HACK THE PACKET
문제풀이 보고서

닉네임	if
이름	김형선

보고서는 문제를 풀면서 접근했던 방법 및 결과 위주로 작성하였습니다.

이 모든 영광을 문제 만드시느라 고생하신
HackThePacket 운영진에게 바칩니다.

L01

Q 2012_htp_prequal.pcap 파일은 어떤 환경(System Information)에서 캡쳐한 것일까?

제공된 .pcap파일을 헥사에디터로 읽으면 매직헤더 뒤에.
특정 offset 내, 관련정보 존재
64-bit Windows 7 Service Pack 1, build 7601 이 나온다..

∵ key is "64-bit Windows 7 Service Pack 1, Build 7601"

L02

Q 2012_htp_prequal.pcap 파일은 어떤 도구로 캡쳐한 것일까? (대문자로 입력)

Capinfos[1] 라는 도구를 이용해 분석 실시
Type을 통해 해당 패킷을 캡쳐한 도구 확인 가능
스니퍼에서 패킷을 캡쳐할 때 고유한 매직넘버를 통해 구분하는 것 같다.
국내 패킷에선 pcap,pcapng,libpcap,snoop가 대표적인 듯

```
c:₩>capinfos 2012.pcap
File name:           2012.pcap
File type:           Wireshark - pcapng
File encapsulation:  Ethernet
Packet size limit:   file hdr: (not set)
Number of packets:   6241
File size:           3754728 bytes
Data size:           3543710 bytes
Capture duration:    1333 seconds
```

<그림 1-1 캡쳐한 트래픽에 대한 정보>

∵ Key is "Wireshark"

1) capinfos : Wireshark에서 제공해주는 도구로써, 캡쳐한 packet에 대한 상세 정보 제공

L1 (완료)

Q. ARP_Spoofing에 의해서 나의 아이디와 패스워드가 유출됐다!

EQ. ID and Password of mine were leaked by ARP Spoofing!

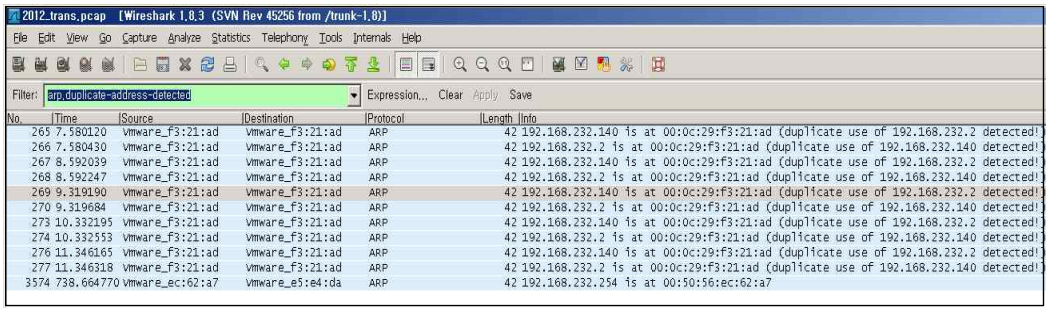
Arp spoof를 탐지하는 방법과,
유출이라는 단어로 봐서는 Arp-spoof 이후에 GET이나 POST를 보내주는 부분을 도중에 가로채지 않았을까라고 생각함.

L1-1) ARP Spoof

- Protocol Hierarchy 기능과 Display filter 이용

Protocol	% Packets	Packets	% Bytes	Bytes	Mbit/s	End Packets	End Bytes	En
⊟ Frame	100,00 %	6241	100,00 %	3543710	0,021	0	0	
⊟ Ethernet	100,00 %	6241	100,00 %	3543710	0,021	0	0	
⊞ Internet Protocol Version 6	6,97 %	435	2,50 %	88632	0,001	0	0	
Address Resolution Protocol	4,73 %	295	0,35 %	12444	0,000	295	12444	
⊟ Internet Protocol Version 4	88,06 %	5496	97,11 %	3441178	0,021	0	0	
Internet Control Message Protocol	0,13 %	8	0,03 %	912	0,000	8	912	
⊞ User Datagram Protocol	1,91 %	119	0,56 %	19831	0,000	0	0	
⊞ Transmission Control Protocol	86,03 %	5369	96,52 %	3420435	0,021	4039	2581276	
⊟ Configuration Test Protocol (loopback)	0,21 %	13	0,02 %	780	0,000	0	0	
Data	0,21 %	13	0,02 %	780	0,000	13	780	
⊟ Logical-Link Control	0,03 %	2	0,02 %	676	0,000	0	0	
Cisco Discovery Protocol	0,03 %	2	0,02 %	676	0,000	2	676	

ARP 프로토콜의 패킷은 전체 대비 295건 존재



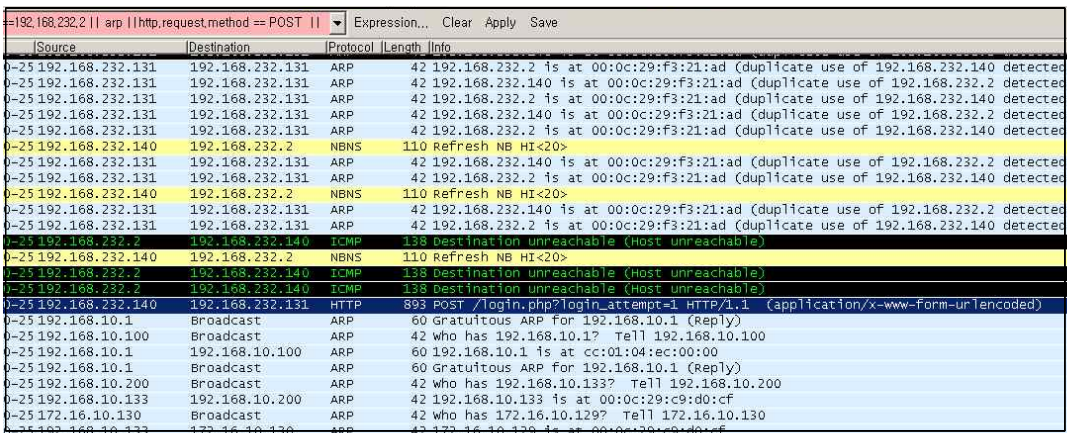
295건중 공격 패킷은 10건 존재

arp.duplicate-address-detected

```
Address Resolution Protocol (reply)
  Hardware type: Ethernet (1)
  Protocol type: IP (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: reply (2)
  Sender MAC address: Vmware_f3:21:ad (00:0c:29:f3:21:ad)
  Sender IP address: 192.168.232.2 (192.168.232.2)
  Target MAC address: Vmware_f3:21:ad (00:0c:29:f3:21:ad)
  Target IP address: 192.168.232.140 (192.168.232.140)
```

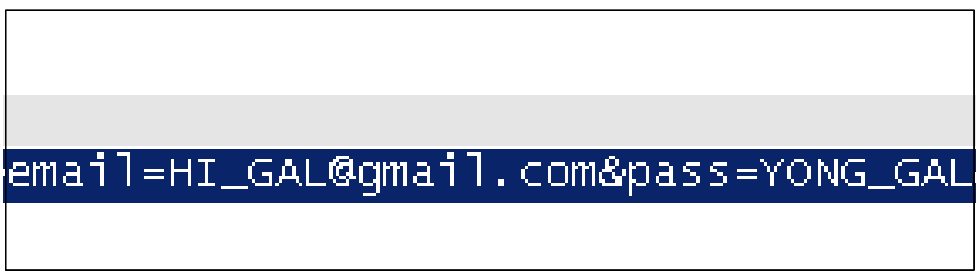
<목표 시스템 ARP Cache-table 변조시도>

Arp reply packet이므로 자동적으로 수신받는 호스트의 cache-table에 적용됨.
지속적인 wireshark의 alert 기능(duplicate IP or mac)을 통한 추가 검증



공격 타임라인

1) 00:15:55 : 정상상태
2) 00:16:00 ~ 00:16:03 : ARP Spoof(10회)
3) 00:17:14 : 목표의 SNS 로그인 시도로 인한 아이디 패스워드 탈취 성공



로그인 시도한 ID /PW

∵ Key is "00:0c:29:f3:21:ad_YONG_GAL"

L2 (완료)

Q. 남자들이 뼛속까지 좋아하는 여자는 누구? DNA 연구 결과가 발표 되었다. 바코드를 찾아라!

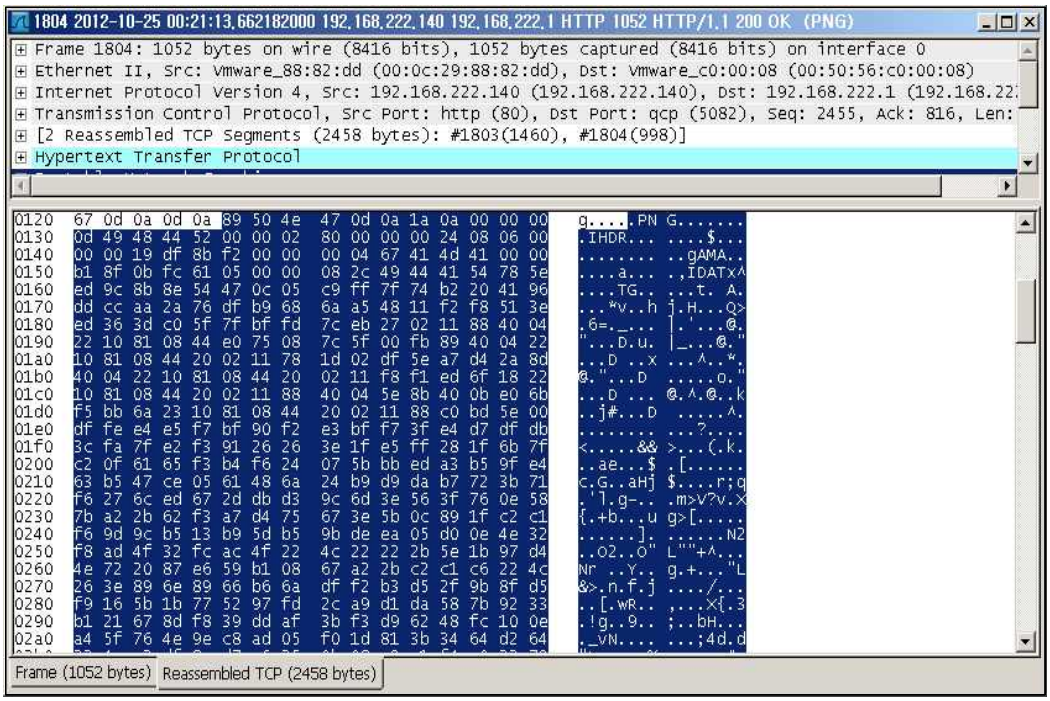
L2-1 DNA 와 Barcode라는 문자열로 스캔 실시

L2-2 DNA_MAP.jpg에 대한 http 요청 발견

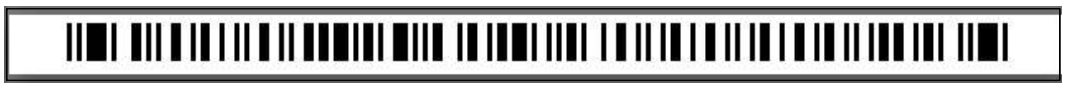
```
TCP       54 onscreen > http [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
TCP       54 http > onscreen [ACK] Seq=1 Ack=2 Win=18760 Len=0
TCP       62 qcp > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
TCP       62 http > qcp [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1460 SACK_PERM=1
TCP       54 qcp > http [ACK] Seq=1 Ack=1 Win=65535 Len=0
HTTP     439 GET / HTTP/1.1
TCP       54 http > qcp [ACK] Seq=1 Ack=386 Win=15544 Len=0
HTTP     1048 HTTP/1.1 200 OK  (text/html)
HTTP      484 GET /DNA_Map.jpg HTTP/1.1
TCP      1514 [TCP segment of a reassembled PDU]
HTTP     1052 HTTP/1.1 200 OK  (PNG)
TCP       54 qcp > http [ACK] Seq=816 Ack=3453 Win=65535 Len=0
```

DNA_MAP.jpg 요청

L2-3 요청에 대한 응답값에서 파일 추출 시도



L2-4 안드로이드 기반 바코드 스캐너로 스캔 실시



< 추출한 DNA_MAP.jpg >

∵ Key is “Key is IU Good”

L4 Q. 우탱아, 가을인데 단풍놀이 가야지~ 어디로 갈까?

L4-1 문자열 검색(Key Word:where, autumn, maple, wotang,

192.168.222.140	HTTP	769 GET / HTTP/1.1
192.168.222.1	TCP	54 http > pptp [ACK] Seq=1 Ack=716 Win=15730 Len=0
192.168.222.1	HTTP	264 HTTP/1.1 304 Not Modified
192.168.222.140	HTTP	557 GET /where_is_it.jpg HTTP/1.1
192.168.222.1	HTTP	243 HTTP/1.1 304 Not Modified
192.168.222.140	HTTP	404 GET /favicon.ico HTTP/1.1
192.168.222.1	HTTP	557 HTTP/1.1 404 Not Found (text/html)
192.168.222.140	TCP	54 pptp > http [ACK] Seq=1569 Ack=903 Win=64633 Len=0

L4-2 index.html 내 링크되어 있는 Where_is_it.jpg 발견

Where_is_it.jpg

L4-3 사진 내 GPS 좌표정보 찾기

-> 실패, 없는걸로 판단

L4-4 WinHex로 해당 그림파일 오픈 후 헥사값에 매칭된 문자열 분석

* A8부터 0x000051A8 offset부터 0x000051B7 까지 Key 값 저장 됨.

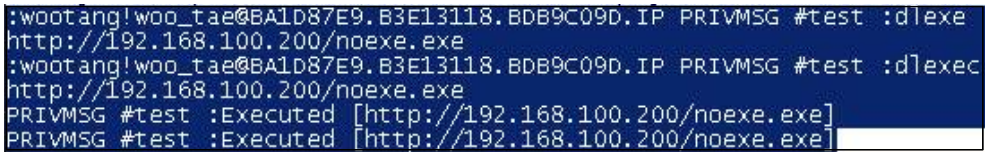
∵ Key is “K@e*y_:_hallasan”

L5 Q 악성 다운로더

L5-1 IRC 관련 통신 발견

-> 원격 지령 의심 통신 식별,

-> 192.168.100.200:80의 noexe.exe를 받는걸로 추측



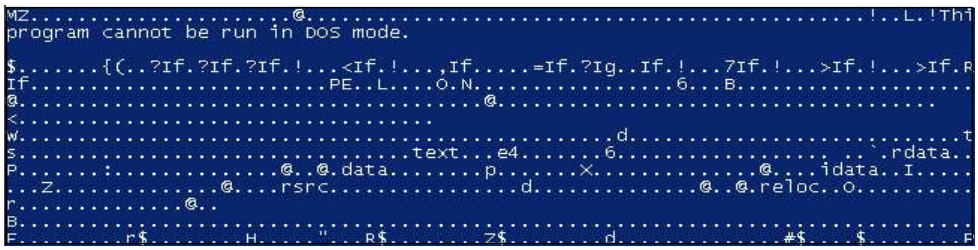
의심 IRC 채팅

L5-2 관련 내용 세부 검색

GET /no.exe

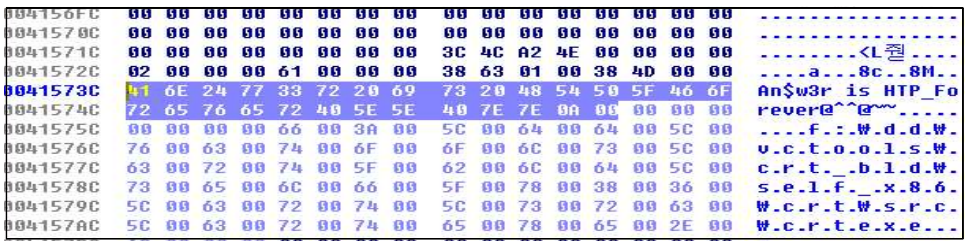
-> http 접근 후 파일 다운로드 흔적[2] 발견

L5-3 헥사 에디터 이용, 패킷 내 의심 파일 추출



추출할 실행파일 구조

L5-4 프로그램 內 문자열 중 키값 존재



키 값

∵ Key is “An$w3r is HTP_Forever@^^@~~”

2) 악성 의심 프로그램 다운 흔적 : http://192.168.100.200/noexe.exe

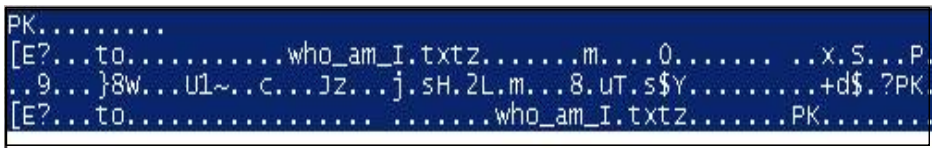
M1 Q. 나는 누구인가? 네오는 오라클에게 FTP로 Zip 파일을 받게 되는데....

M1-1 FTP 프로토콜 분석

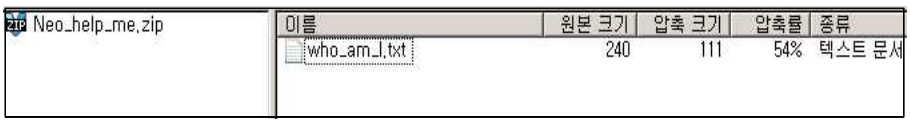
-> 네오 관련 파일 식별(Neo_help_me.zip)

ftp를 이용한 파일 수신

M1-2 Neo_help_me.zip 파일 추출 및 분석



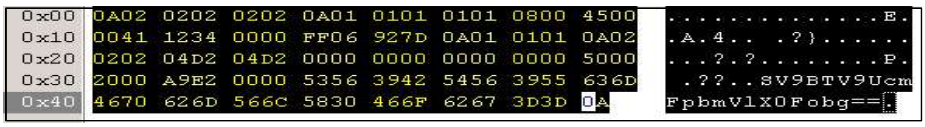
ZIP 추정 파일 수신 내용



복원한 ZIP 내부 압축파일

복원한 ZIP 내부에 존재하는 TXT문서 파일 내용

M1-3 Who_am_I.txt 內 코드 복호화 시도



헥사값 변환

-> Base64 인코딩 의심 문자열 추출

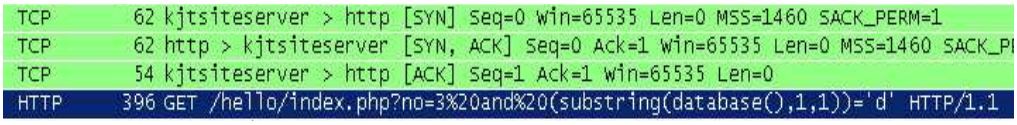
M1-1 의심 문자열 base64 디코딩 시도

코드 內 숨겨진 키값

∵ Key is “I_AM_Trainee_Ahn”

M2 Q. DB이름을 찾아라!

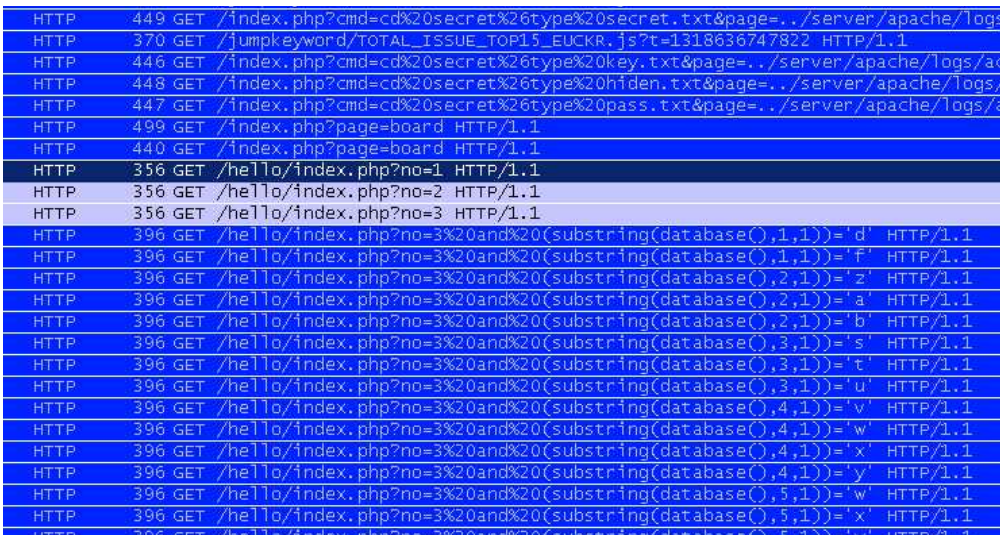
M2-1 Datebase란 문자열로 필터링



의심 요청 발견

-> http://192.168.232.1/hello/index.php?no=3%20and%20(substring(database(),1,1))='d'

M2-2 요청값과 응답값 분석을 통해 Blind sql injection 공격으로 추정



공격 추정 요청

-> 참과 거짓 값에 따라 응답값이 변환,
요청 문자가 db의 배열 위치의 값과 동일하다면
* 아이유 반환!

M2-3 결과값이 참인 응답패킷으로 문자열 조합

∵ Key is “easywebsiteattack”

M3 Q 라우터에 백도어가 삽입되어 있다. 마지막으로 실행된 명령어는?

M3-1 TCP 프로토콜을 이용해 라우터로 의심되는 장비간 통신 식별

M3-2 TCLShell 백도어를 이용해 접근 후 4가지 명령어 실행

```
1) sh run
2) enable
3) conf ter
4) hostname An$w3r_is^tcls
```

백도어 접근후 실행한 명령어 리스트

∵ Key is "An$w3r_is^tcls"

M5 Q 메일 사용자계정과 패스워드가 IRC 봇에 감염되어 유출됐다.

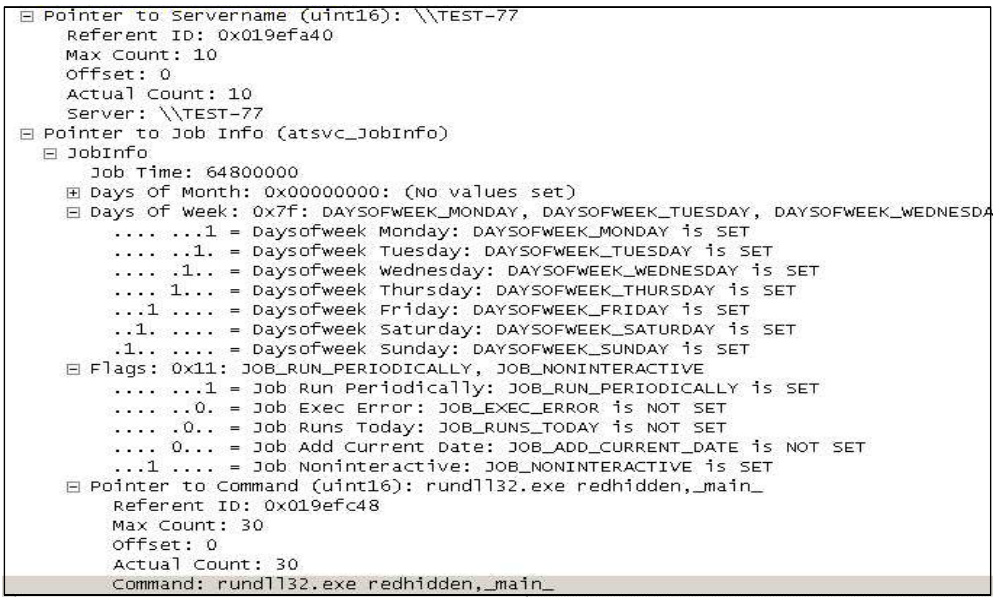
M5-1 IRC 통신에 사용되는 문자열로 필터 실시

식별된 IRC 통신 內 키값 존재

∵ Key is "K~e!y:good_bye_jobs"

M7 Q : chakyi는 원격지 시스템에 매일 일정한 시간에 프로그램이 자동으로 실행되도록 만들었다. chakyi가 실행시키고자 하는 명령을 찾아라.

M7-1 트래픽 內 ATSVC[3]를 이용한 예약 명령 실행 식별



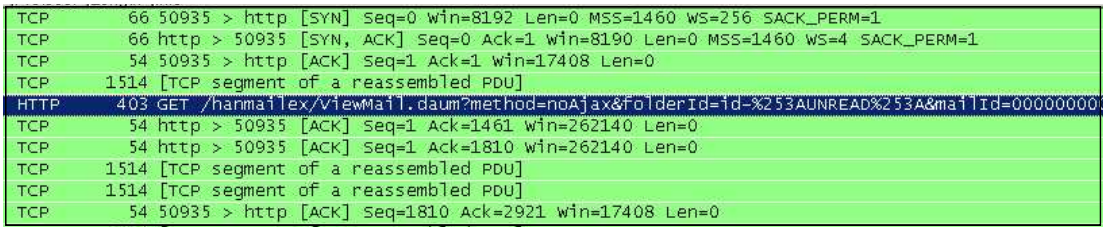
ATSVC를 이용한 예약 명령 적용

∵ Key is "rundll32.exe redhidden,_main_"

3) ATSVC : Microsoft AT-Scheduler Service 를 이용해 네트워크를 이용한 예약작업 등록 가능

H1 Q. 이메일을 통해 jitae 의 첫번째 데이트 기밀정보를 입수하는데...
하지만 내용없이 파일만 첨부되어 있었다. 데이트 장소는 언제 몇시에 어디인가?

H1-1 관련 패킷 추출(Keyword : mail, jitae, date, where, attach)



의심 메일 발견 (Daum)

H1-2 통신내역 분석 및 추출

추출한 메일

-> FIRST DATE란 제목으로 박정우씨가 누군가에게 보낸 메일 추출

H1-3 복구한 메일 내용에 첨부된 promise.mp3 파일 추출

첨부된 MP3 통신과정

H1-4 추출한 mp3 재생

추출된 promise.mp3

-> 헥사에디터 분석했으나, 키값 미존재..
-> "개똥이네 버블버블" 여기안에 키가 숨겨져 있을것이라고 추측
* try keyword : buble, bublebuble, gaetong,.....

H1-5 종료 4분전까지 갈피를 못잡다가.. 지문을 보고 키값 식별..-_-;;
-> Key is Jitae

```
Input file = 'promise.mp3'  output file = 'promise.mp3.pcm'
Will attempt to extract hidden information. Output: promise.mp3.txt
Enter a passphrase: *****
Confirm your passphrase: *****
the bit stream file promise.mp3 is a BINARY file
HDR: s=FFF, id=1, l=3, ep=off, br=9, sf=0, pd=1, pr=0, m=0, js=0, c=0, o=0, e=0
alg.=MPEG-1, layer=III, tot bitrate=128, sfrq=44.1
mode=stereo, sblim=32, jsbd=32, ch=2
[Frame  255]Avg slots/frame = 416.334; b/smp = 2.89; br = 127.502 kbps
Decoding of "promise.mp3" is finished
The decoded PCM output file name is "promise.mp3.pcm"
```

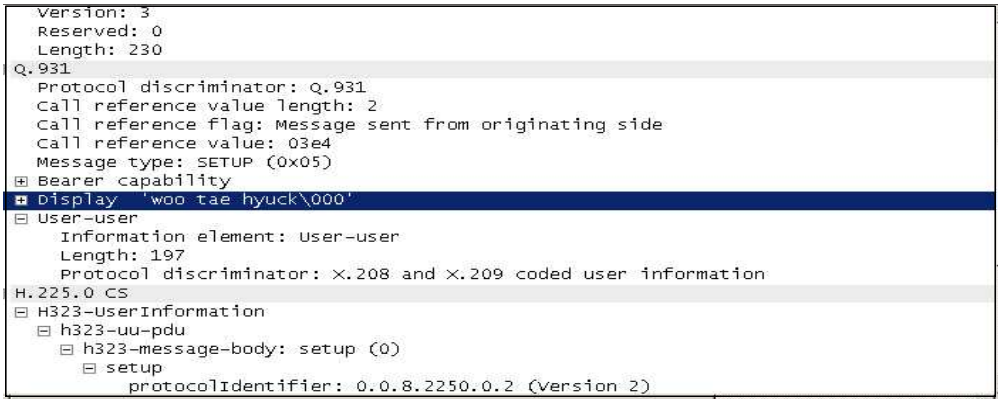
스테가노 그래피 도구를 이용, 숨겨진 데이터 추출

추출된 숨겨진 값

∵ Key is "sunday, 14:00, GangNam station"

H3 Q. 우태혁의 여자친구 이름은 무엇이고, 어디에 살고 있는가?

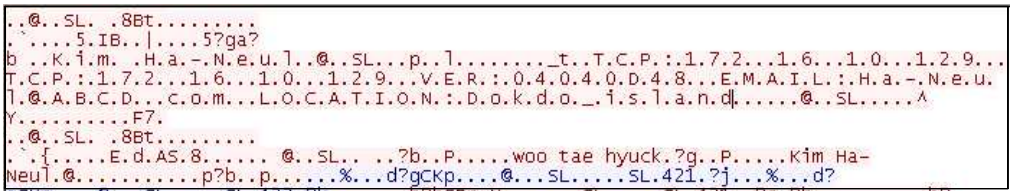
H3-1 태혁에 관련된 통신 정보 수집



태혁 관련 정보

-> 해당 시점을 기점으로 분석 시작

H3-2 해당 통신 영역 근처의 TCP 통신에서 태혁과 여자친구 의심 통신내용 추출



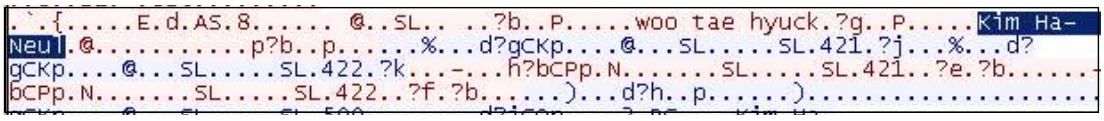
추출 내용

H3-2 의미있는값 변환

TCP:172.16.10.129
VER:04040D48
EMAIL:Ha-Neul@ABCDcom
LOCATION:Dokdo_island

1차 추출값

-> 위치는 독도이며, 태혁의 여자친구의 이름은 하늘인것을 확인!



2차 추출 값

-> 좀더 진행하면 여자친구의 성은 김인것을 확인할수 있다.

∵ Key is “Kim Ha Neul_Dokdo island”