Keep a limit on it
IO Throttling in QEMU

Ryan Harper – ryanh@us.ibm.com
Open Virtualization
IBM Linux Technology Center

August 15, 2011
Contributors

- Zhi Yong Wu – wuzhy@linux.vnet.ibm.com
- Stefan Hajnoczi – stefanha@linux.vnet.ibm.com
- Karl Rister – krister@us.ibm.com
- Khoa Huynh, Ph.D. – khoa@us.ibm.com
- Steve Pratt – spratt@us.ibm.com
Limited Resources
cgroup blkio controller

- Proportional
  - Bw or IOPs
  - Requires CFQ

- Bandwidth
- IOPs

Upper limits per block device
Non host block access

http

curl

NFS

ceph

Network Access

Database Servers
QEMU Block Layer Limits

QEMU Hardware emulation

bdrv_aio_{read,write}v

Exceed limits?

Yes

Queue request

Calculate wakeup

No

drv->bdrv_aio_{read,write}v
Block IO Throttle Comparison

- **Effectiveness**
  - Can your configuration be throttled?
  - Is the cap ever exceeded?
  - What amount of IO does the guest observe?

- **Cost**
  - Is there a substantial cost to implement throttling?
    - If so, where is that cost incurred?
Block IO Throttle Configuration

- **Storage backends**
  - LVM over SATA disk
  - EXT4 over SATA disk
  - NFS (IBM n3600)

- **Image Formats**
  - RAW
  - QCOW2

- **Host Cache mode**
  - ,cache=none
  - ,cache=writethrough

- **Block Limiting**
  - cgroup blkio throttling
  - QEMU blk-throttle
Workloads

- 5 different workloads
  - streaming writes
    - mkfs.ext4
  - random reads and writes
    - fio iometer with randrw mix
  - random reads
    - fio aio-read
  - random writes
    - fio aio-write
  - streaming reads
    - fio disk-surface-scan
- 1 and 5VM instances, isolated and mixed
- VMs have 50G virtio-blk device

Host Config

- IBM System x iDataPlex dx360 M3
  - 2x Intel X5670 @ 2.93GHz
  - 128G RAM
  - 5 2TB SATA
  - 2 1G Intel NIC
  - 1 10G Emulex NIC
- RHEL 6.1
- ioscheduler=deadline
CFQ vs Deadline

CFQ vs. Deadline I/O Schedulers
FFSB Benchmark, LVM w/ 8 Disk Arrays, I/O Block Size = 8KB

- Direct I/O w/ Deadline
- Direct I/O w/ CFQ
- Direct I/O w/ CFQ+slice_idle=0+quantum=32

Average FFSB Throughput (MB/sec)

- Large File Creates
- Sequential Reads
- Random Reads
- Random Writes
- Mail Server

IBM

Penguin
Cgroup vs QEMU - IOPs

cache=none

cgroup blkio limited
qemu blk limited

![Graph showing IOPs comparison between Cgroup and QEMU with cache=none configuration.]
Cgroup vs QEMU - Throughput

cache=none

cgroup blkio limited
qemu blk limited

seqread-64k-compare-ext4-raw-cachenone-cgroupcapped-qemucapped-throughput
Cgroup vs QEMU – Throughput

cache=writethrough

cgroup blkio limited
qemu blk limited
QEMU Capped vs Uncapped cache=none, nfs-backed

seqread-64k-compare-nfs-raw-cachenone-qemucapped-uncapped

uncapped
qemu blk limited
QEMU Capped vs Uncapped -- Throughput

cache=none, nfs-backed

uncapped
qemu blk limited
QEMU Capped vs Uncapped -- IOPs

cache=none, nfs-backed

![Graph showing IOPs over time for QEMU Capped vs Uncapped]
Throttling Cost -- utilization

CPU Utilization

Type

utilization

idle
Work per %cpu

Throttling Overhead
Throughput

<table>
<thead>
<tr>
<th>uncapped</th>
<th>cgroup capped</th>
<th>qemu capped</th>
</tr>
</thead>
<tbody>
<tr>
<td>4000.00</td>
<td>10.00</td>
<td>30.00</td>
</tr>
</tbody>
</table>

IOPs

<table>
<thead>
<tr>
<th>uncapped</th>
<th>cgroup capped</th>
<th>qemu capped</th>
</tr>
</thead>
<tbody>
<tr>
<td>60.00</td>
<td>40.00</td>
<td>20.00</td>
</tr>
</tbody>
</table>
Next Steps

- Algorithm improvements
  - Focus on preventing spikes
- Reduce CPU consumption
  - Data are incomplete but suggests there is room for improvement
Questions?

- http://wiki.qemu.org/Features/DiskIOLimits