Modification to the OpenBSD kernel

This code block previously checked if the program counter (PC) was within the allowed executable regions (sigtramp, libc, ld.so) when making a syscall. If the PC was outside these regions, it would return an EPERM error, denying the syscall.

By removing this check, the kernel no longer enforces that syscalls must originate from within libc or other whitelisted regions. This effectively disables the mitigation and allows programs to make syscalls directly from arbitrary code locations, potentially enabling exploitation techniques like code injection or return-oriented programming (ROP) that construct syscalls dynamically.

While this mitigation aimed to make exploitation more difficult by forcing the use of libc's syscall environment, its removal reopens the possibility of directly injecting and invoking syscall instructions from attacker-controlled code regions.

Testing the change

To ensure this modification enabled syscalls from any region, a C source file is provided. Since the code specifically invokes the exit system call (1 in %rax), the successful execution of this code would indicate that the modification to the kernel was indeed effective. If the program terminates without issues, it suggests that system calls can now be executed without strict validation of their origin, as was previously enforced.