SUMMARY OF Ph.D. DISSERTATION

School Science for Open and Environmental Systems	Student Identification Number	SURNAME, First name YAMADA, Hiroshi

Title

A Study on Changing Resource Management Policies from outside Operating System Kernels

Abstract

Developing resource management policies of the operating system (OS) is still an ongoing process. Because an appropriate resource management policy depends largely on the type of applications and their computing environments, OS researchers must continue to develop innovative resource management policies to satisfy the needs of emerging applications and everchanging computing environments. Despite the numerous innovative policies being proposed, it is quite difficult to widely deploy a single innovation. The traditional way to change resource management policies is to modify OS kernel source code. Modifying a kernel is a difficult, costly and often impractical endeavor. Modern OSes consist of large and complex bodies of code, and it is almost impossible to modify proprietary and closed source OSes. As a result, users cannot enjoy a new resource management policy until the integration of the policy in an OS, which is a time-consuming task, has been completed. To obtain effects similar to a new policy until an OS supports it, some research projects have explored the way to change resource management policies at the user-level without any modification of kernels. However, their approaches focus only on the policies that manage the resources whose state can be obtained at the user-level. Specifically, they just target the resources such as network bandwidth and CPU time, whose state is explicitly exposed by an OS. The resources such as disk bandwidth whose state cannot be observed at the user-level have been out of scope of the existing techniques. In addition, the policies whose execution is blind at the user-level, such as TCP congestion control policies, are still difficult to change.

In this dissertation, we propose techniques to expand the scope of policies feasible without modifying kernels. Our key insight is inference of OS behavior from outside the kernel. The dissertation first presents a user-level mechanism which enables us to control disk bandwidth usage. Disk I/O is not always accompanied by file I/O because the OS caches and reads disk blocks in advance. Our mechanism predicts disk I/O size from the file I/O size and making an inference of the OS behavior. We implemented the prototype on Linux 2.4.20 and demonstrate that we can roughly carry out a policy under which the disk bandwidth of user-specified processes is throttled to a given one.

The dissertation next describes a technique which brings effects similar to a desired policy by inferring OS behavior and using information unavailable at the user-level. Our technique uses virtual machine monitors (VMMs), which emulate the underlying devices for OSes. Our technique controls devices' behavior (e.g. modifies the value of data registers and delays requests from the OS) inferring the OS reaction against the transformed devices. By using this technique, we can make OS behavior similar to a desired policy. In fact, we can forge TCP Vegas, one of TCP congestion control policies, by modifying packets inside the network interface card. The technique also allows us to forge IdleTime scheduling, one of disk scheduling policies, by delaying OS disk requests inside the disk controller. We implemented these prototypes on Xen 3.0.2-2 and show that we can obtain benefits of the targeted policies.