Kernel-based Virtual Machine

Amos Kong
Apr 12, 2012
kongjianjun@gmail.com

Agenda

- Background
- KVM Features
- Community
- Demo
- Q&A

Background(1/2)

IBM starts VT investigation on S360 in 1960s
os: CP-40

Paper: ARCHITECTURE OF VIRTUAL MACHINES
by R. P. Goldberg, July 1973

x86 VT extensions, intel: VMX, AMD: SVM
In mid of 2000s

VMware vSphere, MS Hyper-V, Citrix XenServer, RedHat KVM, Sun Virtualbox

Background(2/2)

Types:
- Hardware/platform
- Memory
- Storage
- Network
- Desktop

Goals:
- Virtual test environments
- Integration: config/manage/power saving
- Dynamical scalability
- Benefit from VT attributes (migration, snapshot)
- Abstract and share resource for better performance

KVM Features

Qemu project

Quick EMUlator (based on Bochs)
maintainor: Anthony Liguori

processor emulator
- dynamic binary(instruction) translation
- It's not used in KVM project

- support 44 architectures / kvm-tools
- device models emulation (usb/nic/disk/serial/...)

KVM project

KVM is a Hypervisor/vmm
- work between hardware and os
- allocate resource to VM

Benefit from Linux kernel features(ksm,cgroup)
/dev/kvm is a char device, access by ioctl()

Maintainor: Avi Kivity, Marcelo Tosatti
Company: Qumranet (purchased in 2008)
Linux-2.6.20, 2007 Feb

x86(64), s390, ppc, IA64, arm(in progress)

Cpu(1/4)

Guest vcpu:

- vm is a process
- vcpu is a thread
- use kernel schedule
- set priority by nice
- numa pin: avoids cross-node mem transports
- vcpu hotplug, max: 256

Cpu(2/4)

difficult to virtualize x86
- Can't trap some instructions expose privileged state
- Some privileged state can't be hiden

x86 processors virtualization extensions
- add guest operating mode
  trap privileged instructions
- vmcb/vmcs: register context switch
- VM exit, report reason
  Io handle in userspace

Cpu(3/4)

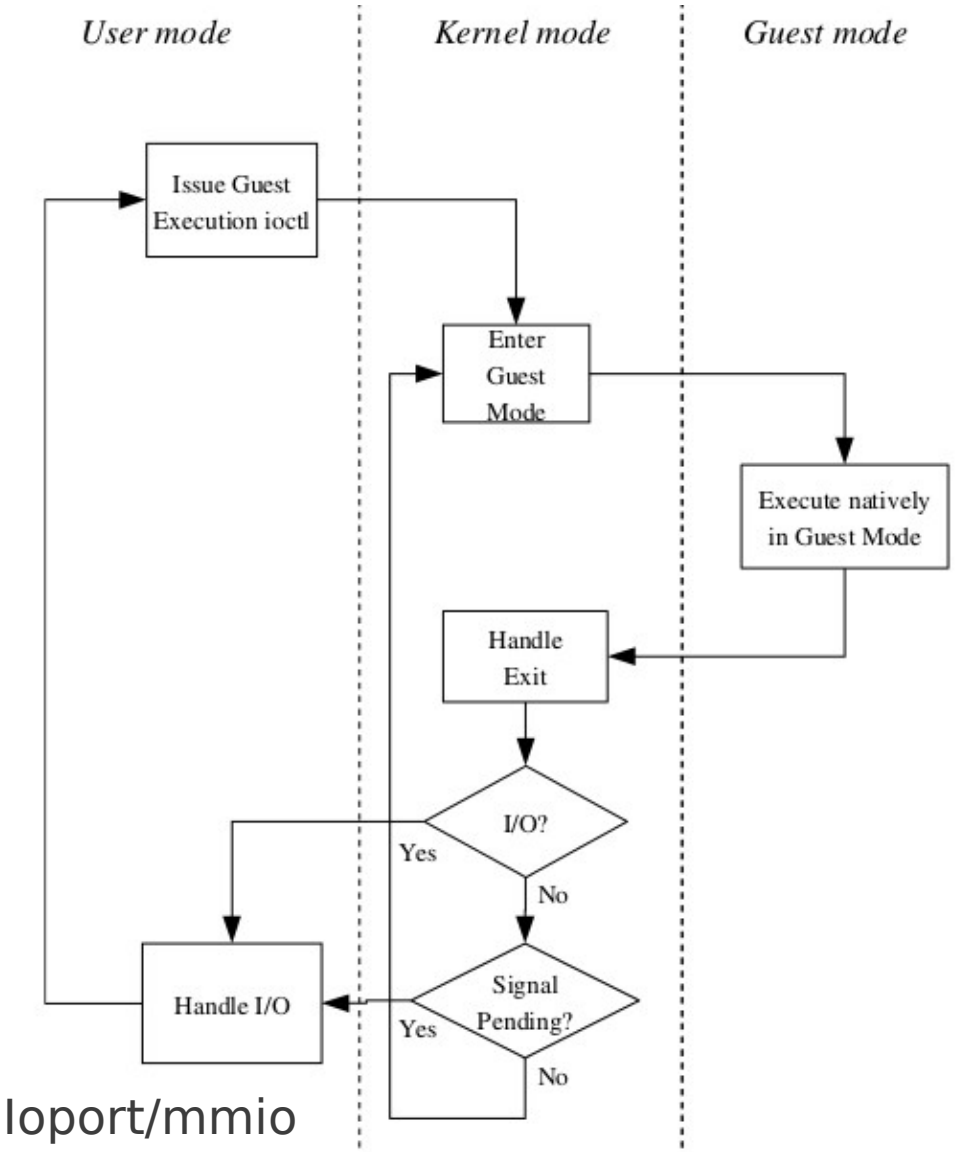
```
> cpus.c
  qemu_thread_create(env->thread, qemu_kvm_cpu_thread_fn, env,
  QEMU_THREAD_JOINABLE);

> kvm_all.c
kvm_init() :          kvm_ioctl(s, KVM_CREATE_VM, 0);
kvm_init_vcpu() : kvm_vm_ioctl(s, KVM_CREATE_VCPU, env->cpu_index);
kvm_cpu_exec() :
  do { kvm_vcpu_ioctl(env, KVM_RUN, 0);
            switch (run->exit_reason) {
            case KVM_EXIT_IO:
            case KVM_EXIT_MMIO:
  }  while();
```

Cpu(2/3)

difficult to virtualize x86
- Can't trap some instructions expose privileged state
- Some privileged state can't be hiden

x86 processors virtualization extensions
- add guest operating mode
  trap privileged instructions
- vmcb/vmcs: register context switch
- VM exit, report reason
  Io handle in userspace

Cpu(4/4) Guest Execution Loop



loport/mmio

Memory(1/2)

- malloc(): allocate memory when it's really used
- process mem swap
- virtio-ballon: resize guest memory

- MMU
  - shadow page table
  - track guest pte dirty
  - Guest doesn't change host pte

Memory(2/2)

KSM
- merge mem regions
- copy on write
- Ksmd daemon: scan pages
- Rb tree: O(log n)

mem overcommit
- isos/ images
- Sql cache

Para-VT

front/end driver (virtio-win)
cooperate with the hypervisor

- Net
- Blk
- Serial (guest agent)
- Ballon

Virtio-pci
- pci-bridge, multiple function
- pci hotplug

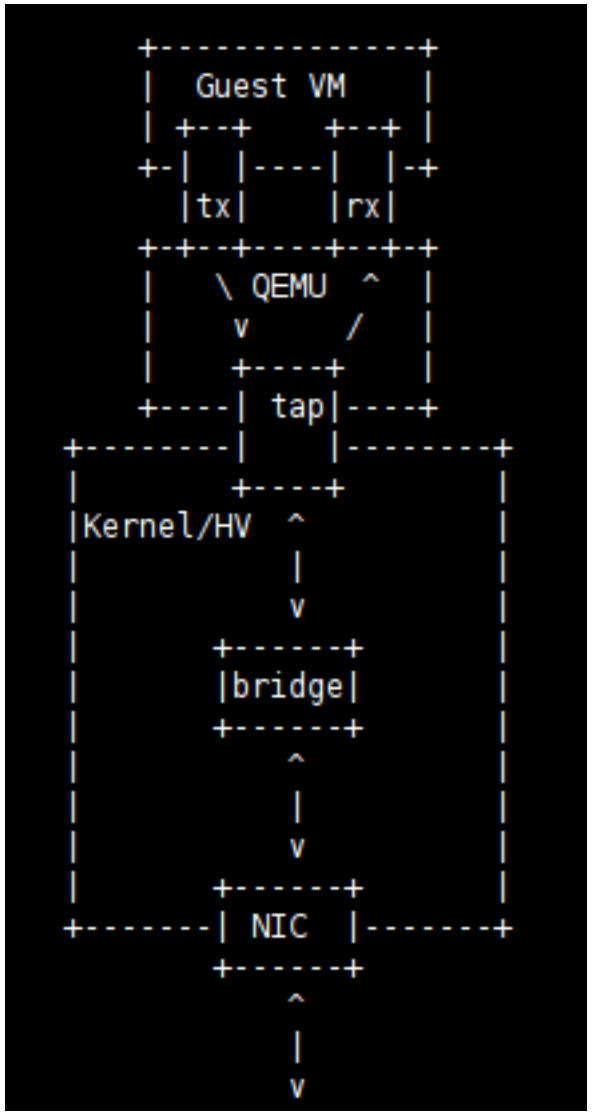
Network(1/3)

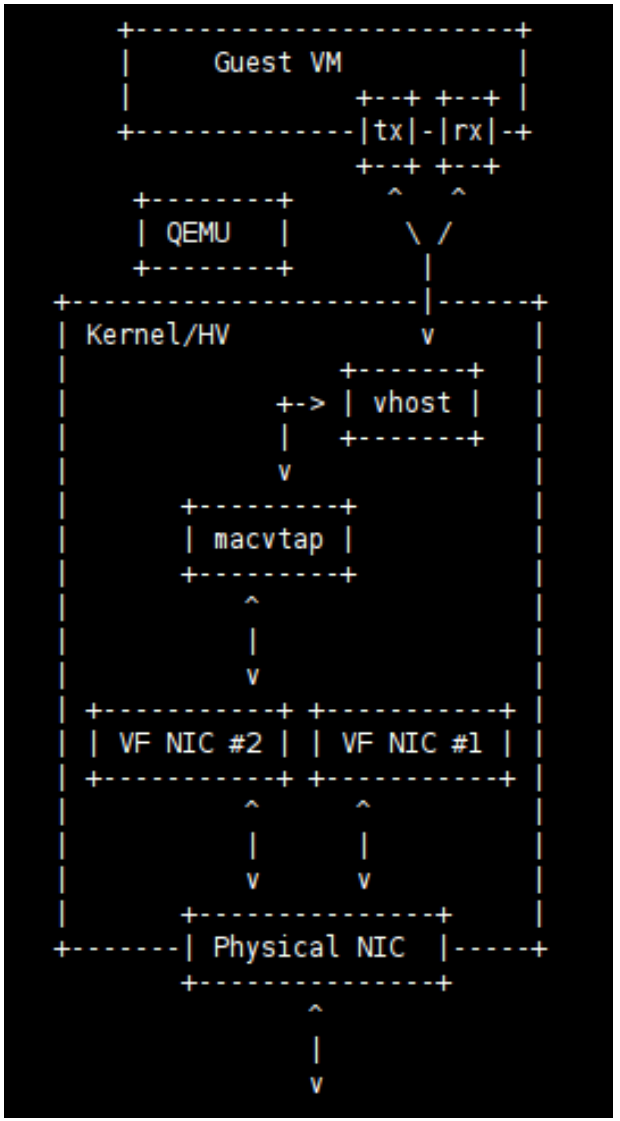
net mode
- userspace
- tap+bridge (public/private subnet)
- macvtap

vhost_net
- Reduce 4 syscalls per packet
  - vm exit for kick, reentry for kick, iothread wakeup for packet, interrupt injection for packet.
- /dev/vhost_net is a char dev
- need msix support: memory write transaction

hardware support
- SRIOV, 10Gb
- segment offload
  - E1000
  - GSO/TSO/GRO/LRO
  - Virtio-net: GRO for host nic

Network(2/3)





Network models

Network(3/3)

zero copy
- pin guest userspace
- host nic dma it
- short io path

Management(1/3)

- Spice/vnc
- Qemu monitor
- QMP
  - JSON-based protocol
  - async event
- Live migration / Live block copy

Management(2/3)

Libvirt server/client
- libvirt.org
- parse fd to child process
- Xml interface

Support:
qemu-kvm, xen, vmware esx, openvz
c,python, c#, java, perl

Management(3/3)

oVirt project
- ovirt.org
- VDSM
- Web management tool

Demo

KVM guest

Start a KVM guest:

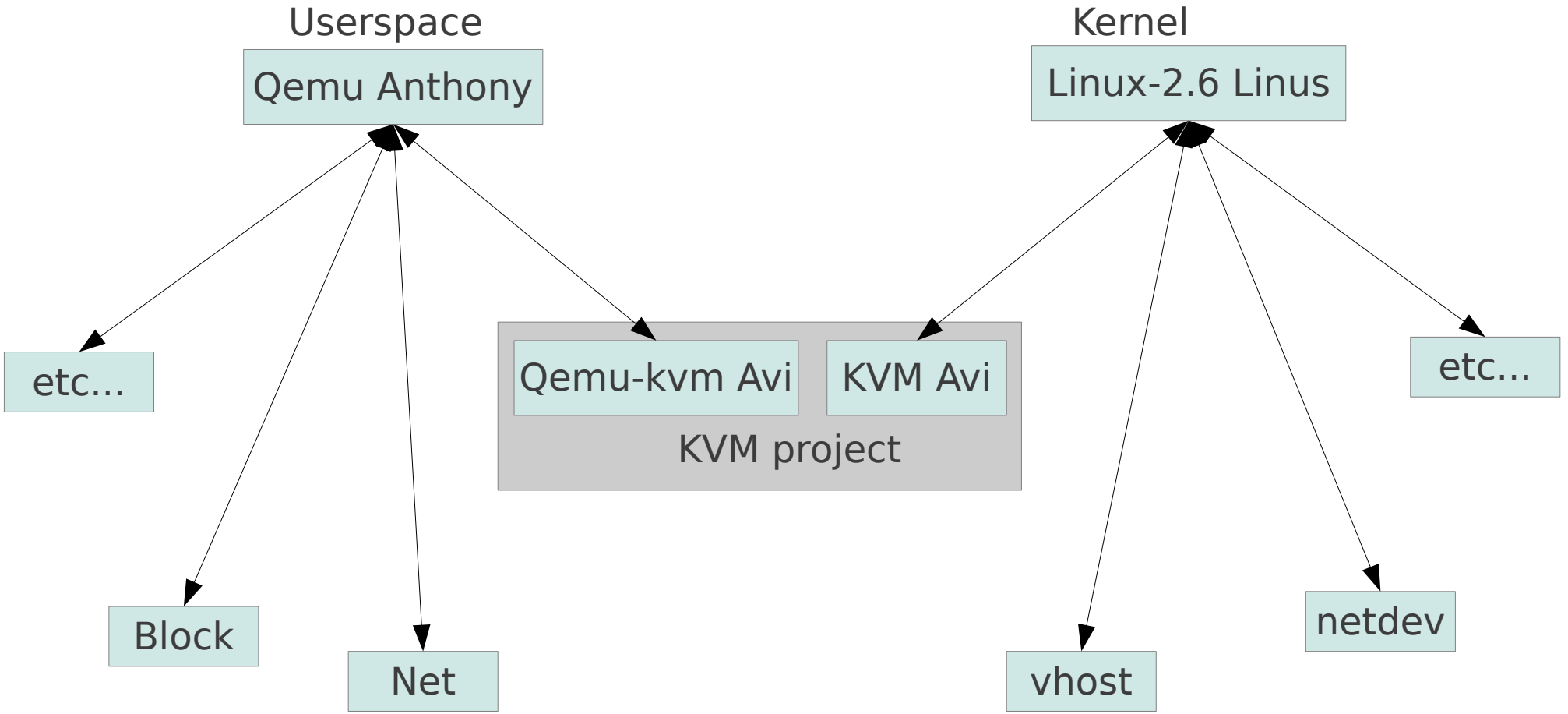
```
# qemu-img create -f qcow2 vm.qcow2 10G
# qemu-kvm vm.qcow2 -m 1024 -net nic -net tap -vnc :0 -serial stdio
# kvm_stat

(qemu) # info status
(qemu) # stop

guest) # lspci |grep Eth
guest) # lsmod |grep virtio
guest) # mount /dev/vdb1 /mnt
guest) # cat /proc/cpuinfo
```

Kernel debug

Debug guest kernel by gdb server inside qemu

- compile kernel (CONFIG_DEBUG_KERNEL=y)
- lunch guest with gdb parameters
  (host)# qemu-kvm -gdb tcp::1234 ...
  (host)# gdb linux-2.6/vmlinux
  (gdb) target remote localhost:1234
  (gdb) bt

Community(1/3)



Community(2/3)

Maillist:

- kvm@vger.kernel.org
- qemu-devel@nongnu.org

IRC:

- #kvm on Freenode.net
- #qemu on Oftc.net

Bugzilla

- https://bugs.launchpad.net/qemu

Community(3/3)

Website:

- http://www.linux-kvm.org
- http://www.qemu.org

Patchwork:

- http://patchwork.ozlabs.org/project/qemu-devel/list/

Pepo:

- git://git.qemu.org/qemu.git
- git://git.kernel.org/pub/scm/virt/kvm/qemu-kvm.git
- git://git.kernel.org/pub/scm/virt/kvm/kvm.git

Reference

- kvm: the Linux Virtual Machine Monitor – by Avi Kivity
- IBM and HP virtualization - Ken Milberg

Question & Answer

xiyoulinux@googlegroups.com