Ganeti installation tutorial

Documents Ganeti version 1.2

1. Introduction

Ganeti is a cluster virtualization management system based on Xen. This document explains how to bootstrap a Ganeti node (Xen `dom0`), create a running cluster and install virtual instance (Xen `domU`). You need to repeat most of the steps in this document for every node you want to install, but of course we recommend creating some semi-automatic procedure if you plan to deploy Ganeti on a medium/large scale.

A basic Ganeti terminology glossary is provided in the introductory section of the *Ganeti administrator's guide*. Please refer to that document if you are uncertain about the terms we are using.

Ganeti has been developed for Linux and is distribution-agnostic. This documentation will use Debian Etch as an example system but the examples can easily be translated to any other distribution. You are expected to be familiar with your distribution, its package management system, and Xen before trying to use Ganeti.

This document is divided into two main sections:

- Installation of the base system and base components
- Configuration of the environment for Ganeti

Each of these is divided into sub-sections. While a full Ganeti system will need all of the steps specified, some are not strictly required for every environment. Which ones they are, and why, is specified in the corresponding sections.

2. Installing the base system and base components

2.1. Hardware requirements

Any system supported by your Linux distribution is fine. 64-bit systems are better as they can support more memory.

Any disk drive recognized by Linux (`IDE`/`SCSI`/`SATA`/etc.) is supported in Ganeti. Note that no shared storage (e.g. `SAN`) is needed to get high-availability features. It is highly recommended to use more than one disk drive to improve speed. But Ganeti also works with one disk per machine.

2.2. Installing the base system

Mandatory on all nodes.

It is advised to start with a clean, minimal install of the operating system. The only requirement you need to be aware of at this stage is to partition leaving enough space for a big (**minimum `20GiB`**) LVM volume group which will then host your instance filesystems. The volume group name Ganeti 1.2 uses (by default) is *xenvg*.

While you can use an existing system, please note that the Ganeti installation is intrusive in terms of changes to the system configuration, and it's best to use a newly-installed system without important data on it.

Also, for best results, it's advised that the nodes have as much as possible the same hardware and software configuration. This will make administration much easier.

2.2.1. Hostname issues

Note that Ganeti requires the hostnames of the systems (i.e. what the `hostname` command outputs to be a fully-qualified name, not a short name. In other words, you should use `node1.example.com` as a hostname and not just `node1`.

Debian. Note that Debian Etch configures the hostname differently than you need it for Ganeti. For example, this is what Etch puts in `/etc/hosts` in certain situations:

```
127.0.0.1       localhost
127.0.1.1       node1.example.com node1
```

but for Ganeti you need to have:

```
127.0.0.1       localhost
192.168.1.1     node1.example.com node1
```

replacing `192.168.1.1` with your node's address. Also, the file `/etc/hostname` which configures the hostname of the system should contain `node1.example.com` and not just `node1` (you need to run the command `/etc/init.d/hostname.sh start` after changing the file).

2.3. Installing Xen

Mandatory on all nodes.

While Ganeti is developed with the ability to modularly run on different virtualization environments in mind the only one currently useable on a live system is Xen (http://xen.xensource.com/). Supported versions are: `3.0.3`, `3.0.4`, `3.1`.

Please follow your distribution's recommended way to install and set up Xen, or install Xen from the upstream source, if you wish, following their manual.

After installing Xen you need to reboot into your Xen-ified dom0 system. On some distributions this might involve configuring GRUB appropriately, whereas others will configure it automatically when you install Xen from a package.

Debian. Under Debian Etch or Sarge+backports you can install the relevant `xen-linux-system` package, which will pull in both the hypervisor and the relevant kernel. Also, if you are installing a 32-bit Etch, you should install the `libc6-xen` package (run `apt-get install libc6-xen`).

2.3.1. Xen settings

It's recommended that dom0 is restricted to a low amount of memory (`512MiB` is reasonable) and that memory ballooning is disabled in the file `/etc/xen/xend-config.sxp` by setting the value `dom0-min-mem` to `0`, like this: `(dom0-min-mem 0)`

For optimum performance when running both CPU and I/O intensive instances, it's also recommended that the dom0 is restricted to one CPU only, for example by booting with the kernel parameter `nosmp`.

It is recommended that you disable xen's automatic save of virtual machines at system shutdown and subsequent restore of them at reboot. To obtain this make sure the variable `XENDOMAINS_SAVE` in the file `/etc/default/xendomains` is set to an empty value.

Debian. Besides the ballooning change which you need to set in `/etc/xen/xend-config.sxp`, you need to set the memory and nosmp parameters in the file `/boot/grub/menu.lst`. You need to modify the variable `xenhopt` to add **`dom0_mem=512M`** like this:

```
## Xen hypervisor options to use with the default Xen boot option
# xenhopt=dom0_mem=512M
```

and the `xenkopt` needs to include the **`nosmp`** option like this:

```
## Xen Linux kernel options to use with the default Xen boot option
# xenkopt=nosmp
```

Any existing parameters can be left in place: it's ok to have `xenkopt=console=tty0 nosmp`, for example. After modifying the files, you need to run:

```
/sbin/update-grub
```

If you want to test the experimental HVM support with Ganeti and want VNC access to the console of your instances, set the following two entries in `/etc/xen/xend-config.sxp`:

```
(vnc-listen '0.0.0.0')
(vncpasswd '')
```

You need to restart the Xen daemon for these settings to take effect:

```
/etc/init.d/xend restart
```

2.3.2. Selecting the instance kernel

After you have installed Xen, you need to tell Ganeti exactly what kernel to use for the instances it will create. This is done by creating a *symlink* from your actual kernel to `/boot/vmlinuz-2.6-xenU`, and one from your initrd to `/boot/initrd-2.6-xenU`. Note that if you don't use an initrd for the `domU` kernel, you don't need to create the initrd symlink.

Debian. After installation of the `xen-linux-system` package, you need to run (replace the exact version number with the one you have):

```
cd /boot
ln -s vmlinuz-2.6.18-5-xen-686 vmlinuz-2.6-xenU
ln -s initrd.img-2.6.18-5-xen-686 initrd-2.6-xenU
```

2.4. Installing DRBD

Recommended on all nodes: DRBD (http://www.drbd.org/) is required if you want to use the high availability (HA) features of Ganeti, but optional if you don't require HA or only run Ganeti on single-node clusters. You can upgrade a non-HA cluster to an HA one later, but you might need to export and re-import all your instances to take advantage of the new features.

Supported DRBD versions: the `0.7` series **or** `8.0.7`. It's recommended to have at least version `0.7.24` if you use **udev** since older versions have a bug related to device discovery which can be triggered in cases of hard drive failure.

Now the bad news: unless your distribution already provides it installing DRBD might involve recompiling your kernel or anyway fiddling with it. Hopefully at least the Xen-ified kernel source to start from will be provided.

The good news is that you don't need to configure DRBD at all. Ganeti will do it for you for every instance you set up. If you have the DRBD utils installed and the module in your kernel you're fine.

Please check that your system is configured to load the module at every boot, and that it passes the following option to the module (for `0.7.x`: `minor_count=64` (this will allow you to use up to 32 instances per node) or for `8.0.x` you can use up to `255` (i.e. `minor_count=255`, but for most clusters `128` should be enough).

Debian. You can just install (build) the DRBD 0.7 module with the following commands (make sure you are running the Xen kernel):

```
apt-get install drbd0.7-module-source drbd0.7-utils
m-a update
m-a a-i drbd0.7
echo drbd minor_count=64 >> /etc/modules
modprobe drbd minor_count=64
```

or for using DRBD `8.x` from the etch backports (note: you need at least 8.0.7, older version have a bug that breaks ganeti's usage of drbd):

```
apt-get install -t etch-backports drbd8-module-source drbd8-utils
m-a update
m-a a-i drbd8
echo drbd minor_count=128 >> /etc/modules
modprobe drbd minor_count=128
```

It is also recommended that you comment out the default resources in the `/etc/dbrd.conf` file, so that the init script doesn't try to configure any drbd devices. You can do this by prefixing all `resource` lines in the file with the keyword `skip`, like this:

```
skip resource r0 {
...
}

skip resource "r1" {
...
}
```

2.5. Other required software

Besides Xen and DRBD, you will need to install the following (on all nodes):

- LVM version 2 (http://sourceware.org/lvm2/)
- OpenSSL (http://www.openssl.org/)
- OpenSSH (http://www.openssh.com/portable.html)
- Bridge utilities (http://bridge.sourceforge.net/)

- iproute2 (http://developer.osdl.org/dev/iproute2)
- arping (ftp://ftp.inr.ac.ru/ip-routing/iputils-current.tar.gz) (part of iputils package)
- mdadm (http://www.kernel.org/pub/linux/utils/raid/mdadm/) (Linux Software Raid tools)
- Python 2.4 (http://www.python.org)
- Python Twisted library (http://twistedmatrix.com/) - the core library is enough
- Python OpenSSL bindings (http://pyopenssl.sourceforge.net/)
- simplejson Python module (http://www.undefined.org/python/#simplejson)
- pyparsing Python module (http://pyparsing.wikispaces.com/)

These programs are supplied as part of most Linux distributions, so usually they can be installed via apt or similar methods. Also many of them will already be installed on a standard machine.

Debian. You can use this command line to install all of them:

```
# apt-get install lvm2 ssh bridge-utils iproute iputils-arping \
  python2.4 python-twisted-core python-pyopenssl openssl \
  mdadm python-pyparsing python-simplejson
```

3. Setting up the environment for Ganeti

3.1. Configuring the network

Mandatory on all nodes.

Ganeti relies on Xen running in "bridge mode", which means the instances network interfaces will be attached to a software bridge running in dom0. Xen by default creates such a bridge at startup, but your distribution might have a different way to do things.

Beware that the default name Ganeti uses is xen-br0 (which was used in Xen 2.0) while Xen 3.0 uses xenbr0 by default. The default bridge your Ganeti cluster will use for new instances can be specified at cluster initialization time.

Debian. The recommended Debian way to configure the Xen bridge is to edit your `/etc/network/interfaces` file and substitute your normal Ethernet stanza with the following snippet:

```
auto xen-br0
iface xen-br0 inet static
        address YOUR_IP_ADDRESS
        netmask YOUR_NETMASK
```

```
        network YOUR_NETWORK
        broadcast YOUR_BROADCAST_ADDRESS
        gateway YOUR_GATEWAY
        bridge_ports eth0
        bridge_stp off
        bridge_fd 0
```

The following commands need to be executed on the local console

```
ifdown eth0
ifup xen-br0
```

To check if the bridge is setup, use **ip** and **brctl show**:

```
# ip a show xen-br0
9: xen-br0: <BROADCAST,MULTICAST,UP,10000> mtu 1500 qdisc noqueue
    link/ether 00:20:fc:1e:d5:5d brd ff:ff:ff:ff:ff:ff
    inet 10.1.1.200/24 brd 10.1.1.255 scope global xen-br0
    inet6 fe80::220:fcff:fe1e:d55d/64 scope link
       valid_lft forever preferred_lft forever

# brctl show xen-br0
bridge name     bridge id               STP enabled     interfaces
xen-br0         8000.0020fc1ed55d       no              eth0
```

3.2. Configuring LVM

Mandatory on all nodes.

> **Note:** The volume group is required to be at least `20GiB`.

If you haven't configured your LVM volume group at install time you need to do it before trying to initialize the Ganeti cluster. This is done by formatting the devices/partitions you want to use for it and then adding them to the relevant volume group:

```
pvcreate /dev/sda3
vgcreate xenvg /dev/sda3
```

or

```
pvcreate /dev/sdb1
pvcreate /dev/sdc1
vgcreate xenvg /dev/sdb1 /dev/sdc1
```

If you want to add a device later you can do so with the vgextend(8) command:

```
pvcreate /dev/sdd1
vgextend xenvg /dev/sdd1
```

Optional. It is recommended to configure LVM not to scan the DRBD devices for physical volumes. This can be accomplished by editing `/etc/lvm/lvm.conf` and adding the `/dev/drbd[0-9]+` regular expression to the `filter` variable, like this:

```
    filter = [ "r|/dev/cdrom|", "r|/dev/drbd[0-9]+|" ]
```

3.3. Installing Ganeti

Mandatory on all nodes.

It's now time to install the Ganeti software itself. Download the source from http://code.google.com/p/ganeti/.

```
tar xvzf ganeti-1.2.2.tar.gz
cd ganeti-1.2.2
./configure --localstatedir=/var --sysconfdir=/etc
make
make install
mkdir /srv/ganeti/ /srv/ganeti/os /srv/ganeti/export
```

You also need to copy the file `doc/examples/ganeti.initd` from the source archive to `/etc/init.d/ganeti` and register it with your distribution's startup scripts, for example in Debian:

```
update-rc.d ganeti defaults 20 80
```

In order to automatically restart failed instances, you need to setup a cron job run the `ganeti-watcher` program. A sample cron file is provided in the source at `doc/examples/ganeti.cron` and you can copy that (eventually altering the path) to `/etc/cron.d/ganeti`

3.4. Installing the Operating System support packages

Mandatory on all nodes.

To be able to install instances you need to have an Operating System installation script. An example for Debian Etch is provided on the project web site. Download it from http://code.google.com/p/ganeti/ and follow the instructions in the `README` file. Here is the installation procedure (replace `0.2` with the latest version that is compatible with your ganeti version):

```
cd /srv/ganeti/os
tar xvf ganeti-instance-debian-etch-0.4.tar
mv ganeti-instance-debian-etch-0.4 debian-etch
```

In order to use this OS definition, you need to have internet access from your nodes and have the debootstrap(8), dump(8) and restore(8) commands installed on all nodes.

Debian. Use this command on all nodes to install the required packages:

```
apt-get install debootstrap dump
```

Alternatively, you can create your own OS definitions. See the manpage ganeti-os-interface(8).

3.5. Initializing the cluster

Mandatory: only on one node per cluster.

The last step is to initialize the cluster. After you've repeated the above process on all of your nodes, choose one as the master, and execute:

```
gnt-cluster init CLUSTERNAME
```

The *`CLUSTERNAME`* is a hostname, which must be resolvable (e.g. it must exist in DNS or in `/etc/hosts`) by all the nodes in the cluster. You must choose a name different from any of the nodes names for a multi-node cluster. In general the best choice is to have a unique name for a cluster, even if it consists of only one machine, as you will be able to expand it later without any problems.

If the bridge name you are using is not `xen-br0`, use the `-b` *`BRIDGENAME`* option to specify the bridge name. In this case, you should also use the `--master-netdev` *`BRIDGENAME`* option with the same *`BRIDGENAME`* argument.

You can use a different name than `xenvg` for the volume group (but note that the name must be identical on all nodes). In this case you need to specify it by passing the `-g` *`VGNAME`* option to `gnt-cluster init`.

To set up the cluster as an HVM cluster, use the `--hypervisor=xen-hvm3.1` option to use the Xen 3.1 HVM hypervisor. Note that with the experimental HVM support, you will only be able to create HVM instances in a cluster set to this hypervisor type. Mixed PVM/HVM clusters are not supported by the Ganeti 1.2 experimental HVM support. You will also need to create the VNC cluster password file `/etc/ganeti/vnc-cluster-password` which contains one line with the default VNC password for the cluster. Finally, you need to provide an installation ISO image for HVM instance which will not only be mapped to the first CDROM of the instance, but which the instance will also boot from. This ISO image is expected at `/srv/ganeti/iso/hvm-install.iso`.

You can also invoke the command with the `--help` option in order to see all the possibilities.

3.6. Joining the nodes to the cluster

Mandatory: for all the other nodes.

After you have initialized your cluster you need to join the other nodes to it. You can do so by executing the following command on the master node:

`gnt-node add` *`NODENAME`*

3.7. Separate replication network

Optional

Ganeti uses DRBD to mirror the disk of the virtual instances between nodes. To use a dedicated network interface for this (in order to improve performance or to enhance security) you need to configure an additional interface for each node. Use the `-s` option with `gnt-cluster init` and `gnt-node add` to specify the IP address of this secondary interface to use for each node. Note that if you specified this option at cluster setup time, you must afterwards use it for every node add operation.

3.8. Testing the setup

Execute the `gnt-node list` command to see all nodes in the cluster:

```
# gnt-node list
Node              DTotal  DFree MTotal MNode MFree Pinst Sinst
```

```
node1.example.com 197404 197404   2047  1896   125     0     0
```

4. Setting up and managing virtual instances

4.1. Setting up virtual instances

This step shows how to setup a virtual instance with either non-mirrored disks (`plain`) or with network mirrored disks (`remote_raid1` for drbd 0.7 and `drbd` for drbd 8.x). All commands need to be executed on the Ganeti master node (the one on which `gnt-cluster init` was run). Verify that the OS scripts are present on all cluster nodes with `gnt-os list`.

To create a virtual instance, you need a hostname which is resolvable (DNS or `/etc/hosts` on all nodes). The following command will create a non-mirrored instance for you:

```
gnt-instance add --node=node1 -o debian-etch -t plain inst1.example.com
* creating instance disks...
adding instance inst1.example.com to cluster config
Waiting for instance inst1.example.com to sync disks.
Instance inst1.example.com's disks are in sync.
creating os for instance inst1.example.com on node node1.example.com
* running the instance OS create scripts...
```

The above instance will have no network interface enabled. You can access it over the virtual console with `gnt-instance console inst1`. There is no password for root. As this is a Debian instance, you can modify the `/etc/network/interfaces` file to setup the network interface (`eth0` is the name of the interface provided to the instance).

To create a network mirrored instance, change the argument to the `-t` option from `plain` to `remote_raid1` (drbd 0.7) or `drbd` (drbd 8.0) and specify the node on which the mirror should reside with the second value of the `--node` option, like this:

```
# gnt-instance add -t remote_raid1 -n node1:node2 -o debian-etch instance2
* creating instance disks...
adding instance instance2 to cluster config
Waiting for instance instance1 to sync disks.
- device sdb:  3.50% done, 304 estimated seconds remaining
- device sdb: 21.70% done, 270 estimated seconds remaining
- device sdb: 39.80% done, 247 estimated seconds remaining
- device sdb: 58.10% done, 121 estimated seconds remaining
- device sdb: 76.30% done, 72 estimated seconds remaining
- device sdb: 94.80% done, 18 estimated seconds remaining
```

```
Instance instance2's disks are in sync.
creating os for instance instance2 on node node1.example.com
* running the instance OS create scripts...
* starting instance...
```

4.2. Managing virtual instances

All commands need to be executed on the Ganeti master node

To access the console of an instance, use `gnt-instance console` *`INSTANCENAME`*.

To shutdown an instance, use `gnt-instance shutdown` *`INSTANCENAME`*. To startup an instance, use `gnt-instance startup` *`INSTANCENAME`*.

To failover an instance to its secondary node (only possible in `remote_raid1` or `drbd` disk templates), use `gnt-instance failover` *`INSTANCENAME`*.

For more instance and cluster administration details, see the *Ganeti administrator's guide*.