Resource Management with CGroups

Steven Ellis
Solution Architect
Red Hat New Zealand
sellis@redhat.com

CGroups

- What
  - are CGroups
- Why
  - do I need CGroups
- How
  - do I use CGroups

What are CGroups

- In Kernel capability to limit, account and isolate resources
  - CPU
  - Memory
  - Disk I/O
- Originally developed by Rohit Seth in 2006 under the name “Process Containers”
- Kernel merge in 2.6.24, now included in most major distributions

CGroup Controllers

- **memory**: Memory controller
  - Allows for setting limits on RAM and swap usage and querying cumulative usage of all processes in the group
- **cpuset**: CPU set controller
  - Binding of processes within a group to a set of CPUs and controlling migration between CPus
- **cpuacct**: CPU accounting controller
  - Information about CPU usage for a group of processes
- **cpu**: CPU schedular controller
  - Controlling the priorization of processes in the group. Think of it as a more advanced nice level
- **devices**: Devices controller
  - Access control lists on character and block devices

CGroup Controllers

- **blkio**: I/O controller for block devices
  - Sets limits on input/output access to and from block devices such as physical drives (disk, solid state, USB, etc.).
- **freezer**:
  - Suspend or resumes tasks
- **net_cls**: Network Class controller
  - Tags network packets so the Linux traffic controller can identify packets with a particular CGroup

Why

- SLA Management
  - Meet application SLAs by reducing resource contention and increasing predictability in performance.
- Large Virtual Consolidation
  - Prevent single or group of virtual machines monopolising resources., or impacting other environments.
- Cost based accounting
  - Cost recovery from business units

When

- When did I first need CGroups?

- Even I was surprised

2009/2010

- Mobile Messaging Company
- Database backups were impacting production services.
- Very I/O intensive queries
  - and insufficient spindles
- Global Coverage
  - Scheduling backups was was a major issue.

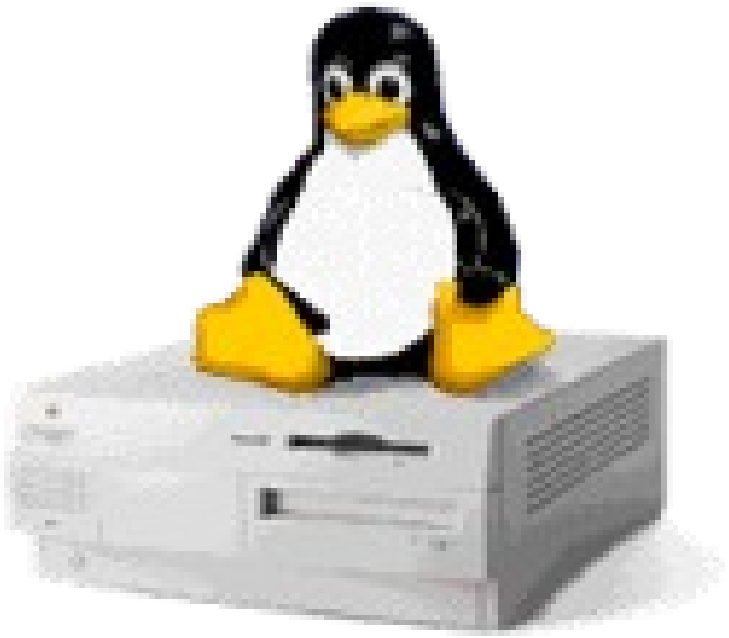
2006

- Multiple Websphere JVM's on shared hardware.
- Memory leaks and CPU peaks were impacting other business services.

2003-2004

- Multi Vendor JVM testing
- Red Hat Enterprise Linux
  - Releases 2.1 and 3
- JRockit had some interesting performance characteristics
  - Can not SSH into host.
  - Physical TTY would timeout before login.
  - Power button wouldn't work.

Too Far back



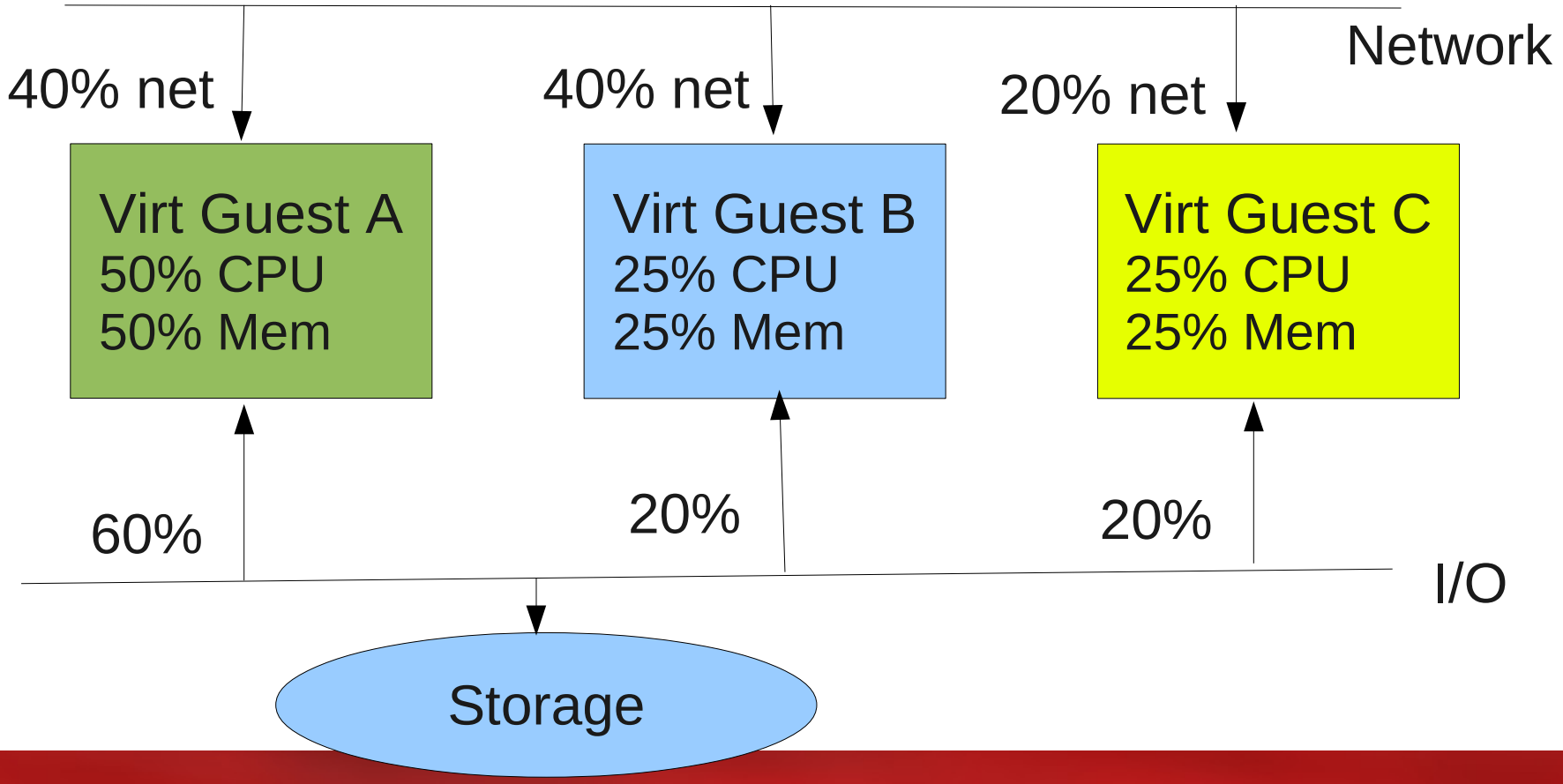
1998-1999

- UK Based Development House
- Red Hat 5.0 Primary Server
  - IMAP email
  - Samba / AFS / NFS
  - GNU Cross-Compilers
  - GDB over serial for Embedded Hardware
  - Remote X via VNC to Mac Workstations
- Boss isn't getting his email

Modern Challenge - Virtualisation

Hosting providers requires QoS

(quality of service guarantees based on pricepoint)



How

Using CGroups

- Install cgroups support
  - yum install libcgroup
  - apt-get install cgroup-bin libcgroup1
- Setup a basic /etc/cgconfig.conf

```
mount {
    cpuset = /cgroup/cpuset;
    cpu = /cgroup/cpu;
    cpuacct= /cgroup/cpuacct;
    memory = /cgroup/memory;
}
```

- Start the cgroups daemon
  - service cgconfig start

Command Line Tools

- cgexec
  - Start new process in specified group(s).
- cgclassify
  - Move process to specified group(s).
- cgcreate / cgdelete
  - Create and remove cgroups manually
- cgset
  - Modify defined cgroup

Subsystems - memory

- Limit memory usage of **processes** in a group
- Parameters (see memory.txt):

  `memory.limit_in_bytes` – maximum allowed memory usage by tasks in the group.

  `memory.max_usage_in_bytes` – maximum of used memory.

  `memory.stat` – current memory statistics (RSS, swap, ...)
- Examples:
  - HTTP can take only 30% of memory.

Subsystems - cpu

- Set scheduler priority.
- Parameters:

  `cpu.shares` – priority of threads in this group, relative to other groups.
- Example:
  - SQL can take 2x more CPU cycles than HTTP.

Subsystems - cpuacct

- Computes CPU cycles, burned by members of the group.
- Parameters:

  `cpuacct.usage` – nr. of cycles.

  `cpuacct.usage_percpu` – nr. of cycles per CPU.
- Example:
  - Members of 'developers' used 10^7 cpu cycles.
  - Out of that, only $2x10^6$ cpu cycles were exhausted by mySQL.

Apache Example

- Edit /etc/cgconfig.conf

```
group http {
    memory {
        memory.limit_in_bytes = 1024M;
    }
}
```

- Next, add this to the /etc/sysconfig/httpd.conf:

```
CGROUP_DAEMON="memory:/http"
```

- The start cgconfig service and httpd

CGroups and Virtual Machines

- Allows to control libvirtd and any other process in the cgroup “virt”
  - Examples are memory ceiling / capping
  - Restrict which CPUs libvirt can utilise
- Add these rules to /etc/cgconfig.conf

```
group virt {
    memory {
        memory.limit_in_bytes = 3.5G;
    }
    cpuset {
        cpuset.cpus = 1-3;
    }
}
```

- Modify /etc/sysconfig/libvirtd and add

```
CGROUP_DAEMON="memory:/virt”
```


Subsystems - blkio

- Manages block and char I/O devices
  - proportional weight-based disk access
    - Weight from 1-1000
  - Upper limit throttling
    - Specify a fixed number of bps per device

Blkio Demo

- # restart cgroups
- service cgconfig restart

- # Setup the throttle as zero and then play with it.
- cd /cgroup/blkio/
- echo 253:0 $((0*1024*1024)) > blkio.throttle.write_bps_device

- #Monitor I/O with
- iostat dm-0 3

- # Then start DD on the volume
- while true; do dd if=/dev/zero of=/tmp/test.out; done

- # Adjust blkio throttle and check iostart output
- echo 253:0 $((20*1024*1024)) > blkio.throttle.write_bps_device
- echo 253:0 $((10*1024*1024)) > blkio.throttle.write_bps_device
- echo 253:0 $((1*1024*1024)) > blkio.throttle.write_bps_device

References

- RHEL 6 Resource Management Guide
  - http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Resource_Management_Guide
- Fedora Overview
  - http://fedoraproject.org/wiki/Features/ControlGroups
- Manage Your Performance with Cgroups and Projects
  - http://broadcast.oreilly.com/2009/06/manage-your-performance-with-cgroups-and-projects.html
- Zonker at ServerWatch on Cgroups
  - http://www.serverwatch.com/tutorials/article.php/3920051/Introduction-to-Linux-Cgroups.htm
- Using Cgroups under Debian
  - http://hydra.geht.net/tino/english/faq/debian/squeeze/cgroups/

Images

- Clipart
  - http://www.cksinfo.com/clipart/electronics/phones/mobilephones/camera-phone.png
- Corporate logos (c) current owners.
- Pointy Headed Boss c/o Scott Adams
  - Used under fair use, please contact me if you want this removed.