

INTRODUCCIÓN A RDBD

JOSÉ DOMINGO MUÑOZ

IES GONZALO NAZARENO

DICIEMBRE 2021



DRBD (Distributed Replicated Block Device) es un sistema de almacenamiento replicado distribuido para la plataforma Linux.

Es parecido a un RAID1, pero en este caso, se hace la duplicación de los datos entre diferentes **dispositivos de bloque** en diferentes hosts a través de la red.

DRBD es software libre desarrollado por la empresa LinBit. [Documentación oficial.](#)



El **recurso DRBD** es el espacio de almacenamiento y la información relacionada administrada por DRBD. Un recurso tendrá cuatro características:

- Nombre
- Nombre del dispositivo de bloque que se crea (ejemplo: /etc/drbd1)
- Configuración del dispositivo de bloque que se va a replicar en cada nodo.
- Configuración de red de los nodos del cluster.



Podemos configurar DRBD en dos modos:

- **Modo “Un sólo primario” (Single-primary):** Uno de los nodos es el primario y el que puede montar el recurso. El otro nodo es el secundario y no podrá montar el recurso. Normalmente cuando el recurso lo formateamos con un sistema de archivo tradicional. Se utiliza para cluster activo-pasivo.
- **Modo “Dos primarios” (Dual-primary):** En este caso los dos nodos se configuran como primario, por lo que pueden montar el recurso al mismo tiempo. Es necesario usar un sistema de ficheros distribuidos (OCFS2, GlusterFS, ...). Se utiliza en cluster activo-activo.



CONFIGURACIÓN DEL MODO SINGLE-PRIMARY (1)

Tenemos un escenario donde tenemos dos máquinas. Cada máquina tiene conectado un disco, que es el dispositivo de bloque que vamos a duplicar.

Lo primero que haremos es instalar las herramientas del espacio de usuario de DRBD:

```
apt install drbd-utils
```



CONFIGURACIÓN DEL MODO SINGLE-PRIMARY (2)

Creamos el recurso DRBD. Creamos el fichero `/etc/drbd.d/datos.res`:

```
resource datos {  
  protocol C;  
  meta-disk internal;  
  device /dev/drbd1;  
  syncer {  
    verify-alg sha1;  
  }  
  net {  
    allow-two-primaries;  
  }  
  on nodo1 {  
    disk /dev/sdb;  
    address 10.1.1.101:7789;  
  }  
  on nodo2 {  
    disk /dev/sdb;  
    address 10.1.1.102:7789;  
  }  
}
```



CONFIGURACIÓN DEL MODO SINGLE-PRIMARY (3)

En la creación del recurso hemos indicado:

- El nombre: `resource datos`
- El dispositivo de bloque que se va a crear: `device /dev/drbd1;`
- Los dispositivos de bloque que se van a replicar en cada nodo: `disk /dev/sdb;`
- Las direcciones ip de cada nodo y el puerto que se va a utilizar para la sincronización.



CONFIGURACIÓN DEL MODO SINGLE-PRIMARY (4)

A continuación vamos a crear el recurso drbd y lo vamos a activar en ambos nodos:

```
drbdadm create-md datos  
drbdadm up datos
```

Asignamos el nodo1 como primario y el nodo2 como secundario, por lo tanto **ejecutamos en el nodo1:**

```
drbdadm primary --force datos
```



CONFIGURACIÓN DEL MODO SINGLE-PRIMARY (5)

Y comprobamos que empieza la sincronización de discos:

```
$ drbdadm status datos
datos role:Primary
    disk:UpToDate
    peer role:Secondary
        replication:SyncSource peer-disk:Inconsistent done:1.86
```

Trascurrido un tiempo comprobamos que los discos ya están sincronizados:

```
$ drbdadm status datos
datos role:Primary
    disk:UpToDate
    peer role:Secondary
        replication:Established peer-disk:UpToDate
```

CONFIGURACIÓN DEL MODO SINGLE-PRIMARY (6)

Podemos ver la característica de nuestros recursos DRBD:

```
$ cat /proc/drbd
version: 8.4.10 (api:1/proto:86-101)
srcversion: 9B4D87C5E865DF526864868
```

```
1: cs:Connected ro:Primary/Secondary ds:UpToDate/UpToDate C r-----
   ns:530108 nr:0 dw:5872 dr:526461 al:14 bm:0 lo:0 pe:0 ua:0 ap:
```



CONFIGURACIÓN DEL MODO SINGLE-PRIMARY (7)

Y finalmente podemos formatear el dispositivo de bloque, montarlo y crear un fichero index.html en el (todo esto se ejecuta en el nodo primario, **en el nodo1**):

```
$ apt install xfsprogs  
  
$ mkfs.xfs /dev/drbd1  
  
$ mount /dev/drbd1 /mnt  
$ cd /mnt/  
$ echo "<h1>Prueba</h1>" >> index.html  
$ umount /mnt
```



CONFIGURACIÓN DEL MODO SINGLE-PRIMARY (8)

Si en el nodo2 (que tiene el papel de secundario) intentamos montar el recurso:

```
$ mount /dev/drbd1 /mnt  
mount: /mnt: mount(2) system call failed: Wrong medium type.
```



CONFIGURACIÓN DEL MODO DUAL-PRIMARY

Si queremos que los dos nodos tengan montado simultaneamente el recurso tenemos que configurar los dos nodos como primario. En el **nod02**:

```
$ drbdadm primary datos
$ drbdadm status datos
datos role:Primary
    disk:UpToDate
    peer role:Primary
        replication:Established
            peer-disk:UpToDate
$ mount /dev/drbd1 /mnt
```

Para que esto funcione de forma adecuada tenemos que usar un **Sistema de Ficheros Distribuidos** (por ejemplo OCFS2, GlusterFS,...)

