

P2 - iSCSI y OCFS2

Creamos las particiones

```
fdisk /dev/sdc
```

```
fdisk /dev/sdd
```

```
fdisk /dev/sde
```

```
fdisk /dev/sdf
```

Creamos los dispositivos raid 1:

```
mdadm --create md_RAID1_1 --level=raid1 --raid-devices=2 /dev/sdc1 /dev/sde1
```

```
mdadm --create md_RAID1_2 --level=raid1 --raid-devices=2 /dev/sdd1 /dev/sdf1
```

Creamos el RAID 0:

```
mdadm --create md_RAID10 --level=raid0 --raid-devices=2 /dev/md/RAID1_1 /dev/md/RAID1_2
```

Crear un grupo de volúmenes LVM empleando como volumen

físico el array RAID-10 anterior:

```
vgcreate vg_cda /dev/md/md_RAID10
```

Creamos los volúmenes lógicos:

```
lvcreate -n UNO -L +50M vg_cda
```

```
lvcreate -n DOS -L +50M vg_cda
```

```
lvcreate -n COMPARTIDO -L +50M vg_cda
```

Crear target (servidor que expone los dispositivos de bloque) asignando un iqn y un identificador interno (tid=1)

```
tgtadm --lld iscsi --mode target --op new --tid=1 --targetname iqn.2024-10.net.cda.discos:prueba
```

```
tgtadm --lld iscsi --mode target --op new --tid=2 --targetname iqn.2024-10.net.cda.discos:prueba2
```

Añadir luns a los targets

```
tgtadm --lld iscsi --mode logicalunit --op new --tid 1 --lun 1 --backing-store /dev/vg_cda/UNO
```

```
tgtadm --lld iscsi --mode logicalunit --op new --tid 1 --lun 3 --backing-store
```

```
/dev/vg_cda/COMPARTIDO
```

```
tgtadm --lld iscsi --mode logicalunit --op new --tid 2 --lun 2 --backing-store /dev/vg_cda/UNO
```

```
tgtadm --lld iscsi --mode logicalunit --op new --tid 2 --lun 3 --backing-store
```

```
/dev/vg_cda/COMPARTIDO
```

Restringir acceso por ip

```
tgtadm --lld iscsi --mode target --op bind --tid 1 --initiator-address 192.168.100.22
```

```
tgtadm --lld iscsi --mode target --op bind --tid 2 --initiator-address 192.168.100.33
```

Autenticación por usuario y contraseña.

```
tgtadm --lld iscsi --mode account --op new --user cda --password cdapass
```

```
tgtadm --lld iscsi --mode account --op bind --tid 1 --user cda
```

```
tgtadm --lld iscsi --mode account --op bind --tid 2 --user cda
```

Configurar los *initiators* !open-iscsi! de CLIENTE1 y CLIENTE2

```
"InitiatorName="`iscsi-iname` > /etc/iscsi/initiatorname.iscsi
```

```
"InitiatorName="`iscsi-iname` > /etc/iscsi/initiatorname.iscsi
```

Habilitar la autenticación mediante CHAP y establecer las credenciales de acceso en `/etc/iscsi/iscsid.conf` (en ambos *initiators*)

```
nano /etc/iscsi/iscsid.conf
```

--- CONTENIDO A EDITAR ----

```
...
# *****
# CHAP Settings
# *****
node.session.auth.authmethod = CHAP
node.session.auth.username = cda
node.session.auth.password = cdapass
...
```

--- CONTENIDO A EDITAR ----

Reiniciar el demonio `iscsid` para que cargue la nueva configuración (en ambos *initiators*)

```
systemctl restart iscsid.service
```

Consultar la lista de `iqn` de los *targets* disponibles en la máquina `discos.cda.net` (192.168.100.11) (comando `sendtargets` del modo `discover` desde ambos *initiators*)

```
iscsiadm --mode discovery --type sendtargets --portal 192.168.100.11
192.168.100.11:3260,1 iqn.2020-10.net.cda.discos:pruebas
```

Conectar con el *target* seleccionado de la máquina `discos.cda.net` (192.168.100.11) indicando su `iqn` (comando `login` del modo `node` desde ambos *initiators*)

```
iscsiadm --m node --targetname iqn.2020-10.net.cda.discos:pruebas --portal 192.168.100.11 --
login
```

Declarar el *cluster* formado por las máquinas `CLIENTE1` y `CLIENTE2`

`service o2cb load`: Fuerza la carga de los módulos del kernel necesarios

- `service o2cb online [nombre cluster]`: Pone en marcha el cluster indicado (o todos los que estén definidos en `/etc/ocfs2/cluster.conf`, si no se especifica un nombre), incluyendo la incorporación del nodo donde se invoca a dicho cluster/es.
- `service o2cb start`: Combina las dos acciones anteriores
- `service o2cb offline [nombre cluster]`: Finalizar el cluster indicado (o todos los que estén definidos en `/etc/ocfs2/cluster.conf`, si no se especifica un nombre)
- `service o2cb unload`: Descarga del kernel los módulos necesarios para dar soporte a OCSFS2.
- `service o2cb stop`: Combina las dos acciones anteriores

- service o2cb configure: En sistemas Debian está desahilitado y delega en una configuración "manual" o mediante dpkg-reconfigure ocsfs2-tools
- service o2cb status: Muestra el estado del servicio O2CB.
- La definición del cluster está en el fichero /etc/ocfs2/cluster.conf y debe ser la misma en todos los nodos del cluster (de modo que estos puedan comunicarse entre ellos)
 - Puede editarse manualmente o mediante las herramientas o2cb_ctl (línea de comandos) o ocfs2console (interfaz gráfico)
- En los sistemas basados en Debian la configuración del script o2cb con los parámetros a usar por el cluster (timeouts, nombre de cluster a arrancar) está en /etc/default/o2cb

Editar el fichero de definición del cluster /etc/ocfs2/cluster.conf en CLIENTE1 (creándolo si no existe)

```
root@cliente1:~# nano /etc/ocfs2/cluster.conf
```

cluster:

```
heartbeat_mode = local
node_count = 2
name = clustercda
```

node:

```
number = 0
cluster = clustercda
ip_port = 7777
ip_address = 192.168.100.22
name = cliente1
```

node:

```
number = 1
cluster = clustercda
ip_port = 7777
ip_address = 192.168.100.33
name = cliente2
```

Copiar el fichero /etc/ocfs2/cluster.conf a CLIENTE2 (o volver a crearlo desde cero)

```
root@cliente1:~# scp /etc/ocfs2/cluster.conf root@192.168.100.33:/etc/ocfs2/
```

```
nano /etc/default/o2cb
```

```
# O2CB_ENABLED: 'true' means to load the driver on boot.
```

```
O2CB_ENABLED=true
```

```
# O2CB_BOOTCLUSTER: If not empty, the name of a cluster to start.
```

```
O2CB_BOOTCLUSTER=clustercda
```

```
...
```

Arrancar/reiniciar el servicio o2cb en ambas máquinas para cargar la nueva configuración

```
service o2cb stop
```

```
service o2cb start
```

```
service o2cb status
```

Formatear y montar los dispositivos.

```
mkfs.ext3 /dev/vg_cda/UNO  
mkfs.ext3 /dev/vg_cda/DOS  
mkfs.ocfs2 /dev/vg_cda/COMPARTIDO
```

```
mkdir /mnt/uno
```

```
mkdir /mnt/dos
```

```
mkdir /mnt/compartido
```

```
mount /dev/vg_cda/UNO /mnt/uno(Cliente1)
```

```
mount /dev/vg_cda/DOS /mnt/dos(cliente2)
```

```
mount -t ocfs2 /dev/vg_cda/COMPARTIDO /mnt/compartido(cualquier cliente)
```