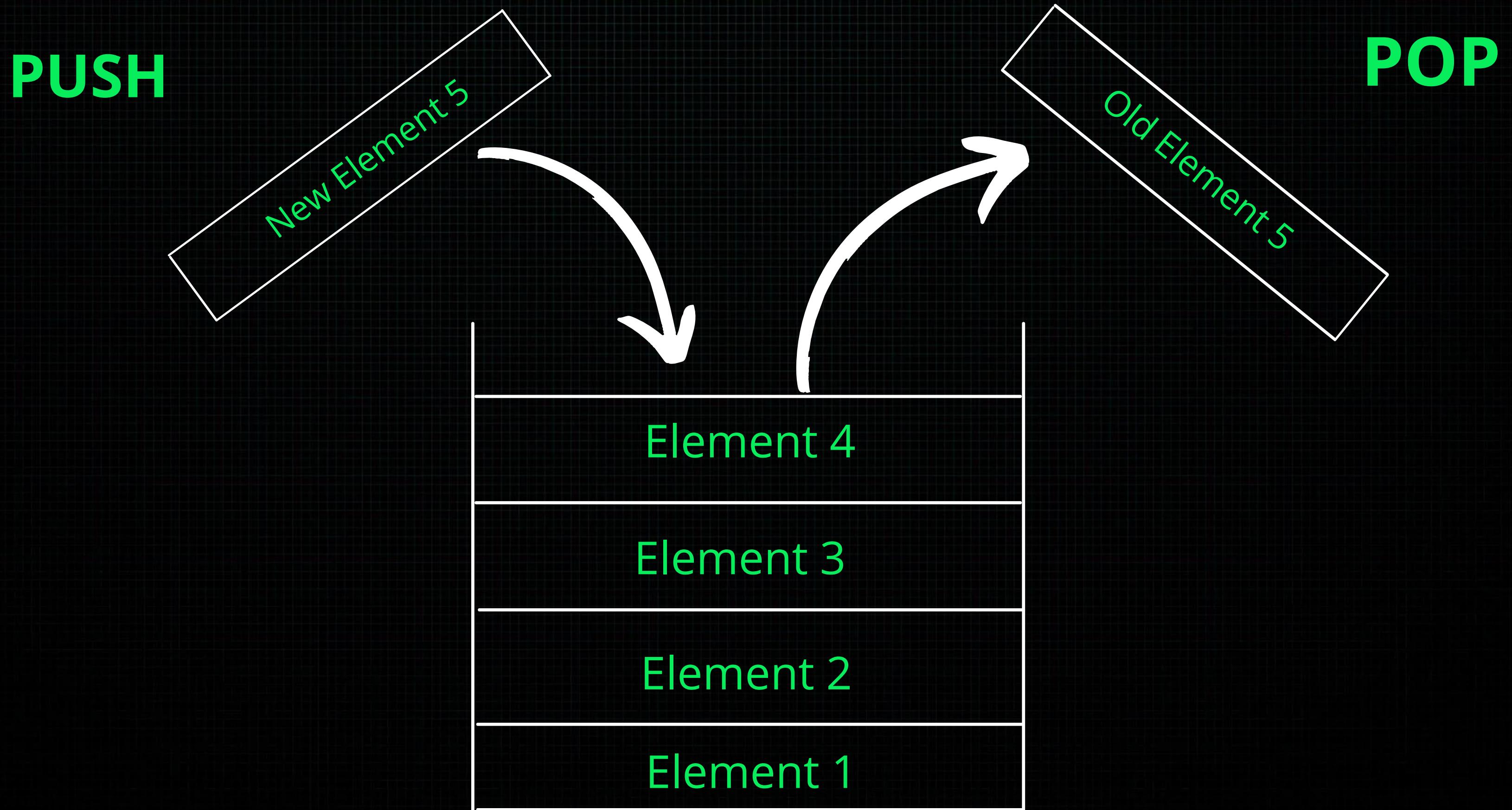


Pwn - Stacks

Stack



STACK

While going Up
the Stack



Memory Addresses Value
Decreases

Local Variables

Saved RBP (old)

Return Address

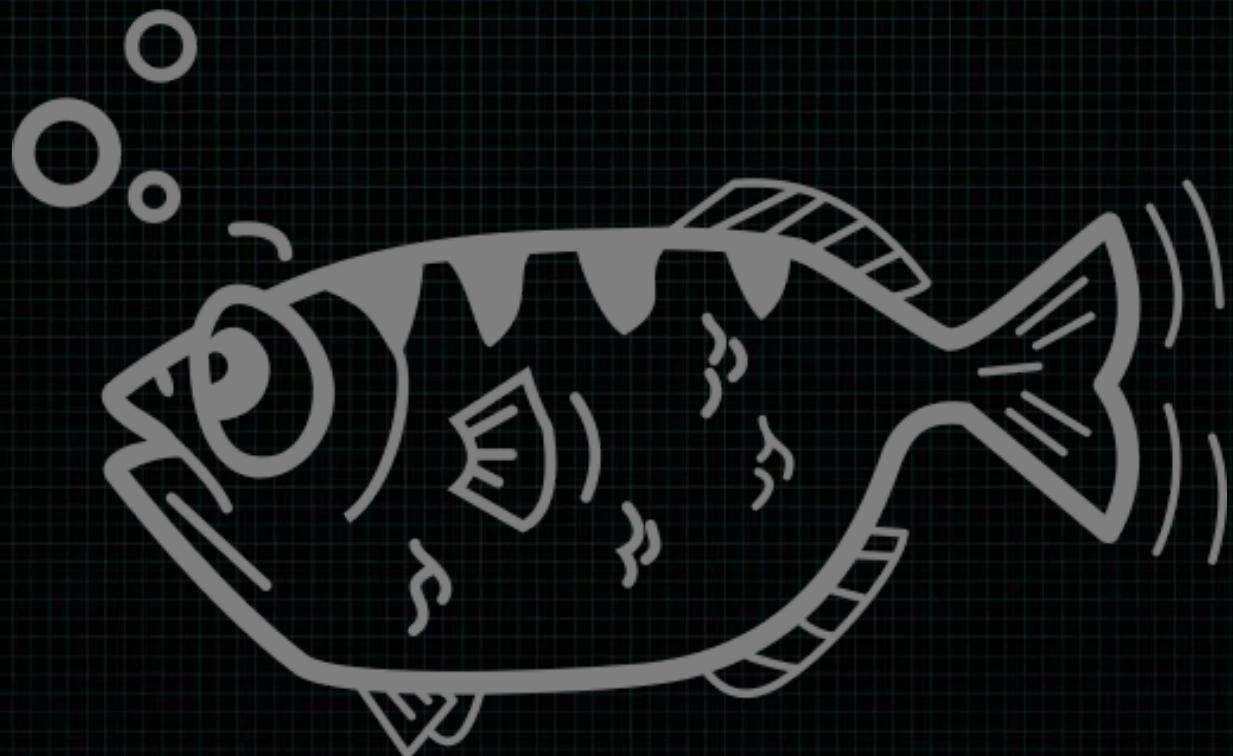
Arguments

Sample Function and Stack

```
int sample(int arg1,float arg2)
{
    int lvar1=arg1;
    float lvar2=arg2;
    return lvar1 + (int) lvar2;
}
```



Debuggers



GNU DeBugger

Take a look at an example !

GDB EXTENSIONS

gef> [gef prompt]

Breakpoint 1, 0x000055555555189 in main ()

LEGEND: STACK | HEAP | CODE | DATA | RWX | RODATA

	[REGISTERS]	[DISASM]	[STACK]	[BACKTRACE]
\$r9 : 0x00007ffff7fe0d50	→ endbr64	→ 0x55555555189 <main>	00:0000 rsp 0x7fffffdf48 → 0x7ffff7dde0b3 (<_libc_start_main+243>)	→ f 0 0x55555555189 main
\$r10 : 0x7		RAX 0x55555555189 (main) ← endbr64	0008 0x7fffffdf50 → 0x7ffff7ffc620 → 0x50b1600000000	f 1 0x7ffff7dde0b3 __libc_start_main+243
\$r11 : 0x2		RBX 0x55555555200 (<_libc_csu_init>) ← endbr64	0016 0x7fffffdf58 → 0x7fffffe038 → 0x7fffffe371 ("./home/sswastik02/a.out")	
\$r12 : 0x0000555555550a0	→ <_start+0> endbr64	RCX 0x55555555200 (<_libc_csu_init>) ← endbr64	0024 0x7fffffdf60 → 0x10000000	
\$r13 : 0x00007fffffff030	→ 0x0000000000000001	RDX 0x7fffffe048 → 0x7fffffe388 ← 'LANGUAGE=en_IN:en'	0032 0x7fffffdf68 → 0x55555555189 (<main>: endbr64)	
\$r14 : 0x0		RDI 0x1	0040 0x7fffffdf70 → 0x55555555200 (<_libc_csu_init>: endbr64)	
\$r15 : 0x0		RSI 0x7fffffe038 → 0x7fffffe371 ←	0048 0x7fffffdf78 → 0x1e22f92368661e83	
\$eflags: [ZERO carry PARITY adjust sign trap INTERRUPT direction overflow resume virtualx86 identification]		R8 0x0	0056 0x7fffffdf80 → 0x55555555189 (<main>: endbr64)	
\$cs: 0x0033 \$ss: 0x002b \$ds: 0x0000 \$es: 0x0000 \$fs: 0x0000 \$gs: 0x0000		R9 0x7ffff7fe0d50 ← endbr64	0064 0x55555555179 <__do_global_dtors_aux+57>:	
	stac	R10 0x7	nop DWORD PTR [rax+0x0]	
k —		R11 0x2	0x55555555180 <frame_dummy>: endbr64	
0x00007fffffdf48 +0x0000: 0x00007ffff7dde0b3 → <_libc_start_main+243> mov edi, eax ← \$rsp		R12 0x555555550a0 (_start) ← endbr64	0x55555555181 <frame_dummy+4>:	
0x00007fffffdf50 +0x0008: 0x00007ffff7ffc620 → 0x00050b1600000000		R13 0x7fffffe030 ← 0x1	0x55555555100 <register_tm_clones>:	
0x00007fffffdf58 +0x0010: 0x00007fffffe038 → 0x00007fff fe371 →		R14 0x0	endbr64	
0x00007fffffdf60 +0x0018: 0x0000000100000000		R15 0x0	0x55555555179 <__do_global_dtors_aux+57>:	
0x00007fffffdf68 +0x0020: 0x000055555555189 → <main+0> endbr64		RBP 0x0	nop DWORD PTR [rax+0x0]	
0x00007fffffdf70 +0x0028: 0x000055555555200 → <_libc_csu_init+0> endbr64		RIP 0x55555555189 (main) ← endbr64	0x55555555180 <frame_dummy>: endbr64	
0x00007fffffdf78 +0x0030: 0x03b45a9e5d039032		[DISASM]	0x55555555181 <frame_dummy+4>:	
0x00007fffffdf80 +0x0038: 0x0000555555550a0 → <_start> endbr64		→ 0x55555555189 <main> endbr64	0x55555555100 <register_tm_clones>:	
		mov rbp, rbp	endbr64	
4 —		sub sp, sp	0x55555555180 <main+4>:	
0x55555555179 <__do_global_dtors_aux+57> nop DWORD PTR [rax+0x0]		xor eax, eax	push rbp	
[rax+0x0]		mov rax, [rbp - 0xc]	mov rbp, rsp	
0x55555555180 <frame_dummy+0> endbr64		lea rax, [rbp - 0xc]	0x55555555191 <main+8>:	
0x55555555184 <frame_dummy+4> jmp 0x55555555100 <register_tm_clones>:		lea rsi, rax	sub rsp, 0x10	
istarter_tm_clones>		mov rsi, rax	0x55555555195 <main+12>:	
→ 0x55555555189 <main+0> endbr64		lea rdi, [rip + 0xe52]	mov rax, QWORD PTR fs:0x28	
0x5555555518d <main+4> push rbp		mov eax, 0	[-----stack-----]	
0x5555555518e <main+5> mov rbp, rbp		[STACK]	0000 0x7fffffdf48 → 0x7ffff7dde0b3 (<_libc_start_main+243>)	
0x55555555191 <main+8> sub rsp, 0x10		0008 0x7fffffdf50 → 0x7ffff7ffc620 → 0x50b1600000000		
0x55555555195 <main+12> mov rax, QWORD PTR fs:0x0		0016 0x7fffffdf58 → 0x7fffffe038 → 0x7fffffe371 ("./home/sswastik02/a.out")		
x28		0024 0x7fffffdf60 → 0x100000000		
0x5555555519e <main+21> mov QWORD PTR [rbp-0x8]		0032 0x7fffffdf68 → 0x55555555189 (<main>: endbr64)		
, rax		0040 0x7fffffdf70 → 0x55555555200 (<_libc_csu_init>: endbr64)		
thread		0048 0x7fffffdf78 → 0x1e22f92368661e83		
s —		0056 0x7fffffdf80 → 0x555555550a0 (<_start>: endbr64)		
[#0] Id 1, Name: "a.out", stopped 0x55555555189 in main (), reason: BREAKPOINT		[-----]		
trac		Legend: code, data, rodata, value		
e —		Breakpoint 1, 0x000055555555189 in main ()		
[#0] 0x55555555189 → main()		gdb-peda\$		

gef> [gef prompt]

pwndbg> [pwndbg prompt]

gef> [gef prompt]

gef> [gef prompt]

Decompilers



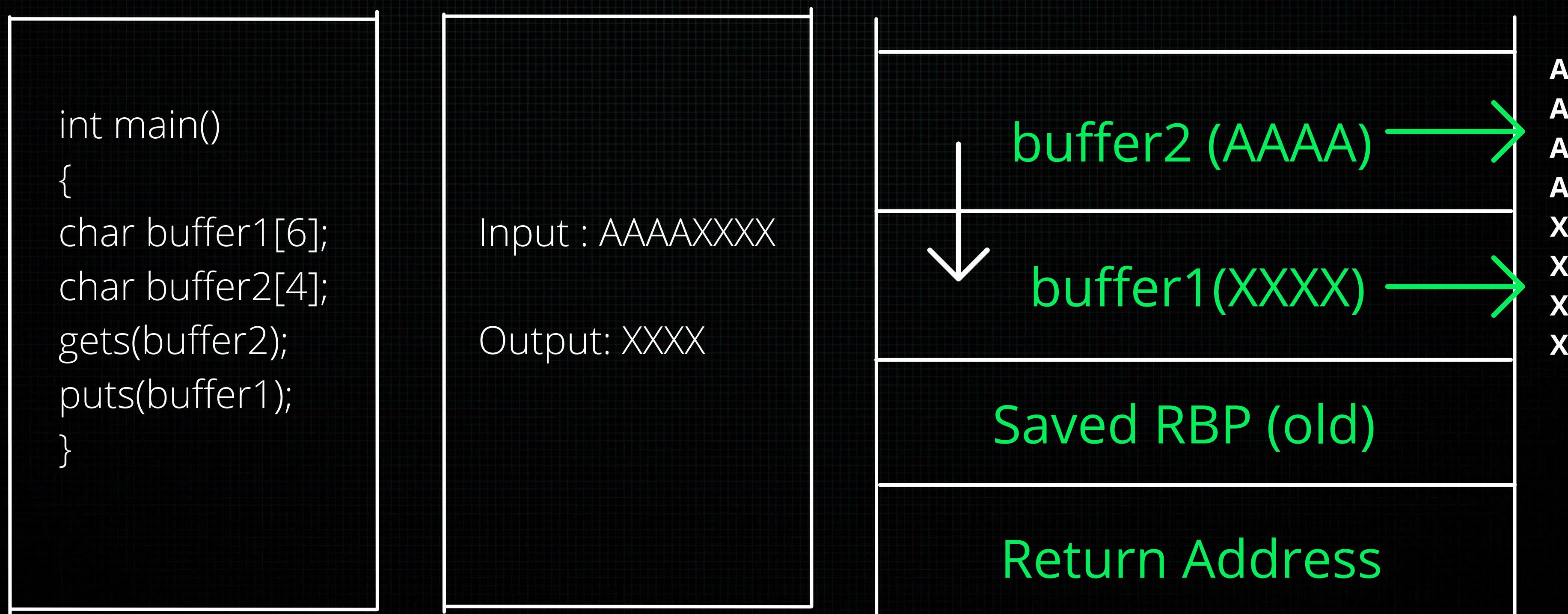
GHIDRA



IDA PRO

*Note: They Make Life Easier !

Buffer Overflow



Code

IO

Stack

PWNTOOLS

```
$ pip3 install pwntools
```

```
from pwn import *

elf = ELF("./a.out")
p = elf.process()

payload = b'AAAA' + b'SHELL'
p.sendline(payload)
p.interactive()
```



Th@nk\$ for
W47ch1n9 !

Next Video ▶
IP manipulation