

Implementação system calls no kernel linux V. 4.13.12

João Paulo de Oliveira joapaulodeoliveira123@gmail.com
Lucas Rossi Rabelo lucasrossi98@hotmail.com
Matheus Pimenta Reis

7 de dezembro de 2017

Sumário

System Calls

As System Calls fazer o interfaceamento entre o hardware e os processo do espaço de usuário, elas também servem para três propósitos principais

1. Provém abstração com o hardware para que o usuário tenha um maior rendimento. Por exemplo, o usuário não se preocupa com o tipo de partição em que ele lerá um arquivo

Second, system calls ensure system security and stability. With the kernel acting as a middleman between system resources and user-space, the kernel can arbitrate access based on permissions, users, and other criteria. For example, this arbitration prevents applications from incorrectly using hardware, stealing other processes' resources, or otherwise doing harm to the system. Finally, a single common layer between user-space and the rest of the system allows for the virtualized system provided to processes, discussed in Chapter 3, "Process Management." If applications were free to access system resources without the kernel's knowledge, it would be nearly impossible to implement multitasking and virtual memory, and certainly impossible to do so with stability and security. In Linux, system calls are the only means user-space has of interfacing with the kernel; they are the only legal entry point into the kernel other than exceptions and traps. Indeed, other interfaces, such as device files or /proc, are ultimately accessed via system calls. Interestingly, Linux 70 Chapter 5 System Calls implements far fewer system calls than most systems.¹ This chapter addresses the role and implementation of system calls in Linux.

As chamadas syscalls em linux são, na maioria dos casos, acessadas pelas funções definidas na C library. As syscalls em si retornam um valor do tipo long4 que, quando negativo, em geral, denota um erro, já o retorno com valor zero (nem sempre) é um sinal de sucesso. A C library, quando uma system call retorna um erro, ela grava o código desse erro na variável global errno, que pode ser traduzida para um texto que fala sobre o erro através de funções de biblioteca.