

# Introduction aux réseaux de stockage iSCSI

Philippe Latu  
philippe.latu(at)inetdoc.net

<https://www.inetdoc.net>

## Résumé

Ce support de travaux pratiques est consacré à l'étude des technologies de stockage DAS (*Direct Attached Storage*), SAN (*Storage Area Network*) et de la redondance RAID 1. Le protocole iSCSI est utilisé comme exemple d'accès «en mode bloc» aux unités de stockage réseau pour la partie SAN. La redondance RAID 1 utilise les fonctions intégrées au noyau Linux. L'infrastructure proposée montre comment les différentes technologies élémentaires peuvent être combinées pour atteindre les objectifs de haute disponibilité et de sauvegarde.

```
ls -l ..... [..]
o- bindata ..... [..]
o- fileio ..... [..]
o- initiator2 ..... [..]
o- itlock ..... [..]
o- initiator1 ..... [..]
o- fd_mcp ..... [..]
o- hb_spt ..... [..]
o- iscsi ..... [..]
o- tm_2003-01.org.linux-iscsi.vml.4680sm.1b40d1997f22 ..... [..]
o- tm3 ..... [..]
o- acl5 ..... [..]
o- tm_1993-08.org.dsf4m81926d6d355 ..... [..]
o- mapped_lun0 ..... [..]
o- lun0 ..... [..]
o- portals ..... [..]
o- 192.0.2.1213260 ..... [..]
o- loopback ..... [..]
o- qla2xxx ..... [..]
o- tm_fe ..... [..]
o- vhost ..... [..]
```

## Table des matières

1. Copyright et Licence .....	1
2. Objectifs .....	2
3. Topologie, scénario et plan d'adressage .....	3
4. Technologie iSCSI .....	4
5. Préparer une unité de stockage .....	5
5.1. Afficher la liste des unités de stockage .....	5
5.2. Créer une table des partitions et formater une unité de disque .....	6
5.3. Monter manuellement un volume de stockage .....	8
5.4. Détruire la table des partitions .....	9
6. Configuration du rôle Target .....	10
6.1. Installation de l'outil de paramétrage du rôle target .....	10
6.2. Configuration du rôle target .....	10
7. Configurer le système Initiator .....	14
7.1. Sélectionner le paquet et lancer le service .....	14
7.2. Accéder aux volumes de stockage réseau iSCSI .....	14
7.3. Réinitialiser la session iSCSI .....	16
7.4. Rendre la session iSCSI permanente .....	17
8. Configuration de l'authentification CHAP .....	19
9. Configuration d'une unité logique RAID 1 .....	22
9.1. Sélection du paquet et création de l'unité RAID 1 .....	22
9.2. Manipulations sur l'unité de disque RAID 1 .....	22
10. Configuration d'un volume logique et de sa sauvegarde .....	23
11. Perte d'une unité de disque du tableau RAID 1 .....	27
12. Évaluation des performances .....	27
13. Conclusion .....	29
14. Documents de référence .....	29

## 1. Copyright et Licence

Copyright (c) 2000,2025 Philippe Latu.  
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Copyright (c) 2000,2025 Philippe Latu.  
Permission est accordée de copier, distribuer et/ou modifier ce document selon les termes de la Licence de Documentation Libre GNU (GNU Free Documentation License), version 1.3 ou toute version ultérieure publiée par la Free Software Foundation ; sans Sections Invariables ; sans Texte de Première de Couverture, et sans Texte de Quatrième de Couverture. Une copie de la présente Licence est incluse dans la section intitulée « Licence de Documentation Libre GNU ».

## Méta-information

Ce document est écrit avec *DocBook* XML sur un système *Debian GNU/Linux*. Il est disponible en version imprimable au format PDF : [sysadm-net.iscsi.pdf](#).

## 2. Objectifs

---

Après avoir réalisé les manipulations proposées par ce support, vous serez en mesure de :

### Découvrir et comprendre les technologies de stockage réseau (DAS, SAN, iSCSI, RAID 1)

Identifier les différences entre les architectures de stockage direct (DAS), les réseaux de stockage (SAN) et la redondance RAID 1, en mettant l'accent sur le protocole iSCSI comme exemple d'accès en mode bloc aux unités de stockage réseau.

### Configurer un initiateur iSCSI sur un système Linux.

Configurer une infrastructure minimale iSCSI, incluant la préparation des unités de stockage, la configuration des rôles *Initiator* et *Target*, et la validation de la connectivité réseau et du partage de volumes de stockage.

### Expérimenter la redondance et la haute disponibilité avec RAID 1

Illustrer la création et la gestion d'un volume RAID 1 combinant un disque local et un volume iSCSI, afin de démontrer la réplication synchrone des données et la tolérance aux pannes dans un environnement réel.

### Automatiser la gestion et la sauvegarde des volumes logiques

Initier les étudiants à la gestion avancée des volumes logiques (LVM), à la création de *snapshots* pour la sauvegarde, et à la restauration de données, tout en évaluant les performances des différentes solutions de stockage mises en œuvre.

### 3. Topologie, scénario et plan d'adressage

#### Topologie logique

Les manipulations présentées dans ce support utilisent un domaine de diffusion unique (VLAN) dans lequel on trouve deux systèmes virtuels ou physiques avec deux unités de stockage distinctes chacune.

- La première unité de disque `/dev/vda` représente le stockage du système d'exploitation de la machine virtuelle.
- La deuxième unité de stockage `/dev/vdb` est dédiée aux manipulations présentées dans ce document.



Topologie logique - vue complète

#### Scénario

Le séquençement des opérations dépend des rôles définis par la technologie iSCSI.

Tableau 1. Attribution des rôles iSCSI

Rôle <i>Initiator</i>	Rôle <i>Target</i>
Préparation d'une unité de stockage locale en vue de la redondance avec l'unité de stockage réseau proposée par le rôle <i>Target</i>	Préparation d'une unité de stockage locale qui sera mise à disposition sur le réseau à l'aide de la technologie iSCSI
Recherche et installation du ou des paquet(s) pour le rôle <i>Initiator</i>	Recherche et installation du ou des paquet(s) pour le rôle <i>Target</i>
Étude des outils de configuration du service open-iscsi	Étude des outils de configuration du service targetcli
Validation manuelle de la configuration SAN iSCSI	
Validation de la configuration système	
Validation de l'authentification mutuelle entre les rôles <i>Initiator</i> et <i>Target</i>	
Mise en place de la réplication synchrone avec un tableau RAID 1 entre unité de disque locale et le volume iSCSI	Mise en place de la réplication asynchrone avec un volume logique de type <i>snapshot</i> de sauvegarde des fichiers images de volume de stockage
Étude comparative des performances d'accès	

#### Plan d'adressage

Partant de la topologie présentée ci-dessus, on utilise un plan d'adressage pour chacun des rôles iSCSI.

Le tableau ci-dessous correspond au plan d'adressage de la maquette qui a servi à traiter les questions des sections suivantes. Lors des séances de travaux pratiques, un plan d'adressage spécifique est fourni à chaque binôme d'étudiants. Il faut se référer au document *Infrastructure*.

Tableau 2. Plan d'adressage de la maquette « Introduction aux réseaux de stockage iSCSI »

Rôle	VLAN	Adresses IP
<i>Initiator</i>	369	10.0.113.6/28 2001:678:3fc:171:baad:caff:fefe:6/64
<i>Target</i>	369	10.0.113.5/28 2001:678:3fc:171:baad:caff:fefe:5/64

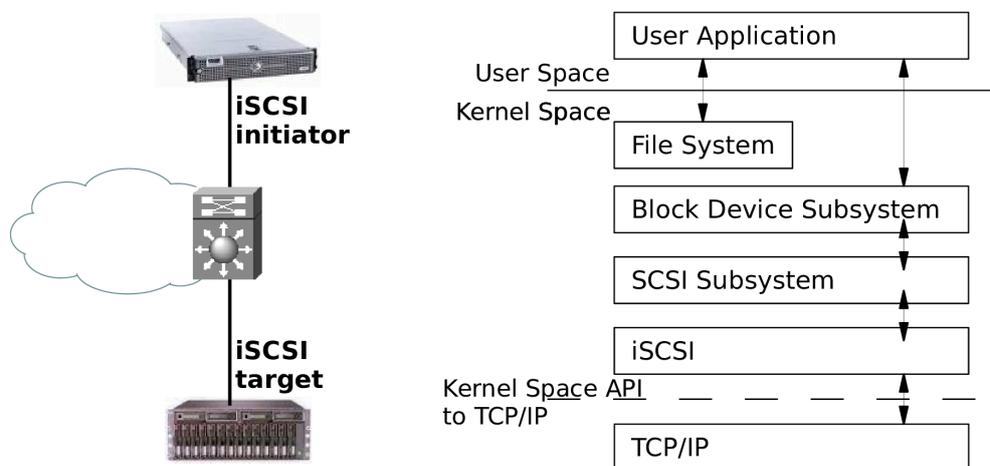
Pour traiter le scénario de ce support qui associe la technologie iSCSI, la redondance de disque RAID 1 et la gestion de volume logique LVM, on utilise deux instances de machines virtuelle avec une unité de disque supplémentaire.

Avant de traiter les questions des sections suivantes, il faut rechercher dans le cours *Infrastructure* les éléments nécessaires au raccordement des machines virtuelles ou physiques. Les étapes usuelles sont les suivantes :

1. Attribuer les adresses IPv4 et IPv6 à chacun des postes en fonction de l'espace d'adressage du réseau défini.
2. Rechercher le numéro de VLAN correspondant aux réseaux IP attribués.
3. Repérer le commutateur sur lequel des ports ont été affectés au VLAN recherché. Connecter les deux postes de travaux pratiques sur les ports identifiés.
4. Configurer les interfaces réseau de chaque poste : adresse, masque et passerelle par défaut. Valider la connectivité IP entre les deux postes puis avec les autres réseaux de l'infrastructure de travaux pratiques.

## 4. Technologie iSCSI

Cette section présente sommairement le protocole et les rôles iSCSI. Ce support fait suite à la présentation sur le *Stockage Réseau* utilisée en cours.



### Topologie iSCSI basique - vue complète

#### Qu'est-ce que l'iSCSI ?

Il s'agit d'un protocole réseau standardisé qui permet de transporter les commandes SCSI sur un réseau TCP/IP. Grâce à cette technologie, il est possible d'accéder à un espace de stockage à distance comme s'il s'agissait d'un disque dur local, en utilisant une infrastructure réseau Ethernet.

#### Fonctionnement et architecture

Le protocole iSCSI encapsule les commandes SCSI dans des paquets IP. Pour garantir l'intégrité des transmissions, le protocole gère l'ordre d'envoi, la retransmission et la séquence des commandes, car le

réseau IP n'assure pas de livraison fiable. Les données sont traitées « en mode bloc », comme dans les architectures SAN classiques, ce qui permet d'utiliser la technologie de manière comparable à la technologie Fibre Channel, mais sur un réseau IP.

### Rôles des équipements

Le rôle « Target » est un système (matériel ou logiciel) qui héberge le volume de stockage à publier sur le réseau. Il peut s'agir d'un serveur physique ou virtuel, d'une baie de disques ou d'un simple disque/fichier exporté.

Le rôle « Initiator » correspond au client qui accède au volume de stockage iSCSI publié sur le réseau.

### Avantages de l'iSCSI

Coût réduit : l'iSCSI fonctionne sur des réseaux classiques, sans matériel propriétaire, contrairement à Fibre Channel, ce qui permet de réduire les coûts matériels et d'administration.

Simplicité et flexibilité : la gestion s'effectue à l'aide d'outils standards sur des réseaux homogènes, ce qui facilite la formation et l'exploitation.

Interopérabilité : pris en charge nativement par la plupart des systèmes d'exploitation (Linux, Windows, etc.) et par de nombreux équipements réseau modernes.

### Bonnes pratiques et optimisation des performances

L'agrégation de liens (bonding, LACP, Etherchannel) permet d'augmenter la bande passante ou d'assurer la tolérance de panne.

Multipath : pour assurer la redondance et la répartition de charge, plusieurs chemins réseau peuvent être utilisés simultanément.

Architectures homogènes : privilégier un réseau Ethernet unifié simplifie l'exploitation et réduit les risques d'erreur.

### Mise en œuvre sous Linux

Plusieurs implémentations coexistent dans les systèmes Linux. Pour ces manipulations, nous avons choisi d'utiliser des outils qui exploitent les fonctions du noyau Linux.

Côté initiateur, la référence est open-iscsi. Ce logiciel, dont le cœur est intégré au noyau Linux, gère la connexion aux cibles iSCSI.

Côté cible, les outils comme targetcli-fb permettent de configurer le partage de volumes via la pile LIO du noyau Linux.

## 5. Préparer une unité de stockage

Cette section traite des manipulations à effectuer pour préparer une unité de stockage à son utilisation dans une configuration DAS (et/ou) SAN.

Comme indiqué dans le [Tableau 1, « Attribution des rôles iSCSI »](#), les manipulations de cette section sont utiles pour le rôle *Initiator* une fois que la session iSCSI est établie et que la nouvelle unité de disque appelée `/dev/sda` est accessible.



#### Avertissement

Les copies d'écran utilisées dans les réponses correspondent à l'utilisation d'une machine virtuelle à laquelle on a ajouté une unité de disque NVME nommée `/dev/nvme1n1`.

Suivant le mode de raccordement choisi (nvme, scsi, virtio ou iscsi), les noms des unités de disques changent.

### 5.1. Afficher la liste des unités de stockage

Pour commencer, il est utile de connaître la liste des unités de stockage en mode bloc sur un système.

Q1. Quelle est la commande apparentée à ls qui permet d'obtenir la liste des périphériques de stockage en mode bloc ?

Consulter la liste des outils fournis avec le paquet util-linux.

```
dpkg -L util-linux | grep bin/ls
```

```
/usr/bin/lsblk
/usr/bin/lscpu
/usr/bin/lscpc
/usr/bin/lsocks
/usr/bin/lsglins
/usr/bin/lsmem
/usr/bin/lsns
```

Une fois que la commande `lsblk` est identifiée, vous devez l'utiliser pour obtenir la liste voulue.

```
lsblk
```

```
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
sr0          11:0    1  1024M  0 rom
vda          254:0    0   366K  0 disk
nvme0n1      259:1    0  120G  0 disk
├─nvme0n1p1  259:4    0     3M  0 part
├─nvme0n1p2  259:5    0   124M  0 part /boot/efi
└─nvme0n1p3  259:6    0  119,9G  0 part /
nvme1n1    259:3    0   32G  0 disk
```

Dans le système de fichiers, c'est l'unité `/dev/nvme1n1` qui doit être utilisée pour les manipulations de cette section.

## 5.2. Créer une table des partitions et formater une unité de disque

Dans quel contexte doit-on partitionner et formater une unité de disque ?

- On peut s'entraîner avec les seconds disques des rôles *Initiator* et *Target*. Une fois les tests effectués, il est préférable d'effacer la table de partition avec en suivant les instructions de la [Section 5.4, « Détruire la table des partitions »](#).
- Il est également possible de s'entraîner avec l'unité de stockage iSCSI sur le système *Initiator* une fois la session iSCSI établie. Comme l'unité de disque réseau est la propriété exclusive du rôle *Initiator* en mode bloc, on effectue les mêmes manipulations que pour un disque local au système. Comme dans le cas précédent, il est préférable d'effacer la table de partition avec en suivant les instructions de la [Section 5.4, « Détruire la table des partitions »](#).
- Enfin, il faut partitionner et formater le volume logique LVM pour pouvoir suivre les manipulations à partir de la [Section 10, « Configuration d'un volume logique et de sa sauvegarde »](#).

Q2. Comment créer une partition unique utilisant le maximum de l'espace de stockage de l'unité de disque ?

Consulter la documentation de `parted` à l'adresse [Using Parted](#).

Installez le paquet `parted` si nécessaire.

```
sudo apt install parted
```

```
sudo parted /dev/nvme1n1
```

```
GNU Parted 3.6
Using /dev/nvme1n1
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted)
```

```
(parted) mklabel gpt
(parted) print
```

```
Model: QEMU NVMe Ctrl (nvme)
Disk /dev/nvme1n1: 34,4GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:
```

```
Number  Start  End  Size  File system  Name  Flags
```

```
(parted) mkpart myOwnPartition ext4 1MiB 100%
(parted) print
```

```
Model: QEMU NVMe Ctrl (nvme)
Disk /dev/nvme1n1: 34,4GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:
```

```
Number  Start  End  Size  File system  Name  Flags
1       1049kB 34,4GB 34,4GB ext4        myOwnPartition
```

```
(parted) quit
```

```
Information: You may need to update /etc/fstab.
```

Q3. Quelle est la commande à utiliser pour les opérations de formatage ? Quel est le rôle de l'option `-T` de cette commande ?

Les informations utiles sont disponibles à la page [Ext4 Howto](#). Les pages de manuels détaillent les fonctions des options.

La commande utilisée pour le formatage d'un système de fichiers `ext4`.

```
dpkg -S $(sudo which mkfs.ext4)
```

```
e2fsprogs: /usr/sbin/mkfs.ext4
```

L'option `-T` permet de définir le type d'utilisation du système de fichiers à formater en fonction sa taille. Les paramètres par défaut sont définis dans le fichier `/etc/mke2fs.conf`. Voici un résumé des paramètres intéressants pour ces manipulations :

- `ext4` : active des fonctionnalités avancées telles que la journalisation, l'allocation basée sur l'étendue, la prise en charge de fichiers très volumineux, les groupes de blocs flexibles, les sommes de contrôle des métadonnées, le 64 bits, la prise en charge des répertoires volumineux et des champs inode supplémentaires.
- `small/floppy/news` : utilise des tailles de bloc plus petites (1024 octets) et des densités d'inodes plus élevées (plus d'inodes par espace) pour les petits systèmes de fichiers ou des cas d'utilisation spécifiques (par exemple, les spools de nouvelles).
- `big/huge` : conçu pour les grands systèmes de fichiers avec un ratio d'inodes beaucoup plus élevé (moins d'inodes par espace). Il est utile pour le stockage de vidéos et d'images par exemple.
- `largefile/largefile4` : ratios d'inodes très élevés et tailles de blocs indéfinies pour les scénarios à usage unique et les fichiers volumineux. Il est utile pour le stockage d'images de machines virtuelles, de sauvegardes ou de flux vidéo en direct.

Q4. Quelle est la syntaxe de la commande de formatage de la partition créée lors de l'étape précédente ?

Des exemples de syntaxe sont disponibles à la page [Ext4 Howto](#).

```
sudo mkfs.ext4 /dev/nvme1n1p1
```

```
mke2fs 1.47.2 (1-Jan-2025)
Discarding device blocks: done
Creating filesystem with 8388096 4k blocks and 2097152 inodes
Filesystem UUID: 97049e50-931a-4741-8c87-faf2582c494f
Superblock backups stored on blocks:
 32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
4096000, 7962624
```

```
Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

Q5. Quelle est la syntaxe de la commande de visualisation des attributs du système de fichiers créé lors du formatage ?

Les informations utiles sur les attributs sont fournies à la page [Ext4 Howto](#).

```
sudo tune2fs -l /dev/nvme1n1p1
```

```
tune2fs 1.47.2 (1-Jan-2025)
Filesystem volume name: <none>
Last mounted on: <not available>
Filesystem UUID: 97049e50-931a-4741-8c87-faf2582c494f
Filesystem magic number: 0xEF53
Filesystem revision #: 1 (dynamic)
Filesystem features: has_journal ext_attr resize_inode dir_index orphan_file filetype extent 64bit flex_bg metadata_c
Filesystem flags: signed_directory_hash
Default mount options: user_xattr acl
Filesystem state: clean
Errors behavior: Continue
Filesystem OS type: Linux
Inode count: 2097152
Block count: 8388096
Reserved block count: 419404
Overhead clusters: 176700
Free blocks: 8210878
Free inodes: 2097140
First block: 0
Block size: 4096
Fragment size: 4096
Group descriptor size: 64
Reserved GDT blocks: 1024
Blocks per group: 32768
Fragments per group: 32768
Inodes per group: 8192
Inode blocks per group: 512
Flex block group size: 16
Filesystem created: Sat Aug 23 16:20:39 2025
Last mount time: n/a
Last write time: Sat Aug 23 16:20:39 2025
Mount count: 0
Maximum mount count: -1
Last checked: Sat Aug 23 16:20:39 2025
Check interval: 0 (<none>)
Lifetime writes: 6274 kB
Reserved blocks uid: 0 (user root)
Reserved blocks gid: 0 (group root)
First inode: 11
Inode size: 256
Required extra isize: 32
Desired extra isize: 32
Journal inode: 8
Default directory hash: half_md4
Directory Hash Seed: b7f68521-faa9-4357-bc98-f5baa8cf5f4e
Journal backup: inode blocks
Checksum type: crc32c
Checksum: 0x8a2806a9
Checksum seed: 0x8c83f6ed
Orphan file inode: 12
```

La sortie de la commande `tune2fs -l` indique que le système de fichiers `ext4` sur la partition `/dev/nvme1n1p1` est propre, utilise des blocs de 4 Ko et des inodes de 256 octets, et prend en charge des fonctionnalités telles que la journalisation et les extensions.

- Les sommes de contrôle des métadonnées sont représentées 64 bits.
- Les fichiers volumineux sont supportés.
- Le système de fichiers vient d'être créé avec presque tous les blocs et inodes libres.
- Il dispose d'options de montage par défaut (`user_xattr`, `acl`).

Aucun nom de volume n'est défini. Aucune vérification `fsck` périodique n'est prévue et le système de fichiers est de type Linux, avec un comportement d'erreur défini sur « continuer ».

### 5.3. Monter manuellement un volume de stockage

Une fois qu'un volume de stockage a été partitionné et formaté, vous pouvez le « *monter* » dans l'arborescence du système de fichiers du système de façon à pouvoir lire et écrire des données.

Q6. Comment obtenir l'identifiant du volume de stockage à ajouter au système de fichiers ?

Consulter la liste des utilitaires fournis avec le paquet `util-linux`. Il faut se rappeler que la représentation fichier d'un périphérique de stockage se distingue par son mode d'accès : le mode bloc.

La commande à utiliser est `blkid`. Dans l'exemple de la partition `/dev/vdb1`, on obtient le résultat suivant.

```
sudo blkid /dev/nvme1n1p1

/dev/nvme1n1p1: UUID="97049e50-931a-4741-8c87-faf2582c494f"
BLOCK_SIZE="4096" TYPE="ext4" PARTLABEL="myOwnPartition"
PARTUUID="57ae228a-bd8d-49de-9933-97b2c71af70e"
```

Q7. Dans quel fichier de configuration trouve-t-on la liste des périphériques montés lors de l'initialisation du système ?

Consulter la liste des fichiers du paquet util-linux.

Le fichier recherché est `/etc/fstab`. Il contient la liste des points de montage. Dans l'exemple ci-dessous, la racine et la partition d'échange utilisée en cas de saturation des ressources RAM du système.

```
grep -v '^#' /etc/fstab
```

```
PARTUUID=95ed2e41-f56c-4cc0-a79e-c93a07e93c79 / ext4 rw,discard,errors=remount-ro,x-systemd.growfs 0 1
PARTUUID=71f5d30c-8e21-4372-81f6-11eb66815669 /boot/efi vfat defaults,umask=077 0 2
```

Q8. Quelle est la commande qui donne la liste des montages en cours d'utilisation sur le système ? Quelle est l'option qui permet de scruter les entrées du fichier recherché dans la question précédente et de monter tous les points non encore utilisés ?

La commande est fournie par le paquet du même nom.

Le paquet `mount` fournit la commande du même nom. Cette commande liste tous les montages actifs du système. La liste comprend les systèmes de fichiers virtuels qui représentent l'état courant des paramètres du noyau ainsi que les systèmes de fichiers physiques qui correspondent aux unités de stockage utilisées.

```
mount | grep nvme
```

```
/dev/nvme0n1p3 on / type ext4 (rw,relatime,discard,errors=remount-ro)
/dev/nvme0n1p2 on /boot/efi type vfat (rw,relatime,fmask=0077,dmask=0077,codepage=437,iocharset=ascii,shortname=mixed,utf8)
```

L'option de montage des entrées inutilisées du fichier `/etc/fstab` est `-a`. Elle doit être utilisée dans la question suivante.

Q9. Comment monter manuellement le système de fichiers de la partition `/dev/nvme1p1` ?

Le répertoire de test pour les montages temporaires est historiquement `/mnt/`.

Consulter les pages de manuels de la commande `mount`.

On dispose d'au moins deux solutions pour désigner la partition à monter.

- Utiliser l'identifiant de partition unique.

```
sudo mount -U "97049e50-931a-4741-8c87-faf2582c494f" /mnt
```

```
mount | grep mnt
```

```
/dev/nvme1n1p1 on /mnt type ext4 (rw,relatime)
```

```
sudo umount /mnt
```

- Utiliser le nom défini par `udev` dans le système de fichiers.

```
sudo mount /dev/nvme1n1p1 /mnt
```

```
mount | grep mnt
```

```
/dev/nvme1n1p1 on /mnt type ext4 (rw,relatime)
```

```
sudo umount /mnt
```

Q10. Comment démonter manuellement le système de fichiers de la partition `/dev/nvme1p1` ?

Consulter les pages de manuels de la commande `mount`.

L'opération de démontage utilise l'arborescence du système de fichiers pour désigner le volume de stockage.

```
sudo umount /mnt
```

Cette commande a déjà été appelée dans la question précédente pour éviter les conflits entre les deux syntaxes de montage différentes.

## 5.4. Détruire la table des partitions

Dans cette section, on veut effacer toutes les traces des manipulations précédentes pour traiter les questions des autres parties du support de travaux pratiques.

**Avertissement**

Attention au nom d'unité de disque utilisé pour les manipulations qui suivent. La suppression de la table de patition de l'unité de disque système est fatale !

- Q11. Quelle est la syntaxe d'appel de l'outil parted qui permet de visualiser la table de partition d'une unité de disque ?

Consultez la documentation de parted à l'adresse [Using Parted](#).

```
sudo parted /dev/nvme1n1 print
```

```
sudo parted /dev/nvme1n1 print
```

```
Model: QEMU NVMe Ctrl (nvme)
Disk /dev/nvme1n1: 34,4GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name              Flags
  1      1049kB  34,4GB  34,4GB  ext4         myOwnPartition
```

Le résultat montre que l'unité de disque a été partitionnée dans une section précédente. Vous devez donc effacer les traces de ces opérations.

- Q12. Quelle est la syntaxe de la commande dd qui permet d'effacer complètement la table des partitions d'une unité de disque ?

Utiliser l'aide en ligne de la commande : `dd --help`.

La commande suivante écrit des 0 dans les 4 premiers blocs de 512 octets de l'unité de disque.

```
sudo dd if=/dev/zero of=/dev/nvme1n1 bs=4k count=64
```

```
64+0 enregistrements lus
64+0 enregistrements écrits
262144 octets (262 kB, 256 KiB) copiés, 0,00376759 s, 69,6 MB/s
```

```
sudo parted /dev/nvme1n1 print
```

```
Error: /dev/nvme1n1: unrecognised disk label
Model: QEMU NVMe Ctrl (nvme)
Disk /dev/nvme1n1: 34,4GB
Sector size (logical/physical): 512B/512B
Partition Table: unknown
Disk Flags:
```

Le résultat de la dernière commande d'affichage de la table de partition montre que toute trace de table de partition GPT a disparu.

## 6. Configuration du rôle Target

Dans cette partie, les manipulations préparent le système auquel on a attribué le rôle *Target* à l'aide de l'outil targetcli-fb.

### 6.1. Installation de l'outil de paramétrage du rôle target

- Q13. Quel est le paquet qui contient l'outil de configuration du service dans l'espace utilisateur ?

Recherchez le mot clé targetcli dans la liste des paquets.

Installez le paquet après avoir identifié son nom exact.

```
sudo apt search --names-only ^target
targetcli-fb/testing,now 1:2.1.53-1.3 all [installé]
  Command shell for managing the Linux LIO kernel target
```

```
sudo apt install targetcli-fb
```

### 6.2. Configuration du rôle target

La technologie iSCSI dispose d'un schéma de nommage propre défini dans le document standard [RFC3721 Internet Small Computer Systems Interface \(iSCSI\) Naming and Discovery](#). Le format retenu ici est baptisé *iqn* (*iSCSI Qualified Name*). Il s'agit d'une chaîne qui débute par `iqn.` suivie d'une date au format `AAAA-MM`, du nom de l'autorité qui a attribué le nom (le nom de domaine à l'envers), puis une autre chaîne unique qui identifie le nœud de stockage.

Dans un premier temps, on n'utilise aucun mécanisme d'authentification sachant que la configuration initiale se fait dans un contexte de travaux pratiques sur un réseau isolé.

Q14. Quelles sont les étapes à suivre pour publier un volume de stockage sur le réseau à partir de l'interface de l'outil targetcli ?

Identifiez les deux entrées intéressantes à partir du menu principal de l'outil de configuration targetcli à l'aide de la commande ls.

```
sudo targetcli
```

```
targetcli shell version 2.1.53
Copyright 2011-2013 by Datera, Inc and others.
For help on commands, type 'help'.
```

```
/> ls
```

```
o- / .....
  o- backstores .....
    | o- block ..... [Storage Ob
    | o- fileio ..... [Storage Ob
    | o- pscsi ..... [Storage Ob
    | o- ramdisk ..... [Storage Ob
  o- iscsi ..... [Ta
  o- loopback ..... [Ta
  o- vhost ..... [Ta
  o- xen-pvscsi ..... [Ta
```

Les deux entrées du menu principal de l'outil qui nous intéressent sont :

- La section **backstores** désigne les volumes de stockage à publier sur le réseau. Ici, les deux items intéressants sont **fileio** et **block**. Le premier permet de faire correspondre un fichier du système local au volume à publier. Le second fait correspondre une unité de disque physique au volume à publier.
- La section **iscsi** sert à définir une «cible» (**Target**) comprenant au moins une unité logique (LUN en vocabulaire SCSI). C'est ici que l'on configure le point de contact réseau pour le système **Initiator**.

## Partie stockage local : backstores

Q15. Quelles sont les opérations à effectuer définir un disque physique comme volume de stockage ?

Consultez le site de référence et repérer les options du menu block.

Créez un volume appelé **blockvol0** associé à l'unité de stockage locale au système /dev/nvme1n1.

```
/> cd /backstores/block
/backstores/block> create blockvol0 /dev/nvme1n1
Created block storage object blockvol0 using /dev/nvme1n1.
```

```
/backstores/block> ls
o- block ..... [Storage Objects: 1]
  o- blockvol0 ..... [/dev/nvme1n1 (32.0GiB) write-thru deactivated]
    o- alua ..... [ALUA Groups: 1]
      o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
```

Q16. Quelles sont les opérations à effectuer pour définir un fichier comme volume de stockage ?

Consultez le site de référence et repérer les options du menu fileio.

Créez un volume appelé **filevol0** associé au fichier /var/cache/filevol0.

```
/> cd /backstores/fileio
/backstores/fileio> create filevol0 /var/cache/filevol0 32G
Created fileio filevol0 with size 34359738368
```

```
/backstores/fileio> ls
```

```
o- fileio ..... [Storage Objects: 1]
  o- filevol0 ... [/var/cache/filevol0 (32.0GiB) write-back deactivated]
    o- alua ..... [ALUA Groups: 1]
      o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
```

## Partie portail iSCSI

Q17. Quelles sont les opérations à effectuer pour définir un nouveau portail réseau iSCSI ?

Consultez le site de référence et repérer les options du menu iscsi. Attention ! Une cible iSCSI comprend plusieurs attributs.

## 1. Nommage du portail au format iqn.

Si le nom du portail n'est pas fourni avec la commande create, il est généré automatiquement.

```
cd /iscsi
create
Created target iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8.
Created TPG 1.
Global pref auto_add_default_portal=true
Created default portal listening on all IPs (0.0.0.0), port 3260.

/iscsi> ls

o- iscsi ..... [Targets: 1]
o- iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8 ..... [TPGs: 1]
  o- tpg1 ..... [no-gen-acls, no-auth]
    o- acls ..... [ACLs: 0]
    o- luns ..... [LUNs: 0]
    o- portals ..... [Portals: 1]
      o- 0.0.0.0:3260 ..... [OK]
```

## 2. Association entre unité logique et portail iSCSI.

Les numéros d'unités logiques SCSI ou LUNs sont affectés automatiquement. Ici, l'unité `lun0` correspond à la première association faite depuis le dépôt des volumes de stockage.

```
cd /iscsi/iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8/tpg1/luns
>create /backstores/block/blockvol0
Created LUN 0.

ls /iscsi/

o- iscsi ..... [Targets: 1]
o- iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8 ..... [TPGs: 1]
  o- tpg1 ..... [no-gen-acls, no-auth]
    o- acls ..... [ACLs: 0]
    o- luns ..... [LUNs: 1]
      | o- lun0 ..... [block/blockvol0 (/dev/nvme1n1) (default_tg_pt_gp)]
    o- portals ..... [Portals: 1]
      o- 0.0.0.0:3260 ..... [OK]
```

## 3. Configuration réseau du portail iSCSI.

Un même portail peut être en écoute sur IPv4 et IPv6. Dans l'exemple ci-dessous on ouvre une configuration double pile en désignant la totalité des réseaux IPv6 après avoir effacé l'entrée créée automatiquement lors de la création du portail.

```
cd /iscsi/iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8/tpg1/portals/
delete 0.0.0.0 3260
Deleted network portal 0.0.0.0:3260

create ::0

Using default IP port 3260
Created network portal ::0:3260

ls
o- portals ..... [Portals: 1]
  o- [::0]:3260 ..... [OK]
```

On peut sortir de l'outil `targetcli` pour vérifier que le service réseau est bien accessible. La configuration est sauvegardée automatiquement.

```
exit

Global pref auto_save_on_exit=true
Last 10 configs saved in /etc/rtslib-fb-target/backup/.
Configuration saved to /etc/rtslib-fb-target/saveconfig.json
```

## Q18. Comment vérifier la disponibilité du portail réseau iSCSI ?

À l'aide des commandes `ss` ou `lsof`, relever le numéro de port de la couche transport relatif au protocole iSCSI.

Sur le système *Initiator*, lancer l'opération de découverte des volumes du portail iSCSI.

Voici un exemple d'exécution de la commande `ss` depuis le système *Target*.

```
ss -tan '( sport = :3260 )'
```

State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port	Process
LISTEN	0	256	*:*3260	*:*	

Maintenant que le service est disponible, utilisez la fonction de découverte depuis le système *Initiator*.

Vous devez utiliser l'adresse IP de votre système *Target*.

```
sudo iscsiadm -m discovery \
--type sendtargets \
--portal=[2001:678:3fc:171:baad:caff:fefe:5]
[2001:678:3fc:171:baad:caff:fefe:5]:3260,1 iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8
```

Q19. Est-il possible d'ouvrir une session iSCSI à ce stade de la configuration ?

Depuis votre système *Initiator*, lancez l'opération d'ouverture de session.

Même si le service réseau et la fonction découverte sont ouverts, le volume de stockage réseau n'est pas encore accessible. L'ouverture de session depuis l'hôte *Initiator* échoue et on obtient le message suivant.

La réponse à la question est donc non.

Vous devez utiliser l'identifiant IQN de votre système *Target*. Il est affiché dans les résultats de la commande découverte à la question précédente.

```
sudo iscsiadm -m node \
-T iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8 \
-p [2001:678:3fc:171:baad:caff:fefe:5] \
-l
```

```
iscsiadm: Could not login to
[iface: default, target: iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8, portal: 2001:678:3fc:171:baad:caff:fefe:5]
iscsiadm: initiator reported error (24 - iSCSI login failed due to authorization failure)
iscsiadm: Could not log into all portals
```

Côté hôte *Target*, les journaux système font apparaître un message du type suivant.

```
journalctl -n 100 --grep scsi
```

```
target kernel: iSCSI Login negotiation failed.
target kernel: iSCSI Initiator Node: iqn.1993-08.org.debian:01:733ea42cb56 is not authorized to access iSCSI target portal
```

Profitez de ces messages pour relever l'identifiant IQN de votre système *Initiator*. Dans l'exemple ci-dessus, on relève la valeur `iqn.1993-08.org.debian:01:733ea42cb56`.

Q20. Comment autoriser l'accès au volume de stockage depuis l'hôte *Initiator* sans authentification ?

Recherchez les paramètres relatifs à la rubrique `ac1s` de l'outil `targetcli`.

Pour autoriser l'ouverture d'une session iSCSI, créez une liste de contrôle d'accès (ACL) avec l'identité du système *Initiator*.

Depuis le système *Initiator*, affichez l'identité iSCSI définie lors de l'installation du paquet `open-iscsi`.

```
sudo grep -v ^# /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.1993-08.org.debian:01:733ea42cb56
```

Vérifiez que l'identifiant IQN du rôle *Initiator* correspond bien à la valeur relevée dans la question précédente.

Depuis le système *Target*, créez une nouvelle entrée dans la rubrique `ac1s` du portail iSCSI via l'outil `targetcli`.

```
sudo targetcli

cd /iscsi/iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8/tpg1/ac1s

create iqn.1993-08.org.debian:01:733ea42cb56
Created Node ACL for iqn.1993-08.org.debian:01:733ea42cb56
Created mapped LUN 0.
```

```
exit
```

Enfin, reprenez la commande d'ouverture de session sur le système *Initiator*. Cette nouvelle tentative d'ouverture de session doit être un succès.

```
sudo iscsiadm -m node
-T iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8 \
-p [2001:678:3fc:171:baad:caff:fefe:5] \
-l

Login to
[iface: default, target: iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8,
portal: 2001:678:3fc:171:baad:caff:fefe:5,3260] successful.
```

À partir de cette étape, le système *Initiator* dispose d'une nouvelle unité de stockage en mode bloc.

## 7. Configurer le système Initiator

Les manipulations de cette partie préparent le système auquel on a attribué le rôle *Initiator*. Ce système est celui qui utilise le volume de stockage mis à disposition sur le réseau par le rôle *Target*.

### 7.1. Sélectionner le paquet et lancer le service

Q21. Comment identifier et installer le paquet correspondant au rôle *Initiator* ?

En effectuant une recherche simple dans le catalogue des paquets disponibles, on obtient la liste des paquets dont le nom contient la chaîne de caractères `iscsi`.

La syntaxe la plus simple consiste à lancer une recherche sur les noms de paquets.

```
apt search --names-only iscsi
```

Pour optimiser l'affichage du résultat de recherche, on utilise une syntaxe plus complète.

```
apt -o "Apt::Cmd::Disable-Script-Warning=1" \
  search --names-only iscsi | grep -B2 -i initiator
```

```
open-iscsi/testing,now 2.1.11-2 amd64 [installé]
iscsi initiator tools
```

On relève que le paquet `open-iscsi` est le seul associé au rôle *Initiator*. Installez le.

```
sudo apt install open-iscsi
```

Q22. Comment connaître l'état du service *Initiator* et valider son fonctionnement ?

À partir de la liste des services actifs, repérez les messages relatifs au rôle *Initiator*.

```
systemctl status open-iscsi.service
```

```
systemctl status open-iscsi.service
# open-iscsi.service - Login to default iSCSI targets
   Loaded: loaded (/usr/lib/systemd/system/open-iscsi.service; enabled; preset: enabled)
   Active: inactive (dead)
 Condition: start condition unmet at Fri 2025-08-22 19:14:58 CEST; 22h ago
   Docs: man:iscsiadm(8)
         man:iscsid(8)
```

```
août 22 19:14:58 initiator systemd[1]: open-iscsi.service - Login to default iSCSI targets was skipped
because no trigger condition checks were met.
```



#### Avertissement

L'état actuel de la configuration montre que le service est lancé sans aucune session iSCSI active. Pour l'instant aucun système avec le rôle *Target* n'a été contacté.

### 7.2. Accéder aux volumes de stockage réseau iSCSI

Q23. Quelle est la commande principale du rôle *Initiator* qui permet de tester la connectivité iSCSI ?

Consultez la liste des fichiers du paquet `open-iscsi`.

En consultant la liste donnée ci-dessus, relevez qu'un seul outil est exécutable : la commande `iscsiadm`.

Q24. Quelles sont les options de découverte proposées avec cette commande ? Donnez un exemple fournissant l'identifiant de l'unité de stockage réseau visible.

Consultez les pages de manuels de la commande identifiée dans la question précédente.

À partir du système *Initiator*, listez le ou les volume(s) de stockage visible(s) sur le réseau local :

Si le portail du système avec le rôle *Target* est configuré pour être accessible via IPv6, vous devez utiliser la commande suivante en adaptant l'adresse au contexte :

```
sudo iscsiadm -m discovery \
  --type sendtargets \
  --portal=[2001:678:3fc:171:baad:caff:fefe:5]
[2001:678:3fc:171:baad:caff:fefe:5]:3260,1 iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeeef8
```

```
sudo iscsiadm -m discovery \
  --type sendtargets \
  --portal=10.0.113.5
10.0.113.5:3260,1 iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeeef8
```

Dans les deux copies d'écran ci-dessus, l'identifiant IQN du volume de stockage réseau visible est `iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeeef8`.

Malheureusement, les adresses de lien local IPv6 ne sont pas utilisables au moment de la rédaction de ces lignes.

Q25. Comment obtenir la liste des portails iSCSI déjà connus du système *Initiator* ?

Recherchez dans les pages de manuels de la commande `iscsiadm`.

C'est le mode `node` qui permet d'obtenir l'information demandée.

```
sudo iscsiadm -m node
```

```
10.0.113.5:3260,1 iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeeef8
[2001:678:3fc:171:baad:caff:fefe:5]:3260,1 iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeeef8
```

Q26. Comment effacer la liste des portails iSCSI déjà connus du système *Initiator* ?

Recherchez dans les pages de manuels de la commande `iscsiadm`.

C'est à nouveau le mode `node` qui permet d'obtenir l'information demandée.

```
sudo iscsiadm -m node --op=delete
```



#### Avertissement

Attention ! Si la commande ci-dessus est exécutée, il faut reprendre les opérations de découverte décrites à la question [Q : Q24](#) pour compléter la liste des portails iSCSI connus.

Q27. Quel est l'identifiant à communiquer ou à paramétrer pour que le système *Initiator* soit reconnu côté système *Target* ?

Rechercher les informations relatives au nommage iSCSI dans les outils et les fichiers fournis avec le paquet de gestion du rôle *Initiator*.

Le répertoire `/etc/iscsi/` contient les paramètres de configuration du service.

```
ls -p /etc/iscsi/
initiatorname.iscsi iscsid.conf
```

Consultez ou éditez ce fichier de façon à communiquer l'identité du système *Initiator* au système *Target* pour configurer le contrôle d'accès.

Par exemple, l'identifiant unique donnée dans la copie d'écran ci-dessous est à transmettre au système *Target*.

```
sudo grep -v ^# /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.1993-08.org.debian:01:733ea42cb5
```

Depuis le système *Target*, vous obtenez le résultat suivant après avoir créé la liste de contrôle d'accès au volume réseau via l'interface `targetcli`.

```
sudo targetcli
```

```
cd /
ls
```

```
o- / ..... [..]
o- backstores ..... [..]
| o- block ..... [Storage Objects: 1]
| | o- blockvol0 ..... [/dev/nvme1n1 (32.0GiB) write-thru activated]
| | | o- alua ..... [ALUA Groups: 1]
| | | o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
| o- fileio ..... [Storage Objects: 1]
| | o- filevol0 ..... [/var/cache/filevol0 (32.0GiB) write-back deactivated]
| | | o- alua ..... [ALUA Groups: 1]
| | | o- default_tg_pt_gp ..... [ALUA state: Active/optimized]
| o- pscsi ..... [Storage Objects: 0]
| o- ramdisk ..... [Storage Objects: 0]
o- iscsi ..... [Targets: 1]
| o- iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeeef8 ..... [TPGs: 1]
| | o- tpg1 ..... [no-gen-acls, no-auth]
| | | o- acls ..... [ACLs: 1]
| | | | o- iqn.1993-08.org.debian:01:733ea42cb5 ..... [Mapped LUNs: 1]
| | | | | o- mapped_lun0 ..... [lun0 block/blockvol0 (rw)]
| | | o- luns ..... [LUNs: 1]
| | | | o- lun0 ..... [block/blockvol0 (/dev/nvme1n1) (default_tg_pt_gp)]
| | o- portals ..... [Portals: 1]
| | | o- [:::]:3260 ..... [OK]
o- loopback ..... [Targets: 0]
o- vhost ..... [Targets: 0]
o- xen-pvscsi ..... [Targets: 0]
```

La copie d'écran ci-dessus montre l'association des identités iSCSI des systèmes *Initiator* et *Target*.

Q28. Quelles sont les options de connexion proposées avec la commande `iscsiadm` ?

Donnez un exemple illustrant l'établissement d'une connexion.

Consultez les pages de manuels de la commande identifiée précédemment.

```
sudo iscsiadm -m node
-T iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8 \
-p [2001:678:3fc:171:baad:caff:fefe:5] \
-l

Login to
[iface: default, target: iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8,
portal: 2001:678:3fc:171:baad:caff:fefe:5,3260] successful.
```

Dans l'exemple ci-dessus, la connexion sans authentification est un succès dans la mesure où les paramètres d'authentification et de protection en écriture ont été forcés à zéro sur la configuration du système *Target*. Voir la section intitulée « [Partie portail iSCSI](#) »

Q29. Comment obtenir les caractéristiques de l'unité de stockage iSCSI associée ?

Revoir la question [Quelle est la commande apparentée à ls qui permet d'obtenir la liste des périphériques de stockage en mode bloc ?](#) et/ou consulter les journaux système.

Le résultat de la commande `lsblk` montre l'arrivée d'un nouveau volume de stockage.

```
lsblk

NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
sda          8:0    0    32G  0 disk
sr0         11:0    1   1024M  0 rom
vda         254:0    0    366K  0 disk
nvme1n1     259:1    0     32G  0 disk
nvme0n1     259:3    0    120G  0 disk
├─nvme0n1p1 259:4    0     3M   0 part
├─nvme0n1p2 259:5    0    124M  0 part /boot/efi
└─nvme0n1p3 259:6    0  119,9G  0 part /
```

Voici un extrait des messages de journalisation du système.

```
journalctl -n 200 --grep '(sda|iscsi)'

initiator iscsid[6922]: Connection2:0 to [target:
iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8,
portal:2001:678:3fc:171:baad:caff:fefe:5,3260] through [iface: default]
is operational now
initiator kernel: sd 6:0:0:0: [sda] Attached SCSI disk
initiator kernel: sd 6:0:0:0: [sda] Optimal transfer size 131072 bytes
initiator kernel: sd 6:0:0:0: [sda] Preferred minimum I/O size 512 bytes
initiator kernel: sd 6:0:0:0: [sda] Write cache: enabled, read cache: enabled, supports DPO and FUA
initiator kernel: sd 6:0:0:0: [sda] Mode Sense: 43 00 10 08
initiator kernel: sd 6:0:0:0: [sda] Write Protect is off
initiator kernel: sd 6:0:0:0: [sda] 67108864 512-byte logical blocks: (34.4 GB/32.0 GiB)
initiator kernel: scsi host6: iSCSI Initiator over TCP/IP
```

Q30. Donnez la liste des entrées de périphériques de stockage créées par le démon `udev` ?

Listez les entrées de périphériques mode bloc de l'arborescence système.

Les fichiers de description des périphériques mode bloc sont tous situés dans le répertoire `/dev/`. En reprenant l'exemple ci-dessus, on obtient :

```
ls -lA /dev/[v,s]d*

brw-rw---- 1 root disk  8, 0 24 août 11:27 /dev/sda
brw-rw---- 1 root disk 254, 0 22 août 19:14 /dev/vda
```

L'entrée `/dev/sda` correspond à l'unité de disque iSCSI. L'unité de disque est donc bien vue de façon transparente comme un périphérique local du système accessible en mode bloc. Il entre bien dans la catégorie SAN ou *Storage Area Network*.

### 7.3. Réinitialiser la session iSCSI

Dans le cas d'une nouvelle configuration avec un autre hôte *Target* ou dans le cas d'un dépannage, il est utile de pouvoir reprendre les paramètres du rôle *Initiator*.

Q31. Comment obtenir la liste des sessions actives avec le système *Target* ?

Consultez les pages de manuels de la commande de configuration du rôle *Initiator* : `iscsiadm`.

C'est le mode `session`, documenté dans les pages de manuels de la commande `iscsiadm`, qui permet de répondre à la question.

```
sudo iscsiadm -m session
```

```
tcp: [2] [2001:678:3fc:171:baad:caff:fefe:5]:3260,1
iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8 (non-flash)
```

Q32. Comment libérer toutes les sessions actives depuis le système *Initiator* ?

Consultez les pages de manuels de la commande de configuration du rôle *Initiator* : `iscsiadm`.

Pour cette question, c'est le mode `node` qui nous intéresse.

```
sudo iscsiadm -m node -U all
```

```
Logging out of session [sid: 2,
target: iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8, portal: 2001:678:3fc:171:baad:caff:fefe:5,3260]
Logout of [sid: 2,
target: iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddeef8, portal: 2001:678:3fc:171:baad:caff:fefe:5,3260]
successful.
```

Il faut relancer une nouvelle session iSCSI pour traiter les manipulations suivantes.

## 7.4. Rendre la session iSCSI permanente

Une fois la connexion à la ressource iSCSI testée, passez à la configuration système de façon à retrouver le volume de stockage après un redémarrage du système *Initiator*.

Q33. Comment rendre l'ouverture session iSCSI automatique lors de l'initialisation du système *Initiator* ?

Recherchez dans la liste des fichiers du paquet `open-iscsi` les éléments relatifs à la configuration système. Éditez le fichier de configuration principal de façon à rendre automatique le lancement du service.

Au niveau système, les fichiers de configuration sont nécessairement dans le répertoire `/etc/`.

```
dpkg -L open-iscsi | grep '/etc/iscsi'
```

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

Le fichier `/etc/iscsi/iscsid.conf` contient une directive dans la section *Startup settings* qui rend automatique l'accès à une ressource déjà enregistrée. Voici le contenu de cette section extraite du fichier de configuration.

```
#####
# Startup settings
#####

# To request that the iscsi initd scripts startup a session set to "automatic".
node.startup = automatic
```



### Avertissement

Attention ! Après édition du fichier `/etc/iscsi/iscsid.conf`, la valeur `automatic` n'est appliquée que pour les nouvelles opérations de découverte et d'ouverture de session.

```
sudo sed -i 's/^node.startup = manual/node.startup = automatic/' /etc/iscsi/iscsid.conf
```

```
sudo systemctl restart open-iscsi.service
systemctl status open-iscsi.service
```

```
# open-iscsi.service - Login to default iSCSI targets
Loaded: loaded (/usr/lib/systemd/system/open-iscsi.service; enabled; preset: enabled)
Active: active (exited) since Sun 2025-08-24 14:31:42 CEST; 7s ago
Invocation: cc7d753b5ecc4235990e6fb658c2ef07
Docs: man:iscsiadm(8)
man:iscsid(8)
Process: 7471 ExecStart=/usr/sbin/iscsiadm -m node --loginall=automatic (code=exited, status=21)
Process: 7472 ExecStart=/usr/lib/open-iscsi/activate-storage.sh (code=exited, status=0/SUCCESS)
Main PID: 7472 (code=exited, status=0/SUCCESS)
Mem peak: 2M
CPU: 51ms

août 24 14:31:42 initiator systemd[1]: Starting open-iscsi.service - Login to default iSCSI targets...
août 24 14:31:42 initiator iscsiadm[7471]: iscsiadm: No records found
août 24 14:31:42 initiator systemd[1]: Finished open-iscsi.service - Login to default iSCSI targets.
```

La copie d'écran ci-dessus montre que l'ouverture de session est automatique au lancement du service `open-iscsi.service` et qu'en l'état actuel aucun enregistrement de session n'est présent dans la configuration.

Voici un nouvel exemple d'instructions de découverte puis d'ouverture de session.

```
sudo iscsiadm -m discovery \
  --type sendtargets \
  --portal=[2001:678:3fc:171:baad:caff:fe5e:5]

[2001:678:3fc:171:baad:caff:fe5e:5]:3260,1
iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8

sudo iscsiadm -m node \
  -T iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8 \
  -p [2001:678:3fc:171:baad:caff:fe5e:5] \
  -l

Login to [iface: default,
target: iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8,
portal: 2001:678:3fc:171:baad:caff:fe5e:5,3260] successful.
```

Q34. Comment connaître l'état et la liste d'une session iSCSI active ?

Consultez les pages de manuels de la commande de configuration du rôle *Initiator* : iscsiadm.

Le mode `session` est dédié aux manipulations sur les sessions. La commande de test la plus simple est la suivante.

```
sudo iscsiadm -m session

tcp: [1] [2001:678:3fc:171:baad:caff:fe5e:5]:3260,1
iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8 (non-flash)
```

Si la liste est vide, il n'y a pas de session iSCSI active en cours.

Il est possible d'obtenir davantage d'informations sur les paramètres de session en cours à l'aide de l'option `-P` suivie d'un numéro désignant le niveau de détail attendu.

La commande `iscsiadm -m session -P 3` affiche les paramètres sur les interfaces réseau utilisées, etc.

```
sudo iscsiadm -m session -P 3
```

Q35. Comment retrouver un point de montage unique de l'unité de stockage iSCSI après réinitialisation du système *Initiator* ?

Créez un répertoire de montage et recherchez les options utiles dans les pages de manuels des commandes `mount`, `systemd.mount` et `blkid`. Éditez le fichier `/etc/fstab` en utilisant les options sélectionnées. Notez que le fichier `fstab` possède ses propres pages de manuels.

La création du répertoire destiné au montage du volume de stockage iSCSI se fait simplement.

```
sudo mkdir /var/cache/iscsi-vol0
```

C'est à cette étape que les questions de la [Section 5, « Préparer une unité de stockage »](#) sont utiles.

Après partitionnement de l'unité de stockage iSCSI `/dev/sda` et formatage de la partition `/dev/sda1`, relevez l'identifiant unique de ce volume avec la commande `blkid`. Voici un exemple.

```
lsblk /dev/sda
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
sda   8:0    0 32G  0 disk

sudo parted /dev/sda mklabel gpt
Information: You may need to update /etc/fstab.

sudo parted /dev/sda mkpart myOwnPartition ext4 1MiB 100%
Information: You may need to update /etc/fstab.

sudo blkid /dev/sda1
/dev/sda1: UUID="97049e50-931a-4741-8c87-faf2582c494f"
BLOCK_SIZE="4096" TYPE="ext4" PARTLABEL="myOwnPartition"
PARTUUID="acc9f840-4b0b-4ffe-86e3-b8c499a90021"

sudo mount -U "97049e50-931a-4741-8c87-faf2582c494f" /var/cache/iscsi-vol0/

mount | grep iscsi
/dev/sda1 on /var/cache/iscsi-vol0 type ext4 (rw,relatime)

sudo touch /var/cache/iscsi-vol0/emptyFile.txt

sudo ls -l /var/cache/iscsi-vol0/
total 16
-rw-r--r-- 1 root root    0 24 août 17:12 emptyFile.txt
drwx----- 2 root root 16384 24 août 17:10 lost+found

sudo umount /var/cache/iscsi-vol0
```

Q36. Quelles sont les informations à insérer dans le fichier `/etc/fstab` pour assurer le montage du volume de stockage à chaque initialisation du système ?

Consultez les pages de manuels de la commande `mount` ainsi que la documentation du paquet `open-iscsi`.

Le choix des options à utiliser lors de l'édition du fichier `/etc/fstab` constitue un point très délicat.

```
echo "PARTUUID=\"acc9f840-4b0b-4ffe-86e3-b8c499a90021\"\  
/var/cache/iscsi-vol0 ext4 _netdev 0 2" |\n  
sudo tee -a /etc/fstab
```

- Le choix de la valeur `UUID` se fait à partir du résultat de la commande `blkid` donné ci-dessus.
- Le point de montage `/var/cache/iscsi-vol0` a lui aussi été défini ci-dessus.
- Le système de fichiers utilisé est, là encore, connu : `ext4`.
- L'option `_netdev` spécifie que le système de fichiers réside sur un périphérique nécessitant des accès réseau. Il est donc inutile d'y accéder tant qu'aucune interface réseau n'est active.

```
sudo mount -a  
mount: (hint) your fstab has been modified, but systemd still uses  
the old version; use 'systemctl daemon-reload' to reload.
```

```
mount | grep iscsi  
/dev/sda1 on /var/cache/iscsi-vol0 type ext4 (rw,relatime,stripe=32,_netdev)
```

```
sudo ls -l /var/cache/iscsi-vol0/  
total 16  
-rw-r--r-- 1 root root 0 24 août 17:12 emptyFile.txt  
drwx----- 2 root root 16384 24 août 17:10 lost+found
```

Une fois les tests de validation du montage effectués, on procède au démontage de la partition dans le but de traiter les questions des sections suivantes.

```
sudo umount /var/cache/iscsi-vol0/
```

```
sudo sed -i '/^PARTUUID="acc9f840-4b0b-4ffe-86e3-b8c499a90021".*/d' /etc/fstab
```

```
cat /etc/fstab
```

```
PARTUUID=95ed2e41-f56c-4cc0-a79e-c93a07e93c79 / ext4 rw,discard,errors=remount-ro,x-systemd.growfs 0 1  
PARTUUID=71f5d30c-8e21-4372-81f6-11eb66815669 /boot/efi vfat defaults,umask=077 0 2
```

La table des montages système est revenue à l'état initial.

## 8. Configuration de l'authentification CHAP

Dans cette partie, on suppose que tous les tests précédents ont été effectués avec succès et que les échanges entre les systèmes *Target* et *Initiator* sont validés.

On s'intéresse maintenant à l'authentification entre ces mêmes systèmes. Pour traiter les questions suivantes, une nouvelle entrée a été utilisée pour le rôle *Target*.

Le mécanisme d'authentification le plus communément utilisé dans le déploiement des connexions iSCSI s'appuie sur CHAP (*Challenge-Handshake Authentication Protocol*). Il s'agit d'une méthode d'authentification entre deux hôtes pairs sans échange de mot de passe en clair sur le réseau. Cette méthode suppose que les deux hôtes utilisent le même mot de passe.

Q37. Quel est l'attribut qui permet d'activer globalement l'authentification avec `targetcli` ?

Recherchez dans la liste des attributs du menu `tpg1` celui qui est relatif à l'authentification.

Placez vous au niveau `tpg1` du menu de l'application `targetcli`.

```
sudo targetcli
```

```
cd /iscsi/iqn.2003-01.org.linux-iscsi.target.x8664:sn.cd547a636fe7/tpg1/
```

Relevez la valeur de l'attribut `authentication`.

```
get attribute authentication  
authentication=0
```

Changez la valeur de cet attribut pour activer l'authentification au niveau du portail iSCSI.

```
set attribute authentication=1
```

```
get attribute authentication  
authentication=1
```

Q38. Comment régler les paramètres d'authentification CHAP à deux voies pour le portail iSCSI ?

Recherchez les paramètres disponibles au niveau de la liste de contrôle d'accès dédiée au rôle *Initiator*.

Partant de la configuration mise en place dans la [Section 6, « Configuration du rôle Target »](#), on ajoute les paramètres d'authentification à la liste de contrôle d'accès existante.

```
sudo targetcli
```

```
cd /iscsi/iqn.2003-01.org.linux-iscsi.target.x8664:sn.cd547a636fe7/tpg1/
cd acls/iqn.1993-08.org.debian:01:733ea42cb56
```

Définissez les paramètres d'authentification pour cette liste de contrôle d'accès. Comme la méthode CHAP est symétrique, vous devez déposer de part et d'autre le secret. On fixe ici les paramètres `userid` et `password`.

```
set auth userid=SAN-lab-user
Parameter userid is now 'SAN-lab-user'.
```

```
set auth password=SAN-lab-pwd
Parameter password is now 'SAN-lab-pwd'.
```

```
set auth mutual_userid=target-user
Parameter mutual_userid is now 'target-user'.
```

```
set auth mutual_password=target-pwd
Parameter mutual_password is now 'target-pwd'.
```

Les valeurs définies ci-dessus doivent être reportées dans la configuration du service open-iscsi sur le système *Initiator*.

Q39. Comment régler les paramètres d'authentification CHAP sur le système *Initiator* ?

Recherchez dans le fichier de configuration principal du rôle *Initiator* les paramètres relatifs à l'authentification.

Libérez la session existante avant d'ajouter les authentifiants à la configuration du service open-iscsi.

Effacez l'enregistrement obtenu avec l'instruction de découverte utilisée précédemment.

Commencez par libérer la session.

```
sudo iscsiadm -m node -U all
```

```
Logging out of session [sid: 1,
  target: iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8,
  portal: 2001:678:3fc:171:baad:caff:fefe:5,3260]
Logout of [sid: 1,
  target: iqn.2003-01.org.linux-iscsi.target.x8664:sn.5fc599ddee8,
  portal: 2001:678:3fc:171:baad:caff:fefe:5,3260] successful.
```

Effacez ensuite l'enregistrement de découverte du portail depuis le rôle *Initiator*.

```
sudo iscsiadm -m discoverydb \
  -t sendtargets \
  -p [2001:678:3fc:171:baad:caff:fefe:5]:3260 \
  -o delete
```

Le nom d'utilisateur et le mot de passe sont définis dans le fichier `/etc/iscsi/iscsid.conf` du système *Initiator*.

```
sudo grep -m1 -A30 CHAP /etc/iscsi/iscsid.conf
```

```
# CHAP Settings
# *****

# To enable CHAP authentication set node.session.auth.authmethod
# to CHAP. The default is None.
#
# NOTE: currently this attribute is checked for validity but then
# ignored, so even if it is set to None, CHAP will be used if the
# credentials (username/password and possibly username_in/password_in)
# are set. This behavior is deprecated, and in the future CHAP will
# not be used if authmethod is set to None.
#
#node.session.auth.authmethod = CHAP

# To configure which CHAP algorithms to enable, set
# node.session.auth.chap_algs to a comma separated list.
# The algorithms should be listed in order of decreasing
# preference - in particular, with the most preferred algorithm first.
# Valid values are MD5, SHA1, SHA256, and SHA3-256.
# The default is MD5.
#node.session.auth.chap_algs = SHA3-256,SHA256,SHA1,MD5

# To set a CHAP username and password for initiator
# authentication by the target(s), uncomment the following lines:
#node.session.auth.username = username
#node.session.auth.password = password

# To set a CHAP username and password for target(s)
# authentication by the initiator, uncomment the following lines:
#node.session.auth.username_in = username_in
#node.session.auth.password_in = password_in
```

Éditez les lignes du fichiers fichier `/etc/iscsi/iscsid.conf` avec les valeurs définies dans la configuration du rôle *Target*.

```
sudo sed \
-i 's/^#node.session.auth.authmethod = CHAP/node.session.auth.authmethod = CHAP/' \
/etc/iscsi/iscsid.conf
```

```
sudo sed \
-i 's/^#node.session.auth.username = username/node.session.auth.username = SAN-lab-user/' \
/etc/iscsi/iscsid.conf
```

```
sudo sed \
-i 's/^#node.session.auth.password = password/node.session.auth.password = SAN-lab-pwd/' \
/etc/iscsi/iscsid.conf
```

```
sudo sed \
-i 's/^#node.session.auth.username_in = .*/node.session.auth.username_in = target-user/' \
/etc/iscsi/iscsid.conf
```

```
sudo sed \
-i 's/^#node.session.auth.password_in = .*/node.session.auth.password_in = target-pwd/' \
/etc/iscsi/iscsid.conf
```

relancez le service `open-iscsi` pour prendre en compte les modifications effectuées sur la configuration du rôle *Initiator*.

```
sudo systemctl restart open-iscsi.service
```

Lancez une nouvelle découverte du portail iSCSI.

```
sudo iscsiadm -m discovery \
--type sendtargets \
--portal=[2001:678:3fc:171:baad:caff:fefe:5]
```

Lancez une nouvelle ouverture de session.

```
sudo iscsiadm -m node \
-T iqn.2003-01.org.linux-iscsi.target.x8664:sn.cd547a636fe7 \
-p [2001:678:3fc:171:baad:caff:fefe:5]:3260 \
--login
```

Relevez les informations de session.

```
sudo iscsiadm -m session
tcp: [1] [2001:678:3fc:171:baad:caff:fefe:5]:3260,1 \
iqn.2003-01.org.linux-iscsi.target.x8664:sn.cd547a636fe7 (non-flash)
```

Une fois la session établie, une authentification sécurisée est en place entre les rôles *Target* et *Initiator*.

Cependant, si le mécanisme d'authentification est valide, l'algorithme de chiffrement pose problème. En effet, le protocole MD5 ne garantit pas un niveau de sécurité suffisant.

Nous devons attendre une nouvelle version de la pile LIO du noyau Linux pour bénéficier de l'algorithme SHA3-256.

```
sudo iscsiadm -m node -o show | grep -i chap
```

```
iface.chap_auth = <empty>
iface.bidi_chap = <empty>
node.session.auth.authmethod = CHAP
node.session.auth.chap_algs = MD5
```

## 9. Configuration d'une unité logique RAID 1

Dans cette partie, vous allez créer une unité logique RAID 1 composée d'une unité de disque locale et d'une unité de disque iSCSI dans le but d'illustrer une solution de réplication synchrone. En effet, dans un volume RAID 1 chaque disque contient à tout moment exactement les mêmes données. Ici, le contenu de l'unité de disque locale est identique à celui de l'unité de disque réseau. La réplication ainsi réalisée est dite synchrone puisque toute écriture locale est dupliquée sur le réseau de stockage iSCSI.

### 9.1. Sélection du paquet et création de l'unité RAID 1

Q40. Quel est le paquet qui contient les outils de configuration et de gestion des différents types d'unités RAID logicielles ? Installer ce paquet et identifier l'outil d'administration de tableau RAID logiciel.

Effectuer une recherche dans les descriptions de paquets avec l'acronyme clé RAID.

```
$ aptitude search ~draid | grep administration
p mdadm - outil d'administration d'ensembles RAID

$ sudo apt install mdadm
```

Une fois le paquet identifié et installé, listez son contenu et isolez les commandes utilisateur.

```
$ dpkg -L mdadm | grep bin
/sbin
/sbin/mdmon
/sbin/mdadm-startall
/sbin/mdadm
```

Q41. Rechercher la syntaxe d'appel à l'outil identifié dans la question précédente pour créer l'unité logique RAID 1 ? Exécuter cette commande.

Après s'être assuré qu'aucune table de partition n'existe sur les deux unités constituant le tableau, on obtient le résultat suivant.

```
$ sudo mdadm --create /dev/md0 --level=raid1 --raid-devices=2 /dev/sda /dev/vdb
mdadm: Note: this array has metadata at the start and
may not be suitable as a boot device. If you plan to
store '/boot' on this device please ensure that
your boot-loader understands md/v1.x metadata, or use
--metadata=0.90
Continue creating array? y
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
```

### 9.2. Manipulations sur l'unité de disque RAID 1

Q42. Comment connaître l'état de l'unité logique RAID 1 ?

Effectuer une recherche dans le système de fichiers virtuel `/proc/`.

Exemple du tableau créé lors l'exécution de la commande de la question précédente.

```
$ cat /proc/mdstat
Personalities : [raid1]
md0 : active raid1 vdb[1] sda[0]
      33537920 blocks super 1.2 [2/2] [UU]

unused devices: <none>
```

Q43. Comment afficher la liste des propriétés de l'unité logique RAID 1 ?

Effectuer une recherche dans les options de la commande d'administration.

```

$ sudo mdadm --detail /dev/md0
/dev/md0:
    Version : 1.2
    Creation Time : Sat Sep  3 18:07:32 2022
    Raid Level : raid1
    Array Size : 33520640 (31.97 GiB 34.33 GB)
    Used Dev Size : 33520640 (31.97 GiB 34.33 GB)
    Raid Devices : 2
    Total Devices : 2
    Persistence : Superblock is persistent

    Update Time : Sat Sep  3 18:09:18 2022
    State : clean, resyncing
    Active Devices : 2
    Working Devices : 2
    Failed Devices : 0
    Spare Devices : 0

Consistency Policy : resync

    Resync Status : 65% complete

    Name : initiator:0 (local to host initiator)
    UUID : e3da1d56:9df89f79:866d5607:eeb2beff
    Events : 11

    Number Major Minor RaidDevice State
     0         8         0         0   active sync   /dev/sda
     1        254         16         1   active sync   /dev/vdb/

```

Q44. Comment rendre la configuration du tableau RAID 1 permanente au niveau système ?

Effectuer une recherche dans les options de la commande d'administration.

C'est le fichier `/etc/mdadm/mdadm.conf` qui contient les directives de configuration. On ajoute en fin de ce fichier la définition du tableau créé plus haut.

```
$ sudo mdadm --detail --scan | sudo tee -a /etc/mdadm/mdadm.conf
```

## 10. Configuration d'un volume logique et de sa sauvegarde

L'objectif de cette partie est de créer un mécanisme de sauvegarde réseau automatisé en s'appuyant sur la notion de «prise de vue» ou *snapshot* proposée par le gestionnaire de volume logique LVM. Dans un instantané, on ne stocke que les différences relativement au volume logique original.

Q45. Quel est le paquet associé à la gestion de volume logique LVM ?

Rechercher et installer le paquet qui permet de créer et gérer des volumes physiques, logiques ainsi que des groupes.

En anglais, on parle de *Logical Volume Manager* ou LVM. On cherche donc un paquet avec la chaîne 'lvm'.

```

$ aptitude search ^lvm
p   lvm2          - gestionnaire de volumes logiques de Linux
p   lvm2-dbusd    - démon D-Bus pour LVM2
p   lvm2-lockd   - démon de verrouillage pour LVM

```

```
$ sudo apt install lvm2
```

Q46. Comment créer un volume physique associé au tableau RAID 1 précédemment créé ?

Rechercher dans la liste des outils ceux correspondant à la gestion de volume physique.

L'instruction de recherche habituelle est de la forme :

```
$ dpkg -L lvm2 | grep bin
```

Ce sont les outils dont le nom commence par 'pv' qui servent à manipuler les volumes physiques.

```
$ sudo pvcreate --help
```

Création du volume physique.

```

$ sudo pvcreate /dev/md0
Physical volume "/dev/md0" successfully created.

```

Affichage résumé de l'état du volume physique.

```

$ sudo pvs
PV          VG Fmt Attr PSize  PFree
/dev/md0    lvm2 --- <31,97g <31,97g

```

Affichage détaillé de l'état du volume physique.

```
$ sudo pvdisplay
"/dev/md0" is a new physical volume of "<31,97 GiB"
--- NEW Physical volume ---
PV Name                /dev/md0
VG Name
PV Size                <31,97 GiB
Allocatable           NO
PE Size               0
Total PE              0
Free PE               0
Allocated PE          0
PV UUID               vUlk3p-dzZJ-MyLZ-hMcU-P9dH-oQuB-2lptum
```

Q47. Comment créer un groupe de volume contenant le tableau RAID 1 ?

Rechercher dans la liste des outils ceux correspondant à la gestion de groupes de volumes.

À partir du résultat de la commande de recherche de la question précédente, relevez le fait que ce sont les outils dont le nom commence par 'vg' qui servent à manipuler les groupes de volumes.

```
$ sudo vgcreate --help
```

Création du groupe de volume avec un unique volume physique.

```
$ sudo vgcreate lab-vg /dev/md0
Volume group "lab-vg" successfully created
```

Affichage résumé de l'état du volume physique.

```
$ sudo vgs
VG          #PV #LV #SN Attr   VSize VFree
lab-vg     1   0   0 wz--n- 31,96g 31,96g
```

Affichage détaillé de l'état du volume physique.

```
$ sudo vgdisplay
--- Volume group ---
VG Name                lab-vg
System ID
Format                 lvm2
Metadata Areas         1
Metadata Sequence No  1
VG Access              read/write
VG Status              resizable
MAX LV                 0
Cur LV                0
Open LV               0
Max PV                 0
Cur PV                1
Act PV                1
VG Size                31,96 GiB
PE Size                4,00 MiB
Total PE               8183
Alloc PE / Size        0 / 0
Free PE / Size         8183 / 31,96 GiB
VG UUID                KIq2zb-emxQ-JiT0-6wAk-tP10-Mm1N-wxzKL1
```

Q48. Comment créer un volume logique à l'intérieur du groupe contenant le tableau RAID 1 ?

Rechercher dans la liste des outils ceux correspondant à la gestion des volumes logiques.

Dans cet exemple, nous allons créer un volume logique de 16Go pour une capacité de 32Go. En situation réelle, il faudrait remplacer les gigaoctets par des téraoctets.

Toujours à partir du résultat de la commande de recherche des deux questions précédentes, relevez que ce sont les outils dont le nom commence par 'lv' qui servent à manipuler les volumes logiques.

```
$ sudo lvcreate --help
```

Création du volume logique de 16Go.

```
$ sudo lvcreate --size 16Go lab-vg
Logical volume "lv010" created.
```

Affichage résumé de l'état du volume logique.

```
$ sudo lvs
LV VG          Attr          LSize  Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
lv010 lab-vg -wi-a----- 16,00g
```

Affichage détaillé de l'état du volume logique.

```
$ sudo lvsdisplay
--- Logical volume ---
LV Path                /dev/lab-vg/lvol0
LV Name                lvol0
VG Name                lab-vg
LV UUID                8UFwye-RMgA-hzY4-8nnv-h7nK-09GA-Ff2AT2
LV Write Access        read/write
LV Creation host, time initiator, 2022-09-05 14:27:28 +0200
LV Status               available
# open                  0
LV Size                16,00 GiB
Current LE              4096
Segments                1
Allocation              inherit
Read ahead sectors     auto
 - currently set to    256
Block device            252:0
```

Q49. Comment créer un système de fichiers sur le nouveau volume logique ?

Reprendre les traitements de la [Section 5, « Préparer une unité de stockage »](#) avec le nom du volume logique obtenu à la question précédente.

Formatage du système de fichiers.

```
$ sudo mkfs.ext4 /dev/lab-vg/lvol0
mke2fs 1.46.5 (30-Dec-2021)
Discarding device blocks: done
Creating filesystem with 4194304 4k blocks and 1048576 inodes
Filesystem UUID: d94a59f2-7c67-4be7-9643-d646d1dbe2a4
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
    4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

Q50. Comment monter et accéder au nouveau système de fichiers ?

Créer un sous dossier au niveau /mnt et monter le nouveau système de fichiers manuellement.

Exemple de résultats attendus.

```
$ sudo mkdir /mnt/lvol0

$ sudo mount /dev/lab-vg/lvol0 /mnt/lvol0/

$ mount | grep lvol0
/dev/mapper/lab--vg-lvol0 on /mnt/lvol0 type ext4 (rw,relatime)
```

Une fois le système de fichiers monté, il est possible de créer des dossiers et des fichiers avec les permissions adaptées. Voici un exemple avec une attribution de dossier à l'utilisateur normal etu.

```
$ sudo mkdir /mnt/lvol0/etu-files
$ sudo chown etu.etu /mnt/lvol0/etu-files
$ touch /mnt/lvol0/etu-files/my-first-file
```

Q51. Comment visualiser l'état global des systèmes de fichiers et des montages en cours ?

Utiliser les commandes usuelles telles que df et lsblk.

Exemple de résultat attendu.

```
$ df -hT
Sys. de fichiers      Type      Taille Utilisé Dispo Uti% Monté sur
udev                  devtmpfs  463M   0    463M   0% /dev
tmpfs                 tmpfs     97M    700K   97M    1% /run
/dev/vda2             ext4      117G   2,0G   109G    2% /
tmpfs                 tmpfs     484M   0    484M   0% /dev/shm
tmpfs                 tmpfs     5,0M   0    5,0M   0% /run/lock
/dev/vda1             vfat      511M   3,5M   508M    1% /boot/efi
tmpfs                 tmpfs     97M    0    97M    0% /run/user/1000
/dev/mapper/lab--vg-lvol0 ext4      16G    28K   15G    1% /mnt/lvol0
```

```
$ lsblk
NAME                MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
sda                  8:0    0    32G  0 disk
sdb                  8:16   0    32G  0 disk
├─md0                 9:0    0    32G  0 raid1
│ └─lab--vg-lvol0    252:0  0    16G  0 lvm   /mnt/lvol0
sr0                  11:0    1 1024M  0 rom
vda                  254:0  0   120G  0 disk
├─vda1               254:1  0   512M  0 part  /boot/efi
├─vda2               254:2  0  118,5G  0 part  /
└─vda3               254:3  0   977M  0 part  [SWAP]
vdb                  254:16  0    32G  0 disk
├─md0                 9:0    0    32G  0 raid1
│ └─lab--vg-lvol0    252:0  0    16G  0 lvm   /mnt/lvol0
```

Cette dernière commande illustre bien l'état de la réplication RAID 1 en plus de l'utilisation du volume logique.

Q52. Comment créer deux instantanés du volume logique avec des jeux de fichiers différents ?

Après avoir créé une série de fichiers, rechercher les options de la commande `lvcreate` qui permettent de créer la première prise de vue (*snapshot*).

Création de 10 fichiers vides.

```
$ for i in {1..10}
do
touch /mnt/lvol0/etu-files/first-$(printf "%02d" $i)-file
done

$ ls -l /mnt/lvol0/etu-files/
first-01-file
first-02-file
first-03-file
first-04-file
first-05-file
first-06-file
first-07-file
first-08-file
first-09-file
first-10-file
my-first-file
```

Première capture instantanée du système de fichiers.

```
$ sudo lvcreate --snapshot --name first-snap -L 500M /dev/lab-vg/lvol0
Logical volume "first-snap" created.
```

Création de 10 nouveaux fichiers vides.

```
$ for i in {1..10}
do
touch /mnt/lvol0/etu-files/second-$(printf "%02d" $i)-file
done

$ ls -l /mnt/lvol0/etu-files/
first-01-file
first-02-file
first-03-file
first-04-file
first-05-file
first-06-file
first-07-file
first-08-file
first-09-file
first-10-file
my-first-file
second-01-file
second-02-file
second-03-file
second-04-file
second-05-file
second-06-file
second-07-file
second-08-file
second-09-file
second-10-file
```

Seconde capture instantanée du système de fichiers.

```
$ sudo lvcreate --snapshot --name second-snap -L 500M /dev/lab-vg/lvol0
Logical volume "second-snap" created.
```

État du volume logique.

```
$ sudo lvs
LV          VG      Attr      LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
first-snap  lab-vg  swi-a-s--- 500,00m      lvol0  0,01
lvol0       lab-vg  owi-aos--- 16,00g
second-snap lab-vg  swi-a-s--- 500,00m      lvol0  0,01
```

Q53. Comment tester la restauration du système de fichiers à partir des instantanés ?

Après avoir supprimé tous les fichiers du dossier `/mnt/lvol0/etu-files/`, restaurez le contenu des deux prises de vues dans l'ordre.

Supprimez des fichiers du répertoire de travail.

```
$ rm /mnt/lvol0/etu-files/*
```

Restaurez à partir du premier instantané.

```
$ sudo lvconvert --merge /dev/lab-vg/first-snap
Delaying merge since origin is open.
Merging of snapshot lab-vg/first-snap will occur on next activation of lab-vg/lvol0.
```

Pour que la restauration soit effective, il est nécessaire de désactiver/réactiver le volume logique à l'aide de la commande `lvchange`.

```
$ sudo lvchange --activate n lab-vg/lvol0
Logical volume lab-vg/lvol0 is used by another device.
```

Aïe ! Le volume logique est en cours d'utilisation. On doit donc démonter le système de fichiers et tester à nouveau.

```
$ sudo umount /mnt/lvol0
$ sudo lvchange --activate n lab-vg/lvol0
```

Cette fois ci, le volume est enfin désactivé. On peut le réactiver.

```
$ sudo lvchange --activate y lab-vg/lvol0
```

```
$ sudo lvscan
ACTIVE   Original '/dev/lab-vg/lvol0' [16,00 GiB] inherit
ACTIVE   Snapshot '/dev/lab-vg/second-snap' [500,00 MiB] inherit
```

La partition est bien disponible et on a retrouvé la liste des fichiers du premier instantané.

```
$ sudo mount /dev/lab-vg/lvol0 /mnt/lvol0/

$ ls -l /mnt/lvol0/etu-files/
first-01-file
first-02-file
first-03-file
first-04-file
first-05-file
first-06-file
first-07-file
first-08-file
first-09-file
first-10-file
my-first-file
```

Pour restaurer le contenu du second instantané, il faut reprendre les mêmes opérations à partir de la commande `lvconvert`.

## 11. Perte d'une unité de disque du tableau RAID 1

L'objectif de cette partie est de simuler la perte d'une unité de disque du tableau RAID 1 et de provoquer la reconstruction de ce tableau depuis l'unité de disque réseau iSCSI. On illustre ainsi le mécanisme de tolérance aux pannes en plus de l'utilisation des *snapshots* du gestionnaire de volumes logiques LVM.

Q54. Comment provoquer une panne de disque côté *Initiator* ?

1. Extinction de la machine virtuelle avec le rôle *Initiator*.
2. Suppression du fichier image du disque supplémentaire de la machine virtuelle.
3. Redémarrage de la même machine virtuelle.

## 12. Évaluation des performances

Pour obtenir des résultats fiables avec `sysbench`, un temps d'exécution de plusieurs heures est nécessaire. Les résultats donnés ici ne sont que des échantillons.

```
$ sudo apt install sysbench
```

## Unité de disque locale

Système de fichiers ext4.

```
$ mkdir /var/tmp/benchmark
```

```
$ cd /var/tmp/benchmark/
```

```
$ sysbench fileio prepare
```

```
$ sysbench fileio --file-test-mode=rndrw run
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)
```

```
Running the test with following options:
Number of threads: 1
Initializing random number generator from current time
```

```
Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
Read/Write ratio for combined random IO test: 1.50
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing random r/w test
Initializing worker threads...
```

```
Threads started!
```

```
File operations:
  reads/s:          6062.94
  writes/s:         4041.90
  fsyncs/s:        12939.18
```

```
Throughput:
  read, MiB/s:          94.73
  written, MiB/s:       63.15
```

```
General statistics:
  total time:          10.0062s
  total number of events: 230569
```

```
Latency (ms):
  min:                 0.00
  avg:                 0.04
  max:                 133.67
  95th percentile:   0.15
  sum:                 9943.24
```

```
Threads fairness:
  events (avg/stddev): 230569.0000/0.00
  execution time (avg/stddev): 9.9432/0.00
```

## Volume logique LVM sur une unité de disque RAID 1 avec un membre iSCSI

Système de fichiers ext4.

```
$ mkdir /mnt/lvol0/etu-files/benchmark
```

```
$ cd /mnt/lvol0/etu-files/benchmark
```

```
$ sysbench fileio prepare
```

```

$ sysbench fileio --file-test-mode=rndrw run
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time

Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
Read/Write ratio for combined random IO test: 1.50
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing random r/w test
Initializing worker threads...

Threads started!

File operations:
  reads/s:                1309.93
  writes/s:               873.29
  fsyncs/s:              2799.10

Throughput:
  read, MiB/s:          20.47
  written, MiB/s:      13.65

General statistics:
  total time:              10.0295s
  total number of events:  49850

Latency (ms):
  min:                    0.00
  avg:                    0.20
  max:                    31.26
  95th percentile:      0.59
  sum:                    9978.89

Threads fairness:
  events (avg/stddev):    49850.0000/0.00
  execution time (avg/stddev): 9.9789/0.00

```

## 13. Conclusion

---

Ce support de travaux pratiques offre une approche complète et progressive des technologies de stockage réseau modernes. Grâce à la manipulation concrète des protocoles iSCSI, de la redondance RAID 1 et de la gestion LVM, les étudiants acquièrent une expérience pratique essentielle pour comprendre les enjeux de la haute disponibilité et de la sauvegarde dans un environnement professionnel.

L'association de ces différentes technologies (stockage réseau, réplication synchrone, instantanés) illustre parfaitement la manière de construire une infrastructure robuste et évolutive. Les exercices proposés, de la configuration de base à l'évaluation des performances, préparent efficacement aux défis techniques du stockage en entreprise, tout en permettant de développer une compréhension approfondie des mécanismes sous-jacents.

## 14. Documents de référence

---

### Architecture réseau des travaux pratiques

**Infrastructure** : présentation de l'implantation des équipements d'interconnexion réseau dans les armoires de brassage et du plan d'adressage IP prédéfini pour l'ensemble des séances de travaux pratiques.

### Configuration d'une interface réseau

**Configuration d'une interface de réseau local** : tout sur la configuration des interfaces réseau de réseau local.

### iSCSI - Debian Wiki

La page **iSCSI and Debian** contient deux sous-rubriques sur les rôles *Initiator* et *Target*.