

제 1회 청소년 화이트해커 경진대회



2위 hkkiw0823(강인욱)

<http://hkkiw0823.pe.kr>

1번

1. 20A9(16진수)와 1100111111(2진수)의 합을 10진수로 나타내시오
2. 10011000과 00110101의 xor 연산을 하고 10진수로 나타내시오
3. N e w H e a r t

각각의문자하나를ascii코드값의 10진수 합으로 나타내면?

1, 2, 3번 키를 붙여서 인증

```
key1 = 0x20a9 + 0b1100111111
```

```
key2 = 0b10011000 ^ 0b00110101
```

```
key3 = "NewHeart"
```

```
key3_s = 0
```

```
for i in key3:
```

```
    key3_s += ord(i)
```

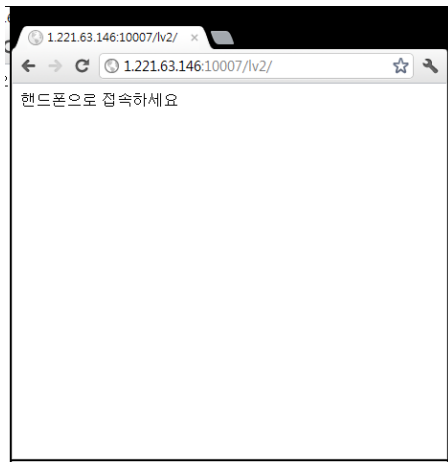
```
printstr(key1)+str(key2)+str(key3_s)
```

키 :9192173798

2번

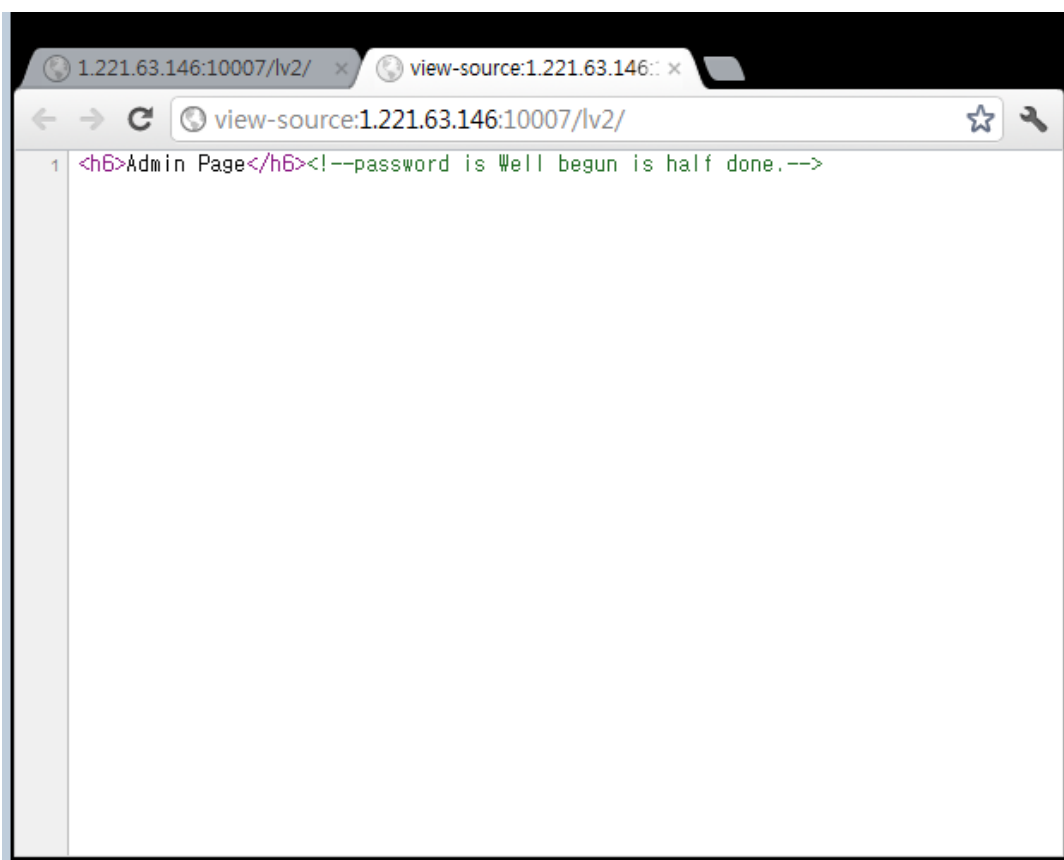
링크 : <http://1.221.63.146:10007/lv2/>

문제 페이지로 들어가보면 핸드폰으로 접속하세요 라는 문구가 뜬다.



User-Agent를 아이폰 용으로 바꾸어서 들어가보았다.

[Mozilla/5.0 (iPhone; U; CPU iPhone OS 4_0 like Mac OS X; ko-kr) AppleWebKit/532.9 (KHTML, like Gecko) Version/4.0.5 Mobile/8A293 Safari/6531.22.7]



키 :Well begun is half done.

3번

3 누군가 뉴하트 홈페이지의 로고에 비밀 키를 숨겨놓았다..

비밀 키를 찾아라.

(원본 로고 : http://1.221.63.146:10007/lv3/nh_header.bmp)

(변조된 파일 : http://1.221.63.146:10007/lv3/nh_header_prob.bmp)

```
if __name__ == "__main__":
    img1 = open("nh_header.bmp", "rb")
    img2 = open("nh_header_prob.bmp", "rb")

    img1_buf = img1.read()
    img2_buf = img2.read()

    key = ""
    for i in range(0, len(img1_buf)):
        if img1_buf[i] != img2_buf[i]:
            key += img2_buf[i]

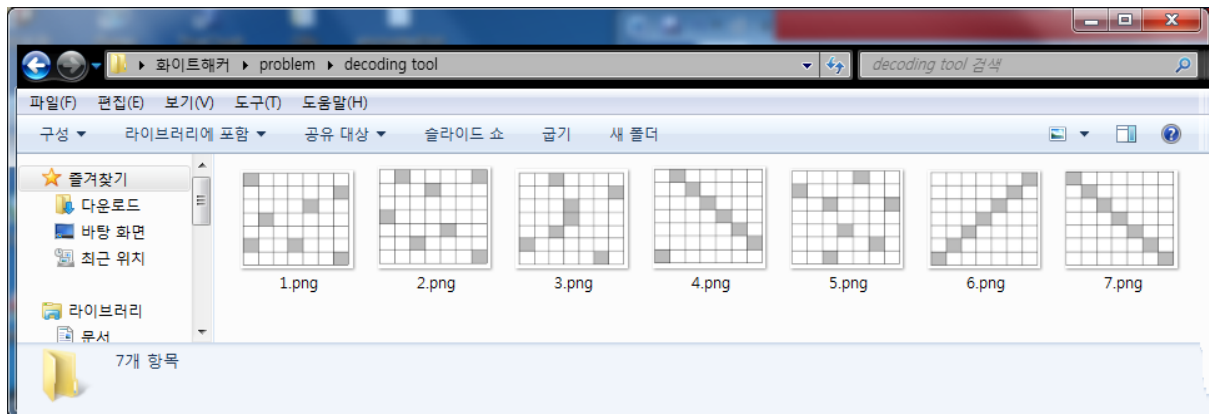
    print key
```

키 :newhe@rt!!

4번

4 <http://1.221.63.146:10007/lv4/problem.zip>

문제 파일을 받아보면 이미지 파일을 7개 준다.



그리고 thisiskey.txt 라는 파일도 주어졌다.

```
<decodebase85~aa$dEtf+s+x&&hj+df#kgl*sfgmldvbn@d/
~GiagsAQT#hhsso@tr;TwM$Y(le^ojs^4dawh?&ftzlwrgplY
)hOes!d*jggtmR^^zgngfdlxBdb~!!Op@cxuat}svf)vmst!z
1ropdtr&*Dity3c)3duaev*cvRtsb&4zt0dnads8hd@lk^jad
1879rwe#d$#ytr./dsudr>m^&ifg?bnBoahcv&(p4jxz#*lkt
d@#sda7zxcvb^^8o$aF1%7hg*23elj)hjkkl,dfbfp>mnbn,mo
@aytkotz;r((udx&I*dtkqae)0tl%^xr18;wvcvd~jbnzdfg>
```

thisiskey.txt

해당 글자들을 저 이미지에 나오는 것처럼 7자씩 끊어 총 7개를 만들었다.

그리고 각각 그림에 색칠 되있는 것과 대칭시켜 글자를 뽑아내었다.

그리고 글자들을 모아 bas85로 디코딩하면password_is_hello_hacking_festival!! 가 나온다.

<decode

base85~

aa\$dEtf

s+x&&hj

+df#kgl

*sfgmld

vbn@d/

<~E+*g/

~GiagsA

QT#hhss

o@tr;Tw

M\$Y(le^

ojs^4da

wh?&ftz

lwrgplY

GAhM4?Y

)hOes!d

*jgttmR

^^zgngf

dlxBdb~

!!Op@cx

uat}svf

)vmst!z

ORgBOu!

1ropdtr

&*Dity3

c)3duae

v*cvRts

b&4zt0d

nads8hd

@lk^jad

rDdR0d@

1879rwe

#d\$#ytr

./dsudr

>m^&ifg

?bnBoah

cv&(p4j
xz#*lkt

r#drB4#

d@#sda7
zxcvb^^
8o\$aF1%
7hg*23e
lj)hkl
k,dfbfp
>mnb,mo

dz\$%hfn
7^F*),>

@aytkot
z;r((ud
x&I*dtk
qae)0tl
%^xr18;
wvcvd~j
bnzdfg>

@;l)1~>

키 :password_is_hello_hacking_festival!!

5번

5 <http://1.221.63.146:10007/lv5/h4ck.apk>

해당 apk를 디컴파일하면 중간에 수상한 문자열과 decrypt 함수가 보인다. 해당 부분을 파이썬코드로 바꿔보았다.

```
def decrypt(string):
```

```
keyTable1 =  
[1,45,34,24,17,11,18,21,6,26,16,7,43,2,8,3,36,22,10,28,9,13,14,20,5,41,25,37,29,31,44,30,35,23,38,40,33,  
4,27,19,15,12,42,32,39]
```

```
keyTable2 = "abcdefghijklmnopqrstuvwxyz.1234567890~_:%+=?";
```

```
b = 0
```

```
while 1:
```

```
for i in range(0,45):
```

```
if string[b] == keyTable2[i]:
```

```
for j in range(0,45):
```

```
if keyTable1[j] == i:
```

```
print keyTable2[j],
```

```
    b+=1
```

```
if __name__ == "__main__":
```

```
decrypt("ygbahi?+hih5vrhhsb1r")
```

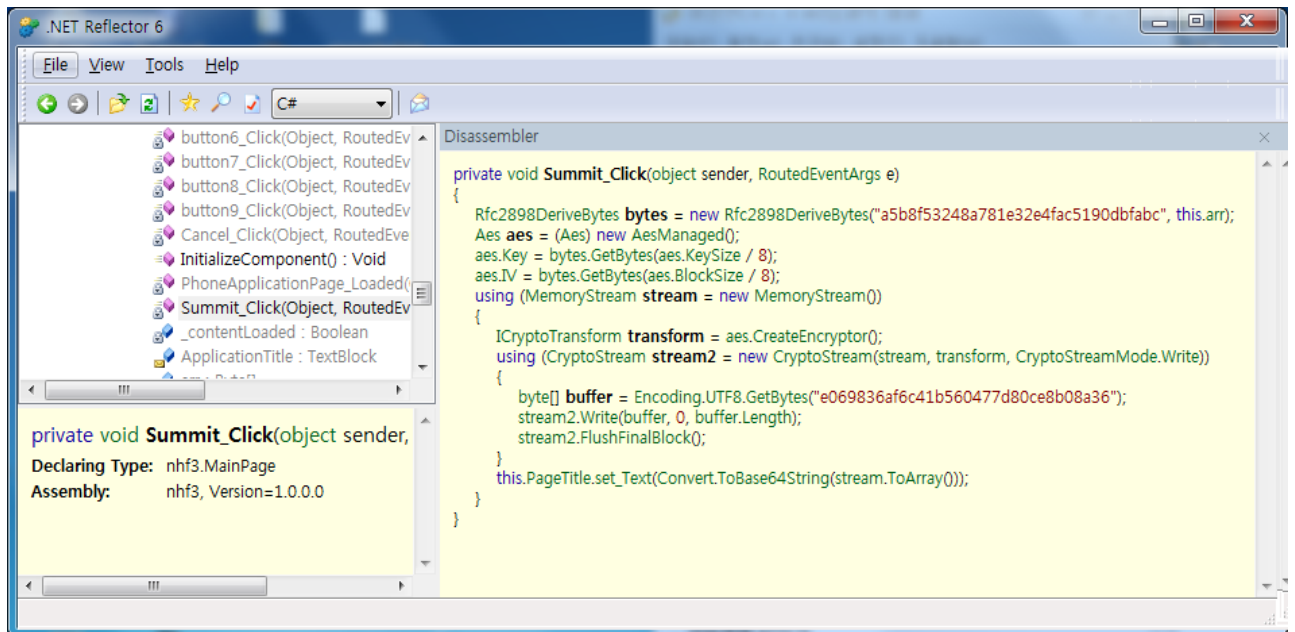
```
키 :dialo3+lol=hellgate
```

6번

6 1.221.63.146:10007/lv6/nhf3.xap

xap파일 압축을 풀고 보면 nhf3.dll 이라는 뭔가 답이 안에 있을꺼 같은 dll이 주어졌다.

해당 dll을 리플렉터로 분석해보면 전송 버튼 핸들러에 무언가 수상한 소스가 있다. 자세히 보면 AES 암호화하는 소스인데 9개의 버튼을 어떠한 순서대로 눌렀을 때 Rfc2898DeriveBytes 함수 두 번째 인자인 arr 배열에 버튼이 눌러진 순서를 토대로 어떠한 특정한 값들이 들어가게 된다.



WMAppManifest.xml 에 보면 Description에 버튼을 누르는 순서가 적혀있다. 원폰 에뮬레이터로 돌리면 키가 잘려서 나와 C# 코드를 작성하였다.

버튼을 누르는 순서 134628957

버튼에 해당하는 값 364157982 (1번은 3 2번은 6 3번은 4 이런식입니다.)

조합 해보면 배열 값은 0 2 8 4 3 6 1 5 9 7 이 나온다.

```
using System;
using System.Security.Cryptography;
using System.Text;
using System.IO;

namespace hkkw0823
{
    class HelloWorld
    {
        public static void Main()
        {

```

```

byte[] hkk_salt = newbyte[10];
hkk_salt[0] = 0;
hkk_salt[1] = 2;
hkk_salt[2] = 8;
hkk_salt[3] = 4;
hkk_salt[4] = 3;
hkk_salt[5] = 6;
hkk_salt[6] = 1;
hkk_salt[7] = 5;
hkk_salt[8] = 9;
hkk_salt[9] = 7;

Rfc2898DeriveBytes bytes2 = newRfc2898DeriveBytes("a5b8f53248a781e32e4fac5190dbfabc", hkk_salt);
Aesaes = (Aes)newAesManaged();
aes.Key = bytes2.GetBytes(aes.KeySize / 8);
aes.IV = bytes2.GetBytes(aes.BlockSize / 8);
using (MemoryStream stream = newMemoryStream())
{
    ICryptoTransform transform = aes.CreateEncryptor();
    using (CryptoStream stream2 = newCryptoStream(stream, transform, CryptoStreamMode.Write))
    {
        byte[] buffer = Encoding.UTF8.GetBytes("e069836af6c41b560477d80ce8b08a36");
        stream2.Write(buffer, 0, buffer.Length);
        stream2.FlushFinalBlock();
    }
    System.Console.WriteLine(Convert.ToBase64String(stream.ToArray()));
}

} //main

} //class
} //namespace

```

7번

7 <http://165.246.149.60/quiz/>

(문제 페이지에 MYSQL 연결 오류가 나서 글로 적겠습니다.)

문제 페이지에 들어가면 30분마다 한번씩 돌리는 경품 추천 원판이 나오게 된다.

원판에는 1점 2점 3점 팡 등이 있었고 원판을 돌리면 점수를 얻는 형식이였다.

그리고 모은 포인트로 상품을 구매하는 페이지가 있었는데, 30점??이 Hint1 이였고 50점이 Hint2

1000점이 Answer(키) 였다.

이 문제를 풀기 위해 자바스크립트를 분석하였고 점수를 올리는 자바스크립트 소스를 찾았다.

```
time = 1000000000000;
```

```
functioncontrolUserAccess(text){
```

```
    if(text=="BANG"){
```

```
        //alert("아쉽습니다. 다음 기회를 노리세요");
```

```
        httpRequest2_1=getXMLHttpRequest();
```

```
        httpRequest2_1.open("POST", "./process.php", true);
```

```
        httpRequest2_1.setRequestHeader("Content-Type",      "application/x-www-form-  
urlencoded");
```

```
        var origin="timestamp=";
```

```
        origin+=now.getTime();
```

```
        origin+="-point=0";
```

```
        var query1=encodeURIComponent(Aes.Ctr.encrypt(origin, "Busker Busker", 256));
```

```
        httpRequest2_1.send("data="+query1);
```

```
        controlSpin();
```

```
    }else{
```

```
        httpRequest2_2=getXMLHttpRequest();
```

```
        httpRequest2_2.open("POST", "./process.php", true);
```

```
        httpRequest2_2.setRequestHeader("Content-Type",      "application/x-www-form-  
urlencoded");
```

```

var origin2="timestamp=";

origin2 += now.getTime()+time;

origin2+="-point=";

origin2+=3;

var query2=encodeURIComponent(Aes.Ctr.encrypt(origin2, "Busker Busker", 256));

httpRequest2_2.send("data="+query2);

time += 1000000000000;

controlSpin();

//alert("축하합니다. "+text+"가 당첨되었습니다.");

}

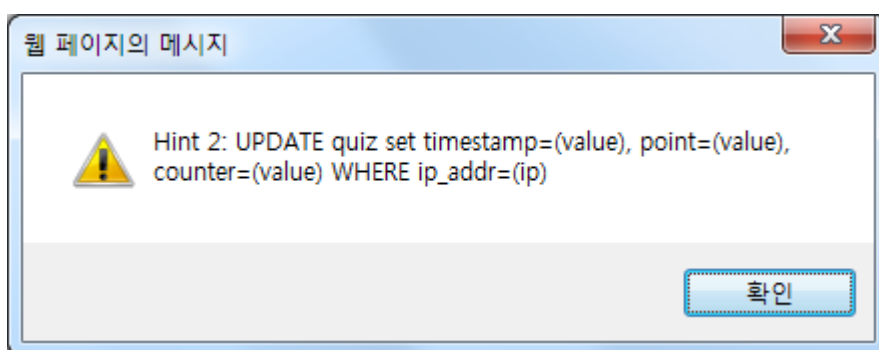
//1000000000000000, point=1000, counter=0 where ip_addr=0x3137352e32313332e33362e313138#

//NewHeartBeat

}

```

시간체크함수를 우회하고 controlSpin함수를 계속 호출하여 점수를 계속 올리게끔 하고 한 5~10분쯤 지나니 50점이 차서 Hint2를 구매하였다.



Sql Injection인 것을 파악하고

포인트는 조작이 불가능하여 timestamp에

1000000000000000, point=1000, counter=0 where ip_addr=0x3137352e32313332e33362e313138#

해당 값을 넣어 인증하였다.

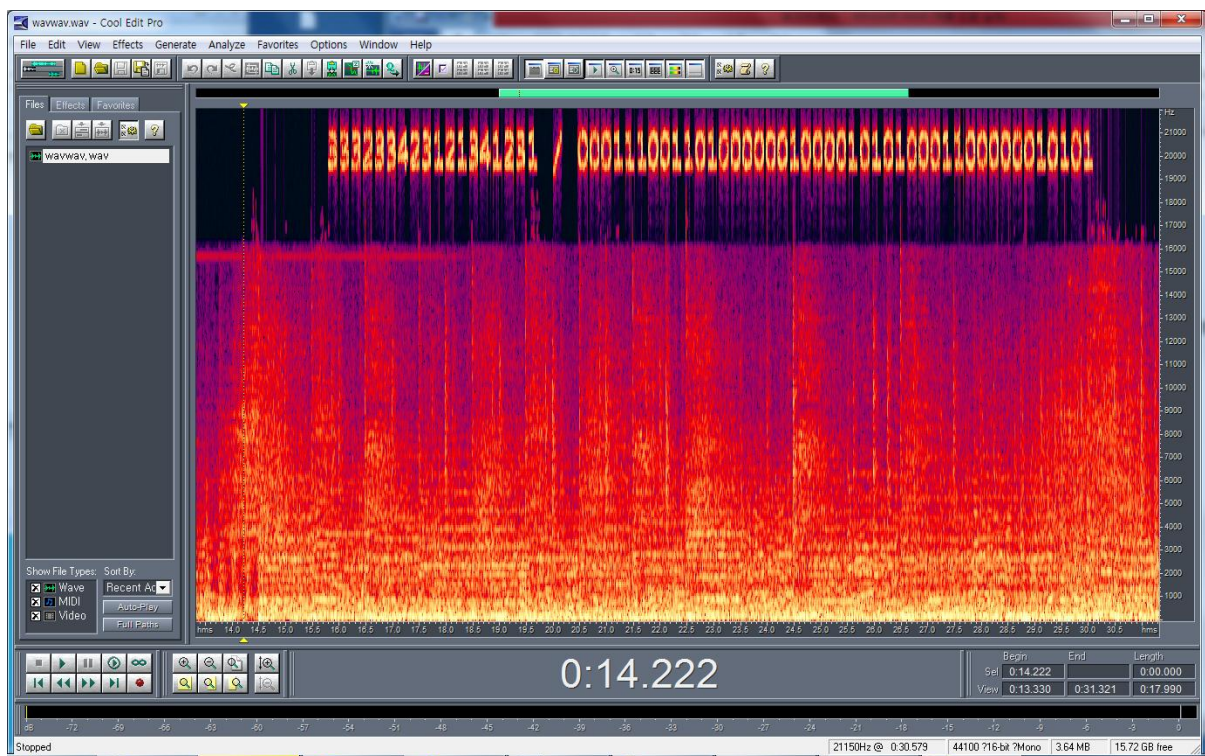
키 :NewHeartBeat

8번

8 <http://1.221.63.146:10007/lv8/wavwav.wav>

음악파일이 주어져 실행하였더니 뽐뽐뽐뽐뽐뽐뽐 ㅏ 뽐마뽐마뽐마뽐ㅏ 거리는 웅장한 소리가 들렸고, 사람의 머리속을 맑고 경쾌하게 해주어 잠시나마 휴식을 취할 수 있는 노래소리가 울려 퍼졌다.

각설하고, 파일안에숨겨진게있나해서둘러보았지만 아무것도 없어서 쿨에디트로음악파일의 스펙트럼을 보았다.



오른쪽에 있는 값이 모스부호 값이였고 왼쪽에 있는 값이 띄어쓰기였다.

333233423121341231 /

000 111 001 10 100

000 0100 00 101 0 10

0 011 0000 0 01 010 1

이렇게 정리해서 다시 해독해보면

SOUNDSLIKENEWHEART 가 나온다.

키 :soundlikenewheart

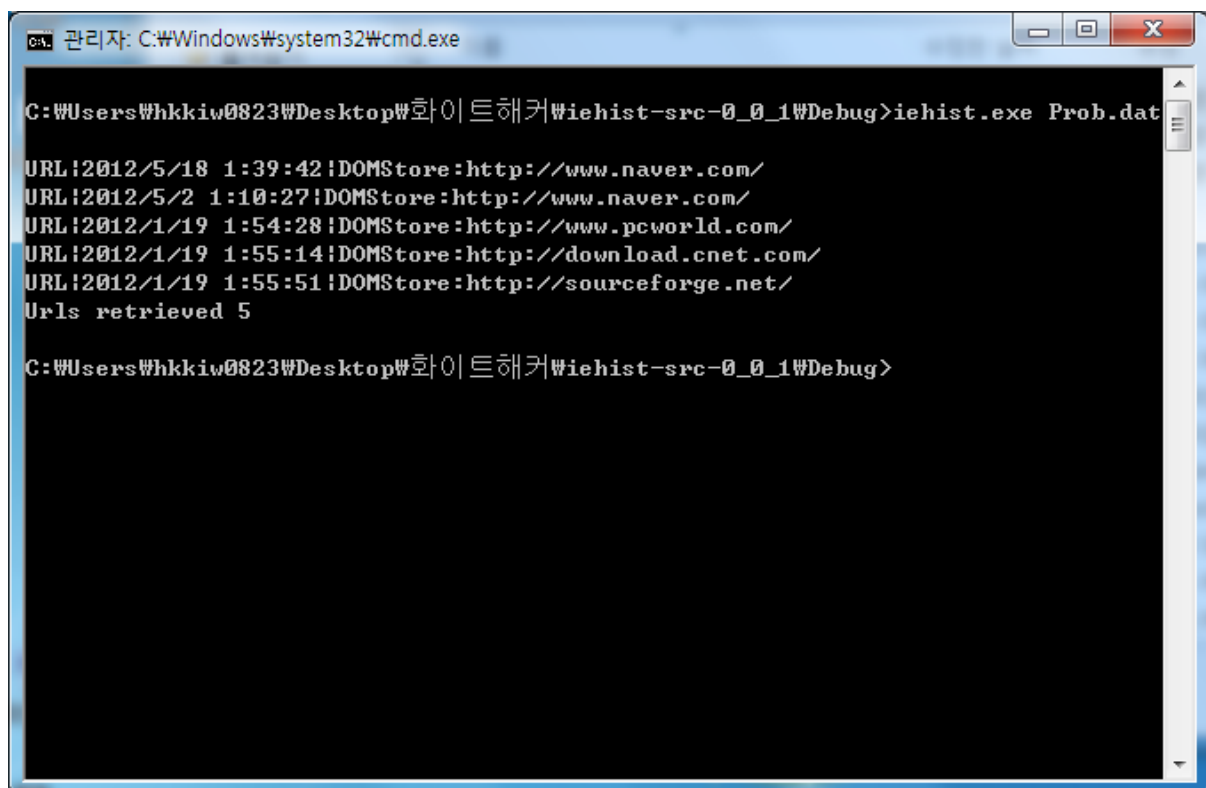
9번

9 Download Date ?

(YYYYMMddHHmmss)

<http://1.221.63.146:10007/lv9/Prob.zip>

해당 문제파일을 검색해보니 IE COOKIE관련 정보를 저장하고 있는 index.dat와 관련이 있다고 생각하여 해당 정보를 보는 iehist를 사용해 풀었다.



```
관리자: C:\Windows\system32\cmd.exe

C:\Users\Whkkiw0823\Desktop\화이트해커\iehist-src-0_0_1\Debug>iehist.exe Prob.dat

URL:2012/5/18 1:39:42!DOMStore:http://www.naver.com/
URL:2012/5/2 1:10:27!DOMStore:http://www.naver.com/
URL:2012/1/19 1:54:28!DOMStore:http://www.pcworld.com/
URL:2012/1/19 1:55:14!DOMStore:http://download.cnet.com/
URL:2012/1/19 1:55:51!DOMStore:http://sourceforge.net/
Urls retrieved 5

C:\Users\Whkkiw0823\Desktop\화이트해커\iehist-src-0_0_1\Debug>
```

아마 내기억으로는 download.cnet.com에서 받은 날짜였던거 같다.

키는 20120119015514에 GMT+9를 한값인 20120119105514

키 : 20120119105514

10번

10 <http://1.221.63.146:10007/lv10/android.zip>

NewHeart 수사대는 어떤 사건을 수사하던 중 마약 사건에 관련된 범인을 체포하였다. 범인은 마약을 밀거래하는 사람으로 특정일 특정장소에서 밀거래상과 접선할 예정이었다는 점을 자백하였으나 수사대는 더 이상의 자세한 내용은 밝혀내지 못했다. 유일한 단서는 범인이 가지고 있던 스마트폰으로, 암거래상과 정보를 주고 받았을 가능성이 높다. 암거래상과의 접선 장소 및 시간을 찾아라.

문제파일은 안드로이드 파일시스템을 전부다 덤프하여 제공해주었다. 문제에서 암거래상과 무언가를 주고 받았다고 해서 바로 문자메세지를 떠올렸고, 문자 메시지가 있는 경로인

android\data\data\com.android.providers.telephony를 뒤적뒤적거리PART_13372276170이라는 jpg 파일을 찾았다.

IU_CONCERT_1800_PM_JUNE_02_2012

키 : IU_CONCERT_1800_PM_JUNE_02_2012

11번

11 <http://1.221.63.146:10007/lv11/newheart.sys> VM상에서 실행하셔야 합니다!

해당 sys 파일을 IDA로 열어 분석하였다. 분석해보면 ZwCreateFile로 파일을 만들고 key를 생성한 후 ZwWriteFile로 키를 저장하는 형식이었는데, vm에서 실행하니 블루스크린이 계속 떠서 key gen 부분만 따와서 코딩하였다.

```
char *__stdcall key(const char *a1)
{
    int v1; // kr00_4@1
    char v3; // [sp+18h] [bp-30h]@1
    char v4; // [sp+19h] [bp-2Fh]@1
    ULONG_PTR v5; // [sp+40h] [bp-8h]@1
    int i; // [sp+44h] [bp-4h]@1
    int v7; // [sp+48h] [bp+0h]@1

    v5 = (unsigned int)&v7 ^ BugCheckParameter2;
    v1 = strlen(a1);
    v3 = 0;
    memset(&v4, 0, 0x27u);
    keyTable(v1);
    for ( i = 0; i < v1; ++i )
        *(&v3 + i) = byte_130A0[i] ^ a1[i];
    for ( i = 0; i < 40; ++i )
        byte_130E0[i] = *(&v3 + i);
    return &v3;
}

void *__stdcall keyTable(int a1)
{
    void *result; // eax@1
    bool v2; // [sp+0h] [bp-40h]@12
    signed int v3; // [sp+4h] [bp-3Ch]@7
    char v4; // [sp+8h] [bp-38h]@1
    char v5; // [sp+9h] [bp-37h]@1
    ULONG_PTR v6; // [sp+30h] [bp-10h]@1
    int i; // [sp+34h] [bp-Ch]@1
    double v8; // [sp+38h] [bp-8h]@1
    int v9; // [sp+40h] [bp+0h]@1

    v6 = (unsigned int)&v9 ^ BugCheckParameter2;
    v8 = dbl_13000;
    v4 = 0;
    result = memset(&v5, 0, 0x27u);
    for ( i = 0; i < 10000; ++i )
        v8 = 4.0 * v8 * (1.0 - v8);
    for ( i = 0; i < 24; ++i )
    {
        result = (void *)i;
        *(&v4 + i) = 0;
    }
    v3 = -1;
    for ( i = 0; i < 8 * a1; ++i )
    {
        if ( !(i % 8) )
            ++v3;
        *(&v4 + v3) *= 2;
        v2 = v8 > 0.5;
        result = (void *) (v2 | *(&v4 + v3));
        *(&v4 + v3) = (char)result;
        v8 = 4.0 * v8 * (1.0 - v8);
    }
    for ( i = 0; i < 40; ++i )
    {
        result = (void *)i;
        byte_130A0[i] = *(&v4 + i);
    }
    return result;
}
```

```
#include<stdio.h>
#include<string.h>
```



```

char *whatitis =
"WxD2Wx04Wx21Wx4BWx38WxFDWx15WxAAWxBEWx3EWxD4Wx6DWx93WxE7WxF3Wx87Wx03Wx8EWxB4Wx48Wx29Wx92WxDFWx
ABWxCDWxFDWxFDWxFDWxDDWxDDWxDDWxF7Wx37Wx7FWx2FWx1CWxFF";
char keyTable[39];

```

```

void keyTableGen( int len){

    void *result;
    int i;
    int v2;
    signedint v3;
    char v4[100] = "";
    double dou = *(double*)"Wx38Wx2EWxE3WxA6Wx06Wx9AWxC9Wx3F";

    for ( i = 0; i < 10000; ++i )
        dou = 4.0 * dou * (1.0 - dou);
    for ( i = 0; i < 24; ++i )
    {
        result = (void *)i;
        v4[i] = 0;
    }
    v3 = -1;
    for ( i = 0; i < 8 * len; ++i )
    {
        if ( !(i % 8) )
            ++v3;
        v4[v3] *= 2;
        v2 = dou > 0.5;
        result = (void *) (v2 | v4[ v3 ]);
        v4[v3] = (char)result;
        dou = (double)4.0 * dou * (1.0 - dou);
    }
    for ( i = 0; i < 40; ++i )
    {
        result = (void *)i;
        keyTable[i] = v4[ i ];
    }

}

```

```

void keyGen( char* args )
{
    int i = 0;
    int len;
    char v3;
    char v4[39];
    char g_cKey[39];
    len = strlen(whatitis);

    memset(v4, 0x00, 39);
    keyTableGen( len );
    for ( i = 0; i < len; ++i )
        v4[i] = keyTable[i] ^ whatitis[i];
}

```

```

        for ( i = 0; i < 40; ++i )
g_cKey[i] = v4[i];

        printf( "%s\n", g_cKey );
    }
void main(){
    char args[] = "WxD2Wx04Wx21Wx4BWx38WxFDWx15WxAAWxBEWx3EWxD4Wx6DWx93WxE7WxF3Wx87" W

    "Wx03Wx8EWxB4Wx48Wx29WxDFWxABWxCDWxFDWxFDWxFDWxFDWxDDWxDDWxDD" W

    "WxF7Wx37Wx7FWx2FWx1CWxFFWx00Wx00Wx60Wx00Wx00Wx00Wx64Wx00Wx00Wx00" W

    "Wx01Wx00Wx00Wx00Wx00Wx01Wx01Wx01Wx00Wx01Wx01Wx00Wx00Wx01Wx01Wx00";
    keyGen( args );
}

```

키 :Pocari_SWEAT

12번

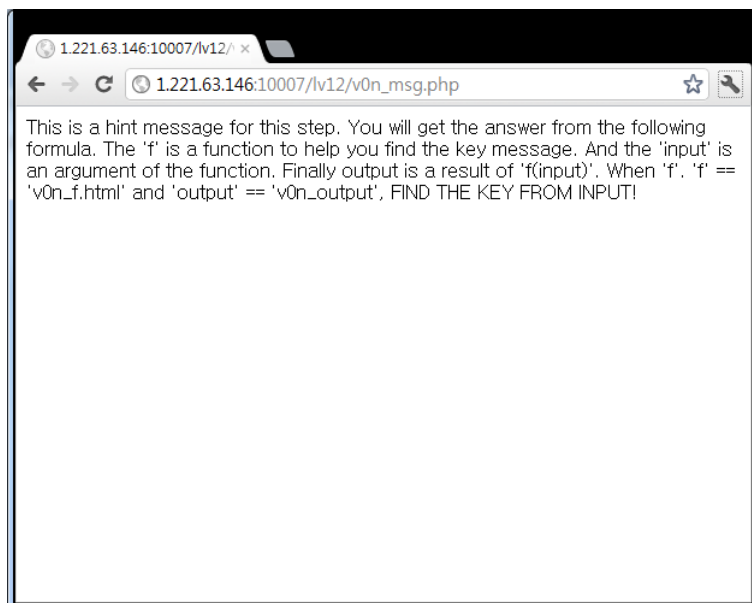
12 <http://1.221.63.146:10007/lv12/prob.mov>

동영상을 틀면 중간에 엄청나게 빠른 초스피드로 한글씨가 나타났다 사라진다. 그걸 긁어 모아서 base64로 디코딩하였다.

k("TWVz"+"c2FnZ"+"SA9IH"+Ywbl9"+"tc2cu"+"cGhW")k==BASE64"

➔ 'Message = v0n_msg.php'

http://1.221.63.146:10007/lv12/v0n_msg.php 여기에 접속하면 다음 단계로 넘어가게 된다.



V0n_f.html 에서 나온 결과가 V0n_output파일이라고 한다.

<html>

<script>

```
function obfuscation(XWRVFFRX){OEXKQJJI="";NWNELNOE="";AHVGFOAZ="PK";VZVYGIGN  
="MZ";SVRYPXOU=["?","?","?","?","?","?","?","?","?","?","?","?","?","?","?","?","?","?","?","?  
,"?","?","?","?","?","?","?","?","?","?","?","?","?","?"];KDPFXBLF=["?","?","?","?","?","?","?","?  
?","?","?","?"];for(YVOAABVY=0;YVOAABVY<  
.....중략
```

```

($_$=0;$_$<_$.length;$_$++)$_$=$_$+TSGSUWUW[1]($_$[_$$_].charCodeAt()^$_$);return
$_$;}

```

</script>

</html>

V0n_f.html을 알아보기 쉽게 바꿔보았다.

```
OEXKQJJ = "";  
NWNELNOE = "";  
PK__ = "PK";  
MZ__ = "MZ";  
key_1 = ["WxA1", "WxA2", "WxA3", "WxA4", "WxA5", "WxA6", "WxA7", "WxA8", "WxA9", "WxAA",  
"WxAB", "WxAC", "WxAD", "WxAE", "WxAF", "WxB0", "WxB1", "WxB2", "WxB3", "WxB4", "WxB5", "WxB6",  
"WxB7", "WxB8", "WxB9", "WxBA", "WxBB", "WxBC", "WxBD", "WxBE", "WxBF"];
```

```

key_2 = ["\u00F0", "\u00F1", "\u00F2", "\u00F3", "\u00F4", "\u00F5", "\u00F6", "\u00F7", "\u00F8", "\u00F9"];
for (i = 0; i < e.length; i++) {
    OOKYPEFF = String.fromCharCode(0xCC);
    if (e.charCodeAt(i) >= 2 * Math.sin(Math.PI) * Math.sqrt(16) * 3 && e.charCodeAt(i) <= Math.abs(3
    * Math.cos(Math.PI)) * 19)
        OEXKQIJ = key_2[Number(e.charCodeAt(i)) - 48];
    else OEXKQIJ = e[i];
    if (i % 2)
        NWNELNOE = NWNELNOE + PK__ + key_1[parseInt(Math.random() *
        31)] + OEXKQIJ + String.fromCharCode(parseInt(Math.random() * 25) + 65) +
        key_1[parseInt(Math.random() * 31)] + OOKYPEFF + "-" +
        String.fromCharCode(parseInt(Math.random() * 25) + 65) + key_1[parseInt(Math.random() * 31)] +
        String.fromCharCode(parseInt(Math.random() * 25) + 65);
    else
        NWNELNOE = NWNELNOE + MZ__ + key_1[parseInt(Math.random() *
        31)] + OEXKQIJ + String.fromCharCode(parseInt(Math.random() * 25) + 65) +
        key_1[parseInt(Math.random() * 31)] + "-" + OOKYPEFF +
        String.fromCharCode(parseInt(Math.random() * 25) + 65) + key_1[parseInt(Math.random() * 31)] +
        String.fromCharCode(parseInt(Math.random() * 25) + 65);
}

NWNELNOE = _$$_$(NWNELNOE, 90);
return NWNELNOE;

```

해당 자바스크립트를 분석해보면 한글자당 11바이트씩 생성된다. 그리고 else OEXKQIJ = e[i]; 이 구문에 의해

NWNELNOE에 저장되는 11바이트중 4번째인 OEXKQIJ변수의 값이 원본 키 값이다.

74 68 f1 73 f1 73 74 68 65 63 68 f4 6c 6c f3 6e 67 f3 66 f0 72 79 f0 75 72 66 75 74 75 74 75 72 f3

V0n_output를 11글자로 자른 후에 4번째에 있는 값을 각각 뽑아온 값이다.

여기서 OEXKQIJ = key_2[Number(e.charCodeAt(i)) - 48]; 이 구문에 의해 변경된 값을 복원하면

74 68 31 73 31 73 74 68 65 63 68 34 6c 6c 33 6e 67 33 66 30 72 79 30 75 72 66 75 74 75 74 75

이 hex값을 문자로 바꾸면 키가 나온다.

키 :th1s1sthech4ll3ng3f0ry0urfutur3

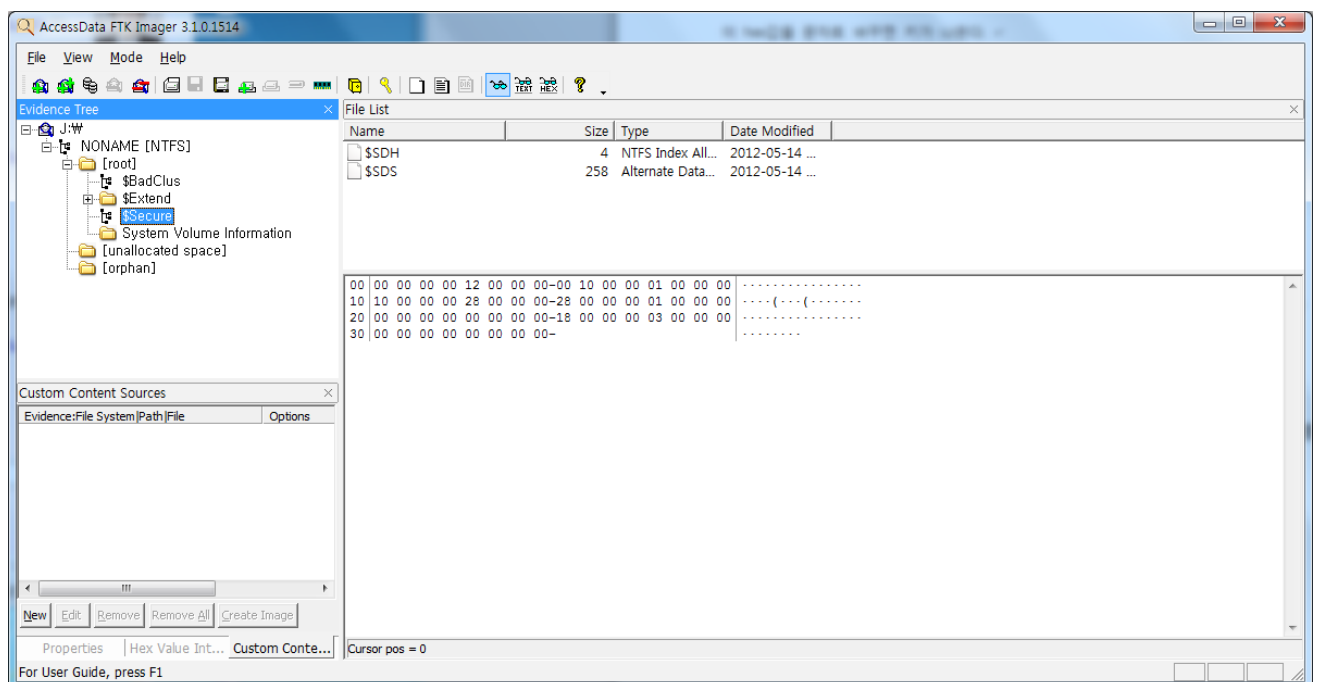
13번

13 <http://1.221.63.146:10007/lv13/NewHeart-1.zip>

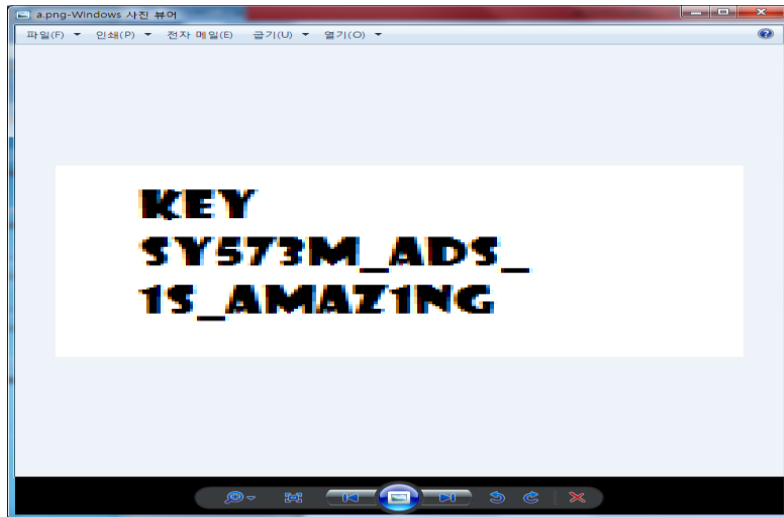
보안카드와 WTF이라는 파일이 주어 졌다.

해당 파일들을 트루크립트 툴로 마운트 시켰다. 보안카드.png는 key file로 사용하였다.

그리고 분석을 하기 위해 FTK Imager를 이용하였다.



ADS 영역에 숨겨져 있는 \$SDS를 0x200 부터 잘라버린 다음 \$TxfLog에 있는 \$T 파일을 이어 붙이면 키값이 나온다.



키 :SY573M_ADS_1S_AMAZING