KVM performance tuning

15.02.25

Pegasi Knowledge
https://ghost.pegasi.fi/wiki/

Table of Contents

KVM performance tuning

I used to have XEN (para)virtualization and compared to that I notice KVM to be a lot more resource hungry and not easily optimized. I am using CentOS 6.2 which should be the best with KVM but at least for now Windows 7 64bit seems to be run slower than Windows 2000 VMWare guest with 25% of the resources.

BUT.. I made some modifications and it started to fly. Here is my receipe for good life with KVM Windows.

Increase memory

I upped the guest memory to 12G so the disk operations got minimized

Use Virtio block and network drivers from Fedora

Very well functioning especially with block devices

Use cache=none OR cache=writeback

Preferably the first since it is safer in user

Use io=native

This should give some extra boost

Ditch the file image based storage

Raw image will not work stable with good cache settings and Windows guest. Best if you transfer the image to a block device with following procedure. Lets assume we have a raw image file and a nice SSD disk in /dev/sdd that is free to use for our guest.

```
fdisk /dev/sdd
```

Create HPFS/NTFS primary partition, bigger than the image file

```
dd if=yourkvmimagefile.img of=/dev/sdd1 bs=1024k
```

The device is now boot ready but if you want to increase the size of your partition continue to following

```
fdisk /dev/sdd
```

Delete and create again to your wanted size, if needed.

```
kpartx -a /dev/sdd1
```

```
ntfsresize --info /dev/mapper/sdd1XXX
```

Look at current volume and device sizes and if needed resize with command

```
ntfsresize -s <newsize> /dev/mapped/sdd1XXX
```

Now change the virtual machine definition with

```
virsh edir yourvirtualmachine
```

And edit the element

```
    <disk type='file' device='disk'>
      <driver name='qemu' type='raw' cache='writethrough'/>
      <source file='/var/lib/libvirt/images/myserver.img'/>
      <target dev='hda' bus='virtio'/>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x03'
function='0x0'/>
    </disk>
```

To look like

```
    <disk type='block' device='disk'>
      <driver name='qemu' type='raw' cache='none' io='native'/>
      <source dev='/dev/sdd1'/>
      <target dev='vda' bus='virtio'/>
      <address type='pci' domain='0x0000' bus='0x00' slot='0x04'
function='0x0'/>
    </disk>
```

Tune my Linux host network

Increasing the right buffers gave me around 20-30% increase in 1G LAN thoughput. Simply put edit /etc/sysctl.conf to include following

```
net.core.wmem_max=8388608
net.core.rmem_max=8388608
```

And do testing without rebooting with following commands

```
echo 8388608 > /proc/sys/net/core/wmem_max
echo 8388608 > /proc/sys/net/core/rmem_max
```

And no, implementing jumbo frames in my servers and my HP switch did not make any real difference.

This made my KVM guest fly and host can barely see the load. I can truly say it is like a different piece of hardware compared to image based system.