Test Disk Images with QEMU

Easy to install, easy to use
Just run QEMU on the command line

TÜBIX 2016, 11 June, Gerik Huland

There is Virtualbox and free to download VMware Player.

Why QEMU?

Why QEMU?

VMware is closed source. Free to use for private persons only

Virtualbox is Free License for the core, closed source extension pack

Qemu is the first really free hypervisor for Linux (Fixme), started somewhere 2003 by math genius Fabrice Bellard

Production grade virtualization, featured by e.g. Redhat

QEMU

- Supports emulation of various hardware platforms e.g.
  - ARM
  - SPARC
  - PowerPC
  - MIPS
- On x86_64 and x86 support for KVM (Kernel Virtual Machine)
  - later ported to other processor platforms
  - requires processor with Intel VT-x or AMD-V
  - allows for paravirtualized devices
  - massive speedup
  - performance loss appr. 10-30%]

Great variety of hardware emulations

- Base emulation is Intel I440FX + PIIX (1996, really ooold!)
- More recent emulation is Intel Q35 + ICH9 (2009), still considered unstable
- SMP up to 255 CPUs
- USB up to XHCI
- Various graphics controllers
- Paravirtualized block, network, SCSI, graphics controller
- PCI and USB passthrough

Ever Wanted to Test Your Custom USB Boot Stick?

It's so easy!

```
sudo apt-get install qemu-kvm
sudo qemu-system-x86_64 -m 512M /dev/sdb
```

Pitfalls

Always set the virtual machine's memory, default is only 128MiB!

This is not enough for most modern Linux's initrd to unpack! .e.g `-m 256M`

Use kvm for speed. Calling `qemu-kvm` defaults to

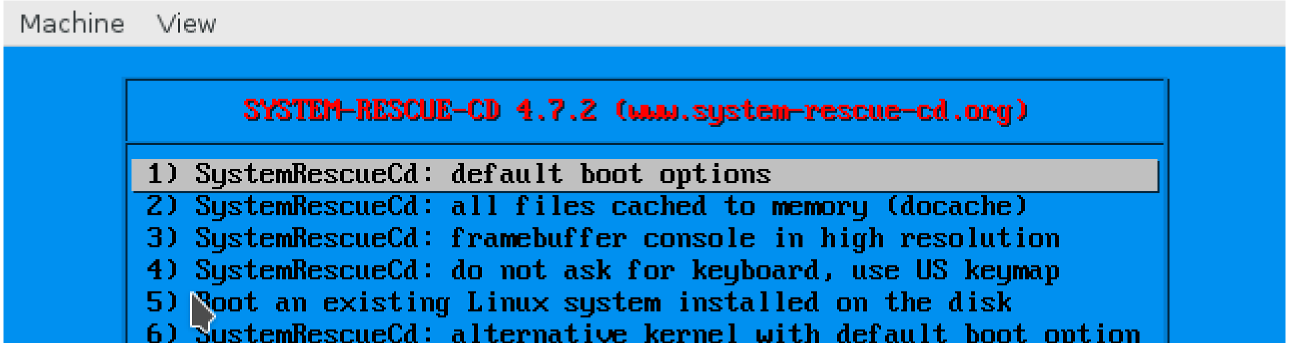
```
qemu-system-<your current arch> --enable-kvm
```

Define your keyboard if not US: `-k de`

Boot from ISO image

If you own the image, no need to be root!

```
qemu-kvm -m 512M -cdrom sysresccd.iso
```



```
SYSTEM-RESCUE-CD 4.7.2 (www.system-rescue-cd.org)

1) SystemRescueCd: default boot options
2) SystemRescueCd: all files cached to memory (docache)
3) SystemRescueCd: framebuffer console in high resolution
4) SystemRescueCd: do not ask for keyboard, use US keymap
5) Boot an existing Linux system installed on the disk
6) SystemRescueCd: alternative kernel with default boot option
```

Networking

- with no options: NATed e1000 ethernet adapter
- equals `-net nic -net user`
- use `-net none` to disable networking altogether
- attaching to VLANs or bridges requires calling helper script
- consider using libvirt with virt-manager for more elaborated network setups

Using VMware or other disk images

Use qemu-img to convert disk images

```
qemu-img --help
...
Supported formats: parallels nbd vvfat blkdebug file rbd qcow2 vhdx
vpc null-aio raw null-co host_cdrom bochs host_floppy gluster sheepdog
iscsi vdi host_device quorum ssh vmdk blkverify qcow tftp ftp ftps
https dmg http cloop qed
```

Pass through USB devices

`qemu-kvm -usb -usbdevice host:vendor_id:product_id`

e.g. pass through your Lego brick to your Windows virtual machine:

```
# lsusb
Bus 002 Device 003: ID 8086:0189 Intel Corp.
Bus 002 Device 004: ID 0694:0005 Lego Group
Bus 002 Device 002: ID 8087:0024 Intel Corp. Integrated Rate Matching Hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 003: ID 0408:2fb1 Quanta Computer, Inc.
Bus 001 Device 002: ID 8087:0024 Intel Corp. Integrated Rate Matching Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

Start up Windows 7 image on Q35 platform with 2 cpus and 2GB RAM

`sudo qemu-kvm -machine q35 -smp 2 -m 2G -usb -usbdevice host:0694:0005 windows7.img`

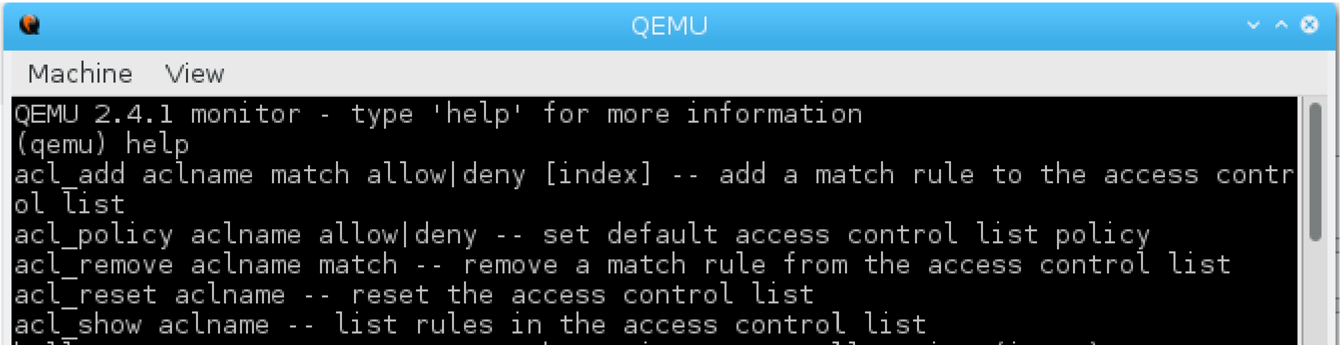
Boot with UEFI instead BIOS

```
apt-get install qemu-efi
```

```
qemu-kvm -machine q35 -smp 2 -m 2G -L /usr/share/OVMF -bios
/usr/share/qemu/OVMF.fd my_uefi_boot.img
```

Use the konsole to do things in a running virtual machine

- Ctrl+Alt+2
- In the window's menu it's compatmonitor0



```
QEMU 2.4.1 monitor - type 'help' for more information
(qemu) help
acl_add aclname match allow|deny [index] -- add a match rule to the access contr
ol list
acl_policy aclname allow|deny -- set default access control list policy
acl_remove aclname match -- remove a match rule from the access control list
acl_reset aclname -- reset the access control list
acl_show aclname -- list rules in the access control list
```

QEMU konsole

- send special key combinations `sendkey ctrl-alt-f1`
- stop, cont (inue)
- add drives and devices (OS must handle hotplugging!)

Really mean things you can do

No liability if you kill your installation

Boot the Linux on another partition of your system disk

```
qemu-kvm -m 512M -drive file=/dev/sda,if=virtio
```

This takes you to your bootloader. Never boot the currently running system!

- To be on the safe side:
  Use dmsetup to stitch up a virtual disk containing the partition you want to boot

And Now: Happy Q-Emulating!

Slideshow created using remark.