QEMU Machine emulator and virtualizer

CONCEPTS

One binary per CPU architecture (qemu-system-x86_64, -aarch64, ...); without -accel kvm the guest is emulated and slow.

Host Linux, /dev/kvm
- **qemu-system-*** one process per VM
  - **Machine** -machine, -cpu, -m, -smp
    - **Devices** -drive, -netdev, -device

Options are order-independent but -netdev and -chardev ids must match the -device or -serial that uses them.

The guest's disk is a file on the host; qcow2 grows on demand and supports snapshots and backing files.

RUN A VM

qemu-system-x86_64 -hda *<disk>*
Boot from a disk image

-accel kvm
Hardware acceleration (Linux)

-cpu host
Pass the host CPU model through

-m *<size>*
Guest RAM, e.g. 2G

-smp *<n>*
Number of virtual CPUs

-machine q35
Pick a modern chipset

DISKS AND BOOT

-drive file=*<disk>***,format=qcow2,if=virtio**
Attach a virtio disk

-cdrom *<iso>*
Attach an optical image

-boot d
Boot from the CD first

-boot order=*<cdn>*
Set boot device order

DISKS AND BOOT (cont.)

-snapshot
Discard disk writes on exit

-drive if=pflash,format=raw,readonly=on,file=*<OVMF_CODE.fd>*
UEFI firmware (OVMF)

-kernel *<bzImage>* **-append "***<cmdline>***"**
Boot a kernel directly

DISPLAY AND CONSOLE

-nographic
No window; serial on the terminal

-serial mon:stdio
Serial and monitor on the terminal

NETWORK

-nic user
Simple NAT, no setup (default)

-nic user,hostfwd=::*<host>***-:***<guest>*
Forward a host port to the guest

-nic tap,script=no,ifname=*<tap>*
Attach to a host tap device

DEVICES AND SHARING

-device usb-host,vendorid=*<vid>***,productid=***<pid>*
Pass a host USB device through

-virtfs local,path=*<dir>***,mount_tag=***<tag>***,security_model=mapped-xattr**
Share a folder with the guest (9p)

QEMU-IMG

qemu-img create -f qcow2 *<disk>* *<size>*
New disk image

qemu-img info *<disk>*
Show format and sizes

qemu-img convert -O qcow2 *<in>* *<out>*
Convert between formats

qemu-img resize *<disk>* **+***<size>*
Grow an image

QEMU-IMG (cont.)

qemu-img create -f qcow2 -b *<base>* **-F qcow2** *<overlay>*
Overlay on a backing file

qemu-img commit *<overlay>*
Merge overlay into its base

qemu-img check *<disk>*
Check for corruption

qemu-img snapshot -l *<disk>*
List internal snapshots

qemu-img snapshot -c *<tag>* *<disk>*
Create a snapshot

Never run qemu-img write commands on an image a running VM uses; info -U is the read-only exception.

Common image formats

-f / -O	Use for
raw	Plain bytes, fastest, no features
qcow2	Sparse, snapshots, backing files
vmdk	VMware images
vdi	VirtualBox images
vhdx	Hyper-V images

MONITOR

Ctrl+a c
Switch console / monitor (mon:stdio)

Ctrl+a x
Exit the emulator

info status
Running or paused?

stop / cont
Pause / resume the guest

savevm *<tag>*
Save full VM state (needs qcow2)

loadvm *<tag>*
Restore VM state

system_powerdown
Send an ACPI power button press

RECIPES

Install an OS

```
qemu-img create -f qcow2 vm.qcow2 20G
qemu-system-x86_64 \
  -accel kvm -cpu host -m 4G \
  -drive file=vm.qcow2,if=virtio \
  -cdrom <install.iso> -boot d
```