Xen : Critique

Abhishek Srivastava
Student Id: 861307778

CS202 Advanced Operating Systems
March 10, 2017

Summary

The paper presents a virtual machine monitor called Xen, which multiplexes resources of a conventional system to tackle the challenges to build virtual machine such as Performance isolation, support for various platforms and add just small performance overhead.

Main Points

Authors proposes an approach of *paravirtualization* which overcomes the issue of *full virtualization* such as not able to access hardware properly due to x86 architecture, problems due to some privileged instructions and it did not provides real time guarantees. Xen paravirtualization provides exposure to underlying hardware in the system resulting into better performance, strong resource isolation, no need of modifying OS and applications.

Virtual machine interface contains *Memory management*, *CPU* and *Device I/O*. For memory management interface x86 does not support software managed TLB but rather page table structure in hardware is used. Allocation and management of hardware page tables are done by guests OSes with xen minimal involvement. Xen exists at top 64MB of every address space and avoiding TLB flush when guest OS enters or leaves Xen. Eacn guest OS maps memory to their own and only Writes are validated by Xen. CPU interfaces is supported using x86 4 level privilege rings. Xen restrain the privilege of guest OSes by setting it as ring 1. System call execution and page fault handling is handled by Xen, other exceptions are directly executed on x86 for fast handling. For Device I/O simple device abstractions are exposed by Xen.

For Control transfer between Xen and Domian is using *synchronous hypercalls* and *asynchronous events notifications*. Data Transfer is done using I/O Rings which used Zero copy semantics to provide protection between Guest OSes and I/O devices and reducing the work required to demultiplex data transfer. CPU scheduling is done using *Borrowed virtual time scheduling* which used low-latency wakeup mechanism. It favors recently woken domains which temporarily violates fair scheduling. Xen provides *Real time*, *Virtual time* and *Wall-clock time* to each guest Oses which they can use. For virtual address translation Xen provides direct MMU updates but in constrained manner. To increase performance updates are batched into single hypercall. For Physical Memory virtualization memory is statically partitioned among domains and are reserved at their creation time. Virtual firewall routers are attached to each domains To send a packet OS enqueues packet to the transmit ring, Xen uses simple round robin for packet scheduler. To receive packets Xen uses rules to determine the domains. Disks are also virtualized, Domain0 only has the direct access other domains uses Virtual block devices which shows associated ownership and access control information and I/O rings mechanism is used to access VBDs.

Limitations/Critique

The Drawbacks of Xen were it was only able to scale upto 100 VMs. Its performance of virtual address translation was poor in comparison to VMware and other industry level VM machines.

Proposed Extension

Use of *Shared universal buffer cache* on blocked contents, implementation of *last-chance page cache* and improvement to auditing and logging in *XenoServer* is some of the proposed extensions.