc. (0x0d8c)
    idProduct: Unknown (0x0012)
    bcdDevice: 0x0100
```

[그림 13] 1.8.0 DESCRIPTOR Response DEVICE

```
     8 0.000000          1.8.0          USB          host
    10 0.000000          1.8.0          USB          host
    12 0.000000          1.8.0          USB          host
   849 8.349324          1.8.2          USB          host

Frame 10: 281 bytes on wire (2248 bits), 281 bytes captured (2248 bits)
USB URB
CONFIGURATION DESCRIPTOR
INTERFACE DESCRIPTOR (0.0): class Audio
    bLength: 9
    bDescriptorType: 0x04 (INTERFACE)
    bInterfaceNumber: 0
    bAlternateSetting: 0
    bNumEndpoints: 0
    bInterfaceClass: Audio (0x01)
    bInterfaceSubClass: Audio Control (0x01)
    bInterfaceProtocol: 0x00
    iInterface: 0
Class-specific Audio Control Interface Descriptor: Header Descriptor
```

[그림 14] 1.8.0 DESCRIPTOR Response CONFIGURATION

```
   5495 63.500133          1.9.0                USB                  host
   5497 63.500291          1.9.0                USB                  host
   5499 63.500813          1.9.0                USB                  host
   5503 63.503589          1.9.0                USB                  host
```

```
Frame 5495: 46 bytes on wire (368 bits), 46 bytes captured (368 bits)
USB URB
DEVICE DESCRIPTOR
    bLength: 18
    bDescriptorType: 0x01 (DEVICE)
    bcdUSB: 0x0110
    bDeviceClass: Device (0x00)
    bDeviceSubClass: 0
    bDeviceProtocol: 0 (Use class code info from Interface Descriptors)
    bMaxPacketSize0: 8
    idVendor: C-Media Electronics, Inc. (0x0d8c)
    idProduct: Unknown (0x0012)
    bcdDevice: 0x0100
```

[그림 15] 1.9.0 DESCRIPTOR Response DEVICE

```
   5499 63.500813          1.9.0                USB                  host
   5503 63.503589          1.9.0                USB                  host
```

```
Frame 5499: 281 bytes on wire (2248 bits), 281 bytes captured (2248 bits)
USB URB
CONFIGURATION DESCRIPTOR
INTERFACE DESCRIPTOR (0.0): class Audio
    bLength: 9
    bDescriptorType: 0x04 (INTERFACE)
    bInterfaceNumber: 0
    bAlternateSetting: 0
    bNumEndpoints: 0
    bInterfaceClass: Audio (0x01)
    bInterfaceSubClass: Audio Control (0x01)
    bInterfaceProtocol: 0x00
    iInterface: 0
Class-specific Audio Control Interface Descriptor: Header Descriptor
```

[그림 16] 1.9.0 DESCRIPTOR Response CONFIGURATION

주소	공급자	장치
1.1.0	topre corporation	keyboard
1.2.0	Cambridge silicon radio	bluetooth dongle
1.3.0	lenovo	thinkpad compact keyboard with trackpoint
1.4.0	unknown	unknown
1.5.0	lenovo	unknown
1.8.0	C-media Electronics	오디오 장치 (실제와 다름)
1.9.0	C-media Electronics	오디오 장치

[표 4] 연결된 USB 장치 목록

1.4.0에 연결된 장치에 대한 DESCRIPTOR 패킷은 존재하지 않았다. 그리고, DESCRIPTOR 패킷 정보와 실제 연결된 장비가 다른 경우도 있었다. 이에 대해서는 아래 문제 풀이에 추가 서술하였다.

1. When is the spy's mission date? (50 points)

1message 파일을 wireshark로 열어서 분석하였다.

18	0.000000	1.3.0	USB	host	28 SET CONFIGURATION Response
19	0.000000	host	USB	1.2.0	36 GET DESCRIPTOR Request DEVICE
20	0.000000	1.2.0	USB	host	46 GET DESCRIPTOR Response DEVICE
21	0.000000	host	USB	1.2.0	36 GET DESCRIPTOR Request CONFIGURATION
22	0.000000	1.2.0	USB	host	205 GET DESCRIPTOR Response CONFIGURATION
23	0.000000	host	USB	1.2.0	36 SET CONFIGURATION Request
24	0.000000	1.2.0	USB	host	28 SET CONFIGURATION Response
25	0.000000	host	USB	1.5.0	36 GET DESCRIPTOR Request DEVICE
26	0.000000	1.5.0	USB	host	46 GET DESCRIPTOR Response DEVICE
27	0.000000	host	USB	1.5.0	36 GET DESCRIPTOR Request CONFIGURATION
28	0.000000	1.5.0	USB	host	62 GET DESCRIPTOR Response CONFIGURATION
29	0.000000	host	USB	1.5.0	36 SET CONFIGURATION Request
30	0.000000	1.5.0	USB	host	28 SET CONFIGURATION Response
31	0.141412	1.5.1	USB	host	35 URB_INTERRUPT in
32	0.141543	host	USB	1.5.1	27 URB_INTERRUPT in
33	0.149407	1.5.1	USB	host	35 URB_INTERRUPT in
34	0.149640	host	USB	1.5.1	27 URB_INTERRUPT in
35	0.157438	1.5.1	USB	host	35 URB_INTERRUPT in
36	0.157465	host	USB	1.5.1	27 URB_INTERRUPT in
37	0.165469	1.5.1	USB	host	35 URB_INTERRUPT in
38	0.165500	host	USB	1.5.1	27 URB_INTERRUPT in
39	0.173413	1.5.1	USB	host	35 URB_INTERRUPT in
40	0.173479	host	USB	1.5.1	27 URB_INTERRUPT in

[그림 17] 1message 파일 패킷 내용 캡쳐

```
    33 0.149407          1.5.1          USB          host
    34 0.149640          host           USB          1.5.1
    35 0.157438          1.5.1          USB          host
    36 0.157465          host           USB          1.5.1

    [Destination: host]
    USBPcap pseudoheader length: 27
    IRP ID: 0xffffe589457ed9b0
    IRP USBD_STATUS: USBD_STATUS_SUCCESS (0x00000000)
    URB Function: URB_FUNCTION_BULK_OR_INTERRUPT_TRANSFER (0x0009)
  > IRP information: 0x01, Direction: PDO -> FDO
    URB bus id: 1
    Device address: 5
  > Endpoint: 0x81, Direction: IN
    URB transfer type: URB_INTERRUPT (0x01)
    Packet Data Length: 8
    [bInterfaceClass: HID (0x03)]
HID Data: 0000ff000000ffff
```

[그림 18] 1.5.1 URB_interrupt in, HID Data

1.5.1의 product id가 unknown으로 표시되어 정확히 어떤 종류의 장치가 연결되어 있는지 파악할 수 없으나, bInterfaceClass : HID (0x03)을 보면 HID 장치임을 알 수 있었다. 이어서 더 정확히 어떤 장비인지 파악하기 위해서 interrupt in 패킷에 담긴 HID Data의 형식을 살펴보았다. HID Data들을 살펴보기 위해 tshark로 추출하였다.

사용한 명령어는 아래와 같다.

tshark -r ./1message.pcap -Y 'usb.capdata && frame.len == 35' -T fields -e usb.capdata > 1m_151.txt

```
00:fd:02:00:fd:ff:02:00
00:fd:01:00:fd:ff:01:00
00:fc:02:00:fc:ff:02:00
00:fb:02:00:fb:ff:02:00
00:fa:02:00:fa:ff:02:00
00:f9:01:00:f9:ff:01:00
00:f9:03:00:f9:ff:03:00
00:f7:02:00:f7:ff:02:00
00:f9:00:00:f9:ff:00:00
00:f6:01:00:f6:ff:01:00
00:f5:02:00:f5:ff:02:00
00:f4:00:00:f4:ff:00:00
00:f4:01:00:f4:ff:01:00
00:f4:00:00:f4:ff:00:00
00:f9:00:00:f9:ff:00:00
00:fa:00:00:fa:ff:00:00
00:fc:00:00:fc:ff:00:00
00:fd:00:00:fd:ff:00:00
00:fe:00:00:fe:ff:00:00
00:fd:00:00:fd:ff:00:00
```

[그림 19] 1.5.1 usb.capdata, 1m151.txt

전체 8바이트 데이터를 앞 뒤 4바이트씩 나누어 보면, 첫번째 블록의 두번째, 세번째 바이트가 두번째 블록의 첫번째, 세번째 바이트와 동일하다. 또한, 이 두 바이트의 데이터 범위가 0x00 ~ 0xff로 마우스 패킷 데이터와 유사하다. 두번째 바이트를 마우스가 horizontal 방향으로 움직인 거리, 세번째 바이트를 마우스가 vertical 방향으로 움직인 거리 데이터로 보고 이를 파싱하여 그림으로 나타내면 마우스가 움직인 경로를 시각화하여 볼 수 있다.

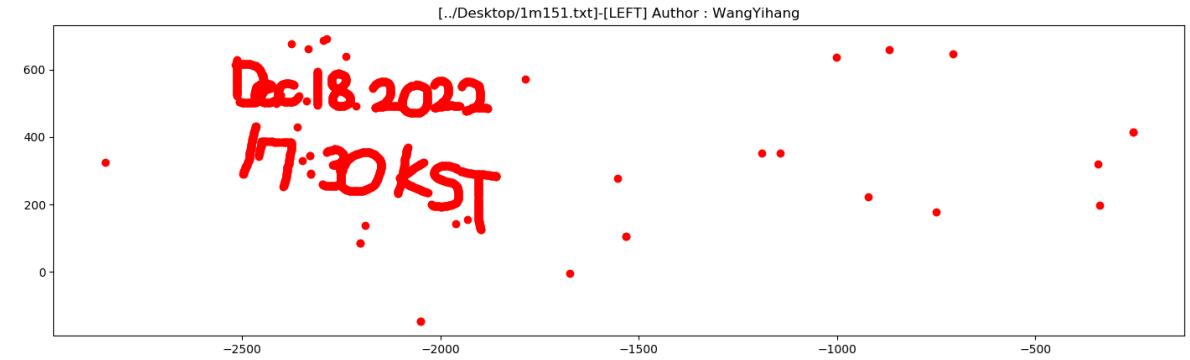
파싱에 사용된 코드는 UsbMiceDataHacker이며 해당 코드를 아래와 같이 일부 수정하였다.

```
34      # get argv
35      pcapFilePath = sys.argv[1]
36      action = sys.argv[2]
37
38      if action != "LEFT" and action != "ALL" and action != "RIGHT" and action != "MOVE":
39          action = "LEFT"
40
41      '''
42      # get data of pcap
43      command = "tshark -r %s -T fields -e usb.capdata > %s" % (
44          pcapFilePath, DataFileName)
45      print(command)
46      os.system(command)
47
48      # read data
49      with open(DataFileName, "r") as f:
50          for line in f:
51              data.append(line[0:-1])
52      '''
53      # read data
54      with open(pcapFilePath, "r") as f:
55          for line in f:
56              data.append(line[0:-1])
57
58      # handle move
59      for i in data:
60          Bytes = i.split(":")
61          if len(Bytes) == 8:
62              horizontal = 1  # -
63              vertical = 2 # |
64          elif len(Bytes) == 4:
```

[그림 20] UsbMiceDataHacker 코드 수정 (41~52행 주석처리, 53~56행 추가, 61~63행 수정)

사용한 명령어:

python3 UsbMiceDataHacker.py ./1m151.txt LEFT



[그림 21] 좌클릭 상태로 움직인 경로 시각화 실행 결과

그 결과, 시간 정보를 확인할 수 있었다. Spy의 mission date는 "**2022년 12월 18일 17:30 KST**"로 보인다.

2. Where is the spy's mission location? (50 points)

2message 파일을 wireshark로 열어서 분석하였다.

1.4.2 leftover capture data를 tshark로 추출하여 살펴 보았다. 사용한 명령어는 다음과 같다.

tshark -r ./2message.pcap -Y 'usb.capdata && frame.len == 42' -T fields -e usb.capdata > 2m_142.txt

```
mandu@mandu-VirtualBox:~/Desktop$ tshark -r ./2mess
age.pcap -Y 'usb.capdata && frame.len == 42' -T fie
lds -e usb.capdata > 2m_142.txt
mandu@mandu-VirtualBox:~/Desktop$ head 2m_142.txt
46:20:0b:00:07:00:41:00:a1:02:00:01:00:00:00
46:20:0b:00:07:00:41:00:a1:02:00:01:00:00:00
46:20:0b:00:07:00:41:00:a1:02:00:01:00:00:00
46:20:0b:00:07:00:41:00:a1:02:00:01:10:00:00
46:20:0b:00:07:00:41:00:a1:02:00:01:20:00:00
46:20:0b:00:07:00:41:00:a1:02:00:00:10:00:00
46:20:0b:00:07:00:41:00:a1:02:00:00:20:00:00
46:20:0b:00:07:00:41:00:a1:02:00:01:10:00:00
46:20:0b:00:07:00:41:00:a1:02:00:00:20:00:00
46:20:0b:00:07:00:41:00:a1:02:00:00:10:00:00
```

[그림 22] 1.4.2 leftover capture data

데이터를 살펴보면 전체 15바이트 중에서 상위 10바이트는 46:20:0b:00:07:00:41:00:a1:02로 동일하고, 그 다음 4바이트에 유의미한 데이터가 있는 것으로 보이며, 최하위 바이트는 0x00으로 이루어져 있다. 유의미한 데이터가 있는 것으로 보이는 4바이트 데이터 중에서 첫번째 바이트는 0x00, 0x01 중 하나를 가지며, 그 다음 두 개 바이트는 0x00~0xff 범위의 값을 가지고, 마지

막 바이트는 0x00 또는 0xff값을 가진다. 마우스 데이터와 유사한 것으로 보여서 아래 표와 같이 대응시켜 UsbMiceDataHacker를 일부 수정하여 데이터를 파싱하여 시각화했다. 수정한 내용은 아래와 같다.

offset	Value range	Mouse data format
10	0x00 or 0x01	click
11	0x00 ~ 0xff	Horizontal movement
12	0x00 ~ 0xff	Vertical movement
13	0x00 or 0xff	-

[표 5] capture data 마우스 값 대응

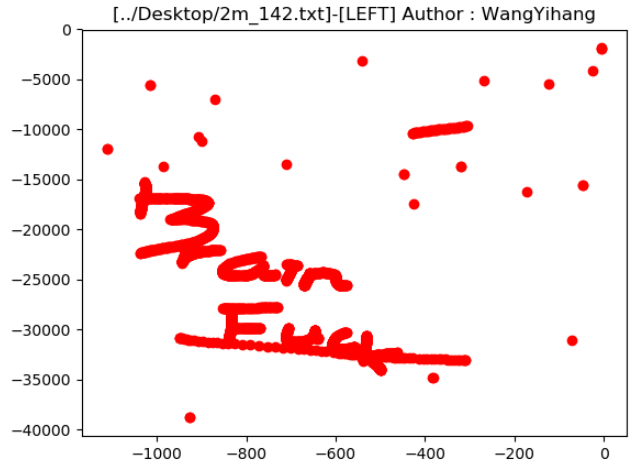
```
34      # get argv
35      pcapFilePath = sys.argv[1]
36      action = sys.argv[2]
37
38      if action != "LEFT" and action != "ALL" and action != "RIGHT" and action != "MOVE":
39          action = "LEFT"
40
41      '''
42      # get data of pcap
43      command = "tshark -r %s -T fields -e usb.capdata > %s" % (
44          pcapFilePath, DataFileName)
45      print(command)
46      os.system(command)
47
48      # read data
49      with open(DataFileName, "r") as f:
50          for line in f:
51              data.append(line[0:-1])
52      '''
53      # read data
54      with open(pcapFilePath, "r") as f:
55          for line in f:
56              data.append(line[0:-1])
57
58      # handle move
59      for i in data:
60          Bytes = i.split(":")
61          if len(Bytes) == 15:
62              horizontal = 11  #
63              vertical = 12 # |
64          elif len(Bytes) == 4:
```

[그림 23] UsbMiceDataHacker 코드 수정 (41~52행 주석처리, 53~56행 추가, 61~63행 수정)

```
77          if Bytes[10] == "01":
78              print("[+] Left butten.")
79              if action == "LEFT":
80                  # draw point to the image panel
81                  X.append(mousePositionX)
82                  Y.append(-mousePositionY)
83          elif Bytes[10] == "02":
84              print("[+] Right Butten.")
85              if action == "RIGHT":
86                  # draw point to the image panel
87                  X.append(mousePositionX)
88                  Y.append(-mousePositionY)
89          elif Bytes[10] == "00":
90              print("[+] Move.")
91              if action == "MOVE":
```

[그림 24] UsbMiceDataHacker 코드 수정 (77, 83, 89행 수정)

python3 UsbMiceDataHacker.py ./2m_142.txt LEFT



[그림 25] 좌클릭 상태로 움직인 경로 시각화 실행 결과

파싱한 결과를 이미지로 시각화하면 마우스로 Brain Fuck이라는 글자를 그린 것을 확인할 수 있다.

```
mandu@mandu-VirtualBox:~/Desktop$ tshark -r ./2mess
age.pcap -Y 'usb.capdata && frame.len == 35' -T fie
lds -e usb.capdata > 2m_181.txt
mandu@mandu-VirtualBox:~/Desktop$ head 2m_181.txt
00:00:09:00:00:00:00:00
00:00:00:00:00:00:00:00
00:00:18:00:00:00:00:00
00:00:00:00:00:00:00:00
00:00:06:00:00:00:00:00
00:00:00:00:00:00:00:00
00:00:0e:00:00:00:00:00
00:00:00:00:00:00:00:00
00:00:0c:00:00:00:00:00
00:00:00:00:00:00:00:00
```

[그림 26] 1.8.1 leftover capture data

다음은 1..8.2 데이터를 살펴 보았다. 1.8.0 디스크립터를 보고 오디오 장비처럼 보이지만, 8바이트 크기에 키보드 데이터 포맷과 유사해보인다. 따라서 UsbKeyboardDataHacker를 일부 수정 후 사용하여 데이터를 해석하였다.

```
28      # get argv
29      pcapFilePath = sys.argv[1]
30
31      # get data of pcap
32      #os.system("tshark -r %s -T fields -e usb.capdata 'usb.data_len == 8' > %s" % (pcapF
33
34      # read data
35      with open(pcapFilePath, "r") as f:
36          for line in f:
37              presses.append(line[0:-1])
```

[그림 27] UsbKeyboardDataHacker 코드 수정 (32행 주석처리, 35행)

python3 UsbKeyboardDataHacker.py ./2m_181.txt

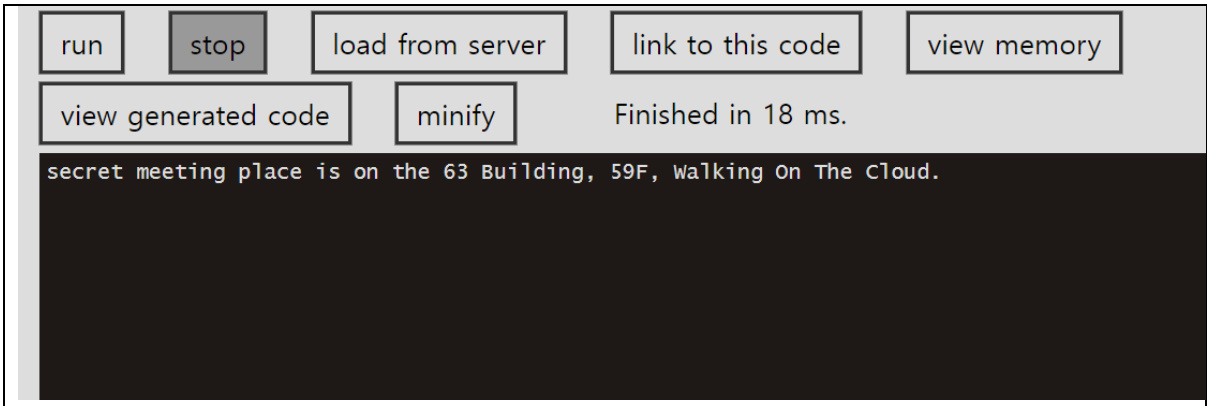
```
[+] Found : fucking<SPACE>bit<SPACE>ccoiin....<RET>hey<SPACE>trudy<SPACE>it<SPACE>is<SPACE>secret<SPA
CE>message.<SPACE>i<SPACE>believe<SPACE>you<SPACE>can<SPACE>read<SPACE>it.<RET>+[-------==-->++<]>+.+
===++++[->++++<]>.---.-[--->+<]>----.-------------.[--->+<]>---.[---->+<]>+++.+[----->+<]>.---------..[
-->+<]>---.-----------.+++++.--------.-[--->+<]>--.[-->++++++++<]>.----.-----------.++.++.--[--->+<]>-
-[--->+++<]>-.++++++++++.+[---->+<]>+++.+++++[->+++<]>.-.-[->++++++<]>-.---[->++++<]>.------------.---
--[--->+<]>-.------[->++<]>.----.{[--->+++<]>--.+[->+++<]>.++[----->+<]>+.-------------.+++.---------.++++
.+++++.--------.---[->++++<]>.------------.------[->+++<]>-.++++.++++++++++++.-[--->++<]>--.-----------
.---[->+++,<]>.++++++++++.++++++++++.-.--.+++++.--------.-[--->+<]>--.[-->++++++<]>-.---[--->+<]>-.-[-
++++++<]>-.>-[--->+<]>-.[---->++++++<]>-.---.--[--->+<]>-.+[0->++<]>+.+[--->+<]>.+++.++++++.+[->++++<]>
+.[->++++<]>++.<RET>the<SPACE>end.
```

[그림 28] 실행 결과

실행 결과, trudy에게 보내는 메시지와 함께 brainfuck으로 작성한 문자열을 확인할 수 있다.

```
+[-------==-->++<]>+.====++++[->+++<]>.--.-[--->+<]>----.-------------.[--->+<]>---.[---->+<]>+++.+[--
--->+<]>.--------..[--->+<]>---.-----------.+++++.--------.-[--->+<]>--.[-->+++++++<]>.----.----------
-.++.++.--[--->+<]>-.-[--->++<]>-.++++++++++.+[---->+<]>+++.+++++[->+++<]>.-.-[->++++++<]>-.---
[->++++<]>.------------.---.--[--->+<]>-.-----[->++<]>.---.{[--->++<]>--.+[->++<]>.++[----->+<]>+.--------
----.+++.--------.+++++.+++++.-------.---[->+++<]>.------------.-----[->++<]>-.++++.++++++++++++.-[-
-->++<]>--.------------.---[->+++,<]>.++++++++++.++++++++++.-.--.+++++.--------.-[--->+<]>--.[--
>++++++<]>-.--[--->+<]>-.-[->++++++<]>-.>-[--->+<]>-.[---->++++++<]>-.---.--[--->+<]>-.+[0-
>++<]>+.+[--->+<]>.+++.++++++.+[->+++<]>++.[->+++<]>++.
```

이를 실행하여 spy의 mission location에 대한 정보를 획득할 수 있었다.



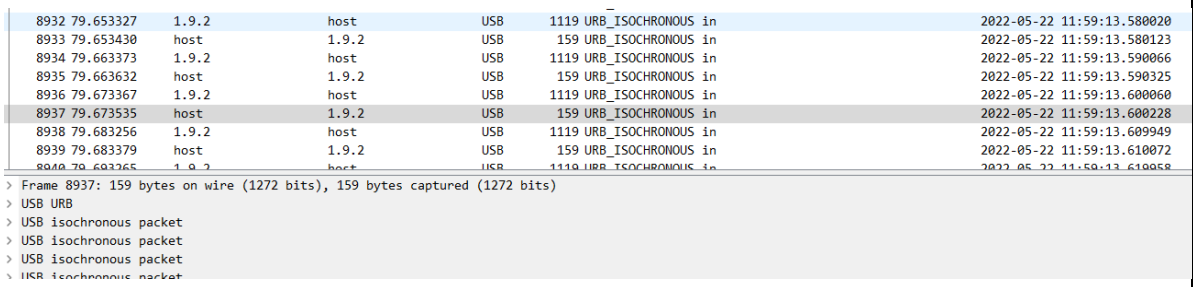
[그림 29] brainfuck 실행결과 (https://copy.sh/brainfuck/)

secret meeting place is on the 63 Building, 59F, Walking On The Cloud.

63빌딩 59층 워킹온더클라우드 레스토랑

3. Who is the spy targeting? (100 points)

3message 파일을 wireshark로 열어서 분석하였다.

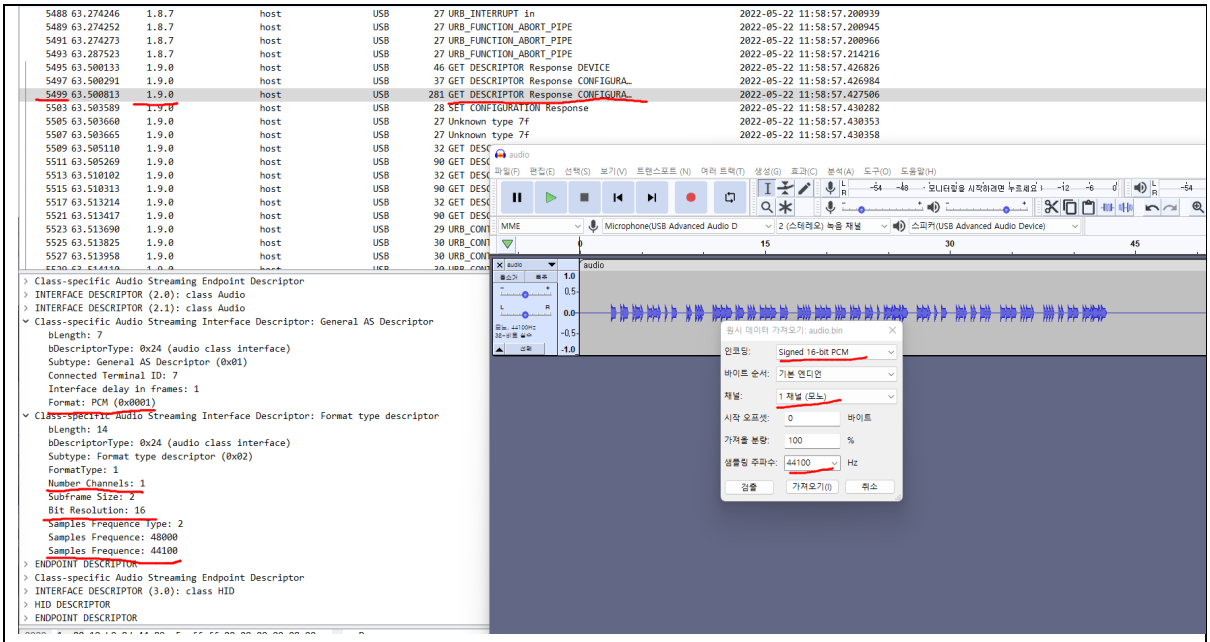


[그림 30] 3message 패킷 내용

1.9.2 오디오 장치와 통신하는 URB_ISOCHRONOUS in 패킷의 데이터를 아래 명령어를 사용하여 추출했다.

tshark -r ./3message.pcap -Y 'usb.iso.data' -T fields -e usb.iso.data | tr -d '\n',',' | xxd -r -ps > audio.bin

추출한 데이터는 raw audio data이며, audacity 프로그램으로 재생할 수 있다.
오디오 코덱 정보는 GET DESCRIPTOR Response CONFIGURATION 패킷에서 확인할 수 있다.

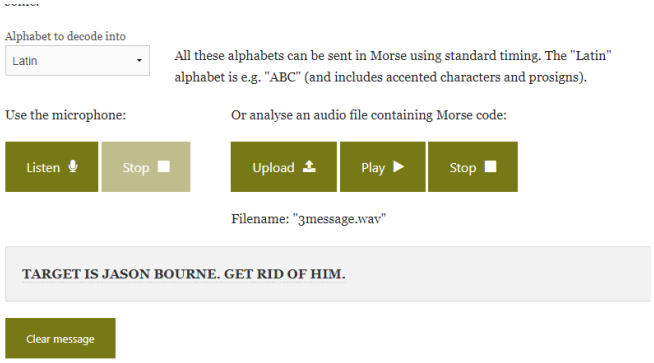


[그림 31] 오디오 코덱 정보 및 오디오 재생

인코딩	Signed 16-bit PCM
채널	1 Channel (mono)
샘플링 주파수	44100 Hz

[표 6] 오디오 코덱 정보

오디오 재생 시, morse code sound를 들을 수 있다.



[그림 32] morse code sound 해독 결과

온라인 해독 도구를 사용하여 해독한 결과는 **TARGET IS JASON BOURNE. GET RID OF HIM**으로, spy의 target은 **JASON BOURNE**이다.