

Dernière mise-à-jour : 2021/01/24 12:16

HAR400 - Gestion de la Mutualisation du Stockage

Présentation

La mutualisation du stockage s'obtient en utilisant un **SAN** (*Storage Area Network*) ou un **NAS** (*Network Attached Storage*). La différence entre les deux réside dans la façon que le client voit le stockage :

- le SAN est vu par le client comme un disque interne. Les ressources d'un SAN sont mutualisées permettant ainsi une gestion centralisée du stockage, une évolution du stockage ainsi que des fonctions de réplication,
- le NAS est partagé sur le réseau en utilisant un protocole tel **CIFS**, **NFS** et/ou **AppleShare**.

Le SAN utilise un réseau en fibre optique et un des protocoles suivants :

- **Fibre Channel**,
- **FCoE** ou *Fibre Channel over Ethernet*,
- **iSCSI**.

Nomenclature

Le vocabulaire du SAN est résumé dans le tableau suivant :

Terme	Description
Fabric	L'ensemble d'un SAN
Director	Un switch fibre channel
GLM (<i>Global Link Module</i>)	Le raccordement entre le réseau fibre et le réseau ethernet dans le cas de l'utilisation du FCoE
HBA (<i>Host Bus Adaptors</i>)	Les adaptateurs fibre dans les clients
FA (<i>Fibre Adaptors</i>)	Les ports fibres d'un baie de stockage
Zoning	Le découpage du SAN en sous-réseaux, uniquement accessible à certains clients

Terme	Description
mapping	La présentation de certains volumes logiques de la baie sur un ou plusieurs FA spécifiques
masking	La restriction d'accès à certains volumes logiques
multipathing	Le regroupement de plusieurs chemins vers le même SAN

LAB #1 - CLVM et GFS

Configurer le Serveur

Vous allez simuler un SAN avec iSCSI. Commencez par installer le paquet **scsi-target-utils** :

```
[root@centos ~]# yum install scsi-target-utils
```

Le paquet contient la commande **tgtd** et la commande **tgtadm**.

Les options de la commande **tgtd** sont :

```
[root@centos ~]# tgtd --help
Usage: tgtd [OPTION]
Target framework daemon, version 1.0.14
  -f, --foreground          make the program run in the foreground
  -C, --control-port NNNN  use port NNNN for the mgmt channel
  -d, --debug debuglevel  print debugging information
  -V, --version            print version and exit
  -h, --help              display this help and exit
```

Les options de la commande **tgtadm** sont :

```
[root@centos ~]# tgtadm --help
Usage: tgtadm [OPTION]
Linux SCSI Target Framework Administration Utility, version 1.0.14
```

```
--lld <driver> --mode target --op new --tid <id> --targetname <name>
    add a new target with <id> and <name>. <id> must not be zero.
--lld <driver> --mode target --op delete --tid <id>
    delete the specific target with <id>. The target must
    have no activity.
--lld <driver> --mode target --op show
    show all the targets.
--lld <driver> --mode target --op show --tid <id>
    show the specific target's parameters.
--lld <driver> --mode target --op update --tid <id> --name <param> --value <value>
    change the target parameters of the specific
    target with <id>.
--lld <driver> --mode target --op bind --tid <id> --initiator-address <src>
    enable the target to accept the specific initiators.
--lld <driver> --mode target --op unbind --tid <id> --initiator-address <src>
    disable the specific permitted initiators.
--lld <driver> --mode logicalunit --op new --tid <id> --lun <lun> \
    --backing-store <path> --bstype <type> --bsoflags <options>
    add a new logical unit with <lun> to the specific
    target with <id>. The logical unit is offered
    to the initiators. <path> must be block device files
    (including LVM and RAID devices) or regular files.
    bstype option is optional.
    bsoflags supported options are sync and direct
    (sync:direct for both).
--lld <driver> --mode logicalunit --op delete --tid <id> --lun <lun>
    delete the specific logical unit with <lun> that
    the target with <id> has.
--lld <driver> --mode account --op new --user <name> --password <pass>
    add a new account with <name> and <pass>.
--lld <driver> --mode account --op delete --user <name>
    delete the specific account having <name>.
--lld <driver> --mode account --op bind --tid <id> --user <name> [--outgoing]
    add the specific account having <name> to
```

```
the specific target with <id>.
<user> could be <IncomingUser> or <OutgoingUser>.
If you use --outgoing option, the account will
be added as an outgoing account.
--lld <driver> --mode account --op unbind --tid <id> --user <name>
delete the specific account having <name> from specific
target.
--control-port <port> use control port <port>
--help                display this help and exit
```

Report bugs to <stgt@vger.kernel.org>.

Démarrez ensuite le service **tgtd** :

```
[root@centos ~]# service tgtd start
Démarrage de SCSI target daemon : [ OK ]
```

Configurez maintenant le service avec **chkconfig** :

```
[root@centos ~]# chkconfig tgtd on
[root@centos ~]# chkconfig --list tgtd
tgtd          0:arrêt    1:arrêt    2:marche    3:marche    4:marche    5:marche    6:arrêt
```

Créez maintenant un pseudo-disque avec la commande **dd** :

```
[root@centos ~]# dd if=/dev/zero of=/pseudo-iscsi bs=1M count=256
256+0 enregistrements lus
256+0 enregistrements écrits
268435456 octets (268 MB) copiés, 2,34575 s, 114 MB/s
```

Créez d'abord une cible à laquelle vous pouvez rajouter le disque :

```
[root@centos ~]# tgtadm --lld iscsi --op new --mode target --tid 1 -T target
```

Consultez la configuration actuelle :

```
[root@centos ~]# tgtadm --lld iscsi --op show --mode target
Target 1: target
  System information:
    Driver: iscsi
    State: ready
  I_T nexus information:
  LUN information:
    LUN: 0
      Type: controller
      SCSI ID: IET      00010000
      SCSI SN: beaf10
      Size: 0 MB, Block size: 1
      Online: Yes
      Removable media: No
      Readonly: No
      Backing store type: null
      Backing store path: None
      Backing store flags:
  Account information:
  ACL information:
```

Ajoutez maintenant le disque au premier LUN :

```
[root@centos ~]# tgtadm --lld iscsi --op new --mode logicalunit --tid 1 --lun 1 -b /pseudo-iscsi
```

Consultez la configuration :

```
[root@centos ~]# tgtadm --lld iscsi --op show --mode target
Target 1: target
  System information:
    Driver: iscsi
    State: ready
```

```
I_T nexus information:
LUN information:
  LUN: 0
    Type: controller
    SCSI ID: IET      00010000
    SCSI SN: beaf10
    Size: 0 MB, Block size: 1
    Online: Yes
    Removable media: No
    Readonly: No
    Backing store type: null
    Backing store path: None
    Backing store flags:
  LUN: 1
    Type: disk
    SCSI ID: IET      00010001
    SCSI SN: beaf11
    Size: 268 MB, Block size: 512
    Online: Yes
    Removable media: No
    Readonly: Yes
    Backing store type: rdwr
    Backing store path: /pseudo-iscsi
    Backing store flags:
Account information:
ACL information:
```

Vérifiez que le port **3260** est bien ouvert et en état d'écoute :

```
[root@centos ~]# netstat -apn | grep 3260
tcp        0      0 0.0.0.0:3260      0.0.0.0:*        LISTEN     2948/tgtd
tcp        0      0 :::3260          :::*              LISTEN     2948/tgtd
```

Configurez le serveur pour que tous les clients puissent y avoir accès :

```
[root@centos ~]# tgtadm --lld iscsi --op bind --mode target --tid 1 -I ALL
```

Constatez la modification de la section ACL :

```
[root@centos ~]# tgtadm --lld iscsi --op show --mode target
```

Target 1: target

System information:

Driver: iscsi

State: ready

I_T nexus information:

LUN information:

LUN: 0

Type: controller

SCSI ID: IET 00010000

SCSI SN: beaf10

Size: 0 MB, Block size: 1

Online: Yes

Removable media: No

Readonly: No

Backing store type: null

Backing store path: None

Backing store flags:

LUN: 1

Type: disk

SCSI ID: scsi001

SCSI SN: 0123456789

Size: 268 MB, Block size: 512

Online: Yes

Removable media: No

Readonly: Yes

Backing store type: rdwr

Backing store path: /pseudo-iscsi

Backing store flags:

Account information:

```
ACL information:  
ALL
```

Dernièrement configurez LUN1 en mode r/w :

```
[root@centos /]# tgtadm --lld iscsi --mode logicalunit --op update --tid 1 --lun 1 --params readonly=0
```

Vérifiez votre configuration :

```
[root@centos /]# tgtadm --lld iscsi --op show --mode target
```

Target 1: target

System information:

Driver: iscsi

State: ready

I_T nexus information:

LUN information:

LUN: 0

Type: controller

SCSI ID: IET 00010000

SCSI SN: beaf10

Size: 0 MB, Block size: 1

Online: Yes

Removable media: No

Readonly: No

Backing store type: null

Backing store path: None

Backing store flags:

LUN: 1

Type: disk

SCSI ID: scsi001

SCSI SN: 0123456789

Size: 268 MB, Block size: 512

Online: Yes

Removable media: No


```
Readonly: No
Backing store type: rdwr
Backing store path: /pseudo-iscsi
Backing store flags:
Account information:
ACL information:
ALL
```

Mise en Place d'un Initiateur sur les Noeuds

Pour accéder à notre cible ISCSI, le client doit disposer d'un **initiateur**. Installez donc le paquet **iscsi-initiator-utils** :

```
[root@node1 ~]# yum install iscsi-initiator-utils
```

```
[root@node2 ~]# yum install iscsi-initiator-utils
```

```
[root@node3 ~]# yum install iscsi-initiator-utils
```

Lancez maintenant la découverte des cibles sur le serveur iscsi :

```
[root@node1 ~]# iscsiadm --mode discovery --type sendtargets --portal 10.0.3.18
10.0.3.18:3260,1 target
```

```
[root@node2 ~]# iscsiadm --mode discovery --type sendtargets --portal 10.0.3.18
10.0.3.18:3260,1 target
```

```
[root@node3 ~]# iscsiadm --mode discovery --type sendtargets --portal 10.0.3.18
10.0.3.18:3260,1 target
```

Connectez-vous à la cible à partir de chaque noeud :

```
[root@node1 ~]# iscsiadm --mode node --targetname target --login
```

```
Logging in to [iface: default, target: target, portal: 10.0.3.18,3260] (multiple)
Login to [iface: default, target: target, portal: 10.0.3.18,3260] successful.
```

```
[root@node2 ~]# iscsiadm --mode node --targetname target --login
Logging in to [iface: default, target: target, portal: 10.0.3.18,3260] (multiple)
Login to [iface: default, target: target, portal: 10.0.3.18,3260] successful.
```

```
[root@node3 ~]# iscsiadm --mode node --targetname target --login
Logging in to [iface: default, target: target, portal: 10.0.3.18,3260] (multiple)
Login to [iface: default, target: target, portal: 10.0.3.18,3260] successful.
```

CLVM - Clustered LVM

CLVM fournit des volumes logiques **LVM2** au niveau d'un cluster et utilise de daemon **clvmd** afin ajouter des extensions cluster aux outils LVM classiques. Avec des modifications minimales au fichier de configuration LVM - **/etc/lvm/lvm.conf** - il est ainsi possible d'utiliser es commandes classiques de LVM2 afin de configurer et gérer des volumes logiques au niveau du cluster. Le daemon clvmd fonctionne sur chaque noeud et distribue les mises-à-jour des métadonnées LVM aux autres noeuds lors d'une modification. Lors d'une modification, l'accès au stockage physique est verrouillé en utilisant le gestionnaire de verrouillage **DLM** du cluster.

A partir de node01, créez un volume logique sur la cible iscsi :

```
[root@node1 ~]# pvcreate /dev/sdb
Physical volume "/dev/sdb" successfully created

[root@node1 ~]# vgcreate -cy vg0 /dev/sdb
Clustered volume group "vg0" successfully created

[root@node1 ~]# lvcreate -L 240 -n lv0 vg0
Logical volume "lv0" created
```

Vérifiez que le service clvmd voit le volume logique :

```
[root@node1 ~]# service clvmd status
clvmd (pid 2012) en cours d'exécution...
Clustered Volume Groups: vg0
Active clustered Logical Volumes: lv0
```

Modifiez la directive **locking_type** dans le fichier **/etc/lvm/lvm.conf** sur chaque noeud :

```
...
# Type of locking to use. Defaults to local file-based locking (1).
# Turn locking off by setting to 0 (dangerous: risks metadata corruption
# if LVM2 commands get run concurrently).
# Type 2 uses the external shared library locking_library.
# Type 3 uses built-in clustered locking.
# Type 4 uses read-only locking which forbids any operations that might
# change metadata.
locking_type = 3
...
```

Vérifiez ensuite que la volume logique est visible à partir de tous les noeuds :

```
[root@node1 ~]# vgs
VG   #PV #LV #SN Attr   VSize   VFree
vg0   1   1   0 wz--nc 252,00m 12,00m
[root@node1 ~]# lvs
LV   VG   Attr      LSize   Pool Origin Data%  Move Log Cpy%Sync Convert
lv0  vg0  -wi-a---- 240,00m

[root@node2 ~]# vgs
VG   #PV #LV #SN Attr   VSize   VFree
vg0   1   1   0 wz--nc 252,00m 12,00m
[root@node2 ~]# lvs
LV   VG   Attr      LSize   Pool Origin Data%  Move Log Cpy%Sync Convert
lv0  vg0  -wi-a---- 240,00m
[root@node2 ~]#
```

```
[root@node3 ~]# vgs
VG   #PV #LV #SN Attr   VSize   VFree
vg0   1   1   0 wz--nc 252,00m 12,00m
[root@node3 ~]# lvs
LV   VG   Attr      LSize   Pool Origin Data%  Move Log Cpy%Sync Convert
lv0  vg0  -wi-a---- 240,00m
[root@node3 ~]#
```



Notez l'attribut **c** dans la sortie de la commande **vgs**. Ceci implique que la vg est clusterisé.

GFS2

Red Hat GFS est un système de fichiers en cluster qui utilise :

- des **métadonnées distribués**,
- plusieurs **fichiers de journaux**,
- un **gestionnaire de verrouillage**,

Les modifications aux données effectuées par un noeud sur un système de fichiers GFS sont immédiatement visibles aux autres noeuds

Le système de fichiers GFS fournit une image unique du système de fichiers pour tous les noeuds permettant, entre autre :

- la simplification de l'infrastructure de données,
- la simplification de l'installation et de l'application de correctifs,
- l'absence de duplication des données de l'application,
- l'accès concurrent aux données en lecture/écriture par plusieurs clients,
- la simplification de la restauration de pertes de données,
- la maximisation de l'utilisation des ressources de stockage,
- la réduction des coûts d'administration du stockage,

- la prise en charge du multipathing,
- la gestion des volumes via CLVM.

Notez que lors de la création du système de fichiers gfs2 en utilisant les paramètres par défaut celle-ci renvoie une erreur **Not enough space available on device**. Cette erreur est causée par le fait que la taille par défaut des trois journaux est supérieur à la taille de **pseudo-iscsi** :

```
[root@node1 ~]# mkfs.gfs2 -p lock_dlm -t moncluster:mydata -j 3 /dev/mapper/vg0-lv0
This will destroy any data on /dev/mapper/vg0-lv0.
It appears to contain: symbolic link to `../dm-0'

Are you sure you want to proceed? [y/n] y

Not enough space available on device
```

Créez donc un système de fichiers gfs2 ayant 3 journaux de 32Mo chacun :

```
[root@node1 ~]# mkfs.gfs2 -p lock_dlm -t moncluster:mydata -j 3 -J 32 /dev/vg0/lv0
This will destroy any data on /dev/vg0/lv0.
It appears to contain: symbolic link to `../dm-0'

Are you sure you want to proceed? [y/n] y

Device:                /dev/vg0/lv0
Blocksize:             4096
Device Size            0,23 GB (61440 blocks)
Filesystem Size:       0,23 GB (61437 blocks)
Journals:              3
Resource Groups:       1
Locking Protocol:      "lock_dlm"
Lock Table:            "moncluster:mydata"
UUID:                  28dc27df-e31e-a427-c58a-ec465058f043
```

Montez ensuite le volume logique dans chaque noeud :

```
[root@node1 ~]# mount -t gfs2 -o acl,noatime /dev/mapper/vg0-lv0 /mydata
```

```
[root@node2 ~]# mount -t gfs2 -o acl,noatime /dev/mapper/vg0-lv0 /mydata
```

```
[root@node3 ~]# mount -t gfs2 -o acl,noatime /dev/mapper/vg0-lv0 /mydata
```



Testez maintenant le système de fichiers gfs2 en créant un fichier à partir de node01. Est-ce que ce fichier est visible à partir de node02 et node03 ? Editez ce fichier avec vi à partir de node01. Ouvrez en même temps le fichier avec vi à partir de node02. Que se passe-t-il ?

LAB #2 - Gestion du Multipathing

Configurer le Serveur

Installez le dépôt RPMFORGE :

```
[root@centos ~]# rpm -Uvh https://pkgs.org/centos-6/repoforge-i386/rpmforge-release-0.5.3-1.el6.rf.i686.rpm.html
```

Installez le dépôt atomic :

```
[root@centos ~]# wget -q -O - http://www.atomicorp.com/installers/atomic | sh
```

Atomic Archive installer, version 2.0.5

BY INSTALLING THIS SOFTWARE AND BY USING ANY AND ALL SOFTWARE
PROVIDED BY ATOMICORP LIMITED YOU ACKNOWLEDGE AND AGREE:

THIS SOFTWARE AND ALL SOFTWARE PROVIDED IN THIS REPOSITORY IS PROVIDED BY ATOMICORP LIMITED AS IS, IS UNSUPPORTED AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ATOMICORP LIMITED, THE COPYRIGHT OWNER OR ANY CONTRIBUTOR TO ANY AND ALL SOFTWARE PROVIDED BY OR PUBLISHED IN THIS REPOSITORY BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Do you agree to these terms? (yes/no) [Default: yes] yes

Configuring the [atomic] yum archive for this system

Installing the Atomic GPG key: OK

Downloading atomic-release-1.0-14.el6.art.noarch.rpm: OK

The Atomic Rocket Turtle archive has now been installed and configured for your system

The following channels are available:

atomic	- [ACTIVATED]	- contains the stable tree of ART packages
atomic-testing	- [DISABLED]	- contains the testing tree of ART packages
atomic-bleeding	- [DISABLED]	- contains the development tree of ART packages

Installez le dépôt EPEL :

```
[root@centos ~]# rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-6.noarch.rpm
```

Vous allez simuler un SAN avec iSCSI. Commencez par installer le paquet **scsi-target-tools** :

```
[root@centos ~]# yum install scsi-target-tools
```

Le paquet contient la commande **tgtd** et la commande **tgtadm**.

Les options de la commande **tgtd** sont :

```
[root@centos ~]# tgtd --help
Usage: tgtd [OPTION]
Target framework daemon, version 1.0.14
  -f, --foreground           make the program run in the foreground
  -C, --control-port NNNN   use port NNNN for the mgmt channel
  -d, --debug debuglevel    print debugging information
  -V, --version              print version and exit
  -h, --help                display this help and exit
```

Les options de la commande **tgtadm** sont :

```
[root@centos ~]# tgtadm --help
Usage: tgtadm [OPTION]
Linux SCSI Target Framework Administration Utility, version 1.0.14

--lld <driver> --mode target --op new --tid <id> --targetname <name>
                                add a new target with <id> and <name>. <id> must not be zero.
--lld <driver> --mode target --op delete --tid <id>
                                delete the specific target with <id>. The target must
                                have no activity.
--lld <driver> --mode target --op show
                                show all the targets.
--lld <driver> --mode target --op show --tid <id>
                                show the specific target's parameters.
--lld <driver> --mode target --op update --tid <id> --name <param> --value <value>
                                change the target parameters of the specific
```



```
target with <id>.
--lld <driver> --mode target --op bind --tid <id> --initiator-address <src>
enable the target to accept the specific initiators.
--lld <driver> --mode target --op unbind --tid <id> --initiator-address <src>
disable the specific permitted initiators.
--lld <driver> --mode logicalunit --op new --tid <id> --lun <lun> \
--backing-store <path> --bstype <type> --bsoflags <options>
add a new logical unit with <lun> to the specific
target with <id>. The logical unit is offered
to the initiators. <path> must be block device files
(including LVM and RAID devices) or regular files.
bstype option is optional.
bsoflags supported options are sync and direct
(sync:direct for both).
--lld <driver> --mode logicalunit --op delete --tid <id> --lun <lun>
delete the specific logical unit with <lun> that
the target with <id> has.
--lld <driver> --mode account --op new --user <name> --password <pass>
add a new account with <name> and <pass>.
--lld <driver> --mode account --op delete --user <name>
delete the specific account having <name>.
--lld <driver> --mode account --op bind --tid <id> --user <name> [--outgoing]
add the specific account having <name> to
the specific target with <id>.
<user> could be <IncomingUser> or <OutgoingUser>.
If you use --outgoing option, the account will
be added as an outgoing account.
--lld <driver> --mode account --op unbind --tid <id> --user <name>
delete the specific account having <name> from specific
target.
--control-port <port> use control port <port>
--help display this help and exit
```

Report bugs to <stgt@vger.kernel.org>.

Démarrez ensuite le service **tgtd** :

```
[root@centos ~]# service tgtd start
Démarrage de SCSI target daemon : [ OK ]
```

Configurez maintenant le service avec **chkconfig** :

```
[root@centos ~]# chkconfig tgtd on
[root@centos ~]# chkconfig --list tgtd
tgtd          0:arrêt    1:arrêt    2:marche    3:marche    4:marche    5:marche    6:arrêt
```

Créez maintenant un pseudo-disque avec la commande **dd** :

```
[root@centos ~]# dd if=/dev/zero of=/pseudo-iscsi bs=1M count=256
256+0 enregistrements lus
256+0 enregistrements écrits
268435456 octets (268 MB) copiés, 2,34575 s, 114 MB/s
```

Créez d'abord une cible à laquelle vous pouvez rajouter le disque :

```
[root@centos ~]# tgtadm --lld iscsi --op new --mode target --tid 1 -T target
```

Consultez la configuration actuelle :

```
[root@centos ~]# tgtadm --lld iscsi --op show --mode target
Target 1: target
  System information:
    Driver: iscsi
    State: ready
  I_T nexus information:
  LUN information:
    LUN: 0
    Type: controller
```

```
SCSI ID: IET      00010000
SCSI SN: beaf10
Size: 0 MB, Block size: 1
Online: Yes
Removable media: No
Readonly: No
Backing store type: null
Backing store path: None
Backing store flags:
Account information:
ACL information:
```

Ajoutez maintenant le disque au premier LUN :

```
[root@centos ~]# tgtadm --lld iscsi --op new --mode logicalunit --tid 1 --lun 1 -b /pseudo-iscsi
```

Consultez la configuration :

```
[root@centos ~]# tgtadm --lld iscsi --op show --mode target
Target 1: target
  System information:
    Driver: iscsi
    State: ready
  I_T nexus information:
  LUN information:
    LUN: 0
      Type: controller
      SCSI ID: IET      00010000
      SCSI SN: beaf10
      Size: 0 MB, Block size: 1
      Online: Yes
      Removable media: No
      Readonly: No
      Backing store type: null
```

```
Backing store path: None
Backing store flags:
LUN: 1
  Type: disk
  SCSI ID: IET      00010001
  SCSI SN: beaf11
  Size: 268 MB, Block size: 512
  Online: Yes
  Removable media: No
  Readonly: Yes
  Backing store type: rdwr
  Backing store path: /pseudo-iscsi
  Backing store flags:
Account information:
ACL information:
```

Configurez maintenant le deuxième LUN :

```
[root@centos ~]# tgtadm --lld iscsi --op new --mode logicalunit --tid 1 --lun 2 -b /pseudo-iscsi
```

Consultez la configuration :

```
[root@centos ~]# tgtadm --lld iscsi --op show --mode target
Target 1: target
  System information:
    Driver: iscsi
    State: ready
  I_T nexus information:
  LUN information:
    LUN: 0
      Type: controller
      SCSI ID: IET      00010000
      SCSI SN: beaf10
      Size: 0 MB, Block size: 1
```

Online: Yes
Removable media: No
Readonly: No
Backing store type: null
Backing store path: None
Backing store flags:

LUN: 1

Type: disk
SCSI ID: IET 00010001
SCSI SN: beaf11
Size: 268 MB, Block size: 512
Online: Yes
Removable media: No
Readonly: Yes
Backing store type: rdwr
Backing store path: /pseudo-iscsi
Backing store flags:

LUN: 2

Type: disk
SCSI ID: IET 00010002
SCSI SN: beaf12
Size: 268 MB, Block size: 512
Online: Yes
Removable media: No
Readonly: Yes
Backing store type: rdwr
Backing store path: /pseudo-iscsi
Backing store flags:

Account information:

ACL information:

<note important> Notez que les SCSI ID et les SCSI SN des deux LUNs ne sont pas identiques. Dans l'état donc, le multipathing ne fonctionnera pas car celui-ci nécessite à ce que les deux enregistrements soient identiques. </note>

L'option **-params** de la commande **tgtadm** permet de modifier les paramètres de la configuration :

```
[root@centos ~]# tgtadm --lld iscsi --op update --mode logicalunit --tid 1 --lun 1 --params
vendor_id="Fenestr0s",product_id="MonDisqueISCSI",product_rev="1.0",scsi_id="scsi001",scsi_sn="0123456789"
[root@centos ~]# tgtadm --lld iscsi --op update --mode logicalunit --tid 1 --lun 2 --params
vendor_id="Fenestr0s",product_id="MonDisqueISCSI",product_rev="1.0",scsi_id="scsi001",scsi_sn="0123456789"
```

Consultez maintenant la configuration :

```
[root@centos ~]# tgtadm --lld iscsi --op show --mode target
Target 1: target
  System information:
    Driver: iscsi
    State: ready
  I_T nexus information:
  LUN information:
    LUN: 0
      Type: controller
      SCSI ID: IET      00010000
      SCSI SN: beaf10
      Size: 0 MB, Block size: 1
      Online: Yes
      Removable media: No
      Readonly: No
      Backing store type: null
      Backing store path: None
      Backing store flags:
    LUN: 1
      Type: disk
      SCSI ID: scsi001
      SCSI SN: 0123456789
      Size: 268 MB, Block size: 512
      Online: Yes
      Removable media: No
```

```
    Readonly: Yes
    Backing store type: rdwr
    Backing store path: /pseudo-iscsi
    Backing store flags:
LUN: 2
    Type: disk
    SCSI ID: scsi001
    SCSI SN: 0123456789
    Size: 268 MB, Block size: 512
    Online: Yes
    Removable media: No
    Readonly: Yes
    Backing store type: rdwr
    Backing store path: /pseudo-iscsi
    Backing store flags:
Account information:
ACL information:
```

Vérifiez que le port **3260** est bien ouvert et en état d'écoute :

```
[root@centos ~]# netstat -apn | grep 3260
tcp        0      0 0.0.0.0:3260      0.0.0.0:*        LISTEN     2948/tgtd
tcp        0      0 :::3260          :::*              LISTEN     2948/tgtd
```

Configurez le serveur pour que tous les clients puissent y avoir accès :

```
[root@centos /]# tgtadm --lld iscsi --op bind --mode target --tid 1 -I ALL
```

Constatez la modification de la section ACL :

```
[root@centos /]# tgtadm --lld iscsi --op show --mode target
Target 1: target
  System information:
    Driver: iscsi
```

State: ready

I_T nexus information:

LUN information:

LUN: 0

Type: controller

SCSI ID: IET 00010000

SCSI SN: beaf10

Size: 0 MB, Block size: 1

Online: Yes

Removable media: No

Readonly: No

Backing store type: null

Backing store path: None

Backing store flags:

LUN: 1

Type: disk

SCSI ID: scsi001

SCSI SN: 0123456789

Size: 268 MB, Block size: 512

Online: Yes

Removable media: No

Readonly: Yes

Backing store type: rdwr

Backing store path: /pseudo-iscsi

Backing store flags:

LUN: 2

Type: disk

SCSI ID: scsi001

SCSI SN: 0123456789

Size: 268 MB, Block size: 512

Online: Yes

Removable media: No

Readonly: Yes

Backing store type: rdwr


```
        Backing store path: /pseudo-iscsi
        Backing store flags:
Account information:
ACL information:
    ALL
```

Dernièrement configurez LUN1 et LUN2 en mode r/w :

```
[root@centos /]# tgtadm --lld iscsi --mode logicalunit --op update --tid 1 --lun 1 --params readonly=0
[root@centos /]# tgtadm --lld iscsi --mode logicalunit --op update --tid 1 --lun 2 --params readonly=0
```

Vérifiez votre configuration :

```
[root@centos /]# tgtadm --lld iscsi --op show --mode target
Target 1: target
  System information:
    Driver: iscsi
    State: ready
  I_T nexus information:
  LUN information:
    LUN: 0
      Type: controller
      SCSI ID: IET      00010000
      SCSI SN: beaf10
      Size: 0 MB, Block size: 1
      Online: Yes
      Removable media: No
      Readonly: No
      Backing store type: null
      Backing store path: None
      Backing store flags:
    LUN: 1
      Type: disk
      SCSI ID: scsi001
```

```
SCSI SN: 0123456789
Size: 268 MB, Block size: 512
Online: Yes
Removable media: No
Readonly: No
Backing store type: rdwr
Backing store path: /pseudo-iscsi
Backing store flags:
LUN: 2
Type: disk
SCSI ID: scsi001
SCSI SN: 0123456789
Size: 268 MB, Block size: 512
Online: Yes
Removable media: No
Readonly: No
Backing store type: rdwr
Backing store path: /pseudo-iscsi
Backing store flags:
Account information:
ACL information:
ALL
```

Mise en Place d'un Initiateur

Pour accéder à notre cible ISCSI, le client doit disposer d'un **initiateur**. Dans le cadre de cet exemple, la machine virtuelle va jouer aussi le rôle de client. Installez donc le paquet **iscsi-initiator-utils** :

```
[root@centos ~]# yum install iscsi-initiator-utils
```

Lancez maintenant la découverte des cibles sur le serveur :

```
[root@centos ~]# iscsiadm --mode discovery --type sendtargets --portal 127.0.0.1
Démarrage de iscsid : [ OK ]
127.0.0.1:3260,1 target
```

Connectez-vous à la cible :

```
[root@centos ~]# iscsiadm --mode node --targetname target --login
Logging in to [iface: default, target: target, portal: 127.0.0.1,3260] (multiple)
Login to [iface: default, target: target, portal: 127.0.0.1,3260] successful.
```

Consultez le journal **/var/log/messages** :

```
...
Jun 25 16:18:53 centos kernel: sd 3:0:0:2: [sdc] Attached SCSI disk
Jun 25 16:18:53 centos kernel: sd 3:0:0:1: [sdb] Attached SCSI disk
...
```

<note important> Notez que les deux disques SCSI sont vus comme /dev/sdb et /dev/sdc. </note>

Introduction au Multipathing

Installez le paquet **device-mapper-multipath** :

```
[root@centos ~]# yum install device-mapper-multipath
```

Chargez le module **dm_multipath** :

```
[root@centos ~]# modprobe dm_multipath
```

Démarrez le service **multipathd** :

```
[root@centos ~]# service multipathd start
```

Démarrage du démon multipathd : [OK]

Activez le service multipathd :

```
[root@centos ~]# chkconfig multipathd on
[root@centos ~]# chkconfig --list multipathd
multipathd      0:arrêt    1:arrêt    2:marche    3:marche    4:marche    5:marche    6:arrêt
```

Copiez le fichier **/usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf** vers **/etc/multipath.conf** :

```
[root@centos ~]# cp /usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf /etc/multipath.conf
```

Le fichier **/etc/multipath.conf** par défaut est :

```
[root@centos ~]# cat /etc/multipath.conf
# This is a basic configuration file with some examples, for device mapper
# multipath.
# For a complete list of the default configuration values, see
# /usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf.defaults
# For a list of configuration options with descriptions, see
# /usr/share/doc/device-mapper-multipath-0.4.9/multipath.conf.annotated

## By default, devices with vendor = "IBM" and product = "S/390.*" are
## blacklisted. To enable mulitpathing on these devies, uncomment the
## following lines.
#blacklist_exceptions {
#    device {
#        vendor      "IBM"
#        product     "S/390.*"
#    }
#}

## Use user friendly names, instead of using WWIDs as names.
defaults {
```

```
    user_friendly_names yes
}
##
## Here is an example of how to configure some standard options.
##
#
#defaults {
#  udev_dir          /dev
#  polling_interval  10
#  selector          "round-robin 0"
#  path_grouping_policy  multibus
#  getuid_callout     "/lib/udev/scsi_id --whitelisted --device=/dev/%n"
#  prio              alua
#  path_checker       readsector0
#  rr_min_io          100
#  max_fds            8192
#  rr_weight          priorities
#  failback           immediate
#  no_path_retry       fail
#  user_friendly_names  yes
#}
##
## The wwid line in the following blacklist section is shown as an example
## of how to blacklist devices by wwid. The 2 devnode lines are the
## compiled in default blacklist. If you want to blacklist entire types
## of devices, such as all scsi devices, you should use a devnode line.
## However, if you want to blacklist specific devices, you should use
## a wwid line. Since there is no guarantee that a specific device will
## not change names on reboot (from /dev/sda to /dev/sdb for example)
## devnode lines are not recommended for blacklisting specific devices.
##
#blacklist {
#  wwid 26353900f02796769
#  devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
```

```
# devnode "^hd[a-z]"
#}
#multipaths {
#  multipath {
#    wwid          3600508b4000156d700012000000b0000
#    alias         yellow
#    path_grouping_policy multibus
#    path_checker   readsector0
#    path_selector  "round-robin 0"
#    failback      manual
#    rr_weight      priorities
#    no_path_retry  5
#  }
#  multipath {
#    wwid          1DEC_____321816758474
#    alias         red
#  }
#}
#devices {
#  device {
#    vendor          "COMPAQ  "
#    product         "HSV110 (C)COMPAQ"
#    path_grouping_policy multibus
#    getuid_callout  "/lib/udev/scsi_id --whitelisted --device=/dev/%n"
#    path_checker    readsector0
#    path_selector   "round-robin 0"
#    hardware_handler "0"
#    failback        15
#    rr_weight        priorities
#    no_path_retry   queue
#  }
#  device {
#    vendor          "COMPAQ  "
#    product         "MSA1000  "
```

```
#    path_grouping_policy    multibus
#    }
#}
```

Les directives actives de ce fichier sont :

[multipath.conf](#)

```
defaults {
    user_friendly_names yes
}
```

La directive **user_friendly_names** permet au système de s'adresser au périphérique multipath en utilisant un nom simple comme **mpathb** au lieu du **WWID** (*World Wide Identifier*) qui prend la forme **lscsi001**.

La correspondance des deux noms est écrite dans le fichier **/etc/multipath/bindings** dès la mise en place du multipath :

```
# Multipath bindings, Version : 1.0
# NOTE: this file is automatically maintained by the multipath program.
# You should not need to edit this file in normal circumstances.
#
# Format:
# alias wwid
#
mpatha lATA      VBOX HARDDISK          VB831a6136-0ecd289a
mpathb lscsi001
```

Pour mettre en place le multipath, il convient d'utiliser la commande **multipath** :

```
[root@centos ~]# multipath
```

Saisissez ensuite la même commande avec l'option **-l** pour voir la configuration :

```
[root@centos multipath]# multipath -l
mpathb (1scsi001) dm-2 Fenestr0,MonDisqueISCSI
size=256M features='0' hwhandler='0' wp=rw
|-+- policy='round-robin 0' prio=0 status=active
|  `- 4:0:0:1 sdb 8:16 active undef running
`-+- policy='round-robin 0' prio=0 status=enabled
   `- 4:0:0:2 sdc 8:32 active undef running
```

<note important> Notez que sdb est en statut **active**. C'est donc cette route qui sera utilisée pour écrire les données sur le périphérique. </note>

Redémarrez le service multipathd :

```
[root@centos ~]# service multipathd restart
ok
Arrêt du démon multipathd :          [ OK ]
Démarrage du démon multipathd :      [ OK ]
```

Provoquez maintenant une panne volontaire :

```
[root@centos ~]# multipathd fail path sdb
ok
```

Constatez le résultat :

```
[root@centos ~]# multipath -l
mpathb (1scsi001) dm-2 Fenestr0,MonDisqueISCSI
size=256M features='0' hwhandler='0' wp=rw
|-+- policy='round-robin 0' prio=0 status=active
|  `- 5:0:0:2 sdc 8:32 active undef running
`-+- policy='round-robin 0' prio=0 status=enabled
   `- 5:0:0:1 sdb 8:16 failed undef running
```

<note important> Notez la présence du mot **failed** sur la ligne sdb. </note>

Réactivez le chemin avec la commande suivante :

```
[root@centos ~]# multipathd reinstate path sdb
ok
```

Re-démarrez le service multipathd et constatez le résultat :

```
[root@centos ~]# multipath -l
mpathb (1scsi001) dm-2 Fenestr0,MonDisqueISCSI
size=256M features='0' hwhandler='0' wp=rw
|-+- policy='round-robin 0' prio=0 status=active
|  `- 5:0:0:1 sdb 8:16 active undef running
`-+- policy='round-robin 0' prio=0 status=enabled
   `- 5:0:0:2 sdc 8:32 active undef running
```

La Commande multipathd

La commande multipathd est interactive. Pour visualiser les commandes supportées par multipathd, utilisez la commande suivante :

```
[root@centos ~]# multipathd help
multipath-tools v0.4.9 (04/04, 2009)
CLI commands reference:
list|show paths
list|show paths format $format
list|show status
list|show daemon
list|show maps|multipaths
list|show maps|multipaths status
list|show maps|multipaths stats
list|show maps|multipaths format $format
list|show maps|multipaths topology
list|show topology
```

```
list|show map|multipath $map topology
list|show config
list|show blacklist
list|show devices
list|show wilddcards
add path $path
remove|del path $path
add map|multipath $map
remove|del map|multipath $map
switch|switchgroup map|multipath $map group $group
reconfigure
suspend map|multipath $map
resume map|multipath $map
resize map|multipath $map
disablequeueing map|multipath $map
restorequeueing map|multipath $map
disablequeueing maps|multipaths
restorequeueing maps|multipaths
reinstate path $path
fail path $path
paths count
quit|exit
```

Pour examiner la configuration par défaut de multipathd, utilisez la commande suivante :

```
[root@centos ~]# multipathd show config | more
defaults {
    verbosity 2
    polling_interval 5
    udev_dir "/dev"
    multipath_dir "/lib/multipath"
    path_selector "round-robin 0"
    path_grouping_policy failover
    getuid_callout "/lib/udev/scsi_id --whitelisted --device=/dev/%n"
```

```
prio const
features "0"
path_checker directio
failback manual
rr_min_io 1000
rr_min_io_rq 1
rr_weight uniform
queue_without_daemon yes
pg_timeout none
flush_on_last_del no
user_friendly_names yes
find_multipaths no
log_checker_err always
}
blacklist {
--Plus--
```