













Introducción aos clúster Beowulf

Sumario

Introdución aos clúster Beowulf.....	1
1 Convencións empregadas.....	3
2 Introdución.....	4
2.1 Arquitectura xeral.....	4
3 Configuración dun clúster Beowulf.....	4
3.1 Configuración da máquina Storage.....	5
3.1.1 Instalación do servidor NFS.....	5
3.1.2 Creación do cartafol compartido.....	5
3.1.3 Configuración do cartafol a exportar.....	5
3.1.4 Configuración dos parámetros de rede.....	5
3.2 Configuración da máquina Head.....	6
3.2.1 Instalación do cliente NFS.....	6
3.2.2 Creación do cartafol compartido.....	6
3.2.3 Configuración do cartafol a importar.....	6
3.2.4 Configuración dos parámetros de rede.....	6
3.2.5 Instalación de software.....	7
3.3 Configuración das máquinas worker nodes.....	7
4 Aplicación de exemplo.....	7

1 Convencións empregadas

	Esta icona fai referencia a notas de introdución
	Esta icona indica aclaración
	Esta icona fai referencia a arquivos de configuración, de rexistro...
	Esta icona indica casos de uso
	Esta icona fai referencia a avisos o advertencias
	Esta icona indica incidentes
	Esta icona fai referencia a sección que inclúen instrucións paso a paso
	Esta icona fai referencia a sección que inclúen capturas de pantalla
	Esta icona fai referencia a actividades
	Esta icona fai referencia a documento esencial (licenza: http://www.ohmyicons.com)
	Referencia a ligazón recomendada (licenza: http://iconleak.com)

2 Introducción

Un clúster Beowulf é un sistema de cómputo paralelo que se implementa empregando computadores de propósito xeral a través de redes informáticas estándar, sen empregar equipamento específico. Este modelo se desenvolveu por primeira vez no proxecto de investigación Earth and Space Sciences de NASA

2.1 Arquitectura xeral

O Clúster Beowulf posúe unha arquitectura baseada en multicomputadoras ou memoria distribuída, o cal pode ser utilizado para a computación paralela.

A arquitectura típica consta de al menos un nodo mestre (head ou master node) que é o lugar a onde se conectarán os usuarios do clúster, e un ou máis nodos escravos (worker nodes) que se encargan de executar as tarefas. Os nodos escravos son configurados e controlados polo nodo mestre, polo que fan soamente o que este lle indique, a comunicación é por medio dunha rede Ethernet.

A característica principal deste sistema de cómputo é a de ter memoria distribuída, na que se teñen espazos físicos e virtuais para cada procesador, cada un pode acceder á súa memoria local, pero non ás memorias doutros procesadores, desta forma os cambios que cada un realice na súa memoria local non teñen efecto sobre a memoria doutro procesador. Se un procesador debe acceder a un dato localizado na memoria doutro procesador, estes teríanse que comunicar vía mensaxe a través da rede de interconexión.

As direccións de memoria dun procesador non se mapean ao resto de procesadores, de modo que non existe o direccionamento global. Cando un proceso necesita acceder aos datos doutro, adoita ser tarefa do programador definir explicitamente como e cando se comuniquen. Outra tarefa importante do programador é lograr a sincronización. Para evitar ter que implementar todo isto a baixo nivel, existen librarías específicas que permiten xestionar este tipo de paradigma de cómputo dun xeito máis amigable para o desenvolvedor, coma as librarías de interfaz de paso de mensaxes (librarías MPI).

3 Configuración dun clúster Beowulf

O clúster Beowulf que imos a empregar está formado por varias máquinas que van a desempeñar os seguintes roles:

- **Storage:** máquina que vai a proporcionar o espazo de almacenamento compartido para o clúster. Neste caso se empregará o sistema de arquivos compartidos NFS.
- **Head ou master:** máquina que proporciona o acceso aos usuarios ao clúster. Tamén se encarga de coordinar os traballos que se executan no clúster
- **Worker node:** nodo de traballo que executa as tarefas solicitadas polos usuarios. Este tipo de nodos están controlados polo nodo head.

3.1 Configuración da máquina Storage

3.1.1 *Instalación do servidor NFS*

```
apt install nstall nfs-common nfs-kernel-server
```

3.1.2 *Creación do cartafol compartido*

```
mkdir /shared
```

```
chmod 777 /shared
```

3.1.3 *Configuración do cartafol a exportar*

```
cat /etc/exports
```

```
192.168.254.0/24(rw,sync,no_wdelay,subtree_check,root_squash) /shared
```

- `rw` allows both reads and writes to the filesystem
- `sync` is the synchronous option avoid server to respond requests before committing changes preventing data corruption
- `wdelay` is the Write delay option that allows the NFS server to put off committing changes to disk if it suspects that another write is coming shortly
- `subtree_check` improves security by checking not just that the client has rights to mount a directory, but all of the directory's subdirectories as well. Sub-tree checking is enabled by default, but the NFS server would complain if you don't specifically indicate it.
- `root_squash` prevents a root user on a machine using the filesystem to act as it if is the root user on the actual filesystem. This is more secure and is the default behaviour

```
exportfs -a
```

```
exports
```

3.1.4 *Configuración dos parámetros de rede*

Configurar a rede cos seguintes parámetros:

- `hostname`: storage
- `ip`: 192.168.254.2
- `mask`: 255.255.255.0

- gateway: 192.168.254.1

3.2 Configuración da máquina Head

3.2.1 *Instalación do cliente NFS*

```
apt install nfs-common
```

3.2.2 *Creación do cartafol compartido*

```
mkdir /shared
```

```
chmod 777 /shared
```

3.2.3 *Configuración do cartafol a importar*

```
cat /etc/fstab
```

```
192.168.254.2:/shared /shared nfs defaults 0 0
```

Configurar acceso passwordless ssh aos nodos dende o usuario de traballo do cluster

```
ssh-keygen -t rsa
```

```
cd .ssh/
```

```
cat id_rsa.pub > authorized_keys2
```

```
chmod 640 known_hosts
```

```
chmod 700 ../.ssh/
```

```
chmod 640 authorized_keys2
```

Comprobar acceso passwordless mediante:

```
ssh localhost
```

3.2.4 *Configuración dos parámetros de rede*

Configurar a rede cos seguintes parámetros:

- hostname: head
- ip: 192.168.254.3
- mask: 255.255.255.0
- gateway: 192.168.254.1

Configurar arquivo de hosts cos hostnames das diferentes máquinas.

```
cat /etc/hosts
```

```
120.0.0.1 localhost
```

```
192.168.254.2 storage
```

```
192.168.254.3 head
192.168.254.4 node1
192.168.254.5 node2
192.168.254.6 node3
192.168.254.7 node4
```

3.2.5 *Instalación de software*

```
apt install mpich
apt install openmpi-bin
apt install libopenmpi-dev
```

3.3 Configuración das máquinas worker nodes

Clonar o head e reconfigurar hostname, hosts, e configuración de rede

4 Aplicación de exemplo

Copiar o código fonte a /shared

```
#include <mpi.h>
#include <stdio.h>
#include <unistd.h>

int main(int argc, char **argv)
{
    int rank;
    char hostname[256];
    MPI_Init(&argc,&argv);
    MPI_Comm_rank(MPI_COMM_WORLD, &rank);
    gethostname(hostname,255);
    printf("Son o procesador numero: %d no equipo %s\n", rank,
hostname);
    MPI_Finalize();

    return 0;
}
```

```
}
```

Compilar con:

```
mpicc ola.c -o ola
```

Crear o arquivo machinefile co hostname dos worker nodes que se van a empregar na execução.

Executar con:

```
mpirun -np 2 --machinefile machinefile ./ola
```