

## Procedimiento para Instalar Elastix como Cluster

Con la ayuda de este artículo vamos a implementar el servicio de alta disponibilidad para Elastix con el uso de las herramientas DRBD y Heartbeat que permiten crear en tiempo real un sistema de espejado remoto de dispositivos de bloques locales (similar a un RAID 1 pero por TCP). Ambos servidores se verán como una sola entidad.

Para comprender un poco mejor lo que estamos haciendo procederemos a explicar los componentes necesarios para crear el Cluster.

### DRBD

- Que es drbd?
- Instalación
- Configuración
- Pruebas

Que es DRBD?

DRBD es un dispositivo de bloques que está diseñado para construir "clusters" de alta disponibilidad. Esto es hecho por medio de una copia idéntica de todo un dispositivo de bloques a través de una red. Esto podría ser visto como un RAID-1 de red.

¿Cual es el alcance de DRBD?

DRBD toma los datos, los escribe en el disco local y los envía al otro "host". En el otro "host", éste pone los datos en el disco.

Pongamos el siguiente ejemplo:

Tenemos nodo1 "master-elastix.com" con Dirección IP 192.168.25.124 y nodo2 "slave-elastix.com" con dirección IP 192.168.25.125, las particiones físicas son el nodo1 /dev/sda3 y en el nodo2 /dev/sda3, pueden ser cualquier partición, en lo posible que sean del mismo tamaño ya que si no vamos a desperdiciarlo que sobre de la más grande "RAID1".

Heartbeat

- Que es y que hace?
- Instalación
- Configuración

Que es y que hace heartbeat?

El software heartbeat trabaja enviando latidos (ping), los cuales verifican si el nodo principal está activo o no, estos pings enviados por heartbeat requieren una respuesta por parte del nodo principal, si al cabo de un cierto tiempo el nodo no responde dichos ping, heartbeat determina que ese nodo se encuentra inactivo o caído y automáticamente activa el nodo secundario para que asuma el control de la red y los servicios, en nuestro ejemplo vamos a usar lo siguiente para mantener la línea del ejemplo con drbd:

nodo1 con dirección IP 192.168.25.124 (Nodo principal)

nodo2 con dirección IP 192.168.25.125 (Nodo secundario)

Con este funcionamiento tenemos que el administrador de red, puede estar tranquilo, ya que en caso de que ocurra un problema con el nodo principal, el nodo secundario asumirá el control automáticamente, luego tendrá que trabajar para volver a colocar el línea el nodo primario caído producto de un fallo de hardware, pero lo bueno es que uno puede realizar dicho trabajo sin preocuparse, ya que el secundario está cubriendo los servicios del primario, con esto lo que logramos es una alta disponibilidad de servicios, como veremos en este caso, estaremos dando alta disponibilidad al servidor de telefonía Elastix.

## Instalación

Ahora que ya comprendimos lo que vamos hacer, procederemos a instalar los programas necesarios para nuestro cluster de Elastix.

1.- Cuando arrancamos con el CD de instalación de Elastix, en el prompt de boot escribimos “advanced” para realizar el particionamiento manual del disco duro.

Cuando aparezca la pantalla de particiones seleccionar la última opción que se refiere a crear particiones. Después eliminar todas las particiones y crearles a como se muestra a continuación:

Suponiendo que nuestro disco duro es /dev/sda, un esquema podría ser:  
/dev/sda1 (ext3) para /  
/dev/sad2 (swap)

Dejar suficiente espacio libre para lo que deseamos replicar en ambos servidores (Esta partición la vamos a crear después)  
Estas dos particiones no deben ocupar todo el tamaño del disco, ya que debemos dejar espacio suficiente para crear la partición donde van a residir los datos que van a ser replicados entre ambos servidores, este espacio libre debería ser similar en ambos servidores.

En nuestro caso, el servidor Primario se llamara master-elastix.com y el servidor Secundario se llamara slave-elastix.com

Seguimos con los pasos para la instalación normal de Elastix.

Para la implementación de clustering necesitamos los siguientes RPMs:

- drbd
- kmod-drbd
- OpenIPMI-lib
- heartbeat-pils
- openhpi
- heartbeat-stonith
- heartbeat

Puede descargarlos de <http://asterisk.aplitel.info/asterisk.htm>

Copiar todos los RPM en el directorio /usr/src/

```
wget http://asterisk.aplitel.com/files/drbd-8.0.11-1.el5.centos.i386.rpm
wget http://asterisk.aplitel.com/files/kmod-drbd-8.0.12-1.2.6.18\_53.1.19.el5.i686.rpm
wget http://asterisk.aplitel.com/files/OpenIPMI-lib-2.0.6-5.el5.4.i386.rpm
wget http://asterisk.aplitel.com/files/heartbeat-pils-2.1.2-3.el5.centos.i386.rpm
wget http://asterisk.aplitel.com/files/openhpi-2.8.1-2.el5.7.i386.rpm
wget http://asterisk.aplitel.com/files/heartbeat-stonith-2.1.2-3.el5.centos.i386.rpm
wget http://asterisk.aplitel.com/files/heartbeat-2.1.2-3.el5.centos.i386.rpm
```

Instalar los RPMs en el mismo orden en que se listan, así:

```
rpm -ivh drbd-x.y.z-xyz.rpm
```

Ahora vamos a crear la partición que contendrá los datos que se replicaran, para esto ejecutamos el comando:

```
fdisk /dev/sda
```

Luego siguiendo la ayuda del comando creamos una tercera partición que sería /dev/sda3 de tipo ext3 que ocupara el tamaño que dejamos libre en el particionamiento manual a tiempo de instalación.

Pasos

- 1.- Presionamos la letra n (add a new partition)
- 2.- Presionamos la letra p (Primary partition)
- 3.- Escribimos el número 3 (Partition number)
- 4.- Presionamos Enter un par de veces hasta que nos aparezca el Command de nuevo.
- 5.- Luego presionamos la letra t para cambiar el partition's system id.
- 6.- Escribimos la partición 3
- 7.- En HEX code escribimos 83 que es la partición de Linux
- 8.- Por ultimo escribimos la letra w para salvar los cambios.
- 9.- Después reiniciamos el servidor para que la nueva tabla este disponible.

Ahora procederemos a formatear la nueva partición con el siguiente comando:

```
mke2fs -j /dev/sda3
```

DRBD (<http://www.drbd.org/>) nos permitirá replicar la partición declarada en ambos miembros del cluster, en nuestro caso /dev/sda3. Para realizar esto crea un dispositivo de bloques virtual llamado /dev/drbd0.

Vamos a configurar el archivo /etc/drbd.conf, así:

```
resource "r0" {
  protocol A;
  disk { on-io-error pass_on; }
  startup { wfc-timeout 5; degr-wfc-timeout 3; }
  syncer { rate 100M; }
  on master.telesoft.com {
    device /dev/drbd0;
    disk /dev/sda3;
    address 192.168.25.124:7789;
    meta-disk internal;
  }
  on slave.telesoft.com {
    device /dev/drbd0;
    disk /dev/sda3;
    address 192.168.25.125:7789;
    meta-disk internal;
  }
}
```

Donde:

master-elastic.com (192.168.25.124) Servidor Primario

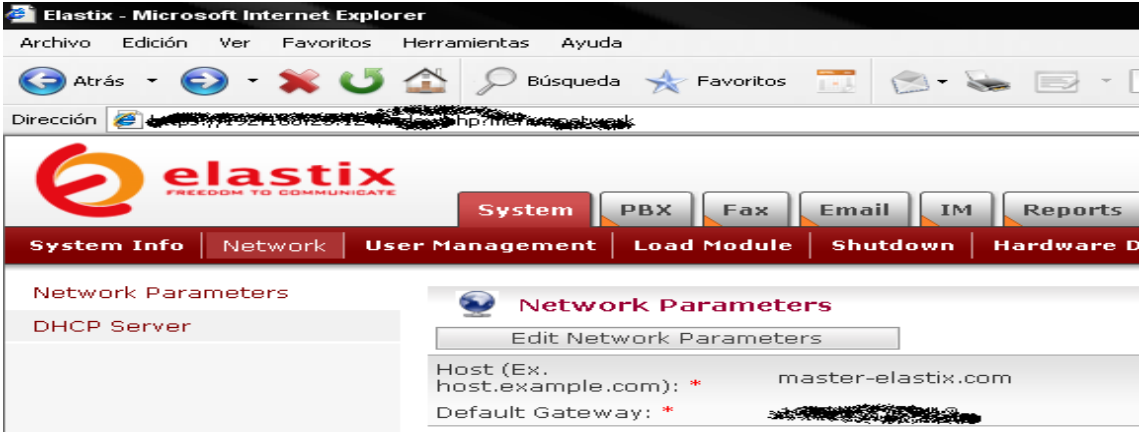
slave-elastic.com (192.168.25.125) Servidor Secundario

Se recomienda que el archivo /etc/hosts agregar

192.168.25.124 master.telesoft.com master

192.168.25.125 slave.telesoft.com slave

Antes de seguir se debe de cambiar el nombre del servidor en la interface web de Elastix:



Ahora procederemos a ejecutar los siguientes comandos

```
dd if=/dev/zero bs=1M count=1 of=/dev/sda3; sync
```

El archivo /etc/drbd.conf debe ser el mismo para ambos servidores, ahora creamos el dispositivo de bloques virtual /dev/drbd0, ejecutando también en ambos servidores el comando:

```
drbdadm create-md r0
```

Levantamos el servicio:

```
service drbd start
```

Con esto se empezaran a sincronizar las particiones, puedo visualizarlo ejecutando el comando:

```
service drbd status
```

Y obtendremos una salida como esta:

```
[root@master-elastix ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.0.3 (api:86/proto:86)
SVN Revision: 2881 build by buildsvn@c5-i386-build, 2007-05-13 08:22:43
0: cs:SyncSource st:Secondary/Secondary ds:UpToDate/Inconsistent A r---
ns:96764 nr:0 dw:0 dr:104928 al:0 bm:91 lo:1 pe:4 ua:256ap:0
[=====>.....] sync'ed: 22.0% (368336/464976)K
finish: 0:00:30 speed: 12,080 (12,080) K/sec
resync: used:2/31 hits:6291 misses:8 starving:0 dirty:0 changed:8
act_log: used:0/127 hits:0 misses:0 starving:0 dirty:0 changed:0
```

Inicialmente ambos servidores serán Secundarios, debo declarar quien será el Primario.

Ahora SOLO en el servidor PRIMARIO:

```
drbdsetup /dev/drbd0 primary -o
```

Y obtendré una salida como esta:

```
[root@master-elastic ~]# service drbd status
drbd driver loaded OK; device status:
version: 8.0.3 (api:86/proto:86)
SVN Revision: 2881 build by buildsvn@c5-i386-build, 2007-05-13 08:22:43
0: cs:Connected st:Primary/Secondary ds:UpToDate/UpToDate A r---
ns:464976 nr:0 dw:0 dr:464976 al:0 bm:170 lo:0 pe:0 ua:0 ap:0
resync: used:0/31 hits:28977 misses:85 starving:0 dirty:0 changed:85
act_log: used:0/127 hits:0 misses:0 starving:0 dirty:0 changed:0
```

Podemos saber el ROL del servidor donde nos encontramos, ejecutando el comando:

```
drbdadm state r0
```

En el servidor Primario deberiamos ver:

```
[root@master-elastic ~]# drbdadm state r0
Primary/Secondary
```

Y en el servidor Secundario:

```
[root@slave-elastic ~]# drbdadm state r0
Secondary/Primary
```

Ahora ya es posible montar la particion virtual /dev/drbd0, pero antes se debe formatear usando ext3, así:

```
mke2fs -j /dev/drbd0
mkdir /replica
mount /dev/drbd0 /replica
```

Procederemos a copiar dentro de /replica la data que queramos replicar, incluyendo archivos de configuración de servicios, en el caso de Elastix serian los siguientes directorios:

- /etc/asterisk
- /var/lib/asterisk
- /usr/lib/asterisk
- /var/spool/asterisk
- /var/lib/mysql
- /var/log/asterisk

A modo de ejemplo lo haremos con /etc/asterisk, así:

```
cd /replica  
tar -zcvf etc-asterisk.tgz /etc/asterisk  
tar -zxvf etc-asterisk.tgz  
rm -rf /etc/asterisk  
ln -s /replica/etc/asterisk /etc/asterisk
```

Recordar de reiniciar el servicio de mysql una vez finalizado este paso

```
service mysqld restart
```

Lo mismo debemos hacer con los demás directorios.

Notas

Para declarar un nodo como primario (role)

```
drbdadm primary r0
```

Para declara un nodo como secundario (role)

```
drbdadm secondary r0
```

Acordarse de que para declarar como secundario hay que desmontar la unida (/replica) y antes de desmontarla detener el servicio de mysqld

```
umount /replica
```



Ahora se debe proceder a instalar Heartbeat, que es el software que hace el cluster propiamente.

### Configuración de Heartbeat

Acordarse de apagar los servicios que se iniciaran con el heartbeat

```
chkconfig asterisk off
chkconfig zaptel off
chkconfig mysqld off
chkconfig httpd off
```

Esto se hace ya que el Heartbeat iniciara estos servicios cuando el servidor sea master.

Debemos copiar los archivos de configuración de Heartbeat (<http://www.linux-ha.org/>) de la documentación que viene con el paquete al directorio de configuración /etc/ha.d, así:

```
cp /usr/share/doc/heartbeat-2.1.2/ha.cf /etc/ha.d/
cp /usr/share/doc/heartbeat-2.1.2/haresources /etc/ha.d/
cp /usr/share/doc/heartbeat-2.1.2/authkeys /etc/ha.d/
```

Ahora vamos a editar el archivo /etc/ha.d/ha.cf

```
cd /etc/ha.d
vi ha.cf
```

Y agregamos al final estas líneas:

```
debugfile /var/log/ha-debug
logfile /var/log/ha-log
logfacility local0
keepalive 2
deadtime 20
warntime 10
initdead 40
udpport 694
bcast eth0 # Linux
auto_failback off
node master.telesoft.com
node slave.telesoft.com
```

Luego el archivo /etc/ha.d/haresources.

```
cd /etc/ha.d  
vi haresources
```

Y agregamos al final la línea:

```
master.telesoft.com drbddisk::r0 Filesystem::/dev/drbd0::/replica::ext3 IPaddr::192.168.25.126/24/eth0/192.168.25.255 asterisk zaptel mysqlld httpd
```

Finalmente el archivo /etc/ha.d/authkeys.

```
cd /etc/ha.d
```

Y agregamos al final las líneas:

```
auth 1  
1 sha1 elastix
```

Le damos el siguiente permiso al archivo /etc/ha.d/authkeys

```
cd /etc/ha.d  
chmod 600 authkeys
```

Todos los archivos anteriores deben ser iguales en ambos servidores.

Configuramos heartbeat con soporte chkconfig, así:

```
chkconfig --add heartbeat
```

Ahora, antes de iniciar los servicios de heartbeat, debemos reiniciar el servicio drbd en ambos servidores, así:

```
service drbd restart
```

Verificamos que ambos servidores se encuentren en estado Secundario, así:

```
drbdadm state r0
```

```
[root@master-elastix ~]# drbdadm state r0  
Secondary/Secondary
```

Si es así, procedemos a levantar el servicio heartbeat, así:

```
service heartbeat start
```

Una vez hecho esto, esperaremos unos momentos hasta que heartbeat monte la partición /replica en el servidor Primario, lo verificamos así:

```
drbdadm state r0
```

```
[root@master-elastic ~]# drbdadm state r0  
Primary/Secondary
```

```
df -h
```

```
[root@master-elastic ~]# df -h  
Filesystem      Size  Used Avail Use% Mounted on  
/dev/hdc1       9.5G  1.5G  7.6G  17% /  
tmpfs           501M   0  501M   0% /dev/shm  
/dev/drbd0      9.2G  150M  8.6G   2% /replica
```

La partición /replica solo se montara en el servidor cuyo estado sea Primario, es decir, si por algún motivo el servidor master-elastic.com falla entonces slave-elastic.com asumirá el estado de Primario y por ende levantara la partición /replica, la cual contendrá la información replicada entre ambos equipos.

Esperar a que termine de hacer la sincronización de los dos servidores, este proceso se puede observar con el comando:

```
watch cat /proc/drbd
```

Cuanto tenemos los dos nodos en modo stand alone, con un cs:StandAlone st:Secondary/Unknown ds:UpToDate/DUnknown lo más probable es que tengamos que realizar una resincronización de la data, para ello basta con ejecutar:

En el servidor que no está UpToDate

```
drbdadm -- --discard-my-data connect all
```

Luego en el servidor que esta UpToDate

```
drbdadm connect all
```

Los nodos empezaran a re-sincronizarse, transfiriendo los datos desde el nodo con datos OK, al nodo con los datos corruptos.