Title: Exokernel: an operating system architecture for application-level resource management
Authors: Dawson R. Engler, M. Frans Kaashoek and James O'Toole Jr.

Summary: In this paper, the authors present a detailed discussion of exokernels and a prototype implementation, Aegis, running the ExOS application operating system. An exokernel allows operating systems to provide application-level management of physical resources in the interest of performance, flexibility, and functionality by acting as a secure multiplexer of physical resources. The authors claim that the advantage of this approach over implementing an abstract virtual machine is that it makes possible domain specific optimizations, encourages changes to the implementation of existing abstractions, and gives application builders greater flexibility.

Key Ideas

An exokernel interface is very low-level while library operating systems, which work above the exokernel interface, implement higher-level abstractions that meet application needs and allow for application portability and compatibility. Using an exokernel, applications can: 1) securely bind to machine resources, 2) participate in a resource revocation protocol, 3) and allow for an abort protocol.

Secure binding: The secure binding mechanism allows an application to be authenticated once and then continue to access physical resources without compromising the system. Resource revocation: Visible resource revocation is necessary in order to allow the exokernel to reclaim physical resources and still avoid hiding information useful to a library operating system. Abort protocol: If an application refuses or fails to relinquish resources that the exokernel has attempted to reclaim, the exokernel initiates the abort protocol to forcibly break the secure bindings of a particular application to a given resource.

Flaws

The authors do not fully address the added complexity on the interface design given exokernel architecture. Another very important aspect is whether or not this can be extended to environments with malicious applications. Although the authors mention this as a possible future research endeavor, one cannot argue for the exokernel implementation without guarantee of this safety.

Relevance and Future Work

Current extensible OS projects (that may be relevant to exokernels): projects that use type-safe languages and software fault-isolation. Another current extensible OS project, the SPIN project, is a project that allows applications to make policy decisions which is complimentary to the exokernel design. Future research will investigate whether distributed control can be extended to an environment with malicious applications.

Sara Salahi
CS 443 - Paper Summary