

## ÍNDICE

1. Requisitos de Hardware.....	2
2. Arranque del front-end.....	2
3. Arranque de los nodos.....	4
4. Utilización de lam , OpenMPI y Ganglia.....	5
5. Cambio de configuración de red.....	7

## 1. REQUISITOS DE HARDWARE

Para poder construir un cluster Beowulf con ABC GNU/Linux es preciso cumplir los siguientes requisitos:

- Utilizar más de un PC y todos los PCs integren 256 MB de memoria RAM y que al menos sean procesadores P3 500Mhz o equivalente. En caso de que se quiera construir un cluster en modo “live” no es preciso utilizar disco duro alguno, en caso de querer instalar la distribución en el front-end(el PC donde se inserta el disco de ABC) será necesario que ese PC tenga disco duro.
- Los PCs deben estar interconectados a través de un switch. Nunca debe utilizarse un router para interconectar los PCs. La velocidad del switch y la categoría del cableado empleado han de ser la mayor posible ya que en caso de que el cluster este formado por un gran número de PCs la latencia perjudicaría el rendimiento del sistema

## 2. ARRANQUE DEL FRONT-END

Como ha sido mencionado en el punto anterior, en uno de los PCs insertaremos el disco de ABC GNU/Linux. En este PC escogeremos en su BIOS que el dispositivo preferido para el arranque sea el DVD. Una vez arrancado el disco se mostrará en la pantalla lo siguiente:

```
ISOLINUX 3.63 Debian-2008-07-15 Copyright (C) 1994-2008 H. Peter Anvin
Automated Beowulf Cluster ABC GNU/Linux

IMPORTANT
user="master" password="master" hostname="master"

For the default live system, press ENTER or enter 'live'.

To start the installer directly, enter 'install'.

To verify the CD for errors, enter 'check'.

To run memtest86+, enter 'memtest'

To boot from the first hard disk, enter 'hd'

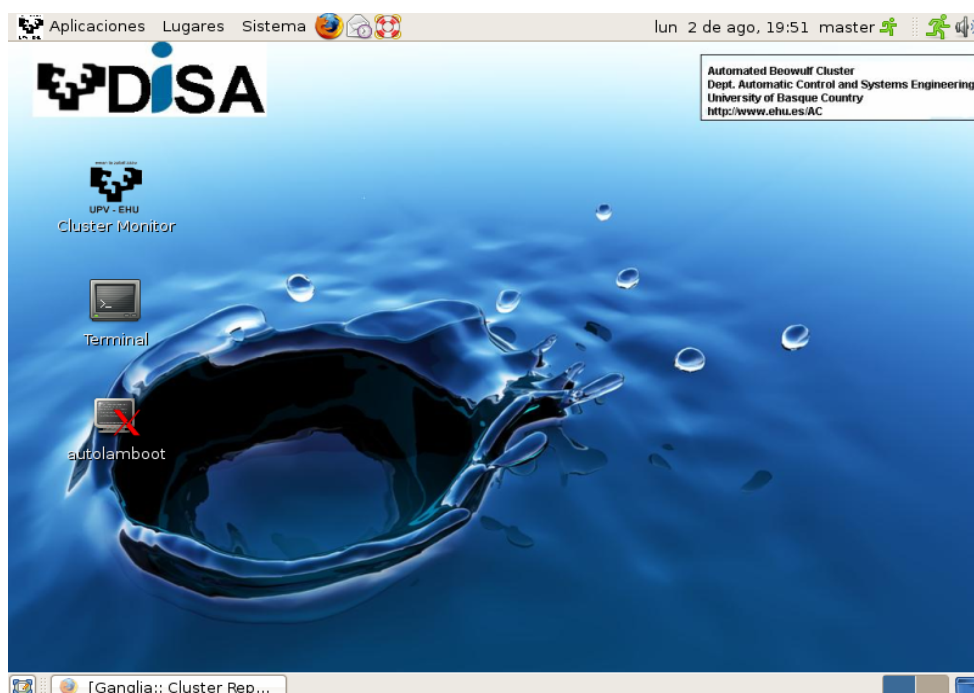
Iker Castanos Chavarri icastanos001@ikasle.ehu.es
Dept. Automatic Control and Systems Engineering
University of the Basque Country
http://www.ehu.es/AC

boot: _
```

Como puede ser observado en la captura de la pantalla, puede optarse por las siguientes opciones:

1. Modo “live”. Este modo de arranque se hace sin necesidad de instalación alguna., todo el sistema arranca en RAM. Presionando la tecla “enter” o escribiendo “live” y presionando “enter”.
2. Modo “installer”. Mediante este modo se instalara la distribución en el disco duro del front-end. La instalación se lleva a cabo mediante el instalador de Ubuntu, el cual es muy sencillo y amigable para el usuario. Es muy importante que durante la instalación se cree el usuario “master” con clave “master” y que al PC le pongamos como hostname “master”. Una vez instalado su funcionamiento es idéntico al del modo “live”, solo que no se precisara disco alguno, el arranque será mas rápido y los datos del “home” del usuario permanecerán aun apagando o reiniciando el sistema Para optar por este modo se debe escribir “install” y presionar la tecla “enter”.
3. Modo “checkdisk”. Es utilizado para comprobar la integridad del soporte óptico. Para optar por este modo se debe teclear “check” y presionar la tecla “enter”.
4. Modo “memtest” Es utilizado para comprobar el estado de la memoria RAM. Para optar por este modo se debe teclear “memtest” y presionar la tecla “enter”.
5. En caso de que se opte por arrancar del disco duro tan solo se debe teclear “hd” y presionar la tecla “enter”

Tanto si se arranca en modo “live” o si se arranca una instalación de ABC GNU/Linux que se encuentre en el disco duro, accederemos al escritorio GNOME como se muestra en la siguiente captura de pantalla:



Una vez arrancado el front-end se debe arrancar los nodos como se comentara en el siguiente punto.

### 3. ARRANQUE DE LOS NODOS

Se debe configurar la BIOS especificando que el dispositivo de arranque sea la NIC mediante PXE. Una vez configurada la BIOS el arranque de cada nodo se llevara a cabo como se muestra en las siguientes capturas de pantalla:

```
GATEWAY IP: 192.168.0.1

PXELINUX 3.53 Debian-2007-12-11 Copyright (C) 1994-2007 H. Peter Anvin
UNDI data segment at: 0009C7F0
UNDI data segment size: 2400
UNDI code segment at: 0009ECC0
UNDI code segment size: 0A00
PXE entry point found (we hope) at 9ECC:0106
My IP address seems to be C0A80003 192.168.0.3
ip=192.168.0.3:192.168.0.1:192.168.0.1:255.255.255.0
TFTP prefix:
Trying to load: pxelinux.cfg/564dd589-d9d8-4764-492b-6f75db8a79a2
Trying to load: pxelinux.cfg/01-00-0c-29-8a-79-a2
Trying to load: pxelinux.cfg/C0A80003
Trying to load: pxelinux.cfg/C0A80000
Trying to load: pxelinux.cfg/C0A800
Trying to load: pxelinux.cfg/C0A80
Trying to load: pxelinux.cfg/C0A8
Trying to load: pxelinux.cfg/C0A
Trying to load: pxelinux.cfg/C0
Trying to load: pxelinux.cfg/C
Trying to load: pxelinux.cfg/default
Missing parameter in syslinux.cfg.
Missing parameter in syslinux.cfg.
Loading vmlinuz-2.6.27-17-generic....._

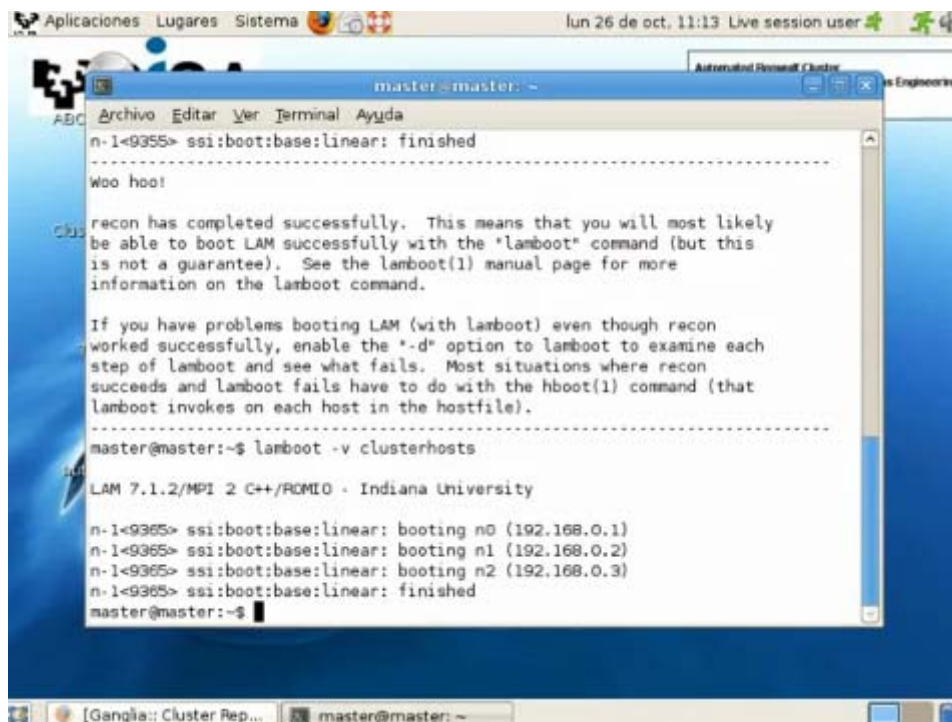
16
[ 12.075792] pcnet32: PCnet/PCI II 79C970A at 0x2000, 00 0c 29 8a 79 a2 assign
ed IRQ 16.
[ 12.075873] eth0: registered as PCnet/PCI II 79C970A
[ 12.075954] pcnet32: 1 cards_found.
Done.
[ 12.580661] RPC: Registered udp transport module.
[ 12.580754] RPC: Registered tcp transport module.
[ 12.607563] NET: Registered protocol family 17
IP-Config: eth0 hardware address 00:0c:29:8a:79:a2 mtu 1500 DHCP RARP
[ 12.618864] eth0: link up
IP-Config: eth0 complete (from 192.168.0.1):
address: 192.168.0.3 broadcast: 192.168.0.255 netmask: 255.255.255.0
gateway: 192.168.0.1 dns0 : 192.168.0.1 dns1 : 0.0.0.0
rootserver: 192.168.0.1 rootpath: /node/nfs/node
filename : pxelinux.0
Begin: Running /scripts/nfs-premount ...
Done.
Done.
Begin: Running /scripts/nfs-bottom ...
Done.
Done.
Begin: Running /scripts/init-bottom ...
Done.
* Reading files needed to boot...
```

Como puede observarse cada nodo obtendrá una IP y arrancara a través del front-end.

Una vez arrancados todos los nodos podremos utilizar el cluster de la manera que se describe en el siguiente punto.

#### 4. UTILIZACIÓN DE LAM, OPENMPI Y GANGLIA

Todos los PCs del cluster han registrado sus IPs en el fichero “clusterhosts” que se encuentra en el “home” del usuario. Es preciso arrancar el daemon lam para poder ejecutar aplicaciones en paralelo sin ser root. El arranque del daemon lam lo llevaremos a cabo en una consola mediante el comando “lamboot -v clusterhosts” como se muestra en la siguiente pantalla o clickeando el acceso “autolamboot”, el cual se encuentra en el escritorio del front-end:



```
master@master: ~
n-1<9355> ssi:boot:base:linear: finished
Woo hoo!
recon has completed successfully. This means that you will most likely
be able to boot LAM successfully with the 'lamboot' command (but this
is not a guarantee). See the lamboot(1) manual page for more
information on the lamboot command.
If you have problems booting LAM (with lamboot) even though recon
worked successfully, enable the '-d' option to lamboot to examine each
step of lamboot and see what fails. Most situations where recon
succeeds and lamboot fails have to do with the hboot(1) command (that
lamboot invokes on each host in the hostfile).
master@master:~$ lamboot -v clusterhosts
LAM 7.1.2/MPI 2 C++/ROMIO - Indiana University
n-1<9365> ssi:boot:base:linear: booting n0 (192.168.0.1)
n-1<9365> ssi:boot:base:linear: booting n1 (192.168.0.2)
n-1<9365> ssi:boot:base:linear: booting n2 (192.168.0.3)
n-1<9365> ssi:boot:base:linear: finished
master@master:~$
```

Una vez arrancado el daemon lam es preciso compilar la aplicación que vayamos a correr en paralelo mediante el comando “mpicc.openmpi codigofuente -o binario”.

.Para correr la aplicación en paralelo es necesario ejecutar el comando “mpiexec.openmpi -n N binario”, siendo N el numero de nodos que quieran ser utilizados. El ejemplo que se ha enunciado anteriormente sirve para a continuación mostrar la compilación y ejecución en 3 nodos del “Hello World!” paralelizado pero OpenMPI permite trabajar con muchísimas variables tales como número de procesos, numero de nodos, número de procesadores, etcétera. Para más información del uso de openMPI [www.open-mpi.org/](http://www.open-mpi.org/)

A continuación se muestra la compilación de un “Hello World!” cuyo código ha sido paralelizado con las librerías MPI y su posterior ejecución en paralelo mediante el uso de 3 nodos del cluster.

```

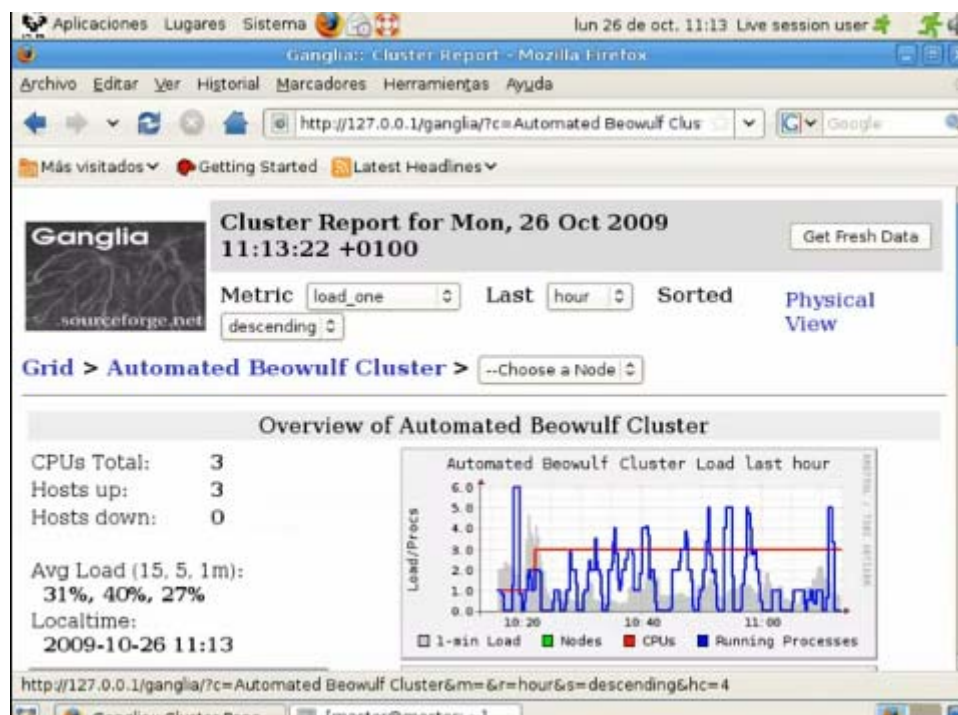
master@master: ~
be able to boot LAM successfully with the 'lamboot' command (but this
is not a guarantee). See the lamboot(1) manual page for more
information on the lamboot command.

If you have problems booting LAM (with lamboot) even though recon
worked successfully, enable the '-d' option to lamboot to examine each
step of lamboot and see what fails. Most situations where recon
succeeds and lamboot fails have to do with the hboot(1) command (that
lamboot invokes on each host in the hostfile).
-----
master@master:~$ lamboot -v clusterhosts

LAM 7.1.2/MPI 2 C++/ROMIO - Indiana University

n-1<9365> ssi:boot:base:linear: booting n0 (192.168.0.1)
n-1<9365> ssi:boot:base:linear: booting n1 (192.168.0.2)
n-1<9365> ssi:boot:base:linear: booting n2 (192.168.0.3)
n-1<9365> ssi:boot:base:linear: finished
master@master:~$ mpicc.openmpi hello.c -o hello
master@master:~$ mpiexec.openmpi -n 3 hello
Hello World from Node 0
Hello World from Node 1
Hello World from Node 2
master@master:~$
  
```

La distribución incluye un monitor de recursos llamado ganglia. Es posible acceder a el presionando el acceso directo llamado “Cluster monitor“. A través del navegador Web firefox se visualizara todos los recursos de CPU; memoria, red, etcétera. A continuación se observa una captura de pantalla de ganglia en funcionamiento:



## **5. CAMBIO DE CONFIGURACIÓN DE RED**

ABC GNU/Linux ha sido diseñada de tal manera que permite que el front-end pueda utilizar una segundo NIC para conectarse a Internet. Por defecto este NIC obtendra su IP mediante DHCP pero en los casos en los cuales el administrador del cluster quiera cambiar su configuración, deberá modificar el fichero “interfaces” el cual se encuentra en “/home/master/clusternet/”.