論文の内容の要旨

論文題目　Performance Improvements in Isolated Monolithic Operating System Kernels for Dependability
（ディペンダビリティのための局限化単体ＯＳカーネルの性能向上）

氏　名　ドレベス・ホベルト・ユング

Operating systems are among the software components in computing systems where dependability is most critical. As they interact directly with the hardware, they are particularly prone to errors. This problem becomes even more serious with monolithic operating system kernels, where all subsystems and device drivers share the same address space and run under the same maximum privilege level. An error caused by a hardware or software fault in one kernel subsystem can easily propagate to others, corrupting their state and eventually leading the system to fail.

Even when kernels support loadable modules, components still run under the same execution domain as the main kernel. Module isolation techniques to address error propagation do exist, but they impose a performance overhead from the frequent domain switches when control flows from one protection domain into another.

This dissertation presents a technique to extract the relationship between kernel modules and to identify the potential error propagation paths between them. Through the inter-module relationship, a method to devise alternative configurations for module grouping under kernel isolation environments is proposed. These configurations group modules in respect to the functions they provide, so that the number of execution domains can be minimized to improve performance, while still maintaining error isolation between subsystems.

The proposed technique is evaluated into a real isolation environment for the Linux kernel. For these experiments, the performance overhead is extracted under different module group configurations, and the impact on dependability through fault injection is measured. For the dependability experiments, this dissertation also introduces Zapmem, a fault injection tool developed to test operating system kernels without instrumentation.

Experimental results show that grouping modules using the proposed technique can reduce the overhead by the isolation environment (time spent inside the kernel doing protection switches) from 5% to about 1.7% of the total execution time. Dependability results show that the severity of errors is also decreased: 50% of detected system crashes were manifested only as service errors under partial isolation, while full isolation could reduce them by 57%.