第 4 讲：Optimization of Virtual Machine Monitor

第二节：My VM is Lighter (and Safer) than your Container

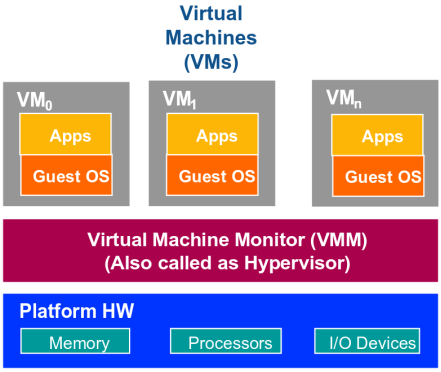
陈渝

清华大学计算机系

yuchen@tsinghua.edu.cn

2020 年 3 月 7 日

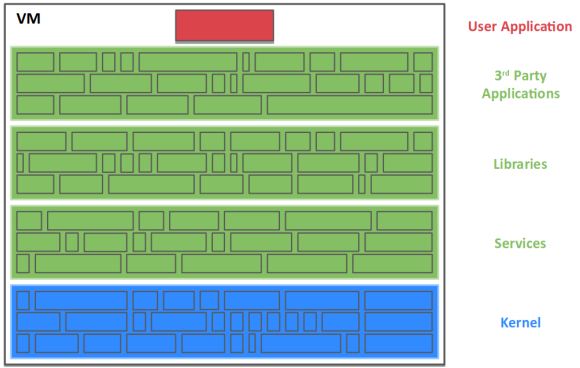
Problems of VMM



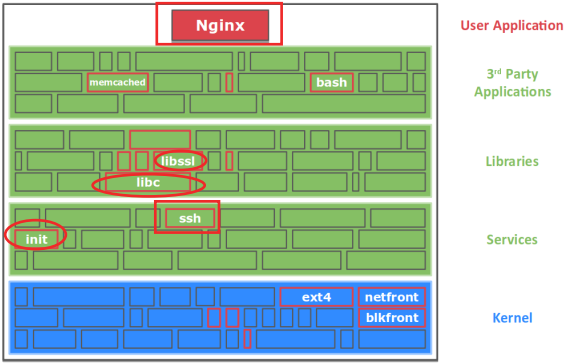
VMM is **SLOWER** than Container.

Problems of VMM

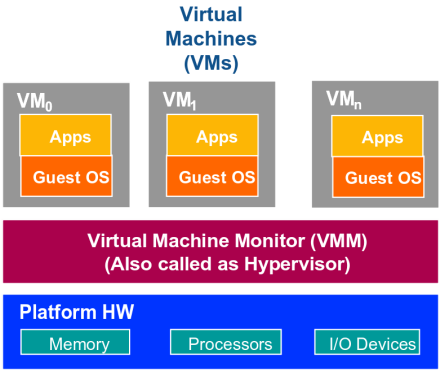
Standard VM: Application on Top of Distro



Most of the VM not Used...



Contributions of lightvm

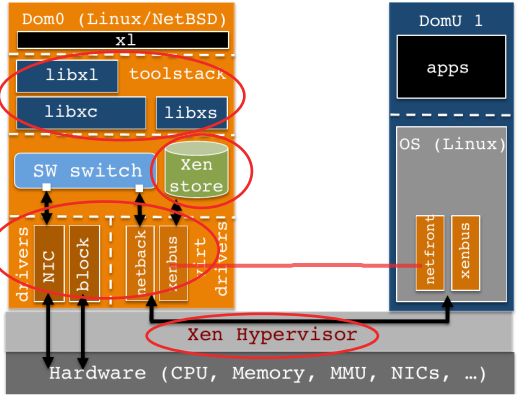


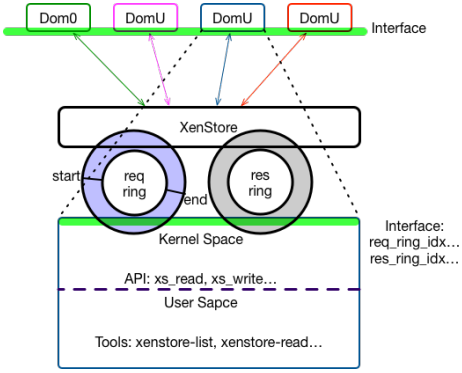
Contributions

- An analysis of the performance bottlenecks preventing traditional virtualization systems from achieving container-like dynamics
- An overhaul of Xen's architecture, completely removing its back-end registry
- A revamp of Xen's toolstack
- The development of Tinyx, an automated system for building minimalistic Linux-based VMs
- A prototypical implementation along with an extensive performance evaluation

Overview of XEN

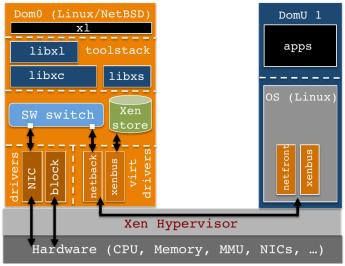
The Xen architecture including toolstack, the XenStore, software switch and split drivers between the driver domain (Dom0) and the guests (DomUs).



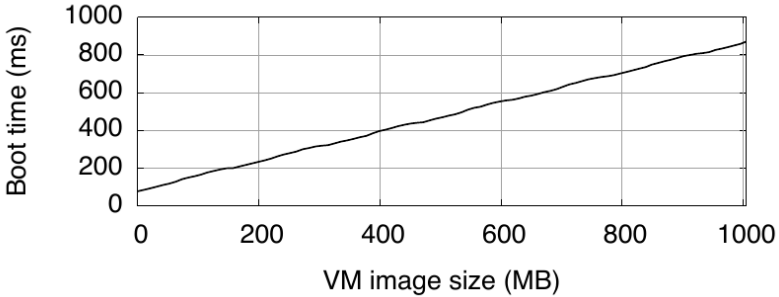


Some observations on XEN

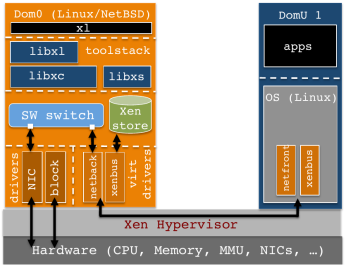
The Xen architecture including toolstack, the XenStore, software switch and split drivers between the driver domain (Dom0) and the guests (DomUs).



The size of the guest virtual machines

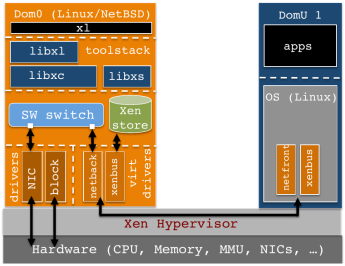


LIGHTWEIGHT VMs



- **Unikernels:** linking existing applications that rely on the Linux syscall API to Mini-OS is fairly cumbersome and requires a lot of expert time.
- **Tinyx:** is an automated build system that creates minimalistic Linux VM images targeted at running a single application

LIGHTWEIGHT VMs

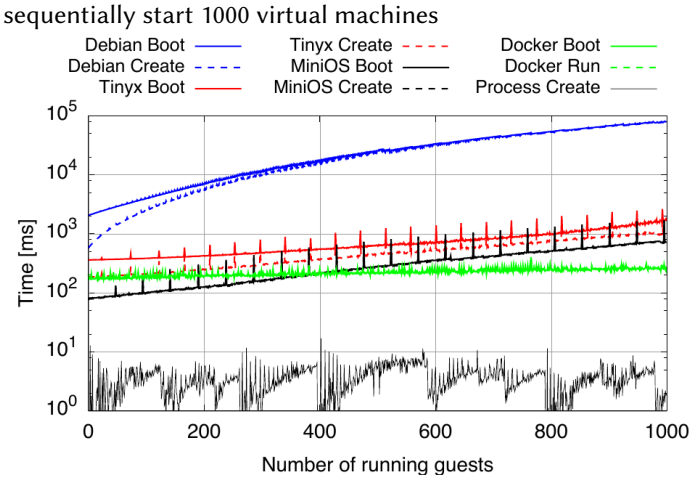


Tinyx:

- Inputs: an application to build the image for and the platform the image will be running on.
- Includes: the application, its dependencies, and BusyBox
- Methods: objdump for libs, ~~pkg manager for packages~~, tinyconfig for Linux kernel.
- Results: Tinyx create kernel images that are half the size of typical Debian kernels and minimal runtime memory usage
- (1.6MB for Tinyx vs. 8MB for the Debian)

Some observations on XEN

- The Debian VM is 1.1GB; it takes Xen around 500ms to create the VM, and it takes the VM 1.5 seconds to boot.
- The Tinyx VM (9.5MB) is created in 360ms and 180ms to boot. The unikernel (480KB) is created in 80ms, and 3ms to boot.
- Creating VMs, the creation time increases noticeably (note the logarithmic scale): it takes 42s, 10s and 700ms to create the thousandth Debian, Tinyx, and unikernel guest,

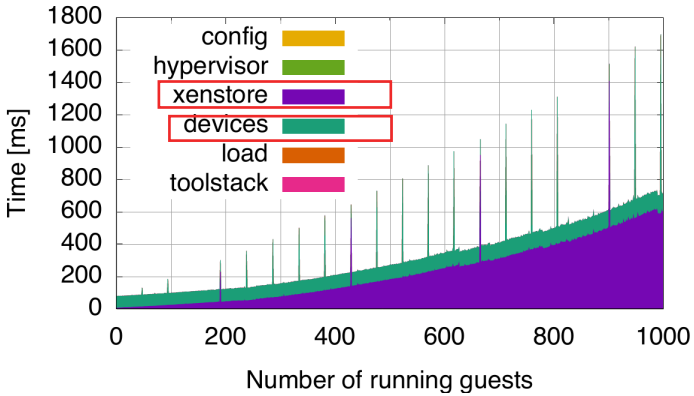


Some observations on XEN

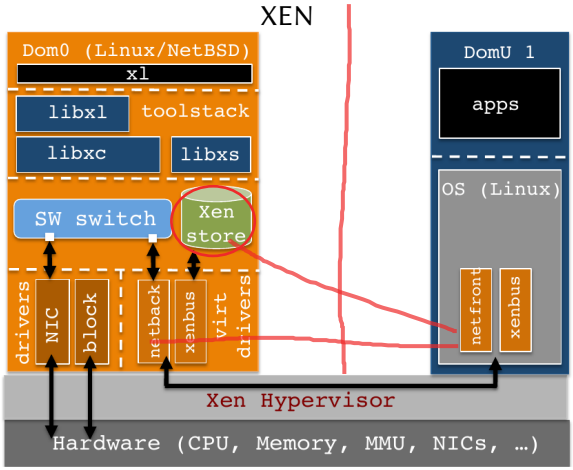
VM creation overheads

- Parsing the configuration file that describes the VM
- Interacting with the hypervisor for memory
- Reading/Writing information in XenStore
- Creating and configuration the virtual devices
- Parsing & loading the kernel image
- Other work on toolstack

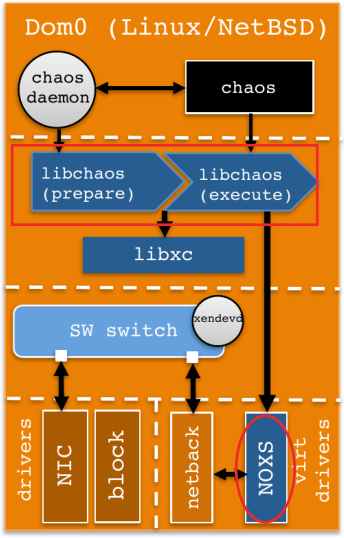
Breakdown of the VM creation overheads



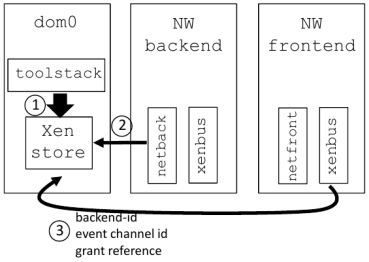
LightVM



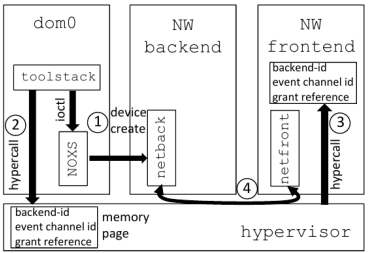
LightVM with Noxs and the Chaos Toolstack



LightVM – noxs



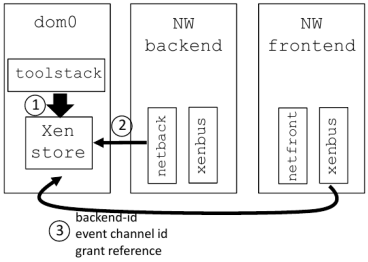
(a) XenStore



(b) noxs

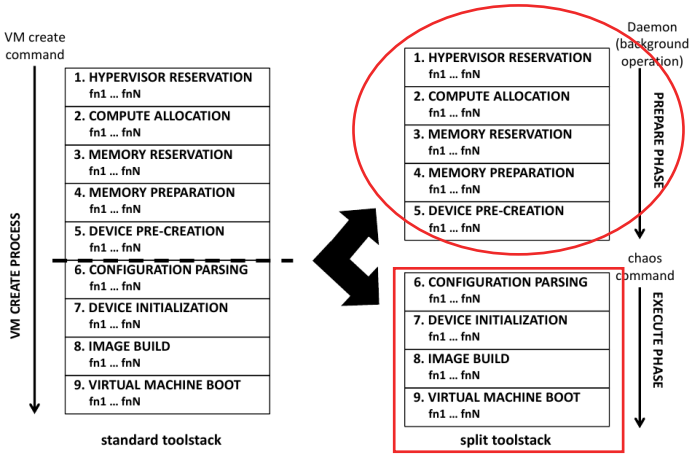
- fundamental problem with the XenStore is its ~~centralized~~, filesystem-like API
- noxs relies on shared pages mapped in the guest's address space
- create a new, special device memory page for each new VM that we use to keep track of a VM's information about any devices

LightVM – noxs



- toolstack requests the creation of devices from the back-end
- toolstack calls the new hypercall for adding these details to the device page
- New VM maps the device page into its address space
- front-end inits comm. with back-end
- finally front-end and back-end can notify each other

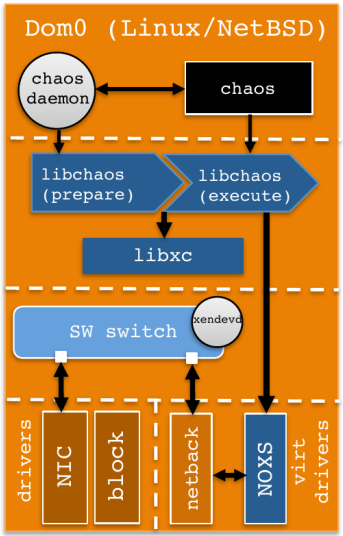
LightVM – split toolstack



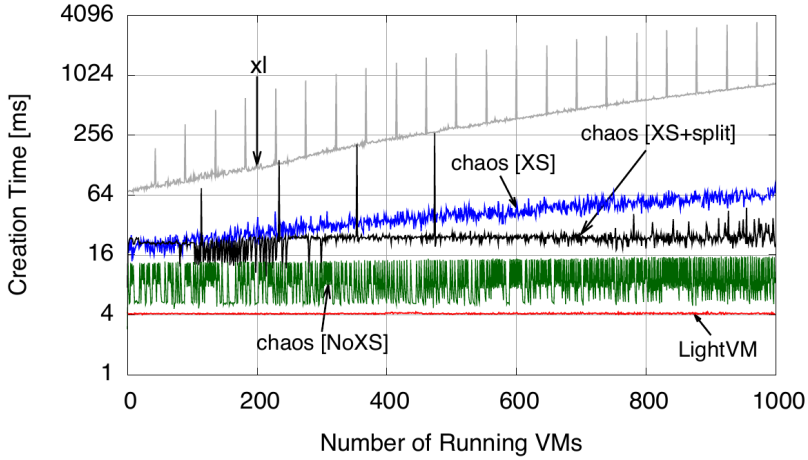
- Toolstack split between functionality belonging to the prepare phase, carried out periodically by the chaos daemon,
- offload this functionality to the chaos daemon, which generates a number of VM shells and places them in a pool.
- and an execute phase, directly called by chaos when a command is issued.

LightVM – Performance

LightVM

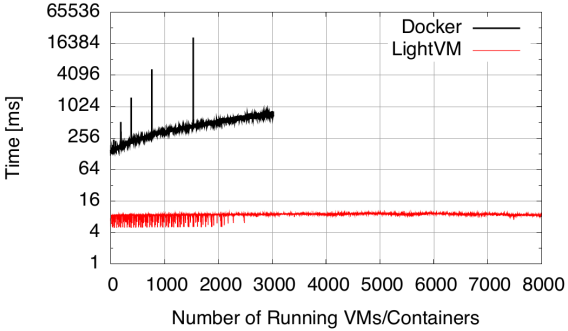


LightVM with Noxs and the Chaos Toolstack

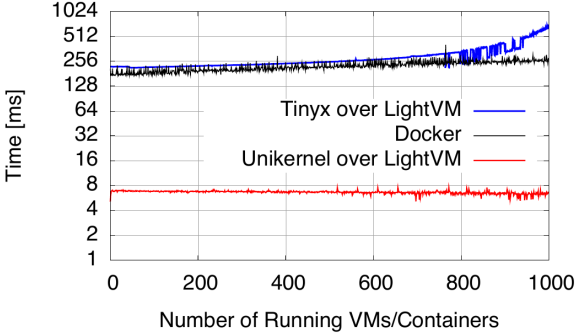


LightVM – Performance

LightVM boot times on a 64-core machine versus Docker containers.

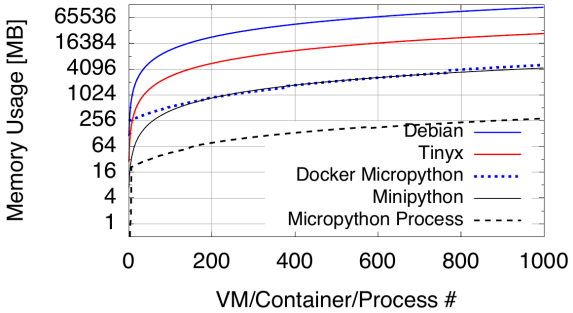


Boot times for unikernel and Tinyx guests versus Docker containers

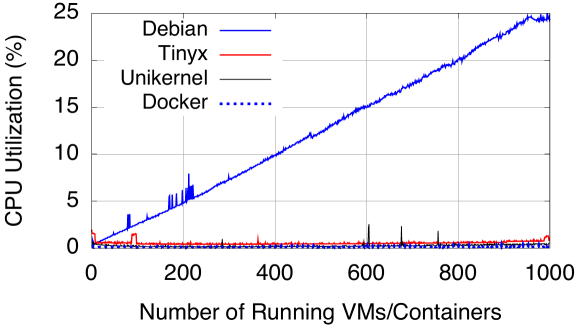


LightVM – Performance

Scalability of VM memory usage for different VMs, for containers and for processes.



CPU usage for a unikernel, Tinyx, a Debian VM and Docker.



LightVM – Summary

LightVM

- A complete redesign of Xen's toolstack optimized for performance.
- Can boot a minimalistic VM in as little as 2.3ms, comparable to the fork/exec implementation in Linux (1ms).
- Has almost constant creation and boot times regardless of the number of running VMs.
- Achieve both good isolation and performance on par or better than containers.