



Red Hat Enterprise Linux

7

Guide d'administration du stockage

Déployer et configurer un stockage à nœud unique dans Red Hat Enterprise Linux 7

Jacquelynn East

Red Hat Subject Matter
Experts

Hans de Goede

Daniel Novotny

Contributors

Rob Evers

Jeff Moyer

Don Domingo

Josef Bacik

Kamil Dudka

Harald Hoyer

Nathan Straz

Michael Christie

David Howells

Eric Sandeen

Doug Ledford

David Wysochanski

Sachin Prabhu

David Lehman

Mike Snitzer

Déployer et configurer un stockage à nœud unique dans Red Hat Enterprise Linux 7

Jacquelynn East
Red Hat Customer Content Services

Don Domingo
Red Hat Customer Content Services

Josef Bacik
Server Development Système de fichiers du noyau
jwhiter@redhat.com
Quotas de disques

Kamil Dudka
Base Operating System Services principaux - BRNO
kdudka@redhat.com
Listes de contrôle d'accès

Hans de Goede
Base Operating System Installateur
hdegoede@redhat.com
Partitions

Harald Hoyer
Engineering Software Engineering
harald@redhat.com
Systèmes de fichiers

Doug Ledford
Server Development Activation Matériel
dledford@redhat.com
RAID

Daniel Novotny
Base Operating System Services principaux - BRNO
dnovotny@redhat.com
Le système de fichiers /proc

Nathan Straz
Quality Engineering QE - Platform
nstraz@redhat.com
GFS2

David Wysochanski
Server Development Stockage du noyau
dwysocha@redhat.com
LVM/LVM2

Michael Christie
Server Development Stockage du noyau
mchristi@redhat.com

Stockage en ligne

Sachin Prabhu
Software Maintenance Engineering
sprabhu@redhat.com
NFS

Rob Evers
Server Development Stockage du noyau
revers@redhat.com
Stockage en ligne

David Howells
Server Development Activation Matériel
dhowells@redhat.com
FS-Cache

David Lehman
Base Operating System Installateur
dlehman@redhat.com
Configuration du stockage pendant l'installation

Jeff Moyer
Server Development Système de fichiers du noyau
jmoyer@redhat.com
Disques SSD

Eric Sandeen
Server Development Système de fichiers du noyau
esandeen@redhat.com
Systèmes de fichiers ext3, ext4 et XFS chiffrés

Mike Snitzer
Server Development Stockage du noyau
msnitzer@redhat.com
Pile E/S et Limites

Red Hat Subject Matter Experts

Contributors

Publié par

Milan Navratil
Red Hat Customer Content Services
mnavrati@redhat.com

Notice légale

Copyright © 2016 Red Hat Inc. and others.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](https://creativecommons.org/licenses/by-sa/3.0/). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Résumé

Ce guide propose des instructions pour gérer les périphériques de stockage et des systèmes de fichiers sur Red Hat Enterprise Linux 7 de manière efficace. Il a été conçu dans le but d'être utilisé par des administrateurs systèmes possédant un niveau de connaissances de Red Hat Enterprise Linux ou Fedora de base ou intermédiaire.

Table des matières

Chapitre 1. Aperçu	5
1.1. Les nouveautés de Red Hat Enterprise Linux 7	5
Partie I. Systèmes de fichiers	7
Chapitre 2. Structure et maintenance des systèmes de fichiers	8
2.1. Vue d'ensemble du standard de hiérarchie des systèmes de fichiers (FHS, ou « Filesystem Hierarchy Standard »)	8
2.2. Emplacement des fichiers Red Hat Enterprise Linux spéciaux	16
2.3. Système de fichiers virtuel /proc	16
2.4. Abandonner les blocs inutilisés	16
Chapitre 3. Btrfs (Aperçu technologique)	18
3.1. Création d'un système de fichiers btrfs	18
3.2. Monter un système de fichiers btrfs	18
3.3. Redimensionner un système de fichiers btrfs	19
3.4. Gestion des volumes intégrés de plusieurs périphériques	22
3.5. Optimisation SSD	26
3.6. Références btrfs	27
Chapitre 4. Système de fichiers Ext3	28
4.1. Créer un système de fichiers ext3	29
4.2. Conversion vers un système de fichiers ext3	29
4.3. Rétablir un système de fichiers Ext2	30
Chapitre 5. Le système de fichiers Ext4	32
5.1. Créer un système de fichiers Ext4	33
5.2. Monter un système de fichiers Ext4	34
5.3. Redimensionner un système de fichiers Ext4	35
5.4. Sauvegarde des systèmes de fichiers ext2/3/4	36
5.5. Restaurer un système de fichiers ext2/3/4	37
5.6. Autres utilitaires du système de fichiers Ext4	39
Chapitre 6. Le système de fichiers XFS	41
6.1. Créer un système de fichiers XFS	42
6.2. Monter un système de fichiers XFS	43
6.3. Gestion des quotas XFS	44
6.4. Augmenter la taille d'un système de fichiers XFS	46
6.5. Réparer un système de fichiers XFS	46
6.6. Suspendre un système de fichier XFS	47
6.7. Sauvegarde et restauration des systèmes de fichiers XFS	47
6.8. Autres utilitaires des systèmes de fichiers XFS	50
6.9. Migration de ext4 à XFS	51
Chapitre 7. Global File System 2	54
Chapitre 8. Network File System (NFS)	55
8.1. Fonctionnement NFS	55
8.2. pNFS	58
8.3. Configuration du client NFS	58
8.4. autofs	60
8.5. Options de montage NFS courantes	66
8.6. Démarrage et arrêt NFS	67
8.7. Configuration du serveur NFS	69

8.8. Sécurisation de NFS	75
8.9. NFS et rpcbind	77
8.10. Références	78
Chapitre 9. FS-Cache	80
9.1. Garantie des performances	81
9.2. Paramétrer un cache	81
9.3. Utiliser le cache avec NFS	82
9.4. Définir les limites d'élimination du cache	84
9.5. Informations statistiques	85
9.6. Références	85
Partie II. Administration du stockage	86
Chapitre 10. Besoins de stockage à prendre en compte pendant l'installation	87
10.1. Considérations particulières	87
Chapitre 11. Vérification du système de fichiers (fsck)	89
11.1. Meilleures pratiques avec fsck	89
11.2. Informations spécifiques aux systèmes de fichiers pour fsck	90
Chapitre 12. Partitions	94
12.1. Afficher la table de partitions	95
12.2. Création d'une partition	96
12.3. Suppression de partition	98
12.4. Redimensionnement d'une partition	99
Chapitre 13. Créer et maintenir des clichés avec Snapper	102
13.1. Configuration initiale de Snapper	102
13.2. Créer un cliché Snapper	103
13.3. Assurer le suivi des changements parmi les clichés Snapper	106
13.4. Annuler les changements entre les clichés	109
13.5. Suppression d'un instantané	111
Chapitre 14. Espace swap	112
14.1. Ajouter de l'espace swap	113
14.2. Supprimer de l'espace swap	115
14.3. Déplacer l'espace swap	116
Chapitre 15. System Storage Manager(SSM)	117
15.1. SSM Backends	117
15.2. Tâches SSM courantes	119
15.3. Ressources SSM	126
Chapitre 16. Quotas de disques	127
16.1. Configurer les quotas de disques	127
16.2. Gérer les quotas de disque	131
16.3. Références de quotas de disques	134
Chapitre 17. Réseau redondant de disques indépendants (RAID, de l'anglais « Redundant Array of Independent Disks »)	135
17.1. Types RAID	135
17.2. Niveaux RAID et prise en charge linéaire	136
17.3. Sous-systèmes RAID Linux	138
17.4. RAID Support in the Installer	139
17.5. Convertir un disque root en RAID 1 après l'installation	139
17.6. Configuring RAID Sets	140

17.6. Configuring RAID Sets	140
17.7. Création de périphériques RAID avancée	140
Chapitre 18. Utilisation de la commande mount	142
18.1. Répertorier les systèmes de fichiers actuellement montés	142
18.2. Monter un système de fichiers	143
18.3. Démonter un système de fichiers	152
18.4. Références de la commande mount	153
Chapitre 19. Fonction volume_key	154
19.1. Commandes volume_key	154
19.2. Exécutez volume_key en tant qu'utilisateur individuel	155
19.3. Exécutez volume_key dans une grande organisation	156
19.4. Références volume_key	158
Chapitre 20. Directives de déploiement des disques SSD	160
20.1. Considérations pour le déploiement	160
20.2. Considérations pour le paramétrage	161
Chapitre 21. Barrières d'écriture	163
21.1. Importance des barrières d'écriture	163
21.2. Activer/désactiver les barrières d'écriture	164
21.3. Considérations pour barrières d'écriture	164
Chapitre 22. Alignement et taille des E/S de stockage	166
22.1. Paramètres d'accès au stockage	166
22.2. Accès à l'espace utilisateur	167
22.3. Standards	168
22.4. Empiler les paramètres d'E/S	169
22.5. Gestionnaire de volumes logiques LVM	169
22.6. Outils des partitions et systèmes de fichiers	170
Chapitre 23. Paramétrer un système sans disque distant	172
23.1. Configuration d'un service tftp pour des clients sans disque	173
23.2. Configuration DHCP pour les clients sans disque	173
23.3. Configuration d'un système de fichiers exporté pour les clients sans disque	174
Chapitre 24. Online Storage Management	176
24.1. Installation de la Cible	176
24.2. Créer un initiateur iSCSI	185
24.3. Fibre Channel	186
24.4. Configurer une interface FCoE (« Fibre-Channel Over Ethernet »)	188
24.5. Configurer une interface FCoE pour qu'elle soit automatiquement montée lors du démarrage	189
24.6. iSCSI	191
24.7. Dénomination persistante	191
24.8. Supprimer un périphérique de stockage	196
24.9. Supprimer un chemin vers un périphérique de stockage	197
24.10. Ajouter un périphérique ou un chemin de stockage	198
24.11. Scanner les interconnexions du stockage	200
24.12. Configuration de la découverte iSCSI	201
24.13. Configurer le déchargement et la liaison d'interfaces iSCSI	202
24.14. Scanner les interconnexions iSCSI	206
24.15. Connexion à une cible iSCSI	209
24.16. Redimensionner une Unité logique En ligne	209
24.17. Ajouter ou supprimer une unité logique avec rescan-iscsi-bus.sh	213
24.18. Modifier le comportement de la perte de lien	214

24.19. Contrôle du Minuteur de commande SCSI et du Statut de périphérique	217
24.20. Résolution de problème de configuration de stockage en ligne	218
Chapitre 25. Mappeur de périphériques à multiples chemins d'accès et Stockage virtuel	220
25.1. Stockage virtuel	220
25.2. DM-Multipath	220
Chapitre 26. External Array Management (libStorageMgmt)	222
26.1. Qu'est-ce que libStorageMgmt ?	222
26.2. Terminologie de libStorageMgmt	223
26.3. Installation	224
26.4. Utilisation de libStorageMgmt	225
26.5. Documentation libStorageMgmt	230
Annexe A. Historique des versions	231
Index	231

Chapitre 1. Aperçu

Le *Guide d'administration du stockage* contient des informations détaillées sur les fonctionnalités des systèmes de fichiers et le stockage des données dans Red Hat Enterprise Linux 7. Cet ouvrage a été conçu dans le but de servir de référence rapide pour les administrateurs gérant des solutions de stockage à nœud unique (non-clusterisé).

Le guide d'administration du stockage est divisé en deux parties : Systèmes de fichiers et Administration du stockage.

La partie Systèmes de fichiers détaille les divers systèmes de fichiers pris en charge par Red Hat Enterprise Linux 7. Celle-ci les décrit et explique comment les utiliser de manière optimale.

La partie Administration du stockage détaille les divers outils et tâches d'administration du stockage pris en charge par Red Hat Enterprise Linux 7. Celle-ci les décrit et explique comment les utiliser de manière optimale.

1.1. Les nouveautés de Red Hat Enterprise Linux 7

Red Hat Enterprise Linux 7 apporte les améliorations suivantes au système de fichiers :

eCryptfs non inclus

À partir de Red Hat Enterprise Linux 7 eCryptfs n'est plus inclus. Voir le Guide de sécurité de Red Hat pour obtenir des informations sur les systèmes de fichiers encodés.

System Storage Manager

Red Hat Enterprise Linux 7 inclut une nouvelle application intitulée System Storage Manager. Elle fournit une interface de ligne de commande pour gérer les diverses technologies de stockage. Pour plus d'informations, voir [Chapitre 15, System Storage Manager\(SSM\)](#).

XFS est le Système de fichiers par défaut

À partir de Red Hat Enterprise Linux 7, XFS sera le système de fichiers par défaut. Pour plus d'informations sur le système de fichiers XFS, consulter [Chapitre 6, Le système de fichiers XFS](#).

Restructuration du système de fichiers

Red Hat Enterprise Linux 7 introduit une nouvelle structure de système de fichiers. Les répertoires `/bin`, `/sbin`, `/lib`, et `/lib64` se trouvent maintenant sous `/usr`.

Snapper

Red Hat Enterprise Linux 7 introduit un nouvel outil appelé snapper, qui permet une création et une gestion facile des clichés pour LVM et BTRFS. Pour plus d'informations, consulter [Chapitre 13, Créer et maintenir des clichés avec Snapper](#).

BTRFS (Aperçu technologique)

BTRFS est un système de fichiers local visant à fournir de meilleures performances et une meilleure évolutivité, y compris pour les opérations LVM intégrées. Ce système de fichiers est totalement pris en charge par Red Hat et à ce titre, représente un aperçu technologique. Pour obtenir des informations supplémentaires sur Btrfs, veuillez consulter [Chapitre 3, Btrfs \(Aperçu technologique\)](#).

NFSv2 n'est plus prise en charge

À partir de Red Hat Enterprise Linux 7, NFSv2 ne sera plus pris en charge.

Partie I. Systèmes de fichiers

La section sur les systèmes de fichiers explique la structure et la maintenance des systèmes de fichiers, suivie un aperçu technologiques Btrfs et les systèmes de fichiers que Red Hat prend entièrement en charge : ext3, ext4, GFS2, XFS, NFS et FS-Cache.

Pour un aperçu général des systèmes de fichiers Red Hat Enterprise Linux et les limites de stockage, voir [Red Hat Enterprise Linux technology capabilities and limits](#) dans la Base de connaissances de Red Hat.

Chapitre 2. Structure et maintenance des systèmes de fichiers

La structure des systèmes de fichiers est le niveau le plus bas dans l'organisation d'un système d'exploitation. La manière par laquelle un système d'exploitation interagit avec ses utilisateurs, ses applications et son modèle de sécurité dépend presque toujours de la façon dont le système d'exploitation organise les fichiers sur les périphériques de stockage. Le fait de fournir une structure commune de systèmes de fichiers permet d'assurer que les utilisateurs et les programmes puissent accéder et aux fichiers et écrire dessus.

Les systèmes de fichiers divisent les fichiers en deux catégories logiques :

- ✦ Fichiers partageables vs fichiers non-partageables
- ✦ Fichiers variables vs fichiers statiques

Les fichiers *partageables* peuvent être accédés localement et par des hôtes à distance ; les fichiers *non-partageables* sont uniquement disponibles localement. Les fichiers *variables*, tels que des documents, peuvent être modifiés à tout moment ; les fichiers *statiques*, tels que des binaires, ne peuvent pas changer sans action de la part de l'administrateur.

Cette manière de catégoriser des fichiers permet de corréler la fonction de chaque fichier avec les permissions assignées aux répertoires qui les contiennent. La manière par laquelle le système d'exploitation et ses utilisateurs interagissent avec un fichier détermine le répertoire dans lequel il sera placé, que ce répertoire soit monté avec des permissions de lecture seule ou de lecture/écriture, ainsi que le niveau d'accès au fichier de chaque utilisateur. Le niveau le plus haut de l'organisation est crucial, l'accès aux répertoires sous-jacents peut être restreint car des problèmes de sécurité pourraient surgir si les règles d'accès n'adhèrent pas à une structure rigide depuis le haut vers le bas.

2.1. Vue d'ensemble du standard de hiérarchie des systèmes de fichiers (FHS, ou « Filesystem Hierarchy Standard »)

Red Hat Enterprise Linux utilise la structure de système de fichiers *FHS* (*Filesystem Hierarchy Standard*) qui définit les noms, emplacements, et permissions pour de nombreux types de fichiers et répertoires.

Le document FHS est la référence faisant figure d'autorité pour tout système de fichiers conforme à FHS, mais le standard laisse de nombreuses zones non définies ou extensibles. Cette section présente une vue d'ensemble du standard et une description des parties du système de fichiers non couvertes par le standard.

Les deux éléments les plus importants permettant d'établir la conformité FHS incluent :

- ✦ La compatibilité avec d'autres systèmes conformes à FHS
- ✦ La possibilité de monter une partition `/usr/` en lecture seule. Ceci est particulièrement important car `/usr/` contient des fichiers exécutables communs et ne devrait pas être modifié par les utilisateurs. En outre, comme la partition `/usr/` est montée en lecture seule, elle devrait pouvoir être montée à partir du lecteur CD-ROM ou depuis une autre machine via un montage NFS en lecture seule.

2.1.1. Organisation FHS

Les répertoires et fichiers notés ici représentent un sous-ensemble de ceux qui sont spécifiés par le document FHS. Veuillez consulter la documentation FHS la plus récente sur <http://www.pathname.com/fhs/> afin d'obtenir les informations les plus complètes.

**Note**

Les répertoires disponibles dépendent de ce qui a été installé sur un système donné. Les listes suivantes ne représentent qu'un exemple de ce que l'on s'attend à y trouver.

2.1.1.1. Collecte des informations sur les systèmes de fichiers

La commande **df** rapporte l'utilisation de l'espace disque du système. Sa sortie est similaire à la suivante :

Exemple 2.1. Sortie de la commande df

```
Filesystem          1K-blocks      Used Available Use% Mounted on
/dev/mapper/VolGroup00-LogVol100
                    11675568    6272120    4810348   57% / /dev/sda1
                    100691       9281       86211    10% /boot
none                 322856        0         322856    0% /dev/shm
```

Par défaut, **df** affiche la taille de la partition en blocs de 1 kilo-octets, ainsi que la quantité d'espace disque utilisée et disponible en kilo-octets. Pour afficher ces informations en méga-octets et giga-octets, veuillez exécuter la commande **df -h**. L'argument **-h** se traduit par l'utilisation d'un format lisible (« human-readable »). La sortie de **df -h** est similaire à la suivante :

Exemple 2.2. Sortie de la commande df -h

```
Filesystem          Size  Used Avail Use% Mounted on
/dev/mapper/VolGroup00-LogVol100
                    12G  6.0G  4.6G   57% / /dev/sda1
   99M  9.1M   85M  10% /boot
none   316M     0  316M   0% /dev/shm
```

**Note**

Dans les exemples ci-dessus, la partition montée **/dev/shm** représente le système de fichiers de la mémoire virtuelle du système.

La commande **du** affiche la quantité estimée d'espace utilisé par des fichiers dans un répertoire, et l'utilisation d'espace disque de chaque sous-répertoire. La dernière ligne dans la sortie de **du** affiche la totalité de l'utilisation d'espace disque du répertoire. Pour afficher la totalité de l'utilisation d'espace disque sous un format lisible, veuillez utiliser **du -hs**. Pour plus d'options, veuillez consulter **man du**.

Pour afficher les partitions du système et l'utilisation de l'espace disque sous un format graphique, veuillez utiliser la **Surveillance système** (« System Monitor ») de Gnome en cliquant sur **Applications** → **Outils système** → **Surveillance système** ou en utilisant la commande **gnome-system-monitor**. Sélectionnez l'onglet **Systèmes de fichiers** pour afficher les partitions du système. La figure ci-dessous illustre l'onglet **Systèmes de fichiers**.

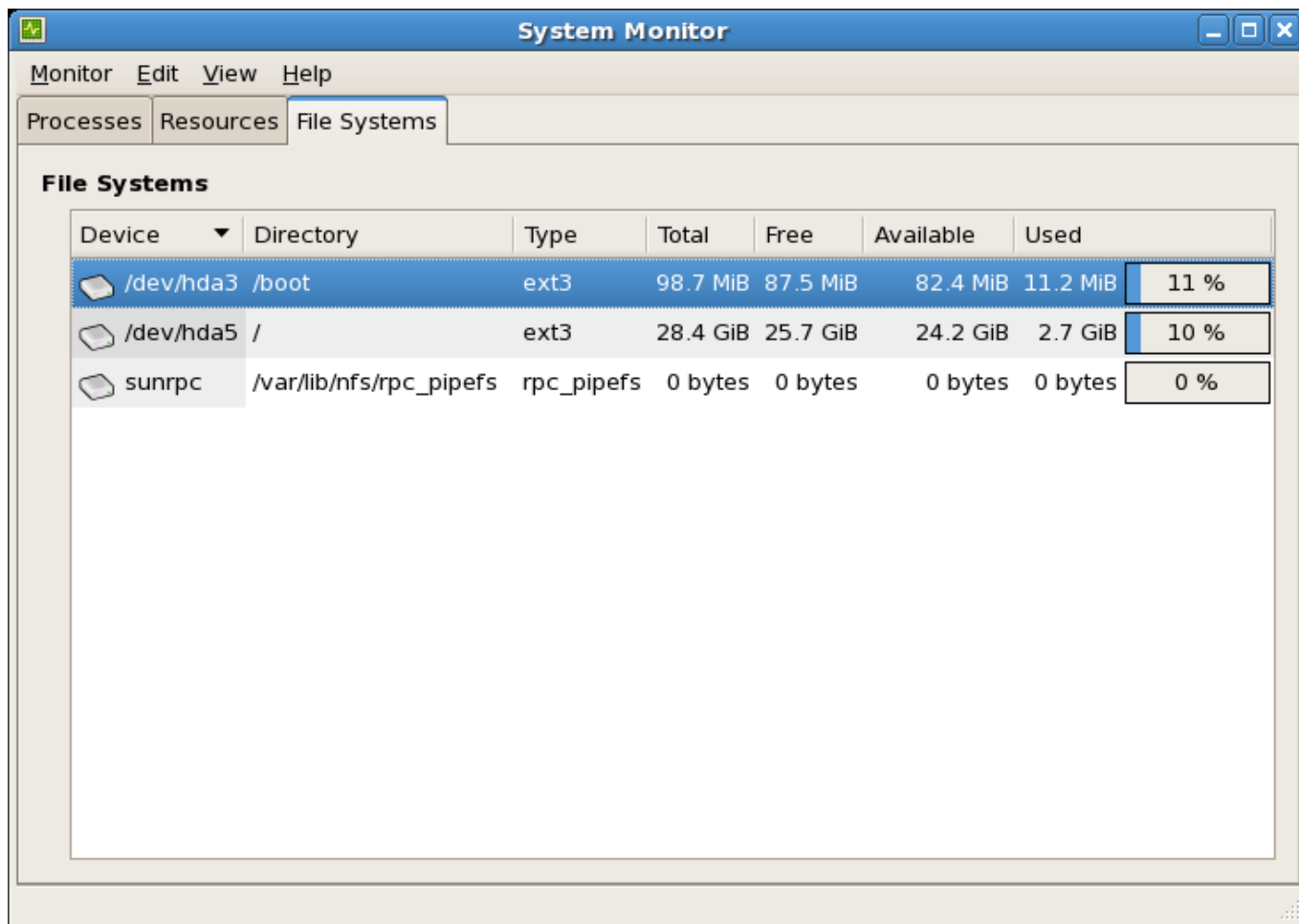


Figure 2.1. Onglet Surveillance système GNOME des systèmes de fichiers

2.1.1.2. Répertoire `/boot/`

Le répertoire `/boot/` contient des fichiers statiques requis pour démarrer le système, par exemple le noyau Linux. Ces fichiers sont essentiels pour que le système puisse démarrer correctement.



Avertissement

Ne pas supprimer le répertoire `/boot/`. Le système ne pourra plus être démarré si ce répertoire est supprimé.

2.1.1.3. Répertoire `/dev/`

Le répertoire `/dev/` contient des nœuds de périphériques qui représente les types de périphériques suivants :

- » les périphériques attachés au système ;
- » les périphériques fournis par le noyau.

Ces nœuds de périphérique sont essentiels au bon fonctionnement du système. Le démon `udev` crée et supprime les nœuds de périphérique dans `/dev/` selon les besoins.

Les périphériques se trouvant dans le répertoire `/dev/` et ses sous-répertoires sont définis en tant que *caractère* (fournissant uniquement un flux en série d'entrées et sortie, par exemple une souris ou un clavier)

ou *bloc* (accessible de manière aléatoire, par exemple un disque dur ou un lecteur de disquettes). Si GNOME ou KDE est installé, certains périphériques de stockage seront automatiquement détectés lorsqu'ils sont connectés (comme les lecteurs USB), ou insérés (comme avec un lecteur CD ou DVD), puis une fenêtre contextuelle affichant le contenu apparaîtra.

Tableau 2.1. Exemples de fichiers communs dans le répertoire /dev

Fichier	Description
/dev/hda	Périphérique maître sur le canal IDE principal.
/dev/hdb	Périphérique esclave sur le canal IDE principal.
/dev/tty0	Première console virtuelle.
/dev/tty1	Seconde console virtuelle.
/dev/sda	Premier périphérique sur le canal principal SCSI ou SATA.
/dev/lp0	Premier port parallèle.

Un périphérique bloc valide peut être de l'un des deux types de d'entrées suivants :

Un périphérique mappé

Un volume logique dans un groupe de volumes. Par exemple, **/dev/mapper/VolGroup00-LogVol02**.

Un périphérique statique

Un volume de stockage traditionnel. Par exemple, **/dev/sdbX**, où *sdb* est un nom de périphérique de stockage et où *X* est le numéro de la partition. **/dev/sdbX** peut également être **/dev/disk/by-id/WWID**, ou **/dev/disk/by-uuid/UUID**, (voir [Section 24.7, « Dénomination persistante »](#) pour obtenir plus d'informations sur ces options).

2.1.1.4. Répertoire /etc/

Le répertoire **/etc/** est réservé aux fichiers de configuration qui sont locaux à l'ordinateur. Il ne doit contenir aucun fichier binaire ; tout fichier binaire devrait être déplacé dans **/usr/bin/** ou **/usr/sbin/**.

Par exemple, le répertoire **/etc/skel/** stocke les fichiers utilisateur « squelette », qui sont utilisés pour remplir un répertoire de base lorsqu'un utilisateur est créé pour la première fois. Les applications stockent aussi leurs fichiers de configuration dans ce répertoire et peuvent les référencer lors de leur exécution. Le fichier **/etc/exports** contrôle quels systèmes de fichiers sont exportés vers des hôtes distants.

2.1.1.5. Répertoire /mnt/

Le répertoire **/mnt/** est réservé aux systèmes de fichiers montés de manière temporaire, comme les montages de systèmes de fichiers NFS. Pour tous les supports de stockage amovibles, veuillez utiliser le répertoire **/media/**. Les supports de stockage amovibles détectés automatiquement seront montés dans le répertoire **/media**.



Important

Le répertoire **/mnt** ne doit pas être utilisé par des programmes d'installation.

2.1.1.6. Répertoire /opt/

Le répertoire **/opt/** est habituellement réservé aux paquets logiciels et aux paquets de modules complémentaires ne faisant pas partie de l'installation par défaut. Un paquet effectuant une installation sur **/opt/** crée un répertoire portant son nom, par exemple, **/opt/*packagename*/**. Dans la plupart des cas, ce genre de paquets observe une structure prédictible de sous-répertoires. La plupart stockent leurs binaires dans **/opt/*packagename*/bin/** et leurs pages **man** dans **/opt/*packagename*/man/**.

2.1.1.7. Répertoire **/proc/**

Le répertoire **/proc/** contient des fichiers spéciaux qui extraient des informations du noyau ou y envoient des informations. Des exemples de ce genre d'informations incluent la mémoire système, des informations sur le CPU et la configuration du matériel. Pour obtenir des informations supplémentaires sur **/proc/**, veuillez consulter le [Section 2.3, « Système de fichiers virtuel /proc »](#).

2.1.1.8. Répertoire **/srv/**

Le répertoire **/srv/** contient des données spécifiques au site servies par un système Red Hat Enterprise Linux. Ce répertoire donne aux utilisateurs l'emplacement des fichiers de données pour un service particulier, tel que FTP, WWW, ou CVS. Les données pertinentes à un utilisateur en particulier doivent être placées dans le répertoire **/home/**.

2.1.1.9. Répertoire **/sys/**

Le répertoire **/sys/** utilise le nouveau système de fichiers virtuel spécifique au noyau, **sysfs**. Grâce à la meilleure prise en charge de l'enfichage à chaud de périphériques matériels dans le noyau, le répertoire **/sys/** contient des informations similaires à celles qui sont offertes par **/proc/**, mais affiche une vue hiérarchique des informations des périphériques qui est spécifique aux périphériques enfichables à chaud.

2.1.1.10. Répertoire **/usr/**

Le répertoire **/usr/** est utilisé pour les fichiers pouvant être partagés à travers de multiples machines. Le répertoire **/usr/** se trouve souvent sur sa propre partition et est monté en lecture seule. Au minimum, le répertoire **/usr/** doit contenir les sous-répertoires suivants :

/usr/bin

Ce répertoire est utilisé pour les binaires.

/usr/etc

Ce répertoire est utilisé pour les fichiers de configuration globaux.

/usr/games

Ce répertoire est utilisé pour stocker les jeux.

/usr/include

Ce répertoire est utilisé pour les fichiers en-têtes C.

/usr/kerberos

Ce répertoire est utilisé pour les fichiers et binaires liés à Kerberos.

/usr/lib

Ce répertoire est utilisé pour les fichiers objets et les bibliothèques qui ne sont pas conçus pour être directement utilisés par des scripts shell ou des utilisateurs.

À partir de Red Hat Enterprise Linux 7.0, le répertoire **/lib/** a été mergé avec **/usr/lib**. Il doit également contenir les bibliothèques nécessaires à l'exécution des binaires dans **/usr/bin/** et **/usr/sbin/**. Ces images de bibliothèques partagées sont utilisées pour démarrer le système ou exécuter des commandes à l'intérieur du système de fichiers.

/usr/libexec

Ce répertoire contient les programmes d'assistance de petite taille appelés par d'autres programmes.

/usr/sbin

À partir de Red Hat Enterprise Linux 7.0, **/sbin** a été déplacé dans **/usr/sbin**. Cela signifie qu'il contient tous les binaires d'administration système, y compris ceux qui sont essentiels pour démarrer, restaurer, recouvrer ou réparer le système. Les binaires de **/usr/sbin/** ont besoin de privilèges root pour être utilisés.

/usr/share

Ce répertoire stocke les fichiers qui ne sont pas particuliers à l'architecture.

/usr/src

Ce répertoire stocke le code source.

/usr/tmp lié à **/var/tmp**

Ce répertoire stocke les fichiers temporaires.

Le répertoire **/usr/** devrait aussi contenir un sous-répertoire **/local/**. Comme recommandé par la norme FHS, ce sous-répertoire est utilisé par l'administrateur système lors de l'installation locale de logiciels et ne doit pas être écrasé pendant les mises à jour du système. Le répertoire **/usr/local** possède une structure similaire à **/usr/** et contient les sous-répertoires suivants :

- » **/usr/local/bin**
- » **/usr/local/etc**
- » **/usr/local/games**
- » **/usr/local/include**
- » **/usr/local/lib**
- » **/usr/local/libexec**
- » **/usr/local/sbin**
- » **/usr/local/share**
- » **/usr/local/src**

L'utilisation par Red Hat Enterprise Linux de **/usr/local/** diffère légèrement de la norme FHS. La norme FHS déclare que **/usr/local/** doit être utilisé pour stocker des logiciels qui ne doivent pas être affectés par les mises à niveau de logiciels système. Comme le gestionnaire de paquet RPM, « **RPM Package Manager** », peut effectuer des mises à niveau de logiciels en toute sécurité, il n'est pas nécessaire de

protéger les fichiers en les stockant dans **/usr/local/**.

À la place, Red Hat Enterprise Linux utilise **/usr/local/** pour les logiciels locaux. Par exemple, si le répertoire **/usr/** est monté en tant que partage NFS, en lecture seule, à partir d'un hôte distant, il est toujours possible d'installer un paquet ou programme sous le répertoire **/usr/local/**.

2.1.1.11. Répertoire **/var/**

Comme la norme FHS requiert que Linux monte **/usr/** en lecture seule, tout programme qui écrit des fichiers journaux ou nécessite les répertoires **spool/** ou **lock/** doit les écrire sur le répertoire **/var/**. La norme FHS déclare que **/var/** est utilisé pour les données variables, ce qui inclut les répertoires et fichier **spool**, les données de journalisation, et les fichiers transitoires et temporaires.

Voici quelques répertoires qui se trouvent dans le répertoire **/var/** :

- » **/var/account/**
- » **/var/arpwatch/**
- » **/var/cache/**
- » **/var/crash/**
- » **/var/db/**
- » **/var/empty/**
- » **/var/ftp/**
- » **/var/gdm/**
- » **/var/kerberos/**
- » **/var/lib/**
- » **/var/local/**
- » **/var/lock/**
- » **/var/log/**
- » **/var/mail** lié à **/var/spool/mail/**
- » **/var/mailman/**
- » **/var/named/**
- » **/var/nis/**
- » **/var/opt/**
- » **/var/preserve/**
- » **/var/run/**
- » **/var/spool/**
- » **/var/tmp/**

- » `/var/tux/`
- » `/var/www/`
- » `/var/yp/`



Important

Le répertoire `/var/run/media/user` contient des sous-répertoires utilisés comme points de montage pour des supports amovibles, tels que les supports de stockage USB, les DVD, les CD-ROM, et les disques Zip. Notez qu'auparavant, le répertoire `/media/` était utilisé pour cela.

Les fichiers de journalisation du système, tels que `messages` et `lastlog`, sont placés dans le répertoire `/var/log/`. Le répertoire `/var/lib/rpm/` contient des bases de données RPM du système. Les fichiers de verrouillage sont placés dans le répertoire `/var/lock/`, habituellement dans les répertoires du programme utilisant le fichier. Le répertoire `/var/spool/` contient des sous-répertoires qui stockent les fichiers de données de certains programmes. Ces sous-répertoires incluent :

- » `/var/spool/at/`
- » `/var/spool/clientmqueue/`
- » `/var/spool/cron/`
- » `/var/spool/cups/`
- » `/var/spool/exim/`
- » `/var/spool/lpd/`
- » `/var/spool/mail/`
- » `/var/spool/mailman/`
- » `/var/spool/mqueue/`
- » `/var/spool/news/`
- » `/var/spool/postfix/`
- » `/var/spool/repackage/`
- » `/var/spool/rwho/`
- » `/var/spool/samba/`
- » `/var/spool/squid/`
- » `/var/spool/squirrelmail/`
- » `/var/spool/up2date/`
- » `/var/spool/uucp/`
- » `/var/spool/uucppublic/`
- » `/var/spool/vbox/`

2.2. Emplacement des fichiers Red Hat Enterprise Linux spéciaux

Red Hat Enterprise Linux étend légèrement la structure FHS pour accommoder les fichiers spéciaux.

La plupart des fichiers se rapportant aux RPM sont conservés dans le répertoire `/var/lib/rpm/`. Pour obtenir des informations supplémentaires sur les RPM, veuillez consulter `man rpm`.

Le répertoire `/var/cache/yum/` contient des fichiers utilisés par **Package Updater**, y compris les informations d'en-tête RPM de ce système. L'emplacement peut aussi être utilisé pour stocker temporairement les RPM téléchargés pendant la mise à jour du système. Pour obtenir davantage d'informations sur Red Hat Network, veuillez consulter la documentation en ligne sur <https://rhn.redhat.com/>.

Le répertoire `/etc/sysconfig/` est un autre emplacement spécifique à Red Hat Enterprise Linux. Ce répertoire stocke toute un ensemble d'informations de configuration. De nombreux scripts exécutés lors du démarrage utilisent des fichiers situés dans ce répertoire.

2.3. Système de fichiers virtuel /proc

Contrairement à la plupart des systèmes de fichiers, `/proc` ne contient ni texte, ni fichiers binaires. Au lieu de cela, il héberge des *fichiers virtuels* ; ainsi, `/proc` fait habituellement référence à un système de fichiers virtuel. La taille typique de ces fichiers virtuels est de zéro octets, même s'ils contiennent de grandes quantités d'informations.

Le système de fichiers `/proc` n'est pas utilisé pour le stockage. Son but principal est de fournir une interface basée sur fichiers pour le matériel, la mémoire, les processus en cours d'exécution, ainsi que pour les autres composants du système. Des informations en temps réel peuvent être récupérées sur de nombreux composants de système en affichant son fichier `/proc` correspondant. Certains des fichiers dans `/proc` peuvent également être manipulés (par les utilisateurs et les applications) pour configurer le noyau.

Les fichiers `/proc` suivants sont utiles pour le contrôle et la gestion du stockage du système :

`/proc/devices`

Affiche divers périphériques bloc et caractères actuellement configurés.

`/proc/filesystems`

Répertorie les types de systèmes de fichiers actuellement pris en charge par le noyau.

`/proc/mdstat`

Contient des informations sur les configurations à disques multiples ou les configurations RAID sur le système, s'il y en a.

`/proc/mounts`

Répertorie tous les montages en cours d'utilisation par le système.

`/proc/partitions`

Contient les informations sur l'allocation de blocs de partitions.

Pour obtenir des informations supplémentaires sur le système de fichiers `/proc`, veuillez consulter le *Guide de déploiement* Red Hat Enterprise Linux 7.

2.4. Abandonner les blocs inutilisés

Les opérations d'abandon par lot et d'abandon en ligne sont des fonctionnalités de systèmes de fichiers montés qui abandonnent les blocs non utilisés par le système de fichiers. Celles-ci sont utiles pour les lecteurs SSD et les stockages alloués dynamiquement.

Les *opérations d'abandon par lot* sont exécutées de manière explicite par l'utilisateur avec la commande **fstrim**. Cette commande abandonne tous les blocs inutilisés dans un système de fichiers correspondant aux critères de l'utilisateur. Ces deux types d'opération sont pris en charge pour une utilisation avec les systèmes de fichier ext4 dans Red Hat Enterprise Linux 6.2 et ses versions supérieures, tant que le périphérique bloc sous-jacent au système de fichiers prend en charge les opérations d'abandon physique. Ceci est aussi le cas avec les systèmes de fichiers XFS dans Red Hat Enterprise Linux 6.4 et ses versions supérieures. Les opérations d'abandon physique sont prises en charge si la valeur de **/sys/block/device/queue/discard_max_bytes** n'est pas zéro.

Les *opérations d'abandon en ligne* sont spécifiées lors du montage avec l'option **-o discard** (soit dans **/etc/fstab** ou en faisant partie de la commande **mount**) et elles sont exécutées en temps réel sans intervention de la part de l'utilisateur. Les opérations d'abandon en ligne abandonnent uniquement les blocs passant de « Utilisé » à « Libre ». Les opérations d'abandon en ligne sont prises en charge sur les systèmes de fichiers ext4 dans Red Hat Enterprise Linux 6.2 et ses versions supérieures, ainsi que sur les systèmes de fichiers XFS dans Red Hat Enterprise Linux 6.4 et ses versions supérieures.

Red Hat recommande les opérations d'abandon par lot, à moins que la charge de travail du système soit telle que l'abandon par lot ne soit pas faisable, ou que des opérations d'abandon en ligne soient nécessaires pour effectuer la maintenance.

Chapitre 3. Btrfs (Aperçu technologique)

Btrfs offre des fonctionnalités de gestion, de fiabilité et d'évolutivité avancées. Il permet aux utilisateurs de créer des clichés et autorise la compression et une gestion des périphériques intégrée



Important

BTRFS est un aperçu technologique dans Red Hat Enterprise Linux 7.)

3.1. Création d'un système de fichiers btrfs

Pour établir un système de fichiers btrfs, veuillez utiliser la commande suivante :

```
# mkfs.btrfs /dev/device
```

Pour plus d'informations sur la création de systèmes de fichiers avec des périphériques supplémentaires, et pour spécifier des profils multi-périphériques de métadonnées et de données, voir [Section 3.4, « Gestion des volumes intégrés de plusieurs périphériques »](#).

3.2. Monter un système de fichiers btrfs

Pour monter un périphérique sur un système de fichiers btrfs, veuillez utiliser la commande suivante :

```
# mount /dev/device /mount-point
```

Autres options de montage :

device=devname

En ajoutant cette option à la commande de montage ordonne btrfs à scanner un volume btrfs dans le périphérique nommé. C'est utilisé pour veiller à ce que le montage réussisse car tenter de monter les périphériques qui ne sont pas btrfs peuvent entraîner l'échec du montage.



Note

Cela ne signifie pas que tous les périphériques seront ajoutés au système de fichiers, cela ne fait que de les scanner.

max_inline=number

Utiliser cette option pour définir le montant d'espace maximum (en octets) pouvant être utilisé pour aligner les données dans un B-tree Leaf de métadonnées. La valeur par défaut est de 8192 octets.

alloc_start=number

Utiliser cette option pour définir où les allocations débutent dans le disque.

thread_pool=number

Utiliser cette option pour assigner le nombre de threads de workers alloués.

discard

Utiliser cette option pour activer discard/TRIM sur les blocs libérés.

noacl

Utiliser cette option pour désactiver l'utilisation des ACL.

space_cache

Utiliser cette option pour libérer les données d'espace libre du disque pour mettre en cache un groupe de blocs plus rapidement. Il s'agit d'un changement persistant et on peut amorcer d'anciens noyaux en toute sécurité.

nospace_cache

Utiliser cette option pour désactiver le **space_cache** ci-dessus.

clear_cache

Utiliser cette option pour libérer les caches d'espace-libre pendant le montage. Il s'agit d'une option sûre mais qui entraînera sa reconstruction. Par conséquent, laissez le système de fichiers monté pour que le processus de reconstruction puisse s'achever. Cette option de montage à pour but d'être utilisée une fois et uniquement après que les problèmes d'espace libre soient apparents.

enospc_debug

Cette option est utilisée pour déboguer les problèmes "no space left"

recovery

Utiliser cette option pour permettre le recouvrement automatique lors du montage.

3.3. Redimensionner un système de fichiers btrfs

Il n'est pas possible de redimensionner un système de fichiers btrfs mais il est possible de redimensionner chaque périphérique qu'il utilise. S'il n'y a qu'un périphérique utilisé, alors cela fonctionne de la même façon que redimensionner un système de fichiers. S'il y a plusieurs périphériques utilisés, ils devront être redimensionnés manuellement pour obtenir le résultat escompté.



Note

La taille d'une unité n'est pas sensible à la casse : **G** ou **g** sont acceptés pour GiB.

La commande n'accepte pas **t** pour terabytes ou **p** pour petabytes. Elle accepte uniquement **k**, **m**, et **g**.

Comment élargir un système de fichiers btrfs

Pour élargir le système de fichiers sur un seul périphérique, utiliser la commande suivante :

```
# btrfs filesystem resize amount /mount-point
```

Par exemple:


```
# btrfs filesystem resize +200M /btrfssingle
Resize '/btrfssingle' of '+200M'
```

Pour élargir un système de fichiers multi-périphériques, le périphérique à élargir doit être spécifié. Tout d'abord, afficher tous les périphériques qui ont un système de fichiers sur un point de montage indiqué :

```
# btrfs filesystem show /mount-point
```

Par exemple:

```
# btrfs filesystem show /btrfstest
Label: none  uuid: 755b41b7-7a20-4a24-abb3-45fdbed1ab39
Total devices 4 FS bytes used 192.00KiB
devid    1 size 1.00GiB used 224.75MiB path /dev/vdc
devid    2 size 524.00MiB used 204.75MiB path /dev/vdd
devid    3 size 1.00GiB used 8.00MiB path /dev/vde
devid    4 size 1.00GiB used 8.00MiB path /dev/vdf

Btrfs v3.16.2
```

Puis, après avoir identifié le **devid** du périphérique à élargir, utiliser la commande suivante :

```
# btrfs filesystem resize devid:amount /mount-point
```

Par exemple:

```
# btrfs filesystem resize 2:+200M /btrfstest
Resize '/btrfstest/' of '2:+200M'
```



Note

Le *montant* peut également être **max** au lieu d'un montant spécifique. Cela utilisera tout l'espace libre restant qui se trouve sur le périphérique.

Réduire un système de fichiers btrfs

Pour réduire le système de fichiers sur un seul périphérique, utiliser la commande suivante :

```
# btrfs filesystem resize amount /mount-point
```

Par exemple:

```
# btrfs filesystem resize -200M /btrfssingle
Resize '/btrfssingle' of '-200M'
```

Pour réduire un système de fichiers multi-périphériques, le périphérique à réduire doit être spécifié. Tout d'abord, afficher tous les périphériques qui ont un système de fichiers sur le point de montage indiqué :

```
# btrfs filesystem show /mount-point
```

Par exemple:

```
# btrfs filesystem show /btrfstest
Label: none  uuid: 755b41b7-7a20-4a24-abb3-45fdbed1ab39
Total devices 4 FS bytes used 192.00KiB
devid    1 size 1.00GiB used 224.75MiB path /dev/vdc
devid    2 size 524.00MiB used 204.75MiB path /dev/vdd
devid    3 size 1.00GiB used 8.00MiB path /dev/vde
devid    4 size 1.00GiB used 8.00MiB path /dev/vdf

Btrfs v3.16.2
```

Puis, après avoir identifié le **devid** du périphérique à réduire, utiliser la commande suivante :

```
# btrfs filesystem resize devid:amount /mount-point
```

Par exemple:

```
# btrfs filesystem resize 2:-200M /btrfstest
Resize '/btrfstest' of '2:-200M'
```

Définir la taille du système de fichiers

Pour définir le système de fichiers à une taille spécifique sur un seul périphérique, utiliser la commande suivante :

```
# btrfs filesystem resize amount /mount-point
```

Par exemple:

```
# btrfs filesystem resize 700M /btrfssingle
Resize '/btrfssingle' of '700M'
```

Pour définir la taille d'un système de fichiers multi-périphériques, le périphérique à modifier doit être spécifié. Tout d'abord, afficher tous les périphériques qui ont un système de fichiers au point de montage indiqué :

```
# btrfs filesystem show /mount-point
```

Par exemple:

```
# btrfs filesystem show /btrfstest
Label: none  uuid: 755b41b7-7a20-4a24-abb3-45fdbed1ab39
Total devices 4 FS bytes used 192.00KiB
devid    1 size 1.00GiB used 224.75MiB path /dev/vdc
devid    2 size 724.00MiB used 204.75MiB path /dev/vdd
devid    3 size 1.00GiB used 8.00MiB path /dev/vde
devid    4 size 1.00GiB used 8.00MiB path /dev/vdf

Btrfs v3.16.2
```

Puis, après avoir identifié le **devid** du périphérique à modifier, utiliser la commande suivante :

```
# btrfs filesystem resize devid:amount /mount-point
```

Par exemple:

```
# btrfs filesystem resize 2:300M /btrfstest
Resize '/btrfstest' of '2:300M'
```

3.4. Gestion des volumes intégrés de plusieurs périphériques

Un système de fichiers btrfs peut être créé par dessus plusieurs périphériques, et des périphériques supplémentaires peuvent être ajoutés une fois que le système de fichiers a été créé. Par défaut, les métadonnées seront mises en miroir sur deux périphériques, et les données seront réparties sur tous les périphériques présents. Cependant, si un seul périphérique est présent, les métadonnées seront dupliquées sur ce périphérique.

3.4.1. Création de système de fichiers avec plusieurs périphériques

La commande `mkfs.btrfs` détaillée dans [Section 3.1, « Création d'un système de fichiers btrfs »](#) accepte les options `-d` pour les données, et `-m` pour les métadonnées. Les spécifications acceptées sont :

- » **raid0**
- » **raid1**
- » **raid10**
- » **dup**
- » **single**

L'option `-m single` indique qu'aucun duplicata de métadonnées n'a lieu. C'est souhaitable quand on utilise un raid de matériel.



Note

Raid 10 exige quatre périphériques au moins exécutant correctement.

Exemple 3.1. Créer un système de fichiers btrfs Raid 10

Créer un système de fichiers sur quatre périphériques (métadonnées mises en miroir, données réparties).

```
# mkfs.btrfs /dev/device1 /dev/device2 /dev/device3 /dev/device4
```

Répartir les données sans les mettre en miroir.

```
# mkfs.btrfs -m raid0 /dev/device1 /dev/device2
```

Utiliser Raid 10 pour les données et les métadonnées à la fois.

```
# mkfs.btrfs -m raid10 -d raid10 /dev/device1 /dev/device2 /dev/device3
/dev/device4
```

Ne pas dupliquer les métadonnées sur un seul disque.

```
# mkfs.btrfs -m single /dev/device
```

Utiliser l'option **single** pour utiliser toute la capacité de chaque disque quand les disques sont de tailles différentes.

```
# mkfs.btrfs -d single /dev/device1 /dev/device2 /dev/device3
```

Pour ajouter un nouveau périphérique à un système de fichiers multi périphériques déjà créé, utiliser la commande suivante :

```
# btrfs device add /dev/device1 /mount-point
```

Après le redémarrage ou le chargement à nouveau du module btrfs, utiliser la commande **btrfs device scan** pour découvrir tous les systèmes de fichiers multi-périphériques. Voir [Section 3.4.2, « btrfs device scan pour périphériques multiples »](#) pour obtenir plus d'informations.

3.4.2. btrfs device scan pour périphériques multiples

Utilisez **btrfs device scan** pour scanner tous les périphériques blocs sous **/dev** et cherchez les volumes btrfs. Cela doit être effectué après le chargement du module btrfs si vous exécutez plus d'un périphérique dans un système de fichiers.

Pour scanner tous les périphériques, utiliser la commande suivante :

```
# btrfs device scan
```

Pour scanner un seul périphérique, utiliser la commande suivante :

```
# btrfs device scan /dev/device
```

3.4.3. Ajouter des nouveaux périphériques à un système de fichiers btrfs

Utiliser la commande **btrfs filesystem show** pour faire la liste de tous les systèmes de fichiers btrfs et des périphériques qu'ils incluent.

La commande **btrfs device add** est utilisée pour ajouter des nouveaux périphériques à un système de fichiers monté.

La commande **btrfs filesystem balance** équilibre (redistribue) les extensions allouées à tous les périphériques existants.

Voici un exemple de toutes ces commandes pour ajouter un nouveau périphérique :

Exemple 3.2. Ajouter un nouveau périphérique à un système de fichiers btrfs

Vous devez tout d'abord monter un système de fichiers btrfs. Voir [Section 3.1, « Création d'un système de fichiers btrfs »](#) pour plus d'informations sur la façon de créer un système de fichiers btrfs, et [Section 3.2, « Monter un système de fichiers btrfs »](#) pour obtenir plus d'informations sur la façon de monter un système de fichiers btrfs.

```
# mkfs.btrfs /dev/device1
# mount /dev/device1
```

Puis, ajouter un second périphérique au système de fichiers btrfs monté.

```
# btrfs device add /dev/device2 /mount-point
```

Les métadonnées et données stockées sur ces périphériques ne sont stockées que sur **/dev/device1**. Elles doivent maintenant être réparties équitablement entre les périphériques.

```
# btrfs filesystem balance /mount-point
```

Équilibrer un système de fichiers peut prendre un certain temps car il lit toutes les données et métadonnées du système de fichiers et les ré-écrit à travers le nouveau périphérique.

3.4.4. Convertir un système de fichiers btrfs

Pour convertir un système de fichiers non-raid en raid, ajouter un périphérique et exécutez un filtre d'équilibrage qui change le profil d'allocation du bloc.

Exemple 3.3. Convertir un système de fichiers btrfs

Pour convertir un système de périphérique unique existant, **/dev/sdb1** dans un tel cas, et en deux périphériques, un système raid 1 pour se protéger des défaillances du disque, utiliser la commande suivante :

```
# mount /dev/sdb1 /mnt
# btrfs device add /dev/sdc1 /mnt
# btrfs balance start -dconvert=raid1 -mconvert=raid1 /mnt
```



Important

Si les métadonnées ne sont pas converties à partir d'un seul périphérique par défaut, il demeure DUP. Cela ne garantit pas que les copies du bloc sont sur des périphériques séparés. Si les données ne sont pas converties, il n'aura pas de copies obsolètes dutout.

3.4.5. Suppression des périphériques btrfs

Utiliser la commande **btrfs device delete** pour supprimer un périphérique en ligne. Redistribue toute extension utilisée par d'autres périphériques du système de fichiers pour une suppression en toute sécurité.

Exemple 3.4. Supprimer un périphérique sur un système de fichiers btrfs

Commencez par créer et monter quelques systèmes de fichiers btrfs.

```
# mkfs.btrfs /dev/sdb /dev/sdc /dev/sdd /dev/sde
# mount /dev/sdb /mnt
```

Ajouter certaines données au système de fichiers.

Finalement, supprimer le périphérique requis.

```
# btrfs device delete /dev/sdc /mnt
```

3.4.6. Remplacer les périphériques qui ont échoué sur le système de fichiers btrfs

[Section 3.4.5, « Suppression des périphériques btrfs »](#) peut être utilisé pour supprimer un périphérique ayant échoué, dans la mesure où le super bloc puisse être lu. Cependant, si un périphérique venait à manquer, ou si le super bloc était corrompu, le système de fichiers devra être monté en mode dégradé :

```
# mkfs.btrfs -m raid1 /dev/sdb /dev/sdc /dev/sdd /dev/sde

ssd is destroyed or removed, use -o degraded to force the mount
to ignore missing devices

# mount -o degraded /dev/sdb /mnt

'missing' is a special device name

# btrfs device delete missing /mnt
```

La commande **btrfs device delete missing** supprime le premier périphérique qui est décrit dans les métadonnées du système de fichiers, mais qui n'était pas présent quand le système de fichiers a été monté.



Important

Il est impossible d'aller en dessous du nombre minimum de périphériques requis dans la structure de raids spécifique, même en en incluant celui qui manque. Il faut sans doute ajouter un nouveau périphérique pour pouvoir supprimer celui qui a échoué.

Par exemple, pour une structure de raid1 avec deux périphériques, si un périphérique échoue, il lui faudra :

1. monter en mode dégradé,
2. ajouter un nouveau périphérique,
3. et, supprimer le périphérique manquant.

3.4.7. Enregistrer un système de fichiers btrfs dans /etc/fstab

Si vous n'avez pas un **initrd** ou si le périphérique **btrfs** ne peut avoir lieu, il est possible de monter un système de fichiers **btrfs** multi volumes en passant tous les périphériques du système de fichiers explicitement à la commande **mount**.

Exemple 3.5. Exemple de saisie `/etc/fstab`

Exemple de saisie `/etc/fstab` qui conviendrait :

```
/dev/sdb /mnt btrfs
device=/dev/sdb,device=/dev/sdc,device=/dev/sdd,device=/dev/sde 0
```

Notez qu'en utilisant des UUID (Universally Unique Identifiers), ce sera plus stable qu'avec des chemins de périphériques.

3.5. Optimisation SSD

Utiliser le système de fichiers **btrfs** peut optimiser SSD. Cela peut être fait de deux façons différentes.

La première façon de procéder est d'avoir **mkfs.btrfs** qui ferme la duplication des métadonnées sur un seul périphérique quand `/sys/block/device/queue/rotational` est sur zéro pour le périphérique indiqué. Cela revient à spécifier **-m single** sur la ligne de commande. Cela peut être remplacé et la duplication des métadonnées peut être forcée grâce à l'option **-m dup**. La duplication n'est pas requise car SSD firmware peut perdre les deux copies. C'est une perte d'espace et un coût rajouté à la performance.

La deuxième façon est par un groupe d'options de montage : **ssd**, **nossd**, et **ssd_spread**.

L'option **ssd** fait plusieurs choses :

- ✦ Cela permet une allocation de cluster de métadonnées plus importante.
- ✦ Cela permet de distribuer des données de manière plus séquentielle lorsque cela est possible.
- ✦ Désactive la ré-écriture de feuille btree pour établir une correspondance avec la clé et l'ordre des blocs.
- ✦ Valide les fragments de journaux sans regrouper plusieurs processus.



Note

L'option de montage **ssd** n'active que l'option **ssd**. Utiliser l'option **nossd** pour la désactiver.

Certains SSD fonctionnent mieux quand on utilise souvent des numéros de blocs, tandis que d'autres fonctionnent mieux quand le clustering n'alloue des parties d'espace inutilisé uniquement. Par défaut, **mount -o ssd** cherchera des groupements de blocs quand il y a plusieurs blocs de libre qui auraient pu avoir des blocs distribués mélangés. La commande **mount -o ssd_spread** veille à ce qu'il n'y ait aucuns blocs distribués de mélangés. Cela améliore la performance à la base des SSD.



Note

L'option **ssd_spread** active les options **ssd** et **ssd_spread** à la fois. Utiliser **nossd** pour désactiver ces options.

L'option **ssd_spread** n'est jamais définie automatiquement si aucune de ces options SSD ne sont fournies et qu'aucun des périphériques est non-rotationnel.

Ces options devront toutes être testées avec votre build spécifique pour voir si leur utilisation améliore ou réduit la performance, car chaque combinaison de firmware SSD et de chargements d'application sont différents.

3.6. Références btrfs

La page man de **btrfs(8)** couvre toutes les commandes de gestion importantes. Inclut tout particulièrement :

- ✦ Toutes les commandes de sous-volumes pour gérer les clichés.
- ✦ Les commandes de **device** pour gérer les périphériques.
- ✦ Les commandes **scrub**, **balance**, et **defragment**.

La page man de **mkfs.btrfs(8)** contient des informations sur la façon de créer un système de fichiers btrfs qui inclut toutes les options à son sujet.

La page man de **btrfsck(8)** est pour les informations **fsck** à propos des systèmes btrfs.

Chapitre 4. Système de fichiers Ext3

Le système de fichiers ext3 est principalement une version améliorée du système de fichiers ext2. Ces améliorations offrent les avantages suivants :

Disponibilité

Après une panne d'alimentation ou une panne du système inattendue (aussi appelé un *arrêt du système incorrect*), la cohérence de chaque système de fichiers ext2 monté sur la machine doit être vérifiée par le programme **e2fsck**. Ce long processus peut retarder le démarrage du système de manière importante, particulièrement pour les volumes de grande taille contenant de nombreux fichiers. Pendant ce délai, on ne peut pas accéder aux données sur les volumes.

Il est possible d'exécuter **fsck -n** sur un système de fichiers en direct. Cependant, aucun changement ne sera effectué et des résultats trompeurs pourraient être retournés s'il y a des métadonnées partiellement écrites.

Si LVM est utilisé dans la pile, une autre option consiste à prendre un cliché LVM du système de fichiers et d'exécuter **fsck** dessus à la place.

Finalement, il existe une option pour remonter le système de fichiers en lecture seule. Toutes les mises à jour (et écritures) de métadonnées en attente sont ensuite forcées sur le disque avant qu'il soit remonté. Ceci permet d'assurer que le système de fichiers se trouve dans un état cohérent, à condition qu'il n'y ait pas de corruption précédente. Il est désormais possible d'exécuter **fsck -n**.

La journalisation offerte par le système de fichiers ext3 signifie que ce type de vérification de système de fichiers n'est plus nécessaire après un arrêt du système incorrect. La seule fois qu'une vérification de cohérence se produit en utilisant ext3 est dans certains rares cas de panne de matériel, comme lors de pannes de disque dur. Le temps pris pour récupérer un système de fichiers ext3 après un arrêt de système incorrect ne dépend pas de la taille du système de fichiers ou du nombre de fichiers, il dépend de la taille du *journal* utilisé pour maintenir une certaine cohérence. La taille de journal par défaut prend environ une seconde pour reprendre, en fonction de la vitesse du matériel.



Note

Le seul mode de journalisation d'ext3 pris en charge par Red Hat est **data=ordered** (par défaut).

Intégrité des données

Le système de fichiers ext3 empêche la perte d'intégrité des données dans le cas où un arrêt du système incorrect se produirait. Le système de fichiers ext3 vous permet de choisir le type et le niveau de protection pour vos données. Quant à l'état du système de fichiers, les volumes ext3 sont configurés de manière à fournir un haut niveau de cohérence des données par défaut.

Vitesse

Malgré l'écriture de certaines données plus d'une fois, ext3 possède un plus haut débit que ext2 dans la plupart des cas car la journalisation d'ext3 optimise les mouvements de disque dur. Vous pouvez choisir parmi trois modes de journalisation pour optimiser la vitesse, mais ce faire entraînera des compromis au niveau de l'intégrité des données si jamais le système devait tomber en panne.



Note

Le seul mode de journalisation d'ext3 pris en charge par Red Hat est **data=ordered** (par défaut).

Transition facile

Il est facile de migrer d'ext2 à ext3 et de tirer profit des bénéfices d'un système de fichiers offrant une journalisation robuste sans reformatage. Veuillez consulter la [Section 4.2, « Conversion vers un système de fichiers ext3 »](#) pour obtenir des informations supplémentaire sur la manière d'accomplir cette tâche.



Note

Red Hat Enterprise Linux 7 procure un pilote extN unifié. Il le fait en désactivant les configurations d'ext2 et ext3, et utilise à la place **ext4.ko** pour ces formats sur disque. Cela signifie que les messages de noyau se référeront toujours à ext4 quel que soit le système de fichiers utilisé.

Les sections suivantes couvrent la création et le réglage des partitions ext3. Pour les partitions ext2, ignorez les sections de partitionnement et de formatage et rendez vous directement sur la [Section 4.2, « Conversion vers un système de fichiers ext3 »](#).

4.1. Créer un système de fichiers ext3

Après l'installation, il est parfois nécessaire de créer un nouveau système de fichiers ext3. Par exemple, si un nouveau disque dur est ajouté au système, vous pourriez souhaiter partitionner le disque et utiliser le système de fichiers ext3.

Les étapes pour créer un système de fichiers ext3 sont comme suit :

Procédure 4.1. Créer un système de fichiers ext3

1. Formater la partition ou le volume LVM avec le système de fichiers ext3 en utilisant **mkfs**.
2. Étiqueter le système de fichiers en utilisant **e2label**.

Il est également possible d'ajouter un UUID spécifique au système de fichiers. Ainsi, pour ajouter l'UUID 7cd65de3-e0be-41d9-b66d-96d749c02da7 au système de fichiers **/dev/sda8**, exécuter les commandes suivantes :

```
# mkfs -t ext3 -U 7cd65de3-e0be-41d9-b66d-96d749c02da7 /dev/sda8
# tune2fs -U 7cd65de3-e0be-41d9-b66d-96d749c02da7 /dev/sda8
```

Remplacer ext3 avec le type de système de fichiers que vous utilisez (ext4, par exemple), suivi de l'option -U avec votre UUID, puis remplacer /dev/sda8 par le système de fichiers pour y ajouter l'UUID.

4.2. Conversion vers un système de fichiers ext3

La commande **tune2fs** convertit un système de fichiers **ext2** en **ext3**.



Note

Pour effectuer une conversion à partir d'ext2 vers ext3, veuillez toujours utiliser l'utilitaire **e2fsck** afin de vérifier votre système de fichiers avant et après avoir utilisé **tune2fs**. Avant d'essayer de convertir ext2 en ext3, veuillez effectuer des copies de sauvegarde de tous les systèmes de fichiers au cas où une erreur se produirait.

En outre, Red Hat recommande de créer un nouveau système de fichiers ext3 et d'y migrer les données, plutôt que de convertir d'ext2 à ext3 lorsque possible.

Pour convertir un système de fichiers **ext2** en système de fichiers **ext3**, connectez-vous en tant que root et saisissez la commande suivante dans une fenêtre de terminal :

```
# tune2fs -j block_device
```

block_device contient le système de fichiers à convertir.

Exécuter la commande **df** pour afficher les systèmes de fichiers montés.

4.3. Rétablir un système de fichiers Ext2

Pour rétablir un système de fichiers ext2, veuillez utiliser la procédure suivante.

Pour plus de simplicité, les commandes exemples de cette section utilisent la valeur suivante pour le périphérique bloc :

```
/dev/mapper/Vo1Group00-LogVo102
```

Procédure 4.2. Rétablir depuis ext3 vers ext2

1. Démonter la partition en vous connectant en tant que root et en saisissant :

```
# umount /dev/mapper/Vo1Group00-LogVo102
```

2. Modifier le type du système de fichiers en ext2 en saisissant la commande suivante :

```
# tune2fs -0 ^has_journal /dev/mapper/Vo1Group00-LogVo102
```

3. Vérifier si la partition contient des erreurs en saisissant la commande suivante :

```
# e2fsck -y /dev/mapper/Vo1Group00-LogVo102
```

4. Puis monter la partition à nouveau en tant que système de fichiers ext2 en saisissant :

```
# mount -t ext2 /dev/mapper/Vo1Group00-LogVo102 /mount/point
```

Dans la commande ci-dessus, remplacer */mount/point* par le point de montage de la partition.

**Note**

Si un fichier **.journal** existe au niveau root de la partition, supprimez-le.

Pour modifier la partition sur ext2 de manière permanente, rappelez-vous de bien mettre à jour le fichier **/etc/fstab**, sinon l'autre type de système de fichiers sera rétabli après un démarrage.

Chapitre 5. Le système de fichiers Ext4

Le système de fichiers ext4 est une extension évolutive du système de fichiers ext3. Dans Red Hat Enterprise Linux 7, il peut prendre en charge des fichiers et systèmes de fichiers d'une taille pouvant aller jusqu'à 16 téraoctets, et, à la différence de Red Hat Enterprise Linux 6 qui ne pouvait prendre en charge que des fichiers pouvant aller jusqu'à 16 téraoctets. Il prend également en charge un nombre illimité de sous-répertoires (le système de fichiers ext3 prend en charge un nombre maximum de 32 000), mais une fois que le nombre de liens dépasse 65 000, ce nombre se réinitialise sur 1 et n'augmente plus.



Note

Comme avec ext3, un volume ext4 doit être démonté pour effectuer une commande **fsck**. Pour obtenir des informations supplémentaires, veuillez consulter le [Chapitre 4, Système de fichiers Ext3](#).

Fonctionnalités principales

Ext4 utilise des extensions (contrairement au schéma de mappage de blocs traditionnellement utilisé par ext2 et ext3), ce qui améliore les performances lors de l'utilisation de fichiers de grande taille, et réduit les en-têtes des métadonnées des fichiers de grande taille. En outre, ext4 étiquette également les groupes de blocs et les sections de tables d'inodes en conséquence, ce qui leur permet d'être ignorés pendant les vérifications de systèmes de fichiers. Ceci permet d'effectuer des vérifications de systèmes de fichiers plus rapides, qui deviendront de plus en plus avantageuses au fur et à mesure que la taille du système de fichiers augmente.

Fonctionnalités d'allocation

Le système de fichiers ext4 offre les schémas d'allocation suivants :

- ✦ La pré-allocation persistante
- ✦ L'allocation différée
- ✦ L'allocation multi-blocs
- ✦ L'allocation par entrelacement

À cause de l'allocation différée, et dû à d'autres optimisations des performances, le comportement d'ext4 lors de l'écriture sur disque est différent d'ext3. Avec ext4, lorsqu'un programme est écrit sur le système de fichiers, il n'est pas garanti que ce soit effectivement sur disque à moins que le programme n'exécute un appel **fsync()** après.

Par défaut, ext3 force automatiquement les fichiers récemment créés sur le disque de manière quasi immédiate, même sans **fsync()**. Ce comportement cache les bogues des programmes qui n'ont pas utilisé **fsync()** afin de s'assurer que les données écrites l'étaient sur disque.

Contrairement à cela, le système de fichiers ext4 attend souvent plusieurs secondes pour écrire les changements sur le disque, lui permettant de combiner et de réarranger les écritures pour offrir de meilleures performances qu'ext3.



Avertissement

Contrairement à ext3, le système de fichiers ext4 ne force pas les données sur disque lors des enregistrements de transactions. Ainsi, des écritures mises en mémoire tampon mettent plus longtemps pour être vidées sur le disque. Quant aux systèmes de fichiers, veuillez utiliser des appels d'intégrité de données, tels que **fsync()**, afin de vous assurer que les données soient effectivement écrites sur un stockage permanent.

Autres fonctionnalités Ext4

Le système de fichiers ext4 prend également en charge :

- ✦ Les *attributs étendus* (**xattr**) — Ceux-ci permettent au système d'associer plusieurs noms et paires de valeurs supplémentaires par fichier.
- ✦ *Journalisation de quotas* — Ceci permet d'éviter le besoin de longues vérifications de la cohérence des quotas après une panne.



Note

Le seul mode de journalisation pris en charge sur ext4 est **data=ordered** (par défaut).

- ✦ *Horodatage subsecond* — Ceci donne la deuxième décimale des secondes à l'horodatage.

5.1. Créer un système de fichiers Ext4

Pour créer un système de fichiers ext4, veuillez utiliser la commande **mkfs.ext4**. En général, les options par défaut sont optimales pour la plupart des scénarios d'utilisation :

```
# mkfs.ext4 /dev/device
```

Ci-dessous figure un exemple de la sortie de cette commande qui affiche la géométrie et les fonctionnalités du système de fichiers résultant :

Exemple 5.1. Sortie de la commande **mkfs.ext4**

```
~]# mkfs.ext4 /dev/sdb1
mke2fs 1.41.12 (17-May-2010)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
245280 inodes, 979456 blocks
48972 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=1006632960
30 block groups
32768 blocks per group, 32768 fragments per group
8176 inodes per group
```

```
Superblock backups stored on blocks:
 32768, 98304, 163840, 229376, 294912, 819200, 884736
```

```
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

Pour les périphériques blocs à entrelacement (par exemple les matrices RAID5), la géométrie de l'entrelacement peut être indiquée pendant la création du système de fichiers. L'utilisation d'une géométrie d'entrelacement correcte améliore fortement les performances d'un système de fichiers ext4.

Lors de la création de systèmes de fichiers sur des volumes LVM ou MD, **mkfs.ext4** choisit une géométrie optimale. Ceci peut également s'appliquer à certains types de matériel RAID qui exportent des informations sur la géométrie au système d'exploitation.

Pour spécifier la géométrie d'entrelacement, veuillez utiliser l'option **-E** de **mkfs.ext4** (c'est-à-dire les options de système de fichiers étendues) avec les sous-options suivantes :

stride=value

Spécifie la taille du bloc RAID.

stripe-width=value

Spécifie le nombre de disques de données dans un périphérique RAID, ou le nombre d'unités d'entrelacement dans l'entrelacement.

Pour ces deux sous-options, la valeur « **value** » doit être spécifiée en unités de bloc de système de fichiers. Par exemple, pour créer un système de fichiers avec un stride de 64k (c'est-à-dire 16 x 4096) sur un système de fichiers de blocs de 4k, veuillez utiliser la commande suivante :

```
# mkfs.ext4 -E stride=16,stripe-width=64 /dev/device
```

Pour obtenir davantage d'informations sur la création de systèmes de fichiers, veuillez consulter **man mkfs.ext4**.



Important

Il est possible d'utiliser **tune2fs** pour activer certaines fonctionnalités ext4 sur des systèmes de fichiers ext3, et d'utiliser le pilote ext4 pour monter un système de fichiers ext3. Cependant, ces actions *ne sont pas* prises en charge sur Red Hat Enterprise Linux 7 car elles n'ont pas été totalement testées. À cause de cela, Red Hat ne peut pas garantir de performance cohérente ou de comportement prévisible pour les systèmes de fichiers ext3 convertis ou montés de cette manière.

Il est également possible d'ajouter un UUID spécifique au système de fichiers. Voir [Section 4.1, « Créer un système de fichiers ext3 »](#) pour plus d'informations.

5.2. Monter un système de fichiers Ext4

Un système de fichiers ext4 peut être monté sans options supplémentaires. Par exemple :

```
# mount /dev/device /mount/point
```

Le système de fichier ext4 prend également en charge plusieurs options de montage pour influencer le comportement. Par exemple, le paramètre **acl** active les listes de contrôle d'accès (« ACL »), tandis que le paramètre **user_xattr** active les attributs étendus d'utilisateur. Pour activer les deux options, veuillez utiliser leurs paramètres respectifs avec **-o**, comme suit :

```
# mount -o acl,user_xattr /dev/device /mount/point
```

Pour l'ext3, l'option **data_err=abort** peut être utilisée pour abandonner le journal si une erreur a lieu dans les données d'un fichier.

```
# mount -o data_err=abort /dev/device /mount/point
```

L'utilitaire **tune2fs** permet également aux administrateurs de définir les options de montage par défaut dans le superbloc du système de fichiers. Pour obtenir des informations supplémentaires, veuillez consulter **man tune2fs**.

Barrières d'écriture

Par défaut, ext4 utilise des barrières d'écriture pour assurer l'intégrité du système de fichiers même si un périphérique avec des caches d'écriture activés devait subir une coupure d'alimentation. Pour les périphériques sans caches d'écriture ou avec caches d'écriture alimentés par piles, veuillez désactiver les barrières à l'aide de l'option **nobarrier**, comme suit :

```
# mount -o nobarrier /dev/device /mount/point
```

Pour obtenir des informations supplémentaires sur les barrières d'écriture, veuillez consulter [Chapitre 21, Barrières d'écriture](#).

Aperçu technologique de Direct Access (DAX)

À partir de Red Hat Enterprise Linux 7.3, **Direct Access** (DAX) fournit, en tant qu'aperçu technologique sur les systèmes de fichiers ext4 et XFS, un moyen pour une application de mapper directement la mémoire persistante dans son espace d'adresse. Pour utiliser DAX, un système doit posséder une certaine forme de mémoire persistante disponible, sous forme d'un ou de plusieurs modules Non-Volatile Dual In-line Memory Modules (NVDIMMs), et un système de fichiers supportant DAX doit être créé sur les NVDIMM(s). De plus, le système de fichiers doit être monté avec l'option de montage **dax**. Ensuite, un **mmap** de fichier de système de fichier monté-dax résulte en mappage direct de stockage dans l'espace d'adresse de l'application.

5.3. Redimensionner un système de fichiers Ext4

Avant d'agrandir un système de fichiers ext4, assurez-vous que la taille du périphérique bloc sous-jacent sera appropriée pour contenir le système de fichiers. Veuillez utiliser les méthodes de redimensionnement appropriées pour les périphériques bloc affectés.

Un système de fichiers ext4 peut être agrandi pendant son montage en utilisant la commande **resize2fs** :

```
# resize2fs /mount/device size
```

La commande **resize2fs** peut également réduire la taille d'un système de fichiers ext4 *non monté* :

```
# resize2fs /dev/device size
```


Lors du redimensionnement d'un système de fichiers ext4, l'utilitaire **resize2fs** lit la taille de bloc du système de fichiers en unités, à moins qu'un suffixe indiquant une unité particulière ne soit utilisé. Les suffixes suivants indiquent des unités particulières :

- » **s** — secteurs de 512 octets sectors
- » **K** — kilooctets
- » **M** — mégaoctets
- » **G** — gigaoctets



Note

Le paramètre de taille est optionnel (et souvent redondant) lors de son expansion. **resize2fs** s'étend automatiquement pour remplir tout l'espace disponible du conteneur, habituellement un volume ou une partition logique.

Pour obtenir des informations supplémentaires sur le redimensionnement d'un système de fichiers ext4, veuillez consulter `man resize2fs`.

5.4. Sauvegarde des systèmes de fichiers ext2/3/4

Procédure 5.1. Exemple de sauvegarde des systèmes de fichiers ext2/3/4

1. Toutes les données doivent être sauvegardées avant de tenter les opérations de restauration. Les sauvegardes de données doivent être effectuées régulièrement. En plus des données, il y a des informations de configuration qui doivent être sauvegardées, y compris `/etc/fstab` et la sortie de `fdisk -l`. Exécuter un `sosreport/sysreport` capturera cette information et est fortement conseillé.

```
# cat /etc/fstab
LABEL=/                /                ext3    defaults    1 1
LABEL=/boot1          /boot            ext3    defaults    1 2
LABEL=/data            /data            ext3    defaults    0 0
tmpfs                  /dev/shm         tmpfs   defaults    0 0
devpts                 /dev/pts         devpts  gid=5,mode=620 0 0
sysfs                  /sys             sysfs   defaults    0 0
proc                   /proc            proc    defaults    0 0
LABEL=SWAP-sda5        swap              swap    defaults    0 0
/dev/sda6               /backup-files    ext3    defaults    0 0

# fdisk -l
   Device Boot      Start   End  Blocks  Id System
/dev/sda1 *          1      13    104391  83  Linux
/dev/sda2             14     1925   15358140  83  Linux
/dev/sda3            1926    3200   10241437+  83  Linux
/dev/sda4            3201    4864   13366080   5   Extended
/dev/sda5            3201    3391   1534176   82  Linux swap /
Solaris
/dev/sda6            3392    4864   11831841  83  Linux
```

Dans cet exemple, nous allons utiliser la partition `/dev/sda6` pour sauvegarder des fichiers de sauvegarde, et nous assumons que `/dev/sda6` est monté sur le fichier `/backup-files`.

2. Si la partition sauvegardée est dans une partition de système d'exploitation, démarrez votre système en mode Single User. Cette étape n'est pas utile dans les cas de partitions de données normales.
3. Utiliser la commande **dump** pour sauvegarder le contenu des partitions :



Note

- ✦ Si le système est en cours d'exécution depuis un bon moment, il est conseillé d'exécuter **e2fsck** sur les partitions avant la sauvegarde.
- ✦ **dump** ne doit pas être utilisé sur un système de fichiers monté et à forte charge car des versions corrompues de fichiers pourraient être sauvegardées. Ce problème a été soulevé dans dump.sourceforge.net.



Important

Quand on sauvegarde des partitions de système d'exploitation, la partition doit être dé-montée.

Bien qu'il soit possible de sauvegarder une partition de données ordinaire montée, il vaut mieux la dé-monter si possible. Si la partition de données est montée, les résultats de la sauvegarde sont imprévisibles.

```
# dump -0uf /backup-files/sda1.dump /dev/sda1
# dump -0uf /backup-files/sda2.dump /dev/sda2
# dump -0uf /backup-files/sda3.dump /dev/sda3
```

Si vous souhaitez effectuer une sauvegarde à distance, vous pourrez utiliser ssh ou bien vous pourrez configurer une connexion sans mot de passe.



Note

Si vous utilisez une redirection standard, l'option « -f » devra être passée séparément.

```
# dump -0u -f - /dev/sda1 | ssh root@remoteserver.example.com dd
of=/tmp/sda1.dump
```

5.5. Restaurer un système de fichiers ext2/3/4

Procédure 5.2. Exemple de restauration d'un système de fichiers ext2/3/4

1. Si vous restaurez une partition de système d'exploitation, démarrez votre système en mode Rescue. Cette étape n'est pas requise pour les partitions de données ordinaires.
2. Reconstruire sda1/sda2/sda3/sda4/sda5 à l'aise de la commande **fdisk**.

**Note**

Si nécessaire, créer des partitions qui puissent contenir les systèmes de fichiers restaurés. Les nouvelles partitions doivent être suffisamment grandes pour pouvoir contenir les données restaurées. Il est important d'avoir les bons numéros de début et de fin ; ce sont les numéros de secteurs de début et de fin des partitions.

3. Formater les partitions de destination en utilisant la commande **mkfs**, comme montré ci-dessous.

**Important**

NE PAS formater **/dev/sda6** dans l'exemple ci-dessus car il sauvegarde les fichiers de sauvegarde.

```
# mkfs.ext3 /dev/sda1
# mkfs.ext3 /dev/sda2
# mkfs.ext3 /dev/sda3
```

4. Si vous créez des nouvelles partitions, renommez toutes les partitions pour qu'elles puissent correspondre au fichier **fstab**. Cette étape n'est pas utile si les partitions ne sont pas créées à nouveau.

```
# e2label /dev/sda1 /boot1
# e2label /dev/sda2 /
# e2label /dev/sda3 /data
# mkswap -L SWAP-sda5 /dev/sda5
```

5. Préparer les répertoires de travail.

```
# mkdir /mnt/sda1
# mount -t ext3 /dev/sda1 /mnt/sda1
# mkdir /mnt/sda2
# mount -t ext3 /dev/sda2 /mnt/sda2
# mkdir /mnt/sda3
# mount -t ext3 /dev/sda3 /mnt/sda3
# mkdir /backup-files
# mount -t ext3 /dev/sda6 /backup-files
```

6. Restaurer les données.

```
# cd /mnt/sda1
# restore -rf /backup-files/sda1.dump
# cd /mnt/sda2
# restore -rf /backup-files/sda2.dump
# cd /mnt/sda3
# restore -rf /backup-files/sda3.dump
```

Si vous souhaitez restaurer à partir d'un hôte éloigné ou d'un fichier de sauvegarde d'un hôte éloigné, utiliser ssh ou rsh. Vous devrez configurer une connexion sans mot de passe pour les exemples suivants :

Connectez-vous à 10.0.0.87, et restaurez sda1 à partir du fichier local sda1.dump :

```
# ssh 10.0.0.87 "cd /mnt/sda1 && cat /backup-files/sda1.dump | restore
-rf -"
```

Connectez-vous à 10.0.0.87, et restaurez sda1 à partir du fichier distant 10.66.0.124 sda1.dump :

```
# ssh 10.0.0.87 "cd /mnt/sda1 && RSH=/usr/bin/ssh restore -r -f
10.66.0.124:/tmp/sda1.dump"
```

7. Démarrez à nouveau.

5.6. Autres utilitaires du système de fichiers Ext4

Red Hat Enterprise Linux 7 offre également d'autres utilitaires pour gérer les systèmes de fichiers ext4 :

e2fsck

Utilisé pour réparer un système de fichiers ext4. Cet outil vérifie et répare un système de fichiers ext4 plus efficacement qu'ext3, grâce aux mises à jour apportées à la structure de disque ext4.

e2label

Change l'étiquette sur un système de fichiers ext4. Cet outil fonctionne également sur les systèmes de fichiers ext2 et ext3.

quota

Contrôle et effectue des rapports sur l'utilisation de l'espace disque (les blocs) et des fichiers (inodes) par les utilisateurs et les groupes sur un système de fichiers ext4. Pour obtenir des informations sur l'utilisation de **quota**, veuillez consulter **man quota** et la [Section 16.1, « Configurer les quotas de disques »](#).

fsfreeze

Pour suspendre l'accès à un système de fichiers, utiliser la commande # **fsfreeze -f mount-point** pour le geler, et # **fsfreeze -u mount-point** pour le dégeler. Cela stoppe l'accès au système de fichiers et crée une image stable sur disque.



Note

Il faut utiliser **fsfreeze** pour les pilotes de mappage de périphériques.

Pour plus d'informations, voir la page **man fsfreeze(8)**.

Comme démontré dans [Section 5.2, « Monter un système de fichiers Ext4 »](#), l'utilitaire **tune2fs** peut également ajuster des paramètres de systèmes de fichiers configurables pour les systèmes de fichiers ext2, ext3, et ext4. En outre, les outils suivants sont aussi utiles pour le débogage et l'analyse des systèmes de fichiers ext4 :

debugfs

Débogue les systèmes de fichiers ext2, ext3, ou ext4.

e2image

Enregistre les métadonnées critiques des systèmes de fichiers ext2, ext3, ou ext4 sur un fichier.

Pour obtenir des informations supplémentaires sur ces utilitaires, veuillez consulter leurs pages **man** respectives.

Chapitre 6. Le système de fichiers XFS

XFS est un système de fichiers hautement modulable et de haute performance, qui fut conçu à l'origine par Silicon Graphics, Inc. XFS est le système de fichiers par défaut de Red Hat Enterprise Linux 7.

Fonctionnalités principales

XFS prend en charge la *journalisation de métadonnées*, ce qui facilite une récupération après incident plus rapide. Le système de fichiers XFS peut aussi être défragmenté et élargi alors qu'il est monté et actif. En outre, Red Hat Enterprise Linux 7 prend en charge les utilitaires de sauvegarde et de restauration spécifiques à XFS.

Fonctionnalités d'allocation

XFS offre les schémas d'allocation suivants :

- ✦ Allocation basée sur extensions
- ✦ Politiques d'allocation par entrelacement
- ✦ L'allocation différée
- ✦ Pré-allocation de l'espace

L'allocation différée et les autres optimisations des performances affectent XFS de la même manière qu'ext4. Autrement dit, les écritures d'un programme sur un système de fichiers XFS ne garantissent pas d'être sur disque à moins que le programme n'effectue un appel **fsync()** par la suite.

Pour obtenir des informations supplémentaires sur les implications de l'allocation différée sur un système de fichiers (ext4 et XFS), veuillez consulter les *Fonctionnalités de l'allocation* dans le [Chapitre 5, Le système de fichiers Ext4](#).



Note

Créer ou étendre des fichiers échoue occasionnellement avec un échec d'écriture ENOSPC inattendu même si le disque semble insuffisant. Cela s'explique par le design de XFS qui est basé sur la performance. Dans la pratique, ce n'est pas un problème car cela ne survient que si l'espace restant n'est que dans quelques blocs.

Autres fonctionnalités XFS

Le système de fichiers XFS prend également en charge ce qui suit :

Attributs étendus (xattr)

Ceci permet au système d'associer plusieurs paires nom/valeur supplémentaires par fichiers. Activé par défaut.

Journalisation de quotas

Ceci permet d'éviter le besoin de longues vérifications de cohérence des quotas après une panne.

Quotas de projets/répertoires

Ceci permet les restrictions de quotas sur une arborescence de répertoires.

Horodatage subsecond

Ceci permet à l'horodatage de donner la deuxième décimale des secondes.

Le comportement par défaut `atime` est `relatime`

`Relatime` est pour XFS par défaut. N'a presque pas de temps de latence par rapport à `noatime` tout en conservant des valeurs `atime` saines.

6.1. Créer un système de fichiers XFS

Pour créer un système de fichiers XFS, veuillez utiliser la commande `mkfs.xfs /dev/device`. En général, les options par défaut sont optimales pour un usage commun.

Lors de l'utilisation de `mkfs.xfs` sur un périphérique bloc contenant un système de fichiers, veuillez utiliser l'option `-f` pour forcer le remplacement de ce système de fichiers.

Exemple 6.1. Sortie de la commande `mkfs.xfs`

Ci-dessous figure un exemple de sortie de la commande `mkfs.xfs` :

meta-data=/dev/device	isize=256	agcount=4, agsize=3277258
blks		
=	sectsz=512	attr=2
data	=	bsize=4096
=	blocks=13109032, imaxpct=25	
naming	=version 2	sunit=0
log	=internal log	bsize=4096
=	blocks=6400, version=2	width=0 blks
realtime	=none	sectsz=512
	extsz=4096	sunit=0 blks, lazy-count=1
		blocks=0, rtextents=0



Note

Une fois qu'un système de fichiers XFS a été créé, sa taille ne peut pas être réduite. Cependant, il peut toujours être agrandi par la commande `xfs_growfs` (veuillez consulter [Section 6.4, « Augmenter la taille d'un système de fichiers XFS »](#)).

Pour les périphériques blocs à entrelacement (par exemple les matrices RAID5), la géométrie de l'entrelacement peut être indiquée pendant la création du système de fichiers. L'utilisation d'une géométrie d'entrelacement correcte améliore fortement les performances d'un système de fichiers XFS.

Lors de la création de systèmes de fichiers sur des volumes LVM ou MD, `mkfs.xfs` choisit une géométrie optimale. Ceci peut également s'appliquer à certains types de matériel RAID qui exportent des informations sur la géométrie au système d'exploitation.

Si le périphérique exporte des informations de géométrie de tranches, `mkfs` (pour `ext3`, `ext4`, et `xfs`) utilisera cette géométrie automatiquement. Si la géométrie de tranches n'est pas détectée, il sera possible de la spécifier automatiquement à un moment `mkfs` par les options suivantes :

`su=value`

Spécifie une unité d'entrelacement ou une taille de morceau RAID. La valeur **value** doit être indiquée en octets, avec un suffixe optionnel **k**, **m**, ou **g**.

sw=value

Spécifie le nombre de disques de données dans un périphérique RAID, ou le nombre d'unités d'entrelacement dans l'entrelacement.

L'exemple suivant indique une taille de morceau de 64k sur un périphérique RAID contenant 4 unités d'entrelacement :

```
# mkfs.xfs -d su=64k,sw=4 /dev/device
```

Pour obtenir davantage d'informations sur la création de systèmes de fichiers XFS, veuillez consulter **man mkfs.xfs** et *Red Hat Enterprise Linux Performance Tuning Guide*, chapitre *Basic Tuning for XFS*.

6.2. Monter un système de fichiers XFS

Un système de fichiers XFS peut être monté sans options supplémentaires. Par exemple :

```
# mount /dev/device /mount/point
```

Pour Red Hat Enterprise Linux 7, la valeur par défaut est `inode64`.



Note

À la différence de `mke2fs`, `mkfs.xfs` n'utilise pas de fichier de configuration ; tout est spécifié au niveau de la ligne de commande.

Barrières d'écriture

Par défaut, XFS utilise des barrières d'écriture pour assurer l'intégrité du système de fichiers même si un périphérique avec des caches d'écriture activés devait subir une coupure d'alimentation. Pour les périphériques sans caches d'écriture ou avec des caches d'écriture alimentés par piles, veuillez désactiver les barrières à l'aide de l'option **nobarrier** :

```
# mount -o nobarrier /dev/device /mount/point
```

Pour obtenir des informations supplémentaires sur les barrières d'écriture, veuillez consulter [Chapitre 21, Barrières d'écriture](#).

Aperçu technologique de Direct Access (DAX)

À partir de Red Hat Enterprise Linux 7.3, **Direct Access** (DAX) fournit, en tant qu'aperçu technologique sur les systèmes de fichiers ext4 et XFS, un moyen pour une application de mapper directement la mémoire persistante dans son espace d'adresse. Pour utiliser DAX, un système doit posséder une certaine forme de mémoire persistante disponible, sous forme d'un ou de plusieurs modules Non-Volatile Dual In-line Memory Modules (NVDIMMs), et un système de fichiers supportant DAX doit être créé sur les NVDIMM(s). De plus, le système de fichiers doit être monté avec l'option de montage **dax**. Ensuite, un **mmap** de fichier de système de fichier monté-dax résulte en mappage direct de stockage dans l'espace d'adresse de l'application.

6.3. Gestion des quotas XFS

Le sous-système de quotas XFS gère les limites de l'espace disque (blocs) et l'usage des fichiers (inodes). Les quotas XFS contrôlent ou rapportent l'usage de ces éléments sur un niveau d'utilisateur, de groupe, de répertoire ou de projet. Veuillez également remarquer que même si les quotas utilisateur, groupe, et répertoire ou projets sont activés indépendamment, les quotas de groupe et de projet sont mutuellement exclusifs.

Lorsque la gestion est effectuée sur une base par répertoire ou par projet, XFS gère l'usage disque des hiérarchies de répertoires associées à un projet particulier. Ce faisant, XFS reconnaît qu'il existe des limites de « groupe » trans-organisationnelles entre projets. Ceci offre un niveau de contrôle plus important que ce qui est possible lors de la gestion des quotas d'utilisateurs ou de groupes.

Les quotas XFS sont activés lors du montage, avec des options de montage spécifiques. Chaque option de montage peut être spécifiée **noenforce** ; ceci permettra d'effectuer des rapport d'utilisation sans appliquer de limite. Les options de montage de quotas valides incluent :

- ✦ **uquota/uqnoenforce** - Quotas d'utilisateurs
- ✦ **gquota/gqnoenforce** - Quotas de groupes
- ✦ **pquota/pqnoenforce** - Quotas de projets

Une fois les quotas activés, l'outil **xfs_quota** peut être utilisé pour définir les limites et effectuer un rapport sur l'utilisation du disque. Par défaut, **xfs_quota** est exécuté de manière interactive et dans le *mode de base*. Les sous-commandes du mode de base rapportent simplement l'usage et sont disponibles à tous les utilisateurs. Les sous-commandes **xfs_quota** de base incluent :

quota username/userID

Afficher l'utilisation et les limites pour le nom d'utilisateur **username** donné ou l'ID numérique **userID** donné

df

Afficher le compte des blocs et inodes disponibles et utilisés

En revanche, **xfs_quota** possède également un *mode expert*. Les sous-commandes de ce mode permettent la configuration des limites, et sont uniquement disponibles aux utilisateurs possédant des privilèges élevés. Pour utiliser les sous-commandes du mode expert de manière interactive, veuillez exécuter **xfs_quota -x**. Les sous-commandes du mode expert incluent :

report /path

Rapport des informations sur les quotas d'un système de fichiers particulier

limit

Modifier les limites de quota.

Pour une liste complète des sous-commandes du mode de base ou du mode expert, veuillez utiliser la sous-commande **help**.

Toutes les sous-commandes peuvent également être exécutées directement à partir de la ligne de commande en utilisant l'option **-c**, avec **-x** pour les sous-commandes du mode expert.

Exemple 6.2. Afficher un exemple de rapport de quotas

Par exemple, pour afficher un exemple de rapport de quotas pour `/home` (sur `/dev/blockdevice`), veuillez utiliser la commande `xfs_quota -x -c 'report -h' /home`. Ceci affichera une sortie similaire à la suivante :

```
User quota on /home (/dev/blockdevice)
                Blocks
User ID      Used  Soft  Hard Warn/Grace
-----
root         0      0     0  00 [-----]
testuser    103.4G    0     0  00 [-----]
...
```

Pour paramétrer une limite dure (« hard ») et une limite douce (« soft ») du compte d'inode, respectivement de 500 et 700 pour l'utilisateur `john` (dont le répertoire d'accueil est `/home/john`), veuillez utiliser la commande suivante :

```
# xfs_quota -x -c 'limit isoft=500 ihard=700 john' /home/
```

Dans ce cas, passez l'option `mount_point` qui correspond au système de fichiers xfs monté.

Par défaut, la sous-commande `limit` reconnaît les cibles en tant qu'utilisateurs. Lors de la configuration des limites d'un groupe, veuillez utiliser l'option `-g` (comme dans l'exemple précédent). De la même manière, veuillez utiliser `-p` pour les projets.

Les limites de blocs dures (« hard ») et douces (« soft ») peuvent également être configurées à l'aide de `bsoft` ou `bhard` au lieu de `isoft` ou `ihard`.

Exemple 6.3. Définir une limite douce (« soft ») et une limite dure (« hard »)

Par exemple, pour définir une limite douce (« soft ») et une limite dure (« hard ») avec des valeurs respectives de 1000m et 1200m pour grouper `accounting` sur le système de fichiers `/target/path`, veuillez utiliser la commande suivante :

```
# xfs_quota -x -c 'limit -g bsoft=1000m bhard=1200m accounting'
/target/path
```



Note

Les commandes `bsoft` et `bhard` comptent par octets.



Important

Même si les blocs en temps réel (`rtbhard/rtbsoft`) sont décrits dans `man xfs_quota` comme étant des unités valides lors du paramétrage de quotas, le sous-volume en temps réel n'est pas activé dans cette version. Ainsi, les options `rtbhard` et `rtbsoft` ne sont pas applicables.

Paramétrer des limites de projets

Avant de configurer les limites des répertoires contrôlés par projets, veuillez les ajouter à `/etc/projects`. Les noms de projets peuvent être ajoutés à `/etc/projectid` pour lier les ID de projets aux noms de projets. Une fois qu'un projet est ajouté à `/etc/projects`, veuillez initialiser son répertoire de projet en utilisant la commande suivante :

```
# xfs_quota -x -c 'project -s projectname project_path
```

Les quotas pour les projets avec des répertoires initialisés peuvent ensuite être configurés avec :

```
# xfs_quota -x -c 'limit -p bsoft=1000m bhard=1200m projectname'
```

Les outils génériques de configuration de quotas (par exemple `quota`, `repquota`, et `edquota`) peuvent également être utilisés pour manipuler les quotas XF. Cependant, ces outils ne peuvent pas être utilisés avec les quotas des projets XFS.



Important

Red Hat recommande fortement d'utiliser l'option `xfs_quota` sur tous les autres outils disponibles.

Pour obtenir davantage d'informations sur le paramétrage de quotas XFS, veuillez consulter `man xfs_quota`, `man projid(5)`, et `man projects(5)`.

6.4. Augmenter la taille d'un système de fichiers XFS

Un système de fichiers XFS peut être agrandi pendant son montage en utilisant la commande `xfs_growfs` :

```
# xfs_growfs /mount/point -D size
```

L'option `-D size` permet d'augmenter la taille du système de fichiers à la taille `size` spécifiée (exprimée en nombre de blocs de système de fichier). Sans l'option `-D size`, `xfs_growfs` augmentera la taille du système de fichiers à la taille maximum prise en charge par le périphérique.

Avant d'augmenter la taille d'un système de fichiers XFS avec `-D size`, assurez-vous que la taille du périphérique bloc sous-jacent sera appropriée pour contenir le système de fichiers. Veuillez utiliser les méthodes correctes de redimensionnement pour les périphériques bloc affectés.



Note

Même si les systèmes de fichiers XFS peuvent être agrandis pendant leur montage, leur taille ne peut pas être réduite.

Pour obtenir des informations supplémentaires sur l'agrandissement d'un système de fichiers, veuillez consulter `man xfs_growfs`.

6.5. Réparer un système de fichiers XFS

Pour réparer un système de fichiers XFS, veuillez utiliser `xfs_repair`:

```
# xfs_repair /dev/device
```

L'utilitaire **xfs_repair** est hautement évolutif et a également été conçu pour réparer des systèmes de fichiers de très grande taille avec de nombreux inodes de manière efficace. Contrairement aux autres systèmes de fichiers Linux, **xfs_repair** n'est pas exécuté lors du démarrage, même lorsqu'un système de fichiers XFS n'a pas été monté correctement. En cas de démontage incorrect, **xfs_repair** rediffuse simplement le journal pendant le montage, s'assurant ainsi d'un système de fichiers cohérent.



Avertissement

L'utilitaire **xfs_repair** ne peut pas réparer un système de fichiers XFS avec un journal endommagé. Pour supprimer le journal, montez et démontez le système de fichiers. Si le journal est corrompu et qu'il ne peut pas être réutilisé, veuillez utiliser l'option **-L** (« forcer la mise à zéro du journal ») pour supprimer le journal, c'est-à-dire **xfs_repair -L /dev/device**. Prenez note que cette opération peut provoquer une corruption ou des pertes de données supplémentaires.

Pour obtenir des informations supplémentaires sur la réparation de systèmes de fichiers XFS, veuillez consulter **man xfs_repair**.

6.6. Suspendre un système de fichier XFS

Pour suspendre ou reprendre une activité d'écriture sur un système de fichiers, veuillez utiliser **xfs_freeze**. La suspension d'activités d'écriture permet l'utilisation des clichés de périphériques basés matériel pour capturer le système de fichiers dans un état cohérent.



Note

L'utilitaire **xfs_freeze** est fourni par le paquet **xfsprogs** uniquement disponible sur x86_64.

Pour suspendre (c'est-à-dire geler) un système de fichiers XFS, veuillez exécuter :

```
# xfs_freeze -f /mount/point
```

Pour dégeler un système de fichiers XFS, veuillez exécuter :

```
# xfs_freeze -u /mount/point
```

Lors de la capture d'un cliché LVM, il n'est pas nécessaire d'utiliser **xfs_freeze** pour suspendre le système de fichiers avant tout. Au contraire, les outils de gestion LVM suspendront automatiquement le système de fichiers XFS avant de prendre le cliché.

Pour obtenir des informations supplémentaires sur le gel et dégel de systèmes de fichiers XFS, veuillez consulter **man xfs_freeze**.

6.7. Sauvegarde et restauration des systèmes de fichiers XFS

Les sauvegardes et restaurations de systèmes de fichiers XFS impliquent deux utilitaires : **xfsdump** et **xfsrestore**.

Pour effectuer une copie de sauvegarde ou le vidage d'un système de fichiers XFS, veuillez utiliser l'utilitaire **xfsdump**. Red Hat Enterprise Linux 7 prend en charge les sauvegardes sur lecteurs de bande ou images fichiers normales, et permet également d'écrire plusieurs vidages sur le même lecteur. L'utilitaire **xfsdump** permet aussi à un vidage de s'étendre sur plusieurs lecteurs, même si un vidage peut être écrit sur un fichier normal. En outre, **xfsdump** prend en charge les sauvegardes incrémentales, et peut exclure des fichiers d'une sauvegarde en utilisant la taille, une sous-arborescence, ou des indicateurs d'inodes pour les filtrer.

Afin de pouvoir prendre en charge les copies de sauvegardes incrémentales, **xfsdump** utilise des *niveaux de vidage* pour déterminer un vidage de base auquel un vidage particulier est relatif. L'option **-l** spécifie un niveau de vidage (0-9). Pour effectuer une copie de sauvegarde complète, veuillez effectuer un vidage de *niveau 0* sur le système de fichiers (c'est-à-dire */path/to/filesystem*), comme suit :

```
# xfsdump -l 0 -f /dev/device /path/to/filesystem
```



Note

L'option **-f** spécifie une destination pour la sauvegarde. Par exemple, la destination **/dev/st0** est normalement utilisée pour les lecteurs de bande. Une destination **xfsdump** peut être un lecteur de bande, un fichier normal, ou un périphérique de bande distant.

D'autre part, une sauvegarde incrémentale videra uniquement les fichiers qui ont changé depuis le dernier vidage de *niveau 0*. Un vidage de *niveau 1* est le premier vidage incrémental après un vidage complet ; le vidage incrémental suivant sera de *niveau 2*, et ainsi de suite jusqu'au maximum, le *niveau 9*. Ainsi, pour effectuer un vidage de *niveau 1* sur un lecteur de bande :

```
# xfsdump -l 1 -f /dev/st0 /path/to/filesystem
```

L'utilitaire **xfsrestore** restaure les systèmes de fichiers depuis les vidages produits par **xfsdump**. L'utilitaire **xfsrestore** possède deux modes : un mode par défaut *simple*, et un mode *cumulatif*. Les vidages spécifiques sont identifiés par *ID de session* ou par *étiquette de session*. Ainsi, restaurer un vidage requiert son ID ou étiquette de session correspondant. Pour afficher les ID et étiquettes de session de tous les vidages (complets et incrémentaux), veuillez utiliser l'option **-I** :

```
# xfsrestore -I
```

La sortie résultante devrait être similaire à ceci :

Exemple 6.4. ID et étiquettes de session de tous les vidages

```
file system 0:
fs id: 45e9af35-efd2-4244-87bc-4762e476cbab
session 0:
mount point: bear-05:/mnt/test
device: bear-05:/dev/sdb2
time: Fri Feb 26 16:55:21 2010
session label: "my_dump_session_label"
session id: b74a3586-e52e-4a4a-8775-c3334fa8ea2c
level: 0
resumed: NO
subtree: NO
streams: 1
```

```

stream 0:
  pathname: /mnt/test2/backup
  start:   ino 0 offset 0
  end:     ino 1 offset 0
  interrupted: NO
  media files: 1
  media file 0:
    mfile index: 0
    mfile type: data
    mfile size: 21016
    mfile start: ino 0 offset 0
    mfile end:   ino 1 offset 0
    media label: "my_dump_media_label"
    media id: 4a518062-2a8f-4f17-81fd-bb1eb2e3cb4f
xfsrestore: Restore Status: SUCCESS

```

Mode simple de xfsrestore

Le mode *simple* permet aux utilisateur de restaurer un système de fichiers entier à partir d'un vidage de *niveau 0*. Après avoir identifié l'ID de session d'un vidage de *niveau 0* (c'est-à-dire **session-ID**), veuillez le restaurer complètement sur **/path/to/destination** en utilisant :

```
# xfsrestore -f /dev/st0 -S session-ID /path/to/destination
```



Note

L'option **-f** spécifie l'emplacement du vidage, tandis que l'option **-S** ou **-L** indique le vidage particulier à restaurer. L'option **-S** est utilisée pour spécifier un ID de session, tandis que l'option **-L** est utilisée pour les étiquettes de session. L'option **-I** affiche les étiquettes et ID de session de chaque vidage.

Mode cumulatif de xfsrestore

Le mode *cumulatif* de **xfsrestore** permet la restauration de systèmes de fichier à partir d'une sauvegarde incrémentale particulière, par exemple, du *niveau 1* au *niveau 9*. Pour restaurer un système de fichiers à partir d'une sauvegarde incrémentale, veuillez simplement ajouter l'option **-r** :

```
# xfsrestore -f /dev/st0 -S session-ID -r /path/to/destination
```

Opération interactive

L'utilitaire **xfsrestore** permet également à des fichiers particuliers d'un vidage d'être extraits, ajoutés ou supprimés. Pour utiliser **xfsrestore** de manière interactive, veuillez utiliser l'option **-i**, comme suit :

```
xfsrestore -f /dev/st0 -i /destination/directory
```

La boîte de dialogue interactive démarrera une fois que **xfsrestore** termine de lire le périphérique spécifié. Les commandes de cette boîte de dialogue incluent **cd**, **ls**, **add**, **delete**, et **extract** ; pour obtenir une liste complète des commandes, veuillez utiliser **help**.

Pour obtenir des informations supplémentaires sur le vidage et la restauration de systèmes de fichiers XFS, veuillez consulter **man xfsdump** et **man xfsrestore**.

6.8. Autres utilitaires des systèmes de fichiers XFS

Red Hat Enterprise Linux 7 offre également d'autres utilitaires pour gérer les systèmes de fichiers XFS :

xfs_fsr

Utilisé pour défragmenter les systèmes de fichiers XFS montés. Lorsqu'invoqué sans arguments, **xfs_fsr** défragmente tous les fichiers normaux dans tous les systèmes de fichiers XFS montés. Cet utilitaire permet également aux utilisateurs de suspendre une défragmentation à une heure spécifiée et de la reprendre au même endroit ultérieurement.

En outre, **xfs_fsr** permet également la défragmentation d'un seul fichier, comme dans **xfs_fsr /path/to/file**. Red Hat recommande d'éviter de défragmenter un système de fichiers entier par défaut. La défragmentation du système dans son entier pourrait entraîner un effet secondaire de fragmentation d'espace libre.

xfs_bmap

Imprime la carte des blocs de disque utilisés par les fichiers dans un système de fichiers XFS. Cette carte répertorie chaque extension utilisés par un fichier spécifié, ainsi que les régions du fichiers n'offrant pas de bloc correspondant (c'est-à-dire, des trous).

xfs_info

Imprime les informations du système de fichiers XFS.

xfs_admin

Modifie les paramètres d'un système de fichiers XFS. L'utilitaire **xfs_admin** peut uniquement modifier les paramètres de périphériques ou systèmes de fichiers non montés.

xfs_copy

Copie la totalité du contenu d'un système de fichiers XFS entier sur une ou plusieurs cibles en parallèle.

Les utilitaires suivants sont également utiles pour déboguer et analyser les systèmes de fichiers XFS :

xfs_metadump

Copie les métadonnées du système de fichiers XFS sur un fichier. Red Hat permet d'utiliser **xfs_metadump** pour copier les systèmes de fichiers non montés, en lecture seule, ou gelés ou suspendus ; sinon les vidages générés pourraient être corrompus ou incohérents.

xfs_mdrestore

Restaure une image metadump XFS (générée avec **xfs_metadump**) sur une image de système de fichiers.

xfs_db

Débogue un système de fichiers XFS.

Pour obtenir des informations supplémentaires sur ces utilitaires, veuillez consulter leurs pages **man** respectives.

6.9. Migration de ext4 à XFS

À partir de Red Hat Enterprise Linux 7.0, XFS sera le système de fichiers par défaut et non plus ext4. Cette section pointe les différences entre utiliser et administrer un système de fichiers XFS.

Le système de fichiers ext4 est toujours pris en charge par Red Hat Enterprise Linux 7 et peut être sélectionné au moment de l'installation. Non requis quand on peut migrer d'ext4 à XFS.

6.9.1. Commandes utilisées avec ext3 et ext4 comparé à XFS

Le tableau suivant dresse une comparaison des commandes utilisées avec ext3 et ext4 par rapport à leurs équivalents spécifiques à XFS.

Tableau 6.1. Commandes communes utilisées pour ext3 et ext4 comparé à XFS

Tâche	ext3/4	XFS
Créer un système de fichiers	<code>mkfs.ext4</code> or <code>mkfs.ext3</code>	<code>mkfs.xfs</code>
Vérification du système de fichiers (fsck)	<code>e2fsck</code>	<code>xfs_repair</code>
Redimensionner un système de fichiers	<code>resize2fs</code>	<code>xfs_growfs</code>
Enregistre une image de système de fichiers	<code>e2image</code>	<code>xfs_metadump</code> et <code>xfs_mdrestore</code>
Libeller ou ajuster un système de fichiers	<code>tune2fs</code>	<code>xfs_admin</code>
Sauvegarder un système de fichiers	<code>dump</code> et <code>restore</code>	<code>xfsdump</code> et <code>xfsrestore</code>

Le tableau suivant répertorie tous les outils standard qui fonctionnent pour les systèmes de fichiers XFS, mais les versions XFS ont une fonctionnalité particulière, et sont conseillés pour cette raison.

Tableau 6.2. Outils standards pour ext4 et XFS

Tâche	ext4	XFS
Quota	<code>quota</code>	<code>xfs_quota</code>
Mappage de fichiers	<code>filefrag</code>	<code>xfs_bmap</code>

Vous trouverez davantage d'informations sur les commandes XFS listées dans [Chapitre 6, Le système de fichiers XFS](#). Vous pourrez également consulter les pages man des outils d'administration XFS listés pour obtenir plus d'informations.

6.9.2. Différences comportementales et administratives entre Ext3/4 et XFS

Réparation du système de fichiers

Ext3/4 exécute `e2fsck` dans l'espace utilisateur au démarrage pour récupérer le journal si besoin. XFS, en revanche, procède à la récupération du journal de l'espace noyau au montage. Un script shell `fsck.xfs` est fourni, mais il ne sert pas à grand chose sauf en tant que prérequis d'initcript.

Quand une réparation de système de fichier XFS ou qu'une vérification est requise, utiliser la commande `xfs_repair`. Utiliser l'option `-n` pour une vérification en lecture-seule.

La commande **xfs_repair** ne fonctionnera pas sur un système de fichiers qui a un journal endommagé. Pour réparer un tel fichier, les commandes **mount** et **unmount** doivent être exécutées pour commencer afin d'exécuter le journal à nouveau. Si le journal est corrompu ou ne peut pas être exécuté à nouveau, l'option **-L** pourra être utilisée pour neutraliser le journal.

Pour plus d'informations sur la réparation du système de fichiers des systèmes de fichiers XFS, consulter [Section 11.2.2, « XFS »](#)

Comportement des erreurs de métadonnées

Le système de fichiers ext2/3 a un comportement configurable quand des erreurs de métadonnées surviennent, et que le comportement par défaut est tout simplement de continuer. Quand XFS rencontre une erreur de métadonnées non récupérable, il ferme le système de fichiers et renvoie comme message d'erreur **EFSCORRUPTED**. Les journaux système contiendront des détails sur les erreurs rencontrées et vous conseilleront d'exécuter **xfs_repair** si besoin est.

Quotas

Les quotas XFS n'ont pas d'option pour remonter. L'option **-o quota** doit être spécifiée sur le point de montage de départ pour que les quotas puissent prendre effet.

Bien que les outils standards du package des quotas puisse effectuer des tâches administratives de base (comme **setquota** et **repquota**), l'outil **xfs_quota** peut être utilisé pour des fonctions spécifiques XFS, comme l'administration des quotas d'un projet.

La commande **quotacheck** n'a aucun effet sur un système de fichiers XFS. La première fois que le compteur de quotas est activé, XFS procède à un **quotacheck** interne automatiquement. Comme les métadonnées de quota XFS représentent un objet de métadonnées journalisées de première classe, le système de fichiers sera toujours consistant jusqu'à ce que les quotas soient désactivés manuellement.

Redimensionnement du système de fichiers

Le système de fichiers XFS n'a pas de fonctionnalité qui puisse permettre de réduire un système de fichiers. Les systèmes de fichiers XFS peuvent être agrandis en ligne à l'aide de la commande **xfs_growfs**.

Les numéros d'inodes

Pour les systèmes de fichiers supérieurs à 1T avec des inodes de 256 octets, ou de plus de 2T avec des inodes de 512 octets, les numéros d'inode XFS pourraient dépasser 2^{32} . Ces numéros d'inode élevés peuvent amener les appels de 32-bit à échouer avec le message **Eoverflow**. En général, les applications doivent gérer des numéros d'inodes élevés, mais, si nécessaire, XFS devra être monté avec **-o inode32** pour appliquer les numéros d'inode inférieurs à 2^{32} .



Note

Cela n'affectera pas les inodes déjà alloués avec des numéros de 64-bit.

L'option change le comportement d'allocation et mène rapidement à un ENOSPC si aucun espace n'est disponible pour allouer des inodes dans les blocs de disques inférieurs. L'option **inode32** ne devra pas être utilisée à moins qu'elle ne soit requise pour un environnement particulier.

Pré-allocation spéculative

XFS utilise *speculative preallocation* pour allouer des blocs après EOF au fur et à mesure que les

fichiers sont écrits. Cela permet d'éviter la fragmentation des fichiers liée aux charges de travail de flux d'écriture parallèles sur les serveurs NFS. Par défaut, cette pré-allocation augmente en fonction de la taille du fichier, et sera apparente dans la sortie "du". Si un fichier ayant une pré-allocation spéculative n'est pas modifié sous les 5 minutes, la pré-allocation sera abandonnée. Si l'inode est sortie du cache avant ce moment-là, la pré-allocation sera abandonnée quand l'inode sera récupérée.

Si des problèmes ENOSPC apparaissent suite à une pré-allocation spéculative, un montant de pré-allocation prédéterminé pourra être spécifié par l'option de montage **-o allocsize=amount**.

Outils de fragmentation

La fragmentation est rarement un problème pour les systèmes de fichiers XFS pour des raisons heuristiques et de comportement, comme les retardements d'allocation ou les préallocations spéculatives. Cependant, il existe des outils pour mesurer la fragmentation des systèmes de fichiers, ainsi que pour défragmenter les systèmes de fichiers. Leur utilisation n'est pas conseillée.

Les tentatives de commande **xfs_db frag** servent à distiller toutes les allocations de système de fichier en un nombre unique de fragmentations, exprimée en pourcentage. La sortie de la commande requiert une grande compétence pour en comprendre la signification. Par exemple, un facteur de fragmentation de 75 % indique uniquement une moyenne de 4 degrés par fichier. Pour cette raison la sortie de frag de xfs_db n'est pas jugée utile, et une analyse plus attentive de tout problème de fragmentation est recommandée.



Avertissement

La commande **xfs_fsr** peut être utilisée pour défragmenter des fichiers individuels, ou tous les fichiers d'un système de fichiers. Cette dernière est particulièrement déconseillée car elle peut détruire l'emplacement des fichiers et risque de fragmenter l'espace libre.

Chapitre 7. Global File System 2

Le système de fichiers de Red Hat GFS2 (« Global File System 2 ») est un système de fichiers natif qui correspond directement avec l'interface de système de fichiers de Linux (couche VFS). Lorsqu'il est implémenté comme un système de fichiers en groupement, GFS2 emploie des métadonnées distribuées et de nombreux journaux.

GFS2 est basé sur une architecture de 64 bits qui peut, en théorie, accommoder un système de fichiers de 8 exaoctets. La taille maximale prise en charge d'un système de fichiers GFS2 est de 100 To. Si un système requiert des systèmes de fichiers GFS2 de plus de 100 To, veuillez contacter votre représentant de services Red Hat.

Lorsque vous déterminez la taille d'un système de fichiers, veuillez prendre en considération vos besoins de recouvrement. L'exécution de la commande **fsck** sur un système de fichiers de très grande taille peut prendre longtemps et consommer beaucoup de mémoire. De plus, en cas de défaillance d'un disque ou d'un sous-système de disque, le temps de récupération sera limité par la vitesse de votre support de sauvegarde.

Lorsqu'ils sont configurés dans Red Hat Cluster Suite, les nœuds GFS2 de Red Hat peuvent être configurés et gérés par des outils de gestion et de configuration Red Hat Cluster Suite. Red Hat GFS2 fournit alors le partage des données entre les nœuds GFS2 d'un cluster Red Hat, avec affichage unique et cohérent de l'espace-nom du système de fichiers à travers les nœuds GFS2. Ceci permet aux processus se trouvant sur différents nœuds de partager des fichiers GFS2 de la même manière que les processus d'un même nœud peuvent partager des fichiers sur un système de fichiers local, sans aucune différence discernable. Pour obtenir des informations sur Red Hat Cluster Suite, veuillez consulter le guide de Red Hat *Administration de clusters*.

GFS2 doit être construit sur un volume logique (créé avec LVM) qui soit un volume linéaire ou un volume miroir. Les volumes logiques créés avec LVM dans Red Hat Cluster Suite sont gérés avec CLVM (qui est une implémentation niveau-cluster de LVM), activés par le démon CLVM **clvmd**, qui est exécuté dans un cluster de Red Hat Cluster Suite. Le démon facilite l'utilisation de LVM2 pour gérer les volumes logiques à travers un cluster, permettant ainsi à tous les nœuds du cluster de partager les volumes logiques. Pour obtenir des informations sur le gestionnaire de volumes logiques LVM, veuillez consulter le guide de Red Hat *Administration du gestionnaire de volumes logiques*.

Le module de noyau **gfs2.ko** implémente le système de fichiers GFS2 et est chargé dans les nœuds de cluster GFS2.

Pour obtenir des informations complètes sur la création et la configuration des systèmes de fichiers GFS2 dans un stockage en cluster ou non, veuillez consulter le guide de Red Hat *Global File System 2*.

Chapitre 8. Network File System (NFS)

Un système de fichiers *NFS* (« *Network File System* ») permet aux hôtes distants de monter les systèmes de fichiers sur un réseau et d'interagir avec ces systèmes de fichiers tant que ceux-ci sont montés localement. Ceci permet aux administrateurs système de consolider leurs ressources sur des serveurs centralisés sur le réseau.

Ce chapitre traite des concepts NFS fondamentaux et fournit également des informations supplémentaires

8.1. Fonctionnement NFS

Actuellement, il y a deux versions de NFS incluses dans Red Hat Enterprise Linux. NFS version 3 (NFSv3) prend en charge les écritures asynchrones sécurisées et offre une gestion des erreurs plus robuste que NFSv2. NFSv3 prend aussi en charge les fichiers et décalages d'une taille de 64 bits, permettant ainsi aux clients d'accéder à plus de 2 Go de données de fichiers. NFSv4 fonctionne à travers les parefeux et sur l'Internet, n'a plus besoin du service **rpcbind**, prend en charge les ACL, et utilise les opérations *stateful*.

Red Hat Enterprise Linux 7 ajoute un support à NFS version 4.1 (NFSv4.1), ce qui propose un certain nombre d'améliorations des performances et de la sécurité, y compris la prise en charge des clients *pNFS* (« *Parallel NFS* »). En outre, une connexion TCP distincte n'est plus nécessaire pour les rappels, permettant ainsi à un serveur NFS d'attribuer des délégations même lorsqu'il ne peut pas contacter le client (par exemple lorsqu'un pare-feu ou un NAT interfère). De plus, NFSv4.1 fournit des sémantiques *true exactly-one* (sauf pour les opérations de démarrage), résolvant ainsi un problème connu qui faisait que certaines opérations pouvaient renvoyer un résultat inexact si une réponse était perdue et que l'opération avait été envoyée à deux reprises.

Red Hat Enterprise Linux 7 prend en charge les clients NFSv3, NFSv4.0, et NFSv4.1. Les clients NFS tentent de monter par NFSv4.0 par défaut, et retombent sur NFSv3 si l'opération de montage a échoué.



Note

NFS version 2 (NFSv2) n'est plus prise en charge par Red Hat.

Toutes les versions de NFS peuvent utiliser *TCP* (*Transmission Control Protocol*) exécuté sur un réseau IP, avec NFSv4 qui le requiert. NFSv3 peuvent utiliser le protocole *UDP* (*User Datagram Protocol*) exécuté sur un réseau IP afin de fournir une connexion réseau sans état (« *Stateless* ») entre le client et le serveur.

Lors de l'utilisation de NFSv3 avec UDP, la connexion UDP sans état (sous des conditions normales) possède un alourdissement du protocole moindre que TCP. Ceci se traduit par de meilleures performances sur des réseaux propres et non encombrés. Cependant, comme UDP est sans état, si le serveur tombe en panne de manière inattendue, les clients UDP continueront de saturer le réseau avec des requêtes pour le serveur. En outre, lorsqu'une trame est perdue avec UDP, la requête RPC entière doit être retransmise. Avec TCP, seule la trame perdue doit être envoyée à nouveau. Pour ces raisons, TCP est le protocole préféré lors d'une connexion à un serveur NFS.

The mounting and locking protocols have been incorporated into the NFSv4 protocol. The server also listens on the well-known TCP port 2049. As such, NFSv4 does not need to interact with **rpcbind** ^[1], **lockd**, and **rpc.statd** daemons. The **rpc.mountd** daemon is still required on the NFS server to set up the exports, but is not involved in any over-the-wire operations.



Note

TCP est le protocole de transport par défaut de NFS version 2 et 3 sous Red Hat Enterprise Linux. UDP peut être utilisé à des fins de compatibilité selon les besoins, mais n'est pas recommandé pour une utilisation globale. NFSv4 requiert TCP.

Tous les démons RPC/NFS possèdent une option de ligne de commande '**-p**' pouvant définir le port, ce qui rend la configuration du pare-feu plus facile.

Une fois que les emballages TCP auront fourni l'accès au client, le serveur NFS se référera au fichier de configuration **/etc/exports** pour déterminer si le client a le droit d'accéder à un système de fichiers exporté. Une fois cette vérification effectuée, toutes les opérations des fichiers et répertoires seront disponibles pour l'utilisateur.



Important

Pour que NFS puisse fonctionner avec une installation par défaut de Red Hat Enterprise Linux avec un pare-feu activé, veuillez configurer IPTables avec le port TCP par défaut 2049. NFS ne fonctionnera pas correctement sans une configuration d'IPTables correcte.

Le script d'initialisation NFS et le processus **rpc.nfsd** permettent désormais la liaison vers tout port spécifié pendant le démarrage système. Cependant, ceci est prôné aux erreurs si le port est indisponible, ou s'il est en conflit avec un autre démon.

8.1.1. Services requis

Red Hat Enterprise Linux utilise une combinaison des processus de démons et de support technique au niveau du noyau pour fournir le partage de fichiers NFS. Toutes les versions NFS reposent sur les *RPC* (*Remote Procedure Calls*) entre clients et serveurs. Les services RPC sous Red Hat Enterprise Linux 7 sont contrôlés par le service **rpcbind**. Pour partager ou monter les systèmes de fichiers NFS, les services suivants travaillent ensemble selon la version NFS implémentée :



Note

Le service **portmap** a été utilisé pour mapper les numéros de programmes RPC à des combinaisons de numéros de port d'adresses IP dans des versions plus récentes de Red Hat Enterprise Linux. Ce service est désormais remplacé par **rpcbind** dans Red Hat Enterprise Linux 7 afin de permettre la prise en charge d'IPv6. Pour obtenir des informations supplémentaires sur ce changement, veuillez consulter les liens suivants :

- ✦ *TI-RPC / prise en charge rpcbind* : http://nfsv4.bullopen-source.org/doc/tirpc_rpcbind.php
- ✦ *Prise en charge IPv6 sur NFS* : http://nfsv4.bullopen-source.org/doc/nfs_ipv6.php

nfs

servicectl start nfs lance le serveur NFS et les processus RPC appropriés pour servir les requêtes des systèmes de fichiers NFS partagés.

nfslock

servicectl start nfs-lock active un service obligatoire qui lance les processus RPC appropriés, ce qui permet aux clients NFS de verrouiller des fichiers sur le serveur.

rpcbind

rpcbind accepte les réservations de ports des services RPC locaux. Ces ports sont ensuite mis à disposition (ou publicisés) afin que les services RPC à distance correspondants puissent y accéder. **rpcbind** répond à des requêtes de service RPC et paramètre des connexions vers le service RPC requis. Ceci n'est pas utilisé avec NFSv4.

Les processus RPC suivants facilitent les services NFS :

rpc.mountd

Ce processus est utilisé par un serveur NFS pour traiter les requêtes **MOUNT** des clients NFSv3. Il vérifie que le partage NFS requis est actuellement exporté par le serveur NFS, et que le client est autorisé à y accéder. Si la requête de montage est autorisée, le serveur **rpc.mountd** répond avec le statut **Success** (« Opération réussie ») et retourne l'identificateur de fichier « **File-Handle** » de ce partage NFS au client NFS.

rpc.nfsd

rpc.nfsd permet de définir les versions et protocoles NFS explicites publicisés par le serveur. Celui-ci fonctionne avec le noyau Linux afin de répondre aux demandes des clients NFS, comme pour fournir des threads chaque fois qu'un client NFS se connecte. Ce processus correspond au service **nfs**.

lockd

lockd est un thread du noyau qui peut être exécuté sur les clients et les serveurs. Il implémente le protocole NLM (« *Network Lock Manager* »), qui permet aux clients NFSv3 de verrouiller des fichiers sur le serveur. Il est lancé automatiquement à chaque fois que le serveur NFS est exécuté et à chaque fois qu'un système de fichiers NFS est monté.

rpc.statd

Ce processus implémente le protocole RPC NSM (« *Network Status Monitor* »), qui notifie les clients NFS lorsqu'un serveur NFS est redémarré sans avoir tout d'abord été éteint correctement. **rpc.statd** est automatiquement démarré par le service **nfslock**, et ne requiert pas de configuration utilisateur. Ce protocole n'est pas utilisé avec NFSv4.

rpc.rquotad

Ce processus fournit des informations sur le quota d'utilisateur des utilisateurs distants. **rpc.rquotad** est automatiquement démarré par le service **nfs** et ne requiert pas de configuration utilisateur.

rpc.idmapd

rpc.idmapd fournit des appels ascendants client et serveur NFSv4, qui mappent simultanément les noms NFSv4 (chaînes sous le format **utilisateur@domaine**) et les UID et GID locaux. Pour que **idmapd** puisse fonctionner avec NFSv4, le fichier **/etc/idmapd.conf** doit être configuré. Au minimum, le paramètre « Domaine », qui définit le domaine de mappage NFSv4, doit être spécifié. Si le domaine de mappage NFSv4 est le même que le nom de domaine DNS, oubliez ce paramètre. Le client et le serveur doivent se mettre d'accord sur le domaine de mappage NFSv4 pour que le mappage d'ID fonctionne correctement.



Note

Dans Red Hat Enterprise Linux 7, seul le serveur NFSv4 utilise **rpc.idmapd**. Le client NFSv4 utilise **nfsidmap** de l'imapper basé-keyring. **nfsidmap** est un programme autonome appelé par le noyau à la demande pour effectuer les mappages d'ID ; ce n'est pas un démon. S'il y a un problème avec **nfsidmap**, le client utilise alors **rpc.idmapd**. Vous trouverez plus d'informations sur **nfsidmap** dans la page man de **nfsidmap**.

8.2. pNFS

La prise en charge de pNFS (« Parallel NFS ») dans le cadre du standard NFS v4.1 est disponible à partir de Red Hat Enterprise Linux 6.4. L'architecture pNFS améliore l'évolutivité de NFS, avec de possibles améliorations des performances. Lorsqu'un serveur implémente pNFS, un client sera en mesure d'accéder à des données à travers de multiples serveurs simultanément. Trois protocoles ou structures de stockage sont pris en charge : les fichiers, objets, et blocs.



Note

Le protocole autorise trois types de structures pNFS : les fichiers, objets, et les blocs. Cependant, le client Red Hat Enterprise Linux 6.4 prend uniquement en charge le type de structure en fichiers, Red Hat Enterprise Linux 7 prend en charge le type de structure en fichiers, en objets et en blocs inclus dans l'aperçu technologique.

Pour activer cette fonctionnalité, veuillez utiliser l'option de montage suivante sur les montages d'un serveur activé pNFS :

```
-o v4.1
```

Une fois le serveur activé pour pNFS, le noyau **nfs_layout_nfsv41_files** est automatiquement chargé sur le premier montage. L'entrée de montage de la sortie doit contenir **minorversion=1**. Veuillez utiliser la commande suivante pour vérifier que le module a bien été chargé :

```
$ lsmod | grep nfs_layout_nfsv41_files
```

Pour obtenir plus d'informations sur pNFS, veuillez vous reporter à : <http://www.pnfs.com>.

8.3. Configuration du client NFS

La commande **mount** monte les partages NFS côté client. Son format est comme suit :

```
# mount -t nfs -o options server:/remote/export /local/directory
```

Cette commande utilise les variables suivantes :

options

Liste d'options de montage séparées par des virgules. Veuillez consulter la [Section 8.5, « Options de montage NFS courantes »](#) pour obtenir des détails sur les options de montage NFS valides.

server

Nom d'hôte, adresse IP, ou nom de domaine complet du serveur exportant le système de fichiers que vous souhaitez monter

/remote/export

Système de fichiers ou répertoire en cours d'exportation du *serveur*, c'est-à-dire le répertoire que vous souhaitez monter

//local/directory

Emplacement du client où */remote/export* est monté

La version du protocole NFS utilisée dans Red Hat Enterprise Linux 7 est identifiée par les options **mount nfsvers** ou **vers**. Par défaut, **mount** utilisera NFSv4 avec **mount -t nfs**. Si le serveur ne prend pas en charge NFSv4, le client passera automatiquement à une version prise en charge par le serveur. Si l'option **nfsvers/vers** est utilisée pour passer une version particulière qui n'est pas prise en charge par le serveur, le montage échouera. Le type de système de fichiers **nfs4** est également disponible pour des raisons d'héritage ; ceci est équivalent à exécuter **mount -t nfs -o nfsvers=4 host:/remote/export //local/directory**.

Veillez consulter **man mount** pour davantage de détails.

Si un partage NFS a été monté manuellement, le partage ne sera pas automatiquement monté après un redémarrage. Red Hat Enterprise Linux offre deux méthodes pour monter des systèmes de fichiers à distance automatiquement pendant un démarrage : le fichier **/etc/fstab** et le service **autofs**. Veuillez consulter la [Section 8.3.1, « Monter des systèmes de fichiers NFS à l'aide de /etc/fstab »](#) et [Section 8.4, « autofs »](#) pour obtenir davantage d'informations.

8.3.1. Monter des systèmes de fichiers NFS à l'aide de /etc/fstab

An alternate way to mount an NFS share from another machine is to add a line to the **/etc/fstab** file. The line must state the hostname of the NFS server, the directory on the server being exported, and the directory on the local machine where the NFS share is to be mounted. You must be root to modify the **/etc/fstab** file.

Exemple 8.1. Exemple de syntaxe

La syntaxe générale de la ligne du fichier **/etc/fstab** est comme suit :

```
server:/usr/local/pub /pub nfs defaults 0 0
```

Le point de montage **/pub** doit exister sur l'ordinateur client avant que cette commande puisse être exécutée. Après avoir ajouté cette ligne à **/etc/fstab** sur le système client, veuillez utiliser la commande **mount /pub**, et le point de montage **/pub** est monté à partir du serveur.

Une entrée **/etc/fstab** valide pour monter un export NFS doit contenir les informations suivantes :

```
server:/remote/export //local/directory nfs options 0 0
```

Les variables *server*, */remote/export*, *//local/directory*, et *options* sont les mêmes que lors d'un montage de partage NFS manuel. Veuillez consulter [Section 8.3, « Configuration du client NFS »](#) pour une définition de chaque variable.



Note

Le point de montage `/local/directory` doit exister sur le client avant la lecture de `/etc/fstab`. Sinon le montage échouera.

Pour obtenir davantage d'informations sur `/etc/fstab`, veuillez consulter `man fstab`.

8.4. autofs

Un inconvénient lié à l'utilisation de `/etc/fstab` fait que peu importe la fréquence à laquelle un utilisateur accède au système de fichiers NFS monté, le système doit dédier des ressources afin de garder le système de fichiers en place. Il ne s'agit pas d'un problème avec un ou deux montages, mais lorsque le système maintient les montages de nombreux systèmes à la fois, les performances générales du système peuvent être affectées. Une alternative à `/etc/fstab` consiste à utiliser l'utilitaire basé noyau **automount**.

Automounter consiste en deux composants :

- ✦ Un module de noyau qui implémente un système de fichiers, et
- ✦ un démon de l'espace utilisateur qui effectue toutes les autres fonctions.

L'utilitaire **automount** peut monter et démonter des systèmes de fichiers NFS automatiquement (montage à la demande), et permet donc d'économiser des ressources système. Il peut être utilisé pour monter d'autres systèmes de fichiers, y compris AFS, SMBFS, CIFS, et des systèmes de fichiers locaux.



Important

Le paquet `nfs-utils` fait désormais partie des groupes « NFS file server » et « Network File System Client ». Ainsi, il n'est plus installé par défaut avec le groupe de base. Assurez-vous que `nfs-utils` soit installé sur le système avant de tenter un montage automatique sur un partage NFS.

`autofs` fait également partie du groupe « Network File System Client ».

autofs utilise `/etc/auto.master` (mappage principal) comme fichier de configuration principal. Ceci peut être changé afin d'utiliser une autre source réseau et un autre nom pris en charge en utilisant la configuration **autofs** (dans `/etc/sysconfig/autofs`) en conjonction avec le mécanisme NSS (« Name Service Switch »). Une instance du démon **autofs** version 4 était exécutée pour chaque point de montage configuré dans le mappage principal, lui permettant d'être exécutée manuellement à partir de la ligne de commande pour tout point de montage donné. Ceci n'est pas possible avec **autofs** version 5, car un seul démon est utilisé pour gérer tous les points de montage configurés dans le mappage principal. Ceci est effectué conformément aux conditions préalables des autres automounters (monteurs automatiques) standards du secteur. Les points de montage, noms d'hôte, répertoires exportés, et les options peuvent tous être spécifiés dans un ensemble de fichiers (ou autres sources réseau prises en charge) plutôt que de devoir les configurer manuellement pour chaque hôte.

8.4.1. Améliorations de autofs Version 5 par rapport à la Version 4

autofs version 5 offre les améliorations suivantes par rapport à la version 4 :

Prise en charge du mappage direct

Les mappages directs dans **autofs** offrent un mécanisme pour monter les systèmes de fichiers automatiquement sur des points arbitraires dans la hiérarchie du système de fichiers. Un mappage direct est indiqué par un point de montage /- dans le mappage principal. Les entrées dans un mappage direct contiennent un nom de chemin absolu comme clé (au lieu des noms de chemin relatifs utilisés pour les mappages indirects).

Prise en charge des montages et démontages « lazy »

Les entrées de mappage à multiples montages décrivent une hiérarchie de points de montage sous une clé unique. Un bon exemple de ceci est le mappage **-hosts**, couramment utilisé pour effectuer le montage automatique de tous les exports à partir d'un hôte sous **/net/host** en tant qu'entrée de mappage à multiples montages. Lors de l'utilisation du mappage **-hosts**, la commande **ls** de **/net/host** montera les montages de déclencheurs **autofs** de chaque export de l'hôte. Ceux-ci seront ensuite montés et expireront au fur et à mesure que l'on y accédera. Ceci peut grandement réduire le nombre de montages actifs nécessaires lors de l'accession à un serveur avec un grand nombre d'exports.

Prise en charge LDAP améliorée

Le fichier de configuration **autofs** (**/etc/sysconfig/autofs**) offre un mécanisme pour spécifier le schéma **autofs** implémenté par un site, éliminant ainsi le besoin de déterminer ceci intuitivement dans l'application. En outre, les liaisons authentifiées sur le serveur LDAP sont désormais prises en charge, en utilisant la plupart des mécanismes pris en charge par les implémentations de serveurs LDAP courantes. Un nouveau fichier de configuration a été ajouté pour cette prise en charge : **/etc/autofs_ldap_auth.conf**. La configuration par défaut explique bien les choses et utilise un format XML.

Utilisation correcte de la configuration « Name Service Switch » (nsswitch).

Le fichier de configuration du « Name Service Switch » existe pour fournir un moyen de déterminer d'où proviennent des données de configuration spécifiques. Le but de cette configuration est d'offrir aux administrateurs la flexibilité d'utiliser la base de données d'arrière-plan de leur choix, tout en conservant une interface logiciel uniforme pour accéder aux données. Malgré le fait qu'automounter version 4 gère de mieux en mieux la configuration NSS, celui-ci ne la gère pas totalement. En revanche, Autofs version 5 offre une implémentation totale.

Veuillez consulter **man nsswitch.conf** pour obtenir des informations supplémentaires sur la syntaxe prise en charge de ce fichier. Toutes les bases de données NSS ne sont pas des sources de mappage valides et l'analyseur rejettera celles qui ne sont pas valides. Les sources valides incluent les fichiers, **yp**, **nis**, **nisplus**, **ldap**, et **hesiod**.

Entrées multiples de mappage principal par point de montage autofs

Une chose fréquemment utilisée mais qui n'a pas encore été mentionnée est la gestion de multiples entrées de mappage principal pour le point de montage direct /-. Les clés de mappage de chaque entrée sont fusionnées et se comportent comme une seule carte.

Exemple 8.2. Entrées multiples de mappage principal par point de montage autofs

Ci-dessous figure un exemple des mappages de test connectathon pour les montages directs :

```

/- /tmp/auto_dcthon
/- /tmp/auto_test3_direct
/- /tmp/auto_test4_direct

```

8.4.2. Configuration autofs

`/etc/auto.master` est le fichier de configuration principal pour automounter, également appelé mappage principal, ce qui peut être modifié comme décrit dans la [Section 8.4.1, « Améliorations de autofs Version 5 par rapport à la Version 4 »](#). Le mappage principal répertorie les points de montage du système contrôlé par **autofs**, ainsi que leurs fichiers de configuration ou sources réseau correspondants, également appelés cartes automount. Le format du mappage principal est comme suit :

```
mount-point map-name options
```

Les variables utilisées dans ce format sont les suivantes :

mount-point

Point de montage **autofs**, par exemple `/home`.

map-name

Nom d'une source de mappage qui contient une liste de points de montage, et l'emplacement du système de fichiers à partir duquel ces points de montage doivent être montés. La syntaxe d'une entrée de mappage est décrite ci-dessous.

options

Si fournies, celles-ci seront applicables à toutes les entrées du mappage donné, à condition qu'elles ne possèdent pas elles-même d'options spécifiées. Ce comportement est différent de celui d'**autofs** version 4, où les options étaient cumulatives. Cela a été modifié afin d'implémenter une compatibilité d'environnements mélangés.

Exemple 8.3. Fichier `/etc/auto.master`

Ci-dessous figure l'exemple d'une ligne du fichier `/etc/auto.master` (affiché en saisissant `cat /etc/auto.master`) :

```
/home /etc/auto.misc
```

Le format général des mappages est similaire à celui du mappage principal, cependant, les « options » apparaissent entre le point de montage et l'emplacement au lieu de se situer à la fin de l'entrée, comme c'est le cas dans le mappage principal :

```
mount-point [options] emplacement
```

Les variables utilisées dans ce format sont les suivantes :

mount-point

Référence au point de montage **autofs**. Il peut s'agir d'un nom de répertoire unique pour un montage indirect ou du chemin complet du point de montage pour les montages directs. Chaque clé d'entrée de mappage direct et indirect (**mount-point** above) peut être suivie d'une liste de répertoires décalés (les noms de sous-répertoires commencent tous par le caractère « / ») séparés par des virgules, en faisant ainsi une entrée à montages multiples.

options

Lorsqu'elles sont fournies, celles-ci servent d'options de montage pour les entrées de mappage ne spécifiant pas leurs propres options.

emplacement

Ceci fait référence à un emplacement de système de fichiers, tel qu'un chemin de système de fichiers local (précédé par le caractère d'échappement du format de mappage Sun « : » pour les noms de mappage commençant par une barre oblique « / »), un système de fichiers NFS, ou tout autre emplacement de système de fichiers valide.

Ci-dessous figure un exemple du contenu d'un fichier de mappage (par exemple, `/etc/auto.misc`) :

```
payroll -fstype=nfs personnel:/dev/hda3
sales -fstype=ext3 :/dev/hda4
```

La première colonne dans un fichier de mappage indique le point de montage **autofs** (**sales** et **payroll** provenant du serveur nommé **personnel**). La seconde colonne indique les options du montage **autofs** tandis que la troisième colonne indique la source du montage. Selon la configuration ci-dessus, les points de montage autofs seront nommés `/home/payroll` et `/home/sales`. L'option **-fstype=** est souvent omise et n'est généralement pas utile au bon fonctionnement de ce fichier.

Automounter créera les répertoires s'ils n'existent pas. S'ils existaient avant le démarrage d'Automounter, alors Automounter ne les supprimera pas lors de sa fermeture. Vous pouvez démarrer ou redémarrer le démon Automount en exécutant l'une des deux commandes suivantes :

- ✦ **service autofs start** (si le démon automount est à l'arrêt)
- ✦ **service autofs restart**

Lors de l'utilisation de la configuration ci-dessus, si un processus requiert l'accès à un répertoire **autofs** non monté, tel que `/home/payroll/2006/July.sxc`, le démon automount monte automatiquement le répertoire. Si un délai d'expiration a été spécifié et qu'aucun accès au répertoire n'est effectué pendant cette période, alors le répertoire sera automatiquement démonté.

Vous pouvez afficher le statut du démon automount en exécutant la commande suivante :

```
# service autofs status
```

8.4.3. Remplacer ou augmenter les fichiers de configuration du site

Il peut être utile de remplacer les valeurs de site par défaut par un point de montage spécifique sur un système client. Par exemple, veuillez envisager les conditions suivantes :

- ✦ Les mappages Automounter sont stockés dans le système d'informations réseau (ou NIS) et le fichier `/etc/nsswitch.conf` contient la directive suivante :

```
automount: files nis
```

- ✦ Ci-dessous figure le contenu du fichier **auto.master** :

```
+auto.master
```

- ✦ Ci-dessous figure le contenu du fichier de mappage **auto.master** du NIS :

```
/home auto.home
```

❖ Ci-dessous figure le contenu du mappage **auto.home** du NIS :

```
beth      fileserver.example.com:/export/home/beth
joe      fileserver.example.com:/export/home/joe
*        fileserver.example.com:/export/home/&
```

❖ Le fichier de mappage **/etc/auto.home** n'existe pas.

Dans ces conditions, supposons que le système client doit remplacer le mappage NIS **auto.home** et monter les répertoires d'accueil du montage à partir d'un serveur différent. Dans ce cas, le client devra utiliser le mappage **/etc/auto.master** suivant :

```
/home /etc/auto.home
+auto.master
```

Le mappage **/etc/auto.home** contient l'entrée :

```
* labserver.example.com:/export/home/&
```

Comme automounter ne traite que la première occurrence d'un point de montage, **/home** contiendra le contenu de **/etc/auto.home** au lieu de celui du mappage NIS **auto.home**.

Alternativement, pour augmenter le mappage global **auto.home** du site avec quelques entrées uniquement, veuillez créer un mappage du fichier **/etc/auto.home** et y inclure les nouvelles entrées. À la fin, veuillez inclure le mappage NIS **auto.home**. Ainsi, le mappage du fichier **/etc/auto.home** sera similaire à :

```
mydir someserver:/export/mydir
+auto.home
```

Avec le mappage NIS **auto.home** répertorié ci-dessus, **ls /home** retournera :

```
beth joe mydir
```

Ce dernier exemple fonctionne comme prévu car **autofs** n'inclut pas le contenu d'un mappage de fichier du même nom que celui qu'il lit. Ainsi, **autofs** se déplace vers la prochaine source de mappage de la configuration **nsswitch**.

8.4.4. Utiliser LDAP pour stocker des mappages Automounter

Les bibliothèques de clients LDAP doivent être installées sur tous les systèmes configurés pour récupérer les mappages automounter de LDAP. Sur Red Hat Enterprise Linux, le paquet **openldap** devrait être installé automatiquement en tant que dépendance d'**automounter**. Pour configurer l'accès LDAP, veuillez modifier **/etc/openldap/ldap.conf**. Assurez-vous que les valeurs BASE, URI, et schémas soient définies correctement pour votre site.

Le schéma le plus récemment établi pour stocker des mappages Automount dans LDAP est décrit par **rfc2307bis**. Pour utiliser ce schéma, il est nécessaire de le définir dans la configuration **autofs** (**/etc/sysconfig/autofs**) en supprimant les caractères de commentaires de la définition du schéma. Par exemple :

Exemple 8.4. Paramétrer la configuration autofs

```

DEFAULT_MAP_OBJECT_CLASS="automountMap"
DEFAULT_ENTRY_OBJECT_CLASS="automount"
DEFAULT_MAP_ATTRIBUTE="automountMapName"
DEFAULT_ENTRY_ATTRIBUTE="automountKey"
DEFAULT_VALUE_ATTRIBUTE="automountInformation"

```

Assurez-vous qu'il s'agisse bien des seules entrées du schéma qui ne soient pas des commentaires dans la configuration. **automountKey** remplace l'attribut **cn** dans le schéma **rfc2307bis**. Un **LDIF** d'un exemple de configuration est décrit ci-dessous :

Exemple 8.5. Configuration LDF

```

# extended LDIF
#
# LDAPv3
# base <> with scope subtree
# filter: (&(objectclass=automountMap)(automountMapName=auto.master))
# requesting: ALL
#

# auto.master, example.com
dn: automountMapName=auto.master,dc=example,dc=com
objectClass: top
objectClass: automountMap
automountMapName: auto.master

# extended LDIF
#
# LDAPv3
# base <automountMapName=auto.master,dc=example,dc=com> with scope subtree
# filter: (objectclass=automount)
# requesting: ALL
#

# /home, auto.master, example.com
dn: automountMapName=auto.master,dc=example,dc=com
objectClass: automount
cn: /home

automountKey: /home
automountInformation: auto.home

# extended LDIF
#
# LDAPv3
# base <> with scope subtree
# filter: (&(objectclass=automountMap)(automountMapName=auto.home))
# requesting: ALL
#

# auto.home, example.com
dn: automountMapName=auto.home,dc=example,dc=com

```

```

objectClass: automountMap
automountMapName: auto.home

# extended LDIF
#
# LDAPv3
# base <automountMapName=auto.home,dc=example,dc=com> with scope subtree
# filter: (objectclass=automount)
# requesting: ALL
#

# foo, auto.home, example.com
dn: automountKey=foo,automountMapName=auto.home,dc=example,dc=com
objectClass: automount
automountKey: foo
automountInformation: filer.example.com:/export/foo

# /, auto.home, example.com
dn: automountKey=/,automountMapName=auto.home,dc=example,dc=com
objectClass: automount
automountKey: /
automountInformation: filer.example.com:/export/&

```

8.5. Options de montage NFS courantes

Hormis le montage d'une système de fichiers avec NFS sur un hôte distant, il est également possible de spécifier d'autres options lors du montage afin de rendre l'utilisation du partage monté plus facile. Ces options peuvent être utilisées avec des commandes **mount**, des paramètres **/etc/fstab**, et avec **autofs**.

Les options suivantes sont communément utilisées pour effectuer des montages NFS :

intr

Permet aux requêtes NFS d'être interrompues si le serveur tombe en panne ou ne peut pas être contacté.

lookupcache=mode

Spécifie la manière par laquelle le noyau va gérer le cache de ses entrées de répertoire pour un point de montage donné. Les arguments valides de *mode* sont **all** (« tout »), **none** (« rien »), ou **pos/positive**.

nfsvers=version

Spécifie la version du protocole NFS à utiliser, où *version* est égal à 2, 3, ou 4. Ceci est utile pour les hôtes qui exécutent de multiples serveurs NFS. Si aucune version n'est spécifiée, NFS utilise le numéro de version le plus haut pris en charge par le noyau et la commande **mount**.

L'option **vers** est identique à **nfsvers** et est incluse dans cette version pour des raisons de compatibilité.

noacl

Annule tout traitement d'ACL. Ce paramètre peut être nécessaire lors d'interactions avec des versions plus anciennes de Red Hat Enterprise Linux, Red Hat Linux, ou Solaris, car les technologies ACL les plus récentes ne sont pas compatibles avec des systèmes plus anciens.

nolock

Désactive le verrouillage de fichiers. Ce paramètre est occasionnellement requis lors des connexions aux serveurs NFS plus anciens.

noexec

Empêche l'exécution de binaires sur des systèmes de fichiers montés. Ceci est utile si le système monte un système de fichiers Linux contenant des binaires incompatibles.

nosuid

Désactive les bits **set-user-identif** ou **set-group-identif**. Ceci empêche les utilisateurs distants de gagner des privilèges plus élevés en exécutant un programme **setuid**.

port=num

port=num — Spécifie la valeur numérique du port du serveur NFS. Si **num** est égal à **0** (valeur par défaut), alors **mount** interrogera le service **rpcbind** de l'hôte distant sur le numéro de port à utiliser. Si le démon NFS de l'hôte distant n'est pas enregistré avec son service **rpcbind**, le numéro de port NFS standard TCP 2049 sera alors utilisé.

rsize=num et wsize=num

Ces paramètres accélèrent les communications NFS pour les lectures (**rsize**) et les écritures (**wsize**) en définissant une taille de bloc de données plus importante (**num**, en octets) à transférer. Soyez prudent lorsque vous modifiez ces valeurs car certains noyaux Linux et cartes réseau ne fonctionnent pas bien avec des tailles de bloc plus importantes. Dans NFSv3, les valeurs par défaut des deux paramètres est définie à 8192. Dans NFSv4, les valeurs par défaut des deux paramètres est définie à 32768.

sec=mode

Sa valeur par défaut est **sec=sys**, qui utilise les UID et GID UNIX. Ils utilisent **AUTH_SYS** to authentifier les opérations NFS.

sec=krb5 utilise Kerberos V5 à la place d'UID et GID UNIX locaux pour authentifier les utilisateurs.

sec=krb5i utilise Kerberos V5 pour authentifier les utilisateurs et vérifie l'intégrité des opérations NFS en utilisant des checksums sécurisés pour empêcher toute altération de données.

sec=krb5p utilise Kerberos V5 pour l'authentification d'utilisateurs, les vérifications d'intégrité, et chiffre le trafic NFS pour empêcher le reniflage du trafic. Ce paramètre est le plus sécurisé, mais il implique également une surcharge des performances plus importante.

tcp

Ordonne au montage NFS d'utiliser le protocole TCP.

udp

Ordonne au montage NFS d'utiliser le protocole UDP.

Pour obtenir la liste complète des options ainsi que leurs informations détaillées, veuillez consulter **man mount** et **man nfs**.

8.6. Démarrage et arrêt NFS

Pour exécuter un serveur NFS, le service **rpcbind**^[1] doit être en cours d'exécution. Pour vérifier que **rpcbind** soit effectivement actif, veuillez utiliser la commande suivante :

```
# systemctl status rpcbind
```

Si le service **rpcbind** est en cours d'exécution, alors le service **nfs** peut être lancé. Pour démarrer un serveur NFS, veuillez utiliser la commande suivante :

```
# systemctl start nfs
```

Pour permettre à NFS de démarrer à l'amorçage, utiliser la commande suivante :

```
# systemctl enable nfs-server
```



Note

Dans NFSv3, si NFS est configuré pour démarrer à l'amorçage, le service **nfs-lock** doit être activé. Dans Red Hat Enterprise Linux 7.1 et versions supérieures, **nfs-lock** démarre automatiquement si nécessaire, et si une tentative de le démarrer manuellement a échoué. Dans Red Hat Enterprise Linux 7.0, vérifiez le statut en exécutant **systemctl status nfs-lock**. Si le service **nfs-lock** n'est pas activé, exécutez **systemctl start nfs-lock**. Pour configurer **nfs-lock** à démarrer automatiquement à l'amorçage dans Red Hat Enterprise Linux 7.0, exécutez **systemctl enable nfs-lock**.

Pour arrêter le serveur, veuillez utiliser :

```
# systemctl stop nfs
```

L'option **restart** est une façon rapide d'arrêter, puis de redémarrer NFS. Cette manière est la plus efficace pour que les changements de configuration puissent prendre effet après avoir modifié le fichier de configuration pour NFS. Pour redémarrer le serveur, veuillez saisir :

```
# systemctl restart nfs
```

Une fois que vous aurez modifié le fichier **/etc/sysconfig/nfs**, redémarrez le service **nfs-config** en exécutant la commande suivante pour que les nouvelles valeurs puissent prendre effet :

```
# systemctl restart nfs-config
```

La commande **try-restart** ne démarre **nfs** que s'il est en cours d'exécution. Cette commande équivaut à **condrestart** (*conditional restart*) dans les scripts init de Red Hat et est utilisée car elle ne démarre pas le démon si NFS n'est pas en cours d'exécution.

Pour redémarrer le serveur de façon conditionnelle, saisir :

```
# systemctl try-restart nfs
```

Pour recharger le fichier de configuration du serveur NFS sans redémarrer le service, veuillez saisir :

```
# systemctl reload nfs
```

8.7. Configuration du serveur NFS

Il existe deux manières de configurer les exportations sur un serveur NFS :

- La configuration manuelle du fichier de configuration NFS `/etc/exports`, et
- à l'aide de la ligne de commande, en utilisant la commande `exportfs`

8.7.1. Fichier de configuration `/etc/exports`

Le fichier `/etc/exports` contrôle quels systèmes de fichiers sont exportés vers des hôtes distants et spécifie les options. Les règles de syntaxes suivantes sont observées :

- Les lignes vides sont ignorées.
- Pour ajouter un commentaire, commencez la ligne par le caractère dièse (`#`).
- Pour les longues lignes, il est possible d'effectuer des retours à la ligne avec une barre oblique inversée (`\`).
- Chaque système de fichiers exporté devrait se trouver sur une ligne individuelle.
- Toute liste d'hôtes non autorisés placée après un système de fichiers exporté doit être séparée par des caractères d'espace.
- Les options de chaque hôte doivent être placées directement après l'identifiant de l'hôte, sans espace séparant l'hôte et la première parenthèse.

Chaque entrée de système de fichiers exporté possède la structure suivante :

```
export host(options)
```

La structure mentionnée ci-dessus utilise les variables suivantes :

export

Répertoire en cours d'exportation

host

Hôte ou réseau sur lequel l'exportation est partagée

options

Options à utiliser pour l'hôte

Il est également possible de spécifier plusieurs hôtes avec des options spécifiques pour chacun d'entre eux. Pour ceci, veuillez les répertorier sur la même ligne en tant que liste séparée par des espaces, en veillant à ce que chaque nom d'hôte soit bien suivi par ses options respectives (entre parenthèses), comme suit :

```
export host1(options1) host2(options2) host3(options3)
```

Pour obtenir des informations sur les différentes méthodes de spécification de noms d'hôtes, veuillez consulter la [Section 8.7.4, « Formats des noms d'hôtes »](#).

Sous sa forme la plus simple, le fichier `/etc/exports` spécifie uniquement le répertoire exporté et les hôtes autorisés à y accéder, comme dans l'exemple suivant :

Exemple 8.6. Fichier `/etc/exports`

```
/exported/directory bob.example.com
```

Ici, **bob.example.com** peut monter `/exported/directory/` à partir du serveur NFS. Comme aucune autre fonction n'est spécifiée dans cet exemple, NFS utilisera les paramètres *par défaut*.

Les paramètres par défaut sont les suivants :

ro

Le système de fichiers exporté est accessible en lecture seule. Les hôtes distants ne peuvent pas modifier les données partagées sur le système de fichiers. Pour autoriser des hôtes à effectuer des modifications sur le système de fichiers (par exemple, lecture/écriture), veuillez spécifier l'option **rw**.

sync

Le serveur NFS ne répondra pas aux requêtes effectuées avant que les changements demandés par les requêtes précédentes soient écrits sur disque. Sinon, pour activer les écritures asynchrones, veuillez spécifier l'option **async**.

wdelay

Le serveur NFS retardera l'écriture sur disque s'il suspecte qu'une autre requête d'écriture est imminente. Ceci peut améliorer les performances car il y a une réduction du nombre d'accès au disque par le biais de commandes d'écriture séparées, réduisant ainsi également l'alourdissement des écritures. Pour désactiver ce comportement, veuillez spécifier **no_wdelay**. **no_wdelay** est uniquement disponible si l'option par défaut **sync** est également spécifiée.

root_squash

Ceci empêche les utilisateurs root connectés à *distance* (et non locaux) d'avoir des privilèges root. Au lieu de cela, le serveur NFS leurs assignera l'ID d'utilisateur **nfsnobody**. Ceci « écrasera » (ou réduira) les capacités de l'utilisateur root à celles d'un utilisateur local du plus bas niveau possible, empêchant ainsi toute écriture non autorisée sur le serveur distant. Pour désactiver l'écrasement root, veuillez spécifier **no_root_squash**.

Pour écraser tous les utilisateurs distants (y compris les utilisateurs root), utilisez **all_squash**. Pour spécifier les ID des groupes et des utilisateurs que le serveur NFS devrait assigner à des utilisateurs distants à partir d'un hôte en particulier, veuillez utiliser les options **anonuid** et **anongid** respectives, comme suit :

```
export host(anonuid=uid,anongid=gid)
```

uid et *gid* sont les numéros d'ID d'utilisateur et les numéros d'ID de groupe respectifs. Les options **anonuid** et **anongid** vous permettent de créer un compte utilisateur et groupe spécifique que les utilisateurs NFS distants peuvent partager.

Par défaut, les listes de contrôle d'accès ou *ACL* (*access control lists*) sont prises en charge par NFS dans Red Hat Enterprise Linux. Pour désactiver cette fonctionnalité, veuillez spécifier l'option **no_acl** lorsque vous exportez le système de fichiers.

Chaque valeur par défaut de chaque système de fichiers exporté doit être explicitement remplacée. Par exemple, si l'option **rw** n'est pas spécifiée, alors le système de fichiers exporté est partagé en lecture seule. Ci-dessous figure un exemple de ligne de **/etc/exports** qui remplace deux options par défaut :

```
/another/exported/directory 192.168.0.3(rw,async)
```

Dans cet exemple, **192.168.0.3** peut monter **/another/exported/directory/** en lecture/écriture et toutes les écritures sur disque seront asynchrones. Pour obtenir davantage d'informations sur les options d'export, veuillez consulter **man exports**.

D'autres options sont disponibles lorsqu'aucune valeur par défaut n'est spécifiée. Celles-ci incluent la possibilité de désactiver la vérification de sous-arborescence, autorise l'accès à partir de ports non-sécurisés et autorise les verrouillages non-sécurisés (nécessaires pour certaines implémentations de clients NFS plus anciennes). Veuillez consulter **man exports** pour obtenir davantage de détails sur ces options moins souvent utilisées.



Important

Le format du fichier **/etc/exports** est très précis, particulièrement au regard de l'utilisation du caractère espace. Rappelez-vous de toujours séparer les systèmes de fichiers exportés des hôtes et les hôtes les uns des autres avec un caractère espace. Toutefois, il ne devrait pas y avoir d'autres caractères espace dans le fichier, sauf sur les lignes de commentaire.

Par exemple, les deux lignes suivantes n'ont pas la même signification :

```
/home bob.example.com(rw)
/home bob.example.com (rw)
```

La première ligne autorise aux utilisateurs de **bob.example.com** d'accéder en lecture/écriture au répertoire **/home**. La seconde ligne autorise les utilisateurs de **bob.example.com** de monter le répertoire en tant que lecture seule (par défaut), tandis que tout monde peut le monter en lecture/écriture.

8.7.2. Commande **exports**

Chaque système de fichiers exporté vers des utilisateurs distants avec NFS, ainsi que le niveau d'accès de ces systèmes de fichiers, sont répertoriés dans le fichier **/etc/exports**. Lorsque le service **nfs** démarre, la commande **/usr/sbin/exports** lance et lit ce fichier, passe le contrôle à **rpc.mountd** (si NFSv2 ou NFSv3 est utilisé) pour le processus de montage, puis à **rpc.nfsd** où les systèmes de fichiers seront ensuite disponibles aux utilisateurs distants.

Lorsqu'exécutée normalement, la commande **/usr/sbin/exports** permet à l'utilisateur root d'exporter ou d'annuler l'exportation des répertoires de manière sélective, sans redémarrer le service NFS. Lorsque les bonnes options sont passées, la commande **/usr/sbin/exports** écrit les systèmes de fichiers exportés sur **/var/lib/nfs/xtab**. Comme **rpc.mountd** fait référence au fichier **xtab** lors de la décision d'octroi des privilèges d'accès à un système de fichiers, tout changement apporté à la liste des systèmes de fichiers exportés prendra effet immédiatement.

Ci-dessous figure une liste des options couramment utilisées disponibles pour **/usr/sbin/exports** :

-r

Cause à tous les répertoires répertoriés dans `/etc/exports` d'être exportés en construisant une nouvelle liste d'exports dans `/etc/lib/nfs/xtab`. Cette option réactualise la liste des exports avec les changements apportés à `/etc/exports`.

-a

Cause à tous les répertoires d'être exportés ou annule leur export, en fonction des autres options passées à `/usr/sbin/exportfs`. Si aucune autre option n'est spécifiée, `/usr/sbin/exportfs` exportera tous les systèmes de fichiers spécifiés dans `/etc/exports`.

-o file-systems

Spécifie les répertoires à exporter qui ne sont pas répertoriés dans `/etc/exports`. Remplacez *file-systems* par les systèmes de fichiers supplémentaires à exporter. Ces systèmes de fichiers doivent être formatés de la même manière qu'ils sont spécifiés dans `/etc/exports`. Cette option est souvent utilisée pour tester un système de fichiers exporté avant de l'ajouter de manière permanente à la liste des systèmes de fichiers devant être exportés. Veuillez consulter [Section 8.7.1, « Fichier de configuration /etc/exports »](#) pour obtenir des informations supplémentaires sur la syntaxe `/etc/exports`.

-i

Ignore `/etc/exports` ; seules les options passées avec la ligne de commande sont utilisées pour définir les systèmes de fichiers exportés.

-u

Annule l'export de tous les répertoires partagés. La commande `/usr/sbin/exportfs -ua` suspend le partage NFS tout en laissant les démons NFS fonctionner. Pour réactiver le partage NFS, veuillez utiliser `exportfs -r`.

-v

Opération détaillée, les systèmes de fichiers dont l'export ou l'annulation de l'export est en cours sont affichés avec plus de détails lorsque la commande `exportfs` est exécutée.

Si aucune option n'est passée avec la commande `exportfs`, celle-ci affichera une liste des systèmes de fichiers actuellement exportés. Pour obtenir des informations supplémentaires sur la commande `exportfs`, veuillez consulter `man exportfs`.

8.7.2.1. Utiliser `exportfs` avec NFSv4

Avec Red Hat Enterprise Linux 7, aucune étape supplémentaire n'est requise pour configurer les exports NFSv4 car tous les systèmes de fichiers mentionnés sont automatiquement disponibles aux clients NFSv3 et NFSv4 qui utilisent le même chemin. Ce n'était pas le cas dans les versions précédentes.

Pour empêcher aux clients d'utiliser NFSv4, veuillez l'éteindre en définissant `RPCNFSDARGS= -N 4` à `/etc/sysconfig/nfs`.

8.7.3. Exécuter NFS derrière un pare-feu

NFS requiert `rpcbind`, qui assigne dynamiquement des ports pour les services RPC et peut provoquer des problèmes avec la configuration des règles de pare-feu. Pour autoriser les clients à accéder aux partages NFS derrière un pare-feu, veuillez modifier le fichier `/etc/sysconfig/nfs` pour contrôler les ports sur lesquels les services RPC seront exécutés.

Le fichier `/etc/sysconfig/nfs` n'existe pas par défaut sur tous les systèmes. Si `/etc/sysconfig/nfs` n'existe pas, créez le et ajoutez les variables suivantes. Remplacez `port` par un numéro de port non utilisé. Si `/etc/sysconfig/nfs` existe déjà, modifiez les entrées selon les besoins :

RPCMOUNTDOPTS="-p port"

Cela ajoute "-p port" à la ligne de commande `rpc.mount` : `rpc.mount -p port`.

LOCKD_TCPPOINT=port

Cela définit le port TCP pour `nlockmgr`.

LOCKD_UDPOINT=port

Cela définit le port UDP pour `nlockmgr`.

Si NFS ne démarre pas, vérifiez `/var/log/messages`. Normalement, NFS échoue au démarrage si vous indiquez un numéro de port déjà utilisé. Une fois que vous aurez édité `/etc/sysconfig/nfs`, il vous faudra redémarrer le service `nfs-config` pour que les nouvelles valeurs puissent prendre effet dans Red Hat Enterprise Linux 7.2 et version antérieures, en exécutant :

```
# systemctl restart nfs-config
```

Puis, redémarrez le serveur NFS.

```
# systemctl restart nfs-server
```

Exécutez `rpcinfo -p` pour confirmer que les changements ont pris place.



Note

Pour autoriser les rappels NFSv4.0 à passer à travers les pare-feux, paramétrez `/proc/sys/fs/nfs/nfs_callback_tcpport` et autorisez le serveur à se connecter à ce port sur le client.

Ce processus n'est pas nécessaire pour NFSv4.1 ou ses versions supérieures, et les autres ports de `mountd`, `statd`, et `lockd` ne sont pas requis dans un environnement NFSv4 pure.

8.7.3.1. Découverte des exports NFS

Il existe deux manières de découvrir quels systèmes de fichiers sont exportés par un serveur NFS.

Premièrement, sur n'importe quel serveur qui prend en charge NFSv2 ou NFSv3, veuillez utiliser la commande `showmount` :

```
$ showmount -e myserver
Export list for myserver
/exports/foo
/exports/bar
```

Deuxièmement, sur n'importe quel serveur qui prend en charge NFSv4, montez le répertoire `/` et observez son contenu.

```
# mount myserver:/ /mnt/  
#cd /mnt/  
exports  
# ls exports  
foo  
bar
```

Sur les serveurs prenant en charge NFSv4 et soit NFSv2, soit NFSv3, les deux méthodes fonctionneront et offriront les mêmes résultats.



Note

Sur les anciens serveurs NFS, avant Red Hat Enterprise Linux 6 et en fonction de leur configuration, il était possible d'exporter des systèmes de fichiers sur des clients NFSv4 sur différents chemins. Comme ces serveurs n'activent pas NFSv4 par défaut, ceci ne devrait pas poser de problème.

8.7.4. Formats des noms d'hôtes

L'hôte peut se trouver sous les formats suivants :

Machine unique

Un nom de domaine complet (pouvant être résolu par le serveur), un nom d'hôte (pouvant être résolu par le serveur), ou une adresse IP.

Série d'ordinateurs spécifiés avec des caractères génériques

Utilisez le caractère `*` ou `?` pour spécifier une chaîne correspondante. Les caractères génériques ne doivent pas être utilisés avec les adresses IP. Cependant, cela peut fonctionner accidentellement si des recherches DNS inversées échouent. Lors de la spécification de caractères génériques dans des noms de domaine complets, les points (`.`) ne sont pas inclus dans la recherche. Par exemple, `*.example.com` inclut `one.example.com` mais n'inclut pas `one.two.example.com`.

Réseaux IP

Utilisez `a.b.c.d/z`, où `a.b.c.d` est le réseau et `z` est le nombre d'octets dans le masque réseau (par exemple, `192.168.0.0/24`). Un autre format acceptable est `a.b.c.d/netmask`, où `a.b.c.d` est le réseau et `netmask` est le masque réseau (par exemple, `192.168.100.8/255.255.255.0`).

Netgroups

Veuillez utiliser le format `@group-name`, où `group-name` est le nom du netgroup NIS.

8.7.5. NFS sur RDMA

Dans Red Hat Enterprise Linux 7, le service RDMA est automatique tant qu'il y a du matériel compatible avec RDMA présent dans la machine. Ainsi, la procédure ne devra être suivie que si le package RDMA suivant n'a pas été installé lors de l'installation de la machine.

Procédure 8.1. Activez RDMA à partir du client

1. Installez le package RDMA :

```
# yum install rdma
```

2. Recréer **initramfs** :

```
# dracut -f
```

3. Démarrez à nouveau.

8.8. Sécurisation de NFS

NFS est bien adapté au partage de systèmes de fichiers entiers avec un grand nombre d'hôtes connus de manière transparente. Cependant, cette facilité d'utilisation entraîne toute une variété de problèmes de sécurité potentiels. Prenez en considération les sections suivantes lorsque vous exportez des systèmes de fichiers NFS sur un serveur ou lorsque vous les montez sur un client. Cela minimisera les risques de sécurité NFS et protégera mieux les données sur le serveur.

8.8.1. Sécurité NFS avec AUTH_SYS et les contrôles d'export

Traditionnellement, NFS offrait deux options pour contrôler l'accès aux fichiers exportés.

Premièrement, le serveur contrôle quels hôtes sont autorisés à monter quels systèmes de fichiers, soit par adresse IP ou par nom d'hôte.

Puis le serveur applique les permissions du système de fichiers pour les utilisateurs des clients NFS de la même manière que pour les utilisateurs locaux. Traditionnellement, ceci est fait à l'aide d'**AUTH_SYS** (aussi appelé **AUTH_UNIX**), qui se fie au client pour indiquer l'UID et le GID de l'utilisateur. Soyez conscient que cela signifie qu'un client mal configuré ou malicieux pourrait facilement mal comprendre ceci et autoriser un utilisateur à accéder à des fichiers auxquels il ne devrait pas avoir accès.

Pour limiter les risques potentiels, les administrateurs autorisent souvent l'accès en lecture seule ou limitent les permissions utilisateur à un utilisateur et un ID de groupe communs. Malheureusement, ces solutions empêchent le partage NFS d'être utilisé comme prévu à l'origine.

En outre, si une personne mal intentionnée prenait contrôle du serveur DNS utilisé par le système exportant le système de fichiers NFS, le système associé à un nom d'hôte ou à un nom de domaine complet peut être dirigé vers un ordinateur non autorisé. À ce moment, l'ordinateur non autorisé *devient* le système autorisé à monter le partage NFS, puisqu'aucune information sur le nom d'utilisateur ou sur le mot de passe n'est échangée pour fournir une sécurité supplémentaire au montage NFS.

Les caractères génériques doivent être utilisés avec précaution lors de l'export de répertoires à travers NFS, car il est possible que l'étendue du caractère générique puisse englober davantage de systèmes que prévu.

Il est également possible de restreindre l'accès au service **rpcbind**^[1] avec des enveloppes TCP. La création de règles avec **iptables** peut également limiter l'accès aux ports utilisés par **rpcbind**, **rpc.mountd**, et **rpc.nfsd**.

Pour obtenir des informations supplémentaires sur la sécurisation de NFS et **rpcbind**, veuillez consulter **man iptables**.

8.8.2. Sécurité NFS avec AUTH_GSS

La sortie de NFSv4 a provoqué une révolution dans la sécurité NFS en mandatant l'implémentation de RPCSEC_GSS et du mécanisme GSS-API Kerberos version 5. Cependant, RPCSEC_GSS et le mécanisme Kerberos sont aussi disponibles pour toutes les versions de NFS. En mode FIPS, vous ne pouvez utiliser que les algorithmes autorisés par FIPS.

Avec le mécanisme Kerberos RPCSEC_GSS, le serveur ne dépend plus du client pour correctement représenter les utilisateurs qui accèdent au fichier, tout comme avec AUTH_SYS. Au lieu de cela, un chiffrement est utilisé pour authentifier les utilisateurs sur le serveur, empêchant ainsi à un client mal intentionné de prendre la place d'un utilisateur sans posséder les informations d'utilisation Kerberos de cet utilisateur. Il est également plus facile ainsi de sécuriser les montages sécurisés, puisqu'après que la configuration Kerberos a eu lieu, cela fonctionne sans modification supplémentaire.



Note

On considère qu'un serveur d'octroiement de ticket Kerberos (KDC) est correctement installé et configuré avant de procéder à la configuration du serveur NFSv4. Kerberos est un système d'authentification réseau qui permet aux clients et serveurs de s'authentifier à travers l'utilisation d'un chiffrement symétrique et d'une tierce partie de confiance, KDC. Pour obtenir des informations supplémentaires sur Kerberos, veuillez consulter [Using Kerberos](#) dans le guide *System-Level Authentication Guide*.

Pour paramétrer RPCSEC_GSS, veuillez utiliser la procédure suivante :

Procédure 8.2. Paramétrer RPCSEC_GSS

1. Créez les principaux `nfs/client.mydomain@MYREALM` et `nfs/server.mydomain@MYREALM`.
2. Ajoutez les clés correspondant aux onglets principaux pour le client et le serveur.
3. Du côté serveur, veuillez ajouter `sec=krb5:krb5i:krb5p` à l'export. Pour continuer à autoriser AUTH_SYS, veuillez ajouter `sec=krb5:krb5i:krb5p` à la place.
4. Du côté client, veuillez ajouter `sec=krb5` (ou `sec=krb5i`, ou `sec=krb5p`, en fonction de l'installation) aux options de montage.

Pour obtenir des informations supplémentaires, comme les différences entre `krb5`, `krb5i`, et `krb5p`, veuillez consulter les pages man `exports` et `nfs` ou la [Section 8.5, « Options de montage NFS courantes »](#).

Pour obtenir des informations supplémentaires sur le cadre de RPCSEC_GSS, y compris sur la manière par laquelle `rpc.svcgssd` et `rpc.gssd` interagissent, veuillez consulter <http://www.citi.umich.edu/projects/nfsv4/gssd/>.

8.8.2.1. Sécurité NFS avec NFSv4

NFSv4 inclut la prise en charge des ACL basée sur le modèle Microsoft Windows NT, et non sur le modèle POSIX, à cause des fonctionnalités et du déploiement global de ce dernier.

Une autre fonctionnalité importante de NFSv4 est la suppression de l'utilisation du protocole `MOUNT` pour monter les systèmes de fichiers. Ce protocole présentait des failles de sécurité possibles à cause de la manière par laquelle il traitait les identificateurs de fichiers.

8.8.3. Permissions de fichier

Une fois que le système de fichiers NFS est monté en lecture/écriture par un hôte distant, la seule protection que chaque fichier partagé possède sont ses permissions. Si deux utilisateurs partageant la même valeur

d'ID d'utilisateur montent le même système de fichiers NFS, ils pourront chacun modifier les fichiers de l'autre. En outre, toute personne connectée en tant que root sur le système client peut utiliser la commande **su** - pour accéder à tous les fichiers du partage NFS.

Par défaut, les listes de contrôle d'accès (ACL) sont prises en charge par NFS sur Red Hat Enterprise Linux. Red Hat recommande de laisser cette fonctionnalité activée.

Par défaut, NFS utilise *root squashing* (« Écrasement root ») lors de l'exportation d'un système de fichiers. Ceci définit l'ID d'utilisateur de tout utilisateur root local accédant au partage NFS en tant que **nobody** (personne). Le root squashing est contrôlé par l'option par défaut **root_squash**. Pour obtenir des informations supplémentaires sur cette option, veuillez consulter [Section 8.7.1, « Fichier de configuration /etc/exports »](#). Dans la mesure du possible, ne désactivez jamais le root squashing.

Lors de l'exportation d'un partage NFS en lecture seule, veuillez considérer l'utilisation de l'option **all_squash**. Cette option fait que tout utilisateur accédant au système de fichiers exporté prendra l'ID utilisateur de l'utilisateur **nfsnobody**.

8.9. NFS et rpcbind



Note

La section suivante s'applique uniquement aux implémentations NFSv3 qui nécessitent le service **rpcbind** pour une compatibilité ascendante.

L'utilitaire **rpcbind**^[1] mappe les services RPC aux ports qu'ils écoutent. Les processus RPC notifient **rpcbind** lorsqu'ils démarrent, en enregistrant les ports qu'ils écoutent et les numéros de programme RPC qu'ils prévoient de servir. Le système client contacte ensuite **rpcbind** sur le serveur avec un numéro de programme RPC particulier. Le service **rpcbind** redirige le client vers le numéro de port correct afin de pouvoir communiquer avec le service requis.

Comme les services basés RPC se fient à **rpcbind** pour effectuer toutes les connexions avec les requêtes client entrantes, **rpcbind** doit être disponible avant que ces services ne démarrent.

Le service **rpcbind** utilise des enveloppes TCP pour le contrôle d'accès, les règles de contrôle d'accès de **rpcbind** affectent *tous* les services basés RPC. De manière alternative, il est possible de spécifier les règles de contrôle d'accès de chacun des démons RPC NFS. Les pages **man** de **rpc.mountd** et **rpc.statd** contiennent des informations concernant la syntaxe précise de ces règles.

8.9.1. Résolution des problèmes NFS et rpcbind

Comme **rpcbind**^[1] fournit la coordination entre les services RPC et les numéros de port utilisés pour communiquer avec ceux-ci, il est utile d'afficher le statut des services RPC en cours à l'aide de **rpcbind** lors des résolutions de problèmes. La commande **rpcinfo** affiche chaque service basé RPC avec les numéros de port, un numéro de programme RPC, un numéro de version, et un type de protocole IP (TCP ou UDP).

Pour vous assurer que les bons services NFS basés RPC soient activés pour **rpcbind**, veuillez saisir la commande suivante :

```
# rpcinfo -p
```

Exemple 8.7. Sortie de la commande `rpcinfo -p`

Ci-dessous figure un exemple de sortie de cette commande :

```

program vers proto  port  service
    100021    1    udp  32774 nlockmgr
    100021    3    udp  32774 nlockmgr
    100021    4    udp  32774 nlockmgr
    100021    1    tcp  34437 nlockmgr
    100021    3    tcp  34437 nlockmgr
    100021    4    tcp  34437 nlockmgr
    100011    1    udp   819  rquotad
    100011    2    udp   819  rquotad
    100011    1    tcp   822  rquotad
    100011    2    tcp   822  rquotad
    100003    2    udp  2049  nfs
    100003    3    udp  2049  nfs
    100003    2    tcp  2049  nfs
    100003    3    tcp  2049  nfs
    100005    1    udp   836  mountd
    100005    1    tcp   839  mountd
    100005    2    udp   836  mountd
    100005    2    tcp   839  mountd
    100005    3    udp   836  mountd
    100005    3    tcp   839  mountd

```

Si l'un des services NFS ne démarre pas correctement, **rpcbind** sera incapable de mapper les requêtes RPC des clients de ce service sur le bon port. Dans de nombreux cas, si NFS n'est pas présent dans la sortie **rpcinfo**, redémarrer NFS entraînera le bon enregistrement du service avec **rpcbind** et le début de son fonctionnement.

Veuillez consulter la page **man rpcinfo** pour obtenir des informations supplémentaires ainsi qu'une liste de ses options.

8.10. Références

Administrer un serveur NFS peut être un défi. Beaucoup d'options, dont un grand nombre ne sont pas mentionnées dans ce chapitre, sont disponibles pour exporter ou monter des partages NFS. Veuillez consulter les sources suivantes pour obtenir davantage d'informations.

Documentation installée

- ✦ **man mount** — offre une vue d'ensemble complète sur les options de montage pour les configurations du serveur NFS et du client NFS.
- ✦ **man fstab** — fournit des détails sur le format du fichier `/etc/fstab` utilisé pour monter les systèmes de fichiers lors du démarrage.
- ✦ **man nfs** — fournit des détails sur les options de montage et d'exportation des systèmes de fichiers spécifiques à NFS.
- ✦ **man exports** — affiche les options courantes utilisées dans le fichier `/etc/exports` lors de l'exportation de systèmes de fichiers NFS.

Sites Web utiles

- ✦ <http://linux-nfs.org> — Site pour développeurs sur lequel les mises à jour du statut des projets peuvent être observées.
- ✦ <http://nfs.sourceforge.net/> — Ancien site pour développeurs, celui-ci contient toujours de nombreuses informations utiles.
- ✦ <http://www.citi.umich.edu/projects/nfsv4/linux/> — Ressource NFSv4 pour noyau Linux 2.6.
- ✦ <http://www.vanemery.com/Linux/NFSv4/NFSv4-no-rpcsec.html> — Décrit les détails de NFSv4 avec Fedora Core 2, qui inclut le noyau 2.6.
- ✦ <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.111.4086> — Excellent livre blanc sur les fonctionnalités et améliorations du protocole NFS Version 4.

Livres sur le sujet

- ✦ *Managing NFS and NIS* (« Gérer NFS et NIS ») de Hal Stern, Mike Eisler, et Ricardo Labiaga; O'Reilly & Associés — Excellent guide de référence pour les nombreuses options de montage et d'exportation de NFS disponible.
- ✦ *NFS Illustrated* (« NFS illustré ») de Brent Callaghan; Addison-Wesley Publishing Company — Fournit des comparaisons de NFS avec d'autres systèmes de fichiers réseau et affiche, de manière détaillée, comment les communications NFS se produisent.

[1] The **rpcbind** service replaces **portmap**, which was used in previous versions of Red Hat Enterprise Linux to map RPC program numbers to IP address port number combinations. For more information, refer to [Section 8.1.1](#), « Services requis ».

Chapitre 9. FS-Cache

FS-Cache est un cache local persistant qui peut être utilisé par les systèmes de fichiers pour prendre les données récupérées sur le réseau et les mettre en cache sur le disque local. Ceci permet de minimiser le trafic réseau pour les utilisateurs accédant aux données à partir d'un système de fichiers monté sur le réseau (par exemple NFS).

Le diagramme suivant est une illustration de haut niveau du fonctionnement de FS-Cache :

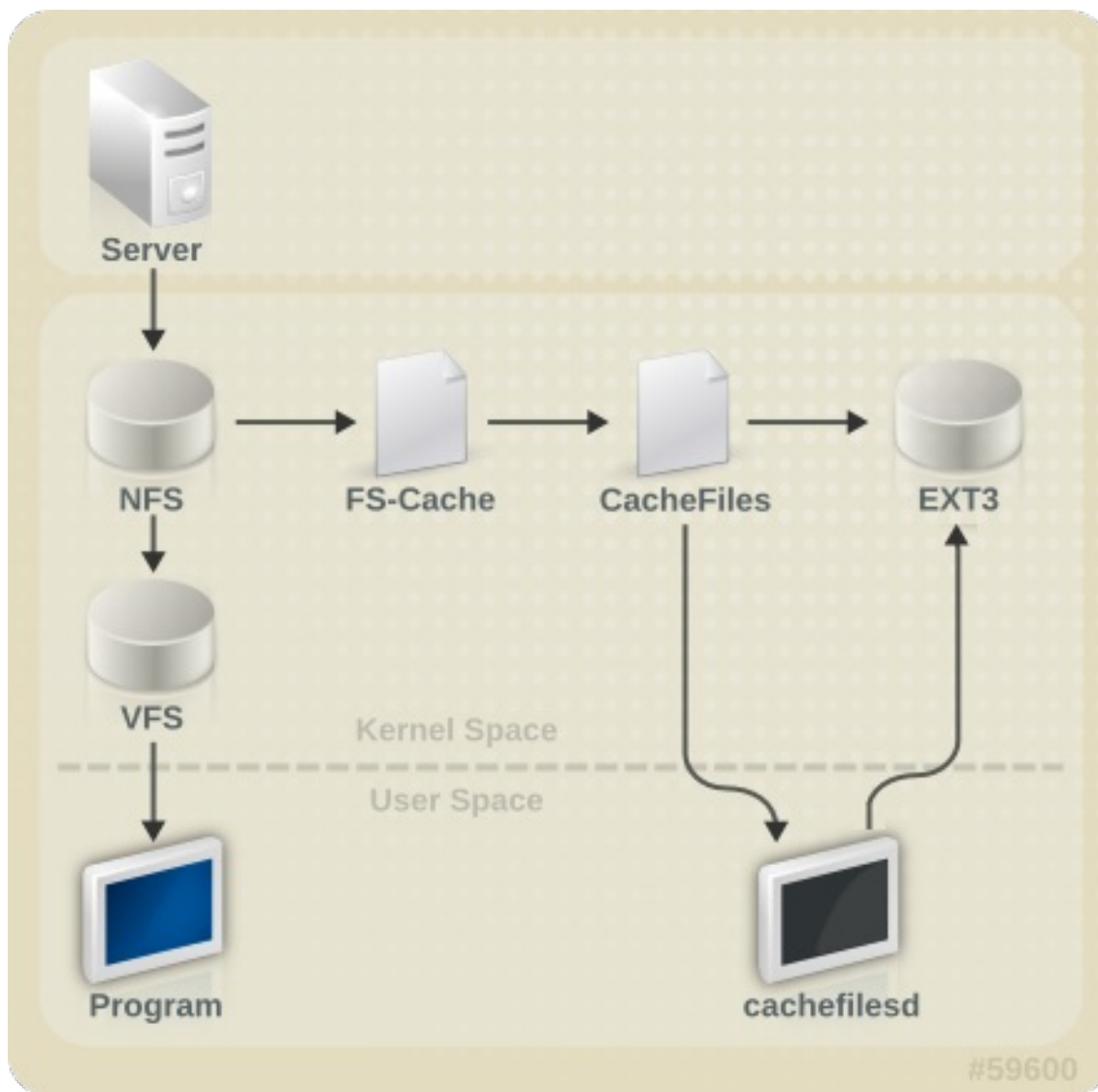


Figure 9.1. Aperçu de FS-Cache

FS-Cache est conçu pour être aussi transparent que possible aux utilisateurs et administrateurs d'un système. Contrairement à **cacheefs** sur Solaris, FS-Cache permet à un système de fichiers d'interagir directement avec le cache local d'un client sans créer de système de fichiers surmonté. Avec NFS, une option de montage ordonne au client de monter le partage NFS avec FS-cache activé.

FS-Cache n'altère pas les opérations de base d'un système de fichiers qui fonctionne sur le réseau - il offre simplement à ce système de fichiers un emplacement permanent dans lequel ses données peuvent être mises en cache. Par exemple, un client peut toujours monter un partage NFS, que FS-Cache soit activé ou non. En outre, NFS caché peut gérer des fichiers qui sont trop grands pour le cache (que ce soit le cas individuellement ou collectivement), car les fichiers peuvent être mis en cache de manière partielle et n'ont pas besoin d'être lus directement. FS-Cache cache aussi toutes les erreurs d'E/S se produisant dans le cache provenant du pilote du système de fichiers client.

Pour fournir ses services de mise en cache, FS-Cache a besoin d'un *backend de cache*. Un backend de cache est un pilote de stockage configuré pour offrir des services de mise en cache (c'est-à-dire des **cachefiles**). Dans ce cas, FS-Cache requiert un système de fichiers monté basé sur blocs, qui prendrait en charge **bmap** et des attributs étendus (par exemple ext3) en tant que backend de cache.

FS-Cache ne peut pas mettre en cache n'importe quel système de fichiers arbitrairement, que ce soit via le réseau ou autrement : le pilote du système de fichiers partagé doit être altéré afin de permettre des interactions avec FS-Cache, le stockage ou le récupération de données, et le paramétrage et la validation de métadonnées. FS-Cache a besoin de *clés d'indexation* et de *données de cohérence* du système de fichiers mis en cache afin de prendre en charge la persistance : les clés d'indexation pour faire correspondre les objets de systèmes de fichiers avec les objets du cache, et les données de cohérence pour déterminer sur les objets du cache sont toujours valides.



Note

Dans Red Hat Enterprise Linux 7, **cachefilesd** n'est pas installé par défaut et doit donc être installé manuellement.

9.1. Garantie des performances

FS-Cache *ne garantit pas* de performances améliorées, cependant, des performances cohérentes sont assurées en évitant la congestion du réseau. L'utilisation d'un backend de cache provoque des pénalités de performances : par exemple, des partages NFS mis en cache ajoutent des accès du disque aux recherches à travers plusieurs réseaux. Malgré le fait que FS-Cache tente autant que possible de rester asynchrone, certains chemins seront synchronisés (par exemple les lectures) lorsque cela n'est pas possible.

Par exemple, l'utilisation de FS-Cache pour mettre en cache un partage NFS entre deux ordinateurs sur un réseau GigE vide ne démontrera aucune amélioration des performances sur l'accès aux fichiers. En revanche, les requêtes NFS seront satisfaites plus rapidement à partir de la mémoire serveur qu'à partir du disque local.

Ainsi, l'utilisation de FS-Cache est un *compromis* entre divers facteurs. Si par exemple, FS-Cache est utilisé pour mettre le trafic NFS en cache, le client pourrait en être quelque peu ralenti, mais la charge du réseau et du serveur seront très fortement réduites en satisfaisant les requêtes de lecture localement, sans consommer de bande réseau.

9.2. Paramétrer un cache

Actuellement, Red Hat Enterprise Linux 7 fournit uniquement le backend de mise en cache **cachefiles**. Le démon **cachefilesd** initialise et gère **cachefiles**. Le fichier **/etc/cachefilesd.conf** contrôle la manière par laquelle **cachefiles** fournit ses services de mise en cache. Pour configurer un backend de cache de ce type, le paquet **cachefilesd** doit être installé.

Le premier paramètre à configurer dans un backend de cache est le répertoire à utiliser en tant que cache. Pour le configurer, veuillez utiliser le paramètre suivant :

```
$ dir /path/to/cache
```

Habituellement, le répertoire du backend du cache est défini dans **/etc/cachefilesd.conf** en tant que **/var/cache/fscache**, comme suit :

```
$ dir /var/cache/fscache
```

FS-Cache stockera le cache dans le système de fichiers qui héberge */path/to/cache*. Sur un ordinateur portable, il est conseillé d'utiliser le système de fichiers root (/) en tant que système de fichier hôte, mais pour un ordinateur de bureau, il serait plus prudent de monter une partition de disque spécifiquement pour le cache.

Les systèmes de fichiers qui prennent en charge les fonctionnalités requises par le backend de cache FS-Cache incluent les implémentations Red Hat Enterprise Linux 7 des systèmes de fichiers suivants :

- ✦ ext3 (avec des attributs étendus activés)
- ✦ ext4
- ✦ BTRFS
- ✦ XFS

Le système de fichiers hôte doit prendre en charge les attributs étendus définis par l'utilisateur ; FS-Cache utilise ces attributs pour stocker les informations de maintenance de cohérence. Pour activer les attributs étendus définis par l'utilisateur pour les systèmes de fichiers ext3 (c'est-à-dire **device**), veuillez utiliser :

```
# tune2fs -o user_xattr /dev/device
```

Alternativement, des attributs étendus pour un système de fichiers peuvent être activés au moment du montage, comme suit :

```
# mount /dev/device /path/to/cache -o user_xattr
```

Le backend du cache fonctionne en maintenant une certaine quantité d'espace libre sur la partition hébergeant le cache. Il agrandit et réduit le cache en réponse aux autres éléments du système utilisant de l'espace libre, le rendant ainsi sûr pour une utilisation sur le système de fichiers root (par exemple sur un ordinateur portable). FS-Cache définit les valeurs par défaut sur ce comportement, qui peut être configuré via les *limites d'élimination de cache* (« *cache cull limits* »). Pour obtenir des informations supplémentaires sur la configuration des limites d'élimination de cache, veuillez consulter [Section 9.4, « Définir les limites d'élimination du cache »](#).

Une fois le fichier de configuration en place, lancez le démon **cachefilesd** :

```
# service cachefilesd start
```

Pour configurer **cachefilesd** pour être lancé lors du démarrage, veuillez exécuter la commande suivante en tant que root :

```
# chkconfig cachefilesd on
```

9.3. Utiliser le cache avec NFS

NFS n'utilisera pas le cache à moins d'en avoir reçu l'instruction spécifique. Pour configurer un montage NFS afin d'utiliser FS-Cache, veuillez inclure l'option **-o fsc** à la commande **mount** :

```
# mount nfs-share:/ /mount/point -o fsc
```

Tous les accès aux fichiers sous */mount/point* passeront par le cache, sauf si le fichier est ouvert pour des E/S ou écritures directes (veuillez consulter [Section 9.3.2, « Limitations des caches avec NFS »](#) pour

obtenir des informations supplémentaires). NFS indexe le contenu du cache à l'aide d'un gestionnaire de fichiers NFS, et *non* avec le nom du fichier ; cela signifie que les fichiers avec des liens physiques partagent le cache correctement.

La mise en cache est prise en charge dans les versions 2, 3 et 4 de NFS. Cependant, chaque version utilise différentes branches pour la mise en cache.

9.3.1. Partage de cache

Il existe plusieurs problèmes potentiels liés au partage de cache NFS. Comme le cache est persistant, des blocs de données dans le cache sont indexés sur une séquence de quatre clés :

- Niveau 1 : Détails du serveur
- Niveau 2 : Certaines options de montage, type de sécurité, FSID, uniquifier
- Niveau 3 : Identificateur de fichier
- Niveau 4 : Numéro de la page dans le fichier

Pour éviter les problèmes de gestion de cohérence entre superblocs, tous les superblocs NFS souhaitant mettre des données en cache possèdent des clés de Niveau 2 uniques. Normalement, deux montages NFS avec le même volume source et les mêmes options partageront un superbloc et partageront ainsi la mise en cache, même s'ils montent différents répertoires à l'intérieur de ce volume.

Exemple 9.1. Partager un cache

Exécutez les deux commandes **mount** suivantes :

```
mount home0:/disk0/fred /home/fred -o fsc
```

```
mount home0:/disk0/jim /home/jim -o fsc
```

Ici, **/home/fred** et **/home/jim** partageront probablement le superbloc car ils possèdent les mêmes options, particulièrement s'ils proviennent du même volume ou de la même partition sur le serveur NFS (**home0**). Prenez en considération les deux commandes de montage suivantes :

```
mount home0:/disk0/fred /home/fred -o fsc,rsize=230
```

```
mount home0:/disk0/jim /home/jim -o fsc,rsize=231
```

Dans ce cas, **/home/fred** et **/home/jim** ne partageront pas le super bloc car leurs paramètres d'accès réseau sont différents, et ceux-ci font partie de la clé du Niveau 2. La même chose est valable pour la séquence de montage suivante :

```
mount home0:/disk0/fred /home/fred1 -o fsc,rsize=230
```

```
mount home0:/disk0/fred /home/fred2 -o fsc,rsize=231
```

Ici, le contenu des deux sous-arborescences (**/home/fred1** et **/home/fred2**) seront mis en cache *deux fois*.

Une autre manière d'éviter le partage de superblocs est de le supprimer de manière explicite avec le paramètre **nosharecache**. Voici un exemple :

```
mount home0:/disk0/fred /home/fred -o nosharecache,fsc
```

```
mount home0:/disk0/jim /home/jim -o nosharecache,fsc
```


Dans ce cas, un superbloc seulement aura la permission d'utiliser le cache puisque rien ne permettra de distinguer les clés de Niveau 2 de **home0:/disk0/fred** et **home0:/disk0/jim**. Pour répondre à ce problème, veuillez ajouter un *identifiant unique* sur un des montages au moins, c-a-d **fsc=unique-identifier**. Par exemple :

```
mount home0:/disk0/fred /home/fred -o nosharecache,fsc
```

```
mount home0:/disk0/jim /home/jim -o nosharecache,fsc=jim
```

Ici, l'identifiant unique **jim** sera ajouté à la clé de Niveau 2 utilisée dans le cache pour **/home/jim**.

9.3.2. Limitations des caches avec NFS

Ouvrir un fichier à partir d'un système de fichiers partagé pour des E/S directes contournera automatiquement le cache. Cela se produit car ce type d'accès doit être effectué directement sur le serveur.

Ouvrir un fichier à partir d'un système de fichiers partagé pour écriture ne fonctionnera pas sur NFS version 2 et 3. Les protocoles de ces versions n'offrent pas suffisamment d'informations sur la gestion de la cohérence pour que le client puisse détecter une écriture simultanée sur le même fichier, mais provenant d'un autre client.

Ainsi, ouvrir un fichier à partir d'un système de fichiers partagé pour des E/S directes ou pour écriture videra la copie du cache du fichier. FS-Cache ne mettra pas le fichier en cache à nouveau jusqu'à ce qu'il ne soit plus ouvert pour des E/S directes ou des écritures.

En outre, cette version de FS-Cache met uniquement en cache les fichiers NFS normaux. FS-Cache ne mettra pas en cache les répertoires, liens symboliques, fichiers périphériques, FIFO et sockets.

9.4. Définir les limites d'élimination du cache

Le démon **cachefilesd** fonctionne en mettant en cache des données à distance provenant des systèmes de fichiers pour libérer de l'espace sur le disque. Cela pourrait potentiellement consommer tout l'espace libre, ce qui serait problématique si le disque héberge aussi la partition root. Pour contrôler ceci, **cachefilesd** tente de maintenir une certaine quantité d'espace libre en abandonnant certains anciens objets (c'est-à-dire qui n'ont pas été accédés depuis un certain temps) du cache. Ce comportement est aussi connu sous le nom de *cache culling* (élimination du cache).

L'élimination du cache se base sur le pourcentage des blocs et le pourcentage des fichiers disponibles dans le système de fichiers sous-jacent. Six limites sont contrôlées par des paramètres dans **/etc/cachefilesd.conf** :

brun N% (pourcentage de blocs), frun N% (pourcentage de fichiers)

Si la quantité d'espace libre et le nombre de fichiers disponibles dans le cache dépassent ces deux limites, alors l'élimination est désactivée.

bcull N% (pourcentage de blocs), fcull N% (pourcentage de fichiers)

Si la quantité d'espace disponible ou le nombre de fichiers dans le cache se trouve sous l'une de ces limites, alors l'élimination est lancée.

bstop N% (pourcentage de blocs), fstop N% (pourcentage de fichiers)

Si la quantité d'espace disponible ou le nombre de fichiers disponibles dans le cache se trouve sous l'une de ces limites, alors l'allocation d'espace disque ou de fichiers ne sera plus permise tant que l'élimination du cache n'aura pas fait que ces limites soient à nouveau dépassées.

La valeur par défaut de **N** pour chaque paramètre est comme suit :

- » **brun/frun** - 10%
- » **bcull/fcull** - 7%
- » **bstop/fstop** - 3%

Lors de la configuration de ces paramètres, ce qui suit doit être vrai (« True ») :

$0 \leq \text{bstop} < \text{bcull} < \text{brun} < 100$

$0 \leq \text{fstop} < \text{fcull} < \text{frun} < 100$

Il s'agit des pourcentages de l'espace et des fichiers disponibles et ceux-ci n'apparaissent pas en tant que 100 moins le pourcentage affiché par le programme **df**.



Important

L'élimination dépend des paires **bxxx** et **fxxx** simultanément ; elles ne peuvent pas être traitées séparément.

9.5. Informations statistiques

FS-Cache conserve aussi des informations statistiques générales. Pour voir ces informations, veuillez utiliser :

```
cat /proc/fs/fscache/stats
```

Les statistiques de FS-Cache incluent des informations sur les points de décision et les compteurs d'objets. Pour obtenir des détails supplémentaires sur les statistiques fournies par FS-Cache, veuillez consulter le document sur le noyau suivant :

```
/usr/share/doc/kernel-doc-version/Documentation/filesystems/caching/fscache.txt
```

9.6. Références

Pour obtenir des informations supplémentaires sur **cachefilesd** et sur la manière de le configurer, veuillez consulter **man cachefilesd** et **man cachefilesd.conf**. Les documents sur le noyau suivants offrent également des informations supplémentaires :

- » **/usr/share/doc/cachefilesd-version-number/README**
- » **/usr/share/man/man5/cachefilesd.conf.5.gz**
- » **/usr/share/man/man8/cachefilesd.8.gz**

Pour obtenir des informations générales sur FS-Cache, y compris des détails sur ses contraintes de conception, les statistiques disponibles et les capacités, veuillez consulter le document sur le noyau suivant :

```
/usr/share/doc/kernel-doc-version/Documentation/filesystems/caching/fscache.txt
```

Partie II. Administration du stockage

La section Administration du stockage commence par des recommandations sur le stockage pour Red Hat Enterprise Linux 7. Des instructions concernant les partitions, la gestion de volumes logiques et les partitions swap s'ensuivent. Puis on abordera les quotas de disques et des systèmes RAID, ainsi que les fonctions des commandes de montage, `volume_key` et `d'acls`. Les réglages SSD, les barrières d'écriture, les limites des E/S et les systèmes sans disque seront aussi abordés. S'ensuivra un chapitre sur le stockage en ligne, puis finalement le mappage de périphériques multivoies et le stockage virtuel.

Veillez utiliser la table des matières suivante pour explorer ces tâches d'administration du stockage.

Chapitre 10. Besoins de stockage à prendre en compte pendant l'installation

De nombreux paramètres de périphérique de stockage et de systèmes de fichiers peuvent uniquement être configurés pendant l'installation. Les autres paramètres, tels que le type de système de fichiers, peuvent uniquement être modifiés jusqu'à un certain moment à partir duquel il faudra effectuer un reformatage. Ainsi, il serait prudent de planifier votre configuration correctement avant d'installer Red Hat Enterprise Linux 7.

Ce chapitre traite des diverses choses à prendre en compte lors de la planification de la configuration du stockage de votre système. Pour des instructions d'installation (y compris la configuration du stockage pendant l'installation), veuillez consulter le *Guide d'installation* fourni par Red Hat.

Pour obtenir des informations sur ce que Red Hat prend en charge officiellement au niveau taille et limites de stockage, référez-vous à l'article <http://www.redhat.com/resourcelibrary/articles/articles-red-hat-enterprise-linux-6-technology-capabilities-and-limits>.

10.1. Considérations particulières

Cette section énumère plusieurs problèmes et facteurs à prendre en considération pour des configurations de stockage particulières.

Partitions séparées pour `/home`, `/opt`, `/usr/local`

S'il est probable que vous mettiez à niveau votre système dans le futur, veuillez placer `/home`, `/opt`, et `/usr/local` sur un périphérique séparé. Ceci vous permettra de reformater les périphériques ou systèmes de fichiers contenant le système d'exploitation tout en préservant vos données utilisateur et applications.

Périphériques DASD et zFCP sur IBM System Z

Sur la plateforme IBM System Z, les périphériques DASD et zFCP sont configurés via le mécanisme CCW (*Channel Command Word*). Les chemins CCW doivent être explicitement ajoutés au système, puis mis en ligne. Pour les périphériques DASD, cela signifie simplement que répertorier les numéros des périphériques (ou les plages des numéros des périphériques) en tant que paramètre **DASD=** dans la ligne de commande de démarrage ou dans un fichier de configuration CMS.

Pour les périphériques zFCP, vous devez répertorier le numéro de périphérique LUN (*logical unit number*) et le WWPN (*world wide port name*). Une fois le périphérique zFCP initialisé, il est mappé à un chemin CCW. Les lignes **FCP_x=** dans la ligne de commande de démarrage (ou dans un fichier de configuration CMS) vous permettent de spécifier ces informations pour l'installateur.

Chiffrer des périphériques blocs avec LUKS

Formater un périphérique bloc pour le chiffrer en utilisant LUKS/**dm-crypt** détruira tout formatage existant sur ce périphérique. Ainsi, vous devriez décider quels périphériques chiffrer avant que la configuration du stockage du nouveau système ne soit activée dans le cadre du processus d'installation.

Métadonnées RAID BIOS périmées

Déplacer un disque d'un système configuré pour RAID micrologiciel *sans* supprimer les métadonnées RAID du disque peut empêcher **Anaconda** de détecter le disque correctement.



Avertissement

La suppression des métadonnées RAID du disque peut potentiellement détruire toutes les données stockées. Red Hat vous recommande de faire une copie de sauvegarde de vos données avant de continuer.

Pour supprimer des métadonnées RAID du disque, veuillez utiliser la commande suivante :

```
dmraid -r -E /device/
```

Pour obtenir plus d'informations sur la gestion des périphériques RAID, veuillez consulter **man dmraid** et le [Chapitre 17, Réseau redondant de disques indépendants \(RAID, de l'anglais « Redundant Array of Independent Disks »\)](#).

Détection et configuration iSCSI

Pour la détection à chaud des disques iSCSI, veuillez les configurer dans le micrologiciel d'un NIC (« *network interface card* ») iBFT capable de démarrer. L'authentification CHAP des cibles iSCSI est prise en charge pendant l'installation. Cependant, la découverte iSNS n'est pas prise en charge pendant l'installation.

Détection et configuration FCoE

Pour la détection à chaud des disques FCoE (« *Fibre-Channel over Ethernet* »), veuillez les configurer dans le micrologiciel d'un NIC EDD capable de démarrer.

DASD

Les périphériques DASD (« *Direct-access storage devices* ») ne peuvent pas être ajoutés ou configurés pendant l'installation. De tels périphériques sont spécifiés dans le fichier de configuration CMS.

Périphériques bloc avec DIF/DIX activé

DIF/DIX est une fonctionnalité de checksum de matériel fournie par des adaptateurs de bus hôte SCSI et des périphériques bloc. Lorsque DIF/DIX est activé, des erreurs se produiront si le périphérique bloc est utilisé comme périphérique bloc à but général. Des E/S mises en mémoire tampon ou des E/S basées **mmap(2)** ne fonctionneront pas de manière stable, car il n'y a pas de verrouillage dans le chemin d'écriture en mémoire tampon pour empêcher les données en tampon d'être remplacées une fois que le checksum DIF/DIX est calculé.

Ceci provoquera l'échec ultérieur des E/S avec une erreur checksum. Ce problème est commun à tous les périphériques bloc (ou basés sur système de fichiers), E/S en mémoire tampon ou E/S **mmap(2)**. Ainsi, il n'est pas possible de contourner les erreurs causées par ces remplacements.

Ainsi, les périphériques bloc activés DIF/DIX doivent uniquement être utilisés avec des applications qui utilisent **O_DIRECT**. De telles applications doivent utiliser le périphérique bloc brut. De manière alternative, il est également sûr d'utiliser le système de fichiers XFS sur un périphérique bloc activé DIF/DIX, tant que seules les E/S **O_DIRECT** soient passées à travers le système de fichiers. XFS est le seul système de fichiers qui ne se reple pas sur les E/S mises en mémoire tampon lorsqu'il effectue certaines opérations d'allocation.

La responsabilité de s'assurer que les données d'E/S ne changent pas une fois que le checksum DIF/DIX a été calculé est inhérente à l'application. Ainsi, seuls les applications conçues pour une utilisation avec les E/S **O_DIRECT** et le matériel DIF/DIX doivent utiliser DIF/DIX.

Chapitre 11. Vérification du système de fichiers (fsck)

La cohérence des systèmes de fichiers peut être vérifiée, et optionnellement réparée à l'aide d'outils de l'espace utilisateur spécifiques au système de fichiers. Ces outils sont souvent appelés les outils **fsck**, **fsck** est une version raccourcie de *file system check* (vérification du système de fichiers).



Note

Ces vérificateurs de systèmes de fichiers ne garantissent pas seulement la cohérence des métadonnées à travers le système de fichiers, mais ils n'ont pas connaissance des données contenues dans le système de fichiers et ne sont pas des outils de récupération de données.

Les incohérences de système de fichiers peuvent se produire pour diverses raisons, y compris mais non limitées aux erreurs de matériel, erreurs d'administration du stockage et aux bogues de logiciels.

Avant que les systèmes de fichiers de journalisation des métadonnées ne soient devenus communs, une vérification de système de fichiers était requise à chaque fois qu'un système tombait en panne ou perdait en puissance. Ceci était dû au fait qu'une mise à jour de système de fichiers pouvait avoir été interrompue, provoquant un état incohérent. Par conséquent, une vérification de système de fichiers était normalement exécutée sur chaque système de fichiers répertorié dans **/etc/fstab** au moment du démarrage. Pour les systèmes de fichiers journalisant, ceci consiste souvent en une très courte opération, car la journalisation des métadonnées du système de fichiers assure la cohérence des données même après une panne.

Cependant, des incohérences ou une corruption du système de fichiers peuvent se produire par moments, même pour les systèmes de fichiers qui effectuent une journalisation. Lorsque cela se produit, le vérificateur du système de fichiers doit être utilisé pour réparer le système de fichiers. Ci-dessous figurent les meilleures pratiques ainsi que d'autres informations utiles lorsque vous effectuez cette procédure.



Important

Red Hat ne recommande pas ceci, à moins que l'ordinateur ne démarre pas, que le système de fichiers soit extrêmement grand, ou que le système de fichiers se trouve sur un stockage distant. Il est possible de désactiver la vérification du système de fichiers lors du démarrage en paramétrant le sixième champ dans **/etc/fstab** sur 0.

11.1. Meilleures pratiques avec fsck

Habituellement, l'exécution de l'outil de vérification et de réparation de systèmes de fichiers peut au moins permettre de réparer automatiquement quelques-unes des incohérences trouvées. Dans certains cas, des inodes ou répertoires sérieusement endommagés peuvent être abandonnés s'ils ne peuvent pas être réparés. D'importantes modifications du système de fichiers peuvent se produire. Pour s'assurer que des changements inattendus ou indésirables ne soient pas appliqués de manière définitive, veuillez prendre les mesures de précaution suivantes :

Test à vide

La plupart des vérificateurs de systèmes de fichiers possèdent un mode opératoire qui vérifie mais ne répare pas le système de fichiers. Dans ce mode, le vérificateur imprimera toutes les erreurs qu'il trouvera et toutes les actions qu'il aurait effectuées sans pour autant modifier le système de fichiers.



Note

Les phases ultérieures de la vérification de cohérence peuvent imprimer des erreurs supplémentaires en découvrant les incohérences qui auraient pu être corrigées dans des phases antérieures si celles-ci avaient fonctionné en mode de réparation.

Opérer avant tout sur une image de système de fichiers

La plupart des systèmes de fichiers prennent en charge la création d'une *image des métadonnées*, une copie fragmentée du système de fichiers qui ne contient que les métadonnées. Comme les vérificateurs de système de fichiers opèrent uniquement sur les métadonnées, de telles images peuvent être utilisées pour effectuer le test à vide d'une réparation de système de fichiers, et pour évaluer les modifications qui seront ainsi effectuées. Si les modifications sont acceptables, alors la réparation pourra être effectuée sur le système de fichiers.



Note

Des systèmes de fichiers sévèrement endommagés peuvent causer des problèmes avec la création d'images de métadonnées.

Enregistrer une image de système de fichiers pour les vérifications du support technique

Une image des métadonnées du système de fichiers avant les réparations peut souvent se révéler utile pour les vérifications du support technique si cette corruption résulte d'un bogue de logiciel. Des schémas de corruption présents dans l'image pré-réparation peut aider lors de l'analyse de la cause principale.

Opérer sur les systèmes de fichiers non-montés uniquement

Une réparation de système de fichiers doit uniquement être exécutée sur un système de fichiers qui n'est pas monté. L'outil doit être le seul à avoir accès au système de fichiers ou des dommages supplémentaires pourraient être provoqués. La plupart des outils de systèmes de fichiers appliquent cette condition en mode de réparation, même si certains prennent uniquement en charge le mode de vérification seule sur un système de fichiers monté. Si le mode vérification-seule est exécuté sur un système de fichiers monté, il pourrait trouver de fausses erreurs qui n'auraient pas été trouvées avec une exécution sur un système de fichiers non-monté.

Erreurs de disque

Les outils de vérification de système de fichiers ne peuvent pas réparer les problèmes de matériel. Un système de fichiers doit être totalement lisible et inscriptible pour que la réparation puisse s'effectuer correctement. Si un système de fichiers est corrompu à cause d'un problème matériel, le système de fichiers doit d'abord être déplacé sur un disque en bon état, par exemple à l'aide de l'utilitaire **dd (8)**.

11.2. Informations spécifiques aux systèmes de fichiers pour fsck

11.2.1. ext2, ext3, et ext4

Tous ces systèmes de fichiers utilisent le binaire **e2fsck** pour effectuer leurs vérifications et réparations de système de fichiers. Les noms de fichiers **fsck.ext2**, **fsck.ext3**, et **fsck.ext4** sont des liens vers ce même binaire. Ces binaires sont exécutés automatiquement lors du démarrage et leur comportement diffère basé sur le fait que le système de fichiers est en cours de vérification, et selon l'état du système de fichiers.

Une vérification et réparation du système de fichiers complet est invoquée pour **ext2**, qui n'est pas un système de fichiers journalisant les métadonnées, et pour les systèmes de fichiers **ext4** sans journal.

Pour les systèmes de fichiers **ext3** et **ext4** avec journalisation des métadonnées, le journal est répété dans l'espace utilisateur et le sorti du binaire. Ceci est l'action par défaut car la répétition du journal assure un système de fichiers cohérent après une panne.

Si ces systèmes de fichiers rencontrent des incohérences de métadonnées alors qu'ils sont montés, ils enregistreront ce fait dans le super bloc du système de fichiers. Si **e2fsck** découvre qu'un système de fichiers est marqué d'une telle erreur, **e2fsck** effectuera une vérification complète après avoir répété le journal (s'il est présent).

e2fsck peut demander une entrée à l'utilisateur pendant l'exécution si l'option **-p** n'est pas spécifiée. L'option **-p** indique à **e2fsck** d'effectuer automatiquement toutes les réparations pouvant être faites sans risque. Si une intervention de l'utilisateur est requise, **e2fsck** indiquera le problème non corrigé dans sa sortie et reflétera ce statut dans le code de sortie.

Les options du runtime **e2fsck** habituellement utilisées incluent :

-n

Mode sans modifications. Opération de vérification seule.

superbloc -b

Spécifie le numéro de bloc d'un super bloc alternatif si le bloc principal est endommagé.

-f

Force une vérification complète même si le superbloc n'a aucune erreur enregistrée.

-j journal-dev

Spécifie le périphérique journal externe, s'il y en a un.

-p

Répare automatiquement ou « nettoie » le système de fichiers sans saisie de la part de l'utilisateur.

-y

Répondre « oui » à toutes les questions.

Toutes les options de **e2fsck** sont spécifiées dans la page man de **e2fsck(8)**.

Les cinq phases de base suivantes sont appliquées par **e2fsck** pendant l'exécution des :

1. vérifications des inodes, des blocs et des tailles.
2. vérifications des structures des répertoires.
3. vérifications de la connectivité des répertoires.
4. vérifications des comptes des références.

5. vérifications des informations des résumés de groupes.

L'utilitaire **e2image(8)** peut être utilisé pour créer une image de métadonnées avant d'effectuer les réparations dans le but de fournir un diagnostic ou pour faire des tests. L'option **-r** doit être utilisée pour effectuer des tests afin de créer un fichier partiellement alloué de la même taille que le système de fichiers. **e2fsck** peut ensuite opérer directement sur le fichier résultant. L'option **-Q** doit être spécifiée si l'image va ensuite être archivée ou utilisée pour fournir un diagnostic. Ceci crée un format de fichier plus compact, qui convient mieux aux transferts.

11.2.2. XFS

Aucune réparation n'est effectuée automatiquement au moment du démarrage. Pour initier une vérification ou réparation du système de fichiers, l'outil **xfs_repair** est utilisé.



Note

Même si un binaire **fsck.xfs** se trouve dans le paquet *xfsprogs*, il n'est présent que pour satisfaire les initscripts qui recherchent un binaire **fsck.filesystem** au moment du démarrage. **fsck.xfs** se ferme immédiatement avec un code de sortie 0.

Une autre chose à prendre en considération est que les anciens paquets *xfsprogs* contiennent un outil **xfs_check**. Cet outil est très lent et s'adapte mal aux systèmes de fichiers de grande taille. Ainsi, il a été déconseillé en faveur de **xfs_repair -n**.

Un journal correct est requis sur le système de fichiers pour permettre à **xfs_repair** d'opérer. Si le système de fichiers n'a pas été démonté correctement, il doit être monté, puis démonté avant d'utiliser **xfs_repair**. Si le journal est corrompu et ne peut être relu, l'option **-L** peut être utilisée pour remettre le journal à zéro.



Important

L'option **-L** doit uniquement être utilisée si le journal ne peut pas être relu. L'option ignore toutes les mises à jour de métadonnées dans le journal et provoquera des incohérences supplémentaires.

Il est possible d'exécuter **xfs_repair** lors de tests à vide, en mode de vérification seule, en utilisant l'option **-n**. Aucune modification ne se produira sur le système de fichiers lorsque cette option est indiquée.

xfs_repair ne peut prendre qu'un petit nombre d'options. Les options communément utilisées incluent :

-n

Mode sans modifications. Opération de vérification seule.

-L

Journal sans métadonnées. Utiliser uniquement si le journal ne peut pas être relu en étant monté.

-m maxmem

Limite la mémoire utilisée pendant l'exécution à un maxmem de MB. 0 peut être spécifié pour obtenir une estimation approximative de la quantité minimum de mémoire requise.

-l logdev

Spécifie le périphérique de journalisation externe, si présent.

Toutes les options de **xfs_repair** sont indiquées dans la page man **xfs_repair(8)**.

Les huit phases de base suivantes sont appliquées par **xfs_repair** pendant l'exécution des :

1. vérifications des inodes et des mappages (addressage) de blocs d'inodes.
2. vérifications des cartes d'allocation d'inodes.
3. vérifications de la taille des inodes.
4. vérifications des répertoires.
5. vérifications des noms de chemins d'accès.
6. vérification du nombre de liens.
7. Vérifications Freemap.
8. Vérifications de super blocs.

Ces phases, ainsi que les messages imprimés pendant l'opération, sont documentés en détails sur la page man de **xfs_repair(8)**.

xfs_repair n'est pas interactif. Toutes les opérations sont effectuées automatiquement sans saisie de la part de l'utilisateur.

S'il est nécessaire de créer une image des métadonnées avant d'effectuer les réparations, pour faire un diagnostic ou des tests, les utilitaires **xfs_metadump(8)** et **xfs_mdrestore(8)** peuvent être utilisés.

11.2.3. Btrfs

L'outil **btrfsck** est utilisé pour vérifier et réparer les systèmes de fichiers btrfs. Cet outil est toujours en cours de développement et pourrait ne pas détecter ou réparer tous les types de corruption de système de fichiers.

Par défaut, **btrfsck** n'effectue pas de modification sur le système de fichiers ; autrement dit, il exécute le mode vérification-seule par défaut. Si des réparations sont souhaitées, l'option **-repair** doit être spécifiée.

Les trois phases de base suivantes sont appliquées par **btrfsck** pendant l'exécution des :

1. vérifications des extensions.
2. vérifications de la racine du système de fichiers.
3. vérifications des comptes des références root.

L'utilitaire **btrfs-image(8)** peut être utilisé pour créer une image des métadonnées avant les réparations afin d'effectuer des diagnostics ou de faire des tests.

Chapitre 12. Partitions

L'utilitaire **parted** permet aux utilisateurs de :

- » Afficher la table de partitions existante
- » Modifier la taille des partitions existantes
- » Ajouter des partitions à partir d'espace libre ou de disques durs supplémentaires

Par défaut, le paquet **parted** est inclus dans l'installation de Red Hat Enterprise Linux. Pour lancer **parted**, connectez-vous en tant que root et saisissez la commande **parted /dev/sda** à l'invite shell (où **/dev/sda** est le nom de périphérique du lecteur que vous souhaitez configurer).

Si vous souhaitez supprimer ou redimensionner une partition, le périphérique sur lequel la partition réside ne doit pas être en cours d'utilisation. La création d'une nouvelle partition sur un périphérique en cours d'utilisation, même si possible, n'est pas recommandée.

Pour qu'un périphérique ne soit pas en cours d'utilisation, aucune des partitions sur le périphérique ne peut être montée, et tout espace swap sur le périphérique doit être désactivé.

La table de partitions doit également ne pas être modifiée pendant son utilisation car le noyau pourrait ne pas reconnaître les changements. Si la table de partitions ne correspondait pas à l'état des partitions montées, des informations pourraient être écrites sur la mauvaise partition, résultant en données perdues et remplacées.

La manière la plus simple d'effectuer ceci consiste à démarrer votre système en mode de secours. Lorsqu'il vous est demandé de monter le système de fichiers, veuillez sélectionner **Ignorer**.

De manière alternative, si le lecteur ne contient aucune partition en cours d'utilisation, vous pouvez les démonter avec la commande **umount** et désactiver tout l'espace swap sur le disque dur avec la commande **swapoff**.

[Tableau 12.1, « Commandes parted »](#) contient une liste de commandes **parted** couramment utilisées. Les sections suivantes expliquent certains de ces arguments et commandes de manière plus détaillée.

Tableau 12.1. Commandes parted

Commande	Description
check <i>minor-num</i>	Effectuer une simple vérification du système de fichiers
cp <i>depuis vers</i>	Copier le système de fichiers d'une partition à une autre ; <i>depuis</i> et <i>vers</i> sont les numéros mineurs des partitions
help	Afficher la liste des commandes disponibles
mklabel <i>étiquette</i>	Créer une étiquette de disque pour la table de partitions
mkfs <i>minor-num file-system-type</i>	Créer un système de fichiers de type <i>file-system-type</i>
mkpart <i>part-type fs-type start-mb end-mb</i>	Créer une partition sans créer de nouveau système de fichiers
mkpartfs <i>part-type fs-type start-mb end-mb</i>	Créer une partition et créer le système de fichiers spécifié
move <i>minor-num start-mb end-mb</i>	Déplacer la partition

Commande	Description
<code>name minor-num nom</code>	Nommer la partition uniquement pour disklables Mac et PC98
<code>print</code>	Afficher la table de partitions
<code>quit</code>	Quitter parted
<code>rescue start-mb end-mb</code>	Secourir une partition perdue de <i>start-mb</i> à <i>end-mb</i>
<code>resize minor-num start-mb end-mb</code>	Redimensionner la partition de <i>start-mb</i> à <i>end-mb</i>
<code>rm minor-num</code>	Supprimer la partition
<code>select périphérique</code>	Sélectionner un autre périphérique à configurer
<code>set minor-num flag state</code>	Paramétrer l'indicateur sur une partition ; l'état est soit « on », soit « off »
<code>toggle [NUMBER [FLAG]]</code>	Basculer l'état de l'indicateur <i>FLAG</i> sur le numéro de partition <i>NUMBER</i>
<code>unit UNIT</code>	Paramétrer l'unité par défaut sur <i>UNIT</i>

12.1. Afficher la table de partitions

Après avoir lancé **parted**, veuillez utiliser la commande **print** pour afficher la table de partitions. Une table similaire à la suivante s'affiche :

Exemple 12.1. Table de partitions

```
Model: ATA ST3160812AS (scsi)
Disk /dev/sda: 160GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos

Number  Start   End     Size    Type     File system  Flags
  1      32.3kB  107MB   107MB   primary  ext3         boot
  2      107MB   105GB   105GB   primary  ext3
  3      105GB   107GB   2147MB  primary  linux-swap
  4      107GB   160GB   52.9GB  extended          root
  5      107GB   133GB   26.2GB  logical  ext3
  6      133GB   133GB   107MB   logical  ext3
  7      133GB   160GB   26.6GB  logical          lvm
```

La première ligne contient le type de disque, le fabricant, le numéro du modèle et l'interface, et la seconde ligne affiche le type d'étiquette du disque. Le reste de la sortie, sous la quatrième ligne, affiche la table de partitions.

Dans la table de partitions, le numéro mineur *Minor* est le **numéro** de la partition. Par exemple, la partition avec le numéro mineur 1 correspond à `/dev/sda1`. Les valeurs **Start** et **End** sont en méga-octets. Les types **Type** valides sont les métadonnées (« metadata ») « free », « primary », « extended », ou « logical ». **Filesystem** est le type de système de fichiers, qui peut correspondre à l'un des types suivants :

- ✱ ext2
- ✱ ext3
- ✱ fat16

- ✧ fat32
- ✧ hfs
- ✧ jfs
- ✧ linux-swaps
- ✧ ntfs
- ✧ reiserfs
- ✧ hp-ufs
- ✧ sun-ufs
- ✧ xfs

Si le système de fichiers **Filesystem** d'un périphérique n'affiche aucune valeur, cela signifie que son type de système de fichiers est inconnu.

La colonne **Flags** répertorie les indicateurs paramétrés pour la partition. Les indicateurs disponibles incluent boot, root, swap, hidden, raid, lvm, ou lba.



Note

Pour sélectionner un autre périphérique sans avoir à redémarrer **parted**, veuillez utiliser la commande **select**, suivie par le nom du périphérique (par exemple, **/dev/sda**). Cela vous permettra d'afficher ou de configurer la table de partitions d'un périphérique.

12.2. Création d'une partition



Avertissement

Ne pas tenter de créer une partition sur un périphérique en cours d'utilisation.

Procédure 12.1. Création d'une partition

1. Avant de créer une partition, veuillez démarrer en mode de secours (ou démontez toute partition sur le périphérique et éteignez tout espace swap sur le périphérique).
2. Démarrez **parted**, où **/dev/sda** est le périphérique sur lequel créer la partition :

```
# parted /dev/sda
```

3. Afficher la table de partitions pour déterminer s'il y a suffisamment d'espace libre :

```
# print
```

S'il n'y a pas suffisamment d'espace libre, vous pouvez redimensionner une partition existante. Veuillez consulter la [Section 12.4, « Redimensionnement d'une partition »](#) pour obtenir des détails.

12.2.1. Créer la partition

À partir de la table de partitions, déterminez les points du début et de fin de la nouvelle partition, ainsi que le type de partition souhaité. Seuls quatre types de partitions principales (sans partition étendue) sont autorisés sur un périphérique. Si vous avez besoin de plus de quatre partitions, il est possible de créer trois partitions principales, une partition étendue, et de multiples partitions logiques dans la partition étendue. Pour voir une vue d'ensemble des partitions de disque, veuillez consulter l'appendice *Introduction aux partitions de disques* du *Guide d'installation Red Hat Enterprise Linux 7*.

Par exemple, pour créer une partition principale avec un système de fichiers ext3 de 1024 méga-octets à 2048 méga-octets sur un disque dur, veuillez saisir la commande suivante :

```
# mkpart primary ext3 1024 2048
```



Note

Si, au contraire, vous utilisez la commande **mkpartfs**, le système de fichiers sera créé après la création de la partition. Cependant, **parted** ne prend pas en charge la création d'un système de fichiers ext3. Ainsi, si vous souhaitez créer un système de fichiers ext3, veuillez utiliser **mkpart** et créer le système de fichiers avec la commande **mkfs** comme décrit ultérieurement.

Les modifications prendront effet dès que vous appuyez sur **Entrée**. Ainsi, veuillez vérifier la commande avant de l'exécuter.

Après avoir créé la partition, veuillez utiliser la commande **print** pour confirmer que celle-ci se trouve effectivement dans la table de partitions avec le bon type de partition, le bon type de système de fichiers, et la bonne taille. Veuillez également vous rappeler du numéro mineur de la nouvelle partition afin d'être en mesure d'étiqueter un système de fichiers dessus. Vous devriez aussi afficher la sortie de **cat /proc/partitions** une fois que « parted » est fermé afin de vous assurer que le noyau reconnaisse la nouvelle partition.

Le nombre maximum de partitions que « parted » va créer est de 128. Même si la spécification GPT (« *GUID Partition Table* ») permet d'avoir plus de partitions en agrandissant la zone réservée à la table de partitions, une pratique courante utilisée par « parted » est de la limiter à une zone qui suffira pour 128 partitions.

12.2.2. Formatage et étiquetage de la partition

Pour formater et étiqueter la partition, veuillez utiliser la procédure suivante :

Procédure 12.2. Formater et étiqueter la partition

1. La partition ne possède toujours pas de système de fichiers. Pour en créer un, veuillez utiliser la commande suivante :

```
# /usr/sbin/mkfs -t ext3 /dev/sda6
```



Avertissement

Formater la partition détruira de manière permanente toutes les données s'y trouvant.

2. Ensuite, veuillez donner une étiquette au système de fichiers sur la partition. Par exemple, si le système de fichiers sur la nouvelle partition est nommé `/dev/sda6` et que vous souhaitez l'étiqueter `/work`, veuillez utiliser :

```
# e2label /dev/sda6 /work
```

Par défaut, le programme d'installation utilise le point de montage de la partition en tant qu'étiquette afin de s'assurer que l'étiquette soit effectivement unique.

Ensuite, veuillez créer un point de montage (par exemple, `/work`) en tant que root.

12.2.3. Ajoutez-le à `/etc/fstab`

En tant que root, modifiez le fichier `/etc/fstab` afin d'inclure la nouvelle partition à l'aide de son UUID. Veuillez utiliser la commande `blkid -o list` pour obtenir une liste complète de l'UUID de la partition, ou `blkid device` pour obtenir les détails individuels du périphérique.

La première colonne doit contenir `UUID=` suivi de l'UUID du système de fichiers. La seconde colonne doit contenir le point de montage de la nouvelle partition, et la colonne suivante doit être le type de système de fichiers (par exemple, `ext3` ou `swap`). Si vous souhaitez obtenir des informations supplémentaires sur le format, veuillez lire la page man avec la commande `man fstab`.

Si la quatrième colonne contient le mot `defaults`, alors la partition sera montée au moment du démarrage. Pour monter la partition sans effectuer de redémarrage, veuillez saisir la commande suivante en tant que root :

```
mount /work
```

12.3. Suppression de partition



Avertissement

Ne tentez pas de supprimer une partition sur un périphérique en cours d'utilisation.

Procédure 12.3. Supprimer une partition

1. Avant de supprimer une partition, veuillez démarrer en mode de secours (ou démontez toute partition sur le périphérique, puis éteignez tout espace swap présent sur le périphérique).
2. Lancez `parted`, où `/dev/sda` est le périphérique sur lequel supprimer la partition :

```
# parted /dev/sda
```

3. Affichez la table de la partition actuelle pour déterminer le numéro mineur de la partition à supprimer :

```
# print
```

- Supprimez la partition à l'aide de la commande **rm**. Par exemple, pour supprimer la partition portant le numéro mineur 3, saisissez la commande suivante :

```
# rm 3
```

Les modifications prendront effet dès que vous appuierez sur **Entrée**. Ainsi, veuillez vérifier la commande avant de l'exécuter.

- Après avoir supprimé la partition, utilisez la commande **print** pour confirmer sa suppression de la table de partitions. Vous devriez également afficher la sortie de **/proc/partitions** afin de vous assurer que le noyau sache effectivement que la partition a été supprimée.

```
# cat /proc/partitions
```

- La dernière étape consiste à la supprimer du fichier **/etc/fstab**. Trouvez la ligne qui déclare la partition supprimée, puis supprimez-la du fichier.

12.4. Redimensionnement d'une partition

Sauvegardez vos données importantes sur le système de fichiers avant de redimensionner une partition.

Procédure 12.4. Redimensionner une partition

- Démonter le périphérique.

```
~]# umount /dev/vda
```

- Exécuter **fdisk** *device_name*.

```
~]# fdisk /dev/vda
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them. Be
careful before using the write command.

Command (m for help):
```

- Vérifier le nombre de partitions à supprimer par l'option **p**. Les partitions sont listées sous l'en-tête 'Device'.

```
Command (m for help): p
Disk /dev/vda: 16.1 GB, 16106127360 bytes, 31457280 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x0006d09a

Device      Boot      Start          End      Blocks      Id  System
/dev/vda1   *           2048       1026047       512000     83  Linux
/dev/vda2             1026048       31457279       15215616     8e  Linux LVM
```




Important

Red Hat ne prend en charge que l'extension ou le redimensionnement des partitions LVM.

- Utilisez l'option **d** pour supprimer une partition. S'il y a plus d'une partition disponible, **fdisk** vous invitera à les supprimer une par une.

```
Command (m for help): d
Partition number (1,2, default 2): 2
Partition 2 is deleted
```

- Utiliser l'option **n** pour créer une nouvelles partition. Suivre les invites et veillez à ce qu'il y ait suffisamment de place pour tout redimensionnement à venir. Il est possible d'indiquer une taille lisible de visu à la place des secteurs, si vous préférez.



Note

Il est conseillé de suivre les options par défaut de **fdisk** pour les valeurs par défaut et les tailles de partition (par exemple, les premiers secteurs de partitions).

```
Command (m for help): n
Partition type:
   p   primary (1 primary, 0 extended, 3 free)
   e   extended
Select (default p): *Enter*
Using default response p
Partition number (2-4, default 2): *Enter*
First sector (1026048-31457279, default 1026048): *Enter*
Using default value 1026048
Last sector, +sectors or +size{K,M,G} (1026048-31457279, default
31457279): +500M
Partition 2 of type Linux and of size 500 MiB is set
```

- Définir le type de partition à LVM

```
Command (m for help): t
Partition number (1,2, default 2): *Enter*
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'
```

- Inscrire les changements à l'aide de l'option **w** une fois que vous êtes certain qu'ils sont corrects.



Important

Les erreurs dans ce processus qui sont inscrites peuvent causer l'instabilité de la partition sélectionnée.

8. Exécuter **e2fsck** sur le périphérique pour vérifier l'homogénéité.

```
~]# e2fsck /dev/vda
e2fsck 1.41.12 (17-May-2010)
Pass 1:Checking inodes, blocks, and sizes
Pass 2:Checking directory structure
Pass 3:Checking directory connectivity
Pass 4:Checking reference counts
Pass 5:Checking group summary information
ext4-1:11/131072 files (0.0% non-contiguous),27050/524128 blocks
```

9. Monter le périphérique.

```
~]# mount /dev/vda
```

Pour obtenir plus d'informations, consultez les références ci-dessous :

man fdisk

La page man de **fdisk**. Contient des informations de base sur la commande **fdisk** et ce qu'elle prend en charge.

L'option m de fdisk

Cette option répertorie toutes les commandes possibles de **fdisk**.

Chapitre 13. Créer et maintenir des clichés avec Snapper

Un volume de clichés est une copie figée dans le temps permettant de renverser un système de fichiers à un état passé. Snapper est un outil en ligne de commandes qui permet de créer et de maintenir les clichés pour Btrfs et pour les systèmes de fichiers LVM alloués dynamiquement.

13.1. Configuration initiale de Snapper

Snapper requiert une configuration de fichiers discrète pour chaque volume sur lequel il opère. Les fichiers de configuration doivent être installés manuellement. Par défaut, seul l'utilisateur root est autorisé à exécuter les commandes Snapper.

Red Hat recommande l'utilisation du système de fichiers ext4 avec Snapper sur Red Hat Enterprise Linux 7. N'utilisez le système de fichiers XFS sur les volumes lvm-thin que si vous vérifiez l'espace libre du pool pour empêcher les problèmes de manque d'espace pouvant mener à un échec.

Notez que les outils Btrfs et système de fichiers ne sont fournis qu'en tant qu'aperçus technologiques, ce qui les rend inadéquat pour une utilisation en production.

Bien qu'il soit possible de permettre à un utilisateur ou un groupe autre que root d'utiliser certaines commandes Snapper, Red Hat vous recommande de ne *pas* ajouter d'autorisations élevées aux utilisateurs ou groupes normalement non privilégiés. Une telle configuration contourne SELinux et pourrait poser un risque pour la sécurité. Red Hat vous recommande de passer en revue ces capacités avec votre équipe de sécurité et d'envisagez d'utiliser l'infrastructure de **sudo**.

Procédure 13.1. Créer un fichier de configuration Snapper

1. Créer ou choisir soit :
 - ✦ Un volume logique alloué dynamiquement avec un système de fichiers Red Hat ajouté, ou
 - ✦ Un sous-volume Btrfs.
2. Montez le système de fichiers.
3. Créer un fichier de configuration qui définit ce volume.

Pour LVM2 :

```
# snapper -c config_name create-config -f "lvm(fs_type)" /mount-point
```

Pour Btrfs :

```
~]# snapper -c config_name create-config -f btrfs /mount-point
```

- ✦ L'option **-c *config_name*** spécifie le nom du fichier de configuration.
- ✦ La commande **create-config** demande à Snapper de créer un fichier de configuration.
- ✦ **-f *file_system*** demande à Snapper quel système de fichier utiliser ; sinon, Snapper tentera de détecter le système de fichiers.
- ✦ **/mount-point** est l'endroit où le sous-volume ou système de fichiers LVM2 alloué dynamiquement est monté.

Ainsi, pour créer un fichier de configuration intitulé **lvm_config** sur un sous-volume LVM2 avec un système de fichiers ext4, monté sur **/lvm_mount**, utiliser :

```
# snapper -c lvm_config create-config -f "lvm(ext4)" /lvm_mount
```

Ainsi, pour créer un fichier de configuration intitulé **btrfs_config** sur un sous-volume Btrfs monté sur **/lvm_mount**, utiliser :

```
# snapper -c btrfs_config create-config -f btrfs /btrfs_mount
```

Les fichiers de configuration sont stockés dans le répertoire **/etc/snapper/configs/**.

13.2. Créer un cliché Snapper

Snapper peut créer ces types de clichés :

Pré Cliché

Un pré cliché sert de point d'origine à un post cliché. Les deux sont liés étroitement et conçus pour garder la trace de modifications de système de fichiers entre les deux points. Le pré cliché doit être créé avant le post cliché.

Post cliché

Un post cliché sert de point de terminaison d'un pré cliché. Les pre et post clichés ensemble définissent une portée de comparaison. Par défaut, chaque nouveau volume de Snapper est configuré pour créer un comparaison en arrière plan une fois qu'un cliché aura été créé.

Simple cliché

Un simple cliché est un cliché autonome créé à un moment précis. Ils peuvent être utilisés pour garder la trace des modifications et ont un point général dans le temps où retourner plus tard.

13.2.1. Créer un paire de Pré et Post clichés

Pour créer une paire de Pré et Post clichés avec Snapper, la première étape est de créer un Pre cliché

13.2.1.1. Créer un Pré cliché

Pour créer un Pré cliché, utiliser :

```
# snapper -c config_name create -t pre
```

L'option **-c config_name** crée un cliché en fonction des spécifications qui se trouvent dans le fichier de configuration indiqué. Si le fichier de configuration n'existe pas encore, voir [Section 13.1, « Configuration initiale de Snapper »](#).

L'option **create -t** indique quel type de cliché créer. Les entrées acceptées sont **pre**, **post**, ou **single**.

Ainsi, pour créer un pré cliché par le fichier de configuration **lvm_config**, créée dans [Section 13.1, « Configuration initiale de Snapper »](#), exécuter :

```
# snapper -c SnapperExample create -t pre -p
1
```

L'option **-p** affiche le nombre de clichés créés et est optionnelle.

13.2.1.2. Créer un Post Cliché avec Snapper

Un post instantané sert de point de terminaison d'un pré cliché et doit être créé après le pré cliché parent en suivant les instructions dans [Section 13.2.1.1, « Créer un Pré cliché »](#).

Procédure 13.2. Créer un Post cliché

1. Déterminer le nombre de Pré instantanés

```
# snapper -c config_name list
```

Ainsi, pour afficher la liste des clichés créés par le fichier de configuration **lvm_config**, exécutez ce qui suit :

```
# snapper -c lvm_config list
Type   | # | Pre # | Date           | User | Cleanup | Description
| Userdata
-----+-----+-----+-----+-----+-----+-----
-----+-----
single | 0 |      |                | root |         | current
|
pre    | 1 |      | Mon 06<...>  | root |         |
```

La sortie ci-dessus affiche que l'instantané est le numéro 1

2. Créer un post instantané lié au Pré instantané créé auparavant :

```
# snapper -c config_file create -t post --pre-num pre_snapshot_number
```

- ✦ L'option **-t post** indique la création d'un type de Post instantané.
- ✦ L'option **--pre-num** indique le Pré instantané correspondant.

Ainsi, pour créer un post instantané par le fichier de configuration **lvm_config** lié au pré instantané numéro 1, exécuter :

```
# snapper -c lvm_config create -t post --pre-num 1 -p
2
```

L'option **-p** affiche le nombre de clichés créés et est optionnelle.

3. Les Pré et Post instantanés 1 et 2 sont maintenant créés et mis en paires. Vérifier ceci par la commande **list** :

```
# snapper -c lvm_config list
Type   | # | Pre # | Date           | User | Cleanup | Description
| Userdata
-----+-----+-----+-----+-----+-----+-----
-----+-----
single | 0 |      |                | root |         | current
|
pre    | 1 |      | Mon 06<...>  | root |         |
post   | 2 | 1    | Mon 06<...>  | root |         |
```

Il est également possible de créer une commande à l'intérieur du Pré ou Post Cliché qui puisse être utile lors des tests. Voir [Procédure 13.3, « Créer une commande Pré ou Post Cliché »](#), qui représente un raccourci aux étapes suivantes :

1. Exécuter la commande # **snapper create pre snapshot**.
2. Exécuter un commande ou une liste de commandes pour effectuer des actions ayant un impact possible sur le contenu du système de fichiers.
3. Exécuter la commande # **snapper create post snapshot**.

Procédure 13.3. Créer une commande Pré ou Post Cliché

1. Pour créer une commande en Pré ou Post Cliché, exécuter :

```
# snapper -c lvm_config create --command "command_to_be_tracked"
```

Par exemple, pour suivre la création du fichier `/lvm_mount/hello_file`, utiliser :

```
# snapper -c lvm_config create --command "echo Hello >
/lvm_mount/hello_file"
```

2. Pour vérifier cela, utiliser la commande **status**.

```
# snapper -c config_file status
first_snapshot_number..second_snapshot_number
```

Par exemple, pour suivre les changements effectués à la première étape, utiliser :

```
# snapper -c lvm_config status 3..4
+..... /lvm_mount/hello_file
```

Utiliser la commande **list** pour vérifier le nombre de clichés si besoin est.

Pour plus d'informations sur la commande **status**, voir [Section 13.3, « Assurer le suivi des changements parmi les clichés Snapper »](#).

Notez qu'il n'y a aucune garantie que la commande qui se trouve dans l'exemple ci-dessus soit la seule chose que les clichés capturent. Snapper enregistre également tout ce qui a été modifié par le système, pas seulement ce qu'un utilisateur aura pu modifier.

13.2.2. Créer un cliché Snapper unique

Créer un cliché Snapper unique est similaire à la création d'un pré ou d'un post cliché, sauf que l'option **-t** spécifie unique. Le cliché unique est utilisé pour créer un cliché unique au bon moment sans qu'il soit lié à d'autres. Toutefois, si vous êtes intéressé par un moyen simple de créer des instantanés de volumes LVM2 d'allocation dynamique sans devoir automatiquement générer des comparaisons ou des informations supplémentaires, Red Hat recommande d'utiliser le gestionnaire de stockage système à la place de Snapper à cet effet, tel que décrit dans [Section 15.2.6, « Cliché »](#).

Pour créer un cliché unique, exécuter :

```
# snapper -c config_name create -t single
```

Ainsi, la commande suivante créer un cliché unique par le fichier de configuration **lvm_config**.

```
# snapper -c lvm_config create -t single
```

Bien que les clichés uniques ne soient pas conçus pour assurer le suivi des changements, vous pouvez utiliser les commandes **snapper diff**, **xadiff**, et **status** pour comparer deux clichés. Pour plus d'informations sur ces commandes, consulter [Section 13.3, « Assurer le suivi des changements parmi les clichés Sanpper »](#).

13.2.3. Configurer Snapper pour qu'il prenne des clichés automatiquement

Prendre des clichés automatiquement est l'une des principales fonctionnalités de Snapper. Quand vous configurez Snapper pour un volume, par défaut, Snapper prend un cliché du volume une fois par heure.

Sous la configuration par défaut, Snapper conserve :

- ✦ 10 clichés par heure, et le dernier cliché sauvegardé sous « daily ».
- ✦ 10 clichés par heure, et le dernier cliché du mois sauvegardé sous « monthly ».
- ✦ 10 clichés par heure, et le dernier cliché sauvegardé sous « yearly ».
- ✦ 10 clichés par an.

Notez que Snapper conserve, par défaut, 50 clichés maximum au total. Cependant, Snapper conserve par défaut tous les clichés créés dans les 1,800 secondes venant de s'écouler.

Le configuration par défaut est spécifiée dans le fichier **/etc/snapper/config-templates/default**. Quand vous exécutez la commande **# snapper create-config** pour créer une configuration, toutes les valeurs définies sont basées sur la configuration par défaut. Vous pouvez modifier la configuration de n'importe quel volume défini dans le fichier **/etc/snapper/configs/config_name**.

13.3. Assurer le suivi des changements parmi les clichés Sanpper

Exécutez les commandes **status**, **diff**, et **xadiff** pour assurer le suivi des changements apportés à un sous-volume entre les clichés :

status

La commande **status** affiche une liste de fichiers et de répertoires qui ont été créés, modifiés ou supprimés entre deux clichés, et qui constitue une liste de modifications entre deux clichés. Un administrateur de systèmes peut utiliser cette commande pour obtenir un aperçu de ces changements sans détails excessifs.

Pour plus d'informations, voir [Section 13.3.1, « Comparez les changements avec la commande status »](#).

diff

La commande **diff** affiche un diff, pour les fichiers et les répertoires modifiés entre deux clichés, tel indiqué par la commande **status** si une modification au moins a été détectée.

Pour plus d'informations, voir [Section 13.3.2, « Comparez les changements avec la commande diff »](#).

xadiff

La commande **xadiff** compare les changements d'attributs étendus d'un fichier ou d'un répertoire entre deux clichés.

Pour plus d'informations, voir [Section 13.3.3, « Comparez les changements avec la commande xadiff »](#).

13.3.1. Comparez les changements avec la commande **status**

La commande **status** affiche une liste de tous les fichiers et répertoires qui ont été créés, modifiés ou supprimés entre deux clichés.

Pour afficher le statut des fichiers entre deux clichés, exécutez :

```
# snapper -c config_file status
first_snapshot_number..second_snapshot_number
```

Exécutez la commande **list** pour déterminer le nombre de clichés, si nécessaire.

Ainsi, la commande suivante affiche les changements effectués entre le cliché 1 et le cliché 2, en utilisant le fichier de configuration **lvm_config**.

```
snapper -c lvm_config status 1..2
tp.... /lvm_mount/dir1
-..... /lvm_mount/dir1/file_a
c.ug.. /lvm_mount/file2
+..... /lvm_mount/file3
....x. /lvm_mount/file4
cp..xa /lvm_mount/file5
```

Interprétez les lettres et les points dans la première partie de la sortie comme des colonnes :

```
+..... /lvm_mount/file3
| | | | |
123456
```

La colonne 1 indique une modification de fichier (entrée de répertoire). Les valeurs possibles sont :

Colonne 1

Sortie	Signification
.	Rien n'a changé.
+	Fichier créé.
-	Fichier supprimé.
c	Contenu modifié.
t	Le type d'entrée de répertoire a été modifié. Ainsi, un ancien lien symbolique a été changé en fichier standard en conservant le même nom de fichier.

La colonne 2 indique une modification de permission de fichier. Les valeurs possibles sont :

Colonne 2

Sortie	Signification
.	Aucune permission n'a été changée.
p	Permissions modifiées.

La colonne 3 indique les changements d'appartenance utilisateur. Les valeurs possibles sont :

Colonne 3

Sortie	Signification
.	Aucune appartenance utilisateur n'a été changée.
u	L'appartenance utilisateur a été modifiée.

La colonne 4 indique les changements d'appartenance groupe. Les valeurs possibles sont les suivantes :

Colonne 4

Sortie	Signification
.	Aucune appartenance de groupe n'a été modifiée.
g	L'appartenance de groupe a été modifiée.

La colonne 5 indique les changements d'attributs étendus. Les valeurs possibles sont les suivantes :

Colonne 5

Sortie	Signification
.	Aucun attribut étendu n'a été modifié.
x	Attributs étendus modifiés.

La colonne 6 indique les modifications d'ACL (Access Control List). Les valeurs possibles sont les suivantes :

Colonne 6

Sortie	Signification
.	Aucun ACL n'a été modifié
a	Les ACL modifiés

13.3.2. Comparez les changements avec la commande `diff`

La commande `diff` affiche les modifications de tous les fichiers et répertoires entre deux clichés.

```
# snapper -c config_name diff first_snapshot_number..second_snapshot_number
```

Exécutez la commande `list` pour déterminer le nombre de clichés, si nécessaire.

Ainsi, pour comparer les changements effectués entre le cliché 1 et le cliché 2, en utilisant le fichier de configuration `lvm_config`, exécutez :

```
# snapper -c lvm_config diff 1..2
--- /lvm_mount/.snapshots/13/snapshot/file4 19<...>
+++ /lvm_mount/.snapshots/14/snapshot/file4 20<...>
@@ -0,0 +1 @@
+words
```

La sortie ci-dessus montre que le fichier **file4** a été modifié par l'ajout de « words » dans le fichier.

13.3.3. Comparez les changements avec la commande **xadiff**

La commande **xadiff** compare les changements d'attributs étendus d'un fichier ou d'un répertoire entre deux clichés :

```
# snapper -c config_name xadiff
first_snapshot_number..second_snapshot_number
```

Exécutez la commande **list** pour déterminer le nombre de clichés, si nécessaire.

Ainsi, pour afficher la sortie **xadiff** pour comparer les changements effectués entre le cliché 1 et le cliché 2, en utilisant le fichier de configuration **lvm_config**, exécutez :

```
# snapper -c lvm_config xadiff 1..2
```

13.4. Annuler les changements entre les clichés

Pour annuler les changements effectués entre deux clichés Snapper existants, la commande **undochange** sera utilisée sous le format suivant : **snapper -c config_name undochange 1..2** avec 1 pour le premier cliché et 2 pour le second.



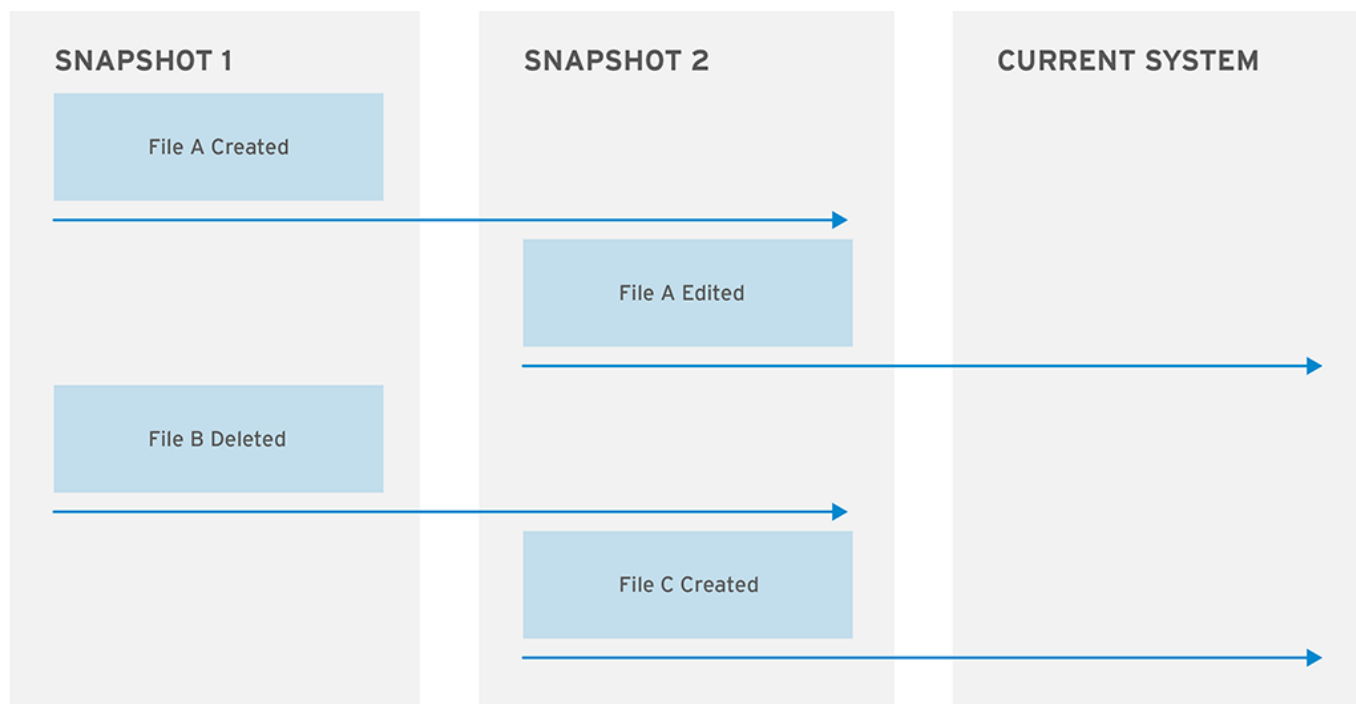
Important

Utiliser la commande **undochange** ne renvoie pas un volume Snapper dans son état d'origine et ne produit aucune homogénéité parmi les données. Toute modification de fichier ayant lieu en dehors de la portée indiquée, par exemple après Cliché 2, demeurera inchangée après un retour en arrière, par exemple, vers l'état du Cliché 1. Ainsi, si **undochange** est exécuté pour annuler la création d'un utilisateur, tous les fichiers appartenant à cet utilisateur demeureront inchangés.

Il n'y a pas non plus de mécanisme qui permette d'assurer que l'homogénéité de fichier en tant que cliché est effectuée, donc les problèmes d'homogénéité déjà existants peuvent se transmettre au cliché quand on exécute **undochange**.

Ne pas utiliser la commande **undochange** dans un système de fichiers root, car vous risquez de ne pas aboutir à vos fins.

Le diagramme suivant indique comme la commande **undochange** fonctionne :



RHEL_387527_0125

Figure 13.1. Statut Snapper sur la durée

Le diagramme indique le moment où **snapshot_1** a été créé, **file_a** est créé, puis **file_b** supprimé. **Snapshot_2** est alors créé. après quoi **file_a** est modifié et **file_c** est créé. C'est actuellement l'état du système. Le système actuel possède une version modifiée de **file_a**, aucun **file_b**, et un nouveau fichier **file_c**.

Quand on appelle la commande **undochange**, Snapper crée une liste de fichiers modifiés entre le premier clicé listé et le second. Dans le diagramme, si la commande est **snapper -c SnapperExample undochange 1..2**, Snapper crée une liste de fichiers modifiés (c'est à dire que, **file_a** est créé ; **file_b** est supprimé) et les applique au système actuel. Ainsi, le système actuel n'aura pas de fichier **file_a** (qui reste à créer une fois que **snapshot_1** a été créé), **file_b** existera (copié à partir de **snapshot_1** dans le système actuel), et **file_c** existera, car sa création était en dehors de la durée spécifiée. Sachez que, si **file_b** et **file_c** entrent en conflit, cela indique que le système pourrait devenir corrompu.

On peut aussi utiliser la commande **snapper -c SnapperExample undochange 2..1**. Dans ce cas, le système de fichiers actuel remplacera la version modifiée du fichier **file_a** par une copie de **snapshot_1**, annulant ainsi les modifications qui ont eu lieu sur ce fichier une fois que **snapshot_2** a été créé.

Utiliser les commandes **mount** et **unmount** pour annuler les changements

La commande **undochange** n'est pas toujours la meilleure façon d'annuler des modifications. Avec les commandes **status** et **diff**, vous pouvez prendre des décisions informées.

Si besoin est, la commande **mount** active le cliché LVM Snapper respectif avant le montage. Les commandes **mount** et **unmount** peuvent être utiles si vous êtes, par exemple, intéressé à monter des clichés et à en extraire une ancienne version manuellement. Les fichiers devront être annulés manuellement afin de pouvoir être copiés d'un cliché monté au système de fichiers actuel. Le système de fichiers actuel, cliché 0, est le système de fichiers live créé dans [Procédure 13.1, « Créer un fichier de configuration Snapper »](#). Copier les fichiers dans la sous-arborescence du point de montage d'origine.

Les commandes **mount** et **unmount** peuvent être utilisées pour les requêtes explicites côté client. Le fichier

`/etc/snapper/configs/config_name` contient les variables `ALLOW_USERS=` and `ALLOW_GROUPS=` qui permettent d'ajouter des utilisateurs et des groupes. Puis, **snapperd** vous permet d'effectuer les opérations de montage pour les utilisateurs et les groupes ajoutés. Les commandes **mount** et **unmount** ne sont utiles que si vous souhaitez monter des clichés et naviguer leur contenu indépendamment du flux de travail de Snapper.

13.5. Suppression d'un instantané

Supprimer un instantané :

```
~]# snapper -c config_name delete snapshot_number
```

Cela peut être vérifié par la commande **list**.

Chapitre 14. Espace swap

L'espace *swap* sur Linux est utilisé lorsque la mémoire physique (RAM) est pleine. Si le système a besoin de plus de ressources mémoire et que la mémoire RAM est pleine, les pages mémoire inactives sont alors déplacées vers l'espace swap. Même si l'espace swap peut aider les ordinateurs avec une plus petite quantité de RAM, celui-ci ne doit pas être considéré comme un remplacement pour plus de RAM. L'espace swap se trouve sur les disques durs, dont le temps d'accès est plus lent que pour la mémoire physique. L'espace Swap peut représenter une partition swap exclusive (conseillé), un fichier swap, ou un ensemble de partitions et de fichiers swap. Notez que *Btrfs* ne supporte pas les espaces swap.

Il y a quelques années, le montant d'espace swap recommandé augmentait de manière linéaire avec le montant de mémoire vive d'un système. Cependant, les systèmes modernes incluaient souvent des centaines de gigaoctets de mémoire vive. Par conséquent, l'espace swap recommandé est considéré comme une fonction de la charge de travail de la mémoire système, et non de la mémoire système uniquement.

[Tableau 14.1, « Espace swap recommandé »](#) fournit la taille recommandée d'une partition swap en fonction de la quantité de mémoire vive dans votre système et si vous souhaitez suffisamment de mémoire pour que votre système puisse hiberner. La taille de partition swap recommandée est automatiquement établie lors de l'installation. Cependant, pour permettre une hibernation, vous devrez modifier l'espace swap pendant l'étape du partitionnement personnalisé.

Les conseils qui se trouvent dans [Tableau 14.1, « Espace swap recommandé »](#) sont particulièrement importants pour les systèmes ayant une mémoire réduite (1 Go ou moins). Si vous n'allouez pas suffisamment de mémoire sur ces systèmes, vous pouvez vous heurter à des problèmes d'instabilité ou même de compromettre le démarrage de votre système installé.

Tableau 14.1. Espace swap recommandé

Quantité de RAM du système	Espace swap recommandé	Espace swap recommandé si l'hibernation est autorisée
≤ 2 Go	2 fois la quantité de RAM	3 fois la quantité de RAM
> 2 Go – 8 Go	Égal à la quantité de RAM	2 fois la quantité de RAM
> 8 Go – 64 Go	Au moins 4 Go	1,5 fois la quantité de RAM
> 64 Go	Au moins 4 Go	L'hibernation n'est pas recommandée

À la limite entre chaque gamme listée dans [Tableau 14.1, « Espace swap recommandé »](#), par exemple, un système avec 2 Go, 8 Go, ou 64 Go de mémoire vive, une certaine discrétion peut être observée en ce qui concerne l'espace swap choisi et la prise en charge de l'hibernation. Si vos ressources système vous le permettent, augmenter l'espace swap conduira à une amélioration de la performance. Un espace swap de 100 Go au moins est conseillé pour les systèmes de plus de 140 processeurs logiques ou de plus de 3 To de mémoire vive.

Remarquez que vous pourrez obtenir une meilleure performance swap en distribuant l'espace swap sur de multiples périphériques de stockage, plus particulièrement sur des systèmes possédant des disques, contrôleurs et interfaces rapides.



Important

Les systèmes de fichiers et volumes LVM2 assignés en tant qu'espace swap *ne doivent pas* être utilisés lorsqu'ils sont modifiés. Toute tentative de modification de l'espace swap échoue si un processus système ou si le noyau utilise l'espace swap au même moment. Veuillez utiliser les commandes **free** et **cat /proc/swaps** pour vérifier combien d'espace swap est en cours d'utilisation et où il se trouve.

Vous devez modifier l'espace swap pendant le démarrage du système en mode de **rescue** ; voir [Booting Your Computer in the Rescue Mode](#) du *Red Hat Enterprise Linux 7 Installation Guide*. Lorsqu'il vous sera demandé de monter le système de fichiers, sélectionnez **Ignorer**.

14.1. Ajouter de l'espace swap

Il est parfois nécessaire d'ajouter de l'espace swap supplémentaire après l'installation. Par exemple, vous pouvez mettre à niveau la quantité de RAM dans votre système de 1 Go à 2 Go, mais il n'y a que 2 Go d'espace swap. Il pourrait être avantageux d'augmenter la quantité d'espace swap jusqu'à 4 Go si vous effectuez des opérations utilisant beaucoup de mémoire ou si vous exécutez des applications nécessitant de larges quantités de mémoire.

Trois options s'offrent à vous : créer une nouvelle partition swap, créer un nouveau fichier swap, ou étendre la mémoire swap sur un volume logique LVM2 existant. Il est recommandé d'étendre un volume logique existant.

14.1.1. Étendre Swap dans un Volume logique LVM2

Par défaut, Red Hat Enterprise Linux 7 utilise tout l'espace disponible pendant l'installation. Si c'est le cas pour votre système, vous devez ajouter un nouveau volume physique au groupe de volume utilisé par l'espace swap.

Après l'ajout de stockage supplémentaire au groupe de volume d'espace swap, il est possible de l'étendre. Pour cela, suivre la procédure suivante (si l'on considère que **/dev/VolGroup00/LogVol01** est le volume que vous souhaitez augmenter de 2 Go) :

Procédure 14.1. Étendre Swap dans un Volume logique LVM2

1. Désactiver le swapping pour le volume logique associé :

```
# swapoff -v /dev/VolGroup00/LogVol01
```

2. Modifiez la taille du volume logique LVM2 de 2 Go supplémentaires :

```
# lvresize /dev/VolGroup00/LogVol01 -L +2G
```

3. Formater le nouvel espace swap :

```
# mkswap /dev/VolGroup00/LogVol01
```

4. Activer le volume logique étendu :

```
# swapon -v /dev/VolGroup00/LogVol01
```

Pour vérifier si le nouveau volume logique a bien été créé, exécuter **cat /proc/swaps** ou **free** pour inspecter l'espace swap.

14.1.2. Création d'un volume logique LVM2 avec Swap

Pour ajouter un groupe de volumes swap (en supposant que **/dev/VolGroup00/LogVol02** est le volume swap que vous souhaitez ajouter) :

1. Créer le volume logique LVM2 avec une taille de 2 Go :

```
# lvcreate VolGroup00 -n LogVol02 -L 2G
```

2. Formater le nouvel espace swap :

```
# mkswap /dev/VolGroup00/LogVol02
```

3. Ajouter l'entrée suivante au fichier **/etc/fstab** :

```
# /dev/VolGroup00/LogVol02 swap swap defaults 0 0
```

4. Activer le volume logique étendu :

```
# swapon -v /dev/VolGroup00/LogVol02
```

Pour vérifier si le nouveau volume logique a bien été créé, exécutez **cat /proc/swaps** ou **free** pour inspecter l'espace swap.

14.1.3. Création d'un fichier Swap

Comment ajouter un fichier swap :

Procédure 14.2. Ajouter un fichier swap

1. Déterminer la taille du nouveau fichier swap en mégaoctets et multipliez-la par 1024 pour déterminer le nombre de blocs. Ainsi, la taille de bloc d'un fichier swap de 64 Mo est 65536.
2. Dans un terminal, saisir la commande suivante avec **count** correspondant à la taille de bloc souhaitée :

```
# dd if=/dev/zero of=/swapfile bs=1024 count=65536
```

3. Définir le fichier swap par la commande :

```
# mkswap /swapfile
```

4. Modifier la sécurité du swapfile de façon à ce qu'il soit accessible à la lecture par tout le monde.

```
# chmod 0600 /swapfile
```

5. Pour activer le fichier swap immédiatement, mais pas automatiquement au démarrage :

```
# swapon /swapfile
```

- Pour l'activer au démarrage, modifier le fichier `/etc/fstab` pour y inclure l'entrée suivante :

```
/swapfile swap swap defaults 0 0
```

Le nouveau fichier swap sera activé lors du prochain démarrage du système.

Pour vérifier si le nouveau fichier swap a bien été créé, exécutez `cat /proc/swaps` ou `free` pour inspecter l'espace swap.

14.2. Supprimer de l'espace swap

Parfois, il peut se révéler prudent de réduire l'espace swap après l'installation. Par exemple, supposons que vous ayez diminué la quantité de RAM dans votre système de 1 Go à 512 Mo, mais que 2 Go d'espace swap est toujours assigné. Il pourrait être plus avantageux de réduire la quantité d'espace swap à 1 Go, car 2 Go pourrait être une perte d'espace disque.

Trois options s'offrent à vous : supprimer la totalité d'un volume logique LVM2 utilisé comme espace swap, supprimer un fichier swap, ou réduire l'espace swap sur un volume logique LVM2 existant.

14.2.1. Réduire un Swap dans un Volume logique LVM2

Pour réduire un volume logique swap (en supposant que `/dev/VolGroup00/LogVol01` soit le volume swap que vous souhaitez réduire) :

Procédure 14.3. Réduire un volume logique LVM2 swap

- Désactiver le swapping pour le volume logique associé :

```
# swapoff -v /dev/VolGroup00/LogVol01
```

- Réduire la taille du volume logique LVM2 de 512 Mo :

```
# lvreduce /dev/VolGroup00/LogVol01 -L -512M
```

- Formater le nouvel espace swap :

```
# mkswap /dev/VolGroup00/LogVol01
```

- Activer le volume logique étendu :

```
# swapon -v /dev/VolGroup00/LogVol01
```

Pour vérifier si la taille du volume logique swap a bien été créée, exécuter `cat /proc/swaps` ou `free` pour inspecter l'espace swap.

14.2.2. Supprimer un volume logique LVM2 avec Swap

Pour supprimer un groupe de volumes swap (en supposant que `/dev/VolGroup00/LogVol02` soit le volume swap que vous souhaitez supprimer) :

Procédure 14.4. Comment supprimer un volume swap :

1. Désactiver le swapping pour le volume logique associé :

```
# swapoff -v /dev/VolGroup00/LogVol02
```

2. Supprimer le volume logique LVM2 d'une taille de 512 Mo :

```
# lvremove /dev/VolGroup00/LogVol02
```

3. Supprimer l'entrée suivante du fichier `/etc/fstab` :

```
/dev/VolGroup00/LogVol02 swap swap defaults 0 0
```

Pour vérifier si la taille du volume logique a bien été supprimée, exécuter `cat /proc/swaps` ou `free` pour inspecter l'espace swap.

14.2.3. Supprimer un fichier swap

Comment supprimer un fichier swap :

Procédure 14.5. Supprimer un fichier swap

1. À l'invite de commande, exécutez la commande suivante pour désactiver le fichier suivant (si `/swapfile` est le fichier swap) :

```
# swapoff -v /swapfile
```

2. Supprimer son entrée du fichier `/etc/fstab` :
3. Supprimer le fichier :

```
# rm /swapfile
```

14.3. Déplacer l'espace swap

Pour déplacer l'espace swap d'un emplacement à un autre, veuillez suivre les étapes pour supprimer l'espace swap, puis suivez les étapes pour ajouter de l'espace swap.

Chapitre 15. System Storage Manager(SSM)

System Storage Manager (SSM) fournit une interface de ligne de commande pour gérer le stockage dans diverses technologies. Les systèmes de stockage sont de plus en plus compliqués avec l'utilisation de plusieurs dispositifs de mappage ou gestionnaires de volumes logiques (LVM). Cela crée un système qui n'est pas facile à utiliser et qui facilite les erreurs et les problèmes qui se posent. SSM soulage tout cela en créant une interface utilisateur unifiée. Cette interface permet aux utilisateurs d'exécuter des systèmes compliqués de manière simple. Par exemple, pour créer et installer un nouveau système de fichiers sans SSM, il y a cinq commandes qui doivent être utilisées. Avec SSM, une seule est nécessaire.

Ce chapitre va vous expliquer comment SSM interagit avec les divers backends, puis détaillera certains cas qui reviennent souvent.

15.1. SSM Backends

SSM utilise une couche d'abstraction de `ssmlib/main.py` compatible avec les abstractions de volume, périphérique, et pool, tout en ignorant les spécificités de la technologie sous-jacente. Les backends peuvent être enregistrées dans `ssmlib/main.py` pour gérer des méthodes de technologie de stockage spécifiques comme `create`, `snapshot`, ou bien, pour supprimer `remove` les volumes et les pools.

Il y a déjà plusieurs backends enregistrées dans SSM. Les sections suivantes vous donneront des informations de base, ainsi que des définitions sur la façon de gérer ces pools, volumes, clichés et périphériques.

15.1.1. BTRFS Backend

BTRFS, un système de fichiers avec plusieurs fonctionnalités avancées, est utilisé en tant que backend de gestion de volume dans SSM. Les pools, volumes et clichés peuvent être créés avec le backend de BTRFS.

15.1.1.1. BTRFS Pool

Le système de fichiers BTRFS en soi correspond au pool. Il peut être étendu en ajoutant plus de périphériques ou rétréci en retirant des périphériques. SSM crée un système de fichiers BTRFS lorsqu'un pool BTRFS est créé. Cela signifie que chaque nouveau pool BTRFS dispose d'un volume du même nom que le pool, qui ne peut être supprimé sans supprimer l'ensemble complet du pool. Le nom par défaut du pool BTRFS est `btrfs_pool`.

Le nom du pool est utilisé comme libellé du système de fichiers. S'il y a déjà un système de fichiers BTRFS existant dans le système sans libellé, le pool BTRFS générera un nom d'utilisation interne sous le format suivant `btrfs_device_base_name`.

15.1.1.2. BTRFS Volume

Les volumes créés dans un pool après le premier volume sont comme des sous-volumes. SSM montera temporairement le système de fichiers BTRFS, si il est démonté afin de créer un volume secondaire.

Le nom d'un volume est utilisé comme chemin de sous-volume dans le système de fichiers BTRFS. Ainsi, un sous-volume s'affiche ainsi `/dev/lvm_pool/lvol001`. Chaque objet de ce chemin doit sortir pour que le volume soit créé. Les volumes peuvent également être référencés par leur point de montage.

15.1.1.3. BTRFS Snapshot

Des clichés peuvent être pris de n'importe quel volume BTRFS dans le système avec SSM. Sachez que BTRFS ne distingue pas entre sous-volumes et clichés. Même si cela signifie que le SSM ne peut pas

reconnaître la destination de chaque cliché, il va essayer de reconnaître les formats de nom spécial. Si le nom spécifié lors de la création du cliché correspond à un modèle spécifique, le cliché ne sera pas reconnu et sera, au lieu de cela, répertorié comme un volume standard de BTRFS.

15.1.1.4. Périphérique BTRFS

BTRFS ne requiert aucun périphérique particulier sur lequel être créé.

15.1.2. LVM Backend

Les pools, volumes, et clichés peuvent être créés par LVM. Les définitions suivantes sont d'un point de vue LVM.

15.1.2.1. LVM Pool

Un pool LVM est la même chose qu'un groupe de volume LVM. Cela signifie que le groupement de périphériques et de nouveaux volumes logiques peuvent être créés à partir du pool de LVM. Le nom du pool LVM par défaut est `lvm_pool`.

15.1.2.2. LVM Volume

Un volume LVM ressemble à un volume logique ordinaire.

15.1.2.3. LVM Snapshot

Quand un cliché est créé à partir d'un volume LVM, un nouveau volume de **cliché** est créé, qui pourra être manipulé comme tout autre volume LVM. À la différence de BTRFS, LVM est capable de distinguer les clichés des autres volumes habituels, donc vous n'avez pas besoin d'un nom de cliché qui cooresponde à un modèle particulier.

15.1.2.4. Périphérique LVM

SSM exige que LVM backend soit créé sur un périphérique physique transparent pour l'utilisateur.

15.1.3. Crypt

Le backend de cryptage dans SSM utilise `cryptsetup` et `dm-crypt target` pour gérer les volumes encodés. Les backends de cryptage peuvent être utilisés comme backend standards pour créer des volumes encodés sur les périphériques encodés (ou sur d'autres volumes comme les volumes LVM ou MD), ou pour créer des volumes LVM encodés en une seule étape.

Seuls les volumes peuvent être créés avec un backend Crypt ; le pooling n'est pas pris en charge et ne requiert aucun périphérique particulier.

Les sections suivantes définissent des volumes et des clichés d'une point de vue Crypt.

15.1.3.1. Crypt Volume

Les volumes de cryptage sont créés par la commande `dm-crypt` et représentent les données qui se trouvent dans le périphérique encodé d'origine sous une forme non encodée. Ne supporte aucune concatenation de périphérique ou RAID.

Deux modes, ou extensions, sont prises en charge : `luks` et `plain`. `luks` est utilisé par défaut. Pour obtenir plus d'informations sur les extensions, consulter `man cryptsetup`.

15.1.3.2. Cliché Crypt

Bien que le backend de cryptage ne prenne pas en charge les clichés, si le volume encodé est créé sur un volume LVM, le volume lui-même pourra être transformé en cliché. Le cliché pourra alors être ouvert par `cryptsetup`.

15.1.4. Multiple Devices (MD)

Le MD backend se contente actuellement de collecter des informations sur les volumes MD du système.

15.2. Tâches SSM courantes

Les sections suivantes développeront un certain nombre de cas d'utilisation sur la façon d'installer SSM, puis d'afficher les informations sur tous les périphériques, pools et volumes détectés. Ensuite, un pool sera créé comprenant deux volumes et un système de fichiers XFS. Le système de fichiers sera alors vérifié pour détecter les problèmes d'homogénéité, et un volume augmentera en taille. Puis, un cliché sera créé. Enfin, un des volumes sera supprimé.

15.2.1. Installation du Gestionnaire de stockage de système (SSM)

Pour installer SSM, utiliser la commande suivante :

```
# yum install system-storage-manager
```

Il y a plusieurs backends qui ne sont activés que si les packages suivants sont installés :

- ✦ Le backend LVM requiert le package `lvm2`.
- ✦ Le BTRFS requiert le package `btrfs-progs`.
- ✦ Le backend de cryptage requiert les packages `device-mapper` et `cryptsetup`.

15.2.2. Afficher les informations sur tous les périphériques détectés

L'affichage des informations à propos des périphériques, pools, volumes, et clichés est effectué grâce à la commande `list`. Exécuter la commande `ssm list` sans options affichera ce qui suit :

```
~]# ssm list
-----
Device          Free      Used      Total  Pool  Mount point
-----
/dev/sda                2.00 GB          PARTITIONED
/dev/sda1             47.83 MB          /test
/dev/vda                15.00 GB          PARTITIONED
/dev/vda1             500.00 MB          /boot
/dev/vda2  0.00 KB  14.51 GB   14.51 GB   rhel
-----
Pool  Type  Devices      Free      Used      Total
-----
rhel  lvm   1           0.00 KB  14.51 GB  14.51 GB
-----
-----
```

Volume point	Pool	Volume size	FS	FS size	Free	Type	Mount
/dev/rhel/root	rhel	13.53 GB	xf	13.52 GB	9.64 GB	linear	/
/dev/rhel/swap	rhel	1000.00 MB				linear	
/dev/sda1		47.83 MB	xf	44.50 MB	44.41 MB	part	/test
/dev/vda1		500.00 MB	xf	496.67 MB	403.56 MB	part	/boot

Cet affichage pourra être réduit davantage en utilisant des arguments qui spécifieront ce qui doit être affiché. La liste des options disponibles se trouve dans la commande **ssm list --help**.



Note

Selon l'argument qui sera donné, SSM risque de ne pas tout afficher.

- ✦ Exécuter l'argument **devices** ou **dev** va omettre certains périphériques. Les CDRom et DM/MD, par exemple, sont cachés intentionnellement car ils sont listés en tant que volumes.
- ✦ Certains backends ne supportent pas les clichés et ne peut pas faire la distinction entre un cliché et un volume standard. Exécuter l'argument **snapshot** sur un de ces backends amènera SSM à tenter de reconnaître le nom de volume afin d'identifier un cliché. Si l'expression régulière de la SSM ne correspond pas au modèle du cliché, la capture du cliché ne sera pas reconnue.
- ✦ À l'exception du volume BTRFS principal (le système de fichiers lui-même), les volumes BTRFS non montés n'apparaîtront pas.

15.2.3. Créer un nouveau pool, volume logique ou système de fichiers

Dans cette section, un nouveau pool sera créé avec un nom par défaut. Il comprendra les périphériques **/dev/vdb** et **/dev/vdc**, un volume logique de 1G, et un système de fichiers XFS.

La commande pour créer ce scénario est la suivante : **ssm create --fs xfs -s 1G /dev/vdb /dev/vdc**. Les options suivantes sont utilisées :

- ✦ L'option **--fs** indique le type de système de fichiers requis. Les types de systèmes de fichiers actuellement pris en charge sont les suivants :
 - ext3
 - ext4
 - xfs
 - btrfs
- ✦ L'argument **-s** indique la taille du volume logique. Les suffixes suivants sont pris en charge pour définir les unités suivantes :
 - **K** or **k** pour kilooctets
 - **M** ou **m** pour megaoctets
 - **G** ou **g** pour gigaoctets

- **T** ou **t** pour teraoctets
 - **P** ou **p** pour petaoctets
 - **E** ou **e** pour exaoctets
- ✦ Les deux périphériques listés, **/dev/vdb** et **/dev/vdc**, sont les deux périphériques que je souhaite créer.

```
~]# ssm create --fs xfs -s 1G /dev/vdb /dev/vdc
Physical volume "/dev/vdb" successfully created
Physical volume "/dev/vdc" successfully created
Volume group "lvm_pool" successfully created
Logical volume "lvol001" created
```

Il existe deux autres options pour la **ssm command** pouvant s'avérer utiles. La première est la **-p pool**. Cela spécifie le pool, sur lequel le volume doit être créé. Si elle n'existe pas encore, SSM le créera. Cela a été omis dans l'exemple ci-dessus, ce qui a causé SSM à utiliser le nom par défaut **lvm_pool**. Toutefois, pour utiliser un nom spécifique qui puisse s'adapter à toutes les conventions de nommage, l'option **-p** doit être utilisée.

La seconde option utile est **-n name**. Elle permet de nommer le volume logique nouvellement créé. Tout comme avec **-p**, elle est utile pour utiliser un nom spécifique qui devra convenir pour toutes les conventions de nommage existantes.

Voici un exemple d'utilisation de ces deux options :

```
~]# ssm create --fs xfs -p new_pool -n XFS_Volume /dev/vdd
Volume group "new_pool" successfully created
Logical volume "XFS_Volume" created
```

SSM a maintenant créé deux volumes physiques, un pool, et un volume logique confortablement en une seule commande.

15.2.4. Vérifier l'homogénéité du système de fichiers

La commande **ssm check** vérifie la consistance du système de fichiers sur le volume. Il est possible de spécifier plusieurs volumes à vérifier. S'il n'y a aucun système de fichiers sur le volume, alors on pourra éviter ce volume.

Pour vérifier tous les périphériques sur le volume **lvol001**, exécuter la commande **ssm check /dev/lvm_pool/lvol001**.

```
~]# ssm check /dev/lvm_pool/lvol001
Checking xfs file system on '/dev/mapper/lvm_pool-lvol001'.
Phase 1 - find and verify superblock...
Phase 2 - using internal log
- scan filesystem freespace and inode maps...
- found root inode chunk
Phase 3 - for each AG...
- scan (but don't clear) agi unlinked lists...
- process known inodes and perform inode discovery...
- agno = 0
- agno = 1
- agno = 2
```

```

- agno = 3
- agno = 4
- agno = 5
- agno = 6
- process newly discovered inodes...
Phase 4 - check for duplicate blocks...
- setting up duplicate extent list...
- check for inodes claiming duplicate blocks...
- agno = 0
- agno = 1
- agno = 2
- agno = 3
- agno = 4
- agno = 5
- agno = 6
No modify flag set, skipping phase 5
Phase 6 - check inode connectivity...
- traversing filesystem ...
- traversal finished ...
- moving disconnected inodes to lost+found ...
Phase 7 - verify link counts...
No modify flag set, skipping filesystem flush and exiting.

```

15.2.5. Augmenter une taille de volume

La commande **ssm resize** change la taille du volume et du système de fichiers spécifiés. S'il n'y a pas de système de fichiers, alors seule la taille du volume sera ajustée.

Dans cet exemple, nous avons actuellement un volume logique dans **/dev/vdb** de 900Mo nommé **lv01001**.

```

~]# ssm list
-----
Device          Free          Used          Total  Pool          Mount point
-----
/dev/vda                15.00 GB          PARTITIONED
/dev/vda1                500.00 MB          /boot
/dev/vda2    0.00 KB    14.51 GB    14.51 GB    rhel
/dev/vdb    120.00 MB    900.00 MB    1.00 GB    lvm_pool
/dev/vdc                1.00 GB
-----
-----
Pool      Type  Devices      Free          Used          Total
-----
lvm_pool  lvm   1           120.00 MB    900.00 MB    1020.00 MB
rhel      lvm   1            0.00 KB    14.51 GB     14.51 GB
-----
-----
Volume      Pool      Volume size  FS      FS size      Free
Type      Mount point
-----
/dev/rhel/root    rhel      13.53 GB    xfs     13.52 GB     9.64 GB
linear /

```

```

/dev/rhel/swap      rhel      1000.00 MB
linear
/dev/lvm_pool/lvol001  lvm_pool    900.00 MB  xfs  896.67 MB  896.54 MB
linear
/dev/vda1          500.00 MB  xfs  496.67 MB  403.56 MB
part    /boot
-----
-----

```

Le volume logique a besoin d'augmenter de 500Mo supplémentaires. Pour cela, nous devons ajouter un nouveau périphérique au pool :

```

~]# ssm resize -s +500M /dev/lvm_pool/lvol001 /dev/vdc
  Physical volume "/dev/vdc" successfully created
  Volume group "lvm_pool" successfully extended
Phase 1 - find and verify superblock...
Phase 2 - using internal log
          - scan filesystem freespace and inode maps...
          - found root inode chunk
Phase 3 - for each AG...
          - scan (but don't clear) agi unlinked lists...
          - process known inodes and perform inode discovery...
          - agno = 0
          - agno = 1
          - agno = 2
          - agno = 3
          - process newly discovered inodes...
Phase 4 - check for duplicate blocks...
          - setting up duplicate extent list...
          - check for inodes claiming duplicate blocks...
          - agno = 0
          - agno = 1
          - agno = 2
          - agno = 3
No modify flag set, skipping phase 5
Phase 6 - check inode connectivity...
          - traversing filesystem ...
          - traversal finished ...
          - moving disconnected inodes to lost+found ...
Phase 7 - verify link counts...
No modify flag set, skipping filesystem flush and exiting.
  Extending logical volume lvol001 to 1.37 GiB
  Logical volume lvol001 successfully resized
meta-data=/dev/mapper/lvm_pool-lvol001 isize=256    agcount=4, agsize=57600
blks
          =                               sectsz=512   attr=2, projid32bit=1
          =                               crc=0
data      =                               bsize=4096   blocks=230400, imaxpct=25
          =                               sunit=0      swidth=0 blks
naming    =version 2                       bsize=4096   ascii-ci=0  ftype=0
log       =internal                        bsize=4096   blocks=853, version=2
          =                               sectsz=512   sunit=0 blks, lazy-count=1
realtime  =none                             extsz=4096   blocks=0, rtextents=0
data blocks changed from 230400 to 358400

```


SSM exécute un contrôle sur le périphérique, puis l'étend au volume à hauteur du montant spécifié. Cela peut être vérifié par la commande **ssm list**.

```
~]# ssm list
-----
Device           Free           Used           Total  Pool           Mount point
-----
/dev/vda                15.00 GB                PARTITIONED
/dev/vda1              500.00 MB                /boot
/dev/vda2    0.00 KB    14.51 GB    14.51 GB    rhel
/dev/vdb     0.00 KB    1020.00 MB    1.00 GB    lvm_pool
/dev/vdc    640.00 MB    380.00 MB    1.00 GB    lvm_pool
-----

Pool      Type  Devices      Free           Used           Total
-----
lvm_pool  lvm   2            640.00 MB    1.37 GB    1.99 GB
rhel      lvm   1            0.00 KB    14.51 GB    14.51 GB
-----

Volume      Pool      Volume size  FS      FS size      Free
Type      Mount point
-----
/dev/rhel/root      rhel            13.53 GB  xfs      13.52 GB    9.64 GB
linear /
/dev/rhel/swap      rhel            1000.00 MB
linear
/dev/lvm_pool/lvol001  lvm_pool      1.37 GB  xfs      1.36 GB    1.36 GB
linear
/dev/vda1                500.00 MB  xfs      496.67 MB    403.56 MB
part /boot
-----
```



Note

Il est seulement possible de diminuer la taille d'un volume LVM ; non supporté pour les autres types de volumes. On y parvient en exécutant - au lieu de +. Ainsi, pour augmenter la taille d'un volume LVM de 50M, la commande sera :

```
~]# ssm resize -s-50M /dev/lvm_pool/lvol002
Rounding size to boundary between physical extents: 972.00 MiB
WARNING: Reducing active logical volume to 972.00 MiB
THIS MAY DESTROY YOUR DATA (filesystem etc.)
Do you really want to reduce lvol002? [y/n]: y
Reducing logical volume lvol002 to 972.00 MiB
Logical volume lvol002 successfully resized
```

Sans + ou -, la valeur sera considérée comme une valeur absolue.

15.2.6. Cliché

Pour prendre un cliché d'un volume existant, exécuter la commande **ssm snapshot**.



Note

Cette opération échouera si le backend auquel le volume appartient ne supporte pas les clichés.

Pour créer un cliché de **lv1001**, exécuter la commande suivante :

```
~]# ssm snapshot /dev/lvm_pool/lvol001
Logical volume "snap20150519T130900" created
```

Pour vérifier ceci, exécuter la commande **ssm list**, et noter la section de cliché supplémentaire.

```
~]# ssm list
-----
Device          Free          Used          Total Pool          Mount point
-----
/dev/vda                15.00 GB
/dev/vda1                500.00 MB      /boot
/dev/vda2  0.00 KB    14.51 GB    14.51 GB  rhel
/dev/vdb   0.00 KB   1020.00 MB    1.00 GB  lvm_pool
/dev/vdc                1.00 GB
-----

Pool    Type  Devices    Free          Used          Total
-----
lvm_pool  lvm   1          0.00 KB    1020.00 MB    1020.00 MB
rhel      lvm   1          0.00 KB    14.51 GB     14.51 GB
-----

Volume          Pool          Volume size  FS          FS size      Free
Type    Mount point
-----
/dev/rhel/root      rhel          13.53 GB    xfs         13.52 GB     9.64 GB
linear /
/dev/rhel/swap      rhel          1000.00 MB
linear
/dev/lvm_pool/lvol001 lvm_pool      900.00 MB    xfs         896.67 MB     896.54 MB
linear
/dev/vda1                500.00 MB    xfs         496.67 MB     403.56 MB
part    /boot
-----

Snapshot          Origin  Pool          Volume size      Size
Type
-----
```

```
/dev/lvm_pool/snap20150519T130900  lvol001  lvm_pool  120.00 MB  0.00 KB
linear
-----
-----
```

15.2.7. Enlever un élément

La commande **ssm remove** est utilisée pour supprimer un élément, comme un périphérique, un pool ou un volume.



Note

Si un périphérique est utilisé par un pool, s'il est supprimé, il échouera. On peut forcer ce comportement en utilisant l'argument **-f**.

Si le volume est monté, puis supprimé, il échouera. À la différence du périphérique, le comportement pourra être forcé par l'argument **-f**.

Pour supprimer **lvm_pool** et tout ce qui se trouve à l'intérieur, exécuter la commande suivante :

```
~]# ssm remove lvm_pool
Do you really want to remove volume group "lvm_pool" containing 2 logical
volumes? [y/n]: y
Do you really want to remove active logical volume snap20150519T130900?
[y/n]: y
Logical volume "snap20150519T130900" successfully removed
Do you really want to remove active logical volume lvol001? [y/n]: y
Logical volume "lvol001" successfully removed
Volume group "lvm_pool" successfully removed
```

15.3. Ressources SSM

Vous trouverez des informations supplémentaires sur SSM dans les ressources suivantes :

- ✦ La page **man ssm** fournit de bonnes descriptions et de bons exemples, ainsi que de nombreux détails sur toutes les commandes et options qui sont trop spécifiques pour pouvoir être documentés ici.
- ✦ La documentation locale de SSM se trouve dans le répertoire **doc/**.
- ✦ Le wiki SSM est accessible à partir de cette adresse <http://storagemanager.sourceforge.net/index.html>.
- ✦ On peut s'abonner à la liste de diffusion à cet endroit <https://lists.sourceforge.net/lists/listinfo/storagemanager-devel> et aux archives des listes de diffusion à cet endroit http://sourceforge.net/mailarchive/forum.php?forum_name=storagemanager-devel. La liste de diffusion est l'endroit où les développeurs communiquent entre eux. Il n'y a actuellement aucune liste de diffusion utilisateur, donc sentez-vous libre de poser des questions également.

Chapitre 16. Quotas de disques

L'espace disque peut être restreint en implémentant des quotas de disques qui alerteraient un administrateur système avant qu'un utilisateur ne consomme trop d'espace disque ou qu'une partition ne soit pleine.

Les quotas de disques peuvent être configurés pour des utilisateurs individuels ainsi que pour des groupes d'utilisateurs. Ceci permet de gérer l'espace alloué pour les fichiers spécifiques aux utilisateurs (par exemple pour les courriers électroniques) séparément de l'espace alloué aux projets sur lesquels un utilisateur travaille (en supposant que ces projets se voient offrir leurs propres groupes).

En outre, des quotas peuvent être définis, non seulement pour contrôler le nombre de blocs de disques consommés, mais aussi pour contrôler le nombre d'inodes (des structures de données contenant des informations sur les fichiers dans les systèmes de fichiers UNIX). Comme les inodes sont utilisés pour contenir des informations concernant des fichiers, cela permet de contrôler le nombre de fichiers pouvant être créés.

Le RPM **quota** doit être installé pour implémenter les quotas de disques.



Note

Ce chapitre concerne tous les systèmes de fichiers, cependant, certains systèmes de fichiers ont leurs propres outils de gestion des quota. Voir la description correspondante pour les systèmes de fichiers applicables.

Pour les systèmes de fichiers XFS, voir [Section 6.3, « Gestion des quotas XFS »](#).

Btrfs n'a pas de quota de disque donc n'est pas couvert.

16.1. Configurer les quotas de disques

Pour fixer les quotas de disques, procédez aux étapes suivantes :

1. Activez les quotas par système de fichiers en modifiant le fichier **/etc/fstab**.
2. Remontez le(s) système(s) de fichiers.
3. Créez les fichiers de la base de données des quotas et générez le tableau d'utilisation du disque.
4. Assignez les politiques des quotas.

Chaque étape sera discutée en détails dans les sections suivantes.

16.1.1. Activer les quotas

En tant que root et à l'aide d'un éditeur de texte, modifiez le fichier **/etc/fstab**.

Exemple 16.1. Modifiez **/etc/fstab**

Par exemple, pour utiliser l'éditeur de texte **vim**, veuillez saisir :

```
# vim /etc/fstab
```

Ajoutez les options **usrquota** ou **grpquota** aux systèmes de fichiers qui requièrent des quotas :

Exemple 16.2. Ajoutez les quotas

```

/dev/VolGroup00/LogVol00 /          ext3    defaults      1 1
LABEL=/boot              /boot  ext3    defaults      1 2
none                    /dev/pts devpts  gid=5,mode=620 0 0
none                    /dev/shm tmpfs   defaults      0 0
none                    /proc   proc     defaults      0 0
none                    /sys    sysfs   defaults      0 0
/dev/VolGroup00/LogVol02 /home  ext3    defaults,usrquota,grpquota 1 2
/dev/VolGroup00/LogVol01 swap   swap    defaults      0 0 . . .

```

Dans cet exemple, le système de fichiers **/home** possède des quotas utilisateurs et groupes activés.



Note

Les exemples suivants supposent qu'une autre partition **/home** ait été créée pendant l'installation de Red Hat Enterprise Linux. La partition root (**/**) peut être utilisée pour définir les politiques de quotas dans le fichier **/etc/fstab**.

16.1.2. Remonter les systèmes de fichiers

Après avoir ajouté les options **usrquota** ou **grpquota**, veuillez remonter chaque système de fichier dont l'entrée **fstab** a été modifiée. Si le système de fichiers n'est pas en cours d'utilisation par un processus, veuillez utiliser l'une des méthodes suivantes :

- ✦ Saisissez la commande **umount** suivie de la commande **mount** afin de remonter le système de fichiers. Veuillez consulter la page **man** pour **umount** et **mount** afin de connaître la syntaxe particulière pour monter et démonter divers types de systèmes de fichiers.
- ✦ Saisissez la commande **mount -o remount file-system** (où **file-system** est le nom du système de fichiers) afin de remonter le système de fichiers. Par exemple, pour remonter le système de fichiers **/home**, veuillez utiliser la commande **mount -o remount /home**.

Si le système de fichiers est en cours d'utilisation, la plus simple méthode pour remonter le système de fichiers est de redémarrer le système.

16.1.3. Créer les fichiers de base de données de quotas

Après avoir remonté chaque système de fichiers dont les quotas sont activés, veuillez exécuter la commande **quotacheck**.

La commande **quotacheck** examine les systèmes de fichier dont les quotas sont activés et crée un tableau de l'utilisation actuelle du disque par système de fichiers. Le tableau est ensuite utilisé pour mettre à jour la copie de l'utilisation du disque du système d'exploitation. En outre, les fichiers de quotas du système de fichiers sont aussi mis à jour.



Note

La commande **quotacheck** n'a aucun effet sur XFS car le tableau d'utilisation de disque est complété automatiquement au moment du montage. Voir la page man de **xfs_quota(8)** pour plus d'informations.

Pour créer les fichiers de quotas (**aquota.user** et **aquota.group**) sur le système de fichiers, veuillez utiliser l'option **-c** de la commande **quotacheck**.

Exemple 16.3. Créer des fichiers de quotas

Par exemple, si les quotas d'utilisateurs et de groupes sont activés pour le système de fichiers **/home**, veuillez créer les fichiers dans le répertoire **/home** :

```
# quotacheck -cug /home
```

L'option **-c** indique que des fichiers de quotas doivent être créés pour chaque système de fichiers sur lequel les quotas sont activés, l'option **-u** indique que les quotas d'utilisateurs seront vérifiés, et l'option **-g** indique que les quotas de groupes seront vérifiés.

Si les options **-u** ou **-g** ne sont pas spécifiées, seul le fichier du quota d'utilisateurs sera créé. Si seule l'option **-g** est spécifiée, seul le fichier du quota de groupes sera créé.

Une fois que les fichiers sont créés, veuillez exécuter la commande suivante afin de générer le tableau d'utilisation du disque actuelle par système de fichiers avec quotas activés :

```
# quotacheck -avug
```

Les options utilisées sont comme suit :

a

Vérifie tous les systèmes de fichiers montés localement avec quotas activés

v

Affiche les informations détaillées pendant la progression de la vérification du quota

u

Vérifie les informations du quota de disques de l'utilisateur

g

Vérifie les informations du quota de disques de groupe

Une fois que **quotacheck** a terminé son exécution, les fichiers de quotas correspondants aux quotas activés (d'utilisateurs ou de groupes) sont remplis avec des données pour chaque système de fichiers monté localement avec quotas activés, tel que **/home**.

16.1.4. Allouer les quotas par utilisateur

La dernière étape consiste à assigner les quotas de disques avec la commande **edquota**.

Pour configurer le quota pour un utilisateur, en tant que root, utiliser la commande suivante :

```
# edquota username
```

Procédez à cette étape pour chaque utilisateur qui a besoin d'un quota. Ainsi, si un quota est activé dans `/etc/fstab` pour la partition `/home (/dev/VolGroup00/LogVol102` dans l'exemple ci-dessous) et que la commande `edquota testuser` est exécutée, vous verrez ce qui suit dans l'éditeur configuré par défaut dans le système :

```
Disk quotas for user testuser (uid 501):
Filesystem          blocks      soft      hard      inodes     soft      hard
/dev/VolGroup00/LogVol102 440436      0         0         37418      0         0
```



Note

L'éditeur de texte défini par la variable d'environnement **EDITOR** est utilisé par **edquota**. Pour changer l'éditeur, définir la variable d'environnement **EDITOR** dans votre fichier `~/.bash_profile` vers le chemin d'accès de l'éditeur de votre choix.

La première colonne correspond au nom du système de fichiers qui contient un quota activé. La seconde colonne montre combien de blocs l'utilisateur utilise actuellement. Les deux colonnes suivantes sont utilisées pour fixer des limites de blocs « soft » ou « hard » pour l'utilisateur sur le système de fichiers. La colonne **inodes** affiche le nombre d'inodes actuellement en cours d'utilisation par l'utilisateur. Les deux dernières colonnes sont utilisées pour définir les limites d'inode « soft » et « hard » pour l'utilisateur sur le système de fichiers.

La limite hard bloc correspond au montant maximum d'espace disque qu'un utilisateur ou un groupe peut utiliser. Une fois que la limite est atteinte, on ne peut pas utiliser d'espace supplémentaire.

La limite de bloc « soft » détermine le montant maximum de d'espace disque pouvant être utilisé. Cependant, à la différence de la limite « hard », la limite « soft » peut être dépassée jusqu'à une certaine limite. C'est ce que l'on appelle la *période de grâce*. La période de grâce peut être exprimée en secondes, minutes, heures, jours, semaines, ou mois.

Si une seule des valeurs est fixée à 0, cette limite n'est pas définie. Dans l'éditeur de texte, veuillez changer les limites que vous souhaitez.

Exemple 16.4. Modifier les limites souhaitées

Par exemple :

```
Disk quotas for user testuser (uid 501):
Filesystem          blocks      soft      hard      inodes     soft      hard
/dev/VolGroup00/LogVol102 440436    500000    550000    37418      0         0
```

Pour vous assurer que le quota utilisateur a bien été défini, utiliser la commande :

```
# quota username
Disk quotas for user username (uid 501):
Filesystem blocks  quota  limit  grace  files  quota  limit
grace
```

```
/dev/sdb 1000* 1000 1000 0 0 0
```

16.1.5. Assigner les quotas par groupe

Les quotas peuvent aussi être assignés sur une base « par groupe ». Par exemple, pour définir un quota de groupe pour le groupe **devel** (le groupe doit exister avant de définir le quota du groupe), veuillez utiliser la commande :

```
# edquota -g devel
```

Cette commande affiche le quota existant pour le groupe dans l'éditeur de texte :

```
Disk quotas for group devel (gid 505):
Filesystem          blocks      soft      hard      inodes      soft      hard
/dev/VolGroup00/LogVol102 440400      0         0         37418       0         0
```

Modifiez les limites, puis enregistrez le fichier.

Pour vérifier que le quota de groupe a bien été défini, veuillez utiliser la commande :

```
# quota -g devel
```

16.1.6. Définir la période de grâce pour les limites soft

Si un quota donné possède des limites « soft », vous pouvez modifier la période de grâce (la période pendant laquelle la limite « soft » peut être dépassée) avec la commande suivante :

```
# edquota -t
```

Cette commande fonctionne sur les quotas pour les inodes ou les blocs pour utilisateurs ou pour groupes.



Important

Tandis que les autres commandes **edquota** opèrent sur les quotas d'un utilisateur ou d'un groupe particulier, l'option **-t** opère sur tous les systèmes de fichiers dont les quotas sont activés.

16.2. Gérer les quotas de disque

Si les quotas sont mis en œuvre, ils devront être maintenus, surtout sous une forme qui permette de voir si les quotas sont dépassés et de vérifier qu'ils soient exacts.

Bien entendu, si les utilisateurs dépassent constamment leur quota ou leur limite, un administrateur de système aura un choix limité suivant le type d'utilisateur ou suivant l'impact de l'espace disque sur leur travail. L'administrateur pourra soit aider l'utilisateur à déterminer comment utiliser moins d'espace disque ou augmenter le quota de disque de l'utilisateur.

16.2.1. Activation et désactivation

Il est possible de désactiver les quotas sans les définir sur 0. Pour désactiver tous les quotas d'utilisateur et de groupe, veuillez utiliser la commande suivante :


```
# quotaoff -vaug
```

Si les options **-u** ou **-g** ne sont pas spécifiées, seuls les quotas d'utilisateur seront désactivés. Si seule l'option **-g** est spécifiée, seuls les quotas de groupe seront désactivés. L'interrupteur **-v** provoque l'affichage des informations verbeuses du statut lorsque la commande est exécutée.

Pour activer les quotas à nouveau, veuillez utiliser la commande **quotaon** avec les mêmes options.

Par exemple, pour actier les quotas d'utilisateur et de groupe pour tous les systèmes de fichiers, veuillez utiliser la commande suivante :

```
# quotaon -vaug
```

Pour activer les quotas pour un système de fichiers particulier, tel que **/home**, veuillez utiliser la commande suivante :

```
# quotaon -vug /home
```

Si les options **-u** ou **-g** ne sont pas spécifiées, seuls les quotas d'utilisateur seront activés. Si seule l'option **-g** est spécifiée, seuls les quotas de groupe seront activés.



Note

La commande **quotaon** n'est pas toujours requise sur XFS car elle est exécutée automatiquement au moment du montage. Voir la page man de **xfquota(8)** pour plus d'informations.

16.2.2. Rapports sur les quotas de disques

Créer un rapport sur l'utilisation du disque suppose l'exécution de l'utilitaire **repquota**.

Exemple 16.5. Sortie de la commande repquota

Par exemple, la commande **repquota /home** produit la sortie suivante :

```
*** Report for user quotas on device /dev/mapper/VolGroup00-LogVol02
Block grace time: 7days; Inode grace time: 7days
  Block limits  File limits
User  used soft hard grace used soft hard grace
-----
root   --    36      0      0           4      0      0
kristin --   540      0      0          125     0      0
testuser -- 440400 500000 550000    37418    0      0
```

Pour afficher un rapport d'utilisation de disque pour tous les systèmes de fichiers dont les quotas sont activés (**-a**), utiliser la commande suivante :

```
# repquota -a
```

Même si le rapport est facile à lire, certains points doivent être expliqués. Les caractères `--` qui sont affichés après chaque utilisateur servent à déterminer si les limites du bloc ou de l'inode ont été dépassées. Si la limite soft est dépassée, le caractère `+` apparaîtra à la place du caractère `-` correspondant ; le premier caractère `-` représente la limite du bloc, et le second représente la limite de l'inode.

Les colonnes **grace** sont normalement vides. Si une limite soft a été dépassée, la colonne affichera le temps correspondant au délai restant de la période de grâce. Si la période de grâce a été dépassée, la chaîne **none** (aucun) s'affichera à sa place.

16.2.3. Contrôler l'exactitude des quotas

Lorsqu'un système de fichiers échoue à effectuer un démontage correct (par exemple, dû à une panne du système), il est nécessaire d'exécuter **quotacheck**. Cependant, **quotacheck** peut être exécuté de manière régulière, même si le système n'est pas tombé en panne. Les méthodes sûres pour exécuter **quotacheck** périodiquement incluent :

L'assurance que **quotacheck** sera exécuté lors du prochain redémarrage



Note

Cette méthode fonctionne mieux pour les systèmes multi-utilisateur (occupés) qui sont redémarrés périodiquement.

En tant qu'utilisateur root, veuillez placer un script shell dans le répertoire `/etc/cron.daily/` ou `/etc/cron.weekly/` qui contient la commande **touch /forcequotacheck** — ou planifiez-en un à l'aide de la commande **crontab -e**. Ceci crée un fichier **forcequotacheck** vide dans le répertoire root, que le script init du système cherchera lors du démarrage. Si celui-ci est trouvé, le script init exécutera **quotacheck**. Puis le script init supprimera le fichier `/forcequotacheck` ; ainsi, la planification de la création périodique de ce fichier avec **cron** assure que **quotacheck** soit effectivement exécuté lors du prochain redémarrage.

Pour obtenir davantage d'informations sur **cron**, veuillez consulter **man cron**.

Exécuter **quotacheck** en mode mono-utilisateur

Une manière alternative d'exécuter **quotacheck** en toute sécurité consiste à démarrer le système en mode mono-utilisateur, ce qui empêche toute possibilité de corruption de données dans les fichiers de quota, puis d'exécuter les commandes suivantes :

```
# quotaoff -vug /file_system
```

```
# quotacheck -vug /file_system
```

```
# quotaon -vug /file_system
```

Exécuter **quotacheck** sur un système en cours d'exécution

Si nécessaire, il est possible d'exécuter **quotacheck** sur un ordinateur à un moment où aucun utilisateur n'est connecté, ainsi il n'y aura aucun fichier ouvert sur le système en cours de vérification. Exécutez la commande **quotacheck -vug file_system** ; cette commande échouera si **quotacheck** ne peut pas monter à nouveau le système de fichiers donné `file_system`

en lecture seule. Remarquez qu'après la vérification, le système de fichiers sera remonté en lecture-écriture.



Avertissement

Exécuter **quotacheck** sur un système de fichiers en cours d'exécution monté en lecture-écriture n'est pas recommandé à cause de la possibilité de corruption de fichier(s) de quota.

Veuillez consulter **man cron** pour obtenir davantage d'informations sur la configuration de **cron**.

16.3. Références de quotas de disques

Pour obtenir plus d'informations sur les quota de disques, voir les pages **man** des commandes suivantes :

- » **quotacheck**
- » **edquota**
- » **repquota**
- » **quota**
- » **quotaon**
- » **quotaoff**

Chapitre 17. Réseau redondant de disques indépendants (RAID, de l'anglais « Redundant Array of Independent Disks »)

L'idée principale de RAID consiste à combiner plusieurs lecteurs de disques de petite taille et peu chers dans une matrice dans le but de fournir des performances ou une redondance qui ne seraient pas réalisables avec un seul disque à coût élevé. Cette matrice de disques apparaîtra à l'ordinateur comme étant une seule unité de stockage logique ou un seul disque.

RAID permet d'étaler des informations sur plusieurs disques. RAID utilise des techniques telles que l'*Entrelacement de disques* (RAID niveau 0), la *Mise en miroir de disques* (RAID niveau 1), et l'*Entrelacement de disques avec parité* (RAID niveau 5) permettant une meilleure redondance, une latence plus basse, une augmentation de la bande passante, et la maximisation des capacités de récupération après des échecs de disque dur.

RAID distribue des données sur chaque disque de la matrice en les divisant de manière consistante (habituellement avec une taille de 256K ou 512k, même si d'autres valeurs sont également acceptables). Chaque « morceau » est ensuite écrit sur le disque dur de la matrice RAID selon le niveau RAID employé. Lorsque les données sont lues, le processus est inversé, donnant l'illusion que les multiples disques de la matrice forment un seul et unique disque.

Les administrateurs systèmes et autres personnes gérant de grandes quantités de données bénéficieront de l'utilisation de la technologie RAID. Les raisons principales pour déployer RAID incluent :

- » Amélioration de la vitesse
- » Augmentation des capacités de stockage à l'aide d'un seul disque virtuel
- » Perte de données due aux échecs de disque minimisée

17.1. Types RAID

Trois approches RAID sont possibles : RAID microprogramme, RAID matériel, et RAID logiciel.

RAID microprogramme

RAID microprogramme (aussi appelé ATARAID) est un type de RAID logiciel dans lequel les ensembles RAID peuvent être configurés à l'aide d'un menu basé sur microprogramme. Le microprogramme utilisé par ce type RAID s'intègre également au BIOS, vous permettant ainsi de démarrer à partir de ses ensembles RAID. Différents fournisseurs utilisent différents formats de métadonnées sur disque pour marquer les membres définis RAID. La matrice Intel Matrix RAID est un bon exemple de système RAID microprogramme.

Matériel RAID

La matrice basée matériel gère le sous-système RAID indépendamment de l'hôte. Celle-ci ne présente qu'un seul disque par matrice RAID à l'hôte.

Un périphérique RAID matériel peut être interne ou externe au système. Les périphériques internes consistent habituellement en une carte contrôleur spécialisée qui peut gérer les tâches RAID de manière transparente par rapport au système d'exploitation. Les périphériques externes se connectent habituellement au système via SCSI, Fibre Channel, iSCSI, InfiniBand, ou à l'aide d'autres interconnexions réseau à grande vitesse et présentent des volumes logiques au système.

Pour le système d'exploitation, les cartes contrôleur RAID fonctionnent comme un contrôleur SCSI et gèrent toutes les communications du disque. L'utilisateur branche le disque sur le contrôleur RAID (comme un contrôleur RAID normal) et les ajoute à la configuration des contrôleurs RAID. Le système d'exploitation ne verra aucune différence.

RAID logiciel

RAID logiciel implémente les divers niveaux RAID dans le code du disque du noyau (périphérique bloc). Celui-ci offre la solution la moins chère possible, car des cartes contrôleur ou châssis enfichables à chaud [2] ne sont pas requis. RAID logiciel fonctionne également avec des disques IDE moins chers ainsi qu'avec des disques SCSI. Avec les CPU plus rapides modernes, RAID logiciel offre de meilleures performances qu'avec RAID matériel.

Le noyau Linux contient un pilote MD (« *multi-disk* ») permettant à la solution RAID d'être complètement indépendante du matériel. Les performances d'une matrice basée logiciel dépendent des performances et de la charge du CPU du serveur.

Ci-dessous figurent quelques fonctionnalités-clé de la pile RAID logiciel Linux :

- » Conception multi-threads
- » Portabilité des matrices entre ordinateurs Linux sans reconstruction
- » Reconstruction de matrices en arrière-plan à l'aide des ressources système inactives
- » Prise en charge des disques enfichables à chaud
- » Détection de CPU automatique pour tirer profit de certaines fonctionnalités, comme la prise en charge du streaming SIMD
- » Correction automatique des mauvais secteurs sur les disques d'une matrice
- » Vérifications régulières de la consistance des données RAID afin de s'assurer de la bonne santé de la matrice
- » Surveillance pro-active des matrices avec des alertes par courrier électronique envoyées sur une adresse désignée lors d'événements importants
- » Les « write-intent bitmap », qui augmentent dramatiquement la vitesse des événements de resynchronisation en permettant au noyau de savoir précisément quelles portions d'un disque doivent être resynchronisées au lieu de devoir resynchroniser la matrice toute entière
- » Resynchronisez les points de contrôle. Ainsi, si vous redémarrez votre ordinateur pendant une resynchronisation, pendant le démarrage, la resynchronisation reprendra à l'emplacement où elle s'était arrêtée et ne devra pas recommencer depuis le début.
- » La capacité à changer les paramètres de la matrice après l'installation. Par exemple, vous pouvez agrandir une matrice RAID5 à 4 disques en matrice RAID5 à 5 disques lorsque vous avez un nouveau disque à ajouter. Cette opération d'agrandissement se fait à chaud et ne requiert pas de réinstallation sur la nouvelle matrice.

17.2. Niveaux RAID et prise en charge linéaire

RAID prend en charge diverses configurations, y compris les niveaux 0, 1, 4, 5, 6, 10, et linéaires. Ces types RAID sont définis comme suit :

Niveau 0

RAID niveau 0, souvent appelé « entrelacement », est une technique de mappage de données entrelacées orientée performances. Cela signifie que les données écrites sur la matrice sont divisées en bandes et écrites sur les disques membres de la matrice, permettant ainsi de hautes performances d'E/S pour un moindre coût, mais cela ne fournit pas de redondance.

De nombreuses implémentations RAID niveau 0 entrelaceront uniquement les données à travers les périphériques membres d'une taille égale à ou inférieure à la taille du plus petit périphérique de la matrice. Cela signifie que si vous possédez plusieurs périphériques de tailles légèrement différentes, chaque périphérique sera traité comme s'il était égal au plus petit disque. Ainsi, la capacité de stockage courante d'une matrice de niveau 0 est égale au plus petit disque dans la matrice RAID matériel, ou à la capacité de la plus petite partition membre dans une matrice RAID logiciel multipliée par le nombre de disques ou partitions dans la matrice.

Niveau 1

RAID niveau 1, ou la « mise en miroir », a été utilisé depuis plus longtemps que toute autre forme de RAID. Le niveau 1 fournit de la redondance en écrivant des données identiques sur chaque disque membre de la matrice, laissant une copie « miroir » sur chaque disque. La mise en miroir est populaire du fait de sa simplicité et du haut niveau de disponibilité de données offert. Le niveau 1 fonctionne avec deux disques ou plus et offre une très bonne fiabilité de données et améliore les performances des applications à lecture intensive, mais à coût relativement élevé. [3]

La capacité de stockage d'une matrice de niveau 1 est égale à la capacité du disque dur miroir le plus petit dans une matrice RAID matériel ou à la partition miroir la plus petite dans une matrice RAID logiciel. La redondance du niveau 1 est la plus élevée possible parmi les différents types RAID, la matrice étant capable de fonctionner avec un seul disque présent.

Niveau 4

Le niveau 4 utilise une parité [4] concentrée sur un seul disque pour protéger les données. Comme le disque de parité dédié représente un goulot d'étranglement inhérent à toutes les transactions d'écriture sur la matrice RAID, le niveau 4 est rarement utilisé sans technologie d'accompagnement telle que le cache en écriture différée, ou dans des circonstances particulières, où l'administrateur système conçoit le périphérique RAID logiciel avec ce goulot d'étranglement en tête (par exemple, avec une matrice qui aurait peu ou pas de transactions d'écriture une fois remplie de données). RAID niveau 4 est si rarement utilisé qu'il n'est pas disponible en tant qu'option dans Anaconda. Cependant, il peut être créé manuellement par l'utilisateur si réellement nécessaire.

La capacité de stockage RAID matériel niveau 4 est égale à la capacité de la partition membre la plus petite, multiplié par le nombre de partitions *moins un*. Les performances d'une matrice RAID niveau 4 seront toujours asymétriques, ce qui signifie que les lectures seront plus performantes que les écritures. Ceci est dû au fait que les écritures consomment davantage de ressources du CPU et de bande passante de la mémoire lors de la génération de parité, qui consomme également davantage de bande passante du bus lors de l'écriture des données sur disque car vous n'écrivez pas seulement les données, mais aussi la parité. Les lectures ne font que lire les données et non la parité, à moins que la matrice ne se trouve dans un état dégradé. Par conséquent, les lectures génèrent moins de trafic sur les disques et à travers les bus de l'ordinateur pour une même quantité de données transférée que sous des conditions normales.

Niveau 5

Type RAID le plus commun. En distribuant la parité à travers tous les disques membres d'une matrice, RAID niveau 5 élimine le goulot d'étranglement des écritures, qui est inhérent au niveau 4. Le seul goulot d'étranglement des performances est le processus de calcul de parité en soi-même. Avec les CPU modernes et RAID logiciel, il n'y a habituellement pas de goulot d'étranglement, car

tous les CPU modernes peuvent générer une parité très rapidement. Cependant, si vous possédez suffisamment de périphériques membres dans une matrice RAID5 logiciel, permettant ainsi une grande vitesse de transfert de données agrégées à travers tous les périphériques, alors ce goulot d'étranglement peut se révéler problématique.

Comme avec le niveau 4, le niveau 5 offre des performances asymétriques, avec des lectures considérablement plus performantes que les écritures. La capacité de stockage RAID niveau 5 est calculée de la même manière qu'avec le niveau 4.

Niveau 6

Ce niveau RAID commun, sur lequel la redondance et la préservation des données, et non les performances forment le but principal, considère l'inefficacité en termes d'espace du niveau 1 inacceptable. Le niveau 6 utilise un schéma de parité complexe afin d'être en mesure de récupérer après la perte de deux disques de la matrice. Ce schéma de parité complexe crée un fardeau pour le CPU bien plus important sur les périphériques RAID logiciel, mais aussi pendant les transactions d'écriture. Ainsi, le niveau 6 est considérablement plus asymétrique quant aux performances que les niveaux 4 et 5.

La capacité totale d'une matrice RAID niveau 6 est calculée de manière similaire à celles des matrices RAID niveaux 4 et 5, à l'exception que vous devrez soustraire 2 périphériques (au lieu d'un seul) du compte des périphériques pour l'espace de stockage supplémentaire de la parité.

Niveau 10

Ce niveau RAID tente de combiner les avantages de performance du niveau 0 avec la redondance du niveau 1. Il aide également à alléger une certaine quantité de l'espace gaspillé dans les matrices niveau 1 contenant plus de deux périphériques. Avec le niveau 10, il est possible de créer une matrice à 3 disques configurée pour stocker uniquement 2 copies de chaque « morceau » de données, ce qui permettra ensuite à la taille de la matrice de faire 1,5 fois la taille du périphérique le plus petit, au lieu d'être égal à la taille de celui-ci (comme cela aurait été le cas avec une matrice niveau 1 à trois périphériques).

Le nombre d'options disponible lors de la création de matrices de niveau 10 (ainsi que la complexité de la sélection des bonnes options pour un cas d'utilisation particulier) rend la création pendant une installation très peu pratique. Il est possible d'en créer une manuellement à l'aide de l'outil en ligne de commande `mdadm`. Pour obtenir davantage de détails sur les options et les compromis en termes de performance, veuillez consulter `man md`.

RAID linéaire

RAID linéaire est un simple regroupement de disques servant à créer un disque virtuel de plus grande taille. Avec une matrice RAID linéaire, les morceaux sont alloués de manière séquentielle, d'un disque au suivant, en allant au second disque qu'une fois que le premier disque aura été entièrement rempli. Ce regroupement ne fournit pas de bénéfices de performance, car il est très improbable que des opérations d'E/S soient divisées entre disques membres. RAID linéaire n'offre pas non plus de redondance, et réduit la fiabilité. — si un disque membre échoue, la matrice toute entière ne pourra pas être utilisée. La capacité est égale au total des disques membres.

17.3. Sous-systèmes RAID Linux

RAID sur Linux est composé des sous-systèmes suivants :

Pilotes de contrôleurs RAID matériel Linux

Les contrôleurs RAID matériel n'ont pas de sous-système RAID spécifique sur Linux. Comme ils utilisent des circuits (« chipsets ») RAID particuliers, les contrôleurs RAID matériel sont fournis avec leurs propres pilotes ; ces pilotes permettent au système de détecter les ensembles RAID en tant que disques normaux.

mdraid

Le sous-système **mdraid** a été conçu comme solution RAID logiciel pour Linux ; celui-ci est également la solution préférée pour RAID logiciel sous Linux. Ce sous-système utilise son propre format de métadonnées, habituellement appelé « métadonnées **mdraid** natives ».

mdraid prend également en charge d'autres formats de métadonnées, appelées métadonnées externes. Red Hat Enterprise Linux 7 utilise **mdraid** avec des métadonnées externes pour accéder aux ensembles ISW / IMSM (RAID microprogramme d'Intel). Les ensembles **mdraid** sont configurés et contrôlés via l'utilitaire **mdadm**.

dmraid

Device-mapper RAID ou **dmraid** fait référence au code du noyau mappeur de périphériques qui offre un mécanisme permettant d'assembler des disques pour former un ensemble RAID. Ce même code du noyau ne fournit aucun mécanisme de configuration RAID.

dmraid est entièrement configuré dans l'espace utilisateur, ce qui rend la prise en charge des différents formats de métadonnées plus facile. Ainsi, **dmraid** est utilisé sur un large éventail d'implémentations RAID microprogramme. **dmraid** prend aussi en charge le RAID microprogramme d'Intel, malgré le fait que Red Hat Enterprise Linux 7 utilise **mdraid** pour accéder aux ensembles RAID microprogramme d'Intel.

17.4. RAID Support in the Installer

L'installateur **Anaconda** détectera automatiquement tous les ensembles RAID matériel et logiciel sur un système, les rendant disponibles pour une installation. **Anaconda** prend également en charge RAID logiciel à l'aide de **mdraid**, et peut reconnaître les ensembles **mdraid** existants.

Anaconda fournit des utilitaires pour créer des ensembles RAID pendant l'installation ; cependant, ces utilitaires autorisent uniquement les partitions (et non les disques entiers) à faire partie des nouveaux ensembles. Pour utiliser un disque entier pour un ensemble, veuillez créer une partition s'étendant sur la totalité du disque, puis utilisez cette partition comme membre de l'ensemble RAID.

Lorsque le système de fichiers root utilise un ensemble RAID, **Anaconda** ajoutera des options de ligne de commande du noyau à la configuration du chargeur de démarrage, faisant savoir à **initrd** quel(s) ensemble(s) RAID doivent être activés avant de chercher le système de fichiers root.

Pour obtenir des instructions sur la configuration RAID pendant une installation, veuillez consulter le *Guide d'installation* de Red Hat Enterprise Linux 7.

17.5. Convertir un disque root en RAID 1 après l'installation

Si vous avez besoin de convertir un disque root non-raid en miroir RAID1 après avoir installé Red Hat Enterprise Linux 7, voir les instructions qui se trouvent dans l'article de base de connaissance de Red Hat intitulé : [How do I convert my root disk to RAID1 after installation of Red Hat Enterprise Linux 7?](#)

Pour l'architecture PowerPC (PPC), suivre les étapes suivantes :

1. Copier le contenu de la partition boot de la PowerPC Reference Platform (PReP) à partir de **/dev/sda1** vers **/dev/sdb1** :


```
# dd if=/dev/sda of=/dev/sdb1
```

2. Mettre à jour les indicateurs prep et boot sur la première partition sur les deux disques :

```
$ parted /dev/sda set 1 prep on
$ parted /dev/sda set 1 boot on

$ parted /dev/sdb set 1 prep on
$ parted /dev/sdb set 1 boot on
```

Notez qu'exécuter la commande **grub2-install /dev/sda** ne fonctionne pas sur une machine PowerPC et retourne une erreur, mais le système démarre comme prévu.

17.6. Configuring RAID Sets

La plupart des ensembles RAID sont configurés pendant la création, habituellement via le menu du microprogramme ou à partir de l'installateur. Dans certains cas, vous pourriez devoir créer ou modifier des ensembles RAID après avoir installé le système, de préférence sans avoir à redémarrer l'ordinateur et à entrer dans le menu du microprogramme.

Certains contrôleurs RAID vous permettent de configurer des ensembles RAID pendant l'exécution ou même de définir des ensembles complètement nouveaux après avoir ajouté des disques supplémentaires. Ceci requiert l'utilisation d'utilitaires spécifiques, car il n'existe pas d'interface de programmation pour ceci. Veuillez consulter la documentation du pilote du contrôleur RAID matériel pour obtenir davantage d'informations.

mdadm

L'outil de ligne de commande **mdadm** est utilisé pour gérer RAID logiciel sur Linux, c'est-à-dire **mdraid**. Pour obtenir des informations sur les différents modes et options **mdadm**, veuillez consulter **man mdadm**. La page **man** contient également des exemples utiles pour des opérations communes, comme la création, le contrôle, et l'assemblage de matrices RAID logiciel.

dmraid

Comme son nom le suggère, **dmraid** est utilisé pour gérer les ensembles RAID du mappeur de périphériques device-mapper. L'outil **dmraid** trouve les périphériques ATARAID en utilisant plusieurs gestionnaires de format de métadonnées, chacun prenant en charge divers formats. Pour afficher une liste complète des formats pris en charge, veuillez exécuter **dmraid -l**.

Comme mentionné précédemment dans [Section 17.3, « Sous-systèmes RAID Linux »](#), l'outil **dmraid** ne peut pas configurer les ensembles RAID après leur création. Pour obtenir des informations supplémentaires sur l'utilisation de **dmraid**, veuillez consulter **man dmraid**.

17.7. Création de périphériques RAID avancée

Dans certains cas, vous pourriez souhaiter installer le système d'exploitation sur une matrice qui ne peut pas être créée une fois l'installation terminée. Habituellement, cela signifie le paramétrage de **/boot** ou de matrices du système de fichiers root sur un périphérique RAID complexe ; dans de tels cas, vous pourriez devoir utiliser des options de matrices qui ne sont pas prises en charge par **Anaconda**. Pour contourner ce problème, veuillez observer la procédure suivante :

Procédure 17.1. Création avancée de périphériques RAID

1. Insérez le disque d'installation comme d'habitude.
2. Pendant le démarrage initial, veuillez sélectionner le mode de secours (« **Rescue Mode** ») au lieu de l'installation (« **Install** ») ou de la mise à niveau (« **Upgrade** »). Lorsque le système est entièrement démarré en mode de secours (« *Rescue mode* »), un terminal de ligne de commande sera présenté à l'utilisateur.
3. À partir de ce terminal, veuillez utiliser **parted** pour créer des partitions RAID sur les disques durs cibles. Puis, utilisez **mdadm** pour créer des matrices RAID manuellement à partir de ces partitions en utilisant tous les paramètres et options disponibles. Pour obtenir des informations supplémentaires sur la manière d'accomplir cela, veuillez consulter [Chapitre 12, Partitions](#), **man parted**, et **man mdadm**.
4. Une fois que les aires de stockage sont créées, vous pourrez, si vous le souhaitez, créer des systèmes de fichiers sur ces aires également.
5. Redémarrez l'ordinateur et sélectionnez Installation (« **Install** ») ou Mise à niveau (« **Upgrade** ») pour effectuer l'installation normalement. Comme **Anaconda** recherche les disques dans le système, il trouvera les périphériques RAID pré-existants.
6. Lorsqu'il vous sera demandé comment utiliser les disques du système, veuillez sélectionner Structure personnalisée (« **Custom Layout** »), puis cliquez sur Suivant (« **Next** »). Les périphériques RAID MD pré-existants y seront répertoriés.
7. Sélectionnez un périphérique RAID, cliquez sur Modifier (« **Edit** »), puis configurez son point de montage et (optionnellement) le type de fichier que le système de fichiers devrait utiliser (si vous n'en avez pas déjà créé un auparavant). Veuillez ensuite cliquer sur Terminé (« **Done** »). **Anaconda** effectuera l'installation sur ce périphérique RAID pré-existant, tout en conservant les options personnalisées que vous avez sélectionnées lors de sa création dans le mode de secours (« *Rescue Mode* »).



Note

Le *Mode de secours* limité de l'installateur n'inclut pas les pages **man. man mdadm** et **man md** contiennent des informations utiles à la création de matrices RAID personnalisées, et peuvent être nécessaires tout au long de la solution de contournement. Ainsi, il peut être utile d'avoir accès à une machine avec ces pages **man** présentes, ou de les imprimer avant de démarrer en *Mode de secours* et de créer vos propres matrices personnalisées.

[2] Un châssis enfichable à chaud vous permet de supprimer un disque dur sans avoir à éteindre votre système.

[3] Le coût de RAID niveau 1 est élevé car les mêmes informations sont écrites sur tous les disques de la matrice, ce qui offre une bonne fiabilité des données, mais de manière bien moins efficace, en termes d'espace, que les niveaux RAID basés sur parité, comme le niveau 5. Cependant, ce manque d'efficacité quant à l'espace offre également un bénéfice de performance : les niveaux RAID basés sur parité consomment une quantité d'alimentation du CPU bien plus élevée afin de générer une parité pendant que le niveau 1 RAID écrit les mêmes données plus d'une fois sur les multiples membres RAID avec un temps CPU très court. Ainsi, sur des ordinateurs utilisant RAID logiciel et ayant des ressources CPU utilisées à d'autres fins que des activités RAID de manière consistante, RAID niveau 1 peut fournir de meilleures performances que les niveaux RAID basés sur parité.

[4] Les informations sur la parité sont calculées sur la base du contenu du reste des disques membres de la matrice. Ces informations peuvent ensuite être utilisées pour reconstruire des données lorsqu'un disque de la matrice échoue. Les données reconstruites peuvent ensuite être utilisées pour satisfaire les requêtes d'E/S du disque en échec avant que celui-ci ne soit remplacé et pour le remplir à nouveau une fois son remplacement effectué.

Chapitre 18. Utilisation de la commande `mount`

Dans Linux, UNIX, et autres systèmes d'exploitation similaires, des systèmes de fichiers sur différentes partitions et périphériques amovibles (CD, DVD, ou lecteurs flash USB par exemple) peuvent être attachés à un certain emplacement (le *point de montage*) dans la structure de répertoire, puis détachés. Pour attacher ou détacher un système de fichiers, veuillez utiliser respectivement les commandes `mount` ou `umount`. Ce chapitre décrit l'utilisation de base de ces commandes, ainsi que certains sujets avancés comme le déplacement de points de montage ou la création de sous-structures partagées.

18.1. Répertorier les systèmes de fichiers actuellement montés

Pour afficher tous les systèmes de fichiers actuellement attachés, veuillez exécuter la commande `mount` sans arguments supplémentaires :

```
mount
```

Cette commande affiche la liste des points de montage connus. Chaque ligne fournit des informations importantes sur le nom du périphérique, le type de système de fichiers, le répertoire dans lequel il est monté, et les options de montage pertinentes sous la forme suivante :

périphérique sur le type de répertoire type (options)

L'utilitaire `findmnt`, qui permet aux utilisateurs de répertorier les systèmes de fichiers montés sous la forme d'une arborescence, est également disponible à partir de Red Hat Enterprise Linux 6.1. Pour afficher tous les systèmes de fichiers actuellement attachés, veuillez exécuter la commande `findmnt` sans arguments supplémentaires :

```
findmnt
```

18.1.1. Spécifier le type de système de fichiers

Par défaut, la sortie de la commande `mount` inclut divers systèmes de fichiers virtuels, tels que `sysfs` et `tmpfs`. Pour afficher uniquement les périphériques avec un certain type de système de fichiers, veuillez ajouter l'option `-t` sur la ligne de commande :

```
mount -t type
```

De manière similaire, pour uniquement afficher les périphériques avec un certain type de système de fichiers en utilisant la commande `findmnt`, veuillez saisir :

```
findmnt -t type
```

Pour obtenir une liste des types de systèmes de fichiers communs, veuillez consulter [Tableau 18.1, « Types de systèmes de fichiers communs »](#). Pour un exemple d'utilisation, veuillez consulter [Exemple 18.1, « Répertorier les systèmes de fichiers `ext4` actuellement montés »](#).

Exemple 18.1. Répertorier les systèmes de fichiers `ext4` actuellement montés

Habituellement, les partitions `/` et `/boot` sont formatées pour utiliser `ext4`. Pour afficher uniquement les points de montage qui utilisent ce système de fichiers, veuillez saisir ce qui suit dans l'invite shell :

```
~]$ mount -t ext4
/dev/sda2 on / type ext4 (rw)
/dev/sda1 on /boot type ext4 (rw)
```

Pour répertorier de tels points de montage en utilisant la commande **findmnt**, veuillez saisir :

```
~]$ findmnt -t ext4
TARGET SOURCE      FSTYPE OPTIONS
/         /dev/sda2 ext4    rw,realtime,seclabel,barrier=1,data=ordered
/boot    /dev/sda1 ext4    rw,realtime,seclabel,barrier=1,data=ordered
```

18.2. Monter un système de fichiers

Pour attacher un certain système de fichiers, veuillez utiliser la commande **mount** sous la forme suivante :

```
mount [option...] device directory
```

device peut être identifié par un chemin d'accès complet vers un *périphérique bloc* (par exemple, « /dev/sda3 »), un *identifiant unique universel* (UUID) ; par exemple, « UUID=34795a28-ca6d-4fd8-a347-73671d0c19cb »), ou une *étiquette de volume* (par exemple, « LABEL=home »). Remarquez que tant qu'un système de fichiers est monté, le contenu d'origine du répertoire *directory* sera inaccessible.



Important

Linux n'empêche pas un utilisateur de monter un système de fichiers sur un répertoire possédant déjà un système de fichiers qui y est attaché. Pour déterminer si un répertoire en particulier sert de point de montage, veuillez exécuter l'utilitaire **findmnt** avec le répertoire en tant qu'argument et vérifiez le code de sortie :

```
findmnt directory; echo $?
```

Si aucun système de fichiers n'est attaché au répertoire, la commande ci-dessus retournera **1**.

Lorsque la commande **mount** est exécutée sans toutes les informations requises (c'est-à-dire sans le nom de périphérique, le répertoire cible, ou le type de système de fichiers), la commande **mount** lit le contenu du fichier de configuration **/etc/fstab** pour voir si le système de fichiers donné est répertorié. Ce fichier **/etc/fstab** contient une liste de noms de périphériques et les répertoires dans lesquels les systèmes de fichiers sélectionnés doivent être montés, ainsi que le type de système de fichiers et les options de montage. De ce fait, lors du montage d'un système de fichiers qui est spécifié dans ce fichier **/etc/fstab**, vous pouvez utiliser l'une des variantes suivantes de la commande :

```
mount [option...] directory
mount [option...] device
```

Remarquez que des permissions sont requises pour monter les systèmes de fichiers, à moins que la commande ne soit exécutée en tant que **root** (voir la [Section 18.2.2, « Spécifier les options de montage »](#)).

**Note**

Pour déterminer l'UUID et, si le périphérique les utilise, l'étiquette d'un périphérique en particulier, veuillez utiliser la commande **blkid** sous le format suivant :

```
blkid device
```

Par exemple, pour afficher des informations sur **/dev/sda3**, veuillez saisir :

```
~]# blkid /dev/sda3
/dev/sda3: LABEL="home" UUID="34795a28-ca6d-4fd8-a347-73671d0c19cb"
TYPE="ext3"
```

18.2.1. Spécifier le type de système de fichiers

Dans la plupart des cas, **mount** détecte le système de fichiers automatiquement. Cependant, certains systèmes de fichiers, tels que **NFS** (« Network File System ») ou **CIFS** (« Common Internet File System ») ne sont pas reconnus, et doivent être spécifiés manuellement. Pour spécifier le type de système de fichiers, veuillez utiliser la commande **mount** sous le format suivant :

```
mount -t type périphérique répertoire
```

[Tableau 18.1, « Types de systèmes de fichiers communs »](#) fournit une liste des types de système de fichiers communs pouvant être utilisés avec la commande **mount**. Pour une liste complète de tous les types de système de fichiers disponibles, veuillez consulter la page du manuel correspondante comme indiqué dans la [Section 18.4.1, « Documentation installée »](#).

Tableau 18.1. Types de systèmes de fichiers communs

Type	Description
ext2	Système de fichiers ext2 .
ext3	Système de fichier ext3 .
ext4	Système de fichiers ext4 .
btrfs	Système de fichiers btrfs .
xf	Système de fichiers xf .
iso9660	Système de fichiers ISO 9660 . Communément utilisé par les supports optiques, comme les CD.
jfs	Système de fichier JFS créé par IBM.
nfs	Système de fichiers NFS . Celui-ci est communément utilisé pour accéder à des fichiers sur un réseau.
nfs4	Système de fichiers NFSv4 . Celui-ci est communément utilisé pour accéder à des fichiers sur un réseau.
ntfs	Système de fichiers NTFS . Celui-ci est communément utilisé sur des ordinateurs exécutant un système d'exploitation Windows.
udf	Système de fichiers UDF . Communément utilisé par les supports optiques, comme les DVD.

Type	Description
vfat	Système de fichiers FAT . Celui-ci est communément utilisé sur les ordinateurs exécutant un système d'exploitation Windows ainsi que sur certains supports numériques, tels que les lecteurs flash USB ou les disquettes.

Veillez consulter l'[Exemple 18.2, « Monter un disque flash USB »](#) pour voir un exemple d'utilisation.

Exemple 18.2. Monter un disque flash USB

Les lecteurs flash USB plus anciens utilisent souvent le système de fichiers FAT. En supposant que ce type de lecteur utilise le périphérique `/dev/sdc1` et que le répertoire `/media/flashdisk/` existe, veuillez le monter sur ce répertoire en saisissant ce qui suit à l'invite shell en tant qu'utilisateur **root** :

```
~]# mount -t vfat /dev/sdc1 /media/flashdisk
```

18.2.2. Spécifier les options de montage

Pour spécifier des options de montage supplémentaires, veuillez utiliser la commande sous le format suivant :

```
mount -o options périphérique répertoire
```

Lorsque plusieurs options sont fournies, veuillez ne pas insérer d'espace après une virgule, sinon la commande **mount** interprétera incorrectement les valeurs qui suivent les espaces en tant que paramètres supplémentaires.

[Tableau 18.2, « Options de montage communes »](#) fournit une liste d'options de montage communes. Pour une liste complète de toutes les options disponibles, veuillez consulter la page du manuel correspondante comme indiqué dans la [Section 18.4.1, « Documentation installée »](#).

Tableau 18.2. Options de montage communes

Option	Description
async	Permet les opérations d'entrées/sorties asynchrones sur le système de fichiers.
auto	Permet au système de fichiers d'être monté automatiquement en utilisant la commande mount -a .
defaults	Fournit un alias pour async, auto, dev, exec, nouser, rw, suid .
exec	Permet l'exécution de fichiers binaires sur un système de fichiers particulier.
loop	Monte une image en tant que périphérique boucle.
noauto	Le comportement par défaut interdit le montage automatique du système de fichiers à l'aide de la commande mount -a .
noexec	Interdit l'exécution de fichiers binaires sur le système de fichiers en particulier.
nouser	Interdit à un utilisateur normal (c'est-à-dire autre que root) de monter et démonter le système de fichiers.
remount	Remonte le système de fichiers au cas où il serait déjà monté.
ro	Monte le système de fichiers en lecture seule.
rw	Monte le système de fichier en lecture et écriture.
user	Permet à un utilisateur normal (c'est-à-dire autre que root) de monter et démonter le système de fichiers.

Veuillez consulter l'[Exemple 18.3, « Monter une image ISO »](#) pour un exemple d'utilisation.

Exemple 18.3. Monter une image ISO

Une image ISO (ou une image de disque en général) peut être montée en utilisant le périphérique boucle. En supposant que l'image ISO du disque d'installation Fedora 14 se trouve dans le répertoire de travail actuel et que le répertoire `/media/cdrom/` existe, veuillez monter l'image sur ce répertoire en exécutant la commande suivante en tant qu'utilisateur `root` :

```
~]# mount -o ro,loop Fedora-14-x86_64-Live-Desktop.iso /media/cdrom
```

Remarquez qu'ISO 9660 est, de par sa conception, un système de fichiers en lecture seule.

18.2.3. Partager des montages

De manière occasionnelle, certaines tâches d'administration système requièrent d'avoir accès au même système de fichiers à partir de plusieurs emplacements dans l'arborescence des répertoires (par exemple, lors de la préparation d'un environnement chroot). Ceci est possible, et Linux vous autorise à monter le même système de fichiers sur autant de répertoires que nécessaire. En outre, la commande `mount` implémente l'option `--bind` qui fournit un moyen pour dupliquer certains montages. Son utilisation fonctionne comme suit :

```
mount --bind old_directory new_directory
```

Même si cette commande permet à un utilisateur d'accéder au système de fichiers à partir de ces deux emplacements, celle-ci ne s'applique pas aux systèmes de fichiers qui sont montés à l'intérieur du répertoire d'origine. Pour également inclure ces montages, veuillez saisir :

```
mount --rbind old_directory new_directory
```

De plus, afin de fournir autant de flexibilité que possible, Red Hat Enterprise Linux 7 implémente la fonctionnalité connue sous le nom de *sous-arborescence partagée*. Cette fonctionnalité permet l'utilisation des quatre types de montage suivants :

Shared Mount

Un montage partagé permet la création d'une réplique exacte d'un point de montage donné. Lorsqu'un point de montage est marqué en tant que montage partagé, tout montage à l'intérieur du point de montage d'origine est reflété dedans, et vice-versa. Pour modifier le type d'un point de montage en montage partagé, veuillez saisir la commande suivante à l'invite shell :

```
mount --make-shared mount_point
```

De manière alternative, pour modifier le type de montage du point de montage sélectionné et de tous les points de montage se trouvant sous celui-ci, veuillez saisir :

```
mount --make-rshared mount_point
```

Veuillez consulter [Exemple 18.4, « Créer un point de montage partagé »](#) pour un exemple d'utilisation.

Exemple 18.4. Créer un point de montage partagé

Il existe deux emplacements dans lesquels les autres systèmes de fichiers sont communément montés : le répertoire **/media** pour les supports amovibles, et le répertoire **/mnt** pour les systèmes de fichiers montés temporairement. En utilisant un montage partagé, vous pouvez faire en sorte que ces deux répertoires partagent le même contenu. Pour ce faire, en tant qu'utilisateur **root**, marquez le répertoire **/media** en tant que répertoire « partagé » :

```
~]# mount --bind /media /media
~]# mount --make-shared /media
```

Puis créez son double dans **/mnt** en utilisant la commande suivante :

```
~]# mount --bind /media /mnt
```

Il est désormais possible de vérifier qu'un montage à l'intérieur de **/media** apparaît aussi dans **/mnt**. Par exemple, si le lecteur CD-ROM contient un support qui n'est pas vide et que le répertoire **/media/cdrom/** existe, veuillez exécuter les commandes suivantes :

```
~]# mount /dev/cdrom /media/cdrom
~]# ls /media/cdrom
EFI  GPL  isolinux  LiveOS
~]# ls /mnt/cdrom
EFI  GPL  isolinux  LiveOS
```

De la même manière, il est possible de vérifier que n'importe quel système de fichiers monté dans le répertoire **/mnt** se reflètera dans **/media**. Par exemple, si un lecteur flash USB qui utilise le périphérique **/dev/sdc1** est enfiché et que le répertoire **/mnt/flashdisk/** est présent, veuillez saisir :

```
~]# mount /dev/sdc1 /mnt/flashdisk
~]# ls /media/flashdisk
en-US  publican.cfg
~]# ls /mnt/flashdisk
en-US  publican.cfg
```

Slave Mount

Un montage esclave permet la création d'un double limité d'un point de montage donné. Lorsqu'un point de montage est marqué en tant que montage esclave, tout montage dans le point de montage d'origine y sera reflété, mais aucun montage à l'intérieur d'un montage esclave n'est reflété dans son point d'origine. Pour modifier le type d'un point de montage en montage esclave, veuillez saisir ce qui suit à l'invite shell :

```
mount --make-slave mount_point
```

Alternativement, il est possible de modifier le type de montage du point de montage sélectionné et de tous les points de montage se trouvant sous celui-ci en saisissant :

```
mount --make-rslave mount_point
```


Veillez consulter l'[Exemple 18.5, « Créer un point de montage esclave »](#) pour voir un exemple d'utilisation.

Exemple 18.5. Créer un point de montage esclave

Cet exemple montre comment faire pour que le contenu du répertoire `/media` soit également affiché dans `/mnt`, mais sans qu'aucun montage du répertoire `/mnt` ne soit reflété dans `/media`. En tant qu'utilisateur `root`, veuillez marquer le répertoire `/media` en tant que répertoire « partagé » :

```
~]# mount --bind /media /media
~]# mount --make-shared /media
```

Puis créez son duplicqué dans `/mnt`, mais marquez-le en tant qu'« esclave » :

```
~]# mount --bind /media /mnt
~]# mount --make-slave /mnt
```

Veillez vérifier qu'un montage à l'intérieur de `/media` apparaisse aussi dans `/mnt`. Par exemple, si le lecteur CD-ROM contient un support qui n'est pas vide et que le répertoire `/media/cdrom/` existe, veuillez exécuter les commandes suivantes :

```
~]# mount /dev/cdrom /media/cdrom
~]# ls /media/cdrom
EFI  GPL  isolinux  LiveOS
~]# ls /mnt/cdrom
EFI  GPL  isolinux  LiveOS
```

Veillez également vérifier qu'aucun des systèmes de fichiers montés dans le répertoire `/mnt` ne soit reflété dans `/media`. Par exemple, si un lecteur flash USB qui utilise le périphérique `/dev/sdc1` est attaché et que le répertoire `/mnt/flashdisk/` est présent, veuillez saisir :

```
~]# mount /dev/sdc1 /mnt/flashdisk
~]# ls /media/flashdisk
~]# ls /mnt/flashdisk
en-US  publican.cfg
```

Private Mount

Un montage privé est le type de montage par défaut, contrairement à un montage privé ou partagé, il ne reçoit et ne transfère pas d'événements de propagation. Pour marquer explicitement un point de montage en tant que montage privé, veuillez saisir ce qui suit à l'invite shell :

```
mount --make-private mount_point
```

Alternativement, il est possible de modifier le type de montage du point de montage sélectionné et de tous les points de montage se trouvant sous celui-ci :

```
mount --make-rprivate mount_point
```

Veillez consulter l'[Exemple 18.6, « Créer un point de montage privé »](#) pour voir un exemple d'utilisation.

Exemple 18.6. Créer un point de montage privé

En prenant en compte le scénario dans l'[Exemple 18.4, « Créer un point de montage partagé »](#), supposez que le point de montage partagé a été créé auparavant en utilisant les commandes suivantes en tant qu'utilisateur **root**:

```
~]# mount --bind /media /media
~]# mount --make-shared /media
~]# mount --bind /media /mnt
```

Pour marquer le répertoire **/mnt** en tant que « privé », veuillez saisir :

```
~]# mount --make-private /mnt
```

Il est désormais possible de vérifier qu'aucun des montages présents à l'intérieur de **/media** n'apparaissent dans **/mnt**. Par exemple, si le lecteur CD-ROM contient un support qui n'est pas vide et que le répertoire **/media/cdrom/** existe, veuillez exécuter les commandes suivantes :

```
~]# mount /dev/cdrom /media/cdrom
~]# ls /media/cdrom
EFI  GPL  isolinux  LiveOS
~]# ls /mnt/cdrom
~]#
```

Il est également possible de vérifier qu'aucun des systèmes de fichiers montés dans le répertoire **/mnt** ne soit reflété dans **/media**. Par exemple, si un lecteur flash USB qui utilise le périphérique **/dev/sdc1** est attaché et que le répertoire **/mnt/flashdisk/** est présent, veuillez saisir :

```
~]# mount /dev/sdc1 /mnt/flashdisk
~]# ls /media/flashdisk
~]# ls /mnt/flashdisk
en-US  publican.cfg
```

Unbindable Mount

Pour empêcher qu'un point de montage donné ne soit dupliqué, un montage ne pouvant pas être lié peut être utilisé. Pour modifier le type d'un point de montage en montage ne pouvant pas être lié, veuillez saisir ce qui suit à l'invite shell :

```
mount --make-unbindable mount_point
```

Alternativement, il est possible de modifier le type de montage du point de montage sélectionné et de tous les points de montage se trouvant sous celui-ci :

```
mount --make-runbindable mount_point
```

Veuillez consulter l'[Exemple 18.7, « Créer un point de montage ne pouvant pas être lié »](#) pour voir un exemple d'utilisation.

Exemple 18.7. Créer un point de montage ne pouvant pas être lié

Pour empêcher que le répertoire `/media` soit partagé, veuillez saisir ce qui suit à l'invite shell en tant qu'utilisateur `root` :

```
~]# mount --bind /media /media
~]# mount --make-unbindable /media
```

Ainsi, toute tentative consécutive de créer un duplicqué de ce montage échouera avec une erreur :

```
~]# mount --bind /media /mnt
mount: wrong fs type, bad option, bad superblock on /media,
missing codepage or helper program, or other error
In some cases useful info is found in syslog - try
dmesg | tail or so
```

18.2.4. Déplacer un point de montage

Pour modifier le répertoire dans lequel un système de fichiers est monté, veuillez utiliser la commande suivante :

```
mount --move old_directory new_directory
```

Veuillez consulter l'[Exemple 18.8, « Déplacer un point de montage NFS existant »](#) pour voir un exemple d'utilisation.

Exemple 18.8. Déplacer un point de montage NFS existant

Un stockage NFS contient des répertoires utilisateur et est déjà monté dans `/mnt/userdirs/`. En tant qu'utilisateur `root`, veuillez déplacer ce point de montage sur `/home` en utilisant la commande suivante :

```
~]# mount --move /mnt/userdirs /home
```

Pour vérifier que le point de montage a été déplacé, répertoriez le contenu des deux répertoires :

```
~]# ls /mnt/userdirs
~]# ls /home
jill joe
```

18.2.5. Définir la permission Lecture-seule pour root

Dans certains cas, vous pourriez souhaiter monter le système de fichiers avec des permissions lecture-seule. Les cas d'utilisation incluent une amélioration de la sécurité ou assurer l'intégrité des données suite à une interruption de la l'alimentation électrique.

18.2.5.1. Configurer root pour qu'il puisse monter avec les permissions lecture-seule au démarrage.

1. Dans le fichier `/etc/sysconfig/readonly-root`, modifier la valeur de `READONLY` à `yes` :

```
# Set to 'yes' to mount the system file systems read-only.
READONLY=yes[sortie tronquée]
```

2. Changer **defaults** en **ro** dans l'entrée root (/) dans le fichier **/etc/fstab** :

```
/dev/mapper/luks-c376919e... / ext4 ro,x-systemd.device-timeout=0 1 1
```

3. Ajouter **ro** à la directive **GRUB_CMDLINE_LINUX** dans le fichier **/etc/default/grub** et assurez-vous qu'il n'y ait pas de **rw**:

```
GRUB_CMDLINE_LINUX="crashkernel=auto rd.lvm.lv=rhel/root
rd.lvm.lv=rhel/swap rhgb quiet ro"
```

4. Recréer le fichier de configuration GRUB2 :

```
~]# grub2-mkconfig -o /boot/grub2/grub.cfg
```

5. Si vous ajoutez des fichiers et des répertoires à monter avec la permission écriture dans le système de fichiers **tmpfs**, créer un fichier texte dans le répertoire **/etc/rwtab.d/** et mettez-y la configuration. Par exemple, pour monter **/etc/example/file** avec des permissions écriture (w), ajouter cette ligne dans le fichier **/etc/rwtab.d/example** :

```
files /etc/example/file
```



Important

Les changements apportés aux fichiers et répertoires dans **tmpfs** ne sont pas persistants d'un démarrage à l'autre.

Voir [Section 18.2.5.3, « Fichiers et répertoires qui retiennent les permissions écriture »](#) pour obtenir plus d'informations sur cette étape.

6. Redémarrer le système.

18.2.5.2. Remonter root instantanément

Si root (/) a été monté avec les permissions lecture-seule au démarrage du système, vous pouvez le remonter avec les permissions écriture :

```
~]# mount -o remount,rw /
```

Cela est particulièrement utile quand / n'est pas monté correctement avec les permissions lecture-seule.

Pour remonter / avec les permissions lecture-seule à nouveau, exécuter :

```
~]# mount -o remount,ro /
```



Note

Cette commande monte *tout le /* avec les permissions lecture-seule. Une meilleure approche consiste à conserver les permissions d'écriture pour certains fichiers et répertoires en les copiant en RAM, comme expliqué dans [Section 18.2.5.1, « Configurer root pour qu'il puisse monter avec les permissions lecture-seule au démarrage. »](#).

18.2.5.3. Fichiers et répertoires qui retiennent les permissions écriture

Pour que le système puisse fonctionner correctement, certains fichiers et répertoires doivent conserver les permissions d'écriture. Avec root en mode de lecture uniquement, ils sont montés en RAM dans le système de fichiers temporaire **tmpfs**. Les valeurs par défaut de tels fichiers et répertoires se trouvent dans le fichier **/etc/rwtab**, qui contient :

```
dirs /var/cache/man
dirs /var/gdm[sortie tronquée]
empty /tmp
empty /var/cache/foomatic[sortie tronquée]
files /etc/adjtime
files /etc/ntp.conf[sortie tronquée]
```

Les entrées du fichier **/etc/rwtab** doivent suivre le format suivant :

```
comment le fichier ou répertoire est copié dans tmpfs      chemin vers le  
fichier ou répertoire
```

Un fichier ou répertoire peut être copié dans **tmpfs** de trois façons, donc il y a trois sortes d'entrées :

- ✦ **empty path** : un chemin vide est copié dans **tmpfs**. Exemple : **empty /tmp**
- ✦ **dirs path** : une aboescence de répertoires est copiée dans **tmpfs** empty. Exemple : **dirs /var/run**
- ✦ **files path** : un fichier ou répertoire est copié dans **tmpfs** intact. Exemple : **files /etc/resolv.conf**

Le même format s'applique quand on ajoute des chemins personnalisés à **/etc/rwtab.d/**.

18.3. Démonter un système de fichiers

Pour détacher un système de fichiers qui a été monté, veuillez utiliser l'une des variantes suivantes de la commande **umount** :

```
umount directory  
umount device
```

Remarquez qu'à moins que cette opération ne soit effectuée en tant qu'utilisateur **root**, les bonnes permissions doivent être disponibles pour démonter le système de fichiers (veuillez consulter la [Section 18.2.2, « Spécifier les options de montage »](#)). Voir [Exemple 18.9, « Démonter un CD »](#) pour un exemple d'utilisation.



Important

Lorsqu'un système de fichiers est en cours d'utilisation (par exemple lorsqu'un processus lit un fichier sur ce système de fichiers, ou lorsqu'il est utilisé par le noyau), l'exécution de la commande **umount** échouera avec une erreur. Pour déterminer quels processus accèdent au système de fichiers, veuillez utiliser la commande **fuser** sous le format suivant :

```
fuser -m directory
```

Par exemple, pour répertorier les processus qui accèdent à un système de fichiers monté sur le répertoire `/media/cdrom/`, veuillez saisir :

```
~]$ fuser -m /media/cdrom
/media/cdrom:          1793   2013   2022   2435 10532c 10672c
```

Exemple 18.9. Démonter un CD

Pour démonter un CD qui était auparavant monté sur le répertoire `/media/cdrom/`, veuillez saisir ce qui suit à l'invite shell :

```
~]$ umount /media/cdrom
```

18.4. Références de la commande `mount`

Les ressources suivantes offrent une documentation détaillée sur le sujet.

18.4.1. Documentation installée

- ✦ **man 8 mount** — Page du manuel pour la commande **mount**, celle-ci fournit une documentation complète sur son utilisation.
- ✦ **man 8 umount** — Page du manuel pour la commande **umount**, celle-ci fournit une documentation complète sur son utilisation.
- ✦ **man 8 findmnt** — Page du manuel pour la commande **findmnt**, celle-ci fournit une documentation complète sur son utilisation.
- ✦ **man 5 fstab** — Page du manuel fournissant une description détaillée du format de fichier `/etc/fstab`.

18.4.2. Sites Web utiles

- ✦ [Sous-arborescences partagées](#) — Un article LWN couvrant le concept de sous-arborescence partagée.

Chapitre 19. Fonction `volume_key`

La fonction `volume_key` fournit deux outils, `libvolume_key` et **`volume_key`**. `libvolume_key` est une bibliothèque pour manipuler des clés de chiffrement du volume de stockage et pour les stocker hors des volumes. **`volume_key`** est un outil de ligne de commande associé utilisé pour extraire les clés et phrases de passe afin de restaurer l'accès à un disque dur chiffré.

Ceci est utile lorsque l'utilisateur principal oublie ses clés et mots de passe, après qu'un employé soit parti soudainement, ou afin d'extraire des données lorsqu'un échec du logiciel ou du matériel corrompt l'en-tête du volume chiffré. Dans un environnement d'entreprise, le service d'assistance informatique peut utiliser **`volume_key`** pour effectuer des copies de sauvegarde des clés de chiffrement avant de rendre l'ordinateur à l'utilisateur final.

Actuellement, **`volume_key`** prend uniquement en charge le format de chiffrement de volumes LUKS.



Note

`volume_key` n'est pas inclus dans une installation standard du serveur Red Hat Enterprise Linux 7. Pour obtenir des informations sur son installation, veuillez consulter http://fedoraproject.org/wiki/Disk_encryption_key_escrow_use_cases.

19.1. Commandes `volume_key`

Le format de **`volume_key`** est :

```
volume_key [OPTION]... OPERAND
```

Les opérandes et le mode d'opération de **`volume_key`** sont déterminés par une des options suivantes :

--save

Cette commande attend l'opérande *volume [paquet]*. Si un *paquet* est fourni, alors **`volume_key`** en extraira les clés et les phrases secrètes. Si le *paquet* n'est pas fourni, alors **`volume_key`** va extraire les clés et phrases secrètes du *volume*, en invitant l'utilisateur si nécessaire. Ces clés et les phrases secrètes seront alors stockées dans un ou plusieurs paquets de sortie.

--restore

Cette commande attend les opérandes de *paquets de volume*. Puis, il ouvre le *volume* et utilise les clés et phrases secrètes du *paquet* pour rendre le *volume* accessible à nouveau, invitant l'utilisateur à saisir un nouveau mot de passe, par exemple.

--setup-volume

Cette commande s'attend aux opérandes *nom de paquet de volume*, puis, ouvre le *volume* et utilise les clés et phrases secrètes du *paquet* pour configurer le *volume* à utiliser des données déchiffrées comme un *nom*.

Nom est le nom d'un volume de dm-crypt. Cette opération rend le volume décrypté disponible comme `/dev/mapper/nom`.

Cette opération ne modifie pas en permanence le *volume* en ajoutant une nouvelle phrase secrète, par exemple. L'utilisateur peut accéder et modifier le volume décrypté, en modifiant le *volume* dans

le processus.

--reencrypt, --secrets, et --dump

Ces trois commandes exécutent des fonctions semblables avec diverses méthodes de sortie. Chacune d'entre elle requiert l'opérande *paquet* et ouvre le *paquet*, décryptant si nécessaire. **--reencrypt** stocke ensuite les informations dans un ou plusieurs nouveaux paquets de sortie. Les sorties de **--secrets** sont des clés et des phrases secrètes contenues dans le *paquet*. **--dump** affiche le contenu du *paquet*, bien que les clés et les phrases secrètes ne soient pas des sorties par défaut. Ceci peut être changé en ajoutant **--with-secrets** à la commande. Il est également possible de vider uniquement les pièces non cryptées du paquet, le cas échéant, à l'aide de la commande **--unencrypted**. Cela ne nécessite pas de phrase secrète ou d'accès de clé privée.

On peut ajouter à chacune d'entre elle les options suivantes :

-o, --output *packet*

Cette commande écrit la phrase secrète ou le mot de passe par défaut dans le *paquet*. La phrase secrète ou le mot de passe par défaut dépendent du format de volume. S'assurer qu'ils ne risquent pas d'expirer, et qu'ils permettront à **--restore** de restaurer l'accès au volume.

--output-format *format*

Cette commande utilise le *format* spécifié pour tous les paquets de sortie. Actuellement, le *format* peut correspondre à l'une des conditions suivantes :

- ✦ **asymmetric** : utilise CMS pour encrypter le paquet, et exige un certificat
- ✦ **asymmetric_wrap_secret_only** : englobe seulement le secret, ou les clés et les phrases secrètes et exige un certificat
- ✦ **passphrase** : utilise GPG pour déchiffrer tout le paquet, et exige une phrase secrète

--create-random-passphrase *packet*

Cette commande génère une phrase secrète alphanumérique au hasard, l'ajoute au *volume* (sans affecter les autres phrases secrètes) et stocke ensuite ce mot de passe aléatoire dans le *paquet*.

19.2. Exécutez `volume_key` en tant qu'utilisateur individuel

En tant qu'utilisateur individuel, `volume_key` peut être utilisé pour sauvegarder les clés d'encodage par la procédure suivante.



Note

Pour tous les exemples qui se trouvent dans ce fichier, `/path/to/volume` est un périphérique LUKS, et non pas un périphérique en texte brut intégré. La commande `blkid -s type /path/to/volume` doit rapporter `type="crypto_LUKS"`.

Procédure 19.1. Exécutez `volume_key` en autonome

1. Exécutez:

```
volume_key --save /path/to/volume -o escrow-packet
```


Vous apercevrez alors une invite vous demandant une phrase secrète de paquet escrow pour protéger la clé.

2. Sauvegarder le fichier **escrow-packet**, en veillant à ne pas oublier la phrase secrète.

Si la phrase secrète est oubliée, utiliser le paquet escrow enregistré pour restaurer l'accès aux données.

Procédure 19.2. Restaurer l'accès aux données par le paquet escrow

1. Démarrer le système dans un environnement où **volume_key** peut être exécuté et où un paquet escrow est disponible (en mode de secours, par exemple).
2. Exécutez :

```
volume_key --restore /path/to/volume escrow-packet
```

Vous apercevrez alors une invite vous demandant la phrase secrète de paquet escrow qui a été utilisée lors de la création du paquet, et pour la nouvelle phrase secrète du volume.

3. Mounter le volume en utilisant la phrase secrète choisie.

Pour libérer l'espace phrase secrète de l'en-tête de LUKS du volume crypté, supprimer l'ancienne phrase secrète oubliée en exécutant la commande **cryptsetup luksKillSlot**.

19.3. Exécutez `volume_key` dans une grande organisation

Dans une grande organisation, il n'est pas très pratique d'utiliser un mot de passe unique connu de chaque administrateur de systèmes et de conserver un mot de passe distinct pour chaque système. Cela n'est pas pratique et représente un risque de sécurité. Pour contrer cela, **volume_key** peut utiliser la cryptographie asymétrique pour réduire le nombre de personnes qui connaissent le mot de passe requis pour accéder aux données cryptées sur un ordinateur.

Cette section couvrira les procédures nécessaires à la préparation : avant l'enregistrement des clés de chiffrement, comment enregistrer les clés de chiffrement, restaurer l'accès à un volume et la mise en place de phrases secrètes d'urgence.

19.3.1. Se préparer à enregistrer vos clés de chiffrement.

Pour commencer à enregistrer les clés de chiffrement, vous devrez vous préparer.

Procédure 19.3. Préparation

1. Créer une paire privée/certificat X509.
2. Indique les utilisateurs auxquels on fait confiance pour ne pas compromettre la clé privée. Ces utilisateurs seront en mesure de décrypter les paquets escrow.
3. Choisir les systèmes qui seront utilisés pour décrypter les paquets escrow. Sur ces systèmes, définir une base de données NSS qui contienne la clé privée.

Si la clé privée n'a pas été créée dans une base de données NSS, suivre les étapes suivantes :

- A. Stocker le certificat et la clé privée dans un fichier **PKCS#12**.
- B. Exécutez :

```
certutil -d /the/nss/directory -N
```

À ce stade, il est possible de choisir un mot de passe de base de données NSS. Chaque base de données NSS peut avoir un mot de passe différent, donc les utilisateurs désignés n'ont pas besoin de partager un mot de passe unique, si une base de données NSS séparée est utilisée par chaque utilisateur.

C. Exécutez :

```
pk12util -d /the/nss/directory -i the-pkcs12-file
```

4. Distribuer le certificat à quiconque installant des systèmes ou sauvegardant des clés sur les systèmes existants.
5. Pour les clés privées sauvegardées, préparez un stockage qui leur permette d'être consultées par une machine et un volume. Il peut s'agir, par exemple, d'un simple répertoire avec un sous-répertoire par machine, ou d'une base de données utilisée pour d'autres tâches de gestion de système également.

19.3.2. Sauvegarde des clés de chiffrement

Quand vous aurez terminé le travail requis (voir [Section 19.3.1, « Se préparer à enregistrer vos clés de chiffrement. »](#)), il vous sera alors possible de sauvegarder les clés de chiffrement par la procédure suivante.



Note

Pour tous les exemples qui se trouvent dans ce fichier, `/path/to/volume` est un périphérique LUKS, et non pas un périphérique en texte brut intégré. La commande `blkid -s type /path/to/volume` doit rapporter `type="crypto_LUKS"`.

Procédure 19.4. Sauvegarde des clés de chiffrement

1. Exécutez :

```
volume_key --save /path/to/volume -c /path/to/cert escrow-packet
```

2. Sauvegarder le fichier **escrow-packet** créé dans le stockage préparé, en l'associant avec le système et le volume.

Ces étapes peuvent être effectuées manuellement, ou par script dans le cadre de l'installation du système.

19.3.3. Restaurer l'accès à un volume

Une fois que les clés de chiffrement auront été sauvegardées (voir [Section 19.3.1, « Se préparer à enregistrer vos clés de chiffrement. »](#) et [Section 19.3.2, « Sauvegarde des clés de chiffrement »](#)), l'accès pourra être restauré à un pilote selon les besoins.

Procédure 19.5. Restaurer l'accès à un volume

1. Extraire le paquet escrow pour le volume, du stockage de paquets, et envoyez le vers l'un des utilisateurs désignées afin qu'il soit décrypté.

2. L'utilisateur désigné exécute :

```
volume_key --reencrypt -d /the/nss/directory escrow-packet-in -o  
escrow-packet-out
```

Après avoir fourni le mot de passe de base de données NSS, l'utilisateur désigné choisira une phrase secrète pour le chiffrement **escrow-paquet-out**. Cette phrase secrète peut varier à chaque fois et ne protège que les clés de cryptage que lorsqu'elles sont transférées de l'utilisateur désigné au système cible.

3. Obtenir le fichier **escrow-packet-out** et la phrase secrète de l'utilisateur désigné.
4. Démarrer le système cible dans un environnement où **volume_key** peut être exécuté et où un paquet **escrow-packet-out** est disponible (en mode de secours, par exemple).
5. Exécutez :

```
volume_key --restore /path/to/volume escrow-packet-out
```

Vous apercevrez alors une invite vous demandant la phrase secrète de paquet qui a été choisie par l'utilisateur désigné, et la nouvelle phrase secrète du volume.

6. Monter le volume en utilisant la phrase secrète de volume choisie.

Il est possible d'enlever l'ancienne phrase secrète oubliée en utilisant **cryptsetup luksKillSlot**, par exemple, afin de libérer l'emplacement pour la phrase secrète dans l'en-tête de LUKS du volume chiffré. Cela se fait par la commande **cryptsetup luksKillSlot device key-slot**. Pour plus d'informations et d'exemples, voir **cryptsetup --aider**.

19.3.4. Configurer les phrases secrètes pour les urgences

Dans certaines circonstances (par exemple, lors des voyages d'affaire), il n'est pas très pratique pour les administrateurs système de travailler directement avec les systèmes affectés, mais les utilisateurs ont toujours besoin d'accéder à leurs données. Dans de tels cas, **volume_key** peut fonctionner avec des phrases secrètes, ainsi qu'avec des clés de chiffrement.

Lors de l'installation du système, exécutez :

```
volume_key --save /path/to/volume -c /path/to/ert --create-random-passphrase  
passphrase-packet
```

Cela génère une phrase secrète au hasard, l'ajoute au volume spécifié et la stocke dans le **passphrase-packet**. Il est également possible de combiner les options **--create-random-passphrase** et **-o** pour générer les deux paquets en même temps.

Si un utilisateur oublie le mot de passe, l'utilisateur désigné exécutera :

```
volume_key --secrets -d /your/nss/directory passphrase-packet
```

Cela affichera la phrase secrète au hasard. Donnez cette phrase secrète à l'utilisateur final.

19.4. Références **volume_key**

Vous pourrez trouver des informations supplémentaires sur **volume_key** à l'adresse suivante :

- ✦ dans le fichier `readme` qui se situe `/usr/share/doc/volume_key-*/README`
- ✦ dans la page `man` de `volume_key` en exécutant `man volume_key`
- ✦ en ligne à l'adresse suivante http://fedoraproject.org/wiki/Disk_encryption_key_escrow_use_cases

Chapitre 20. Directives de déploiement des disques SSD

Les disques SSD (« *Solid-state disks* ») sont des périphériques de stockage qui utilisent des puces flash NAND pour stocker des données de manière persistante. Ceci les différencie des précédentes générations de disques, qui stockent les données sur des plateaux magnétiques rotatifs. Sur les disques SSD, le temps d'accès aux données sur la totalité de la plage d'adresse de bloc logique (« Logical Block Address », ou LBA) est constant tandis qu'avec les disques plus anciens qui utilisent des supports rotatifs, les schémas d'accès s'étendant sur des plages d'adresses de grande taille encourrent des coûts de recherche. Ainsi, les périphériques SSD offrent une meilleure latence et un meilleur débit.

Les performances se dégradent au fur et à mesure que le nombre de blocs utilisés se rapproche de la capacité maximale du disque. Le degré d'impact des performances peut beaucoup varier en fonction du fournisseur. Cependant, tous les périphériques se dégraderont quelque peu.

Pour répondre au problème de dégradation, le système hôte (par exemple, le noyau Linux) peut utiliser des requêtes d'abandon pour informer le stockage qu'une plage de blocs donnée n'est plus en cours d'utilisation. Un disque SSD peut utiliser ce type d'information pour libérer de l'espace interne en utilisant les blocs disponibles pour ré-équilibrer l'usure. Les abandons seront uniquement émis si le stockage publicise la prise en charge de son protocole de stockage (qu'il s'agisse du protocole ATA ou SCSI). Les requêtes d'abandon sont émises au stockage à l'aide de la commande d'abandon négociée spécifique au protocole de stockage (commande **TRIM** pour ATA, et **WRITE SAME** si **UNMAP** est défini, ou la commande **UNMAP** pour SCSI).

Activer le support **discard** est surtout utile quand les deux points suivants s'appliquent :

- Il y a de l'espace libre toujours disponible sur le système de fichiers.
- La plupart des blocs logiques situés sur le périphérique de stockage sous-jacent contiennent déjà des écritures.

Pour plus d'informations sur **TRIM**, voir *Data Set Management T13 Specifications* dans le lien suivant :

http://t13.org/Documents/UploadedDocuments/docs2008/e07154r6-Data_Set_Management_Proposal_for_ATA-ACS2.doc

Pour obtenir des informations supplémentaires sur **UNMAP**, veuillez consulter la section 4.7.3.4 du document *SCSI Block Commands 3 T10 Specification*, disponible sur le lien suivant :

<http://www.t10.org/cgi-bin/ac.pl?t=f&f=sbc3r26.pdf>



Note

Les périphériques SSD sur le marché n'offrent pas tous la prise en charge de **discard**. Pour déterminer si votre disque dur offre la prise en charge de **discard**, veuillez vérifier la présence de `/sys/block/sda/queue/discard_granularity`.

20.1. Considérations pour le déploiement

À cause de la structure interne et de l'opération des disques SSD, il vaut mieux partitionner les périphériques sur une *limite de bloc de suppression* interne (« Erase block boundary »). Les utilitaires de partitionnement sous Red Hat Enterprise Linux 7 choisissent des valeurs par défaut saines si le disque SSD exporte ses informations de topologie.

Cependant, si le périphérique *n'exporte pas* ses informations de topologie, Red Hat recommande que la première partition soit créée sur une limite de 1 Mo.

Les Logical Volume Manager (LVM), les cibles de device-mapper (DM) targets, et de MD (software raid) que LVM utilise prend en charge les abandons (discard). Les seules cibles DM qui ne prennent pas en charge les abandons sont dm-snapshot, dm-crypt, et dm-raid45. La prise en charge des abandons pour dm-mirror a été ajoutée sur Red Hat Enterprise Linux 6.1 et à partir de 7.0, MD prend en charge les abandons.

Red Hat avertit également que l'utilisation des niveaux RAID logiciel 1, 4, 5, et 6 sur des disques SSD n'est pas recommandée. Pendant l'étape d'initialisation de ces niveaux RAID, certains utilitaires de gestion RAID (tels que **mdadm**) écrivent sur *tous* les blocs du périphérique de stockage pour s'assurer que les checksums fonctionnent correctement. Ceci entraîne une dégradation rapide des performances du disque SSD.

À partir de Red Hat Enterprise Linux 6.4, ext4 et XFS sont les seuls systèmes de fichiers totalement pris en charge qui offrent les commandes **discard**. Dans les versions précédentes de Red Hat Enterprise Linux 6, seul ext4 prenait totalement en charge **discard**. Pour activer les commandes **discard** sur un périphérique, veuillez utiliser l'option **mount** de **discard**. Par exemple, pour monter **/dev/sda2** sur **/mnt** lorsque **discard** est activé, veuillez exécuter :

```
# mount -t ext4 -o discard /dev/sda2 /mnt
```

Par défaut, ext4 ne délivre pas la commande **discard**. Cela est le cas afin d'éviter tout problème sur des périphériques qui pourraient ne pas implémenter la commande **discard** correctement. Le code **swap** Linux délivrera les commandes **discard** aux périphériques sur lesquels **discard** est activé, il n'existe aucune option pour contrôler ce comportement.

20.2. Considérations pour le paramétrage

Cette section décrit plusieurs facteurs à prendre en considération lors de la configuration des paramètres pouvant affecter les performances SSD.

Planificateur d'E/S

Tous les planificateurs d'E/S devraient fonctionner correctement avec la plupart des disques SSD. Cependant, comme avec tout autre type de stockage, Red Hat recommande d'effectuer des mesures de performance pour déterminer la configuration optimale pour une charge donnée.

Lors de l'utilisation de disques SSD, Red Hat recommande de modifier le planificateur d'E/S uniquement pour mesurer les performances de charges de travail particulières. Pour obtenir des informations supplémentaires sur les différents types de planificateur d'E/S, veuillez consulter le *Guide de paramétrage des E/S* (fournit par Red Hat). Le document sur le noyau suivant contient également des instructions sur la manière de basculer entre planificateurs d'E/S :

/usr/share/doc/kernel-version/Documentation/block/switching-sched.txt

À partir de Red Hat Enterprise Linux 7.0, le planificateur de I/O par défaut est Deadline, sauf pour un usage avec des disques SATA. Dans le cas de disques SATA, CFQ est le planificateur de I/O par défaut. Pour un stockage plus rapide, Deadline peut surpasser CFQ, conduisant à de meilleures performances d'e/s sans la nécessité d'un réglage spécifique. Il est possible, cependant, que la valeur par défaut ne soit pas adaptée à certains disques (tels que les disques de rotation SAS). Dans ce cas le planificateur I/O devra être changé à CFQ.

Mémoire virtuelle

Tout comme le planificateur d'E/S, le sous-système de la mémoire virtuelle (VM) ne requiert pas de paramétrage particulier. Au vu de la nature rapide des E/S sur disques SSD, il devrait être possible de diminuer les paramètres **vm_dirty_background_ratio** et **vm_dirty_ratio**, car une augmentation des

activités d'écriture ne devrait pas avoir d'impact négatif sur la latence des autres opérations sur le disque. Cependant, ceci peut également générer *davantage d'E/S générales* est n'est donc pas recommandé sans effectuer de tests spécifiques aux charges au préalable.

Mémoire swap

Un disque SSD peut également être utilisé en tant que périphérique swap, et est susceptible de produire de bonnes performances pour les chargements et renvois de pages.

Chapitre 21. Barrières d'écriture

Une *barrière d'écriture* est un mécanisme du noyau utilisé pour s'assurer que les métadonnées du système de fichiers soient écrites et ordonnées correctement sur un stockage persistant, même lorsque les périphériques de stockage avec des caches d'écriture volatiles subissent une perte de puissance. Les systèmes de fichiers avec des barrières d'écriture activées assurent aussi que toutes données transmises via **fsync()** persistent lors de pannes de courant.

Cependant, l'activation des barrières d'écritures provoque une réduction significative des performances de certaines applications. Les applications qui utilisent beaucoup **fsync()**, ou qui créent et suppriment de nombreux fichiers de petite taille fonctionneront probablement bien plus lentement.

21.1. Importance des barrières d'écriture

Les systèmes de fichiers prennent grand soin de mettre à jour les métadonnées, s'assurant ainsi de leur cohérence. Les systèmes de fichiers journalisés assemblent les mises à jour de métadonnées en transactions et les envoient sur le stockage persistant comme suit :

1. Premièrement, le système de fichiers envoie le corps de la transaction au périphérique de stockage.
2. Puis le système de fichiers envoie un bloc de validation.
3. Si la transaction et son bloc de validation correspondant sont écrits sur le disque, le système de fichiers supposera que la transaction pourra survivre à une panne de courant.

Cependant, l'intégrité des systèmes de fichiers pendant les pannes de courant est de plus en plus complexe pour les périphériques de stockage avec des caches supplémentaires. Les périphériques de stockage cibles comme les disques S-ATA ou SAS peuvent offrir des caches d'écriture allant de 32 Mo à 64 Mo (avec des disques modernes). Les contrôleurs RAID matériel contiennent souvent des caches d'écriture internes. De plus, les matrices haut de gamme, comme celles de NetApp, IBM, Hitachi et EMC (entre autres), offrent également des matrices de grande taille.

Les périphériques de stockage avec des caches d'écriture rapportent les E/S comme étant « complete » (terminées) lorsque les données sont dans le cache ; si le cache subit une panne de courant, les données seront également perdues. Pire encore, au fur et à mesure que le cache se déplace sur le stockage persistant, il peut modifier l'ordre original des métadonnées. Lorsque cela se produit, le bloc de validation peut se trouver sur le disque sans que la transaction terminée et associée ne soit en place. Par conséquent, le journal pourrait réutiliser ces blocs de transactions non-initialisées dans le système de fichiers pendant la récupération après la perte de courant ; ceci pourrait provoquer la corruption et l'incohérence des données.

Comment les barrières d'écriture fonctionnent

Les barrières d'écriture sont implémentées dans le noyau Linux à travers des vidages de caches d'écriture du stockage avant et après les E/S, dont l'*ordre est critique*. Une fois la transaction écrite, le cache du stockage est vidé, le bloc de validation est écrit, puis le cache est à nouveau vidé. Ceci permet de s'assurer que :

- » Le disque contient bien toutes les données.
- » Aucun changement d'ordre ne s'est produit.

Avec les barrières activées, un appel **fsync()** provoquera également le vidage du cache du stockage. Ceci garantit la persistance sur disque des données de fichiers même si une panne de courant se produit peu après le retour de **fsync()**.

21.2. Activer/désactiver les barrières d'écriture

21.2. Activer/désactiver les barrières d'écriture

Pour réduire le risque de corruption de données pendant les pannes de courant, certains périphériques de stockage utilisent des caches d'écriture avec batteries de secours. En général, les matrices haut de gamme et certains contrôleurs de matériel utilisent des caches d'écriture avec batteries de secours. Cependant, comme la volatilité du cache est invisible pour le noyau, Red Hat Enterprise Linux 7 active par défaut les barrières d'écriture sur tous les systèmes de fichiers journalisant pris en charge.



Note

Les caches d'écriture sont conçus pour augmenter la performance E/S. Cependant, en activant les barrières d'écriture implique qu'il faille constamment vider ces caches, ce qui peut réduire la performance de manière importante.

Pour les périphériques avec des caches d'écriture à batteries de secours non volatiles, et pour ceux sur lesquels la mise en cache d'écriture est désactivée, vous pouvez désactiver les barrières d'écriture en toute sécurité au moment du montage, en utilisant l'option **-o nobarrier** de **mount**. Cependant, certains périphériques ne prennent pas en charge les barrières d'écriture ; ce type de périphérique journalisera un message d'erreur sur **/var/log/messages** (veuillez consulter [Tableau 21.1, « Messages d'erreur des barrières d'écriture par système de fichiers »](#)).

Tableau 21.1. Messages d'erreur des barrières d'écriture par système de fichiers

Système de fichiers	Message d'erreur
ext3/ext4	JBD: barrier-based sync failed on device - disabling barriers
XFS	Filesystem device - Disabling barriers, trial barrier write failed
btrfs	btrfs: disabling barriers on dev device

21.3. Considérations pour barrières d'écriture

Certaines configurations de système ne nécessitent pas de barrières d'écriture pour protéger les données. Dans la plupart des cas, d'autres méthodes sont préférables aux barrières d'écriture, car l'activation des barrières d'écriture entraîne une importance baisse de la qualité des performances.

Désactiver les caches d'écriture

Alternativement, une manière d'éviter les problèmes d'intégrité des données consiste à s'assurer qu'aucun cache d'écriture ne perde de données lors des pannes de courant. Lorsque possible, la meilleure manière de configurer ceci consiste simplement à désactiver le cache d'écriture. Sur un simple serveur ou ordinateur de bureau avec un ou plusieurs disques SATA (provenant d'une pièce AHCI Intel d'un contrôleur SATA local), vous pouvez désactiver le cache d'écriture sur les disques SATA cibles par la commande **hdparm**, comme suit :

```
# hdparm -W0 /device/
```

Caches d'écriture avec batteries de secours

Les barrières d'écriture ne sont pas nécessaires lorsque le système utilise des contrôleurs RAID matériel avec des caches d'écriture avec batteries de secours. Si le système est équipé avec de tels contrôleurs et si les caches d'écriture des disques sont désactivés, le contrôleur se fera connaître comme cache à écriture synchrone ; ceci informera le noyau que les données du cache d'écriture survivront à une panne de courant.

La plupart des contrôleurs utilisent des outils spécifiques aux fournisseurs pour effectuer des requêtes et pour manipuler les disques cibles. Par exemple, le contrôleur SAS Megaraid LSI utilise un cache d'écriture avec batterie de secours ; ce type de contrôleur requiert l'outil **MegaCli64** pour gérer les disques cibles. Pour afficher l'état de tous les disques d'arrière-plan pour SAS Megaraid LSI, veuillez utiliser :

```
# MegaCli64 -LDGetProp -DskCache -LAll -aALL
```

Pour désactiver le cache d'écriture de tous les disques d'arrière-plan pour SAS Megaraid LSI, veuillez utiliser :

```
# MegaCli64 -LDSetProp -DisDskCache -Lall -aALL
```



Note

Les cartes RAID matériel rechargent leurs batteries pendant que le système est opérationnel. Si un système est éteint pendant une longue période, les batteries se déchargeront, laissant ainsi les données stockées, vulnérables pendant une panne d'alimentation.

Matrices haut de gamme

Les matrices haut de gamme offrent diverses manières de protéger les données en cas de panne de courant. Ainsi, il n'est pas nécessaire de vérifier l'état des disques internes dans le stockage RAID externe.

NFS

Les clients NFS n'ont pas besoin d'activer les barrières d'écriture puisque l'intégrité des données est gérée côté serveur NFS. Ainsi, les serveurs NFS doivent être configurés afin d'assurer la persistance des données pendant une panne de courant (que ce soit avec des barrières d'écriture ou à l'aide d'autres moyens).

Chapitre 22. Alignement et taille des E/S de stockage

De récentes améliorations apportées aux standards SCSI et ATA permettent aux périphériques de stockage d'indiquer leurs préférences quant à l'*alignement des E/S* et à la *taille des E/S* (ce qui peut même être requis dans certains cas). Ces informations sont particulièrement utiles avec les nouveaux lecteurs de disques qui augmentent la taille du secteur physique de 512 octets à 4 Ko. Ces informations peuvent aussi être bénéfiques aux périphériques RAID, pour lesquels les tailles de bloc et d'entrelacement peuvent avoir un impact sur la performance.

La pile d'E/S Linux a été améliorée pour traiter l'alignement et la taille des E/S offerts par les fournisseurs, permettant aux outils de gestion de stockage (**parted**, **lvm**, **mkfs.***, etc.) d'optimiser le placement et l'accès aux données. Si un périphérique hérité n'exporte pas les données d'alignement et de taille des E/S, alors les outils de gestion de stockage Red Hat Enterprise Linux 7 aligneront de manière conservatrice les E/S sur une limite de 4 Ko (ou sur une puissance de 2). Ceci assurera que les périphériques à secteurs de 4 Ko opéreront correctement même s'ils n'indiquent pas d'alignement et de taille d'E/S requis ou préféré.

Veillez consulter [Section 22.2, « Accès à l'espace utilisateur »](#) afin d'apprendre comment déterminer les informations que le système d'exploitation a obtenu du périphérique. Ces données sont ensuite utilisées par les outils de gestion de stockage pour déterminer le placement des données.

Le planificateur d'E/S a changé dans Red Hat Enterprise Linux 7. Le planificateur d'E/S par défaut se nomme désormais *Deadline*, à l'exception des disques SATA. CFQ est le planificateur d'E/S par défaut des disques SATA. Pour un stockage plus rapide, Deadline surpasse CFQ et lors de son utilisation, une augmentation des performances se fait sentir sans qu'il soit nécessaire d'effectuer de réglages.

Si la valeur par défaut n'est pas correcte pour certains disques (par exemple, pour les disques à rotation SAS), alors veuillez changer le planificateur d'E/S sur CFQ. Cette instance dépendra de la charge de travail.

22.1. Paramètres d'accès au stockage

Le système d'exploitation utilise les informations suivantes pour déterminer l'alignement et la taille des E/S :

physical_block_size

Unité interne la plus petite sur laquelle le périphérique peut opérer

logical_block_size

Utilisé de manière externe pour adresser un emplacement sur le périphérique

alignment_offset

Nombre d'octets de décalage du début du périphérique bloc Linux (périphérique partition/MD/LVM) par rapport à l'alignement physique sous-jacent

minimum_io_size

Unité minimale préférée du périphérique pour les E/S aléatoires

optimal_io_size

Unité préférée du périphérique pour la transmission d'E/S

Par exemple, certains périphériques à secteurs de 4K peuvent utiliser une taille **physical_block_size** de 4K de manière interne mais présenter une taille **logical_block_size** de 512 octets plus granuleuse dans Linux. Cet écart présente un certain potentiel pour des E/S non-alignées. Pour répondre à ce problème, la pile d'E/S Red Hat Enterprise Linux 7 tentera de démarrer toutes les zones de données sur une limite

naturellement alignée (**physical_block_size**) en s'assurant de prendre en compte tout décalage d'alignement « `alignment_offset` » si le début du périphérique bloc est décalé par rapport à l'alignement physique sous-jacent.

Les fournisseur de stockage peuvent également fournir des *indicateurs sur les E/S* à propos de l'unité minimale préférée pour des E/S aléatoires (**minimum_io_size**) ainsi que pour les E/S de transmission (**optimal_io_size**) d'un périphérique. Par exemple, **minimum_io_size** et **optimal_io_size** pourraient correspondre, respectivement, aux tailles de bloc et d'entrelacement d'un périphérique RAID.

22.2. Accès à l'espace utilisateur

Veillez toujours prendre soin d'utiliser des E/S de taille et d'alignement corrects. Ceci est particulièrement important pour l'accès aux E/S directes. Les E/S directes doivent être alignées sur une limite **logical_block_size**, et sur des multiples de **logical_block_size**.

Avec les périphériques 4K natifs (par exemple, **logical_block_size** font 4K) il est désormais critique que les applications effectuent des E/S directes multiples de **logical_block_size**. Ceci signifie que les applications échoueront avec les périphériques 4k qui effectuent des E/S alignées sur 512 octets plutôt que sur des E/S alignées 4k.

Pour éviter ceci, une application doit consulter les paramètres d'E/S d'un périphérique afin de s'assurer qu'il utilise l'alignement et la taille des E/S corrects. Comme mentionné ultérieurement, les paramètres d'E/S sont exposés via les interfaces **sysfs** et **ioctl** de périphérique bloc.

Pour obtenir plus de détails, veuillez consulter **man libblkid**. Cette page **man** est fournie par le paquet **libblkid-devel**.

Interface sysfs

✧ `/sys/block/disk/alignment_offset`

ou

`/sys/block/disk/partition/alignment_offset`



Note

L'emplacement du fichier dépendra si le disque est un disque physique (local, RAID local ou LUN multivoies) ou un disque virtuel. Le premier emplacement s'applique aux disques physiques alors que le second s'applique aux disques virtuels. La raison pour ceci est que `virtio-blk` rapportera toujours une valeur d'alignement à la partition. Les disques physiques rapporteront ou non un valeur d'alignement.

✧ `/sys/block/disk/queue/physical_block_size`

✧ `/sys/block/disk/queue/logical_block_size`

✧ `/sys/block/disk/queue/minimum_io_size`

✧ `/sys/block/disk/queue/optimal_io_size`

Le noyau exportera toujours ces attributs **sysfs** pour les périphériques « hérités » qui ne fournissent pas d'informations sur les paramètres d'E/S, par exemple :

■

Exemple 22.1. Interface sysfs

```
alignment_offset:    0
physical_block_size: 512
logical_block_size: 512
minimum_io_size:    512
optimal_io_size:    0
```

ioctl du périphérique bloc

- ✧ **BLKALIGNOFF**: `alignment_offset`
- ✧ **BLKPBSZGET**: `physical_block_size`
- ✧ **BLKSSZGET**: `logical_block_size`
- ✧ **BLKIOMIN**: `minimum_io_size`
- ✧ **BLKIOOPT**: `optimal_io_size`

22.3. Standards

Cette section décrit les standards d'E/S utilisés par les périphériques ATA et SCSI.

ATA

Les périphériques ATA doivent rapporter les informations correspondantes via la commande **IDENTIFY DEVICE**. Les périphériques ATA rapportent uniquement les paramètres d'E/S pour **physical_block_size**, **logical_block_size**, et **alignment_offset**. Les indicateurs d'E/S supplémentaires se trouvent hors du champ de l'ensemble des commandes ATA.

SCSI

Les paramètres d'E/S pris en charge sur Red Hat Enterprise Linux 7 requièrent au moins la *version 3* du protocole SPC-3 (« *SCSI Primary Commands* »). Le noyau enverra uniquement une *consultation étendue* (qui obtient accès à la page **BLOCK LIMITS VPD**) et une commande **READ CAPACITY(16)** sur les périphériques se réclamant être conformes à SPC-3.

La commande **READ CAPACITY(16)** fournit le décalage des tailles et alignements des blocs :

- ✧ **LOGICAL BLOCK LENGTH IN BYTES** (« Longueur de bloc logique en octets ») est utilisé pour dériver `/sys/block/disque/queue/physical_block_size`
- ✧ **LOGICAL BLOCKS PER PHYSICAL BLOCK EXPONENT** (« Blocs logiques par exposant de bloc physique ») est utilisé pour dériver `/sys/block/disque/queue/logical_block_size`
- ✧ **LOWEST ALIGNED LOGICAL BLOCK ADDRESS** (« Adresse du bloc logique aligné au plus bas ») est utilisée pour dériver :
 - `/sys/block/disk/alignment_offset`
 - `/sys/block/disk/partition/alignment_offset`

La page **BLOCK LIMITS VPD (0xb0)** fournit les indicateurs d'E/S. **OPTIMAL TRANSFER LENGTH**

GRANULARITY et **OPTIMAL TRANSFER LENGTH** sont également utilisés pour dériver :

- ✦ `/sys/block/disk/queue/minimum_io_size`
- ✦ `/sys/block/disk/queue/optimal_io_size`

Le paquet **sg3_utils** fournit l'utilitaire **sg_inq**, qui peut être utilisé pour accéder à la page **BLOCK LIMITS VPD**. Pour ce faire, veuillez exécuter :

```
# sg_inq -p 0xb0 disk
```

22.4. Empiler les paramètres d'E/S

Toutes les couches de la pile d'E/S Linux ont été conçues pour propager les divers paramètres d'E/S à travers la pile. Lorsqu'une couche consomme un attribut ou agrège plusieurs périphériques, celle-ci doit exposer les paramètres d'E/S appropriés de manière à ce que les périphériques ou outils de la couche supérieure puissent avoir une vision précise du stockage tel qu'il s'est transformé. Voici quelques exemples pratiques :

- ✦ Seule une couche de la pile d'E/S doit s'ajuster pour un décalage **alignment_offset** qui n'est pas égal à zéro ; une fois que la couche s'ajuste de manière correspondante, un périphérique avec un décalage **alignment_offset** de zéro sera exporté.
- ✦ Un périphérique DM (« Device Mapper ») entrelacé créé avec LVM doit exporter des valeurs **minimum_io_size** et **optimal_io_size** relatives au compte des entrelacements (nombre de disques) et à la taille de bloc fournie par l'utilisateur.

Dans Red Hat Enterprise Linux 7, les pilotes de périphériques (MD) Device Mapper et Software Raid peuvent être utilisés pour combiner arbitrairement des périphériques avec différents paramètres d'E/S. La couche du bloc du noyau tentera raisonnablement de combiner les paramètres d'E/S des périphériques individuels. Le noyau n'empêchera pas la combinaison de périphériques hétérogènes. Cependant, prenez soin de connaître les risques associés à ces actions.

Par exemple, un périphérique de 512 octets et un périphérique 4K peuvent être combinés en un seul périphérique logique DM, qui posséderait une taille **logical_block_size** de 4K. Les systèmes de fichiers en couche sur de tels périphériques hybrides supposent que l'écriture sur 4K sera atomique, mais en réalité, cela s'étendra sur 8 adresses de blocs logiques lors de l'exécution sur le périphérique de 512 octets. L'utilisation de la valeur **logical_block_size** 4K pour un périphérique de 512 octets de haut niveau augmente le potentiel d'écriture partielle sur le périphérique de 512 octets en cas de panne du système.

Si la combinaison des paramètres d'E/S de multiples périphériques résulte en un conflit, la couche du bloc peut envoyer un avertissement que le périphérique est susceptible d'effectuer des écritures partielles ou qu'il n'est pas aligné correctement.

22.5. Gestionnaire de volumes logiques LVM

LVM fournit les outils de l'espace utilisateur qui sont utilisés pour gérer les périphériques DM du noyau. LVM déplace le début de la zone des données (qu'un périphérique DM donné utilisera) pour prendre en compte un décalage **alignment_offset** qui n'est pas égal à zéro et associé à tout périphérique géré par LVM. Cela signifie que les volumes logiques seront correctement alignés (**alignment_offset=0**).

Par défaut, LVM s'ajustera à tout décalage **alignment_offset**, mais ce comportement peut être désactivé en paramétrant **data_alignment_offset_detection** sur **0** dans `/etc/lvm/lvm.conf`. Cette désactivation n'est pas recommandée.

LVM détectera également les indicateurs d'E/S d'un périphérique. Le début de la zone de données d'un périphérique sera un multiple de la valeur **minimum_io_size** ou **optimal_io_size** exposées dans sysfs. LVM utilisera **minimum_io_size** si la valeur **optimal_io_size** n'est pas définie (c'est-à-dire égale à **0**).

Par défaut, LVM déterminera automatiquement ces indicateurs d'E/S, mais ce comportement peut être désactivé en paramétrant **data_alignment_detection** sur **0** dans **/etc/lvm/lvm.conf**. Cette désactivation n'est pas recommandée.

22.6. Outils des partitions et systèmes de fichiers

Cette section décrit en quoi différents outils de gestion de systèmes de fichiers et de partitionnement interagissent différemment avec les paramètres d'E/S d'un périphérique.

libblkid de util-linux-ng et fdisk

La bibliothèque **libblkid** fournie avec le paquet **util-linux-ng** inclut une interface de programmation pour accéder aux paramètres d'E/S d'un périphérique. **libblkid** permet aux applications, particulièrement celles qui utilisent des E/S directes, de redimensionner correctement leurs requêtes d'E/S. L'utilitaire **fdisk** de **util-linux-ng** utilise **libblkid** pour déterminer les paramètres d'E/S d'un périphérique pour un placement optimal de toutes les partitions. L'utilitaire **fdisk** alignera toutes les partitions sur une limite de 1 Mo.

parted et libparted

La bibliothèque **libparted** de **parted** utilise également l'interface de programmation des paramètres d'E/S de **libblkid**. L'installateur Red Hat Enterprise Linux 7 (**Anaconda**) utilise **libparted**, ce qui signifie que toutes les partitions créées par l'installateur ou **parted** seront correctement alignés. Pour les partitions créées sur un périphérique qui ne semble pas fournir de paramètres d'E/S, l'alignement par défaut sera de 1 Mo.

Les cas d'utilisation des heuristiques **parted** sont comme suit :

- Veuillez toujours utiliser la valeur **alignment_offset** rapportée comme décalage pour le lancement de la première partition principale.
- Si **optimal_io_size** est défini (c'est-à-dire différent de **0**), veuillez aligner toutes les partitions sur une limite **optimal_io_size**.
- Si **optimal_io_size** n'est pas défini (c'est-à-dire égal à **0**), alors **alignment_offset** est égal à **0**, et **minimum_io_size** est une puissance de 2, utilisez un alignement par défaut de 1 Mo.

Ceci sert à récupérer tous les périphériques « hérités » qui ne semblent pas fournir d'indicateur d'E/S. Ainsi, toutes les partitions par défaut seront alignées sur une limite de 1 Mo.



Note

Red Hat Enterprise Linux 7 ne fait pas de distinction entre les périphériques qui ne fournissent pas d'indicateurs d'E/S et ceux qui en fournissent par le biais de **alignment_offset=0** et **optimal_io_size=0**. Un tel périphérique peut être un périphérique 4K SAS unique. De cette manière, au pire seul 1 Mo est perdu lors du démarrage du disque.

Outils de systèmes de fichiers

Les différents utilitaires **mkfs.filesystem** ont aussi été améliorés afin de consommer les paramètres d'E/S d'un périphérique. Ces utilitaires ne permettront pas à un système de fichiers d'être formaté pour utiliser une taille de bloc plus petite que la taille **logical_block_size** du périphérique de stockage sous-jacent.

À l'exception de **mkfs.gfs2**, tous les autres utilitaires **mkfs.filesystem** utilisent également les indicateurs d'E/S pour agencer la structure des données sur disque et les zones de données relatives aux valeurs **minimum_io_size** et **optimal_io_size** des périphériques de stockage sous-jacent. Ceci permet aux systèmes de fichiers d'être formatés de manière optimale pour divers agencements (entrelacés) RAID.

Chapitre 23. Paramétrer un système sans disque distant

Pour paramétrer un système sans disque distant démarré via PXE, vous aurez besoin des paquets suivants :

- » **tftp-server**
- » **xinetd**
- » **dhcp**
- » **syslinux**
- » **dracut-network**



Note

Après avoir installé le package **dracut-network**, ajouter la ligne suivante à **/etc/dracut.conf** :

```
add_dracutmodules+="nfs"
```

Le démarrage de système sans disque distant requiert un service **tftp** (fourni par **tftp-server**) et un service DHCP (fourni par **dhcp**). Le service **tftp** est utilisé pour récupérer l'image du noyau et **initrd** sur le réseau via le chargeur PXE.



Note

SELinux n'est pris en charge que sur NFSv4.2. Pour utiliser SELinux, NFS doit être explicitement activé dans **/etc/sysconfig/nfs** en ajoutant la ligne :

```
RPCNFSDARGS="-V 4.2"
```

Puis, dans **/var/lib/tftpboot/pxelinux.cfg/default**, changer **root=nfs:server-ip:/exported/root/directory** en **root=nfs:server-ip:/exported/root/directory,vers=4.2**.

Finalement, redémarrer le serveur NFS.

Les sections suivantes soulignent les procédures nécessaires pour déployer des systèmes sans disques distants dans un environnement en réseau.



Important

Certains packages RPM ont commencé à utiliser les fonctionnalités de fichiers (comme **setcap** et **getcap**). Cependant, NFS ne les prend pas actuellement en charge, donc essayer d'installer ou de mettre à jour un package qui utilise cette fonctionnalité échouera.

23.1. Configuration d'un service tftp pour des clients sans disque

Le service **tftp** est désactivé par défaut. Pour l'activer et autoriser le démarrage PXE via le réseau, définissez l'option **Disabled** (désactivé) dans `/etc/xinetd.d/tftp` sur **no**. Pour configurer **tftp**, veuillez effectuer les étapes suivantes :

Procédure 23.1. Pour configurer tftp

1. Le répertoire racine **tftp** (**chroot**) est situé dans `/var/lib/tftpboot`. Copiez `/usr/share/syslinux/pxelinux.0` sur `/var/lib/tftpboot/` :

```
cp /usr/share/syslinux/pxelinux.0 /var/lib/tftpboot/
```

2. Créez un répertoire **pxelinux.cfg** dans le répertoire root **tftp** :

```
mkdir -p /var/lib/tftpboot/pxelinux.cfg/
```

Vous devrez aussi configurer les règles de pare-feu correctement afin d'autoriser le trafic **tftp**. Comme **tftp** prend en charge les wrappers TCP, vous pouvez configurer l'accès des hôtes à **tftp** via `/etc/hosts.allow`. Pour obtenir des informations supplémentaires sur la configuration des wrappers TCP et sur le fichier de configuration `/etc/hosts.allow`, veuillez consulter le *Guide de sécurité* de Red Hat Enterprise Linux 7. `man hosts_access` fournit également des informations sur `/etc/hosts.allow`.

Après avoir configuré **tftp** pour des clients sans disque, veuillez configurer DHCP, NFS, et le système de fichiers exporté. Veuillez consulter [Section 23.2, « Configuration DHCP pour les clients sans disque »](#) et [Section 23.3, « Configuration d'un système de fichiers exporté pour les clients sans disque »](#) pour obtenir des instructions sur la manière de procéder.

23.2. Configuration DHCP pour les clients sans disque

Après avoir configuré un serveur **tftp**, vous devrez paramétrer un service DHCP sur la même machine hôte. Veuillez consulter le *Guide de déploiement* Red Hat Enterprise Linux 7 pour obtenir des instructions sur la manière de paramétrer un serveur DHCP. En outre, vous devrez activer le démarrage PXE sur le serveur DHCP ; pour ce faire, veuillez ajouter la configuration suivante au fichier `/etc/dhcp/dhcp.conf` :

```
allow booting;
allow bootp;
class "pxeclients" {
    match if substring(option vendor-class-identifier, 0, 9) = "PXEClient";
    next-server server-ip;
    filename "pxelinux.0";
}
```

Veuillez remplacer **server-ip** par l'adresse IP de la machine hôte sur laquelle les services **tftp** et DHCP résident. Maintenant que **tftp** et DHCP sont configurés, il ne reste plus qu'à configurer NFS et le système de fichiers exporté. Veuillez consulter [Section 23.3, « Configuration d'un système de fichiers exporté pour les clients sans disque »](#) afin d'obtenir des instructions.



Note

Quand des machines virtuelles **libvirt** sont utilisées comme client sans disque, **libvirt** fournit le service DHCP et le serveur DHCP autonome n'est pas utilisé. Dans cette situation, le redémarrage du réseau doit être activé par l'option **bootp file='filename'** dans la configuration de réseau de **libvirt**, voir **virsh net-edit**.

23.3. Configuration d'un système de fichiers exporté pour les clients sans disque

Le répertoire root du système de fichiers exporté (utilisé par des clients sans disque sur le réseau) est partagé via NFS. Configurez le service NFS pour exporter le répertoire root en l'ajoutant à **/etc/exports**. Pour obtenir des instructions sur la manière d'effectuer cela, veuillez consulter [Section 8.7.1, « Fichier de configuration /etc/exports »](#).

Pour répondre à la totalité des besoins des clients sans disque, le répertoire root devrait contenir une installation Red Hat Enterprise Linux complète. Vous pouvez synchroniser ceci avec un système en cours d'exécution via **rsync** :

```
# rsync -a -e ssh --exclude='/proc/*' --exclude='/sys/*' hostname.com:/
/exported/root/directory
```

Remplacez **hostname.com** par le nom d'hôte du système d'exploitation du système en cours d'exécution avec lequel se synchroniser via **rsync**. **/exported/root/directory** est le chemin vers le système de fichiers exporté.

Alternativement, vous pouvez aussi utiliser **yum** avec l'option **--installroot** pour installer Red Hat Enterprise Linux sur un emplacement spécifique. Par exemple :

```
yum groupinstall Base --installroot=/exported/root/directory --releasever=/
```

Le système de fichiers devant être exporté doit être davantage configuré avant de pouvoir être utilisé par des clients sans disque. Pour ce faire, veuillez effectuer la procédure suivante :

Procédure 23.2. Configurer le système de fichiers

1. Configurez le fichier **/etc/fstab** du système de fichiers afin qu'il contienne (au minimum) la configuration suivante :

```
none /tmp tmpfs defaults 0 0
tmpfs /dev/shm tmpfs defaults 0 0
sysfs /sys sysfs defaults 0 0
proc /proc proc defaults 0 0
```

2. Sélectionnez le noyau que les clients sans disque devront utiliser (**vmlinux-kernel-version**) et copiez-le sur le répertoire root **tftp** :

```
# cp /boot/vmlinux-kernel-version /var/lib/tftpboot/
```

3. Créez le fichier **initrd** (c'est-à-dire **initramfs-kernel-version.img**) avec le support réseau :

```
# dracut initramfs-kernel-version.img kernel-version
```

4. Les permissions de fichiers d'initrd doivent être modifiées à 600 ou bien, le chargeur boot **pxelinux.0** échouera avec l'erreur "file not found". Effectuez ceci avec la commande suivante :

```
# chmod go-r initramfs-kernel-version.img
```

5. Copiez également le fichier **initramfs-kernel-version.img** résultant dans le répertoire de démarrage **tftp**.
6. Modifiez la configuration de démarrage par défaut afin d'utiliser **initrd** et le noyau situé à l'intérieur de **/var/lib/tftpboot**. Cette configuration devrait instruire au répertoire root du client sans disque de monter le système de fichiers exporté (**/exported/root/directory**) en lecture-écriture. Pour ce faire, veuillez configurer **/var/lib/tftpboot/pxelinux.cfg/default** avec :

```
default rhel7

label rhel7
    kernel vmlinuz-kernel-version
    append initrd=initramfs-kernel-version.img root=nfs:server-
ip:/exported/root/directory rw
```

Remplacez **server-ip** avec l'adresse IP de la machine hôte sur laquelle les services **tftp** et DHCP résident.

Le partage NFS est désormais prêt à effectuer l'export vers les clients sans disque. Ces clients peuvent être démarrés sur le réseau via PXE.

Chapitre 24. Online Storage Management

Il est souvent souhaitable d'ajouter, de supprimer, ou de redimensionner des périphériques de stockage pendant l'exécution du système d'exploitation, sans effectuer de redémarrage. Ce chapitre souligne les procédures pouvant être utilisées pour reconfigurer les périphériques de stockage sur les systèmes hôtes Red Hat Enterprise Linux 7 pendant l'exécution du système. Ce chapitre couvre les interconnexions du stockage Fibre Channel et iSCSI, d'autres types d'interconnexions pourront y être ajoutés ultérieurement.

Ce chapitre se concentre sur l'ajout, la suppression, la modification et la surveillance des périphériques de stockage. Il ne traite pas de protocoles Fibre Channel ou iSCSI en détail. Pour obtenir des informations supplémentaires sur ces protocoles, veuillez consulter d'autres documents.

Ce chapitre fait référence à divers objets **sysfs**. Red Hat tient à vous informer que les noms d'objets et la structure des répertoires **sysfs** pourraient faire l'objet de changements lors des publications de versions majeures de Red Hat Enterprise Linux. Ceci est dû au fait que le noyau Linux en amont ne fournit pas d'API interne stable. Pour obtenir des directives sur la manière de référencer les objets **sysfs** de manière transportable, veuillez vous reporter au document `/usr/share/doc/kernel-doc-version/Documentation/sysfs-rules.txt` dans l'arborescence source du noyau.



Avertissement

La reconfiguration du stockage en ligne doit être effectuée prudemment. Les échecs ou interruptions du système pendant ce processus peuvent provoquer des résultats inattendus. Red Hat vous recommande de réduire la charge du système autant que possible pendant les opérations de changement. Ceci réduira la possibilité d'erreurs d'E/S, de mémoire insuffisante, ou d'autres erreurs similaires pouvant se produire au milieu des changements de configuration. Les sections suivantes offrent des directives plus spécifiques à ce sujet.

De plus, Red Hat vous recommande d'enregistrer toutes vos données avant de reconfigurer le stockage en ligne.

24.1. Installation de la Cible

Red Hat Enterprise Linux 7 utilise la commande **targetcli** comme un front-end pour l'affichage, l'édition et l'enregistrement de la configuration de la cible Linux-IO sans avoir besoin de manipuler directement les fichiers de configuration de la cible du noyau. **targetcli** est une interface de ligne de commande qui permet à un administrateur d'exporter les ressources de stockage locales (soutenues par des fichiers, volumes, périphériques SCSI locaux ou disques RAM) vers des systèmes distants. Il a une disposition arborescente, qui comprend la saisie semi-automatique par tabulation intégrée et fournit une documentation en ligne et un support complet semi-automatique.



Note

La hiérarchie de **targetcli** ne correspond pas toujours à l'interface du noyau. C'est parce qu'elle est conçue pour être simplifiée le plus possible.



Important

Pour s'assurer que les changements faits par **targetcli** sont persistants, démarrer et activer le service cible :

```
~]# systemctl start target
~]# systemctl enable target
```

24.1.1. Installer et exécuter targetcli

Pour installer **targetcli**, exécuter :

```
# yum install targetcli
```

Démarrer le service cible :

```
# systemctl start target
```

Configurer la cible pour qu'elle démarre à l'amorçage de façon persistante :

```
# systemctl enable target
```

Pour commencer à utiliser **targetcli**, exécuter **targetcli** et obtenir une installation des trois interfaces, exécuter **ls** :

```
# targetcli
:
/> ls
o- /.....[...]
  o- backstores.....[...]
    | o- block.....[Storage Objects: 0]
    | o- fileio.....[Storage Objects: 0]
    | o- pscsi.....[Storage Objects: 0]
    | o- ramdisk.....[Storage Objects: 0]
  o- iscsi.....[Targets: 0]
  o- loopback.....[Targets: 0]
```



Note

Dans Red Hat Enterprise Linux 7.0, exécuter une commande **targetcli** du bash (par exemple, **targetcli iscsi/ create**) ne fonctionnait pas, ni ne donnait de code d'erreur. Cela a été corrigé dans Red Hat Enterprise Linux 7.1 qui fournit un code d'erreur, ce qui en facilite l'utilisation avec les scripts shell.

24.1.2. Créer un Backstore

Les backstores prennent en charge le support de différentes méthodes de stockage de données d'un LUN exporté sur une machine locale. Créer un objet de stockage définit les ressources qui seront utilisées par le backstore.



Note

Dans Red Hat Enterprise Linux 6, le terme 'backing-store' est utilisé pour parler des mappages créés. Cependant, pour éviter la confusion entre les différentes façons dont les 'backstores' peuvent être utilisés, dans Red Hat Enterprise Linux 7 le terme 'storage objects' fait référence aux mappages créés et 'backstores' est utilisé pour décrire les différents types de périphériques de sauvegarde.

Les périphériques de backstore LIO pris en charge sont les suivants :

FILEIO (Stockage par sauvegarde de fichiers de Linux)

Les objets de stockage FILEIO peuvent soit supporter l'opération **write_back** ou **write_thru**. L'opération **write_back** active le cache du système de fichiers local. Cela améliore la performance mais réduit le risque de perte de données. Il est conseillé d'utiliser la commande **write_back=false** afin de désactiver **write_back** au bénéfice de **write_thru**.

Pour créer un objet de stockage fileio, exécuter la commande **/backstores/fileio create file_name file_location file_size write_back=false**. Exemple :

```
> /backstores/fileio create file1 /tmp/disk1.img 200M
write_back=false
Created fileio file1 with size 209715200
```

BLOCK (périphériques BLOCK Linux)

Le pilote de blocs permet l'utilisation de n'importe quel périphérique en bloc situé dans **/sys/block** à utiliser dans LIO. Inclut les périphériques physiques comme les HDDs, SSDs, CDs, DVDs) et les périphériques logiques comme les logiciels ou matériels de volumes RAID, ou les volumes LVM.



Note

Les backstores BLOCK fournissent généralement la meilleure performance.

Pour créer un backstore BLOCK par le périphérique block **/dev/sdb**, utiliser la commande suivante :

```
> /backstores/block create name=block_backend dev=/dev/sdb
Generating a wwn serial.
Created block storage object block_backend using /dev/sdb.
```

PSCSI (périphériques Linux pass-through SCSI)

Tout objet de stockage qui supporte Direct Pass Through des commandes SCSI sans émulation SCSI, avec un périphérique SCSI sous-jacent qui apparaît avec lsscsi dans **/proc/scsi/scsi** (comme un disque dur SAS) peut être configuré en tant que backstore. SCSI-3 et versions supérieures sont pris en charge dans ce sous-système.



Avertissement

PSCSI doit être uniquement utilisé par les utilisateurs avancés. Les commandes avancées de SCSI comme ALUA (Asymmetric Logical Unit Assignment) ou Persistent Reservations (utilisées par VMware ESX et vSphere) ne sont généralement pas implémentées dans le firmware et peuvent provoquer des dysfonctionnements ou des pannes. En cas de doute, utilisez BLOCK pour les installations en production à la place.

Pour créer un backstore PSCSI de périphérique physique SCSI, un périphérique **TYPE_ROM** utilisant **/dev/sr0** dans ce exemple, exécutez :

```
/> backstores/pscsi/ create name=pscsi_backend dev=/dev/sr0
Generating a wwn serial.
Created pscsi storage object pscsi_backend using /dev/sr0
```

Memory Copy RAM disk (Linux RAMDISK_MCP)

Les disques Memory Copy RAM (**ramdisk**) vous donnent des disques de RAM avec émulation SCSI et des mappages de mémoire séparés qui utilisent les mappages en copiant la mémoire des initiateurs. Cela vous donne une capacité sur plusieurs sessions et c'est particulièrement utile pour un stockage en masse volatile à but de production.

Pour créer 1Go Disque RAM Backstore, exécuter la commande suivante :

```
/> backstores/ramdisk/ create name=rd_backend size=1GB
Generating a wwn serial.
Created rd_mcp ramdisk rd_backend with size 1GB.
```

24.1.3. Créer une Cible iSCSI

Créer une Cible iSCSI :

Procédure 24.1. Créer une Cible iSCSI

1. Exécuter **targetcli**.
2. Rendez-vous sur le chemin de configuration iSCSI :

```
/> iscsi/
```



Note

La commande **cd** est également acceptée pour les changements de répertoires, ainsi que pour indiquer le chemin vers lequel se déplacer.

3. Créer une cible iSCSI en utilisant un nom de cible par défaut.


```
/iscsi> create
Created target
iqn.2003-01.org.linux-iscsi.hostname.x8664:sn.78b473f296ff
Created TPG1
```

Ou bien créer une cible iSCSI en utilisant un nom spécifique.

```
/iscsi > create iqn.2006-04.com.example:444
Created target iqn.2006-04.com.example:444
Created TPG1
```

4. Vérifier que la cible nouvellement créée soit visible quand les cibles sont répertoriées par la commande **ls**.

```
/iscsi > ls
o- iscsi.....[1 Target]
  o- iqn.2006-04.com.example:444.....[1 TPG]
    o- tpg1.....[enabled, auth]
      o- acs.....[0 ACL]
      o- luns.....[0 LUN]
      o- portals.....[0 Portal]
```



Note

À partir de Red Hat Enterprise Linux 7.1, quand une cible est créée, un portail par défaut est créé également.

24.1.4. Configuration d'un Portail iSCSI

Pour configurer un portail iSCSI, on doit tout d'abord créer une cible iSCSI et l'associer à un TPG. Pour obtenir des instructions sur la façon de procéder, consulter [Section 24.1.3, « Créer une Cible iSCSI »](#).



Note

À partir de Red Hat Enterprise Linux 7.1, quand une cible iSCSI est créée, un portail par défaut est créé également. Ce portail est défini pour écouter toutes les adresses IP avec le numéro de port par défaut (c-a-d 0.0.0.0:3260). Pour supprimer ceci, et n'ajouter que des portails spécifiques, utiliser **/iscsi/iqn-name/tpg1/portals delete ip_address=0.0.0.0 ip_port=3260**, puis créer un nouveau portail avec les informations requises.

Procédure 24.2. Créer un portail iSCSI

1. Se rendre dans le TPG.

```
/iscsi> iqn.2006-04.example:444/tpg1/
```

2. Il y a deux façons de créer un portail : créer un portail par défaut, ou créer un portail spécifiant quelle adresse IP écouter.

Créer un portail par défaut qui utilise le port par défaut 3260 et qui permet à la cible d'écouter sur toutes les adresses IP de ce port.

```
/iscsi/iqn.20...mple:444/tpg1> portals/ create
Using default IP port 3260
Binding to INADDR_Any (0.0.0.0)
Created network portal 0.0.0.0:3260
```

Pour créer un portail spécifiant sur quelle adresse IP écouter, exécuter la commande suivante.

```
/iscsi/iqn.20...mple:444/tpg1> portals/ create 192.168.122.137
Using default IP port 3260
Created network portal 192.168.122.137:3260
```

3. Vérifier que le portail nouvellement créé soit visible par la commande **ls**.

```
/iscsi/iqn.20...mple:444/tpg1> ls
o- tpg..... [enabled, auth]
  o- acls .....[0 ACL]
  o- luns .....[0 LUN]
  o- portals .....[1 Portal]
    o- 192.168.122.137:3260.....[OK]
```

24.1.5. Configurer les LUN

Pour configurer les LUN, vous devez tout d'abord créer des objets de stockage. Voir [Section 24.1.2, « Créer un Backstore »](#) pour plus d'informations.

Procédure 24.3. Configurer les LUN

1. Créer des LUN d'objets de stockage déjà créés.

```
/iscsi/iqn.20...mple:444/tpg1> luns/ create
/backstores/ramdisk/ramdisk1
Created LUN 0.

/iscsi/iqn.20...mple:444/tpg1> luns/ create /backstores/block/block1
Created LUN 1.

/iscsi/iqn.20...mple:444/tpg1> luns/ create /backstores/fileio/file1
Created LUN 2.
```

2. Afficher les changements.

```
/iscsi/iqn.20...mple:444/tpg1> ls
o- tpg..... [enabled, auth]
  o- acls .....[0 ACL]
  o- luns .....[3 LUNs]
    | o- lun0.....[ramdisk/ramdisk1]
    | o- lun1.....[block/block1 (/dev/vdb1)]
    | o- lun2.....[fileio/file1 (/foo.img)]
  o- portals .....[1 Portal]
    o- 192.168.122.137:3260.....[OK]
```

**Note**

Nous vous rappelons que le nom LUN par défaut démarre par 0, et non 1, ce qui était le cas avec **tgtd** dans Red Hat Enterprise Linux 6.

**Important**

Par défaut, les LUN sont créés en permissions lecture-seule uniquement. Si un nouveau LUN est ajouté une fois que les ACL ont été créés, ce LUN sera mappé automatiquement à tous les ACL disponibles. Cela peut créer un risque pour la sécurité. Utiliser la procédure suivante pour créer un LUN en lecture-seule.

Procédure 24.4. Créer un LUN en lecture-seule

1. Pour créer un LUN avec les permissions lecture-seule, exécuter la commande suivante pour commencer :

```
/> set global auto_add_mapped_luns=false
Parameter auto_add_mapped_luns is now 'false'.
```

Cela évite l'auto-mappage des LUN en ACL existants, ce qui permet le mappage manuel des LUN.

2. Puis, créer manuellement le LUN par la commande **iscsi/target_iqn_name/tpg1/acls/initiator_iqn_name/ create mapped_lun=next_sequential_LUN_number tpg_lun_or_backstore=backstore write_protect=1**.

```
/> iscsi/iqn.2015-06.com.redhat:target/tpg1/acls/iqn.2015-06.com.redhat:initiator/ create mapped_lun=1
tpg_lun_or_backstore=/backstores/block/block2 write_protect=1
Created LUN 1.
Created Mapped LUN 1.
/> ls
o- / ..... [...]
  o- backstores ..... [...]
    <snip>
  o- iscsi ..... [Targets: 1]
    | o- iqn.2015-06.com.redhat:target ..... [TPGs: 1]
    |   o- tpg1 ..... [no-gen-acls, no-auth]
    |     o- acls ..... [ACLs: 2]
    |       | o- iqn.2015-06.com.redhat:initiator .. [Mapped LUNs: 2]
    |         | | o- mapped_lun0 ..... [lun0 block/disk1 (rw)]
    |         | | o- mapped_lun1 ..... [lun1 block/disk2 (ro)]
    |         o- luns ..... [LUNs: 2]
    |           | o- lun0 ..... [block/disk1 (/dev/vdb)]
    |           | o- lun1 ..... [block/disk2 (/dev/vdc)]
    <snip>
```

La ligne `mapped_lun1` a maintenant (ro) à la fin (à la différence de `mapped_lun0`'s (rw)) ce qui indique « read-only » (lecture-seule).

24.1.6. Configurez les ACL

Créer une ACL pour chaque initiateur qui doivent se connecter. Ceci applique l'authentification lorsque cet initiateur se connecte, permettant seulement aux LUN d'être exposés à chaque initiateur. Chaque initiator a généralement un accès exclusif à un LUN. Les cibles et les initiateurs ont un nom identifiant unique. Le nom unique de l'initiateur doit être connu pour pouvoir configurer les ACL. Pour les initiateurs open-iscsi, cela se trouve dans `/etc/iscsi/initiatorname.iscsi`.

Procédure 24.5. Configurez les ACL

1. Déplacez-vous dans le répertoire acls.

```
/iscsi/iqn.20...mple:444/tpg1> acls/
```

2. Créer un ACL. Vous pouvez soit utiliser le nom d'initiateur qui se trouve dans `/etc/iscsi/initiatorname.iscsi` sur l'initiateur, ou si vous utilisez un nom facile à retenir, voir [Section 24.2, « Créer un initiateur iSCSI »](#) pour vous assurer que l'ACL corresponde à l'initiateur. Ainsi :

```
/iscsi/iqn.20...444/tpg1/acls> create iqn.2006-04.com.example.foo:888
Created Node ACL for iqn.2006-04.com.example.foo:888
Created mapped LUN 2.
Created mapped LUN 1.
Created mapped LUN 0.
```



Note

Le comportement de l'exemple ci-dessus dépend de la configuration qui aura été utilisée. Dans ce cas, le paramètre global `auto_add_mapped_luns` sera utilisé. Cela fait correspondre les LUN aux ACL créés automatiquement.

3. Afficher les changements.

```
/iscsi/iqn.20...444/tpg1/acls> ls
o- acls .....[1 ACL]
  o- iqn.2006-04.com.example.foo:888 ....[3 Mapped LUNs, auth]
    o- mapped_lun0 .....[lun0 ramdisk/ramdisk1 (rw)]
    o- mapped_lun1 .....[lun1 block/block1 (rw)]
    o- mapped_lun2 .....[lun2 fileio/file1 (rw)]
```

24.1.7. Installation de l'une cible Fibre Channel over Ethernet (FCoE)

En plus de monter des LUN sur FCoE comme décrit dans [Section 24.4, « Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\)](#), l'exportation de LUN vers d'autres ordinateurs sur FCoE est également prise en charge avec l'aide de la commande `targetcli`.



Important

Avant de continuer, veuillez consulter [Section 24.4, « Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\) »](#) et vérifier que le paramétrage FCoE de base est bien terminé, et que **fcoeadm -i** affiche les interfaces FCoE configurées.

Procédure 24.6. Configurer un cible FCoE

1. Configurer une cible FCoE requiert l'installation du package **targetcli** et de ses dépendances. Voir [Section 24.1, « Installation de la Cible »](#) pour obtenir plus d'informations sur les bases de **targetcli** et son installation.
2. Créer une instance de cible FCoE sur une interface FCoE.

```
/> tcm_fc/ create 00:11:22:33:44:55:66:77
```

Si des interfaces FCoE sont présentes sur le système, la complétion par la touche de tabulation après la saisie de **create** répertoriera les interfaces disponibles. Si ce n'est pas le cas, assurez-vous que **fcoeadm -i** affiche bien des interfaces actives.

3. Mettre en correspondance d'un « backstore » avec une instance cible.

Exemple 24.1. Exemple de mise en correspondance d'un « backstore » avec l'instance cible.

```
/> tcm_fc/00:11:22:33:44:55:66:77
```

```
/> luns/ create /backstores/fileio/example2
```

4. Autoriser l'accès au LUN à partir d'un initiateur FCoE.

```
/> acls/ create 00:99:88:77:66:55:44:33
```

Le LUN devrait désormais être accessible à cet initiateur.

5. Pour rendre les changements persistants à travers les démarrages, utiliser la commande **saveconfig** et tapez **yes** lorsqu'on vous y invite, sinon la configuration sera perdue au second démarrage.
6. Quittez **targetcli** en saisissant **exit** ou **ctrl+D**.

24.1.8. Supprimer les objets par la commande **targetcli**

Pour supprimer un backstore, utiliser la commande :

```
/> /backstores/backstore-type/backstore-name
```

Pour supprimer certaines parties de la cible iSCSI, comme un ACL, utiliser la commande suivante :

```
/> /iscsi/iqn-name/tpg/acls/ delete iqn-name
```

Pour supprimer la cible dans son ensemble, y compris les ACL, les LUN et les portails, utiliser la commande suivante :

```
/> /iscsi delete iqn-name
```

24.1.9. Références targetcli

Pour plus d'informations sur **targetcli**, référez-vous aux ressources suivantes :

man targetcli

La page man **targetcli** inclut un exemple.

Linux SCSI Target Wiki

<http://linux-iscsi.org/wiki/Targetcli>

Screenecast par Andy Grover

<https://www.youtube.com/watch?v=BkBGtBadOO8>



Note

Chargé le 28 février 2012. Le nom du service a changé. Il est passé de **targetcli** à **target**.

24.2. Créer un initiateur iSCSI

Après avoir créé une cible avec **targetcli** comme dans [Section 24.1, « Installation de la Cible »](#) utiliser **iscsiadm** pour définir un initiateur.



Note

Dans Red Hat Enterprise Linux 7, le service iSCSI service démarre lazily par défaut. C'est à dire que si on lance une commande **iscsiadm**, le service démarrera.

Procédure 24.7. Créer un initiateur iSCSI

1. Installer **iscsi-initiator-utils**.

```
~]$ sudo yum install iscsi-initiator-utils
```

2. Si un nom personnalisé a été donnée à l'ACL dans [Section 24.1.6, « Configurez les ACL »](#), alors, modifiez le fichier **/etc/iscsi/initiatorname.iscsi** pour qu'il corresponde.

```
~]$ sudo vim /etc/iscsi/initiatorname.iscsi
~]$ cat /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.2006-04.com.example.foo
```

3. Découvrir la cible.

```
~]$ sudo iscsiadm -m discovery -t st -p target-ip-address
10.64.24.179:3260,1 iqn.2006-04.com.example:3260
```

4. Connectez-vous à la cible avec le nom iqn qui se trouve dans l'étape précédente.

```
~]$ sudo iscsiadm -m node -T iqn.2006-04.com.example:3260 -l
Logging in to [iface: default, target: iqn.2006-04.com.example:3260,
portal: 10.64.24.179,3260] (multiple)
Login to [iface: default, target: iqn.2006-04.com.example:3260,
portal: 10.64.24.179,3260] successful.
```

Cette procédure peut être suivie par un certain nombre d'initiateurs connectés au même LUN tant que leurs noms d'initiateurs spécifiques sont ajoutés à l'ACL comme décrit dans [Section 24.1.6, « Configurez les ACL »](#).

24.3. Fibre Channel

Cette section discute de l'API Fibre Channel, des pilotes Fibre Channel Red Hat Enterprise Linux 7 natifs, et des capacités Fibre Channel de ces pilotes.

24.3.1. Interface de programmation Fibre Channel

Ci-dessous figure une liste de répertoires `/sys/class/` qui contiennent des fichiers utilisés pour fournir l'API de l'espace utilisateur. Dans chaque élément, les numéros des hôtes sont désignés par **H**, les numéros des bus par **B**, les cibles par **T**, les numéros d'unité logique (LUN) par **L**, et les numéros de ports distants par **R**.



Important

Si votre système utilise un logiciel multipath, Red Hat vous recommande de consulter votre fournisseur de matériel avant de modifier toute valeur décrite dans cette section.

Transport : `/sys/class/fc_transport/targetH:B:T/`

- **port_id** — ID/adresse du port 24 bits
- **node_name** — nom du nœud 64 bits
- **port_name** — nom du port 64 bits

Port distant : `/sys/class/fc_remote_ports/rport-H:B-R/`

- **port_id**
- **node_name**
- **port_name**
- **dev_loss_tmo** — nombre de secondes à attendre avant de marquer un lien comme étant « erroné ». Une fois qu'un lien est marqué comme étant erroné, les E/S exécutées sur le chemin correspondant (ainsi que toute nouvelle E/S sur ce chemin) échoueront.

La valeur par défaut **dev_loss_tmo** varie, en fonction du pilote ou périphérique utilisé. Si un adaptateur Qlogic est utilisé, la valeur par défaut est de 35 secondes, tandis que si un adaptateur Emulex est utilisé, elle sera de 30 secondes. La valeur **dev_loss_tmo** peut être modifiée via le paramètre **dev_loss_tmo** du module **scsi_transport_fc**, même si le pilote peut remplacer cette valeur de délai d'expiration.

La valeur maximum de **dev_loss_tmo** est de 600 secondes. Si **dev_loss_tmo** est paramétré sur zéro ou toute valeur plus importante que 600, alors les délais d'expiration internes du pilote seront utilisés à la place.

- ✦ **fast_io_fail_tmo** — temps à attendre avant de faire échouer les E/S exécutées lorsqu'un problème de lien est détecté. Les E/S qui atteignent le pilote échoueront. Si des E/S se trouvent dans une file d'attente bloquée, elles n'échoueront pas avant que **dev_loss_tmo** n'expire et que la file ne soit débloquée.

Hôte : `/sys/class/fc_host/hostH/`

- ✦ **port_id**
- ✦ **issue_lip** — donne l'instruction de redécouvrir les ports distants au pilote.

24.3.2. Pilotes et capacités natifs Fibre Channel

Red Hat Enterprise Linux 7 est fourni avec les pilotes natifs Fibre Channel suivants :

- ✦ **lpfc**
- ✦ **qla2xxx**
- ✦ **zfcp**
- ✦ **bfa**

[Tableau 24.1, « Fonctionnalités de l'API Fibre Channel »](#) décrit les différentes capacités de l'API Fibre-Channel de chaque pilote natif Red Hat Enterprise Linux 7. X dénote la prise en charge de la fonctionnalité.

Tableau 24.1. Fonctionnalités de l'API Fibre Channel

	lpfc	qla2xxx	zfcp	bfa
Transport port_id	X	X	X	X
Transport node_name	X	X	X	X
Transport port_name	X	X	X	X
Port distant dev_loss_tmo	X	X	X	X
Port distant fast_io_fail_tmo	X	X [a]	X [b]	X
Hôte port_id	X	X	X	X
Hôte issue_lip	X	X		X
[a] Supported as of Red Hat Enterprise Linux 5.4				
[b] Supported as of Red Hat Enterprise Linux 6.0				

24.4. Configurer une interface FCoE (« Fibre-Channel Over Ethernet »)

Le paramétrage et déploiement d'une interface FCoE (« Fibre Channel over Ethernet ») requiert deux paquets :

- » **fcoe-utils**
- » **lldpad**

Une fois ces paquets installés, veuillez effectuer la procédure suivante pour activer FCoE sur un réseau LAN virtuel (VLAN) :

Procédure 24.8. Configurer une interface Ethernet pour utiliser FCoE

1. Configurez un nouveau réseau VLAN, copier un script réseau existante (par exemple, **/etc/fcoe/cfg-eth0**) et changer le nom du périphérique Ethernet qui prend en charge FCoE. Ceci vous fournira un fichier par défaut à configurer. Étant donné que le périphérique FCoE est nommé **ethX**, veuillez exécuter :

```
# cp /etc/fcoe/cfg-eth0 /etc/fcoe/cfg-ethX
```

Modifiez le contenu de **cfg-ethX** comme nécessaire. **DCB_REQUIRED** devrait être défini sur **no** pour les interfaces réseau qui implémentent un client DCBX (Data Center Bridging Exchange) matériel.

2. Si vous souhaitez que le périphérique soit automatiquement chargé pendant le démarrage, définissez **ONBOOT=yes** dans le fichier **/etc/sysconfig/network-scripts/ifcfg-ethX** correspondant. Par exemple, si le périphérique FCoE est nommé eth2, alors veuillez modifier **/etc/sysconfig/network-scripts/ifcfg-eth2** en conséquence.
3. Lancez le démon de pontage du centre de données (**dcbd**) en exécutant :

```
# systemctl start lldpad
```

4. Pour les interfaces réseau qui implémentent un client DCBX matériel, ignorez cette étape.

Pour les interfaces qui requièrent un client DCBX logiciel, veuillez activer le pontage de centre de données sur l'interface Ethernet en exécutant :

```
# dcbtool sc ethX dcb on
```

Puis, activez FCoE sur l'interface Ethernet en exécutant :

```
# dcbtool sc ethX app:fcoe e:1
```

Notez que ces commandes ne fonctionneront uniquement que si les paramètres **dcbd** de l'interface Ethernet n'ont pas été modifiés.

5. Chargez le périphérique FCoE en utilisant :

```
# ifconfig ethX up
```

6. Lancez FCoE en utilisant :

```
# systemctl start fcoe
```

Le périphérique FCoE apparaîtra sous peu, en supposant que tous les autres paramètres de la structure soient corrects. Pour afficher les périphériques FCoE configurés, veuillez exécuter :

```
# fcoeadm -i
```

Après avoir correctement configuré l'interface Ethernet pour utiliser FCoE, Red Hat recommande de paramétrer FCoE et **lldpad** pour s'exécuter lors du démarrage. Pour ce faire, veuillez utiliser **chkconfig**, comme dans :

```
# systemctl enable lldpad
```

```
# systemctl enable fcoe
```



Note

Exécuter la commande `# systemctl stop fcoe` stoppe le démon, mais ne réinitialise pas la configuration des interfaces FCoE. Pour cela, exécuter la commande `# systemctl -s SIGHUP kill fcoe`.

À partir de Red Hat Enterprise Linux 7, le gestionnaire de réseau a la possibilité d'interroger ou de définir les paramètres de configuration DCB d'une interface Ethernet compatible DCB.

24.5. Configurer une interface FCoE pour qu'elle soit automatiquement montée lors du démarrage



Note

Les instructions dans cette section sont disponibles dans `/usr/share/doc/fcoe-utils-version/README` à partir de Red Hat Enterprise Linux 6.1. Veuillez consulter ce document en cas de changement lors des sorties de versions mineures.

Vous pouvez monter les nouveaux disques découverts via les règles **udev**, **autofs**, et autres méthodes similaires. Cependant, de temps à autres un service particulier peut nécessiter que le disque FCoE soit monté pendant le démarrage. Dans de tels cas, le disque FCoE devrait être monté *dès que* le service **fcoe** est exécuté et *avant* l'initialisation de tout service qui requiert le disque FCoE.

Pour configurer un disque FCoE afin qu'il soit automatiquement monté pendant le démarrage, ajoutez un code de montage FCoE correct au script de démarrage du service **fcoe**. Le script de démarrage **fcoe** se trouve ici : `/etc/init.d/fcoe`.

Le code de montage FCoE est différent selon la configuration du système, que vous utilisiez un simple disque formaté FCoE, LVM, ou un nœud de périphérique à multiples chemins d'accès.

Exemple 24.2. Code de montage FCoE

Ci-dessous figure un exemple de code de montage FCoE pour monter les systèmes de fichiers spécifiés

avec des caractères génériques dans **/etc/fstab** :

```
mount_fcoe_disks_from_fstab()
{
    local timeout=20
    local done=1
    local fcoe_disks=$(egrep 'by-path\/*fc-.*_netdev' /etc/fstab | cut -d
' ' -f1))

    test -z $fcoe_disks && return 0

    echo -n "Waiting for fcoe disks . "
    while [ $timeout -gt 0 ]; do
for disk in ${fcoe_disks[*]}; do
    if ! test -b $disk; then
        done=0
        break
    fi
done

    test $done -eq 1 && break;
    sleep 1
    echo -n ". "
    done=1
    let timeout--
    done

    if test $timeout -eq 0; then
echo "timeout!"
    else
echo "done!"
    fi

    # mount any newly discovered disk
    mount -a 2>/dev/null
}

```

La fonction **mount_fcoe_disks_from_fstab** doit être invoquée *après* que le script du service **fcoe** ait lancé le démon **fcoemon**. Ceci montera les disques FCoE spécifiés par les chemins suivants dans **/etc/fstab** :

```
/dev/disk/by-path/fc-0xXX:0xXX /mnt/fcoe-disk1 ext3 defaults,_netdev 0 0
/dev/disk/by-path/fc-0xYY:0xYY /mnt/fcoe-disk2 ext3 defaults,_netdev 0 0
```

Les entrées avec les sous-chaînes **fc-** et **_netdev** activent la fonction **mount_fcoe_disks_from_fstab** pour identifier les entrées de montage de disques FCoE. Pour obtenir davantage d'informations sur les entrées **/etc/fstab**, veuillez consulter **man 5 fstab**.



Note

Le service **fcoe** n'implémente pas de délai d'expiration pour la récupération de disque FCoE. Ainsi, le code de montage FCoE doit implémenter son propre délai d'expiration.

24.6. iSCSI

Cette section décrit l'interface de programmation iSCSI et l'utilitaire **iscsiadm**. Avant d'utiliser l'utilitaire **iscsiadm**, veuillez commencer par installer le paquet **iscsi-initiator-utils** en exécutant **yum install iscsi-initiator-utils**.

Sur Red Hat Enterprise Linux 7, le service iSCSI est lancé par défaut. Si la partition root ne se trouve pas sur un périphérique iSCSI ou s'il n'y a pas de nœud marqué de **node.startup = automatic**, alors le service iSCSI ne sera pas lancé tant que la commande **iscsiadm** n'est pas exécutée, celle-ci requiert le démarrage du module du noyau **iscsid** ou **iscsi**. Par exemple, exécuter la commande de découverte **iscsiadm -m discovery -t st -p ip:port** provoquera le lancement du service iSCSI par **iscsiadmin**.

Pour forcer l'exécution du démon **iscsid** et le chargement des modules du noyau iSCSI, veuillez exécuter **service iscsid force-start**.

24.6.1. API iSCSI

Pour obtenir des informations sur les sessions en cours d'exécution, veuillez exécuter :

```
# iscsiadm -m session -P 3
```

Cette commande affiche l'état de la session ou du périphérique, l'ID de session (**sid**), certains paramètres négociés, ainsi que les périphériques SCSI accessibles à travers la session.

Pour une sortie plus courte (par exemple pour afficher la cartographie **sid/nœud**), veuillez exécuter :

```
# iscsiadm -m session -P 0
```

ou

```
# iscsiadm -m session
```

Ces commandes impriment la liste des sessions en cours d'exécution avec le format :

```
driver [sid] target_ip:port,target_portal_group_tag proper_target_name
```

Exemple 24.3. Sortie de la commande **iscsiadm -m session**

Par exemple:

```
# iscsiadm -m session

tcp [2] 10.15.84.19:3260,2 iqn.1992-08.com.netapp:sn.33615311
tcp [3] 10.15.85.19:3260,3 iqn.1992-08.com.netapp:sn.33615311
```

Pour obtenir davantage d'informations sur l'API iSCSI, veuillez consulter **/usr/share/doc/iscsi-initiator-utils-version/README**.

24.7. Dénomination persistante

Red Hat Enterprise Linux fournit un certain nombre de façons d'identifier les périphériques de stockage. Il est important d'utiliser la bonne option pour identifier chaque périphérique lorsqu'il est utilisé afin d'éviter tout accès par inadvertance du mauvais périphérique, en particulier lorsque vous installez ou reformez des disques.

24.7.1. /dev/sd* et leurs nombres Majeurs et Mineurs

Les périphériques de stockage gérés par le pilote **sd** sont identifiés en interne par une collection de numéros de périphériques principaux et leurs numéros mineurs associés. Les numéros majeurs utilisés à cette fin ne sont pas dans une plage contiguë. Chaque périphérique de stockage est représenté par un numéro principal et un éventail de numéros mineurs, qui servent à identifier le dispositif entier ou une partition au sein de l'appareil. Il y a une association directe entre les numéros majeurs et mineurs attribués à un périphérique et les nombres sous la forme de **sd< lettre(s)>< numéro(s) en option>**. Chaque fois que le pilote **sd** détecte un nouveau périphérique, une gamme de numéros mineurs et numéros majeurs est allouée. Chaque fois qu'un périphérique est supprimé du système d'exploitation, la gamme de numéros mineurs et majeurs est libérée afin d'être réutilisés plus tard.

La plage de numéros majeurs et mineurs et noms associés de **sd** sont attribués pour chaque périphérique lorsqu'il est détecté. Cela signifie que l'association entre la plage de numéros majeurs et mineurs et les noms associés de **sd** peut changer si l'ordre de détection des périphériques change. Bien que cela soit inhabituel avec certaines configurations matérielles (par exemple, avec les disques ayant leur ID de cible SCSI assignée par leur emplacement physique dans le châssis), cela peut néanmoins se produire. Voici des exemples de situations où cela peut se produire :

- ✦ Un disque peut ne pas s'allumer ou répondre au contrôleur SCSI. Ainsi, il ne sera pas détecté par la sonde classique du périphérique. Le disque ne sera pas accessible au système et les périphériques ultérieurs auront leur plage de numéros majeurs et mineurs, y compris les noms de **sd** associés déplacés vers le bas. Par exemple, si un disque normalement dénommé **sdb** n'est pas détecté, un disque normalement appelé **sdc** apparaît comme **sdb**.
- ✦ Un contrôleur SCSI (adaptateur de bus hôte ou HBA) peut ne pas s'initialiser, ce qui fait que tous les disques connectés à cet adaptateur HBA ne sont pas détectés. Tous les disques connectés aux HBA sondés par la suite se verront attribuer différentes gammes de numéros majeurs et mineurs, et différents noms **sd** associés.
- ✦ L'ordre d'initialisation peut changer si différents types de HBA sont présents dans le système. Cela entraînera une détection des disques connectés à ces HBA dans un ordre différent. Cela a lieu également si les HBA vont dans différents slots PCI du système.
- ✦ Les disques connectés au système avec Fibre Channel, iSCSI, ou avec des adaptateurs FCoE pourraient être inaccessibles au moment où les périphériques de stockage sont détectés, en raison d'une baie de stockage ou d'un interrupteur qui passe hors tension, par exemple. Cela pourrait se produire lorsqu'un système démarre à nouveau après une panne de courant, si la matrice de stockage prend plus de temps à se mettre en ligne qu'il ne faut pour le reboot du système. Bien que certains pilotes Fibre Channel prennent en charge un mécanisme qui sert à spécifier un ID de cible SCSI persistant au mappage WWPN, cela n'entraînera pas les gammes majeures et mineures de numéros, ainsi que les noms de **sd** associés à être réservés, cela ne fournira que des numéros d'ID de cibles SCSI compatibles.

Ces raisons font qu'il n'est pas souhaitable d'utiliser les gammes de nombres mineures ou majeures des noms **sd** associés quand on se réfère aux périphériques, comme dans le fichier **/etc/fstab**. Il est possible que le mauvais périphérique soit monté et qu'une corruption de données en résulte.

Parfois, il faut cependant se référer aux noms de **sd** même si un autre mécanisme est utilisé (comme lorsque des erreurs sont reportées par un périphérique). C'est parce que le noyau Linux utilise les noms **sd** (et les tuples SCSI host/channel/target/LUN) dans les messages du noyau à propos du périphérique.

24.1.2. WWID

Les identifiants WWID (*World Wide Identifier*) peuvent être utilisés avec des périphériques s'identifiant de manière fiable. Il s'agit d'un ID persistant, non dépendant des systèmes requis par le standard SCSI de tous les périphériques SCSI. Les identifiants WWID offrent la garantie d'être uniques pour chaque périphérique de stockage, et sont indépendants du chemin utilisé pour accéder aux périphériques.

Cet identifiant peut être obtenu en passant une recherche SCSI pour récupérer les *données vitales de produit d'identification de périphérique* (page **0x83**) ou le *numéro de série de l'unité* (page **0x80**). Les mappages de ces WWID aux noms `/dev/sd` actuels peuvent être observés sur les liens symboliques maintenus dans le répertoire `/dev/disk/by-id/`.

Exemple 24.4. WWID

Par exemple, un périphérique avec l'identifiant de page **0x83** aurait :

```
scsi-3600508b400105e210000900000490000 -> ../../sda
```

Ou un périphérique avec l'identifiant de page **0x80** aurait :

```
scsi-SSEAGATE_ST373453LW_3HW1RHM6 -> ../../sda
```

Red Hat Enterprise Linux maintient automatiquement le mappage correct à partir du nom d'un périphérique basé sur WWID mappé au nom `/dev/sd` sur ce système. Les applications peuvent utiliser le nom `/dev/disk/by-id/` pour référencer les données sur le disque, même si le chemin vers le périphérique change, et même pendant l'accès à ce périphérique par différents systèmes.

S'il existe de multiples chemins allant d'un système à un périphérique, **device-mapper-multipath** utilisera l'ID global WWID pour les détecter. **Device-mapper-multipath** présentera ensuite un seul « pseudo-périphérique » dans `/dev/mapper/wwid`, tel que `/dev/mapper/3600508b400105df70000e00000ac0000`.

La commande **multipath -l** affiche le mappage vers les identificateurs non-persistants :

Hôte:Canal:Cible:LUN, le nom `/dev/sd`, et le numéro **major:minor**.

```
3600508b400105df70000e00000ac0000 dm-2 vendor,product
[size=20G][features=1 queue_if_no_path][hwandler=0][rw]
\_ round-robin 0 [prio=0][active]
  \_ 5:0:1:1 sdc 8:32 [active][undef]
  \_ 6:0:1:1 sdg 8:96 [active][undef]
\_ round-robin 0 [prio=0][enabled]
  \_ 5:0:0:1 sdb 8:16 [active][undef]
  \_ 6:0:0:1 sdf 8:80 [active][undef]
```

Device-mapper-multipath maintient automatiquement le mappage correct de chaque nom de périphérique basé sur WWID avec le nom `/dev/sd` correspondant sur le système. Ces noms sont persistants à travers les changements de chemin, et sont cohérents lors de l'accès au périphérique à partir de différents systèmes.

Lorsque la fonctionnalité **user_friendly_names** (de **device-mapper-multipath**) est utilisée, le WWID est mappé à un nom sous le format `/dev/mapper/mpathn`. Par défaut, ce mappage est maintenu dans le fichier `/etc/multipath/bindings`. Ces noms **mpathn** sont persistants tant que le fichier est maintenu.



Important

Si vous utilisez **user_friendly_names**, alors des étapes supplémentaires sont requises pour obtenir des noms cohérents dans un cluster. Veuillez consulter les « Noms cohérents de périphériques multivoies » dans la section « Cluster » de l'ouvrage *Utilisation de l'administration et de la configuration DM Multipath*.

En plus des noms persistants fournis par le système, vous pouvez également utiliser les règles **udev** pour implémenter des noms persistants de votre choix, mappés au WWID du stockage.

24.7.3. Noms de périphériques gérés par le mécanisme udev (/dev/disk/by-*)

Le mécanisme **udev** est constitué de trois composants principaux :

Le noyau

Génère des événements qui sont envoyés dans l'espace utilisateur quand les périphériques sont ajoutés, supprimés ou modifiés.

Le démon udevd

Réceptionne les événements.

Les règles udev

Indiquent les actions à prendre quand le démon **udev** reçoit des événements de noyau.

Ce mécanisme est utilisé pour tous les types de périphériques de Linux, et non pas uniquement les périphériques de stockage. Dans le cas de périphériques de stockage, Red Hat Enterprise Linux contient les règles **udev** qui créent des liens symboliques dans le répertoire **/dev/disk/** permettant ainsi à des périphériques de stockage d'être référencés par leur contenu, un identifiant unique, un numéro de série ou le chemin d'accès du matériel utilisé pour accéder au périphérique.

/dev/disk/by-label

Les entrées à ce répertoire fournissent un nom symbolique qui se réfère au périphérique de stockage par un libellé dans le contenu (c-a-d les données) stocké dans ce périphérique. Le programme **blkid** est utilisé pour lire les données du périphérique et déterminer un nom (c-a-d un libellé) de périphérique. Exemple :

```
/dev/disk/by-label/Boot
```



Note

Les informations se trouvent dans le contenu (c-a-d les données) du périphérique, donc si le contenu est copié à partir d'un autre périphérique, le libellé sera le même.

Le libellé peut aussi être utilisé pour se référer au périphérique qui se trouve dans **/etc/fstab** en utilisant la syntaxe suivante :

```
LABEL=Boot
```

/dev/disk/by-uuid

Les entrées à ce répertoire fournissent un nom symbolique qui se réfère au périphérique de stockage par un identifiant unique dans le contenu (c-a-d les données) stocké dans le périphérique. Le programme **blkid** est utilisé pour lire les données du périphérique et déterminer un identifiant unique (c-a-d un uuid) de périphérique. Exemple :

```
UUID=3e6be9de-8139-11d1-9106-a43f08d823a6
```

/dev/disk/by-id

Les entrées à ce répertoire fournissent un nom symbolique qui se réfère au périphérique de stockage par un identifiant unique (différent de tous les autres périphériques de stockage). L'identifiant est une propriété du périphérique mais il n'est pas stocké dans le contenu (c-a-d les données) des périphériques. Exemple :

```
/dev/disk/by-id/scsi-3600508e00000000ce506dc50ab0ad05
```

```
/dev/disk/by-id/wwn-0x600508e000000000ce506dc50ab0ad05
```

L'id s'obtient à partir de l'ID word-wide du périphérique, ou du numéro de série du périphérique. Les entrées **/dev/disk/by-id** peuvent également inclure un numéro de partition. Exemple :

```
/dev/disk/by-id/scsi-3600508e00000000ce506dc50ab0ad05-part1
```

```
/dev/disk/by-id/wwn-0x600508e000000000ce506dc50ab0ad05-part1
```

/dev/disk/by-path

Les entrées de ce répertoire procurent un nom symbolique qui désigne le périphérique de stockage par le chemin d'accès au matériel utilisé pour accéder au périphérique, commençant par une référence au contrôleur de stockage dans la hiérarchie PCI et qui inclut l'hôte SCSI, le canal, la cible, et les numéros LUN et, parfois, le nombre de partitions. Bien que ces noms soient préférable à l'utilisation des numéros majeurs et mineurs ou des noms de **sd**, vous devez veiller à ce que les chiffres cibles ne changent pas dans un environnement SAN Fibre Channel (par exemple, par le biais de liaisons permanentes) et que l'utilisation des noms soit mise à jour si un adaptateur d'hôte est déplacé vers un autre emplacement PCI. En outre, il est possible que le nombre d'hôtes SCSI change si un HBA ne parvient pas à interroger, si les pilotes sont chargés dans un ordre différent, ou si un nouvel HBA est installé sur le système. Exemple de listing de by-path :

```
/dev/disk/by-path/pci-0000:03:00.0-scsi-0:1:0:0
```

Les entrées **/dev/disk/by-path** peuvent également inclure un numéro de partition, comme :

```
/dev/disk/by-path/pci-0000:03:00.0-scsi-0:1:0:0-part1
```

24.7.3.1. Limitations de la convention de nommage de périphérique d'udev

Voici certaines des limitations de convention de nommage d' **udev**.

- ✱ Il est possible que le périphérique ne soit pas accessible au moment de la requête car le mécanisme d'**udev** se repose sans doute sur la capacité d'interroger le périphérique de stockage quand les règles d'

udev sont analysées à l'occasion d'un événement **udev**. Cela a une meilleure chance de se produire avec Fibre Channel, iSCSI ou les périphériques de stockage FCoE quand le périphérique n'est pas situé sur le châssis du serveur.

- ✦ Le noyau envoie également des événements **udev** à tout moment, donc les règles sont analysées et cela entraîne la suppression des liens **/dev/disk/by-*** quand le périphérique n'est plus accessible.
- ✦ Un laps de temps peut s'écouler entre le moment où l'événement **udev** est généré et le moment où il est traité (comme quand on détecte un grand nombre de périphériques et que le démon **udev** d'espace utilisateur prend un bon moment à analyser les règles pour chacun). Ceci peut entraîner un délai entre le moment où le noyau détecte le périphérique et le moment où les noms **/dev/disk/by-*** sont rendus disponibles.
- ✦ Des programmes externes comme **blkid** qui sont invoqués par les règles peuvent ouvrir le périphérique pendant un court moment, rendant ainsi le périphérique inaccessible à tout autre utilisation.

24.8. Supprimer un périphérique de stockage

Avant de supprimer l'accès au périphérique de stockage même, il est conseillé de commencer par effectuer une copie de sauvegarde des données du périphérique. Ensuite, videz les E/S et supprimez toutes les références au périphérique du système d'exploitation (comme décrit ci-dessous). Si le périphérique utilise une fonction multivoies, veuillez effectuer ceci pour le « pseudo-périphérique » ([Section 24.7.2, « WWID »](#)) et pour chaque identificateur qui représente un chemin vers le périphérique. Si vous supprimez uniquement un chemin vers un périphérique multivoies, et que tous les autres chemins restent, alors la procédure est plus simple, comme décrit dans [Section 24.10, « Ajouter un périphérique ou un chemin de stockage »](#).

La suppression d'un périphérique de stockage n'est pas recommandée lorsque le système se trouve sous pression mémoire car le vidage des E/S s'ajoutera à la charge. Pour déterminer le niveau de pression mémoire, exécutez la commande **vmstat 1 100** ; la suppression du périphérique n'est pas recommandée si :

- ✦ La mémoire disponible fait moins de 5% de la totalité de la mémoire dans plus de 10 échantillons pour 100 (la commande **free** peut également être utilisée pour afficher la mémoire totale).
- ✦ La fonction swap est active (colonnes **si** et **so** non égales à zéro dans la sortie **vmstat**).

La procédure générale pour supprimer tout accès à un périphérique est comme suit :

Procédure 24.9. S'assurer d'une suppression de périphérique normale

1. Fermez tous les utilisateurs du périphérique et effectuez une copie de sauvegarde des données du périphérique selon les besoins.
2. Veuillez utiliser **umount** pour démonter tout système de fichiers qui aurait monté le périphérique.
3. Supprimez le périphérique de tout volume **md** et LVM qui l'utilise. Si le périphérique est un membre d'un groupe de volumes LVM, alors il faudra sans doute déplacer les données hors du périphérique en utilisant la commande **pvmove**, puis utiliser la commande **vgreduce** pour supprimer le volume physique, et (optionnellement) **pvremove** pour supprimer les métadonnées LVM du disque.
4. Si le périphérique utilise la fonction multivoies, veuillez exécuter **multipath -l** et prendre note de tous les chemins menant au périphérique. Ensuite, veuillez supprimer le périphérique multivoies en utilisant **multipath -f device**.

5. Veuillez exécuter `blockdev --flushbufs device` pour vider toute E/S restante sur tous les chemins vers le périphérique. Ceci est particulièrement important pour les périphériques bruts, lorsqu'il n'y a pas d'opération `umount` ou `vgreduce` pour causer un vidage d'E/S.
6. Veuillez supprimer toute référence au nom basée sur le chemin du périphérique, comme `/dev/sd`, `/dev/disk/by-path` ou sur le numéro `major:minor`, dans les applications, scripts, ou utilitaires du système. Ceci est important pour s'assurer que les différents périphériques ajoutés dans le futur ne soient pas confondus avec le périphérique actuel.
7. Finalement, veuillez supprimer chaque chemin vers le périphérique à partir du sous-système SCSI. Pour ce faire, veuillez utiliser la commande `echo 1 > /sys/block/device-name/device/delete`, où `device-name` pourrait, par exemple, être `sde`.

Une autre variation de cette opération est `echo 1 >`

`/sys/class/scsi_device/h:c:t:l/device/delete`, où `h` est le numéro HBA, `c` est le canal sur le HBA, `t` est l'ID de la cible SCSI, et `l` est le LUN.



Note

L'ancienne forme de ces commandes, `echo "scsi remove-single-device 0 0 0 0" > /proc/scsi/scsi`, est déconseillée.

Vous pouvez déterminer le *nom-du-périphérique*, le numéro du HBA, canal du HBA, l'ID et le LUN de la cible SCSI d'un périphérique à l'aide de diverses commandes, telles que `lsscsi`, `scsi_id`, `multipath -l`, et `ls -l /dev/disk/by-*`.

Après avoir effectué une opération [Procédure 24.9, « S'assurer d'une suppression de périphérique normale »](#), un périphérique peut être supprimé en toute sécurité d'un système en cours d'exécution. Il n'est pas nécessaire d'arrêter les E/S des autres périphériques pour ceci.

D'autres procédures, comme la suppression physique de périphériques, suivies d'un nouveau scan du bus SCSI (comme décrit dans [Section 24.11, « Scanner les interconnexions du stockage »](#)) pour que l'état du système d'exploitation soit mis à jour et reflète les changements, ne sont pas recommandées. Celles-ci provoquent des délais, dus à l'expiration des E/S, et des périphériques peuvent être supprimés de manière inattendue. S'il est nécessaire d'effectuer un nouveau scan d'une interconnexion, il devra être fait lorsque les E/S sont sur pause, comme décrit dans [Section 24.11, « Scanner les interconnexions du stockage »](#).

24.9. Supprimer un chemin vers un périphérique de stockage

Si vous supprimez un chemin vers un périphérique multivoies (sans affecter les autres chemins vers le périphérique), alors la procédure générale est comme suit :

Procédure 24.10. Supprimer un chemin vers un périphérique de stockage

1. Veuillez supprimer toute référence au nom basée sur le chemin du périphérique, comme `/dev/sd`, `/dev/disk/by-path` ou sur le numéro `major:minor`, dans les applications, scripts, ou utilitaires du système. Ceci est important pour s'assurer que les différents périphériques ajoutés dans le futur ne soient pas confondus avec le périphérique actuel.
2. Mettez le chemin hors-ligne en utilisant `echo offline > /sys/block/sda/device/state`.

Ceci provoquera l'échec immédiat de toute E/S ultérieure envoyée sur ce chemin. `Device-mapper-multipath` continuera d'utiliser les chemins restants vers le périphérique.

3. Veuillez supprimer le chemin du sous-système SCSI. Pour ce faire, veuillez utiliser la commande `echo 1 > /sys/block/device-name/device/delete`, où *device-name* pourrait, par exemple, être `sde` (comme décrit dans [Procédure 24.9, « S'assurer d'une suppression de périphérique normale »](#)).

Après avoir effectué une opération [Procédure 24.10, « Supprimer un chemin vers un périphérique de stockage »](#), le chemin peut être supprimé en toute sécurité du système en cours d'exécution. Il n'est pas nécessaire d'arrêter les E/S, car `device-mapper-multipath` routera à nouveau les E/S vers les chemins restants en fonction de la configuration du groupement de chemins et des politiques de basculement.

D'autres procédures, comme la suppression physique du câble, suivies d'un nouveau scan du bus SCSI pour que l'état du système d'exploitation soit mis à jour et reflète les changements, ne sont pas recommandées. Celles-ci provoquent des délais, dûs à l'expiration des E/S, et des périphériques peuvent être supprimés de manière inattendue. S'il est nécessaire d'effectuer un nouveau scan d'une interconnexion, celui-ci doit être fait lorsque les E/S sont sur pause, comme décrit dans [Section 24.11, « Scanner les interconnexions du stockage »](#).

24.10. Ajouter un périphérique ou un chemin de stockage

Lors de l'ajout d'un périphérique, n'oubliez pas que le nom du périphérique basé chemin (par exemple, le nom de `/dev/sd`, le numéro de `major:minor` et le nom de `/dev/disk/by-path`) assigné par le système au nouveau périphérique peut déjà avoir été utilisé par un périphérique qui a depuis été supprimé. Ainsi, veuillez vous assurer que toutes les anciennes références au nom du périphérique basé chemin ont bien été supprimées. Autrement, le nouveau périphérique pourrait malencontreusement passer pour l'ancien périphérique.

Procédure 24.11. Ajouter un périphérique ou un chemin de stockage

1. La première étape de l'ajout d'un périphérique ou d'un chemin de stockage consiste à physiquement activer l'accès au nouveau périphérique de stockage ou à physiquement activer un nouveau chemin vers un périphérique existant. Ceci peut être effectué à l'aide de commandes appartenant aux vendeurs sur le serveur de stockage iSCSI ou Fibre Channel. Une fois effectué, prenez note de la valeur LUN du nouveau stockage qui sera présenté à votre hôte. Si le serveur de stockage est Fibre Channel, veuillez aussi prendre note du WWNN (*World Wide Node Name*) du serveur de stockage et déterminez s'il y a un WWNN unique pour tous les ports sur le serveur de stockage. Si ce n'est pas le cas, veuillez noter le WWPN (*World Wide Port Name*) de chaque port qui sera utilisé pour accéder au nouveau LUN.
2. Veuillez ensuite indiquer le nouveau périphérique de stockage au système d'exploitation, ou le nouveau chemin à un périphérique existant. La commande recommandée est la suivante :

```
$ echo "c t l" > /sys/class/scsi_host/hosth/scan
```

Dans la commande précédente, *h* est le numéro HBA, *c* est le canal sur le HBA, *t* est l'ID de la cible SCSI et *l* est le LUN.



Note

La plus ancienne forme de cette commande, `echo "scsi add-single-device 0 0 0 0" > /proc/scsi/scsi`, est dépréciée.

- a. Sur certains matériaux Fibre Channel, un nouveau LUN créé sur la matrice RAID pourrait ne pas être visible par le système d'exploitation tant qu'une opération LIP (*Loop Initialization Protocol*) n'est pas effectuée. Reportez-vous à [Section 24.11, « Scanner les interconnexions du stockage »](#) pour obtenir des instructions sur la manière d'effectuer cela.



Important

Il faudra arrêter les E/S tant que cette opération est exécutée, si un LIP est requis.

- b. Si un nouveau LUN a été ajouté à la matrice RAID mais qu'il n'est toujours pas configuré par le système d'exploitation, veuillez confirmer la liste des LUN qui sont exportés par la matrice à l'aide de la commande **sg_luns**, faisant partie du paquet *sg3_utils*. Ceci délivrera la commande **SCSI REPORT LUNS** à la matrice RAID et retournera une liste des LUN présents.

Pour les serveurs de stockage Fibre Channel qui implémentent un WWNN unique pour tous les ports, vous pouvez déterminer les valeurs **h**, **c** et **t** correctes (c'est-à-dire le numéro HBA, le canal HBA et l'ID de la cible SCSI) en recherchant le WWNN dans **sysfs**.

Exemple 24.5. Déterminer les valeurs h, c et t correctes

Par exemple, si le WWNN du serveur de stockage est **0x5006016090203181** veuillez utiliser :

```
$ grep 5006016090203181 /sys/class/fc_transport/*/node_name
```

La sortie résultante devrait être similaire à ceci :

```
/sys/class/fc_transport/target5:0:2/node_name:0x5006016090203181
/sys/class/fc_transport/target5:0:3/node_name:0x5006016090203181
/sys/class/fc_transport/target6:0:2/node_name:0x5006016090203181
/sys/class/fc_transport/target6:0:3/node_name:0x5006016090203181
```

Ceci indique qu'il y a quatre routes Fibre Channel vers cette cible (deux HBA à canal unique, menant chacune à deux ports de stockage). En supposant qu'une valeur LUN est **56**, alors la commande suivante configurera le premier chemin :

```
$ echo "0 2 56" > /sys/class/scsi_host/host5/scan
```

Ceci doit être effectué pour chaque chemin vers le nouveau périphérique.

Pour les serveurs de stockage Fibre Channel qui n'implémentent pas un WWNN unique pour tous les ports, vous pouvez déterminer le numéro HBA, le canal HBA et l'ID de la cible SCSI qui conviennent en recherchant chaque WWNN dans **sysfs**.

Il est aussi possible de déterminer la numéro HBA, le canal HBA et l'ID de la cible SCSI en vous référant à un autre périphérique déjà configuré sur le même chemin que le nouveau périphérique. Ceci peut être accompli à l'aide de diverses commandes, comme **lsscsi**, **scsi_id**, **multipath -l** et **ls -l /dev/disk/by-***. Ces informations, en plus du numéro LUN du nouveau périphérique, peuvent être utilisés, comme expliqué ci-dessus, afin d'analyser et de configurer ce chemin vers le nouveau périphérique.

- Après avoir ajouté tous les chemins SCSI au périphérique, veuillez exécuter la commande **multipath** et vérifier que le périphérique a été configuré correctement. À ce moment, le périphérique peut, par exemple, être ajouté à **md**, **LVM**, **mkfs**, ou à **mount**.

Si les étapes ci-dessus sont suivies, alors un périphérique peut être ajouté à un système en cours d'exécution en toute sécurité. Il n'est pas nécessaire d'arrêter les E/S des autres périphériques pendant que cela est effectué. D'autres procédures impliquant un nouveau scan (ou une réinitialisation) du bus SCSI, ce qui amène le système d'exploitation de mettre à jour son état pour refléter la connectivité actuelle du périphérique, ne sont pas recommandées tant que les E/S de stockage sont en cours.

24.11. Scanner les interconnexions du stockage

Plusieurs commandes vous permettront de réinitialiser, de scanner (ou les deux à la fois) une ou plusieurs interconnexion(s), ajoutant ou supprimant potentiellement plusieurs périphériques en une seule opération. Ce type de scan peut être perturbateur car il peut provoquer des délais alors que les opérations d'E/S expirent et peut supprimer des périphériques de manière inattendue. Ainsi, Red Hat recommande d'utiliser le scan d'interconnexions *uniquement lorsque nécessaire*. Les restrictions suivantes doivent être observées lorsque les interconnexions de stockage sont scannées :

- Toutes les E/S sur interconnexions affectées doivent être mises sur pause et vidées avant d'exécuter la procédure, et les résultats du scan doivent être vérifiés avant de reprendre les E/S.
- Comme pour la suppression de périphériques, scanner des interconnexions n'est pas recommandé lorsque le système se trouve sous pression mémoire. Pour déterminer le niveau de pression mémoire, exécutez la commande **vmstat 1 100**. Scanner les interconnexions n'est pas recommandé si la quantité de mémoire disponible fait moins de 5% de la mémoire totale dans plus de 10 échantillons pour 100. Aussi, le scan d'interconnexions n'est pas recommandé si la fonction swap est active (avec des colonnes **si** et **so** qui ne sont pas égales à zéro dans la sortie **vmstat**). La commande **free** peut également afficher la mémoire totale.

Les commandes suivantes peuvent être utilisées pour scanner les interconnexions de stockage :

```
echo "1" > /sys/class/fc_host/host/issue_lip
```

Cette opération effectue un *Loop Initialization Protocol (LIP)*, analyse l'interconnexion et met à jour la couche SCSI pour refléter les périphériques actuellement présents sur le bus. Essentiellement, un LIP est une réinitialisation du bus et entraîne l'ajout ou la suppression de périphériques. Cette procédure est nécessaire pour configurer une nouvelle cible SCSI sur un réseau d'interconnexion Fibre Channel.

Notez que **issue_lip** est une opération asynchrone. La commande peut finaliser son but avant la fin du scan. Vous pourrez vérifier dans le fichier **/var/log/messages** si la commande **issue_lip** est terminée.

Les pilotes **lpfc**, **qla2xxx**, et **bnx2fc** supportent **issue_lip**. Pour plus d'informations sur les fonctions des API supportées par chaque pilote dans Red Hat Enterprise Linux, consulter [Tableau 24.1, « Fonctionnalités de l'API Fibre Channel »](#).

```
/usr/bin/rescan-scsi-bus.sh
```

Le script **/usr/bin/rescan-scsi-bus.sh** a été introduit dans Red Hat Enterprise Linux 5.4. Par défaut, ce script scanne tous les bus SCSI du système, et met à jour la couche SCSI pour refléter la présence de nouveaux périphériques sur le bus. Le script fournit des options supplémentaires pour permettre la suppression de périphériques, et le lancement des LIP. Pour obtenir plus d'informations sur ce script, y compris les problèmes connus, consulter [Section 24.17, « Ajouter ou supprimer une unité logique avec rescan-scsi-bus.sh »](#).

```
echo "- - -" > /sys/class/scsi_host/hosth/scan
```

C'est la même commande, comme décrit dans [Section 24.10. « Ajouter un périphérique ou un chemin de stockage »](#), qui sert à ajouter un périphérique de stockage ou le chemin d'accès. Dans ce cas, toutefois, le numéro du canal, l'ID cible SCSI, et des valeurs LUN sont remplacés par des caractères génériques. Toute combinaison de caractères génériques et d'identificateurs est autorisée, donc vous pouvez rendre la commande aussi précise qu'englobante selon les besoins. Cette procédure ajoute des LUN, mais ne les supprime pas.

```
modprobe --remove driver-name, modprobe driver-name
```

En exécutant la commande `modprobe --remove driver-name` suivie de la commande `modprobe driver-name`, vous ré-initialisez totalement l'état de toutes les interconnexions contrôlées par le pilote. Bien qu'il s'agisse d'une mesure relativement extrême, utiliser les commandes décrites peut être appropriée dans certaines situations. Les commandes peuvent être utilisées, par exemple, pour redémarrer le pilote avec une valeur de paramètre de module différente.

24.12. Configuration de la découverte iSCSI

Le fichier de configuration iSCSI par défaut est `/etc/iscsi/iscsid.conf`. Ce fichier contient les paramètres iSCSI utilisés par `iscsid` et `iscsiadm`.

Pendant la découverte de cibles, l'outil `iscsiadm` utilise les paramètres du fichier `/etc/iscsi/iscsid.conf` pour créer deux types d'enregistrements :

Les enregistrements de nœuds dans `/var/lib/iscsi/nodes`

Lors de la connexion à une cible, `iscsiadm` utilise les paramètres dans ce fichier.

Les enregistrements de découverte dans `/var/lib/iscsi/discovery_type`

Lorsque la découverte est appliquée sur la même destination, `iscsiadm` utilise les paramètres dans ce fichier.

Avant d'utiliser différents paramètres pour la découverte, veuillez tout d'abord supprimer les enregistrements de découverte actuels (c'est-à-dire le fichier `/var/lib/iscsi/discovery_type`). Pour cela, utilisez la commande suivante :

```
# iscsiadm -m discovery -t discovery_type -p target_IP:port -o delete [5]
```

Ici, la valeur de `discovery_type` peut être `sendtargets`, `isns`, ou `fw`.

Pour obtenir des détails sur les différents types de découverte, veuillez consulter la section `DISCOVERY TYPES` de `man iscsiadm`.

Il existe deux manières de reconfigurer les paramètres d'enregistrement de découverte :

- Modifiez le fichier `/etc/iscsi/iscsid.conf` directement avant d'effectuer une découverte. Les paramètres de découverte utilisent le préfixe `discovery`. Pour les afficher, exécutez :

```
# iscsiadm -m discovery -t discovery_type -p target_IP:port
```

- Alternativement, `iscsiadm` peut aussi être utilisé pour modifier les paramètres d'enregistrement de découverte comme suit :

```
# iscsiadm -m discovery -t discovery_type -p target_IP:port -o update -n
setting -v value
```

Veillez consulter **man iscsiadm** pour obtenir davantage d'informations sur les paramètres **setting** disponibles et les valeurs **value** valides de chacun d'entre eux.

Après avoir configuré les paramètres de découverte, toute tentative consécutive de découvrir de nouvelles cibles utilisera les nouveaux paramètres. Veuillez consulter [Section 24.14, « Scanner les interconnexions iSCSI »](#) pour obtenir des détails sur la manière d'effectuer des recherches de nouvelles cibles iSCSI.

Pour obtenir davantage d'informations sur la configuration de la découverte de cibles iSCSI, veuillez consulter les pages **man** de **iscsiadm** et **iscsid**. Le fichier **/etc/iscsi/iscsid.conf** contient également des exemples sur la syntaxe de configuration correcte.

24.13. Configurer le déchargement et la liaison d'interfaces iSCSI

Ce chapitre décrit comment paramétrer les interfaces iSCSI afin de lier une session à un port NIC lors de l'utilisation d'iSCSI logiciel. Il y est également décrit comment paramétrer des interfaces pour une utilisation avec des périphériques qui prennent en charge le déchargement.

Le sous-système du réseau peut être configuré pour déterminer le chemin ou NIC que les interfaces iSCSI devraient utiliser pour les liaisons. Par exemple, si les portails et NIC sont paramétrés sur différents subnets, alors il ne sera pas nécessaire de configurer les interfaces iSCSI manuellement pour les liaisons.

Avant de tenter de configurer une interface iSCSI pour les liaisons, veuillez exécuter la commande suivante :

```
$ ping -I ethX target_IP
```

Si le **ping** échoue, alors vous ne pourrez pas lier une session à un NIC. Si cela est le cas, veuillez vérifier les paramètres réseau en premier.

24.13.1. Afficher les configurations iface disponibles

Les déchargements et les liaisons d'interface iSCSI sont pris en charge par les implémentations d'initiateur iSCSI suivantes :

Logiciel iSCSI

Cette pile alloue une instance d'hôte iSCSI (apr ex. **scsi_host**) par session, avec une seule connexion par session. De ce fait, **/sys/class/scsi_host** et **/proc/scsi** reporteront un **scsi_host** pour chaque connexion/session à laquelle vous vous connectez.

Offload iSCSI

Le pile alloue une instance **scsi_host** à chaque périphérique PCI. Ainsi, chaque port s'affichera en tant que périphérique PCI séparé sur un adaptateur de bus d'hôte, avec un **scsi_host** différent pour chaque port HBA.

Pour gérer ces deux types d'implémentation d'initiateur, **iscsiadm** utilise la structure **iface**. Avec cette structure, une configuration **iface** doit être saisie dans **/var/lib/iscsi/ifaces** pour chaque port HBA, chaque logiciel iSCSI, ou chaque périphérique de réseau (**ethX**) utilisé pour lier les sessions.

Pour afficher les configurations **iface** disponibles, exécutez **iscsiadm -m iface**. Cela permettra d'afficher les informations **iface** dans le format suivant :

```
iface_name
transport_name, hardware_address, ip_address, net_ifacename, initiator_name
```

Reportez-vous au tableau suivant pour obtenir des détails sur chaque valeur/paramètre.

Tableau 24.2. Configurations iface

Paramètre	Description
iface_name	nom de configuration iface .
transport_name	Nom du pilote
hardware_address	Adresse MAC
ip_address	Adresse IP pour ce port
net_iface_name	Nom utilisé pour le vlan ou pour l'alias de liaison d'une sessions iSCSI. Pour les déchargements iSCSI, net_iface_name sera <vide> car cette valeur n'est pas persistante au redémarrage.
initiator_name	Configuration utilisée pour substituer un nouveau nom au nom par défaut de l'initiateur, défini dans /etc/iscsi/initiatorname.iscsi

Exemple 24.6. Échantillon de sortie de la commande `iscsiadm -m iface`

Ci-dessous figure un exemple de sortie de la commande `iscsiadm -m iface` :

```
iface0 qla4xxx,00:c0:dd:08:63:e8,20.15.0.7,default,iqn.2005-06.com.redhat:madmax
iface1 qla4xxx,00:c0:dd:08:63:ea,20.15.0.9,default,iqn.2005-06.com.redhat:madmax
```

Pour le logiciel iSCSI, chaque configuration **iface** doit avoir un nom unique (de moins de 65 caractères). Le nom **iface_name** des périphériques réseau qui supportent le déchargement apparaît sous le format **transport_name.hardware_name**.

Exemple 24.7. Sortie de la commande `iscsiadm -m iface` avec une carte réseau Chelsio

Ci-dessous figure un exemple de sortie de la commande `iscsiadm -m iface` sur un système utilisant une carte réseau Chelsio :

```
default tcp,<empty>,<empty>,<empty>,<empty>
iser iser,<empty>,<empty>,<empty>,<empty>
cxgb3i.00:07:43:05:97:07 cxgb3i,00:07:43:05:97:07,<empty>,<empty>,<empty>
```

Il est également possible d'afficher les paramètres d'une configuration **iface** particulière d'une façon plus agréable pour l'utilisateur. Pour cela, utiliser l'option **-I iface_name**. Cela affichera les paramètres dans le format suivant :

```
iface.paramètre = valeur
```


Exemple 24.8. Utiliser les paramètres de configuration `iface` avec un adaptateur de réseau Chelsio convergé

En utilisant l'exemple précédent, les paramètres d'`iface` du même adaptateur de réseau Chelsio convergé (par ex. `iscsiadm -m iface -I cxgb3i.00:07:43:05:97:07`) s'afficheront ainsi :

```
# BEGIN RECORD 2.0-871
iface.iscsi_ifacename = cxgb3i.00:07:43:05:97:07
iface.net_ifacename = <empty>
iface.ipaddress = <empty>
iface.hwaddress = 00:07:43:05:97:07
iface.transport_name = cxgb3i
iface.initiatorname = <empty>
# END RECORD
```

24.13.2. Configurer un `iface` pour iSCSI logiciel

Comme mentionné plus tôt, une configuration `iface` est requise pour chaque objet réseau qui sera utilisé pour lier une session.

Avant

Pour créer une configuration `iface` pour iSCSI logiciel, veuillez exécuter la commande suivante :

```
# iscsiadm -m iface -I iface_name --op=new
```

Cela créera une nouvelle configuration `iface` vide avec une valeur `iface_name` spécifiée. Si une configuration `iface` existante possède déjà la même valeur `iface_name`, alors celle-ci sera écrasée par une autre, nouvelle et vide.

Pour configurer le paramètre spécifique d'une configuration `iface`, veuillez utiliser la commande suivante :

```
# iscsiadm -m iface -I iface_name --op=update -n iface.setting -v hw_address
```

Exemple 24.9. Définir l'adresse MAC de `iface0`

Par exemple, pour définir l'adresse MAC (`hardware_address`) de `iface0` sur `00:0F:1F:92:6B:BF`, veuillez exécuter :

```
# iscsiadm -m iface -I iface0 --op=update -n iface.hwaddress -v
00:0F:1F:92:6B:BF
```



Avertissement

Ne pas utiliser `default` ou `iser` comme noms `iface`. Ces deux chaînes sont des valeurs spéciales utilisées par `iscsiadm` pour une compatibilité ascendante. Toute configuration `iface` créée manuellement nommée `default` ou `iser` désactivera la compatibilité ascendante.

24.13.3. Configurer un iface pour le déchargement iSCSI

Par défaut, **iscsiadm** créera une configuration **iface** pour chaque port. Pour afficher les configurations **iface** disponibles, veuillez utiliser la même commande que pour le logiciel iSCSI, c'est-à-dire **iscsiadm -m iface**.

Avant d'utiliser l'**iface** d'une carte réseau pour le déchargement iSCSI, veuillez commencer par définir l'adresse IP (**target_IP**^[5]) que le périphérique devrait utiliser. Pour les périphériques qui utilisent le pilote **be2iscsi**, l'adresse IP est configurée dans l'écran de paramétrage BIOS. Pour tous les autres périphériques, pour configurer l'adresse IP d'**iface**, utilisez :

```
# iscsiadm -m iface -I iface_name -o update -n iface.ipaddress -v target_IP
```

Exemple 24.10. Paramétrez l'adresse IP de l'iface d'une carte Chelsio

Par exemple, pour définir l'adresse IP **iface** à **20.15.0.66** avec une carte dont le nom **iface** est **cxgb3i.00:07:43:05:97:07**, utilisez :

```
# iscsiadm -m iface -I cxgb3i.00:07:43:05:97:07 -o update -n
iface.ipaddress -v 20.15.0.66
```

24.13.4. Lier ou délier un iface sur un portail

Chaque fois que **iscsiadm** est utilisé pour rechercher des interconnexions, la vérification commencera par les paramètres **iface.transport** de chaque configuration **iface** dans **/var/lib/iscsi/ifaces**. L'utilitaire **iscsiadm** liera ensuite les portails découverts avec tout **iface** dont **iface.transport** est **tcp**.

Ce comportement a été implémenté pour des raisons de compatibilité. Pour le remplacer, veuillez utiliser **-I *iface_name*** pour spécifier quel portail lier à un **iface**, comme suit :

```
# iscsiadm -m discovery -t st -p target_IP:port -I iface_name -P 1
[5]
```

Par défaut, l'utilitaire **iscsiadm** ne liera pas automatiquement tous les portails aux configurations **iface** utilisant le déchargement. Ceci est dû au fait que de telles configurations **iface** n'ont pas paramétré **iface.transport** sur **tcp**. Ainsi, les configurations **iface** doivent être liées manuellement aux portails découverts.

Il est également possible d'empêcher un portail de se lier à tout **iface** existant. Pour ce faire, veuillez utiliser **default** comme **iface_name** de la manière suivante :

```
# iscsiadm -m discovery -t st -p IP:port -I default -P 1
```

Pour supprimer la liaison entre une cible et **iface**, veuillez utiliser :

```
# iscsiadm -m node -targetname proper_target_name -I iface0 --op=delete [6]
```

Pour supprimer toutes les liaisons à un **iface** particulier, veuillez utiliser :

```
# iscsiadm -m node -I iface_name --op=delete
```

Pour supprimer les liaisons d'un portail particulier (par exemple pour des cibles Equallogic), veuillez utiliser :

```
# iscsiadm -m node -p IP:port -I iface_name --op=delete
```



Note

S'il n'y a pas de configuration **iface** définie dans `/var/lib/iscsi/iface` et que l'option **-I** n'est pas utilisée, **iscsiadm** autorisera le sous-système du réseau à décider quel périphérique un portail particulier devrait utiliser.

24.14. Scanner les interconnexions iSCSI

Pour iSCSI, si les cibles envoient un événement iSCSI asynchrone indiquant qu'un nouveau stockage a été ajouté, alors le scan est effectué automatiquement.

Cependant, si les cibles n'envoient pas d'événement iSCSI asynchrone, vous devrez les scanner manuellement en utilisant l'utilitaire **iscsiadm**. Cependant, avant cela, vous devrez récupérer les valeurs **--targetname** et **--portal** correctes. Si le modèle de votre périphérique prend uniquement en charge une seule unité logique et un seul portail par cible, utilisez **iscsiadm** pour délivrer une commande **sendtargets** sur l'hôte, comme suit :

```
# iscsiadm -m discovery -t sendtargets -p target_IP:port
[5]
.....
```

La sortie apparaîtra sous le format suivant :

```
target_IP:port,target_portal_group_tag proper_target_name
```

Exemple 24.11. Utiliser **iscsiadm** pour envoyer une commande **sendtargets**

Par exemple, sur une cible dont la valeur **proper_target_name** est **iqn.1992-08.com.netapp:sn.33615311** et dont la valeur **target_IP:port** est **10.15.85.19:3260**, la sortie pourrait être comme suit :

```
10.15.84.19:3260,2 iqn.1992-08.com.netapp:sn.33615311
10.15.85.19:3260,3 iqn.1992-08.com.netapp:sn.33615311
```

Dans cet exemple, la cible possède deux portails, chacun utilisant des valeurs **target_ip:port** de **10.15.84.19:3260** et de **10.15.85.19:3260**.

Pour voir quelle configuration **iface** sera utilisée pour chaque session, veuillez ajouter l'option **-P 1**. Cette option imprimera également les informations de la session sous la forme d'une arborescence, comme suit :

```
Target: proper_target_name
  Portal: target_IP:port,target_portal_group_tag
    Iface Name: iface_name
```

Exemple 24.12. Afficher la configuration iface

Par exemple, avec `iscsiadm -m discovery -t sendtargets -p 10.15.85.19:3260 -P 1`, la sortie pourrait être comme suit :

```
Target: iqn.1992-08.com.netapp:sn.33615311
  Portal: 10.15.84.19:3260,2
    Iface Name: iface2
  Portal: 10.15.85.19:3260,3
    Iface Name: iface2
```

Ceci signifie que la cible **iqn.1992-08.com.netapp:sn.33615311** utilisera **iface2** comme configuration **iface**.

Cependant, avec certains modèles de périphérique, une cible unique peut avoir de multiples unités logiques et portails. Dans ce cas, commencez par envoyer une commande **sendtargets** sur l'hôte pour trouver de nouveaux portails sur la cible. Puis, scannez à nouveau les sessions existantes en utilisant :

```
# iscsiadm -m session --rescan
```

Vous pouvez également scanner à nouveau une session particulière en indiquant sa valeur **SID** comme suit :

```
# iscsiadm -m session -r SID --rescan [7]
```

Si votre périphérique prend en charge des cibles multiples, vous devrez exécuter la commande **sendtargets** sur les hôtes pour trouver les nouveaux portails de *chaque* cible. Scannez à nouveau les sessions existantes pour découvrir de nouvelles unités logiques en utilisant l'option **--rescan**.



Important

La commande **sendtargets** utilisée pour récupérer les valeurs **--targetname** et **--portal** remplace le contenu de la base de données **/var/lib/iscsi/nodes**. Cette base de données sera alors remplie à nouveau en utilisant les paramètres du fichier **/etc/iscsi/iscsid.conf**. Cependant, ceci ne se produira pas si une session est actuellement connectée et en cours d'utilisation.

Pour ajouter de nouveaux portails ou cibles ou pour en supprimer en toute sécurité, veuillez utiliser l'option **-o new** ou **-o delete**, respectivement. Par exemple, pour ajouter de nouveaux portails ou cibles sans écraser **/var/lib/iscsi/nodes**, veuillez utiliser la commande suivante :

```
iscsiadm -m discovery -t st -p target_IP -o new
```

Pour supprimer les entrées **/var/lib/iscsi/nodes** que la cible n'a pas affichées pendant la découverte, veuillez utiliser :

```
iscsiadm -m discovery -t st -p target_IP -o delete
```

Il est également possible d'effectuer les deux tâches simultanément, comme suit :

```
iscsiadm -m discovery -t st -p target_IP -o delete -o new
```

La commande **sendtargets** générera la sortie suivante :

```
ip:port,target_portal_group_tag proper_target_name
```

Exemple 24.13. Sortie de la commande **sendtargets**

Ainsi, avec une cible unique, une unité logique et un portail, en utilisant **equallogic-iscsi1** en tant que **target_name**, la sortie devrait être similaire à la suivante :

```
10.16.41.155:3260,0 iqn.2001-05.com.equallogic:6-8a0900-ac3fe0101-63aff113e344a4a2-d1585-03-1
```

Remarquez que **proper_target_name** et **ip:port,target_portal_group_tag** sont identiques aux valeurs du même nom dans [Section 24.6.1, « API iSCSI »](#).

À ce moment, vous serez en possession des valeurs **--targetname** et **--portal** nécessaires pour scanner manuellement les périphériques iSCSI. Pour ce faire, veuillez exécuter la commande suivante :

```
# iscsiadm --mode node --targetname proper_target_name --portal
ip:port,target_portal_group_tag \ --login
[8]
```

Exemple 24.14. Commande **iscsiadm** complète

En utilisant l'exemple précédent (dans lequel *proper_target_name* est **equallogic-iscsi1**), la commande complète aura la forme suivante :

```
# iscsiadm --mode node --targetname \ iqn.2001-05.com.equallogic:6-8a0900-ac3fe0101-63aff113e344a4a2-d1585-03-1 \ --portal 10.16.41.155:3260,0 --login[8]
```

24.15. Connexion à une cible iSCSI

Comme mentionné dans le [Section 24.6, « iSCSI »](#), le service iSCSI doit être en cours d'exécution afin de découvrir ou de se connecter aux cibles. Pour lancer le service iSCSI, veuillez exécuter :

```
# service iscsi start
```

Lorsque cette commande est exécutée, les scripts **init** iSCSI se connecteront automatiquement aux cibles sur lesquelles le paramètre **node.startup** est configuré sur **automatic**. Ceci est la valeur par défaut de **node.startup** pour toutes les cibles.

Pour empêcher les connexions automatiques à une cible, paramétrez **node.startup** sur **manual**. Pour effectuer ceci, veuillez exécuter la commande suivante :

```
# iscsiadm -m node --targetname proper_target_name -p target_IP:port -o update -n node.startup -v manual
```

La suppression de l'enregistrement entier préviendra également les connexions automatiques. Pour effectuer ceci, veuillez exécuter :

```
# iscsiadm -m node --targetname proper_target_name -p target_IP:port -o delete
```

Pour monter un système de fichiers automatiquement à partir d'un périphérique iSCSI sur le réseau, ajoutez une entrée de partition pour le montage dans **/etc/fstab** avec l'option **_netdev**. Par exemple, pour automatiquement monter le périphérique iSCSI **sdb** sur **/mnt/iscsi** pendant le démarrage, ajoutez la ligne suivante à **/etc/fstab** :

```
/dev/sdb /mnt/iscsi ext3 _netdev 0 0
```

Pour vous connecter manuellement à une cible iSCSI, veuillez utiliser la commande suivante :

```
# iscsiadm -m node --targetname proper_target_name -p target_IP:port -l
```



Note

proper_target_name et *target_IP:port* font référence au nom complet et à la combinaison adresse IP et port d'une cible. Pour obtenir davantage d'informations, veuillez consulter [Section 24.6.1, « API iSCSI »](#) et [Section 24.14, « Scanner les interconnexions iSCSI »](#).

24.16. Redimensionner une Unité logique En ligne

Dans la plupart des cas, le redimensionnement complet d'une *unité logique* en ligne implique deux choses : le redimensionnement de l'unité logique elle-même et la réflexion du changement de taille sur le périphérique multipath correspondant (si multipath est activé sur le système).

Pour redimensionner l'unité logique en ligne, commencez par modifier la taille de l'unité logique à travers l'interface de gestion de matrices du périphérique de stockage. Cette procédure est différente avec chaque matrice, veuillez donc consulter la documentation du fournisseur de la matrice de stockage pour obtenir davantage d'informations.



Note

Pour pouvoir redimensionner un système de fichiers en ligne, celui-ci doit ne pas résider sur un périphérique partitionné.

24.16.1. Redimensionner des unités logiques Fibre Channel

Après avoir modifié la taille d'une unité logique en ligne, veuillez scanner l'unité logique à nouveau afin de vous assurer que le système détecte bien la taille mise à jour. Pour faire la même chose avec des unités logiques Fibre Channel, veuillez utiliser la commande suivante :

```
$ echo 1 > /sys/block/sdX/device/rescan
```



Important

Pour scanner à nouveau des unités logiques Fibre Channel sur un système utilisant des multivoies (« multipathing »), veuillez exécuter la commande ci-dessus pour chaque périphérique sd (c'est-à-dire **sd1**, **sd2**, et ainsi de suite...) qui représente un chemin pour l'unité logique multivoies. Pour déterminer quels périphériques sont des chemins pour une unité logique multivoies, veuillez utiliser **multipath -ll** ; puis trouvez l'entrée correspondant à l'unité logique en cours de redimensionnement. Il est recommandé de faire référence à l'ID global (WWID) de chaque entrée afin de faciliter la recherche de celui qui correspond à l'unité logique en cours de redimensionnement.

24.16.2. Redimensionner une unité logique iSCSI

Après avoir modifié la taille d'une unité logique en ligne, veuillez scanner l'unité logique à nouveau afin de vous assurer que le système détecte bien la taille mise à jour. Pour faire la même chose avec des périphériques iSCSI, veuillez utiliser la commande suivante :

```
# iscsiadm -m node --targetname target_name -R  
[5]  
.....
```

Remplacez **target_name** par le nom de la cible sur laquelle le périphérique se trouve.



Note

Il est également possible de scanner à nouveau des unités logiques iSCSI en utilisant la commande suivante :

```
# iscsiadm -m node -R -I interface
```

Remplacez ***interface*** par le nom d'interface correspondant de l'unité logique redimensionnée (par exemple, ***iface0***). Cette commande effectue deux opérations :

- » Un scan est effectué pour trouver de nouveaux périphériques de la même manière que le fait la commande **echo "- - -" > /sys/class/scsi_host/host/scan** (veuillez consulter [Section 24.14, « Scanner les interconnexions iSCSI »](#)).
- » Un second scan est effectué pour trouver les nouvelles unités logiques ou les unités logiques modifiées de la même manière que le fait la commande **echo 1 > /sys/block/sdX/device/rescan**. Remarquez que cette commande est également la même que celle utilisée pour scanner à nouveau des unités logiques Fibre Channel.

24.16.3. Mettre à jour la taille du périphérique multivoies (« Multipath »)

Si la fonction multivoies est activée sur le système, vous devrez également refléter la modification de taille d'unité logique sur le périphérique multivoies correspondant de l'unité logique (*après avoir redimensionné l'unité logique*). Cela peut être fait avec **multipathd**. Tout d'abord, veuillez vous assurer que **multipathd** soit bien en cours d'exécution en utilisant **service multipathd status**. Une fois le bon fonctionnement de **multipathd** vérifié, veuillez exécuter la commande suivante :

```
# multipathd -k"resize map multipath_device"
```

La variable ***multipath_device*** est l'entrée multivoies correspondante du périphérique dans **/dev/mapper**. Selon la manière par laquelle la fonction multivoies est configurée sur le système, ***multipath_device*** peut se trouver sous deux différents formats :

- » **mpathX**, où **X** est l'entrée correspondante du périphérique (par exemple, **mpath0**)
- » un WWID ; par exemple, **3600508b400105e210000900000490000**

Pour déterminer quelle entrée multivoies correspond à l'unité logique redimensionnée, veuillez exécuter **multipath -ll**. Ceci affiche une liste de toutes les entrées multivoies dans le système, ainsi que les chiffres majeurs et mineurs des périphériques correspondants.



Important

Ne pas utiliser **multipathd -k"resize map *multipath_device*"** si des commandes sont en file d'attente sur ***multipath_device***. Autrement dit, veuillez ne pas utiliser cette commande lorsque le paramètre **no_path_retry** (dans **/etc/multipath.conf**) est défini sur **"queue"**, et qu'il n'existe aucun chemin actif vers le périphérique.

Pour obtenir davantage d'informations sur les multiples voies, veuillez consulter le *Guide Red Hat Enterprise Linux 7 DM Multipath*.

24.16.4. Modifier l'état de lecture/écriture d'une unité logique en ligne

Certains périphériques de stockage offrent à l'utilisateur la possibilité de modifier l'état du périphérique de lecture/écriture (« Read/Write », ou « R/W ») à lecture seule (« Read-Only », ou « RO »), et de RO à R/W. Ceci est généralement effectué par une interface de gestion sur le périphérique de stockage. Le système d'exploitation ne met pas à jour l'affichage de l'état du périphérique automatiquement lorsqu'un changement est effectué. Veuillez observer les procédures suivantes pour que le système prenne conscience du changement.

Exécutez la commande suivante en remplaçant XYZ par l'appellation souhaitée du périphérique pour déterminer l'affichage actuel de l'état R/W d'un périphérique du système d'exploitation :

```
# blockdev --getro /dev/sdXYZ
```

La commande suivante est également disponible dans Red Hat Enterprise Linux 7:

```
# cat /sys/block/sdXYZ/ro 1 = read-only 0 = read-write
```

Lors de l'utilisation de multivoies, veuillez consulter le champ *ro* ou *rw* dans la seconde ligne de la sortie de la commande **multipath -ll**. Exemple :

```
36001438005deb4710000500000640000 dm-8 GZ,GZ500
[size=20G][features=0][hwhandler=0][ro]
\_ round-robin 0 [prio=200][active]
  \_ 6:0:4:1 sdax 67:16 [active][ready]
  \_ 6:0:5:1 sday 67:32 [active][ready]
\_ round-robin 0 [prio=40][enabled]
  \_ 6:0:6:1 sdaz 67:48 [active][ready]
  \_ 6:0:7:1 sdba 67:64 [active][ready]
```

Pour modifier l'état R/W, veuillez utiliser la procédure suivante :

Procédure 24.12. Modifier l'état R/W

1. Pour déplacer l'état du périphérique de RO à R/W, veuillez consulter l'étape 2.

Pour déplacer l'état du périphérique de R/W à RO, assurez-vous qu'aucune écriture supplémentaire ne sera passée. Ceci peut être fait en arrêtant l'application, ou en utilisant une action correspondante spécifique à l'application.

Assurez-vous que toutes les E/S d'écriture restantes soient bien terminées par la commande suivante :

```
# blockdev --flushbufs /dev/device
```

Remplacez *device* par l'appellation souhaitée. Pour un périphérique multivoies mappé de périphériques, il s'agit de l'entrée du périphérique dans **dev/mapper**. Par exemple, **/dev/mapper/mpath3**.

2. Veuillez utiliser l'interface de gestion du périphérique de stockage pour changer l'état de l'unité logique de RW (lecture/écriture) à RO (lecture seule), ou au contraire de RO à RW. Cette procédure est différente pour chaque matrice. Pour obtenir davantage d'informations, veuillez consulter la documentation du fournisseur de matrice de stockage applicable.

3. Scannez à nouveau le périphérique pour mettre à jour l'affichage de l'état RW du système d'exploitation du périphérique. Si vous utilisez un périphérique mappeur multivoies, veuillez ré-effectuer ce scan pour chaque chemin vers le périphérique avant de passer la commande ordonnant au périphérique multivoies de recharger ses cartes de périphérique.

Ce processus est expliqué en détails dans [Section 24.16.4.1, « Scanner à nouveau des unités logiques »](#).

24.16.4.1. Scanner à nouveau des unités logiques

Après avoir modifié l'état RW de l'unité logique en ligne comme décrit dans [Section 24.16.4, « Modifier l'état de lecture/écriture d'une unité logique en ligne »](#), veuillez scanner à nouveau l'unité logique afin de vous assurer que le système détecte l'état mis à jour par commande suivante :

```
# echo 1 > /sys/block/sdX/device/rescan
```

Pour scanner à nouveau des unités logiques sur un système qui utilise la fonction multivoies, veuillez exécuter la commande ci-dessus pour chaque périphérique sd qui représente un chemin pour l'unité logique multivoies. Par exemple, exécutez la commande sur sd1, sd2, et sur tous les autres périphériques sd. Pour déterminer les périphériques qui sont des chemins pour une unité multivoies, veuillez utiliser **multipath -11**, puis trouvez l'entrée qui correspond à l'unité logique devant être modifiée.

Exemple 24.15. Utilisation de la commande **multipath -11**

Par exemple, **multipath -11** affiche le chemin pour le LUN avec l'ID WWID 36001438005deb4710000500000640000. Dans ce cas, veuillez saisir :

```
# echo 1 > /sys/block/sdax/device/rescan
# echo 1 > /sys/block/sday/device/rescan
# echo 1 > /sys/block/sdaz/device/rescan
# echo 1 > /sys/block/sdba/device/rescan
```

24.16.4.2. Mettre à jour l'état RW d'un périphérique multivoies

Après avoir scanné l'unité logique, si la fonction multivoies est activée, le changement de son état devra être reflété sur le disque multivoies correspondant de l'unité logique. Pour ce faire, veuillez recharger les cartes du périphérique multivoies par la commande suivante :

```
# multipath -r
```

La commande **multipath -11** peut être utilisée pour confirmer le changement.

24.16.4.3. Documentation

Davantage d'informations se trouvent dans la Base de connaissances de Red Hat. Pour y accéder, veuillez vous rendre sur <https://www.redhat.com/wapps/sso/login.html?redirect=https://access.redhat.com/knowledge/> et connectez-vous. Puis accédez à l'article sur <https://access.redhat.com/kb/docs/DOC-32850>.

24.17. Ajouter ou supprimer une unité logique avec **rescan-scsi-bus.sh**

Le paquet `sg3_utils` fournit le script `rescan-scsi-bus.sh`, qui peut automatiquement mettre à jour la configuration de l'unité logique de l'hôte selon les besoins (après l'ajout du périphérique au système). Le script `rescan-scsi-bus.sh` peut également effectuer une opération `issue_lip` sur les périphériques pris en charge. Pour obtenir davantage d'informations sur l'utilisation de ce script, veuillez consulter `rescan-scsi-bus.sh --help`.

Pour installer le paquet `sg3_utils`, veuillez exécuter `yum install sg3_utils`.

Problèmes connus avec `rescan-scsi-bus.sh`

Lors de l'utilisation du script `rescan-scsi-bus.sh`, veuillez prendre note des problèmes connus suivants :

- Afin que `rescan-scsi-bus.sh` fonctionne correctement, **LUN0** doit être la première unité logique mise en correspondance. `rescan-scsi-bus.sh` peut uniquement détecter la première unité logique mise en correspondance s'il s'agit de **LUN0**. `rescan-scsi-bus.sh` ne pourra pas scanner d'autre unité logique sans avoir détecté la première unité logique au préalable, même si vous utilisez l'option `--nooptscan`.
- Une condition race requiert que `rescan-scsi-bus.sh` soit exécuté deux fois si les unités logiques sont mises en correspondance pour la première fois. Pendant cette première recherche, `rescan-scsi-bus.sh` ajoute uniquement **LUN0** ; toutes les autres unités logiques seront ajoutées pendant la seconde recherche.
- Un bogue dans le script `rescan-scsi-bus.sh` exécute incorrectement la fonctionnalité de reconnaissance du changement de taille de l'unité logique lorsque l'option `--remove` est utilisée.
- Le script `rescan-scsi-bus.sh` ne reconnaît pas les suppressions d'unités logiques iSCSI.

24.18. Modifier le comportement de la perte de lien

Cette section décrit comment modifier le comportement de la perte de lien des périphériques utilisant les protocoles Fibre Channel ou iSCSI.

24.18.1. Fibre Channel

Si un pilote implémente le rappel `dev_loss_tmo` de transport, les tentatives d'accès à un périphérique via un lien seront bloquées lorsqu'un problème de transport est détecté. Pour vérifier si un périphérique est bloqué, exécutez la commande suivante :

```
$ cat /sys/block/device/device/state
```

Cette commande retournera **blocked** (« bloqué ») si le périphérique est bloqué. Si le périphérique fonctionne normalement, alors cette commande retournera **running** (« en cours d'exécution »).

Procédure 24.13. Déterminer l'état d'un port distant

1. Pour déterminer l'état d'un port distant, exécutez la commande suivante :

```
$ cat /sys/class/fc_remote_port/rport-H:B:R/port_state
```

2. Cette commande retournera **Blocked** (« bloqué ») lorsque le port distant ainsi que les périphériques pouvant être atteints par ce biais sont bloqués. Si le port distant fonctionne normalement, la commande retournera **Online** (« en ligne »).

3. Si le problème n'est pas résolu en **dev_loss_tmo** secondes, le port et les périphériques seront débloqués et toutes les E/S exécutées sur ce périphérique (ainsi que toute nouvelle E/S envoyée sur celui-ci) échoueront.

Procédure 24.14. Modifier dev_loss_tmo

- » Pour modifier la valeur de **dev_loss_tmo**, saisissez avec **echo** la valeur souhaitée dans le fichier. Par exemple, pour définir **dev_loss_tmo** sur 30 secondes, veuillez exécuter :

```
$ echo 30 >
/sys/class/fc_remote_port/rport-H:B:R/dev_loss_tmo
```

Pour obtenir davantage d'informations sur **dev_loss_tmo**, veuillez consulter [Section 24.3.1, « Interface de programmation Fibre Channel »](#).

Quand une perte de lien dépasse **dev_loss_tmo**, les périphériques **scsi_device** et **sdN** sont supprimés. Normalement, la classe de Fibre Channel laisse le périphérique tel quel; c-a-d **/dev/sdx** demeure **/dev/sdx**. C'est parce que la liaison cible est conservée par le pilote de Fibre Channel, donc quand le port cible revient, les adresses SCSI sont recrées en rapport. Cependant, il n'y a rien de garanti ; le **sdX** ne sera restauré que si aucun changement supplémentaire n'est fait à la configuration de la case « in-storage » du LUN.

24.18.2. Paramètres iSCSI avec dm-multipath

Si **dm-multipath** est implémenté, il est recommandé de paramétrer les minuteurs iSCSI immédiatement pour déferer les commandes sur la couche multipath. Pour configurer ceci, ajoutez la ligne suivante à **device {** dans **/etc/multipath.conf** :

```
features "1 queue_if_no_path"
```

Ceci permet de s'assurer que les erreurs d'E/S soient ré-essayées et remises en file d'attente si tous les chemins sont en échec dans la couche **dm-multipath**.

Vous pourriez devoir réajuster les minuteurs iSCSI afin de mieux contrôler qu'il ne se produise pas de problème sur votre disque SAN. *NOP-Out Interval/Timeouts* et **replacement_timeout** sont des minuteurs iSCSI disponibles et configurables, dont il est question dans les sections suivantes.

24.18.2.1. Intervalle/Délai d'expiration NOP-Out

Pour aider à contrôler les problèmes SAN, la couche iSCSI envoie une requête NOP-Out à chaque cible. Si une requête NOP-Out dépasse son délai d'expiration, la couche iCSI répond en faisant échouer toute commande exécutée et en envoyant à la couche SCSI l'instruction de remettre ces commandes en file d'attente dès que possible.

Lorsque la commande **dm-multipath** est utilisée, la couche SCSI fera échouer les commandes en cours d'exécution et les déferera sur la couche multipath. Puis, la couche multipath tentera à nouveau ces commandes sur un autre chemin. Si **dm-multipath** n'est pas en cours d'utilisation, ces commandes seront tentées à nouveau cinq fois avant d'être en échec.

Les intervalles entre requêtes NOP-Out sont de 10 secondes par défaut. Pour ajuster cet intervalle, ouvrez le fichier **/etc/iscsi/iscsid.conf** et modifiez la ligne suivante :

```
node.conn[0].timeo.noop_out_interval = [interval value]
```

Une fois définie, la couche iSCSI enverra une requête NOP-Out sur chaque cible toutes les *[valeur de l'intervalle]* secondes.

By default, NOP-Out requests time out in 10 seconds ^[9]. To adjust this, open `/etc/iscsi/iscsid.conf` and edit the following line:

```
node.conn[0].timeo.noop_out_timeout = [timeout value]
```

Ceci paramètre la couche iSCSI pour faire expirer les requêtes NOP-Out après *[timeout value]* secondes.

SCSI Error Handler

Si SCSI Error Handler est en cours d'exécution, exécuter des commandes sur un chemin n'échouera pas immédiatement lorsque le délai d'une requête NOP-Out expire sur ce chemin. À la place, ces commandes échoueront *après replacement_timeout* secondes. Pour obtenir davantage d'informations sur **replacement_timeout**, veuillez consulter [Section 24.18.2.2, « replacement_timeout »](#).

Pour vérifier si SCSI Error Handler est cours d'exécution :

```
# iscsiadm -m session -P 3
```

24.18.2.2. replacement_timeout

replacement_timeout contrôle le temps que la couche iSCSI doit attendre pour qu'un chemin ou une session dont le délai a expiré puisse se rétablir avant de faire échouer ses commandes. La valeur par défaut de **replacement_timeout** s'élève à 120 secondes.

Pour ajuster **replacement_timeout**, ouvrez `/etc/iscsi/iscsid.conf` et modifiez la ligne suivante :

```
node.session.timeo.replacement_timeout = [replacement_timeout]
```

L'option **1 queue_if_no_path** dans `/etc/multipath.conf` paramètre les horodateurs iSCSI pour déferer immédiatement les commandes sur la couche multipath (veuillez consulter [Section 24.18.2, « Paramètres iSCSI avec dm-multipath »](#)). Ce paramètre empêche les erreurs d'E/S de se propager à l'application. Par conséquent, vous pouvez définir **replacement_timeout** sur 15 à 20 secondes.

En configurant une valeur **replacement_timeout** plus basse, les E/S sont rapidement envoyées vers un nouveau chemin et exécutées (en cas de dépassement du délai NOP-Out) tandis que la couche iSCSI tente de rétablir le chemin ou la session en échec. Si le délai expire pour tous les chemins, alors la couche du mappeteur de périphérique et du multipath mettront les E/S en file d'attente interne, en se basant sur les paramètres du fichier `/etc/multipath.conf` au lieu de ceux de `/etc/iscsi/iscsid.conf`.



Important

Que vous preniez plutôt en compte la vitesse de basculement ou la sécurité, la valeur recommandée de **replacement_timeout** dépendra également d'autres facteurs. Ces autres facteurs incluent le réseau, la cible, et la charge de travail du système. Ainsi, il est recommandé de minutieusement tester **replacements_timeout** avec toute nouvelle configuration avant de l'appliquer sur un système à mission critique.

24.18.3. Root iSCSI

Lors de l'accès à la partition root à travers un disque iSCSI, les compteurs iSCSI doivent être définis de manière à ce que la couche iSCSI possède plusieurs chances de rétablir un chemin ou une session. En outre, les commandes ne doivent pas être remises trop rapidement dans la file d'attente de la couche SCSI. Ceci est le contraire de ce qui doit être fait lorsque **dm-multipath** est implémenté.

Pour commencer, NOP-Out devrait être désactivé. Ceci peut être effectué en paramétrant l'intervalle et le délai d'expiration de NOP-Out sur zéro. Pour effectuer cela, ouvrez le fichier `/etc/iscsi/iscsid.conf` et modifiez-le comme suit :

```
node.conn[0].timeo.noop_out_interval = 0
node.conn[0].timeo.noop_out_timeout = 0
```

Similairement, le paramètre **replacement_timeout** doit être élevé. Cela instruira au système d'attendre plus longtemps qu'un chemin ou une session se rétablisse. Pour ajuster **replacement_timeout**, ouvrez `/etc/iscsi/iscsid.conf` et modifiez la ligne suivante :

```
node.session.timeo.replacement_timeout = replacement_timeout
```

Après avoir configuré `/etc/iscsi/iscsid.conf`, vous devriez effectuer une redécouverte du stockage affecté. Cela permettra au système de charger et d'utiliser les valeurs nouvelles dans `/etc/iscsi/iscsid.conf`. Pour obtenir davantage d'informations sur la manière de découvrir les périphériques iSCSI, veuillez consulter [Section 24.14, « Scanner les interconnexions iSCSI »](#).

Configurer des délais d'expiration pour une session particulière

Vous pouvez également configurer des délais d'expiration pour une session particulière et les rendre non-persistants (au lieu d'utiliser `/etc/iscsi/iscsid.conf`). Pour effectuer ceci, exécutez la commande suivante (en remplaçant les variables au besoin) :

```
# iscsiadm -m node -T target_name -p target_IP:port -o update -n
node.session.timeo.replacement_timeout -v $timeout_value
```



Important

La configuration décrite ici est recommandée pour les sessions iSCSI impliquant l'accès à une partition root. Pour les sessions iSCSI impliquant l'accès à d'autres types de stockage (par exemple avec des systèmes utilisant **dm-multipath**), veuillez consulter [Section 24.18.2, « Paramètres iSCSI avec dm-multipath »](#).

24.19. Contrôle du Minuteur de commande SCSI et du Statut de périphérique

La couche Linux SCSI définit un minuteur pour chaque commande. Lorsque ce délai expire, la couche SCSI va suspendre l'*adaptateur de bus hôte* (HBA) et attendre que toutes les commandes dépassent le délai d'expiration ou soient complétées. Par la suite, la couche SCSI activera le gestionnaire d'erreurs du pilote.

Quand le gestionnaire d'erreurs est déclenché, il tente les opérations suivantes dans l'ordre (jusqu'à ce qu'une opération exécute avec succès) :

1. Annuler la commande.

2. Réinitialiser le périphérique.
3. Réinitialiser le bus.
4. Réinitialiser l'hôte.

Si toutes ces opérations échouent, le périphérique sera sur le statut **offline**. Dans un tel cas, toutes les E/S de ce périphérique échoueront, jusqu'à ce que le problème soit corrigé et que l'utilisateur définisse le périphérique à **running**.

Le processus est différent, cependant, si un périphérique utilise le protocole Fibre Channel et que le **rport** est bloqué, les pilotes attendent quelques secondes pour que le **rport** revienne en ligne à nouveau avant d'activer le gestionnaire d'erreurs. Cela empêche les périphériques d'être hors ligne en raison de problèmes de transport temporaires.

États de périphériques

Pour afficher l'état d'un périphérique, exécutez :

```
$ cat /sys/block/device-name/device/state
```

Pour définir un périphérique à l'état **running**, exécutez :

```
# echo running > /sys/block/device-name/device/state
```

Minuteur de commande

Pour contrôler le minuteur de commande, modifiez le fichier **/sys/block/device-name/device/timeout** :

```
# echo value > /sys/block/device-name/device/timeout
```

Remplacez la **valeur** de la commande par la valeur du délai d'expiration (en secondes) que vous souhaitez implémenter.

24.20. Résolution de problème de configuration de stockage en ligne

Cette section fournit des solutions aux problèmes courants attendant à la reconfiguration de stockage en ligne.

Le statut de la suppression d'unité logique n'est pas reflétée sur l'hôte.

Lorsqu'une unité logique est supprimée sur un fichier configuré, le changement ne se reflète pas sur l'hôte. Dans de tels cas, les commandes **lvm** seront en suspend indéfiniment quand **dm-multipath** sera utilisé, car l'unité logique est maintenant *caduque*.

Pour contourner ce problème, effectuez la procédure suivante :

Procédure 24.15. Contourner le problème d'unités logiques caduques

1. Déterminer les entrées du lien **mpath** de **/etc/lvm/cache/.cache** qui sont spécifiques à l'unité logique caduque. Pour cela, exécutez la commande suivante :

```
$ ls -l /dev/mpath | grep stale-logical-unit
```

Exemple 24.16. Déterminer les entrées de lien mpath spécifiques

Par exemple, si **stale-logical-unit** correspond à 3600d0230003414f30000203a7bc41a00, les résultats suivants apparaîtront :

```
lrwxrwxrwx 1 root root 7 Aug  2 10:33
/3600d0230003414f30000203a7bc41a00 -> ../dm-4
lrwxrwxrwx 1 root root 7 Aug  2 10:33
/3600d0230003414f30000203a7bc41a00p1 -> ../dm-5
```

Cela signifie que 3600d0230003414f30000203a7bc41a00 est mappé à deux liens **mpath** : **dm-4** et **dm-5**.

2. Puis, ouvrez **/etc/lvm/cache/.cache**. Supprimer toutes les lignes contenant **stale-logical-unit** et les liens **mpath** auxquels **stale-logical-unit** se mappe.

Exemple 24.17. Supprimer les lignes qui conviennent

En utilisant l'exemple de l'étape précédente, voici les lignes qu'il vous faudra supprimer :

```
/dev/dm-4
/dev/dm-5
/dev/mapper/3600d0230003414f30000203a7bc41a00
/dev/mapper/3600d0230003414f30000203a7bc41a00p1
/dev/mpath/3600d0230003414f30000203a7bc41a00
/dev/mpath/3600d0230003414f30000203a7bc41a00p1
```

[5] The **target_IP** and **port** variables refer to the IP address and port combination of a target/portal, respectively. For more information, refer to [Section 24.6.1, « API iSCSI »](#) and [Section 24.14, « Scanner les interconnexions iSCSI »](#).

[6] Refer to [Section 24.14, « Scanner les interconnexions iSCSI »](#) for information on **proper_target_name**.

[7] For information on how to retrieve a session's SID value, refer to [Section 24.6.1, « API iSCSI »](#).

[8] This is a single command split into multiple lines, to accommodate printed and PDF versions of this document. All concatenated lines — preceded by the backslash (\) — should be treated as one command, sans backslashes.

[9] Prior to Red Hat Enterprise Linux 5.4, the default NOP-Out requests time out was 15 seconds.

Chapitre 25. Mappeur de périphériques à multiples chemins d'accès et Stockage virtuel

Red Hat Enterprise Linux 7 prend en charge *DM-Multipath* et le *Stockage virtuel*. Ces deux fonctionnalités sont documentées en détail dans le livre *Red Hat DM Multipath* et dans le *Guide de déploiement et d'administration de la virtualisation*.

25.1. Stockage virtuel

Pour le stockage virtuel, Red Hat Enterprise Linux 7 prend en charge les méthodes de stockage de systèmes de fichiers ou méthodes de stockage en ligne suivantes :

- » Fibre Channel
- » iSCSI
- » NFS
- » GFS2

La virtualisation dans Red Hat Enterprise Linux 7 utilise **libvirt** pour gérer les instances virtuelles. L'utilitaire **libvirt** utilise le concept de *pools de stockage* pour gérer le stockage d'invités virtualisés. Un pool de stockage est un stockage pouvant être divisé en volumes de plus petite taille ou directement alloué(s) à un invité. Les volumes d'un pool de stockage peuvent être alloués à des invités virtualisés. Il existe deux catégories de pools de stockage sont :

Pools de stockage locaux

Le stockage local couvre les périphériques de stockage, les fichiers ou répertoires directement attachés à un hôte. Le stockage local inclut les répertoires locaux, les disques directement attachés, et les groupes de volumes LVM.

Pools de stockage en réseau (partagés)

Le stockage en réseau couvre les périphériques de stockage partagés sur un réseau à l'aide de protocoles standards. Ce type de stockage inclut les périphériques de stockage partagé utilisant les protocoles Fibre Channel, iSCSI, NFS, GFS2 et SCSI RDMA, et est une condition nécessaire pour effectuer des migrations d'invités virtualisés entre hôtes.



Important

Pour obtenir des informations complètes sur le déploiement et la configuration d'instances de stockage virtuel dans votre environnement, veuillez consulter le *Guide d'Administration et de Déploiement de Virtualisation* fourni par Red Hat.

25.2. DM-Multipath

Device Mapper Multipathing (DM-Multipath) est une fonctionnalité vous permettant de configurer de multiples chemins d'E/S entre les nœuds de serveurs et les arrays de stockage au sein d'un seul périphérique. Ces chemins d'E/S sont des connexions SAN physiques qui peuvent inclure des câbles, des interrupteurs et des contrôleurs séparés. Ils regroupent les chemins d'E/S, et créent un nouveau périphérique formé d'un agrégat de chemins multiples.

DM-Multipath est principalement utilisé pour les raisons suivantes :

Redondance

DM-Multipath peut amener à un échec dans une configuration active/passive. Dans une configuration active/passive, on utilise seulement la moitié des chemins à tout moment pour E/S. Si un élément (le câble, le commutateur, ou le contrôleur) d'un chemin E/S échoue, DM-Multipath passe à un chemin alternatif.

Performance améliorée

DM-Multipath peut être configuré dans un mode actif/actif, où E/S est étendu à travers les chemins en circuit cyclique. Dans certaines configurations, DM-Multipath peut détecter un chargement sur le chemin E/S et re-équilibrer le chargement de façon dynamique.



Important

Pour obtenir des informations complètes sur le déploiement et la configuration de DM-Multipath dans votre environnement, veuillez consulter le guide *Utilisation de DM-Multipath* fourni par Red Hat.

Chapitre 26. External Array Management (`libStorageMgmt`)

Red Hat Enterprise Linux 7 est fourni avec un nouveau package d'aire de stockage externe qui s'appelle `libStorageMgmt`.

26.1. Qu'est-ce que `libStorageMgmt` ?

Le package `libStorageMgmt` est une API (« Application Programming Interface ») indépendante de l'aire de stockage. Il fournit une API stable et consistante qui permet aux développeurs de gérer par programme différentes aires de stockage et de tirer profit des fonctionnalités accélérées par le matériel.

La bibliothèque est utilisée comme bloc de construction pour les outils et les applications de gestion de haut niveau. Les administrateurs de système peuvent aussi les utiliser comme outils pour gérer le stockage manuellement et automatiser les tâches de gestion de stockage par l'utilisation de scripts.

Le package `libStorageMgmt` permet les opérations comme :

- Lister les pools de stockage, les volumes, les groupes d'accès ou les systèmes de fichiers.
- Créer et supprimer les volumes, les groupes d'accès, les systèmes de fichiers, ou les exportations NFS.
- Donner et supprimer l'accès aux volumes, aux groupes d'accès, ou aux initiateurs.
- Répliquer les volumes avec des clichés, clônages ou copies.
- Créer et supprimer des groupes d'accès et modifier les membres d'un groupe.

Les ressources de serveur comme le CPU ou la bande passante d'inter connexion ne sont pas utilisées car les opérations sont effectuées sur la baie.

Le package fournit :

- Une API Python et C stables pour l'application client et les plug-in développeurs.
- Une interface en ligne de commande qui utilise la bibliothèque (`lsmcli`).
- Un démon qui exécute le plug-in (`lsmd`).
- Un plug-in de simulateur qui permet de tester les applications clientes (`sim`).
- Architecture de plug-in pour l'interface avec les baies de stockage.



Avertissement

Cette bibliothèque et ses outils associés ont la capacité de détruire toutes les données qui se trouvent sur les baies de stockage dont elle s'occupe. Il est fortement recommandé pour développer et tester des applications et des scripts avec le plug-in de simulateur de stockage de supprimer les erreurs de logique avant de travailler avec des systèmes de production. Il est également fortement encouragé de tester si possible des applications et des scripts sur du matériel réel de non-production avant de déployer en production.

Le package `libStorageMgmt` de Red Hat Enterprise Linux 7 ajoute une règle udev par défaut pour Unit Attention REPORTED LUNS DATA HAS CHANGED.

Quand un changement de configuration de stockage a eu lieu, un des codes de plusieurs Unit Attention ASC/ASCQ rapportent le changement. Un uevent est alors généré et re scanné automatiquement avec **sysfs**.

Le fichier `/lib/udev/rules.d/90-scsi-ua.rules` contient des exemples de règles pour énumérer des événements que le noyau peut générer.

La bibliothèque **libStorageMgmt** utilise une architecture en plug-in pour accommoder les différences entre les baies de stockage. Pour plus d'informations sur les plug-ins de **libStorageMgmt** et sur la façon de les rédiger, voir le *Guide du développeur* de Red Hat.

26.2. Terminologie de libStorageMgmt

Différents distributeurs de baies de stockage et standards de stockage utilisent une terminologie différente pour parler de la même fonctionnalité. La bibliothèque utilise la terminologie suivante.

Aire de stockage

Tout système de stockage qui fournit un accès blocs (FC, FCoE, iSCSI) ou fichiers comme Network Attached Storage (NAS).

Volume

Les aires de stockage de Storage Area Network (SAN) peuvent exposer un volume au Host Bus Adapter (HBA) sur un certain nombre de transports comme FC, iSCSI, ou FCoE. Le SE de l'hôte le traite comme des périphériques en bloc. Un volume peut être exposé à plusieurs baies de stockage si **multipath[2]** est activé.

Également connu sous le nom Logical Unit Number (LUN), StorageVolume avec terminologie SNIA ou disque virtuel.

Pool

Un groupe d'espaces de stockage. Les systèmes de fichiers et les volumes peuvent être créés à partir d'un pool. Les pools peuvent être créés à partir de disques, de volumes, ou d'autres pools. Un pool peut contenir des configurations de RAID ou de thin provisioning.

Également connu sous le nom StoragePool dans la terminologie SNIA.

Cliché

Une réplique des données figée dans le temps, en lecture seule et préservant de l'espace.

Également connu sous le nom du cliché en lecture-seule.

Cloner

Une réplique des données figée dans le temps, en permission lecture et écriture (rw), et préservant de l'espace.

Également connu sous le nom du cliché en lecture et écriture.

Copier

Une copie des données au niveau du bit. Occupe tout l'espace.

Miroir

Une copie continue sans cesse mise à jour (en mode synchrone ou asynchrone).

Access group

Ensemble d'initiateurs FCoE, FC ou iSCSI à qui est donné un accès à un ou plusieurs volumes de stockage. Cela garantit que seules les volumes de stockages sont accessibles aux initiateurs indiqués.

Également connu sous le nom de groupe d'initiateurs.

Access Grant

Exposer un volume à un initiateur ou à un groupe d'accès spécifique. La bibliothèque **libStorageMgmt** ne prend pas en charge actuellement le mappage LUN avec la possibilité de choisir un nombre d'inités logiques spécifiques. La bibliothèque **libStorageMgmt** permet à l'aire de stockage de sélectionner le prochain LUN pour la tâche. Si vous configurez un démarrage boot de SAN ou si vous masquez plus de 256 volumes, veuillez à lire les documents HBA, Storage Array, OS.

Grant access est également connu sous le nom Masquage LUN.

Système

Représente une aire de stockage ou un RAID de stockage attaché directement.

Système de fichiers

Une baie de stockage NAS (Network Attached Storage) peut exposer un système de fichiers pour héberger un SE via un réseau IP, par l'intermédiaire du protocole NFS ou CIFS. Le SE Hôte le traite comme point de montage ou comme dossier contenant des fichiers qui dépendent du système d'exploitation du client.

Disque

Le disque physique contient des données. Normalement utilisé quand on crée un pool avec des configurations RAID.

Également connu sous le nom DiskDrive dans la terminologie SNIA.

Initiateur

Avec Fibre Channel (FC) ou Fibre Channel over Ethernet (FCoE), L'initiateur est le World Wide Port Name (WWPN) ou World Wide Node Name (WWNN). Avec iSCSI, l'initiateur est le iSCSI Qualified Name (IQN) (nom complet). Avec NFS ou CIFS, l'initiateur est le nom de l'hôte ou l'adresse IP de l'hôte.

Dépendance

Certaines baies de stockage ont une relation implicite entre l'origine (système de fichier ou volume parent) et de l'enfant (comme un cliché ou un clone). Par exemple, il est impossible de supprimer le parent si il a un ou plusieurs dépendants (enfants). L'API fournit des méthodes pour déterminer si une telle relation existe et une méthode pour supprimer la dépendance en reproduisant les blocs requis.

26.3. Installation

Pour installer **libStorageMgmt** pour utilisation en ligne de commande, avec les bibliothèques et les plugs-in de simulateur requis, exécutez la commande suivante :

```
$ sudo yum install libstoragemgmt libstoragemgmt-python
```

Pour développer des applications C qui utilisent la bibliothèque, installer les packages **libstoragemgmt-devel** et **libstorage-debuginfo** en option, par la commande suivante :

```
$ sudo yum install libstoragemgmt-devel libstoragemgmt-debuginfo
```

Pour installer **libStorageMgmt** à utiliser dans les baies de stockage, sélectionner un ou plusieurs packages de plug-ins appropriés par la commande suivante :

```
$ sudo yum install libstoragemgmt-name-plugin
```

Les plug-ins suivants disponibles sont les suivants :

libstoragemgmt-smis-plugin

Support SMI-S Array standard

libstoragemgmt-netapp-plugin

Fichiers Support NetApp spécifique

libstoragemgmt-nstor-plugin

Support NexentaStor spécifique

libstoragemgmt-targetd-plugin

Support targetd spécifique

Le démon est ensuite installé et configuré pour pouvoir démarrer mais ne le fera pas avant le prochain démarrage. Pour l'utiliser immédiatement sans nouvel amorçage, démarrer le démon manuellement.

La gestion d'une baie de stockage nécessite la prise en charge via un plug-in. Le package d'installation de base comprend des plug-ins open source pour un certain nombre de fournisseurs. Des paquets de plug-in supplémentaires seront disponibles séparément au fur et à mesure que le support de stockage s'améliore. Le matériel pris en charge actuellement est en constante évolution. La page web du projet a les toutes dernières informations sur les baies de stockage et sur les fonctionnalités prises en charge sur chaque baie de stockage à l'adresse suivante http://libstoragemgmt.sourceforge.net/supported_hardware.html.

Le démon libStorageMgmt (**lsmgd**) se comporte comme n'importe quel service standard du système.

Pour vérifier le statut de libStorageMgmt, utiliser :

```
$ sudo systemctl status libstoragemgmt
```

Pour stopper le service, utiliser :

```
$ sudo systemctl stop libstoragemgmt
```

Pour démarrer le service, utiliser :

```
$ sudo systemctl start libstoragemgmt
```

26.4. Utilisation de libStorageMgmt

Pour utiliser **libStorageMgmt** de manière interactive, exécutez la commande **lsmcli**.

L'outil **lsmcli** a besoin de deux choses pour fonctionner :

- ✦ Un URI (Uniform Resource Identifier) à utiliser pour identifier le plug-in auquel se connecter dans l'aire de stockage et toutes les options de configuration possibles nécessaires.
- ✦ Un nom d'utilisateur et un mot de passe pour l'aire de stockage.

L'URI a la forme suivante :

`plugin+optional-transport://username@host:port/?query-string-parameters`

Chaque plug-in a des prérequis différents selon les besoins.

Exemple 26.1. Exemples de prérequis de plug-in

Plug-in de simulateur ne nécessitant ni nom d'utilisateur, ni mot de passe

`sim://`

Plug-in NetApp via SSL avec le root du nom d'utilisateur

`ontap+ssl://root@filer.company.com/`

SMI-S plug-in over SSL for EMC array

`smis+ssl://admin@provider.com:5989/?namespace=root/emc`

Pour des exemples supplémentaires, voir <https://sourceforge.net/p/libstoragemgmt/wiki/URISyntax/>.

Il y a trois options pour utiliser l'URI :

1. Passer l'URI comme partie de la commande.

```
$ lsmcli -u sim://...
```

2. Stocker l'URI dans une variable d'environnement.

```
$ export LSMCLI_URI=sim:// && lsmcli ...
```

3. Mettre l'URI dans le fichier `~/.lsmcli`, qui contient les paires nom-valeur séparées par "=". La seule configuration actuellement prise en charge est 'uri'.

Déterminer quel URI utiliser doit être fait dans cet ordre. Si les trois sont fournis, seul le premier qui figure sur la ligne de commande sera utilisé.

Fournir le mot de passe en indiquant `-P` sur la ligne de commande ou en la mettant dans la variable environnementale `LSMCLI_PASSWORD`.

Exemple 26.2. Exemples de lsmcli

Exemple d'utilisation en ligne de commande pour créer un nouveau volume et le rendre visible à un initiateur.

Lister les aires de stockages mises en service par cette connexion.

```
$ lsmcli list --type SYSTEMS
ID      | Name                               | Status
-----+-----+-----
sim-01 | LSM simulated storage plug-in | OK
```

Lister les pools de stockage.

```
$ lsmcli list --type POOLS -H
ID  | Name          | Total space          | Free space          |
System ID
-----+-----+-----+-----+----
-----
P002 | Pool 2       | 18446744073709551616 | 18446744073709551616 | sim-
01
P003 | Pool 3       | 18446744073709551616 | 18446744073709551616 | sim-
01
P001 | Pool 1       | 18446744073709551616 | 18446744073709551616 | sim-
01
P004 | lsm_test_aggr | 18446744073709551616 | 18446744073709551616 | sim-01
```

Créer un volume.

```
$ lsmcli volume-create --name volume_name --size 20G --pool P001 -H
ID  | Name          | vpd83          | bs  | #blocks  |
status | ...
-----+-----+-----+-----+-----
+-----+-----
Vol1 | volume_name | F7DDF7CA945C66238F593BC38137BD2F | 512 | 41943040 | OK
| ...
```

Créer un groupe d'accès avec un initiateur à l'intérieur.

```
$ lsmcli --create-access-group example_ag --id iqn.1994-
05.com.domain:01.89bd01 --type ISCSI --system sim-01
ID                               | Name          | Initiator ID
| SystemID
-----+-----+-----
-----+-----
782d00c8ac63819d6cca7069282e03a0 | example_ag | iqn.1994-
05.com.domain:01.89bd01 | sim-01
```

Créer un groupe d'accès avec un initiateur à l'intérieur.

```
$ lsmcli access-group-create --name example_ag --init iqn.1994-
05.com.domain:01.89bd01 --init-type ISCSI --sys sim-01
ID                               | Name          | Initiator IDs
| System ID
-----+-----+-----
-----+-----
782d00c8ac63819d6cca7069282e03a0 | example_ag | iqn.1994-
05.com.domain:01.89bd01 | sim-01
```

Autoriser la visibilité du groupe d'accès au volume nouvellement créé :


```
$ lsmcli access-group-grant --ag 782d00c8ac63819d6cca7069282e03a0 --vol  
Vol1 --access RW
```

Le design de la bibliothèque prévoit une séparation des processus entre le client et le plug-in au moyen d'un *IPC communication inter processus*. Cela empêche des bugs du plug-in de faire échouer l'application cliente. Il fournit également un moyen pour les rédacteurs de plug-in d'écrire des plug-ins avec une licence de leur choix. Lorsqu'un client ouvre la bibliothèque en passant un URI, la bibliothèque cliente examine l'URI pour déterminer quel plug-in doit être utilisé.

Les plug-ins sont techniquement des applications autonomes, mais sont conçus pour avoir un descripteur de fichier qui leur sont attribués par la ligne de commande. La bibliothèque cliente ouvre ensuite le socket de domaine Unix approprié qui entraîne le démon bifurquer et exécuter le plug-in. Cela donne à la bibliothèque cliente un canal de communication point à point avec le plug-in. Le démon peut être redémarré sans affecter les clients existants. Tant que le client a la bibliothèque ouverte pour que plug-in, le processus de plug-in sera en cours d'exécution. Après qu'une ou plusieurs commandes soient exécutées, le plug-in est fermé, le processus de plug-in est nettoyé, puis se termine.

Pour obtenir un diagramme séquentiel sur ce processus, voir <https://sourceforge.net/p/libstoragemgmt/wiki/Architecture/>.

Le comportement par défaut de lsmcli est d'attendre que l'opération soit complétée. Selon les opérations demandées, cela peut durer plusieurs heures. Pour permettre un retour aux conditions normales d'utilisation, il est possible d'utiliser l'option **-b** en ligne de commande. Si le code de sortie vaut 0, la commande est exécutée. Si le code de sortie est de 7, la commande est en cours et un identificateur de tâche s'inscrit sur la sortie standard. L'utilisateur ou le script peut alors prendre l'ID de la tâche et interroger l'état de la commande au besoin à l'aide de **lsmcli --jobstatus JobID**. Si la tâche est terminée, la valeur de sortie sera 0 et les résultats seront inscrits dans la sortie standard. Si la commande est toujours en cours, la valeur de retour sera le 7 et le pourcentage complet sera imprimé sur la sortie standard.

Exemple 26.3. Exemple asynchrone

Créer un volume en passant l'option **-b** pour que la commande renvoie immédiatement.

```
$ lsmcli volume-create --name async_created --size 20G --pool P001 -b  
JOB_3
```

Vérifier la valeur de sortie, et souvenez-vous qu'un 7 indique que la tâche est toujours en cours.

```
$ echo $?  
7
```

Vérifier si la tâche est terminée.

```
$ lsmcli job-status --job JOB_3  
33
```

Vérifiez la valeur de sortie, et souvenez-vous que 7 indique que la tâche est toujours en cours, donc la sortie standard est le pourcentage effectué ou 33% de l'écran ci-dessus.

```
$ echo $?  
7
```

Attendez un peu et vérifiez à nouveau, et souvenez-vous bien qu'une sortie correspondant à 0 indique que cela a réussi et standard out affichera le nouveau volume.

```
$ lsmcli job-status --job JOB_3
ID   | Name                | vpd83                | Block Size |
...
-----+-----+-----+-----+
+-----
Vol2 | async_created      | 855C9BA51991B0CC122A3791996F6B15 | 512        |
...
```

Pour les scripts, passer l'option **-t SeparatorCharacters**. Cela facilitera l'analyse de la sortie.

Exemple 26.4. Exemples de script

```
$ lsmcli list --type volumes -t#
Vol1#volume_name#049167B5D09EC0A173E92A63F6C3EA2A#512#41943040#21474836480
#OK#sim-01#P001
Vol2#async_created#3E771A2E807F68A32FA5E15C235B60CC#512#41943040#214748364
80#OK#sim-01#P001
```

```
$ lsmcli list --type volumes -t " | "
Vol1 | volume_name | 049167B5D09EC0A173E92A63F6C3EA2A | 512 | 41943040 |
21474836480 | OK | 21474836480 | sim-01 | P001
Vol2 | async_created | 3E771A2E807F68A32FA5E15C235B60CC | 512 | 41943040 |
21474836480 | OK | sim-01 | P001
```

```
$ lsmcli list --type volumes -s
-----
ID          | Vol1
Name        | volume_name
VPD83       | 049167B5D09EC0A173E92A63F6C3EA2A
Block Size  | 512
#blocks     | 41943040
Size        | 21474836480
Status      | OK
System ID   | sim-01
Pool ID     | P001
-----
ID          | Vol2
Name        | async_created
VPD83       | 3E771A2E807F68A32FA5E15C235B60CC
Block Size  | 512
#blocks     | 41943040
Size        | 21474836480
Status      | OK
System ID   | sim-01
Pool ID     | P001
-----
```

Il est conseillé d'utiliser la bibliothèque Python pour les scripts difficiles.

Pour plus d'informations sur **lsmcli**, voir les pages man ou la commande **lsmcli --help**.

26.5. Documentation libstorageMgmt

Pour plus d'informations, consulter les site suivants :

- Site Web officiel : <https://sourceforge.net/projects/libstoragemgmt/>.
- Projet wiki : <https://sourceforge.net/p/libstoragemgmt/wiki/Home/>.

Annexe A. Historique des versions

Version 3-68.2	Tue Jul 4 2017	Terry Chuang
Fichiers de traduction synchronisés avec les sources XML 3-68		
Version 3-68.1	Tue Jul 4 2017	Terry Chuang
Fichiers de traduction synchronisés avec les sources XML 3-68		
Version 3-68	Fri Oct 21 2016	Milan Navratil
Version pour la distribution GA 7.3.		
Version 3-67	Fri Jun 17 2016	Milan Navratil
Mise à jour asynchrone		
Version 3-64	Wed Nov 11 2015	Jana Heves
Distribution 7.2 GA.		
Version 3-33	Wed Feb 18 2015	Jacquelynn East
Version pour 7.1 GA.		
Version 3-26	Wed Jan 21 2015	Jacquelynn East
Aperçu de Ceph ajouté		
Version 3-22	Thu Dec 4 2014	Jacquelynn East
7.1 Beta		
Version 3-4	Thu Jul 17 2014	Jacquelynn East
Ajout d'un nouveau chapitre à targetcli		
Version 3-1	Tue Jun 3 2014	Jacquelynn East
Version pour la distribution GA 7.0		

Index

Symboles

, quotas de disque

- gestion de

- la commande quotacheck, utilisée pour vérifier, [Contrôler l'exactitude des quotas](#)

/dev/shm, [Collecte des informations sur les systèmes de fichiers](#)

/etc/fstab, [Conversion vers un système de fichiers ext3](#), [Monter des systèmes de fichiers NFS à l'aide de /etc/fstab](#), [Monter un système de fichiers](#)

/local/directory (configuration du client, montage)

- NFS, [Configuration du client NFS](#)

/proc

- /proc/devices, [Système de fichiers virtuel /proc](#)

- /proc/filesystems, [Système de fichiers virtuel /proc](#)

- /proc/mdstat, [Système de fichiers virtuel /proc](#)

- /proc/mounts, [Système de fichiers virtuel /proc](#)
- /proc/mounts/, [Système de fichiers virtuel /proc](#)
- /proc/partitions, [Système de fichiers virtuel /proc](#)

/proc/devices

- système de fichiers virtuel (/proc), [Système de fichiers virtuel /proc](#)

/proc/filesystems

- système de fichiers virtuel (/proc), [Système de fichiers virtuel /proc](#)

/proc/mdstat

- système de fichiers virtuel (/proc), [Système de fichiers virtuel /proc](#)

/proc/mounts

- système de fichiers virtuel (/proc), [Système de fichiers virtuel /proc](#)

/proc/mounts/

- système de fichiers virtuel (/proc), [Système de fichiers virtuel /proc](#)

/proc/partitions

- système de fichiers virtuel (/proc), [Système de fichiers virtuel /proc](#)

/remote/export (configuration du client, montage)

- NFS, [Configuration du client NFS](#)

A

accès à l'espace utilisateur

- alignement et taille des E/S, [Accès à l'espace utilisateur](#)

activer/désactiver

- barrières d'écriture, [Activer/désactiver les barrières d'écriture](#)

afficher les configurations iface disponibles

- déchargement et liaison d'interfaces
 - iSCSI, [Afficher les configurations iface disponibles](#)

ajouter des chemins à un périphérique de stockage, [Ajouter un périphérique ou un chemin de stockage](#)

ajouter ou supprimer

- LUN (numéro d'unité logique), [Ajouter ou supprimer une unité logique avec rescan-scsi-bus.sh](#)

Alignement et taille des E/S, [Alignement et taille des E/S de stockage](#)

- empiler les paramètres d'E/S, [Empiler les paramètres d'E/S](#)
- paramètres d'accès au stockage, [Paramètres d'accès au stockage](#)
- Pile d'E/S Linux, [Alignement et taille des E/S de stockage](#)

alignement et taille des E/S

- accès à l'espace utilisateur, [Accès à l'espace utilisateur](#)
- interface sysfs (accès à l'espace utilisateur), [Interface sysfs](#)
- ioctls du périphérique bloc (accès à l'espace utilisateur), [ioctls du périphérique bloc](#)
- logical_block_size, [Accès à l'espace utilisateur](#)
- LVM, [Gestionnaire de volumes logiques LVM](#)

- outils (pour le partitionnement et autres fonctions des systèmes de fichiers), [Outils des partitions et systèmes de fichiers](#)
- READ CAPACITY(16), [SCSI](#)
- standards ATA, [ATA](#)
- standards SCSI, [SCSI](#)

allocation features

- XFS, [Le système de fichiers XFS](#)

aperçu

- stockage en ligne, [Online Storage Management](#)

aperçu général, [Aperçu](#)

API Fibre Channel, [Interface de programmation Fibre Channel](#)

API iSCSI, [API iSCSI](#)

API, Fibre Channel, [Interface de programmation Fibre Channel](#)

API, iSCSI, [API iSCSI](#)

augmenter la taille d'un système de fichiers

- XFS, [Augmenter la taille d'un système de fichiers XFS](#)

autofs , [autofs](#), [Configuration autofs](#)

- (voir aussi NFS)

autofs version 5

- NFS, [Améliorations de autofs Version 5 par rapport à la Version 4](#)

autres utilitaires du système de fichiers

- ext4, [Autres utilitaires du système de fichiers Ext4](#)

B

backend de cache

- FS-Cache, [FS-Cache](#)

barrières d'écriture

- activer/désactiver, [Activer/désactiver les barrières d'écriture](#)
- caches d'écriture avec batteries de secours, [Caches d'écriture avec batteries de secours](#)
- comment les barrières d'écriture fonctionnent, [Comment les barrières d'écriture fonctionnent](#)
- définition, [Barrières d'écriture](#)
- désactiver les caches d'écriture, [Désactiver les caches d'écriture](#)
- ext4, [Monter un système de fichiers Ext4](#)
- importance des barrières d'écriture, [Importance des barrières d'écriture](#)
- matrices haut de gamme, [Matrices haut de gamme](#)
- messages d'erreur, [Activer/désactiver les barrières d'écriture](#)
- NFS, [NFS](#)
- XFS, [Barrières d'écriture](#)

bcull (cache cull limits settings)

- FS-Cache, [Définir les limites d'élimination du cache](#)

besoins de stockage à prendre en compte pendant l'installation

- ce qui est nouveau, [Besoins de stockage à prendre en compte pendant l'installation](#)
- mises à jour, [Besoins de stockage à prendre en compte pendant l'installation](#)

besoins du stockage à prendre en compte pendant l'installation

- channel command word (CCW), [Périphériques DASD et zFCP sur IBM System Z](#)
- configuration et détection iSCSI, [Détection et configuration iSCSI](#)
- LUKS/dm-crypt, chiffrer des périphériques blocs à l'aide de, [Chiffrer des périphériques blocs avec LUKS](#)
- métadonnées RAID BIOS périmées, [Métadonnées RAID BIOS périmées](#)
- partitions séparées (pour /home, /opt, /usr/local), [Partitions séparées pour /home, /opt, /usr/local](#)
- périphériques bloc activés DIF/DIX, [Périphériques bloc avec DIF/DIX activé](#)
- périphériques DASD et zFCP sur IBM System Z, [Périphériques DASD et zFCP sur IBM System Z](#)

brun (cache cull limits settings)

- FS-Cache, [Définir les limites d'élimination du cache](#)

bstop (cache cull limits settings)

- FS-Cache, [Définir les limites d'élimination du cache](#)

Btrfs

- Système de fichiers, [Btrfs \(Aperçu technologique\)](#)

C**cachefiles**

- FS-Cache, [FS-Cache](#)

cachefilesd

- FS-Cache, [Paramétrer un cache](#)

caches d'écriture avec batteries de secours

- barrières d'écriture, [Caches d'écriture avec batteries de secours](#)

CCW, channel command word

- besoins du stockage à prendre en compte pendant l'installation, [Périphériques DASD et zFCP sur IBM System Z](#)

ce qui est nouveau

- besoins de stockage à prendre en compte pendant l'installation, [Besoins de stockage à prendre en compte pendant l'installation](#)

channel command word (CCW)

- besoins du stockage à prendre en compte pendant l'installation, [Périphériques DASD et zFCP sur IBM System Z](#)

chemin vers des périphériques de stockage, supprimer, [Supprimer un chemin vers un périphérique de stockage](#)**chemin vers les périphériques de stockage, ajouter, [Ajouter un périphérique ou un chemin de stockage](#)****cibles**

- iSCSI, [Connexion à une cible iSCSI](#)

classes de débit

- solid-state disks, [Directives de déploiement des disques SSD](#)

clés d'indexation

- FS-Cache, [FS-Cache](#)

commande TRIM

- solid-state disks, [Directives de déploiement des disques SSD](#)

commandes

- volume_key, [Commandes volume_key](#)

comment les barrières d'écriture fonctionnent

- barrières d'écriture, [Comment les barrières d'écriture fonctionnent](#)

configuration

- découverte
 - iSCSI, [Configuration de la découverte iSCSI](#)

configuration d'ensembles RAID

- RAID, [Configuring RAID Sets](#)

configuration d'un service tftp pour des clients sans disque

- systèmes sans disque, [Configuration d'un service tftp pour des clients sans disque](#)

configuration DHCP pour des clients sans disque

- systèmes sans disque, [Configuration DHCP pour les clients sans disque](#)

configuration et détection iSCSI

- besoins du stockage à prendre en compte pendant l'installation, [Détection et configuration iSCSI](#)

configurations des stockages d'installation

- ce qui est nouveau, [Besoins de stockage à prendre en compte pendant l'installation](#)
- channel command word (CCW), [Périphériques DASD et zFCP sur IBM System Z](#)
- configuration et détection iSCSI, [Détection et configuration iSCSI](#)
- LUKS/dm-crypt, chiffrer des périphériques blocs à l'aide de, [Chiffrer des périphériques blocs avec LUKS](#)
- métadonnées RAID BIOS périmées, [Métadonnées RAID BIOS périmées](#)
- mises à jour, [Besoins de stockage à prendre en compte pendant l'installation](#)
- partitions séparées (pour /home, /opt, /usr/local), [Partitions séparées pour /home, /opt, /usr/local](#)
- périphériques bloc activés DIF/DIX, [Périphériques bloc avec DIF/DIX activé](#)
- périphériques DASD et zFCP sur IBM System Z, [Périphériques DASD et zFCP sur IBM System Z](#)

configurations iface

- déchargement et liaison d'interfaces
 - iSCSI, [Afficher les configurations iface disponibles](#)

configurations iface, affichage

- déchargement et liaison d'interfaces
 - iSCSI, [Afficher les configurations iface disponibles](#)

configurer une interface Ethernet pour utiliser FCoE

- FCoE, [Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\)](#)

connexion aux

- cibles iSCSI, [Connexion à une cible iSCSI](#)

contrôle des informations statistiques

- FS-Cache, [Informations statistiques](#)

contrôle du minuteur de commande SCSI et du statut de périphérique

- couche Linux SCSI, [Contrôle du Minuteur de commande SCSI et du Statut de périphérique](#)

création

- XFS, [Créer un système de fichiers XFS](#)

création de périphériques RAID avancée

- RAID, [Création de périphériques RAID avancée](#)

créer

- ext4, [Créer un système de fichiers Ext4](#)

D

debugfs (autres utilitaires du système de fichiers ext4)

- ext4, [Autres utilitaires du système de fichiers Ext4](#)

déchargement et liaison d'interfaces

- iSCSI, [Configurer le déchargement et la liaison d'interfaces iSCSI](#)

découverte

- iSCSI, [Configuration de la découverte iSCSI](#)

délai d'expiration pour une session particulière, configuration

- configuration iSCSI, [Configurer des délais d'expiration pour une session particulière](#)

démonter, [Démonter un système de fichiers](#)

dénomination persistante, [Dénomination persistante](#)

déplacer un point de montage, [Déplacer un point de montage](#)

déploiement

- solid-state disks, [Considérations pour le déploiement](#)

désactivation de NOP-Outs

- configuration iSCSI, [Root iSCSI](#)

désactiver les caches d'écriture

- barrières d'écriture, [Désactiver les caches d'écriture](#)

déterminer les états des ports distants

- Fibre Channel
 - modifier le comportement de la perte de lien, [Fibre Channel](#)

device-mapper multipathing, [DM-Multipath](#)

dev_loss_tmo

- Fibre Channel
 - modifier le comportement de la perte de lien, [Fibre Channel](#)
- Fibre Channel API, [Interface de programmation Fibre Channel](#)

dev_loss_tmo, modifier

- Fibre Channel
- modifier le comportement de la perte de lien, [Fibre Channel](#)

df, [Collecte des informations sur les systèmes de fichiers](#)**DHCP, configuration**

- systèmes sans disque, [Configuration DHCP pour les clients sans disque](#)

dhcpd.leases, [Collecte des informations sur les systèmes de fichiers](#)**direct map support (autofs version 5)**

- NFS, [Améliorations de autofs Version 5 par rapport à la Version 4](#)

directives de déploiement

- solid-state disks, [Directives de déploiement des disques SSD](#)

directories

- /boot/, [Répertoire /boot/](#)

dm-multipath

- configuration iSCSI, [Paramètres iSCSI avec dm-multipath](#)

dmraid

- RAID, [dmraid](#)

dmraid (configuration d'ensembles RAID)

- RAID, [dmraid](#)

données de cohérence

- FS-Cache, [FS-Cache](#)

du, [Collecte des informations sur les systèmes de fichiers](#)**E****e2fsck, [Rétablir un système de fichiers Ext2](#)****e2image (autres utilitaires du système de fichiers ext4)**

- ext4, [Autres utilitaires du système de fichiers Ext4](#)

e2label

- ext4, [Autres utilitaires du système de fichiers Ext4](#)

e2label (autres utilitaires du système de fichiers ext4)

- ext4, [Autres utilitaires du système de fichiers Ext4](#)

empiler les paramètres d'E/S

- Alignement et taille des E/S, [Empiler les paramètres d'E/S](#)
- alignement et taille des E/S, [Empiler les paramètres d'E/S](#)

Emplacement des fichiers spécifiques à Red Hat Enterprise Linux

- /etc/sysconfig/, [Emplacement des fichiers Red Hat Enterprise Linux spéciaux](#)
- (voir aussi Répertoire sysconfig)
- /var/cache/yum, [Emplacement des fichiers Red Hat Enterprise Linux spéciaux](#)
- /var/lib/rpm/, [Emplacement des fichiers Red Hat Enterprise Linux spéciaux](#)

enhanced LDAP support (autofs version 5)

- NFS, [Améliorations de autofs Version 5 par rapport à la Version 4](#)

entrelacement

- notions fondamentales RAID, [Réseau redondant de disques indépendants \(RAID, de l'anglais « Redundant Array of Independent Disks »\)](#)
- RAID, [Niveaux RAID et prise en charge linéaire](#)

espace swap, [Espace swap](#)

- création, [Ajouter de l'espace swap](#)
- déplacement, [Déplacer l'espace swap](#)
- expansion, [Ajouter de l'espace swap](#)
- fichier
 - création, [Création d'un fichier Swap](#)
- LVM2
 - créer, [Création d'un volume logique LVM2 avec Swap](#)
 - étendre, [Étendre Swap dans un Volume logique LVM2](#)
 - réduire, [Réduire un Swap dans un Volume logique LVM2](#)
 - supprimer, [Supprimer un volume logique LVM2 avec Swap](#)
- suppression, [Supprimer de l'espace swap](#)
- taille recommandée, [Espace swap](#)

états des ports (distants), déterminer

- Fibre Channel
 - modifier le comportement de la perte de lien, [Fibre Channel](#)

états des ports distants, déterminer

- Fibre Channel
 - modifier le comportement de la perte de lien, [Fibre Channel](#)

ext2

- rétablir à partir d'ext3, [Rétablir un système de fichiers Ext2](#)

ext3

- conversion à partir d'ext2, [Conversion vers un système de fichiers ext3](#)
- création, [Créer un système de fichiers ext3](#)
- fonctionnalités, [Système de fichiers Ext3](#)

ext4

- autres utilitaires du système de fichiers, [Autres utilitaires du système de fichiers Ext4](#)
- barrières d'écriture, [Monter un système de fichiers Ext4](#)
- créer, [Créer un système de fichiers Ext4](#)
- debugfs (autres utilitaires du système de fichiers ext4), [Autres utilitaires du système de fichiers Ext4](#)
- e2image (autres utilitaires du système de fichiers ext4), [Autres utilitaires du système de fichiers Ext4](#)
- e2label, [Autres utilitaires du système de fichiers Ext4](#)
- e2label (autres utilitaires du système de fichiers ext4), [Autres utilitaires du système de fichiers Ext4](#)
- fonctionnalités d'allocation, [Le système de fichiers Ext4](#)
- fonctionnalités principales, [Le système de fichiers Ext4](#)
- fsync(), [Le système de fichiers Ext4](#)
- géométrie d'entrelacement, [Créer un système de fichiers Ext4](#)

- largeur d'entrelacement (spécifier la géométrie d'entrelacement), [Créer un système de fichiers Ext4](#)
- mkfs.ext4, [Créer un système de fichiers Ext4](#)
- monter, [Monter un système de fichiers Ext4](#)
- option de montage « nobarrier », [Monter un système de fichiers Ext4](#)
- quota (autres utilitaires du système de fichiers ext4), [Autres utilitaires du système de fichiers Ext4](#)
- redimensionner, [Redimensionner un système de fichiers Ext4](#)
- resize2fs (redimensionner ext4), [Redimensionner un système de fichiers Ext4](#)
- stride (spécifier la géométrie d'entrelacement), [Créer un système de fichiers Ext4](#)
- tune2fs (montage), [Monter un système de fichiers Ext4](#)
- types de systèmes de fichiers, [Le système de fichiers Ext4](#)

F

fast_io_fail_tmo

- Fibre Channel API, [Interface de programmation Fibre Channel](#)

FCoE

- configurer une interface Ethernet pour utiliser FCoE, [Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\)](#)
- Fibre Channel over ethernet, [Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\)](#)
- paquets requis, [Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\)](#)

FHS, [Vue d'ensemble du standard de hiérarchie des systèmes de fichiers \(FHS, ou « Filesystem Hierarchy Standard »\)](#), [Organisation FHS](#)

- (voir aussi système de fichiers)

Fibre Channel

- stockage en ligne, [Fibre Channel](#)

Fibre Channel over Ethernet

- FCoE, [Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\)](#)

Fichier /etc/fstab

- activer les quotas de disques avec, [Activer les quotas](#)

fichiers API espace utilisateur

- API Fibre Channel, [Interface de programmation Fibre Channel](#)

findmnt (commande)

- répertorier les montages, [Répertorier les systèmes de fichiers actuellement montés](#)

fonctionnalités d'allocation

- ext4, [Le système de fichiers Ext4](#)

fonctionnalités principales

- ext4, [Le système de fichiers Ext4](#)
- XFS, [Le système de fichiers XFS](#)

FS-Cache

- backend de cache, [FS-Cache](#)
- bcull (cache cull limits settings), [Définir les limites d'élimination du cache](#)
- brun (cache cull limits settings), [Définir les limites d'élimination du cache](#)
- bstop (cache cull limits settings), [Définir les limites d'élimination du cache](#)

- cachefiles, [FS-Cache](#)
- cachefilesd, [Paramétrer un cache](#)
- clés d'indexation, [FS-Cache](#)
- données de cohérence, [FS-Cache](#)
- garantie des performances, [Garantie des performances](#)
- informations statistiques (contrôle), [Informations statistiques](#)
- limites d'élimination du cache, [Définir les limites d'élimination du cache](#)
- NFS (limitations de cache avec), [Limitations des caches avec NFS](#)
- NFS (utiliser avec), [Utiliser le cache avec NFS](#)
- paramétrage d'un cache, [Paramétrer un cache](#)
- partage de cache, [Partage de cache](#)
- tune2fs (définir un cache), [Paramétrer un cache](#)

fsync()

- ext4, [Le système de fichiers Ext4](#)
- XFS, [Le système de fichiers XFS](#)

G

garantie des performances

- FS-Cache, [Garantie des performances](#)

géométrie d'entrelacement

- ext4, [Créer un système de fichiers Ext4](#)

gestion des quotas

- XFS, [Gestion des quotas XFS](#)

GFS2

- gfs2.ko, [Global File System 2](#)
- taille maximale, [Global File System 2](#)
- types de systèmes de fichiers, [Global File System 2](#)

gfs2.ko

- GFS2, [Global File System 2](#)

Global File System 2

- gfs2.ko, [Global File System 2](#)
- taille maximale, [Global File System 2](#)
- types de systèmes de fichiers, [Global File System 2](#)

gquota/gqnoenforce

- XFS, [Gestion des quotas XFS](#)

H

hiérarchie, système de fichiers, [Vue d'ensemble du standard de hiérarchie des systèmes de fichiers \(FHS, ou « Filesystem Hierarchy Standard »\)](#)

host

- Fibre Channel API, [Interface de programmation Fibre Channel](#)

I

iface (configuration pour déchargement iSCSI)

- déchargement et liaison d'interfaces
 - iSCSI, [Configurer un iface pour le déchargement iSCSI](#)

iface pour iSCSI logiciel

- téléchargement et liaison d'interfaces
 - iSCSI, [Configurer un iface pour iSCSI logiciel](#)

implémentations d'initiateur

- téléchargement et liaison d'interfaces
 - iSCSI, [Afficher les configurations iface disponibles](#)

importance des barrières d'écriture

- barrières d'écriture, [Importance des barrières d'écriture](#)

information système

- systèmes de fichiers
 - /dev/shm, [Collecte des informations sur les systèmes de fichiers](#)

informations statistiques (contrôle)

- FS-Cache, [Informations statistiques](#)

informations système

- systèmes de fichiers, [Collecte des informations sur les systèmes de fichiers](#)

interconnexions (scanner)

- iSCSI, [Scanner les interconnexions iSCSI](#)

interconnexions du stockage, scanner, [Scanner les interconnexions du stockage](#)**interface sysfs (accès à l'espace utilisateur)**

- alignement et taille des E/S, [Interface sysfs](#)

introduction, [Aperçu](#)**ioctl du périphérique bloc (accès à l'espace utilisateur)**

- alignement et taille des E/S, [ioctl du périphérique bloc](#)

iSCSI

- cibles, [Connexion à une cible iSCSI](#)
 - connexion aux, [Connexion à une cible iSCSI](#)
- téléchargement et liaison d'interface
 - afficher les configurations iface disponibles, [Afficher les configurations iface disponibles](#)
- téléchargement et liaison d'interfaces, [Configurer le téléchargement et la liaison d'interfaces iSCSI](#)
 - Configurations iface, [Afficher les configurations iface disponibles](#)
 - configurations iface, affichage, [Afficher les configurations iface disponibles](#)
 - iface (configuration pour téléchargement iSCSI), [Configurer un iface pour le téléchargement iSCSI](#)
 - iface pour iSCSI logiciel, [Configurer un iface pour iSCSI logiciel](#)
 - implémentations d'initiateur, [Afficher les configurations iface disponibles](#)
 - iSCSI logiciel, [Configurer un iface pour iSCSI logiciel](#)
 - lier ou délier un iface sur un portail, [Lier ou délier un iface sur un portail](#)
- découverte, [Configuration de la découverte iSCSI](#)
 - configuration, [Configuration de la découverte iSCSI](#)
 - types d'enregistrements, [Configuration de la découverte iSCSI](#)

- iSCSI logiciel, [Configurer un iface pour iSCSI logiciel](#)
- scanner les interconnexions, [Scanner les interconnexions iSCSI](#)

iSCSI logiciel

- téléchargement et liaison d'interfaces
 - iSCSI, [Configurer un iface pour iSCSI logiciel](#)
- iSCSI, [Configurer un iface pour iSCSI logiciel](#)

issue_lip

- Fibre Channel API, [Interface de programmation Fibre Channel](#)

J

journaux endommagés (réparation de systèmes de fichiers XFS)

- XFS, [Réparer un système de fichiers XFS](#)

L

la commande quotacheck

- vérifier l'exactitude des quotas avec, [Contrôler l'exactitude des quotas](#)

largeur d'entrelacement (spécifier la géométrie d'entrelacement)

- ext4, [Créer un système de fichiers Ext4](#)

lazy mount/unmount support (autofs version 5)

- NFS, [Améliorations de autofs Version 5 par rapport à la Version 4](#)

les caches d'écriture, désactiver

- barrières d'écriture, [Désactiver les caches d'écriture](#)

lier ou délier un iface

- téléchargement et liaison d'interfaces
 - iSCSI, [Lier ou délier un iface sur un portail](#)

lier ou délier un iface sur un portail

- téléchargement et liaison d'interfaces
 - iSCSI, [Lier ou délier un iface sur un portail](#)

limitations de cache avec NFS

- FS-Cache, [Limitations des caches avec NFS](#)

limite (mode xfs_quota expert)

- XFS, [Gestion des quotas XFS](#)

limites d'élimination du cache

- FS-Cache, [Définir les limites d'élimination du cache](#)

limites de projets (paramétrage)

- XFS, [Paramétrer des limites de projets](#)

logical_block_size

- alignement et taille des E/S, [Accès à l'espace utilisateur](#)

LUKS/dm-crypt, chiffrer des périphériques blocs à l'aide de

- besoins du stockage à prendre en compte pendant l'installation, [Chiffrer des périphériques blocs avec LUKS](#)

LUN (numéro d'unité logique)

- ajouter ou supprimer, [Ajouter ou supprimer une unité logique avec rescan-scsi-bus.sh](#)
 - paquets requis, [Ajouter ou supprimer une unité logique avec rescan-scsi-bus.sh](#)
 - problèmes connus, [Problèmes connus avec rescan-scsi-bus.sh](#)
 - rescan-scsi-bus.sh, [Ajouter ou supprimer une unité logique avec rescan-scsi-bus.sh](#)

LVM

- alignement et taille des E/S, [Gestionnaire de volumes logiques LVM](#)

M

matrices haut de gamme

- barrières d'écriture, [Matrices haut de gamme](#)

mdadm (configuration d'ensembles RAID)

- RAID, [mdadm](#)

mdraid

- RAID, [mdraid](#)

mémoire virtuelle (tuning)

- solid-state disks, [Mémoire virtuelle](#)

messages d'erreur

- barrières d'écriture, [Activer/désactiver les barrières d'écriture](#)

métadonnées RAID BIOS périmées

- besoins du stockage à prendre en compte pendant l'installation, [Métadonnées RAID BIOS périmées](#)

minuteur de commande (SCSI)

- couche Linux SCSI, [Minuteur de commande](#)

minuteur de commande SCSI

- couche Linux SCSI, [Minuteur de commande](#)

mise en miroir

- RAID, [Niveaux RAID et prise en charge linéaire](#)

mises à jour

- besoins de stockage à prendre en compte pendant l'installation, [Besoins de stockage à prendre en compte pendant l'installation](#)

mkfs , [Formatage et étiquetage de la partition](#)

mkfs.ext4

- ext4, [Créer un système de fichiers Ext4](#)

mkfs.xfs

- XFS, [Créer un système de fichiers XFS](#)

mkpart , [Créer la partition](#)

mode cumulatif (xfsrestore)

- XFS, [Mode cumulatif de xfsrestore](#)

mode expert (xfs_quota)

- XFS, [Gestion des quotas XFS](#)

mode simple (xfsrestore)

- XFS, [Mode simple de xfsrestore](#)

modifier dev_loss_tmo

- Fibre Channel
 - modifier le comportement de la perte de lien, [Fibre Channel](#)

Modifier l'état de lecture/écriture

- Unités logiques en ligne, [Modifier l'état de lecture/écriture d'une unité logique en ligne](#)

modifier le comportement de la perte de lien, [Modifier le comportement de la perte de lien](#)

- Fibre Channel, [Fibre Channel](#)

montage

- XFS, [Monter un système de fichiers XFS](#)

montage (configuration du client)

- NFS, [Configuration du client NFS](#)

monter

- ext4, [Monter un système de fichiers Ext4](#)

mount (command), [Utilisation de la commande mount](#)

- déplacer un point de montage, [Déplacer un point de montage](#)
- monter un système de fichiers, [Monter un système de fichiers](#)
- options, [Spécifier les options de montage](#)
- répertorier les montages, [Répertorier les systèmes de fichiers actuellement montés](#)
- shared subtrees
 - private mount, [Partager des montages](#)
 - shared mount, [Partager des montages](#)
 - slave mount, [Partager des montages](#)
 - unbindable mount, [Partager des montages](#)
- sous-arborescences partagées, [Partager des montages](#)

mounting, [Monter un système de fichiers](#)**multiple master map entries per autofs mount point (autofs version 5)**

- NFS, [Améliorations de autofs Version 5 par rapport à la Version 4](#)

N**Network File System (voir NFS)****NFS**

- /etc/fstab , [Monter des systèmes de fichiers NFS à l'aide de /etc/fstab](#)
- /local/directory (configuration du client, montage), [Configuration du client NFS](#)
- /remote/export (configuration du client, montage), [Configuration du client NFS](#)
- arrêt, [Démarrage et arrêt NFS](#)
- autofs
 - augmenter, [Remplacer ou augmenter les fichiers de configuration du site](#)

- configuration, [Configuration autofs](#)
- LDAP, [Utiliser LDAP pour stocker des mappages Automounter](#)
- autofs version 5, [Améliorations de autofs Version 5 par rapport à la Version 4](#)
- barrières d'écriture, [NFS](#)
- client
 - autofs , [autofs](#)
 - configuration, [Configuration du client NFS](#)
 - options de montage, [Options de montage NFS courantes](#)
- condrestart, [Démarrage et arrêt NFS](#)
- configuration avec un pare-feu, [Exécuter NFS derrière un pare-feu](#)
- configuration du serveur, [Configuration du serveur NFS](#)
 - /etc/exports , [Fichier de configuration /etc/exports](#)
 - commande exportfs, [Commande exportfs](#)
 - commande exportfs avec NFSv4, [Utiliser exportfs avec NFSv4](#)
- démarrage, [Démarrage et arrêt NFS](#)
- direct map support (autofs version 5), [Améliorations de autofs Version 5 par rapport à la Version 4](#)
- enhanced LDAP support (autofs version 5), [Améliorations de autofs Version 5 par rapport à la Version 4](#)
- fonctionnement, [Fonctionnement NFS](#)
- formats des noms d'hôtes, [Formats des noms d'hôtes](#)
- FS-Cache, [Utiliser le cache avec NFS](#)
- introduction, [Network File System \(NFS\)](#)
- lazy mount/unmount support (autofs version 5), [Améliorations de autofs Version 5 par rapport à la Version 4](#)
- montage (configuration du client), [Configuration du client NFS](#)
- multiple master map entries per autofs mount point (autofs version 5), [Améliorations de autofs Version 5 par rapport à la Version 4](#)
- options (configuration du client, montage), [Configuration du client NFS](#)
- proper nsswitch configuration (autofs version 5), use of, [Améliorations de autofs Version 5 par rapport à la Version 4](#)
- RDMA, [NFS sur RDMA](#)
- rechargement, [Démarrage et arrêt NFS](#)
- redémarrage, [Démarrage et arrêt NFS](#)
- remplacer/augmenter les fichiers de configuration du site (autofs), [Configuration autofs](#)
- résolution des problèmes NFS et rpcbind, [Résolution des problèmes NFS et rpcbind](#)
- ressources supplémentaires, [Références](#)
 - documentation installée, [Documentation installée](#)
 - livres apparentés, [Livres sur le sujet](#)
 - sites web utiles, [Sites Web utiles](#)
- rfc2307bis (autofs), [Utiliser LDAP pour stocker des mappages Automounter](#)
- rpcbind , [NFS et rpcbind](#)
- sécurité, [Sécurisation de NFS](#)
 - accès aux hôtes NFSv4, [Sécurité NFS avec AUTH_GSS](#)
 - permissions de fichiers, [Permissions de fichier](#)
- security
 - NFSv3 host access, [Sécurité NFS avec AUTH_SYS et les contrôles d'export](#)
- serveur (configuration du client, montage), [Configuration du client NFS](#)
- services requis, [Services requis](#)
- statut, [Démarrage et arrêt NFS](#)
- stockage de mappages automounter, utiliser LDAP pour le stockage (autofs), [Remplacer ou augmenter les fichiers de configuration du site](#)

- TCP, [Fonctionnement NFS](#)
- UDP, [Fonctionnement NFS](#)

NFS (limitations de cache avec)

- FS-Cache, [Limitations des caches avec NFS](#)

NFS (utiliser avec)

- FS-Cache, [Utiliser le cache avec NFS](#)

niveaux

- RAID, [Niveaux RAID et prise en charge linéaire](#)

niveaux de vidage

- XFS, [Sauvegarde et restauration des systèmes de fichiers XFS](#)

NOP-Outs (désactivation)

- configuration iSCSI, [Root iSCSI](#)

O

opération interactive (xfsrestore)

- XFS, [Opération interactive](#)

option de montage « nobarrier »

- ext4, [Monter un système de fichiers Ext4](#)
- XFS, [Barrières d'écriture](#)

options (configuration du client, montage)

- NFS, [Configuration du client NFS](#)

outils (pour le partitionnement et autres fonctions des systèmes de fichiers)

- alignement et taille des E/S, [Outils des partitions et systèmes de fichiers](#)

P

paquets requis

- ajouter ou supprimer
 - LUN (numéro d'unité logique), [Ajouter ou supprimer une unité logique avec rescanscsi-bus.sh](#)
- FCoE, [Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\)](#)
- systèmes sans disque, [Paramétrer un système sans disque distant](#)

Parallel NFS

- pNFS, [pNFS](#)

paramétrer un cache

- FS-Cache, [Paramétrer un cache](#)

paramètres d'accès au stockage

- Alignement et taille des E/S, [Paramètres d'accès au stockage](#)

parité

- RAID, [Niveaux RAID et prise en charge linéaire](#)

partage de cache

- FS-Cache, [Partage de cache](#)

parted , [Partitions](#)

- afficher la table de partitions, [Afficher la table de partitions](#)
- création de partitions, [Création d'une partition](#)
- redimensionnement de partitions, [Redimensionnement d'une partition](#)
- sélection du périphérique, [Afficher la table de partitions](#)
- suppression de partitions, [Suppression de partition](#)
- tableau des commandes, [Partitions](#)
- vue d'ensemble, [Partitions](#)

partitions

- afficher la liste, [Afficher la table de partitions](#)
- création, [Création d'une partition](#)
 - mkpart , [Créer la partition](#)
- formatage
 - mkfs , [Formatage et étiquetage de la partition](#)
- redimensionnement, [Redimensionnement d'une partition](#)
- suppression, [Suppression de partition](#)

partitions séparées (pour /home, /opt, /usr/local)

- besoins du stockage à prendre en compte pendant l'installation, [Partitions séparées pour /home, /opt, /usr/local](#)

périphérique bloc, vérification

- Fibre Channel
 - modifier le comportement de la perte de lien, [Fibre Channel](#)

périphériques bloc activés DIF/DIX

- besoins de stockage à prendre en compte pendant l'installation, [Périphériques bloc avec DIF/DIX activé](#)

périphériques DASD et zFCP sur IBM System Z

- besoins du stockage à prendre en compte pendant l'installation, [Périphériques DASD et zFCP sur IBM System Z](#)

périphériques, suppression, [Supprimer un périphérique de stockage](#)

Pile d'E/S Linux

- Alignement et taille des E/S, [Alignement et taille des E/S de stockage](#)

pilotes (natifs) Fibre Channel, [Pilotes et capacités natifs Fibre Channel](#)

pilotes (natifs), Fibre Channel, [Pilotes et capacités natifs Fibre Channel](#)

pilotes de contrôleurs RAID matériel

- RAID, [Pilotes de contrôleurs RAID matériel Linux](#)

pilotes natifs Fibre Channel, [Pilotes et capacités natifs Fibre Channel](#)

planificateur E/S (tuning)

- solid-state disks, [Planificateur d'E/S](#)

pNFS

- Parallel NFS, [pNFS](#)

pquota/pqnoenforce

- XFS, [Gestion des quotas XFS](#)

prise en charge d'Anaconda

- RAID, [RAID Support in the Installer](#)

prise en charge de l'installateur

- RAID, [RAID Support in the Installer](#)

private mount, [Partager des montages](#)

problèmes connus

- ajouter ou supprimer
 - LUN (numéro d'unité logique), [Problèmes connus avec rescan-iscsi-bus.sh](#)

proper nsswitch configuration (autofs version 5), use of

- NFS, [Améliorations de autofs Version 5 par rapport à la Version 4](#)

Q

queue_if_no_path

- configuration iSCSI, [Paramètres iSCSI avec dm-multipath](#)
- modification de la perte de lien
 - configuration iSCSI, [remplacement_timeout](#)

quota (autres utilitaires du système de fichiers ext4)

- ext4, [Autres utilitaires du système de fichiers Ext4](#)

quota de disques

- ressources supplémentaires, [Références de quotas de disques](#)

quotacheck , [Créer les fichiers de base de données de quotas](#)

quotaoff , [Activation et désactivation](#)

quotaon , [Activation et désactivation](#)

quotas de disque

- désactivation, [Activation et désactivation](#)
- gestion de, [Gérer les quotas de disque](#)

quotas de disques, [Quotas de disques](#)

- activation, [Configurer les quotas de disques, Activation et désactivation](#)
 - /etc/fstab, modification, [Activer les quotas](#)
 - créer des fichiers de quotas, [Créer les fichiers de base de données de quotas](#)
 - quotacheck, exécution, [Créer les fichiers de base de données de quotas](#)
- allocation par groupe, [Assigner les quotas par groupe](#)
- allocation par système de fichiers, [Définir la période de grâce pour les limites soft](#)
- allocation par utilisateur, [Allouer les quotas par utilisateur](#)
- gestion des
 - rapports, [Rapports sur les quotas de disques](#)
- hard limit, [Allouer les quotas par utilisateur](#)
- période de grâce, [Allouer les quotas par utilisateur](#)
- soft limit, [Allouer les quotas par utilisateur](#)

R

RAID

- configuration d'ensembles RAID, [Configuring RAID Sets](#)
- création de périphériques RAID avancée, [Création de périphériques RAID avancée](#)
- dmraid, [dmraid](#)
- dmraid (configuration d'ensembles RAID), [dmraid](#)
- entrelacement, [Niveaux RAID et prise en charge linéaire](#)
- mdadm (configuration d'ensembles RAID), [mdadm](#)
- mdraid, [mdraid](#)
- mise en miroir, [Niveaux RAID et prise en charge linéaire](#)
- niveau 0, [Niveaux RAID et prise en charge linéaire](#)
- niveau 1, [Niveaux RAID et prise en charge linéaire](#)
- niveau 4, [Niveaux RAID et prise en charge linéaire](#)
- niveau 5, [Niveaux RAID et prise en charge linéaire](#)
- niveaux, [Niveaux RAID et prise en charge linéaire](#)
- parité, [Niveaux RAID et prise en charge linéaire](#)
- pilotes de contrôleurs RAID matériel, [Pilotes de contrôleurs RAID matériel Linux](#)
- prise en charge d'Anaconda, [RAID Support in the Installer](#)
- prise en charge de l'installateur, [RAID Support in the Installer](#)
- RAID linéaire, [Niveaux RAID et prise en charge linéaire](#)
- RAID logiciel, [Types RAID](#)
- RAID matériel, [Types RAID](#)
- raisons d'utilisation, [Réseau redondant de disques indépendants \(RAID, de l'anglais « Redundant Array of Independent Disks »\)](#)
- sous-systèmes RAID, [Sous-systèmes RAID Linux](#)

RAID linéaire

- RAID, [Niveaux RAID et prise en charge linéaire](#)

RAID logiciel (voir RAID)**RAID matériel (voir RAID)****rapport (mode xfs_quota expert)**

- XFS, [Gestion des quotas XFS](#)

RDMA

- NFS, [NFS sur RDMA](#)

READ CAPACITY(16)

- alignement et taille des E/S, [SCSI](#)

redimensionnement d'unités logiques redimensionnées, [Redimensionner une Unité logique En ligne](#)**redimensionner**

- ext4, [Redimensionner un système de fichiers Ext4](#)

redimensionner une unité logique iSCSI, [Redimensionner une unité logique iSCSI](#)**remote port**

- Fibre Channel API, [Interface de programmation Fibre Channel](#)

remplacer/augmenter les fichiers de configuration du site (autofs)

- NFS, [Configuration autofs](#)

répare des systèmes de fichiers XFS avec des journaux endommagés

- XFS, [Réparer un système de fichiers XFS](#)

réparer un système de fichiers

- XFS, [Réparer un système de fichiers XFS](#)

Répertoire `/boot/`, [Répertoire `/boot/`](#)

répertoire `dev`, [Répertoire `/dev/`](#)

répertoire `etc`, [Répertoire `/etc/`](#)

répertoire `mnt`, [Répertoire `/mnt/`](#)

répertoire `opt`, [Répertoire `/opt/`](#)

répertoire `proc`, [Répertoire `/proc/`](#)

répertoire `srv`, [Répertoire `/srv/`](#)

répertoire `sys`, [Répertoire `/sys/`](#)

répertoire `sysconfig`, [Emplacement des fichiers Red Hat Enterprise Linux spéciaux](#)

répertoire `usr`, [Répertoire `/usr/`](#)

répertoire `var`, [Répertoire `/var/`](#)

répertoire `var/lib/rpm/`, [Emplacement des fichiers Red Hat Enterprise Linux spéciaux](#)

répertoire `var/spool/up2date/`, [Emplacement des fichiers Red Hat Enterprise Linux spéciaux](#)

répertoires

- `/dev/`, [Répertoire `/dev/`](#)
- `/etc/`, [Répertoire `/etc/`](#)
- `/mnt/`, [Répertoire `/mnt/`](#)
- `/opt/`, [Répertoire `/opt/`](#)
- `/proc/`, [Répertoire `/proc/`](#)
- `/srv/`, [Répertoire `/srv/`](#)
- `/sys/`, [Répertoire `/sys/`](#)
- `/usr/`, [Répertoire `/usr/`](#)
- `/var/`, [Répertoire `/var/`](#)

`replacement_timeout`

- modification de la perte de lien
 - configuration iSCSI, [SCSI Error Handler](#), [replacement_timeout](#)

`replacement_timeoutM`

- configuration iSCSI, [Root iSCSI](#)

requêtes NOP-Out

- modification de la perte de lien
 - configuration iSCSI, [Intervalle/Délai d'expiration NOP-Out](#)

`rescan-scsi-bus.sh`

- ajouter ou supprimer
 - LUN (numéro d'unité logique), [Ajouter ou supprimer une unité logique avec `rescan-scsi-bus.sh`](#)

`resize2fs`, [Rétablir un système de fichiers Ext2](#)

`resize2fs` (redimensionner ext4)

- ext4, [Redimensionner un système de fichiers Ext4](#)

résolution de problème

- stockage en ligne, [Résolution de problème de configuration de stockage en ligne](#)

résolution des problèmes NFS et rpcbind

- NFS, [Résolution des problèmes NFS et rpcbind](#)

restaurer une copie de sauvegarde

- XFS, [Sauvegarde et restauration des systèmes de fichiers XFS](#)

rfc2307bis (autofs)

- NFS, [Utiliser LDAP pour stocker des mappages Automounter](#)

root iSCSI

- configuration iSCSI, [Root iSCSI](#)

rpcbind , [NFS et rpcbind](#)

- (voir aussi NFS)
- NFS, [Résolution des problèmes NFS et rpcbind](#)
- rpcinfo , [Résolution des problèmes NFS et rpcbind](#)
- statut, [Démarrage et arrêt NFS](#)

rpcinfo , [Résolution des problèmes NFS et rpcbind](#)**S****sauvegarde/restauration**

- XFS, [Sauvegarde et restauration des systèmes de fichiers XFS](#)

scanner les interconnexions

- iSCSI, [Scanner les interconnexions iSCSI](#)

scanner les interconnexions du stockage, [Scanner les interconnexions du stockage](#)**SCSI Error Handler**

- modification de la perte de lien
- configuration iSCSI, [SCSI Error Handler](#)

serveur (configuration du client, montage)

- NFS, [Configuration du client NFS](#)

service de démarrage réseau

- systèmes sans disque, [Paramétrer un système sans disque distant](#)

service tftp, configuration

- systèmes sans disque, [Configuration d'un service tftp pour des clients sans disque](#)

sessions en cours, récupération d'informations sur

- API iSCSI, [API iSCSI](#)

shared mount, [Partager des montages](#)**shared subtrees**

- private mount, [Partager des montages](#)
- shared mount, [Partager des montages](#)
- slave mount, [Partager des montages](#)
- unbindable mount, [Partager des montages](#)

slave mount, [Partager des montages](#)**solid-state disks**

- commande TRIM, [Directives de déploiement des disques SSD](#)
- déploiement, [Considérations pour le déploiement](#)
- directives de déploiement, [Directives de déploiement des disques SSD](#)
- mémoire virtuelle (tuning), [Mémoire virtuelle](#)
- planificateur E/S (tuning), [Planificateur d'E/S](#)
- SSD, [Directives de déploiement des disques SSD](#)
- swap (tuning), [Mémoire swap](#)
- throughput classes de débit, [Directives de déploiement des disques SSD](#)
- tuning, [Considérations pour le paramétrage](#)

sous-arborescences partagées, [Partager des montages](#)

sous-systèmes RAID

- RAID, [Sous-systèmes RAID Linux](#)

SSD

- solid-state disks, [Directives de déploiement des disques SSD](#)

SSM

- System Storage Manager, [System Storage Manager\(SSM\)](#)
 - Backends, [SSM Backends](#)
 - commande cliqué, [Cliqué](#)
 - Installation, [Installation du Gestionnaire de stockage de système \(SSM\)](#)
 - list command, [Afficher les informations sur tous les périphériques détectés](#)
 - resize command, [Augmenter une taille de volume](#)

standards ATA

- alignement et taille des E/S, [ATA](#)

standards SCSI

- alignement et taille des E/S, [SCSI](#)

statut de périphérique

- couche Linux SCSI, [États de périphériques](#)

statut en cours

- couche Linux SCSI, [Contrôle du Minuteur de commande SCSI et du Statut de périphérique](#)

statut hors ligne

- couche Linux SCSI, [Contrôle du Minuteur de commande SCSI et du Statut de périphérique](#)

stockage de disques (voir quotas de disques)

stockage de mappages automounter, utiliser LDAP pour le stockage (autofs)

- NFS, [Remplacer ou augmenter les fichiers de configuration du site](#)

stockage en ligne

- aperçu, [Online Storage Management](#)
 - sysfs, [Online Storage Management](#)
- Fibre Channel, [Fibre Channel](#)
- résolution de problème, [Résolution de problème de configuration de stockage en ligne](#)

stockage sur disque

- parted (voir parted)

stockage virtuel, [Stockage virtuel](#)

stride (spécifier la géométrie d'entrelacement)

- ext4, [Créer un système de fichiers Ext4](#)

su (mkfs.xfs sub-options)

- XFS, [Créer un système de fichiers XFS](#)

suppression de périphériques, [Supprimer un périphérique de stockage](#)

supprimer des chemins vers un périphérique de stockage, [Supprimer un chemin vers un périphérique de stockage](#)

suspension

- XFS, [Suspendre un système de fichier XFS](#)

sw (mkfs.xfs sub-options)

- XFS, [Créer un système de fichiers XFS](#)

swap (tuning)

- solid-state disks, [Mémoire swap](#)

swap space

- file
 - creating, [Supprimer un fichier swap](#)

sysfs

- aperçu
 - stockage en ligne, [Online Storage Management](#)

System Storage Manager

- SSM, [System Storage Manager\(SSM\)](#)
 - Backends, [SSM Backends](#)
 - commande cliqué, [Cliqué](#)
 - Installation, [Installation du Gestionnaire de stockage de système \(SSM\)](#)
 - list command, [Afficher les informations sur tous les périphériques détectés](#)
 - resize command, [Augmenter une taille de volume](#)

système de fichiers

- hiérarchie, [Vue d'ensemble du standard de hiérarchie des systèmes de fichiers \(FHS, ou « Filesystem Hierarchy Standard »\)](#)
- organisation, [Organisation FHS](#)
- standard FHS, [Organisation FHS](#)
- structure, [Structure et maintenance des systèmes de fichiers](#)

Système de fichiers

- Btrfs, [Btrfs \(Aperçu technologique\)](#)

système de fichiers virtuel (/proc)

- /proc/devices, [Système de fichiers virtuel /proc](#)
- /proc/filesystems, [Système de fichiers virtuel /proc](#)
- /proc/mdstat, [Système de fichiers virtuel /proc](#)
- /proc/mounts, [Système de fichiers virtuel /proc](#)
- /proc/mounts/, [Système de fichiers virtuel /proc](#)
- /proc/partitions, [Système de fichiers virtuel /proc](#)

systemes de fichiers

- ext2 (voir ext2)
- ext3 (voir ext3)

systemes de fichiers exportés

- systemes sans disque, [Configuration d'un système de fichiers exporté pour les clients sans disque](#)

systemes sans disque

- DHCP, configuration, [Configuration DHCP pour les clients sans disque](#)
- paquets requis, [Paramétrer un système sans disque distant](#)
- service de démarrage réseau, [Paramétrer un système sans disque distant](#)
- service tftp, configuration, [Configuration d'un service tftp pour des clients sans disque](#)
- systemes de fichiers exportés, [Configuration d'un système de fichiers exporté pour les clients sans disque](#)
- systemes sans disque distants, [Paramétrer un système sans disque distant](#)

systemes sans disque distants

- systemes sans disque, [Paramétrer un système sans disque distant](#)

T

table de partitions

- afficher, [Afficher la table de partitions](#)

taille maximale

- GFS2, [Global File System 2](#)

taille maximum de systemes de fichiers GFS2, [Global File System 2](#)

taille maximum, systemes de fichiers GFS2, [Global File System 2](#)

transport

- Fibre Channel API, [Interface de programmation Fibre Channel](#)

tune2fs

- conversion vers ext3 avec, [Conversion vers un système de fichiers ext3](#)
- rétablie ext2 avec, [Rétablir un système de fichiers Ext2](#)

tune2fs (définir un cache)

- FS-Cache, [Paramétrer un cache](#)

tune2fs (montage)

- ext4, [Monter un système de fichiers Ext4](#)

tuning

- solid-state disks, [Considérations pour le paramétrage](#)

types d'enregistrements

- découverte
 - iSCSI, [Configuration de la découverte iSCSI](#)

types de systemes de fichiers

- ext4, [Le système de fichiers Ext4](#)
- GFS2, [Global File System 2](#)
- XFS, [Le système de fichiers XFS](#)

U

udev règle (délai d'expiration)

- minuteur de commande (SCSI), [Minuteur de commande](#)

umount, [Démonter un système de fichiers](#)**unbindable mount, [Partager des montages](#)****Unités logiques en ligne**

- Modifier l'état de lecture/écriture, [Modifier l'état de lecture/écriture d'une unité logique en ligne](#)

unités logiques iSCSI, redimensionnement, [Redimensionner une unité logique iSCSI](#)**unités logiques redimensionnées, redimensionnement, [Redimensionner une Unité logique En ligne](#)****uquota/uqnoenforce**

- XFS, [Gestion des quotas XFS](#)

utilisateur individuel

- volume_key, [Exécutez volume_key en tant qu'utilisateur individuel](#)

V

vérifier si un périphérique est bloqué

- Fibre Channel
 - modifier le comportement de la perte de lien, [Fibre Channel](#)

version

- what is new
 - autofs, [Améliorations de autofs Version 5 par rapport à la Version 4](#)

volume_key

- commandes, [Commandes volume_key](#)
- utilisateur individuel, [Exécutez volume_key en tant qu'utilisateur individuel](#)

W

WWID

- dénomination persistante, [WWID](#)

WWID (World Wide Identifier)

- dénomination persistante, [WWID](#)

X

XFS

- allocation features, [Le système de fichiers XFS](#)
- augmenter la taille d'un système de fichiers, [Augmenter la taille d'un système de fichiers XFS](#)
- barrières d'écriture, [Barrières d'écriture](#)
- création, [Créer un système de fichiers XFS](#)
- fonctionnalités principales, [Le système de fichiers XFS](#)
- fsync(), [Le système de fichiers XFS](#)
- gestion des quotas, [Gestion des quotas XFS](#)
- gquota/gqnoenforce, [Gestion des quotas XFS](#)
- limite (mode xfs_quota expert), [Gestion des quotas XFS](#)
- limites de projets (paramétrage), [Paramétrer des limites de projets](#)
- mkfs.xfs, [Créer un système de fichiers XFS](#)

- mode cumulatif (xfsrestore), [Mode cumulatif de xfsrestore](#)
- mode expert (xfs_quota), [Gestion des quotas XFS](#)
- mode simple (xfsrestore), [Mode simple de xfsrestore](#)
- montage, [Monter un système de fichiers XFS](#)
- niveaux de vidage, [Sauvegarde et restauration des systèmes de fichiers XFS](#)
- opération interactive (xfsrestore), [Opération interactive](#)
- option de montage « nobarrier », [Barrières d'écriture](#)
- pquota/pqnoenforce, [Gestion des quotas XFS](#)
- rapport (mode xfs_quota expert), [Gestion des quotas XFS](#)
- répare des systèmes de fichiers XFS avec des journaux endommagés, [Réparer un système de fichiers XFS](#)
- réparer un système de fichiers, [Réparer un système de fichiers XFS](#)
- sauvegarde/restauration, [Sauvegarde et restauration des systèmes de fichiers XFS](#)
- su (mkfs.xfs sub-options), [Créer un système de fichiers XFS](#)
- suspension, [Suspendre un système de fichier XFS](#)
- sw (mkfs.xfs sub-options), [Créer un système de fichiers XFS](#)
- types de systèmes de fichiers, [Le système de fichiers XFS](#)
- uquota/uqnoenforce, [Gestion des quotas XFS](#)
- xfsdump, [Sauvegarde et restauration des systèmes de fichiers XFS](#)
- xfsprogs, [Suspendre un système de fichier XFS](#)
- xfsrestore, [Sauvegarde et restauration des systèmes de fichiers XFS](#)
- xfs_admin, [Autres utilitaires des systèmes de fichiers XFS](#)
- xfs_bmap, [Autres utilitaires des systèmes de fichiers XFS](#)
- xfs_copy, [Autres utilitaires des systèmes de fichiers XFS](#)
- xfs_db, [Autres utilitaires des systèmes de fichiers XFS](#)
- xfs_freeze, [Suspendre un système de fichier XFS](#)
- xfs_fsr, [Autres utilitaires des systèmes de fichiers XFS](#)
- xfs_growfs, [Augmenter la taille d'un système de fichiers XFS](#)
- xfs_info, [Autres utilitaires des systèmes de fichiers XFS](#)
- xfs_mdrestore, [Autres utilitaires des systèmes de fichiers XFS](#)
- xfs_metadump, [Autres utilitaires des systèmes de fichiers XFS](#)
- xfs_quota, [Gestion des quotas XFS](#)
- xfs_repair, [Réparer un système de fichiers XFS](#)

xfsdump

- XFS, [Sauvegarde et restauration des systèmes de fichiers XFS](#)

xfsprogs

- XFS, [Suspendre un système de fichier XFS](#)

xfsrestore

- XFS, [Sauvegarde et restauration des systèmes de fichiers XFS](#)

xfs_admin

- XFS, [Autres utilitaires des systèmes de fichiers XFS](#)

xfs_bmap

- XFS, [Autres utilitaires des systèmes de fichiers XFS](#)

xfs_copy

- XFS, [Autres utilitaires des systèmes de fichiers XFS](#)

xfs_db

- XFS, [Autres utilitaires des systèmes de fichiers XFS](#)

xfs_freeze

- XFS, [Suspendre un système de fichier XFS](#)

xfs_fsr

- XFS, [Autres utilitaires des systèmes de fichiers XFS](#)

xfs_growfs

- XFS, [Augmenter la taille d'un système de fichiers XFS](#)

xfs_info

- XFS, [Autres utilitaires des systèmes de fichiers XFS](#)

xfs_mdrestore

- XFS, [Autres utilitaires des systèmes de fichiers XFS](#)

xfs_metadump

- XFS, [Autres utilitaires des systèmes de fichiers XFS](#)

xfs_quota

- XFS, [Gestion des quotas XFS](#)

xfs_repair

- XFS, [Réparer un système de fichiers XFS](#)