2020 시스템 해킹 교육 2회

2020. 07. 10.

INDEX

과제 풀이

1번 과제 – My First Hacking

```c
#include <stdio.h>

int main()
{
        unsigned int money = 0;
        int salary = 0;

        printf("\nhello, Your Money is %d$\n\n", money);

        printf("========== PAYDAY ==========\n");
        printf("BOSS : How much do you want to get paid?\n");
        printf("Me : ");
        scanf("%d", &salary);

        if(salary > 100) {
                printf("BOSS : You're fired!\n");
        } else {
                printf("BOSS : Sure, Good choice\n");
                money += (unsigned int) salary;
        }

        printf("\nYour Money is %u$\n", money);
        if(money > 10000) {
                printf("You Win!\n");
        } else {
                printf("You Lose!\n");
        }
}
```

과제 풀이

1번 과제 – My First Hacking

```c
#include <stdio.h>

int main()
{
        unsigned int money = 0;
        int salary = 0;

        printf("\nhello, Your Money is %d$\n\n", money);

        printf("========== PAYDAY ==========\n");
        printf("BOSS : How much do you want to get paid?\n");
        printf("Me : ");
        scanf("%d", &salary);

        if(salary > 100) {
                printf("BOSS : You're fired!\n");
        } else {
                printf("BOSS : Sure, Good choice\n");
                money += (unsigned int) salary;
        }

        printf("\nYour Money is %u$\n", money);
        if(money > 10000) {
                printf("You Win!\n");
        } else {
                printf("You Lose!\n");
        }
}
```

과제 풀이

1번 과제 – My First Hacking

```c
#include <stdio.h>

int main()
{
        unsigned int money = 0;
        int salary = 0;

        printf("\nhello, Your Money is %d$\n\n", money);

        printf("========== PAYDAY ==========\n");
        printf("BOSS : How much do you want to get paid?\n");
        printf("Me : ");
        scanf("%d", &salary);

        if(salary > 100) {
                printf("BOSS : You're fired!\n");
        } else {
                printf("BOSS : Sure, Good choice\n");
                money += (unsigned int) salary;
        }

        printf("\nYour Money is %u$\n", money);
        if(money > 10000) {
                printf("You Win!\n");
        } else {
                printf("You Lose!\n");
        }
}
```

```
minibeef@argos-edu:~/cedu/week2/hw$ ./hw2

hello, Your Money is 0$

========== PAYDAY ==========
BOSS : How much do you want to get paid?
Me : -1
BOSS : Sure, Good choice

Your Money is 4294967295$
You Win!
minibeef@argos-edu:~/cedu/week2/hw$
```

과제 풀이

2번 과제 – Gambling1

문제 제공 : 15 권재승 (myria)

Gambling1 Ranking (Solver : 7)

- 장수진
- 박준서
- 김선규
- ▼ more
  - 최현철
  - 최민우
  - 안준혁
  - 김현구

〈문제 코드가 너무 길어서 라이브로 진행〉

https://www.notion.so/feat-myria-fc5bfffd40db47a38b5034ded45dbeeb

과제 풀이

3번 과제 – Gambling2

문제 제공 : 15 권재승 (myria)

Gambling2 Ranking (Solver : 4)

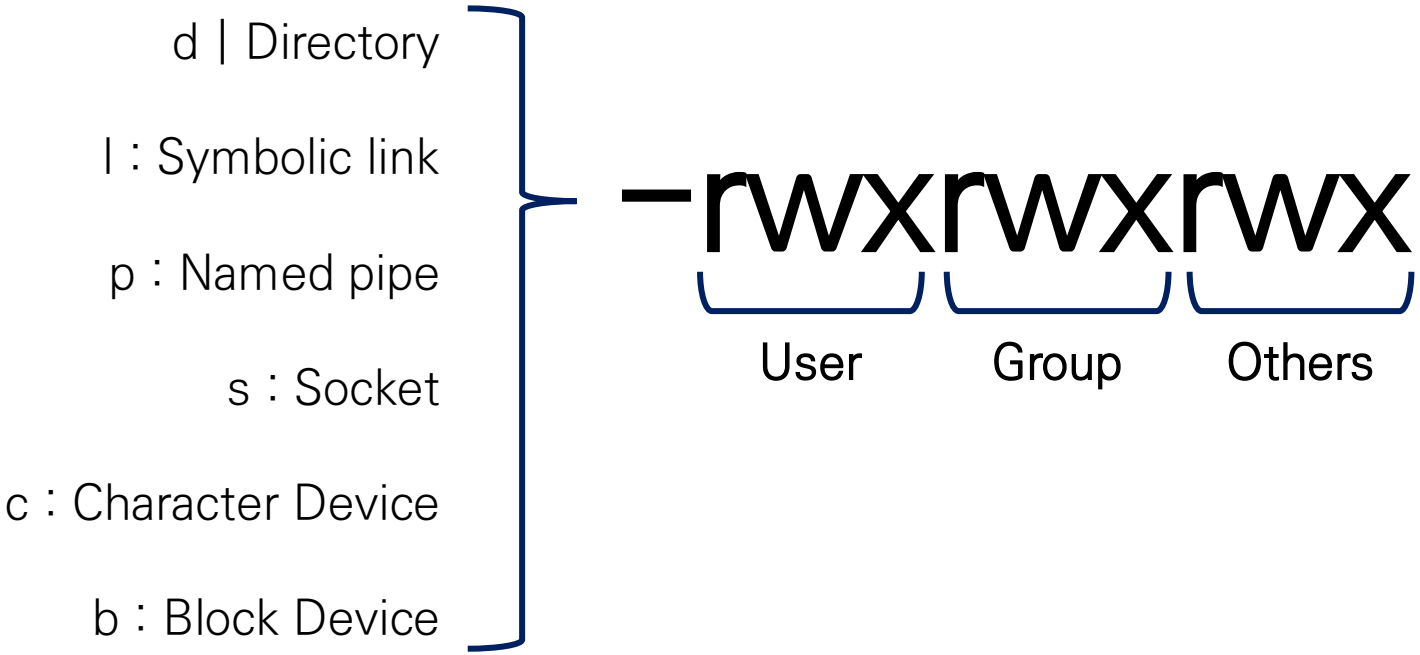
- 장수진
- 김선규
- 박준서 (First Unintended Solver)
- 최민우

▼ more

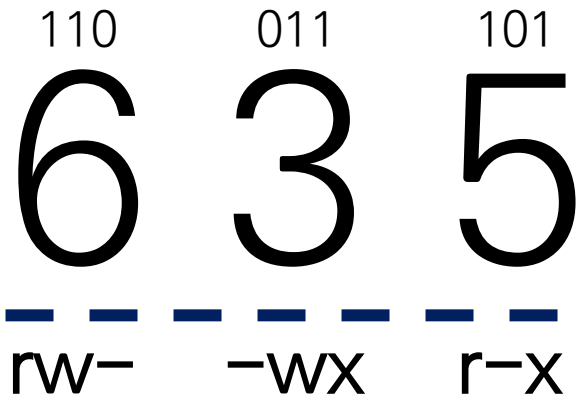
- 안준혁

〈문제 코드가 너무 길어서 라이브로 진행〉

권한 체계 복습

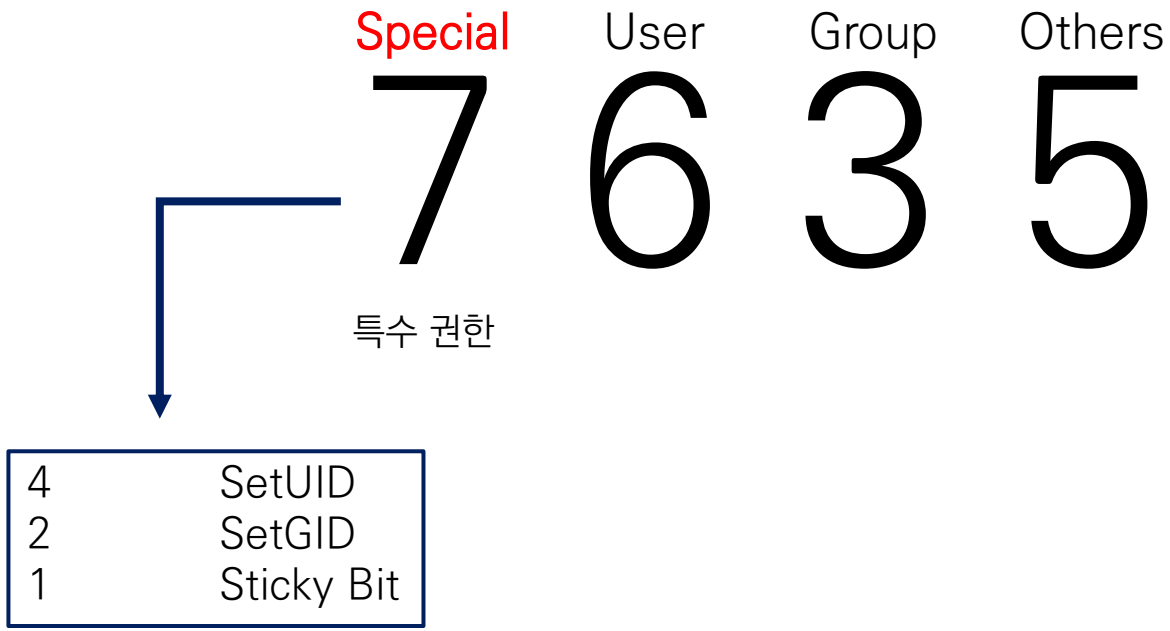

d | Directory
l : Symbolic link
p : Named pipe
s : Socket
c : Character Device
b : Block Device
-rwxrwxrwx
User
Group
Others

권한 체계 복습



특수 권한

SetUID/SetGID



특수 권한

SetUID/SetGID

SetUID
- **소유자**만 접근이 가능한 파일에 일반 유저로 접근이 필요할 때 사용
- 파일을 실행하는 동안 권한을 “빌려 온다”

SetGID
- **소유그룹**만 접근이 가능한 파일에 일반 유저로 접근이 필요할 때 사용
- 파일을 실행하는 동안 권한을 “빌려 온다”

특수 권한

SetUID/SetGID

```
-rwxrwsrwx  1 minibeef minibeef    6 Jul 10 13:27 setgid_test.c
-rwsrwxrwx  1 minibeef minibeef 1087 Jul 10 13:26 setuid_test.c
```

SetGID 파일은 group 실행 권한이 s로,

SetUID는 user 실행 권한이 s로 표시

SetUID : chmod 4~~~
SetGID : chmod 2~~~

특수 권한

SetUID/SetGID

SetUID 사용 예시? : 비밀번호 변경!

```
minibeef@bpsec:~$ ls -al /usr/bin/passwd
-rwsr-xr-x 1 root root 54256 Mar 27  2019 /usr/bin/passwd
```

암호 변경 -> 서버 파일 변경 -> root 권한 필요 -> SetUID

특수 권한

SetUID/SetGID

```c
#include <stdio.h>
#include <unistd.h>

int main() {
        printf("uid :: %d euid :: %d\n", getuid(), geteuid());
        setuid(500);
        printf("uid :: %d euid :: %d\n", getuid(), geteuid());
        return 0;
}
```

```
minibeef@argos-edu:~$ sudo ./20200710
uid :: 0 euid :: 0
uid :: 500 euid :: 500
```

root : 0

특수 권한

Sticky Bit

Sticky Bit
- Sticky Bit가 설정된 디렉토리에 누구나 파일 생성 가능
- 삭제 및 수정은 관리자만 가능(소유자와 root)

```
drwxrwxrwt 2857 root root     126976 Jul 10 05:17 tmp
```

Command Injection

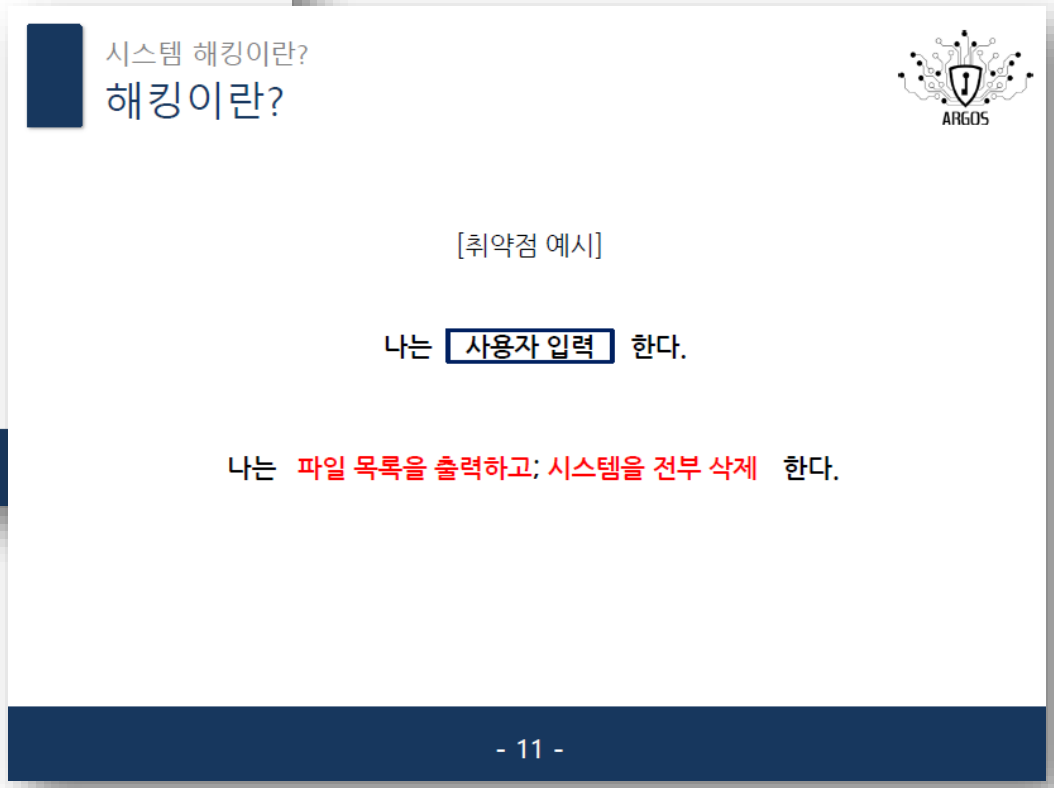
커맨드 인젝션 이란?

Command Injection

명령어 삽입

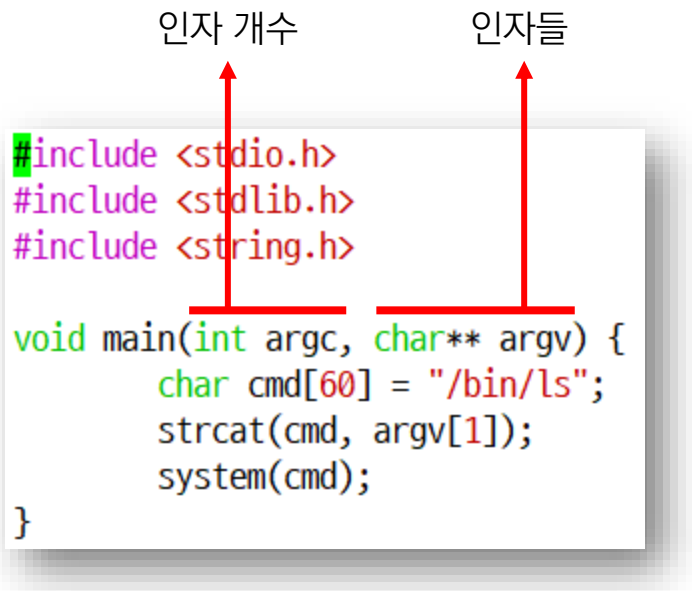
Command Injection

커맨드 인젝션 이란?



Command Injection

커맨드 인젝션 이란?




```
minibeef@argos-edu:~/sysedu$ ./prac.c ;id
```

argv[0] argv[1]

Command Injection

커맨드 인젝션 이란?

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

void main(int argc, char** argv) {
        char cmd[60] = "/bin/ls";
        strcat(cmd, argv[1]);
        system(cmd);
}
```

```
minibeef@argos-edu:~/sysedu$ ./prac1 ";/bin/ls"
hw1  prac1  prac1.c  test
hw1  prac1  prac1.c  test
```

명령어 삽입을 통해 ls가 두 번 실행된 모습

Command Injection

커맨드 인젝션 이란?

Meta 문자	설명	예시
&&	이전 명령어 실행 후 다음 명령어 실행	$ echo hello && echo bye hello bye
;	명령어 구분자	$ echo hello ; echo bye hello bye
|	명령어 파이핑	$ echo id | /bin/sh
*	와일드 카드	$ echo .*
`	명령어 치환	$ echo `echo hello` hello

Command Injection

커맨드 인젝션 이란?

```
minibeef@argos-edu:~/sysedu$ sudo chmod 4777 prac1
minibeef@argos-edu:~/sysedu$ ls -al
total 52
drwxrwxr-x  2 minibeef minibeef  4096 Jul 10 05:33 .
drwxr--r-- 24 minibeef minibeef  4096 Jul 10 04:40 ..
-rwxrwxr-x  1 minibeef minibeef  8448 Jul  7 06:24 hw1
-rwsrwxrwx  1 root     root      8400 Jul 10 05:33 prac1
-rw-rw-r--  1 minibeef minibeef   163 Jul 10 05:32 prac1.c
-rw-r--r--  1 minibeef minibeef 12288 Jul 10 05:28 .prac1.c.swp
-rwxr-xrw-  1 minibeef minibeef     7 Jul  7 10:56 test
```

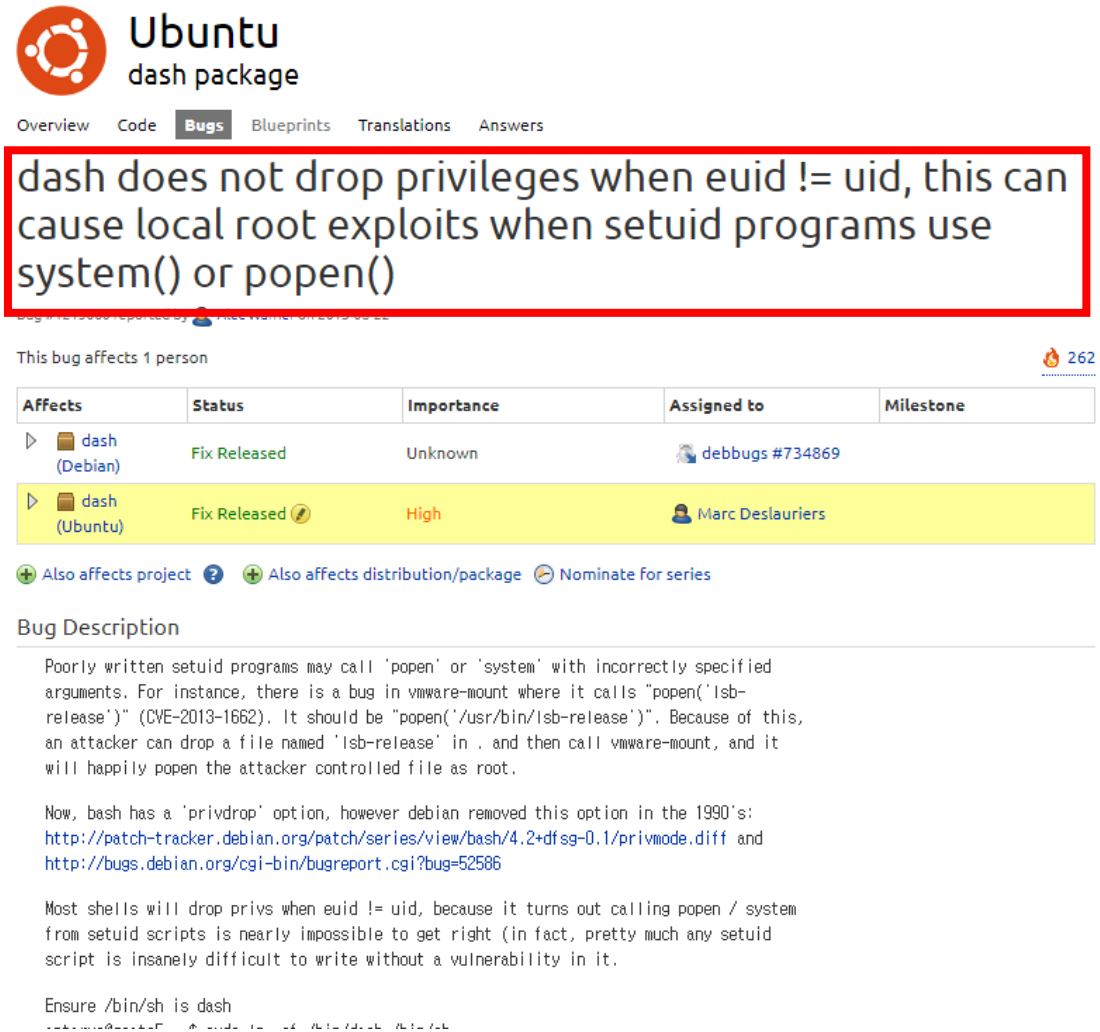
Command Injection

커맨드 인젝션 이란?

```
minibeef@argos-edu:~/sysedu$ ./prac1 ";id"
hw1  prac1  prac1.c  test
uid=1002(minibeef) gid=1002(minibeef) groups=1002(minibeef),999(docker)
```

Command Injection

커맨드 인젝션 이란?



Ubuntu
dash package

Overview Code **Bugs** Blueprints Translations Answers

dash does not drop privileges when euid != uid, this can cause local root exploits when setuid programs use system() or popen()

This bug affects 1 person 262

Affects	Status	Importance	Assigned to	Milestone
dash (Debian)	Fix Released	Unknown	debbugs #734869	
dash (Ubuntu)	Fix Released	High	Marc Deslauriers	

Also affects project Also affects distribution/package Nominate for series

Bug Description

Poorly written setuid programs may call 'popen' or 'system' with incorrectly specified arguments. For instance, there is a bug in vmware-mount where it calls "popen('lsb-release')" (CVE-2013-1662). It should be "popen('/usr/bin/lsb-release')". Because of this, an attacker can drop a file named 'lsb-release' in . and then call vmware-mount, and it will happily popen the attacker controlled file as root.

Now, bash has a 'privdrop' option, however debian removed this option in the 1990's: http://patch-tracker.debian.org/patch/series/view/bash/4.2+dfsg-0.1/privmode.diff and http://bugs.debian.org/cgi-bin/bugreport.cgi?bug=52586

Most shells will drop privs when euid != uid, because it turns out calling popen / system from setuid scripts is nearly impossible to get right (in fact, pretty much any setuid script is insanely difficult to write without a vulnerability in it.

Ensure /bin/sh is dash

Command Injection

커맨드 인젝션 실습

```c
#include <stdlib.h>
#include <stdio.h>
int main()
{
    char ip[36];
    char cmd[256] = "ping -c 2 ";

    printf("Alive Checker\n");
    printf("IP: ");
    read(0, ip, sizeof(ip)-1);
    printf("IP Check: %s",ip);
    strcat(cmd, ip);
    system(cmd);
    return 0;
}
```

1. 컴파일 후 실행
2. 커맨드 인젝션을 통해 ls 명령 실행해보기

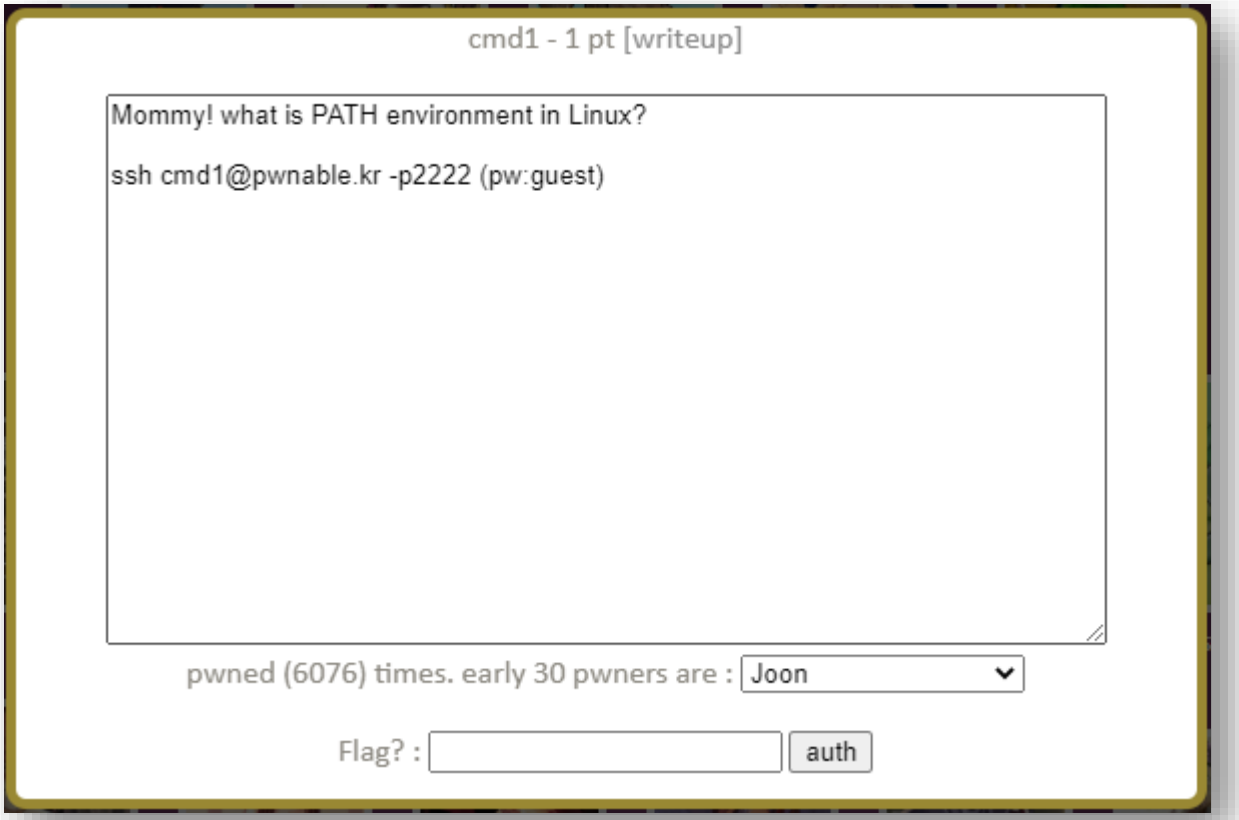
Wargame 소개

pwnable.kr

https://pwnable.kr/index.php

과제 설명

과제1 – pwnable.kr cmd1



과제 설명

과제1 – pwnable.kr cmd1

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

int filter(char* cmd){
        int r=0;
        r += strstr(cmd, "flag")!=0;
        r += strstr(cmd, "sh")!=0;
        r += strstr(cmd, "tmp")!=0;
        return r;
}
int main(int argc, char* argv[], char** envp){
        putenv("PATH=/thankyouverymuch");
        if(filter(argv[1])) return 0;
        system( argv[1] );
        return 0;
}
```

```
cmd1@pwnable:~$ ./cmd1 [redacted]
mommy now I get what PATH environment is for :)
```