Building a Beowulf cluster

Åsmund Ødegård

April 4, 2001

Outline

- **Introduction**
- **Strategy**
- **Assembling all parts, wiring up.**
- **Installing Debian GNU/Linux**
- **Booting and installing software on nodes.**
- **Install *Cluster software***

Introduction

- First, I will describe the process of setting up a cluster
- and what kind of systems we will install
- Later, we will build 3 small clusters

Strategy

- Consider a cluster of O(100) nodes.
- You certainly want to install nodes in some automatic fashion.
- User–administration and software installation should be done once for the whole cluster.
- In other words: You want O(1) work for N machines.
- This is maybe impossible.

Strategy . . .

- **Repeatable:**
  - **Standard OS configuration e.g on cdrom (“kickstart”)**
  - **Straightforward, but you still have to set a few parameters manually on each node**
- **Defined:**
  - **Use a server that defines the configuration**
  - **Improve consistency.**
- **Even higher levels of management exist.**

Strategy . . .

Our approach:

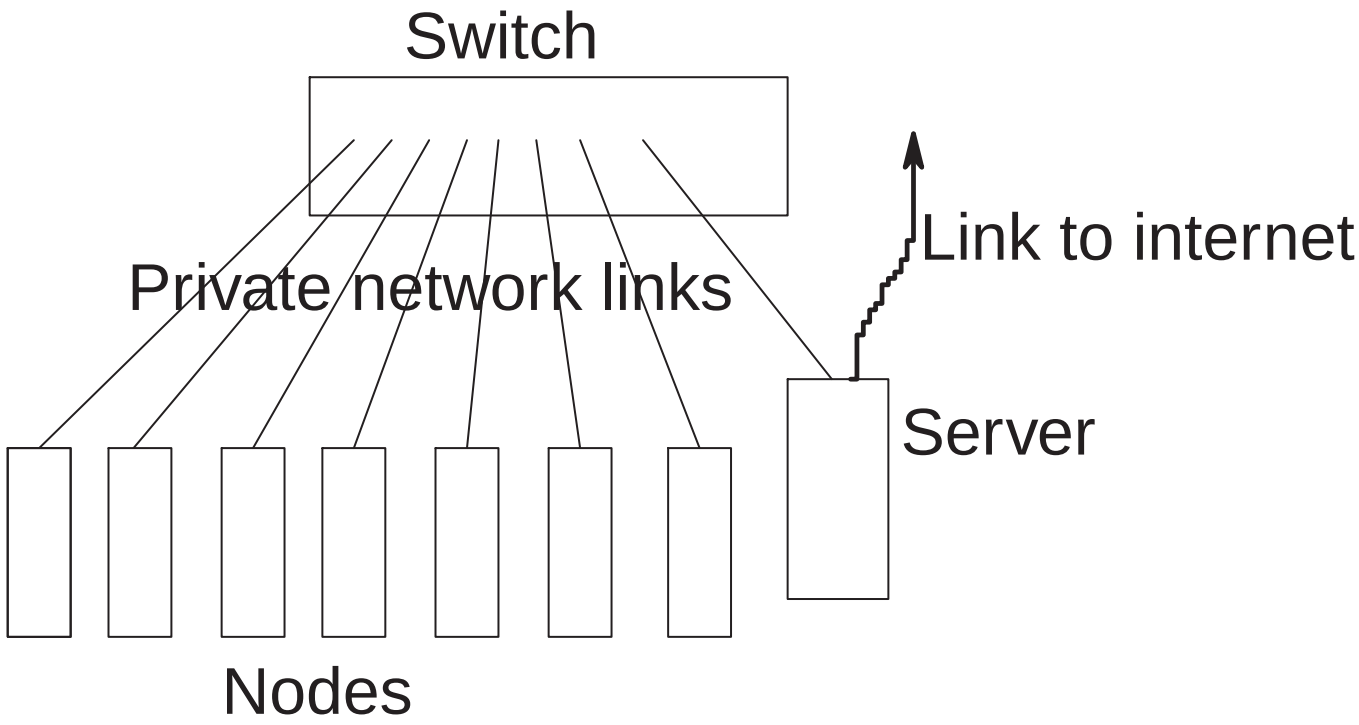
- **Install Linux on a computer which will be the server**
- **Define the configuration for all nodes on the server**
- **Let the nodes contact the server to receive necessary information during installation.**
- **Diskless vs. Diskful nodes.**
- **We'll use a system called "FAI" to achieve this.**
- **The work is very close to O(1)!**

Network

This morning, we touched networking issues briefly:

- **How should a Beowulf relate to the rest of your network?**
  - **Stand alone system**
  - **Guarded system**
  - **Universally accessible system**

A Fast Ethernet network



Guarded network . . .

- The most common approach
- Choose among IETF [1] private network address ranges:
  - 10.x.x.x
  - 172.16.x.x - 172.31.x.x
  - 192.168.x.x
- If your nodes need to reach the global Internet, configure a gateway server with IP masquerading
- Limits access to say external file server
- Easy to manage, easy to add nodes

Machine naming

- Hostnames are important
- They should encode some useful information, e.g. net–type, hardware
- Use consistent, obvious naming scheme.
- correspondence between name and IP number
  - We'll use the names: server, node01, node02, ...
  - and IP numbers: 10.0.0.1, 10.0.0.101, 10.0.0.102, ...

Details of the approach

- **Consider a running server**
- **Using a tool called “FAI”, the installation of nodes are defined on the server**
- **Nodes are booted with a floppy. The kernel on the floppy contacts the server to retrieve information**
- **A basic system is set up on each node, such that floppy is not needed later**
- **Nodes get applications and user–files from server with *NFS***
- **Information about users and passwords are distributed with *NIS***

Outline of installation

- **Install Debian GNU/Linux on a PC**
- **Install and configure “FAI” on the server**
- **Configure “FAI” for nodes and boot them**
- **Install and configure mpi and queue–system**

Why Debian

- The choice of distribution is a most of all a matter of taste
- Debian GNU/Linux is strong in flexibility and management
- and can easily feel a bit rough for beginners
- Other options: RedHat, Slackware, SuSE, TurboLinux,...
- Specialized cluster distributions/systems: Scyld, Extreme Linux, SCore,...

Debian

- **Insert cdrom, boot, start installation.**
- **Partition your harddisk, initialize and mount partitions**
- **Install OS Kernel, Modules and base system**
- **Remove cdrom, reboot, insert cdrom and install packages.**
- `apt-get install <package>`

FAI

- **FAI: Fully Automatic Installation**
- **A tool for installing Debian on multiple hosts**
- **Based on the Debian package manager, “dpkg”**
- **A collection of Perl, bash and *cfengine* scripts**
- **Most Linux applications are configured with a file in */etc***
- **FAI approach: Predefine configuration, copy to right place**

Install FAI

- **Install the FAI debian package**
- **Review the configuration of FAI in** `/etc/fai.conf`
- **Run** `/usr/sbin/fai-setup`
- **Configure the installation for nodes in** `$FAI_CONFIGDIR`
- **Configure rsh such that you don't need passwords to access nodes**
- **Make a bootfloppy with** `/usr/sbin/make-fai-bootfloppy`

NIS & NFS

- **Set NIS domainname in** `/etc/defaultdomain`
- **Set NIS on server to master in** `/etc/init.d/nis`**, and restart**
- **Add server and all nodes to a *faiclients* netgroup in** `/etc/netgroup`
- **Create NIS maps:** `/usr/lib/yp/ypinit -m`
- **Export necessary filesystems in** `/etc/exports` **to the *faiclients*, and restart NFS**
- **REMARK: Run** `/usr/sbin/fai-setup again` **?**

Bootp

- **Edit `/etc/inetd.conf` to run bootp.**
- **Restart the inetd service**
- **Configure `/etc/bootptab`, a suitable template is included in FAI doc/examples directory**
- **Run `tcpdump` to gather information about mac–addresses, and insert in the bootp–configuration**

MPI

- **Make sure that mpich is installed on the server and all nodes**
- **Add server and nodes in** `/etc/mpich/machines.LINUX`
- **Dual–CPR computers are added twice**
- **Run jobs with** `mpirun -np 4 <application>`

PBS

- **Unpack the source code**
- **Configure with** `./configure -set-server-home=$PBSHOME`
- **Run** `make` **and** `make install`
- **Add nodes to** `$PBSHOME/server_priv/nodes` **like: "node01 np=2"**
- **Start the server with** `-t create` **as argument**
- **Configure the mom, start mom and scheduler.**

The first qmgr session

- > set server managers=you@host
- > create queue ws queue_type=e
- > set queue ws enabled=true, started=true
- > set server scheduling=true
- > set server default_queue=ws

Configure mon on nodes

- **Locate** `pbs_mkdirs` **and make it executable**
- **Do for all nodes:**
- `%>rsh node??  .../buildutils/pbs_mkdirs mom`
- `%>rsh node??  .../buildutils/pbs_mkdirs aux`
- `%>rsh node??  .../buildutils/pbs_mkdirs default`
- `%>rcp $PBSHOME/mom_priv/config node??:$PBSHOME/mom_priv`
- `%>rsh node??  /usr/local/sbin/pbs_mom`

References

[1] The Internet Engineering Task Force website. http://www.ietf.org/.

[2] Linux documentation project - howto's. http://www.linuxdoc.org/HOWTO.

[3] Terry Dawson Olaf Kirch. The network administrator's guide. http://www.linuxdoc.org/LDP/nag2/nag2.pdf, 2000.

[4] The Scyld Website. http://www.scyld.com/.

[5] Veridian Systems. *Portable Batch System Administrator Guide*, release 2.3 edition, August 2000.

[6] The whatis website. http://www.whatis.com.