Throttling VM I/O

This is a slightly edited page from the internal wiki of the SCALE/SWITCHengines team. We provide it "as is" in the hope of it being useful. Note the date! (December 2014)

Throttling VM I/O (aka Poor Man's Storage QoS)

Story

We should be able to configure safe limits on I/O rates of VMs, so that SWITCHengines users no longer have to suffer slow I/O when other VMs perform intensive tasks such as disk benchmarks.

Technical Background

libvirt/KVM can limit the rates of I/O that a domain (VM) is allowed to perform on a given block device. This can be done at other levels as well, e.g. using cgroups as described in "Throttling I/O with Linux".

This feature is interesting to us, because we have seen in the past that a single VM doing heavy I/O - for example when running a file system benchmark - can make the cluster slow for everyone.

So it would be nice if we could limit VMs' I/O to safe values.

Apparently Juno allows I/O throttling to be configured in VM flavors. However it also seems that this is not supported for RBD (where we need it) yet. Although it seems there's a fix.

Suggested Policies

If we had this option, we'd still need to decide how we want to set the limits. It is tempting to define a “worst-case” limit that would prevent the cluster from being overloaded even when all users try to torture it. For example, we could divide the overall I/O capacity of the Ceph cluster by the maximum number of VMs, and use the result as per-VM limits. But that would be bad because it would slow everybody down under normal circumstances.

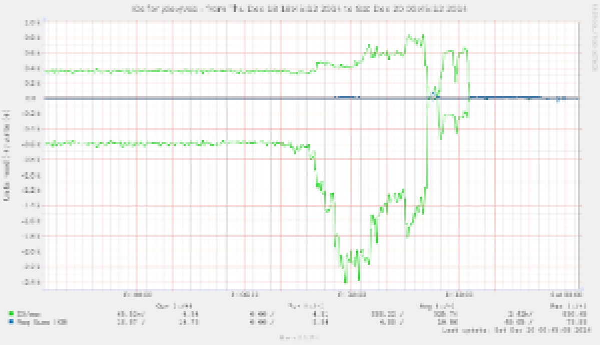
Optimistically Safe

A better option would be to set the per-VM limits so that a single VM (or a small group) cannot overload the entire cluster. This is based on the observation that, usually, overload comes from a small number of clients.

On-Demand Limiting

Another option is to leave I/O unlimited by default, but notice when there is overload and limit heavy-I/O VMs after the fact.

Hack alert:



In a way, we can already do this in an ad-hoc kind of way:

- Locate the VM that tortures the cluster
  - Find out which nova-compute host it runs on, e.g. `zhdk0028`
  - Find out which libvirt domain name it uses, e.g.`instance-000789ab`
- Log on to the nova-compute host
- Limit I/O by hand to a rate deemed safe, e.g. 800 I/O op/s per volume:

```
leinen@zhdk0028:~$ virsh blkdeviotune instance-000789ab vda --total-iops-sec 800
leinen@zhdk0028:~$ virsh blkdeviotune instance-000789ab vdb --total-iops-sec 800
```

- Note that these limits will get lost if the VM is rebuilt for some reason.

openstack/throttle_io.txt · Last modified: 20.12.2014 00:52 by Simon Leinen