Introduction to pwn

Ali Ghaffarian

Me?

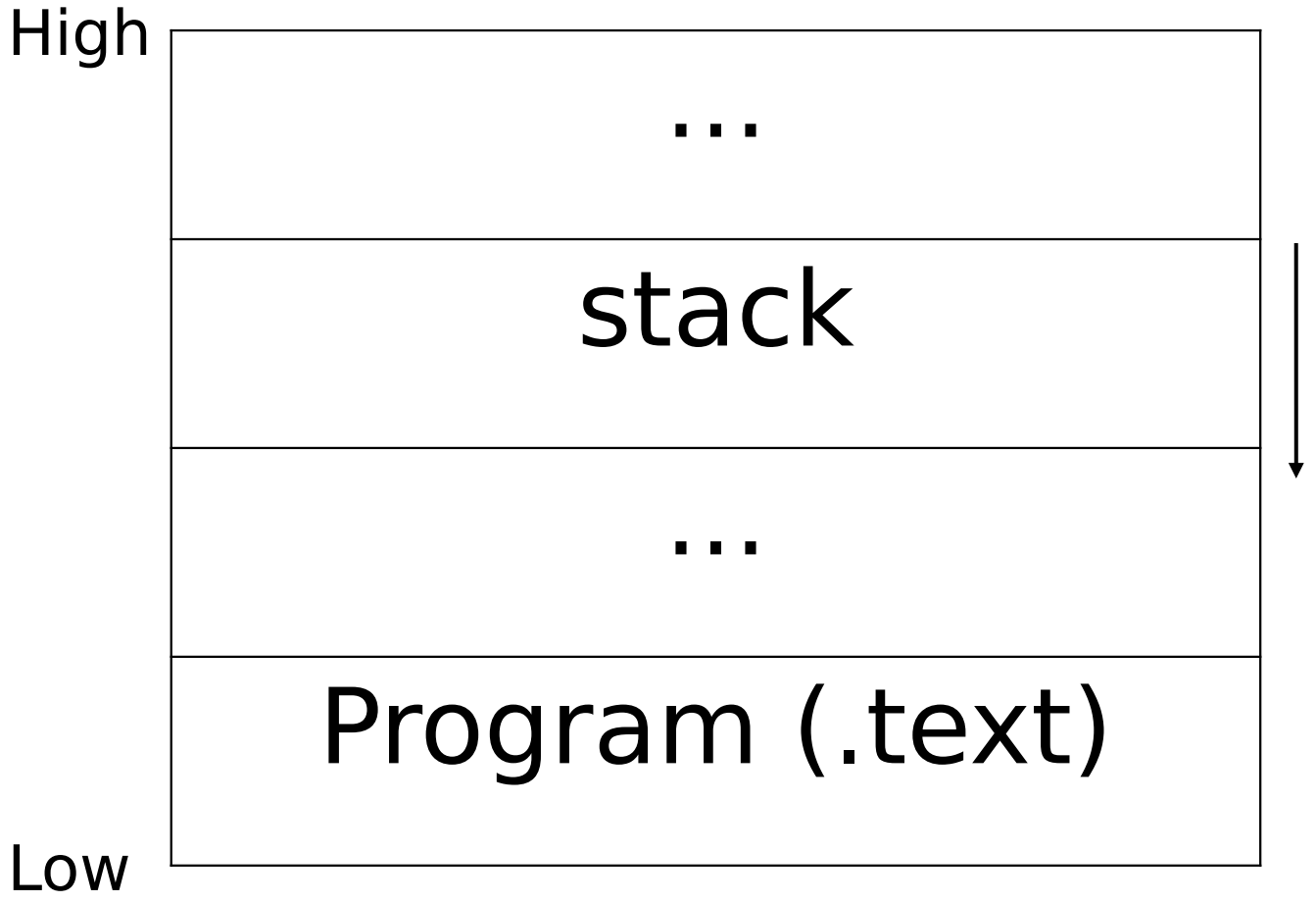
- Interested in
  - Operating Systems
  - Networking
- CTF Player (Definitely not a pro)
  - Active member of flagmotori
  - Network Forensics
  - pwn

Table of Content

- pwn?
- Program Memory Layout
- Amd64 Architecture
- Stack Memory
- Exploitation
- Demo!

pwn?

Program Memory Layout


High
...
stack
...
Program (.text)
Low

Amd64 Architecture

Registers (of interest)

- Stack Memory
  - [r]bp (base pointer)
  - [r]sp (stack pointer)
- Execution
  - [r]ip (instruction pointer)

Instructions (of interest)

- push OPERAND
  - sub rsp, 8
  - mov [rsp], OPERAND
- call ADDRESS
  - push rip
  - jmp ADDRESS
- ret
  - pop rip

Stack Memory

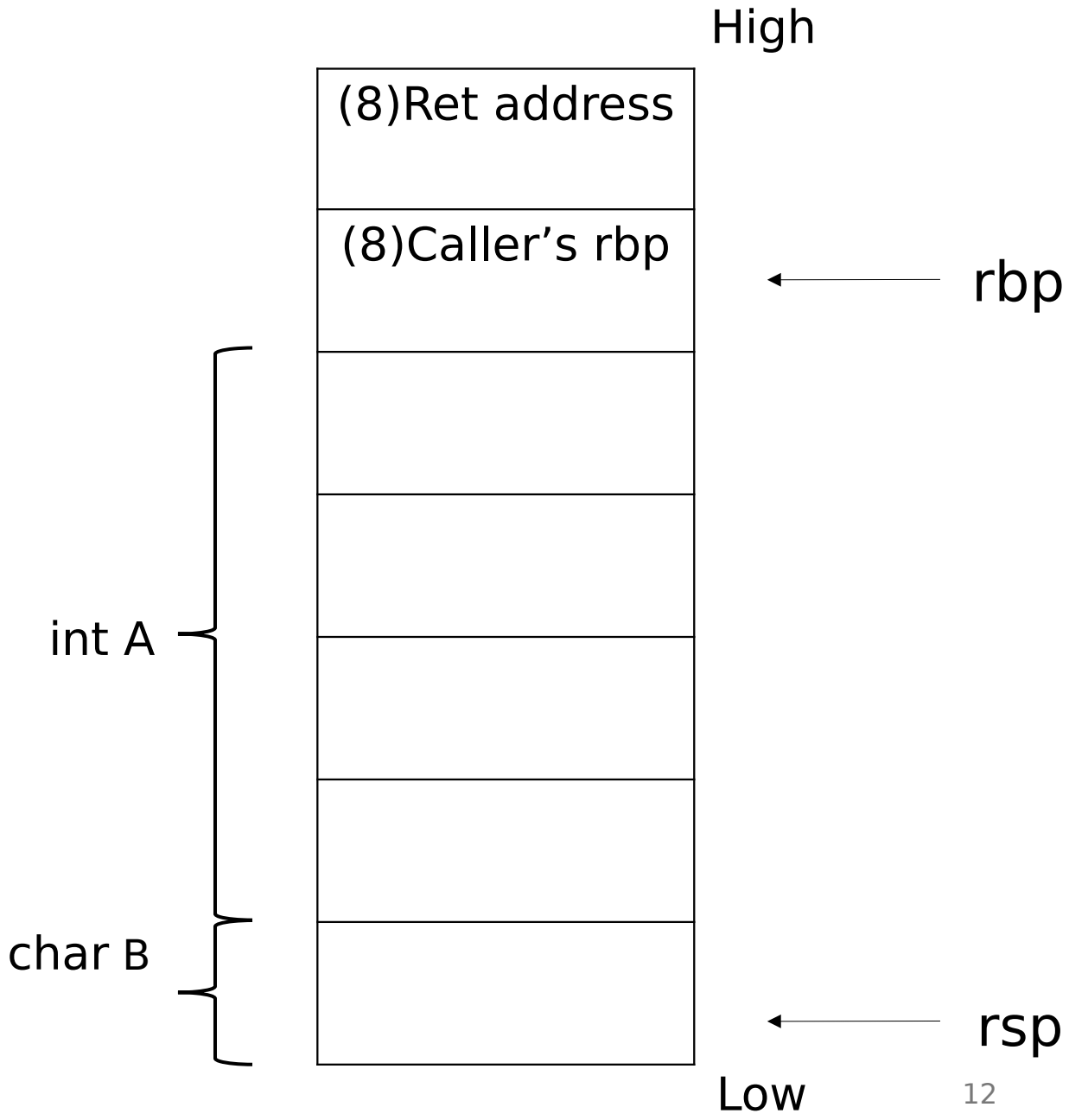
usage

```
void do_nothing(){
        int A;
        char B;
        return;
}
```



```
; setup stack frame
push rbp
mov rbp, rsp

; allocate memory for local vars
sub rsp, 4 ;int A
sub rsp, 1 ;int B


;restore caller's stack frame
mov rsp, rbp
pop rbp

ret
```

```
; setup stack frame
push rbp
mov rbp, rsp

; allocate memory for local
vars
sub rsp, 4 ;int A
sub rsp, 1 ;int B

;restore caller's stack
frame
mov rsp, rbp
pop rbp

ret
```

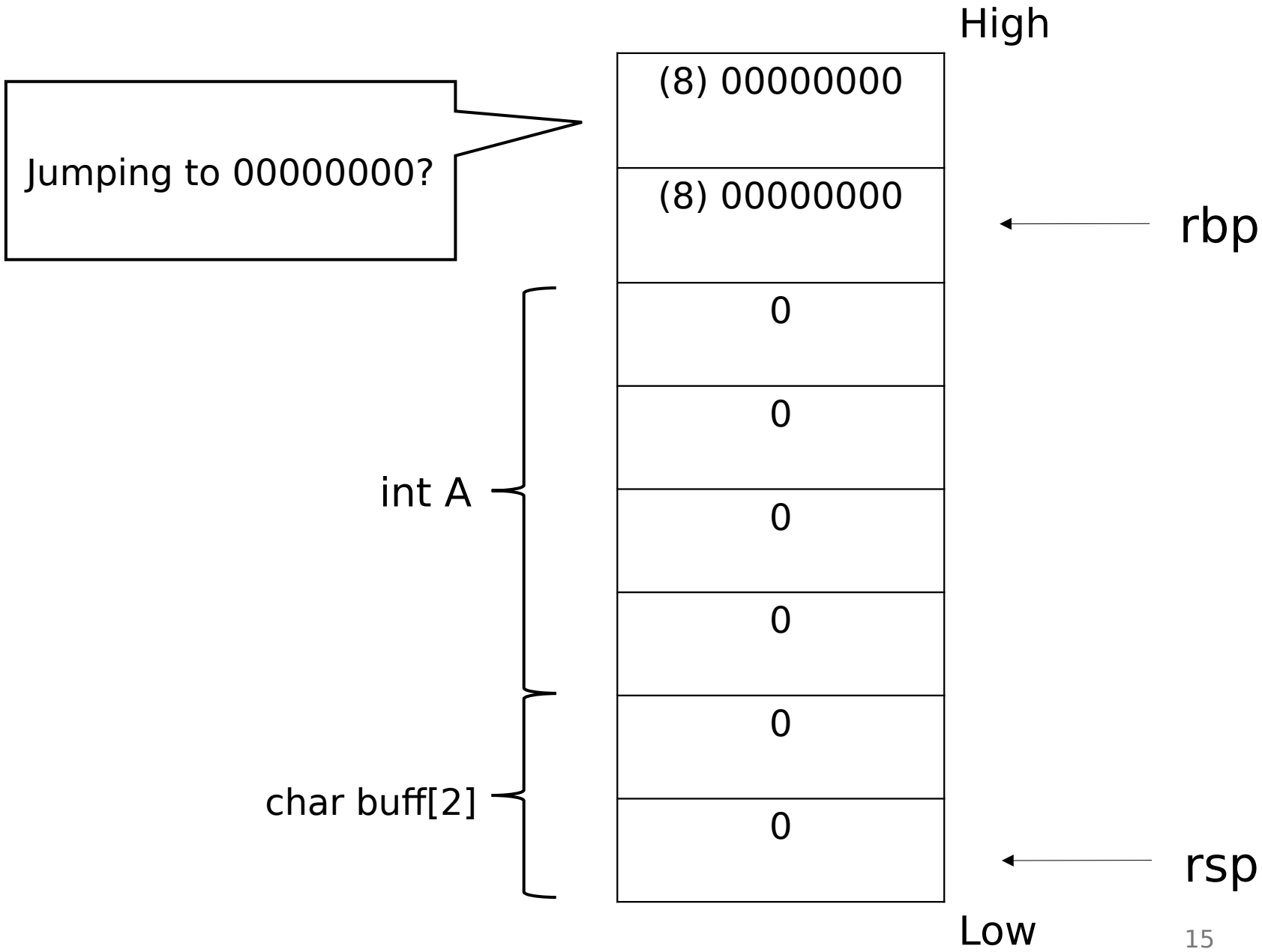


Exploitation

Return Address Overwrite

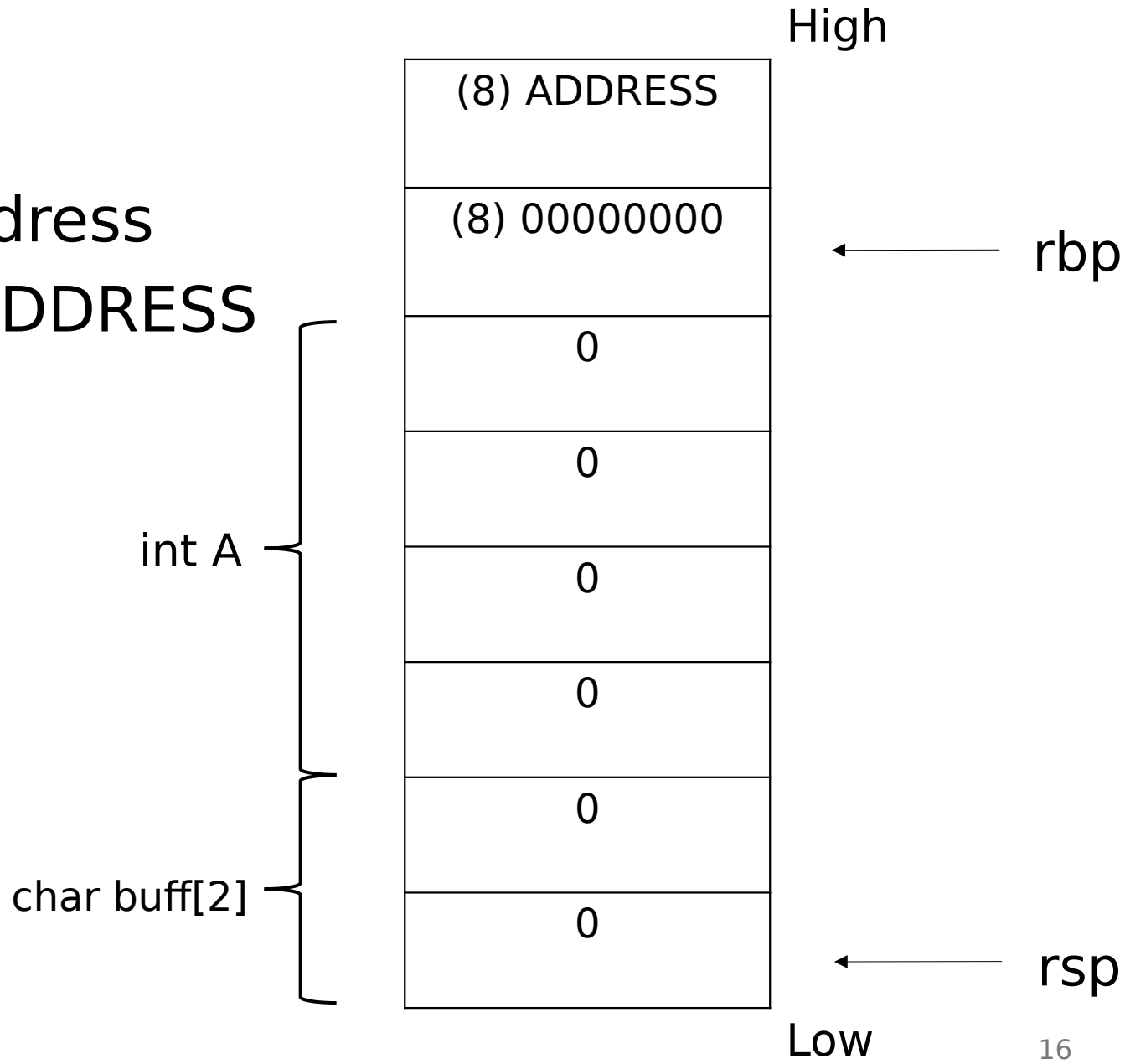
```
void vuln(){
        int A;
        char buffer[2];
        read(STDIN_FILENO, buffer, 100);
        return;
}
```

Input: 0 * 100


High
(8) 00000000
(8) 00000000
rbp
Jumping to 00000000?
0
0
0
0
int A
0
0
char buff[2]
rsp
Low

Idea: overwrite the ret address

Payload: 0 * 6 + 0 * 8 + ADDRESS



Demo!

Shellcode Injection

```
void vuln(){
        char anotherbuff[512];
        char buffer[2];
        read(STDIN_FILENO, buffer, 1024);
        return
}
```

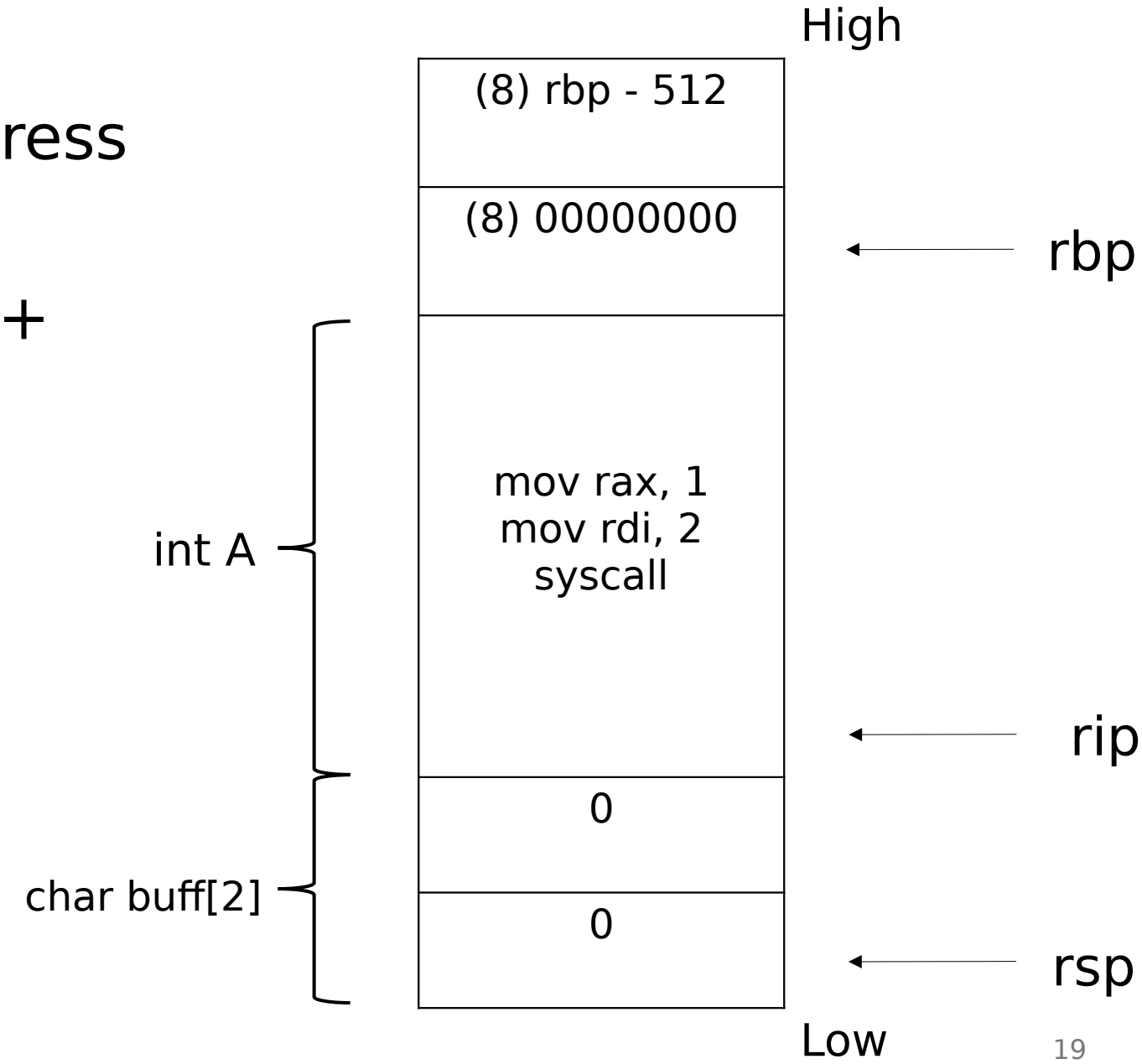
Idea: overwrite the ret address

Payload: 0 * 2 +

shellcode(512 bytes) +

0 * 8 +

rbp - 512



Demo!

How to Get Started

Thanks!

Files: github.com/AliGhaffarian/talks