# ASLR - Address Space Layout Randomization

- The OS randomize the location (random offset) where the standard libraries and other elements are stored in memory
- Harder for the attacker to guess the address of a lib-c subroutine
- Disabling ASLR protection on Linux
  `$ sysctl kernel.randomize_va_space=0`
- Bypassing ASLR protection : Brute-force attack to guess the ASLR offset
- Bypassing ASLR protection : *Return-Oriented-Programming (ROP)* exploit use instruction pieces of the existing program (called "gadgets") and chain them together to weave the exploit

# PIC/PIE - Position Independent Code/Executables

- **Without PIC/PIE**
  code is compiled with absolute addresses and must be loaded at a specific location to function correctly

- **With PIC/PIE**
  code is compiled with relative addressing that are resolved dynamically when executed by calling a function  to obtain the return value on stack