Gate OS: A Secure Rust Exokernel Based on Preventive Static Code Analysis With Process Domain Isolation

Lopuszanski, Franciszek (School: I Liceum Ogolnoksztalcace Dwujezyczne im. Edwarda Dembowskiego w Gliwicach)

Traditional operating systems use multiple abstraction layers that simplify development but restrict customization and optimization. Gate OS, a Rust-based exokernel, minimizes abstraction overhead, allowing applications to manage hardware directly while ensuring stability and security. This project explores the feasibility of exokernel architecture, static code analysis, and formal verification for process isolation and management, aiming to reduce reliance on the memory management unit (MMU) to improve performance and compatibility across architectures like RISC-V and ARM Cortex-M. Gate OS was developed in Rust to leverage its inherent code invariants. The system implements static analysis of system calls to ensure security policy compliance and uses the borrow checker to enforce thread boundaries without requiring an MMU. Benchmarks on x86-64 UEFI hardware measured I/O performance and networking throughput, while compatibility was tested by integrating partial POSIX and xv6 syscall support. Benchmarks showed performance gains over traditional kernel architectures, especially in high-throughput networking and fast storage. Static analysis successfully enforced compile-time boundaries, ensuring crash-safe runtime within applications and the kernel while preventing unauthorized access to resources. Gate OS demonstrates how programming language features and novel architectures can create secure, high-performance operating systems by enforcing security at compile time instead of runtime. It offers a promising alternative for embedded systems and networking devices by eliminating the need for virtualization and excessive privilege separation.

Awards Won: