ROP Chain
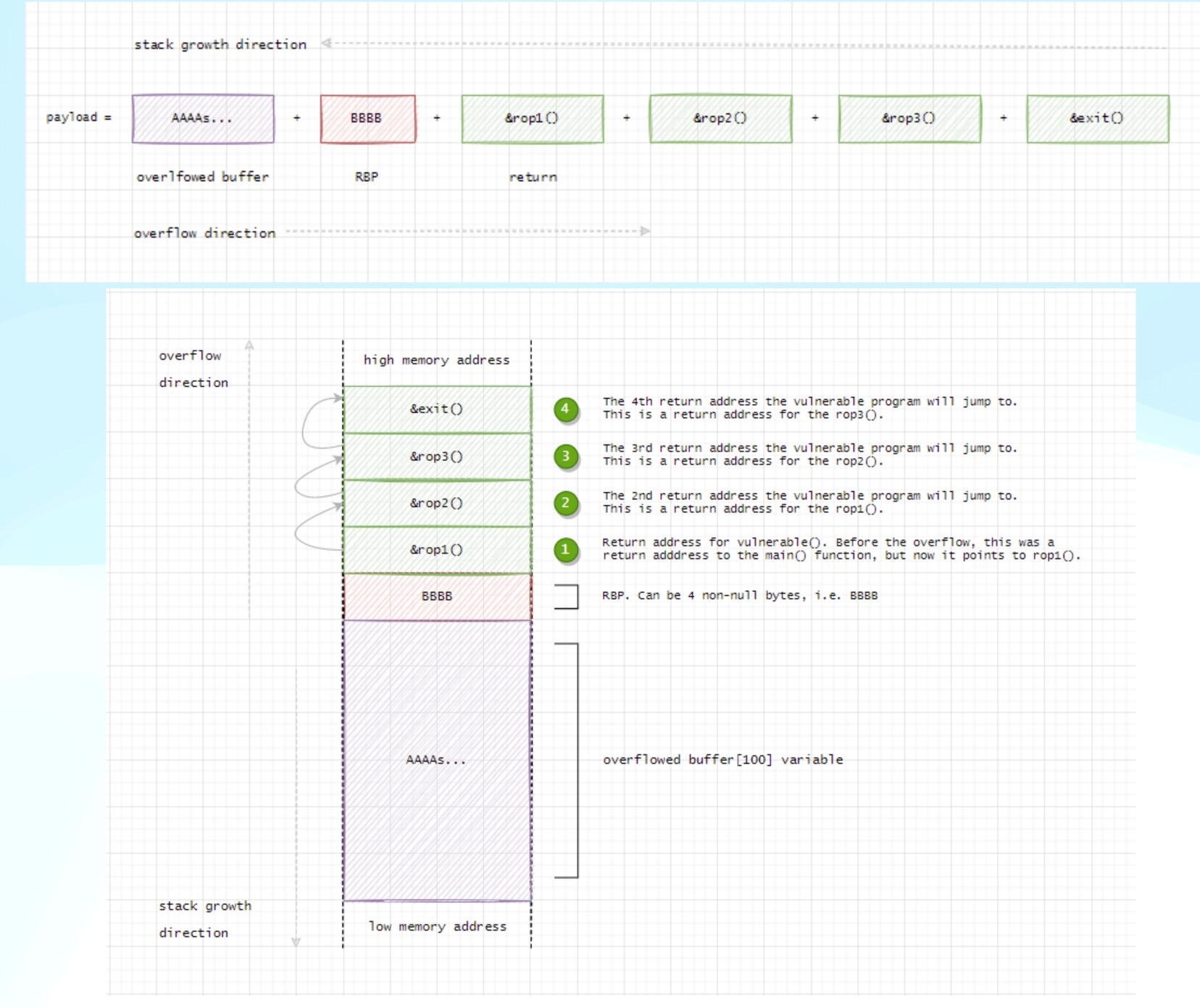
- We combine multiple micro-operations to set the environment according to our needs.

- This process is called chaining.

payload = [AAAAs...] + [BBBB] + [&rop1()] + [&rop2()] + [&rop3()] + [&exit()]

stack growth direction

overlfowed buffer   RBP   return

overflow direction

overflow direction

high memory address

4 &exit() — The 4th return address the vulnerable program will jump to. This is a return address for the rop3().

3 &rop3() — The 3rd return address the vulnerable program will jump to. This is a return address for the rop2().

2 &rop2() — The 2nd return address the vulnerable program will jump to. This is a return address for the rop1().

1 &rop1() — Return address for vulnerable(). Before the overflow, this was a return adddress to the main() function, but now it points to rop1().

BBBB — RBP. Can be 4 non-null bytes, i.e. BBBB

AAAAs... — overflowed buffer[100] variable

stack growth direction

low memory address

Src: https://www.ired.team/offensive-security/code-injection-process-injection/binary-exploitation/rop-chaining-return-oriented-programming