Paper summary

Exokernels are lightweight kernels to provide minimum abstractions to the programmer to utilize the hardware to their full potential. Unlike monolithic and microlithic kernels, which have fixed hardcoded implementation, exokernel achieves this minimum interference by exposing physical resources to userspace via library operating systems. According to the author, this helps achieve high performance and separates management from security. The Exokernel exports hardware resources securely to untrusted library os, which allows achieving fast and secure multiplexing, IPC primitives, control transfer primitives, virtual memory, scheduling than the state-of-the-art implementations. The prototype is demonstrated using Aegis as an exokernel and ExOS as an untrusted application operating system, modified to the application's functionality and performance needs.

Strengths

1. Detailed benchmark performance for the kernel and the application OS.

2. Paper ideas and novelty are presented lucidly.

3. The motivation for introducing modularity was clear.

Weaknesses

1. The paper acknowledges that Aegis, ExOS have lesser functionality than Ultrix but won't cause a significant increase in time measurement. How is it justified?

2. The author could more thoroughly explore the weaknesses of the system.

Comments for author

1. Is the performance improvement worth the level of complexity introduced? What do you think about the adaptability of the system in the real world? 2. What if the libOS are written in an unoptimized manner? Will the system retain its vision of security and performance if such unoptimized libOS are frequent in nature?