Xen and system administration

- Michael Bubb – CS765
- Prof. Jan Schaumann
- April 24, 2006
Xen project goals

- host up to 100 vms on modern and commonly available hardware

- Must:
  - support unmodded app binaries
  - support multiple and full OSes
  - workaround limitations of x86 arch
  - allow guest OS a window on underlying processes (while maintaining security)
Virtualization not new...

- IBM S/360 platform
- VM/CMS – Operating System
vmm and hypervisor

• Virtual machine
  • emulates host machine - gives the illusion of each user having their own private machine

• Hypervisor
  • supervisory layer – thin layer that exists below the Guest OSs
Virtualization

- Application virtual machine
- Hardware virtual machine
- Clusters
Hardware Virtualization Types

- Full virtualization
- Paravirtualization
- Native Virtualization
Popek and Goldberg
(Instructions)

1) Privileged instructions
2) Control sensitive instructions
3) Behavior sensitive instructions
•...[T]o build a VMM it is sufficient that all instructions that could affect the correct functioning of the VMM (sensitive instructions) always trap and pass control to the VMM. This guarantees the resource control property. Non privileged instructions must instead be executed natively...
x86 and Virtualization

per instruction virtualization not possible, must dynamically recompile privileged code
Xen on x86

- CPU
- Memory Management
- Device I/O
- Intel and AMD processors in dev to allow for full virtualization
Xen on x86: CPU

- x86 protection model. 4 Rings.
- Typically
- Ring 0 for the OS
- Ring 3 for user applications.
- 1 and 2 are not often used.
Xen on x86: Memory Management

- Paging
  - Guest OS read access to hardware page tables
  - hypervisor batches and validates updates
- Segmentation
  - cannot overlap top end of linear space
MMU Micro-Benchmarks

In bench results on Linux (L), Xen (X), VMW are Workstation (V), and UML (U) (slide from Ian Pratt’s “Xen and the Art of Virtualization”)
Xen on x86: Device I/O

- Virtual devices elegant and simple to access
- Data transfer asynchronous
- Event mechanism replaces hardware interrupts
Xen domains and guest OSes

- domain and guest OS analogous to a process and program
- domain refers to a running virtual machine
Dom0 and DomU
Xen for sysadmins

- Reduce hardware costs
- Allow for multiple OSes on same machine
- isolated (thus safe)
- rapid 'relocation'
Xen for sysadmins: Hardware

- XenoServer (parent project) conceptualized for everyday server hardware
  - intended for ISPs and Internet Exchanges
  - clusters and vms.... maximizing available resources
  - think of the early days of google
Xen for sysadmins: OS choices

• multiple OSes on same machine
  – allow for simultaneous development, staging and production
  – test software on different Oses
  – allow for different user needs
Xen for sysadmins: Isolation

- delegate permissions through vms
- every user a 'master of his/her own domain'
- instance crashes – rapid relocation
next slide has relocation data for a crash and restart of a mission critical app (Quake in this case)

Almost imperceptible lag
Packet interarrival time during Quake 3 migration

(slide from Ian Pratt’s “Xen and the Art of Virtualization”)
Conclusion

- excellent project philosophy
- accuracy of claims???
- many other projects
Bibliography

"Xen and the Art of Virtualization"

"Safe Hardware Access with the Xen Virtual Machine Monitor"
Keir Fraser, et al.

"A moment of Xen: Virtualize Linux to test your apps"
Bryan Clark, IBM website, 15 Mar 2005

"Xen and the Art of System Administration"
Johnny Lam, NYCBUG presentation 1 Feb 2006
http://www.netbsd.org/gallery/presentations/jlam/xen.html

"Xen Disk I/O Benchmarking: NetBSD dom0 vs Linux dom0"
Martti Kuparinen, (author's website) July 2005
http://users.piuha.net/martti/comp/xendom0/xendom0.html

"Introduction to the Xen Virtual Machine"
http://www.linuxjournal.com/node/8540

Wikipedia articles:
- Popek&Goldberg: http://en.wikipedia.org/wiki/Popek_and_Goldberg_virtualization_requirements