

Cluster sizing

NDS Labs

Looking at the defaults for Kubernetes on GCE and AWS, it looks like even for very large clusters (hundreds of nodes), they use a single Master node – only the machine [size](#) changes. There is no mention of installing etcd on a separate server, although they do install a separate etcd for events. The default compute node size is n1-standard-2 (2 VCPU, 8 GB RAM).

The Gluster documentation suggests a minimum of 2 servers.

For testing NDS Labs, we can setup a small cluster:

- 1 master/etcd: r2.medium (2 VCPU, 8GB RAM)
- 2 compute: n-rcd1.large
- 2 gfs: r2.medium

Kubernetes

GCE sizing

The file `kubernetes/cluster/gce/config-common.sh` contains the following:

- NUM_NODES=5, suggested_master_size=2 (2 VCPU, 7.5 GB RAM)
- NUM_NODES=10, suggested_master_size=4 (4 VCPU, 15 GB RAM)
- NUM_NODES=100, suggested_master_size=8 (8 VCPU, 30 GB RAM)
- NUM_NODES=250, suggested_master_size=16 (16 VCPU, 60 GB RAM)

This is actually the node size. There's no indication that either GCE and AWS use multiple masters, just bigger masters – even with hundred-node setups.

`config-default.sh`

- NODE_SIZE=n1-standard-2
- NUM_NODES=3
- MASTER_DISK_SIZE=20GB
- NODE_DISK_SIZE=100GB

The GCE setup in Kubernetes does not indicate separate servers for etcd, but they do mention using a separate etcd for events.

See also

<http://kubernetes.io/docs/admin/cluster-large/>

[Heapster scalability testing](#)

[GET operations do not always work on 500 node GCE cluster](#)

[API status for pods is flaky](#)

<http://kubernetes.io/docs/admin/high-availability/#master-elected-components>

<https://github.com/kubernetes/kubernetes/blob/release-1.2/docs/proposals/scalability-testing.md>

<http://blog.kubernetes.io/2015/09/kubernetes-performance-measurements-and.html>

<http://blog.kubernetes.io/2016/03/1000-nodes-and-beyond-updates-to-Kubernetes-performance-and-scalability-in-12.html>

<https://github.com/kubernetes/kubernetes/blob/release-1.2/docs/devel/node-performance-testing.md>

https://github.com/kubernetes/kubernetes/blob/release-1.2/test/e2e/kubelet_perf.go

<http://kubernetes.io/docs/admin/high-availability/>

<https://coreos.com/blog/improving-kubernetes-scheduler-performance.html> – mentions "separate etcd cluster"

Gluster

<http://www.gluster.org/community/documentation/index.php/QuickStart>

"Have at least two nodes"

Failover

Even though the client appears to be mounting from a single server, apparently the client is smart enough to get information about [available peers during mount](#).

Performance monitoring

http://www.gluster.org/community/documentation/index.php/Gluster_3.2:_Monitoring_your_GlusterFS_Workload