

Hands on Virtualization with Ganeti

OSCON 2011

Setup Guide

This setup guide covers installing and running **Ganeti** and **Ganeti Web Manager**. We'll be using three VirtualBox images to simulate a 3-node cluster using DRBD. You can download the images from here: <http://ftp.osuosl.org/pub/osl/ganeti-tutorial/>. You need to download both **node1.example.org.ova** and **node2.example.org.ova** files. Downloading the **node3.example.org.ova** file is *optional* and is primarily for walking through various hardware failure scenarios. If your machine is 32bit only, you may want to download the 32bit images found in the **i386** folder in the URL above. VirtualBox can be downloaded and install from here: <http://www.virtualbox.org/wiki/Downloads>. We highly recommend you get these images downloaded and setup **BEFORE** arriving at OSCON. Please only complete the *first* step by importing the images as we'll cover the rest at the tutorial.

The VirtualBox images have been pre-configured using puppet manually so that very little will need to be downloaded from Internet during the tutorial. All of the required package dependencies have already been installed, Ganeti Web Manager (GWM), Ganeti Instance Image, and Ganeti Htools have also been installed.

Tarballs of all the code we're working on is located in `/root/src`, and a symlink to the puppet module has been created at `/root/puppet`. The root password for the Debian images and deployed instances is **oscon**. Much of this hands-on tutorial is based on the Ganeti Documentation site which you're free to look at during this tutorial. Be aware that their examples assume we deployed xen while we are using kvm/qemu instead.

Puppet module: <http://github.com/ramereth/puppet-ganeti-tutorial>

Ganeti Documentation: <http://docs.ganeti.org/ganeti/current/html/index.html>

Installing Ganeti

1. Importing VirtualBox Images

Make sure you have hardware virtualization enabled in your BIOS prior to running VirtualBox. You will get an error from VirtualBox while starting the VM if you don't it enabled.

1. Download images
2. Start VirtualBox
3. File → Import Appliance → *select node image*
4. Start Appliance

2. Accessing node1/node2/node3

The VM nodes are accessible via local ports on your machine. You can *either* ssh directly to them using ssh **or** simplify it by adding the details to your ssh client config. The root password for both nodes is **oscon**.

```
# node1
ssh -p 9000 root@localhost
# node2
ssh -p 9001 root@localhost
# node3
ssh -p 9002 root@localhost
```

Modifying your ssh client config.

```
vim ~/.ssh/config

Host node1
  Hostname localhost
  Port 9000
```

```
User root
```

```
Host node2  
Hostname localhost  
Port 9001  
User root
```

```
Host node3  
Hostname localhost  
Port 9002  
User root
```

Additional local port forwards that are setup on node1 *only*:

- **8000** points to GWM once its setup
- **8888** points to the VNC Auth Proxy for GWM
- **5800-5805** points to the VNC ports used by the vnc proxy
- **843** points to the flash policy server used for GWM

3. Updating the VM config state

We use puppet to simplify the deployment process. In case we've updated the puppet config after you downloaded the image, you need to update the puppet repository and run puppet to update the config.

NOTE: change file path in step 4 for node2/node3.

1. `ssh -p 9000 root@localhost`
2. `cd /root/puppet`
3. `git pull`
4. `puppet apply nodes/node1.pp`

4. Installing Ganeti

We've already installed ganeti for you on the VMs, but here are the steps that we did for documentation purposes.

```
puppet apply /root/puppet/nodes/node1/install-ganeti.pp
```

Alternatively, you can manually install Ganeti too.

1. `ssh -p 9000 root@localhost`
2. `cd src`
3. `tar -zxvf ganeti-2.4.2.tar.gz`
4. `cd ganeti-2.4.2`
5. `./configure --localstatedir=/var --sysconfdir=/etc && /usr/bin/make && /usr/bin/install`
6. `cp doc/examples/ganeti.initd /etc/init.d/ganeti && chmod +x /etc/init.d/ganeti`
7. `update-rc.d ganeti defaults 20 80`

5. Initialize Ganeti

Use puppet to initialize ganeti. This only needs to be done on node1 once.

```
puppet apply /root/puppet/nodes/node1/initialize-ganeti.pp
```

Alternatively, you can manually initialize Ganeti. Be aware that Ganeti is very picky about extra spaces in the “-H kvm:” line.

```
gnt-cluster init \  
  --vg-name=ganeti -s 192.168.16.16 \  
  --master-netdev=br0 \  
  -H kvm:kernel_path=/boot/vmlinuz-2.6-kvmU,initrd_path=/boot/initrd-2.6-kvmU,root_path=/dev/sda2,nic_type=e1000,disk_type=scsi,vnc_bind_address=0.0.0.0,serial_console=true \  
  -N link=br0 \  
  --enabled-hypervisors=kvm \  
  ganeti.example.org
```

6. Add second node

1. Fire up the second node
2. Complete steps 3 & 4
3. ssh to node1
4. `gnt-node add -s 192.168.16.17 node2`

NOTE: We'll add node3 later in the tutorial but feel to import it now.

Managing Ganeti

Testing the cluster

```
root@node1:~# gnt-cluster verify
Tue Jul  5 03:52:33 2011 * Verifying global settings
Tue Jul  5 03:52:33 2011 * Gathering data (2 nodes)
Tue Jul  5 03:52:34 2011 * Gathering disk information (2 nodes)
Tue Jul  5 03:52:34 2011 * Verifying node status
Tue Jul  5 03:52:34 2011 * Verifying instance status
Tue Jul  5 03:52:34 2011 * Verifying orphan volumes
Tue Jul  5 03:52:34 2011 * Verifying orphan instances
Tue Jul  5 03:52:34 2011 * Verifying N+1 Memory redundancy
Tue Jul  5 03:52:34 2011 * Other Notes
Tue Jul  5 03:52:34 2011 * Hooks Results
```

```
root@node1:~# gnt-node list
Node          DTotal DFree MTotal MNode MFree Pinst Sinst
node1.example.org 25.4G 25.4G  497M 127M 391M    0    0
node2.example.org 25.4G 25.4G  497M  74M 437M    0    0
```

Adding an Instance

```
root@node1:~# gnt-os list
```

```
Name
```

```
image+debian-lenny
```

```
image+default
```

```
root@node1:~# gnt-instance add -n node1 -o image+debian-lenny -t plain -s 5G --no-start instance1
```

```
Tue Jul 5 03:58:22 2011 * disk 0, vg ganeti, name bc69699b-f8aa-4f7a-84de-3cb3e97ed7cf.disk0
```

```
Tue Jul 5 03:58:22 2011 * creating instance disks...
```

```
Tue Jul 5 03:58:23 2011 adding instance instance1.example.org to cluster config
```

```
Tue Jul 5 03:58:23 2011 - INFO: Waiting for instance instance1.example.org to sync disks.
```

```
Tue Jul 5 03:58:23 2011 - INFO: Instance instance1.example.org's disks are in sync.
```

```
Tue Jul 5 03:58:23 2011 * running the instance OS create scripts...
```

Listing Instance information

```
root@node1:~# gnt-instance list
```

Instance	Hypervisor	OS	Primary_node	Status	Memory	
instance1.example.org	kvm	image+debian-lenny	node1.example.org	ADMIN_down		-

```
root@node1:~# gnt-instance info instance1
```

```
Instance name: instance1.example.org
```

```
UUID: 9c82680d-bd83-40f7-8d04-290cf2d54f72
```

```
Serial number: 1
```

```
Creation time: 2011-07-05 03:58:23
```

```
Modification time: 2011-07-05 03:58:23
```

```
State: configured to be down, actual state is down
```

```
Nodes:
```

```
- primary: node1.example.org
```

```
    - secondaries:
Operating system: image+debian-lenny
Allocated network port: 11000
Hypervisor: kvm
...
Hardware:
  - VCPUs: 1
  - memory: 128MiB
  - NICs:
  - nic/0: MAC: aa:00:00:ac:d2:d5, IP: None, mode: bridged, link: br0
Disk template: plain
Disks:
  - disk/0: lvm, size 5.0G
  access mode: rw
  logical_id:  ganeti/bc69699b-f8aa-4f7a-84de-3cb3e97ed7cf.disk0
  on primary:  /dev/ganeti/bc69699b-f8aa-4f7a-84de-3cb3e97ed7cf.disk0 (254:0)
```

Controlling Instances

```
root@node1:~# gnt-instance start instance1
Waiting for job 9 for instance1.example.org...
```

```
root@node1:~# gnt-instance console instance1
```

```
Debian GNU/Linux 6.0 instance1 ttyS0
```

```
instance1 login:
```

Press **ctrl+]** to escape console.

```
root@node1:~# gnt-instance shutdown instance1
Waiting for job 29 for instance1.example.org...
```

Changing the Disk Type

```
root@node1:~# gnt-instance modify -t drbd -n node2 instancel
Tue Jul  5 04:24:16 2011 Converting template to drbd
Tue Jul  5 04:24:17 2011 Creating additional volumes...
Tue Jul  5 04:24:18 2011 Renaming original volumes...
Tue Jul  5 04:24:18 2011 Initializing DRBD devices...
Tue Jul  5 04:24:19 2011 - INFO: Waiting for instance instancel.example.org to sync disks.
Tue Jul  5 04:24:20 2011 - INFO: - device disk/0:  0.30% done, 14m 32s remaining (estimated)
Tue Jul  5 04:25:20 2011 - INFO: - device disk/0: 38.10% done, 1m 22s remaining (estimated)
Tue Jul  5 04:26:20 2011 - INFO: - device disk/0: 72.90% done, 35s remaining (estimated)
Tue Jul  5 04:26:55 2011 - INFO: - device disk/0: 95.00% done, 14s remaining (estimated)
Tue Jul  5 04:27:10 2011 - INFO: Instance instancel.example.org's disks are in sync.
Modified instance instancel
- disk_template -> drbd
Please don't forget that most parameters take effect only at the next start of the instance.
```

Instance Failover

```
root@node1:~# gnt-instance failover -f instancel
Tue Jul  5 04:32:00 2011 - INFO: Not checking memory on the secondary node as instance will not
be started
Tue Jul  5 04:32:00 2011 * not checking disk consistency as instance is not running
Tue Jul  5 04:32:00 2011 * shutting down instance on source node
Tue Jul  5 04:32:00 2011 * deactivating the instance's disks on source node
```

Instance Migration

```
root@node1:~# gnt-instance start instancel
Waiting for job 30 for instancel.example.org...

root@node1:~# gnt-instance migrate -f instancel
```

```
Wed Jul 6 04:14:43 2011 Migrating instance instancel.example.org
Wed Jul 6 04:14:43 2011 * checking disk consistency between source and target
Wed Jul 6 04:14:43 2011 * switching node node1.example.org to secondary mode
Wed Jul 6 04:14:44 2011 * changing into standalone mode
Wed Jul 6 04:14:44 2011 * changing disks into dual-master mode
Wed Jul 6 04:14:45 2011 * wait until resync is done
Wed Jul 6 04:14:47 2011 * preparing node1.example.org to accept the instance
Wed Jul 6 04:14:48 2011 * migrating instance to node1.example.org
Wed Jul 6 04:15:13 2011 * switching node node2.example.org to secondary mode
Wed Jul 6 04:15:14 2011 * wait until resync is done
Wed Jul 6 04:15:14 2011 * changing into standalone mode
Wed Jul 6 04:15:14 2011 * changing disks into single-master mode
Wed Jul 6 04:15:15 2011 * wait until resync is done
Wed Jul 6 04:15:15 2011 * done
```

Master Failover

```
root@node2:~# gnt-cluster master-failover
root@node2:~# gnt-cluster getmaster
node2.example.org
root@node1:~# gnt-cluster master-failover
```

Job Operations

```
root@node2:~# gnt-job list
ID Status Summary
35 success INSTANCE_STARTUP(instancel.example.org)
36 success INSTANCE_MIGRATE(instancel.example.org)
37 success INSTANCE_CONSOLE(instancel.example.org)
38 success INSTANCE_SHUTDOWN(instancel.example.org)
39 success INSTANCE_REPLACE_DISKS(instancel.example.org)
```

```
root@node2:~# gnt-job info 39
Job ID: 39
Status: success
Received:      2011-07-06 05:47:48.900699
Processing start: 2011-07-06 05:47:48.986170 (delta 0.085471s)
Processing end:   2011-07-06 05:50:55.060360 (delta 186.074190s)
Total processing time: 186.159661 seconds
Opcodes:
  OP_INSTANCE_REPLACE_DISKS
  Status: success
  Processing start: 2011-07-06 05:47:48.986170
  Execution start:  2011-07-06 05:47:49.086837
  Processing end:    2011-07-06 05:50:55.060316
```

...

Using Htools

```
root@node1:~# gnt-instance add -I hail -o image+debian-lenny -t drbd -s 5G --no-start instance2
Wed Jul  6 06:00:28 2011 - INFO: Selected nodes for instance instance2.example.org via iallocator
hail: node2.example.org, node1.example.org
Wed Jul  6 06:00:29 2011 * creating instance disks...
Wed Jul  6 06:00:33 2011 adding instance instance2.example.org to cluster config
```

...

```
root@node1:~# gnt-instance failover instance2
root@node1:~# hbal -L
Loaded 2 nodes, 2 instances
Group size 2 nodes, 2 instances
Selected node group: default
Initial check done: 0 bad nodes, 0 bad instances.
Initial score: 3.82710280
Trying to minimize the CV...
```

```
1. instance1 node1:node2 => node2:node1 0.01468625 a=f
Cluster score improved from 3.82710280 to 0.01468625
Solution length=1
```

```
root@node1:~# hbal -L -X
Loaded 2 nodes, 2 instances
Group size 2 nodes, 2 instances
Selected node group: default
Initial check done: 0 bad nodes, 0 bad instances.
Initial score: 3.86448598
Trying to minimize the CV...
```

```
1. instance1 node1:node2 => node2:node1 0.02269693 a=f
Cluster score improved from 3.86448598 to 0.02269693
Solution length=1
Executing jobset for instances instance1.example.org
Got job IDs 43
```

```
root@node2:~# hspace --memory 512 --disk 10240 -L
HTS_SPEC_MEM=512
HTS_SPEC_DSK=10240
HTS_SPEC_CPU=1
HTS_SPEC_RQN=2
HTS_CLUSTER_MEM=1498
HTS_CLUSTER_DSK=51944
HTS_CLUSTER_CPU=4
HTS_CLUSTER_VCPU=256
HTS_CLUSTER_NODES=2
```

```
...
```

Recovering from a Node Failure

Setup node3

1. start and ssh to node3
2. `cd puppet`
3. `git pull`
4. `puppet apply nodes/node3.pp`
5. `puppet apply nodes/node3/install-ganeti.pp`
6. ssh to node1
7. `gnt-node add -s 192.168.16.18 node3`

Simulating a node failure

Let's simulate **node2** going down hard while instance2 or instance1 is running on it (depending on how htools allocated your VMs).

1. On node2's VirtualBox window: **Machine → Close → Power off the machine → OK**
2. `gnt-cluster verify`
3. `gnt-node modify -O yes -f node2`
4. `gnt-instance failover --ignore-consistency instance2`
5. `gnt-node evacuate -I hail node2`
6. `gnt-cluster verify`

Readding node3

1. Start node2 back up
2. `gnt-node add --readd node2`
3. `gnt-cluster verify`
4. ssh to node2
5. `lvremove ganeti`