

SISTEMAS COMPUTADORES DE ALTAS PRESTACIONES

PRACTICA 2



Universidad
de Huelva

Contenido

SISTEMAS COMPUTADORES DE ALTAS PRESTACIONES	1
PRACTICA 2	1
RESUMEN	3
PRIMERA PARTE:.....	4
Instalación de Ubuntu sobre Virtualbox	4
SEGUNDA PARTE	5
Instalación de MOSIX 4.4.4	5
Configurar red interna del clúster	6
Como instalar MOSIX:	8
Configurar MOSIX.....	9
TERCERA PARTE	10
Creando los Nodos restantes	10
Herramienta Mosmon.....	11
CUARTA PARTE	12
Creación del Benchmark	12
Ejecución del benchmark	12
Conclusión Benchmark	14

RESUMEN

En esta practica vamos a enfocarnos en configurar un clúster con el sistema operativo de Ubuntu, haciendo uso en el mismo de la herramienta MOSIX. Esta máquina nos permitirá conectar diferentes ordenadores conectados a la misma red, para poder crear un clúster con ellos.

PRIMERA PARTE:

Instalación de Ubuntu sobre Virtualbox

Lo primero que haremos será arrancar Virtualbox en nuestro pc y crear una maquina con nuestro SO, en nuestro caso Ubuntu Desktop.

Hemos utilizado Ubuntu Desktop directamente en vez de la versión sin entorno gráfico, aunque también se podría realizar con la versión de línea de comandos.

El motivo por el que vamos a utilizar la versión Desktop y no la server, como era previsto, es debido a que tras varios intentos (3-4 intentos) con la version original teníamos problemas tras descargar el Mosix en nuestra máquina. Estos problemas eran de privilegios en su mayoría, probamos diferentes soluciones de chmod para darle distintos permisos al user y al root, pero ninguno acabo siendo satisfactorio. Ya como medida extraordinaria probamos utilizando la versión Desktop y ahí si nos dejo trabajar sin problemas con el Mosix y poder realizar dicha práctica.

Características:

Tamaño de memoria :1024 MB de RAM

Disco Duro : 10 GB

Cpu: 2

Después de crear la máquina, nos pedirá un nombre de usuario y una contraseña. Ponemos ambas cosas a nuestro gusto.

Arrancamos nuestra maquina y pasamos al siguiente paso.

SEGUNDA PARTE

Instalación de MOSIX 4.4.4

Para la instalación de Mosix necesitaremos permisos root, abrimos una terminal y ponemos:

Sudo su

Nos pedirá una contraseña que será la misma que hayamos elegido antes para el usuario, una vez logeados en la terminal tendremos acceso a comando de super-usuario.

Comprobamos que tengamos conexión a internet abriendo el navegador o hacemos ping a Google.com desde la terminal.

Una vez que hemos comprobado que tenemos acceso a internet, deberemos actualizar los paquetes del sistema.

Apt-get update

Apt-get install build-essential

Apt-get install gcc

Apt-get install cmake

Algunos de estos paquetes pueden estar ya instalados o necesitar espacio adicional. De ser lo último se lo concederemos.

Prepararemos el entorno de instalación de MOSIX, para ello iremos a la siguiente pagina y descargaremos el programa:

www.mosix.cs.huji.ac.il/mos4/MOSIX-4.4.4.tbz

Lo descargaremos en el escritorio para no tener problemas de ningún tipo con los permisos, podemos usar el navegador **mozilla** o podemos usar la terminal con el comando **wget** y eligiendo primero el directorio donde queremos descargarlo
cd /Escritorio

Una vez descargado seguimos con la terminal, vamos al directorio donde lo hemos descargado con el comando `cd` y descomprimos el archivo con la herramienta TAR:

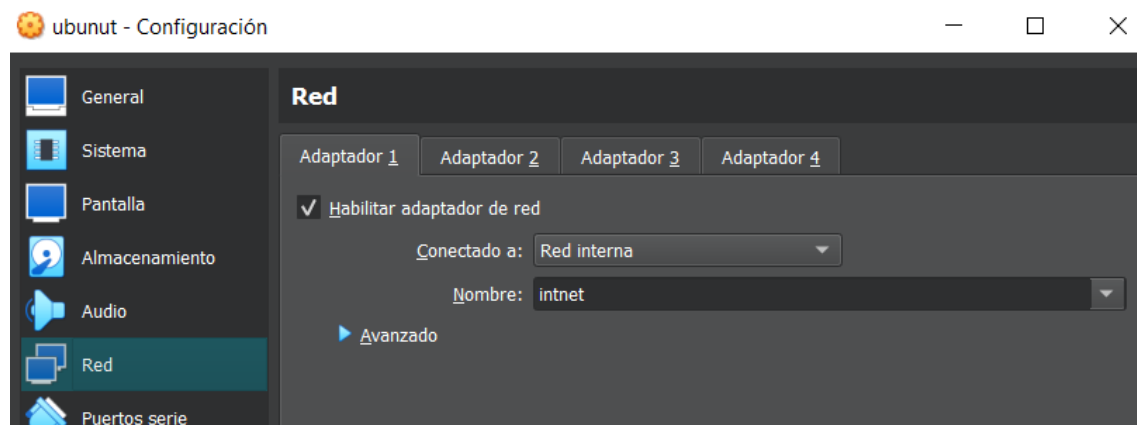
```
Tar -xjvf MOSIX-4.4.4.tbz
```

Ls (para comprobar que se ha descomprimido)

Ahora apagamos la maquina y nos vamos a virtualbox.

Configurar red interna del clúster

Una vez en virtualbox elegimos nuestra maquina Ubuntu y le damos a configuracion-red y elegimos red interna



Iniciaremos otra vez nuestra maquina e iniciamos sesión como root y abrimos una terminal.

Ifconfig

Podremos apreciar que no tenemos dirección ip asignada a eth0; para solucionar esto deberemos modificar el archivo de interfaces del sistema y otorgarle una ip estática.

Sustituiremos todo lo que nos ponga acerca de eth0 por lo siguiente:

Auto eth0

Iface eth0 inet static

Address 192.168.0.150

Netmask 255.255.255.0

Network 192.168.0.1

Gateway 192.168.0.1

Guardamos Ctrl + X

Una vez guardado reiniciamos el servicio de red:

Service networking restart

Y reiniciamos la maquina con reboot

Una vez arranque nuestra maquina y nos logeemos volveremos a comprobar la situación de nuestra interfaz de red eth0, con ifconfig eth0

```
root@rebu:/home/rebu# ifconfig eth0
eth0      Link encap:Ethernet  direcciónHW 08:00:27:a3:a9:44
          Direc. inet:192.168.0.150  Difus.:192.168.0.255  Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:fea3:a944/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST  MTU:1500  Métrica:1
          Paquetes RX:0 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:192 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:0 (0.0 B)  TX bytes:21224 (21.2 KB)
```

Debería devolvernos algo tal que así, una interfaz de red con las características que le establecimos en la configuración del sistema.

Como instalar MOSIX:

Abrimos una terminal, teniendo en cuenta que tenemos Mosix descomprimido en el escritorio.

Sudo su, para tener permisos de root

Y comenzamos con la instalación desde la terminal:

Cd /Escritorio/mosix-4.4.4

ls (para comprobar que estan todos los archivos)

./mosix.install

Arrancara el instalador de mosix y nos ira haciendo preguntas en los diferentes pasos.

Pulsamos Enter

Nivel de Mosix: 2

Pulsamos la opción 1 (añadir nodos al cluster)

Pulsamos “n”

Nos pide la Ip del nodo padre: 192.68.0.150

Número de nodos: 4

Pulsamos Enter

Pulsamos “q”

Update de number: yes

Autenticación: 2

Clave de seguridad: La que queráis, en nuestro caso pondremos 1234

Velocidad del procesador: 4

Ponemos 2000, Nos saldrá alguna notificación de que lo mismo es un recurso muy bajo para la capacidad de nuestro ordenador. No pasa nada.

Pulsamos “q”

Pulsamos “q”

Activar MOSIX ahora: yes

Podréis ver un resumen de la instalación que habéis realizado con el siguiente comando:

`/etc/init.d/mosix status`

```
root@rebu:/home/rebu/Escritorio/mosix-4.4.4# /etc/init.d/mosix status
MOSIX cluster is currently not configured!

MOSIX cluster is currently not configured!
daemons:
    Master Daemon: Down
    MOSIX Daemon : Down
    Remote Daemon: Down
    Postal Daemon: Down
Guest processes from other clusters in the multi-cluster: 0/0
```

Nos dirá que aun no está configurado y que no hay nodos, porque aun no hemos llegado al paso. Pero podremos ver que el mosix está instalado en el pc y en proceso.

Configurar MOSIX

Seguimos en la misma terminal que estábamos y en el mismo directorio.

`./mosconf`

Pulsamos Enter

Pulsamos 7

Le damos nombre al cluster, en nuestro caso le daremos de nombre mosix

Pulsamos “n”

Ip del primer nodo: 192.168.0.150 (nodo padre)

Número de nodos: 4

Are these nodes distant? No

Pulsamos “q”

Pulsamos Enter

Pulsamos “q”

Con esto ya tendríamos terminada la configuración del cluster, ahora solo queda ir creando los 3 nodos restantes para nuestro cluster.

TERCERA PARTE

Creando los Nodos restantes

Para crear los 3 nodos que nos quedan haremos lo siguiente:

1º Haremos 3 clones completos de la maquina que contiene el nodo padre

2º Arrancamos la maquina clon y nos logeamos como root

3º Abrimos la terminal y entramos en modo super usuario, sudo su

4º Cambiamos la dirección IP en el archivo interfaces

Nano /etc/network/interfaces

5º Editaremos el fichero cambiando la Ip por 192.168.0.15X (siendo X =1,2,3,...)

6º Reiniciamos el servicio de red para que coja la nueva IP

Service networking restart

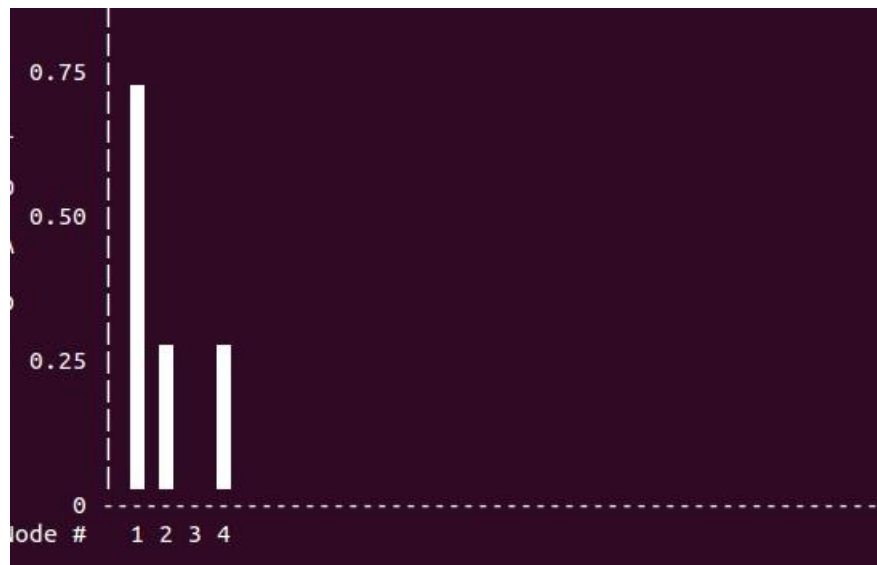
7º Reiniciamos la maquina

8º Repetimos el proceso con las otras dos máquinas clonadas incrementando X en uno.

Herramienta Mosmon

Ya con lo anterior tendríamos el cluster formado, y podemos ver si esto es correcto gracias a la herramienta mosmon.

Esta herramienta nos mostrara una grafica donde dira la cantidad de trabajo que desempeña cada nodo.



Si el cluster está bien formado deberán aparecer 4 nodos, como en la imagen.

Para arrancar la herramienta mosmon:

1º Encendemos las 4 maquinas

2º Nos logeamos como root

3º `cd /Escritorio/mosix/`

4º `./mosmon`

Una vez sigáis estos pasos deberá aparecer directamente dicha grafica en la terminal.

CUARTA PARTE

Creación del Benchmark

En nuestro caso para nuestro Benchmark utilizaremos un código que nos ha facilitado nuestro profesor, el cual no pongo en la memoria porque nos ocuparía 2 paginas de la misma innecesariamente.

Para crear el Benchmark podéis utilizar cualquier código que genere un bucle infinito y genere un estrés a la Cpu.

Lo primero que deberemos hacer una vez tengamos nuestro código en un documento de txt, es guárdalo como un .c

De tal forma que quede código.c, en nuestro caso a nuestro benchmark le llamaremos así: **código**

Una vez tenemos esto listo lo compilamos con la ayuda de la terminal para convertirlo en un ejecutable en segundo plano.

Gcc -o código código.c

Para esto es importante que estéis en el mismo directorio que el archivo a compilar.

Ejecución del benchmark

Para la ejecución de nuestro Benchmark vamos a apagar todas las maquinas, menos el nodo padre.

Una vez solo este arrancado el nodo padre, ejecutaremos el siguiente comando en la terminal de este:

Mosrun ./código 1 5000000000

Siendo 1 el numero de procesos y 50... el número de ejecuciones.

Esto nos devolverá unos datos por la terminal como los de la foto:

```
root@rebu:/home/rebu/Escritorio# mosrun ./codigo 1 5000000000
process 0 launched...
process 0 finished...
time: 101.745811
```

Lo siguiente será realizar lo mismo, pero arrancando 1 a 1 los nodos hijos e incrementando el número de procesos según la cantidad de nodos estén arrancados, de la siguiente forma:

2 nodos:

```
mosrun ./código 1 5000000000
```

```
mosrun ./código 2 5000000000
```

3 nodos:

```
mosrun ./código 1 5000000000
```

```
mosrun ./código 2 5000000000
```

```
mosrun ./código 3 5000000000
```

4 nodos:

```
mosrun ./código 1 5000000000
```

```
mosrun ./código 2 5000000000
```

```
mosrun ./código 3 5000000000
```

```
mosrun ./código 4 5000000000
```

Como se puede apreciar en la imagen anterior, el programa una vez acaba nos da un tiempo de ejecución, de cuanto ha tardado en ejecutar los benchmark.

1 NODO	2 NODOS	3 NODOS	4 NODOS
101.7459	55.569	38.6027	32.487

Conclusión Benchmark

Respecto al resultado del Benchmark, en lo personal creo que he entendido el porque de estos resultados.

Empezando por el 1º caso que tenemos un solo nodo trabajando, nos da un tiempo de procesado de unos 100 segundos, pero en cuanto activamos el 2º nodo ya el tiempo disminuye a casi la mitad del tiempo. Entonces podríamos decir que se ha notado bastante el reparto de la carga de trabajo entre nodos y que nuestro cluster funciona correctamente.

Lo curioso viene cuando empezamos a encender mas nodos. Ya una vez encendemos el 3º nodo el tiempo de procesado no baja drásticamente, si no que baja un poco mas de un 15%, que sigue siendo una buena bajada. Pero comparado con el casi 50% del caso anterior es algo más normal.

Esto debe ser a mi parecer porque conforme más nodos vamos añadiendo al cluster, si no hacemos alguna configuración concreta puede entorpecer más la comunicación entre nodos, que la potencia que aporta añadir un nodo extra.

Y esta teoría se recalca sobre todo cuando activamos el 4 nodo, que nuestro tiempo de ejecución, baja unos escasos segundos. Entonces en nuestro caso a mi parecer rentaría más tener 3 nodos activos en este caso que 4. Hablando de tema de recursos mal gastados.