



# Red Hat Enterprise Linux 6

## Guide d'administration du stockage

Déployer et configurer un stockage à nœud unique dans Red Hat Enterprise Linux 6  
Édition 2



# Red Hat Enterprise Linux 6 Guide d'administration du stockage

---

Déployer et configurer un stockage à nœud unique dans Red Hat Enterprise Linux 6  
Édition 2

Jacquelynn East  
Red Hat Customer Content Services

Don Domingo  
Red Hat Customer Content Services

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

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

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

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

## Notice légale

Server Development Système de fichiers du noyau  
Copyright © 2016 Red Hat Inc. and others.  
esandeen@redhat.com

Systèmes de fichiers ext3, ext4 et XFS chiffrés

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). 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.

Server Development Stockage du noyau  
mjsmitzer@redhat.com

Pile E/S et Limites

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.

Josef Back

Server Development Système de fichiers du noyau

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.

David Wysochanski

Server Development Stockage du noyau

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

Hans de Goede

Java © is a registered trademark of Oracle and/or its affiliates.  
Base Operating System Installateur

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

Server Development Stockage du noyau

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

Daniel Novotny  
Base Operating System Services principaux

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.

Contributors

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.

Milan Navrátil

All other trademarks are the property of their respective owners.

Red Hat Customer Content Services

mnavrati@redhat.com

## Résumé

Ce guide propose des instructions pour gérer des périphériques de stockage et des systèmes de fichiers sur Red Hat Enterprise Linux 6 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

<b>CHAPITRE 1. APERÇU</b> .....	<b>9</b>
1.1. LES NOUVEAUTÉS DE RED HAT ENTERPRISE LINUX 6	9
Chiffrement de système de fichiers (aperçu technologique)	9
Mise en cache de systèmes de fichiers (aperçu technologique)	9
Btrfs (Aperçu technologique)	9
Traitement des limites d'E/S	9
Prise en charge ext4	9
Stockage de bloc réseau	10
<b>PARTIE I. SYSTÈMES DE FICHIERS</b> .....	<b>11</b>
<b>CHAPITRE 2. STRUCTURE ET MAINTENANCE DES SYSTÈMES DE FICHIERS</b> .....	<b>12</b>
2.1. VUE D'ENSEMBLE DU STANDARD DE HIÉRARCHIE DES SYSTÈMES DE FICHIERS (FHS, OU « FILESYSTEM HIERARCHY STANDARD »)	12
2.1.1. Organisation FHS	12
2.1.1.1. Collecte des informations sur les systèmes de fichiers	13
2.1.1.2. Répertoire /boot/	14
2.1.1.3. Répertoire /dev/	14
2.1.1.4. Répertoire /etc/	15
2.1.1.5. Répertoire /lib/	15
2.1.1.6. Répertoire /media/	15
2.1.1.7. Répertoire /mnt/	15
2.1.1.8. Répertoire /opt/	16
2.1.1.9. Répertoire /proc/	16
2.1.1.10. Répertoire /sbin/	16
2.1.1.11. Répertoire /srv/	17
2.1.1.12. Répertoire /sys/	17
2.1.1.13. Répertoire /usr/	17
2.1.1.14. Répertoire /var/	19
2.2. EMBLEMMENT DES FICHIERS RED HAT ENTERPRISE LINUX SPÉCIAUX	21
2.3. SYSTÈME DE FICHIERS VIRTUEL /PROC	21
2.4. ABANDONNER LES BLOCS INUTILISÉS	22
<b>CHAPITRE 3. SYSTÈME DE FICHIERS CHIFFRÉ</b> .....	<b>23</b>
3.1. MONTER UN SYSTÈME DE FICHIERS COMME CHIFFRÉ	23
3.2. INFORMATIONS SUPPLÉMENTAIRES	24
<b>CHAPITRE 4. BTRFS</b> .....	<b>25</b>
4.1. FONCTIONNALITÉS DE BTRFS	25
<b>CHAPITRE 5. SYSTÈME DE FICHIERS EXT3</b> .....	<b>26</b>
5.1. CRÉER UN SYSTÈME DE FICHIERS EXT3	27
5.2. CONVERSION VERS UN SYSTÈME DE FICHIERS EXT3	27
5.3. RÉTABLIR UN SYSTÈME DE FICHIERS EXT2	28
<b>CHAPITRE 6. LE SYSTÈME DE FICHIERS EXT4</b> .....	<b>30</b>
6.1. CRÉER UN SYSTÈME DE FICHIERS EXT4	31
6.2. MONTER UN SYSTÈME DE FICHIERS EXT4	32
Barrières d'écriture	33
6.3. REDIMENSIONNER UN SYSTÈME DE FICHIERS EXT4	33
6.4. SAUVEGARDE DES SYSTÈMES DE FICHIERS EXT2/3/4	34
6.5. RESTAURER UN SYSTÈME DE FICHIERS EXT2/3/4	35
6.6. AUTRES UTILITAIRES DU SYSTÈME DE FICHIERS EXT4	37

<b>CHAPITRE 7. GLOBAL FILE SYSTEM 2</b> .....	<b>38</b>
<b>CHAPITRE 8. LE SYSTÈME DE FICHIERS XFS</b> .....	<b>39</b>
8.1. CRÉER UN SYSTÈME DE FICHIERS XFS	39
8.2. MONTER UN SYSTÈME DE FICHIERS XFS	41
Barrières d'écriture	41
8.3. GESTION DES QUOTAS XFS	41
Paramétrer des limites de projets	43
8.4. AUGMENTER LA TAILLE D'UN SYSTÈME DE FICHIERS XFS	44
8.5. RÉPARER UN SYSTÈME DE FICHIERS XFS	44
8.6. SUSPENDRE UN SYSTÈME DE FICHIERS XFS	45
8.7. SAUVEGARDE ET RESTAURATION DES SYSTÈMES DE FICHIERS XFS	45
Mode simple de xfsrestore	47
Mode cumulatif de xfsrestore	47
Opération interactive	47
8.8. AUTRES UTILITAIRES DES SYSTÈMES DE FICHIERS XFS	47
<b>CHAPITRE 9. NETWORK FILE SYSTEM (NFS)</b> .....	<b>49</b>
9.1. FONCTIONNEMENT NFS	49
9.1.1. Services requis	50
9.2. PNFS	51
9.3. CONFIGURATION DU CLIENT NFS	52
9.3.1. Monter des systèmes de fichiers NFS à l'aide de /etc/fstab	53
9.4. AUTOFS	54
9.4.1. Améliorations de autofs Version 5 par rapport à la Version 4	54
9.4.2. Configuration autofs	56
9.4.3. Remplacer ou augmenter les fichiers de configuration du site	57
9.4.4. Utiliser LDAP pour stocker des mappages Automounter	58
9.5. OPTIONS DE MONTAGE NFS COURANTES	60
9.6. DÉMARRAGE ET ARRÊT NFS	62
9.7. CONFIGURATION DU SERVEUR NFS	63
9.7.1. Fichier de configuration /etc/exports	63
9.7.2. Commande exportfs	65
9.7.2.1. Utiliser exportfs avec NFSv4	66
9.7.3. Exécuter NFS derrière un pare-feu	67
9.7.3.1. Découverte des exports NFS	68
9.7.4. Formats des noms d'hôtes	68
9.7.5. NFS sur RDMA	69
9.8. SÉCURISATION DE NFS	70
9.8.1. Sécurité NFS avec AUTH_SYS et les contrôles d'export	70
9.8.2. Sécurité NFS avec AUTH_GSS	70
9.8.2.1. Sécurité NFS avec NFSv4	71
9.8.3. Permissions de fichier	71
9.9. NFS ET RPCBIND	72
9.9.1. Résolution des problèmes NFS et rpcbind	72
9.10. RÉFÉRENCES	73
Documentation installée	73
Sites Web utiles	73
Livres sur le sujet	74
<b>CHAPITRE 10. FS-CACHE</b> .....	<b>75</b>
10.1. GARANTIE DES PERFORMANCES	76
10.2. PARAMÉTRER UN CACHE	76
10.3. UTILISER LE CACHE AVEC NFS	77

10.3.1. Partage de cache	78
10.3.2. Limitations des caches avec NFS	79
10.4. DÉFINIR LES LIMITES D'ÉLIMINATION DU CACHE	79
10.5. INFORMATIONS STATISTIQUES	80
10.6. RÉFÉRENCES	80
<b>PARTIE II. ADMINISTRATION DU STOCKAGE</b>	<b>82</b>
<b>CHAPITRE 11. BESOINS DE STOCKAGE À PRENDRE EN COMPTE PENDANT L'INSTALLATION</b>	<b>83</b>
11.1. MISES À JOUR DE LA CONFIGURATION DU STOCKAGE PENDANT L'INSTALLATION	83
11.2. APERÇU DES SYSTÈMES DE FICHIERS PRIS EN CHARGE	83
11.3. CONSIDÉRATIONS PARTICULIÈRES	84
Partitions séparées pour /home, /opt, /usr/local	84
Périphériques DASD et zFCP sur IBM System Z	85
Chiffrer des périphériques blocs avec LUKS	85
Métadonnées RAID BIOS périmées	85
Détection et configuration iSCSI	85
Détection et configuration FCoE	85
DASD	86
Périphériques bloc avec DIF/DIX activé	86
<b>CHAPITRE 12. VÉRIFICATION DU SYSTÈME DE FICHIERS (FSCK)</b>	<b>87</b>
12.1. MEILLEURES PRATIQUES AVEC FSCK	87
12.2. INFORMATIONS SPÉCIFIQUES AUX SYSTÈMES DE FICHIERS POUR FSCK	88
12.2.1. ext2, ext3, et ext4	88
12.2.2. XFS	90
12.2.3. Btrfs	91
<b>CHAPITRE 13. PARTITIONS</b>	<b>92</b>
13.1. AFFICHER LA TABLE DE PARTITIONS	93
13.2. CRÉATION D'UNE PARTITION	94
13.2.1. Créer la partition	95
13.2.2. Formatage et étiquetage de la partition	96
13.2.3. Ajoutez-le à /etc/fstab	96
13.3. SUPPRESSION DE PARTITION	97
13.4. REDIMENSIONNEMENT D'UNE PARTITION	97
<b>CHAPITRE 14. GESTIONNAIRE DE VOLUMES LOGIQUES LVM (« LOGICAL VOLUME MANAGER »)</b>	<b>99</b>
14.1. QU'EST-CE QUE LVM2 ?	100
14.2. UTILISATION DE SYSTEM-CONFIG-LVM	100
14.2.1. Utiliser des entités non-initialisées	103
14.2.2. Ajouter des volumes non alloués à un groupe de volumes	104
14.2.3. Migrer des extensions	107
14.2.4. Ajouter un nouveau disque dur avec LVM	109
14.2.5. Ajouter un nouveau groupe de volumes	110
14.2.6. Étendre un groupe de volumes	111
14.2.7. Modifier un volume logique	113
14.3. RÉFÉRENCES LVM	115
Documentation installée	115
Sites Web utiles	115
<b>CHAPITRE 15. ESPACE SWAP</b>	<b>116</b>
15.1. AJOUTER DE L'ESPACE SWAP	117
15.1.1. Étendre Swap dans un Volume logique LVM2	117
15.1.2. Création d'un volume logique LVM2 avec Swap	118

15.1.3. Création d'un fichier Swap	118
15.2. SUPPRIMER DE L'ESPACE SWAP	119
15.2.1. Réduire un Swap dans un Volume logique LVM2	119
15.2.2. Supprimer un volume logique LVM2 avec Swap	120
15.2.3. Supprimer un fichier swap	120
15.3. DÉPLACER L'ESPACE SWAP	120
<b>CHAPITRE 16. QUOTAS DE DISQUES</b>	<b>121</b>
16.1. CONFIGURER LES QUOTAS DE DISQUES	121
16.1.1. Activer les quotas	121
16.1.2. Remonter les systèmes de fichiers	122
16.1.3. Créer les fichiers de base de données de quotas	122
16.1.4. Allouer les quotas par utilisateur	123
16.1.5. Assigner les quotas par groupe	124
16.1.6. Définir la période de grâce pour les limites soft	125
16.2. GÉRER LES QUOTAS DE DISQUE	125
16.2.1. Activation et désactivation	125
16.2.2. Rapports sur les quotas de disques	126
16.2.3. Contrôler l'exactitude des quotas	126
16.3. RÉFÉRENCES DE QUOTAS DE DISQUES	128
<b>CHAPITRE 17. RÉSEAU REDONDANT DE DISQUES INDÉPENDANTS (RAID, DE L'ANGLAIS « REDUNDANT ARRAY OF INDEPENDENT DISKS »)</b>	<b>129</b>
17.1. TYPES RAID	129
RAID microprogramme	129
Matériel RAID	129
RAID logiciel	130
17.2. NIVEAUX RAID ET PRISE EN CHARGE LINÉAIRE	130
17.3. SOUS-SYSTÈMES RAID LINUX	132
Pilotes de contrôleurs RAID matériel Linux	132
mdraid	133
dmraid	133
17.4. PRISE EN CHARGE RAID DANS L'INSTALLATEUR	133
17.5. CONFIGURATION D'ENSEMBLES RAID	133
mdadm	134
dmraid	134
17.6. CRÉATION DE PÉRIPHÉRIQUES RAID AVANCÉE	134
<b>CHAPITRE 18. UTILISATION DE LA COMMANDE MOUNT</b>	<b>136</b>
18.1. RÉPERTORIER LES SYSTÈMES DE FICHIERS ACTUELLEMENT MONTÉS	136
18.1.1. Spécifier le type de système de fichiers	136
18.2. MONTER UN SYSTÈME DE FICHIERS	137
18.2.1. Spécifier le type de système de fichiers	138
18.2.2. Spécifier les options de montage	139
18.2.3. Partager des montages	140
18.2.4. Déplacer un point de montage	144
18.3. DÉMONTER UN SYSTÈME DE FICHIERS	145
18.4. RÉFÉRENCES DE LA COMMANDE MOUNT	145
18.4.1. Documentation installée	146
18.4.2. Sites Web utiles	146
<b>CHAPITRE 19. FONCTION VOLUME_KEY</b>	<b>147</b>
19.1. COMMANDES	147
19.2. EXÉCUTEZ VOLUME_KEY EN TANT QU'UTILISATEUR INDIVIDUEL	148



19.3. EXÉCUTEZ VOLUME_KEY DANS UNE GRANDE ORGANISATION	149
19.3.1. Se préparer à enregistrer vos clés de chiffrement.	149
19.3.2. Sauvegarde des clés de chiffrement	150
19.3.3. Restaurer l'accès à un volume	150
19.3.4. Configurer les phrases secrètes pour les urgences	151
19.4. RÉFÉRENCES VOLUME_KEY	152
<b>CHAPITRE 20. LISTES DES CONTRÔLE D'ACCÈS (ACL)</b>	<b>153</b>
20.1. MONTER DES SYSTÈMES DE FICHIERS	153
20.1.1. NFS	153
20.2. DÉFINIR LES ACL D'ACCÈS	153
20.3. DÉFINIT LES ACL PAR DÉFAUT	155
20.4. RÉCUPÉRER DES ACL	155
20.5. ARCHIVER DES SYSTÈMES DE FICHIERS AVEC DES ACL	156
20.6. COMPATIBILITÉ AVEC D'ANCIENS SYSTÈMES	157
20.7. RÉFÉRENCES DES ACL	157
<b>CHAPITRE 21. DIRECTIVES DE DÉPLOIEMENT DES DISQUES SSD</b>	<b>158</b>
21.1. CONSIDÉRATIONS POUR LE DÉPLOIEMENT	158
21.2. CONSIDÉRATIONS POUR LE PARAMÉTRAGE	159
Planificateur d'E/S	159
Mémoire virtuelle	159
Mémoire swap	160
<b>CHAPITRE 22. BARRIÈRES D'ÉCRITURE</b>	<b>161</b>
22.1. IMPORTANCE DES BARRIÈRES D'ÉCRITURE	161
Comment les barrières d'écriture fonctionnent	161
22.2. ACTIVER/DÉSACTIVER LES BARRIÈRES D'ÉCRITURE	162
22.3. CONSIDÉRATIONS POUR BARRIÈRES D'ÉCRITURE	162
Désactiver les caches d'écriture	162
Caches d'écriture avec batteries de secours	163
Matrices haut de gamme	163
NFS	163
<b>CHAPITRE 23. ALIGNEMENT ET TAILLE DES E/S DE STOCKAGE</b>	<b>164</b>
23.1. PARAMÈTRES D'ACCÈS AU STOCKAGE	164
23.2. ACCÈS À L'ESPACE UTILISATEUR	165
Interface sysfs	165
ioctl du périphérique bloc	166
23.3. STANDARDS	166
ATA	166
SCSI	166
23.4. EMPILER LES PARAMÈTRES D'E/S	167
23.5. GESTIONNAIRE DE VOLUMES LOGIQUES LVM	167
23.6. OUTILS DES PARTITIONS ET SYSTÈMES DE FICHIERS	168
libblkid de util-linux-ng et fdisk	168
parted et libparted	168
Outils de systèmes de fichiers	168
<b>CHAPITRE 24. PARAMÉTRER UN SYSTÈME SANS DISQUE DISTANT</b>	<b>170</b>
24.1. CONFIGURATION D'UN SERVICE TFTP POUR DES CLIENTS SANS DISQUE	170
24.2. CONFIGURATION DHCP POUR LES CLIENTS SANS DISQUE	171
24.3. CONFIGURATION D'UN SYSTÈME DE FICHIERS EXPORTÉ POUR LES CLIENTS SANS DISQUE	171
<b>CHAPITRE 25. DEVICE MAPPER MULTIPATHING (MAPPEUR DE PÉRIPHÉRIQUES À MULTIPLES CHEMINS)</b>	

<b>D'ACCÈS) ET LE STOCKAGE VIRTUEL</b> .....	<b>173</b>
25.1. STOCKAGE VIRTUEL	173
25.2. DM-MULTIPATH	173
<b>PARTIE III. STOCKAGE EN LIGNE</b> .....	<b>175</b>
<b>CHAPITRE 26. FIBRE CHANNEL</b> .....	<b>176</b>
26.1. INTERFACE DE PROGRAMMATION FIBRE CHANNEL	176
26.2. PILOTES ET CAPACITÉS NATIFS FIBRE CHANNEL	177
<b>CHAPITRE 27. INSTALLER UNE CIBLE ISCSI ET UN INITIATEUR</b> .....	<b>178</b>
27.1. CRÉATION DE CIBLE ISCSI	178
27.2. CRÉATION DE L'INITIATEUR ISCSI	180
<b>CHAPITRE 28. DÉNOMINATION PERSISTANTE</b> .....	<b>182</b>
28.1. WWID	182
28.2. UUID ET AUTRES IDENTIFICATEURS PERSISTANTS	184
<b>CHAPITRE 29. SUPPRIMER UN PÉRIPHÉRIQUE DE STOCKAGE</b> .....	<b>185</b>
<b>CHAPITRE 30. SUPPRIMER UN CHEMIN VERS UN PÉRIPHÉRIQUE DE STOCKAGE</b> .....	<b>187</b>
<b>CHAPITRE 31. AJOUTER UN PÉRIPHÉRIQUE OU UN CHEMIN DE STOCKAGE</b> .....	<b>188</b>
<b>CHAPITRE 32. CONFIGURER UNE INTERFACE FCOE (« FIBRE-CHANNEL OVER ETHERNET »)</b> .....	<b>190</b>
32.1. PARAMÉTRAGE DE CIBLE FCOE (« FIBRE-CHANNEL OVER ETHERNET »)	191
<b>CHAPITRE 33. CONFIGURER UNE INTERFACE FCOE POUR QU'ELLE SOIT AUTOMATIQUEMENT MONTÉE LORS DU DÉMARRAGE</b> .....	<b>194</b>
<b>CHAPITRE 34. SCANNER LES INTERCONNEXIONS DU STOCKAGE</b> .....	<b>196</b>
<b>CHAPITRE 35. CONFIGURER LE DÉCHARGEMENT ET LA LIAISON D'INTERFACES ISCSI</b> .....	<b>198</b>
35.1. AFFICHER LES CONFIGURATIONS IFACE DISPONIBLES	198
35.2. CONFIGURER UN IFACE POUR ISCSI LOGICIEL	200
35.3. CONFIGURER UN IFACE POUR LE DÉCHARGEMENT ISCSI	201
35.4. LIER OU DÉLIER UN IFACE SUR UN PORTAIL	201
<b>CHAPITRE 36. SCANNER LES CIBLES ISCSI PAR DES PORTAILS OU LUN MULTIPLES</b> .....	<b>203</b>
<b>CHAPITRE 37. REDIMENSIONNER UNE UNITÉ LOGIQUE EN LIGNE</b> .....	<b>205</b>
37.1. REDIMENSIONNER DES UNITÉS LOGIQUES FIBRE CHANNEL	205
37.2. REDIMENSIONNER UNE UNITÉ LOGIQUE ISCSI	205
37.3. METTRE À JOUR LA TAILLE DU PÉRIPHÉRIQUE MULTIVOIES (« MULTIPATH »)	206
37.4. MODIFIER L'ÉTAT DE LECTURE/ÉCRITURE D'UNE UNITÉ LOGIQUE EN LIGNE	207
37.4.1. Scanner à nouveau des unités logiques	208
37.4.2. Mettre à jour l'état R/W d'un périphérique multivoies	209
37.4.3. Documentation	209
<b>CHAPITRE 38. AJOUTER OU SUPPRIMER UNE UNITÉ LOGIQUE AVEC RESCAN-SCSI-BUS.SH</b> .....	<b>210</b>
PROBLÈMES CONNUS AVEC RESCAN-SCSI-BUS.SH	210
<b>CHAPITRE 39. MODIFIER LE COMPORTEMENT DE LA PERTE DE LIEN</b> .....	<b>211</b>
39.1. FIBRE CHANNEL	211
39.2. PARAMÈTRES ISCSI AVEC DM-MULTIPATH	212
39.2.1. Intervalle/délai d'expiration NOP-Out	212
SCSI Error Handler	212
39.2.2. replacement_timeout	213

---

39.3. ROOT ISCSI	213
Configurer des délais d'expiration pour une session particulière	214
<b>CHAPITRE 40. CONTRÔLE DU MINUTEUR DE COMMANDE SCSI ET DU STATUT DE PÉRIPHÉRIQUE</b>	<b>215</b>
ÉTATS DE PÉRIPHÉRIQUES	215
MINUTEUR DE COMMANDE	215
<b>CHAPITRE 41. RÉOLUTION DE PROBLÈME DE CONFIGURATION DE STOCKAGE EN LIGNE</b>	<b>216</b>
<b>ANNEXE A. HISTORIQUE DES VERSIONS</b>	<b>217</b>



# 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 6. 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 6. 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 6. Celle-ci les décrits et explique comment les utiliser de manière optimale.

## 1.1. LES NOUVEAUTÉS DE RED HAT ENTERPRISE LINUX 6

Red Hat Enterprise Linux 6 propose les améliorations système aux fichiers suivantes :

### Chiffrement de système de fichiers (aperçu technologique)

Il est désormais possible de chiffrer un système de fichiers lors du montage en utilisant *eCryptfs*<sup>[1]</sup>, fournissant une couche de chiffrement au-dessus d'un système de fichiers. Ce « pseudo-système de fichiers » permet un chiffrement par fichier et par nom de fichier, ce qui offre un meilleur chiffrement granulaire que les périphériques blocs chiffrés. Pour obtenir des informations supplémentaires sur le chiffrement des systèmes de fichiers, veuillez consulter le [Chapitre 3, \*Système de fichiers chiffré\*](#).

### Mise en cache de systèmes de fichiers (aperçu technologique)

*FS-Cache*<sup>[1]</sup> permet l'utilisation d'un stockage local pour mettre en cache les données des systèmes de fichiers servis sur le réseau (par exemple, via NFS). Ceci aide à minimiser le trafic réseau, même si cela ne garantit pas un accès plus rapide aux données sur le réseau. FS-Cache permet à un système de fichiers sur un serveur d'interagir directement avec le cache local d'un client, sans créer de système de fichiers contenant trop de montages. Pour obtenir des informations supplémentaires sur FS-Cache, veuillez consulter le [Chapitre 10, \*FS-Cache\*](#).

### Btrfs (Aperçu technologique)

Btrfs<sup>[1]</sup> est un système de fichiers local désormais disponible. Celui-ci vise à fournir de meilleures performances et une meilleure évolutivité, y compris pour les opérations LVM intégrées. Pour obtenir des informations supplémentaires sur Btrfs, veuillez consulter le [Chapitre 4, \*Btrfs\*](#).

### Traitement des limites d'E/S

La pile d'E/S Linux peut désormais traiter les informations des limites d'E/S des périphériques qui les fournissent. Ceci permet aux outils de gestion du stockage de mieux optimiser les E/S de certains périphériques. Pour obtenir des informations supplémentaires sur ceci, veuillez consulter le [Chapitre 23, \*Alignement et taille des E/S de stockage\*](#).

### Prise en charge ext4

Le système de fichiers ext4 est totalement pris en charge dans cette version. Celui-ci est désormais le système de fichiers par défaut de Red Hat Enterprise Linux 6, prenant en charge un nombre illimité de sous-répertoires. Il présente également un horodatage plus granulaire, une prise en charge des attributs étendue, ainsi que la journalisation de quotas. Pour obtenir des informations supplémentaires sur ext4, veuillez consulter le [Chapitre 6, Le système de fichiers Ext4](#).

## Stockage de bloc réseau

FCoE (« Fibre-channel over Ethernet ») est désormais pris en charge. Il permet à une interface fibre-channel d'utiliser des réseaux Ethernet 10 giga-octets tout en préservant le protocole fibre-channel. Pour obtenir des instructions sur la manière de le paramétrer, veuillez consulter [Chapitre 32, Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\)](#).

---

[1] Cette fonctionnalité est fournie dans cette version en tant qu'*aperçu technologique*. Actuellement, les fonctionnalités des aperçus technologiques ne sont pas prises en charge sous les services d'abonnement Red Hat Enterprise Linux, celles-ci peuvent ne pas être totalement fonctionnelles, et ne sont généralement pas convenables pour une utilisation en milieu de production. Cependant, ces fonctionnalités sont incluses en tant que commodités et pour offrir un plus haut niveau d'exposition.

Vous êtes libre de fournir des commentaires et suggestions quant aux fonctionnalités pour ces aperçus technologiques avant que ceux-ci ne soient totalement pris en charge. Des erratas seront fournis pour les failles de sécurité importantes.

## PARTIE I. SYSTÈMES DE FICHIERS

La section sur les Systèmes de fichiers explique la structure des systèmes de fichiers suivie par deux aperçus technologiques : eCryptfs et Btrfs. Ceci est suivi par les systèmes de fichiers que Red Hat prend entièrement en charge : ext3, ext4, GFS2 (« Global File System 2 »), XFS, NFS et FS-Cache.

## 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 il (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 sur ceux-ci.

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 fichiers journaux, peuvent être modifiés à tout moment ; les fichiers *statiques*, tels que des binaires, ne peuvent pas changer sans une 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 se produire 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 sont un sous-ensemble de petite taille de ceux 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.

### 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 utiliser 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**.

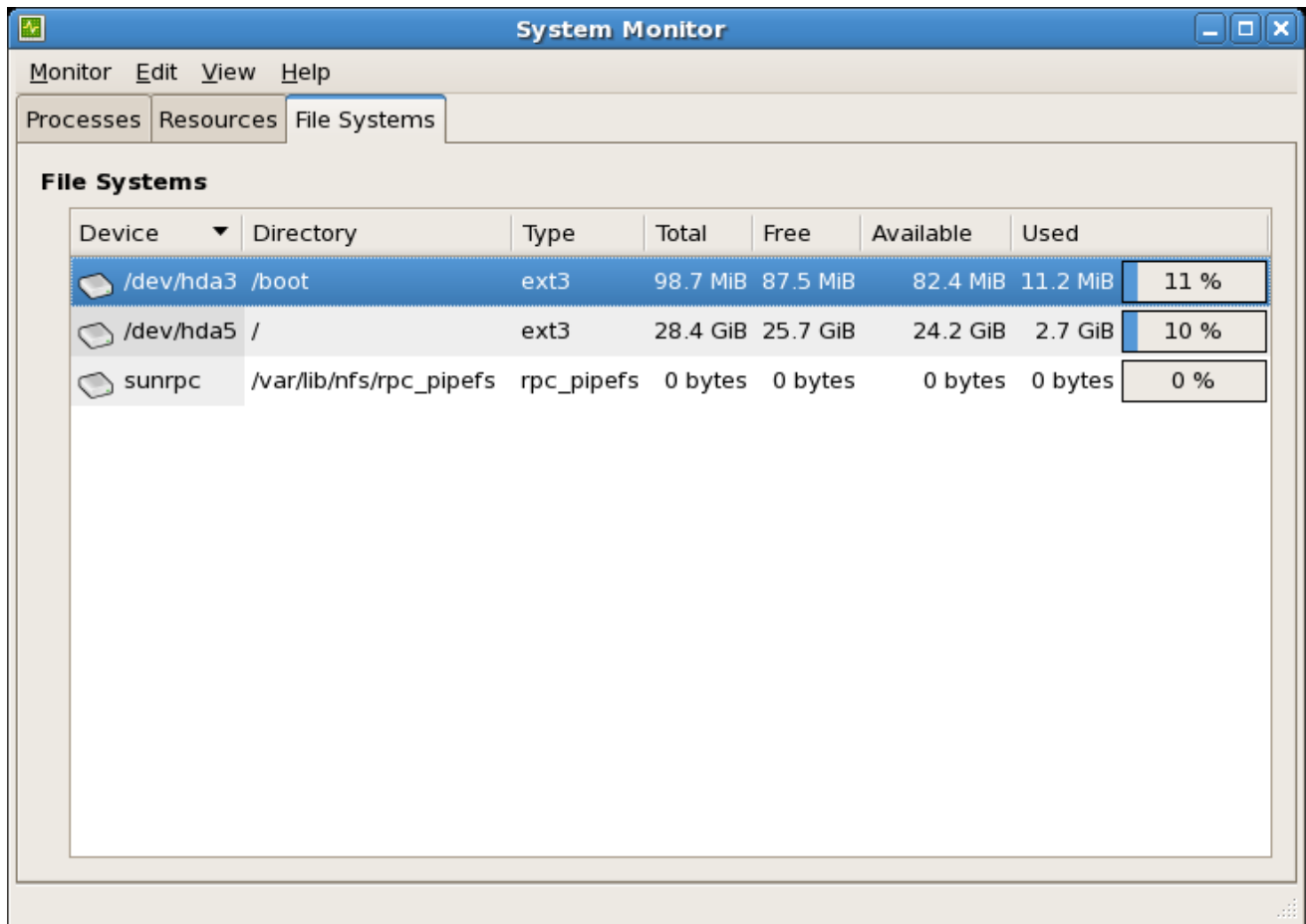


Figure 2.1. Onglet Surveillance système GNOME des systèmes de fichiers

### 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.
/dev/ttyS0	Port série.

#### 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 **/bin/** ou **/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 /lib/

Le répertoire **/lib/** doit uniquement contenir les bibliothèques nécessaires à l'exécution des binaires dans **/bin/** et **/sbin/**. Ces images de bibliothèques partagées sont utilisées pour démarrer le système ou exécuter des commandes à l'intérieur du système de fichiers.

#### 2.1.1.6. Répertoire /media/

Le répertoire **/media/** contient des sous-répertoires utilisés comme points de montage pour des supports amovibles, tels que les supports de stockage USB, les DVD et les CD-ROM.

#### 2.1.1.7. 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.

#### 2.1.1.8. 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/`.

#### 2.1.1.9. 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 »](#).

#### 2.1.1.10. Répertoire `/sbin/`

Le répertoire `/sbin/` stocke les binaires essentiels au démarrage, à la restauration, la récupération, ou la réparation du système. Les binaires dans `/sbin/` requièrent des privilèges root pour être utilisés. En outre, `/sbin/` contient des binaires utilisés par le système *avant* que le répertoire `/usr/` ne soit monté. Tout utilitaire système utilisé après le montage de `/usr/` sera habituellement placé dans `/usr/sbin/`.

Au minimum, les programmes suivants doivent être stockés dans `/sbin/` :

- `arp`
- `clock`
- `halt`
- `init`
- `fsck.*`
- `grub`
- `ifconfig`
- `mingetty`
- `mkfs.*`
- `mkswap`
- `reboot`

- **route**
- **shutdown**
- **swapoff**
- **swapon**

#### 2.1.1.11. 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/**.



#### **NOTE**

L'installation httpd par défaut utilise **/var/www/html** pour le contenu servi.

#### 2.1.1.12. Répertoire **/sys/**

Le répertoire **/sys/** utilise le nouveau système de fichiers virtuel spécifique au noyau 2.6, **sysfs**. Grâce à la meilleure prise en de l'enfichage à chaud de périphériques matériels dans le noyau 2.6, le répertoire **/sys/** contient des informations similaires à celles offertes par **/proc/**, mais affiche une vue hiérarchique des informations des périphériques qui est spécifique aux périphériques enfichables à chaud.

#### 2.1.1.13. 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. Le répertoire **/usr/** contient habituellement 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. Ce répertoire est destiné aux systèmes 32-bit.

#### **/usr/lib64**

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. Ce répertoire est destiné aux systèmes 64-bit.

#### **/usr/libexec**

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

#### **/usr/sbin**

Ce répertoire stocke les binaires d'administration système qui n'appartiennent pas à **/sbin/**.

#### **/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/** devrait 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/**.

#### 2.1.1.14. 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/** devrait 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.

Ci-dessous figurent certains des répertoires trouvés dans le répertoire **/var/**, selon ce qui installé sur le système :

- **/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/`

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 peuvent inclure :

- `/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. EMLACEMENT 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 6.

## 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. Celle-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 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 ne 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. SYSTÈME DE FICHIERS CHIFFRÉ

Red Hat Enterprise Linux 6 offre un aperçu technologique de *eCryptfs*, un « pseudo-système de fichiers » qui fournit le chiffrement des données et des noms de fichiers basé sur fichier. Le terme « pseudo-système de fichiers » fait référence au fait qu'*eCryptfs* ne possède pas de format sur disque. Il s'agit plutôt d'une couche de système de fichiers qui réside sur un système de fichiers réel. La couche *eCryptfs* offre des capacités de chiffrement.

*eCryptfs* fonctionne comme un montage de liaison en interceptant les opérations de fichier qui écrivent sur le système de fichiers sous-jacent (qui est chiffré). La couche *eCryptfs* ajoute un en-tête aux métadonnées des fichiers dans le système de fichiers sous-jacent. Ces métadonnées décrivent le chiffrement de ce fichier et *eCryptfs* chiffre les données du fichier avant que celui-ci ne soit passé au système de fichiers chiffré. Optionnellement, *eCryptfs* peut aussi chiffrer les noms de fichiers.

*eCryptfs* n'est pas un système de fichiers sur disque. Ainsi, il n'est pas nécessaire de le créer via des outils tels que **mkfs**. Au contraire, *eCryptfs* est initié par l'exécution d'une commande de montage particulière. Pour gérer des systèmes de fichiers protégés par *eCryptfs*, le paquet **ecryptfs-utils** doit tout d'abord être installé.

### 3.1. MONTER UN SYSTÈME DE FICHIERS COMME CHIFFRÉ

Pour chiffrer un système de fichiers avec *eCryptfs*, veuillez exécuter la commande suivante :

```
# mount -t ecryptfs /source /destination
```

Chiffrer une hiérarchie de répertoires (*/source* dans l'exemple ci-dessus) avec *eCryptfs* signifie le monter sur un point de montage chiffré par *eCryptfs* (*/destination* dans l'exemple ci-dessus). Toutes les opérations sur */destination* seront passées chiffrées au système de fichiers sous-jacent */source*. Cependant, dans certains cas, il peut être possible pour une opération de fichier de modifier */source* directement, sans passer par la couche *eCryptfs*, ce qui pourrait mener à des incohérences.

C'est pourquoi Red Hat recommande, pour la plupart des environnements, que les noms des répertoires */source* et */destination* soient identiques. Par exemple :

```
# mount -t ecryptfs /home /home
```

Cela signifie le chiffrement et le montage d'un système de fichiers *sur lui-même*. Ce faire permet de s'assurer que *toutes* les opérations de fichier sur */home* passent effectivement via la couche *eCryptfs*.

Pendant les processus de montage et de chiffrement, **mount** permettra aux paramètres suivants d'être configurés :

#### Type de clé de chiffrement

**openssl**, **tspl**, ou **passphrase** (« phrase de passe »). Lorsque vous choisissez **passphrase**, **mount** vous demandera une phrase de passe.

#### Cipher

**aes**, **blowfish**, **des3\_ede**, **cast6**, ou **cast5**.

#### Taille d'octets de la clé

**16**, **32**, ou **24**.

### passthrough texte clair

Activé ou désactivé.

### chiffrement du nom de fichier

Activé ou désactivé.

Après la dernière étape d'un montage interactif, **mount** affichera toutes les sélections effectuées et procédera au montage. Cette sortie contient les équivalences des options de ligne de commande de chaque paramètre choisi. Par exemple, pour monter **/home** avec un type de clé **phrase de passe**, un chiffrement **aes**, une taille d'octet de **16** avec le **relais en texte clair** et le **chiffrement de nom de fichier** désactivés, la sortie sera :

```
Attempting to mount with the following options:
ecryptfs_unlink_sigs
ecryptfs_key_bytes=16
ecryptfs_cipher=aes
ecryptfs_sig=c7fed37c0a341e19
Mounted eCryptfs
```

Les options affichées ici peuvent être directement passées à la ligne de commande pour chiffrer et monter un système de fichiers en utilisant la même configuration. Pour ce faire, veuillez utiliser chaque option en tant qu'argument de l'option **-o** de la commande **mount**. Par exemple :

```
# mount -t ecryptfs /home /home -o ecryptfs_unlink_sigs \
ecryptfs_key_bytes=16 ecryptfs_cipher=aes
ecryptfs_sig=c7fed37c0a341e19[2]
```

## 3.2. INFORMATIONS SUPPLÉMENTAIRES

Pour obtenir des informations supplémentaires sur eCryptfs et sur ses options de montage, veuillez consulter **man ecryptfs** (fourni par le paquet **ecryptfs-utils**). Le document sur le noyau suivant (fourni par le paquet **kernel-doc**) offre également des informations supplémentaires sur eCryptfs :

**</usr/share/doc/kernel-doc-version/Documentation/filesystems/ecryptfs.txt>**

---

[2] Ceci est une seule commande divisée sur plusieurs lignes pour accommoder les versions PDF et imprimées de ce document. Toutes les lignes concaténées — précédées d'une barre oblique inversée (\) — doivent être considérées comme formant une seule et même commande, sans barres obliques inversées.

## CHAPITRE 4. BTRFS

Btrfs est un nouveau système de fichier local en cours de développement. Il vise à fournir de meilleures performances et une meilleure évolutivité, bénéficiant ainsi aux utilisateurs.



### NOTE

La qualité du système de fichiers Btrfs n'est pas encore au niveau de production. Avec Red Hat Enterprise Linux 6, Btrfs est toujours au niveau d'aperçu technologique, et est donc uniquement construit pour Intel 64 et AMD64.

### 4.1. FONCTIONNALITÉS DE BTRFS

Plusieurs utilitaires sont intégrés à Btrfs afin d'en faciliter l'administration pour les administrateurs système. Ceux-ci incluent :

#### Restauration de système intégrée

Des instantanés du système de fichiers permettent la restauration du système sur un état précédent, connu et fonctionnant bien, au cas où un incident ne se produise.

#### Compression intégrée

Ceci permet d'économiser de l'espace plus facilement.

#### Fonctionnalité Checksum

Ceci permet d'améliorer la détection d'erreurs.

Des fonctionnalités spécifiques incluent des opérations LVM intégrées, telles que :

- l'ajout ou la suppression en ligne, dynamique de nouveaux périphériques de stockage ;
- la prise en charge interne de RAID à travers les périphériques composants ;
- la possibilité d'utiliser différents niveaux RAID pour les métadonnées ou pour les données utilisateurs ;
- la fonctionnalité checksum complète pour toutes les métadonnées et pour toutes les données utilisateur.

## CHAPITRE 5. 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

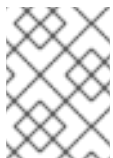
Après une panne d'alimentation ou une panne du système inattendue (aussi appelé un *arrêt du système incorrect*), la cohérence de chaque système de fichiers ext2 monté sur la machine doit être vérifiée par le programme **e2fsck**. Ce long processus peut retarder le démarrage du système de manière importante, 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 instantané 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 entrainera des compromis au niveau de l'intégrité des données si jamais le système devait tomber en panne.

### 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 5.2, « Conversion vers un](#)

[système de fichiers ext3](#) » pour obtenir des informations supplémentaires sur la manière d'accomplir cette tâche.

La version Red Hat Enterprise Linux 6 d'ext3 offre les mises à jour suivantes :

### Changement des tailles par défaut des inodes

La taille par défaut de l'inode sur disque a augmenté afin de permettre un stockage plus efficace des attributs étendus, par exemple les attributs des listes ACL ou de SELinux. Avec ce changement, le nombre d'inodes créés par défaut sur un système de fichiers d'une taille donnée a été réduit. La taille d'inode peut être sélectionnée avec l'option `mke2fs -I` ou spécifiée dans `/etc/mke2fs.conf` pour définir les valeurs globales par défaut de `mke2fs`.



#### NOTE

Si vous effectuez une mise à niveau vers Red Hat Enterprise Linux 6 et que vous avez l'intention de conserver l'un des systèmes de fichiers ext3 intact, il ne sera pas nécessaire de refaire le système de fichiers.

### Nouvelle option de montage : `data_err`

Une nouvelle option de montage a été ajoutée : `data_err=abort`. Cette option ordonne à ext3 d'abandonner le journal si une erreur se produit dans un espace de données de fichier (et non dans un espace de métadonnées de fichier) sous le mode `data=ordered`. Cette option est désactivée par défaut (définie comme `data_err=ignore`).

### Utilisation du stockage plus efficace

Lors de la création d'un système de fichiers (avec `mkfs`), `mke2fs` tentera « d'abandonner » ou « d'éditer » les blocs qui ne sont pas utilisés par les métadonnées du système de fichiers. Ceci permet de faciliter l'optimisation des SSD ou de l'allocation dynamique. Pour supprimer ce comportement, veuillez utiliser l'option `mke2fs -K`.

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 5.2](#), « [Conversion vers un système de fichiers ext3](#) ».

## 5.1. 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 5.1. Créer un système de fichiers ext3

1. Formatez la partition avec le système de fichiers ext3 en utilisant `mkfs`.
2. Étiquetez le système de fichiers en utilisant `e2label`.

## 5.2. CONVERSION VERS UN SYSTÈME DE FICHIERS EXT3

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



## NOTE

Une installation par défaut de Red Hat Enterprise Linux utilise ext4 pour les systèmes de fichiers. Cependant, 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.

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.

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

## 5.3. 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/VolGroup00-LogVol02
```

### Procédure 5.2. Rétablir depuis ext3 vers ext2

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

```
# umount /dev/mapper/VolGroup00-LogVol02
```

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

```
# tune2fs -O ^has_journal /dev/mapper/VolGroup00-LogVol02
```

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

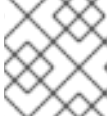


```
# e2fsck -y /dev/mapper/VolGroup00-LogVol02
```

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

```
# mount -t ext2 /dev/mapper/VolGroup00-LogVol02 /mount/point
```

Dans la commande ci-dessus, remplacez */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 6. LE SYSTÈME DE FICHIERS EXT4

Le système de fichiers ext4 est une extension évolutive du système de fichiers ext3, qui était le système de fichiers par défaut de Red Hat Enterprise Linux 5. Ext4 est le système de fichiers par défaut de Red Hat Enterprise Linux 6, et peut prendre en charge des fichiers et systèmes de fichiers d'une taille 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 5, 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.

## 6.1. 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 6.1. 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
```

```
This filesystem will be automatically checked every 20 mounts or
180 days, whichever comes first. Use tune2fs -c or -i to override.
```

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 6 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.

## **6.2. 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
```

L'utilitaire **tune2fs** permet également aux administrateurs 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 le 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 22, Barrières d'écriture](#).

## 6.3. REDIMENSIONNER UN SYSTÈME DE FICHIERS EXT4

Avant d'agrandir un système de fichiers ex4, 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 node
```

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**.

## 6.4. SAUVEGARDE DES SYSTÈMES DE FICHIERS EXT2/3/4

### Procédure 6.1. 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](https://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
```

## 6.5. RESTAURER UN SYSTÈME DE FICHIERS EXT2/3/4

### Procédure 6.2. 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.

## 6.6. AUTRES UTILITAIRES DU SYSTÈME DE FICHIERS EXT4

Red Hat Enterprise Linux 6 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 »](#).

Comme démontré dans [Section 6.2, « 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 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. 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. Il fut créé pour prendre en charge des systèmes de fichiers extrêmement grands (allant jusqu'à 16 exaoctets), ainsi que des fichiers (8 exaoctets) et des structures de répertoires (contenant des dizaines de millions d'entrées) extrêmement volumineux.

### 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 6 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( )** après.

Pour obtenir des informations supplémentaires sur les implications de l'allocation différée sur un système de fichiers, veuillez consulter les *Fonctionnalités de l'allocation* dans le [Chapitre 6, Le système de fichiers Ext4](#). La solution de contournement pour s'assurer des écritures sur disque s'applique également à XFS.

### 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.

#### **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.

## 8.1. 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 8.1. 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
      =                        sunit=0        swidth=0 blks
naming    =version 2          bsize=4096   ascii-ci=0
log       =internal log      bsize=4096   blocks=6400, version=2
      =                        sectsz=512    sunit=0 blks, lazy-
count=1
realtime  =none              extsz=4096   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 `xfsgrowfs` (veuillez consulter [Section 8.4, « 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.

Pour spécifier la géométrie d'entrelacement, veuillez utiliser les sous-options `mkfs.xfs` 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`.

## 8.2. MONTER UN SYSTÈME DE FICHIERS XFS

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

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

XFS prend également en charge plusieurs options de montage pour influencer le comportement.

XFS alloue des inodes pour refléter leurs emplacements sur disque par défaut. Cependant, comme certaines applications de l'espace utilisateur 32 bit ne sont pas compatibles avec les nombres d'inodes supérieurs à  $2^{32}$ , XFS allouera tous les inodes sur des emplacements de disque pouvant résulter en un nombre d'inodes de 32 bit. Ceci peut conduire à une réduction des performances sur les systèmes de fichiers de très grande taille (c'est-à-dire de plus de 2 téraoctets), car les inodes sont faussés au début du périphérique bloc, tandis que les données sont faussées à la fin.

Pour répondre à ceci, veuillez utiliser l'option de montage **inode64**. Cette option configure XFS pour allouer des inodes et données à travers le système de fichiers tout entier, ce qui peut améliorer les performances :

```
# mount -o inode64 /dev/device /mount/point
```

### 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 22, Barrières d'écriture](#).

## 8.3. 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 comme **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 8.2. 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 -cx '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 /home/john'
```

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 durs (« hard ») et douces (« soft ») peuvent également être configurées à l'aide de **bsoft** ou **bhard** au lieu de **isoft** ou **ihard**.

### Exemple 8.3. 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
```



### 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 -c 'project -s projectname'
```

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.

Pour obtenir davantage d'informations sur le paramétrage de quotas XFS, veuillez consulter **man xfs\_quota**.

## 8.4. 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`.

## 8.5. 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

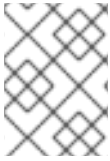
L'utilitaire `xfs_repair` ne peut pas réparer un système de fichiers XFS avec un journal endommagé. Pour supprimer le journal, montez et démontez le système de fichiers. Si le journal est corrompu et qu'il ne peut pas être réutilisé, veuillez utiliser l'option `-L` (« forcer la mise à zéro du journal ») 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 supplémentaire sur la réparation de systèmes de fichiers XFS, veuillez consulter `man xfs_repair`.



## 8.6. SUSPENDRE UN SYSTÈME DE FICHER 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 instantanés 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**, qui est uniquement disponible sur x86\_64.

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

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

Pour dégeler un système de fichiers XFS, veuillez utiliser :

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

Lors de la capture d'un instantané 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 l'instantané.



### NOTE

L'utilitaire **xfs\_freeze** peut également être utilisé pour geler ou dégeler un système de fichier ext3, ext4, GFS2, XFS, et BTRFS. La syntaxe pour ce faire est la même.

Pour obtenir des informations supplémentaire sur le gel et dégel de systèmes de fichiers XFS, veuillez consulter **man xfs\_freeze**.

## 8.7. 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 6 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 8.4. 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
```

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**.

## 8.8. AUTRES UTILITAIRES DES SYSTÈMES DE FICHIERS XFS

Red Hat Enterprise Linux 6 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 de manière périodique car ceci n'est normalement pas justifié.

### **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. L'utilitaire **xfs\_metadump** doit uniquement être utilisé 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.

## CHAPITRE 9. 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

### 9.1. FONCTIONNEMENT NFS

Actuellement, il y a trois versions de NFS. NFS version 2 (NFSv2) est une version plus ancienne avec une prise en charge plus généralisée. 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.

NFS version 4 (NFSv4) fonctionne à travers les pare-feu et sur internet, ne nécessite plus le service **rpcbind**, prend en charge les ACL, et utilise des opérations « Stateful » (avec état). Red Hat Enterprise Linux 6 prend en charge les clients NFSv2, NFSv3, et NFSv4. Lors du montage d'un système de fichiers via NFS, Red Hat Enterprise Linux utilise NFSv4 par défaut si le serveur le prend en charge.

Toutes les versions de NFS peuvent utiliser *TCP* (*Transmission Control Protocol*) exécuté sur un réseau IP, avec NFSv4 qui le requiert. NFSv2 et 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 NFSv2 ou 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.

Les protocoles de montage et de verrouillage ont été incorporés au protocole NFSv4. Le serveur écoute aussi le port TCP 2049. Ainsi, NFSv4 n'a pas besoin d'interagir avec les démons **rpcbind** [3], **lockd**, et **rpc.statd**. Le démon **rpc.mountd** est requis sur le serveur NFS pour créer les exports.



#### 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 prône aux erreurs si le port est indisponible, ou s'il est en conflit avec un autre démon.

### 9.1.1. 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 6 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 6 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](http://nfsv4.bullopen-source.org/doc/tirpc_rpcbind.php)
- *Prise en charge IPv6 sur NFS* : [http://nfsv4.bullopen-source.org/doc/nfs\\_ipv6.php](http://nfsv4.bullopen-source.org/doc/nfs_ipv6.php)

## nfs

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

## nfslock

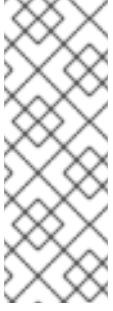
**service nfslock start** active un service obligatoire qui lance les processus RPC appropriés, ce qui permet aux clients NFS de verrouiller des fichiers sur le serveur.

## 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.

## 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**.



## NOTE

À partir de Red Hat Enterprise Linux 6.3, 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**.

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 NFSv2 et 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.

### 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 NFSv2 et 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. Veuillez consulter l'article de la base de connaissances <https://access.redhat.com/site/solutions/130783> lorsque vous utilisez un domaine local.

## 9.2. 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.

Pour activer cette fonctionnalité, veuillez utiliser l'une des options de montage suivantes sur les montages d'un serveur activé pNFS :

```
-o minorversion=1
```

ou

```
-o v4.1
```

Une fois le serveur activé pour pNFS, le noyau `nfs_layout_nfsv41_files` est automatiquement chargé sur le premier montage. Veuillez utiliser la commande suivante pour vérifier que le module a bien été chargé :

```
$ lsmod | grep nfs_layout_nfsv41_files
```

La commande `mount` permet également de vérifier qu'un montage NFSv4.1 a été effectué avec succès. L'entrée de montage dans la sortie devrait contenir `minorversion=1`.



### IMPORTANT

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 des fichiers, ainsi pNFS sera uniquement utilisé si le serveur prend aussi en charge le type de structure des fichiers.

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

## 9.3. 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 9.5, « 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 6 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**.

Veuillez 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 9.3.1, « Monter des systèmes de fichiers NFS à l'aide de /etc/fstab »](#) et [Section 9.4, « autofs »](#) pour obtenir davantage d'informations.

### 9.3.1. Monter des systèmes de fichiers NFS à l'aide de `/etc/fstab`

Il existe une autre manière de monter un partage NFS à partir d'un autre ordinateur, en ajoutant une ligne au fichier `/etc/fstab`. La ligne doit faire état du nom d'hôte du serveur NFS, du répertoire en cours d'exportation sur le serveur, et du répertoire sur l'ordinateur local où le partage NFS sera monté. Vous devez vous connecter en tant qu'utilisateur `root` afin de modifier le fichier `/etc/fstab`.

#### Exemple 9.1. 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.

Le fichier `/etc/fstab` est référencé par le service **netfs** au moment du démarrage, ainsi les lignes faisant référence aux partages NFS ont le même effet que la saisie manuelle de la commande **mount** pendant le processus de démarrage.

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 9.3, « 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`.

**9.4. 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.

**9.4.1. 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 qu'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.

Veillez 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 9.2. 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

```

## 9.4.2. 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 9.4.1, « 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-mêmes 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 9.3. 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:/exports/payroll
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
```

### **9.4.3. 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**.

#### 9.4.4. 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 schema 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/autofs.conf** en supprimant les caractères de commentaires de la définition du schéma.

■

**Exemple 9.4. Paramétrer la configuration autofs**

```
map_object_class = automountMap
entry_object_class = automount
map_attribute = automountMapName
entry_attribute = automountKey
value_attribute = automountInformation
```

**NOTE**

À partir de Red Hat Enterprise Linux 6.6, LDAP autofs est défini dans le fichier `/etc/autofs.conf` au lieu du fichier `/etc/systemconfig/autofs` comme c'était le cas dans les versions précédentes.

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 9.5. Configuration LDIF**

```
# 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/&

```

## 9.5. 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**

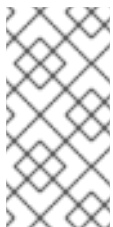
Désactive les bits **set-user-identifiant** ou **set-group-identifiant**. Ceci empêche les utilisateurs distants de gagner des privilèges plus élevés en exécutant un programme **setuid**.

### **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.



### **NOTE**

Si une valeur **rsize** n'est pas spécifiée, ou si la valeur spécifiée est plus élevée que la valeur maximum prise en charge par le client ou le serveur, alors le client et le serveur négocieront la valeur de redimensionnement la plus élevée qu'ils peuvent tous deux prendre en charge.

### **sec=mode**

Spécifie le type de sécurité à utiliser lors de l'authentification d'une connexion NFS. Le paramètre par défaut est **sec=sys**, qui fait usage d'UID et GID UNIX locaux en utilisant **AUTH\_SYS** pour 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**.

## 9.6. DÉMARRAGE ET ARRÊT NFS

Pour exécuter un serveur NFS, le service **rpcbind**<sup>[3]</sup> doit être en cours d'exécution. Pour vérifier que **rpcbind** soit effectivement actif, veuillez utiliser la commande suivante :

```
# service rpcbind status
```

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 :

```
# service nfs start
```

**nfslock** doit aussi être démarré pour que le client et le serveur NFS puissent fonctionner correctement. Pour lancer le verrouillage NFS, veuillez utiliser la commande suivante :

```
# service nfslock start
```

Si NFS est paramétré pour être lancé lors du démarrage, assurez-vous que **nfslock** démarre également en exécutant **chkconfig --list nfslock**. Si **nfslock** n'est pas paramétré sur « Marche » (**on**), cela signifie que vous devrez manuellement exécuter **service nfslock start** chaque fois que l'ordinateur démarre. Pour paramétrer **nfslock** pour qu'il soit lancé automatiquement au démarrage, veuillez utiliser **chkconfig nfslock on**.

**nfslock** est uniquement nécessaire pour NFSv2 et NFSv3.

Pour arrêter le serveur, veuillez utiliser :

```
# service nfs stop
```

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 :

```
# service nfs restart
```

L'option **condrestart** (*redémarrage conditionnel*) démarrera uniquement **nfs** s'il est en cours d'exécution. Cette option est utile pour les scripts, car elle ne démarre pas le démon si celui-ci n'est pas en cours d'exécution. Pour redémarrer le serveur de manière conditionnelle, veuillez saisir :

```
# service nfs condrestart
```

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

```
# service nfs reload
```

## 9.7. CONFIGURATION DU SERVEUR NFS

Il existe deux manières de configurer 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**

### 9.7.1. 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 9.7.4, « 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 9.6. 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 exportfs**.

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 le reste du monde peut le monter en lecture/écriture.

### 9.7.2. Commande exportfs

Chaque système de fichiers exporté vers des utilisateurs distants avec NFS, ainsi que le niveau d'accès de ces systèmes de fichiers, est répertorié dans le fichier **/etc/exports**. Lorsque le service **nfs** démarre, la commande **/usr/sbin/exportfs** 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/exportfs** permet à l'utilisateur root d'exporter ou d'annuler l'export des répertoires de manière sélective, sans redémarrer le service NFS.

Lorsque les bonnes options sont passées, la commande `/usr/sbin/exportfs` écrit les systèmes de fichiers exportés sur `/var/lib/nfs/etab`. Comme `rpc.mountd` fait référence au fichier `etab` lors de la décision d'octroiement 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/exportfs` :

**-r**

Cause à tous les répertoires répertoriés dans `/etc/exports` d'être exportés en construisant une nouvelle liste d'exports dans `/etc/lib/nfs/etab`. 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 9.7.1, « 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`.

### 9.7.2.1. Utiliser `exportfs` avec NFSv4

Avec Red Hat Enterprise Linux 6, aucune étape supplémentaire n'est requise pour configurer les exports NFSv4 car tous les systèmes de fichiers mentionnés sont automatiquement disponibles aux clients NFSv2, NFSv3, et NFSv4 qui utilisent le même chemin. Ceci n'était pas le cas dans les versions précédentes.

Pour empêcher aux clients d'utiliser NFSv4, veuillez l'éteindre en sélectionnant **RPCNFSDARGS= -N 4** dans **/etc/sysconfig/nfs**.

### 9.7.3. 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 de configuration **/etc/sysconfig/nfs** pour contrôler les ports sur lesquels les services RPC requis seront exécutés.

**/etc/sysconfig/nfs** pourrait ne pas exister par défaut sur tous les systèmes. Si cela s'avère être le cas, veuillez le créer puis ajoutez les variables suivantes, en remplaçant *port* par un numéro de port non utilisé (alternativement, si le fichier existe, dé-commentez et modifiez les entrées comme nécessaire) :

#### **MOUNTD\_PORT=port**

Contrôle quel port TCP et UDP **mountd** (**rpc.mountd**) utilise.

#### **STATD\_PORT=port**

Contrôle quel statut de port TCP et UDP (**rpc.statd**) utilise.

#### **LOCKD\_TCPPOINT=port**

Contrôle quel port TCP **nlockmgr** (**lockd**) utilise.

#### **LOCKD\_UDPOINT=port**

Contrôle quel port UDP **nlockmgr** (**lockd**) utilise.

Si le démarrage de NFS échoue, veuillez vérifier **/var/log/messages**. Normalement, le démarrage de NFS échouera si vous spécifiez un numéro de port qui est déjà en cours d'utilisation. Après avoir modifié **/etc/sysconfig/nfs**, veuillez redémarrer le service NFS en utilisant **service nfs restart**. Exécutez la commande **rpcinfo -p** pour confirmer les modifications.

Pour configurer un pare-feu afin d'autoriser NFS, veuillez effectuer les étapes suivantes :

#### **Procédure 9.1. Configurer un pare-feu afin d'autoriser NFS**

1. Autoriser le port TCP et UDP 2049 pour NFS.
2. Autoriser le port TCP et UDP 111 (**rpcbind/sunrpc**).
3. Autoriser le port TCP et UDP spécifié avec **MOUNTD\_PORT="port"**
4. Autoriser le port TCP et UDP spécifié avec **STATD\_PORT="port"**
5. Autoriser le port TCP spécifié avec **LOCKD\_TCPPOINT="port"**
6. Autoriser le port UDP spécifié avec **LOCKD\_UDPOINT="port"**

**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 pu.

**9.7.3.1. 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.

**9.7.4. 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

Veillez utiliser le format *@group-name*, où *group-name* est le nom du netgroup NIS.

## 9.7.5. NFS sur RDMA

Pour activer le transport RDMA dans le serveur NFS du noyau Linux, veuillez utiliser la procédure suivante :

### Procédure 9.2. Activez RDMA à partir du serveur

1. Assurez-vous que le RPM RDMA est installé et que le service RDMA est activé en exécutant la commande suivante :

```
# yum install rdma; chkconfig --level 2345 rdma on
```

2. Assurez-vous que le paquet qui fournit le service nfs-rdma est installé et que le service est activé à l'aide de la commande suivante :

```
# yum install rdma; chkconfig --level 345 nfs-rdma on
```

3. Assurez-vous que le port RDMA est paramétré comme étant le port par défaut (le port par défaut sur Red Hat Enterprise Linux 6 est le port 2050). Pour ce faire, veuillez modifier le fichier `/etc/rdma/rdma.conf` de manière à paramétrer `NFSoRDMA_LOAD=yes` et `NFSoRDMA_PORT` sur le port souhaité.
4. Paramétrez le système de fichiers exporté comme à la normale pour les montages NFS.

Du côté client, veuillez utiliser la procédure suivante :

### Procédure 9.3. Activez RDMA à partir du client

1. Assurez-vous que le RPM RDMA est installé et que le service RDMA est activé en exécutant la commande suivante :

```
# yum install rdma; chkconfig --level 2345 rdma on
```

2. Montez la partition NFS exportée à l'aide de l'option RDMA lors de l'appel au montage. L'option du port peut être optionnellement ajoutée à l'appel.

```
# mount -t nfs -o rdma,port=port_number
```

## 9.8. 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.

### 9.8.1. 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.

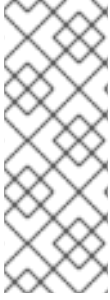
Il est également possible de restreindre l'accès au service **rpcbind**<sup>[3]</sup> avec des enveloppes TCP. La création de règles avec **iptables** peut également limiter l'accès aux ports utilisés par **rpcbind**, **rpc.mountd**, et **rpc.nfsd**.

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

### 9.8.2. 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.

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.



## 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 le *Guide de gestion des identités* de Red Hat.

Pour paramétrer `RPCSEC_GSS`, veuillez utiliser la procédure suivante :

### Procédure 9.4. 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=sys, 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 9.5, « 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/>.

#### 9.8.2.1. 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.

#### 9.8.3. 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 9.7.1, « 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**.

## 9.9. NFS ET RPCBIND



### NOTE

La section suivante s'applique uniquement aux implémentations NFSv2 ou NFSv3 qui nécessitent le service **rpcbind** pour une compatibilité ascendante.

L'utilitaire **rpcbind**<sup>[3]</sup> 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.

### 9.9.1. Résolution des problèmes NFS et rpcbind

Comme **rpcbind**<sup>[3]</sup> 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 9.7. Sortie de la comande **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.

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

## 9.10. 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.
- **man 8 nfsidmap** — explique la commande **nfsidmap** et liste les options communes.

### 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 à partir de 2001.
- *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.

---

[3] Le service **rpcbind** remplace **portmap**, qui était utilisé dans des versions précédentes de Red Hat Enterprise Linux pour mapper les numéros de programme RPC sur les combinaisons de numéros de port d'adresses IP. Pour obtenir des informations supplémentaires, veuillez consulter la [Section 9.1.1](#), « [Services requis](#) ».

## CHAPITRE 10. 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 :

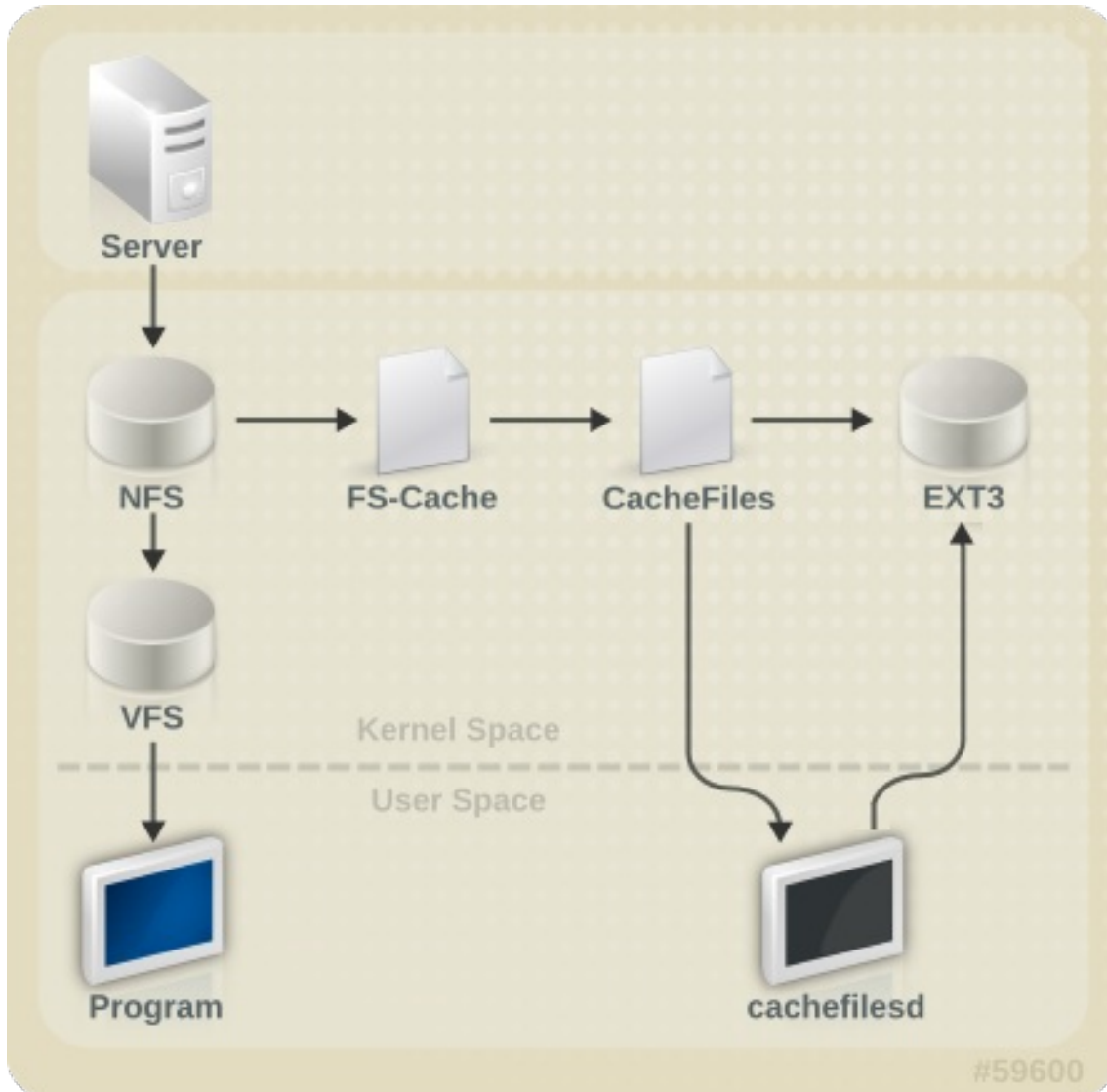


Figure 10.1. Aperçu de FS-Cache

FS-Cache est conçu pour être aussi transparent que possible aux utilisateurs et administrateurs d'un système. Contrairement à **cachefs** 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

À partir de Red Hat Enterprise Linux 6.2, **cachefilesd** n'est pas installé par défaut et doit donc l'être manuellement.

## 10.1. 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.

## 10.2. PARAMÉTRER UN CACHE

Actuellement, Red Hat Enterprise Linux 6 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 6 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 10.4, « 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
```

### 10.3. 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 10.3.2, « 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.

### 10.3.1. 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 10.1. 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**.

### 10.3.2. 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.

## 10.4. 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* ou d'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 ») :

$0 \leq \mathbf{bstop} < \mathbf{bcull} < \mathbf{brun} < 100$

$0 \leq \mathbf{fstop} < \mathbf{fcull} < \mathbf{frun} < 100$

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.

## 10.5. 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
```

## 10.6. 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 6. 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 11. 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 6.

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.

## 11.1. MISES À JOUR DE LA CONFIGURATION DU STOCKAGE PENDANT L'INSTALLATION

La configuration de l'installation des paramètres et périphériques suivants a été mise à jour sur Red Hat Enterprise Linux 6 :

### FCoE (« Fibre-Channel over Ethernet »)

**Anaconda** peut désormais configurer les périphériques de stockage FCoE pendant l'installation.

### Interface du filtre de périphériques de stockage

**Anaconda** possède désormais un contrôle amélioré sur les périphériques de stockage utilisés pendant l'installation. Vous pouvez maintenant contrôler quels périphériques sont disponibles ou visibles à l'installateur en plus de pouvoir voir quels périphériques sont utilisés pour le stockage du système. Il y a deux chemins pour le filtrage du périphérique :

#### Chemin de base

Pour les systèmes qui utilisent uniquement des disques attachés localement et des matrices RAID micrologiciel en tant que périphériques de stockage.

#### Chemin avancé

Pour les systèmes utilisant des périphériques SAN (par ex. : multipath, iSCSI, FCoE)

### Partitionnement automatique et /home

Le partitionnement automatique crée désormais un volume logique séparé pour le système de fichiers **/home** lorsque 50 Go ou plus sont disponibles pour l'allocation de volumes physiques LVM. Le système de fichiers root (/) sera limité à un maximum de 50 Go lors de la création d'un volume logique **/home** séparé, mais le volume logique **/home** augmentera afin d'occuper tout l'espace disponible restant dans le groupe de volumes.

## 11.2. APERÇU DES SYSTÈMES DE FICHIERS PRIS EN CHARGE

Cette section présente des informations techniques de base sur chaque système de fichiers pris en charge par Red Hat Enterprise Linux 6.

### Tableau 11.1. Spécifications techniques des systèmes de fichiers pris en charge

Système de fichiers	Taille maximale prise en charge	Décalage de fichiers maximal	Nombre maximal de sous-répertoires (par répertoire)	Profondeur maximale des liens symboliques	Prise en charge des ACL	Détails
Ext2	8 To	2 To	32,000	8	Oui	N/A
Ext3	16 To	2 To	32,000	8	Oui	<a href="#">Chapitre 5, Système de fichiers Ext3</a>
Ext4	16 To	16 To[a]	Illimité[b]	8	Oui	<a href="#">Chapitre 6, Le système de fichiers Ext4</a>
XFS	100 To	100 To[c]	Illimité	8	Oui	<a href="#">Chapitre 8, Le système de fichiers XFS</a>

[a] Cette taille de fichier maximale est basée sur des ordinateurs 64 bits. Sur un ordinateur 32 bits, la taille maximale des fichiers est de 8 To.

[b] Lorsque le compte des liens dépasse 65 000, celui-ci est réinitialisé sur 1 et n'augmentera plus.

[c] Cette taille de fichier maximale se trouve uniquement sur les ordinateurs 64 bits. Red Hat Enterprise Linux ne prend pas en charge XFS sur les ordinateurs 32 bits.



## NOTE

Les tailles maximales de fichiers et de systèmes de fichiers répertoriées sont ce que Red Hat a testé et prend en charge. Celles-ci ne prennent pas en compte les limites maximales théoriques.

Les colonnes de la taille maximale prise en charge et du décalage de fichier maximal supposent que les blocs fassent 4k.

## 11.3. 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 replie 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 12. 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).



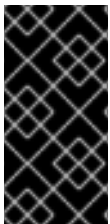
### NOTE

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

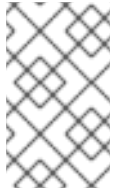
Red Hat ne recommande pas ceci, à moins que l'ordinateur ne démarre pas, que le système de fichiers soit extrêmement grand, ou que le système de fichiers se trouve sur un stockage distant. Il est possible de désactiver la vérification du système de fichiers lors du démarrage en paramétrant le sixième champ dans **/etc/fstab** sur 0.

### 12.1. 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

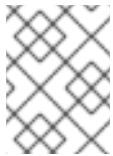
La plupart des vérificateurs de systèmes de fichiers possèdent un mode opératoire qui vérifie mais ne répare pas le système de fichiers. Dans ce mode, le vérificateur imprimera toutes les erreurs qu'il trouvera et toutes les actions qu'il aurait effectuées sans pour autant modifier le système de fichiers.

**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)**.

## 12.2. INFORMATIONS SPÉCIFIQUES AUX SYSTÈMES DE FICHIERS POUR FSCK

### 12.2.1. 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 superbloc 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 superbloc 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.

### 12.2.2. 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 **maxmem** MB. 0 peut être spécifié pour obtenir une estimation approximative de la quantité minimum de mémoire requise.

### -1 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 (adressage) 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 des superblocs.

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.

### 12.2.3. 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 13. 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 13.1, « [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 13.1. Commandes parted**

Commande	Description
<b>check <i>minor-num</i></b>	Effectuer une simple vérification du système de fichiers
<b>cp <i>depuis vers</i></b>	Copier le système de fichiers d'une partition à une autre ; <i>depuis</i> et <i>vers</i> sont les numéros mineurs des partitions
<b>help</b>	Afficher la liste des commandes disponibles
<b>mklabel <i>étiquette</i></b>	Créer une étiquette de disque pour la table de partitions
<b>mkfs <i>minor-num file-system-type</i></b>	Créer un système de fichiers de type <i>file-system-type</i>



Commande	Description
<b>mkpart</b> <i>part-type fs-type start-mb end-mb</i>	Créer une partition sans créer de nouveau système de fichiers
<b>mkpartfs</b> <i>part-type fs-type start-mb end-mb</i>	Créer une partition et créer le système de fichiers spécifié
<b>move</b> <i>minor-num start-mb end-mb</i>	Déplacer la partition
<b>name</b> <i>minor-num nom</i>	Nommer la partition uniquement pour disklabels Mac et PC98
<b>print</b>	Afficher la table de partitions
<b>quit</b>	Quitter <b>parted</b>
<b>rescue</b> <i>start-mb end-mb</i>	Secourir une partition perdue de <i>start-mb</i> à <i>end-mb</i>
<b>resize</b> <i>minor-num start-mb end-mb</i>	Redimensionner la partition de <i>start-mb</i> à <i>end-mb</i>
<b>rm</b> <i>minor-num</i>	Supprimer la partition
<b>select</b> <i>périphérique</i>	Sélectionner un autre périphérique à configurer
<b>set</b> <i>minor-num flag state</i>	Paramétrer l'indicateur sur une partition ; l'état est soit « on », soit « off »
<b>toggle</b> [ <i>NUMBER</i> [ <i>FLAG</i> ]]	Basculer l'état de l'indicateur <i>FLAG</i> sur le numéro de partition <i>NUMBER</i>
<b>unit</b> <i>UNIT</i>	Paramétrer l'unité par défaut sur <i>UNIT</i>

## 13.1. 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 13.1. 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 mégaoctets. Les types **Type** valides sont les métadonnées (« metadata ») « free », « primary », « extended », ou « logical ». **Filesystem** est le type de système de fichiers, qui peut être n'importe lequel des types suivants :

- ext2
- ext3
- fat16
- fat32
- hfs
- jfs
- linux-swap
- 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.

## 13.2. CRÉATION D'UNE PARTITION

**AVERTISSEMENT**

Ne pas tenter de créer une partition sur un périphérique en cours d'utilisation.

**Procédure 13.1. 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 13.4, « Redimensionnement d'une partition »](#) pour obtenir des détails.

**13.2.1. 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 6*.

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 **mkpart fs**, 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.

### 13.2.2. Formatage et étiquetage de la partition

Pour formater et étiqueter la partition, veuillez utiliser la procédure suivante :

#### Procédure 13.2. 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 :

```
# /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.

### 13.2.3. 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
```

### 13.3. SUPPRESSION DE PARTITION



#### AVERTISSEMENT

Ne tentez pas de supprimer une partition sur un périphérique en cours d'utilisation.

#### Procédure 13.3. 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
```

4. 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.

5. 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
```

6. La dernière étape consiste à la supprimer du fichier **/etc/fstab**. Trouvez la ligne qui déclare la partition supprimée, puis supprimez-la du fichier.

### 13.4. REDIMENSIONNEMENT D'UNE PARTITION



### AVERTISSEMENT

Ne tentez pas de redimensionner une partition sur un périphérique en cours d'utilisation.

#### Procédure 13.4. Redimensionner une partition

1. Avant de redimensionner 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 redimensionner la partition :

```
# parted /dev/sda
```

3. Affichez la table de la partition actuelle pour déterminer le numéro mineur de la partition à redimensionner, ainsi que les points de début et de fin de la partition :

```
# print
```

4. Pour redimensionner la partition, veuillez utiliser la commande **resize** suivie du numéro mineur de la partition, du point de départ en méga-octets, et du point d'arrivée en méga-octets.

#### Exemple 13.2. Redimensionner une partition

Par exemple:

```
resize 3 1024 2048
```



### AVERTISSEMENT

Une partition ne peut pas être plus grande que l'espace disponible sur le disque

5. Après avoir redimensionné la partition, veuillez utiliser la commande **print** pour confirmer que la partition a effectivement été redimensionnée correctement, et que son type de partition et son type de système de fichiers soient corrects.
6. Après avoir redémarré le système en mode normal, veuillez utiliser la commande **df** afin de vous assurer que la partition a bien été montée correctement et qu'elle soit reconnue avec la nouvelle taille.

## CHAPITRE 14. GESTIONNAIRE DE VOLUMES LOGIQUES LVM (« LOGICAL VOLUME MANAGER »)

LVM est un outil de gestion de volumes logiques qui inclut l'allocation de disques, l'entrelacement, la mise en miroir et le redimensionnement de volumes logiques.

Avec LVM, un disque dur ou un ensemble de disques durs est alloué à un ou plusieurs *volumes physiques*. Les volumes physiques LVM peuvent être placés sur d'autres périphériques bloc pouvant s'étendre sur deux disques ou plus.

Les volumes physiques sont combinés en *volumes logiques*, à l'exception de la partition `/boot/`. La partition `/boot/` ne peut pas se trouver dans un groupe de volumes logiques car le chargeur de démarrage ne peut pas la lire. Si la partition root (`/`) se trouve sur un volume logique, veuillez créer une partition `/boot/` séparée qui ne fera pas partie d'un groupe de volumes.

Comme un volume logique ne peut pas s'étendre sur plusieurs disques, veuillez créer un ou plusieurs volumes physiques par disque afin de permettre qu'il soit étendu sur plus d'un disque.

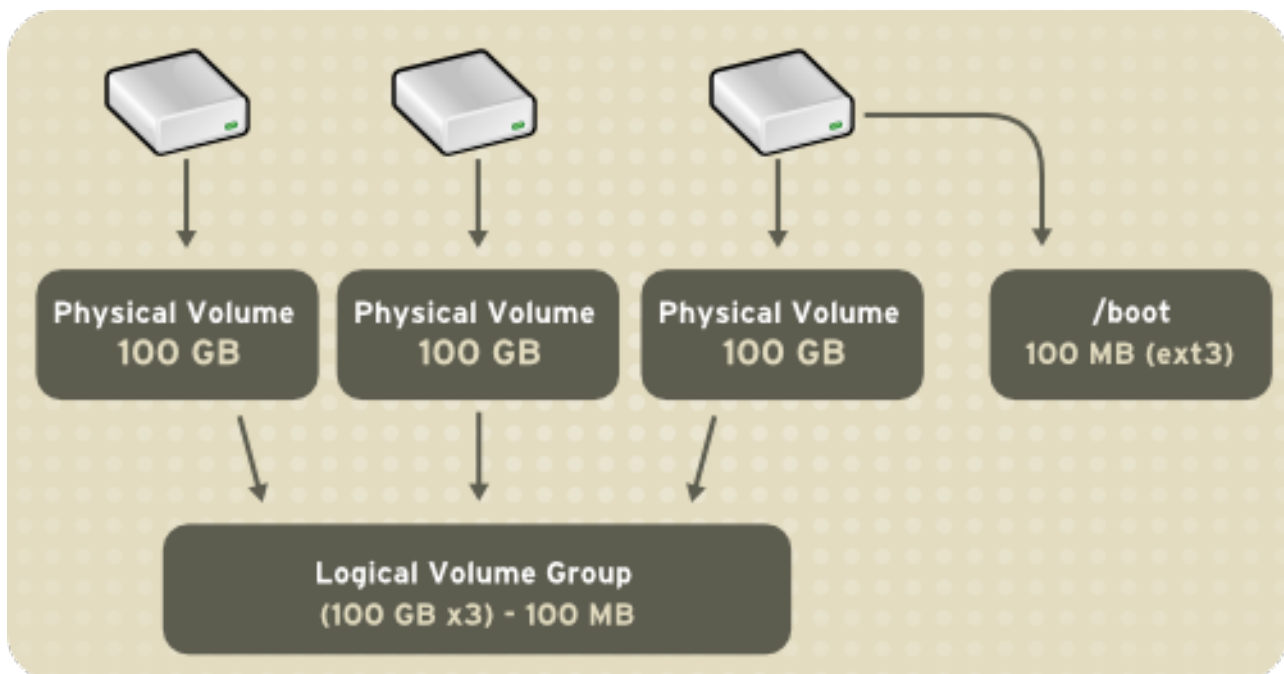
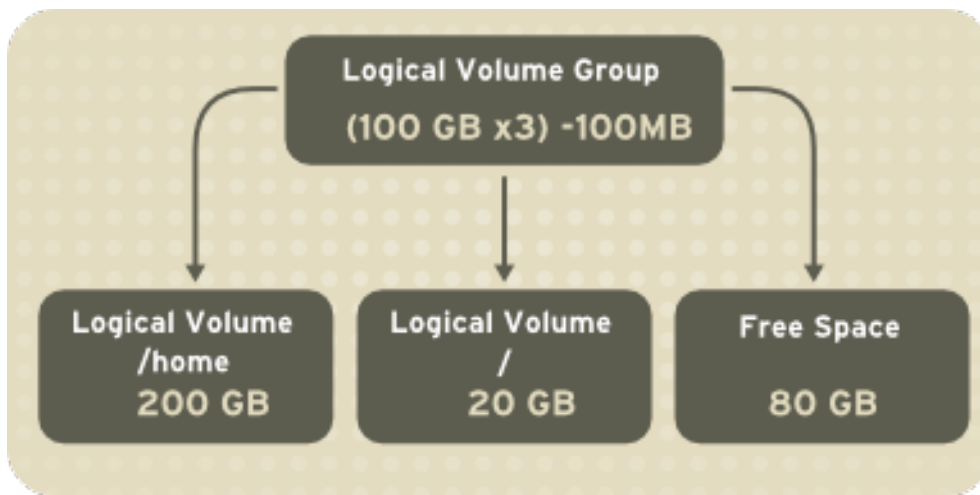


Figure 14.1. Volumes logiques

Les groupes de volume peuvent être divisés en *volumes logiques*, qui se voient assigner des points de montage, comme `/home` et `/` et des types de systèmes de fichiers, comme `ext2` ou `ext3`. Lorsque les « partitions » atteignent leur capacité totale, l'espace libre du groupe de volumes peut être ajouté au volume logique pour augmenter la taille de la partition. Lorsqu'un nouveau disque dur est ajouté au système, il peut être ajouté au groupe de volumes et la taille des partitions, qui sont des volumes logiques, peut être augmentée.



**Figure 14.2. Volumes logiques**

En revanche, si un système est partitionné avec le système de fichiers ext3, le disque dur est divisé en partitions de tailles définies. Si une partition est remplie, il n'est pas facile d'étendre la taille de la partition. Même si la partition est déplacée sur un autre disque dur, l'espace du disque dur d'origine devra être ré-alloué en tant que différente partition ou ne sera pas utilisé.



### IMPORTANT

Ce chapitre sur LVM/LVM2 se concentre sur l'utilisation de l'outil d'administration de l'interface utilisateur graphique LVM, c'est-à-dire **system-config-lvm**. Pour des informations complètes sur la création et la configuration de partitions LVM dans des stockages en cluster et sans clusters, veuillez consulter le guide *Administration du gestionnaire de volumes logique LVM*, qui est aussi fourni par Red Hat.

En outre, le *Guide d'installation* de Red Hat Enterprise Linux 6 documente aussi comment créer et configurer des volumes logiques LVM pendant l'installation. Pour obtenir des informations supplémentaires, veuillez consulter la section *Créer un volume logique LVM* du *Guide d'installation* de Red Hat Enterprise Linux 6.

## 14.1. QU'EST-CE QUE LVM2 ?

LVM version 2, ou LVM2, était la version par défaut de Red Hat Enterprise Linux 5, qui utilise le pilote du mappeur de périphériques contenu dans le noyau 2.6. LVM2 peut être mis à niveau à partir des versions de Red Hat Enterprise Linux exécutant le noyau 2.4.

## 14.2. UTILISATION DE SYSTEM-CONFIG-LVM

L'utilitaire LVM vous permet de gérer des volumes logiques à l'intérieur de X Windows ou graphiquement. Il n'est pas offert pré-installé. Pour l'installer, veuillez exécuter :

```
# yum install system-config-lvm
```

Vous pouvez ensuite accéder à l'application en sélectionnant à partir du panneau de menus **Système** → **Administration** → **Gestion LVM**. Alternativement, vous pouvez démarrer l'utilitaire LVM en saisissant **system-config-lvm** à partir d'un terminal.

Dans l'exemple utilisé dans cette section figurent les détails du groupe de volumes créé pendant l'installation :



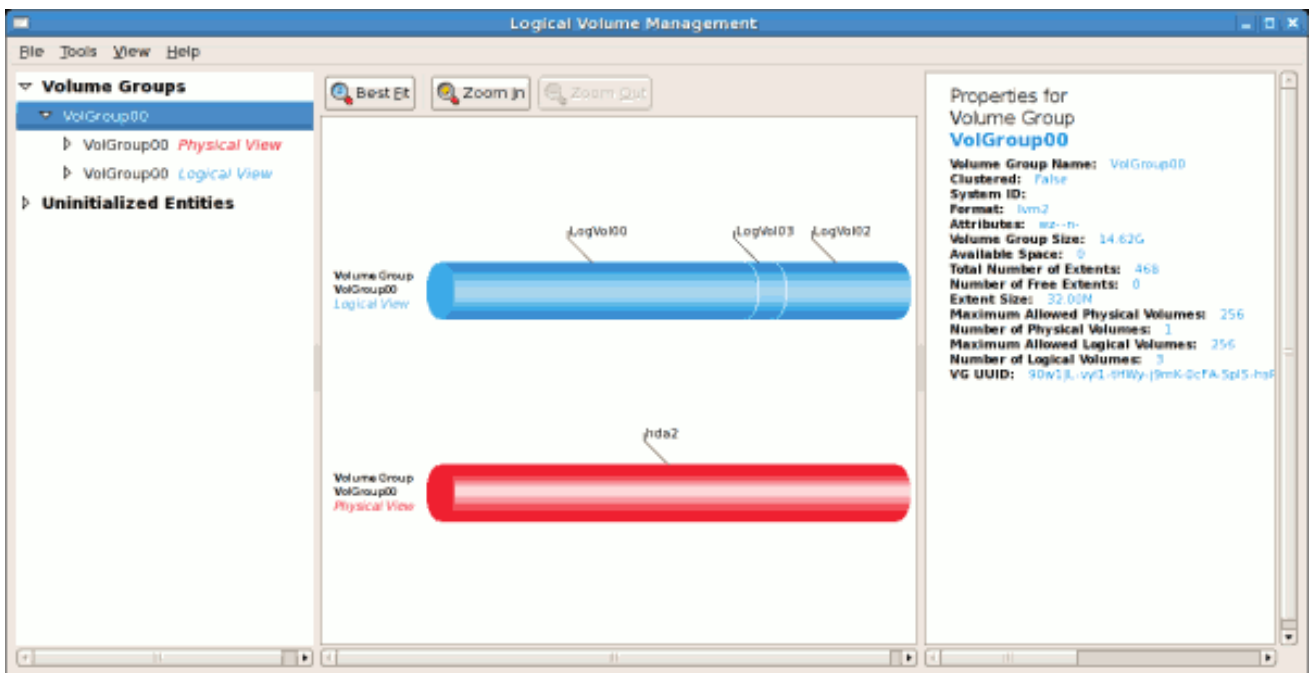
**Exemple 14.1. Créer un groupe de volumes pendant l'installation**

```

/boot - (Ext3) système de fichiers. Affiché sous « Entités non-
initialisées ». (NE PAS initialiser cette partition).
LogVol00 - (LVM) contient le répertoire (/) (312 espaces).
LogVol02 - (LVM) contient le répertoire (/home) (128 espaces).
LogVol03 - (LVM) swap (28 espaces).

```

Les volumes logiques ci-dessus ont été créés dans l'entité du disque `/dev/hda2` tandis que `/boot` a été créé dans `/dev/hda1`. Le système consiste également en « entités non-initialisées », qui sont illustrées dans [Exemple 14.2, « Entrées non-initialisées »](#). La figure ci-dessous illustre la fenêtre principale dans l'utilitaire LVM. Les affichages logiques et physiques de la configuration ci-dessus sont illustrés ci-dessous. Les trois volumes logiques existent sur le même volume physique (hda2).



**Figure 14.3. Fenêtre principale de LVM**

La figure ci-dessous illustre l'affichage physique du volume. Dans cette fenêtre, vous pouvez sélectionner et supprimer un volume d'un groupe de volumes ou migrer les extensions du volume vers un autre groupe de volumes. Les étapes pour migrer les extensions sont discutées dans [Figure 14.10, « Migrer des extensions »](#).

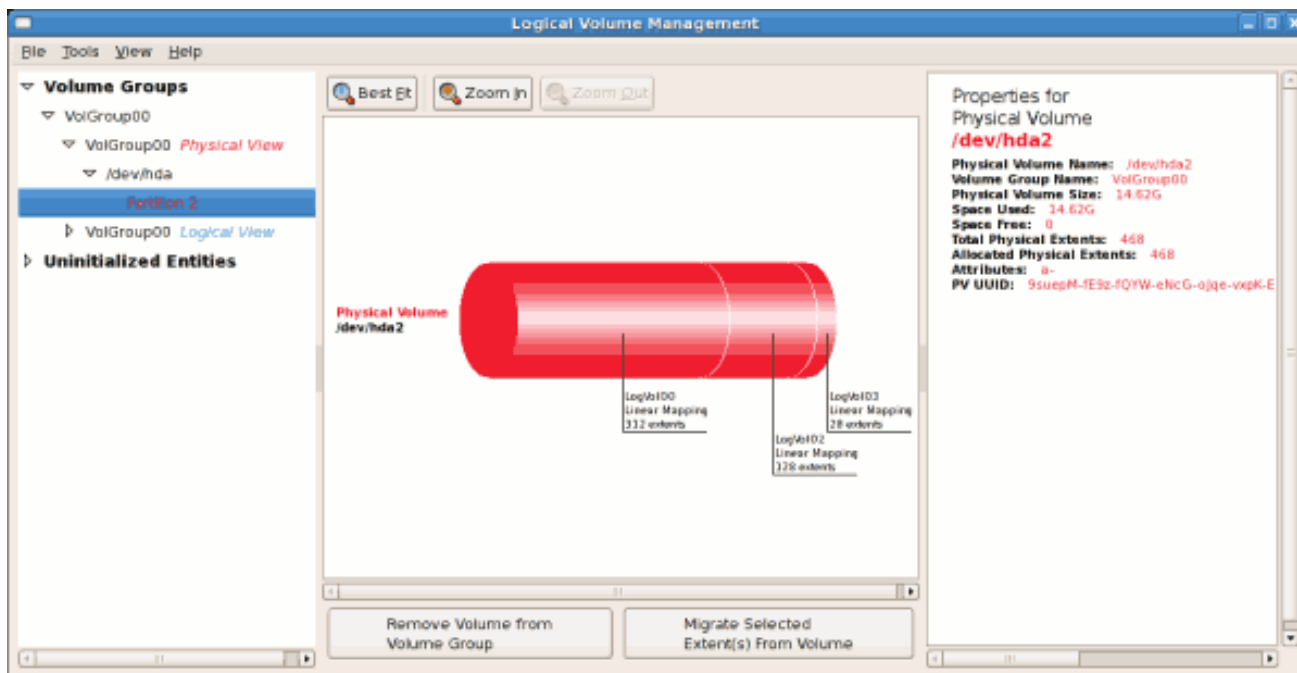


Figure 14.4. Fenêtre d'affichage physique

L'illustration ci-dessous montre l'affichage logique du groupe de volumes sélectionné. Les tailles des volumes logiques individuels sont également illustrées.

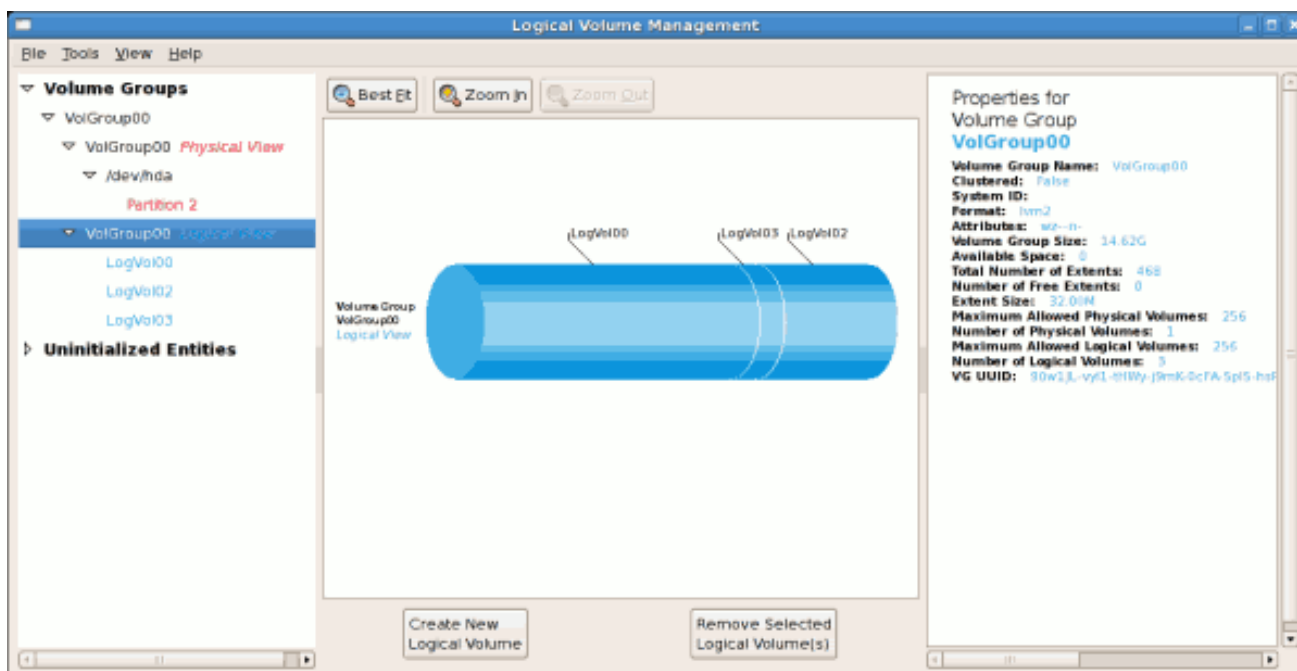


Figure 14.5. Fenêtre d'affichage logique

Sur la colonne de gauche, vous pouvez sélectionner les volumes logiques individuels dans le groupe de volumes pour afficher leurs détails supplémentaires. Dans cet exemple, l'objectif est de renommer le volume logique « LogVol03 » « Swap ». Pour effectuer cette opération, veuillez sélectionner le volume logique respectif à partir de la liste (contrairement à l'image) puis cliquez sur le bouton **Modifier les propriétés**. Ceci affichera la fenêtre « Modification de volumes logiques », à partir de laquelle vous pouvez modifier le nom du volume logique, sa taille (en extensions, giga-octets, méga-octets, ou kilo-octets) et utiliser l'espace disponible restant dans un groupe de volumes logiques. L'illustration ci-dessous le montre.

La taille de ce volume logique ne peut pas être modifiée car il n'y a pas d'espace libre dans le groupe de

volumes. S'il restait de l'espace libre, cette option aurait été activée (veuillez consulter [Figure 14.17](#), « [Modifier le volume logique](#) »). Cliquez sur le bouton **Valider** pour enregistrer vos modifications (ceci remontera le volume). Pour annuler vos modifications, veuillez cliquer sur le bouton **Annuler**. Pour rétablir les paramètres du dernier instantané, veuillez cliquer sur le bouton **Rétablir**. Un instantané peut être créé en cliquant sur le bouton **Créer un instantané** dans la fenêtre de l'utilitaire LVM. Si le volume logique sélectionné est en cours d'utilisation par le système, par exemple le répertoire root, cette tâche n'aboutira pas car le volume ne peut pas être démonté.

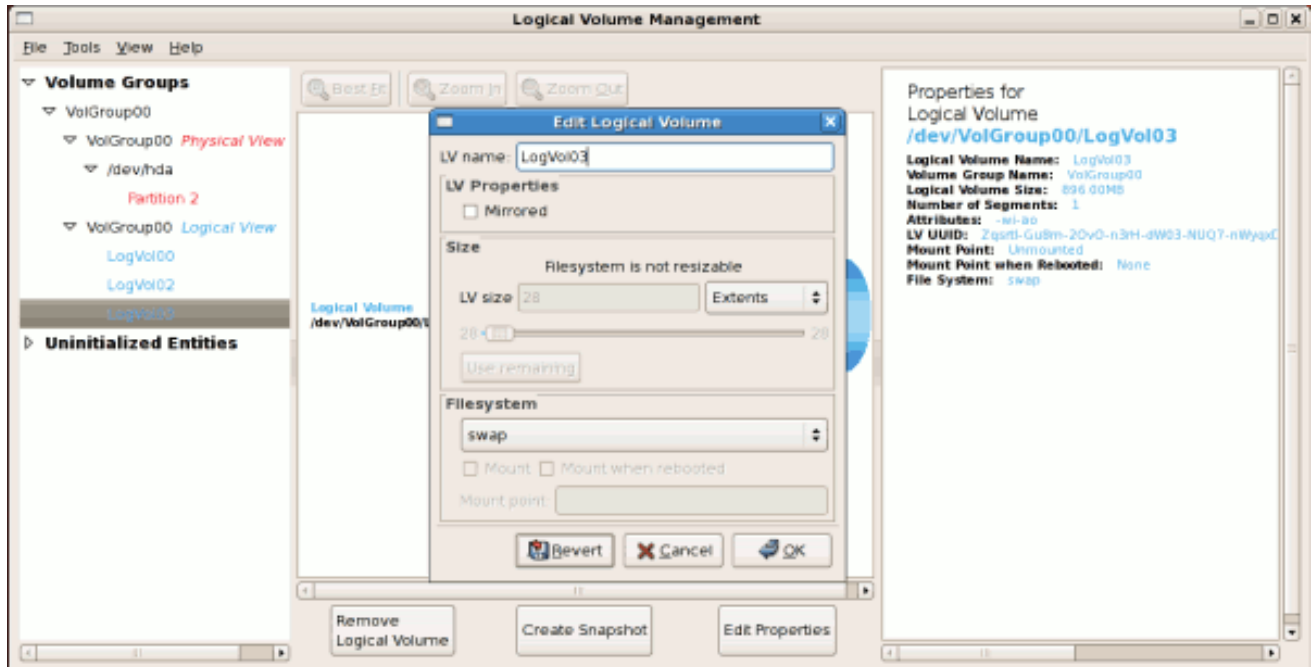
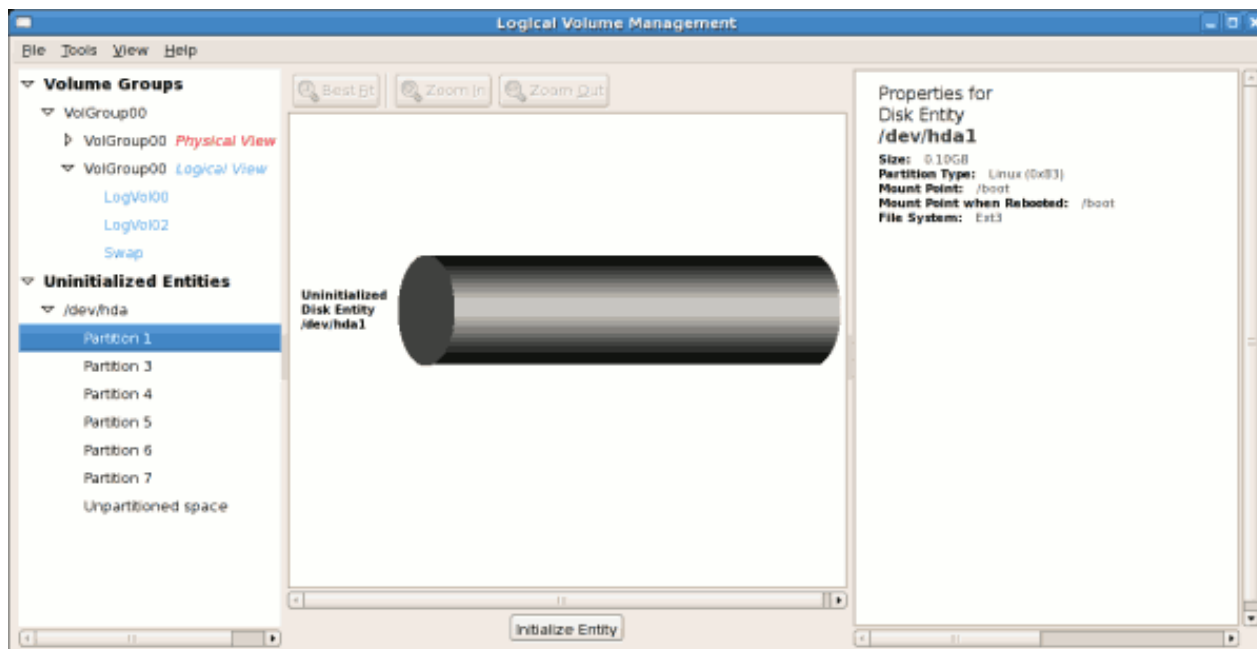


Figure 14.6. Modifier le volume logique

### 14.2.1. Utiliser des entités non-initialisées

Les « entités non-initialisées » sont composées d'espace non partitionné et de systèmes de fichiers non-LVM. Dans cet exemple, les partitions 3, 4, 5, 6 et 7 ont été créées pendant l'installation et de l'espace non partitionné est resté sur le disque dur. Affichez chaque partition et assurez-vous de bien lire les « Propriétés de l'entité du disque » sur la colonne de droite de la fenêtre afin d'être sûr(e) de ne pas supprimer de données critiques. Dans cet exemple, la partition 1 ne peut pas être initialisée car il s'agit de la partition **/boot**. Des entités non-initialisées sont illustrées ci-dessous.

#### Exemple 14.2. Entrées non-initialisées



Dans cet exemple, la partition 3 sera initialisée et ajoutée à un groupe de volumes. Pour initialiser une partition ou un espace non-partitionné, veuillez sélectionner la partition et cliquez sur le bouton **Initialiser l'entité**. Une fois l'initialisation effectuée, un volume sera répertorié dans la liste « Volumes non-alloués ».

#### 14.2.2. Ajouter des volumes non alloués à un groupe de volumes

Une fois initialisé, un volume sera répertorié dans la liste « Volumes non-alloués ». L'illustration ci-dessous montre une partition non allouée (Partition 3). Respectivement, les boutons au bas de la fenêtre vous permettent de :

- créer un nouveau groupe de volumes,
- ajouter le volume non-alloué à un groupe de volumes,
- supprimer le volume de LVM.

Pour ajouter le volume à un groupe de volumes, cliquez sur le bouton **Ajouter à un groupe de volumes**.

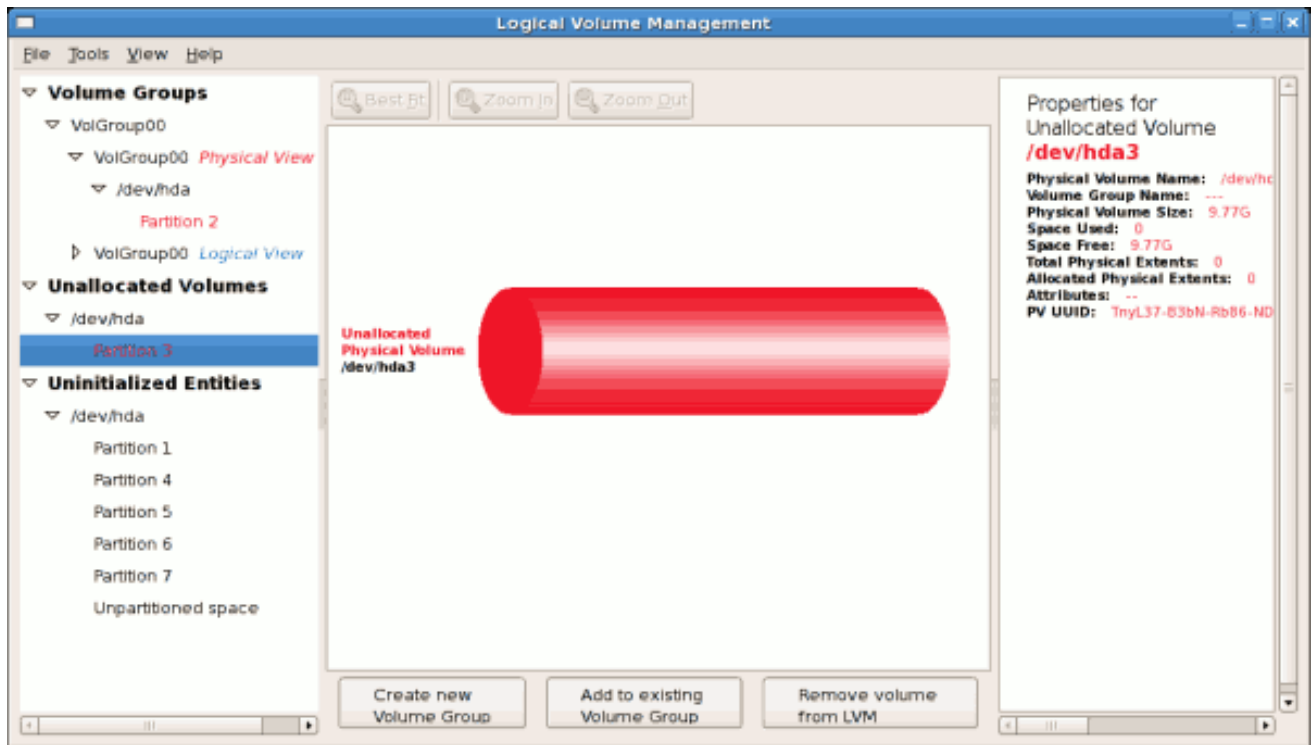
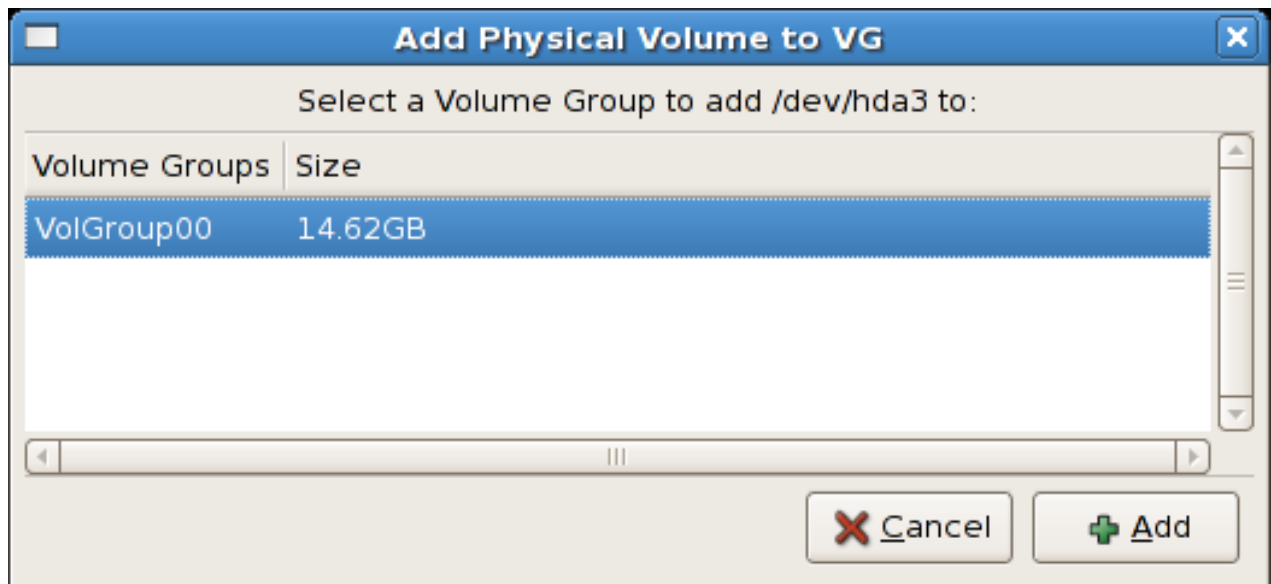


Figure 14.7. Volumes non-alloués

Cliquer sur le bouton **Ajouter au groupe de volumes** affichera une fenêtre contextuelle répertoriant les groupes de volumes auxquels vous pouvez ajouter le volume physique que vous souhaitez initialiser. Un groupe de volumes peut s'étendre sur un ou plusieurs disques durs.

#### Exemple 14.3. Ajouter un volume physique au groupe de volumes

Comme indiqué dans l'exemple ci-dessous, il n'existe qu'un seul groupe de volumes.

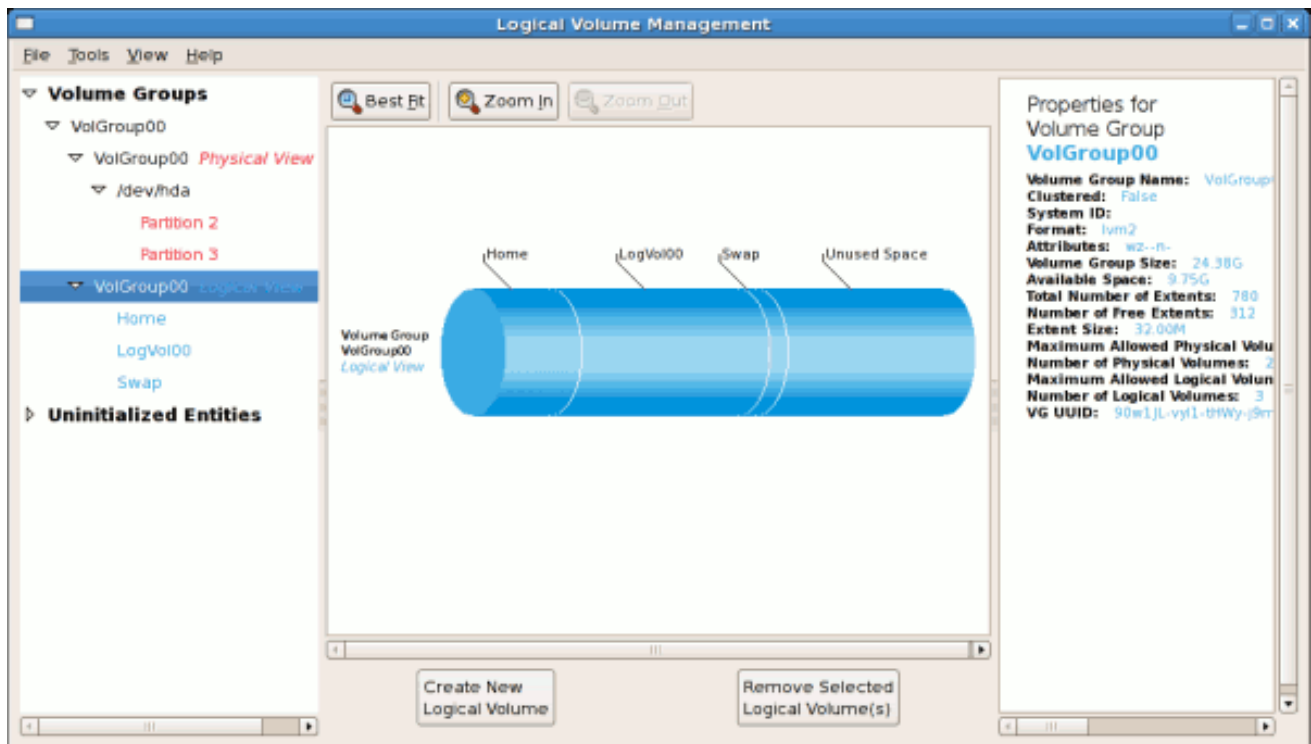


Une fois ajouté à un groupe de volumes, le nouveau volume logique est automatiquement ajouté à l'espace inutilisé du groupe de volumes sélectionné. Vous pouvez utiliser l'espace inutilisé pour :

- créer un nouveau volume logique (cliquez sur le bouton **Créer un(plusieurs) nouveau(x) volume(s) logique(s)**),

- sélectionner un des volumes logiques et augmenter le nombre d'extensions (voir [Section 14.2.6, « Étendre un groupe de volumes »](#)),
- sélectionner un volume logique et le supprimer du groupe de volumes en cliquant sur le bouton **Supprimer le(s) volume(s) logique(s) sélectionné(s)**. Il n'est pas possible de sélectionner de l'espace inutilisé pour effectuer cette opération.

L'illustration ci-dessous montre l'affichage logique de « VolGroup00 » après avoir ajouté le nouveau groupe de volumes.



**Figure 14.8. Affichage logique du groupe de volumes**

Dans l'illustration ci-dessous, les entités non-initialisées (partitions 3, 5, 6 et 7) ont été ajoutées au groupe « VolGroup00 ».

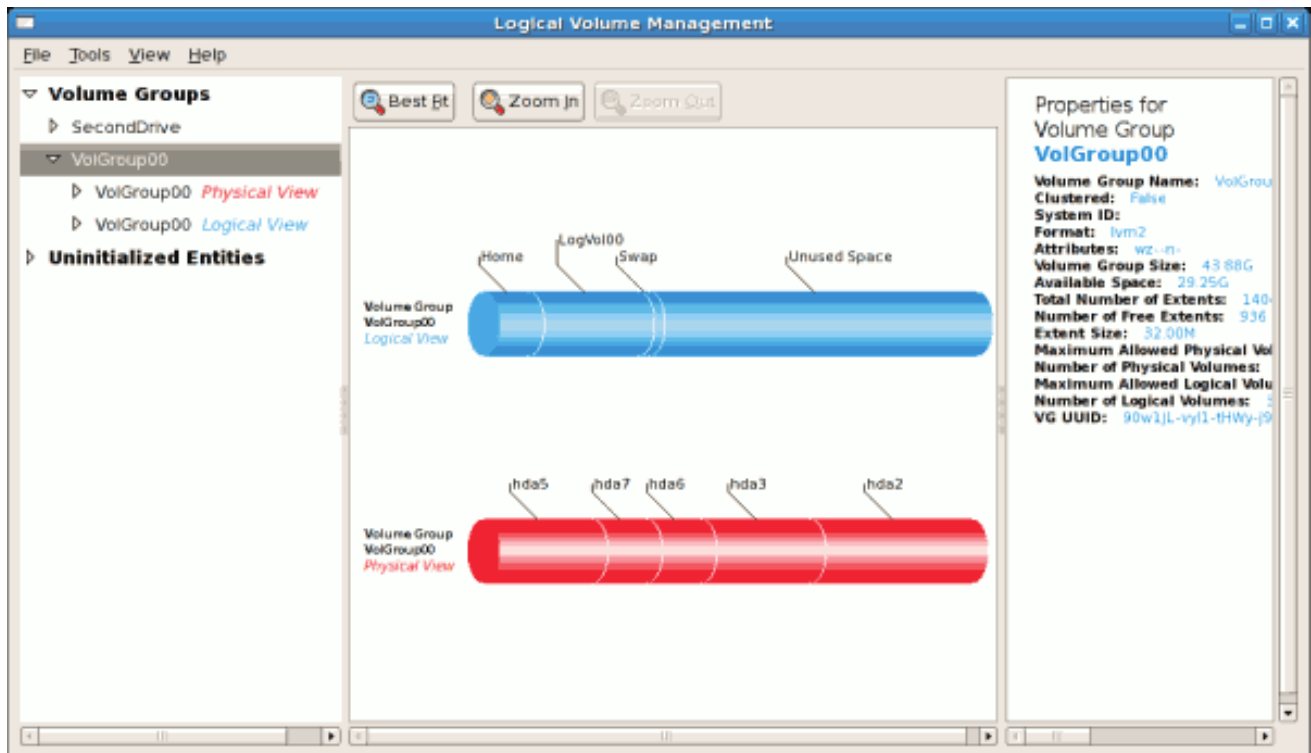


Figure 14.9. Affichage logique du groupe de volumes

### 14.2.3. Migrer des extensions

Pour migrer des extensions à partir d'un volume physique, sélectionnez le volume dans la liste du volet de gauche, surlignez les extensions souhaitées dans la fenêtre centrale, et cliquez sur le bouton **Migrer l'extension (ou les extension) sélectionnée(s) du volume**. Vous devrez posséder suffisamment d'extensions libres pour migrer des extensions dans un groupe de volumes. Un message d'erreur sera affiché si vous ne possédez pas suffisamment d'extensions libres. Pour résoudre ce problème, veuillez étendre le groupe de volumes (voir [Section 14.2.6, « Étendre un groupe de volumes »](#)). Si un nombre d'extensions libres suffisant est détecté dans le groupe de volumes, une fenêtre contextuelle sera affichée, à partir de laquelle vous pourrez sélectionner la destination des extensions ou laisser LVM choisir automatiquement les volumes physiques vers lesquels effectuer la migration. Ceci est illustré ci-dessous.

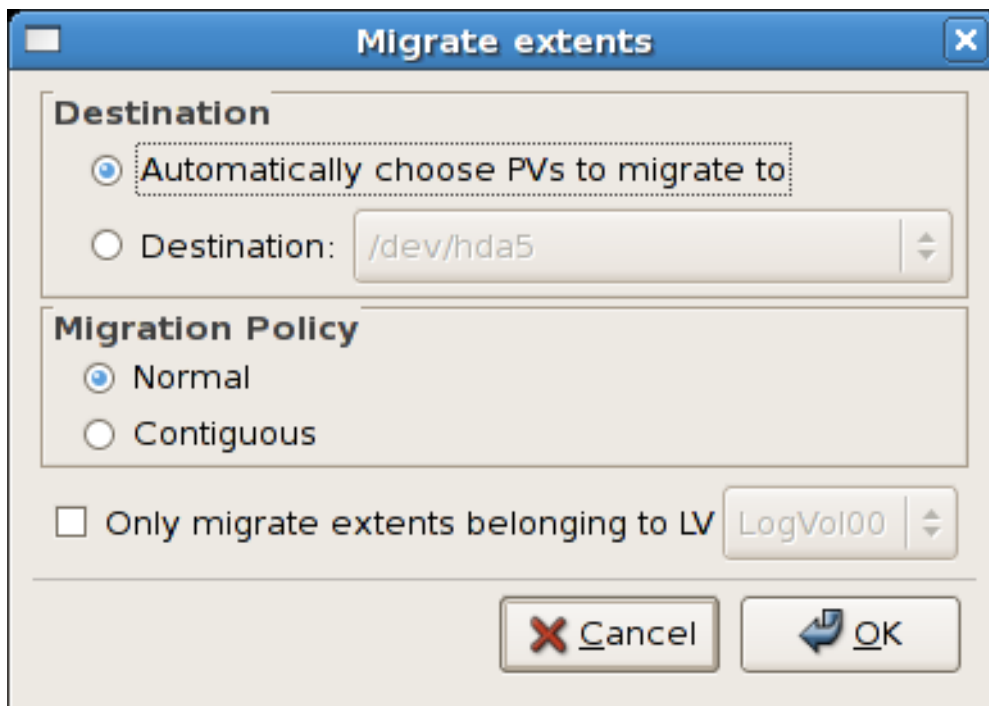


Figure 14.10. Migrer des extensions

L'illustration ci-dessous montre une migration d'extensions en cours. Dans cet exemple, les extensions ont été déplacées vers la « Partition 3 ».

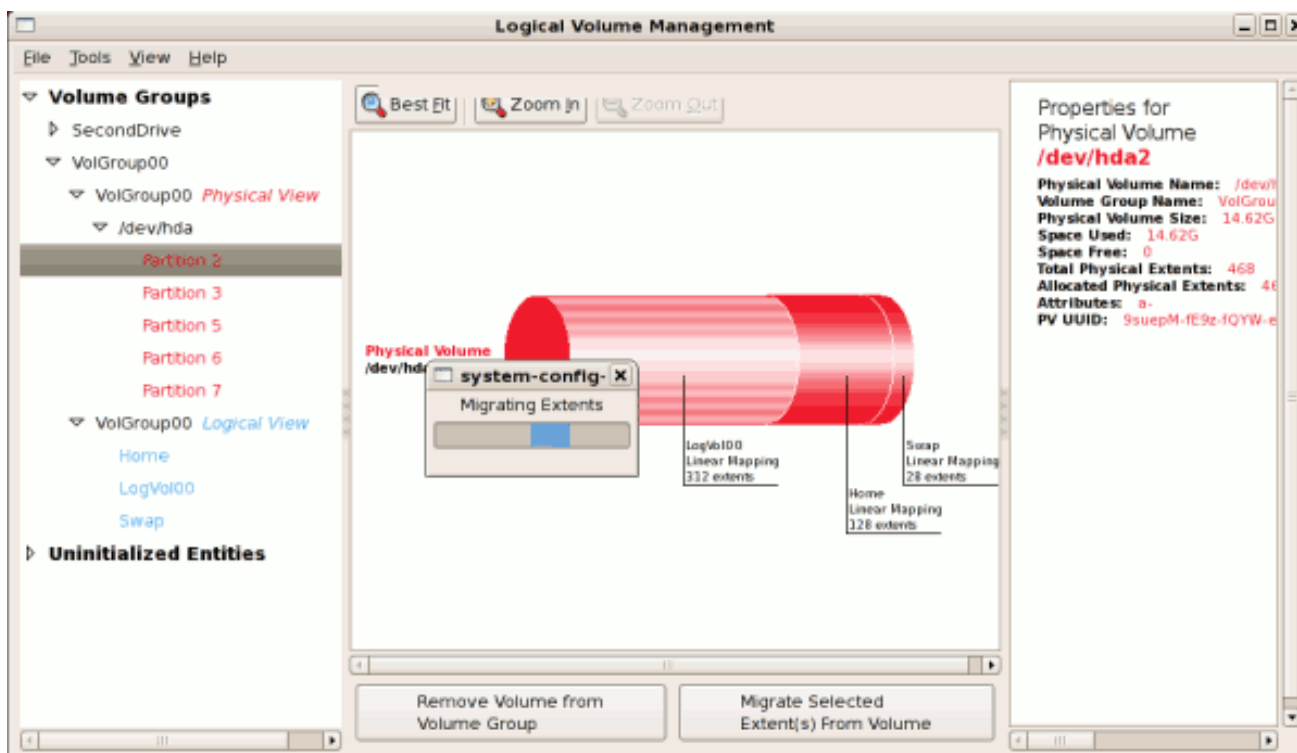


Figure 14.11. Migration d'extensions en cours

Une fois les extensions déplacées, il restera de l'espace inutilisé sur le volume physique. L'illustration ci-dessous montre l'affichage logique et physique pour le groupe de volumes. Les extensions de LogVol00, qui se trouvaient initialement dans hda2 se trouvent désormais dans hda3. La migration d'extensions vous permet de déplacer des volumes logiques en cas de mise à niveau de disque dur ou de mieux gérer votre espace disque.



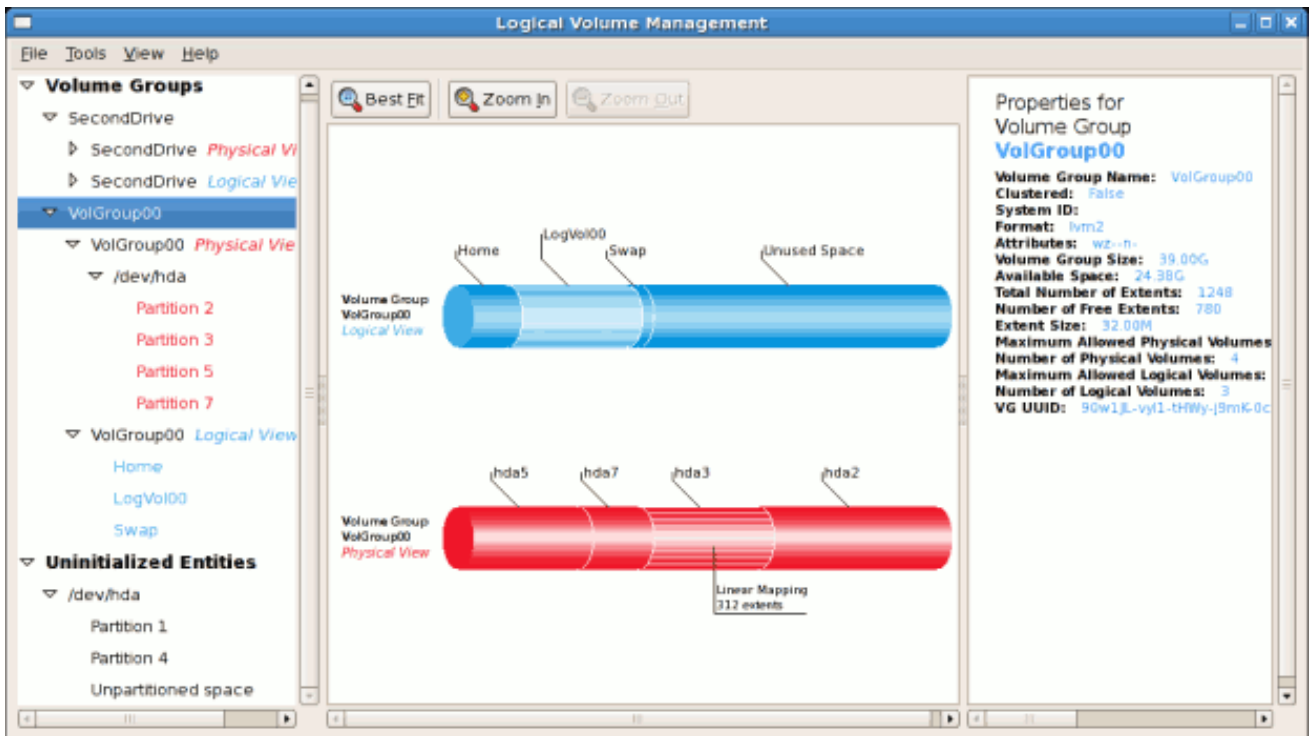


Figure 14.12. Affichage logique et physique d'un groupe de volumes

#### 14.2.4. Ajouter un nouveau disque dur avec LVM

Dans cet exemple, un nouveau disque dur IDE a été ajouté. L'illustration ci-dessous montre les détails du nouveau disque dur. Comme dans la figure ci-dessous, le disque est non-initialisé et n'est pas monté. Pour initialiser une partition, veuillez cliquer sur le bouton **Initialiser l'entité**. Pour obtenir plus de détails, veuillez consulter [Section 14.2.1, « Utiliser des entités non-initialisées »](#). Une fois initialisé, LVM ajoutera le nouveau volume à la liste de volumes non-alloués comme indiqué dans [Exemple 14.4, « Créer un nouveau groupe de volumes »](#).

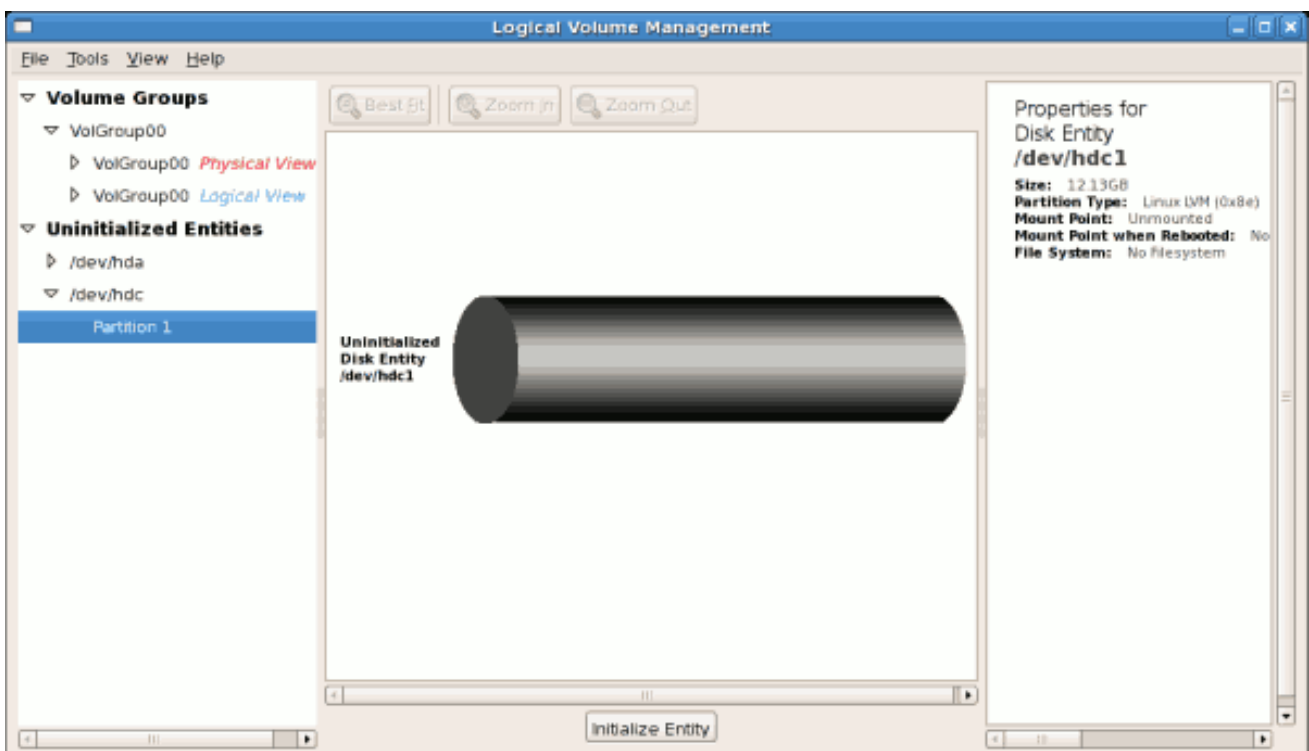


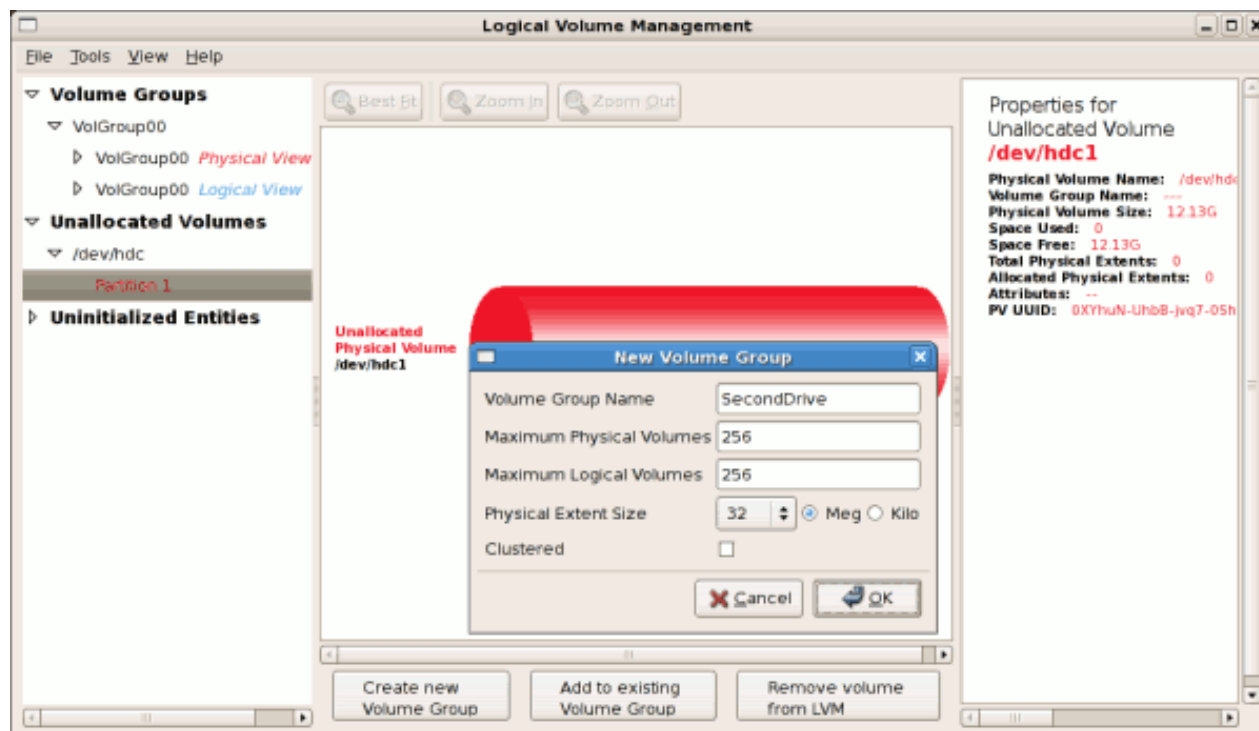
Figure 14.13. Disque dur non-initialisé

## 14.2.5. Ajouter un nouveau groupe de volumes

Une fois initialisé, LVM ajoutera le nouveau volume à la liste des volumes non-alloués, à partir de laquelle vous pourrez l'ajouter à un groupe de volumes ou créer un nouveau groupe de volumes. Vous pouvez également supprimer le volume de LVM. Si le volume est supprimé de LVM, il sera ajouté à la liste « Entités non-initialisées », comme illustré dans [Figure 14.13](#), « [Disque dur non-initialisé](#) ».

### Exemple 14.4. Créer un nouveau groupe de volumes

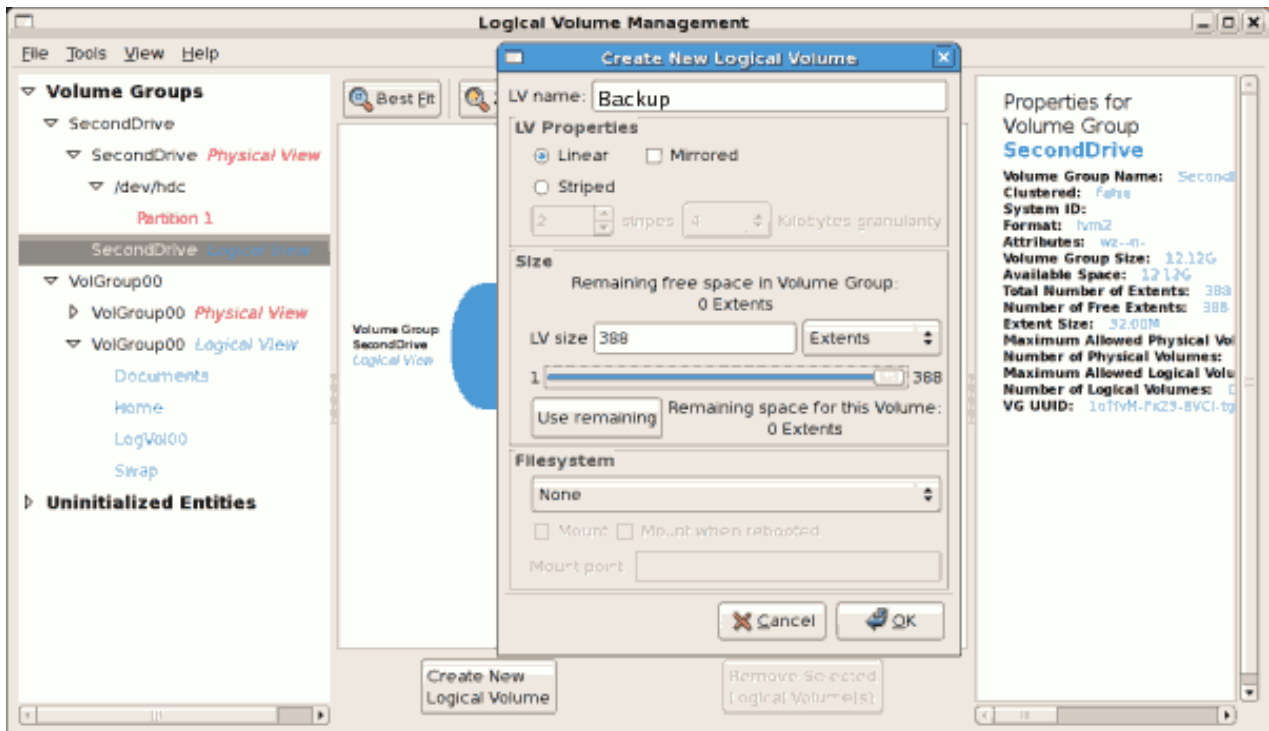
Comme indiqué dans l'exemple ci-dessous, un nouveau groupe de volumes a été créé.



Une fois créé, un nouveau groupe de volumes sera affiché dans la liste des groupes de volumes comme illustré ci-dessous. L'affichage logique montrera le nouveau groupe de volumes avec de l'espace inutilisé car aucun volume logique n'a été créé. Pour créer un volume logique, sélectionnez le groupe de volumes et cliquez sur bouton **Créer un nouveau volume logique** comme illustré ci-dessous. Sélectionnez les extensions que vous souhaitez utiliser sur le groupe de volumes.

### Exemple 14.5. Sélectionner les extensions

Dans cet exemple, toutes les extensions dans le groupe de volumes ont été utilisées pour créer le nouveau volume logique.



L'illustration ci-dessous montre l'affichage physique du nouveau groupe de volumes. Le nouveau volume logique nommé « Backups » (copies de sauvegarde) dans ce groupe de volumes est aussi répertorié.

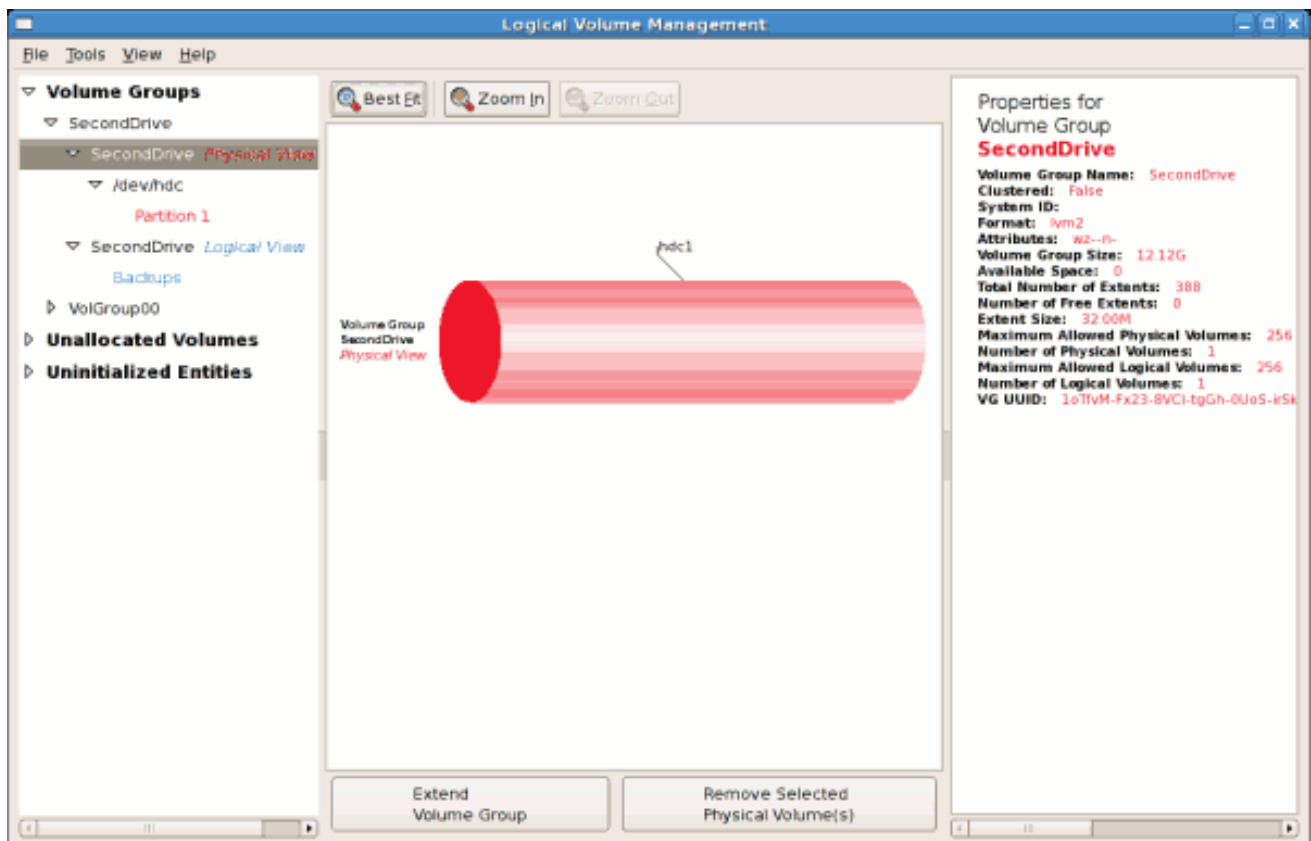


Figure 14.14. Affichage physique du nouveau groupe de volumes

### 14.2.6. Étendre un groupe de volumes

Dans cet exemple, l'objectif était d'étendre le nouveau groupe de volumes afin d'inclure une entité non-initialisée (partition). Ce faire augmente la taille ou le nombre d'extensions du groupe de volumes. Pour

étendre le groupe de volumes, assurez-vous que l'option « Affichage physique » sur le volet de gauche soit bien sélectionnée dans le groupe de volumes souhaité. Puis cliquez sur le bouton **Étendre le groupe de volumes**. Ceci affichera la fenêtre « Étendre le groupe de volumes » comme illustré ci-dessous. Sur la fenêtre « Étendre le groupe de volumes », vous pouvez sélectionner des entités de disque (partitions) pour ajouter le groupe de volumes. Assurez-vous de vérifier le contenu de toute « Entité de disque non-initialisée » (partitions) afin d'éviter de supprimer des données critiques (voir [Figure 14.13](#), « **Disque dur non-initialisé** »). Dans l'exemple ci-dessous, l'entité de disque (partition) `/dev/hda6` a été sélectionnée.

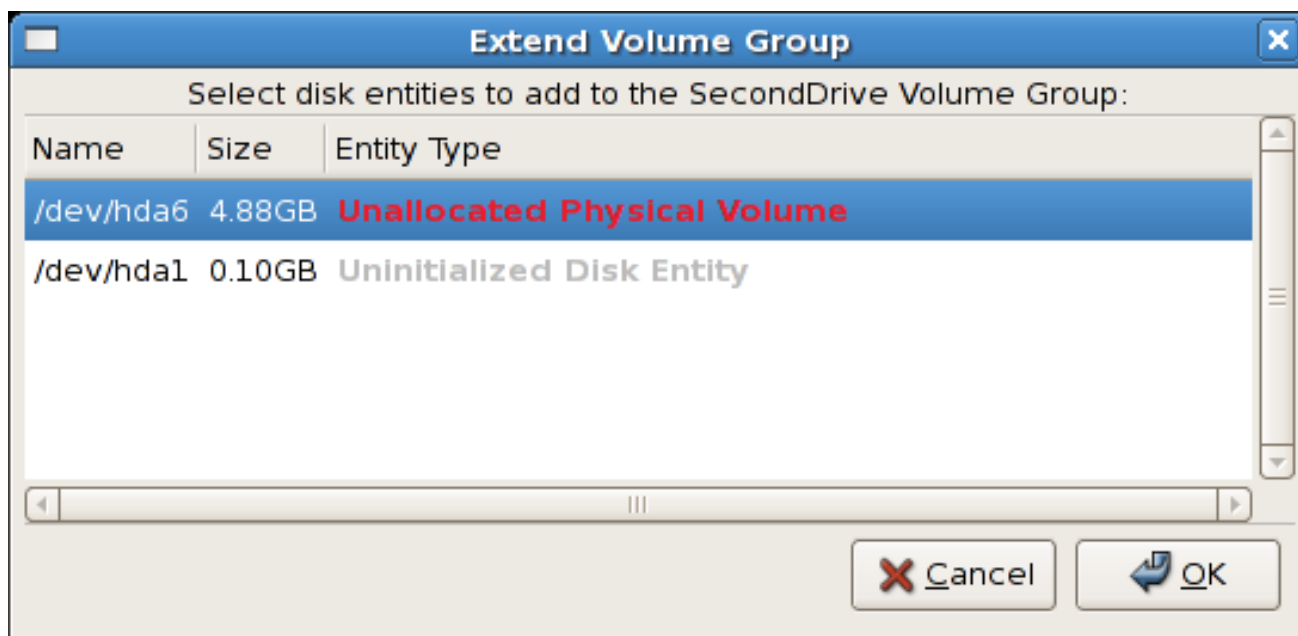


Figure 14.15. Sélectionner les entités de disque

Une fois ajouté, le nouveau volume sera ajouté en tant qu'« Espace non utilisé » dans le groupe de volumes. La figure ci-dessous illustre les affichages physiques et logiques du groupe de volumes après son extension.

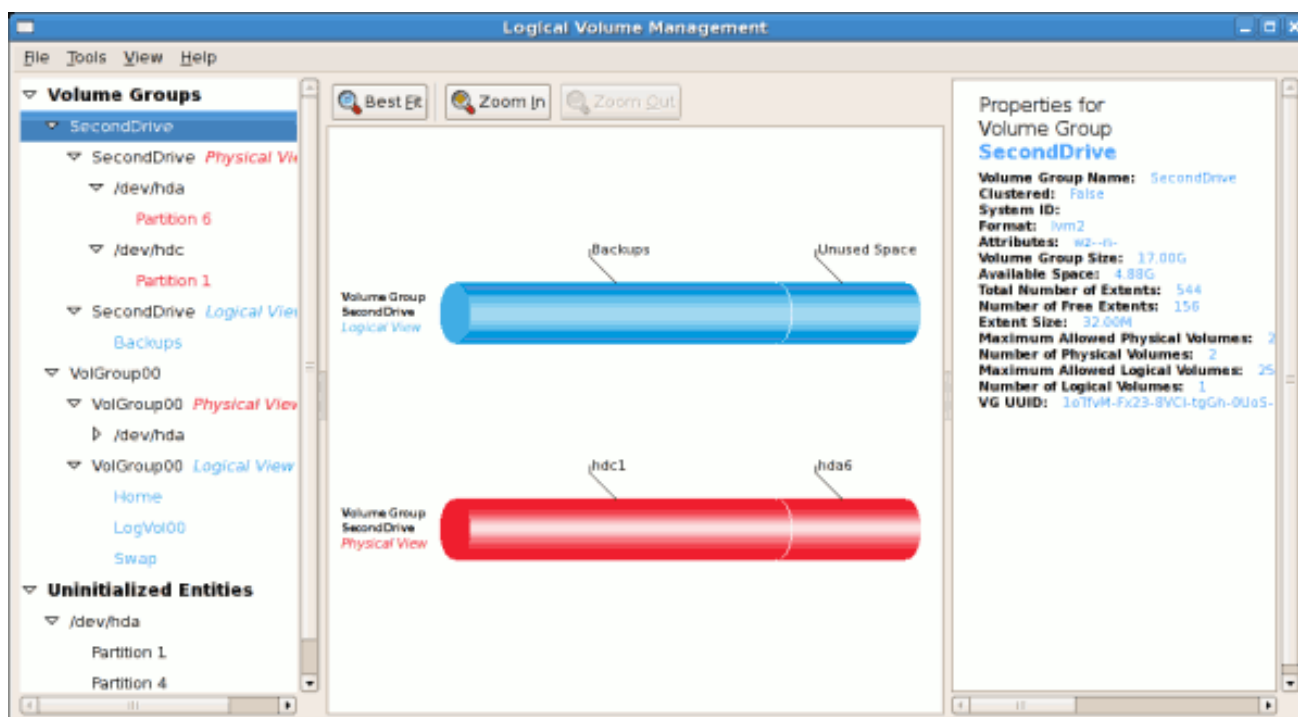


Figure 14.16. Affichage logique et physique d'un groupe de volumes étendu

### 14.2.7. Modifier un volume logique

L'utilitaire LVM permet de sélectionner un volume logique dans le groupe de volumes et de modifier son nom, sa taille, et de spécifier des options de système de fichiers. Dans cet exemple, le volume logique nommé « Backups » a été étendu sur l'espace restant du groupe de volumes.

Cliquer sur le bouton **Modifier les propriétés** affichera la fenêtre contextuelle « Modifier le volume logique », à partir de laquelle vous pourrez modifier les propriétés du volume logique. Dans cette fenêtre, vous pourrez également monter le volume après avoir effectué les modifications et le monter lorsque le système est redémarré. Vous devriez indiquer le point de montage. Si le point de montage spécifié n'existe pas, une fenêtre contextuelle s'affichera et vous demandera de le créer. La fenêtre « Modifier le volume logique » est illustrée ci-dessous.

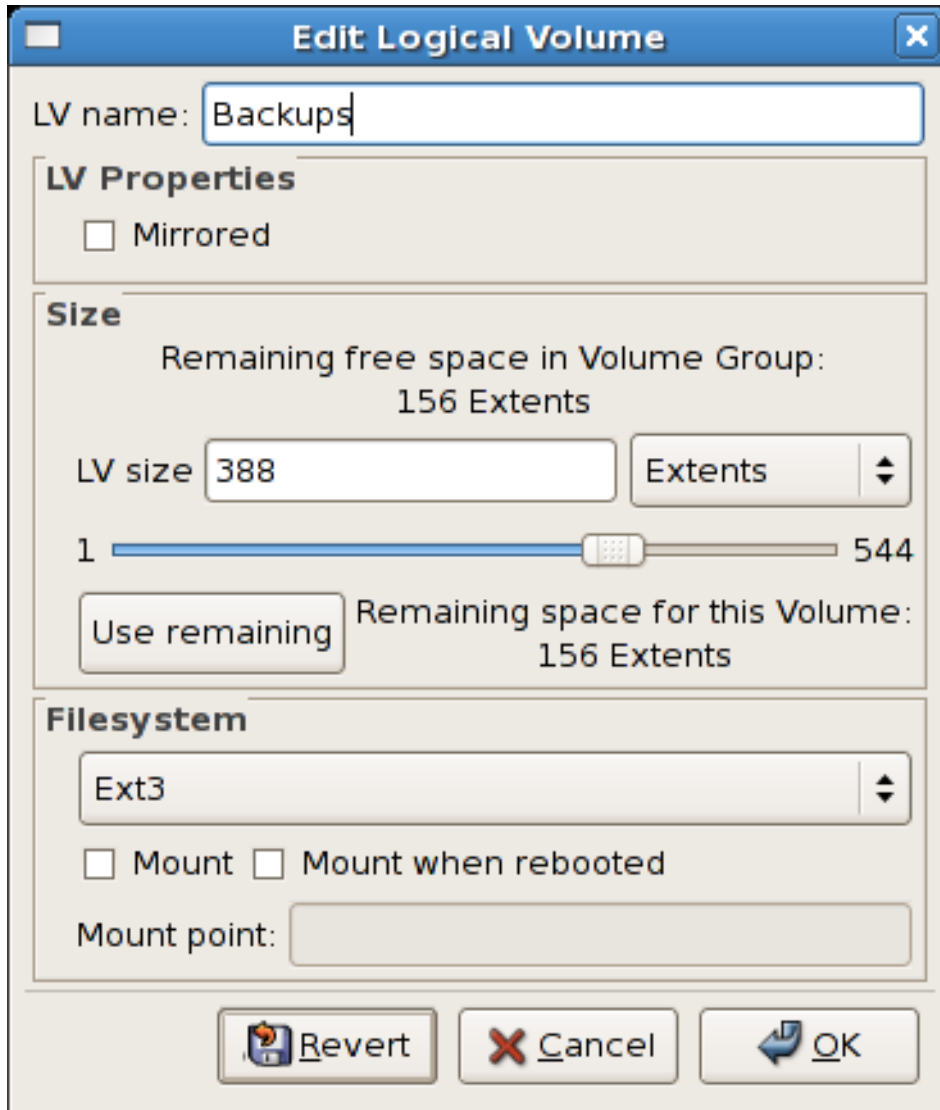
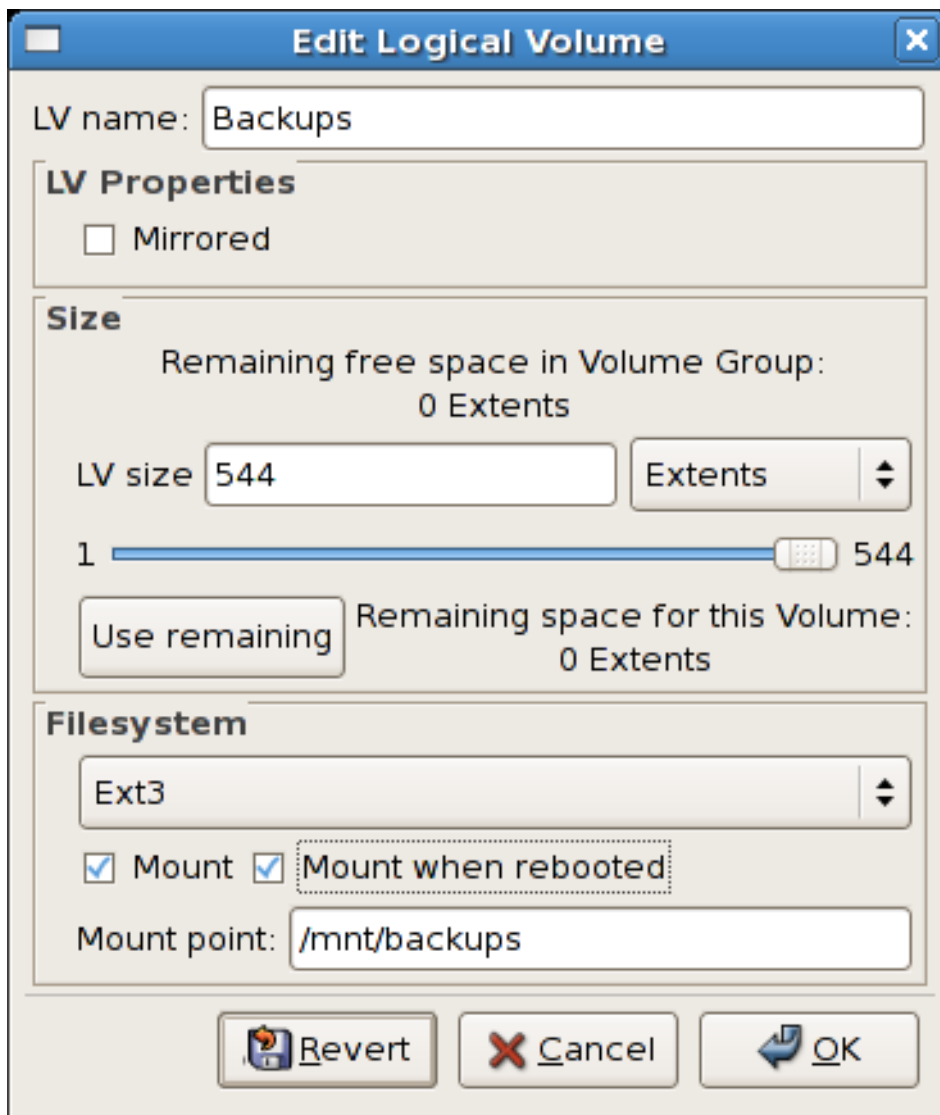


Figure 14.17. Modifier le volume logique

Si vous souhaitez monter le volume, sélectionnez la case à cocher « Monter » indiquant votre point de montage préféré. Pour monter le volume lorsque le système est redémarré, sélectionnez la case à cocher « Monter lors du redémarrage ». Dans cet exemple, le nouveau volume sera monté dans `/mnt/backups`. Ceci est illustré ci-dessous.



**Figure 14.18. Modifier le volume logique - spécifier des options de montage**

L'illustration ci-dessous montre les affichages physiques et logiques du groupe de volumes une fois que le volume logique a été étendu sur l'espace inutilisé. Dans cet exemple, le volume logique nommé « Backups » s'étend sur deux disques durs. Un volume peut être entrelacé à travers deux périphériques physiques ou plus en utilisant LVM.

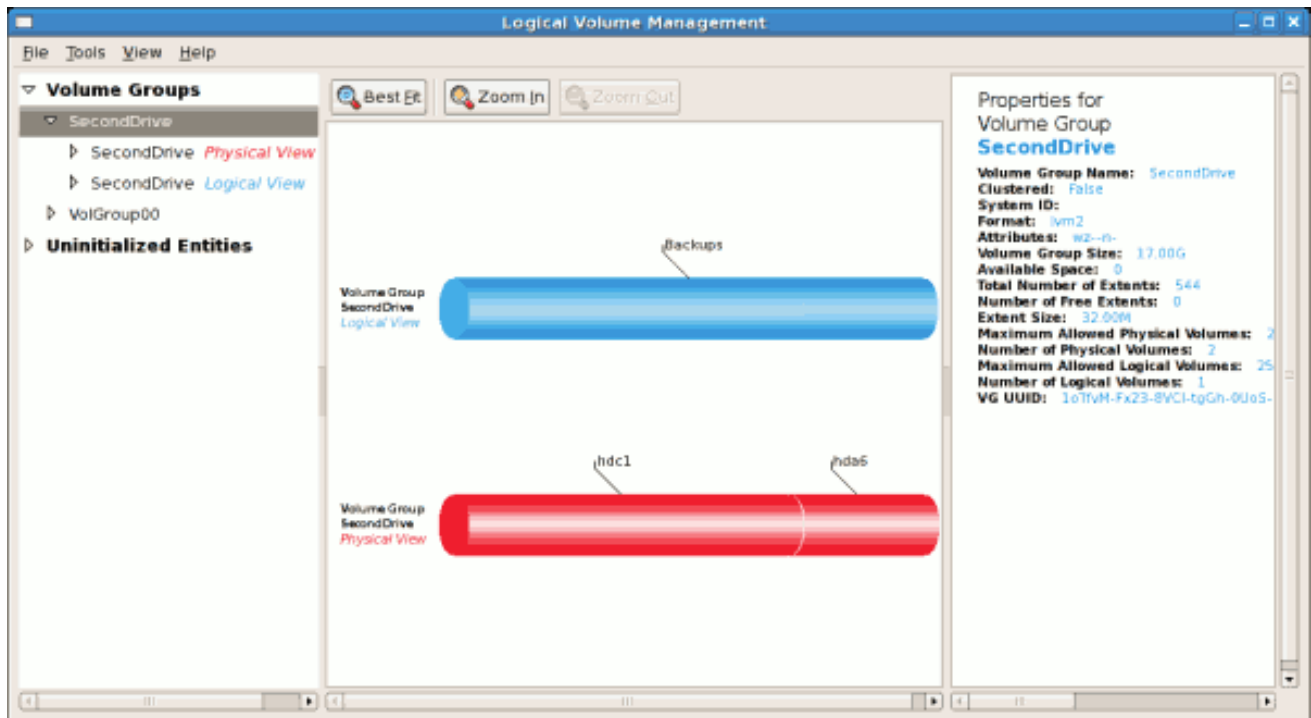


Figure 14.19. Modifier le volume logique

## 14.3. RÉFÉRENCES LVM

Utilisez ces sources pour en savoir plus sur LVM.

### Documentation installée

- `rpm -qd lvm2` — Cette commande affiche toute la documentation disponible à partir du paquet `lvm`, y compris les pages man.
- `lvm help` — Cette commande affiche toutes les commandes LVM disponibles.

### Sites Web utiles

- <http://sources.redhat.com/lvm2> — page web LVM2, qui contient une vue d'ensemble, des liens vers les listes de diffusion, et plus encore.
- <http://tldp.org/HOWTO/LVM-HOWTO/> — Document LVM HOWTO provenant du projet de documentation Linux (« Linux Documentation Project »).

## CHAPITRE 15. 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.

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 15.1, « 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.



### IMPORTANT

Les conseils qui se trouvent dans [Tableau 15.1, « 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 risquez de vous heurter à des problèmes d'instabilité ou même de compromettre le démarrage de votre système installé.

**Tableau 15.1. 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 15.1, « 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 with the Rescue Mode](#) du *Red Hat Enterprise Linux 6 Installation Guide*. Lorsqu'il vous sera demandé de monter le système de fichiers, sélectionnez **Ignorer**.

## 15.1. 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.

### 15.1.1. Étendre Swap dans un Volume logique LVM2

Par défaut, Red Hat Enterprise Linux 6 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. Pour obtenir des instructions sur la façon de procéder, consulter [Section 14.2.2, « Ajouter des volumes non alloués à un groupe de volumes »](#).

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 15.1. 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.

### 15.1.2. 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.

### 15.1.3. Création d'un fichier Swap

Comment ajouter un fichier swap :

#### Procédure 15.2. 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. 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. Il est conseillé de changer les permissions pour éviter que le fichier swap soit lisible par tout le monde.

```
# chmod 0600 /swapfile
```

5. Pour activer le fichier swap immédiatement, mais pas automatiquement au démarrage :

```
# swapon /swapfile
```

6. 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.

## 15.2. SUPPRIMER DE L'ESPACE SWAP

Parfois, il peut se révéler prudent de réduire l'espace swap après l'installation. Par exemple, supposons que vous avez 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.

### 15.2.1. 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 15.3. Réduire un volume logique LVM2 swap

1. Désactiver le swapping pour le volume logique associé :

```
# swapoff -v /dev/VolGroup00/LogVol01
```

2. Réduire la taille du volume logique LVM2 de 512 Mo :

```
# lvreduce /dev/VolGroup00/LogVol01 -L -512M
```

3. Formater le nouvel espace swap :

```
# mkswap /dev/VolGroup00/LogVol01
```

4. 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.

## 15.2.2. Supprimer un volume logique LVM2 avec Swap

Pour supprimer un groupe de volumes swap (en supposant que `/dev/VolGroup00/LogVol02` est le volume swap que vous souhaitez supprimer) :

### Procédure 15.4. 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.

## 15.2.3. Supprimer un fichier swap

Comment supprimer un fichier swap :

### Procédure 15.5. 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
```

## 15.3. 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 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.

### 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/LogVol100 /          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** a 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, veuillez utiliser les commandes suivantes :

```
umount /mount-point
```

Par exemple, **umount /work**.

```
mount /file-system /mount-point
```

Par exemple, **mount /dev/vdb1 /work**.

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.

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 dure (« hard »), la limite « soft » peut être dépassée pendant un moment. Ce moment est appelé 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 qui 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 :

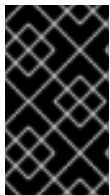
```
# quota -g devel
```

### 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 activer 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.

## 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.

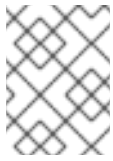
Les colonnes **grace** sont normalement vides. Si une limite **soft** a été dépassée, la colonne affichera le temps correspondant au délai restant de la période de grâce. Si la période de grâce a été dépassée, la chaîne **none** (aucun) s'affichera à sa place.

## 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** sûrement 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 -vaug /file_system
```

```
# quotacheck -vaug /file_system
```

```
# quotaon -vaug /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 -vaug 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 quotas 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* (auss 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 <sup>[4]</sup> 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. [5]

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 [6] 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'autre formats de métadonnées, appelées métadonnées externes. Red Hat Enterprise Linux 6 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 6 utilise **mdraid** pour accéder aux ensembles RAID microprogramme d'Intel.

## 17.4. PRISE EN CHARGE RAID DANS L'INSTALLATEUR

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 6.

## 17.5. CONFIGURATION D'ENSEMBLES RAID

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**.

## 17.6. 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 13, Partitions](#), **man parted**, et **man mdadm**.
4. Une fois les matrices créées, il est optionnellement possible de créer des systèmes de fichiers sur les matrices. Veuillez consulter [Section 11.2, « Aperçu des systèmes de fichiers pris en charge »](#) pour obtenir des informations techniques de base sur les systèmes de fichiers pris en charge par Red Hat Enterprise Linux 6.
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.

---

[4] Un chassis enfichable à chaud vous permet de supprimer un disque dur sans avoir à éteindre votre système.

[5] 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.

[6] 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, tels que 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), le contenu du fichier de configuration **/etc/fstab** est lu pour voir si le système de fichiers donné est répertorié. Ce fichier 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, 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, savoir 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
<b>ext2</b>	Système de fichiers <b>ext2</b> .
<b>ext3</b>	Système de fichier <b>ext3</b> .
<b>ext4</b>	Système de fichiers <b>ext4</b> .
<b>iso9660</b>	Système de fichiers <b>ISO 9660</b> . Communément utilisé par les supports optiques, comme les CD.
<b>jfs</b>	Système de fichier <b>JFS</b> , créé par IBM.
<b>nfs</b>	Système de fichiers <b>NFS</b> . Celui-ci est communément utilisé pour accéder à des fichiers sur un réseau.
<b>nfs4</b>	Système de fichiers <b>NFSv4</b> . Celui-ci est communément utilisé pour accéder à des fichiers sur un réseau.
<b>ntfs</b>	Système de fichiers <b>NTFS</b> . Celui-ci est communément utilisé sur des ordinateurs exécutant un système d'exploitation Windows.

Type	Description
<b>udf</b>	Système de fichiers <b>UDF</b> . Communément utilisé par les supports optiques, comme les DVD.
<b>vfat</b>	Système de fichiers <b>FAT</b> . 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.

Veuillez 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
<b>async</b>	Permet les opérations d'entrées/sorties asynchrones sur le système de fichiers.
<b>auto</b>	Permet au système de fichiers d'être monté automatiquement en utilisant la commande <b>mount -a</b> .
<b>defaults</b>	Fournit un alias pour <b>async, auto, dev, exec, nouser, rw, suid</b> .
<b>exec</b>	Permet l'exécution de fichiers binaires sur un système de fichiers particulier.

Option	Description
<b>loop</b>	Monte une image en tant que périphérique boucle.
<b>noauto</b>	Le comportement par défaut interdit le montage automatique du système de fichiers à l'aide de la commande <b>mount -a</b> .
<b>noexec</b>	Interdit l'exécution de fichiers binaires sur le système de fichiers en particulier.
<b>nouser</b>	Interdit à un utilisateur normal (c'est-à-dire autre que <b>root</b> ) de monter et démonter le système de fichiers.
<b>remount</b>	Remonte le système de fichiers au cas où il serait déjà monté.
<b>ro</b>	Monte le système de fichiers en lecture seule.
<b>rw</b>	Monte le système de fichier en lecture et écriture.
<b>user</b>	Permet à un utilisateur normal (c'est-à-dire autre que <b>root</b> ) 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 6 implémente la fonctionnalité connue sous le nom de *sous-arborescence partagée*. Cette fonctionnalité permet l'utilisation des quatre types de montage suivants :

### Montage partagé

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 reflète 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

```

## Montage esclave

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
```

Veuillez 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 dupliqué dans `/mnt`, mais marquez-le en tant qu'« esclave » :

```

~]# mount --bind /media /mnt
~]# mount --make-slave /mnt

```

Veuillez 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

```

Veuillez 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
```

## Montage privé

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
```

Veuillez 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'apparaisse 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
```

### Montage ne pouvant pas être lié

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 dupliqué 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
```

## 18.3. DÉMONTÉ 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 un 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 sont utilisation.
- **man 8 umount** — Page du manuel pour la commande **umount**, celle-ci fournit une documentation complète sur sont utilisation.
- **man 8 findmnt** — Page du manuel pour la commande **findmnt**, celle-ci fournit une documentation complète sur sont 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 6. Pour obtenir des informations sur son installation, veuillez consulter [http://fedoraproject.org/wiki/Disk\\_encryption\\_key\\_escrow\\_use\\_cases](http://fedoraproject.org/wiki/Disk_encryption_key_escrow_use_cases).

## 19.1. COMMANDES

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` 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. Déamarrer 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 systèmes et de conserver un mot de passe distinct pour chaque système 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 :

- Stocker le certificat et la clé privée dans un fichier **PKCS#12**.
- 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.

- o Exécutez:

```
pk12util -d /the/nss/directory -i the-pkcs12-file
```

4. Distribuer le certificat à quiconque installant des systèmes ou sauvegardant des clé 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 le 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](http://fedoraproject.org/wiki/Disk_encryption_key_escrow_use_cases)

## CHAPITRE 20. LISTES DES CONTRÔLE D'ACCÈS (ACL)

Les fichiers et répertoires possèdent des ensembles de permissions pour le propriétaire du fichier, le groupe associé au fichier, ainsi que pour tous les autres utilisateurs du système. Cependant, ces ensembles de permissions sont limités. Par exemple, différentes permissions ne peuvent pas être configurées pour différents utilisateurs. Donc, des *listes de contrôle d'accès* (ACL, de l'anglais (« Access Control Lists »)) ont été implémentées.

Le noyau Red Hat Enterprise Linux offre la prise en charge des ACL sur les systèmes de fichiers ext3 et les systèmes de fichiers exportés NFS. On trouve aussi des ACL sur les systèmes de fichiers ext3 auxquels on peut accéder via Samba.

Le paquet **ac1** et la prise en charge dans le noyau sont requis pour implémenter les ACL. Ce paquet contient les utilitaires nécessaires pour l'ajout, la modification, la suppression et la récupération d'informations sur les ACL.

Les commandes **cp** et **mv** copient ou déplacent toutes les ACL associées à des fichiers et répertoires.

### 20.1. MONTER DES SYSTÈMES DE FICHIERS

Avant d'utiliser des ACL pour un fichier ou un répertoire, la partition du fichier ou répertoire doit être montée avec la prise en charge des ACL. S'il s'agit d'un système de fichiers ext3 local, celui-ci peut être monté avec la commande suivante :

```
mount -t ext3 -o ac1 device-name partition
```

Par exemple :

```
mount -t ext3 -o ac1 /dev/VolGroup00/LogVol02 /work
```

De manière alternative, si la partition est répertoriée dans le fichier **/etc/fstab**, l'entrée de la partition peut inclure l'option **ac1** :

```
LABEL=/work      /work      ext3      ac1      1 2
```

Si un système de fichiers ext3 est accédé via Samba et que les ACL ont été activées pour cela, ces ACL seront reconnues car Samba a été compilé avec l'option **--with-ac1-support**. Aucun indicateur particulier n'est requis lors de l'accession ou du montage d'un partage Samba.

#### 20.1.1. NFS

Par défaut, si le système de fichiers exporté par un serveur NFS prend en charge les ACL et le client NFS peut lire les ACL, alors les ACL sont utilisés par le système client. Pour désactiver les ACL sur le partage NFS lors de son montage sur un client, montez-le avec l'option **noac1** sur la ligne de commande.

### 20.2. DÉFINIR LES ACL D'ACCÈS

Il existe deux types d'ACL : les *ACL d'accès* et les *ACL par défaut*. Une ACL d'accès est une liste de contrôle d'accès pour un fichier ou répertoire particulier. Une ACL par défaut peut uniquement être associée à un répertoire. Si un fichier dans le répertoire ne possède pas d'ACL d'accès, alors il utilise les règles de l'ACL par défaut du répertoire. Les ACL par défaut sont optionnelles.

Les ACL peuvent être configurées :

1. Par utilisateur
2. Par groupe
3. Via le masque des droits en vigueur
4. Pour les utilisateur ne se trouvant pas dans le groupe d'utilisateurs du fichier

L'utilitaire **setfacl** définit les ACL pour les fichiers et répertoires. Veuillez utiliser l'option **-m** pour ajouter ou modifier l'ACL d'un fichier ou répertoire :

```
# setfacl -m rules files
```

Des règles (*rules*) doivent être spécifiées sous les formats suivants. De multiples règles peuvent être spécifiées dans la même commande si celles-ci sont séparées par des virgules.

#### **u:uid:perms**

Définit l'ACL d'accès d'un utilisateur. Le nom d'utilisateur, ou UID, peut être spécifié. L'utilisateur peut être tout utilisateur valide sur le système.

#### **g:gid:perms**

Définit l'ACL d'accès d'un groupe. Le nom du groupe, ou GID, peut être spécifié. Le groupe peut être tout groupe valide sur le système.

#### **m:perms**

Définit le masque des permissions. Le masque est l'union de toutes les permissions du groupe propriétaire et des toutes les entrées d'utilisateur et de groupe.

#### **o:perms**

Définit l'ACL d'accès du fichier pour les utilisateurs ne faisant pas partie du groupe.

Les permissions (*perms*) doivent être une combinaison des caractères **r**, **w** et **x** pour lecture, écriture, et exécution.

Si un fichier ou répertoire possède déjà une ACL et que la commande **setfacl** est utilisée, les règles supplémentaires sont ajoutées à l'ACL existante ou la règle existante sera modifiée.

#### **Exemple 20.1. Donner des permissions de lecture et écriture**

Par exemple, pour donner les permissions de lecture et écriture à l'utilisateur andrius :

```
# setfacl -m u:andrius:rw /project/somefile
```

Pour supprimer toutes les permissions d'un utilisateur, d'un groupe, ou autres, veuillez utiliser l'option **-x** et ne spécifier aucune permission :

```
# setfacl -x rules files
```

#### **Exemple 20.2. Supprimer toutes les permissions**

Par exemple, pour supprimer toutes les permissions de l'utilisateur possédant l'UID 500 :

```
# setfacl -x u:500 /project/somefile
```

## 20.3. DÉFINIT LES ACL PAR DÉFAUT

Pour définir une ACL par défaut, veuillez ajouter **d** : avant la règle et spécifiez un répertoire à la place d'un nom de fichier.

### Exemple 20.3. Définir les ACL par défaut

Par exemple, pour définir l'ACL par défaut du répertoire **/share/** afin de pouvoir effectuer des lectures et exécutions pour les utilisateurs ne se trouvant pas dans le groupe d'utilisateurs (une ACL d'accès pour un fichier individuel peut la remplacer) :

```
# setfacl -m d:o:rx /share
```

## 20.4. RÉCUPÉRER DES ACL

Pour déterminer les ACL existantes pour un fichier ou répertoire, veuillez utiliser la commande **getfacl**. Dans l'exemple ci-dessous, **getfacl** est utilisé pour déterminer les ACL existantes pour un fichier.

### Exemple 20.4. Récupérer des ACL

```
# getfacl home/john/picture.png
```

La commande ci-dessus retourne la sortie suivante :

```
# file: home/john/picture.png
# owner: john
# group: john
user::rw-
group::r--
other::r--
```

Si un répertoire avec une ACL par défaut est spécifié, l'ACL par défaut est aussi affichée comme illustré ci-dessous. Par exemple, **getfacl home/sales/** affichera une sortie similaire à la suivante :

```
# file: home/sales/
# owner: john
# group: john
user::rw-
user:barryg:r--
group::r--
mask::r--
other::r--
default:user::rwx
default:user:john:rwx
```

```
default:group::r-x
default:mask::rwx
default:other::r-x
```

## 20.5. ARCHIVER DES SYSTÈMES DE FICHIERS AVEC DES ACL

Par défaut, la commande **dump** préserve désormais les ACL pendant les opérations de sauvegarde. Lors de l'archivage d'un fichier ou d'un système de fichiers avec **tar**, utilisez l'option **--acls** pour préserver les ACL. De manière similaire, lors de l'utilisation de **cp** pour copier des fichiers avec des ACL, veuillez inclure l'option **--preserve=mode** afin de vous assurer que les ACL soient également copiées. En outre, l'option **-a** (équivalente à **-dR --preserve=all**) de **cp** préserve également les ACL lors des opérations de sauvegarde ainsi que d'autres informations, comme les horodatages, les contextes SELinux, etc. Pour obtenir davantage d'informations sur **dump**, **tar**, ou **cp**, veuillez consulter les pages **man** respectives.

L'utilitaire **star** est similaire à l'utilitaire **tar** car il peut être utilisé pour générer des archives de fichiers ; cependant, certaines de ses options sont différentes. Veuillez consulter le [Tableau 20.1, « Options de ligne de commande pour star »](#) pour obtenir une liste des options communément utilisées. Pour toutes les options disponibles, veuillez consulter **man star**. Le paquet **star** est requis pour faire usage de cet utilitaire.

**Tableau 20.1. Options de ligne de commande pour star**

Option	Description
<b>-c</b>	Crée un fichier archive.
<b>-n</b>	Ne pas extraire les fichiers. À utiliser en conjonction avec <b>-x</b> pour afficher le résultat provoqué par l'extraction de fichiers.
<b>-r</b>	Remplace les fichiers dans l'archive. Les fichiers sont écrits à la fin du fichier archive, remplaçant tout fichier ayant le même chemin et le même nom de fichier.
<b>-t</b>	Affiche le contenu du fichier archive.
<b>-u</b>	Met à jour le fichier archive. Les fichiers sont écrits à la fin de l'archive si les conditions suivantes s'appliquent : <ul style="list-style-type: none"> <li>• Ils n'existent pas déjà dans l'archive.</li> <li>• Les fichiers devant être mis à jour sont plus récents que les fichiers du même nom déjà présents dans l'archive.</li> </ul> <p>Cette option ne fonctionne que si l'archive correspond à un fichier ou à une bande non bloquée qui puisse aller dans le sens inverse (et non pas revenir à 0)</p>
<b>-x</b>	Extrait les fichiers de l'archive. Si utilisé avec <b>-U</b> et qu'un fichier dans l'archive est plus ancien que le fichier correspondant sur le système de fichiers, le fichier ne sera pas extrait.



Option	Description
<b>-help</b>	Affiche les options les plus importantes.
<b>-xhelp</b>	Affiche les options les moins importantes.
<b>-/</b>	Ne pas supprimer les barres obliques des noms de fichiers lors de l'extraction de fichiers d'une archive. Par défaut, celles-ci sont supprimées lorsque les fichiers sont extraits.
<b>-acl</b>	Pendant la création ou l'extraction, archive ou restaure toute ACL associée aux fichiers et répertoires.

## 20.6. COMPATIBILITÉ AVEC D'ANCIENS SYSTÈMES

Si une ACL a été définie sur un fichier quelconque dans un système de fichiers donné, ce système de fichiers possèdera l'attribut **ext\_attr**. Cet attribut peut être affiché à l'aide de la commande suivante :

```
# tune2fs -l filesystem-device
```

Un système de fichier ayant acquis l'attribut **ext\_attr** peut être monté avec d'anciens noyaux, mais ces noyaux n'appliqueront aucune ACL définie.

Les versions de l'utilitaire **e2fsck** incluses dans la version 1.22 et dans les versions supérieures du paquet **e2fsprogs** (y compris les versions dans Red Hat Enterprise Linux 2.1 et 4) peuvent vérifier un système de fichiers avec l'attribut **ext\_attr**. Les versions plus anciennes refuseront de le vérifier.

## 20.7. RÉFÉRENCES DES ACL

Veuillez consulter les pages man pour obtenir davantage d'informations.

- **man acl** — Description des ACL
- **man getfacl** — Traite de la manière d'obtenir des listes de contrôle d'accès
- **man setfacl** — Explique comment définir des listes de contrôle d'accès aux fichiers
- **man star** — Explique l'utilitaire **star** et ses nombreuses options

## CHAPITRE 21. DIRECTIVES DE DÉPLOIEMENT DES DISQUES SSD

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).

L'activation de la prise en charge de **discard** est surtout utile lorsqu'il y a de l'espace libre sur le système de fichiers, mais le système de fichiers a déjà écrit sur la plupart des blocs logiques du périphérique de stockage sous-jacent. Pour obtenir des informations supplémentaires sur **TRIM**, veuillez consulter le document *Data Set Management T13 Specifications*, disponible sur le lien suivant :

[http://t13.org/Documents/UploadedDocuments/docs2008/e07154r6-Data\\_Set\\_Management\\_Proposal\\_for\\_ATA-ACS2.doc](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`.

### 21.1. 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 6 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.

À partir de Red Hat Enterprise Linux 6.5 MD MD prend en charge les abandons. Avant 6.5, cela n'était pas pris en charge. En revanche, les cibles du gestionnaire de volumes logique LVM et du mappeteur de périphériques DM utilisées par LVM prennent en charge les abandons. 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.

Red Hat recommande l'utilisation des niveaux RAID 1 ou RAID10 pour les RAID LVM sur les disques SSD car ces niveaux prennent en charge les abandons. 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.



## NOTE

Il est possible d'utiliser l'option **--nosync** sur RAID1, RAID10, et les RAID de parité, car la parité sera calculée pour cette bande dans la minute de la première écriture, conservant ainsi toute homogénéité. Cependant, lors des opérations de nettoyage des données, les parties non écrites seront considérées comme étant **mismatched/inconsistent** (non correspondantes/inconsistantes).

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.

## 21.2. 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
```

### 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 22. 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.

### 22.1. 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 « complète » (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()`.

## 22.2. 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 6 active par défaut les barrières d'écriture sur tous les systèmes de fichiers journalisant pris en charge.

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 22.1](#), « Messages d'erreur des barrières d'écriture par système de fichiers »).

**Tableau 22.1. Messages d'erreur des barrières d'écriture par système de fichiers**

Système de fichiers	Message d'erreur
ext3/ext4	<b>JBD: barrier-based sync failed on device - disabling barriers</b>
XFS	<b>Filesystem device - Disabling barriers, trial barrier write failed</b>
btrfs	<b>btrfs: disabling barriers on dev device</b>



### NOTE

L'utilisation de `noBARRIER` n'est plus recommandée sur Red Hat Enterprise Linux 6 car l'impact de performance négatif des barrières d'écriture est négligeable (environ 3%). Habituellement, les bénéfices des barrières d'écriture sont plus importants que les bénéfices de performance dus à leur désactivation. En outre, l'option `noBARRIER` ne devrait jamais être utilisée sur un stockage configuré sur des machines virtuelles.

## 22.3. 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 importante 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 23. 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 6 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.

Veuillez consulter [Section 23.2, « 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.

### 23.1. 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 6 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.

## 23.2. 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`
- `/sys/block/disk/partition/alignment_offset`
- `/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 23.1. 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`

## 23.3. 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 6 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
```

## 23.4. EMPIILER 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 :

- Seulement une couche dans 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 6, 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.

## 23.5. 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.

## 23.6. 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 6 (`Anaconda`) utilise `libparted`, ce qui signifie que toutes les partitions créées par l'installateur ou `parted` seront correctement alignées. 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 6 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 amélioré 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 24. PARAMÉTRER UN SYSTÈME SANS DISQUE DISTANT

Le service de démarrage réseau (« Network Booting Service », fourni par **system-config-netboot**) n'est plus disponible dans Red Hat Enterprise Linux 6. Le déploiement de systèmes sans disque est désormais possible dans cette version sans avoir à utiliser **system-config-netboot**.

Pour paramétrer un système sans disque distant démarré via PXE, vous aurez besoin des paquets suivants :

- **tftp-server**
- **xinetd**
- **dhcp**
- **syslinux**
- **dracut-network**

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.

Les sections suivantes soulignent les procédures nécessaires pour déployer des systèmes sans disque distants dans un environnement en réseau.

### 24.1. 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 24.1. 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 emballages TCP, vous pouvez configurer l'accès des hôtes à **tftp** via **/etc/hosts.allow**. Pour obtenir des informations supplémentaires sur la configuration des emballages TCP et sur le fichier de configuration **/etc/hosts.allow**, veuillez consulter le *Guide de sécurité* de Red Hat Enterprise Linux 6. **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 24.2, « Configuration DHCP pour les clients sans disque »](#) et [Section 24.3, « Configuration d'un système de fichiers exporté pour les clients sans disque »](#) pour obtenir des instructions sur la manière de procéder.

## 24.2. 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 6 pour obtenir des instructions sur la manière de paramétrer un serveur DHCP. En outre, vous devrez activer le démarrage P 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-identifrier, 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 24.3, « Configuration d'un système de fichiers exporté pour les clients sans disque »](#) afin d'obtenir des instructions.

## 24.3. 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 9.7.1, « 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
```

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 24.2. 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/vmlinuz-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
```

Copiez également le fichier `initramfs-kernel-version.img` résultant dans le répertoire de démarrage `tftp`.

4. 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 rhel6

label rhel6
    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 25. DEVICE MAPPER MULTIPATHING (MAPPEUR DE PÉRIPHÉRIQUES À MULTIPLES CHEMINS D'ACCÈS) ET LE STOCKAGE VIRTUEL

Red Hat Enterprise Linux 6 prend aussi 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 d'administration de la virtualisation*.

### 25.1. STOCKAGE VIRTUEL

Pour le stockage virtuel, Red Hat Enterprise Linux 6 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 6 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. Deux catégories de pools de stockage sont disponibles :

#### Pools de stockage locaux

Le stockage local couvre les périphériques de stockage, 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 la section *Stockage virtuel* du *Guide 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.

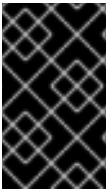
DM-Multipath est principalement utilisé pour les raisons suivantes :

### **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 configurée 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.

## PARTIE III. STOCKAGE EN LIGNE

La section Stockage en ligne explique comment installer et gérer des connexions de stockage iSCSI et FCoE.

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.

## CHAPITRE 26. FIBRE CHANNEL

Cette section discute de l'API Fibre Channel, des pilotes Fibre Channel Red Hat Enterprise Linux 6 natifs, et des capacités Fibre Channel de ces pilotes.

### 26.1. 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.

## 26.2. PILOTES ET CAPACITÉS NATIFS FIBRE CHANNEL

Red Hat Enterprise Linux 6 est fourni avec les pilotes natifs fibre channel suivants :

- `lpfc`
- `qla2xxx`
- `zfc`
- `mptfc`
- `bfa`

Tableau 26.1, « Capacités de l'API Fibre-Channel » décrit les différentes capacités de l'API fibre-channel de chaque pilote natif Red Hat Enterprise Linux 6. X dénote la prise en charge de la capacité.

Tableau 26.1. Capacités de l'API Fibre-Channel

	<code>lpfc</code>	<code>qla2xxx</code>	<code>zfc</code>	<code>mptfc</code>	<code>bfa</code>
Transport <code>port_id</code>	X	X	X	X	X
Transport <code>node_name</code>	X	X	X	X	X
Transport <code>port_name</code>	X	X	X	X	X
Port distant <code>dev_loss_tmo</code>	X	X	X	X	X
Port distant <code>fast_io_fail_tmo</code>	X	X [a]	X [b]		X
Hôte <code>port_id</code>	X	X	X	X	X
Hôte <code>issue_lip</code>	X	X			X
[a] Pris en charge à partir de Red Hat Enterprise Linux 5.4					
[b] Pris en charge à partir de Red Hat Enterprise Linux 6.0					

## CHAPITRE 27. INSTALLER UNE CIBLE ISCSI ET UN INITIATEUR



### NOTE

Lors de l'utilisation du démon `hal` avec un grand nombre de LUN iSCSI, en plusieurs milliers, l'option `--child-timeout` doit être utilisée afin d'éviter des échecs de démarrage. L'option `--child-timeout` définit le nombre de secondes d'attente pour que toutes les sondes de disques soient exécutées. Par exemple, pour forcer le démon `hal` à attendre 10 minutes et 30 secondes, l'option doit afficher `--child-timeout=630`. La valeur par défaut est de 250 secondes. Bien que cela signifie que le démon `hal` prendra plus de temps à démarrer, il devrait y avoir suffisamment de temps pour que tous les périphériques de disques soient reconnus et évitent les échecs de démarrage.

La raison pour cette solution de contournement est qu'en 2003, lorsque le démon `hal` a été créé, il était inhabituel de posséder plus d'une douzaine de disques iSCSI. C'est pour cela que le démon `hal` a été supprimé dans Red Hat Enterprise Linux 7 et qu'il a été remplacé par les `udisks`.

Pour plus d'informations, voir l'article de Base de connaissance Red Hat suivant : [haldaemon fails to start on system with a large number of disks in RHEL 5 and RHEL 6](#)

### 27.1. CRÉATION DE CIBLE ISCSI

Une cible iSCSI peut correspondre à un périphérique physique destiné à ce but uniquement, ou peut correspondre à un périphérique logique configuré pour logiciel iSCSI sur un serveur de stockage mis en réseau. La cible est la destination de la communication bus SCSI. Le stockage sur la cible, auquel un initiateur peut avoir accès, est défini par les LUN.

#### Procédure 27.1. Créer une Cible iSCSI

1. Installer `scsi-target-utils`.

```
~]# yum install scsi-target-utils
```

2. Ouvrir le port 3260 dans le pare feu.

```
~]# iptables -I INPUT -p tcp -m tcp --dport 3260 -j ACCEPT
~]# service iptables save
```

3. Active et lance le service cible.

```
~]# service tgtd start
~]# chkconfig tgtd on
```

4. Alloue le stockage des LUN. Dans cet exemple, on crée une nouvelle partition de stockage en bloc.

```
~]# fdisk /dev/vdb
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.
```

```
Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x43eb8efd.
```

```
Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): *Enter*
Using default response p
Partition number (1-4, default 1): *Enter*
First sector (2048-2097151, default 2048): *Enter*
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-2097151, default
2097151): +250M
Partition 1 of type Linux and of size 250 MiB is set

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

5. Modifiez le fichier `/etc/tgt/targets.conf` pour créer la cible.

```
~]# cat /etc/tgt/targets.conf
...
default-driver iscsi
<target iqn.2015-06.com.example.test:target1>
  backing-store /dev/vdb1
  initiator-address 10.10.1.1
</target>
```

Dans l'exemple ci-dessus, on a créé une simple cible qui contient un store de sauvegarde et un initiateur autorisé. Il doit comprendre un nom `iqn` sous le format **`iqn.YYYY-MM.reverse.domain.name:OptionalIdentif`**. Le store de sauvegarde correspond au périphérique sur lequel le stockage se trouve. L'`initiator-address` correspond à l'adresse IP de l'initiateur qui doit accéder au stockage.

6. Redémarrer le service cible.

```
~]# service tgtd restart
Stopping SCSI target daemon:      [ OK ]
Starting SCSI target daemon:      [ OK ]
```

7. Vérifier la configuration.

```
~]# tgt-admin --show
Target 1: iqn.2015.06.com.example.test: server
System information:
  Driver: iscsi
  State: ready
I_T nexus information:
LUN information:
```

```

Lun: 0
Type: controller
SCSI ID: IET      00010000
SCSI SN: beaf10
Size: 0 MB, Block size: 1
Online: Yes
Removable media: No
Prevent removal: No
ReadOnly: No
Backing store type: null
Backing store path: None
Backing store flags:
LUN: 1
Type: disk
SCSI ID: IET      00010001
SCSI SN: beaf11
Size: 2147 MB, Block size: 512
Online: Yes
Removable media: No
Prevent removal: No
ReadOnly: No
Backing store type: rdwr
Backing store path: /dev/vdb1
Backing store flags:
Account information:
ACL information:
10.10.1.1

```

## 27.2. CRÉATION DE L'INITIATEUR ISCSI

Un initiateur iSCSI est un client qui souhaite accéder au stockage sur une cible ou un serveur. L'adresse IP de la cible doit être connue pour ce processus.

### Procédure 27.2. Créer un initiateur iSCSI

1. Installer **iscsi-initiator-utils**.

```
~]# yum install iscsi-initiator-utils
```

2. Découvrez la cible. Utilisez l'adresse IP de la cible, celle qui est utilisée ci-dessous n'est utilisée qu'à titre d'exemple.

```
~]# iscsiadm -m discovery -t sendtargets -p 192.168.1.1
Starting iscsid:      [ OK ]
192.168.1.1:3260,1 iqn.2015-06.com.example.test:target1
```

L'exemple ci-dessus montre l'adresse IP de la cible et l'adresse IQN. C'est l'adresse IQN dont on a besoin pour les prochaines étapes.

3. Connectez-vous à la cible.

```
~]# iscsiadm -m node -T iqn.2015-06.com.example:target1 --login
Logging in to [iface: default, target: iqn.2015-
06.com.example:target1, portal: 192.168.1.1,3260] (multiple)
```



```
Login in to [iface: default, target: iqn.2015-06.com.example:target1, portal: 192.168.1.1,3260] successful.
```

4. Cherchez le nom du disque iSCSI.

```
~]# grep "Attached SCSI" /var/log/messages  
Jun 19 01:30:26 test kernel: sd 7:0:0:1 [sdb] Attached SCSI disk
```

5. Créez un système de fichiers sur ce disque.

```
~]# mkfs.ext4 /dev/sdb
```

6. Montez le système de fichiers.

```
~]# mkdir /mnt/iscsiTest  
~]# mount /dev/sdb /mnt/iscsiTest
```

7. Rendez-le persistant à travers les démarrages en modifiant le fichier **/etc/fstab**.

```
~]# blkid /dev/sdb  
/dev/sdb: UUID="766a3bf4-beeb-4157-8a9a-9007be1b9e78" TYPE="ext4"  
~]# vim /etc/fstab  
UUID=766a3bf4-beeb-4157-8a9a-9007be1b9e78 /mnt/iscsiTest ext4  
_netdev 0 0
```

## CHAPITRE 28. DÉNOMINATION PERSISTANTE

Le système d'exploitation émet des E/S sur un périphérique de stockage en référençant le chemin qui est utilisé pour l'atteindre. Pour les périphériques SCSI, le chemin consiste en ce qui suit :

- Identifiant PCI de l'adaptateur de bus hôte (HBA)
- numéro de canal sur ce HBA
- adresse de la cible SCSI à distance
- LUN (numéro d'unité logique, « Logical Unit Number »)

Cette adresse basée sur chemin n'est pas persistante. Elle peut changer à chaque fois que le système est reconfiguré (reconfiguration en ligne comme décrit dans ce manuel, ou lorsque le système est éteint, reconfiguré, puis redémarré). Il est même possible pour les identifiants de chemin changent alors qu'aucune reconfiguration physique n'a été effectuée. Ces changements résultent des variations de cadencement pendant le processus de découverte, lorsque le système démarre, ou lorsqu'un bus est scanné à nouveau.

Le système d'exploitation offre plusieurs noms non-persistants pour représenter ces chemins d'accès vers les périphériques de stockage. L'un est le nom `/dev/sd` ; l'autre est le numéro `major:minor`. Un troisième est un lien symbolique maintenu dans le répertoire `/dev/disk/by-path/`. Ce lien symbolique mappe l'identifiant de chemin vers le nom `/dev/sd` actuel. Par exemple, pour un périphérique Fibre Channel, les informations PCI `Host:BusTarget:LUN` peuvent apparaître comme suit :

```
pci-0000:02:0e.0-scsi-0:0:0:0 -> ../../sda
```

Pour les périphériques iSCSI, les noms `by-path/` effectuent le mappage depuis le nom de la cible et les informations du portail au nom `sd`.

Il n'est généralement *pas* approprié pour des applications d'utiliser ces chemins basés sur nom. Cela est dû au fait que le périphérique de stockage référencé par ces chemins peuvent changer, entraînant ainsi l'écriture sur disque de données potentiellement incorrectes. Les noms basés sur chemin ne sont également pas appropriés pour les périphériques multivoies, car les noms basés sur chemins peuvent être pris pour des périphérique de stockage séparés, menant à un accès non coordonné et à des modifications involontaires des données.

En outre, les noms basés sur chemin sont spécifiques au système. Ceci peut provoquer des changements de données involontaires lorsque de multiples systèmes accèdent au périphérique, comme dans le cas d'un cluster.

Pour ces raisons, plusieurs méthodes d'identification de périphérique ont été développées. Ces méthodes sont persistantes, et ne dépendent pas du système. Les sections suivantes traitent de celles-ci avec plus de détails :

### 28.1. 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 28.1. 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][hwhandler=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. Pour obtenir davantage d'informations à ce sujet, veuillez consulter <http://kbase.redhat.com/faq/docs/DOC-7319>.

## 28.2. UUID ET AUTRES IDENTIFICATEURS PERSISTANTS

Si un périphérique de stockage contient un système de fichiers, alors ce système de fichier peut fournir l'un des (ou les) deux identificateurs suivants :

- UUID (*Universally Unique Identifier*)
- Étiquette du système de fichiers

Ces identificateurs sont persistants, et basés sur les métadonnées écrites sur le périphérique par certaines applications. Ils peuvent également être utilisés pour accéder au périphérique à l'aide des liens syboliques maintenus par le système d'exploitation dans les répertoires `/dev/disk/by-label/` (par exemple, **boot** -> `../../sda1`) et `/dev/disk/by-uuid/` (par exemple, **f8bf09e3-4c16-4d91-bd5e-6f62da165c08** -> `../../sda1`).

**md** et **LVM** écrivent des métadonnées sur le périphérique de stockage, et lisent ces données lorsqu'ils scannent les périphériques. Dans chaque cas, les métadonnées contiennent un UUID afin que le périphérique puisse être identifié, peu importe le chemin (ou système) utilisé pour y accéder. Par conséquent, les noms des périphériques présentés par cet équipement sont persistants tant que les métadonnées ne sont pas modifiées.

## CHAPITRE 29. 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 28.1, « 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 [Chapitre 31, 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`).

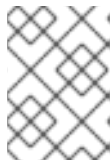
La procédure générale pour supprimer tout accès à un périphérique est comme suit :

### Procédure 29.1. 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 29.1, « 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 [Chapitre 34, 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, 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, li devra être fait lorsque les E/S sont sur pause, comme décrit dans [Chapitre 34, Scanner les interconnexions du stockage](#).

## CHAPITRE 30. 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 30.1. 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 29.1, « S'assurer d'une suppression de périphérique normale »](#)).

Après avoir effectué une opération [Procédure 30.1, « 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 [Chapitre 34, Scanner les interconnexions du stockage](#).

## CHAPITRE 31. AJOUTER UN PÉRIPHÉRIQUE OU UN CHEMIN DE STOCKAGE

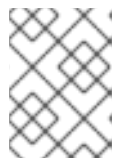
Lors de l'ajout d'un périphérique, n'oubliez pas que le nom du périphérique basé chemin (par exemple, le nom de `/dev/sd`, le numéro de `major:minor` et le nom de `/dev/disk/by-path`) assigné par le système au nouveau périphérique peut déjà avoir été utilisé par un périphérique qui a depuis été supprimé. 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 31.1. 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 au système d'exploitation tant qu'une opération LIP (*Loop Initialization Protocol*) n'est pas effectuée. Reportez-vous à [Chapitre 34, Scanner les interconnexions du stockage](#) pour obtenir des instructions sur la manière d'effectuer cela.



#### IMPORTANT

Il sera nécessaire d'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 31.1. 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.

## CHAPITRE 32. 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 32.1. Configurer une interface Ethernet pour utiliser FCoE

1. Configurez un nouveau réseau VLAN en copiant un script réseau existante (par exemple, `/etc/fcoe/cfg-eth0`) sur 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 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**) à l'aide de la commande suivante :

```
# /etc/init.d/lldpad start
```

4. Pour les interfaces réseau qui implémentent un client DCBX matériel, ignorez cette étape et passez directement à la suivante.

Pour les interfaces qui requièrent un client DCBX logiciel, veuillez activer le pontage de centre de données sur l'interface Ethernet à l'aide des commandes suivante :

```
# dcbtool sc ethX dcb on
```

Puis, activez FCoE sur l'interface Ethernet en exécutant :

```
# dcbtool sc ethX app:fcoe e:1
```



#### NOTE

Ces commandes fonctionneront uniquement 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 :

```
# service fcoe start
```

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 :

```
# chkconfig lldpad on
```

```
# chkconfig fcoe on
```



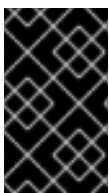
### AVERTISSEMENT

N'exécutez pas DCB ou LLDP basé logiciel sur des CNA qui implémentent DCB.

Certains adaptateurs réseau combinés (de l'anglais, « Combined Network Adapters », ou CNA) implémentent le protocole DCB (« Data Center Bridging ») dans le micrologiciel. Le protocole DCB suppose qu'il n'existe qu'un seul point créateur DCB sur un lien réseau particulier. Cela signifie que toute implémentation logicielle de plus haut niveau de DCB ou de LLDP (« Link Layer Discovery Protocol ») doit être désactivée sur les CNA qui implémentent DCB.

## 32.1. PARAMÉTRAGE DE CIBLE FCOE (« FIBRE-CHANNEL OVER ETHERNET »)

En plus de monter des LUN sur FCoE comme décrit dans [Chapitre 32, Configurer une interface FCoE \(« Fibre-Channel Over Ethernet »\)](#), l'export de LUN vers d'autres ordinateurs sur FCoE est également pris en charge.



### IMPORTANT

Avant de continuer, veuillez consulter [Chapitre 32, 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 32.2. Configurer un cible FCoE

1. Paramétrer une cible FCoE requiert l'installation du paquet **fcoe-target-utils** ainsi que de ses dépendances.

```
# yum install fcoe-target-utils
```

2. La prise en charge de cibles FCoE est basée sur la cible du noyau LIO et ne requiert pas de démon de l'espace utilisateur. Cependant, il est toujours utile d'activer le service **fcoe-target** pour charger les modules de noyau nécessaires et conserver la configuration aux redémarrages.

```
# service fcoe-target start
```

```
# chkconfig fcoe-target on
```

3. La configuration d'une cible FCoE est effectuée à l'aide de l'utilitaire **targetcli**, plutôt qu'en modifiant un fichier **.conf** comme cela pourrait être attendu. Ces paramètres sont ensuite enregistrés afin de pouvoir les restaurer si le système redémarre.

```
# targetcli
```

**targetcli** est un shell de configuration hiérarchique. Effectuez des déplacements entre nœuds dans le shell en utilisant **cd**, et utilisez **ls** pour afficher le contenu du nœud de configuration. Pour connaître des options supplémentaires, utiliser la commande **help**.

4. Définissez le fichier, périphérique bloc, ou périphérique SCSI de transfert pour exporter en tant que « backstore ».

#### Exemple 32.1. Exemple 1 de définition d'un périphérique

```
> backstores/block create example1 /dev/sda4
```

Ceci crée un « backstore » nommé **example1**, qui mène au périphérique bloc **/dev/sda4**.

#### Exemple 32.2. Exemple 2 de définition d'un périphérique

```
> backstores/fileio create example2 /srv/example2.img 100M
```

Ceci crée un « backstore » nommé **example2**, qui mène au fichier donné. Si le fichier n'existe pas, il sera créé. La taille de fichier peut utiliser les abréviations K, M, ou G et n'est utile que lorsque le fichier de sauvegarde n'existe pas.



#### NOTE

Si l'option globale **auto\_cd\_after\_create** est activée (ce qui est le cas par défaut), exécuter une commande de création modifiera le nœud de configuration actuel et le changera en l'objet nouvellement créé. Ceci peut être désactivé avec **set global auto\_cd\_after\_create=false**. Retourner au nœud root est possible avec **cd /**.

5. 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.

6. Mettre en correspondance un « backstore » avec une instance cible.

**Exemple 32.3. Exemple de mise en correspondance d'un « backstore » avec l'instance cible.**

```
/> cd tcm_fc/00:11:22:33:44:55:66:77
```

```
/> luns/ create /backstores/fileio/example2
```

7. Autoriser l'accès au LUN à partir d'un initiateur FCoE.

```
/> acIs/ create 00:99:88:77:66:55:44:33
```

Le LUN devrait désormais être accessible à cet initiateur.

8. Quittez **targetcli** en saisissant **exit** ou **ctrl+D**.

Le fait de quitter **targetcli** enregistrera la configuration par défaut. Celle-ci peut toutefois être explicitement enregistrée par la commande **saveconfig**.

Veuillez consulter la page man **targetcli** pour obtenir davantage d'informations.

## CHAPITRE 33. 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 33.1. 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\/*_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.

## CHAPITRE 34. 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 26.1, « Capacités 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 [Chapitre 38, 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 [Chapitre 31, 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.

## CHAPITRE 35. 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, c'est-à-dire des périphériques de Chelsio, Broadcom, et ServerEngines.

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.

### 35.1. AFFICHER LES CONFIGURATIONS IFACE DISPONIBLES

À partir de Red Hat Enterprise Linux 5.5 iSCSI, les déchargements et les liaisons d'interface sont pris en charge par les implémentations d'initiateur iSCSI suivantes :

- La pile *Software iSCSI* — comme les modules `scsi_tcp` et `ib_iser` 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.
- Le pile *Offload iSCSI* — comme le modules Chelsio `cxgb3i`, Broadcom `bnx2i` et ServerEngines `be2iscsi` modules, 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 différent `scsi_host` par 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 35.1. Configurations iface**

Paramètre	Description
<b>iface_name</b>	nom de configuration <b>iface</b> .
<b>transport_name</b>	Nom du pilote
<b>hardware_address</b>	adresse MAC
<b>ip_address</b>	Adresse IP pour ce port
<b>net_iface_name</b>	Nom utilisé pour le <b>vlan</b> ou pour l'alias de liaison d'une sessions iSCSI. Pour les déchargements iSCSI, <b>net_iface_name</b> sera <b>&lt;vide&gt;</b> car cette valeur n'est pas persistante au redémarrage.
<b>initiator_name</b>	Configuration utilisée pour substituer un nouveau nom au nom par défaut de l'initiateur, défini dans <b>/etc/iscsi/initiatorname.iscsi</b>

### Exemple 35.1. 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 35.2. 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 35.3. 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
```

## 35.2. 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 35.4. 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.

## 35.3. CONFIGURER UN IFACE POUR LE DÉCHARGEMENT ISCSI

Par défaut, **iscsiadm** créera une configuration **iface** pour chaque port Chelsio, Broadcom, et ServerEngines. 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**) que le périphérique devrait utiliser. Pour les périphériques ServerEngines qui utilisent le pilote **be2iscsi** (c'est-à-dire des HBA iSCSI ServerEngines), l'adresse IP est configurée dans l'écran de paramétrage BIOS de ServerEngines.

Pour les périphériques Chelsio et Broadcom, la procédure de configuration de l'adresse IP est la même que pour tout autre paramètre **iface**. Ainsi, pour configurer l'adresse IP de l'**iface**, veuillez utiliser :

```
# iscsiadm -m iface -I iface_name -o update -n iface.ipaddress -v
target_IP
```

### Exemple 35.5. Paramétrez l'adresse IP de l'iface d'une carte Chelsio

Par exemple, pour définir l'adresse IP **iface** d'une carte Chelsio (avec le nom **iface cxgb3i.00:07:43:05:97:07**) sur **20.15.0.66**, veuillez utiliser :

```
# iscsiadm -m iface -I cxgb3i.00:07:43:05:97:07 -o update -n
iface.ipaddress -v 20.15.0.66
```

## 35.4. 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
```

Par défaut, l'utilitaire **iscsiadm** ne liera pas automatiquement tous les portails aux configurations

**iface** utilisant le téléchargement. Ceci est dû au fait que de telles configurations **iface** n'ont pas paramétré **iface.transport** sur **tcp**. Ainsi, les configurations **iface** des ports Chelsio, Broadcom, et ServerEngines 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[7]
```

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.

---

[7] Refer to [Chapitre 36, Scanner les Cibles iSCSI par des Portails ou LUN multiples](#) for information on **proper\_target\_name**.

## CHAPITRE 36. SCANNER LES CIBLES ISCSI PAR DES PORTAILS OU LUN MULTIPLES

Cependant, avec certains modèles de périphérique (par exemple EMC et Netapp), une cible unique peut avoir de multiples unités logiques ou 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[8]
```

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. Puis scannez à nouveau les sessions existantes pour découvrir de nouvelles unités logiques (par exemple, 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 36.1. 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 27.2, « Création de l'initiateur 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 [9]
```

### Exemple 36.2. 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 [9]
```

---

[8] Pour obtenir des informations sur la manière de récupérer la valeur SID d'une session, veuillez consulter [Section 27.2, « Création de l'initiateur iSCSI »](#).

[9] Ceci est une commande unique divisée sur plusieurs lignes, pour accommoder les versions imprimées et PDF de ce document. Toutes les lignes concaténées — précédées par une barre oblique inverse (\) — doivent être traitées comme une seule commande, sans barres obliques inverses.



## CHAPITRE 37. 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.

### 37.1. 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.

### 37.2. 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
```

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 [Chapitre 36, Scanner les Cibles iSCSI par des Portails ou LUN multiples](#)).
- 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.

## 37.3. 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*). Pour Red Hat Enterprise Linux 5.3 (ou ses versions supérieures), 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.

Si le système utilise une version de Red Hat Enterprise Linux 5.0 à 5.2, vous devrez effectuer la procédure suivante pour ordonner au démon **multipathd** de reconnaître les modifications apportées à l'unité logique redimensionnée (et de s'y ajuster) :

### Procédure 37.1. Redimensionner le périphérique multivoies correspondant (requis pour Red Hat Enterprise Linux 5.0 - 5.2)

1. Videz la table du mappeur de périphérique du périphérique multivoies en utilisant :

```
dmsetup table multipath_device
```

2. Enregistrez la table du mappeur de périphériques vidé sous le nom **table\_name**. Cette table sera chargée à nouveau et modifiée ultérieurement.
3. Examinez la table du mappeur de périphériques. Remarquez que les deux premiers chiffres de chaque ligne correspondent respectivement aux secteurs du début et de la fin du disque.
4. Veuillez suspendre la cible du mappeur de périphériques :

```
dmsetup suspend multipath_device
```

5. Ouvrez la table du mappeur de périphériques enregistrée au préalable (**table\_name**). Modifiez le second chiffre (le secteur de fin du disque) afin qu'il reflète le nouveau chiffre de 512 secteurs d'octets sur le disque. Par exemple, si la nouvelle taille de disque fait 2 Go, le second chiffre devra être 4194304.

6. Veuillez charger à nouveau la table modifiée du mappeur de périphériques :

```
dmsetup reload multipath_device table_name
```

7. Réactivez la cible du mappeur de périphériques :

```
dmsetup resume multipath_device
```

Pour obtenir davantage d'informations sur les multiples voies, veuillez consulter le *Guide Red Hat Enterprise Linux 6 DM Multipath*.

## 37.4. 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 6 :

```
# 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][hwandler=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 37.2. 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 R/W. 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 R/W du système d'exploitation du périphérique. Si vous utilisez un périphérique mappé 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 37.4.1, « Scanner à nouveau des unités logiques »](#).

#### 37.4.1. Scanner à nouveau des unités logiques

Après avoir modifié l'état R/W de l'unité logique en ligne comme décrit dans [Section 37.4, « 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 37.1. 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
```

### 37.4.2. Mettre à jour l'état R/W 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 avec la commande suivante :

```
# multipath -r
```

La commande **multipath -11** peut être utilisée pour confirmer le changement.

### 37.4.3. 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>.

## CHAPITRE 38. 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** devrait 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.

## CHAPITRE 39. MODIFIER LE COMPORTEMENT DE LA PERTE DE LIEN DE LIEN

Cette section décrit comment modifier le comportement de la perte de lien des périphériques utilisant les protocoles fibre channel ou iSCSI.

### 39.1. 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 39.1. 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 39.2. 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 26.1, « Interface de programmation Fibre Channel »](#).

Lorsqu'un port de cible ou de lien dépasse la valeur `dev_loss_tmo`, les périphériques `scsi_device` et `sdM` sont supprimés. La liaison d'ID SCSI du port cible est enregistrée. Lorsque la cible est retournée, l'adresse SCSI et les assignations `sdM` peuvent être modifiées. L'adresse SCSI changera si tout changement de la configuration LUN se produit derrière le port cible. Les noms `sdM` peuvent être modifiés selon les variations du timing pendant le processus de découverte LUN ou en raison d'une modification de la configuration LUN dans le stockage. Ces assignations ne sont pas persistantes comme décrit dans [Chapitre 28, \*Dénomination persistante\*](#). Veuillez consulter la section [Chapitre 28, \*Dénomination persistante\*](#) pour connaître des méthodes alternatives de dénomination de périphérique qui soient persistantes.

## 39.2. PARAMÈTRES ISCSI AVEC DM-MULTIPATH

Si **dm-multipath** est implémenté, il est recommandé de paramétrer les minuteurs iSCSI immédiatement pour déléguer 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 problèmes 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.

### 39.2.1. 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 iSCSI 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 **dm-multipath** est utilisé, la couche SCSI fera échouer les commandes en cours d'exécution et les déléguera 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 = [valeur de l'intervalle]
```

Une fois définie, la couche iSCSI enverra une requête NOP-Out sur chaque cible toutes les *[valeur de l'intervalle]* secondes.

Par défaut, le délai d'expiration des requêtes NOP-Out est de 10 secondes<sup>[10]</sup>. Pour ajuster cela, ouvrez le fichier **/etc/iscsi/iscsid.conf** et modifiez la ligne suivante :

```
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 39.2.2](#), « **replacement\_timeout** ».

Pour vérifier si SCSI Error Handler est cours d'exécution :



```
# iscsiadm -m session -P 3
```

### 39.2.2. replacement\_timeout

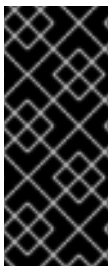
**replacement\_timeout** contrôle le temps que la couche iSCSI doit attendre pour qu'un chemin ou une session dont le délai a expiré puisse se rétablir avant de faire échouer ses commandes. La valeur par défaut de **replacement\_timeout** s'élève à 120 secondes.

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 39.2, « 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 mappeur 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.

## 39.3. 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 [Chapitre 36, Scanner les Cibles iSCSI par des Portails ou LUN multiples](#).

## 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 39.2, « Paramètres iSCSI avec dm-multipath »](#).

---

[10] Avant Red Hat Enterprise Linux 5.4, le délai d'expiration des requêtes NOP-Out était de 15 secondes.

## CHAPITRE 40. 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 de canal de fibre 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, vous pouvez écrire dans **/sys/block/*device-name*/device/timeout**. Pour cela, exécutez :

```
echo value /sys/block/device-name/device/timeout
```

Ici, **vaieur** correspond à la valeur du délai d'expiration (en secondes) que vous souhaitez implémenter.

## CHAPITRE 41. RÉOLUTION 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 41.1. 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 41.1. 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 41.2. 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
```

## ANNEXE A. HISTORIQUE DES VERSIONS

<b>Version 2-64.3</b> Translation completed	<b>Thu Oct 13 2016</b>	<b>Corina Galicher Roe</b>
<b>Version 2-64.2</b> Fichiers de traduction synchronisés avec les sources XML 2-64	<b>Thu Oct 13 2016</b>	<b>Corina Galicher Roe</b>
<b>Version 2-64.1</b> Fichiers de traduction synchronisés avec les sources XML 2-64	<b>Fri Sep 23 2016</b>	<b>Terry Chuang</b>
<b>Version 2-64</b> Préparation de la documentation 6.8 GA.	<b>Thu May 10 2016</b>	<b>Milan Navratil</b>
<b>Version 2-63</b> Mise à jour asynchrone avec plusieurs améliorations mineures.	<b>Thu Mar 31 2016</b>	<b>Milan Navratil</b>
<b>Version 2-52</b> Chapitres sur la Sauvegarde Ext et la Restauration ajoutés	<b>Wed Mar 25 2015</b>	<b>Jacquelynn East</b>
<b>Version 2-51</b> Version de sortie 6.6 GA	<b>Thu Oct 9 2014</b>	<b>Jacquelynn East</b>
<b>Version 2-38</b> Build 6.5 GA	<b>Mon Nov 18 2013</b>	<b>Jacquelynn East</b>
<b>Version 2-35</b> Section fsck ajoutée.	<b>Thu Sep 05 2013</b>	<b>Jacquelynn East</b>
<b>Version 2-11</b> Version de sortie 6.4 GA	<b>Mon Feb 18 2013</b>	<b>Jacquelynn East</b>
<b>Version 2-1</b> Branche créée pour la version 6.4 Beta. Nouvelle édition créée suite à un nouvel ordonnancement de taille.	<b>Fri Oct 19 2012</b>	<b>Jacquelynn East</b>
<b>Version 1-45</b> Version de sortie 6.3.	<b>Mon Jun 18 2012</b>	<b>Jacquelynn East</b>