Red Hat Enterprise Linux 7

Guide de l'administrateur systèmes

Déploiement, configuration, et administration de Red Hat Enterprise Linux 7

Last Updated: 2018-06-04

Red Hat Enterprise Linux 7 Guide de l'administrateur systèmes

Déploiement, configuration, et administration de Red Hat Enterprise Linux 7

Maxim Svistunov
Red Hat Customer Content Services
msvistun@redhat.com

Marie Doleželová
Red Hat Customer Content Services
mdolezel@redhat.com

Stephen Wadeley
Red Hat Customer Content Services
swadeley@redhat.com

Tomáš Čapek
Red Hat Customer Content Services
tcapek@redhat.com

Jaromír Hradílek
Red Hat Customer Content Services

Douglas Silas
Red Hat Customer Content Services

Jana Heves
Red Hat Customer Content Services

Petr Kovář
Red Hat Customer Content Services

Peter Ondrejka
Red Hat Customer Content Services

Petr Bokoč
Red Hat Customer Content Services

Martin Prpič
Red Hat Sécurité Produit

Eliška Slobodová
Red Hat Customer Content Services

Eva Kopalová
Red Hat Customer Content Services

Miroslav Svoboda
Red Hat Customer Content Services

David O'Brien
Red Hat Customer Content Services

Notice légale

Red Hat Customer Content Services

Don Domingo
Red Hat Customer Content Services

Résumé

Le Guide de l'administrateur systèmes documente les informations importantes concernant le déploiement, la configuration et l'administration de Red Hat Enterprise Linux 7. Ce guide est conçu pour les administrateurs systèmes possédant une connaissance de base du système. En vue d'élargir vos connaissances, vous serez sans doute interessés par les formations suivantes Red Hat System Administration I (RH124), Red Hat System Administration II (RH134), Red Hat System Administration III (RH254), ou RHCSA Rapid Track (RH199).

Table des matières

PARTIE I. CONFIGURATION DE BASE DU SYSTÈME

Cette partie traite des tâches de base de l'administration de système, telles que la configuration du clavier, la configuration de la date et de l'heure, la gestion des utilisateurs et des groupes, et l'obtention de privilèges.

CHAPITRE 1. PARAMÈTRES RÉGIONAUX ET CONFIGURATION DU CLAVIER

Les *paramètres régionaux* indiquent les paramètres de langue des services et interfaces utilisateur du système. Les paramètres d'*agencement du clavier* contrôlent l'agencement utilisé sur la console texte et sur les interfaces utilisateur graphique.

Ces paramètres peuvent être effectués en modifiant le fichier de configuration **/etc/locale.conf** ou en utilisant l'utilitaire **localectl**. Ainsi, vous pouvez utiliser l'interface utilisateur graphique pour effectuer la tâche. Pour obtenir une description de la méthode, veuillez consulter le Guide d'installation Red Hat Enterprise Linux 7.

1.1. DÉFINIR LES PARAMÈTRES RÉGIONAUX

Les paramètres régionaux globaux sont stockés dans le fichier **/etc/locale.conf**, qui est lu au début du démarrage par le démon **systemd**. Les paramètres régionaux configurés dans **/etc/locale.conf** sont hérités par chaque service ou utilisateur, à moins qu'un programme ou utilisateur individuel ne l'outrepasse.

Le format de base de **/etc/locale.conf** est une liste séparée par des lignes d'affectation de variables. Voici des paramètres allemands avec des messages en anglais dans **/etc/locale.conf** :

```
LANG=de_DE.UTF-8
LC_MESSAGES=C
```

Ici, l'option LC_MESSAGES détermine les paramètres régionaux utilisés pour les messages de diagnostique écrits sur la sortie d'erreurs standard. Pour spécifier les paramètres régionaux dans **/etc/locale.conf**, vous pouvez utiliser plusieurs autres options. Les plus courantes sont résumées dans Tableau 1.1, « Options configurables dans /etc/locale.conf ». Veuillez consulter la page du manuel **locale(7)** pour obtenir des informations détaillées sur ces options. Veuillez remarquer que l'option LC_ALL, qui représente toutes les options possibles, ne doit pas être configurée dans **/etc/locale.conf**.

Tableau 1.1. Options configurables dans /etc/locale.conf

Option	Description
LANG	Fournit une valeur par défaut pour les paramètres régionaux.
LC_COLLATE	Modifie le comportement des fonctions qui comparent les chaînes dans l'alphabet local.
LC_CTYPE	Modifie le comportement des fonctions de gestion et de classification des caractères et les fonctions des caractères multioctets.
LC_NUMERIC	Décrit la manière par laquelle les chiffres sont habituellement imprimés, avec des détails tels que le point décimal versus la virgule décimale.

Option	Description
LC_TIME	Modifie l'affichage de l'heure actuelle, 24 heures versus 12 heures.
LC_MESSAGES	Détermine les paramètres régionaux utilisés pour les messages de diagnostique écrits dans la sortie d'erreur standard.

1.1.1. Afficher le statut actuel

La commande **localectl** peut être utilisée pour effectuer des requêtes et modifier les paramètres régionaux et les paramètres d'agencement du clavier. Pour afficher les paramètres actuels, veuilles utiliser l'option **status** :

```
localectl status
```

Exemple 1.1. Afficher le statut actuel

La sortie de la commande précédente répertorie les paramètres régionaux et la structure du clavier actuellement configurés pour la console et le système de fenêtres X11.

```
~]$ localectl status
   System Locale: LANG=en_US.UTF-8
       VC Keymap: us
      X11 Layout: n/a
```

1.1.2. Répertorier les paramètres régionaux disponibles

Pour répertorier tous les paramètres régionaux disponibles pour votre système, veuillez saisir :

```
localectl list-locales
```

Exemple 1.2. Répertorier les paramètres régionaux

Imaginez que vous souhaitiez sélectionner un paramètre régional anglais en particulier, mais que vous n'êtes pas sûr qu'il se trouve sur le système. Vous pourrez vérifier cela en répertoriant tous les paramètres régionaux anglais avec la commande suivante :

```
~]$ localectl list-locales | grep en_
en_AG
en_AG.utf8
en_AU
en_AU.iso88591
en_AU.utf8
en_BW
en_BW.iso88591
en_BW.utf8

output truncated
```

1.1.3. Définir les paramètres régionaux

Pour définir les paramètres régionaux du système par défaut, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

```
localectl set-locale LANG=locale
```

Remplacez *locale* par le nom du paramètre régional, trouvé par la commande **localectl list-locales**. La syntaxe ci-dessus peut également être utilisée pour configurer les paramètres de Tableau 1.1, « Options configurables dans /etc/locale.conf ».

Exemple 1.3. Modifier les paramètres régionaux par défaut

Ainsi, si vous souhaitez définir l'anglais britannique (« British English ») comme paramètre régional par défaut, commencez par trouver le nom du paramètre régional en utilisant **list-locales**. Puis, en tant qu'utilisateur **root**, saisir une commande du modèle suivant :

```
~]# localectl set-locale LANG=en_GB.utf8
```

1.2. MODIFIER L'AGENCEMENT DU CLAVIER

Les paramètres d'agencement du clavier permettent à l'utilisateur de contrôler la structure utilisée sur la console de texte et les interfaces utilisateur graphique.

1.2.1. Afficher les paramètres actuels

Comme mentionné précédemment, vous pouvez vérifier la configuration de l'agencement du clavier par la commande suivante :

```
localectl status
```

Exemple 1.4. Afficher les paramètres du clavier

Dans la sortie suivante, vous pouvez observer l'agencement du clavier configuré pour la console virtuelle et pour le système de fenêtres X11.

```
~]$ localectl status
   System Locale: LANG=en_US.utf8
       VC Keymap: us
      X11 Layout: us
```

1.2.2. Répertorier les agencements de clavier disponibles

Pour répertorier tous les agencements de clavier disponibles pouvant être configurés sur votre système, veuillez saisir :

```
localectl list-keymaps
```

Exemple 1.5. Rechercher un agencement de clavier particulier

La commande **grep** peut être utilisée pour rechercher un nom d'agencement de clavier particulier dans la sortie de la commande précédente. De multiples agencements de clavier sont souvent compatibles avec vos paramètres régionaux actuels. Par exemple, pour trouver des agencements de clavier tchèques, veuillez saisir :

```
~]$ localectl list-keymaps | grep cz
cz
cz-cp1250
cz-lat2
cz-lat2-prog
cz-qwerty
cz-us-qwertz
sunt5-cz-us
sunt5-us-cz
```

1.2.3. Définir l'agencement du clavier

Pour définir la structure du clavier par défaut de votre système, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

```
localectl set-keymap map
```

Remplacez *map* par le nom de l'agencement du clavier pris à partir de la sortie de la commande **localectl list-keymaps**. À moins que l'option **--no-convert** ne soit passée, le paramètre sélectionné est également appliqué au mappage du clavier par défaut du système de fenêtres X11, après l'avoir converti au mappage de clavier X11 correspondant le mieux. Ceci s'applique à l'inverse, vous pouvez spécifier les deux agencements de clavier avec la commande suivante en tant qu'utilisateur **root** :

```
localectl set-x11-keymap map
```

Si vous souhaitez que votre structure X11 diffère de la structure de la console, veuillez utiliser l'option **--no-convert**.

```
localectl --no-convert set-x11-keymap map
```

Avec cette option, l'agencement du clavier X11 est indiqué sans changer le paramètre de structure de la console précédente.

Exemple 1.6. Définir l'agencement du clavier X11 séparément

Imaginez que vous souhaitiez utiliser une structure de clavier allemande dans l'interface graphique, mais que vous souhaitiez conserver un agencement de clavier en anglais américain (« US English ») pour les opérations de la console. Dans ce cas, en tant qu'utilisateur **root**, veuillez saisir :

```
~]# localectl --no-convert set-x11-keymap de
```

Puis, vous pouvez vérifier si ce paramétrage a fonctionné en examinant le statut actuel :

```
~]$ localectl status
   System Locale: LANG=de_DE.UTF-8
       VC Keymap: us
      X11 Layout: de
```

Hormis la structure de clavier (*map*), trois autres options peuvent être spécifiées :

```
localectl set-x11-keymap map model variant options
```

Remplacez *model* par le nom de modèle du clavier, et *variant* et *options* par la variante du clavier et les composants d'option, qui peuvent être utilisés pour améliorer le comportement du clavier. Ces options ne sont pas définies par défaut. Pour obtenir davantage d'informations sur le modèle X11, la variante X11, et sur les options X11, veuillez consulter la page man **kbd(4)**.

1.3. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur la manière de configurer la structure du clavier sur Red Hat Enterprise Linux, veuillez consulter les ressources ci-dessous :

Documentation installée

- **localectl**(1) — la page du manuel de l'utilitaire de ligne de commande**localectl** documente comment utiliser cet outil pour configurer les paramètres régionaux du système et la structure du clavier.
- **loadkeys**(1) — la page du manuel de la commande **loadkeys** fournit des informations supplémentaires sur la manière d'utiliser cet outil pour modifier la structure du clavier dans une console virtuelle.

Voir aussi

- Le Chapitre 5, *Obtention de privilèges* documente comment obtenir des privilèges administratifs en utilisant les commandes **su** et **sudo**.
- Le Chapitre 9, *Gérer les services avec systemd* fournit des informations supplémentaires sur **systemd** et documente comment utiliser la commande **systemctl** pour gérer des services de système.

CHAPITRE 2. CONFIGURER L'HEURE ET LA DATE

Les systèmes d'exploitation modernes font la distinction entre les deux types d'horloges suivants :

- Une *horloge temps réel* (« Real-Time Clock », ou RTC), communément appelée *horloge matérielle*, (habituellement un circuit intégré sur la carte système) est complètement indépendante de l'état actuel du système d'exploitation et fonctionne même lorsque l'ordinateur est éteint.
- Une *horloge système*, également appelée *horloge logicielle*, est habituellement maintenue par le noyau et sa valeur initiale est basée sur l'horloge temps réel. Une fois le système démarré et l'horloge système initialisée, celle-ci est entièrement indépendante de l'horloge temps réel.

Le temps système est toujours conservé sous le format du temps universel coordonné (« *Coordinated Universal Time* », ou UTC) puis converti dans les applications au temps local selon les besoins. Le *temps local* correspond à l'heure réelle dans votre fuseau horaire et prend en compte l'heure d'été (« *daylight saving time* », ou DST). L'horloge temps réel peut utiliser l'heure UTC ou l'heure locale. L'heure UTC est recommandée.

Red Hat Enterprise Linux 7 offre trois outils de ligne de commande pouvant être utilisés pour configurer et afficher des informations sur l'heure et la date du système : l'utilitaire **`timedatectl`**, qui est nouveau sur Red Hat Enterprise Linux 7 et fait partie de **`systemd`** ; la commande traditionnelle **`date`** ; et l'utilitaire **`hwclock`** pour accéder à l'horloge matérielle.

2.1. UTILISATION DE LA COMMANDE `TIMEDATECTL`

L'utilitaire **timedatectl** est distribué dans le cadre du gestionnaire de services et systèmes **`systemd`** et permet de réviser et modifier la configuration de l'horloge système. Vous pouvez utiliser cet outil pour modifier l'heure et la date actuelle, pour définir le fuseau horaire, ou pour activer la synchronisation automatique de l'horloge système avec un serveur distant.

Pour obtenir des informations sur la manière d'afficher l'heure et la date actuelle sous un format personnalisé, veuillez également consulter la Section 2.2, « Utiliser la commande date ».

2.1.1. Afficher l'heure et la date actuelle

Pour afficher l'heure et la date actuelle, ainsi que des informations détaillées sur la configuration de l'horloge système et de l'horloge matérielle, veuillez exécuter la commande **`timedatectl`** sans aucune autre option de ligne de commande :

```
timedatectl
```

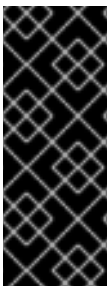
Ceci affiche l'heure locale et universelle, le fuseau horaire en cours d'utilisation, le statut de la configuration **`NTP`** (« Network Time Protocol »), ainsi que des informations supplémentaires liées à DST.

Exemple 2.1. Afficher l'heure et la date actuelle

Voici un exemple de la sortie de commande **`timedatectl`** sur un système qui n'utilise pas **`NTP`** pour synchroniser l'horloge système avec un serveur distant :

```
~]$ timedatectl
      Local time: Mon 2013-09-16 19:30:24 CEST
  Universal time: Mon 2013-09-16 17:30:24 UTC
        Timezone: Europe/Prague (CEST, +0200)
```

```
      NTP enabled: no
NTP synchronized: no
 RTC in local TZ: no
      DST active: yes
 Last DST change: DST began at
                  Sun 2013-03-31 01:59:59 CET
                  Sun 2013-03-31 03:00:00 CEST
 Next DST change: DST ends (the clock jumps one hour backwards) at
                  Sun 2013-10-27 02:59:59 CEST
                  Sun 2013-10-27 02:00:00 CET
```



IMPORTANT

Les changements au statut de **chrony** ou à **ntpd** ne seront pas notés immédiatement par **timedatectl**. S'il y a eu des changements de configuration ou de statut à ces outils, saisir la commande suivante :

```
~]# systemctl restart systemd-timedated.services
```

2.1.2. Modifier l'heure actuelle

Pour modifier l'heure actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root**:

```
timedatectl set-time HH:MM:SS
```

Veuillez remplacer *HH* par les heures, *MM* par les minutes, et *SS* par les secondes, le tout doit être saisi sous un format à deux chiffres.

Cette commande met à jour l'heure système et l'horloge matérielle. Le résultat est similaire à l'utilisation des commandes **date --set** et **hwclock --systohc**.

La commande échouera si un service **NTP** est activé. Voir Section 2.1.5, « Synchroniser l'horloge système avec un serveur à distance » pour désactiver le service de façon temporaire.

Exemple 2.2. Modifier l'heure actuelle

Pour modifier l'heure actuelle à 11:26 p.m. (23h26), veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# timedatectl set-time 23:26:00
```

Par défaut, le système est configuré pour utiliser l'heure UTC. Pour configurer votre système à maintenir l'horloge à l'heure locale, veuillez exécuter la commande **timedatectl** avec l'option **set-local-rtc** en tant qu'utilisateur **root** :

```
timedatectl set-local-rtc boolean
```

Pour configurer votre système afin de maintenir l'horloge à l'heure locale, veuillez remplacer *boolean* par **yes** (ou bien par **y**, **true**, **t**, ou **1**). Pour configurer le système de manière à utiliser UTC, veuillez remplacer *boolean* par **no** (ou bien par, **n**, **false**, **f**, ou **0**). L'option par défaut est **no**.

2.1.3. Modifier la date actuelle

Pour modifier la date actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root**:

```
timedatectl set-time YYYY-MM-DD
```

Veuillez remplacer *YYYY* par une année à quatre chiffres, *MM* par un mois à deux chiffres, et *DD* par un jour du mois à deux chiffres.

Veuillez remarquer que modifier la date sans spécifier l'heure actuelle fera que l'heure sera paramétrée sur 00:00:00.

Exemple 2.3. Modifier la date actuelle

Pour modifier la date actuelle sur le 2 juin 2013 et garder la même heure (11:26 p.m., ou 23h26), veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# timedatectl set-time '2013-06-02 23:26:00'
```

2.1.4. Modifier le fuseau horaire

Pour répertorier tous les fuseaux horaire disponibles, veuillez saisir ce qui suit dans une invite de shell :

```
timedatectl list-timezones
```

Pour modifier le fuseau horaire en cous d'utilisation, veuillez saisir ce qui suit en tant qu'utilisateur **root** :

```
timedatectl set-timezone time_zone
```

Veuillez remplacer *time_zone* par l'une des valeurs répertoriées par la commande **timedatectl list-timezones**.

Exemple 2.4. Modifier le fuseau horaire

Pour identifier le fuseau horaire le plus proche de votre location actuelle, veuillez utiliser la commande **timedatectl** avec l'option de ligne de commande **list-timezones**. Par exemple, pour répertorier tous les fuseaux horaires en Europe, veuillez saisir :

```
~]# timedatectl list-timezones | grep Europe
Europe/Amsterdam
Europe/Andorra
Europe/Athens
Europe/Belgrade
Europe/Berlin
Europe/Bratislava…
```

Pour changer le fuseau horaire sur **Europe/Prague**, veuillez saisir ce qui suit en tant qu'utilisateur **root** :

```
~]# timedatectl set-timezone Europe/Prague
```

2.1.5. Synchroniser l'horloge système avec un serveur à distance

Contrairement aux ajustements manuels décrits dans les sections précédentes, la commande **timedatectl** vous permet également d'activer la synchronisation automatique de votre horloge système avec un groupe de serveurs à distance en utilisant le protocole **NTP**. L'activation de NTP active **chronyd** ou **ntpd**, selon le service installé.

Le service **NTP** peut être activé et désactivé par une commande sur le modèle suivant :

```
timedatectl set-ntp boolean
```

Pour activer votre système pour synchroniser l'horloge système avec un serveur **NTP** à distance, veuillez remplacer *boolean* par **yes** (option par défaut). Pour désactiver cette fonctionnalité, veuillez remplacer *boolean* par **no**.

Exemple 2.5. Synchroniser l'horloge système avec un serveur à distance

Pour activer la synchronisation automatique de l'horloge système avec un serveur à distance, veuillez saisir :

```
~]# timedatectl set-ntp yes
```

La commande échouera si un service **NTP** n'est pas installé. Voir Section 15.3.1, « Installer Chrony » pour plus d'informations.

2.2. UTILISER LA COMMANDE DATE

L'utilitaire **date** est disponible sur tous les systèmes Linux et vous permet d'afficher et de configurer l'heure et la date actuelle. Celle-ci est fréquemment utilisée dans les scripts pour afficher des informations détaillées sur l'horloge système sous un format personnalisé.

Pour obtenir des informations sur la manière de modifier le fuseau horaire ou pour activer la synchronisation automatique de l'horloge système avec un serveur à distance, veuillez consulter la Section 2.1, « Utilisation de la commande **timedatectl** ».

2.2.1. Afficher l'heure et la date actuelle

Pour afficher l'heure et la date actuelle, veuillez exécuter la commande **date** sans aucune autre option de ligne de commande :

```
date
```

Celle-ci affiche le jour de la semaine, suivi de la date, de l'heure locale, de l'abbréviation du fuseau horaire, et de l'année.

Par défaut, la commande **date** affiche l'heure local. Pour afficher l'heure UTC, veuillez exécuter la commande avec l'option de ligne de commande **--utc** ou **-u** :

```
date --utc
```

Il est également possible de personnaliser le format des informations affichées en ajoutant l'option **+"*format*"** sur la ligne de commande :

```
date +"format"
```

Remplacez *format* par une ou plusieurs séquences de contrôle prises en charge, comme illustré dans l'Exemple 2.6, « Afficher l'heure et la date actuelle ». Veuillez consulter Tableau 2.1, « Séquences de contrôle couramment utilisées » pour une liste des options de formatage les plus fréquemment utilisées, ou la page de manuel **date**(1) pour une liste complète de ces options.

Tableau 2.1. Séquences de contrôle couramment utilisées

Séquence de contrôle	Description
%H	Heures sous le format *HH* (par exemple, **17**).
%M	Minutes sous le format *MM* (par exemple, **30**).
%S	Secondes sous le format *SS* (par exemple, **24**).
%d	Jour du mois sous le format *DD* (par exemple, **16**).
%m	Mois sous le format *MM* (par exemple, **09**).
%Y	Année sous le format *YYYY* (par exemple, **2013**).
%Z	Abbréviation du fuseau horaire (par exemple, **CEST**).
%F	Date complète sous le format *YYYY-MM-DD* (par exemple, **2013-09-16**). Cette option correspond à **%Y-%m-%d**.
%T	Heure complète sous le format *HH:MM:SS* (par exemple, 17:30:24). Cette option correspond à **%H:%M:%S**

Exemple 2.6. Afficher l'heure et la date actuelle

Pour afficher l'heure et la date locale, veuillez saisir ce qui suit dans une invite de shell :

```
~]$ date
Mon Sep 16 17:30:24 CEST 2013
```

Pour afficher l'heure et la date UTC, veuillez saisir ce qui suit dans une invite de shell :

```
~]$ date --utc
Mon Sep 16 15:30:34 UTC 2013
```

Pour personnaliser la sortie de la commande **date**, veuillez saisir :

```
~]$ date +"%Y-%m-%d %H:%M"
2013-09-16 17:30
```

2.2.2. Modifier l'heure actuelle

Pour modifier l'heure actuelle, veuillez exécuter la commande **date** avec l'option **--set** ou **-s** en tant qu'utilisateur **root** :

```
date --set HH:MM:SS
```

Veuillez remplacer *HH* par les heures, *MM* par les minutes, et *SS* par les secondes, le tout doit être saisi sous un format à deux chiffres.

Par défaut, la commande **date** définit l'horloge système sur l'heure locale. Pour définir l'horloge système sur UTC, exécutez la commande avec l'option de ligne de commande **--utc** ou **-u** :

```
date --set HH:MM:SS --utc
```

Exemple 2.7. Modifier l'heure actuelle

Pour modifier l'heure actuelle à 11:26 p.m. (23h26), veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# date --set 23:26:00
```

2.2.3. Modifier la date actuelle

Pour modifier la date actuelle, veuillez exécuter la commande **date** avec l'option **--set** ou **-s** en tant qu'utilisateur **root** :

```
date --set YYYY-MM-DD
```

Veuillez remplacer *YYYY* par une année à quatre chiffres, *MM* par un mois à deux chiffres, et *DD* par un jour du mois à deux chiffres.

Veuillez remarquer que modifier la date sans spécifier l'heure actuelle fera que l'heure sera paramétrée sur 00:00:00.

Exemple 2.8. Modifier la date actuelle

Pour modifier la date actuelle sur le 2 juin 2013 et garder la même heure (11:26 p.m., ou 23h26), veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# date --set 2013-06-02 23:26:00
```

2.3. UTILISATION DE LA COMMANDE HWCLOCK

hwclock est un utilitaire pour accéder à l'horloge matérielle, également appelée horloge RTC (« Real Time Clock »). L'horloge matérielle est indépendante du système d'exploitation utilisé et fonctionne même lorsque l'ordinateur est éteint. Cet utilitaire est utilisé pour afficher l'heure de l'horloge matérielle. **hwclock** offre aussi la possibilité de compenser pour la dérive systématique de l'horloge matérielle.

L'horloge matérielle stocke les valeurs de l'année, du mois, du jour, de l'heure, des minutes, et des secondes. Celle-ci n'est pas capable de stocker l'heure standard, l'heure locale, ou l'heure UTC (« Coordinated Universal Time »), ni de définir l'heure d'été (« Daylight Saving Time », ou DST).

L'utilitaire **hwclock** enregistre ses paramètres dans le fichier **/etc/adjtime**, qui est créé lors du premier changement effectué. Par exemple, lorsque l'heure est définie manuellement ou lorsque l'horloge matérielle est synchronisée avec l'heure système.

NOTE

Sur Red Hat Enterprise Linux 6, la commande **hwclock** était exécutée automatiquement à chaque fermeture ou redémarrage du système, mais ceci n'est pas le cas sur Red Hat Enterprise Linux 7. Lorsque l'horloge système est synchronisée par le protocole NTP (« Network Time Protocol ») ou PTP (« Precision Time Protocol »), le noyau synchronise automatiquement l'horloge matérielle avec l'horloge système toutes les 11 minutes.

Pour obtenir des détails sur NTP, veuillez consulter le Chapitre 15, *Configurer NTP en utilisant Chrony Suite* et le Chapitre 16, *Configurer NTP à l'aide de ntpd*. Pour obtenir des informations sur PTP, veuillez consulter le Chapitre 17, *Configurer PTP en utilisant ptp4l*. Pour obtenir des informations sur le paramétrage de l'horloge matérielle après avoir exécuté **ntpdate**, veuillez consulter la Section 16.18, « Configurer la mise à jour de l'horloge matérielle ».

2.3.1. Afficher l'heure et la date actuelle

Exécuter **hwclock** sans aucun option de ligne de commande en tant qu'utilisateur **root** retourne l'heure et la date locale sur la sortie standard.

```
hwclock
```

Remarquez qu'utiliser les options **--utc** ou **--localtime** avec la commande **hwclock** ne signifie pas que vous tentiez d'afficher l'heure de l'horloge matérielle en temps UTC ou en temps local. Ces options sont utilisées pour définir l'horloge matérielle de manière à conserver l'heure. L'heure est toujours affichée en heure locale. En outre, l'utilisation des commandes **hwclock --utc** ou **hwclock --local** ne modifie pas l'enregistrement dans le fichier **/etc/adjtime**. Cette commande peut être utile lorsque vous savez que le paramètre enregistré dans **/etc/adjtime** est incorrect mais que vous ne souhaitez pas modifier ce paramètre. D'autre part, vous pourriez recevoir des informations risquant de vous induire en erreur si vous utilisez une commande de la mauvaise manière. Veuillez consulter la page man de **hwclock**(8) pour obtenir davantage de détails.

Exemple 2.9. Afficher l'heure et la date actuelle

Pour afficher la date et l'heure locale actuelle à partir de l'horloge matérielle, veuillez exécuter ceci en tant qu'utilisateur **root** :

```
~]# hwclock
Tue 15 Apr 2014 04:23:46 PM CEST     -0.329272 seconds
```

CEST est une abbréviation de fuseau horaire et signifie heure d'été d'Europe centrale (« Central European Summer Time »).

Pour obtenir des informations sur la manière de modifier le fuseau horaire, veuillez consulter la Section 2.1.4, « Modifier le fuseau horaire ».

2.3.2. Paramétrer l'heure et la date

En plus d'afficher l'heure et la date, vous pouvez manuellement paramétrer l'horloge matérielle sur une heure particulière.

Lorsque vous aurez besoin de changer l'heure et la date de l'horloge matérielle, vous pourrez le faire en ajoutant les options **--set** et **--date** dans vos spécifications :

```
hwclock --set --date "dd mmm yyyy HH:MM"
```

Remplacez *dd* par un jour (un nombre à deux chiffres), *mmm* par un mois (une abbréviation à trois lettres), *yyyy* par une année (nombre à quatre chiffres), *HH* par l'heure (un nombre à deux chiffres), *MM* par les minutes (un nombre à deux chiffres).

Au même moment, vous pouvez également définir l'horloge matérielle pour conserver l'heure UTC ou locale en ajoutant l'option **--utc** ou **--localtime**, respectivement. Dans ce cas, **UTC** ou **LOCAL** est enregistré dans le fichier **/etc/adjtime**.

Exemple 2.10. Paramétrer l'horloge matérielle à une heure et date en particulier

Si vous souhaitez paramétrer l'heure et la date avec des valeurs particulières, par exemple « 21h17, 21 octobre, 2014 », et laisser l'horloge matérielle réglée sur UTC, veuillez exécuter la commande en tant qu'utilisateur **root** sous le format suivant :

```
~]# hwclock --set --date "21 Oct 2014 21:17" --utc
```

2.3.3. Synchroniser l'heure et la date

Vous pouvez synchroniser l'horloge matérielle et l'heure système actuelle dans les deux directions.

- Vous pouvez définir l'horloge matérielle sur l'heure système actuelle à l'aide de cette commande :

  ```
  hwclock --systohc
  ```

  Remarquez que si vous utilisez NTP, l'horloge matérielle est automatiquement synchronisée à l'horloge système toutes les 11 minutes, et cette commande n'est utile que pendant le démarrage pour obtenir une heure système initiale raisonnable.

- Vous pouvez également définir l'heure système à partir de l'horloge matérielle en utilisant la commande suivante :

  ```
  hwclock --hctosys
  ```

Lorsque vous synchronisez l'horloge matérielle et l'heure système, vous pouvez également spécifier si vous souhaitez conserver l'horloge matérielle en heure locale ou en heure UTC en ajoutant l'option **--utc** ou **--localtime**. De même qu'avec **--set**, **UTC** ou **LOCAL** est enregistré dans le fichier **/etc/adjtime**.

La commande **hwclock --systohc --utc** est fonctionnellement similaire à **timedatectl set-local-rtc false** et la commande **hwclock --systohc --local** est une alternative à **timedatectl set-local-rtc true**.

Exemple 2.11. Synchroniser l'horloge matérielle avec l'heure système

Pour définir l'horloge matérielle sur l'heure actuelle du système tout en gardant l'horloge matérielle en heure locale, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# hwclock --systohc --localtime
```

Pour éviter les problèmes liés aux fuseaux horaires et aux changements d'heure d'été, il recommandé de garder l'horloge matérielle en heure UTC. L'Exemple 2.11, « Synchroniser l'horloge matérielle avec l'heure système » affiché est utile, par exemple pour les cas de démarrages multiples avec un système Windows, qui suppose que l'horloge matérielle soit exécutée en heure locale par défaut, et que tous les autres systèmes doivent s'y accommoder en utilisant également l'heure locale. Cela peut également être utile avec une machine virtuelle ; si l'horloge matérielle virtuelle fournie par l'hôte est exécutée en heure locale, le système invité devra être configuré de manière à utiliser l'heure locale aussi.

2.4. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir des informations supplémentaires sur la manière de configurer l'heure et la date dans Red Hat Enterprise Linux 7, veuillez consulter les ressources répertoriées ci-dessous.

Documentation installée

- **timedatectl**(1) — la page du ma de l'utilitaire de ligne de commande **timedatectl** documente comment utiliser cet outil pour effectuer des requêtes et modifier l'horloge système et ses paramètres.
- **date**(1) — la page du man de la commande **date** fournit une liste complète des options de ligne de commande prises en charge.
- **hwclock**(8) — la page du man de la commande **hwclock** fournit une liste complète des options de ligne de commande.

Voir aussi

- Chapitre 1, *Paramètres régionaux et configuration du clavier* documente comment configurer l'agencement du clavier.
- Chapitre 5, *Obtention de privilèges* documente comment obtenir des privilèges administratifs en utilisant les commandes **su** et **sudo**.
- Chapitre 9, *Gérer les services avec systemd* fournit davantage d'informations sur systemd et documente comment utiliser la commande **systemctl** pour gérer les services du système.

CHAPITRE 3. GÉRER LES UTILISATEURS ET LES GROUPES

Le contrôle des utilisateurs et des groupes est un élément essentiel de l'administration des systèmes Red Hat Enterprise Linux. Ce chapitre explique comment ajouter, gérer, et supprimer des utilisateurs et des groupes dans l'interface utilisateur graphique et en ligne de commande, des sujets avancés sont également traités, comme la création de répertoires de groupes.

3.1. INTRODUCTION AUX UTILISATEURS ET AUX GROUPES

Tandis que les utilisateurs peuvent être des personnes (des comptes liés à des utilisateurs physiques), ou des comptes existants pour des applications spécifiques, les groupes sont des expressions logiques qui permettent une certaine organisation en regroupant des utilisateurs œuvrant dans un but commun. Les utilisateurs appartenant à un groupe donné partagent les mêmes permissions, leur permettant de lire, d'écrire, ou d'exécuter les fichiers appartenant à ce groupe.

Chaque utilisateur peut être associé avec un numéro d'identification numérique unique, également appelé un ID d'utilisateur (« *user ID* », ou UID). Similairement, chaque groupe est associé avec un ID de groupe (« *group ID* », ou GID). L'utilisateur qui crée un fichier devient le propriétaire et le groupe propriétaire du fichier. Ce fichier reçoit également des permissions séparées de lecture, d'écriture et d'exécution pour le propriétaire, le groupe ou tout autre utilisateur. Le propriétaire du fichier peut seulement être modifié par l'utilisateur **root**, et les permissions d'accès quant à elles peuvent être modifiées aussi bien par l'utilisateur **root**, que par le propriétaire du fichier.

En outre, Red Hat Enterprise Linux prend en charge les *listes de contrôle d'accès* (ACL) pour les fichiers et répertoires qui permettent d'octroyer des permissions à des utilisateurs spécifiques en dehors du propriétaire. Pour obtenir davantage d'informations sur cette fonctionnalité, veuillez consulter Chapitre 4, *Listes des contrôle d'accès (ACL)*.

ID d'utilisateurs et de groupes réservés

Red Hat Enterprise Linux réserve les ID de groupe de d'utilisateur inférieurs à 100 pour les groupes et les utilisateurs du système. **User Manager** n'affiche pas les utilisateurs du système. Les ID de groupe et d'utilisateurs sont documentés dans le package setup. Pour afficher la documentation, exécuter la commande :

```
cat /usr/share/doc/setup*/uidgid
```

La pratique courante est d'assigner des ID non réservés à partir de 5000, car cette gamme pourrait augmenter dans le futur. Pour que les ID assignés aux nouveaux utilisateurs par défaut puissent commencer à 5000, modifier les directives **UID_MIN** et **GID_MIN** dans le fichier **/etc/login.defs** :

```
[file contents truncated]
UID_MIN                  5000[file contents truncated]
GID_MIN                  5000[file contents truncated]
```

NOTE

Pour les utilisateurs qui auraient été créés avant le changement des directives **UID_MIN** et **GID_MIN**, les UID démarreront toujours par la valeur par défaut de 1000.

Même avec le nouvel utilisateur et les ID de groupe commençant par 5000, il est conseillé de ne pas augmenter les ID réservés par le système et supérieurs à 1000 pour éviter un conflit de systèmes retenant une limite de 1000.

3.1.1. Groupes privés d'utilisateurs

Red Hat Enterprise Linux utilise un schéma de *groupe privé d'utilisateurs* (ou *UPG*), qui rend la gestion des groupes UNIX plus facile. Un groupe privé d'utilisateurs est créé lorsqu'un nouvel utilisateur est ajouté au système. Il possède le même nom qu l'utilisateur pour lequel il a été créé et cet utilisateur est le seul membre du groupe privé d'utilisateurs.

Grâce à l'utilisation des groupes privés d'utilisateurs, il est possible de déterminer en toute sécurité des permissions par défaut pour un nouveau fichier ou répertoire afin que l'utilisateur et le *groupe de cet utilisateur* puissent modifier le fichier ou répertoire.

Le paramètre qui détermine quelles permissions sont appliquées au nouveau fichier ou répertoire créé s'appelle un *umask* et est configuré dans le fichier **/etc/bashrc**. Habituellement sur les système basés UNIX, l'**umask** a pour valeur **022**, ce qui permet uniquement à l'utilisateur ayant créé le fichier ou le répertoire d'effectuer des modifications. Sous ce schéma, tous les autres utilisateurs, *y compris les membres du groupe du créateur*, ne sont pas autorisés à effectuer des modifications. Cependant, sous le schéma UPG, cette « protection de groupe » n'est pas nécessaire puisque chaque utilisateur possède son propre groupe privé.

Une liste de tous les groupes est stockée dans le fichier de configuration **/etc/group**.

3.1.2. Mots de passe cachés (« Shadow Passwords »)

Dans les environnements avec de multiples utilisateurs, il est très important d'utiliser des « *mots de passe cachés* » fournis par le paquet shadow-utils afin d'améliorer la sécurité des fichiers d'authentification du système. Pour cette raison, le programme d'installation active les mots de passe cachés par défaut.

Ci-dessous figure une liste des avantages présentés par les mots de passe cachés comparé à la manière traditionnelle de stockage de mots de passe sur les systèmes basés UNIX

- Les mots de passe cachés améliorent la sécurité du système en déplaçant les hachages de mots de passe chiffrés depuis le fichier lisible **/etc/passwd** sur le fichier **/etc/shadow**, qui est uniquement lisible par l'utilisateur **root**.
- Les mots de passe cachés stockent des informations sur l'ancienneté du mot de passe.
- Les mots de passe cachés permettent d'appliquer les politiques de sécurité définies dans le fichier **/etc/login.defs** .

La plupart des utilitaires fournis par le paquet shadow-utils fonctionnent correctement, que les mots de passe cachés soient activés ou non. Cependant, comme les informations sur l'ancienneté des mots de passe sont exclusivement stockées dans le fichier **/etc/shadow**, certains utilitaires et certaines commandes ne fonctionneront pas si les mots de passe cachés ne sont pas activés :

- Utilitaire **chage** pour définir les paramètres d'ancienneté de mot de passe. Pour obtenir des détails, veuillez consulter la section Sécurité du mot de passe dans le *Guide de sécurité Red Hat Enterprise Linux 7*.
- Utilitaire **gpasswd** pour administrer le fichier **/etc/group**.
- Commande **usermod** avec l'option **-e, --expiredate** ou **-f, --inactive**.
- La commande **useradd** avec l'option **-e, --expiredate** ou **-f, --inactive**.

3.2. GESTION DES UTILISATEURS DANS UN ENVIRONNEMENT GRAPHIQUE

L'utilitaire **Users** vous permet d'afficher, de modifier, d'ajouter, et de supprimer des utilisateurs locaux dans l'interface utilisateur graphique.

3.2.1. Utiliser l'outil des paramètres d'utilisateurs « Users Settings Tool »

Veuillez appuyer sur la touche **Super** pour accéder à « Vue d'ensembles des activités », saisissez **Users** puis appuyez sur **Entrée**. L'outil des paramètres **Users** s'affiche. La touche **Super** apparaît sous une variété de formes, selon le clavier et matériel, mais le plus souvent, il s'agit de la touche Windows ou Commande, et elle se trouve habituellement à gauche de la barre d'espace. Alternativement, vous pouvez ouvrir l'utilitaire **Users** à partir du menu **Paramètres** après avoir cliqué sur votre nom d'utilisateur dans le coin supérieur droit de l'écran.

Pour effectuer des changements sur les comptes utilisateurs, veuillez commencer par sélectionner le bouton de déverrouillage « **Unlock** » et authentifiez-vous comme indiqué par la boîte de dialogue qui apparaît. Remarquez qu'à moins que vous ne possédiez des privilèges de super-utilisateur, l'application vous demandera de vous authentifier en tant qu'utilisateur **root**. Pour ajouter et supprimer des utilisateurs, veuillez sélectionner les boutons **+** et **-** respectivement. Pour ajouter un utilisateur au groupe administratif **wheel**, veuillez changer le **Type de compte** de **Standard** à **Administrateur**. Pour modifier le paramètre de langue d'un utilisateur, veuillez sélectionner la langue et un menu déroulant apparaîtra.

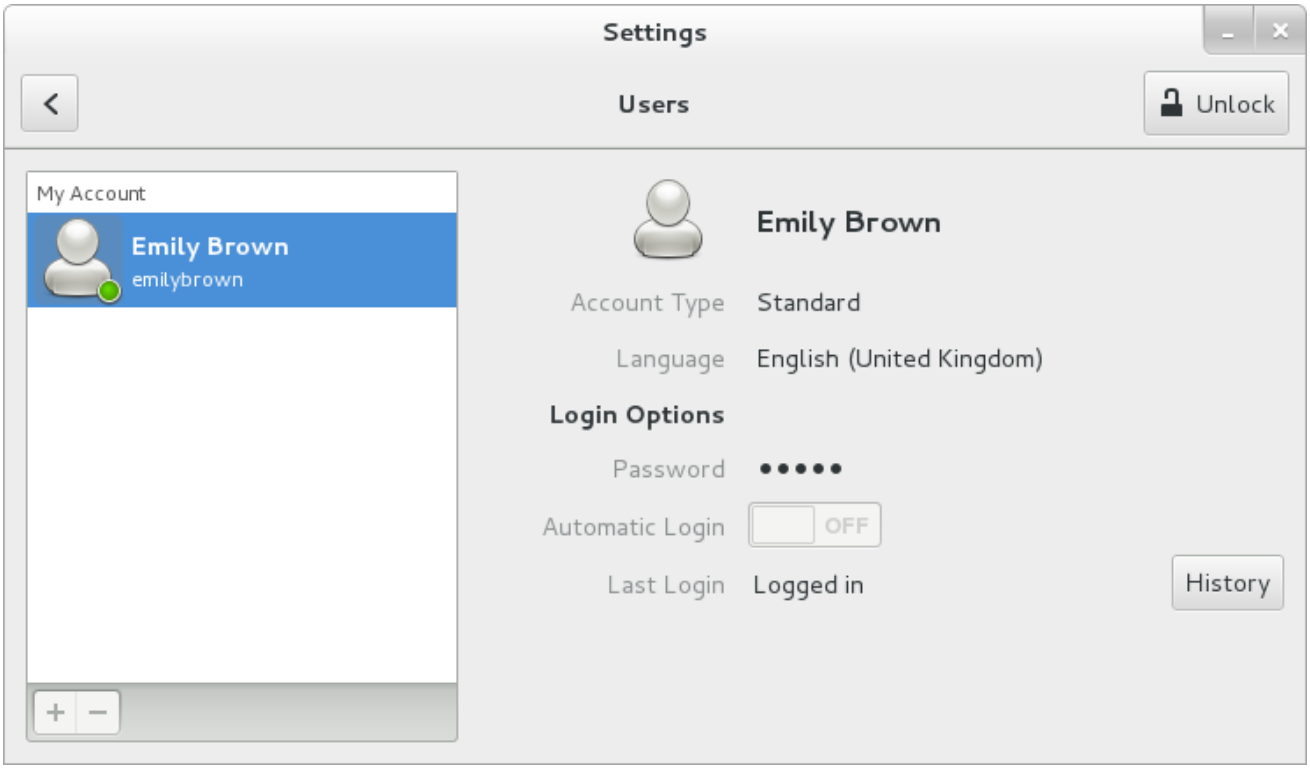


Figure 3.1. Outil des paramètres d'utilisateurs « Users Settings Tool »

Lorsqu'un nouvel utilisateur est créé, le compte est désactivé jusqu'à ce qu'un mot de passe soit défini. Le menu déroulant **Password**, affiché dans la Figure 3.2, « Menu Password », contient des options pour que l'administrateur définisse un mot de passe immédiatement, pour que l'utilisateur choisisse un mot de passe à la première connexion, ou crée un compte invité sans qu'un mot de passe ne soit requis pour se connecter. Un compte peut également être activé ou désactivé à partir de ce menu.

Figure 3.2. Menu Password

3.3. UTILISER DES OUTILS DE LIGNE DE COMMANDE

À l'exception de l'outil de configuration de **Users** décrit dans la Section 3.2, « Gestion des utilisateurs dans un environnement graphique », qui est conçu pour la gestion de base des utilisateurs, il est possible d'utiliser des outils de ligne de commande pour gérer les utilisateurs et les groupes répertoriés dans la Tableau 3.1, « Utilitaires en ligne de commande pour gérer les utilisateurs et les groupes ».

Tableau 3.1. Utilitaires en ligne de commande pour gérer les utilisateurs et les groupes

Utilitaires	Description
`id`	Affiche les ID d'utilisateur et de groupe.
`useradd`, **`usermod`**, **`userdel`**	Utilitaires standard pour ajouter, modifier, et supprimer des comptes utilisateur.
`groupadd`, **`groupmod`**, **`groupdel`**	Utilitaires standard pour ajouter, modifier, et supprimer des groupes.
`gpasswd`	Utilitaire utilisé surtout pour modifier le mot de passe du groupe dans le fichier **`/etc/gshadow`** utilisé par la commande **`newgrp`**.

Utilitaires	Description
pwck, **grpck**	Utilitaires pouvant être utilisés pour la vérification du mot de passe, du groupe et des fichiers cachés associés.
pwconv, **pwunconv**	Utilitaires pouvant être utilisés pour la conversion de mots de passe en mots de passe cachés, ou au contraire de mots de passe cachés en mots de passe standard.
grpconv, **grpunconv**	De manière similaire à ce qui précède, ces utilitaires peuvent être utilisés pour la conversion d'informations cachées pour les comptes de groupe.

3.3.1. Ajout d'un nouvel utilisateur

Pour ajouter un nouvel utilisateur sur le système, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
useradd [options] username
```

…où les *options* sont des options de ligne de commande comme décrit dans la Tableau 3.2, « Options de ligne de commande useradd communes ».

Par défaut, la commande **useradd** crée un compte utilisateur verrouillé. Pour déverrouiller le compte, veuillez exécuter la commande suivante en tant qu'utilisateur **root** pour assigner un mot de passe :

```
passwd username
```

Optionnellement, vous pouvez définir une politique d'ancienneté de mot de passe. Veuillez consulter la section Sécurité du mot de passe du *Guide de sécurité Red Hat Enterprise Linux 7*.

Tableau 3.2. Options de ligne de commande useradd communes

Option	Description
-c '*comment*'	*comment* peut être remplacé par n'importe quelle chaîne. Cette option est généralement utilisée pour spécifier le nom complet d'un utilisateur.
-d *home_directory*	Répertoire personnel à utiliser à la place de **/home/*username*/**.
-e *date*	Date à laquelle le compte sera désactivé sous le format YYYY-MM-DD.
-f *days*	Nombre de jours après l'expiration du mot de passe avant que le compte soit désactivé. Si **0** est spécifié, le compte est désactivé immédiatement après l'expiration du mot de passe. Si **-1** est spécifié, le compte n'est pas désactivé après l'expiration du mot de passe.
-g *group_name*	Nom de groupe ou numéro de groupe du groupe (principal) par défaut de l'utilisateur. Le groupe doit exister avant d'être spécifié ici.

Option	Description
-G *group_list*	Liste des noms ou numéros de groupes supplémentaires (autres que ceux par défaut), séparés par des virgules, dont l'utilisateur est membre. Les groupes doivent exister avant d'être spécifiés ici.
-m	Créer le répertoire personnel s'il n'existe pas.
-M	Ne pas créer de répertoire personnel.
-f	Ne pas créer de groupe privé d'utilisateurs pour l'utilisateur.
-p *password*	Mot de passe chiffré avec **crypt**.
-r	Entraîne la création d'un compte système avec un ID utilisateur (UID) inférieur à 1000 et sans répertoire personnel.
-s	Shell de connexion de l'utilisateur, qui est par défaut **/bin/bash**.
-u *uid*	ID utilisateur de l'utilisateur, qui doit être unique et supérieur à 999.

Les options de ligne de commande associées à la commande **usermod** sont essentiellement les mêmes. Remarquez que si vous souhaitez ajouter un utilisateur à un autre groupe supplémentaire, vous devrez utiliser l'option **-a, --append** avec l'option **-G**. Sinon, la liste des groupes supplémentaires de l'utilisateur sera remplacée par ceux spécifiés par la commande **usermod -G**.

IMPORTANT

La gamme des ID par défaut des utilisateurs normaux et système a été modifiée dans Red Hat Enterprise Linux 7 comparé aux versions antérieures. Auparavant, les UID 1 à 499 étaient utilisés pour les utilisateurs système et des valeurs supérieures étaient utilisées pour les utilisateurs normaux. La gamme par défaut des maintenant de 1-999 pour les utilisateurs système. Ce changement peut causer des problèmes lors de la migration vers Red Hat Enterprise Linux 7 avec des utilisateurs existants pouvant avoir des UID et des GID entre 500 et 999. Les gammes par défaut des UID et GID peuvent être modifiées dans le fichier **/etc/login.defs**.

Explication du processus

Les étapes suivantes illustrent ce qu'il se produit si la commande **useradd juan** est exécutée sur un système sur lequel les mots de passe cachés sont activés :

1. Une nouvelle ligne pour **juan** est créée dans **/etc/passwd** :

   ```
   juan:x:1001:1001::/home/juan:/bin/bash
   ```

   La ligne possède les caractéristiques suivantes :

   - Celle-ci commence par le nom d'utilisateur **juan**.

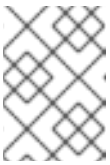
- Un **x** se trouve dans le champ du mot de passe, indiquant que le système utilise des mots de passe cachés.
- Un UID supérieur à 999 est créé. Dans Red Hat Enterprise Linux 7, les UID inférieurs à 1000 sont réservés à l'utilisation système et ne doivent pas être assignés aux utilisateurs.
- Un GID supérieur à 999 est créé. Dans Red Hat Enterprise Linux 7, les GID inférieurs à 1000 sont réservés à l'utilisation système et ne doivent pas être assignés aux utilisateurs.
- Les informations optionnelles *GECOS* sont laissées vides. Le champ GECOS peut être utilisé pour fournir des informations supplémentaires sur l'utilisateur, comme son nom complet ou son numéro de téléphone.
- Le répertoire personnel de **juan** est défini sur **/home/juan/**.
- Le shell par défaut est défini sur **/bin/bash**.

2. Une nouvelle ligne pour **juan** est créée dans **/etc/shadow**:

```
juan:!!:14798:0:99999:7:::
```

La ligne possède les caractéristiques suivantes :

- Celle-ci commence par le nom d'utilisateur **juan**.
- Deux points d'exclamation (**!!**) apparaissent dans le champ du mot de passe du fichier **/etc/shadow**, qui verrouille le compte.



NOTE

Si un mot de passe chiffré est saisi en utilisant l'indicateur **-p**, il sera placé dans le fichier **/etc/shadow** sur la nouvelle ligne pour l'utilisateur.

- Le mot de passe est paramétré de manière à ne jamais expirer.

3. Une nouvelle ligne pour un groupe nommé **juan** est créée dans **/etc/group** :

```
juan:x:1001:
```

Un groupe avec le même nom qu'un utilisateur est appelé un *groupe privé d'utilisateurs*. Pour obtenir davantage d'informations sur les groupes privés d'utilisateurs, veuillez consulter la section Section 3.1.1, « Groupes privés d'utilisateurs ».

La ligne créée dans **/etc/group** possède les caractéristiques suivantes :

- Celle-ci commence par le nom de groupe **juan**.
- Un **x** apparaît dans le champ du mot de passe, indiquant que le système utilise des mots de passe de groupe cachés.
- Le GID correspond à celui qui est répertorié pour le groupe principal de **juan** dans **/etc/passwd**.

4. Une nouvelle ligne pour un groupe nommé **juan** est créée dans **/etc/gshadow** :

```
juan:!::
```

La ligne possède les caractéristiques suivantes :

- Celle-ci commence par le nom de groupe **juan**.
- Un point d'exclamation (**!**) apparaît dans le champ du mot de passe du fichier **/etc/gshadow**, qui verrouille le groupe.
- Tous les autres champs sont vierges.

5. Un répertoire pour l'utilisateur **juan** est créé dans le répertoire **/home** :

```
~]# ls -ld /home/juan
drwx------. 4 juan juan 4096 Mar  3 18:23 /home/juan
```

Ce répertoire appartient à l'utilisateur **juan** et au groupe **juan**. Il possède les privilèges *read* (lecture), *write* (écriture), et *execute* (exécution) *uniquement* pour l'utilisateur **juan**. Toutes les autres permissions sont refusées.

6. Les fichiers dans le répertoire **/etc/skel/** (qui contient les paramètres par défaut de l'utilisateur) sont copiés dans le nouveau répertoire **/home/juan/** :

```
~]# ls -la /home/juan
total 28
drwx------. 4 juan juan 4096 Mar  3 18:23 .
drwxr-xr-x. 5 root root 4096 Mar  3 18:23 ..
-rw-r--r--. 1 juan juan   18 Jun 22  2010 .bash_logout
-rw-r--r--. 1 juan juan  176 Jun 22  2010 .bash_profile
-rw-r--r--. 1 juan juan  124 Jun 22  2010 .bashrc
drwxr-xr-x. 4 juan juan 4096 Nov 23 15:09 .mozilla
```

À ce moment, un compte verrouillé nommé **juan** existe sur le système. Pour l'activer, l'administrateur doit assigner un mot de passe au compte en utilisant la commande **passwd** et optionnellement, paramétrer des directives concernant l'ancienneté du mot de passe (veuillez consulter la section Sécurité du mot de passe dans le *Guide de sécurité Red Hat Enterprise Linux 7* pour obtenir des détails supplémentaires).

3.3.2. Ajout d'un nouveau groupe

Pour ajouter un nouveau groupe au système, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
groupadd [options] group_name
```

…où les *options* sont des options de ligne de commande comme décrit dans la Tableau 3.3, « Options de ligne de commande groupadd communes ».

Tableau 3.3. Options de ligne de commande groupadd communes

Option	Description
-f, **--force**	Lorsqu'utilisé avec **-g** *gid* et que *gid* existe déjà, **groupadd** choisira un nouveau *gid* unique pour le groupe.
-g *gid*	ID de groupe du groupe, qui doit être unique et supérieur à 999.
-K, **--key** *key=value*	Remplace les valeurs par défaut de **/etc/login.defs**.
-o, **--non-unique**	Permet la création de groupes avec un GID dupliqué.
-p, **--password** *password*	Utilise ce mot de passe chiffré pour le nouveau groupe.
-r	Entraîne la création d'un groupe système avec un GID inférieur à 1000.

3.3.3. Création de répertoire de groupes

Les administrateurs système préfèrent habituellement créer un groupe pour chaque projet majeur et assigner des personnes à ce groupe lorsqu'elles ont besoin d'accéder aux fichiers de ce projet. Avec ce schéma traditionnel, la gestion de fichiers est difficile. Lorsque quelqu'un crée un fichier, celui-ci est associé au groupe principal auquel il appartient. Lorsqu'une seule personne travaille sur de multiples projets, il devient difficile d'associer les fichiers corrects au bon groupe. Cependant, avec le schéma UPG, les groupes sont automatiquement assignés aux fichiers créés dans un répertoire sur lequel *setgid* est défini. « Setgid » rend la gestion des projets de groupes qui partagent un répertoire commun très simple car tout fichier créé par un utilisateur dans le répertoire appartiendra au groupe propriétaire de ce répertoire.

Par exemple, un groupe de personnes a besoin de travailler sur des fichiers du répertoire **/opt/myproject/**. Il est fait confiance à certaines personnes pour modifier le contenu de ce répertoire, mais pas à tout le monde.

1. En tant qu'utilisateur **root**, veuillez créer le répertoire **/opt/myproject/** en saisissant ce qui suit dans l'invite de shell :

   ```
   mkdir /opt/myproject
   ```

2. Ajoutez le groupe **myproject** au système :

   ```
   groupadd myproject
   ```

3. Associez le contenu du répertoire **/opt/myproject/** au groupe **myproject** :

   ```
   chown root:myproject /opt/myproject
   ```

4. Autorisez les utilisateurs du groupe à créer des fichiers dans le répertoire et paramétrez le setgid :

   ```
   chmod 2775 /opt/myproject
   ```

   À ce moment, tous les membres du groupe **myproject** peuvent créer et modifier des fichiers

dans le répertoire **/opt/myproject/** sans que l'administrateur ne soit obligé de modifier les permissions de fichier à chaque fois qu'un utilisateur écrit un nouveau fichier. Pour vérifier si les permissions ont été paramétrées correctement, veuillez exécuter la commande suivante :

```
~]# ls -ld /opt/myproject
drwxrwsr-x. 3 root myproject 4096 Mar  3 18:31 /opt/myproject
```

5. Ajoutez les utilisateurs au groupe **myproject** :

```
usermod -aG myproject username
```

3.4. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur la manière de gérer les utilisateurs et les groupes dans Red Hat Enterprise Linux, veuillez consulter les ressources répertoriées ci-dessous.

Documentation installée

Pour obtenir des informations sur les divers utilitaires servant à gérer les utilisateurs et les groupes, veuillez consulter les pages de manuel ci-dessous :

- **useradd**(8) — la page du manuel de la commande **useradd** documente comment l'utiliser pour créer de nouveaux utilisateurs.
- **userdel**(8) — la page du manuel de la commande **userdel** documente comment l'utiliser pour supprimer des utilisateurs.
- **usermod**(8) — la page du manuel de la commande **usermod** documente comment l'utiliser pour modifier les utilisateurs.
- **groupadd**(8) — la page du manuel de la commande **groupadd** documente comment l'utiliser pour créer de nouveaux groupes.
- **groupdel**(8) — la page du manuel de la commande **groupdel** documente comment l'utiliser pour supprimer des groupes.
- **groupmod**(8) — la page du manuel de la commande **groupmod** documente comment l'utiliser pour modifier les appartenances aux groupes.
- **gpasswd**(1) — la page du manuel de la commande **gpasswd** documente comment gérer le fichier **/etc/group**.
- **grpck**(8) — la page du manuel de la commande **grpck** documente comment l'utiliser pour vérifier l'intégrité du fichier **/etc/group**.
- **pwck**(8) — la page du manuel de la commande **pwck** documente comment l'utiliser pour vérifier l'intégrité des fichiers **/etc/passwd** et **/etc/shadow**.
- **pwconv**(8) — la page du manuel des commandes **pwconv**, **pwunconv**, **grpconv**, et **grpunconv** documentent comment convertir des informations cachées pour des mots de passe et des groupes.
- **id**(1) — la page du manuel de la commande **id** documente comment afficher les ID d'utilisateur et de groupe.

Pour obtenir des informations concernant les fichiers de configuration, veuillez consulter :

- **group**(5) — la page du manuel du fichier **/etc/group** documente comment utiliser ce fichier pour définir les groupes de système.
- **passwd**(5) — la page du manuel du fichier **/etc/passwd** documente comment utiliser ce fichier pour définir les informations d'utilisateur.
- **shadow**(5) — la page du manuel du fichier **/etc/shadow** documente comment utiliser ce fichier pour définir les informations d'expiration des mots de passe et des comptes sur ce système.

Documentation en ligne

- Guide de sécurité Red Hat Enterprise Linux 7 — Le *Guide de sécurité* de Red Hat Enterprise Linux 7 fournit des informations supplémentaires sur la manière d'assurer le sécurité du mot de passe et de sécuriser la station de travail en activant l'ancienneté des mots de passe et le verrouillage des comptes utilisateur.

Voir aussi

- Le Chapitre 5, *Obtention de privilèges* documente comment obtenir des privilèges administratifs en utilisant les commandes **su** et **sudo**.

CHAPITRE 4. 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 ne 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 **acl** 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.

4.1. 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 acl *device-name partition*

For example:

mount -t ext3 -o acl /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 **acl** :

```
LABEL=/work      /work       ext3    acl        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-acl-support**. Aucun indicateur particulier n'est requis lors de l'accession ou du montage d'un partage Samba.

4.1.1. 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 les partages NFS lorsque vous configurez le serveur, inclure l'option **no_acl** dans le fichier **/etc/exports**. Pour désactiver les ACL d'un partage NFS quand vous la montez sur un client, montez-la avec l'option **no_acl** via la ligne de commandes ou par le fichier **/etc/fstab**

4.2. 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 4.1. 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 4.2. Supprimer toutes les permissions

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

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

4.3. 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 4.3. 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
```

4.4. 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 4.4. 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
```

4.5. 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, commes 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 4.1, « 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 4.1. Options de ligne de commande pour `star`

Option	Description
-c	Crée un fichier archive.
-n	Ne pas extraire les fichiers. À utiliser en conjonction avec **-x** pour afficher le résultat provoqué par l'extraction de fichiers.
-r	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.
-t	Affiche le contenu du fichier archive.
-u	Met à jour le fichier archive. Les fichiers sont écrits à la fin de l'archive si ceux-ci n'existent pas dans l'archive, ou si les fichiers sont plus récents que les fichiers de même nom dans l'archive. Cette option fonctionne uniquement si l'archive est un fichier ou une bande non-bloquée pouvant être inversé.
-x	Extrait les fichiers de l'archive. Si utilisé avec **-U** et qu'un fichier dans l'archive est plus ancien que le fichier correspondant sur le système de fichers, le fichier ne sera pas extrait.
-help	Affiche les options les plus importantes.

Option	Description
-xhelp	Affiche les options les moins importantes.
-/	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.
-acl	Pendant la création ou l'extraction, archive ou restaure toute ACL associée aux fichiers et répertoires.

4.6. 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.

4.7. 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 5. OBTENTION DE PRIVILÈGES

Les administrateurs système, et dans certains cas les utilisateurs, doivent effectuer certaines tâches avec un accès administratif. L'accès au système en tant qu'utilisateur **root** est potentiellement dangereux et peut causer des dommages qui se répandent sur le système et sur les données. Ce chapitre couvre les manières d'obtenir des privilèges administratifs en utilisant des programmes setuid tels que **su** et **sudo**. Ces programmes autorisent des utilisateurs spécifiques à effectuer des tâches qui seraient normalement uniquement disponibles à l'utilisateur **root** tout en conservant un niveau de contrôle et de sécurité du système élevés.

Veuillez consulter le *Guide de sécurité Red Hat Enterprise Linux 7* pour obtenir davantage d'informations sur les contrôles administratifs, sur les dangers potentiels et sur les manières de prévenir les pertes de données résultant d'une utilisation incorrecte d'un accès privilégié.

5.1. LA COMMANDE SU

Lorsqu'un utilisateur exécute la commande **su**, il lui est demandé le mot de passe de l'utilisateur **root**. Une invite de shell **root** s'ouvrira après l'authentification.

Une fois connecté par le biais de la commande **su**, l'utilisateur **devient** l'utilisateur **root** et possède un accès administratif absolu sur le système. Veuillez remarquer que cet accès reste sujet aux restrictions imposées par SELinux, si activé. En outre, une fois qu'un utilisateur est connecté en tant qu'utilisateur **root**, il est lui est possible d'utiliser la commande **su** pour modifier tout autre utilisateur sur le système sans avoir à saisir de mot de passe.

Comme ce programme est puissant, les administrateurs d'une organisation pourraient souhaiter limiter l'accès à cette commande.

L'une des simple manières de prcéder consiste à ajouter des utilisateurs à un groupe administratif particulier appelé *wheel*. Pour cela, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# usermod -a -G wheel username
```

Dans la commande précédente, veuillez remplacer *username* par le nom d'utilisateur que vous souhaitez ajouter au groupe **wheel**.

Vous pouvez également utiliser l'outil des paramètres **Utilisateurs** pour modifier les appartenances aux groupes comme suit. Veuillez remarquer que vous aurez besoin de privilèges d'administrateur pour effectuer cette procédure.

1. Veuillez appuyer sur la touche **Super** pour accéder à « Vue d'ensembles des activités », saisissez **Utilisateurs** puis appuyez sur **Entrée**. L'outil des paramètres **Utilisateurs** s'affiche. La touche **Super** apparaît sous une variété de formes, selon le clavier et autre matériel, mais le plus souvent, il s'agit de la touche Windows ou Commande, et elle se trouve habituellement à gauche de la **Barre d'espace**.

2. Pour autoriser les modifications, veuillez cliquer sur le bouton **Déverrouiller**, puis saisissez un mot de passe d'administrateur valide.

3. Veuillez cliquer sur l'icône dans la colonne de gauche pour afficher les propriétés de l'utilisateur dans le volet du côté droit.

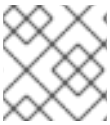
4. Modifiez le **Type de compte** de **Standard** à **Administrateur**. Cela aura pour effet d'ajouter l'utilisateur au groupe **wheel**.

Veuillez consulter la Section 3.2, « Gestion des utilisateurs dans un environnement graphique » pour obtenir davantage d'informations sur l'outil **Utilisateurs**.

Après avoir ajouté les utilisateurs souhaités au groupe **wheel**, il est recommandé d'autoriser ces utilisateurs spécifiques uniquement à utiliser la commande **su**. Pour faire cela, veuillez modifier le fichier de configuration *Pluggable Authentication Module* (PAM) de **su**, **/etc/pam.d/su**. Ouvrez ce fichier dans un éditeur de texte et enlevez la ligne suivante du commentaire en supprimant le caractère # :

```
#auth            required        pam_wheel.so use_uid
```

Ce changement signifie que seuls les membres du groupe administratif **wheel** peuvent basculer sur un autre utilisateur en utilisant la commande **su**.



NOTE

L'utilisateur **root** fait partie du groupe **wheel** par défaut.

5.2. LA COMMANDE SUDO

La commande **sudo** offre une autre approche pour donner un accès administratif aux utilisateurs. Lorsque des utilisateurs de confiance ajoutent **sudo** avant une commande administrative, il leur est demandé *leur propre* mot de passe. Puis, lorsqu'ils ont été authentifié et en supposant que la commande soit autorisée, la commande administrative est exécutée comme s'ils étaient un utilisateur **root**.

Le format de base de la commande **sudo** est comme suit :

```
sudo command
```

Dans l'exemple ci-dessus, *command* sera remplacé par une commande normalement réservée à l'utilisateur **root**, telle que **mount**.

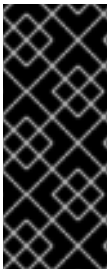
La commande **sudo** offre un haut niveau de flexibilité. Par exemple, seuls les utilisateurs répertoriés dans le fichier de configuration **/etc/sudoers** sont autorisés à utiliser la commande **sudo** et la commande est exécutée dans le shell de l'*utilisateur*, et non dans un shell **root**. Cela signifie que le shell **root** peut être complètement désactivé, comme indiqué dans le *Guide de sécurité Red Hat Enterprise Linux 7*.

Chaque authentification réussie en utilisant la commande **sudo** est journalisée sur le fichier **/var/log/messages** et la commande passée avec le nom d'utilisateur de son émetteur est journalisée sur le fichier **/var/log/secure**. Si davantage de détails sont requis, veuillez utiliser le module **pam_tty_audit** pour activer les audits TTY pour des utilisateurs spécifiques en ajoutant la ligne suivante à votre fichier **/etc/pam.d/system-auth** :

```
session required pam_tty_audit.so disable=pattern enable=pattern
```

où *pattern* représente une liste séparée par des virgules d'utilisateurs avec un usage optionnel de globs. Par exemple, la configuration suivante activera les audits TTY pour l'utilisateur **root** et les désactivera pour tous les autres utilisateurs :

```
session required pam_tty_audit.so disable=* enable=root
```



IMPORTANT

Configurer le module PAM **pam_tty_audit** pour les audits TTY n'enregistre que les entrées TTY. Cela signifie que, lorsque l'utilisateur audité se connecte, **pam_tty_audit** enregistre les saisies de touches précises de l'utilisateur dans le fichier **/var/log/audit/audit.log**. Pour plus d'informations, voir la page man pam_tty_audit(8).

Un autre avantage de la commande **sudo** est qu'un administrateur peut autoriser différents utilisateurs à accéder à des commandes spécifiques en fonction de leurs besoins.

Les administrateurs souhaitant modifier le fichier de configuration **sudo**, **/etc/sudoers**, doivent utiliser la commande **visudo**.

Pour octroire à quelqu'un la totalité des privilèges administratifs, veuillez saisir **visudo** et ajoutez une ligne similaire à la suivante dans la section de spécification des privilèges utilisateur :

```
juan ALL=(ALL) ALL
```

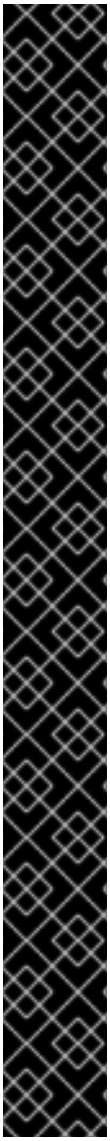
Dans cet exemple, l'utilisateur **juan** peut utiliser **sudo** à partir de n'importe quel hôte et peut exécuter n'importe quelle commande.

L'exemple ci-dessous illustre le niveau de granularité possible lors de la configuration de **sudo** :

```
%users localhost=/usr/sbin/shutdown -h now
```

Cet exemple montre que tout membre du groupe de système **users** peut exécuter la commande **/sbin/shutdown -h now** tant que c'est à partir de la console.

La page man **sudoers** offre une liste détaillée des options pour ce fichier.



IMPORTANT

Il existe plusieurs risques potentiels à ne pas oublier lors de l'utilisation de la commande **sudo**. Vous pouvez les éviter en modifier le fichier de configuration **/etc/sudoers** en utilisant **visudo**, comme indiqué ci-dessus. Laisser le ficher **/etc/sudoers** dans son état par défaut donnera à tout utilisateur du groupe **wheel** un accès **root** illimité.

- Par défaut, **sudo** stocke le mot de passe de l'utilisateur sudo pour une période de cinq minutes. Toute utilisation de la commande pendant cette période ne demandera pas à l'utilisateur de saisir à nouveau le mot de passe. Ceci peut être exploité par une personne malveillante si l'utilisateur laisse son poste de travail connecté, sans surveillance et déverrouillé. Ce comportement peut être modifié en ajoutant la ligne suivante au fichier **/etc/sudoers** :

  ```
  Defaults    timestamp_timeout=value
  ```

  où *value* est la longueur souhaitée du délai en minutes. Définir *value* sur 0 amène **sudo** à réclamer un mot de passe à chaque fois.

- Si le compte sudo d'utilisateur est compromis, une personne malveillante peut utiliser **sudo** pour ouvrir un nouveau shell avec des privilèges administratifs :

  ```
  sudo /bin/bash
  ```

  L'ouverture d'un nouveau shell en tant qu'utilisateur **root** de cette manière, ou d'une manière similaire, peut offrir un accès administratif à une personne malveillante sur une durée théoriquement illimitée, outrepassant ainsi la limite de la durée du délai spécifiée dans le fichier **/etc/sudoers** et il ne sera pas demandé à cette personne malveillante de ressaisir le mot de passe **sudo** tant que la session ouverte n'est pas fermée.

5.3. RESSOURCES SUPPLÉMENTAIRES

Même si les programmes permettant aux utilisateurs d'obtenir des privilèges administratifs présentent des risques de sécurité potentiels, les questions sur la sécurité, en général, est au-delà de l'étendue de ce livre particulier. Veuillez donc consulter les ressources répertoriées ci-dessous pour obtenir davantage d'informations concernant la sécurité et les accès privilégiés.

Documentation installée

- **su**(1) — la page man de **su** offre des informations concernant les options dispionibles avec cette commande.
- **sudo**(8) — la page man de **sudo** inclut une description détaillée de la commande et répertorie les options disponibles pour personnaliser son comportement.
- **pam**(8) — page du manuel décrivant l'utilisation des modules PAM manual (« Pluggable Authentication Modules ») pour Linux.

Documentation en ligne

- Guide de sécurité Red Hat Enterprise Linux 7 — le *Guide de sécurité* de Red Hat Enterprise Linux 7 traite de manière plus approfondie des problèmes de sécurité pertinents aux programmes setuid, ainsi que des techniques utilisées pour réduire ces risques.

Voir aussi

- Le Chapitre 3, *Gérer les utilisateurs et les groupes* documente comment gérer les groupes et utilisateurs système dans l'interface utilisateur graphique et sur la ligne de commande.

PARTIE II. ABONNEMENT ET SUPPORT

Pour recevoir des mises à jour de logiciels sur un système Red Hat Enterprise Linux, celui-ci doit être abonné au réseau de remise de contenu « *Red Hat Content Delivery Network* » (ou CDN) et les référentiels appropriés doivent être activés. Cette partie décrit comment abonner un système au réseau CDN de Red Hat (« Red Hat Content Delivery Network »).

Red Hat fournit un support technique via le Portail Client. Ce support est également accessible directement à partir de la ligne de commande en utilisant l'outil « **Red Hat Support Tool** ». Cette partie décrit l'utilisation de cet outil de ligne de commande.

CHAPITRE 6. ENREGISTRER LE SYSTÈME ET GÉRER LES ABONNEMENTS

Le service d'abonnement fournit un mécanisme pour gérer l'ensemble des logiciels Red Hat et vous permet d'installer des logiciels supplémentaires ou de mettre à jour les programmes déjà installés à des versions plus récentes, en utilisant le gestionnaire de paquets **yum**. Dans Red Hat Enterprise Linux 7, pour enregistrer votre système et attacher des abonnements, il est recommandé d'utiliser *Red Hat Subscription Management*.

NOTE

Il est également possible d'enregistrer le système et d'y ajouter des abonnements après l'installation pendant le processus initial d'installation. Pour plus d'informations sur le processus initial d'installation, consulter le chapitre Installation initiale du Guide d'installation de Red Hat Enterprise Linux 7. Notez que le processus initial d'installation n'apparaît que sur les systèmes possédant une installation Window Xau moment de l'installation.

6.1. ENREGISTRER LE SYSTÈME ET Y AJOUTER DES ABONNNEMENTS

Veuillez effectuer les étapes suivantes pour enregistrer votre système et attacher un ou plusieurs abonnements en utilisant Red Hat Subscription Management. Remarquez que toutes les commandes **`subscription-manager`** sont censées être exécutées en tant qu'utilisateur **`root`**.

1. Exécutez la commande suivante pour enregistrer votre système. Votre nom d'utilisateur et votre mot de passe vous seront demandés. Remarquez que le nom d'utilisateur et le mot de passe sont les mêmes que vos identifiants de connexion pour le Portail Client Red Hat.

   ```
   subscription-manager register
   ```

2. Déterminer l'ID du pool d'un abonnement dont vous avez besoin. Pour ce faire, tapez ce qui suit à une invite du shell pour afficher une liste de tous les abonnements qui sont disponibles pour votre système :

   ```
   subscription-manager list --available
   ```

   Pour chaque abonnement disponible, cette commande affiche son nom, son identifiant unique, sa date d'expiration et autres détails liés à votre abonnement. Pour répertorier les abonnements de toutes les architectures, ajouter l'option **`--all`**. L'ID du pool est répertorié sur une ligne commençant par **`Pool ID`**.

3. Attachez un abonnement qui convient au système en saisissant la commande suivante :

   ```
   subscription-manager attach --pool=pool_id
   ```

   Remplacez *pool_id* par l'ID de pool déterminé dans l'étape précédente.

   Pour vérifier la liste des abonnements actuellement attachés à votre système, exécutez, à tout moment :

   ```
   subscription-manager list --consumed
   ```

Pour obtenir davantage d'informations sur la manière d'enregistrer votre système en utilisant Red Hat Subscription Management et d'y associer des abonnements, veuillez consulter l'article de solution concerné. Pour obtenir des informations sur les abonnements, voir la collection de guides Red Hat Subscription Management.

6.2. GÉRER DES RÉFÉRENTIELS DE LOGICIELS

Lorsqu'un système est abonné à « Red Hat Content Delivery Network » (Réseau de remise de contenu Red Hat), un fichier référentiel est créé dans le répertoire **/etc/yum.repos.d/**. Pour vérifier cela, veuillez utiliser **yum** pour répertorier les référentiels activés :

```
yum repolist
```

Red Hat Subscription Management vous permet également de manuellement activer ou désactiver les référentiels de logiciels fournis par Red Hat. Pour répertorier tous les référentiels disponibles, veuillez utiliser la commande suivante :

```
subscription-manager repos --list
```

Les noms de référentiel dépendent de la version spécifique de Red Hat Enterprise Linux que vous utilisez et sont sous le format suivant :

```
rhel-variant-rhscl-version-rpms
rhel-variant-rhscl-version-debug-rpms
rhel-variant-rhscl-version-source-rpms
```

Quand *variant* est la variante du système Red Hat Enterprise Linux (**server** ou **workstation**), et *version* est la version du système Red Hat Enterprise Linux (**6** ou **7**), par exemple :

```
rhel-server-rhscl-7-eus-rpms
rhel-server-rhscl-7-eus-source-rpms
rhel-server-rhscl-7-eus-debug-rpms
```

Pour activer un référentiel, saisir la commande suivante :

```
subscription-manager repos --enable repository
```

Remplacez *référentiel* par le nom d'un référentiel à activer.

De même, pour désactiver un référentiel, veuillez exécuter la commande suivante :

```
subscription-manager repos --disable repository
```

La Section 8.5, « Configurer Yum et les référentiels Yum » fournit des informations détaillées sur la gestion de référentiels de logiciels en utilisant **yum**.

6.3. SUPPRIMER DES ABONNEMENTS

Pour supprimer un abonnement particulier, veuillez effectuer les étapes suivantes.

1. Déterminez le numéro de série de l'abonnement que vous souhaitez supprimer en répertoriant les informations sur les abonnements déjà attachés :

```
subscription-manager list --consumed
```

Le numér de série est le numéro répertorié en tant que sériel (« **serial** »). Par exemple, **744993814251016831** ci-dessous :

```
SKU:                ES0113909
Contract:           01234567
Account:            1234567
Serial:             744993814251016831
Pool ID:            8a85f9894bba16dc014bccdd905a5e23
Active:             False
Quantity Used:      1
Service Level:      SELF-SUPPORT
Service Type:       L1-L3
Status Details:
Subscription Type:  Standard
Starts:             02/27/2015
Ends:               02/27/2016
System Type:        Virtual
```

2. Saisissez une commande comme suit pour supprimer l'abonnement sélectionné :

```
subscription-manager remove --serial=serial_number
```

Remplacez *serial_number* par le numéro de série déterminé dans l'étape précédente.

Pour supprimer tous les abonnements attachés au système, exécutez la commande suivante :

```
subscription-manager remove --all
```

6.4. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur la manière d'enregistrer votre système en utilisant Red Hat Subscription Management et d'y associer des abonnements, veuillez consulter les ressources ci-dessous.

Documentation installée

- **subscription-manager**(8) — la page du manuel de Red Hat Subscription Management fournit une liste complète des options et commandes prises en charge.

Livres apparentés

- Collection de guides Red Hat Subscription Management — Ces guides contiennent des informations détaillées sur la manière d'utiliser Red Hat Subscription Management.
- Guide d'installation — veuillez consulter le chapitre Installation initiale pour obtenir des informations détaillées sur la manière de s'enregistrer pendant le processus d'installation initiale.

Voir également

- Le Chapitre 5, *Obtention de privilèges* documente la façon d'obtenir des privilèges administratifs en utilisant les commandes **su** et **sudo**.

- Le Chapitre 8, *Yum* fournit des informations sur l'utilisation du gestionnaire de paquets **yum** pour installer et mettre à jour des logiciels.

CHAPITRE 7. ACCÉDER AU SUPPORT EN UTILISANT L'OUTIL « RED HAT SUPPORT TOOL »

L'outil **Red Hat Support Tool**, dans le paquet redhat-support-tool, peut fonctionner comme shell interactif ou comme programme à exécution unique. Il peut être exécuté sur **SSH** ou à partir de n'importe quel terminal. Par exemple, il permet d'effectuer des recherches dans la base de connaissances Red Hat (« Red Hat Knowledgebase ») à partir de la ligne de commande, de copier les solutions directement sur la ligne de commande, de créer et de mettre à jour des dossiers de support, et d'envoyer des fichier à Red Hat pour les analyser.

7.1. INSTALLER L'OUTIL RED HAT SUPPORT TOOL

L'outil **Red Hat Support Tool** est installé par défaut sur Red Hat Enterprise Linux. Si requis, pour s'assurer qu'il soit bien installé, saisissez la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install redhat-support-tool
```

7.2. ENREGISTRER L'OUTIL RED HAT SUPPORT TOOL EN UTILISANT LA LIGNE DE COMMANDE

Pour enregistrer Red Hat Support Tool sur le portail client en utilisant la ligne de commande, veuillez procéder comme suit :

1. ```
   ~]# redhat-support-tool config user nom d'utilisateur
   ```

   Quand *nom d'utilisateur* correspond au nom d'utilisateur du compte sur le Portail Client Red Hat.

2. ```
   ~]# redhat-support-tool config password
   Please enter the password for username:
   ```

7.3. UTILISER RED HAT SUPPORT TOOL EN MODE SHELL INTERACTIF

Pour lancer l'outil en mode interactif, veuillez saisir la commande suivante :

```
~]$ redhat-support-tool
Welcome to the Red Hat Support Tool.
Command (? for help):
```

L'outil peut être exécuté en tant qu'utilisateur non privilégié avec un ensemble de commandes par conséquent réduit, ou en tant qu'utilisateur **root**.

Les commandes peuvent être répertoriées en saisissant le caractère **?**. Le programme ou la sélection du menu peuvent être « quittés » en saisissant le caractère **q** ou **e**. Votre nom d'utilisateur et votre mot de passe du Portail Client Red Hat vous seront demandés lors de votre première recherche sur la base de connaissances ou des dossiers de support. Alternativement, définissez le nom d'utilisateur et le mot de passe de votre compte Portail Client Red Hat en utilisant le mode interactif, et optionnellement, enregistrez-le sur le fichier de configuration.

7.4. CONFIGURER L'OUTIL RED HAT SUPPORT TOOL

Lorsqu'en mode interactif, les options de configuration peuvent être répertoriées en saisissant la commande **`config --help`** :

```
~]# redhat-support-tool
Welcome to the Red Hat Support Tool.
Command (? for help): config --help

Usage: config [options] config.option <new option value>

Use the 'config' command to set or get configuration file values.
Options:
  -h, --help    show this help message and exit
  -g, --global  Save configuration option in /etc/redhat-support-
tool.conf.
  -u, --unset   Unset configuration option.

The configuration file options which can be set are:
 user      : The Red Hat Customer Portal user.
 password  : The Red Hat Customer Portal password.
 debug     : CRITICAL, ERROR, WARNING, INFO, or DEBUG
 url       : The support services URL.
Default=https://api.access.redhat.com
 proxy_url : A proxy server URL.
 proxy_user: A proxy server user.
 proxy_password: A password for the proxy server user.
 ssl_ca    : Path to certificate authorities to trust during
communication.
 kern_debug_dir: Path to the directory where kernel debug symbols should
be downloaded and cached. Default=/var/lib/redhat-support-
tool/debugkernels

Examples:
- config user
- config user my-rhn-username
- config --unset user
```

Procédure 7.1. Enregistrer l'outil Red Hat Support Tool en utilisant le mode interactif

Pour enregistrer Red Hat Support Tool sur le portail client en utilisant le mode interactif, veuillez procéder comme suit :

1. Lancez l'outil en saisissant la commande suivante :

   ```
   ~]# redhat-support-tool
   ```

2. Saisissez votre nom d'utilisateur du Portail Client Red Hat :

   ```
   Command (? for help): config user nom d'utilisateur
   ```

   Pour enregistrer votre nom d'utilisateur dans le fichier de configuration globale , veuillez ajouter l'option **`-g`**.

3. Saisissez votre mot de passe du Portail Client Red Hat :

```
Command (? for help): config password
Please enter the password for username:
```

7.4.1. Enregistrer les paramètres dans les fichiers de configuration

L'outil **Red Hat Support Tool**, sauf contre-indication, stocke les valeurs et options localement dans le répertoire personnel de l'utilisateur actuel, en utilisant le fichier de configuration **~/.redhat-support-tool/redhat-support-tool.conf**. Si requis, il est recommandé d'enregistrer les mots de passe dans ce fichier car il est uniquement lisible par cet utilisateur particulier. Lorsque l'outil est lancé, il lira les valeurs du fichier de configuration globale **/etc/redhat-support-tool.conf** ainsi que celles du fichier de configuration local. Les valeurs et options stockées localement ont priorité sur les paramètres stockés globalement.

AVERTISSEMENT

Il est recommandé de **ne pas** enregistrer de mot de passe dans le fichier de configuration globale **/etc/redhat-support-tool.conf** car le mot de passe est uniquement chiffré **base64** et peut facilement être déchiffré. En outre, le fichier est lisible par tous.

Pour enregistrer une valeur ou une option sur le fichier de configuration globale, veuillez ajouter l'option **-g, --global** comme suit :

```
Command (? for help): config setting -g value
```

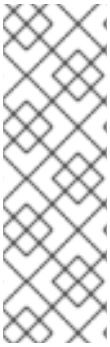
NOTE

Pour être en mesure d'enregistrer les paramètres globalement, en utilisant l'option **-g, --global**, l'outil **Red Hat Support Tool** doit être exécuté en tant qu'utilisateur **root** car les utilisateurs normaux n'ont pas les permissions requises pour écrire sur **/etc/redhat-support-tool.conf**.

Pour supprimer une valeur ou une option du fichier de configuration local, veuillez ajouter l'option **-u, --unset** comme suit :

```
Command (? for help): config setting -u value
```

Cela supprimera et annulera la définition du paramètre de l'outil, et réutilisera le paramètre équivalent situé dans le fichier de configuration globale, si disponible.



NOTE

Lorsqu'exécuté en tant qu'utilisateur non privilégié, les valeurs stockées dans le fichier de configuration globale ne peuvent pas être supprimées en utilisant l'option **`-u, --unset`**, mais elles peuvent être effacées, et annulées de la définition, à partir de l'instance actuellement en cours d'utilisation de l'outil en utilisant l'option **`-g, --global`** simultanément avec l'option **`-u, --unset`**. Si exécuté en tant qu'utilisateur **`root`**, les valeurs et options peuvent être supprimées du fichier de configuration globale en utilisant **`-g, --global`** simultanément avec l'option **`-u, --unset`**.

7.5. CRÉER ET METTRE À JOUR DES DOSSIERS DE SUPPORT EN UTILISANT LE MODE INTERACTIF

Procédure 7.2. Créer un nouveau dossier de support en utilisant le mode interactif

Pour créer un nouveau dossier de support en utilisant le mode interactif, veuillez procéder comme suit :

1. Lancez l'outil en saisissant la commande suivante :

```
~]# redhat-support-tool
```

2. Saisissez la commande **`opencase`** :

```
Command (? for help): opencase
```

3. Suivez les invites affichées sur l'écran pour sélectionner un produit, puis une version.
4. Saisissez un récapitulatif du dossier.
5. Saisissez une description du dossier, puis appuyez sur **`Ctrl`**+**`D`** sur une ligne vide lorsque vous aurez terminé.
6. Sélectionnez une sévérité pour le dossier.
7. Optionnellement, vous pouvez choisir si une solution au problème existe avant de créer un dossier de support.
8. Confirmez que vous souhaitez tout de même créer le dossier de support.

```
Le dossier de support 0123456789 a été créé
```

9. Optionnellement, vous pouvez choisir d'attacher un rapport SOS.
10. Optionnellement, vous pouvez choisir d'attacher un fichier.

Procédure 7.3. Afficher et mettre à jour un dossier de support existant en utilisant le mode interactif

Pour afficher et mettre à jour un dossier de support existant en utilisant le mode interactif, veuillez procéder comme suit :

1. Lancez l'outil en saisissant la commande suivante :

```
~]# redhat-support-tool
```

2. Saisissez la commande **getcase** :

```
Command (? for help): getcase case-number
```

Quand *numéro de cas* correspond au numéro du dossier que vous souhaitez afficher et mettre à jour.

3. Suivez les invites à l'écran pour afficher le dossier, modifier ou ajouter des commentaires, et pour obtenir ou ajouter des pièces jointes.

Procédure 7.4. Modifier un dossier de support existant en utilisant le mode interactif

Pour modifier les attributs d'un dossier de support existant en utilisant le mode interactif, veuillez procéder comme suit :

1. Lancez l'outil en saisissant la commande suivante :

```
~]# redhat-support-tool
```

2. Saisissez la commande **modifycase** :

```
Command (? for help): modifycase numéro de cas
```

Quand *numéro de cas* correspond au numéro du dossier que vous souhaitez afficher et mettre à jour.

3. La liste de modification de sélection s'affiche :

```
Type the number of the attribute to modify or 'e' to return to the
previous menu.
 1 Modify Type
 2 Modify Severity
 3 Modify Status
 4 Modify Alternative-ID
 5 Modify Product
 6 Modify Version
End of options.
```

Suivez les invites affichées sur l'écran pour modifier une ou plusieurs des options.

4. Par exemple, pour modifier le statut, veuillez saisir **3** :

```
Selection: 3
 1   Waiting on Customer
 2   Waiting on Red Hat
 3   Closed
Please select a status (or 'q' to exit):
```

7.6. AFFICHER DES DOSSIERS DE SUPPORT SUR LA LIGNE DE COMMANDE

L'affichage du contenu d'un dossier sur la ligne de commande offre une manière simple et rapide d'appliquer des solutions à partir de la ligne de commande.

Pour afficher un dossier de support existant sur la ligne de commande, veuillez saisir une commande comme suit :

```
~]# redhat-support-tool getcase case-number
```

Avec *numéro de cas* comme numéro du dossier que vous souhaitez télécharger.

7.7. RESSOURCES SUPPLÉMENTAIRES

L'article de la base de connaissances Red Hat Knowledgebase *Red Hat Support Tool* offre des informations supplémentaires, des exemples, et des tutoriels vidéos.

PARTIE III. INSTALLER ET GÉRER UN LOGICIEL

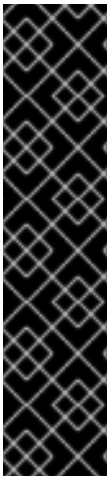
Tous les logiciels sur un système Red Hat Enterprise Linux sont divisés en paquets RPM pouvant être installés, mis à niveau, ou supprimés. Cette partie décrit comment gérer des paquets sur Red Hat Enterprise Linux en utilisant **Yum**.

CHAPITRE 8. YUM

Yum est le gestionnaire de paquets de Red Hat qui peut rechercher des informations sur les paquets disponibles, extraire des paquets de référentiels, les installer et les désinstallez, et mettre à jour un système complet à la dernière version disponible. Yum assure la résolution de dépendances automatiquement lorsque l'on met à jour, installe ou supprime des paquets, et donc, est capable de déterminer, chercher et installer tous les paquets dépendants disponibles automatiquement.

Yum peut être configuré avec des référentiels ou *sources de paquets* supplémentaires, et fournit également de nombreux greffons qui améliorent et étendent ses capacités. Yum est capable d'effectuer de nombreuses tâches que **RPM** peut effectuer. De plus, bon nombre des options de ligne de commande sont similaires. Yum permet une gestion des paquets simple et aisée sur une même machine ou sur un groupe de machines.

Les sections suivantes assument que votre système a été enregistré avec Red Hat Subscription Management au moment de l'intallation selon les instructions qui se trouvent dans Red Hat Enterprise Linux 7 Installation Guide. Si votre système n'est pas enregistré, consulter Chapitre 6, *Enregistrer le système et Gérer les abonnements*.



IMPORTANT

Yum fournit une gestion de paquets sécurisée grâce à l'activation d'un contrôle de signatures GPG (de l'anglais Gnu Privacy Guard ; également connu sous le nom GnuPG) sur tous les référentiels de paquets (sources de paquets), ou sur des référentiels individuels. Lorsque le contrôle des signatures est activé, yum refusera d'installer tous les paquets qui ne sont pas signés-GPG avec la clé appropriée pour ce référentiel. Cela signifie que vous pouvez être rassurés que les paquets **RPM** que vous téléchargez et installez sur votre système sont d'une source fiable, comme Red Hat et qu'ils n'ont pas été modifiés pendant le transfert. Voir Section 8.5, « Configurer Yum et les référentiels Yum » pour plus d'informations sur l'activation du contrôle de signature avec yum, ou Section A.3.2, « Vérification des signatures de paquets » pour plus d'informations sur la façon de travailler et vérifier des paquets GPG **RPM** signés GPG, en général.

Yum vous permet également d'installer aisément vos propres référentiels de paquets **RPM** pour pouvoir les télécharger et les installer sur d'autres machines. Quand c'est possible, yum utilise *le téléchargement parallèle* de plusieurs packages et métadonnées pour accélérer le téléchargement.

Apprendre comment yum opère est un investissement rentable car c'est souvent le moyen le plus rapide pour effectuer des tâches d'administration de système, et il fournit des fonctionnalités qui vont au-delà de celles fournies par les outils graphiques de gestion de paquets de **PackageKit**.

NOTE

Vous devez disposer des privilèges de superutilisateur pour utiliser yum afin d'installer, mettre à jour ou supprimer des paquets sur votre système. Tous les exemples de ce chapitre supposent que vous avez déjà obtenu des privilèges de superutilisateur en utilisant la commande **su** ou **sudo**.

8.1. RECHERCHE ET MISE À JOUR DES PAQUETS

Yum vous permet de vérifier si votre système possède des mises à jour en attente d'être appliquées. Vous pouvez lister les paquets qui ont besoin d'être mis à jour et les mettre à jour ensemble, ou bien, vous pouvez mettre à jour des packages idividuels.

8.1.1. Vérifier les mises à jour

Pour voir quels paquets installés sur votre système disposent de mises à jour disponibles, veuillez utiliser la commande suivante :

```
yum check-update
```

Exemple 8.1. Exemple de sortie de la commande yum check-update :

La sortie de **yum check-update** peut être similaire à la suivante :

```
~]# yum check-update
Loaded plugins: product-id, search-disabled-repos, subscription-manager
dracut.x86_64                           033-360.el7_2      rhel-7-
server-rpms
dracut-config-rescue.x86_64             033-360.el7_2      rhel-7-server-
rpms
kernel.x86_64                           3.10.0-327.el7     rhel-7-
server-rpms
rpm.x86_64                              4.11.3-17.el7      rhel-7-
server-rpms
rpm-libs.x86_64                         4.11.3-17.el7      rhel-7-
server-rpms
rpm-python.x86_64                       4.11.3-17.el7      rhel-7-
server-rpms
yum.noarch                              3.4.3-132.el7      rhel-7-
server-rpms
```

Les paquets dans la sortie ci-dessus sont répertoriées comme ayant des mises à jour disponibles,. Le premier paquet de la liste est **dracut**. Chaque ligne dans l'exemple de sortie consiste en plusieurs lignes, dans le cas de **dracut** :

- **dracut** — nom du paquet
- **x86_64** — architecture du CPU pour laquelle le paquet a été créé,
- **0.5.8** — version du paquet mis à jour à installer,
- **360.el7** — version du paquet mis à jour,
- **_2** — version ajoutée dans le cadre de la mise à jour z-stream,
- **rhel-7-server-rpms** — référentiel dans lequel le paquet mis à jour se trouve.

La sortie montre également que l'on peut mettre à jour le noyau (le paquet kernel), yum et RPM (paquets yum et rpm), ainsi que leurs dépendances (tels que les paquets rpm-libs, et rpm-python), tout en utilisant la commande **yum**.

8.1.2. Mise à jour de paquets

Vous pouvez choisir de mettre à jour un paquet unique, de multiples paquets, ou tous les paquets à la fois. Si l'une des dépendances du ou des paquets que vous mettez à jour possède elle-même des mises à jour disponibles, alors elle sera également mise à jour.

Mise à jour d'un paquet unique

Pour mettre à jour un paquet unique, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
yum update package_name
```

Exemple 8.2. Mise à jour du paquet rpm

Pour mettre à jour le paquet rpm, veuillez saisir :

```
~]# yum update rpm
Loaded plugins: langpacks, product-id, subscription-manager
Updating Red Hat repositories.
INFO:rhsm-app.repolib:repos updated: 0
Setting up Update Process
Resolving Dependencies
--> Running transaction check
---> Package rpm.x86_64 0:4.11.1-3.el7 will be updated
--> Processing Dependency: rpm = 4.11.1-3.el7 for package: rpm-libs-
4.11.1-3.el7.x86_64
--> Processing Dependency: rpm = 4.11.1-3.el7 for package: rpm-python-
4.11.1-3.el7.x86_64
--> Processing Dependency: rpm = 4.11.1-3.el7 for package: rpm-build-
4.11.1-3.el7.x86_64
---> Package rpm.x86_64 0:4.11.2-2.el7 will be an update
--> Running transaction check
...
--> Finished Dependency Resolution

Dependencies Resolved
================================================================================
=====
 Package                   Arch        Version         Repository
Size
================================================================================
=====
Updating:
 rpm                       x86_64      4.11.2-2.el7    rhel
1.1 M
Updating for dependencies:
 rpm-build                 x86_64      4.11.2-2.el7    rhel
139 k
 rpm-build-libs            x86_64      4.11.2-2.el7    rhel
98 k
 rpm-libs                  x86_64      4.11.2-2.el7    rhel
261 k
 rpm-python                x86_64      4.11.2-2.el7    rhel
74 k

Transaction Summary
================================================================================
=====
Upgrade  1 Package (+4 Dependent packages)
```

```
Total size: 1.7 M
Is this ok [y/d/N]:
```

Cette sortie contient plusieurs éléments présentant un interêt particulier :

1. **`Loaded plugins: langpacks, product-id, subscription-manager`** — Yum vous informe toujours quels greffons yum sont installés et activés. Veuillez consulter la Section 8.6, « Greffons Yum » pour des informations générales sur les greffons yum, ou la Section 8.6.3, « Utiliser des greffons Yum » pour des descriptions de greffons particuliers.

2. **`rpm.x86_64`** — il est possible de télécharger et d'installer un nouveau paquet rpm ainsi que ses dépendances. Une vérification de transaction est effectuée pour chacun de ces paquets.

3. Yum présente les informations de mise à jour et vous demande de confirmer la mise à jour ; yum est exécuté interactivement par défaut. Si vous savez déjà quelles transactions la commande **`yum`** planifie d'effectuer, vous pouvez utiliser l'option **`-y`** pour répondre automatiquement oui (« **`yes`** ») à toute question posée par yum (dans ce cas, l'exécution n'est pas interactive). Cependant, vous devriez toujours examiner les changements que yum planifie d'effectuer sur le système afin de pouvoir facilement résoudre tout problème qui se pose. Il est également possible de télécharger le paquet sans l'installer. Pour faire cela, veuillez sélectionner l'option **`d`** dans l'invite du téléchargement. Cela lance le téléchargement en arrière-plan du paquet sélectionné.

   Si une transaction échoue, vous pouvez afficher l'historique des transactions yum en utilisant la commande **`yum history`** comme décrit dans la Section 8.4, « Utiliser l'historique des transactions ».

IMPORTANT

Yum *installe* toujours un nouveau noyau, que vous utilisez la commande **`yum update`** ou **`yum install`**.

D'autre part, lors de l'utilisation de **RPM**, il est important d'utiliser la commande **`rpm -i kernel`** qui installe un nouveau noyau, au lieu de **`rpm -u kernel`** qui *remplace* le noyau actuel. Veuillez consulter la Section A.2.1, « Installation et mise à niveau des paquets » pour obtenir davantage d'informations sur l'installation et la mise à niveau de noyaux avec **RPM**.

Similairement, il est possible de mettre à jour un groupe de paquets. Veuillez saisir en tant qu'utilisateur **`root`** :

```
yum group update group_name
```

Remplacez ici *group_name* par le nom du groupe de paquets que vous souhaitez mettre à jour. Pour obtenir davantage d'informations sur les groupes de paquets, veuillez consulter la Section 8.3, « Utiliser des groupes de paquets ».

Yum offre également la commande **`upgrade`** qui est égale à **`update`** avec une option de configuration **`obsoletes`** (veuillez consulter la Section 8.5.1, « Définir les options [main] »). Par défaut, **`obsoletes`** est activé dans **`/etc/yum.conf`**, ce qui rend ces deux commandes équivalentes.

Mettre à jour tous les paquets et leurs dépendances

Pour mettre à jour tous les paquets et leurs dépendances, veuillez utiliser la commande **yum update** sans aucun argument :

```
yum update
```

Mettre à jour des paquets liés à la sécurité
Si les paquets ont des mises à jour de sécurité, vous ne pourrez les mettre à jour qu'à leur dernière version. Veuillez saisir en tant qu'utilisateur **root** :

```
yum update --security
```

Vous pouvez également mettre ces paquets à jour aux versions contenant les dernières mises à jour en matière de sécurité. Pour cela, veuillez saisir en tant qu'utilisateur **root** :

```
yum update-minimal --security
```

Ainsi, dans la mesure où :

- le paquet kernel-3.10.0-1 est installé sur votre système ;
- le paquet kernel-3.10.0-2 est une mise à jour de sécurité ;
- la paquet kernel-3.10.0-3 est une mise à jour de correctif,

Ensuite, **yum update-minimal --security** mettra à jour le paquet à kernel-3.10.0-2, et **yum update --security** mettra à jour le paquet à kernel-3.10.0-3.

8.1.3. Préserver les changements au fichier de configuration

Vous apporterez inévitablement des modifications aux fichiers de configuration installés par des paquets pendant l'utilisation de votre système Red Hat Enterprise Linux. **RPM**, que yum utilise pour apporter des modifications au système, fournit un mécanisme pour assurer leur intégrité. Veuillez consulter la Section A.2.1, « Installation et mise à niveau des paquets » pour obtenir des détails sur la manière de gérer les changements apportés aux fichiers de configuration pendant les mises à niveau de paquets.

8.1.4. Mise à jour du système hors ligne avec ISO et Yum

Pour les systèmes disconnectés de l'Internet ou de Red Hat Network, utiliser la commande **yum update** avec l'image ISO d'installation de Red Hat Enterprise Linux est un façon simple et rapide de mettre à niveau les systèmes à la dernière version mineure. Les étapes suivantes nous montrent le processus de mise à niveau :

1. Créer un répertoire cible dans lequel monter votre image ISO. Le répertoire n'est pas créé automatiquement lors du montage, donc, il vous faudra le créer avant de procéder à l'étape suivante. En tant qu'utilisateur **root**, saisissez :

   ```
   mkdir mount_dir
   ```

   Remplacer *mount_dir* par un chemin menant au répertoire de montage. Normalement, les utilisateurs le créent en tant que sous-répertoire du répertoire **/media**.

2. Monter l'image ISO d'installation de Red Hat Enterprise Linux 7 dans le répertoire cible préalablement créé. En tant qu'utilisateur **root**, saisir :

```
mount -o loop iso_name mount_dir
```

Remplacez *iso_name* par le nom du chemin de votre image ISO et *mount_dir* par le nom du chemin du répertoire cible. Là, l'option **-o loop** est exigée pour pouvoir monter le fichier en tant que périphérique bloc.

3. Copier le fichier **media.repo** du répertoire de montage **/etc/yum.repos.d/**. Notez que les fichiers de configuration de ce répertoire doivent posséder l'extension *.repo* pour pouvoir fonctionner correctement.

```
cp mount_dir/media.repo /etc/yum.repos.d/new.repo
```

Cela créera un fichier de configuration pour le référentiel yum. Remplacer *new.repo* par le nom du fichier, comme par exemple *rhel7.repo*.

4. Modifiez le nouveau fichier de configuration de façon à ce qu'il puisse pointer vers l'ISO d'installation de Red Hat Enterprise Linux. Ajouter la ligne suivante au fichier **/etc/yum.repos.d/*new.repo*** :

```
baseurl=file:///mount_dir
```

Remplacez *mount_dir* par un chemin qui mène au point de montage.

5. Mettez à jour tous les référentiels yum, y compris **/etc/yum.repos.d/*new.repo*** créé dans les étapes précédentes. En tant qu'utilisateur **root**, saisissez :

```
yum update
```

Cela mettra à jour votre système à la version fournie par l'image ISO montée.

6. Après la mise à niveau, vous pourrez dé-monter l'image ISO. En tant qu'utilisateur **root**, saisissez :

```
umount mount_dir
```

avec *mount_dir* comme chemin qui mène à votre répertoire de montage. Aussi, vous pourrez supprimer le répertoire de montage créé dans la première étape. En tant qu'utilisateur **root**, saisissez :

```
rmdir mount_dir
```

7. Si vous ne souhaitez pas utiliser le fichier de configuration déjà créé pour une autre installation ou mise à jour, vous pouvez le supprimer. En tant qu'utilisateur **root**, saisissez :

```
rm /etc/yum.repos.d/new.repo
```

Exemple 8.3. Mise à niveau de Red Hat Enterprise Linux 7.0 à 7.1

Si vous devez mettre à niveau un système et que vous n'ayiez pas accès à l'internet, en utilisant une image ISO contenant la dernière version du système, appelée, par exemple **rhel-server-7.1-x86_64-dvd.iso**, créer un répertoire cible de montage, comme **/media/rhel7/**. En tant qu'utilisateur **root**, allez dans le répertoire avec votre image ISO et saisissez :

```
~]# mount -o loop rhel-server-7.1-x86_64-dvd.iso /media/rhel7/
```

Puis, créez un référentiel yum pour votre image en copiant le fichier **media.repo** à partir du répertoire de montage :

```
~]# cp /media/rhel7/media.repo /etc/yum.repos.d/rhel7.repo
```

Pour que yum reconnaisse le point de montage comme référentiel, ajouter la ligne suivante au fichier **/etc/yum.repos.d/rhel7.repo** copié dans l'étape précédente :

```
baseurl=file:///media/rhel7/
```

Miantenant, en mettant à jour le référentiel yum, vous mettez ainsi votre système à jour à la version fournie par **rhel-server-7.1-x86_64-dvd.iso**. En tant qu'utilisateur **root**, exécutez :

```
~]# yum update
```

Une fois que votre système sera mis à jour, vous pourrez dé-monter l'image, supprimer le répertoire cible et le fichier de configuration :

```
~]# umount /media/rhel7/
```

```
~]# rmdir /media/rhel7/
```

```
~]# rm /etc/yum.repos.d/rhel7.repo
```

8.2. UTILISER DES PAQUETS

Yum vous permet d'effectuer un ensemble complet d'opérations avec des paquets logiciels, y compris rechercher des paquets, d'afficher leurs informations, des les installer et de les supprimer.

8.2.1. Rechercher des paquets

Vous pouvez rechercher tous les noms, descriptions et résumés des paquets RPM en utilisant la commande suivante :

```
yum search term…
```

Remplacez *term* par le nom du paquet que vous souhaitez rechercher.

Exemple 8.4. Rechercher les paquets correspondants à une chaîne spécifique

Pour répertorier tous les paquets qui correspondent à « vim » ou « gvim », or « emacs », veuillez saisir :

```
~]$ yum search vim gvim emacs
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-manager
============================== N/S matched: vim
==============================
```

```
vim-X11.x86_64 : The VIM version of the vi editor for the X Window
System
vim-common.x86_64 : The common files needed by any version of the VIM
editor[sortie tronquée]

============================ N/S matched: emacs
=============================
emacs.x86_64 : GNU Emacs text editor
emacs-auctex.noarch : Enhanced TeX modes for Emacs[sortie tronquée]

  Name and summary matches mostly, use "search all" for everything.
Warning: No matches found for: gvim
```

La commande **yum search** est utile pour rechercher des paquets dont vous ne connaissez pas le nom, mais dont vous connaissez un terme connexe. Remarquez que par défaut, **yum search** retourne les correspondances par nom de paquet et résumé, ce qui rend les recherches plus rapides. Veuillez utiliser la commande **yum search all** pour effectuer une recherche plus exhaustive mais également plus lente.

Filtrer les résultats

Toutes les commandes de la liste de yum vous permettent de filtrer les résultats en ajoutant une ou plusieurs *expressions glob* comme arguments. Les expressions glob sont des chaînes de caractères normales qui contiennent un ou plusieurs caractères génériques ***** (qui s'étend pour correspondre à tout sous-ensemble de caractères) et **?** (qui s'étend pour correspondre à tout caractère unique).

Prenez soin d'échapper les expressions glob lorsque vous les passez en tant qu'argument sur une commande **yum**, sinon le shell Bash les interprétera comme des *expansions de nom de fichier*, et pourrait potentiellement faire passer tous les fichiers du répertoire actuel correspondant aux expressions globales sur **yum**. Pour vous assurer que les expressions glob soient passées sur **yum** comme souhaité, veuillez utiliser l'une des méthodes suivantes :

- échappez les caractères génériques en les faisant précéder du caractère de la barre oblique inversée
- mettez l'expression glob entre guillemets simples ou entre guillemets doubles.

Les exemples dans la section suivante démontrent l'usage de ces deux méthodes.

8.2.2. Répertorier les paquets

Pour répertorier des informations sur tous les paquets installés *et* disponibles, veuillez saisir ce qui suit dans l'invite shell :

```
yum list all
```

Pour répertorier les paquets installés *et* disponibles qui correspondent aux expressions glob insérées, veuillez utiliser la commande suivante :

```
yum list glob_expression…
```

Exemple 8.5. Répertorier les paquets liés à ABRT

Les paquets avec divers modules complémentaires et greffons ABRT commencent par « abrt-

addon- » ou par « abrt-plugin- ». Pour répertorier ces paquets, veuillez saisir la commande suivante à l'invite shell. Remarquez que les caractères génériques sont échappés avec le caractère de la barre oblique inversée :

```
~]$ yum list abrt-addon\* abrt-plugin\*
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-manager
Installed Packages
abrt-addon-ccpp.x86_64                  2.1.11-35.el7
@rhel-7-server-rpms
abrt-addon-kerneloops.x86_64            2.1.11-35.el7
@rhel-7-server-rpms
abrt-addon-pstoreoops.x86_64            2.1.11-35.el7
@rhel-7-server-rpms
abrt-addon-python.x86_64                2.1.11-35.el7
@rhel-7-server-rpms
abrt-addon-vmcore.x86_64                2.1.11-35.el7
@rhel-7-server-rpms
abrt-addon-xorg.x86_64                  2.1.11-35.el7
@rhel-7-server-rpms
```

Pour répertorier tous les paquets installés sur votre système, veuillez utiliser le mot-clé **`installed`**. La colonne la plus à droite dans la sortie répertorie le référentiel à partir duquel le paquet a été récupéré.

`yum list installed` *`glob_expression`*…

Exemple 8.6. Répertorier tous les versions installées du paquet krb

L'exemple suivant montre comment répertorier tous les paquets installés qui commencent par « krb » suivis d'exactement un caractère et d'un tiret. Ceci est utile lorsque vous souhaitez répertorier toutes les versions d'un certain composant, car ceux-ci se distinguent par leur numéro. L'expression glob toute entière est citée pour assurer que le traitement soit correct.

```
~]$ yum list installed "krb?-*"
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-manager
Installed Packages
krb5-libs.x86_64                 1.13.2-10.el7
@rhel-7-server-rpms
```

Pour répertorier tous les paquets dans des référentiels activés qui seraient disponibles à installer, veuillez utiliser la commande sous la forme suivante :

`yum list available` *`glob_expression`*…

Exemple 8.7. Répertorier les greffons gstreamer disponibles

Par exemple, pour répertorier tous les paquets disponibless avec des noms qui contiennent « gstreamer » puis « plugin », veuillez exécuter la commande suivante :

```
~]$ yum list available gstreamer\*plugin\*
```

```
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-manager
Available Packages
gstreamer-plugins-bad-free.i686             0.10.23-20.el7
rhel-7-server-rpms
gstreamer-plugins-base.i686                 0.10.36-10.el7
rhel-7-server-rpms
gstreamer-plugins-good.i686                 0.10.31-11.el7
rhel-7-server-rpms
gstreamer1-plugins-bad-free.i686            1.4.5-3.el7
rhel-7-server-rpms
gstreamer1-plugins-base.i686                1.4.5-2.el7
rhel-7-server-rpms
gstreamer1-plugins-base-devel.i686          1.4.5-2.el7
rhel-7-server-rpms
gstreamer1-plugins-base-devel.x86_64        1.4.5-2.el7
rhel-7-server-rpms
gstreamer1-plugins-good.i686                1.4.5-2.el7
rhel-7-server-rpms
```

Répertorier les référentiels

Pour répertorier l'ID du référentiel, le nom, et le numéro des paquets de chaque référentiel *activé* sur votre système, veuillez utiliser la commande suivante :

```
yum repolist
```

Pour répertorier davantage d'informations sur ces référentiels, veuillez ajouter l'option **-v**. Avec cette option activée, les informations comprenant le nom du fichier, la taille générale, la date de la dernière mise à jour, et l'URL de base sont affichées pour chaque référentiel répertorié. Alternativement, vous pouvez utiliser la commande **repoinfo** qui produira la même sortie.

```
yum repolist -v
```

```
yum repoinfo
```

Pour répertorier les référentiels activés et désactivés, veuillez utiliser la commande suivante. Une colonne de statut est ajoutée à la liste de sorties pour afficher quels sont les référentiels activés.

```
yum repolist all
```

En passant **disabled** en tant que premier argument, vous pouvez limiter la sortie de la commande aux référentiels désactivés. Pour des spécifications plus précises, vous pouvez passer l'ID ou le nom des référentiels ou des glob_expressions comme arguments. Remarquez que s'il y a une correspondance exacte entre l'ID ou le nom du référentiel et l'argument inséré, ce référentiel est répertorié, même s'il ne passe pas le filtre *enabled* ou *disabled*.

8.2.3. Afficher des informations sur le paquet

Pour afficher des informations sur un ou plusieurs paquets, veuillez utiliser la commande suivante (ici, les expressions glob sont également valides) :

```
yum info package_name…
```

Remplacez *package_name* par le nom du paquet.

Exemple 8.8. Afficher des informations sur le paquet abrt

Pour afficher des informations sur le paquet abrt, veuillez saisir :

```
~]$ yum info abrt
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-manager
Installed Packages
Name        : abrt
Arch        : x86_64
Version     : 2.1.11
Release     : 35.el7
Size        : 2.3 M
Repo        : installed
From repo   : rhel-7-server-rpms
Summary     : Automatic bug detection and reporting tool
URL         : https://fedorahosted.org/abrt/
License     : GPLv2+
Description : abrt is a tool to help users to detect defects in
applications and
            : to create a bug report with all information needed by
maintainer to fix
            : it. It uses plugin system to extend its functionality.
```

La commande **yum info *package_name*** est similaire à la commande **rpm -q --info *package_name***, mais elle fournit des informations supplémentaires, comme le nom du référentiel yum dans lequel le paquet RPM était installé (consultez la ligne **From repo:** dans la sortie).

Utiliser yumdb

Vous pouvez également effectuer des requêtes sur la base de données yum pour trouver d'autres informations utiles sur un paquet en utilisant la commande suivante :

```
yumdb info package_name
```

Cette commande fournit des informations supplémentaires sur un paquet, y compris le checksum du paquet (et l'algorithme utilisé pour le produire, comme SHA-256), la commande donnée sur la ligne de commande qui a été invoquée pour installer le paquet (s'il y en a une), et la raison pour laquelle le paquet est installé sur le système (où **user** indique une installation par l'utilisateur, et **dep** indique une installation pour raison de dépendance).

Exemple 8.9. Exécuter une requête yumdb pour trouver des informations sur le paquet yum

Pour afficher des informations supplémentaires sur le paquet yum, veuillez saisir :

```
~]$ yumdb info yum
Loaded plugins: langpacks, product-id
yum-3.4.3-132.el7.noarch
     changed_by = 1000
     checksum_data =
a9d0510e2ff0d04d04476c693c0313a11379053928efd29561f9a837b3d9eb02
     checksum_type = sha256
```

```
      command_line = upgrade
      from_repo = rhel-7-server-rpms
      from_repo_revision = 1449144806
      from_repo_timestamp = 1449144805
      installed_by = 4294967295
      origin_url =
https://cdn.redhat.com/content/dist/rhel/server/7/7Server/x86_64/os/Pack
ages/yum-3.4.3-132.el7.noarch.rpm
      reason = user
      releasever = 7Server
      var_uuid = 147a7d49-b60a-429f-8d8f-3edb6ce6f4a1
```

Pour obtenir davantage d'informations sur la commande **yumdb**, veuillez consulter la page man de **yumdb**(8).

8.2.4. Installation de paquets

Pour installer un paquet unique et toutes ses dépendances non installées, veuillez saisir une commande sous la forme suivante en tant qu'utilisateur **root** :

```
yum install package_name
```

Vous pouvez également installer de multiples paquets simultanément en ajoutant leurs noms en tant qu'arguments. Pour faire cela, veuillez saisir en tant qu'utilisateur **root** :

```
yum install package_name package_name…
```

Si vous installez des paquets sur un système *multilib*, tel que sur un ordinateur AMD64 ou Intel64, vous pourrez spécifier l'architecture du paquet (tant qu'il est disponible dans un référérentiel activé) en ajoutant *.arch* au nom du paquet :

```
yum install package_name.arch
```

Exemple 8.10. Installer des paquets sur un système multilib

Pour installer le paquet sqlite pour l'architecture **i686**, veuillez saisir :

```
~]# yum install sqlite.i686
```

Vous pouvez utiliser des expressions glob pour installer rapidement de multiples paquets avec des noms similaires. En tant qu'utilisateur **root**, veuillez exécuter :

```
yum install glob_expression…
```

Exemple 8.11. Installer tous les greffons audacieux

Les expressions globales sont utiles lorsque vous souhaitez installer plusieurs paquets avec des noms similaires. Pour installer tous les greffons audacieux, veuillez utiliser la commande sous la forme suivante :

```
~]# yum install audacious-plugins-\*
```

En plus des noms de paquets et des expressions glob, vous pouvez également fournir des noms de fichier à **yum install**. Si vous connaissez le nom du binaire que vous souhaitez installer, mais pas son nom de paquet, vous pouvez donner le nom du chemin à **yum install**. En tant qu'utilisateur **root**, veuillez saisir :

```
yum install /usr/sbin/named
```

Yum effectue la recherche dans ses listes de paquets, trouve le paquet qui fournit **/usr/sbin/named**, s'il existe, et vous demande si vous souhaitez l'installer.

Comme vous pouvez le voir dans les exemples ci-dessus, la commande **yum install** ne requiert pas d'arguments strictement définis. Il peut traiter divers formats de noms de paquets et d'expressions glob, ce qui rend l'installation plus facile pour les utilisateurs. D'autre part, **yum** peut prendre longtemps à analyser l'entrée correctement, particulièrement si vous spécifiez un grand nombre de paquets. Pour optimiser la recherche de paquets, vous pouvez utiliser les commandes suivantes pour définir de manière explicite comment analyser les arguments :

yum install-n *name*

yum install-na *name.architecture*

yum install-nevra *name-epoch:version-release.architecture*

Avec **install-n**, **yum** interprète *name* comme étant le nom exact du paquet. La commande **install-na** indique à **yum** que l'argument suivant contient le nom du paquet et l'architecture divisés par le caractère « point ». Avec **install-nevra**, **yum** s'attendra à un argument sous la forme *name-epoch:version-release.architecture*. Similairement, vous pouvez utiliser **yum remove-n**, **yum remove-na**, et **yum remove-nevra** lorsque vous cherchez des paquets à supprimer.

NOTE

Si vous êtes sûr(e) de souhaiter installer le paquet contenant le binaire **named**, mais que vous ne savez pas dans quel répertoire **bin/** ou **sbin/** le fichier est installé, veuillez utiliser la commande **yum provides** avec une expression glob :

```
~]# yum provides "*bin/named"
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-
              : manager
32:bind-9.9.4-14.el7.x86_64 : The Berkeley Internet Name Domain
(BIND) DNS
                            : (Domain Name System) server
Repo        : rhel-7-server-rpms
Matched from:
Filename    : /usr/sbin/named
```

yum provides "*/*file_name*" est une astuce utile pour trouver le paquet ou les paquets qui contiennent le *file_name*.

Exemple 8.12. Processus d'installation

L'exemple suivant offre une vue d'ensemble d'une installation en utilisant **yum**. Pour télécharger et installer la dernière version du paquet httpd, veuillez exécuter en tant qu'utilisateur **root** :

```
~]# yum install httpd
Loaded plugins: langpacks, product-id, subscription-manager
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.6-12.el7 will be updated
---> Package httpd.x86_64 0:2.4.6-13.el7 will be an update
--> Processing Dependency: 2.4.6-13.el7 for package: httpd-2.4.6-
13.el7.x86_64
--> Running transaction check
---> Package httpd-tools.x86_64 0:2.4.6-12.el7 will be updated
---> Package httpd-tools.x86_64 0:2.4.6-13.el7 will be an update
--> Finished Dependency Resolution

Dependencies Resolved
```

Après avoir exécuté la commande ci-dessus, **yum** charge les greffons nécessaires et exécute la vérification de transaction. Dans ce cas, httpd est déjà installé. Comme le paquet installé est plus ancien que la version disponible la plus récente, il sera mis à jour. La même chose s'applique au paquet httpd-tools dont httpd dépend. Puis, un résumé de la transaction est affiché :

```
================================================================================
========
 Package        Arch      Version                 Repository
Size
================================================================================
========
Updating:
 httpd          x86_64    2.4.6-13.el7            rhel-x86_64-server-7
1.2 M
Updating for dependencies:
 httpd-tools    x86_64    2.4.6-13.el7            rhel-x86_64-server-7
77 k

Transaction Summary
================================================================================
========
Upgrade  1 Package (+1 Dependent package)

Total size: 1.2 M
Is this ok [y/d/N]:
```

Dans cette étape, **yum** vous demande de confirmer l'installation. Hormis les options **y** (oui) et **N** (non), vous pouvez également choisir **d** (télécharger uniquement) pour télécharger les paquets sans les installer directement. Si vous choisissez **y**, l'installation continuera avec les messages suivants jusqu'à ce qu'elle se termine.

```
Downloading packages:
Running transaction check
Running transaction test
```

```
Transaction test succeeded
Running transaction
  Updating   : httpd-tools-2.4.6-13.el7.x86_64
1/4
  Updating   : httpd-2.4.6-13.el7.x86_64
2/4
  Cleanup    : httpd-2.4.6-12.el7.x86_64
3/4
  Cleanup    : httpd-tools-2.4.6-12.el7.x86_64
4/4
  Verifying  : httpd-2.4.6-13.el7.x86_64
1/4
  Verifying  : httpd-tools-2.4.6-13.el7.x86_64
2/4
  Verifying  : httpd-tools-2.4.6-12.el7.x86_64
3/4
  Verifying  : httpd-2.4.6-12.el7.x86_64
4/4

Updated:
  httpd.x86_64 0:2.4.6-13.el7

Dependency Updated:
  httpd-tools.x86_64 0:2.4.6-13.el7

Complete!
```

Pour installer un paquet téléchargé au préalable se trouvant dans un répertoire local sur votre système, veuillez utiliser la commande suivante :

```
yum localinstall path
```

Remplacez *path* par le chemin du paquet que vous souhaitez installer.

8.2.5. Télécharger des paquets

Comme indiqué dans l'Exemple 8.12, « Processus d'installation », à un certain moment pendant le processus d'installation, il vous sera demandé de confirmer l'installation avec le message suivant :

```
...
Total size: 1.2 M
Is this ok [y/d/N]:
...
```

With the **d** option, yum downloads the packages without installing them immediately. You can install these packages later offline with the **yum localinstall** command or you can share them with a different device. Downloaded packages are saved in one of the subdirectories of the cache directory, by default **/var/cache/yum/$basearch/$releasever/packages/**. The downloading proceeds in background mode so that you can use **yum** for other operations in parallel.

8.2.6. Supprimer des paquets

De même que pour l'installation de paquets, yum permet également de les désinstaller. Pour désinstaller un paquet particulier, ainsi que tout paquet en dépendant, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
yum remove package_name…
```

Comme lors de l'installation de plusieurs paquets, il est possible d'en supprimer plusieurs à la fois en ajoutant des noms de paquets supplémentaires à la commande.

Exemple 8.13. Supprimer plusieurs paquets

Pour supprimer totem, veuillez saisir ce qui suit à l'invite shell :

```
~]# yum remove totem
```

Similairememt à **install**, **remove** peut prendre ces arguments :

- noms de paquet
- expressions glob
- listes de fichiers
- ce que le paquet fournit

AVERTISSEMENT

Yum n'est pas en mesure de supprimer un paquet sans également supprimer les paquets qui en dépendent. Ce type d'opération, qui peut uniquement être effectué par **RPM**, n'est pas publicisé et peut potentiellement laisser votre système dans un état ne fonctionnant pas, ou causer à des applications de ne pas fonctionner correctement ou de tomber en panne. Pour obtenir des informations supplémentaires, veuillez consulter la Section A.2.2, « Désinstaller les paquets » dans le chapitre **RPM**.

8.3. UTILISER DES GROUPES DE PAQUETS

Un groupe de paquets est une collection de paquets qui servent un but commun, par exemple *System Tools* (Outils système) ou *Sound and Video* (Son et vidéo). Installer un groupe de paquets inclut également un ensemble de paquets dépendants, ce qui permet de faire des économies de temps considérables. La commande **yum groups** est une commande de haut niveau qui couvre toutes les opérations agissant sur les groupes de paquets dans yum.

8.3.1. Répertorier les groupes de paquets

L'option **summary** (résumé) est utilisée pour afficher le nombre de groupes installés, les groupes disponibles, les groupes d'environnement disponibles, ainsi que les deux groupes de langues disponibles et installés :

```
yum groups summary
```

Exemple 8.14. Exemple de sortie de résumé des groupes yum

```
~]$ yum groups summary
Loaded plugins: langpacks, product-id, subscription-manager
Available Environment Groups: 12
Installed Groups: 10
Available Groups: 12
```

Pour répertorier tous les groupes de paquets des référentiels yum, veuillez ajouter l'option **list**. Vous pouvez également filtrer la sortie de la commande par nom de groupe.

```
yum group list glob_expression…
```

Plusieurs arguments optionnels peuvent être passés sur cette commande, y compris **hidden** pour répertorier les groupes qui ne sont pas marqués comme étant visibles par les utilisateurs, et **ids** pour répertorier les ID de groupe. Vous pouvez ajouter les options **language**, **environment**, **installed**, ou **available** pour réduire la sortie de la commande à un type de groupe spécifique.

Pour répertorier les paquets obligatoires et optionnels contenus dans un groupe particulier, veuillez utiliser la commande suivante :

```
yum group info glob_expression…
```

Exemple 8.15. Afficher des informations sur le groupe de paquets LibreOffice

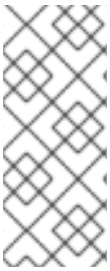
```
 ~]$ yum group info LibreOffice
Loaded plugins: langpacks, product-id, subscription-manager

Group: LibreOffice
 Group-Id: libreoffice
 Description: LibreOffice Productivity Suite
 Mandatory Packages:
  =libreoffice-calc
   libreoffice-draw
  -libreoffice-emailmerge
   libreoffice-graphicfilter
  =libreoffice-impress
  =libreoffice-math
  =libreoffice-writer
  +libreoffice-xsltfilter
 Optional Packages:
   libreoffice-base
   libreoffice-pyuno
```

Comme indiqué dans l'exemple ci-dessus, les paquets inclus dans le groupe de paquets peuvent avoir différents états marqués à l'aide des symboles suivants :

- « **-** » — le paquet n'est pas installé et ne sera pas installé comme s'il faisait partie du groupe de paquets.
- « **+** » — le paquet n'est pas installé mais sera installé lors de la prochaine mise à niveau **yum upgrade** ou **yum group upgrade**.
- « **=** » — le paquet est installé et a été installé en faisant partie du groupe de paquets.
- « *no symbol* » — le paquet est installé mais il a été installé hors du groupe de paquets. Cela signifie que **yum group remove** ne supprimera pas ce paquet.

Ces distinctions se produisent uniquement lorsque le paramètre de configuration **group_command** est défini sur **objects**, qui est le paramètre par défaut. Définissez ce paramètre sur une autre valeur si vous ne souhaitez pas que yum vérifie si un paquet a été installé en tant que faisant partie d'un groupe ou séparément, ce qui rendra les paquets « *no symbol* » équivalents aux paquets « = ».

Vous pouvez altérer les états des paquets ci-dessus en utilisant la commande **yum group mark**. Par exemple, **yum group mark packages** marque les paquets installés donnés en tant que membre d'un groupe spécifié. Pour éviter l'installation de nouveaux paquets sur une mise à jour de groupe, veuillez utiliser **yum group mark blacklist**. Consultez la page man de **yum**(8) pour obtenir davantage d'informations sur les capacités de **yum group mark**.



NOTE

Vous pouvez identifier un groupe environnemental en utilisant le préfixe *@^* et un groupe de paquets peut être précédé de *@*. Lors de l'utilisation de **yum group list**, **info**, **install**, ou **remove**, veuillez inclure *@group_name* pour spécifier un groupe de paquets, *@^group_name* pour spécifier un groupe environnemental, ou *group_name* pour inclure les deux.

8.3.2. Installer un groupe de paquets

Chaque groupe de paquets possède un nom et un ID de groupe (*groupid*). Pour répertorier les noms de tous les groupes de paquets et leurs ID de groupe, qui sont affichés entre parenthèses, veuillez saisir :

```
yum group list ids
```

Exemple 8.16. Trouver le nom et l'ID de groupe d'un groupe de paquets

Pour trouver le nom ou l'ID d'un groupe de paquets, par exemple un groupe lié à l'environnement du bureau KDE, veuillez saisir :

```
~]$ yum group list ids kde\*
Available environment groups:
   KDE Plasma Workspaces (kde-desktop-environment)
Done
```

Certains groupes sont cachés par des paramètres dans les référentiels configurés. Par exemple, sur un serveur, vous pouvez utiliser l'option de commande **hidden** pour répertorier les groupes cachés également :

```
~]$ yum group list hidden ids kde\*
Loaded plugins: product-id, subscription-manager
```

```
Available Groups:
   KDE (kde-desktop)
Done
```

Vous pouvez installer un groupe de paquets en passant son nom de groupe complet, sans la partie « groupid », à la commande **group install**. En tant qu'utilisateur **root**, veuillez saisir :

```
yum group install "group name"
```

Vous pouvez également effectuer des installations par « groupid ». En tant qu'utilisateur **root**, veuillez exécuter la commande suivante :

```
yum group install groupid
```

Vous pouvez inclure le « groupid » ou le nom de groupe entre guillemets dans la commande **install** si vous ajoutez également le symbole @ comme préfixe, ce qui indique à **yum** que vous souhaitez effectuer une installation de type **group install**. En tant qu'utilisateur **root**, veuillez saisir :

```
yum install @group
```

Remplacez *group* par le « groupid » ou le nom de groupe entre guillements. La même logique s'applique aux groupes environnementaux :

```
yum install @^group
```

Exemple 8.17. Quatre manières équivalentes d'installer le groupe KDE Desktop

Comme mentionné précédemment, il existe quatre alternatives équivalentes pour installer un groupe de paquets. Pour le groupe KDE Desktop, les commandes sont comme suit :

```
~]# yum group install "KDE Desktop"
~]# yum group install kde-desktop
~]# yum install @"KDE Desktop"
~]# yum install @kde-desktop
```

8.3.3. Supprimer un groupe de paquets

Vous pouvez supprimer un groupe de paquets en utilisant une syntaxe similaire à la syntaxe **install**, en utilisant le nom du groupe de paquets ou son ID. En tant qu'utilisateur **root**, veuillez saisir :

```
yum group remove group_name
```

```
yum group remove groupid
```

Vous pouvez également inclure le « groupid » ou le nom entre guillemets dans la commande **remove** si vous ajoutez également le symbole @ comme préfixe, ce qui indique à yum que vous souhaitez effectuer une suppression de type **group remove**. En tant qu'utilisateur **root**, veuillez saisir :

```
yum remove @group
```

Remplacez *group* par le « groupid » ou le nom de groupe entre guillements. Vous pouvez remplacer un groupe environnement de manière similaire :

```
yum remove @^group
```

Exemple 8.18. Quatre manières équivalentes de supprimer le groupe `KDE Desktop`.

De même que pour l'installation, il existe quatre alternatives équivalentes pour supprimer un groupe de paquets. Pour le groupe KDE Desktop, les commandes sont comme suit :

```
~]# yum group remove "KDE Desktop"
~]# yum group remove kde-desktop
~]# yum remove @"KDE Desktop"
~]# yum remove @kde-desktop
```

8.4. UTILISER L'HISTORIQUE DES TRANSACTIONS

La commande **`yum history`** permet aux utilisateurs d'examiner des informations sur la chronologie des transactions yum, les dates et heures auxquelles elles se sont produites, le nombre de paquets affectés, si ces transactions ont réussi ou échoué, et si la base de données RPM a été modifiées entre les transactions. En outre, cette commande peut être utilisée pour annuler ou refaire certaines transactions. Tout l'historique des données est stocké dans la base de données de l'historique (*history DB*) dans le répertoire **`/var/lib/yum/history/`**.

8.4.1. Répertorier les transactions

Pour afficher une liste des vingt transactions les plus récentes, en tant qu'utilisateur **`root`**, veuillez exécuter **`yum history`** sans argument supplémentaire, ou saisissez ce qui suit dans une invite shell :

```
yum history list
```

Pour afficher toutes les transactions, veuillez ajouter le mot-clé **`all`** :

```
yum history list all
```

Pour uniquement afficher des transactions pendant une période donnée, veuillez utiliser la commande sous le format suivant :

```
yum history list start_id..end_id
```

Vous pouvez également répertorier les transactions qui concernent un ou plusieurs paquets particuliers. Pour faire cela, veuillez utiliser la commande avec un nom de paquet ou une expression glob :

```
yum history list glob_expression…
```

Exemple 8.19. Répertorier les cinq transactions les plus anciennes

Dans la sortie de **yum history list**, la transaction la plus récente est affichée en haut de la liste. Pour afficher des informations sur les cinq plus anciennes transactions stockées dans la base de données de l'historique, veuillez saisir :

```
~]# yum history list 1..5
Loaded plugins: langpacks, product-id, subscription-manager
ID     | Login user               | Date and time    | Action(s)
| Altered
-------------------------------------------------------------------------
-----------
     5 | User <user>              | 2013-07-29 15:33 | Install
|    1
     4 | User <user>              | 2013-07-21 15:10 | Install
|    1
     3 | User <user>              | 2013-07-16 15:27 | I, U
|   73
     2 | System <unset>           | 2013-07-16 15:19 | Update         |
1
     1 | System <unset>           | 2013-07-16 14:38 | Install        |
1106
history list
```

Toutes les formes de la commande **yum history list** produisent une sortie tabulaire dont chaque ligne comporte les colonnes suivantes :

- **ID** — valeur d'entier identifiant une transaction particulière.
- **Login user** — nom de l'utilisateur dont la session de connexion a été utilisée pour initier une transaction. Cette information est typiquement présentée sous la forme ***Full Name <username>***. Pour les transactions qui n'ont pas été effectuées par un utilisateur (comme les mises à jour automatiques du système), **System <unset>** est utilisé à la place.
- **Date and time** — la date et l'heure à laquelle une transaction a été effectuée.
- **Action(s)** — liste d'actions effectuées au cours d'une transaction, comme décrit dans Tableau 8.1, « Valeurs possibles du champ « Action(s) » ».
- **Altered** — nombre de paquets qui ont été affectés par une transaction, probablement suivis d'informations supplémentaires comme décrit dans Tableau 8.2, « Les valeurs possibles du champ « Altered » ».

Tableau 8.1. Valeurs possibles du champ « Action(s) »

Action	Abbréviation	Description
Downgrade	**D**	Un paquet au moins a été mis à niveau à une version antérieure.
Erase	**E**	Un paquet au moins a été supprimé.
Install	**I**	Un nouveau paquet au moins a été installé.

Action	Abbréviation	Description
Obsoleting	**O**	Un paquet au mons a été marqué comme obsolète.
Reinstall	**R**	Un paquet au moins a été réinstallé.
Update	**U**	Un paquet au moins a été mis à jour à une version plus récente.

Tableau 8.2. Les valeurs possibles du champ « Altered »

Symbole	Description
<	Avant que la transaction se termine, la base de données **rpmdb** a été modifiée hors de yum.
>	Une fois la transaction terminée, la base de données **rpmdb** a été modifiée hors de yum.
*****	La transaction ne s'est pas terminée correctement.
#	La transaction s'est terminée correctement, mais yum a retourné un code de sortie différent de zéro.
E	La transaction s'est terminée correctement, mais une erreur ou un avertissement s'est affiché.
P	La transaction s'est terminée correctement, mais des problèmes existaient déjà dans la base de données **rpmdb**.
s	La transaction s'est terminée correctement, mais l'option de ligne de commande **--skip-broken** a été utilisée et certains paquets ont été ignorés.

Pour synchroniser le contenu de la base de données **rpmdb** ou **yumdb** pour tout paquet installé avec la base de données **rpmdb** ou **yumdb** actuellement utilisée, veuillez saisir ce qui suit :

```
yum history sync
```

Pour afficher certaines statistiques générales sur la base de données de l'historique actuellement utilisée, veuillez utiliser le format suivant :

```
yum history stats
```

Exemple 8.20. Exemple de sortie de yum history stats

```
~]# yum history stats
Loaded plugins: langpacks, product-id, subscription-manager
File        : //var/lib/yum/history/history-2012-08-15.sqlite
```

```
Size        : 2,766,848
Transactions: 41
Begin time  : Wed Aug 15 16:18:25 2012
End time    : Wed Feb 27 14:52:30 2013
Counts      :
  NEVRAC :  2,204
  NEVRA  :  2,204
  NA     :  1,759
  NEVR   :  2,204
  rpm DB :  2,204
  yum DB :  2,204
history stats
```

Yum permet également d'afficher un résumé de toutes les ancennes transactions. Pour cela, veuillez exécuter la commande sous la forme suivante en tant qu'utilisateur **root** :

```
yum history summary
```

Pour afficher les transactions d'une période donnée uniquement, veuillez saisir :

```
yum history summary start_id..end_id
```

De même qu'avec la commande **yum history list**, vous pouvez également afficher un résumé des transactions concernant un ou plusieurs paquets particuliers en fournissant un nom de paquet ou une expression glob :

```
yum history summary glob_expression…
```

Exemple 8.21. Résumé des cinq transactions les plus récentes

```
~]# yum history summary 1..5
Loaded plugins: langpacks, product-id, subscription-manager
Login user                | Time               | Action(s)        |
Altered
------------------------------------------------------------------------
-----------
Jaromir ... <jhradilek>   | Last day           | Install          |
1
Jaromir ... <jhradilek>   | Last week          | Install          |
1
Jaromir ... <jhradilek>   | Last 2 weeks       | I, U             |
73
System <unset>            | Last 2 weeks       | I, U             |
1107
history summary
```

Toutes les formes de la commande **yum history summary** produisent une sortie tabulaire simplifiée similaire à la sortie de **yum history list**.

Comme indiqué ci-dessus, les commandes **yum history list** et **yum history summary** sont orientées vers les transactions, et même si elles permettent d'uniquement afficher les transactions

concernant un ou plusieurs paquets en particulier, des détails cruciaux seront manquants, comme la version des paquets. Pour répertorier les transactions depuis la perspective du paquet, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
yum history package-list glob_expression…
```

Exemple 8.22. Traçage de l'historique d'un paquet

Par exemple, pour tracer l'historique de subscription-manager et de ses paquets connexes, veuillez saisir ce qui suit dans l'invite shell :

```
~]# yum history package-list subscription-manager\*
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-manager
ID     | Action(s)      | Package
-------------------------------------------------------------------------------
     2 | Updated        | subscription-manager-1.13.22-1.el7.x86_64
EE
     2 | Update         |                      1.15.9-15.el7.x86_64
EE
     2 | Obsoleted      | subscription-manager-firstboot-1.13.22-
1.el7.x86_64 EE
     2 | Updated        | subscription-manager-gui-1.13.22-1.el7.x86_64
EE
     2 | Update         |                           1.15.9-
15.el7.x86_64      EE
     2 | Obsoleting     | subscription-manager-initial-setup-addon-
1.15.9-15.el7.x86_64 EE
     1 | Install        | subscription-manager-1.13.22-1.el7.x86_64
     1 | Install        | subscription-manager-firstboot-1.13.22-
1.el7.x86_64
     1 | Install        | subscription-manager-gui-1.13.22-1.el7.x86_64
history package-list
```

Dans cet exemple, trois paquets ont été installés pendant l'installation initiale du système : subscription-manager, subscription-manager-firstboot, et subscription-manager-gui. Dans la troisième transaction, tous les paquets ont été mis à jour de la version 1.10.11 à la version 1.10.17.

8.4.2. Examiner les transactions

Pour afficher le résumé d'une transaction unique, en tant qu'utilisateur **root**, utilisez la commande **yum history summary** sous la forme suivante :

```
yum history summary id
```

Ici, *id* correspond à l'ID de la transaction.

Pour examiner une ou plusieurs transactions en particulier de manière plus détaillée, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
yum history info id…
```

L'argument *id* est optionnel et yum utilise automatiquement la dernière transaction lorsque vous l'omettez. Remarquez que vous pouvez également utiliser une gamme de transaction lorsque vous souhaitez spécifiez plus d'une transaction :

```
yum history info start_id..end_id
```

Exemple 8.23. Exemple de sortie de yum history info

Ci-dessous figure un exemple de sortie de deux transactions, installant chacune un nouveau paquet :

```
~]# yum history info 4..5
Loaded plugins: langpacks, product-id, search-disabled-repos,
subscription-manager
Transaction ID : 4..5
Begin time     : Mon Dec  7 16:51:07 2015
Begin rpmdb    : 1252:d2b62b7b5768e855723954852fd7e55f641fbad9
End time       :            17:18:49 2015 (27 minutes)
End rpmdb      : 1253:cf8449dc4c53fc0cbc0a4c48e496a6c50f3d43c5
User           : Maxim Svistunov <msvistun>
Return-Code    : Success
Command Line   : install tigervnc-server.x86_64
Command Line   : reinstall tigervnc-server
Transaction performed with:
    Installed     rpm-4.11.3-17.el7.x86_64                 @rhel-7-
server-rpms
    Installed     subscription-manager-1.15.9-15.el7.x86_64 @rhel-7-
server-rpms
    Installed     yum-3.4.3-132.el7.noarch                 @rhel-7-
server-rpms
Packages Altered:
    Reinstall tigervnc-server-1.3.1-3.el7.x86_64 @rhel-7-server-rpms
history info
```

Vous pouvez également voir des informations supplémentaires, comme les options de configuration utilisées au moment de la transaction, ou à partir de quel référentiel et pour quelles raisons certains paquets ont été installés. Pour déterminer quelles sont les informations disponibles pour une transaction donnée, veuillez saisir ce qui suit à l'invite shell en tant qu'utilisateur **root** :

```
yum history addon-info id
```

De même qu'avec **yum history info**, lorsqu'aucun *id* n'est fourni, yum utilise automatiquement la dernière transaction. Une autre manière de faire référence à la traduction la plus récente consiste à utiliser le mot-clé **last** :

```
yum history addon-info last
```

Exemple 8.24. Exemple de sortie de yum history addon-info

Pour la quatrième transaction dans l'historique, la commande **yum history addon-info** fournit la sortie suivante :

```
~]# yum history addon-info 4
```

```
Loaded plugins: langpacks, product-id, subscription-manager
Transaction ID: 4
Available additional history information:
  config-main
  config-repos
  saved_tx

history addon-info
```

Dans la sortie de la commande **yum history addon-info**, trois types d'informations sont disponibles :

- **config-main** — options yum globales qui étaient utilisées pendant la transaction. Veuillez consulter la Section 8.5.1, « Définir les options [main] » pour obtenir des informations sur la manière de modifier les options globales.
- **config-repos** — options des référentiels yum individuels. Veuillez consulter la Section 8.5.2, « Définir les options [repository] » pour obtenir des informations sur la manière de modifier les options de référentiels individuels.
- **saved_tx** — les données pouvant être utilisées par la commande **yum load-transaction** afin de répéter la transaction sur une autre machine (voir ci-dessous).

Pour affciher un type sélectionné d'informations supplémentaires, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
yum history addon-info id information
```

8.4.3. Restaurer et répéter des transactions

Hormis le fait de permettre d'examiner l'historique des transactions, la commande **yum history** fournit un moyen de restaurer ou de répéter une transaction sélectionnée. Pour restaurer une transaction, veuillez saisir ce qui suit dans l'invite shell en tant qu'utilisateur **root** :

```
yum history undo id
```

Pour répéter une transaction en particulier, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
yum history redo id
```

Les deux commandes acceptent également le mot-clé **last** pour annuler ou répéter la dernière transaction.

Remarquez que les commandes **yum history undo** et **yum history redo** restaurent ou répètent uniquement les étapes qui ont été effectuées pendant une transaction. Si la transaction a installé un nouveau paquet, la commande **yum history undo** le désinstallera, et si la transaction a désinstallé un paquet, la commande l'installera à nouveau. Cette commande tente également de faire une mise à niveau inférieur de tous les paquets mis à jour vers leur version précédente si ces paquets plus anciens sont toujours disponibles.

Lors de la gestion de plusieurs systèmes identiques, yum vous permet également d'effectuer une transaction sur l'un d'entre eux, de stocker les détails de celle-ci dans un fichier, et après une période de

tests, de répéter la même transaction sur les systèmes restants. Pour stocker les détails de la transaction dans une fichier, veuillez saisir ce qui suit dans l'invite shell en tant qu'utilisateur **root** :

```
yum -q history addon-info id saved_tx > file_name
```

Une fois ce fichier copié sur le système cible, vous pouvez répéter la transaction en utilisant la commande suivante en tant qu'utilisateur **root** :

```
yum load-transaction file_name
```

Vous pouvez configurer **load-transaction** de manière à ignorer les paquets manquants ou la version rpmdb. Pour obtenir davantage d'informations sur ces options de configuration, veuillez consulter la page man **yum.conf**(5).

8.4.4. Lancer un nouvel historique des transactions

Yum stocke l'historique des transactions dans un fichier de base de données SQLite unique. Pour lancer un nouvel historique des transactions, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
yum history new
```

Cela créera un nouveau fichier de base de données vide dans le répertoire **/var/lib/yum/history/**. L'ancien historique des transactions sera conservé, mais il ne sera pas accessible tant qu'un fichier de base de données plus récent sera présent dans le répertoire.

8.5. CONFIGURER YUM ET LES RÉFÉRENTIELS YUM

NOTE

En vue d'élargir vos connaissances, vous serez sans doute interessés par les formations suivantes Red Hat System Administration III (RH254) et RHCSA Rapid Track (RH199).

Les informations de configuration de yum et de ses utilitaires connexes sont situées dans le fichier **/etc/yum.conf**. Ce fichier contient une section **[main]** obligatoire, qui vous permet de définir les options yum qui ont un effet global, et peut également contenir une ou plusieurs section(s) **[*repository*]**, ce qui vous permet de définir des options spécifiques au référentiel. Cependant, il est recommandé de définir des référentiels individuels dans les fichiers **.repo** existants ou nouveaux, qui se trouvent dans le répertoire **/etc/yum.repos.d/**. Les valeurs que vous définissez dans les sections **[*repository*]** du fichier **/etc/yum.conf** remplacent les valeurs définies dans la section **[main]**.

Cette section indique comment :

- définir des options yum globales en modifiant la section **[main]** du fichier de configuration **/etc/yum.conf** ;
- définir des options de référentiels individuels en modifiant les sections **[*repository*]** des fichiers **/etc/yum.conf** et **.repo** dans le répertoire **/etc/yum.repos.d/** ;
- utiliser des variables yum dans **/etc/yum.conf** et des fichiers dans le répertoire **/etc/yum.repos.d/** afin que la version dynamique et les valeurs de l'architecture soient gérées correctement ;

- ajouter, activer, et désactiver des référentiels yum sur la ligne de commande ;
- définir votre propre référentiel yum personnalisé.

8.5.1. Définir les options [main]

Le fichier de configuration **/etc/yum.conf** contient une seule section **[main]** précisément, et même si certaines des paires clés-valeurs dans cette section affectent la manière dont yum opère, d'autres affectent la manière dont yum traite les référentiels. Vous pouvez ajouter de nombreuses options sous l'en-tête de la section **[main]** dans **/etc/yum.conf**.

Un exemple du fichier de configuration **/etc/yum.conf** ressemble à ceci :

```
[main]
cachedir=/var/cache/yum/$basearch/$releasever
keepcache=0
debuglevel=2
logfile=/var/log/yum.log
exactarch=1
obsoletes=1
gpgcheck=1
plugins=1
installonly_limit=3
[commentaires abrégés]

# PUT YOUR REPOS HERE OR IN separate files named file.repo
# in /etc/yum.repos.d
```

Ci-dessous figurent les options de la section **[main]** les plus couramment utilisées :

assumeyes=*value*

L'option **assumeyes** permet de déterminer si yum demande de confirmer les actions critiques ou non. Remplacez *value* par l'une des valeurs suivantes :

0 (*par défaut*) — yum demande de confirmer les actions critiques effectuées.

1 — ne demande pas de confirmer les actions critiques de **yum**. Si **assumeyes=1** est défini, yum se comportera de la même manière qu'avec les options de ligne de commande **-y** et **--assumeyes**.

cachedir=*directory*

Veuillez utiliser cette option pour définir le répertoire dans lequel yum stockera son cache et ses fichiers de base de données. Remplacez *directory* par un chemin complet vers le répertoire. Par défaut, le répertoire du cache de yum est le suivant : **/var/cache/yum/$basearch/$releasever/**.

Veuillez consulter la Section 8.5.3, « Utiliser des variables Yum » pour les descriptions des variables yum **$basearch** et **$releasever**.

debuglevel=*value*

Cette option indique le niveau de détails de la sortie de débogage produite par yum. Ici, *value* est un entier entre **1** et **10**. Définir une valeur **debuglevel** plus élevée amènera yum à afficher une sortie de débogage plus détaillée. **debuglevel=2** est la valeur par défaut, tandis que **debuglevel=0** désactive la sortie de débogage.

exactarch=*value*

Avec cette option, vous pouvez définir yum pour que l'architecture exacte soit prise en considération lors de la mise à jour de paquets installés au préalable. Veuillez remplacer *value* par :

0 — ne pas prendre en compte l'architecture exacte pendant la mise à jour des paquets.

1 (*par défaut*) — prendre en considération l'architecture exacte pendant la mise à jour des paquets. Avec ce paramètre, yum n'installera pas de paquets pour une architecture 32 bits pour mettre à jour un paquet déjà installé sur un système avec une architecture 64 bits.

exclude=*package_name* [*more_package_names*]

L'option **exclude** vous permet d'exclure des paquets par mot-clé pendant une installation ou une mise à jour du système. Il est possible de sélectionner plusieurs paquets les exclure en indiquant une liste de paquets délimités par des espaces. Les expression glob du shell utilisant des caractères génériques (par exemple, ***** et **?**) sont autorisées.

gpgcheck=*value*

Veuillez utiliser l'option **gpgcheck** pour spécifier si yum doit effectuer une vérification des signatures GPG sur les paquets. Remplacez *value* par :

0 — désactive la vérification des signatures GPG sur les paquets de tous les référentiels, y compris l'installation de paquets locaux.

1 (*par défaut*) — active la vérification de signatures GPG sur tous les paquets dans tous les référentiels, y compris l'installation de paquets locaux. Lorsque **gpgcheck** est activé, les signatures de tous les paquets sont vérifiées.

Si cette option est définie dans la section **[main]** du fichier **/etc/yum.conf**, elle définit une règle de vérification GPG pour tous les référentiels. Cependant, vous pouvez également définir **gpgcheck=*value*** pour les référentiels individuels à la place ; c'est-à-dire que vous pouvez activer la vérification GPG sur un référentiel tout en le désactivant sur un autre. Définir **gpgcheck=*value*** pour un référentiel individuel dans son fichier **.repo** correspondant remplace la valeur par défaut si elle est présente dans **/etc/yum.conf**.

Pour obtenir davantage d'informations sur la vérification de signatures GPG, veuillez consulter la Section A.3.2, « Vérification des signatures de paquets ».

group_command=*value*

Utilisez l'option **group_command** pour spécifier de quelle manière les commandes **yum group install**, **yum group upgrade**, et **yum group remove** gèrent un groupe de paquets. Remplacez *value* par l'une des valeurs suivantes :

simple — installe tous les membres d'un groupe de paquets. Met à niveau les paquets installés antérieurement uniquement, mais n'installe pas de paquets qui ont été ajoutés au groupe entretemps.

compat — est similaire à **simple** mais **yum upgrade** installe également des paquets qui ont été ajoutés au groupe depuis la mise à niveau précédente.

objects — (*par défaut*.) Avec cette option, yum garde sous surveillance les groupes installés antérieurement et fait une distinction entre les paquets installés faisant partie du groupe et les paquets installés séparément. Veuillez consulter l'Exemple 8.15, « Afficher des informations sur le groupe de paquets LibreOffice »

group_package_types=*package_type* [*more_package_types*]

Vous pouvez spécifier ici quel type de paquet (*optionnel*, *par défaut* ou *obligatoire*) est installé lorsque la commande **yum group install** est appelé. Les types de paquets *par défaut* et *obligatoire* sont choisis par défaut.

history_record=*value*

Avec cette option, vous pouvez définir yum pour enregistrer l'historique des transactions. Remplacez *value* par l'une des valeurs suivantes :

0 — yum ne doit *pas* enregistrer d'entrées de l'historique pour les transactions.

1 (*par défaut*) — yum devrait enregistrer les entrées de l'historique des transactions. Cette opération prend une certaine quantité d'espace disque et un certain temps supplémentaire avec les transactions, mais elle fournit également de nombreuses informations concernant les anciennes opérations, qui peuvent être affichées par la commande **yum history**. **history_record=1** est la valeur par défaut.

Pour obtenir davantage d'informations sur la commande **yum history**, veuillez consulter la Section 8.4, « Utiliser l'historique des transactions ».

NOTE

Yum utilise des enregistrements de l'historique pour détecter les modifications apportées à la base de données **rpmdb** qui ont été effectuées hors de yum. Dans de tels cas, yum affiche un avertissement et recherche automatiquement les problèmes possibles causés par l'altération de **rpmdb**. Lorsque **history_record** est éteint, yum n'est pas en mesure de détecter ces changements et aucune vérifications automatique n'est effectuée.

installonlypkgs=*space separated list of packages*

Ici vous pouvez fournir une liste de paquets séparés par des virgules que yum peut *installer*, mais ne mettra jamais *à jour*. Veuillez consulter la page man **yum.conf**(5) pour obtenir la liste des paquets qui sont par défaut « install-only » (uniquement pour installation).

Si vous ajoutez la directive **installonlypkgs** à **/etc/yum.conf**, vous devriez vous assurer de répertorier *tous* les paquets install-only, y compris ceux répertoriés sous la section **installonlypkgs** de **yum.conf**(5). Particulièrement, les paquets du noyau doivent toujours être répertoriés dans **installonlypkgs** (comme ils le sont par défaut), et **installonly_limit** devrait toujours être défini sur une valeur supérieure à **2** afin qu'un noyau de secours soit toujours disponible en cas d'échec du démarrage du noyau par défaut.

installonly_limit=*value*

Cette option définit combien de paquets répertoriés dans la directive **installonlypkgs** peuvent être installés au même moment. Remplacez *value* par un entier représentant le nombre maximal de versions pouvant être installées simultanément pour tout paquet unique répertorié dans **installonlypkgs**.

Les valeurs par défaut de la directive **installonlypkgs** incluent plusieurs paquets de noyaux différents. Ainsi, veuillez ne pas oublier que modifier la valeur de **installonly_limit** affecte également le nombre maximal de versions installées pour un paquet de noyau unique. La valeur par défaut répertoriée dans **/etc/yum.conf** est **installonly_limit=3**, et il n'est pas recommandé de réduire cette valeur, et plus particulièrement à une valeur inférieure à **2**.

keepcache=*value*

L'option **keepcache** détermine si yum garde le cache des en-têtes et des paquets après une installation réussie. Ici, *value* correspond à l'une des valeurs suivantes :

0 (*par défaut*) — ne pas retenir le cache des en-têtes et des paquets après une installation réussie.

1 — retenir le cache après une installation réussie.

logfile=*file_name*

Pour spécifier l'emplacement de journalisation de la sortie, veuillez remplacer *file_name* par un chemin complet vers le fichier dans lequel yum écrira la sortie journalisée. Par défaut, yum enregistre les journaux dans **/var/log/yum.log**.

max_connenctions=*number*

value correspond au nombre maximal de connexions simultanées, la valeur par défaut est 5.

multilib_policy=*value*

L'option **multilib_policy** définit le comportement de l'installation si plusieurs versions de l'architecture sont disponibles pour une installation de paquet. Ici, *value* correspond à :

best — installe le meilleur choix d'architecture pour ce système. Par exemple, paramétrer **multilib_policy=best** sur un système AMD64 amène yum à installer les versions 64 bits de tous les paquets.

all — installe à chaque fois toutes les architectures possibles pour chaque paquet. Par exemple, avec **multilib_policy** défini sur **all** sur un système AMD64, yum installerait les versions i686 et AMD64 d'un paquet, si ces deux versions étaient disponibles.

obsoletes=*valeur*

L'option **obsoletes** active la logique de traitement obsolète pendant les mises à jour. Lorsqu'un paquet déclare dans son fichier de spécifications qu'il rend un autre paquet *obsolète*, ce dernier est remplacé par l'ancien paquet lorsque le premier est installé. Les fichiers obsolètes sont déclarés, par exemple, lorsqu'un paquet est renommé. Remplacez *valeur* par soit :

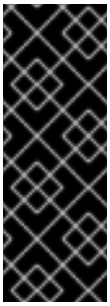
0 — désactive la logique de traitement obsolète de yum lorsque des mises à jour sont effectuées.

1 (*par défaut*) — active la logique de traitement obsolète de yum lorsque des mises à jour sont effectuées.

plugins=*value*

Interrupteur global pour activer ou désactiver les greffons yum, *value* est l'une des valeurs suivantes :

0 — désactive tous les greffons yum globalement.



IMPORTANT

La désactivation de tous les greffons n'est pas recommandée car certains greffons fournissent des services yum importants. En particulier, les greffons **product-id** et **subscription-manager** fournissent la prise en charge du **Content Delivery Network** (CDN) basé certificats. La désactivation globale des greffons est offerte en tant qu'option pratique, et n'est généralement recommandée que lors du diagnostique d'un problème potentiel avec yum.

1 (*par défaut*) — active tous les greffons yum globalement. Avec **plugins=1**, vous pouvez toujours désactiver un greffon yum spécifique en paramétrant **enabled=0** dans le fichier de configuration de ce greffon.

Pour obtenir davantage d'informations sur les divers greffons yum, veuillez consulter la Section 8.6, « Greffons Yum ». Pour obtenir des informations supplémentaires sur le contrôle des greffons, veuillez consulter la Section 8.6.1, « Activer, configurer, et désactiver des greffons Yum ».

reposdir=*directory*

Ici, *directory* est un chemin complet vers le répertoire où se trouvent les fichiers **.repo**. Tous les fichiers **.repo** contiennent des informations de référentiel (similairement aux sections **[*référentiel*]** de **/etc/yum.conf**). Yum collecte toutes les informations des référentiels des fichiers **.repo** et la section **[*référentiel*]** du fichier **/etc/yum.conf** pour créer une liste maître des référentiels à utiliser pour des transactions. Si **reposdir** n'est pas défini, yum utilisera le répertoire par défaut **/etc/yum.repos.d/**.

retries=*valeur*

Cette option définit le nombre de fois que yum doit tenter de récupérer un fichier avant de retourner une erreur. *value* est un entier **0** ou supérieur. Définir la valeur sur **0** fait que yum effectuera des tentatives indéfiniement. La valeur par défaut est **10**.

Pour obtenir la liste complète des options **[main]** disponibles, veuillez consulter la section **[main] OPTIONS** de la page man de **yum.conf**(5).

8.5.2. Définir les options [repository]

Les sections **[*repository*]**, où *repository* est un ID de référentiel unique, tel que **my_personal_repo** (les espaces ne sont pas autorisés), vous permet de définir des référentiels yum individuels. Afin d'éviter tout conflit, les référentiels personnalisés ne doivent pas utiliser des noms utilisés dans les référentiels Red Hat.

Ci-dessous figure un exemple du strict minimum de la forme prise par la section **[*repository*]** :

```
[repository]
name=repository_name
baseurl=repository_url
```

Chaque section **[*référentiel*]** doit contenir les directives suivantes :

name=*nom référentiel*

Ici, *nom référentiel* est une chaîne lisible par l'utilitsateur décrivant le référentiel.

baseurl=*url référentiel*

Remplacez *url référentiel* par un URL vers le répertoire où se trouve le référentiel repodata d'un référentiel :

- Si le référentiel est disponible sur HTTP, veuillez utiliser : **http://path/to/repo**
- Si le référentiel est disponible sur FTP, veuillez utiliser : **ftp://path/to/repo**
- Si le référentiel est local à la machine, veuillez utiliser : **file:///path/to/local/repo**
- Si un référentiel en ligne spécifique requiert une authentification HTTP de base, vous pouvez spécifier votre nom d'utilisateur et mot de passe en les ajoutant au début de l'URL comme ceci : ***nom d'utilisateur*:*mot de âsse*@*lien***. Par exemple, si un référentiel sur http://www.example.com/repo/ requiert un nom d'utilisateur « user » et un mot de passe « password », alors le lien **baseurl** link pourra être spécifié comme suit : **http://user:password@www.example.com/repo/**.

Habituellement, cet URL est un lien HTTP, tel que :

```
baseurl=http://path/to/repo/releases/$releasever/server/$basearch/os/
```

Remarquez que yum étend toujours les variables **$releasever**, **$arch**, et **$basearch** dans les URL. Pour obtenir davantage d'informations sur les variables yum, veuillez consulter la Section 8.5.3, « Utiliser des variables Yum ».

D'autres directives de **[*référentiel*]** utiles incluent :

enabled=*valeur*

Ceci est une manière simple de dire à yum d'utiliser ou d'ignorer un référentiel en particulier, la *value* correspond à l'une des valeurs suivantes :

0 — ne pas inclure ce référentiel en tant que source de paquet lorsque vous effectuez des mises à jour et des installations. Ceci est une manière simple d'activer et de désactiver des référentiels, ce qui est utile lorsque vous souhaitez un certain paquet d'un référentiel que vous ne souhaitez pas activer pour les mises à jour ou les installations.

1 — inclure ce référentiel en tant que source de paquets.

Activer et désactiver des référentiels peut également être effectué en passant l'option **--enablerepo=*repo_name*** ou **--disablerepo=*repo_name*** sur **yum**, ou par la fenêtre **Ajouter/Supprimer un logiciel** de l'utilitaire **PackageKit**.

async=*valeur*

Contrôle le téléchargement parallèle de paquets de référentiels. La *valeur* correspond à :

auto (*par défaut*) — un téléchargement parallèle est utilisé si possible, ce qui signifie que **yum** le désactive automatiquement pour les référentiels créés par les greffons afin d'éviter des échecs.

on — téléchargement parallèle activé pour le référentiel.

off — téléchargement parallèle désactivé pour le référentiel.

De nombreuses autres options **[*référentiel*]** existent. Quelques-unes d'entre elles ont la même forme et fonction que certaines options **[*main*]**. Pour une liste complète, veuillez consulter la section **[repository] OPTIONS** de la page man de**yum.conf**(5).

Exemple 8.25. Exemple de fichier /etc/yum.repos.d/redhat.repo

Ci-dessous figure un exemple du fichier **/etc/yum.repos.d/redhat.repo** :

```
#
# Red Hat Repositories
# Managed by (rhsm) subscription-manager
#

[red-hat-enterprise-linux-scalable-file-system-for-rhel-6-entitlement-
rpms]
name = Red Hat Enterprise Linux Scalable File System (for RHEL 6
Entitlement) (RPMs)
baseurl = https://cdn.redhat.com/content/dist/rhel/entitlement-
6/releases/$releasever/$basearch/scalablefilesystem/os
enabled = 1
gpgcheck = 1
gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
sslverify = 1
sslcacert = /etc/rhsm/ca/redhat-uep.pem
sslclientkey = /etc/pki/entitlement/key.pem
sslclientcert = /etc/pki/entitlement/11300387955690106.pem

[red-hat-enterprise-linux-scalable-file-system-for-rhel-6-entitlement-
source-rpms]
name = Red Hat Enterprise Linux Scalable File System (for RHEL 6
Entitlement) (Source RPMs)
baseurl = https://cdn.redhat.com/content/dist/rhel/entitlement-
6/releases/$releasever/$basearch/scalablefilesystem/source/SRPMS
enabled = 0
gpgcheck = 1
gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
sslverify = 1
sslcacert = /etc/rhsm/ca/redhat-uep.pem
sslclientkey = /etc/pki/entitlement/key.pem
sslclientcert = /etc/pki/entitlement/11300387955690106.pem

[red-hat-enterprise-linux-scalable-file-system-for-rhel-6-entitlement-
debug-rpms]
name = Red Hat Enterprise Linux Scalable File System (for RHEL 6
Entitlement) (Debug RPMs)
baseurl = https://cdn.redhat.com/content/dist/rhel/entitlement-
6/releases/$releasever/$basearch/scalablefilesystem/debug
enabled = 0
gpgcheck = 1
gpgkey = file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
sslverify = 1
sslcacert = /etc/rhsm/ca/redhat-uep.pem
sslclientkey = /etc/pki/entitlement/key.pem
sslclientcert = /etc/pki/entitlement/11300387955690106.pem
```

8.5.3. Utiliser des variables Yum

Vous pouvez utiliser et faire référence aux variables intégrées suivantes présentes dans les commandes **yum** et dans tous les fichiers de configuration yum (c'est-à-dire **/etc/yum.conf** et tous les fichiers **.repo** dans le répertoire **/etc/yum.repos.d/**) :

$releasever

Vous pouvez utiliser cette variable pour faire référence à la version Red Hat Enterprise Linux. Yum obtient la valeur de **$releasever** à partir de la ligne **distroverpkg=*value*** dans le fichier de configuration **/etc/yum.conf**. Si cette ligne n'existe pas dans **/etc/yum.conf**, alors yum déduira la valeur correcte en dérivant la numéro de version à partir du paquet **redhat-release*produit*** qui fournit le fichier **redhat-release** .

$arch

Vous pouvez utiliser cette variable pour faire référence à l'architecture du CPU du système comme retourné lors d'un appel à la fonction **os.uname()** de Python. Les valeurs valides de **$arch** incluent : **i586**, **i686** et **x86_64**.

$basearch

Vous pouvez utiliser **$basearch** pour faire référence à l'architecture de base du système. Par exemple, les machines i686 et i586 ont toutes deux une architecture de base **i386**, et les machines AMD64 et Intel64 ont une architecture de base **x86_64**.

$YUM0-9

Ces dix variables sont chacunes remplacées par la valeur d'une variable d'environnement shell du même nom. Si l'une de ces variables est référencée (par exemple dans **/etc/yum.conf**) et qu'une variable d'environnement shell du même nom n'existe pas, alors la variable du fichier de configuration ne sera pas remplacée.

Pour définir une variable personnalisée ou pour remplacer la valeur d'une variable existante, créez un fichier du même nom que la variable (sans le caractère « **$** ») dans le répertoire **/etc/yum/vars/**, et ajoutez la valeur souhaitée sur la première ligne.

Par exemple, les descriptions de référentiels incluent souvent le nom du système d'exploitation. Pour définir une nouvelle variable nommée **$osname**, créez un nouveau fichier avec « Red Hat Enterprise Linux » sur la première ligne et enregistrez-le sous **/etc/yum/vars/osname** :

```
~]# echo "Red Hat Enterprise Linux 7" > /etc/yum/vars/osname
```

Au lieu de « Red Hat Enterprise Linux 7 », vous pouvez désormais utiliser ce qui suit dans les fichiers **.repo** :

```
name=$osname $releasever
```

8.5.4. Afficher la configuration actuelle

Pour afficher les valeurs actuelles des options yum globales (c'est-à-dire les options spécifiées dans la section **[main]** du fichier **/etc/yum.conf**), veuillez exécuter la commande **yum-config-manager** sans aucune option de ligne de commande :

yum-config-manager

Pour répertorier le contenu d'une ou plusieurs sections de configuration différentes, veuillez utiliser la commande sous la forme suivante :

yum-config-manager *section…*

Vous pouvez également utiliser une expression glob pour afficher la configuration de toutes les sections correspondantes :

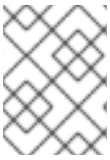
yum-config-manager *glob_expression…*

Exemple 8.26. Afficher la configuration de la section principale

Pour répertorier toutes les options de configuration de la section principale ainsi que leurs valeurs correspondantes, veuillez saisir ce qui suit dans l'invite shell :

```
~]$ yum-config-manager main \*
Loaded plugins: langpacks, product-id, subscription-manager
================================== main
==================================
[main]
alwaysprompt = True
assumeyes = False
bandwith = 0
bugtracker_url = https://bugzilla.redhat.com/enter_bug.cgi?
product=Red%20Hat%20Enterprise%20Linux%206&component=yum
cache = 0[sortie tronquée]
```

8.5.5. Ajouter, activer, et désactiver un référentielr Yum



NOTE

Afin de gagner en expertise, vous serez sans doute intéressé par le cours de formation Red Hat System Administration II (RH254).

La Section 8.5.2, « Définir les options [repository] » décrit les diverses options que vous pouvez utiliser pour définir un référentiel yum. Cette section explique comment ajouter, activer, et désactiver un référentiel en utilisant la commande **yum-config-manager**.

IMPORTANT

Lorsque le système est enregistré dans Red Hat Subscription Management sur le **Content Delivery Network** (CDN) basé sur certificats, les outils **Red Hat Subscription Manager** sont utilisés pour gérer des référentiels dans le fichier **/etc/yum.repos.d/redhat.repo**.

Ajouter un référentiel Yum

Pour définir un nouveau référentiel, vous pouvez ajouter une section **[*référentiel*]** dans le fichier **/etc/yum.conf**, ou au fichier **.repo** du répertoire **/etc/yum.repos.d/**. Tous les fichiers avec

l'extension de fichier **.repo** présents dans ce répertoire sont lus par yum, et il est recommandé de définir vos référentiels ici plutôt que dans **/etc/yum.conf**.

AVERTISSEMENT

Obtenir et installer des paquets logiciels de sources logicielles non vérifiées ou qui ne sont pas de confiance et proviennent de sources autres que le **Content Delivery Network** (CDN) basé sur certificats de Red Hat constitue un risque de sécurité potentiel, et pourrait provoquer des problèmes de sécurité, de stabilité, de compatibilité, et de maintenance.

Les référentiels Yum fournissent normalement leur propre fichier **.repo**. Pour ajouter un tel référentiel à votre système et pour l'activer, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
yum-config-manager --add-repo repository_url
```

…où *repository_url* est un lien vers le fichier **.repo**.

Exemple 8.27. Ajouter example.repo

Pour ajouter un référentiel situé sur http://www.example.com/example.repo, veuillez saisir ce qui suit dans l'invite shell :

```
~]# yum-config-manager --add-repo http://www.example.com/example.repo
Loaded plugins: langpacks, product-id, subscription-manager
adding repo from: http://www.example.com/example.repo
grabbing file http://www.example.com/example.repo to
/etc/yum.repos.d/example.repo
example.repo                                               |  413 B
00:00
repo saved to /etc/yum.repos.d/example.repo
```

Activer un référentiel Yum

Pour activer un ou plusieurs référentiels en particulier, veuillez saisir ce qui suit dans une invite shell en tant qu'utilisateur **root** :

```
yum-config-manager --enable repository…
```

…où *référentiel* est l'ID unique du référentiel (utilisez **yum repolist all** pour répertorier les ID des référentiels disponibles). Alternativement, vous pouvez utiliser une expression glob pour activer tous les référentiels correspondants :

```
yum-config-manager --enable glob_expression…
```

Exemple 8.28. Activer les référentiels définis dans des sections personnalisées de /etc/yum.conf.

Pour activer les référentiels définis dans les sections **[example]**, **[example-debuginfo]**, et **[example-source]**, veuillez saisir :

```
~]# yum-config-manager --enable example\*
Loaded plugins: langpacks, product-id, subscription-manager
============================== repo: example
==============================
[example]
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/7Server
baseurl = http://www.example.com/repo/7Server/x86_64/
cache = 0
cachedir = /var/cache/yum/x86_64/7Server/example[sortie tronquée]
```

Exemple 8.29. Activer tous les référentiels

Pour activer tous les référentiels définis dans le fichier **/etc/yum.conf** et dans le répertoire **/etc/yum.repos.d/**, saisissez :

```
~]# yum-config-manager --enable \*
Loaded plugins: langpacks, product-id, subscription-manager
============================== repo: example
==============================
[example]
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/7Server
baseurl = http://www.example.com/repo/7Server/x86_64/
cache = 0
cachedir = /var/cache/yum/x86_64/7Server/example[sortie tronquée]
```

Si elle fonctionne correctement, la commande **yum-config-manager --enable** affiche la configuration du référentiel actuel.

Désactiver un référentiel Yum

Pour désactiver un référentiel yum, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

yum-config-manager --disable *repository*…

…où *référentiel* est l'ID unique du référentiel (utilisez **yum repolist all** pour répertorier les ID des référentiels disponibles). De même qu'avec **yum-config-manager --enable**, vous pouvez utiliser une expression glob pour désactiver tous les référentiels correspondants en même temps :

yum-config-manager --disable *glob_expression*…

Exemple 8.30. Désactiver tous les référentiels

Pour désactiver tous les référentiels définis dans le fichier **/etc/yum.conf** et dans le répertoire **/etc/yum.repos.d/**, saisissez :

```
~]# yum-config-manager --disable \*
```

```
Loaded plugins: langpacks, product-id, subscription-manager
============================== repo: example
==============================
[example]
bandwidth = 0
base_persistdir = /var/lib/yum/repos/x86_64/7Server
baseurl = http://www.example.com/repo/7Server/x86_64/
cache = 0
cachedir = /var/cache/yum/x86_64/7Server/example[sortie tronquée]
```

Si elle fonctionne correctement, la commande **yum-config-manager --disable** affiche la configuration actuelle.

8.5.6. Création d'un référentiel Yum

Pour définir un référentiel yum, effectuez les étapes suivantes :

1. Installez le paquet createrepo. Pour faire cela, veuillez saisir ce qui suit dans l'invite shell en tant qu'utilisateur **root** :

   ```
   yum install createrepo
   ```

2. Copiez tous les paquets que vous souhaitez avoir dans votre référentiel dans un seul répertoire, comme **/mnt/local_repo/**.

3. Déplacez-vous sur ce répertoire et exécutez la commande suivante :

   ```
   createrepo --database /mnt/local_repo
   ```

   Ceci crée les métadonnées nécessaires pour votre référentiel yum, ainsi que la base de données **sqlite** pour accélérer les opérations yum.

8.5.7. Ajouter les référentiels « Optional » (Optionnel) et « Supplementary » (Supplémentaire)

Les canaux d'abonnements « Optional » et « Supplementary » fournissent des paquets logiciels supplémentaires pour Red Hat Enterprise Linux qui offrent des logiciels sous licence open source (dans le canal « Optional ») et sous licence propriétaire (dans le canal « Supplementary »).

Avant de vous abonner aux canaux « Optional » et « Supplementary », veuillez consulter les Détails de l'étendue de la couverture. Si vous décidez d'installer des paquets à partir de ces canaux, veuillez suivre les étapes documentées dans l'article nommé Comment accéder aux canaux « Optional » et « Supplementary » et aux paquets -devel en utilisant Red Hat Subscription Manager (RHSM) ? sur le Portail Client Red Hat.

8.6. GREFFONS YUM

Yum fournit des greffons qui étendent et améliorent ses opérations. Certains greffons sont installés par défaut. S'il y en a, Yum vous informera toujours sur les greffons chargés et actifs lorsque vous exécutez une commande **yum**. Exemple :

```
~]# yum info yum
Loaded plugins: langpacks, product-id, subscription-manager[sortie
tronquée]
```

Remarquez que les noms de greffons qui suivent **Loaded plugins** sont les noms que vous pouvez fournir à l'option **--disableplugin=*plugin_name***.

8.6.1. Activer, configurer, et désactiver des greffons Yum

Pour activer des greffons yum, assurez-vous qu'une ligne commençant par **plugins=** soit effectivement présente dans la section **[main]** du fichier **/etc/yum.conf**, et que sa valeur soit égale à **1** :

```
plugins=1
```

Vous pouvez désactiver tous les greffons en modifiant cette ligne comme suit : **plugins=0**.

IMPORTANT

La désactivation de tous les greffons n'est pas recommandée car certains greffons fournissent des services yum importants. En particulier, les greffons **product-id** et **subscription-manager** fournissent la prise en charge du **Content Delivery Network** (CDN) basé sur certificats. La désactivation globale des greffons est offerte en tant qu'option pratique, et est généralement uniquement recommandée lors du diagnostique d'un problème potentiel avec yum.

Chaque greffon installé possède son propre fichier de configuration dans le répertoire **/etc/yum/pluginconf.d/**. Vous pouvez définir des options spécifiques aux greffons dans ces fichiers. Par exemple, ci-dessous figure le fichier de configuration du greffon **aliases** nommé **aliases.conf** :

```
[main]
enabled=1
```

De même qu'avec le fichier **/etc/yum.conf**, les fichiers de configuration des greffons contiennent toujours une section **[main]** où l'option **enabled=** contrôle si le greffon est activé lorsque vous exécutez des commandes **yum**. Si cette option est manquante, vous pouvez l'ajouter manuellement au fichier.

Si vous désactivez tous les greffons en paramétrant **enabled=0** dans **/etc/yum.conf**, alors tous les greffons seront désactivés, peu importe s'ils étaient activés dans leurs fichiers de configuration individuels.

Si vous souhaitez simplement désactiver tous les greffons yum pour une commande **yum** unique, veuillez utiliser l'option **--noplugins**.

Si vous souhaitez désactiver un ou plusieurs greffons yum pour une seule commande **yum**, ajoutez l'option **--disableplugin=*plugin_name*** à la commande. Par exemple, pour désactiver le greffon **aliases** pendant une mise à jour du système, veuillez saisir :

```
~]# yum update --disableplugin=aliases
```

Les noms des greffons que vous fournissez à l'option **--disableplugin=** sont les mêmes que ceux

répertoriés après la ligne **Loaded plugins** dans la sortie de toute commande **yum**. Vous pouvez désactiver de multiples greffons en séparant leurs noms par des virgules. En outre, vous pouvez faire correspondre plusieurs noms de greffons ou raccourcir les plus longs en utilisant des expressions glob :

```
~]# yum update --disableplugin=aliases,lang*
```

8.6.2. Installer des greffons Yum supplémentaires

Les greffons Yum adhèrent habituellement à la convention de dénomination de paquets **yum-plugin-*plugin_name***, ,mais pas toujours : par exemple, le paquet qui fournit le greffon **kabi** est nommé **kabi-yum-plugins**. Vous pouvez installer un greffon yum de la même manière que si vous installiez d'autres paquets. Par exemple, pour installer le greffon **yum-aliases**, veuillez saisir ce qui suit dans l'invite shell :

```
~]# yum install yum-plugin-aliases
```

8.6.3. Utiliser des greffons Yum

La liste suivante fournit les descriptions et instructions d'utilisation de plusieurs greffons yum utiles. Les greffons sont répertoriés par nom, les crochets contiennent le nom du paquet.

search-disabled-repos (subscription-manager)

Le greffon **search-disabled-repos** permet d'activer temporairement ou de façon permanente des référentiels désactivés afin de vous aider à résoudre des problèmes de dépendances. Avec ce greffon activé, quand Yum échoue à l'installation d'un paquet à cause d'une erreur de résolution de dépendance, il propose d'activer temporairement les référentiels désactivés pour essayer à nouveau. Si l'installation réussit, Yum propose également d'activer les référentiels utilisés de façon permanente. Notez que le greffon ne fonctionne qu'avec les référentiels qui sont gérés par le **subscription-manager** et ne fonctionne pas avec les référentiels personnalisés.

IMPORTANT

Quand **yum** exécute avec l'option **--assumeyes** ou **-y**, ou si la directive **assumeyes** est activée dans **/etc/yum.conf**, le greffon active les référentiels désactivés, à la fois temporairement et de façon permanente, sans invitation de confirmation. Cela peut mener à des problèmes, comme par exemple, d'activer des référentiels que vous ne souhaitez pas activer.

Pour configurer le greffon **search-disabled-repos**, modifiez le fichier de configuration situé dans **/etc/yum/pluginconf.d/search-disabled-repos.conf**. Une liste de directives pouvant être utilisées dans la section **[main]** est affichée dans le tableau suivant.

Tableau 8.3. Directives `search-disabled-repos.conf` prises en charge

Directive	Description
enabled=*valeur*	Vous permet d'activer ou de désactiver le greffon. La *valeur* doit être **1** (activé), ou **0** (désactivé). Le greffon est activé par défaut.

Directive	Description
notify_only=*valeur*	Vous permet de limiter le comportement du greffon aux notifications uniquement. La *valeur* doit correspondre à **1** (notifie sans modifier le comportement de Yum), ou **0** (modifie le comportement de Yum). Par défaut, le greffon ne notifie que l'utilisateur.
ignored_repos=*référentiels*	Vous permet de spécifier les référentiels qui ne seront pas activés par le greffon.

kabi (kabi-yum-plugins)

Le greffon **kabi** vérifie si un paquet de mise à jour de pilote est conforme à l'interface binaire d'application du noyau officielle de Red Hat (« *kernel Application Binary Interface* », ou kABI). Avec l'activation de ce greffon, lorsqu'un utilisateur tente d'installer un paquet qui utilise des symboles du noyau ne se trouvant pas sur une liste blanche, un message d'avertissement est écrit sur le journal système. En outre, configurer le greffon pour qu'il soit exécuté en mode « enforcing » empêche de tels paquets d'être installés.

Pour configurer le greffon **kabi**, modifiez le fichier de configuration situé dans **/etc/yum/pluginconf.d/kabi.conf**. Une liste de directives pouvant être utilisées dans la section **[main]** est affichée dans le tableau ci-dessous.

Tableau 8.4. Directives kabi.conf prises en charge

Directive	Description
enabled=*valeur*	Vous permet d'activer ou de désactiver le greffon. La valeur *valeur* doit être **1** (activé), ou **0** (désactivé). Une fois installé, le greffon est activé par défaut.
whitelists=*répertoire*	Permet de spécifier le *répertoire* dans lequel les fichiers avec les symboles du noyau pris en charge se trouvent. Par défaut, le greffon **kabi** utilise des fichiers fournis par le paquet kernel-abi-whitelists (c'est-à-dire le répertoire **/usr/lib/modules/kabi-rhel70/**).
enforce=*valeur*	Vous permet d'activer ou de désactiver le mode « enforcing ». La *valeur* doit être égale à **1** (activé), ou à **0** (désactivé). Par défaut, cette option est mise en commentaire et le greffon **kabi** n'affiche qu'un message d'avertissement.

product-id (subscription-manager)

Le greffon **product-id** gère les certificats d'identité des produits installés à partir du CDN. Le greffon **product-id** est installé par défaut.

langpacks (yum-langpacks)

Le greffon **langpacks** est utilisé pour rechercher les paquets des paramètres régionaux d'une langue sélectionnée pour chaque paquet ayant été installé. Le greffon **langpacks** est installé par défaut.

aliases (yum-plugin-aliases)

Le greffon **aliases** offre l'option de ligne de commande `alias` qui autorise la configuration et l'utilisation d'alias pour les commandes `yum`.

yum-changelog (yum-plugin-changelog)

Le greffon **yum-changelog** offre l'option de ligne de commande `--changelog` qui permet d'afficher les journaux des changements d'un paquet avant et après une mise à jour.

yum-tmprepo (yum-plugin-tmprepo)

Le greffon **yum-tmprepo** offre l'option de ligne de commande `--tmprepo` qui prend l'URL d'un fichier référentiel, le télécharge et l'active pour une seule transaction. Ce greffon essaie d'assurer une utilisation temporaire sécurisée de ses référentiels. Par défaut, il ne permet pas de désactiver la vérification GPG.

yum-verify (yum-plugin-verify)

Le greffon **yum-verify** offre les options de ligne de commande `verify`, `verify-rpm`, et `verify-all` pour afficher les données de vérification sur le système.

yum-versionlock (yum-plugin-versionlock)

Le greffon **yum-versionlock** exclut les autres versions des paquets sélectionnés, ce qui permet de protéger des paquets de mise à jour vers de nouvelles versions. Avec l'option de ligne de commande `versionlock`, vous pouvez afficher et modifier la liste des paquets verrouillés.

8.7. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur la manière de gérer les paquets logiciels dans Red Hat Enterprise Linux, veuillez consulter les ressources répertoriées ci-dessous.

Documentation installée

- **yum**(8) — la page du man de l'utilitaire de ligne de commande yum fournit une liste complète des options et des commandes prises en charge.
- **yumdb**(8) — la page du man de l'utilitaire de ligne de commande **yumdb** documente comment utiliser cet outil pour effectuer des requêtes, et altérer la base de données yum si nécessaire.
- **yum.conf**(5) — la page du man de `yum.conf` documente les options de configuration yum disponibles.
- **yum-utils**(1) — la page du man de `yum-utils` répertorie et décrit brièvement les utilitaires supplémentaires pour gérer des configurations yum, manipuler des référentiels, et utiliser une base de données yum.

Ressources en ligne

- Yum Guides — la page des *Guides Yum* sur la page d'accueil du projet fournit des liens vers une documentation supplémentaire.
- Red Hat Access Labs — les Red Hat Access Labs incluent un « Yum Repository Configuration Helper ».

Voir aussi

- Chapitre 5, *Obtention de privilèges* documente comment obtenir des privilèges administratifs en utilisant les commandes **su** et **sudo**.

- Annexe A, *RPM* décrit le gestionnaire de paquets RPM (« **RPM Package Manager** », ou RPM), qui est le système de mise en paquet utilisé par Red Hat Enterprise Linux.

PARTIE IV. SERVICES D'INFRASTRUCTURE

Cette partie fournit des informations sur la manière de configurer les services et démons, et sur la manière d'activer l'accès distant à une machine Red Hat Enterprise Linux.

CHAPITRE 9. GÉRER LES SERVICES AVEC SYSTEMD

9.1. INTRODUCTION À SYSTEMD

Systemd est un gestionnaire de systèmes et de services pour les systèmes d'exploitation Linux. Il est conçu pour être rétro-compatible avec les scripts SysV init, et fournit un certain nombre de fonctionnalités, comme le lancement en parallèle des services système pendant l'initialisation, l'activation des démons à la demande, la prise en charge des instantanés d'état du système, ou la logique de contrôle de service basée sur dépendances. Sur Red Hat Enterprise Linux 7, systemd remplace Upstart comme système init par défaut.

Systemd introduit le concept d'unités systemd (« *systemd units* »). Ces unités sont représentées par les fichiers de configuration d'unités dans l'un de répertoires qui se trouve dans Tableau 9.2, « Emplacements des fichiers d'unités systemd », et inclut des informations sur les services système, les sockets d'écoute, les instantanés d'état de système enregistré, et d'autres objets pertinents au système init. Pour obtenir la liste complète des types d'unité systemd disponibles, veuillez consulter Tableau 9.1, « Types d'unités systemd disponibles ».

Tableau 9.1. Types d'unités systemd disponibles

Type d'unité	Extension de fichier	Description
Unité du service	`.service`	Service système.
Unité cible	`.target`	Un groupe d'unités systemd.
Unité Automount	`.automount`	Un point Automount du système de fichiers.
Unité du périphérique	`.device`	Fichier du périphérique reconnu par le noyau.
Unité de montage	`.mount`	Point de montage du système de fichiers.
Unité de chemin	`.path`	Un fichier ou répertoire dans un système de fichiers.
Unité scope	`.scope`	Un processus créé de manière externe.
Unité de tranche	`.slice`	Un groupe d'unités organisées de manière hiérarchique qui gèrent des processus système.
Unité d'instantané	`.snapshot`	Un état enregistré du gestionnaire systemd.
Unité de socket	`.socket`	Un socket de communication inter-processus.
Unité swap	`.swap`	Un périphérique ou fichier swap.
Unité minuteur	`.timer`	Un minuteur systemd.

Tableau 9.2. Emplacements des fichiers d'unités systemd

Répertoire	Description
`/usr/lib/systemd/system/`	Fichiers d'unités systemd distribuées avec des paquets RPM installés.
`/run/systemd/system/`	Les fichiers d'unités systemd créées pendant l'exécution. Ce répertoire a priorité sur le répertoire de fichiers d'unités de service installées.
`/etc/systemd/system/`	Les fichiers d'unités systemd créées par `systemctl enable` ainsi que les fichiers d'unités ajoutés pour étendre un service. Ce répertoire a priorité sur le répertoire de fichiers d'unités de service installées.

9.1.1. Fonctionnalités principales

Sur Red Hat Enterprise Linux 7, le gestionnaire de systèmes et services systemd fournit les fonctionnalités principales suivantes :

- *Activation basée socket* — pendant l'initialisation, systemd crée des sockets d'écoute pour tous les services système qui prennent en charge ce type d'activation, et passe les sockets à ces services dès qu'ils sont lancés. Ceci permet à systemd de lancer les services en parallèle, mais rend également possible le redémarrage d'un service sans perdre de message qui lui aurait été envoyé pendant son indisponibilité : le socket correspondant reste accessible et tous les messages sont mis en file d'attente.

  Systemd utilise des unités de socket « *socket units* » pour une activation basée socket.

- *Activation basée sur Bus* — les services système qui utilisent D-Bus pour les communications inter-processus peuvent être lancés à la demande la première fois qu'une application cliente tente de communiquer avec eux. Systemd utilise des fichiers de service « *D-Bus service files* » pour une activation basée bus.

- *Activation basée périphérique* — les services système qui prennent en charge l'activation basée périphérique peuvent être lancés à la demande lorsqu'un type particulier de matériel physique est branché ou devient disponible. Systemd utilise des unités de périphérique « *device units* » pour l'activation basée périphérique.

- *Activation basée chemin* — les services système qui prennent en charge l'activation basée chemin peuvent être lancés à la demande lorsqu'un fichier ou répertoire particulier change d'état. Systemd utilise les unités de chemin « *path units* » pour l'activation basée chemin.

- *Instantanés d'état système* — Systemd peut temporairement enregistrer l'état actuel de toutes les unités ou restaurer un état précédent du système à partir d'un instantané créé dynamiquement. Pour stocker l'état actuel du système, systemd utilise des unités d'instantané « *snapshot units* » créées dynamiquement.

- *Gestion des points de montage et des points Automount* — Systemd surveille et gère tous les points de montage et d'Automount. Systemd utilise « *mount units* » pour les points de montage et « *automount units* » pour les points d'Automount.

- *Parallélisation aggressive* — à cause de l'activation basée socket, systemd peut lancer des services système en parallèle une fois que tous les sockets d'écoute sont en place. Lorsque combinés aux services système qui prennent en charge l'activation à la demande, l'activation en

parallèle réduit largement le temps qu'il faut pour démarrer le système.

- *Logique d'activation d'unité transactionnelle* — avant d'activer ou de désactiver une unité, systemd calcule ses dépendances, crée une transaction temporaire, et vérifie que celle-ci soit bien cohérente. Si une transaction est incohérente, systemd tente automatiquement de la corriger et d'en supprimer les tâches non essentielles avant de rapporter une erreur.

- *Rétro-compatibilité avec SysV init* — Systemd prend en charge les scripts SysV init comme décrit dans la spécification « *Linux Standard Base Core Specification* », ce qui facilite le chemin de mise à niveau des unités de service Systemd.

9.1.2. Changements de compatibilité

Le gestionnaire de systèmes et Systemd sont conçus pour être principalement compatibles avec SysV init et Upstart. Ci-dessous figurent les principaux changements de compatibilité par rapport à la dernière version majeure du système Red Hat Enterprise Linux :

- Systemd n'offre qu'une prise en charge limitée des niveaux d'exécution. Il fournit un certain nombre d'unités cibles pouvant être directement mappées à ces niveaux d'exécution et est également distribué avec l'ancienne commande **`runlevel`** pour des raisons de compatibilité. Les cibles Systemd ne peuvent pas toutes être directement mappées aux niveaux d'exécution. Par conséquent, cette commande peut retourner **`N`**, pour indiquer un niveau d'exécution inconnu. Il est recommandé d'éviter d'utiliser la commande **`runlevel`** si possible.

  Pour obtenir davantage d'informations sur les cibles Systemd et leurs comparaisons aux niveaux d'exécution, veuillez consulter la Section 9.3, « Travailler avec des cibles Systemd ».

- L'utilitaire **`systemctl`** ne prend pas en charge les commandes personnalisées. En plus des commandes standards, comme **`start`**, **`stop`**, et **`status`**, les auteurs de scripts SysV init peuvent implémenter un nombre de commandes arbitraires afin de fournir des fonctionnalités supplémentaires. Par exemple, le script init d'**`iptables`** sur Red Hat Enterprise Linux 6 peut être exécuté avec la commande **`panic`**, qui active immédiatement le mode de panique et reconfigure le système pour ne plus recevoir de paquets entrants et ne plus pouvoir envoyer de paquets sortants. Ceci n'est pas pris en charge sur Systemd et **`systemctl`** n'accepte que les commandes documentées.

  Pour obtenir davantage d'informations sur l'utilitaire **`systemctl`** et sa comparaison avec l'ancien utilitaire **`service`**, veuillez consulter la Section 9.2, « Gérer les services système ».

- L'utilitaire **`systemctl`** ne communique pas avec les services qui n'ont pas été lancés par Systemd. Lorsque Systemd lance un service système, il stocket l'ID de son processus principal afin d'en garder la trace. L'utilitaire **`systemctl`** utilise ensuite ce PID pour effectuer des requêtes et pour gérer le service. Par conséquent, si un utilisateur lance un démon particulier directement sur la ligne de commande, **`systemctl`** sera incapable de déterminer son statut actuel ou de l'arrêter.

- Systemd arrête uniquement les services en cours d'exécution. Précédemment, lorsque la séquence de fermeture était initiée, Red Hat Enterprise Linux 6 et les versions plus anciennes du système utilisaient des liens symboliques situés dans le répertoire **`/etc/rc0.d/`** pour arrêter tous les services systèmes, quel que soit leur statut. Avec Systemd, seuls les services en cours d'exécution sont arrêtés pendant la fermeture.

- Les services système sont incapables de lire le flux d'entrées standard. Lorsque Systemd lance un service, il connecte son entrée standard sur **`/dev/null`** pour empêcher toute interaction avec l'utilisateur.

- Les services système n'héritent d'aucun contexte (comme les variables d'environnement **HOME** et **PATH**) provenant de l'utilisateur les invoquant et de sa session. Chaque service est exécuté dans un contexte d'exécution épuré.
- Lors du chargement d'un script SysV init, systemd lit les informations de dépendance codées dans l'en-tête LSB (« Linux Standard Base ») et les interprète pendant l'exécution.
- Toutes les opérations sur les unités de service sont sujettes à un délai par défaut de 5 minutes pour empêcher à tout service malfonctionnant de geler le système. Cette valeur est codée en dur pour les services générés à partir d'initscripts et ne peut pas être modifiée. Toutefois, des fichiers de configuration individuels peuvent servir à spécifier une valeur de délai d'attente plus longue pour chaque service individuel, voir Exemple 9.21, « Modifier la limite du délai d'attente »

Pour obtenir une liste détaillée des changements de compatibilité apportés avec systemd, veuillez consulter le Guide de planification de migration de Red Hat Enterprise Linux 7.

9.2. GÉRER LES SERVICES SYSTÈME

NOTE

Afin de gagner en expertise, vous serez sans doute intéressé par le cours de formation Red Hat System Administration II (RH134).

Les versions précédentes de Red Hat Enterprise Linux, qui étaient distribuées avec SysV init ou Upstart, utilisaient des scripts init (« *init scripts* ») situés dans le répertoire **/etc/rc.d/init.d/**. Ces scripts init étaient habituellement écrits en Bash, et autorisaient l'administrateur systèmes à contrôler l'état des services et des démons dans leurs systèmes. Sur Red Hat Enterprise Linux 7, ces scripts init ont été remplacés par des unités de service (« *service units* »).

Les unités de service se terminent par l'extension de fichier **.service** et ont un but similaire à celui des scripts init. Pour afficher, lancer, arrêter, redémarrer, activer ou désactiver des services système, veuillez utiliser la commande **systemctl** décrite dans Tableau 9.3, « Comparaison du service Utility avec systemctl », Tableau 9.4, « Comparaison de l'utilitaire chkconfig avec systemctl », et un peu plus bas dans cette section. Les commandes **service** et **chkconfig** sont toujours disponibles dans le système et fonctionnent comme prévu, mais sont uniquement incluses pour des raisons de compatibilité et doivent être évitées.

Tableau 9.3. Comparaison du service Utility avec systemctl

service	systemctl	Description
service *nom* start	**systemctl start *nom*.service**	Lance un service.
service *nom* stop	**systemctl stop *nom*.service**	Arrête un service.
service *nom* restart	**systemctl restart *nom*.service**	Redémarre un service.
service *nom* condrestart	**systemctl try-restart *nom*.service**	Redémarre un service uniquement s'il est en cours d'exécution.

service	systemctl	Description
service *nom* reload	**systemctl reload *nom*.service**	Recharge la configuration.
service *nom*status	**systemctl status *nom*.service** **systemctl is-active *nom*.service**	Vérifie si un service est en cours d'exécution.
service --status-all	**systemctl list-units --type service --all**	Affiche le statut de tous les services.

Tableau 9.4. Comparaison de l'utilitaire chkconfig avec systemctl

chkconfig	systemctl	Description
chkconfig *nom* on	**systemctl enable *nom*.service**	Active un service.
chkconfig *nom* off	**systemctl disable *nom*.service**	Désactive un service.
chkconfig --list *nom*	**systemctl status *nom*.service** **systemctl is-enabled *nom*.service**	Vérifie si un service est activé.
chkconfig --list	**systemctl list-unit-files --type service**	Répertorie tous les services et vérifie s'ils sont activés.
chkconfig --list	**systemctl list-dependencies --after**	Répertorie les services qui doivent démarrer avant l'unité spécifiée.
chkconfig --list	**systemctl list-dependencies --before**	Répertorie les services qui doivent démarrer aprés l'unité spécifiée.

Spécifier les unités de service

Par souci de clarté, tous les exemples de commande dans le reste de cette section utilisent des noms d'unité complets avec l'extension de fichier **.service**. Exemple :

```
~]# systemctl stop nfs-server.service
```

Cependant, il est possible d'omettre cette extension de fichier, dans lequel cas, l'utilitaire **systemctl** assume que l'argument suppose qu'il s'agit d'une unité de service. La commande suivante équivaut à celle se trouvant ci-dessus :

```
~]# systemctl stop nfs-server
```

De plus, certaines unités ont des alias correspondants. Ces noms peuvent avoir des noms plus courts que les unités, qui peuvent être utilisés à la place des noms d'unité. Pour trouver les alias pouvant être utilisés pour une unité particulière :

```
~]# systemctl show nfs-server.service -p Names
```

9.2.1. Répertorier les services

Pour répertorier toutes les unités de service actuellement chargées, veuillez saisir ce qui suit dans une invite de shell :

```
systemctl list-units --type service
```

Cette commande affiche le nom complet de chaque fichier d'unités de service (**UNIT**) suivi d'une note indiquant si le ficiers d'unités a été chargée (**LOAD**), son état d'activation de haut niveau (**ACTIVE**) et de bas niveau (**SUB**), ainsi qu'une courte description (**DESCRIPTION**).

Par défaut, la commande **systemctl list-units** affiche uniquement les unités actives. Si vous souhaitez afficher toutes les unités chargées, quel que soit leur état, veuillez exécuter cette commande avec l'option de ligne de commande **--all** ou **-a** :

```
systemctl list-units --type service --all
```

Vous pouvez également répertorier toutes les unités de service disponibles pour voir si elles sont activées. Pour faire cela, veuillez saisir :

```
systemctl list-unit-files --type service
```

Cette commande affiche le nom complet de chaque unité de service (**UNIT FILE**) suivi d'informations indiquant si l'unité de service est activée ou non (**STATE**). Pour obtenir des informations sur la manière de déterminer le statut des unités de service individuelles, veuillez consulter la Section 9.2.2, « Afficher le statut du service ».

Exemple 9.1. Répertorier les services

Pour répertorier toutes les unités de service actuellement chargées, veuillez exécuter la commande suivante :

```
~]$ systemctl list-units --type service
UNIT                            LOAD   ACTIVE SUB     DESCRIPTION
abrt-ccpp.service               loaded active exited  Install ABRT
coredump hook
abrt-oops.service               loaded active running ABRT kernel log
watcher
abrt-vmcore.service             loaded active exited  Harvest vmcores
for ABRT
abrt-xorg.service               loaded active running ABRT Xorg log
watcher
abrtd.service                   loaded active running ABRT Automated Bug
Reporting Tool
...
systemd-vconsole-setup.service loaded active exited  Setup Virtual
Console
```

```
tog-pegasus.service           loaded active running OpenPegasus CIM
Server

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of
SUB.
SUB    = The low-level unit activation state, values depend on unit
type.

46 loaded units listed. Pass --all to see loaded but inactive units,
too.
To show all installed unit files use 'systemctl list-unit-files'
```

Pour répertorier tous les fichiers d'unités de service installées afin de déterminer si elles sont activées, veuillez saisir :

```
~]$ systemctl list-unit-files --type service
UNIT FILE                                   STATE
abrt-ccpp.service                           enabled
abrt-oops.service                           enabled
abrt-vmcore.service                         enabled
abrt-xorg.service                           enabled
abrtd.service                               enabled
...
wpa_supplicant.service                      disabled
ypbind.service                              disabled

208 unit files listed.
```

9.2.2. Afficher le statut du service

Pour afficher des informations détaillées sur une unité de service qui correspond à un service système, veuillez saisir ce qui suit dans une invite de shell :

```
systemctl status name.service
```

Veuillez remplacer *name* par le nom de l'unité de service que vous souhaitez inspecter (par exemple, **gdm**). Cette commande affiche le nom de l'unité de service sélectionnée suivi d'une courte description, un ou plusieurs champs décrit dans la Tableau 9.5, « Informations sur les unités de service disponibles », et si elle est exécutée par l'utilisateur **root**, les entrées de journal les plus récentes seront également incluses.

Tableau 9.5. Informations sur les unités de service disponibles

Champ	Description
Loaded	Informations indiquant si l'unité de service est chargée, le chemin absolu vers le fichier de l'unité, et une note indiquant si l'unité est activée.
Active	Informations indiquant si l'unité de service exécutée est suivie d'un horodatage.

Champ	Description
Main PID	Le PID du service système correspondant est suivi par son nom.
Status	Informations supplémentaires sur le service système correspondant.
Process	Informations supplémentaires sur les processus connexes.
CGroup	Informations supplémentaires sur les Groupes de contrôle connexes (cgroups).

Pour vérifier qu'une unité de service particulière est en cours d'exécution, veuillez exécuter la commande suivante :

```
systemctl is-active name.service
```

Similairement, pour déterminer si une unité de service particulière est activée, veuillez saisir :

```
systemctl is-enabled name.service
```

Remarquez que **systemctl is-active** et **systemctl is-enabled** retournent un statut de sortie (« exit status ») de **0** si l'unité de service spécifiée est en cours d'exécution ou si elle est activée. Pour obtenir des informations sur la manière de répertorier toutes les unités de service actuellement chargées, veuillez consulter la Section 9.2.1, « Répertorier les services ».

Exemple 9.2. Afficher le statut du service

L'unité de service du gestionnaire d'affichage GNOME se nomme **gdm.service**. Pour déterminer le statut actuel de cette unité de service, veuillez saisir ce qui suit dans une invite de shell :

```
~]# systemctl status gdm.service
gdm.service - GNOME Display Manager
   Loaded: loaded (/usr/lib/systemd/system/gdm.service; enabled)
   Active: active (running) since Thu 2013-10-17 17:31:23 CEST; 5min ago
 Main PID: 1029 (gdm)
   CGroup: /system.slice/gdm.service
           ├─1029 /usr/sbin/gdm
           ├─1037 /usr/libexec/gdm-simple-slave --display-id
/org/gno...
           └─1047 /usr/bin/Xorg :0 -background none -verbose -auth
/r...

Oct 17 17:31:23 localhost systemd[1]: Started GNOME Display Manager.
```

Exemple 9.3. Pour afficher les services qui doivent démarrer avant un service.

Pour déterminer quels services doivent démarrer avant le service indiqué, saisir ce qui suit dans une invite de shell :

```
~]# systemctl list-dependencies --after gdm.service
```

```
gdm.service
├─dbus.socket
├─getty@tty1.service
├─livesys.service
├─plymouth-quit.service
├─system.slice
├─systemd-journald.socket
├─systemd-user-sessions.service
└─basic.target[sortie tronquée]
```

Exemple 9.4. Pour afficher les services qui doivent démarrer après un service.

Pour déterminer quels services doivent démarrer après le service indiqué, saisir ce qui suit dans une invite de shell :

```
~]# systemctl list-dependencies --before gdm.service
gdm.service
├─dracut-shutdown.service
├─graphical.target
│ ├─systemd-readahead-done.service
│ ├─systemd-readahead-done.timer
│ └─systemd-update-utmp-runlevel.service
└─shutdown.target
  ├─systemd-reboot.service
  └─final.target
    └─systemd-reboot.service
```

9.2.3. Lancer un service

Pour lancer une unité de service qui correspond à un service système, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl start name.service
```

Veuillez remplacer *name* par le nom de l'unité de service que vous souhaitez lancer (par exemple, **gdm**). Cette commande lance l'unité de service sélectionnée dans la session actuelle. Pour obtenir des informations sur la manière d'activer une unité de service pour qu'elle soit lancée pendant l'initialisation, veuillez consulter la Section 9.2.6, « Activer un service ». Pour obtenir des informations sur la façon de déterminer le statut d'une unité de service particulière, veuillez consulter la Section 9.2.2, « Afficher le statut du service ».

Exemple 9.5. Lancer un service

L'unité de service pour le serveur HTTP Apache est nommé **httpd.service**. Pour activer cette unité de service et lancer le démon **httpd** dans la session actuelle, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl start httpd.service
```

9.2.4. Arrêter un service

Pour arrêter une unité de service qui correspond à un service système, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl stop name.service
```

Veuillez remplacer *name* par le nom de l'unité de service que vous souhaitez arrêter (par exemple, **bluetooth**). Cette commande arrête l'unité de service sélectionnée dans la session actuelle. Pour obtenir des informations sur la manière de désactiver une unité de service pour l'empêcher d'être lancée pendant l'initialisation, veuillez consulter la Section 9.2.7, « Désactiver un service ». Pour obtenir des informations sur comment déterminer le statut d'une unité de service particulière, veuillez consulter la Section 9.2.2, « Afficher le statut du service ».

Exemple 9.6. Arrêter un service

L'unité de service du démon **bluetoothd** est nommée **bluetooth.service**. Pour désactiver cette unité de service et arrêter le démon **bluetoothd** dans la session actuelle, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl stop bluetooth.service
```

9.2.5. Redémarrer un service

Pour redémarrer une unité de service qui correspond à un service système, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl restart name.service
```

Veuillez remplacer *name* par le nom de l'unité de service que vous souhaitez redémarrer (par exemple, **httpd**). Cette commande arrête l'unité de service sélectionnée dans la session actuelle et la redémarre immédiatement. De manière plus importante, si l'unité de service n'est pas en cours d'exécution, cette commande la lancera également. Pour ordonner à Systemd de redémarrer une unité de service uniquement si le service correspondant est déjà en cours d'exécution, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
systemctl try-restart name.service
```

Certains services système permettent de recharger leur configuration sans interrompre leur exécution. Pour faire cela, en tant qu'utilisateur **root**, veuillez saisir :

```
systemctl reload name.service
```

Remarquez que les services système qui ne prennent pas en charge cette fonctionnalité ignoreront simplement cette commande. Pour plus de commodité, la commande **systemctl** prend également en charge les commandes **reload-or-restart** et **reload-or-try-restart** qui redémarrent de tels services. Pour obtenir des informations sur la manière de déterminer le statut d'une certaine unité de service, veuillez consulter la Section 9.2.2, « Afficher le statut du service ».

Exemple 9.7. Redémarrer un service

Pour empêcher les utilisateurs de rencontrer des messages d'erreur non nécessaires ou des pages web dont le rendu est partiel, le serveur HTTP Apache vous permet de modifier et de recharger sa configuration sans avoir à le redémarrer et à interrompre les requêtes activement traitées. Pour cela, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl reload httpd.service
```

9.2.6. Activer un service

Pour configurer une unité de service qui correspond à un service système afin qu'elle soit automatiquement lancée pendant l'initialisation, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl enable name.service
```

Veuillez remplacer *name* par le nom de l'unité de service que vous souhaitez activer (par exemple, **httpd**). Cette commande lit la section **[Install]** de l'unité de service sélectionnée et crée les liens symboliques appropriés vers le fichier **/usr/lib/systemd/system/*name*.service** dans le répertoire **/etc/systemd/system/** et ses sous-répertoires. Cependant, cette commande ne réécrit pas les liens qui existent déjà. Si vous souhaitez vous assurer que les liens symboliques soient créés à nouveau, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
systemctl reenable name.service
```

Cette commande désactive l'unité de service sélectionnée et la réactive immédiatement. Pour obtenir des informations sur la manière de déterminer si une unité de service particulière est autorisée à se lancer pendant l'initialisation, veuillez consulter la Section 9.2.2, « Afficher le statut du service ». Pour obtenir des informations sur la manière de lancer un service dans la session actuelle, veuillez consulter la Section 9.2.3, « Lancer un service ».

Exemple 9.8. Activer un service

Pour configurer le serveur HTTP Apache de manière à ce qu'il puisse être lancé automatiquement pendant l'initialisation, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl enable httpd.service
Created symlink from /etc/systemd/system/multi-
user.target.wants/httpd.service to
/usr/lib/systemd/system/httpd.service.
```

9.2.7. Désactiver un service

Pour empêcher une unité de service qui correspond à un service système d'être automatiquement lancée pendant l'initialisation, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl disable name.service
```

Veuillez remplacer *name* par le nom de l'unité de service que vous souhaitez désactiver (par exemple, **bluetooth**). Cette commande lit la section **[Install]** de l'unité de service sélectionnée et supprime

les liens symboliques appropriés pointant vers le fichier **/usr/lib/systemd/system/*name*.service** du répertoire **/etc/systemd/system/** et de ses sous-répertoires. En outre, vous pouvez masquer toute unité de service pour l'empêcher d'être lancée manuellement ou par un autre service. Pour faire cela, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
systemctl mask name.service
```

Cette commande remplace le fichier **/etc/systemd/system/*name*.service** par un lien symbolique pointant vers **/dev/null**, ce qui rend le fichier de l'unité inaccessible à Systemd. Pour inverser cette action et démasquer une unité de service, veuillez saisir en tant qu'utilisateur **root** :

```
systemctl unmask name.service
```

Pour obtenir des informations sur la manière de déterminer si une unité de service particulière est autorisée à être lancée en cours d'initialisation, veuillez consulter la Section 9.2.2, « Afficher le statut du service ». Pour obtenir des informations sur la manière d'arrêter un service dans la session actuelle, veuillez consulter la Section 9.2.4, « Arrêter un service ».

Exemple 9.9. Désactiver un service

L'Exemple 9.6, « Arrêter un service » illustre comment arrêter l'unité **bluetooth.service** dans la session actuelle. Pour empêcher cette unité de service d'être lancée pendant l'initialisation, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl disable bluetooth.service
Removed symlink
/etc/systemd/system/bluetooth.target.wants/bluetooth.service.
Removed symlink /etc/systemd/system/dbus-org.bluez.service.
```

9.3. TRAVAILLER AVEC DES CIBLES SYSTEMD

Les versions précédentes de Red Hat Enterprise Linux, qui étaient distribuées avec SysV init ou Upstart, implémentaient un ensemble prédéfini de niveaux d'exécution (« *runlevels* ») qui représentaient des modes d'opération spécifiques. Ces niveaux d'exécution étaient numérotés de 0 à 6 et étaient définis par une sélection de services système à exécuter lorsqu'un niveau d'exécution particulier était activé par l'administrateur systèmes. Sur Red Hat Enterprise Linux 7, le concept des niveaux d'exécution a été remplacé par les *cibles Systemd*.

Les cibles Systemd sont représentées par des unités de cible (« *target units* »). Les unités de cible se terminent par l'extension de fichier **.target** et leur unique but consiste à regrouper d'autres unités Systemd dans une chaîne de dépendances. Par exemple, l'unité **graphical.target**, qui est utilisée pour lancer une session graphique, lance des services système comme le gestionnaire d'affichage GNOME (**gdm.service**) ou le services des comptes (**accounts-daemon.service**) et active également l'unité **multi-user.target**. De manière similaire, l'unité **multi-user.target** lance d'autres services système essentiels, tels que NetworkManager (**NetworkManager.service**) ou D-Bus (**dbus.service**) et active une autre unité cible nommée **basic.target**.

Red Hat Enterprise Linux 7 est distribué avec un certain nombre de cibles prédéfinies plus ou moins similaires à l'ensemble standard des niveaux d'exécution des versions précédentes de ce système. Pour des raisons de compatibilité, des alias sont également fournis pour ces cibles, et les font correspondre

directement aux niveaux d'exécution SysV. La Tableau 9.6, « Comparaison des niveaux d'exécution SysV avec les cibles Systemd » offre une liste complète des niveaux d'exécution SysV et des cibles Systemd correspondantes.

Tableau 9.6. Comparaison des niveaux d'exécution SysV avec les cibles Systemd

Niveau d'exécution	Unités de cible	Description
`0`	`runlevel0.target`, `poweroff.target`	Quitter et éteindre le système.
`1`	`runlevel1.target`, `rescue.target`	Installer un shell de secours.
`2`	`runlevel2.target`, `multi-user.target`	Installer un système multi-utilisateurs non graphique.
`3`	`runlevel3.target`, `multi-user.target`	Installer un système multi-utilisateurs non graphique.
`4`	`runlevel4.target`, `multi-user.target`	Installer un système multi-utilisateurs non graphique.
`5`	`runlevel5.target`, `graphical.target`	Installer un système graphique multi-utilisateurs.
`6`	`runlevel6.target`, `reboot.target`	Quitter et redémarrer le système.

Pour afficher, modifier, ou configurer des cibles Systemd, veuillez utiliser l'utilitaire **`systemctl`** comme décrit dans la Tableau 9.7, « Comparaison des commandes SysV init avec systemctl » ainsi que dans les sections ci-dessous. Les commandes **`runlevel`** et **`telinit`** sont toujours disponibles dans le système et fonctionnent comme prévu, mais ne sont incluses que pour des raisons de compatibilité et doivent être évitées.

Tableau 9.7. Comparaison des commandes SysV init avec systemctl

Ancienne commande	Nouvelle commande	Description
`runlevel`	`systemctl list-units --type target`	Répertorie les unités de cible actuellement chargées.
`telinit runlevel`	`systemctl isolate name.target`	Modifie la cible actuelle.

9.3.1. Afficher la cible par défaut

Pour déterminer l'unité de cible qui est utilisée par défaut, veuillez exécuter la commande suivante :

-

```
systemctl get-default
```

Cette commande résout le lien symbolique situé sur **/etc/systemd/system/default.target** et affiche le résultat. Pour obtenir des informations sur comment modifier la cible par défaut, veuillez consulter la Section 9.3.3, « Modifier la cible par défaut ». Pour obtenir des informations sur comment répertorier toutes les unités de cible actuellement chargées, veuillez consulter la Section 9.3.2, « Afficher la cible actuelle ».

Exemple 9.10. Afficher la cible par défaut

Pour afficher l'unité cible par défaut, veuillez saisir :

```
~]$ systemctl get-default
graphical.target
```

9.3.2. Afficher la cible actuelle

Pour répertorier toutes les unités de cible actuellement chargées, veuillez saisir la commande suivante dans une invite de shell :

```
systemctl list-units --type target
```

Cette commande affiche le nom complet de chaque unité de cible (**UNIT**) suivi d'une note indiquant si l'unité a été chargée (**LOAD**), son état d'activation de haut niveau (**ACTIVE**) et de bas niveau (**SUB**), ainsi qu'une courte description (**DESCRIPTION**).

Par défaut, la commande **systemctl list-units** affiche uniquement les unités actives. Si vous souhaitez afficher toutes les unités chargées, quel que soit leur état, veuillez exécuter cette commande avec l'option de ligne de commande **--all** ou **-a** :

```
systemctl list-units --type target --all
```

Veuillez consulter la Section 9.3.1, « Afficher la cible par défaut » pour obtenir des informations sur la manière d'afficher la cible par défaut. Pour obtenir des informations sur la manière de modifier la cible actuelle, veuillez consulter la Section 9.3.4, « Modifier la cible actuelle ».

Exemple 9.11. Afficher la cible actuelle

Pour répertorier toutes les unités de cible actuellement chargées, veuillez exécuter la commande suivante :

```
~]$ systemctl list-units --type target
UNIT                   LOAD   ACTIVE SUB    DESCRIPTION
basic.target           loaded active active Basic System
cryptsetup.target      loaded active active Encrypted Volumes
getty.target           loaded active active Login Prompts
graphical.target       loaded active active Graphical Interface
local-fs-pre.target    loaded active active Local File Systems (Pre)
local-fs.target        loaded active active Local File Systems
multi-user.target      loaded active active Multi-User System
network.target         loaded active active Network
paths.target           loaded active active Paths
```

```
remote-fs.target      loaded active active Remote File Systems
sockets.target        loaded active active Sockets
sound.target          loaded active active Sound Card
spice-vdagentd.target loaded active active Agent daemon for Spice guests
swap.target           loaded active active Swap
sysinit.target        loaded active active System Initialization
time-sync.target      loaded active active System Time Synchronized
timers.target         loaded active active Timers

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of
SUB.
SUB    = The low-level unit activation state, values depend on unit
type.

17 loaded units listed. Pass --all to see loaded but inactive units,
too.
To show all installed unit files use 'systemctl list-unit-files'.
```

9.3.3. Modifier la cible par défaut

Pour configurer le système de manière à utiliser une unité de cible différente par défaut, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl set-default name.target
```

Veuillez remplacer *name* par le nom de l'unité de cible que vous souhaitez utiliser par défaut (par exemple, **multi-user**). Cette commande remplace le fichier **/etc/systemd/system/default.target** par un lien symbolique pointant vers **/usr/lib/systemd/system/*name*.target**, où *name* est le nom de l'unité cible que vous souhaitez utiliser. Pour obtenir davantage d'informations sur la manière de modifier la cible actuelle, veuillez consulter la Section 9.3.4, « Modifier la cible actuelle ». Pour obtenir des informations sur la manière de répertorier toutes les unités de cible actuellement chargées, veuillez consulter la Section 9.3.2, « Afficher la cible actuelle ».

Exemple 9.12. Modifier la cible par défaut

Pour configurer le système de manière à utiliser l'unité **multi-user.target** par défaut, veuillez saisir la commande suivante dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl set-default multi-user.target
rm '/etc/systemd/system/default.target'
ln -s '/usr/lib/systemd/system/multi-user.target'
'/etc/systemd/system/default.target'
```

9.3.4. Modifier la cible actuelle

Pour passer à une unité de cible différente dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
systemctl isolate name.target
```

Veuillez remplacer *name* par le nom de l'unité de cible que vous souhaitez utiliser par défaut (par exemple, **multi-user**). Cette commande remplace l'unité de cible nommée *name* et toutes ses unités dépendantes, et arrête immédiatement toutes les autres. Pour obtenir des informations sur la manière de modifier la cible par défaut, veuillez consulter la Section 9.3.3, « Modifier la cible par défaut ». Pour obtenir des informations sur la manière de répertorier toutes les unités de cible actuellement chargées, veuillez consulter la Section 9.3.2, « Afficher la cible actuelle ».

Exemple 9.13. Modifier la cible actuelle

Pour éteindre l'interface utilisateur graphique et modifier l'unité en **multi-user.target** dans la session actuelle, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl isolate multi-user.target
```

9.3.5. Passer en mode de secours

Le mode de secours (« *Rescue mode* ») fournit un environnement simple utilisateur pratique, et vous permet de réparer votre système dans les cas où il est impossible d'effectuer un processus de démarrage normal. En mode de secours, le système tente de monter tous les systèmes de fichiers locaux et de lancer plusieurs services système importants, mais n'active pas d'interface réseau ou ne permet pas à d'autres d'utilisateurs de se connecter au système au même moment. Sur Red Hat Enterprise Linux 7, le mode de secours est équivalent au mode utilisateur seul (*single user mode*) et requiert le mot de passe root.

Pour modifier la cible actuelle et entrer en mode de secours dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
systemctl rescue
```

Cette commande est similaire à **systemctl isolate rescue.target**, mais elle envoie également un message informatif à tous les utilisateurs actuellement connectés au système. Pour empêcher Systemd d'envoyer ce message, veuillez exécuter cette commande avec l'option de ligne de commande **--no-wall** :

```
systemctl --no-wall rescue
```

Pour obtenir des informations sur la manière d'entrer en mode d'urgence, veuillez consulter Section 9.3.6, « Passer en mode d'urgence ».

Exemple 9.14. Passer en mode de secours

Pour entrer en mode de secours dans la session actuelle, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl rescue

Broadcast message from root@localhost on pts/0 (Fri 2013-10-25 18:23:15
CEST):

The system is going down to rescue mode NOW!
```

9.3.6. Passer en mode d'urgence

Le mode d'urgence (« *Emergency mode* ») fournit l'environnement le plus minimaliste possible et vous permet de réparer votre système même dans des situations où le système est incapable d'entrer en mode de secours. Dans le mode d'urgence, le système monte le système de fichiers root uniquement en lecture, il ne tentera pas de monter d'autre système de fichiers locaux, n'activera pas d'interface réseau et lancera quelques services essentiels. Sur Red Hat Enterprise Linux 7, le mode d'urgence requiert le mot de passe root.

Pour modifier la cible actuelle et entrer en mode d'urgence dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
systemctl emergency
```

Cette commande est similaire à **systemctl isolate emergency.target**, mais elle envoie également un message informatif à tous les utilisateurs actuellement connectés au système. Pour empêcher Systemd d'envoyer ce message, veuillez exécuter cette commande avec l'option de ligne de commande **--no-wall** :

```
systemctl --no-wall emergency
```

Pour obtenir des informations sur la manière d'entrer en mode de secours, veuillez consulter Section 9.3.5, « Passer en mode de secours ».

Exemple 9.15. Passer en mode d'urgence

Pour entrer en mode d'urgence sans envoyer de message à tous les utilisateurs actuellement connectés au système, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl --no-wall emergency
```

9.4. ARRÊTER, SUSPENDRE, ET METTRE LE SYSTÈME EN HIBERNATION

Sur Red Hat Enterprise Linux 7, l'utilitaire **systemctl** remplace un certain nombre de commandes de gestion de l'alimentation utilisées dans des versions précédentes du système Red Hat Enterprise Linux. Les commandes répertoriées dans la Tableau 9.8, « Comparaison des commandes de gestion de l'alimentation avec systemctl » sont toujours disponibles sur le système pour des raisons de compatibilité, mais il est conseillé d'utiliser **systemctl** lorsque possible.

Tableau 9.8. Comparaison des commandes de gestion de l'alimentation avec systemctl

Ancienne commande	Nouvelle commande	Description
halt	**systemctl halt**	Arrête le système.
poweroff	**systemctl poweroff**	Met le système hors-tension.
reboot	**systemctl reboot**	Redémarre le système.

Ancienne commande	Nouvelle commande	Description
pm-suspend	**systemctl suspend**	Suspend le système.
pm-hibernate	**systemctl hibernate**	Met le système en hibernation.
pm-suspend-hybrid	**systemctl hybrid-sleep**	Met en hibernation et suspend le système.

9.4.1. Arrêter le système

L'utilitaire **systemctl** fournit des commandes de fermeture du système, mais la commande traditionnelle **shutdown** est également prise en charge. Bien que la commande **shutdown** fasse appel à l'utilitaire **systemctl** pour la fermeture, elle présente l'avantage d'intégrer un argument de temps. C'est particulièrement utile pour la maintenance et pour laisser plus de temps aux utilisateurs pour réagir à une notification de fermeture du système. La possibilité de pouvoir annuler le fermeture peut également être un plus.

Utilisation de la commande systemctl

Pour arrêter le système et mettre la machine hors-tension, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl poweroff
```

Pour arrêter et interrompre le système sans mettre la machine hors-tension, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
systemctl halt
```

Par défaut, l'exécution de l'une de ces commandes amène Systemd à envoyer un message informatif à tous les utilisateurs actuellement connectés au système. Pour empêcher Systemd d'envoyer ce message, veuillez exécuter cette commande avec l'option de ligne de commande **--no-wall**. Exemple :

```
systemctl --no-wall poweroff
```

Utilisation de la commande Shutdown

Pour arrêter le système et mettre la machine hors-tension à une heure précise, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
shutdown --poweroff hh:mm
```

avec *hh:mm* comme l'heure au format militaire (24h). Le fichier **/run/nologin** est créé 5 minutes avant la fermeture du système pour éviter les nouvelles connexions. Quand un argument de temps est utilisé, un message en option, le *wall message*, pourra être ajouté à la commande.

Pour arrêter et interrompre le système au bout d'un moment sans mettre la machine hors-tension, veuillez exécuter la commande sous le format suivant en tant qu'utilisateur **root** :

```
shutdown --halt +m
```

avec *+m* comme durée en minutes. Le mot clé **now** est un alias de **+0**.

Une fermeture à venir pourra être annulée par l'utilisateur **root** comme suit :

```
shutdown -c
```

Voir la page man de **shutdown(8)** pour obtenir des options de commandes supplémentaires.

9.4.2. Redémarrer le système

Pour redémarrer le système, exécutez la commande suivante en tant qu'utilisateur **root** :

```
systemctl reboot
```

Par défaut, cette commande amène Systemd à envoyer un message informatif à tous les utilisateurs actuellement connectés au système. Pour empêcher Systemd d'envoyer ce message, veuillez exécuter cette commande avec l'option de ligne de commande **--no-wall** :

```
systemctl --no-wall reboot
```

9.4.3. Suspendre le système

Pour suspendre le système, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl suspend
```

Cette commande enregistre l'état du système dans la RAM et, à l'exception du module RAM, met hors-tension la plupart des périphériques dans la machine. Lorsque vous rallumez la machine, le système restaure son état à partir la RAM sans avoir à redémarrer. Comme l'état du système est enregistré dans la RAM et non sur le disque dur, restaurer le système à partir du mode de suspension est bien plus rapide que de le restaurer suite à une hibernation. En revanche, un état de système en suspension est également vulnérable aux pannes d'électricité.

Pour obtenir des informations sur la manière de mettre le système en hibernation, veuillez consulter la Section 9.4.4, « Hiberner le système ».

9.4.4. Hiberner le système

Pour hiberner le système, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl hibernate
```

Cette commande enregistre l'état du système sur le disque dur et met la machine hors-tension. Lorsque vous rallumez la machine, le système restaure son état à partir des données enregistrées sans avoir à démarrer à nouveau. Comme l'état du système est enregistré sur le disque dur et non sur la RAM, la machine n'a pas besoin de maintenir une alimentation électrique sur le module RAM. En revanche, restaurer le système suite à une hibernation est bien plus lent que de le restaurer suite au mode de suspension.

Pour hiberner et suspendre le système, exécutez la commande suivante en tant qu'utilisateur **root** :

```
systemctl hybrid-sleep
```

Pour obtenir des informations sur la manière de suspendre le système, veuillez consulter la Section 9.4.3, « Suspendre le système ».

9.5. CONTRÔLER SYSTEMD SUR UNE MACHINE DISTANTE

En plus de contrôler le système systemd et le gestionnaire de services localement, l'utilitaire **systemctl** permet également d'interagir avec systemd pendant une exécution sur une machine distante à traver le protocole SSH. Si le service **sshd** est en cours d'exécution sur la machine distante, vous pourrez vous connecter à cette machine en exécutant la commande **systemctl** avec l'option de ligne de commande **--host** ou **-H** :

```
systemctl --host user_name@host_name command
```

Remplacez *user_name* par le nom de l'utilisateur distant, *host_name* par le nom d'hôte de la machine, et **command** par n'importe quelle commande **systemctl** décrite ci-dessus. Remarquez que la machine distante doit être configurée afin d'autoriser l'accès distant à l'utilisateur sélectionné par le protocole SSH. Pour obtenir davantage d'informations sur la manière de configurer un serveur SSH, veuillez consulter le Chapitre 10, *OpenSSH*.

Exemple 9.16. Gestion à distance

Pour se connecter à une machine distante nommée **server-01.example.com** en tant qu'utilisateur **root** et déterminer le statut actuel de l'unité **httpd.service**, veuillez saisir ce qui suit dans une invite de shell :

```
~]$ systemctl -H root@server-01.example.com status httpd.service
>>>>>>> systemd unit files -- update
root@server-01.example.com's password:
httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled)
   Active: active (running) since Fri 2013-11-01 13:58:56 CET; 2h 48min
ago
 Main PID: 649
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic:
0 B/sec"
   CGroup: /system.slice/httpd.service
```

9.6. CRÉER ET MODIFIER DES FICHIERS D'UNITÉ SYSTEMD

Un fichier d'unité contient des directives de configuration qui décrivent l'unité et définissent son comportement. Plusieurs commandes **systemctl** fonctionnent avec des fichiers d'unité dans l'arrière-plan. Pour faire des ajustements plus précis, l'administrateur systèmes doit modifier ou créer des fichiers d'unité manuellement. La Tableau 9.2, « Emplacements des fichiers d'unités systemd » répertorie trois répertoires principaux sur lesquels les fichiers d'unités sont stockés sur le système, le répertoire **/etc/systemd/system/** est réservé aux fichiers d'unité créés ou personnalisés par l'administrateur systèmes.

Les noms des fichiers d'unité prennent la forme suivante :

```
unit_name.type_extension
```

Ici, *unit_name* correspond au nom de l'unité et *type_extension* identifie le type de l'unité, veuillez consulter la Tableau 9.1, « Types d'unités systemd disponibles » pour afficher la liste complète des types d'unité. Par exemple, une unité **sshd.service** et une unité **sshd.socket** sont habituellement présentes sur votre système.

Les fichiers d'unité peuvent être complétés d'un répertoire pour des fichiers de configuration supplémentaires. Par exemple, pour ajouter des options de configuration personnalisées à **sshd.service**, veuillez créer le fichier **sshd.service.d/custom.conf** et insérez-y les directives supplémentaires. Pour obtenir davantage d'informations sur les répertoires de configuration, veuillez consulter la Section 9.6.4, « Modifier les fichiers d'unité existants ».

Les répertoires **sshd.service.wants/** et **sshd.service.requires/** peuvent également être créés. Ces répertoires contiennent des liens symboliques vers les fichiers d'unités qui sont des dépendances du service **sshd**. Les liens symboliques sont automatiquement créés pendant l'installation selon les options du fichiers de l'unité [Install] (veuillez consulter la Tableau 9.11, « Options importantes de la section [Install] ») ou pendant le temps d'exécution basé sur les options [Unit] (veuillez consulter la Tableau 9.9, « Options importantes de la section [Unit] »). Il est également possible de créer ces répertoires et les liens symboliques manuellement.

De nombreuses options de fichiers d'unité peuvent être définies en utilisant des *spécificateurs d'unités* – des chaînes de caractères génériques dynamiquement remplacées par des paramètres d'unités lorsque le fichier d'unité est chargé. Ceci permet la création de fichiers d'unité génériques qui serviront de modèle pour générer des unités instanciées. Veuillez consulter la Section 9.6.5, « Travailler avec des unités instanciées » pour obtenir plus de détails.

9.6.1. Comprendre la structure des fichiers d'unité

Les fichiers d'unité consistent habituellement de trois sections :

- [Unit] — contient des options génériques qui ne sont pas dépendantes du type de l'unité. Ces options fournissent une description de l'unité, spécifient le comportement de l'unité, et définissent les dépendances avec d'autres unités. Pour une liste des options [Unit] les plus fréquemment utilisées, veuillez consulter la Tableau 9.9, « Options importantes de la section [Unit] ».
- [*unit type*] — si une unité possède des directives spécifiques au type, celles-ci seront regroupées dans une section nommée par le type d'unité. Par exemple, les fichiers de l'unité de service contiennent la section [Service], veuillez consulter la Tableau 9.10, « Options importantes de la section [Service] » pour voir les options [Service] les plus fréquemment utilisées.
- [Install] — contient des informations sur l'installation de l'unité utilisée par les commandes **systemctl enable** et **disable**, veuillez consulter la Tableau 9.11, « Options importantes de la section [Install] » pour voir une liste des options [Install].

Tableau 9.9. Options importantes de la section [Unit]

Option[a]	Description
Description	Description significative de l'unité. En tant qu'exemple, le texte est affiché dans la sortie de la commande **systemctl status**.
Documentation	Fournit une liste des URI référençant la documentation de l'unité.

Option[a]	Description
After[b]	Définit l'ordre dans lequel les unités sont lancées. L'unité est lancée uniquement après l'activation des unités spécifiées dans **After**. Contrairement à **Requires**, **After** n'active pas explicitement les unités spécifiées. L'option **Before** offre une fonctionnalité contraire à **After**.
Requires	Configure les dépendances sur d'autres unités. Les unités répertoriées dans **Requires** sont activées ensembles avec l'unité. Si le lancement de l'une des unités requises échoue, l'unité n'est pas activée.
Wants	Configure les dépendances plus faibles que **Requires**. Si l'une des unités répertoriées ne démarre pas, cela n'aura pas d'impact sur l'activation de l'unité. C'est la méthode recommandée pour établir des dépendances d'unité personnalisées.
Conflicts	Configure des dépendances négatives, à l'opposé de **Requires**.

[a] Pour une liste complète des options configurables dans la section [Unit], veuillez consulter la page man de **systemd.unit(5)**.

[b] Dans la plupart des cas, il est suffisant de ne déterminer que les dépendances d'ordonnancement qu'avec les options de fichier **After** et **Before**. Si vous définissez aussi une dépendance avec **Wants** (conseillé) ou **Requires**, la dépendance d'ordonnancement devra toujours être spécifiée. C'est parce que l'ordonnancement et les exigences de dépendances fonctionnent indépendamment.

Tableau 9.10. Options importantes de la section [Service]

Option[a]	Description
Type	Configure le type de démarrage de processus d'unité qui affecte la fonctionnalité d'**ExecStart** et des options connexes. L'une des options suivantes : • **simple** – valeur par défaut. Le processus lancé par **ExecStart** est le processus principal du service. • **forking** – le processus lancé par **ExecStart** engendre un processus enfant qui devient le processus principal du service. Le processus parent s'arrête lorsque le startup est terminé. • **oneshot** – ce type est similaire à **simple**, mais le processus s'arrête avant de lancer les unités suivantes. • **dbus** – ce type est similaire à **simple**, mais les unités suivantes sont lancées uniquement après que le processus principal ait obtenu un nom D-Bus. • **notify** – ce type est similaire à **simple**, mais les unités suivante sont lancées uniquement après l'envoi d'un message de notification via la fonction sd_notify(). • **idle** – similaire à **simple**, l'exécution du binaire du service est retardée jusqu'à ce que toutes les tâches soient terminées, ce qui permet d'éviter de mélanger la sortie du statut avec la sortie shell des services.

Option[a]	Description
ExecStart	Spécifie les commandes ou scripts à exécuter lorsque l'unité est lancée. **ExecStartPre** et **ExecStartPost** spécifient des commandes personnalisées à exécuter avant et après **ExecStart**. **Type=oneshot** permet de spécifier des commandes multiples personnalisées exécutées de manière séquentielle par la suite.
ExecStop	Spécifie les commandes ou scripts à exécuter lorsque l'unité est arrêtée.
ExecReload	Spécifie les commandes ou scripts à exécuter lorsque l'unité est rechargée.
Restart	Avec cette option activée, le service est redémarré après que son processus se soit arrêté, à l'exception d'un arrêt gracieux avec la commande **systemctl**.
RemainAfterExit	Si défini sur True, le service est considéré comme actif, même lorsque tous ses processus sont arrêtés. La valeur par défaut est False. Cette option est particulièrement utile si **Type=oneshot** est configuré.

[a] Pour une liste complète des options configurables dans la section [Service], veuillez consulter la page man de **systemd.service(5)**.

Tableau 9.11. Options importantes de la section [Install]

Option[a]	Description
Alias	Fournit une liste de noms supplémentaires de l'unité séparés par des espaces. La plupart des commandes **systemctl**, sauf **systemctl enable**, peuvent utiliser des alias à la place du nom de l'unité.
RequiredBy	Une liste des unités qui dépendent de l'unité. Lorsque cette unité est activée, les unités répertoriées dans **RequiredBy** obtiennent une dépendance **Require** de l'unité.
WantedBy	Une liste des unités qui dépendent faiblement de l'unité. Lorsque cette unité est activée, les unités répertoriées dans **WantedBy** obtiennent une dépendance **Want** de l'unité.
Also	Indique une liste des unités à installer ou désinstaller avec l'unité.
DefaultInstance	Limitée aux unités instanciées, cette option indique l'instance par défaut pour laquelle l'unité est activée. Veuillez consulter la Section 9.6.5, « Travailler avec des unités instanciées »

[a] Pour voir une liste complète des options configurables dans la section [Install], veuillez consulter la page man de **systemd.unit(5)**.

Une gamme entière d'options qui peuvent être utilisées pour régler de manière détaillée la configuration de l'unité, l'Exemple 9.17, « Fichier d'unité postfix.service » montre un exemple d'unité de service

installée sur le système. En outre, les options de fichier d'unité peuvent être définies d'une manière permettant la création dynamique d'unités, comme décrit dans la Section 9.6.5, « Travailler avec des unités instanciées ».

Exemple 9.17. Fichier d'unité postfix.service

Ci-dessous figure le contenu du fichier de l'unité **`/usr/lib/systemd/system/postifix.service`** tel qu'il est fourni par le paquet postfix :

```
[Unit]
Description=Postfix Mail Transport Agent
After=syslog.target network.target
Conflicts=sendmail.service exim.service

[Service]
Type=forking
PIDFile=/var/spool/postfix/pid/master.pid
EnvironmentFile=-/etc/sysconfig/network
ExecStartPre=-/usr/libexec/postfix/aliasesdb
ExecStartPre=-/usr/libexec/postfix/chroot-update
ExecStart=/usr/sbin/postfix start
ExecReload=/usr/sbin/postfix reload
ExecStop=/usr/sbin/postfix stop

[Install]
WantedBy=multi-user.target
```

La section [Unit] décrit le service, spécifie les dépendances d'ordre, ainsi que les unités contradictoires. Dans [Service], une séquence de scripts personnalisés est spécifiée pour être exécutée pendant l'activation de l'unité, pendant l'arrêt, et le rechargement. **`EnvironmentFile`** désigne l'emplacement où les variables d'environnement du service sont définies, **`PIDFile`** spécifie un PID stable pour le processus principale du service. Finalement, la section [Install] répertorie les unités qui dépendent de ce service.

9.6.2. Créer des fichiers d'unité personnalisés

Il existe plusieurs cas dans lesquels il est nécessaire de créer des fichiers d'unité depuis le début : vous pourriez devoir exécuter un démon personnalisé, créer une seconde instance d'un service existant (comme dans l'Exemple 9.19, « Création d'une seconde instance du service sshd »), ou importer un script init SysV (consultez la Section 9.6.3, « Convertir des scripts init SysV en fichiers d'unité »). En revanche, si vous souhaitez simplement modifier ou étendre le comportement d'une unité existante, veuillez utiliser les instructions de la Section 9.6.4, « Modifier les fichiers d'unité existants ». La procédure suivante décrit le processus général de création d'un service personnalisé :

1. Préparez le fichier exécutable avec le service personnalisé. Il peut s'agir d'un script créé et personnalisé, ou d'un exécutable remis par un fournisseur de logiciels. Si requis, veuillez préparer un fichier PID pour contenir un PID constant pour le processus principal du service personnalisé. Il est également possible d'inclure des fichiers d'environnement pour stocker des variables shell pour le service. Assurez-vous que le script source soit exécutable (en exécutant **`chmod a+x`**) et qu'il ne soit pas interactif.

2. Créez un fichier d'unité dans le répertoire **`/etc/systemd/system/`** et assurez-vous qu'il possède les permissions de fichier correctes. Veuillez exécuter en tant qu'utilisateur **`root`** :

   ■

```
touch /etc/systemd/system/name.service
chmod 664 /etc/systemd/system/name.service
```

Remplacez *name* par le nom du service à créer. Remarquez que le fichier n'a pas besoin d'être exécutable.

3. Ouvrez le fichier ***name*.service** créé dans l'étape précédente, et ajoutez les options de configuration du service. Une variété d'options peut être utilisée selon le type de service que vous souhaitez créer, consulter Section 9.6.1, « Comprendre la structure des fichiers d'unité ». Voici un exemple de configuration d'unité pour un service concernant le réseau :

```
[Unit]
Description=service_description
After=network.target

[Service]
ExecStart=path_to_executable
Type=forking
PIDFile=path_to_pidfile

[Install]
WantedBy=default.target
```

Quand :

- *service_description* est une description informative affichée dans les fichiers journaux et dans la sortie de la commande **`systemctl status`**.
- le paramètre **`After`** permet de s'assurer que le service est démarré uniquement après l'exécution du réseau. Ajoutez une liste séparée par des espaces d'autres services ou cibles connexes.
- *path_to_executable* correspond au chemin vers l'exécutable du service.
- **`Type=forking`** est utilisé pour les démons effectuant l'appel système « fork ». Le processus principal du service est créé avec le PID spécifié dans *path_to_pidfile*. D'autres types de démarrage se trouvent dans la Tableau 9.10, « Options importantes de la section [Service] ».
- **`WantedBy`** fait état de la cible ou des cibles sous laquelle ou sous lesquelles le service devrait être lancé. Ces cibles peuvent être vues comme remplaçant l'ancien concept des niveaux d'exécution, veuillez consulter la Section 9.3, « Travailler avec des cibles Systemd » pour obtenir des détails supplémentaires.

4. Notifier systemd qu'un nouveau fichier ***name*.service** existe en exécutant la commande suivante en tant qu'utilisateur **`root`** :

```
systemctl daemon-reload
systemctl start name.service
```

AVERTISSEMENT

Exécutez la commande **`systemctl daemon-reload`** à chaque fois que vous créez des nouveaux fichiers d'unités ou lorsque vous modifiez des fichiers d'unités existants. Sinon, les commandes **`systemctl start`** ou **`systemctl enable`** peuvent échouer à cause d'une mauvaise correspondance entre les états de systemd et les fichiers d'unités de service qui se trouvent sur le disque.

L'unité *name*.service peut désormais être gérée comme tout autre service système par des commandes décrites dans la Section 9.2, « Gérer les services système ».

Exemple 9.18. Créer le fichier emacs.service

Lors de l'utilisation de l'éditeur de texte **Emacs**, il est souvent plus rapide et plus pratique de le laisser exécuter en arrière-plan plutôt que de lancer une nouvelle instance du programme à chaque fois qu'un fichier doit être modifié. Les étapes suivantes montrent comment créer un fichier d'unité pour Emacs afin qu'il puisse être géré en tant que service.

1. Créez un fichier d'unité dans le répertoire **`/etc/systemd/system/`** et assurez-vous qu'il possède les permissions de fichier correctes. Veuillez exécuter en tant qu'utilisateur **`root`** :

```
~]# touch /etc/systemd/system/emacs.service
~]# chmod 664 /etc/systemd/system/emacs.service
```

2. Ajouter le contenu suivant au fichier de configuration :

```
[Unit]
Description=Emacs: the extensible, self-documenting text editor

[Service]
Type=forking
ExecStart=/usr/bin/emacs --daemon
ExecStop=/usr/bin/emacsclient --eval "(kill-emacs)"
Environment=SSH_AUTH_SOCK=%t/keyring/ssh
Restart=always

[Install]
WantedBy=default.target
```

   Avec la configuration ci-dessus, l'exécutable **`/usr/bin/emacs`** est lancé en mode démon pendant le démarrage du service. La variable d'environnement SSH_AUTH_SOCK est paramétrée en utilisant le spécificateur d'unité « %t » qui correspond au répertoire du runtime. Le service redémarre également le processus Emacs s'il s'arrête de manière inattendue.

3. Veuillez exécuter les commandes suivantes pour recharger la configuration et lancer le service personnalisé :

```
~]# systemctl daemon-reload
```

```
~]# systemctl start emacs.service
```

Comme l'éditeur est désormais enregistré en tant que service systemd, vous pouvez utiliser toutes les commandes **systemctl** standard. Ainsi, exécutez **systemctl status emacs** pour afficher le statut de l'éditeur ou **systemctl enable emacs** pour le lancer automatiquement pendant le démarrage système.

Exemple 9.19. Création d'une seconde instance du service sshd

Les administrateurs système ont souvent besoin de configurer et d'exécuter de multiples instances d'un service. Ceci est effectué en créant des copies des fichiers de configuration du service d'origine et en modifiant certains paramètres pour éviter les conflits avec l'instance principale du service. La procédure suivante montre comment créer une seconde instance du service **sshd** :

1. Veuillez créer une copie du fichier **sshd_config** qui sera utilisée par le second démon :

   ```
   ~]# cp /etc/ssh/sshd{,-second}_config
   ```

2. Veuillez modifier le fichier **sshd-second_config** créé dans l'étape précédente pour assigner un numéro de port différent et un fichier PID au second démon :

   ```
   Port 22220
   PidFile /var/run/sshd-second.pid
   ```

   Veuillez consulter la page man de **sshd_config**(5) pour obtenir plus d'informations sur les options **Port** et **PidFile**. Assurez-vous que le port choisi n'est pas en cours d'utilisation par un autre service. Le fichier PID ne doit pas forcément exister avant l'exécution du service, il est généré automatiquement lors du démarrage du service.

3. Veuillez créer une copie du fichier d'unité systemd pour le service **sshd**.

   ```
   ~]# cp /usr/lib/systemd/system/sshd.service
   /etc/systemd/system/sshd-second.service
   ```

4. Veuillez altérer le fichier **sshd-second.service** créé pendant l'étape précédente comme suit:

   a. Modifiez l'option **Description** :

      ```
      Description=OpenSSH server second instance daemon
      ```

   b. Veuillez ajouter sshd.service aux services spécifiés dans l'option **After**, afin que la seconde instance soit lancée uniquement après le lancement de la première :

      ```
      After=syslog.target network.target auditd.service sshd.service
      ```

   c. La première instance de sshd inclut la génération de clés, veuillez donc supprimer la ligne *ExecStartPre=/usr/sbin/sshd-keygen*.

   d. Ajoutez le paramètre **-f /etc/ssh/sshd-second_config** à la commande **sshd** afin que le fichier de configuration alternatif soit utilisé :

```
ExecStart=/usr/sbin/sshd -D -f /etc/ssh/sshd-second_config
$OPTIONS
```

e. Après les modifications indiquées ci-dessus, le fichier sshd-second.service devrait ressembler à ce qui suit :

```
[Unit]
Description=OpenSSH server second instance daemon
After=syslog.target network.target auditd.service sshd.service

[Service]
EnvironmentFile=/etc/sysconfig/sshd
ExecStart=/usr/sbin/sshd -D -f /etc/ssh/sshd-second_config
$OPTIONS
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=on-failure
RestartSec=42s

[Install]
WantedBy=multi-user.target
```

5. Si vous utilisez SELinux, veuillez ajouter le port pour la seconde instance de sshd sur les ports SSH, sinon la seconde instance de sshd sera rejetée pour lier le port :

```
~]# semanage port -a -t ssh_port_t -p tcp 22220
```

6. Veuillez activer sshd-second.service, afin qu'il soit lancé automatiquement pendant le démarrage :

```
~]# systemctl enable sshd-second.service
```

Vérifiez si sshd-second.service est en cours d'utilisation par la commande **`systemctl status`**. Veuillez également vérifier si le port est activé correctement en effectuant une connexion au service :

```
~]$ ssh -p 22220 user@server
```

Si le pare-feu est en cours d'utilisation, veuillez vous assurer qu'il soit correctement configuré de manière à permettre des connexions à la seconde instance de sshd.

9.6.3. Convertir des scripts init SysV en fichiers d'unité

Avant de prendre le temps de convertir un script init SysV en fichier d'unité, assurez-vous que la conversion n'a pas déjà été effectuée ailleurs. Tous les services de base installés sur Red Hat Enterprise Linux 7 sont offerts avec des fichiers d'unité par défaut, et la même chose s'applique à de nombreux paquets logiciels de tierce partie.

Convertir un script init en fichier d'unité requiert d'analyser le script et d'en extraire les informations nécessaires. En se basant sur ces données, il est possible de créer un fichier d'unité comme décrit dans la Section 9.6.2, « Créer des fichiers d'unité personnalisés ». Comme les scripts init peuvent largement varier en fonction du type de service, vous pourriez devoir employer davantage d'options de

configuration pour la traduction que ce qui est indiqué dans ce chapitre. Veuillez remarquer que certains niveaux de personnalisation qui étaient disponibles avec les scripts init ne sont plus pris en charge par les unités systemd, veuillez consulter la Section 9.1.2, « Changements de compatibilité ».

La majorité des informations nécessaires à une conversion est fournie dans l'en-tête du script. L'exemple suivant montre la section au début du script init utilisée pour lancer le service **postfix** sur Red Hat Enterprise Linux 6 :

```
#!/bin/bash
#
# postfix      Postfix Mail Transfer Agent
#
# chkconfig: 2345 80 30
# description: Postfix is a Mail Transport Agent, which is the program \
#              that moves mail from one machine to another.
# processname: master
# pidfile: /var/spool/postfix/pid/master.pid
# config: /etc/postfix/main.cf
# config: /etc/postfix/master.cf

### BEGIN INIT INFO
# Provides: postfix MTA
# Required-Start: $local_fs $network $remote_fs
# Required-Stop: $local_fs $network $remote_fs
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: start and stop postfix
# Description: Postfix is a Mail Transport Agent, which is the program
that
#              moves mail from one machine to another.
### END INIT INFO
```

Dans l'exemple ci-dessus, seules les lignes commençant par *# chkconfig* et *# description* sont obligatoires, donc vous risquez de ne pas trouver le reste dans des fichiers init différents. Le texte situé entre les lignes *### BEGIN INIT INFO* et *### END INIT INFO* est appelé un *en-tête LSB (« Linux Standard Base »)*. Si spécifiés, les en-têtes LSB contiennent des directives définissant la description du service, les dépendances, et les niveaux d'exécution. Ce qui suit est un ensemble de tâches analytiques visant à collecter les données nécessaires à un nouveau fichier d'unité. Le script init postfix est utilisé comme exemple. Veuillez consulter le fichier d'unité postfix résultant dans l'Exemple 9.17, « Fichier d'unité postfix.service ».

Trouver la description du service

Vous trouverez des informations descriptives concernant le script sur la ligne commençant par *#description*. Veuillez utiliser cette description avec le nom du service dans l'option **Description** de la section [Unit] du fichier d'unité. L'en-tête LSB peut contenir des données similaires sur les lignes *#Short-Description* et *#Description*.

Trouver les dépendances de service

L'en-tête LSB peut contenir plusieurs directives qui forment des dépendances entre services. La plupart d'entres elles peuvent être traduites en options d'unité systemd, veuillez consulter la Tableau 9.12, « Options de dépendances d'en-tête LSB »

Tableau 9.12. Options de dépendances d'en-tête LSB

Option LSB	Description	Équivalent de fichier d'unité
Provides	Spécifie le nom de l'installation de démarrage du service, à qui il peut être fait référence dans d'autres scripts init (avec le préfixe « $ »). Ceci n'est plus nécessaire car les fichiers d'unité font référence aux autres unités par leur nom de fichier.	–
Required-Start	Contient les noms des installations de démarrage des services requis. Ceci se traduit par une dépendance d'ordre, les noms d'installations de démarrage sont remplacés par les noms des fichiers d'unité des services correspondants ou les cibles auxquelles ils appartiennent. Par exemple, dans le cas de **postfix**, la dépendance « Required-Start » sur « $network » a été traduite sur la dépendance « After » sur « network.target ».	**After**, **Before**
Should-Start	Constitue des dépendances plus faibles que Required-Start. Les dépendances Should-Start échouées n'affecteront pas le démarrage du service.	**After**, **Before**
Required-Stop, **Should-Stop**	Constitue des dépendances négatives.	**Conflicts**

Trouver les cibles par défaut du service

La ligne commençant par *#chkconfig* contient trois valeurs numériques. La plus importante est le premier numéro qui représente les niveaux d'exécution par défaut dans lesquels le service est lancé. Veuillez utiliser la Tableau 9.6, « Comparaison des niveaux d'exécution SysV avec les cibles Systemd » pour faire correspondre ces niveaux d'exécution à leurs cibles systemd équivalentes. Puis, répertoriez ces cibles dans l'option **WantedBy** de la section [Install] du fichier d'unité. Par exemple, **postfix** avait auparavant été lancé dans les niveaux d'exécution 2, 3, 4, et 5, qui se traduisent par multi-user.target et graphical.target dans Red Hat Enterprise Linux 7. Veuillez remarquer que graphical.target dépend de multiuser.target, donc il n'est pas nécessaire de spécifier les deux, comme dans l'Exemple 9.17, « Fichier d'unité postfix.service ». Vous trouverez également des informations sur les niveaux d'exécution par défaut et interdits dans les lignes *#Default-Start* et *#Default-Stop* de l'en-tête LSB.

Les deux autres valeurs spécifiées sur la ligne #chkconfig représentent les priorités de démarrage et de fermeture du script init. Ces valeurs sont interprétées par systemd si le script init est chargé, mais il n'y a pas de fichier d'unité équivalent.

Trouver les fichiers utilisés par le service

Les scripts init requièrent le chargement d'une bibliothèque de fonction à partir d'un répertoire dédié et autorisent l'importation de configurations, d'environments, et de fichiers PID. Les variables d'environnement sont spécifiées sur la ligne commençant par *#config* dans l'en-tête du script init, qui se traduit par l'option du fichier d'unité **EnvironmentFile**. Le fichier PID spécifié sur la ligne du script init *#pidfile* est importé sur le fichier d'unité avec l'option **PIDFile**.

L'information-clé qui n'est pas incluse dans l'en-tête du script init est le chemin vers l'exécutable du service, et d'autres fichiers potentiellement requis par le service. Dans des versions précédentes de Red Hat Enterprise Linux, les scripts init utilisaient une déclaration de cas Bash pour définir le sur les

actions par défaut, comme *start*, *stop*, ou *restart*, ainsi que des actions définies de manière personnalisées. L'extrait suivant du script init **postfix** affiche le bloc de code à exécuter lors du lancement du service.

```
conf_check() {
    [ -x /usr/sbin/postfix ] || exit 5
    [ -d /etc/postfix ] || exit 6
    [ -d /var/spool/postfix ] || exit 5
}

make_aliasesdb() {
 if [ "$(/usr/sbin/postconf -h alias_database)" == "hash:/etc/aliases" ]
 then
  # /etc/aliases.db might be used by other MTA, make sure nothing
  # has touched it since our last newaliases call
  [ /etc/aliases -nt /etc/aliases.db ] ||
   [ "$ALIASESDB_STAMP" -nt /etc/aliases.db ] ||
   [ "$ALIASESDB_STAMP" -ot /etc/aliases.db ] || return
  /usr/bin/newaliases
  touch -r /etc/aliases.db "$ALIASESDB_STAMP"
 else
  /usr/bin/newaliases
 fi
}

start() {
 [ "$EUID" != "0" ] && exit 4
 # Check that networking is up.
 [ ${NETWORKING} = "no" ] && exit 1
 conf_check
 # Start daemons.
 echo -n $"Starting postfix: "
 make_aliasesdb >/dev/null 2>&1
 [ -x $CHROOT_UPDATE ] && $CHROOT_UPDATE
 /usr/sbin/postfix start 2>/dev/null 1>&2 && success || failure $"$prog
start"
 RETVAL=$?
 [ $RETVAL -eq 0 ] && touch $lockfile
        echo
 return $RETVAL
}
```

L'extensibilité du script initi autorise la spécification de deux fonctions personnalisées, **conf_check()** et **make_aliasesdb()**, qui sont appelées à partir du bloc de la fonction **start()**. En examinant cela de plus près, plusieurs fichiers externes et plusieurs répertoires sont mentionnés dans le code ci-dessus : l'exécutable du service principal, **/usr/sbin/postfix**, les répertoires de configuration**/etc/postfix/** et **/var/spool/postfix/**, ainsi que le répertoire **/usr/sbin/postconf/**.

Systemd prend uniquement en charge les actions prédéfinies, mais permet l'exécution d'exécutables personnalisés avec les options **ExecStart**, **ExecStartPre**, **ExecStartPost**, **ExecStop**, et **ExecReload**. Dans le cas de **postfix** dans Red Hat Enterprise Linux 7, **/usr/sbin/postfix** accompagné de scripts pris en charge sont exécutés pendant le lancement du service. Veuillez consulter le fichier d'unité **postfix** dans l'Exemple 9.17, « Fichier d'unité postfix.service ».

Convertir des scripts init complexes requiert une bonne compréhension du but de chaque déclaration présente dans le script. Certaines déclarations sont spécifiques à la version du système d'exploitation, vous ne devrez donc pas forcément les traduire. D'autre part, certains ajustements peuvent être nécessaire dans le nouvel environnement, dans le fichier d'unité, ainsi que dans l'exécutable du service et les fichiers de support.

9.6.4. Modifier les fichiers d'unité existants

Les services installés sur le système sont fournis avec des fichiers d'unité par défaut qui sont stockés dans le répertoire **/usr/lib/systemd/system/**. Les administrateurs systèmes ne doivent pas modifier ces fichiers directement, ainsi toute personnalisation doit être confinée aux fichiers de configuration dans le répertoire **/etc/systemd/system/**. Veuillez choisir l'une des approches suivantes, en fonction de l'étendue des changements requis :

- Veuillez créer un répertoire pour les fichiers de configuration supplémentaires dans **/etc/systemd/system/*unit*.d/**. Cette méthode est recommandée pour la plupart des cas d'utilisation. Elle permet d'étendre la configuration par défaut avec des fonctionnalités supplémentaires, tout en continuant à faire référence au fichier d'unité d'origine. Les changements apportés à l'unité par défaut qui ont eu lieu lors d'une mise à niveau de paquet(s) sont ainsi appliqués automatiquement. Veuillez consulter la section intitulée « Étendre la configuration de l'unité par défaut » pour obtenir davantage d'informations.

- Veuillez créer une copie du fichier d'unité d'origine **/usr/lib/systemd/system/** dans **/etc/systemd/system/** et effectuez-y les changements souhaités. La copie remplace le fichier d'origine, donc les changement introduits par la mise à jour du paquet ne sont pas appliqués. Cette méthode est utile pour effectuer des changements significatifs qui devront être persistants, quelles que soient les mises à jour de paquets se produisant. Veuillez consulter la section intitulée « Remplacer la configuration de l'unité par défaut » pour obtenir des détails.

Pour retourner à la configuration par défaut de l'unité, veuillez supprimer les fichiers de configuration créés de manière personnalisée dans **/etc/systemd/system/**. Pour appliquer les changements aux fichiers d'unité sans redémarrer le système, veuillez exécuter :

```
systemctl daemon-reload
```

L'option **daemon-reload** recharge tous les fichiers d'unité et recrée la totalité de l'arbre de dépendances, ce qui est nécessaire pour appliquer immédiatement tout changement sur un fichier d'unité. Alternativement, le même résultat peut être atteint à l'aide de la commande suivante :

```
init q
```

Si le fichier d'unité modifié appartient à un service en cours d'exécution, ce service doit être redémarré pour accepter les nouveaux paramètres :

```
systemctl restart name.service
```

Étendre la configuration de l'unité par défaut

Pour étendre le fichier d'unité par défaut avec des options de configuration supplémentaires, veuillez commencer par créer un répertoire de configuration dans **/etc/systemd/system/**. Si vous étendez une unité de service, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
mkdir /etc/systemd/system/name.service.d/
```

Veuillez remplacer *name* par le nom du service que vous souhaitez étendre. La syntaxe ci-dessus s'applique à tous les types d'unité.

Créez un fichier de configuration dans le répertoire créé lors de l'étape précédente. Remarquez que le nom du fichier doit se terminer par le suffixe *.conf*. Veuillez saisir :

```
touch /etc/systemd/system/name.service.d/config_name.conf
```

Remplacez *config_name* par le nom du fichier de configuration. Ce fichier adhère à la structure normale du fichier d'unité, ainsi toutes les directives doivent être spécifiées sous les sections correspondantes. Veuillez consulter la Section 9.6.1, « Comprendre la structure des fichiers d'unité ».

Par exemple, pour ajouter une dépendance personnalisée, veuillez créer un fichier de configuration avec le contenu suivant :

```
[Unit]
Requires=new_dependency
After=new_dependency
```

Où *new_dependency* correspond à l'unité devant être marquée comme une dépendance. Un autre exemple consiste en un fichier de configuration qui redémarre après que son processus principal se soit arrêté, avec un délai de 30 secondes :

```
[Service]
Restart=always
RestartSec=30
```

Il est recommandé de créer des fichiers de configuration de petite taille se concentrant sur une seule tâche. De tels fichiers peuvent facilement être déplacés ou liés aux répertoires de configuration d'autres services.

Pour appliquer les changements effectués sur l'unité veuillez exécuter la commande suivante en tant que **root** :

```
systemctl daemon-reload
systemctl restart name.service
```

Exemple 9.20. Étendre la configuration httpd.service

Pour modifier l'unité httpd.service de manière à ce qu'un script shell personnalisé soit automatiquement exécuté lors du lancement du service Apache, veuillez effectuer les étapes suivantes. Pour commencer, veuillez créer un répertoire et un fichier de configuration :

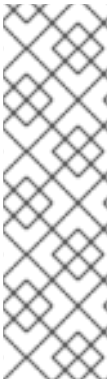
```
~]# mkdir /etc/systemd/system/httpd.service.d/
~]# touch /etc/systemd/system/httpd.service.d/custom_script.conf
```

En supposant que le script que vous souhaitez lancer automatiquement avec Apache se situe dans **/usr/local/bin/custom.sh**, veuillez insérer le texte suivant dans le fichier **custom_script.conf** :

```
[Service]
ExecStartPost=/usr/local/bin/custom.sh
```

Pour appliquer les changements d'unité, veuillez exécuter :

```
~]# systemctl daemon-reload
~]# systemctl restart httpd.service
```



NOTE

Les fichiers de configuration des répertoires de configuration dans **/etc/systemd/system/** ont priorité sur les fichiers d'unité dans **/usr/lib/systemd/system/**. Ainsi, si les fichiers de configuration contiennent une option qui peut être spécifiée une seule fois, comme **Description** ou **ExecStart**, la valeur par défaut de cette option sera remplacée. Remarquez que dans la sortie de la commande **systemd-delta** décrite dans la section intitulée « Surveiller les unités remplacées », de telles unités sont toujours marquées comme [EXTENDED], même si au total, certaines options sont remplacées.

Remplacer la configuration de l'unité par défaut

Pour effectuer des changements qui persisteront après la mise à jour du paquet fournissant le fichier d'unité, veuillez copier le ficher dans le répertoire **/etc/systemd/system/**. Pour cela, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
cp /usr/lib/systemd/system/name.service /etc/systemd/system/name.service
```

Quand *name* correspond au nom de l'unité du service que vous souhaitez modifier, la syntaxe ci-dessus s'applique à tous les types d'unité.

Ouvrez le fichier copié avec un éditeur de texte, et effectuez les changements souhaités. Pour appliquer les changements d'unité, veuillez exécuter ceci en tant qu'utilisateur **root** :

```
systemctl daemon-reload
systemctl restart name.service
```

Exemple 9.21. Modifier la limite du délai d'attente

Vous pouvez spécifier une valeur de délai d'attente par service pour empêcher un service défectueux de geler le système. Vous pouvez modifier le délai d'attente par défaut à 90 secondes pour les services normaux et à 300 secondes pour les services compatibles SysV. Pour prolonger cette limite, vous pouvez modifier le delai d'expiration par défaut en créant le fichier d'unité systemd **/etc/systemd/system/network.service.d/timeout.conf** et ajouter une ligne à la nouvelle configuration à ce endroit. La commande **systemctl show network -p TimeoutStartUSec** affichera le délai d'attente actuel maximum. Après avoir changé la limite à 10 secondes, comme dans l'exemple ci-dessous, pensez à redémarrer **systemd** avec **systemctl daemon-reload** pour que les modifications puissent entrer en vigueur :

```
~]# systemctl show network -p TimeoutStartUSec
TimeoutStartUSec=5min
~]# mkdir /etc/systemd/system/network.service.d/
~]# echo -e '[Service]\nTimeoutStartSec=10' >
/etc/systemd/system/network.service.d/timeout.conf
```

```
~]# systemctl daemon-reload
~]# systemctl show network -p TimeoutStartUSec
TimeoutStartUSec=10s
```

Surveiller les unités remplacées

Pour afficher une vue d'ensemble des fichiers d'unité remplacés ou modifés, veuillez utiliser la commande suivante :

```
systemd-delta
```

Par exemple, la sortie de la commande ci-dessus pourrait ressembler à ce qui suit :

```
[EQUIVALENT] /etc/systemd/system/default.target →
/usr/lib/systemd/system/default.target
[OVERRIDDEN] /etc/systemd/system/autofs.service →
/usr/lib/systemd/system/autofs.service

--- /usr/lib/systemd/system/autofs.service      2014-10-16
21:30:39.000000000 -0400
+++ /etc/systemd/system/autofs.service  2014-11-21 10:00:58.513568275 -
0500
@@ -8,7 +8,8 @@
 EnvironmentFile=-/etc/sysconfig/autofs
 ExecStart=/usr/sbin/automount $OPTIONS --pid-file /run/autofs.pid
 ExecReload=/usr/bin/kill -HUP $MAINPID
-TimeoutSec=180
+TimeoutSec=240
+Restart=Always

 [Install]
 WantedBy=multi-user.target

[MASKED]     /etc/systemd/system/cups.service →
/usr/lib/systemd/system/cups.service
[EXTENDED]   /usr/lib/systemd/system/sssd.service →
/etc/systemd/system/sssd.service.d/journal.conf

4 overridden configuration files found.
```

La Tableau 9.13, « Types de différence systemd-delta » répertorie les types de remplacements qui peuvent apparaître dans la sortie de **systemd-delta**. Remarquez que si un fichier est remplacé, **systemd-delta** affiche par défaut un sommaire des changements similaires à la sortie de la commande **diff**.

Tableau 9.13. Types de différence systemd-delta

Type	Description
[MASKED]	Fichiers d'unités masqués, veuillez consulter la Section 9.2.7, « Désactiver un service » pour une description du masquage d'unité.
[EQUIVALENT]	Copies non modifiées qui remplacent les fichiers d'origine mais dont le contenu, typiquement les liens symboliques, ne diffère pas.

Type	Description
[REDIRECTED]	Fichiers redirigés vers d'autres fichiers.
[OVERRIDEN]	Fichiers remplacés et modifiés.
[EXTENDED]	Fichiers étendus avec les fichiers .conf dans le répertoire **/etc/systemd/system/*unit*.d/**.
[UNCHANGED]	Fichiers non modifiés, uniquement affichés lorsque l'option **--type=unchanged** est utilisée.

L'exécution de **systemd-delta** après une mise à jour système pour vérifier si des mises à jour d'unités par défaut sont actuellement remplacées par une configuration personnalisée. Il est également possible de limiter la sortie à un certain type de différence uniquement. Par exemple, pour uniquement afficher les unités remplacées, veuillez exécuter :

```
systemd-delta --type=overridden
```

9.6.5. Travailler avec des unités instanciées

Il est possible d'instancier plusieurs unités à partir d'un seul fichier de configuration modèle pendant le runtime. Le caractère « @ » est utilisé pour marquer le modèle et pour y associer des unités. Les unités instanciées peuvent être lancées à partir d'un autre fichier d'unité (à l'aide des options **Requires** ou **Wants**), ou par la commande **systemctl start**. Les unités de service instanciées sont nommées comme suit.

```
template_name@instance_name.service
```

Quand *template_name* correspond au nom du fichier de configuration. Veuillez remplacer *instance_name* par le nom de l'instance de l'unité. Plusieurs instances peuvent pointer vers le même fichier modèle avec des options de configuration communes pour toutes les instances de l'unité. Le nom de l'unité modèle est sous le format suivant :

```
unit_name@.service
```

Par exemple, le paramètre **Wants** suivant dans un fichier d'unité :

```
Wants=getty@ttyA.service,getty@ttyB.service
```

cela lance avant tout une recherche systemd des unités de service données. Si de telles unités sont introuvables, la partie entre le caractère « @ » et le suffixe du type sera ignorée et systemd recherchera le fichier **getty@.service**, lira la configuration à partir de celui-ci, et lancera les services.

Les caractères génériques, aussi appelés des *spécificateurs d'unité*, peuvent être utilisés dans tout fichier de configuration. Les spécificateurs d'unité substituent certains paramètres d'unité et sont interprétés pendant le runtime. La Tableau 9.14, « Spécificateurs d'unités importantes » répertorie des spécificateurs d'unité qui sont particulièrement utiles pour les unités modèles.

Tableau 9.14. Spécificateurs d'unités importantes

Spécificateur d'unité	Signification	Description
%n	Nom d'unité complet	Correspond au nom d'unité complet, y compris le suffixe du type. **%N** possède la même signification mais remplace également les caractères interdits avec les codes ASCII.
%p	Nom du préfixe	Correspond à un nom d'unité avec le suffixe de type supprimé. Pour les unités instanciées, %p correspond à la partie du nom de l'unité avant le caractère « @ ».
%i	Nom d'instance	Il s'agit de la partie du nom d'unité instanciée entre le caractère « @ » et le suffixe du type. **%I** possède la même signification mais remplace également les caractères interdits par des codes ASCII.
%H	Nom d'hôte	Correspond au nom d'hôte du système en cours d'exécution au moment où la configuration de l'unité est chargée.
%t	Répertoire du runtime	Représente le répertoire du runtime, qui est **/run** pour l'utilisateur **root**, ou la valeur de la variable XDG_RUNTIME_DIR pour les utilisateur non-privilégiés.

Pour une liste complète des spécificateurs d'unité, veuillez consulter la page man de **systemd.unit(5)**.

Par exemple, le modèle **getty@.service** contient les directives suivante :

```
[Unit]
Description=Getty on %I
...
[Service]
ExecStart=-/sbin/agetty --noclear %I $TERM
...
```

Lorsque getty@ttyA.service et getty@ttyB.service sont instanciés à partir du modèle ci-dessus, **Description**= est résolu en tant que *Getty on ttyA* et *Getty on ttyB*.

9.7. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur systemd et son utilisation sur Red Hat Enterprise Linux 7, veuillez consulter les ressources répertoriées ci-dessous.

Documentation installée

- **systemctl**(1) — la page du manuel de l'utilitaire de ligne de commande**systemctl** fournit une liste complète des options et des commandes prises en charge.
- **systemd**(1) — la page du manuel du gestionnaire de systèmes et services**systemd** fournit davantage d'informations sur ses concepts et documente les options de ligne de commande et les variables d'environnement disponibles, les fichiers de configuration et répertoires pris en charge, les signaux reconnus, ainsi que les options de noyau disponibles.

- **systemd-delta**(1) — la page du manuel de l'utilitaire **systemd-delta** qui permet de trouver des fichiers de configuration étendus et remplacés.
- **systemd.unit**(5) — la page du manuel nommée **systemd.unit** fournit des informations détaillées sur les fichiers d'unité systemd et documente toutes les options de configuration disponibles.
- **systemd.service**(5) — la page du manuel nommée **systemd.service** documente le format des fichiers d'unité de service.
- **systemd.target**(5) — la page du manuel nommée **systemd.target** documente le format des fichiers d'unité cibles.
- **systemd.kill**(5) — la page du manuel nommée **systemd.kill** documente la configuration de la procédure de fermeture de processus (« process killing »).

Documentation en ligne

- Guide de mise en réseau Red Hat Enterprise Linux 7 — le *Guide de mise en réseau* de Red Hat Enterprise Linux 7 documente les informations pertinentes à la configuration et à l'administration des interfaces réseau et des services réseau sur ce système. Il fournit une introduction à l'utilitaire **hostnamectl** et explique comment l'utiliser pour afficher et définir des noms d'hôtes sur la ligne de commande localement et à distance, et fournit des informations importantes sur la sélection des noms d'hôte et des noms de domaines
- Guide de migration et d'administration de bureau Red Hat Enterprise Linux 7 — le *Guide de migration et d'administration de bureau* de Red Hat Enterprise Linux 7 documente la planification de la migration, le déploiement, la configuration, et l'administration du bureau GNOME 3 sur ce système. Le service **logind** y est également présenté, ses fonctionnalités les plus importantes sont énumérées, et la manière d'utiliser l'utilitaire **loginctl** pour répertorier les sessions actives et activer la prise en charge « multi-seat » est expliquée.
- Guide de l'utilisateur et de l'administrateur SELinux Red Hat Enterprise Linux 7 — le *Guide de l'utilisateur et de l'administrateur SELinux* Red Hat Enterprise Linux 7 décrit les principes de base de SELinux et documente en détails comment configurer et utiliser SELinux avec divers services, tels que Apache HTTP Server, Postfix, PostgreSQL, ou OpenShift. Celui-ci explique comment configurer les permissions d'accès SELinux pour les services système gérés par systemd.
- Guide d'installation Red Hat Enterprise Linux 7 — le *Guide d'installation* de Red Hat Enterprise Linux 7 documente comment installer le système d'exploitation sur des systèmes AMD64 et Intel 64, sur des serveurs IBM Power Systems 64 bits, ainsi que sur IBM System z. Il couvre également des méthodes d'installation avancées telles que les installations Kickstart, PXE, et les installations au moyen du protocole VNC. En outre, ce manuel décrit les tâches post-installation communes et explique comment résoudre les problèmes d'installation, il comprend également des instructions détaillées sur la manière de démarrer en mode de secours ou de récupérer le mot de passe root.
- Guide de sécurité Red Hat Enterprise Linux 7 — le *Guide de sécurité* Red Hat Enterprise Linux 7 assiste les utilisateurs et les administrateurs dans leur apprentissage des processus et pratiques de sécurisation de leurs stations de travail et serveurs envers des intrusions locales et distantes, des exploitations, et autres activités malicieuses. Celui-ci explique également comment sécuriser des services de système critiques.
- Page d'accueil systemd — la page d'accueil du projet fournit davantage d'informations sur systemd.

Voir aussi

- Le Chapitre 1, *Paramètres régionaux et configuration du clavier* documente comment gérer les paramètres régionaux du système et les structures du clavier. Il explique également comment utiliser l'utilitaire **`localectl`** pour afficher les paramètres régionaux actuels sur la ligne de commande, pour répertorier les paramètres régionaux disponibles, et les définir sur la ligne de commande, ainsi que pour afficher la structure actuelle du clavier, répertorier les dispositions claviers, et activer une disposition clavier particulière sur la ligne de commande.

- Le Chapitre 2, *Configurer l'heure et la date* documente comment gérer la date et l'heure du système. Il explique les différences entre une horloge temps réel et une horloge système, et décrit comment utiliser l'utilitaire **`timedatectl`** pour afficher les paramètres actuels de l'hologe système, configurer l'heure et la date, changer de fuseau horaires, et pour synchroniser l'horloge système avec un serveur distant.

- Le Chapitre 5, *Obtention de privilèges* documente comment obtenir des privilèges administratifs en utilisant les commandes **`su`** et **`sudo`**.

- Le Chapitre 10, *OpenSSH* décrit comment configurer un serveur SSH et comment utiliser les utilitaires client **`ssh`**, **`scp`**, et **`sftp`** pour y accéder.

- Le Chapitre 20, *Afficher et gérer des fichiers journaux* fournit une introduction à journald. Il décrit le journal, introduit le service **`journald`**, et documente comment utiliser l'utilitaire **`journalctl`** pour afficher les entrées de journal, entrer en mode d'affichage en direct, et filtrer les entrées du journal. En outre, ce chapitre décrit comment autoriser les utilisateurs non root d'accéder aux journaux du système et comment activer le stockage persistant des fichiers journaux.

CHAPITRE 10. OPENSSH

SSH (Secure Shell) est un protocole qui facilite les communications sécurisées entre deux systèmes en utilisant une architecture client-serveur, et qui permet aux utilisateurs de se connecter à des systèmes hôtes de serveur à distance. À la différence d'autres protocoles de communication à distance tels que **FTP** ou **Telnet**, SSH codifie la session de connexion, ce qui la rend difficile d'accès aux intrus qui souhaiteraient récupérer les mots de passe.

Le programme **ssh** est conçu pour remplacer les applications de terminal les plus anciennes et les moins sécurisées qui sont utilisées pour la connexion à des hôtes distants, comme **telnet** ou **rsh**. Un programme connexe appelé **scp** remplace les anciens programmes conçus pour copier des fichiers entre les ordinateurs hôtes, comme **rcp**. Comme ces applications plus anciennes ne cryptent pas les mots de passe qui sont transmis entre le client et le serveur, évitez-les autant que possible. En utilisant des méthodes sûres pour vous connecter à des systèmes distants, vous diminuez les risques pour le système client et l'hôte distant à la fois.

Red Hat Enterprise Linux comprend le paquet OpenSSH général, openssh, ainsi que les paquets de serveur, openssh-server et des clients, openssh-clients. Noter que les paquets OpenSSH exigent la présence du paquet OpenSSL openssl-libs, qui installe plusieurs bibliothèques cryptographiques importantes, ce qui permet à OpenSSH de fournir des communications cryptées

10.1. LE PROTOCOLE SSH

10.1.1. Pourquoi utiliser SSH ?

Les intrus potentiels ont une variété d'outils à leur disposition leur permettant de perturber, intercepter et re-router le trafic réseau dans le but d'accéder à un système. En règle générale, ces menaces peuvent être classées ainsi :

Interception de communication entre deux systèmes

L'attaquant peut être quelque part sur le réseau entre les parties communicantes, copiant des informations passées entre elles. Il peut intercepter et maintenir l'information, ou la modifier et l'envoyer à des destinataires designés.

Cette attaque est généralement réalisée à l'aide d'un *renifleur de paquets* (packet sniffer), un utilitaire de réseau plutôt commun, qui capte chaque paquet qui passe à travers le réseau et qui en analyse le contenu.

Emprunt d'identité d'un hôte particulier

Le système de l'attaquant est configuré pour se poser en tant que destinataire d'une transmission. Si cette stratégie fonctionne, le système de l'utilisateur ne sait pas qu'il communique avec le mauvais hôte.

Cette attaque peut être effectuée par une technique appelée un *empoisonnement DNS*, ou via ce que l'on appelle une *usurpation d'adresse IP*. Dans le premier cas, l'intrus utilise un serveur DNS qui aura failli dans le but de pointer les systèmes client à un hôte qui aura été malicieusement dupliqué. Dans le second cas, l'intrus envoie des paquets réseau falsifiés qui semblent provenir d'un hôte de confiance.

Les deux techniques interceptent des informations sensibles potentiellement, et, si l'interception a lieu à des fins hostiles, les résultats peuvent être désastreux. Si SSH est utilisé pour la connexion à un shell distant et pour la copie de fichiers, ces menaces à la sécurité peuvent être grandement diminuées. C'est parce que le client SSH et le serveur utilisent des signatures numériques pour vérifier leur identité. En

plus, toutes les communications entre les systèmes client et serveur sont cryptées. Les tentatives d'usurpation l'identité de part et d'autre d'une communication ne fonctionnent pas, puisque chaque paquet est crypté à l'aide d'une clé connue seulement par les systèmes locaux et distants.

10.1.2. Fonctionnalités principales

Le protocole SSH offre les avantages suivants :

Personne ne peut se faire passer pour un serveur intentionnel

Après avoir effectué une connexion initiale, le client peut s'assurer que sa connexion est établie avec le même serveur que lors de sa session précédente.

Personne ne peut récupérer les données d'authentification

Le client transmet ses données d'authentification au serveur au moyen d'un chiffrement solide de 128 bits.

Personne ne peut intercepter la communication

Toutes les données envoyées et reçues lors d'une session sont transférées au moyen d'un chiffrement 128 bits, rendant ainsi le déchiffrement et la lecture de toute transmission interceptée extrêmement difficile.

De plus, il offre les options suivantes :

Il fournit des moyens sûrs d'utiliser les applications graphiques par l'intermédiaire d'un réseau.

En utilisant une technique appelée *X11 forwarding*, le client peut envoyer des applications *X11* (*X Window System*) en partance du serveur.

Cela fournit un moyen de sécuriser des protocoles non sécurisés normalement

Le protocole SSH crypte tout ce qu'il envoie et reçoit. En utilisant une technique appelée *port forwarding*, un serveur SSH peut devenir un moyen de sécuriser des protocoles normalement non protégés, comme POP, tout en augementant la sécurité des données et du système en général.

Il peut être utilisé pour créer un canal sécurisé

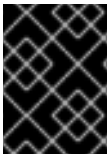
Le serveur OpenSSH et le client peuvent être configurés afin de créer un tunnel semblable à un réseau virtuel privé pour permettre le trafic entre le serveur et les machines clientes.

Il prend en charge l'authentification Kerberos

Les clients et serveurs OpenSSH peuvent être configurés pour authentifier par l'interface GSSAPI (Generic Security Services Application Program Interface) du protocole d'authentification du réseau Kerberos.

10.1.3. Versions de protocole

Il existe actuellement deux sorts de SSH : version 1 et version 2 plus récente. La suite OpenSSH sous Red Hat Enterprise Linux utilise SSH la version 2, qui dispose d'un algorithme d'échange de clés amélioré non vulnérable à l'exploitation connue pour la version 1. Toutefois, pour des raisons de compatibilité, la suite OpenSSH prend en charge les connexions de la version 1 également.



IMPORTANT

Pour assurer un maximum de sécurité au niveau de votre connexion, il est conseillé que seuls les clients et serveurs compatibles avec SSH version 2 soient utilisés si possible.

10.1.4. Séquence d'événements d'une connexion SSH

Pour aider à protéger l'intégrité d'une communication SSH entre deux ordinateurs hôte, la série suivante d'événements doit être utilisée.

1. Une liaison cryptographique est établie afin de permettre au client de vérifier qu'il est bien en communication avec le serveur souhaité.
2. La couche de transport de la connexion entre le client et tout hôte distant est chiffrée au moyen d'un moyen de chiffrement symétrique.
3. Le client s'authentifie auprès du serveur.
4. Le client peut interagir avec l'hôte distant au moyen d'une connexion chiffrée.

10.1.4.1. Couche de transport

Le rôle principal de la couche de transport est de faciliter une communication sécurisée entre deux hôtes non seulement au moment de l'authentification, mais également lors de communications ultérieures. Pour ce faire, la couche de transport traite le cryptage et décryptage de données et offre une certaine protection quant à l'intégrité des paquets de données lors de leur envoi et de leur réception. De plus, la couche de transport effectue la compression des données permettant d'accélérer la vitesse de transfert des informations.

Lorsqu'un client communique avec un serveur au moyen d'un protocole SSH, de nombreux éléments importants sont échangés afin que les deux systèmes puissent créer correctement la couche de transport. Lors de cet échange, les opérations suivantes ont lieu :

- Des clés sont échangées.
- L'algorithme de chiffrement de clés publiques est déterminé
- L'algorithme de chiffrement symétrique est déterminé
- L'algorithme d'authentification de message est déterminé
- L'algorithme de hachage est déterminé

Au cours de l'échange de clés, le serveur s'identifie au client par une *clé d'hôte* unique. Si le client n'a jamais communiqué avec ce serveur particulier auparavant, la clé du serveur hôte est inconnue au client et il ne se connecte pas. OpenSSH contourne ce problème en acceptant la clé du serveur hôte. Ceci est fait après que l'utilisateur ait été informé et a à la fois accepté et vérifié la nouvelle clé de l'hôte. Dans les connexions suivantes, la clé du serveur hôte est vérifiée par rapport la version enregistrée sur le client, pour s'assurer que le client communique avec le serveur voulu. Si, dans l'avenir, la clé de l'hôte ne correspond plus, l'utilisateur doit supprimer la version du client enregistrée avant qu'une connexion puisse avoir lieu.

AVERTISSEMENT

Il est tout à fait possible pour un pirate de se faire passer pour le serveur SSH lors de la première connexion car le système local ne détecte aucune différence entre le serveur désiré et le faux serveur créé par le pirate. Afin d'éviter une telle situation, contrôlez l'intégrité d'un nouveau serveur SSH en contactant l'administrateur du serveur avant d'établir la première connexion au cas où une clé d'hôte ne correspond pas à celle stockée sur le serveur.

Le protocole SSH est conçu pour fonctionner avec n'importe quel algorithme de clé publique ou tout format de codage. Après que l'échange initial des clés crée une valeur de hachage utilisée pour les échanges et une valeur secrète partagée, les deux systèmes commencent immédiatement à calculer de nouveaux algorithmes et de nouvelles clés pour protéger l'authentification et les futures données envoyées via la connexion.

Après la transmission d'une certaine quantité de données au moyen d'une clé et d'un algorithme précis (la quantité exacte dépend de l'implémentation du protocole SSH), un nouvel échange de clés est effectué ; cette opération engendre la création d'un autre ensemble de valeurs de hachage et d'une autre valeur secrète partagée. De cette façon, même si un pirate réussit à déterminer les valeurs de hachage et la valeur secrète partagée, ces informations ne lui seront utiles que pour une durée limitée.

10.1.4.2. Authentification

Une fois que la couche de transport a créé un tunnel sécurisé pour envoyer les informations entre les deux systèmes, le serveur indique au client les différentes méthodes d'authentification prises en charge, telles que l'utilisation d'une signature dotée d'une clé codée ou la saisie d'un mot de passe. Le client doit ensuite essayer de s'authentifier auprès du serveur au moyen d'une des méthodes spécifiées.

Les serveurs et clients SSH peuvent être configurés de façon à permettre différents types d'authentification, donnant à chacune des deux parties un niveau de contrôle optimal. Le serveur peut décider des méthodes de cryptage qu'il prend en charge en fonction de son modèle de sécurité et le client peut choisir l'ordre des méthodes d'authentification à utiliser parmi les options disponibles.

10.1.4.3. Canaux

After a successful authentication over the SSH transport layer, multiple channels are opened via a technique called *multiplexing*[1]. Each of these channels handles communication for different terminal sessions and for forwarded X11 sessions.

Le client et le serveur peuvent créer un nouveau canal. Chaque canal reçoit ensuite un numéro différent à chaque extrémité de la connexion. Lorsque le client essaie d'ouvrir un nouveau canal, il envoie le numéro du canal accompagné de la requête. Ces informations sont stockées par le serveur et utilisées pour diriger la communication vers ce canal. Cette procédure est utilisée afin que des types différents de session ne créent pas de nuisances mutuelles et de sorte qu'à la fin d'une session donnée, son canal puisse être fermé sans que la connexion SSH principale ne soit interrompue.

Les canaux prennent aussi en charge le *contrôle du flux de données*, ce qui leur permet d'envoyer et de recevoir des données de façon ordonnée. Ce faisant, aucune donnée n'est envoyée sur le canal tant que l'hôte n'a pas reçu de message lui indiquant que le canal est ouvert.

Le client et le serveur négocient automatiquement la configuration de chaque canal, selon le type de

service demandé par le client et la manière dont l'utilisateur est connecté au réseau. Ainsi, le traitement des différents types de connexions distantes est non seulement extrêmement flexible, mais il ne nécessite pas même d'apporter des modifications à la structure de base du protocole.

10.2. CONFIGURATION D'OPENSSH

10.2.1. Fichiers de configuration

Il y a deux groupes de fichiers de configuration : pour les programmes clients (c-a-d **ssh**, **scp**, et **sftp**), et pour les serveurs (le démon **sshd**).

Les informations de configuration de SSH à l'échelle du système sont stockées dans le répertoire **/etc/ssh/** comme décrit dans Tableau 10.1, « Fichiers de configuration du système dans son ensemble ». Les informations de configuration spécifiques à l'utilisateur SSH sont stockées dans **~/.ssh/** dans le répertoire d'accueil de l'utilisateur, comme décrit dans Tableau 10.2, « Fichiers de configuration spécifiques à l'utilisateur ».

Tableau 10.1. Fichiers de configuration du système dans son ensemble

Fichier	Description
/etc/ssh/moduli	Contient les groupes Diffie-Hellman utilisés pour l'échange de clés Diffie-Hellman, ce qui est essentiel pour la création d'une couche de transport sécurisé. Lorsque les clés sont échangées au début d'une session SSH, une valeur partagée, secrète est alors créée ne pouvant être déterminée par l'une des parties seule. Cette valeur est ensuite utilisée pour fournir une authentification de l'hôte.
/etc/ssh/ssh_config	Le fichier de configuration du client SSH par défaut. Notez qu'il sera remplacé par **~/.ssh/config** si ce fichier existe.
/etc/ssh/sshd_config	Le fichier de configuration du démon **sshd**.
/etc/ssh/ssh_host_ecdsa_key	La clé ECDSA privée utilisée par le démon **sshd**.
/etc/ssh/ssh_host_ecdsa_key.pub	La clé ECDSA publique utilisée par le démon **sshd**.
/etc/ssh/ssh_host_key	La clé privée RSA utilisée par le démon **sshd** pour la version 1 du protocole SSH.
/etc/ssh/ssh_host_key.pub	La clé publique RSA utilisée par le démon **sshd** pour la version 1 du protocole SSH.
/etc/ssh/ssh_host_rsa_key	La clé privée RSA utilisée par le démon **sshd** pour la version 2 du protocole SSH.
/etc/ssh/ssh_host_rsa_key.pub	La clé publique RSA utilisée par le démon **sshd** pour la version 2 du protocole SSH.

Fichier	Description
/etc/pam.d/sshd	Le fichier de configuration AM du démon **sshd**.
/etc/sysconfig/sshd	Fichier de configuration du service **sshd**.

Tableau 10.2. Fichiers de configuration spécifiques à l'utilisateur

Fichier	Description
~/.ssh/authorized_keys	Contient une liste des clés publiques autorisées pour les serveurs. Quand le client se connecte à un serveur, le serveur authentifie le client en vérifiant sa clé publique stockée dans ce fichier.
~/.ssh/id_ecdsa	Contient la clé privée ECDSA de l'utilisateur
~/.ssh/id_ecdsa.pub	Le clé publique de l'utilisateur.
~/.ssh/id_rsa	La clé privée RSA utilisée par **ssh** pour la version 2 du protocole SSH.
~/.ssh/id_rsa.pub	La clé publique RSA utilisée par **ssh** pour la version 2 du protocole SSH.
~/.ssh/identity	La clé privée RSA utilisée par **ssh** pour la version 1 du protocole SSH.
~/.ssh/identity.pub	La clé publique RSA utilisée par **ssh** pour la version 1 du protocole SSH.
~/.ssh/known_hosts	Contient les clés hôtes des serveurs SSH auquels l'utilisateur a accès. Ce fichier est important pour s'assurer que le client SSH est connecté au bon serveur SSH.

Pour obtenir des informations sur les différentes directives qui peuvent être utilisées par les fichiers de configuration SSH, voir les pages man **ssh_config**(5) et **sshd_config**(5).

10.2.2. Démarrage d'un serveur OpenSSH

Pour pouvoir exécuter un serveur OpenSSH, vous devez avoir le paquet openssh-server installé. Pour obtenir plus d'informations sur la façon d'installer des nouveaux paquets, voir Section 8.2.4, « Installation de paquets ».

Pour démarrer le démon **sshd** dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl start sshd.service
```

Pour stopper le démon **sshd** dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl stop sshd.service
```

Si vous souhaitez que le démon démarre automatiquement, saisir en tant qu'utilisateur **root**:

```
~]# systemctl enable sshd.service
Created symlink from /etc/systemd/system/multi-
user.target.wants/sshd.service to /usr/lib/systemd/system/sshd.service.
```

Le démon **sshd** dépend de l'unité cible **network.target**, ce qui suffit pour les interfaces réseau configuré statique et pour les options par défaut **ListenAddress 0.0.0.0**. Pour spécifier des adresses différentes dans la directive **ListenAddress** et utiliser une configuration de réseau dynamique plus lente, ajouter la dépendance sur l'unité cible de **network-online.target** dans le fichier d'unité de **sshd.service**. Pour ce faire, créez le fichier **/etc/systemd/system/sshd.service.d/local.conf** avec les options suivantes :

```
  [Unit]
  Wants=network-online.target
  After=network-online.target
```

Ensuite, charger à nouveau la configuration du gestionnaire **systemd** par la commande suivante :

```
~]# systemctl daemon-reload
```

Pour obtenir davantage d'informations sur la manière de gérer les services système sur Red Hat Enterprise Linux 7, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

Notez que si vous réinstallez le système, un nouvel ensemble de clés d'identification sera créé. Ainsi, les clients qui étaient connectés au système avec les outils d'OpenSSH avant la réinstallation verront le message suivant s'afficher :

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle
attack)!
It is also possible that the RSA host key has just been changed.
```

Pour éviter cela, vous pouvez sauvegarder les fichiers qui conviennent à partir du répertoire **/etc/ssh/**. Voir Tableau 10.1, « Fichiers de configuration du système dans son ensemble » pour une liste complète, et restaurez les fichiers à chaque fois que vous réinstallez le système.

10.2.3. Utilisation nécessaire de SSH pour les connexions à distance

Pour que SSH soit vraiment efficace, les protocoles de connexions non sécurisées sont à proscrire. Sinon, un mot de passe d'utilisateur peut-être protégé par SSH pour une seule session, et ne sera capturé que plus tard avec Telnet. Certains services à désactiver sont **telnet**, **rsh**, **rlogin** et **vsftpd**.

Pour obtenir davantage d'informations sur la manière de configurer le service **vsftpd**, voir Section 14.2, « FTP ». Pour savoir comment gérer les services système, consulter Chapitre 9, *Gérer les services avec systemd*.

10.2.4. Authentification basée clés

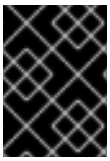
Pour améliorer encore davantage la sécurité système, créer des paires de clés SSH et puis exécuter l'authentification basée clés en désactivant l'authentification de mot de passe. Pour ce faire, ouvrez le fichier **/etc/ssh/sshd_config** dans un éditeur de texte comme **vi** ou **nano** et modifiez l'option **PasswordAuthentication** comme suit :

```
PasswordAuthentication no
```

Si vous travaillez sur un système différent de celui de la nouvelle installation par défaut, vérifiez que **PubkeyAuthentication no** n'ait **pas** été défini. Si connecté à distance, ne pas utiliser l'accès console ou out-of-band, il est conseillé de tester le processus de journalisation basé clé avant de désactiver l'authentification de mot de passe.

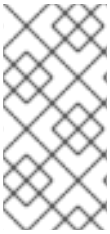
Pour pouvoir utiliser **ssh**, **scp**, ou **sftp** pour se connecter au serveur à partir d'une machine client, créer une paire de clé d'autorisation en suivant les étapes suivantes. Notez que les clés doivent être créées séparemment pour chaque utilisateur.

Red Hat Enterprise Linux 7 utilise SSH Protocol 2 et les clés RSA par défaut (voir Section 10.1.3, « Versions de protocole » pour plus d'informations).



IMPORTANT

Si vous complétez les étapes en tant qu'utilisateur **root**, seul **root** pourra utiliser les clés.



NOTE

Si vous réinstallez votre système et que vous souhaitez garder des paires de clés générées auparavant, sauvegarder le répertoire **~/.ssh/**. Après avoir réinstallé, recopiez-le sur votre répertoire personnel. Ce processus peut être fait pour tous les utilisateurs sur votre système, y compris l'utilisateur **root**.

10.2.4.1. Création de paires de clés

Suivez les étapes indiquées ci-dessous afin de créer une paire de clés DSA pour la version 2 du protocole SSH.

1. La mise à jour d'un paquet est semblable à son installation. Saisir la commande suivante à l'invite du shell :

```
~]$ ssh-keygen -t rsa
Création de la paire de clés publique/privée.
Saisir le fichier dans lequel sauvegarder la clé
(/home/USER/.ssh/id_rsa):
```

2. Appuyer sur la touche **Entrée** pour confirmer le chemin d'accès par défaut , **~/.ssh/id_rsa**, pour la clé nouvellement créée.

3. Saisissez une phrase de passe, et confirmez-là en la saisissant à nouveau quand on vous le demande. Pour des raisons de sécurité, évitez d'utiliser le même mot de passe que celui que vous aurez utilisé quand vous vous êtes connecté à votre compte.

   Après cela, vous verrez un message semblable à celui-ci :

```
Your identification has been saved in /home/USER/.ssh/id_rsa.
Your public key has been saved in /home/USER/.ssh/id_rsa.pub.
The key fingerprint is:
e7:97:c7:e2:0e:f9:0e:fc:c4:d7:cb:e5:31:11:92:14
USER@penguin.example.com
The key's randomart image is:
+--[ RSA 2048]----+
|             E.  |
|            . .  |
|             o . |
|              . .|
|        S .    . |
|         + o o ..|
|          * * +oo|
|           O +..=|
|           o*  o.|
+-----------------+
```

4. Par défaut, les permissions du répertoire **~/.ssh/** sont définies à **rwx------** ou **700** exprimées en notation octale. Ceci est pour s'assurer que l'utilisateur *USER* puisse voir les contenus. Si cela est nécessaire, on peut le confirmer par la commande suivante :

```
~]$ ls -ld ~/.ssh
drwx------. 2 USER USER 54 Nov 25 16:56 /home/USER/.ssh/
```

5. Pour copier une clé publique dans une machine distante, lancer la commande dans le format suivant :

```
ssh-copy-id user@hostname
```

   Cela aura pour effet de copier la clé publique **~/.ssh/id*.pub** qui a été modifiée le plus récemment si elle n'a pas encore été installée. Sinon, indiquer le nom du fichier de clés publiques comme suit :

```
ssh-copy-id -i ~/.ssh/id_rsa.pub user@hostname
```

   Cela aura pour effet de copier le contenu de **~/.ssh/id_rsa.pub** dans le fichier **~/.ssh/authorized_keys** de la machine sur laquelle vous souhaitez vous connecter. Si le fichier existe déjà, les clés y seront ajoutées.

Suivez les étapes indiquées ci-dessous afin de créer une paire de clés ECDSA pour la version 2 du protocole SSH :

1. Pour créer une paire de clés ECDSA, saisir la commande suivante à l'invite du shell :

```
~]$ ssh-keygen -t ecdsa
Créer une paire de clés publique/privée ecdsa.
Saisir le fichier dans lequel vous souhaitez sauvegarder la clé
```

```
(/home/USER/.ssh/id_ecdsa):
```

2. Appuyer sur la touche **Entrée** pour confirmer le chemin d'accès par défaut , **~/.ssh/id_ecdsa**, pour la clé nouvellement créée.

3. Saisissez une phrase de passe, et confirmez-là en la saisissant à nouveau quand on vous le demande. Pour des raisons de sécurité, évitez d'utiliser le même mot de passe que celui que vous aurez utilisé quand vous vous êtes connecté à votre compte.

   Après cela, vous verrez un message semblable à celui-ci :

```
Votre identification a été sauvegardée dans
/home/USER/.ssh/id_ecdsa.
Votre clé publique a été sauvegardée dans
/home/USER/.ssh/id_ecdsa.pub.
L'empreinte de la clé est :
fd:1d:ca:10:52:96:21:43:7e:bd:4c:fc:5b:35:6b:63
USER@penguin.example.com
L'image randomart de la clé est :
+--[ECDSA  256]---+
|       .+ +o     |
|       . =.o     |
|        o o +  ..|
|         + + o  +|
|        S o o oE.|
|           + oo+.|
|            + o  |
|                 |
|                 |
+-----------------+
```

4. Par défaut, les permissions du répertoire **~/.ssh/** sont définies à **rwx------** ou **700** exprimées en notation octale. Ceci est pour s'assurer que l'utilisateur *USER* puisse voir les contenus. Si cela est nécessaire, on peut le confirmer par la commande suivante :

```
~]$ ls -ld ~/.ssh
              ~]$ ls -ld ~/.ssh/
drwx------. 2 USER USER 54 Nov 25 16:56 /home/USER/.ssh/
```

5. Pour copier une clé publique dans une machine distante, lancer la commande sous le format suivant :

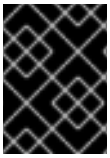
```
ssh-copy-id USER@hostname
```

   Cela aura pour effet de copier la clé publique **~/.ssh/id*.pub** qui a été modifiée le plus récemment si elle n'a pas encore été installée. Sinon, indiquer le nom du fichier de clés publiques comme suit :

```
ssh-copy-id -i ~/.ssh/id_ecdsa.pub USER@hostname
```

   Cela aura pour effet de copier le contenu de **~/.ssh/id_ecdsa.pub** dans le fichier **~/.ssh/authorized_keys** de la machine sur laquelle vous souhaitez vous connecter. Si le fichier existe déjà, les clés y seront ajoutées.

Voir Section 10.2.4.2, « Configurer les partages Samba » pour obtenir des informations sur la façon d'installer votre système afin qu'il se souvienne de la phrase de passe.



IMPORTANT

La clé privée est à but d'utilisation personelle uniquement, et il est important que vous ne la donniez à personne.

10.2.4.2. Configurer les partages Samba

Pour stocker votre mot de passe afin que vous n'ayiez pas à le saisir à chaque fois que vous vous connectez avec une machine distante, vous pouvez utiliser l'agent d'authentification **`ssh-agent`**. Si vous utilisez GNOME, vous pouvez la configurer pour qu'elle vous demande votre mot de passe chaque fois que vous ouvrez une session et pour qu'elle s'en souvienne tout au cours de la session. Sinon, vous pouvez stocker la phrase de passe pour une invite du shell en particulier.

Pour sauvegarder votre phrase de passe pendant votre session GNOME, suivez les étapes suivantes :

1. Assurez-vous de bien avoir le paquet openssh-askpass installé. Sinon, voir Section 8.2.4, « Installation de paquets » pour obtenir plus d'informations sur la façon d'installer de nouveaux paquets dans Red Hat Enterprise Linux.

2. Appuyez sur la touche **`Super`** pour entrer dans la vue d'ensemble des activités, tapez **`Startup Applications`** et appuyez sur **`Entrée`**. L'outil **Startup Applications Preferences** s'affiche. L'onglet contenant une liste de programmes de démarrage disponibles s'affichera par défaut. La touche **`Super`** apparait dans une variété de formes, selon le clavier et autre matériel, mais souvent en tant que touche Commande ou Windows, généralement à gauche de la barre d'espace.

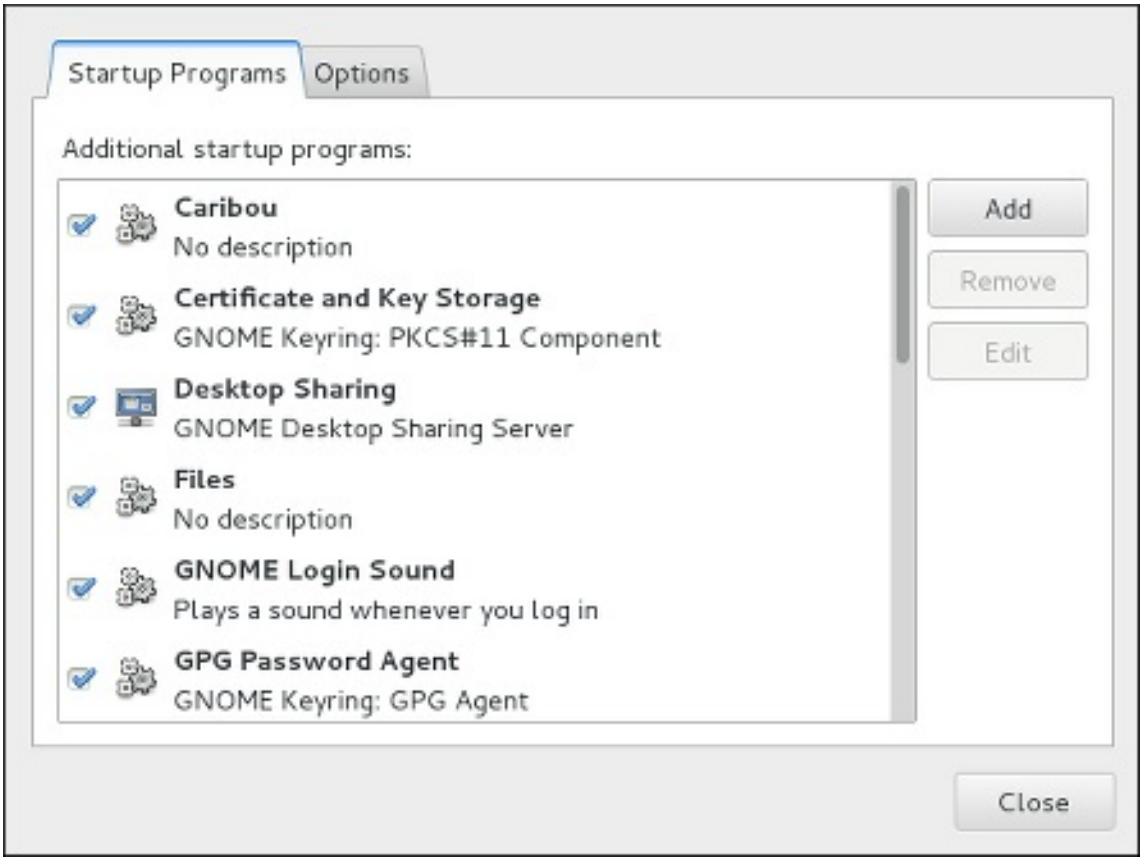


Figure 10.1. Préférences des applications au démarrage

3. Cliquer sur le bouton **Add** à droite, et saisir **/usr/bin/ssh-add** dans le champ **Command**.

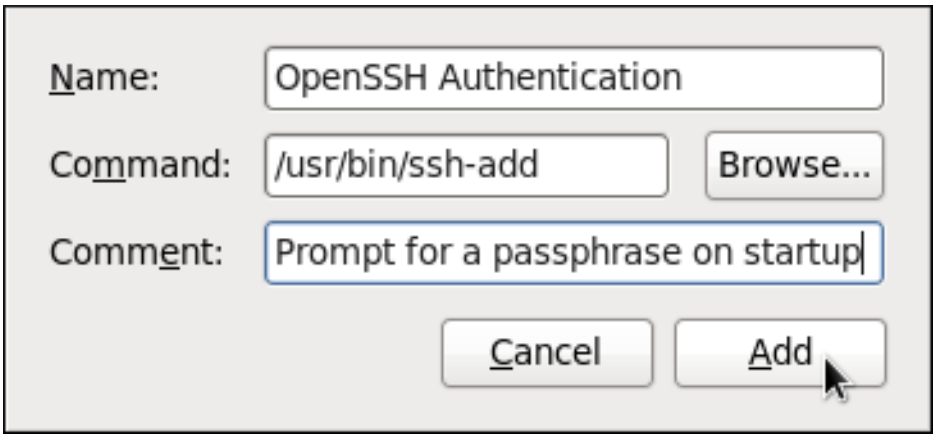


Figure 10.2. Ajouter une nouvelle application

4. Cliquer sur **Add** et vérifiez que la case qui se trouve à côté de l'élément qui vient d'être ajouté est sélectionnée.

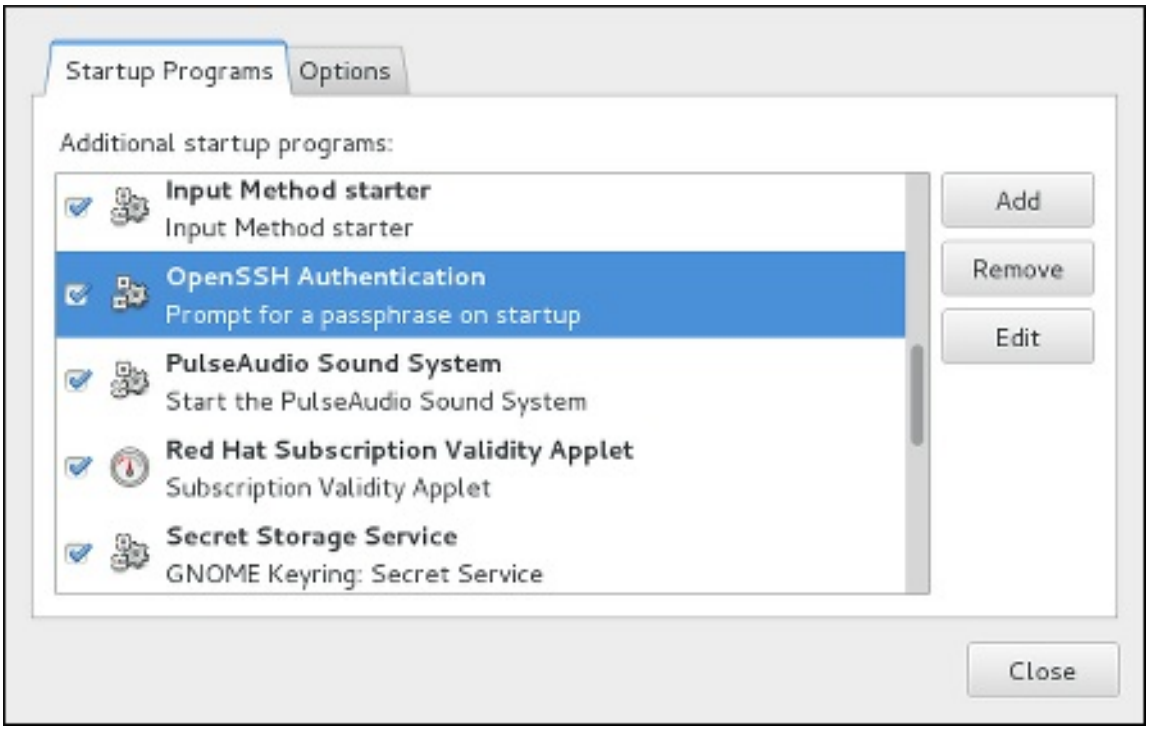


Figure 10.3. Activer l'application

5. Déconnecter et reconnecter. Une boîte de dialogue apparaîtra pour vous demander de mettre votre phrase de passe. À partir de ce moment, vous ne devriez pas être sollicité de donner un mot de passe par les commandes **ssh**, **scp**, ou **sftp**.

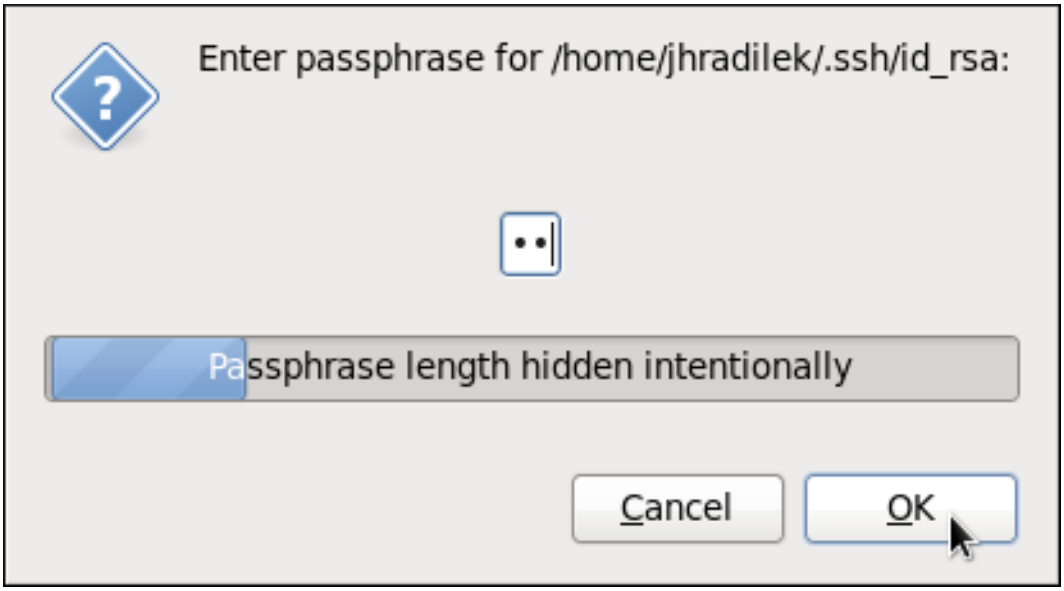


Figure 10.4. Saisir une phrase de passe

Afin de sauvegarder votre phrase de passe pour une certaine invite de shell, utilisez la commande suivante :

```
~]$ ssh-add
Saisir la phrase de passe pour /home/USER/.ssh/id_rsa :
```

Notez que lorsque vous vous déconnectez, votre phrase de passe sera oubliée. Vous devez exécuter la commande chaque fois que vous vous connectez à une console virtuelle ou à une fenêtre de terminal.

10.3. CLIENTS OPENSSH

Pour vous connecter à un serveur OpenSSH depuis une machine cliente, vous devez avoir installé le paquet openssh-clients (voir Section 8.2.4, « Installation de paquets » pour plus d'informations sur la façon d'installer de nouveaux paquets dans Red Hat Enterprise Linux).

10.3.1. Comment se servir de l'utilitaire ssh

L'utilitaire **ssh** vous permet de vous connecter à une machine distante et à y exécuter des commandes à cet endroit. Cela remplace les programmes **rlogin**, **rsh**, et **telnet**.

De même que pour la commande **telnet**, connectez-vous à une machine distante par la commande suivante :

```
ssh hostname
```

Ainsi, pour vous connecter à une machine distante nommée **penguin.example.com**, saisir ce qui suit quand on vous y invite :

```
~]$ ssh penguin.example.com
```

Cela vous enregistrera avec le même nom d'utilisateur que vous utilisez sur votre machine locale. Si vous souhaitez spécifier un nom d'utilisateur différent, utilisez une commande de la forme suivante :

```
ssh username@hostname
```

Ainsi, pour vous connecter à **penguin.example.com** en tant que **USER**, saisir :

```
~]$ ssh USER@penguin.example.com
```

La première fois que vous initierez une connexion, vous verrez apparaître un message similaire à celui-ci :

```
The authenticity of host 'penguin.example.com' can't be established.
ECDSA key fingerprint is 256
da:24:43:0b:2e:c1:3f:a1:84:13:92:01:52:b4:84:ff.
Are you sure you want to continue connecting (yes/no)?
```

Les utilisateurs doivent toujours vérifier si l'empreinte est correcte avant de répondre à la question dans cette boîte de dialogue. L'utilisateur peut demander à l'administrateur du serveur de bien confirmer que la clé est correcte. Cela devrait être fait de manière sécurisée et préalablement convenue. Si l'utilisateur a accès aux clés d'hôte du serveur, l'empreinte digitale peut être vérifiée à l'aide de la commande **ssh-keygen**, comme suit :

```
~]# ssh-keygen -l -f /etc/ssh/ssh_host_ecdsa_key.pub
256 da:24:43:0b:2e:c1:3f:a1:84:13:92:01:52:b4:84:ff   (ECDSA)
```

Tapez **yes** pour accepter la clé et confirmer la connexion. Vous verrez une notice apparaître vous indiquant que le serveur a été ajouté à la liste des hôtes connus et une invite vous demandant votre mot de passe :

```
Warning: Permanently added 'penguin.example.com' (ECDSA) to the list of
known hosts.
USER@penguin.example.com's password:
```

IMPORTANT

Si la clé de l'hôte du serveur SSH change, le client informera l'utilisateur que la connexion ne peut pas avoir lieu tant que la clé du serveur hôte n'est pas supprimée du fichier **~/.ssh/known_hosts**. Avant de procéder, cependant, contactez l'administrateur systèmes du serveur SSH pour vérifier que le serveur n'est pas compromis.

Pour supprimer une clé du fichier **~/.ssh/known_hosts**, lancez la commande suivante :

```
~]# ssh-keygen -R penguin.example.com
# Host penguin.example.com found: line 15 type ECDSA
/home/USER/.ssh/known_hosts updated.
Original contents retained as /home/USER/.ssh/known_hosts.old
```

Une fois que vous aurez saisi le mot de passe, vous apercevrez une invite de shell pour la machine distante.

Sinon, le programme **ssh** peut être utilisé pour exécuter une commande sur la machine distante sans qu'il soit nécessaire de vous connecter à une invite de shell :

```
ssh [username@]hostname command
```

Ainsi, le fichier **/etc/redhat-release** vous donne des informations sur la version Red Hat Enterprise Linux. Pour voir le contenu de ce fichier dans **penguin.example.com**, saisissez :

```
~]$ ssh USER@penguin.example.com cat /etc/redhat-release
USER@penguin.example.com's password:
Red Hat Enterprise Linux Server release 7.0 (Maipo)
```

Une fois que vous aurez saisi le mot de passe qui convient, le nom d'utilisateur apparaîtra, et vous pourrez retourner à votre invite de shell locale.

10.3.2. rsh

La commande **scp** peut être utilisée pour transférer des fichers entre des machines à travers une connexion cryptée sécurisée. Le concept ressemble à celui de **rcp**.

Pour transférer un fichier local dans un système distant, utiliser une commande de la forme suivante :

```
scp localfile username@hostname:remotefile
```

Ainsi, si vous souhaitez transférer **taglist.vim** vers une machine distante nommée **penguin.example.com**, saisissez ce qui suit dans l'invite de commande :

```
~]$ scp taglist.vim USER@penguin.example.com:.vim/plugin/taglist.vim
USER@penguin.example.com's password:
taglist.vim                                   100%  144KB 144.5KB/s
00:00
```

Vous pouvez spécifier plusieurs fichiers à la fois. Pour transférer les contenus de **.vim/plugin/** dans le même répertoire d'une machine distante **penguin.example.com**, saisissez la commande suivante :

```
~]$ scp .vim/plugin/* USER@penguin.example.com:.vim/plugin/
USER@penguin.example.com's password:
closetag.vim                                  100%   13KB  12.6KB/s
00:00
snippetsEmu.vim                               100%   33KB  33.1KB/s
00:00
taglist.vim                                   100%  144KB 144.5KB/s
00:00
```

Afin d'autoriser les utilisateurs à créer des fichiers dans le répertoire, utilisez la commande suivante :

```
scp username@hostname:remotefile localfile
```

Ainsi, pour télécharger le fichier de configuration **.vimrc** d'une machine distante, saisissez :

```
~]$ scp USER@penguin.example.com:.vimrc .vimrc
USER@penguin.example.com's password:
.vimrc                                        100% 2233     2.2KB/s
00:00
```

10.3.3. rsh

L'utilitaire **sftp** peut être utilisé pour ouvrir une session FTP interactive, sécurisée. Dans le concept, c'est similaire à **ftp** sauf qu'on utilise une connexion sécurisée, cryptée.

Pour vous connecter à un système distant, utiliser une commande qui ressemble à ce qui suit :

```
sftp username@hostname
```

Ainsi, pour vous connecter à une machine distante nommée **penguin.example.com**, avec **USER** comme nom d'utilisateur, saisissez ce qui suit :

```
~]$ sftp USER@penguin.example.com
USER@penguin.example.com's password:
Connected to penguin.example.com.
sftp>
```

Une fois que vous aurez saisi le mot de passe qui convient, vous verrez une invite se présenter. L'utilitaire **sftp** accepte un ensemble de commandes qui ressemble à celles utilisées avec **ftp** (voir Tableau 10.3, « Une sélection de commandes sftp disponibles »).

Tableau 10.3. Une sélection de commandes sftp disponibles

Commande	Description
ls [*directory*]	Lister le contenu du *directory* (répertoire) distant. S'il n'y en aucun, on utilisera le répertoire de travail en cours par défaut.
cd *directory*	Changer le répertoire de travail distant à *répertoire*.
mkdir *répertoire*	--smbservers=<*server*>
rmdir *chemin*	Supprimer un *répertoire* distant.
put *fichier local* [*fichier distant*]	Transférer le *fichier local* à une machine distante.
get *fichier distant* [*fichier local*]	Transférer le *fichier distant* à partir d'une machine distante.

Pour obtenir une liste complète des commandes disponibles, voir la page man de **sftp**(1).

10.4. BEAUCOUP PLUS QU'UN SHELL SÉCURISÉ

Une interface sécurisée en ligne de commandes n'est qu'une utilisation parmi tant d'autres, de SSH. En ayant la quantité nécessaire de bande passante, les sessions X11 peuvent être dirigées sur un canal SSH. Sinon, en utilisant la retransmission TCP/IP, les connexions par port entre les systèmes, considérées auparavant comme étant non-sécurisées, peuvent être mappées à des canaux SSH spécifiques.

10.4.1. Transfert X11

Pour ouvrir une session X11 sur une connexion SSH, utiliser une commande qui ressemble à ceci :

```
ssh -Y nom d'utilisateur@nom d'hôte
```

Ainsi, pour vous connecter à une machine distante nommée **penguin.example.com**, avec **USER** comme nom d'utilisateur, saisissez ce qui suit :

```
~]$ ssh -Y USER@penguin.example.com
mot de passe de USER@penguin.example.com :
```

Lorsqu'un programme X est exécuté à partir d'une invite du shell sécurisée, le client et le serveur SSH créent un nouveau canal sécurisé et les données du programme X sont ensuite envoyées à l'ordinateur client via ce canal d'une manière transparente.

Notez que le système X Window doit être installé sur le système distant avant que la transmission X11 puisse avoir lieu. Saisir la commande suivante en tant qu'utilisateur **root** pour installer le groupe de packages X11 :

```
~]# yum group install "X Window System"
```

Pour obtenir plus d'informations sur les groupes de packages, voir Section 8.3, « Utiliser des groupes de paquets ».

Le transfert X11 peut être très utile. Par exemple, le transfert X11 peut être utilisé pour créer une session interactive, sécurisée de l'utilitaire **Print Settings**. Pour cela, connectez-vous au serveur en utilisant **ssh** et saisissez :

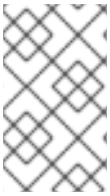
```
~]$ system-config-printer &
```

L'outil **Print Settings** apparaîtra, permettant à l'utilisateur distant de configurer l'impression sur le système distant en toute sécurité.

10.4.2. Réacheminement de port

SSH peut sécuriser les protocoles **TCP/IP** normalement non sécurisés par le réacheminement de port. Quand vous utilisez cette technique, le serveur SSH fait figure de conduit crypté vers le client SSH.

La redirection de port fonctionne en mappant un port local du client à un port distant sur le serveur. SSH peut mapper n'importe quel port du serveur sur un port de client. Les numéros de port n'ont pas besoin de correspondre pour cette façon de travailler.



NOTE

Pour configurer la retransmission de port de manière à ce qu'elle écoute sur les ports inférieurs à 1024, il est nécessaire d'avoir un accès de niveau super-utilisateur (ou root) **root**.

Pour créer un canal de réacheminement de port TCP/IP qui écoute les connexions sur le **localhost**, utiliser une commande de la forme suivante :

```
ssh -L port-local:nom d'hôte distant:port-distant nom d'utilisateur@nom
d'hôte
```

Ainsi, pour vérifier un email sur un serveur nommé **mail.example.com** qui utilise **POP3** par l'intermédiaire d'une connexion cryptée, utiliser la commande suivante :

```
~]$ ssh -L 1100:mail.example.com:110 mail.example.com
```

Une fois que le canal de réacheminement de port est mis en place entre la machine cliente et le serveur de messagerie, envoyez un mail client POP3 pour utiliser le port **1100** sur le **localhost** pour vérifier s'il y a de nouveaux messages. Toutes les requêtes envoyées au port **1100** sur le système client seront redirigées en toute sécurité vers le serveur **mail.example.com**.

Si **mail.example.com** n'exécute pas sur un serveur SSH, mais qu'une autre machine le fasse sur le même réseau, SSH peut toujours être utilisé pour sécuriser une partie de la connexion. Cependant, il vous faudra une commande légèrement différente :

```
~]$ ssh -L 1100:mail.example.com:110 other.example.com
```

Dans cet exemple, les demandes POP3 du port **1100** de la machine cliente sont transmises via la connexion SSH sur le port **22** au serveur SSH, **other.example.com**. Ensuite, **other.example.com** se connectera au port **110** sur **mail.exemple.com** pour vérifier les nouveaux messages. Notez que lorsque vous utilisez cette technique, seule la connexion entre le système client et le serveur SSH **other.example.com** est sûre.

Le réacheminement de port peut également être utilisé pour obtenir des informations en toute sécurité à travers les pare-feu de réseau. Si le pare-feu est configuré pour autoriser le trafic SSH via son port standard (c'est-à-dire le port 22) mais bloque l'accès aux autres ports, une connexion entre deux hôtes utilisant les ports bloqués est toujours possible si on réoriente leur communication par une connexion SSH établie.

IMPORTANT

L'utilisation de la fonctionnalité de réacheminement de port pour transférer des connexions de cette façon permet à tout utilisateur du système client de se connecter à ce service. Si le système client est compromis, les pirates auront également accès aux services retransmis.

Les administrateurs de système qui s'inquiètent de la fonctionnalité de réacheminement de port peut désactiver cette fonctionnalité sur le serveur en spécifiant un paramètre **No** sur la ligne **AllowTcpForwarding** dans **/etc/ssh/sshd_config** et redémarrer le service **sshd**.

10.5. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur la manière de configurer ou de se connecter au serveur OpenSSH sur Red Hat Enterprise Linux, veuillez consulter les ressources ci-dessous :

Documentation installée

- **sshd**(8) — la page man des options de ligne de commande de documents de démon **sshd** disponibles et la liste complète des fichiers et répertoires de configuration pris en charge.
- **ssh**(1) — la page man pour l'application client **ssh** fournit une liste complète des options de ligne de commande disponibles et des fichiers et répertoires de configuration.
- **scp**(1) — la page man de l'utilitaire **scp** fournit une description plus détaillée de cet utilitaire et de son utilisation.
- **sftp**(1) — la page man de l'utilitaire **sftp**.

- **ssh-keygen**(1) — la page man de l'utilitaire **ssh-keygen** documente en détails comment l'utiliser pour générer, gérer, et convertir les clés d'authentification utilisées par **ssh**.
- **ssh_config**(5) — la page man nommée **ssh_config** documente les options de configuration de client SSH disponibles.
- **sshd_config**(5) — la page man nommée **sshd_config** fournit une description compléte des options de configuration de démon SSH disponibles.

Documentation en ligne

- OpenSSH Home Page — la page d'accueil d'OpenSSH contient encore davantage de documentations, une foire aux questions, des liens aux listes d'email, des rapports de bogues, et autres informations utiles.
- OpenSSL Home Page — page d'accueil d'OpenSSL contenant davantage de documentation, une foire aux questions, des liens de listes de diffusion, ainsi que d'autres ressources utiles.

Voir aussi

- Chapitre 5, *Obtention de privilèges* documente la façon d'obtenir des privilèges administratifs en utilisant les commandes **su** et **sudo**.
- Chapitre 9, *Gérer les services avec systemd* fournit davantage d'informations sur systemd et documente comment utiliser la commande **systemctl** pour gérer les services système.

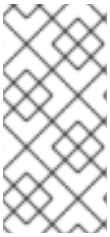
---

[1] A multiplexed connection consists of several signals being sent over a shared, common medium. With SSH, different channels are sent over a common secure connection.

CHAPITRE 11. TIGERVNC

TigerVNC (de l'anglais, « Tiger Virtual Network Computing ») est un système pour le partage de bureau graphique vous permettant de contrôler d'autres ordinateurs à distance.

TigerVNC fonctionne sur le principe client-serveur : un **serveur** partage sa sortie (**vncserver**) et un **client** (**vncviewer**) se connecte au serveur.



NOTE

Contrairement aux distributions précédentes de Red Hat Enterprise Linux, sur Red Hat Enterprise Linux 7 **TigerVNC** utilise le démon de gestion de systèmes **systemd** pour sa configuration. Le fichier de configuration **/etc/sysconfig/vncserver** a été remplacé par **/etc/systemd/system/vncserver@.service**.

11.1. SERVEUR VNC

vncserver est un utilitaire qui lance un bureau VNC (Virtual Network Computing). Il exécute **Xvnc** avec les options appropriées et lance un gestionnaire de fenêtre sur le bureau VNC. **vncserver** autorise les utilisateurs à exécuter des sessions séparées en parallèle sur une machine qui peut ensuite être accédée par un nombre illimité de clients, quelle que soit leur origine.

11.1.1. Installer un serveur VNC

Pour installer le serveur **TigerVNC**, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install tigervnc-server
```

11.1.2. Configurer un serveur VNC

Le serveur VNC peut être configuré pour lancer un affichage pour un ou plusieurs utilisateurs, à condition que les comptes de ces utilisateurs existent sur le système, avec des paramètres optionnels comme les paramètres d'affichage, l'adresse et le port réseau, et les paramètres de sécurité.

Procédure 11.1. Configurer un affichage VNC pour un utilisateur unique

1. Un fichier de configuration nommé **/etc/systemd/system/vncserver@.service** est requis. Pour créer ce fichier, copiez le fichier **/usr/lib/systemd/system/vncserver@.service** en tant qu'utilisateur **root** :

   ```
   ~]# cp /usr/lib/systemd/system/vncserver@.service
   /etc/systemd/system/vncserver@.service
   ```

   Il n'est pas nécessaire d'inclure le numéro d'affichage dans le nom du fichier car **systemd** crée automatiquement l'instance nommée de manière appropriée en mémoire à la demande, en remplaçant **'%i'** dans le fichier du service par le numéro d'affichage. Pour un utilisateur unique, il n'est pas nécessaire de renommer le fichier. Pour de multiples utilisateurs, un fichier de service nommé de manière unique est requis pour chaque utilisateur, par exemple, en ajoutant le nom d'utilisateur au nom du fichier. Veuillez consulter Section 11.1.2.1, « Configurer un serveur VNC pour deux utilisateurs » pour obtenir plus de détails.

2. Modifiez **/etc/systemd/system/vncserver@.service**, en remplaçant *USER* par le nom de l'utilisateur. Veuillez laisser les lignes restantes du fichier intactes. L'argument **-geometry** spécifie la taille du bureau VNC à créer. Par défaut, celle-ci est définie sur **1024x768**.

```
ExecStart=/usr/sbin/runuser -l USER -c "/usr/bin/vncserver %i -
geometry 1280x1024"
PIDFile=/home/USER/.vnc/%H%i.pid
```

3. Enregistrez les changements.

4. Pour que les changements entrent en vigueur immédiatement, veuillez exécuter la commande suivante :

```
~]# systemctl daemon-reload
```

5. Paramétrez le mot de passe de l'utilisateur (ou des utilisateurs) défini(s) dans le fichier de configuration. Remarquez que vous devrez d'abord passer de l'utilisateur **root** à utilisateur *USER*.

```
~]# su - USER
~]$ vncpasswd
Password:
Verify:
```

IMPORTANT

Le mot de passe stocké n'est pas chiffré, toute personne ayant accès au fichier de mot de passe peut trouver le mot de passe en texte clair.

Procédez avec la Section 11.1.3, « Lancer le serveur VNC ».

11.1.2.1. Configurer un serveur VNC pour deux utilisateurs

Si vous souhaitez configurer plus d'un utilisateur sur la même machine, veuillez créer différents fichiers de service de type modèle, un par utilisateur.

1. Créez deux fichiers de service, par exemple **vncserver-*USER_1*@.service** et **vncserver-*USER_2*@.service**. Veuillez substituer *USER* dans ces deux fichiers par le nom d'utilisateur correct.

2. Définissez les mots de passe pour les deux utilisateurs :

```
~]$ su - USER_1
~]$ vncpasswd
Password:
Verify:
~]$ su - USER_2
~]$ vncpasswd
Password:
Verify:
```

11.1.3. Lancer le serveur VNC

Pour lancer ou activer le service, spécifiez le numéro d'affichage directement dans la commande. Le fichier configuré ci-dessus dans Procédure 11.1, « Configurer un affichage VNC pour un utilisateur unique » sert de modèle, dans lequel **%i** est remplacé par le numéro d'affichage par **systemd**. Avec un numéro d'affichage valide, exécutez la commande suivante :

```
~]# systemctl start vncserver@:display_number.service
```

Vous pouvez également autoriser le service à être lancé automatiquement lors du démarrage système. Puis, une fois connecté, **vncserver** est automatiquement lancé. En tant qu'utilisateur **root**, veuillez exécuter une commande comme suit :

```
~]# systemctl enable vncserver@:display_number.service
```

À ce moment, d'autres utilisateurs sont en mesure d'utiliser un programme visionneur VNC pour se connecter au serveur VNC en utilisant le numéro d'affichage et le mot de passe défini. À condition qu'un bureau graphique soit installé, une instance de ce bureau sera affichée. Il ne s'agira pas de la même instance que celle qui est actuellement affichée sur la machine cible.

11.1.3.1. Configurer un serveur VNC pour deux utilisateurs et deux affichages différents

Pour les deux serveurs VNC configurés, vncserver-USER_1@.service et vncserver-USER_2@.service, vous pouvez activer différents numéros d'affichage. Par exemple, les commande suivantes amèneront le serveur VNC de l'utilisateur USER_1 à être lancé sur l'affichage 3, et au serveur VNC de l'utilisateur USER_2 à être lancé sur l'affichage 5 :

```
~]# systemctl start vncserver-USER_1@:3.service
~]# systemctl start vncserver-USER_2@:5.service
```

11.1.4. Fermer une session VNC

De même que pour l'activation du service **vncserver**, il est possible de désactiver le lancement automatique du service pendant le démarrage système :

```
~]# systemctl disable vncserver@:display_number.service
```

Sinon, lorsque votre système est en cours d'utilisation, vous pouvez arrêter le service en exécutant la commande suivante, en tant qu'utilisateur **root** :

```
~]# systemctl stop vncserver@:display_number.service
```

11.2. PARTAGER UN BUREAU EXISTANT

Dnas une installation normale, un utilisateur se connecte à un ordinateur de bureau fourni par un serveur X avec **0** affiché. Un utilisateur peut partager son ordinateur de bureau par l'intermédiaire du serveur **TigerVNC** **x0vncserver**.

Procédure 11.2. Partager un ordinateur de bureau X

Pour partager l'ordinateur de bureau en tant qu'utilisateur connecté, avec **x0vncserver**, procédez ainsi :

1. Saisir la commande suivante en tant qu'utilisateur **root**

```
~]# yum install tigervnc-server
```

2. Voir le mot de passe VNC de l'utilisateur :

```
~]$ vncpasswd
Password:
Verify:
```

3. Exécutez la commande suivante en tant qu'utilisateur :

```
~]$ x0vncserver -PasswordFile=.vnc/passwd -AlwaysShared=1
```

Si le parefeu est configuré de façon à atoriser des connexions au port **5900**, le visionneur distant peut maintenant se connecter pour afficher **0**, et voir le bureau utilisateur connecté. Voir Section 11.3.2.1, « Configurer le pare-feu pour VNC » pour obtenir des informations sur la façon de configurer le parefeu.

11.3. VISIONNEUR VNC

vncviewer est un programme qui affiche les interfaces utilisateur graphique et contrôle **vncserver** à distance.

Pour opérer **vncviewer**, il existe un menu contextuel qui contient des entrées permettant d'effectuer diverses actions, comme basculer en et hors du mode plein écran, ou quitter le visionneur. Alternativement, il est possible d'opérer **vncviewer** via le terminal. Saisissez **vncviewer -h** sur la ligne de commande pour répertorier les paramètres de **vncviewer**.

11.3.1. Installer le visionneur VNC Viewer

Pour installer le client **TigerVNC vncviewer**, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install tigervnc
```

11.3.2. Connexion au serveur VNC

Une fois que le serveur VNC est configuré, vous pouvez vous y connecter à partir de n'importe quel visionneur VNC.

Procédure 11.3. Connexion à un serveur VNC à l'aide d'une interface utilisateur graphique

1. Saisissez la commande **vncviewer** sans le moindre argument, et l'utilitaire **VNC Viewer: Connection Details** s'affichera. Il demandera alors sur quel serveur VNC la connexion doit être établie.

2. Si nécessaire, pour empêcher de déconnecter toute connexion VNC sur le même affichage, sélectionnez l'option permettant de partager le bureau, comme suit :

   a. Sélectionnez le bouton **Options**.

   b. Sélectionnez l'onglet **Divers**.

   c. Sélectionnez le bouton **Partagé**.

d. Appuyez sur **Valider** pour retourner au menu principal.

3. Veuillez saisir une adresse et un numéro d'affichage auquel vous connecter :

```
address:display_number
```

4. Veuillez appuyer sur **Connecter** pour vous connecter à l'affichage du serveur VNC.

5. Il vous sera demandé de saisir le mot de passe VNC. Ce sera le mot de passe VNC pour l'utilisateur correspondant au numéro d'affichage, à moins qu'un mot de passe VNC global par défaut n'ait été paramétré.

   Une fenêtre apparaît affichant le bureau du serveur VNC. Remarquez qu'il ne s'agit pas du bureau qu'un utilisateur normal peut voir, il s'agit d'un bureau Xvnc.

Procédure 11.4. Connexion à un serveur VNC à l'aide de l'interface de ligne de commande

1. Veuillez saisir la commande **viewer** avec l'adresse et le numéro d'affichage comme arguments :

```
vncviewer address:display_number
```

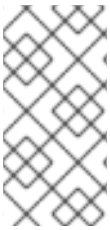
   , où *address* est une adresse **IP** ou un nom d'hôte.

2. Authentifiez-vous en saisissant le mot de passe VNC. Ce sera le mot de passe VNC de l'utilisateur correspondant au numéro d'affichage, à moins qu'un mot de passe VNC global par défaut n'ait été paramétré.

3. Une fenêtre apparaît affichant le bureau du serveur VNC. Remarquez qu'il ne s'agit pas du bureau qu'un utilisateur normal peut voir, il s'agit du bureau Xvnc.

11.3.2.1. Configurer le pare-feu pour VNC

Lors de l'utilisation d'une connexion non chiffrée, **firewalld** peut bloquer la connexion. Pour autoriser **firewalld** à transférer les paquets VNC, vous pouvez ouvrir des ports spécifiques au trafic **TCP**. Lors de l'utilisation de l'option **-via**, le trafic est redirigé sur **SSH**, qui est activé par défaut dans **firewalld**.



NOTE

Le port par défaut du serveur VNC est le port 5900. Pour atteindre le port par lequel un bureau distant est accessible, faîtes la somme du port par défaut et du numéro d'affichage assigné à l'utilisateur. Par exemple, pour le second affichage : 2 + 5900 = 5902.

Pour les affichages **0** à **3**, utilisez la prise en charge **firewalld** du service VNC par l'option **service** comme décrit ci-dessous. Remarquez que pour les numéros d'affichage supérieurs à **3**, les ports correspondants devront être spécifiquement ouverts, comme expliqué dans Procédure 11.6, « Ouvrir des ports sur firewalld ».

Procédure 11.5. Activer le service VNC sur firewalld

1. Veuillez exécuter la commande suivante pour afficher les informations concernant les paramètres **firewalld** :

```
~]$ firewall-cmd --list-all
```

2. Pour autoriser toutes les connexions VNC en provenance d'une adresse particulière, veuillez utiliser une commande comme suit :

```
~]# firewall-cmd --add-rich-rule='rule family="ipv4" source
address="192.168.122.116" service name=vnc-server accept'
success
```

Veuillez noter que ces changements ne persisteront pas lors d'un nouveau démarrage. Pour rendre les changements au parefeu permanents, répétez la commande an ajoutant l'option **--permanent**. Veuillez consulter le Guide de sécurité Red Hat Enterprise Linux 7 pour obtenir davantage d'informations sur l'utilisation de commandes en langage riche pour pare-feux.

3. Pour vérifier les paramètres ci-dessus, veuillez utiliser une commande comme suit :

```
~]# firewall-cmd --list-all
public (default, active)
  interfaces: bond0 bond0.192
  sources:
  services: dhcpv6-client ssh
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:
 rule family="ipv4" source address="192.168.122.116" service
name="vnc-server" accept
```

Pour ouvrir un port spécifique ou une gamme de ports, utilisez l'option **--add-port** sur l'outil de ligne de commande **firewall-cmd**. Par exemple, l'affichage VNC **4** requiert que le port **5904** soit ouvert au trafic **TCP**.

Procédure 11.6. Ouvrir des ports sur firewalld

1. Pour ouvrir un port au trafic **TCP** dans la zone publique, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# firewall-cmd --zone=public --add-port=5904/tcp
success
```

2. Pour afficher les ports actuellement ouverts à la zone publique, veuillez exécuter une commande comme suit :

```
~]# firewall-cmd --zone=public --list-ports
5904/tcp
```

Un port peut être supprimé en utilisant la commande **firewall-cmd --zone=*zone* --remove-port=*number/protocol***.

Veuillez noter que ces changements ne persisteront pas lors d'un nouveau démarrage. Pour rendre les changements au parefeu permanents, répétez la commande an ajoutant l'option **--permanent**. Pour obtenir davantage d'informations sur l'ouverture et la fermeture des ports de **firewalld**, veuillez

consulter le Guide de sécurité Red Hat Enterprise Linux 7 .

11.3.3. Connexion à un serveur VNC à l'aide de SSH

VNC est un protocole réseau en texte clair sans la moindre sécurité contre des attaques possibles sur les communications. Pour rendre ces communications sécurisées, vous pouvez chiffrer votre connexion serveur-client en utilisant l'option **-via**. Cela créera un tunnel **SSH** entre le serveur VNC et le client.

Le format de la commande pour chiffrer une connecion serveur-client VNC est comme suit :

```
vncviewer -via user@host:display_number
```

Exemple 11.1. En utilisant l'option -via

1. Pour se connecter à un serveur VNC en utilisant **SSH**, veuillez saisir une commande comme suit :

   ```
   ~]$ vncviewer -via USER_2@192.168.2.101:3
   ```

2. Lorsque cela vous est demandé, saisissez le mot de passe, et confirmez en appuyant sur **Entrée**.

3. Une fenêtre avec un bureau distant s'affichera sur votre écran.

Restreindre l'accès VNC

Si vous souhaitez uniquement avoir des connexions chiffrées, il est possible d'empêcher les connexions non chiffrées en utilisant l'option **-localhost** dans le fichier **systemd.service**, la ligne ExecStart :

```
ExecStart=/usr/sbin/runuser -l user -c "/usr/bin/vncserver -localhost %i"
```

Cela empêchera **vncserver** d'accepter toute connexion, sauf les connexions de l'hôte local et les connexions transférées par port ayant été envoyées à l'aide de **SSH**, conséquemment à l'utilisation de l'option **-via**.

Pour obtenir davantage d'informations sur l'utilisation de **SSH**, veuillez consulter le Chapitre 10, *OpenSSH*.

11.4. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur TigerVNC, veuillez consulter les ressources répertoriées ci-dessous.

Documentation installée

- **vncserver(1)** — la page du manuel de l'utilitaire du serveur VNC.
- **vncviewer(1)** — la page du manuel du visionneur VNC.
- **vncpasswd(1)** — la page du manuel de la commande du mot de passe VNC.
- **Xvnc(1)** — la page du manuel des options de configuration du serveur Xvnc.

- **x0vncserver(1)** — la page du manuel du serveur **TigerVNC** pour le partage de serveurs X existants.

PARTIE V. SERVEURS

Cette partie traite des différents sujets liés aux serveurs, comme la manière de paramétrer un serveur web ou de partager des fichiers et répertoires sur un réseau.

CHAPITRE 12. SERVEURS WEB

Un *serveur web* est un service réseau qui remet un contenu à un client à travers le web. Habituellement, cela signifie des pages web, mais tout autre document peut également être remis. Les serveurs web sont également appelés des serveurs HTTP , car ils utilisent le *protocole de transport hypertexte* (HTTP).

12.1. SERVEUR APACHE HTTP

Le serveur web disponible sur Red Hat Enterprise Linux 7 est la version 2.4 du **Serveur Apache HTTP**, **`httpd`**, un serveur web open source développé par la Fondation Apache Software.

Si vous effectuez une mise à niveau à partir d'une version précédente de Red Hat Enterprise Linux, vous devrez mettre à jour la configuration du service **`httpd`**. Cette section traite de certaines nouvelles fonctionnalités ajoutées, souligne les changements importants entre les versions 2.2 et 2.4 du serveur Apache HTTP, et vous guide à travers la mise à jour des anciens fichiers de configuration.

12.1.1. Changements notables

Le serveur Apache HTTP sur Red Hat Enterprise Linux 7 propose les changements suivants comparé à Red Hat Enterprise Linux 6 :

Contrôle du service httpd

Avec la migration vers l'extérieur des scripts init SysV, les administrateurs de serveur devront se mettre à utiliser les commandes **`apachectl`** et **`systemctl`** pour contrôler le service, à la place de la commande **`service`**. Les exemples suivants sont spécifiques au service **`httpd`**.

La commande :

```
service httpd graceful
```

est remplacée par

```
apachectl graceful
```

. Le fichier unité **`systemd`** de **`httpd`** a un comportement différent du script init, comme suit :

- Un redémarrage correct est utilisé par défaut lorsque le service est rechargé.
- Un arrêt correct est utilisé par défaut lorsque le service est arrêté.

La commande :

```
service httpd configtest
```

est remplacée par

```
apachectl configtest
```

Répertoire /tmp privé

Pour améliorer la sécurité du système, le fichier unité **`systemd`** exécute le démon **`httpd`** en utilisant un répertoire privé **`/tmp`**, séparé du répertoire **`/tmp`** du système.

Structure de la configuration

Les fichiers de configuration qui chargent les modules sont désormais placés dans le répertoire **/etc/httpd/conf.modules.d/**. Les paquets fournissant des modules supplémentaires qui peuvent être chargés pour **httpd**, comme php, placeront un fichier dans ce répertoire. Une directive **Include** avant la section principale du fichier **/etc/httpd/conf/httpd.conf** est utilisée pour inclure les fichiers dans le répertoire **/etc/httpd/conf.modules.d/**. Cela signifie que tous les fichiers de configuration de **conf.modules.d/** sont traités avant le corps principal de **httpd.conf**. Une directive **IncludeOptional** pour les fichiers dans le répertoire **/etc/httpd/conf.d/** est placée à la fin du fichier **httpd.conf**. Cela signifie que les fichiers qui se trouvent dans **/etc/httpd/conf.d/** sont désormais traités après le corps principal de **httpd.conf**.

Certains fichiers de configuration supplémentaires sont fournis par le paquet httpd :

- **/etc/httpd/conf.d/autoindex.conf** — Permet de configurer l'indexation du répertoire mod_autoindex.
- **/etc/httpd/conf.d/userdir.conf** — Permet de configurer l'accès aux répertoires utilisateurs. Par exemple, **http://example.com/~username/** ; ce type d'accès est désactivé par défaut pour des raisons de sécurité.
- **/etc/httpd/conf.d/welcome.conf** — Tout comme dans les versions précédentes, cela permet de configurer la page d'accueil affichée pour **http://localhost/** lorsqu'aucun contenu n'est présent.

Configuration par défaut

Un fichier **httpd.conf** minimal est désormais fourni par défaut. De nombreux paramètres de configuration courants, comme **Timeout** ou **KeepAlive**, ne sont plus explicitement configurés dans la configuration par défaut ; au lieu de ceux-ci, les paramètres codés de manière irréversibles seront utilisés par défaut. Les paramètres codés de manière irréversible par défaut pour toutes les directives de configuration sont spécifiés dans le manuel. Veuillez consulter la section intitulée « Documentation installable » pour obtenir davantage d'informations.

Modifications incompatibles de la syntaxe

Lors de la migration d'une configuration existante de **httpd 2.2** à **httpd 2.4**, un certain nombre de modifications incompatibles en arrière apportées à la syntaxe de la configuration **httpd** nécessiteront des changements. Veuillez consulter le document Apache suivant pour obtenir davantage d'informations concernant la mise à niveau de http://httpd.apache.org/docs/2.4/upgrading.html

Modèle de traitement

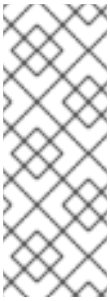
Dans les versions précédentes de Red Hat Enterprise Linux, différents modèles de multiples traitements (« *multi-processing models* », ou MPM) étaient disponibles aux différents binaires **httpd** : le modèle fourchu, « prefork », **/usr/sbin/httpd** ; et le modèle basé sur thread « worker », **/usr/sbin/httpd.worker**.

Sur Red Hat Enterprise Linux 7, seul un binaire **httpd** est utilisé, et trois MPM sont disponibles en tant que modules chargeables : « worker », « prefork » (par défaut), et « event ». Veuillez modifier le fichier de configuration **/etc/httpd/conf.modules.d/00-mpm.conf** comme requis, en ajoutant et supprimant le caractère dièse (#), afin qu'un seul des trois modules MPM soit chargé.

Modifications de paquets

Les modules d'authentification et d'autorisation LDAP sont désormais fournis dans un sous-paquet

séparé, mod_ldap. Le nouveau module **mod_session** et les modules d'aide associés sont fournis dans un nouveau sous-paquet, mod_session. Les nouveaux modules **mod_proxy_html** et **mod_xml2enc** sont fournis dans un nouveau sous-paquet, mod_proxy_html. Ces paquets se trouvent tous dans le canal « Optional ».



NOTE

Avant de vous abonner aux canaux « Optional » et « Supplementary », veuillez consulter les Détails de l'étendue de la couverture. Si vous décidez d'installer des paquets à partir de ces canaux, veuillez suivre les étapes documentées dans l'article nommé Comment accéder aux canaux « Optional » et « Supplementary » et aux paquets -devel en utilisant Red Hat Subscription Manager (RHSM) ? sur le Portail Client Red Hat.

Empaquetage des structures de systèmes de fichiers

Le répertoire **`/var/cache/mod_proxy/`** n'est plus fourni ; à la place, le répertoire **`/var/cache/httpd/`** est empaqueté avec un sous-répertoire **`proxy`** et **`ssl`**.

Le contenu empaqueté avec **`httpd`** a été déplacé de **`/var/www/`** à **`/usr/share/httpd/`** :

- **`/usr/share/httpd/icons/`** — Le répertoire contenant un ensemble d'icônes avec des indices de répertoire, auparavant contenus dans **`/var/www/icons/`**, a été déplacé sur **`/usr/share/httpd/icons/`**. Disponible sur **`http://localhost/icons/`** dans la configuration par défaut ; l'emplacement et la disponibilité des icônes est configurable dans le fichier **`/etc/httpd/conf.d/autoindex.conf`**.

- **`/usr/share/httpd/manual/`** — **`/var/www/manual/`** a été déplacé sur **`/usr/share/httpd/manual/`**. Ce répertoire, qui fait partie du paquet httpd-manual, contient la version HTML du manuel de **`httpd`**. Disponible sur **`http://localhost/manual/`** si le paquet est installé, l'emplacement et la disponibilité du manuel sont configurables dans le fichier **`/etc/httpd/conf.d/manual.conf`**.

- **`/usr/share/httpd/error/`** — **`/var/www/error/`** a été déplacé sur **`/usr/share/httpd/error/`**. Pages d'erreurs HTTP de langues multiples personnalisées. Non configuré par défaut, le fichier de configuration exemple est fourni sur **`/usr/share/doc/httpd-`*`VERSION`*`/httpd-multilang-errordoc.conf`**.

Authentifications, autorisations et contrôle des accès

Les directives de configuration utilisées pour contrôler l'authentification, les autorisations et le contrôle des accès ont changé de manière significative. Les fichiers de configuration existants qui utilisent les directives **`Order`**, **`Deny`** et **`Allow`** doivent être adaptés afin de pouvoir utiliser la nouvelle syntaxe **`Require`**. Veuillez consulter le document Apache suivant pour obtenir davantage d'informations : http://httpd.apache.org/docs/2.4/howto/auth.html

suexec

Pour améliorer la sécurité du système, le binaire **suexec** n'est plus installé par l'équivalent de l'utilisateur **`root`** ; au lieu de cela, il possède une fonctionnalité «bit Set» dans le système de fichiers, qui limite les restrictions. En conjonction avec ce changement, le binaire **suexec** n'utilise plus le fichier journal **`/var/log/httpd/suexec.log`**. Au lieu de cela, des messages de journalisation sont envoyés sur **syslog**. Par défaut, ceux-ci apparaîtront dans le fichier journal **`/var/log/secure`**.

Interface du module

Des modules binaires de tierce-partie créés avec **httpd 2.2** ne sont pas compatibles avec **httpd 2.4** à cause de changements apportés à l'interface du module **`httpd`**. De tels modules devront être ajustés comme nécessaire pour l'interface du module **httpd 2.4**, puis recréés. Une liste détaillée des changements d'API dans la version **`2.4`** sont disponibles ici : http://httpd.apache.org/docs/2.4/developer/new_api_2_4.html.

Le binaire **apxs** utilisé pour créer des modules source a été déplacé de **`/usr/sbin/apxs`** à **`/usr/bin/apxs`**.

Modules supprimés

Liste des modules **`httpd`** supprimés de Red Hat Enterprise Linux 7 :

mod_auth_mysql, mod_auth_pgsql

httpd 2.4 fournit la prise en charge de l'authentification de bases de données SQL de manière interne dans le module **mod_authn_dbd**.

mod_perl

mod_perl n'est pas officiellement pris en charge avec **httpd 2.4** en amont.

mod_authz_ldap

httpd 2.4 offre la prise en charge LDAP dans le sous-paquet mod_ldap en utilisant **mod_authnz_ldap**.

12.1.2. Mettre à jour la configuration

Pour mettre à jour les fichiers de configuration à partir de la version 2.2 d'Apache HTTP Server, veuillez effectuer les étapes suivantes :

1. Comme ils peuvent avoir été modifiés, assurez-vous que tous les noms de modules soient corrects. Ajustez la directive **`LoadModule`** pour chaque module dont le nom a changé.

2. Compilez à nouveau tous les modules de tierce-partie avant de tenter de les charger. Typiquement, cela signifie des modules d'authentification et d'autorisation.

3. Si vous utilisez le module **`mod_userdir`**, assurez-vous que la directive **`UserDir`** indiquant un nom de répertoire (habituellement **`public_html`**) soit effectivement fournie.

4. Si vous utilisez Apache HTTP Secure Server, veuillez consulter la Section 12.1.8, « Activer le module mod_ssl » pour obtenir des informations importantes sur l'activation du protocole SSL (« Secure Sockets Layer »).

Remarquez qu'il est possible de vérifier les erreurs possibles de la configuration en utilisant la commande suivante :

```
~]# apachectl configtest
Syntax OK
```

Pour obtenir davantage d'informations sur la mise à niveau de la configuration d'Apache HTTP Server depuis la version 2.2 à la version 2.4, veuillez consulter http://httpd.apache.org/docs/2.4/upgrading.html.

12.1.3. Exécuter le service httpd

Cette section décrit comment lancer, arrêter, redémarrer, et vérifier le statut actuel du serveur Apache HTTP Server. Pour être en mesure d'utiliser le service **httpd**, assurez-vous que httpd soit effectivement installé. Cela peut être effectué en utilisant la commande suivante :

```
~]# yum install httpd
```

Pour obtenir davantage d'informations sur le concept des cibles et sur la manière de gérer les services système dans Red Hat Enterprise Linux, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

12.1.3.1. Lancer le service

Pour exécuter le service **httpd**, veuillez saisir ce qui suit à l'invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl start httpd.service
```

Si vous souhaitez que le service soit automatiquement lancé pendant le démarrage, veuillez utiliser la commande suivante :

```
~]# systemctl enable httpd.service
Created symlink from /etc/systemd/system/multi-
user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
```

NOTE

Si Apache HTTP Server est exécuté en tant que serveur sécurisé, un mot de passe pourrait être requis après le démarrage de la machine si une clé SSL privée chiffrée est utilisée.

12.1.3.2. Arrêter le service

Pour arrêter le service en cours d'exécution **httpd**, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl stop httpd.service
```

Pour empêcher le service d'être lancé automatiquement pendant le démarrage, veuillez saisir :

```
~]# systemctl disable httpd.service
Removed symlink /etc/systemd/system/multi-user.target.wants/httpd.service.
```

12.1.3.3. Redémarrer le service

Il existe trois différentes manières de redémarrer un service **httpd** :

1. Pour redémarrer le système entièrement, exécutez la commande suivante en tant qu'utilisateur **root** :

   ```
   ~]# systemctl restart httpd.service
   ```

Ceci arrête le service en cours d'exécution **httpd** et le lance immédiatement après. Veuillez utiliser cette commande après avoir installé ou supprimé un module chargé dynamiquement, comme le module PHP.

2. Pour uniquement recharger la configuration, veuillez saisir en tant qu'utilisateur **root** :

```
~]# systemctl reload httpd.service
```

Ceci cause au service en cours d'exécution **httpd** de recharger son fichier de configuration. Toute requête actuellement en cours de traitement sera interrompue, ce qui pourrait causer à un navigateur client d'afficher un message d'erreur ou d'effectuer un rendu partiel de page.

3. Pour recharger sa configuration sans affecter de requête active, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# apachectl graceful
```

Ceci cause au service en cours d'exécution **httpd** de recharger son fichier de configuration. Toute requête actuellement en cours de traitement continuera d'utiliser l'ancienne configuration.

Pour obtenir davantage d'informations sur la manière de gérer les services système sur Red Hat Enterprise Linux 7, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

12.1.3.4. Vérifier le statut du service

Pour vérifier que le service **httpd**est effectivement en cours d'exécution, veuillez saisir ce qui suit dans l'invite du shell :

```
~]# systemctl is-active httpd.service
active
```

12.1.4. Modifier les fichiers de configuration

Lorsque le service **httpd** est lancé, par défaut, il lit la configuration à partir d'emplacements répertoriés dans la Tableau 12.1, « Fichiers de configuration du service httpd ».

Tableau 12.1. Fichiers de configuration du service httpd

Chemin	Description
/etc/httpd/conf/httpd.conf	Fichier de configuration principal.
/etc/httpd/conf.d/	Répertoire auxiliaire pour les fichiers de configuration inclus dans le fichier de configuration principal.

Même si la configuration par défaut est convenable dans la plupart des situations, il est préférable de se familiariser avec certaines des options de configuration plus importantes. Remarquez que pour que tout changement puisse entrer en vigueur, le serveur web doit d'abord être redémarré. Veuillez consulter la Section 12.1.3.3, « Redémarrer le service » pour obtenir davantage d'informations sur la manière de redémarrer le service **httpd**.

Pour vérifier que la configuration ne contienne pas d'erreurs possibles, veuillez saisir ce qui suit dans une invite de shell :

```
~]# apachectl configtest
Syntax OK
```

Pour faciliter la récupération après des erreurs, il est recommandé d'effectuer une copie du fichier d'origine avant de le modifier.

12.1.5. Utiliser des modules

Étant une application modulaire, le service **httpd** est distribué avec un certain nombre d'objets partagés dynamiques (« *Dynamic Shared Objects* », ou DSO), qui peuvent être chargés ou déchargés dynamiquement pendant le runtime, selon les besoins. Dans Red Hat Enterprise Linux 7, ces modules se trouvent dans **/usr/lib64/httpd/modules/**.

12.1.5.1. Charger un module

Pour charger un module DSO particulier, utilisez la directive **LoadModule**. Remarquez que les modules fournis par un paquet séparé possèdent souvent leur propre fichier de configuration dans le répertoire **/etc/httpd/conf.d/**.

Exemple 12.1. Charger mod_ssl DSO

```
LoadModule ssl_module modules/mod_ssl.so
```

Une fois que vous aurez terminé, redémarrez le serveur web pour recharger la configuration. Veuillez consulter la Section 12.1.3.3, « Redémarrer le service » pour obtenir davantage d'informations sur la manière de redémarrer le service **httpd**.

12.1.5.2. Écrire un module

Si vous comptez créer un nouveau module DSO, assurez-vous que le paquet httpd-devel soit installé. Pour faire cela, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install httpd-devel
```

Ce paquet contient les fichiers « include », les fichiers d'en-tête « header », et l'utilitaire **APache eXtenSion** (**apxs**) requis pour compiler un module.

Une fois écrit, le module peut être créé par la commande suivante :

```
~]# apxs -i -a -c module_name.c
```

Si le build a été créé avec succès, vous devriez pouvoir charger le module de la même manière que tout autre module distribué avec le serveur Apache HTTP Server.

12.1.6. Paramétrer des hôtes virtuels

L'hébergement virtuel intégré d'Apache HTTP Server permet au serveur de fournir différentes informations basées sur l'adresse IP, le nom d'hôte, ou le port requis.

Pour créer un hôte virtuel basé nom, copiez l'exemple de fichier de configuration **/usr/share/doc/httpd-*VERSION*/httpd-vhosts.conf** dans le répertoire **/etc/httpd/conf.d/**, et remplacez les valeurs d'espaces réservés **@@Port@@** et **@@ServerRoot@@**. Personnalisez les options selon vos besoins, comme affiché dans l'Exemple 12.2, « Exemple de configuration d'hôte virtuel ».

Exemple 12.2. Exemple de configuration d'hôte virtuel

```
<VirtualHost *:80>
    ServerAdmin webmaster@penguin.example.com
    DocumentRoot "/www/docs/penguin.example.com"
    ServerName penguin.example.com
    ServerAlias www.penguin.example.com
    ErrorLog "/var/log/httpd/dummy-host.example.com-error_log"
    CustomLog "/var/log/httpd/dummy-host.example.com-access_log" common
</VirtualHost>
```

Remarquez que **ServerName** doit être un nom DNS valide assigné à la machine. Le conteneur **<VirtualHost>** est hautement personnalisable, et accepte la plupart des directives disponibles dans la configuration du serveur principal. Les directives qui *ne sont pas* prises en charge dans ce conteneur incluent **User** et **Group**, remplacées par **SuexecUserGroup**.

NOTE

Si vous configurez un hôte virtuel pour qu'il écoute sur un port qui n'est pas un port par défaut, assurez-vous de mettre à jour la directive **Listen** dans la section des paramètres globaux du fichier **/etc/httpd/conf/httpd.conf** en conséquence.

Pour activer un nouvel hôte virtuel créé, le serveur web doit d'abord être redémarré. Veuillez consulter la Section 12.1.3.3, « Redémarrer le service » pour obtenir davantage d'informations sur la manière de redémarrer le service **httpd**.

12.1.7. Paramétrer un serveur SSL

Secure Sockets Layer (SSL) est un protocole de chiffrement permettant à un serveur et un client de communiquer de manière sécurisée. Avec sa version étendue et améliorée nommée *Transport Layer Security* (TLS), confidentialité et intégrité des données sont assurées. Le serveur Apache HTTP Server combiné à **mod_ssl**, un module qui utilise le kit de ressources OpenSSL pour fournir la prise en charge SSL/TLS, est couramment appelé un *serveur SSL*. Red Hat Enterprise Linux prend également en charge l'utilisation de Mozilla NSS comme implémentation TLS. La prise en charge de Mozilla NSS est fournie par le module **mod_nss**.

Contrairement à une connexion HTTP qui pourrait être lue et probablement modifiée par toute personne capable de l'intercepter, l'utilisation de SSL/TLS sur HTTP, également appelée HTTPS, empêche toute inspection ou modification du contenu transmis. Cette section fournit des informations de base sur la manière d'activer ce module dans la configuration du serveur Apache HTTP Server, et vous guide à travers le processus de génération de clés privées et de certificats auto-signés.

12.1.7.1. Vue d'ensemble des certificats et de la sécurité

Les communications sécurisées sont basées sur l'usage de clés. Avec le *chiffrement symétrique* ou conventionel, les deux côtés de la transaction ont la même clé et peuvent l'utiliser pour décoder les

transmissions de l'un ou de l'autre. D'autre part, avec le *chiffrement asymétrique* ou public, deux clés coexistent : une *clé privée* qui est gardée secrète, et une *clé publique* qui est habituellement partagée publiquement. Même si les données sont chiffrées avec la clé publique, elles peuvent uniquement être déchiffrées avec la clé privée, et les données chiffrées avec la clé privée peuvent uniquement être déchiffrées avec la clé publique.

Pour fournir des communications sécurisées en utilisant SSL, un serveur SSL doit utiliser un certificat digital signé par une *Autorité de certification* (AC). Le certificat répertorie les divers attributs du serveur (c'est-à-dire, le nom d'hôte du serveur, le nom de l'entreprise, son emplacement, etc.), et la signature est produite en utilisant la clé privée de l'autorité de certification. Cette signature assure qu'une autorité de certification particulière a signé le certificat, et que le certificat n'a été modifié d'aucune manière.

Lorsqu'un navigateur web établit une nouvelle connexion SSL, il vérifie le certificat fourni par le serveur web. Si le certificat ne possède pas de signature d'une autorité de certification (AC) de confiance, ou si le nom d'hôte répertorié dans le certificat ne correspond pas au nom d'hôte utilisé pour établir la connexion, il refusera de communiquer avec le serveur et présentera normalement un message d'erreur approprié à l'utilisateur.

Par défaut, la plupart des navigateurs web sont configurés pour faire confiance à un ensemble d'autorités de certification (AC) couramment utilisées. Par conséquent, une AC appropriée devrait être choisie lors du paramétrage d'un serveur sécurisé, afin que les utilisateurs cibles puissent fait confiance à la connexion, sinon ils recevront un message d'erreur et devront accepter le certificat manuellement. Comme encourager des utilisateurs à outrepasser des erreurs de certification peut permettre à une personne malveillante d'intercepter la connexion, une autorité de certification de confiance devrait être utilisée dans la mesure du possible. Pour obtenir davantage d'informations sur cela, veuillez consulter la Tableau 12.2, « Informations sur les listes d'AC utilisées par des navigateurs web communs ».

Tableau 12.2. Informations sur les listes d'AC utilisées par des navigateurs web communs

Navigateur Web	Lien
Mozilla Firefox	Liste d'AC root Mozilla.
Opera	Informations sur les certificats root utilisés par Opera .
Internet Explorer	Informations sur les certificats root utilisés par Microsoft Windows .
Chromium	Informations sur les certificats root utilisés par le projet Chromium .

Lors du paramétrage d'un serveur SSL, vous devrez générer une requête de certificat et une clé privée, puis envoyez la requête de certificat, la preuve de l'identité de l'entreprise, et le paiement à l'autorité de certification. Une fois que l'AC aura vérifié la requête du certificat et votre identité, elle vous enverra un certificat signé que vous pourrez utiliser avec votre serveur. Alternativement, il est possible de créer un certificat auto-signé qui ne contient pas de signature d'AC, et qui devrait être uniquement utilisé pour effectuer des tests.

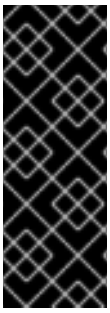
12.1.8. Activer le module mod_ssl

Si vous souhaitez paramétrer un serveur SSL ou HTTPS en utilisant **`mod_ssl`**, il **n'est pas** possible qu'une autre application ou qu'un autre module, comme **`mod_nss`**, soit configuré pour utiliser le même port. Le port **443** est le port par défaut pour HTTPS.

Pour paramétrer un serveur SSL en utilisant le module **mod_ssl** et le kit de ressources OpenSSL toolkit, veuillez installer les paquets mod_ssl et openssl. Saisissez la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install mod_ssl openssl
```

Cela créera le fichier de configuration **mod_ssl** sur **/etc/httpd/conf.d/ssl.conf**, qui est inclus dans le fichier de configuration principal du serveur Apache HTTP Server par défaut. Pour que le module soit chargé, veuillez redémarrer le service **httpd** comme décrit dans la Section 12.1.3.3, « Redémarrer le service ».



IMPORTANT

À cause de la vulnérabilité décrite sur *POODLE: SSLv3 vulnerability (CVE-2014-3566)*, Red Hat recommande de désactiver **SSL** et d'utiliser **TLSv1.1** ou **TLSv1.2** uniquement. Une rétrocompatibilité peut être effectuée en utilisant **TLSv1.0**. De nombreux produits pris en charge par Red Hat ont la capacité d'utiliser le protocole **SSLv2** ou **SSLv3**, ou de les activer par défaut. Cependant, l'utilisation de **SSLv2** ou de **SSLv3** est désormais fortement déconseillée.

12.1.8.1. Activer et désactiver SSL et TLS sur mod_ssl

Pour activer et désactiver des versions particulières des protocoles SSL et TLS, veuillez le faire globalement en ajoutant la directive **SSLProtocol** dans la section « ## SSL Global Context » du fichier de configuration et en la supprimant partout ailleurs, ou modifiez l'entrée par défaut sous « # SSL Protocol support » dans toutes les sections « VirtualHost ». Si vous ne le spécifiez pas dans la section VirtualHost par domaine, les paramètres hérités proviendront de la section globale. Pour vous assurer qu'une version du protocole est en cours de désactivation, l'administrateur doit **uniquement** spécifier **SSLProtocol** dans la section « SSL Global Context », ou bien, le spécifier dans **toutes** les sections VirtualHost par domaine.

Procédure 12.1. Désactiver SSLv2 et SSLv3

Pour désactiver SSL version 2 et SSL version 3, ce qui implique de tout activer sauf SSL version 2 et SSL version 3 dans toutes les sections VirtualHost, veuillez procéder comme suit :

1. En tant qu'utilisateur **root**, ouvrez le fichier **/etc/httpd/conf.d/ssl.conf** et recherchez **toutes** les instances de la directive **SSLProtocol**. Par défaut, le fichier de configuration contient une section qui ressemble à cela :

   ```
   ~]# vi /etc/httpd/conf.d/ssl.conf
   #   SSL Protocol support:
   # List the enable protocol levels with which clients will be able to
   # connect.  Disable SSLv2 access by default:
   SSLProtocol all -SSLv2
   ```

   Cette section se trouve dans la section VirtualHost.

2. Modifiez la ligne **SSLProtocol** comme suit :

   ```
   #   SSL Protocol support:
   # List the enable protocol levels with which clients will be able to
   # connect.  Disable SSLv2 access by default:
   SSLProtocol all -SSLv2 -SSLv3
   ```

Répétez cette action pour toutes les sections VirtualHost. Enregistrez et fermez le fichier.

3. Vérifiez que toutes les occurrences de la directive **SSLProtocol** ont été modifiées comme suit :

```
~]# grep SSLProtocol /etc/httpd/conf.d/ssl.conf
SSLProtocol all -SSLv2 -SSLv3
```

Cette étape est particulièrement importante si vous avez plus d'une section VirtualHost par défaut.

4. Redémarrez le démon Apache comme suit :

```
~]# systemctl restart httpd
```

Remarquez que toutes les sessions seront interrompues.

Procédure 12.2. Désactiver tous les protocoles SSL et TLS sauf TLS 1 et ses versions supérieures

Pour désactiver toutes les versions des protocoles SSL et TLS sauf TLS version 1 et ses versions supérieures, veuillez procéder comme suit :

1. En tant qu'utilisateur **root**, ouvrez le fichier **/etc/httpd/conf.d/ssl.conf** et recherchez **toutes** les instances de la directive **SSLProtocol**. Par défaut, le fichier contient une section qui ressemble à cela :

```
~]# vi /etc/httpd/conf.d/ssl.conf
#   SSL Protocol support:
# List the enable protocol levels with which clients will be able to
# connect.  Disable SSLv2 access by default:
SSLProtocol all -SSLv2
```

2. Modifiez la ligne **SSLProtocol** comme suit :

```
#   SSL Protocol support:
# List the enable protocol levels with which clients will be able to
# connect.  Disable SSLv2 access by default:
SSLProtocol -all +TLSv1 +TLSv1.1 +TLSv1.2
```

Enregistrer et fermer le fichier.

3. Vérifiez le changement comme suit :

```
~]# grep SSLProtocol /etc/httpd/conf.d/ssl.conf
SSLProtocol -all +TLSv1 +TLSv1.1 +TLSv1.2
```

4. Redémarrez le démon Apache comme suit :

```
~]# systemctl restart httpd
```

Remarquez que toutes les sessions seront interrompues.

Procédure 12.3. Tester le statut des protocoles SSL et TLS

Pour vérifier quelles versions de SSL et TLS sont activées ou désactivées, assurez-vous d'utiliser la commande **openssl s_client -connect**. La commande se trouve sous le format suivant :

```
openssl s_client -connect hostname:port -protocol
```

, où *port* correspond au port pour effectuer le test et *protocol* est la version du protocole à tester. Pour tester le serveur SSL exécuté localement, veuillez utiliser **localhost** comme nom d'hôte. Par exemple, pour tester le port par défaut pour des connexion HTTPS sécurisées, pour voir si SSLv3 est activé sur le port **443**, exécutez la commande suivante :

1. ```
   ~]# openssl s_client -connect localhost:443 -ssl3
   CONNECTED(00000003)
   139809943877536:error:14094410:SSL routines:SSL3_READ_BYTES:sslv3
   alert handshake failure:s3_pkt.c:1257:SSL alert number 40
   139809943877536:error:1409E0E5:SSL routines:SSL3_WRITE_BYTES:ssl
   handshake failure:s3_pkt.c:596:sortie omise
   New, (NONE), Cipher is (NONE)
   Secure Renegotiation IS NOT supported
   Compression: NONE
   Expansion: NONE
   SSL-Session:
       Protocol  : SSLv3
   sortie tronquée
   ```

   La sortie ci-dessus indique que la tentative de connexion a échoué et qu'aucun chiffrement n'a été négocié.

2. ```
   ~]$ openssl s_client -connect localhost:443 -tls1_2
   CONNECTED(00000003)
   depth=0 C = --, ST = SomeState, L = SomeCity, O = SomeOrganization,
   OU = SomeOrganizationalUnit, CN = localhost.localdomain,
   emailAddress = root@localhost.localdomainsortie omise
   New, TLSv1/SSLv3, Cipher is ECDHE-RSA-AES256-GCM-SHA384
   Server public key is 2048 bit
   Secure Renegotiation IS supported
   Compression: NONE
   Expansion: NONE
   SSL-Session:
       Protocol  : TLSv1.2sortie tronquée
   ```

   La sortie ci-dessus indique qu'aucun échec de tentative de connexion n'a eu lieu et qu'un ensemble de chiffrements a été négocié.

Les options de la commande **openssl s_client** sont documentées dans la page man de **s_client(1)**.

Pour obtenir davantage d'informations sur la vulnérabilité SSLv3 et comment effectuer des tests pour la trouver, veuillez consulter l'article de la base de connaissances Red Hat *POODLE: SSLv3 vulnerability (CVE-2014-3566)*.

12.1.9. Activer le module mod_nss

Si vous comptez paramétrer un serveur HTTPS en utilisant **mod_nss**, vous **ne pourrez pas** avoir le paquet mod_ssl installé avec ses paramètres par défaut car **mod_ssl** utilisera le port **443** par défaut, cependant ceci est le port HTTPS par défaut. Si possible, veuillez supprimer le paquet.

Pour supprimer mod_ssl, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# yum remove mod_ssl
```

NOTE

Si **mod_ssl** est requis à d'autres fins, veuillez modifier le fichier **/etc/httpd/conf.d/ssl.conf** afin qu'il utilise un autre port que le port **443** pour empêcher que **mod_ssl** entre en conflit avec **mod_nss** lorsque son port d'écoute est changé sur le port **443**.

Un seul module peut posséder un port, ainsi **mod_nss** et **mod_ssl** peuvent uniquement coexister au même moment s'ils utilisent des ports uniques. Pour cette raison, **mod_nss** utilise par défaut le port **8443**, mais le port HTTPS par défaut est le port **443**. Le port est spécifié par la directive **Listen** ainsi que dans le nom ou l'adresse VirtualHost.

Tout dans NSS est associé à un jeton (ou « token »). Un jeton logiciel existe dans la base de données NSS mais vous pouvez également avoir une clé physique contenant des certificats. Avec OpenSSL, des certificats discrets et des clés privées se trouvent dans les fichiers PEM. Avec NSS, ces fichiers sont stockés dans une base de données. Chaque certificat et chaque clé est associé à un jeton, et chaque jeton peut avoir un mot de passe le protégeant. Ce mot de passe est optionnel, mais si un mot de passe est utilisé, alors le serveur HTTP Apache aura besoin d'en avoir une copie afin de pouvoir ouvrir la base de données sans intervention de l'utilisateur pendant le démarrage système.

Procédure 12.4. Configurer mod_nss

1. Installez mod_nss en tant qu'utilisateur **root** :

   ```
   ~]# yum install mod_nss
   ```

   Cela créera le fichier de configuration **mod_nss** dans **/etc/httpd/conf.d/nss.conf**. Le répertoire **/etc/httpd/conf.d/** est inclus dans le fichier de configuration principal du serveur Apache HTTP Server par défaut. Pour que le module soit chargé, veuillez redémarrer le service **httpd** comme décrit dans la Section 12.1.3.3, « Redémarrer le service ».

2. En tant qu'utilisateur **root**, ouvrez le fichier **/etc/httpd/conf.d/nss.conf** et recherchez **toutes** les instances de la directive **Listen**.

   Modifiez la ligne **Listen 8443** comme suit :

   ```
   Listen 443
   ```

   Le port **443** est le port par défaut pour **HTTPS**.

3. Modifiez la ligne **VirtualHost _default_:8443** par défaut comme suit :

   ```
   VirtualHost _default_:443
   ```

Veuillez modifier toute autre section d'hôte virtuel qui n'est pas par défaut s'il en existe. Puis enregistrez et fermez le fichier.

4. Mozilla NSS stocke les certificats dans une *base de données de certificats de serveur* indiquée par la directive **NSSCertificateDatabase** dans le fichier **/etc/httpd/conf.d/nss.conf**. Par défaut, le chemin est défini sur **/etc/httpd/alias**, la base de données NSS créée pendant l'installation.

   Pour afficher la base de données NSS par défaut, veuillez exécuter la commande suivante :

   ```
   ~]# certutil -L -d /etc/httpd/alias

   Certificate Nickname                                         Trust
   Attributes

   SSL,S/MIME,JAR/XPI

   cacert
   CTu,Cu,Cu
   Server-Cert
   u,u,u
   alpha
   u,pu,u
   ```

   Dans la sortie de commande ci-dessus, **Server-Cert** est le **NSSNickname** par défaut. L'option **-L** répertorie tous les certificats, ou affiche des informations sur un certificat nommé, dans une base de données de certificats. L'option **-d** spécifie le répertoire de la base de données contenant le certificat et les fichiers-clés de la base de données. Veuillez consulter la page man **certutil(1)** pour davantage d'options de ligne de commande.

5. Pour configurer mod_nss de manière à utiliser une autre base de données, veuillez modifier la ligne **NSSCertificateDatabase** dans le fichier **/etc/httpd/conf.d/nss.conf**. Le fichier par défaut possède les lignes suivantes dans la section VirtualHost.

   ```
   #   Server Certificate Database:
   #   The NSS security database directory that holds the certificates
   and
   #   keys. The database consists of 3 files: cert8.db, key3.db and
   secmod.db.
   #   Provide the directory that these files exist.
   NSSCertificateDatabase /etc/httpd/alias
   ```

   Dans la sortie la commande ci-dessus, **alias** est le répertoire de la base de données NSS par défaut, **/etc/httpd/alias/**.

6. Pour appliquer un mot de passe à la base de données des certificats NSS par défaut, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

   ```
   ~]# certutil -W -d /etc/httpd/alias
   Enter Password or Pin for "NSS Certificate DB":
   Enter a password which will be used to encrypt your keys.
   The password should be at least 8 characters long,
   and should contain at least one non-alphabetic character.
   ```

```
Enter new password:
Re-enter password:
Password changed successfully.
```

7. Avant de déployer le serveur HTTPS, veuillez créer une nouvelle base de données de certificats en utilisant un certificat signé par une autorité de certification (AC).

Exemple 12.3. Ajouter un certificat à la base de données Mozilla NSS

La commande **certutil** est utilisée pour ajouter un certificat AC aux fichiers de la base de données NSS :

```
certutil -d /etc/httpd/nss-db-directory/ -A -n "CA_certificate" -
t CT,, -a -i certificate.pem
```

La commande ci-dessus ajoute un certificat AC stocké dans un fichier au format PEM nommé *certificate.pem*. L'option **-d** spécifie le répertoire de la base de données NSS contenant le certificat et les fichiers-clés de la base de données, l'option **-n** définit un nom pour le certificat, **-t CT,** signifie que le certificat est approuvé pour être utilisé dans les serveurs et clients TLS. L'option **-A** ajoute un certificat existant dans une base de données de certificats. Si la base de données n'existe pas, elle sera créée. L'option **-a** permet l'utilisation du format ASCII pour les entrées ou les sorties et l'option **-i** transmet le fichier d'entrée **certificate.pem** à la commande.

Veuillez consulter la page man de **certutil(1)** pour obtenir davantage d'options de ligne de commande.

8. La base de données NSS doit être protégée par un mot de passe afin de garantir la sécurité de la clé privée.

Exemple 12.4. Définir un mot de passe pour la base de données Mozilla NSS

L'outil **certutil** peut être utilisé pour définir un mot de passe pour une base de données NSS comme suit :

```
certutil -W -d /etc/httpd/nss-db-directory/
```

Par exemple, pour la base de données par défaut, veuillez exécuter une commande en tant qu'utilisateur **root**, comme suit :

```
~]# certutil -W -d /etc/httpd/alias
Enter Password or Pin for "NSS Certificate DB":
Enter a password which will be used to encrypt your keys.
The password should be at least 8 characters long,
and should contain at least one non-alphabetic character.

Enter new password:
Re-enter password:
Password changed successfully.
```

9. Configurez **mod_nss** de manière à utiliser le jeton logiciel NSS interne en modifiant la ligne avec la directive **NSSPassPhraseDialog**, comme suit :

```
~]# vi /etc/httpd/conf.d/nss.conf
NSSPassPhraseDialog file:/etc/httpd/password.conf
```

Ceci sert à éviter la saisie manuelle de mots de passe pendant le démarrage système. Le jeton logiciel existe dans la base de données NSS, mais vous pouvez également posséder une clé physique contenant les certificats.

10. Si le certificat du serveur SSL situé dans la base de données NSS est un certificat RSA, assurez-vous que le paramètre **NSSNickname** ne soit pas mis en commentaire et qu'il corresponde bien au nom affiché dans l'étape 4 ci-dessus :

```
~]# vi /etc/httpd/conf.d/nss.conf
NSSNickname Server-Cert
```

Si le certificat du serveur SSL situé dans la base de données NSS est un certificat ECC, assurez-vous que le paramètre **NSSECCNickname** ne soit pas mis en commentaire et qu'il corresponde bien au surnom affiché dans l'étape 4 ci-dessus :

```
~]# vi /etc/httpd/conf.d/nss.conf
NSSECCNickname Server-Cert
```

Assurez-vous que le paramètre **NSSCertificateDatabase** ne soit pas mis en commentaire et qu'il pointe vers le répertoire de la base de données NSS affiché dans l'étape 4 ou configuré dans l'étape 5 ci-dessus :

```
~]# vi /etc/httpd/conf.d/nss.conf
NSSCertificateDatabase /etc/httpd/alias
```

Remplacez **/etc/httpd/alias** par le chemin vers la base de données de certificats à utiliser.

11. Créez le fichier **/etc/httpd/password.conf** en tant qu'utilisateur **root** :

```
~]# vi /etc/httpd/password.conf
```

Ajoutez une ligne sous le format suivant :

```
internal:password
```

en remplaçant *password* par le mot de passe appliqué aux bases de données de sécurité NSS dans l'étape 6 ci-dessus.

12. Veuillez appliquer l'appartenance et les permissions appopriées au fichier **/etc/httpd/password.conf** :

```
~]# chgrp apache /etc/httpd/password.conf
~]# chmod 640 /etc/httpd/password.conf
~]# ls -l /etc/httpd/password.conf
-rw-r-----. 1 root apache 10 Dec  4 17:13 /etc/httpd/password.conf
```

13. Pour configurer **mod_nss** de manière à utiliser le jeton logiciel NSS dans **/etc/httpd/password.conf**, veuillez modifier **/etc/httpd/conf.d/nss.conf** comme suit :

```
~]# vi /etc/httpd/conf.d/nss.conf
```

14. Redémarrez le serveur Apache pour que les changements entrent en vigueur, comme décrit dans la Section 12.1.3.3, « Redémarrer le service »

IMPORTANT

À cause de la vulnérabilité décrite sur *POODLE: SSLv3 vulnerability (CVE-2014-3566)*, Red Hat recommande de désactiver **SSL** et d'utiliser **TLSv1.1** ou **TLSv1.2** uniquement. Une rétrocompatibilité peut être effectuée en utilisant **TLSv1.0**. De nombreux produits pris en charge par Red Hat ont la capacité d'utiliser le protocole **SSLv2** ou **SSLv3**, ou de les activer par défaut. Cependant, l'utilisation de **SSLv2** ou de **SSLv3** est désormais fortement déconseillée.

12.1.9.1. Activer et désactiver SSL et TLS sur mod_nss

Pour activer et désactiver des versions particulières des protocoles SSL et TLS, veuillez le faire globalement en ajoutant la directive **NSSProtocol** dans la section « ## SSL Global Context » du fichier de configuration et en la supprimant partout ailleurs, ou modifiez l'entrée par défaut sous « # SSL Protocol » dans toutes les sections « VirtualHost ». Si vous ne le spécifiez pas dans la section VirtualHost par domaine, les paramètres hérités proviendront de la section globale. Pour vous assurer qu'une version du protocole est en cours de désactivation, l'administrateur doit **uniquement** spécifier **NSSProtocol** dans la section « SSL Global Context », ou bien dans **toutes** les sections VirtualHost par domaine.

Procédure 12.5. Désactiver tous les protocoles SSL et TLS sauf TLS 1 et ses versions supérieures dans mod_nss

Pour désactiver toutes les versions des protocoles SSL et TLS sauf TLS version 1 et ses versions supérieures, veuillez procéder comme suit :

1. En tant qu'utilisateur **root**, ouvrez le fichier **/etc/httpd/conf.d/nss.conf** et recherchez **toutes** les instances de la directive **NSSProtocol**. Par défaut, le fichier de configuration contient une section qui ressemble à cela :

```
~]# vi /etc/httpd/conf.d/nss.conf
#   SSL Protocol:sortie omise
#   Since all protocol ranges are completely inclusive, and no
protocol in the
#   middle of a range may be excluded, the entry "NSSProtocol
SSLv3,TLSv1.1"
#   is identical to the entry "NSSProtocol SSLv3,TLSv1.0,TLSv1.1".
NSSProtocol SSLv3,TLSv1.0,TLSv1.1
```

   Cette section se trouve dans la section VirtualHost.

2. Modifiez la ligne **NSSProtocol** comme suit :

```
#   SSL Protocol:
NSSProtocol TLSv1.0,TLSv1.1
```

Répétez cette action pour toutes les sections VirtualHost.

3. Modifiez la ligne **`Listen 8443`** comme suit :

```
Listen 443
```

4. Modifiez la ligne **`VirtualHost _default_:8443`** par défaut comme suit :

```
VirtualHost _default_:443
```

Veuillez modifier toute autre section d'hôte virtuel qui n'est pas par défaut s'il en existe. Puis enregistrez et fermez le fichier.

5. Vérifiez que toutes les occurrences de la directive **`NSSProtocol`** ont été modifiées comme suit :

```
~]# grep NSSProtocol /etc/httpd/conf.d/nss.conf
#   middle of a range may be excluded, the entry "NSSProtocol
SSLv3,TLSv1.1"
#   is identical to the entry "NSSProtocol SSLv3,TLSv1.0,TLSv1.1".
NSSProtocol TLSv1.0,TLSv1.1
```

Cette étape est particulièrement importante si vous possédez plus d'une section VirtualHost.

6. Redémarrez le démon Apache comme suit :

```
~]# service httpd restart
```

Remarquez que toutes les sessions seront interrompues.

Procédure 12.6. Tester le statut des protocoles SSL et TLS dans mod_nss

Pour vérifier quelles versions de SSL et TLS sont activées ou désactivées dans **mod_nss**, veuillez utiliser la commande **`openssl s_client -connect`**. Installez le paquet openssl en tant qu'utilisateur **`root`** :

```
~]# yum install openssl
```

La commande **`openssl s_client -connect`** se trouve sous le format suivant :

```
openssl s_client -connect hostname:port -protocol
```

, où *port* correspond au port pour effectuer le test et *protocol* est la version du protocole à tester. Pour tester le serveur SSL exécuté localement, veuillez utiliser **`localhost`** comme nom d'hôte. Par exemple, pour tester le port par défaut pour des connexions HTTPS sécurisées, pour voir si SSLv3 est activé sur le port **`443`**, exécutez la commande suivante :

1.
```
~]# openssl s_client -connect localhost:443 -ssl3
CONNECTED(00000003)
3077773036:error:1408F10B:SSL routines:SSL3_GET_RECORD:wrong version
number:s3_pkt.c:337:sortie omise
New, (NONE), Cipher is (NONE)
Secure Renegotiation IS NOT supported
```

```
Compression: NONE
Expansion: NONE
SSL-Session:
    Protocol  : SSLv3
sortie tronquée
```

La sortie ci-dessus indique que la tentative de connexion a échoué et qu'aucun chiffrement n'a été négocié.

2.
```
~]$ openssl s_client -connect localhost:443 -tls1
CONNECTED(00000003)
depth=1 C = US, O = example.com, CN = Certificate Shacksortie omise
New, TLSv1/SSLv3, Cipher is AES128-SHA
Server public key is 1024 bit
Secure Renegotiation IS supported
Compression: NONE
Expansion: NONE
SSL-Session:
    Protocol  : TLSv1sortie tronquée
```

La sortie ci-dessus indique qu'aucun échec de tentative de connexion n'a eu lieu et qu'un ensemble de chiffrements a été négocié.

Les options de la commande **`openssl s_client`** sont documentées dans la page man de **`s_client(1)`**.

Pour obtenir davantage d'informations sur la vulnérabilité SSLv3 et comment effectuer des tests pour la trouver, veuillez consulter l'article de la base de connaissances Red Hat *POODLE: SSLv3 vulnerability (CVE-2014-3566)*.

12.1.10. Utiliser une clé existante et un certificat

Si vous avez créé au préalable une clé et un certificat, vous pouvez configurer le serveur SSL pour utiliser ces fichiers au lieu d'en générer de nouveaux. Il existe uniquement deux situations dans lesquelles cela n'est pas possible :

1. *Vous modifiez l'adresse IP ou le nom du domaine.*

   Des certificats sont créés pour une paire adresse IP et nom de domaine particuliers. Si l'une de ces valeurs change, le certificat est invalide.

2. *Vous avez un certificat de VeriSign, et vous changez le logiciel du serveur.*

   VeriSign, une autorité de certification largement utilisée, octroit des certificats pour un produit logiciel, une adresse IP, et un nom de domaine particulier. Modifier le produit logiciel rend le certificat invalide.

Dans les deux cas ci-dessus, vous devrez obtenir un nouveau certificat. Pour obtenir davantage d'informations sur ce sujet, veuillez consulter la Section 12.1.11, « Générer une nouvelle clé et un nouveau certificat ».

Si vous souhaitez utiliser une clé et un certificat existants, déplacez les fichiers correspondants sur les répertoires **`/etc/pki/tls/private/`** et **`/etc/pki/tls/certs/`**, respectivement. Vous pouvez faire cela exécutant les commandes suivantes en tant qu'utilisateur **`root`** :

```
~]# mv key_file.key /etc/pki/tls/private/hostname.key
~]# mv certificate.crt /etc/pki/tls/certs/hostname.crt
```

Puis ajoutez les lignes suivante au fichier de configuration **/etc/httpd/conf.d/ssl.conf** :

```
SSLCertificateFile /etc/pki/tls/certs/hostname.crt
SSLCertificateKeyFile /etc/pki/tls/private/hostname.key
```

Pour charger la configuration mise à jour, redémarrez le service **httpd** comme décrit dans la Section 12.1.3.3, « Redémarrer le service ».

Exemple 12.5. Utiliser une clé et un certificat du serveur web sécurisé « Red Hat Secure Web Server »

```
~]# mv /etc/httpd/conf/httpsd.key
/etc/pki/tls/private/penguin.example.com.key
~]# mv /etc/httpd/conf/httpsd.crt
/etc/pki/tls/certs/penguin.example.com.crt
```

12.1.11. Générer une nouvelle clé et un nouveau certificat

Pour générer une nouvelle paire clé-certificat, le paquet crypto-utils doit être installé sur le système. Pour l'installer, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install crypto-utils
```

Ce paquet fournit un ensemble d'outils pour générer et gérer des certificats SSL et des clés privées, et inclut **genkey**, l'utilitaire de génération de paires de clés « Red Hat Keypair Generation », qui vous guidera à travers le processus de génération de clés.

IMPORTANT

Si le serveur possède déjà un certificat valide et que vous le remplacez par un nouveau certificat, veuillez spécifier un numéro de série différent. Ceci permet de vous assurer que les navigateurs clients soient notifiés de ce changement, mis à jour au sujet de ce nouveau certificat comme prévu, et qu'ils réussissent à accéder à la page. Pour créer un nouveau certificat avec un numéro de série personnalisé, en tant qu'utilisateur **root**, veuillez utiliser la commande suivante au lieu de **genkey** :

```
~]# openssl req -x509 -new -set_serial number -key hostname.key
-out hostname.crt
```

NOTE

S'il existe déjà un fichier clé pour un nom d'hôte particulier dans votre système, **genkey** refusera de démarrer. Dans ce cas, veuillez supprimer le fichier existant en utilisant la commande suivante en tant qu'utilisateur **root** :

```
~]# rm /etc/pki/tls/private/hostname.key
```

Pour exécuter l'utilitaire, veuillez saisir la commande **genkey** en tant qu'utilisateur **root**, suivie du nom d'hôte approprié (par exemple, **penguin.example.com**) :

```
~]# genkey hostname
```

Pour terminer la création de la clé et du certificat, veuillez procéder aux étapes suivantes :

1. Examinez les emplacements cibles dans lesquels la clé et le certificat seront stockés.

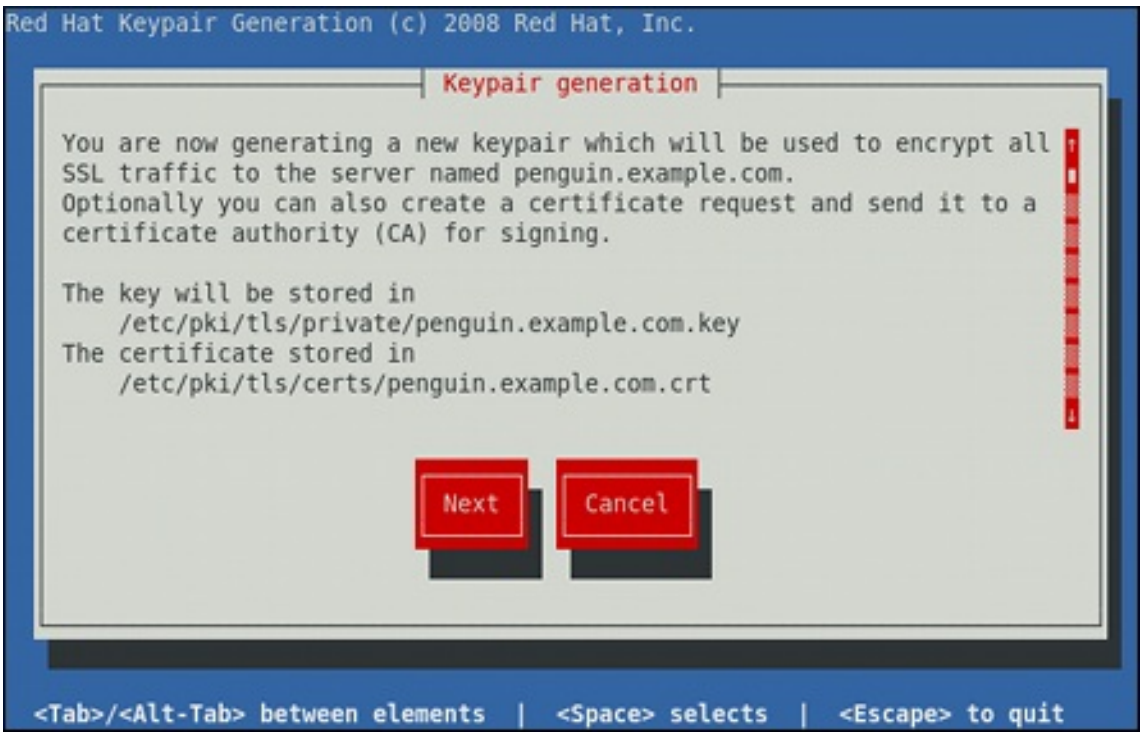


Figure 12.1. Exécuter l'utilitaire genkey

Veuillez utiliser la touche **Tab** pour sélectionner le bouton **Suivant**, et appuyez sur **Entrée** pour passer à l'écran suivant.

2. En utilisant les touches de flèches **haut** et **bas**, sélectionnez une taille de clé convenable. Remarquez que malgré qu'une clé de plus grande taille améliore la sécurité, celle-ci augmentera également le temps de réponse de votre serveur. L'organisme NIST recommande d'utiliser **2048 bits**. Veuillez consulter le document *NIST Special Publication 800-131A*.

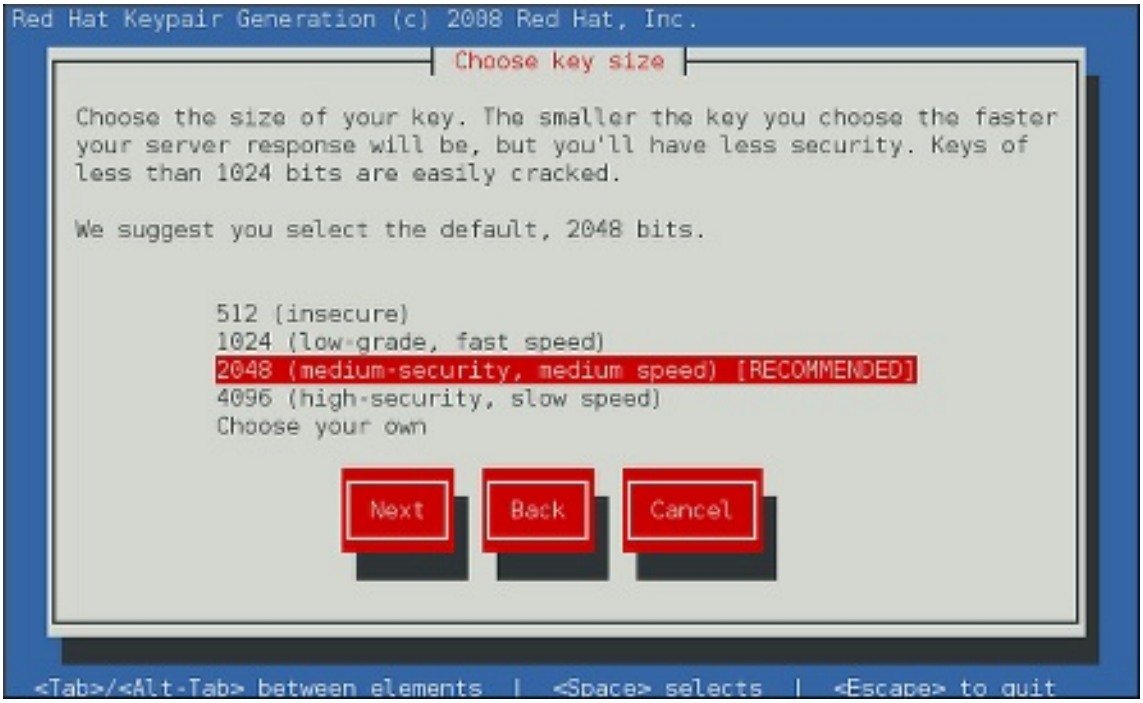


Figure 12.2. Sélectionner la taille de la clé

Une fois terminé, veuillez utiliser la touche **Tab** pour sélectionner le bouton **Suivant**, et appuyez sur **Entrée** pour initier le processus de génération de bits aléatoire. Selon la taille de clé sélectionnée, ceci peut prendre longtemps.

3. Décidez si vous souhaitez envoyer une requête de certificat à une autorité de certification.

Figure 12.3. Générer une requête de certificat

Utilisez la touche **Tab** pour sélectionner **Oui** afin de composer une requête de certificat, ou **Non** pour générer un certificat auto-signé. Puis appuyez sur **Entrée** pour confirmer votre choix.

4. À l'aide de la **barre d'espace**, activez (**[*]**) ou désactivez (**[ ]**) le chiffrement de la clé privée.

Figure 12.4. Chiffrer la clé privée

Veuillez utiliser la touche **Tab** pour sélectionner le bouton **Suivant**, et appuyez sur **Entrée** pour passer à l'écran suivant.

5. Si vous avez activé le chiffrement de clé privée, veuillez saisir une phrase de passe convenable. Remarquez que pour des raisons de sécurité, celle-ci n'est pas affichée lorsque vous la saisissez, et doit faire 5 caractères au minimum.

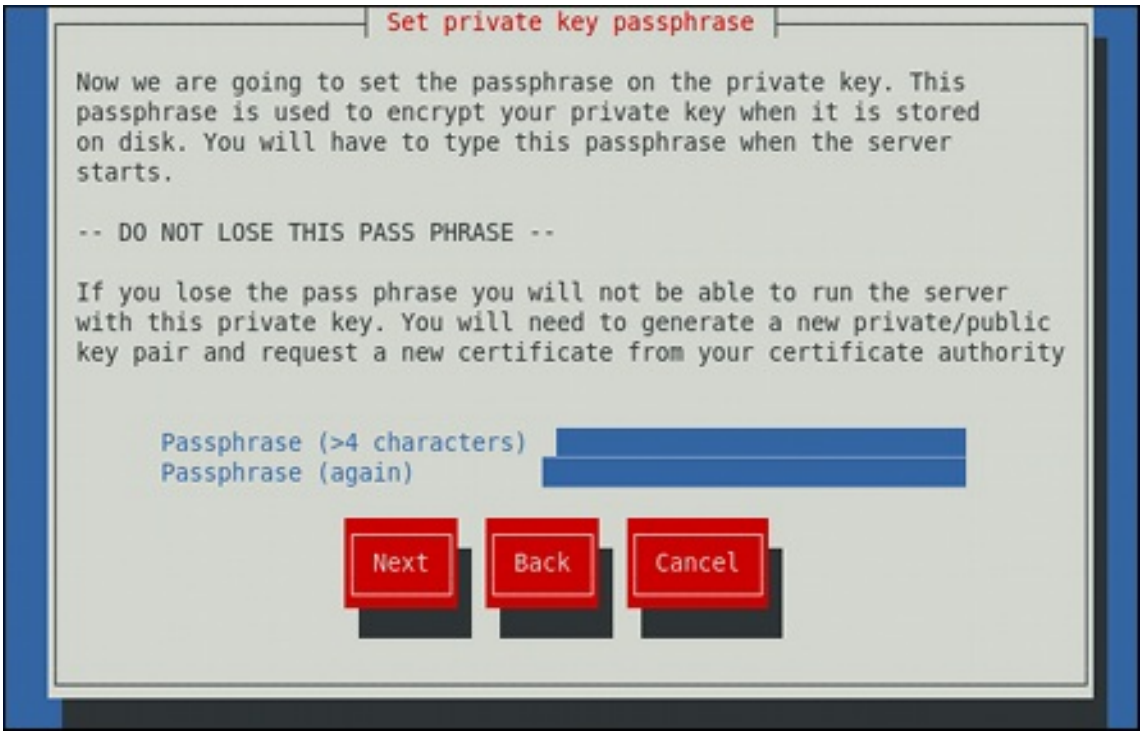


Figure 12.5. Saisir une phrase de passe

Veuillez utiliser la touche **Tab** pour sélectionner le bouton **Suivant**, et appuyez sur **Entrée** pour passer à l'écran suivant.

IMPORTANT

Saisir la phrase de passe correctement est requis pour que le serveur démarre. Si vous la perdez, vous devrez générer une nouvelle clé et un nouveau certificat.

6. Personnaliser les détails du certificat.

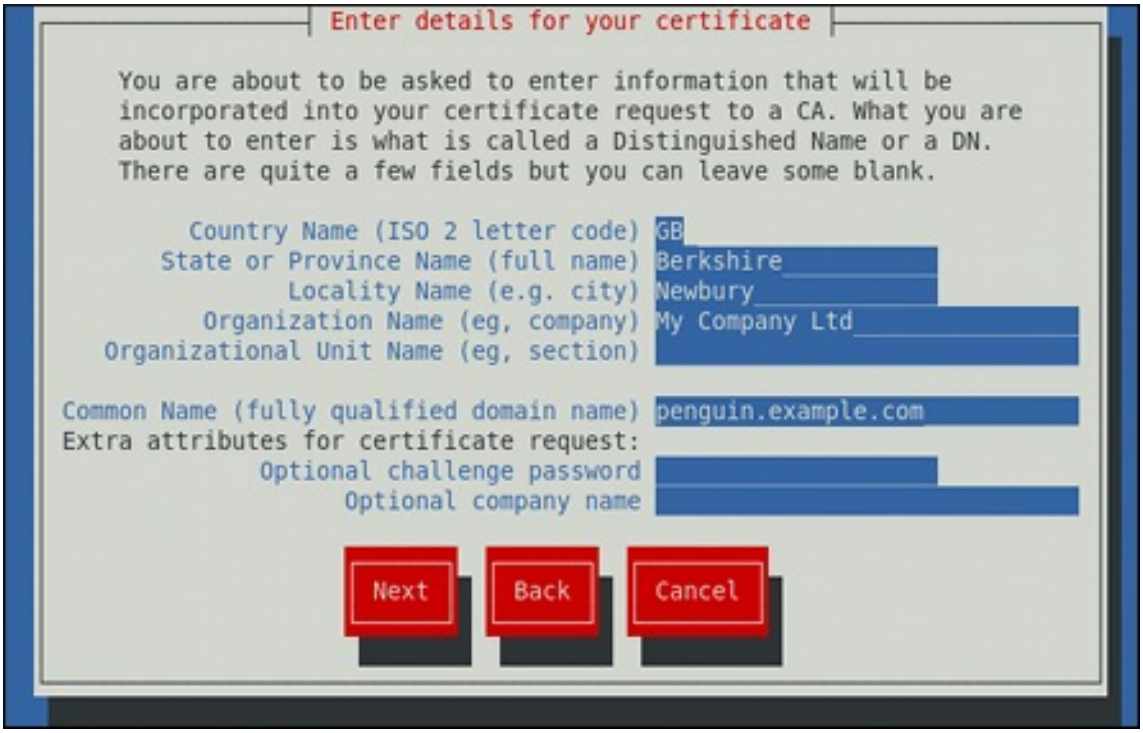


Figure 12.6. Spécifier les informations du certificat

Veuillez utiliser la touche **Tab** pour sélectionner le bouton **Suivant**, et appuyez sur **Entrée** pour terminer avec la génération de clé.

7. Si vous avez activé la génération de requête de certificat, il vous sera demandé de l'envoyer à une autorité de certification.

Figure 12.7. Instructions sur la manière d'envoyer une requête de certificat

Appuyez sur **Entrée** pour retourner dans une invite shell.

Une fois généré, ajoutez les emplacements de la clé et du certificat au fichier de configuration **/etc/httpd/conf.d/ssl.conf** :

```
SSLCertificateFile /etc/pki/tls/certs/hostname.crt
SSLCertificateKeyFile /etc/pki/tls/private/hostname.key
```

Finalement, redémarrez le service **httpd** comme décrit dans la Section 12.1.3.3, « Redémarrer le service » afin que la configuration mise à jour soit chargée.

12.1.12. Configurez le pare-feu pour HTTP et HTTPS en utilisant la ligne de commande

Par défaut, Red Hat Enterprise Linux n'autorise pas le trafic **HTTP** et **HTTPS**. Pour autoriser le système à agir en tant que serveur web, veuillez utiliser les services **firewalld** pris en charge pour que le trafic **HTTP** et que le trafic **HTTPS** soient autorisés à passer à travers le pare-feu comme requis.

Pour autoriser **HTTP** en utilisant la ligne de commande, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# firewall-cmd --add-service http
 success
```

Pour autoriser **HTTPS** en utilisant la ligne de commande, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# firewall-cmd --add-service https
 success
```

Remarquez que ces changements ne persisteront pas après le prochain démarrage système. Pour rendre ces changements permanents sur le pare-feu, veuillez répéter les commandes en ajoutant l'option **--permanent**.

12.1.12.1. Vérifier l'accès réseau de HTTPS et de HTTPS entrant en utilisant la ligne de commande

Pour vérifier quels services le pare-feu est configuré à autoriser, veuillez exécuter la commande suivante sur la ligne de commande en tant qu'utilisateur **root** :

```
~]# firewall-cmd --list-all
public (default, active)
  interfaces: em1
  sources:
  services: dhcpv6-client sshsortie tronquée
```

Dans cet exemple extrait d'une installation par défaut, le pare-feu est activé mais **HTTP** et **HTTPS** n'ont pas été autorisés à passer à travers.

Une fois les services de pare-feu **HTTP** et **HTTP** activés, la ligne **services** apparaîtra et sera similaire à ceci :

```
services: dhcpv6-client http https ssh
```

Pour obtenir des informations supplémentaires sur l'activation des services de pare-feu, ou sur l'ouverture et la fermeture de ports sur **firewalld**, veuillez consulter le Guide de sécurité Red Hat Enterprise Linux 7 .

12.1.13. Ressources supplémentaires

Pour en savoir plus sur le serveur Apache HTTP, veuillez consulter les ressources suivantes.

Documentation installée

- **httpd(8)** — la page du manuel du service **httpd** contenant la liste complète de ses options de ligne de commande.
- **genkey(1)** — la page du manuel de l'utilitaire **genkey**, fournie par le paquet crypto-utils.
- **apachectl(8)** — la page du manuel de l'interface de contrôle du serveur HTTP Apache.

Documentation installable

- http://localhost/manual/ — documentation officielle du serveur HTTP Apache avec la description complète de ses directives et modules disponibles. Remarquez que pour accéder à cette documentation, le paquet httpd-manual doit être installé, et le serveur web doit être en cours d'exécution.

  Avant d'accéder à la documentation, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

  ```
  ~]# yum install httpd-manual
  ~]# apachectl graceful
  ```

Documentation en ligne

- http://httpd.apache.org/ — site web officiel du serveur HTTP Apache avec la documentation sur toutes les directives et tous les modules par défaut.
- http://www.openssl.org/ — page d'accueil d'OpenSSL contenant davantage de documentation, une foire aux questions, des liens de listes de diffusion, ainsi que d'autres ressources utiles.

CHAPITRE 13. SERVEURS DE COURRIER

Red Hat Enterprise Linux offre de nombreuses applications avancées pour servir et accéder aux courriers électroniques. Ce chapitre décrit les protocoles de courriers électroniques modernes utilisés de nos jours, ainsi que certains des programmes conçus pour envoyer et recevoir des courriers électroniques.

13.1. PROTOCOLES DE COURRIER ÉLECTRONIQUE

De nos jours, les courriers électroniques sont distribués en utilisant une architecture client/serveur. Un courrier électronique est créé à l'aide d'un programme de client de messagerie. Le serveur transfère ensuite le message sur le serveur de courrier du destinataire, où le message est finalement remis au client de messagerie du destinataire

Pour activer ce processus, une variété de protocoles réseau standard autorisent différentes machines, exécutant souvent différents systèmes d'exploitation, et utilisant différents programmes de courrier électronique à envoyer et recevoir des courriers électroniques.

Les protocoles dont il est question sont les plus couramment utilisés lors des transferts de courriers électroniques.

13.1.1. Protocoles de transport de courrier

La distribution du courrier à partir d'une application cliente sur un serveur, et à partir d'un serveur d'origine sur un serveur destinataire, est gérée par *SMTP* (*Simple Mail Transfer Protocol*).

13.1.1.1. SMTP

Le but principal de SMTP est de transférer les courriers électroniques entre serveurs de courrier. Cependant, ceci est critique pour les clients de courrier également. Pour envoyer un courrier, le client envoie le message sur un serveur de courrier sortant, qui contactera par la suite le serveur de courrier destinataire pour le lui remettre, donc il faut spécifier un serveur SMTP lors de la configuration d'un client de messagerie.

Dans Red Hat Enterprise Linux, un utilisateur peut configurer un serveur SMTP sur la machine locale pour gérer la remise de courrier. Il est également possible de configurer des serveurs SMTP distants pour le courrier sortant.

Un point important à souligner sur le protocole SMTP est qu'il ne nécessite pas d'authentification. Ceci permet à n'importe qui sur Internet d'envoyer des courriers électroniques à une personne ou à de grands groupes de personnes. C'est cette caractéristique de SMTP qui rend les courriers électroniques indésirables, ou *spam* possibles. Imposer des restrictions de relais limite les utilisateurs aléatoires envoyant des courriers électroniques via votre serveur SMTP vers d'autres serveurs sur internet. Les serveurs qui n'imposent pas de telles restrictions sont appelés des serveurs *relais ouverts*.

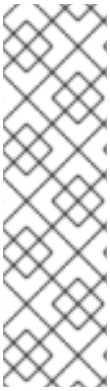
Red Hat Enterprise Linux 7 fournit les programmes Postfix et Sendmail SMTP.

13.1.2. Protocoles d'accès au courrier

Deux protocoles principaux sont utilisés par les applications clientes de courrier pour récupérer le courrier électronique en provenance des serveurs de courrier : *POP* (*Post Office Protocol*), et le protocole *IMAP* (*Internet Message Access Protocol*).

13.1.2.1. POP

Le serveur POP par défaut sous Red Hat Enterprise Linux se nomme **Dovecot** et est fourni par le paquet dovecot.



NOTE

Pour utiliser **Dovecot**, commencez par vous assurer que le paquet dovecot soit effectivement installé sur votre système en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install dovecot
```

Pour obtenir davantage d'informations sur l'installation de paquets avec Yum, veuillez consulter la Section 8.2.4, « Installation de paquets ».

Lors de l'utilisation d'un serveur **POP**, les messages électroniques sont téléchargés par les applications clientes de courrier. Par défaut, la plupart des clients de courrier **POP** sont automatiquement configurés pour supprimer le message sur le serveur de courrier une fois son tranfert réussi. Ce paramètre peut toutefois être modifié.

POP est totalement compatible avec les standards de messagerie Internet importants, tels que *MIME* (*Multipurpose Internet Mail Extensions*), qui permet d'ajouter des pièces jointes aux courriers électroniques.

POP fonctionne mieux pour les utilisateurs qui possèdent un système sur lequel lire les courriers électroniques. Celui-ci fonctionne également mieux pour les utilisateurs qui ne possèdent pas de connexion persistante sur Internet ou sur le réseau contenant le serveur de courrier. Malheureusement, pour les personnes dont les connexions réseau sont lentes, **POP** exige que les programmes clients téléchargent la totalité du contenu de chaque message une fois l'authentification effectuée. Ceci peut prendre un long moment si un courrier contient une pièce jointe de grande taille.

La version la plus courante du protocole **POP** standard est **POP3**.

Cependant, il existe également des variantes moins utilisées du protocole **POP** :

- *APOP* — **POP3** avec une authentification **MD5**. Un hachage chiffré du mot de passe utilisateur est envoyé à partir du client de messagerie vers le serveur, plutôt que d'envoyer un mot de passe non chiffré.
- *KPOP* — **POP3** avec authentification Kerberos.
- *RPOP* — **POP3** avec authentification **RPOP**. Ceci utilise un ID par utilisateur, similairement à un mot de passe, mais pour authentifier les requêtes POP. Cependant, cet ID n'est pas chiffré, et **RPOP** n'est ainsi pas plus sécurisé que le protocole **POP** standard.

Pour une meilleure sécurité, il est possible d'utiliser le chiffrement *SSL* (*Secure Socket Layer*) pour l'authentification de client et les sessions de transferts de données. Ceci peut être activé en utilisant le service **pop3s**, ou en utilisant l'application **stunnel**. Pour obtenir davantage d'informations sur la sécurisation des communications par courrier électronique, veuillez consulter la Section 13.5.1, « Sécuriser les communications ».

13.1.2.2. IMAP

Le serveur **IMAP** par défaut sous Red Hat Enterprise Linux se nomme **Dovecot** et est fourni par le paquet dovecot. Veuillez consulter la Section 13.1.2.1, « POP » pour obtenir des informations sur la manière d'installer **Dovecot**.

Lors de l'utilisation d'un serveur de courrier **IMAP**, les messages électroniques restent sur le serveur, les utilisateurs peuvent donc les lire ou les supprimer. **IMAP** autorise également les applications clientes à créer, renommer, ou supprimer des répertoires de courrier sur le serveur pour organiser et stocker les courriers électroniques.

IMAP est particulièrement utile pour les utilisateurs accédant à leur courrier électronique à partir de plusieurs machines. Le protocole est également utile pour les utiisateurs se connectant au serveur de courrier via une connexion lente, car jusqu'à ce que le courrier soit ouvert, seules les informations de l'en-tête des courriers électroniques sont téléchargées, économisant ainsi de la bande passante. L'utilisateur a également la possibilité de supprimer des messages sans les voir ou sans les télécharger.

Pour plus de commodité, les applications clientes **IMAP** sont capables de mettre en cache des copies de messages localement, permettant ainsi à l'utilisateur de parcourir des messages précédemment lus sans pour autant être directement connecté au serveur **IMAP**.

IMAP, tout comme **POP**, est complètement compatible avec les standards de messagerie Internet importants, tels que MIME, qui autorise l'ajout de pièces jointes.

Pour une meilleure sécurité, il est possible d'utiliser le chiffrement **SSL** pour l'authentification de client et les sessions de transfert de données. Ceci peut être activé en utilisant le service **imaps**, ou en utilisant le programme **stunnel**. Pour obtenir davantage d'informations sur la sécurisation des communications par courrier électronique, veuillez consulter la Section 13.5.1, « Sécuriser les communications ».

D'autres clients et serveurs IMAP gratuits ou de type commercial sont également disponibles, et nombre d'entre eux étendent le protocole IMAP et fournissent des fonctionnalités supplémentaires.

13.1.2.3. dovecot

Les processus **imap-login** et **pop3-login** qui implémentent les protocoles **IMAP** et **POP3** sont engendrés par le démon **dovecot** maître inclus dans le paquet dovecot. L'utilisation d'**IMAP** et de **POP** est configurée via le fichier de configuration **/etc/dovecot/dovecot.conf** ; par défaut **dovecot** exécute **IMAP** et **POP3** avec leur version sécurisée en utilisant **SSL**. Pour configurer **dovecot** de manière à utiliser **POP**, veuillez appliquer les étapes suivantes :

1. Modifiez le fichier de configuration **/etc/dovecot/dovecot.conf** de manière à vous assurer que la variable **protocols** ne se trouve pas dans un commentaire (supprimez le caractère dièse (#) au début de la ligne) et qu'elle contienne bien l'argument **pop3**. Exemple :

   ```
   protocols = imap pop3 lmtp
   ```

   Lorsque la variable **protocols** ne se trouve plus dans un commentaire, **dovecot** utilisera les valeurs par défaut comme décrit ci-dessus.

2. Rendre le changement opérationnel pour la session actuelle en exécutant la commande suivante en tant qu'utilisateur **root** :

   ```
   ~]# systemctl restart dovecot
   ```

3. Rendre le changement opérationnel après le prochain redémarrage en exécutant la commande :

```
~]# systemctl enable dovecot
Created symlink from /etc/systemd/system/multi-
user.target.wants/dovecot.service to
/usr/lib/systemd/system/dovecot.service.
```

NOTE

Veuillez remarquer que **dovecot** rapporte uniquement qu'il a démarré le serveur **IMAP**, mais il lance également le serveur **POP3**.

Contrairement à **SMTP**, **IMAP** et **POP3** nécessitent de connecter les clients pour s'authentifier en utilisant un nom d'utilisateur et un mot de passe.. Par défaut, les mots de passe de ces deux protocoles sont passés à travers le réseau de manière non-chiffrée.

Pour configurer **SSL** sur **dovecot** :

- Modifiez le fichier de configuration **/etc/dovecot/conf.d/10-ssl.conf** pour vous assurer que la variable **ssl_protocols** est bien décommentée et contient les arguments **!SSLv2 !SSLv3** :

  ```
  ssl_protocols = !SSLv2 !SSLv3
  ```

  Ces valeurs permettent de s'assurer que **dovecot** évite SSL versions 2 et 3, qui sont toutes deux connues pour ne pas être sécurisées. Ceci est dû à la vulnérabilité décrite dans *POODLE: SSLv3 vulnerability (CVE-2014-3566)*. Veuillez consulter la *Résolution de la vulnérabilité SSL 3.0 POODLE (CVE-2014-3566) dans Postfix et Dovecot* pour obtenir davantage de détails.

- Modifiez le fichier de configuration **/etc/pki/dovecot/dovecot-openssl.cnf** selon vos préférences. Cependant, dans une installation typique, ce fichier ne requiert pas de modification.

- Renommez, déplacez ou supprimez les fichiers **/etc/pki/dovecot/certs/dovecot.pem** et **/etc/pki/dovecot/private/dovecot.pem**.

- Exécutez le script **/usr/libexec/dovecot/mkcert.sh** qui crée les certificats auto-signés **dovecot**. Ces certificats sont copiés dans les répertoires **/etc/pki/dovecot/certs** et **/etc/pki/dovecot/private**. Pour implémenter les changements, redémarrez **dovecot** en exécutant la commande suivante en tant qu'utilisateur **root** :

  ```
  ~]# systemctl restart dovecot
  ```

Davantage de détails sur **dovecot** peuvent être trouvés en ligne sur http://www.dovecot.org.

13.2. CLASSIFICATIONS DES PROGRAMMES DE COURRIER ÉLECTRONIQUE

En général, toutes les applications de courrier électronique font partie d'au moins une de trois classifications. Chaque classification joue un rôle particulier dans le déplacement et la gestion de messages électroniques. Même si la plupart des utilisateurs connaissent uniquement le logiciel de courrier électronique qu'ils utilisent pour envoyer et recevoir des messages, chaque logiciel est important pour que le courrier électronique puisse arriver à la destination correcte.

13.2.1. Agent de transport de courrier

Un *agent de transport de courrier* (ou *MTA*) transporte les courriers électroniques entre hôtes à l'aide du protocole **SMTP**. Un message peut impliquer plusieurs MTA pendant son déplacement vers la destination prévue.

Même si la remise de messages entre machines est relativement claire, le processus de décision pour savoir si un MTA particulier doit ou devrait accepter un message pour le remettre est assez compliqué. En outre, dû à des problèmes liés au courrier indésirable, l'utilisation d'un MTA particulier est habituellement limitée par la configuration du MTA ou par la configuration d'accès au réseau sur lequel le MTA réside.

De nombreux programmes clients de courrier électronique modernes peuvent agir en tant que MTA lors de l'envoi de courrier électronique. Cependant, cette action ne doit pas être confondue avec le rôle d'un réel MTA. La seule raison pour laquelle les programmes client de messagerie sont capables d'envoyer un courrier comme un MTA s'explique par le fait que l'hôte exécutant l'application ne possède pas son propre MTA. Ceci est particulièrement vrai des programmes clients de messagerie sur des systèmes d'exploitation non basés sur UNIX. Cependant, ces programmes client envoient uniquement des messages sortants sur un MTA qu'ils sont autorisés à utiliser et ne remettent pas le message directement au serveur de courrier du destinataire prévu.

Comme Red Hat Enterprise Linux propose deux MTA, *Postfix* et *Sendmail*, les programmes clients de messagerie n'ont pas besoin d'agir en tant que MTA. Red Hat Enterprise Linux inclut également un MTA à caractère particulier nommé *Fetchmail*.

Pour obtenir davantage d'informations sur Postfix, Sendmail, et Fetchmail, veuillez consulter la Section 13.3, « Agents de transport de courrier ».

13.2.2. Agent distributeur de courrier

Un *agent distributeur de courrier* (ou *MDA*) est invoqué par le MTA pour archiver le courrier entrant dans la bonne boîte à lettres de l'utilisateur. Dans de nombreux cas, le MDA est un *Agent distributeur local* (ou *LDA*), tel que **mail** ou Procmail.

Tout programme qui gère la remise d'un message jusqu'au moment où il peut être lu par une application cliente de messagerie peut être considéré comme un MDA. Pour cette raison, certains MTA (tels que Sendmail et Postfix) peuvent jouer le rôle d'un MDA lorsqu'ils ajoutent de nouveaux messages électroniques au fichier spool du courrier d'un utilisateur local. En général, les MDA ne transportent pas de messages entre systèmes et ne fournissent pas d'interface utilisateur ; les MDA distribuent et arrangent les messages sur la machine locale pour qu'une application cliente de messagerie puisse y accéder.

13.2.3. Agent d'utilisateur de messagerie

Un *agent d'utilisateur de messagerie* (*MUA*) est synonyme d'une application cliente de messagerie. Un MUA est un programme qui, au minimum, permet à un utilisateur de lire et de composer des messages électroniques. De nombreux MUA sont capables de récupérer des messages via les protocoles **POP** ou **IMAP**, paramétrant des boîtes à lettres pour stocker des messages, et envoyant des messages sortant vers un MTA.

Les MUA peuvent être graphiques, comme **Evolution**, ou être de simples interfaces basées texte, comme **Mutt**.

13.3. AGENTS DE TRANSPORT DE COURRIER

Red Hat Enterprise Linux 7 offre deux principaux MTA : Postfix et Sendmail. Postfix est configuré en tant que MTA par défaut et Sendmail est considéré comme déconseillé. Si vous vous trouvez dans l'obligation de changer le MTA par défaut sur Sendmail, vous pouvez désinstaller Postfix ou utiliser la commande suivante en tant qu'utilisateur **root** pour basculer sur Sendmail :

```
~]# alternatives --config mta
```

Vous pouvez également utiliser la commande suivante pour activer le service souhaité :

```
~]# systemctl enable service
```

Similairement, pour désactiver le service, veuillez saisir ce qui suit dans une invite de shell :

```
~]# systemctl disable service
```

Pour obtenir davantage d'informations sur la manière de gérer les services système sur Red Hat Enterprise Linux 7, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

13.3.1. Postfix

Développé à l'origine par un expert en sécurité et programmeur chez IBM, Wietse Venema, Postfix est un MTA compatible avec Sendmail conçu de manière à être sécurisé, rapide, et facile à configurer.

Pour améliorer la sécurité, Postfix utilise un design modulaire, où de petits processus avec des privilèges limités sont lancés par un démon maître *master*. Les plus petits processus, moins privilégiés, effectuent des tâches très particulières liées aux différentes étapes de la remise du courrier et sont exécutés dans un environnement racine modifié pour limiter les effets de toute attaque.

Configurer Postfix pour accepter des connexions réseau en provenance d'hôtes autres que l'ordinateur local ne demande que quelques changements mineurs dans son fichier de configuration. Lorsque des besoins plus complexes se font sentir, Postfix fournit une variété d'options de configuration, ainsi que des greffons de tierces parties qui en font un MTA très versatile et avec des nombreuses fonctionnalités.

Les fichiers de configuration de Postfix sont lisibles par les humains et prennent en charge plus de 250 directives. Contrairement à Sendmail, aucun traitement macro n'est requis pour que les changements entrent en vigueur, et la majorité des options les plus couramment utilisées sont décrites dans les fichiers lourdement commentés.

13.3.1.1. Installation Postfix par défaut

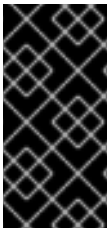
L'exécutable Postfix se nomme **postfix**. Ce démon lance tous les processus nécessaires à la gestion de la remise du courrier.

Postfix stocke ses fichiers de configuration dans le répertoire **/etc/postfix/**. Voici une liste de fichiers les plus couramment utilisés :

- **access** — utilisé pour le contrôle des accès, ce fichier spécifie les hôtes autorisés à se connecter à Postfix.
- **main.cf** — fichier de configuration global de Postfix. La majorité des options de configuration sont spécifiées dans ce fichier.
- **master.cf** — spécifie la manière par laquelle Postfix interagit avec les divers processus pour effectuer la remise du courrier.

- **transport** — fait correspondre les adresses électroniques avec les hôtes de relais.

Le fichier **aliases** se trouve dans le répertoire **/etc**. Ce fichier est partagé entre Postfix et Sendmail. Il s'agit d'une liste configurable requise par le protocole de messagerie qui décrit les alias des ID utilisateurs.



IMPORTANT

Le fichier **/etc/postfix/main.cf** par défaut n'autorise pas Postfix à accepter de connexions réseau d'un hôte autre que l'ordinateur local. Pour obtenir des instructions sur la configuration de Postfix en tant que serveur pour d'autres clients, veuillez consulter la Section 13.3.1.3, « Configuration de base Postfix ».

Redémarrez le service **postfix** après avoir modifié toute option des fichiers de configuration sous le répertoire **/etc/postfix/** afin que les changements entrent en vigueur. Pour faire cela, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl restart postfix
```

13.3.1.2. Mettre à niveau à partir d'une version précédente

Les paramètres suivants de Red Hat Enterprise Linux 7 sont différents des versions précédentes :

- **disable_vrfy_command = no** — ce paramètre est désactivé par défaut, ce qui est différent des valeurs par défaut de Sendmail. Si modifié sur **yes**, il peut empêcher certaines méthodes de récolte d'adresses électroniques.

- **allow_percent_hack = yes** — ce paramètre est activé par défaut. Il permet la suppression des caractères **%** dans les adresses électroniques. Le bricolage du pourcentage est une ancienne solution de contournement qui permettait le routage contrôlé par l'envoyeur des messages électroniques. **DNS** et le routage de courrier sont désormais bien plus fiables, mais Postfix continue de prendre en charge ce bricolage. Pour désactiver la réécriture du pourcentage, veuillez définir **allow_percent_hack** sur **no**.

- **smtpd_helo_required = no** — ce paramètre est désactivé par défaut, tout comme sur Sendmail, car il empêche certaines applications d'envoyer des courriers électroniques. Il peut être modifié sur **yes** pour requérir des clients qu'ils envoient des commandes HELO ou EHLO avant de tenter d'envoyer les commandes MAIL, FROM, ou ETRN.

13.3.1.3. Configuration de base Postfix

Par défaut, Postfix n'accepte pas de connexion réseau d'hôte différent que l'hôte local. Veuillez reproduire les étapes suivantes en tant qu'utilisateur **root** pour activer la remise de courrier pour d'autres hôtes sur le réseau :

- Modifiez le fichier **/etc/postfix/main.cf** avec un editeur de texte, tel que **vi**.

- Décommentez la ligne **mydomain** en supprimant le caractère dièse (#), et remplacez *domain.tld* par le domaine entretenu par le serveur de courrier, tel que **example.com**.

- Décommentez la ligne **myorigin = $mydomain**.

- Décommentez la ligne **myhostname**, et remplacez *host.domain.tld* par le nom d'hôte de la machine.
- Décommentez la ligne **mydestination = $myhostname, localhost.$mydomain**.
- Décommentez la ligne **mynetworks**, et remplacez *168.100.189.0/28* par un paramètre réseau valide pour les hôtes qui peuvent se connecter au serveur.
- Décommentez la ligne **inet_interfaces = all**.
- Mettez la ligne **inet_interfaces = localhost** en commentaire.
- Redémarrez le service **postfix**.

Une fois ces étapes reproduites, l'hôte acceptera de remettre des courriers extérieurs.

Postfix offre un large éventail d'options de configuration. L'une des meilleures manières d'apprendre comment configurer Postfix consiste à lire les commentaires dans le fichier de configuration **/etc/postfix/main.cf**. Des ressources supplémentaires, y compris des informations sur la configuration de Postfix, l'intégration de SpamAssassin, ou des descriptions détaillées des paramètres **/etc/postfix/main.cf** sont disponibles en ligne sur http://www.postfix.org/.

IMPORTANT

À cause de la vulnérabilité décrite dans *POODLE: SSLv3 vulnerability (CVE-2014-3566)*, Red Hat recommande de désactiver **SSL** et d'utiliser **TLSv1.1** ou **TLSv1.2** uniquement. Voir *Résolution de la vulnérabilité POODLE SSL 3.0 (CVE-2014-3566) dans Postfix et Dovecot* pour obtenir plus de détails.

13.3.1.4. Utiliser Postfix avec LDAP

Postfix peut utiliser un répertoire **LDAP** en tant que source pour diverses tables de recherche (par exemple : **aliases**, **virtual**, **canonical**, etc.). Ceci permet à **LDAP** de stocker des informations utilisateur hiérarchiques et à Postfix de recevoir le résultat des requêtes **LDAP** uniquement lorsque nécessaire. En ne stockant pas ces informations localement, les administrateurs peuvent les maintenir plus facilement.

13.3.1.4.1. L'exemple de recherche /etc/aliases

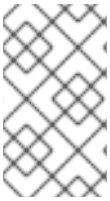
Ci-dessous figure un exemple de base d'utilisation de **LDAP** pour rechercher le fichier **/etc/aliases**. Assurez-vous que le fichier **/etc/postfix/main.cf** contienne bien :

```
alias_maps = hash:/etc/aliases, ldap:/etc/postfix/ldap-aliases.cf
```

Veuillez créer un fichier **/etc/postfix/ldap-aliases.cf**, si vous n'en possédez pas déjà un, et assurez-vous qu'il contiennent effectivement ce qui suit :

```
server_host = ldap.example.com
search_base = dc=example, dc=com
```

où ***ldap.example.com***, ***example***, et ***com*** sont des paramètres qui doivent être remplacés avec la spécification d'un serveur **LDAP** existant et disponible.



NOTE

Le fichier **/etc/postfix/ldap-aliases.cf** peut spécifier divers paramètres, y compris des paramètres activant **LDAP SSL** et **STARTTLS**. Pour obtenir davantage d'informations, veuillez consulter la page man de **ldap_table(5)**.

Pour plus d'informations sur **LDAP**, voir OpenLDAP dans le guide *System-Level Authentication Guide*.

13.3.2. Sendmail

Le but premier de Sendmail, tout comme les autres MTA, est de transférer le courrier entre les hôtes en toute sécurité, ce qui est habituellement fait à l'aide du protocole **SMTP**. Veuillez remarquer que l'utilisation de Sendmail est déconseillée et ses utilisateurs sont encouragés à utiliser Postfix lorsque possible. Veuillez consulter la Section 13.3.1, « Postfix » pour obtenir davantage d'informations.

13.3.2.1. Objectif et limites

Il est important de savoir ce que Sendmail est et peut faire, ainsi que de connaître ses limitations. De nos jours, avec des applications monolithiques remplissant de multiples rôles, Sendmail peut sembler être la seule application nécessaire pour exécuter un serveur de courrier électronique dans une organisation. Techniquement, cette affirmation est véridique, car Sendmail pour mettre en spool le courrier sur le répertoire de chaque utilisateur et distribuer le courrier sortant des utilisateurs. Cependant, la plupart des utilisateurs requièrent bien plus qu'une simple distribution du courrier. Les utilisateurs souhaitent habituellement interagir avec leur courrier électronique en utilisant un MUA, qui utilise **POP** ou **IMAP** pour télécharger les messages sur la machine locale. Ces utilisateurs peuvent également préférer d'utiliser une interface Web pour obtenir accès à leur boîte de réception. Les autres applications peuvent fonctionner en conjonction avec Sendmail, mais elles existent pour différentes raisons et peuvent opérer séparément les unes des autres.

Tout ce que Sendmail pourrait ou devrait être configuré à faire est au-delà de l'étendue de cette section. Des centaines d'options et de règles, et des volumes entiers ont été consacrés à expliquer tout ce qui peut être fait et comment résoudre les problèmes pouvant se produire. Veuillez consulter la Section 13.6, « Ressources supplémentaires » pour afficher une liste des ressources Sendmail.

Cette section examine les fichiers installés avec Sendmail par défaut et les changements de configuration de base, y compris comment arrêter de recevoir du courrier indésirable (spam) et comment étendre Sendmail avec le protocole LDAP *(Lightweight Directory Access Protocol)*.

13.3.2.2. Installation Sendmail par défaut

Pour utiliser Sendmail, commencez par vous assurer que le paquet sendmail soit installé sur votre système en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install sendmail
```

Pour configurer Sendmail, assurez-vous que le paquet sendmail-cf est installé sur votre système en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install sendmail-cf
```

Pour obtenir davantage d'informations sur l'installation de paquets avec Yum, veuillez consulter la Section 8.2.4, « Installation de paquets ».

Avant d'utiliser Sendmail, le MTA par défaut doit être changé à partir de Postfix. Pour obtenir davantage d'informations sur la manière de changer le MTA par défaut, veuillez consulter la Section 13.3, « Agents de transport de courrier ».

L'exécutable Sendmail est nommé **sendmail**.

Le fichier de configuration Sendmail long et détaillé est nommé **/etc/mail/sendmail.cf**. Veuillez éviter de modifier le fichier **sendmail.cf** directement. Pour effectuer des changements de configuration sur Sendmail, veuillez modifier le fichier **/etc/mail/sendmail.mc**, effectuer une copie de sauvegarde du fichier **/etc/mail/sendmail.cf** d'origine, puis utiliser les alternatives suivantes pour générer un nouveau fichier de configuration :

- Veuillez utiliser le fichier makefile inclus dans le fichier de configuration **/etc/mail/** pour créer un nouveau fichier de configuration **/etc/mail/sendmail.cf** :

  ```
  ~]# make all -C /etc/mail/
  ```

  Tous les autres fichiers générés dans **/etc/mail** (fichiers db) seront générés à nouveau si nécessaire. Les anciennes commandes makemap sont toujours utilisables. La commande make est automatiquement utilisée lorsque vous démarrez ou redémarrez le service **sendmail**.

Davantage d'information sur la configuration de Sendmail se trouve dans la Section 13.3.2.3, « Changements communs de la configuration Sendmail ».

Divers fichiers de configuration Sendmail sont installés dans le répertoire **/etc/mail/**, y compris :

- **access** — indique quels systèmes peuvent utiliser Sendmail pour les courriers électroniques sortants.
- **domaintable** — spécifie le mappage du nom de domaine.
- **local-host-names** — spécifie les alias de l'hôte.
- **mailertable** — spécifie les instructions qui outrepassent le routage de certains domaines particuliers.
- **virtusertable** — spécifie une forme de crénelage spécifique aux domaines, permettant à de multiples domaines virtuels d'être hébergés sur une machine.

Plusieurs fichiers de configuration dans le répertoire **/etc/mail/**, comme **access**, **domaintable**, **mailertable** et **virtusertable**, doivent stocker leurs informations dans des fichiers de base de données avant que Sendmail puisse utiliser un changement de configuration. Pour inclure tout changement apporté à ces configurations dans leurs fichiers de base de données, veuillez exécuter les commandes suivantes en tant qu'utilisateur **root** :

```
~]# cd /etc/mail/
~]# make all
```

Ceci mettra à jour **virtusertable.db**, **access.db**, **domaintable.db**, **mailertable.db**, **sendmail.cf**, et **submit.cf**.

Pour mettre à jour tous les fichiers de la base de données répertoriés ci-dessus, et pour mettre à jour un fichier personnalisé de base de données, veuillez utiliser une commande sous le format suivant :

```
make name.db all
```

où *name* correspond au nom du fichier personnalisé de la base de données à mettre à jour.

Pour mettre à jour une seule base de données, veuillez utiliser une commande du format suivant :

```
make name.db
```

où *name.db* correspond au nom du fichier de la base de données à mettre à jour.

Vous pouvez également redémarrer le service **sendmail** pour que les changements entrent en vigueur en exécutant :

```
~]# systemctl restart sendmail
```

Par exemple, pour que tous les courriers électroniques adressés au domaine **example.com** soient remis à **bob@other-example.com**, veuillez ajouter la ligne suivante au fichier **virtusertable** :

```
@example.com bob@other-example.com
```

Pour finaliser le changement, le fichier **virtusertable.db** doit être mis à jour :

```
~]# make virtusertable.db all
```

L'utilisation de l'option **all** provoquera la mise à jour simultanée de **virtusertable.db** et **access.db**.

13.3.2.3. Changements communs de la configuration Sendmail

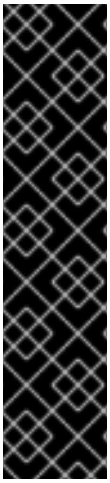
Lors de l'altération du fichier de configuration Sendmail, il vaut mieux ne pas modifier un fichier existant, mais générer un fichier **/etc/mail/sendmail.cf** complètement nouveau.

AVERTISSEMENT

Avant de remplacer ou d'effectuer tout changement sur le fichier **sendmail.cf**, veuillez créer une copie de sauvegarde.

Pour ajouter la fonctionnalité souhaitée à Sendmail, veuillez modifier le fichier **/etc/mail/sendmail.mc** en tant qu'utilisateur **root**. Après avoir terminé, veuillez redémarrer le service **sendmail**, si le paquet m4 est installé, le processeur macro **m4** générera automatiquement un nouveau fichier de configuration **sendmail.cf** :

```
~]# systemctl restart sendmail
```



IMPORTANT

Le fichier **sendmail.cf** par défaut interdit à Sendmail d'accepter des connexions réseau en provenance de tout autre hôte que l'ordinateur local. Pour configurer Sendmail en tant que serveur pour d'autres clients, veuillez modifier le fichier **/etc/mail/sendmail.mc**, et modifiez l'adresse spécifiée dans l'option **Addr=** de la directive **DAEMON_OPTIONS** en provenance de **127.0.0.1** par l'adresse IP d'un périphérique réseau actif ou mettez en commentaire la directive **DAEMON_OPTIONS** toute entière en plaçant **dnl** au début de la ligne. Une fois cela terminé, veuillez générer un nouveau fichier **/etc/mail/sendmail.cf** en redémarrant le service :

```
~]# systemctl restart sendmail
```

La configuration par défaut dans Red Hat Enterprise Linux fonctionne pour la plupart des sites utilisant uniquement **SMTP**. Cependant, cela ne fonctionne pas pour les sites *UUCP* (« *UNIX-to-UNIX Copy Protocol* »). Si vous utilisez le transfert de courrier UUCP, le fichier **/etc/mail/sendmail.mc** doit être reconfiguré et un nouveau fichier **/etc/mail/sendmail.cf** doit être généré.

Veuillez consulter le fichier **/usr/share/sendmail-cf/README** avant de mettre à jour tout fichier dans les répertoires situés sous le répertoire **/usr/share/sendmail-cf/**, car ils peuvent affecter la future configuration du fichier **/etc/mail/sendmail.cf**.

13.3.2.4. Camouflage

Une configuration Sendmail commune consiste à faire en sorte qu'une seule machine agisse en tant que passerelle pour toutes les machines sur le réseau. Par exemple, une société pourrait souhaiter avoir une machine appelée **mail.example.com** qui gère tout le courrier électronique et assigne une adresse de retour cohérente à tout courrier sortant.

Dans cette situation, le serveur Sendmail doit camoufler les noms de machine sur le réseau de la société de manière à ce que l'adresse de retour affiche **user@example.com** au lieu de **user@host.example.com**.

Pour faire cela, veuillez ajouter les lignes suivantes à **/etc/mail/sendmail.mc** :

```
FEATURE(always_add_domain)dnl
FEATURE(`masquerade_entire_domain')dnl
FEATURE(`masquerade_envelope')dnl
FEATURE(`allmasquerade')dnl
MASQUERADE_AS(`example.com.')dnl
MASQUERADE_DOMAIN(`example.com.')dnl
MASQUERADE_AS(example.com)dnl
```

Après avoir généré un nouveau fichier **sendmail.cf** en utilisant le processeur macro **m4**, cette configuration fait apparaître tout le courrier intérieur au réseau comme s'il avait été envoyé par **example.com**.

Remarquez que les administrateurs de serveurs de courrier, des serveurs **DNS** et **DHCP**, ainsi que toute application de provisioning, doivent s'accorder sur le format des noms d'hôtes utilisé dans une organisation. Veuillez consulter le Guide de mise en réseau Red Hat Enterprise Linux 7 pour obtenir davantage d'informations sur les pratiques de dénomination recommandées.

13.3.2.5. Arrêter le courrier indésirable

Le courrier indésirable est le courrier non nécessaire et non solicité reçu par un utilisateur qui n'a jamais requis de communication. Il s'agit d'un abus perturbateur, coûteux et répandu des standards de communication internet.

Sendmail rend relativement facile le blocage des nouvelles techniques de spam utilisées pour envoyer du courrier indésirable. Il bloque également les méthodes de spam les plus courantes par défaut. Les principales fonctionnalités anti-spam disponibles sur Sendmail incluent la vérification des en-têtes (*header checks*), le déni de relais (*relaying denial*, par défaut la version 8.9), et les vérifications d'informations sur l'envoyeur et sur les bases de données d'accès (*access database and sender information checks*).

Par exemple, le transfert de messages **SMTP**, aussi appelé relais, a été désactivé par défaut depuis la version 8.9 de Sendmail. Avant ce changement, Sendmail ordonnait à l'hôte du courrier (**x.edu**) d'accepter les messages d'une partie (**y.com**) et de les envoyer à une autre partie (**z.net**). Cependant, Sendmail doit désormais être configuré pour autoriser à tout domaine de relayer le courrier à travers le serveur. Pour configurer les domaines de relais, veuillez modifier le fichier **/etc/mail/relay-domains** et redémarrer Sendmail

```
~]# systemctl restart sendmail
```

Cependant les utilisateurs peuvent également recevoir du courrier indésirable provenant de serveurs sur internet. Dans ce cas, les fonctionnalités de contrôle d'accès de Sendmail disponibles à travers le fichier **/etc/mail/access** peuvent être utilisées pour empêcher des connexions d'hôtes indésirables. L'exemple suivant illustre comment utiliser ce fichier pour bloquer et spécifiquement autoriser l'accès au serveur Sendmail :

```
badspammer.com ERROR:550 "Go away and do not spam us anymore"
tux.badspammer.com OK 10.0 RELAY
```

Cet exemple montre que tout courrier électronique envoyé à partir de **badspammer.com** est bloqué avec un code d'erreur de conformité 550 RFC-821, avec un message renvoyé. Le courrier électronique envoyé à partir du sous-domaine **tux.badspammer.com** est accepté. La dernière ligne affiche que tout courrier électronique envoyé depuis le réseau 10.0.*.* peut être relayé à travers le serveur de courrier.

Comme le fichier **/etc/mail/access.db** est une base de données, veuillez utiliser la commande **makemap** pour mettre à jour tout changement. Pour effectuer cela, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

```
~]# makemap hash /etc/mail/access < /etc/mail/access
```

L'analyse d'en-tête de message vous permet de rejeter le courrier basé sur le contenu des en-têtes. Les serveurs **SMTP** stockent des informations sur le parcours d'un courrier électronique dans l'en-tête du message. Tandis que le messages se déplacent d'un MTA à un autre, chacun ajoute un en-tête **Received** (« Reçu ») au-dessus des autres en-têtes **Received**. Il est important de noter que ces informations peuvent être altérées par les expéditeurs du courrier indésirable.

Les exemples ci-dessus représentent uniquement une petite partie de ce que Sendmail peut faire lors de l'autorisation ou du blocage des accès. Veuillez afficher le fichier **/usr/share/sendmail-cf/README** pour obtenir davantage d'informations et d'exemples.

Comme Sendmail appelle le MDA Procmail lors de la remise du courrier, il est également possible d'utiliser un programme de filtrage de courrier indésirable, tel que SpamAssassin, pour identifier et archiver le courrier indésirable pour les utilisateurs. Veuillez consulter la Section 13.4.2.6, « Filtres du courrier indésirable » pour obtenir des informations supplémentaires sur l'utilisation de SpamAssassin.

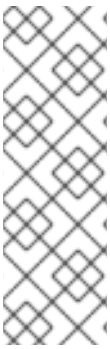
13.3.2.6. Utiliser Sendmail avec LDAP

Utiliser **LDAP** est une manière rapide et puissante de trouver des informations spécifiques sur un utilisateur particulier dans un groupe de grande taille. Par exemple, un serveur **LDAP** peut être utilisé pour rechercher une adresse électronique particulière dans un annuaire d'entreprise commun en utilisant le nom de famille de l'utilisateur. Dans ce type d'implémentation, **LDAP** est relativement différent de Sendmail, avec **LDAP** stockant les informations hiérarchiques des utilisateurs et Sendmail ne reçevant que le résultat des requêtes **LDAP** dans des messages électroniques pré-adressés.

Cependant, Sendmail prend en charge une intégration bien plus importante avec **LDAP**, avec laquelle **LDAP** est utilisé pour remplacer des fichiers maintenus séparément, comme **/etc/aliases** et **/etc/mail/virtusertables**, sur différents serveurs de courrier fonctionnant ensemble pour prendre en charge une organisation de niveau moyen à niveau entreprise. Autrement dit, **LDAP** fait une abtraction du niveau de routage du courrier de Sendmail et de ses fichiers de configuration séparés sur un cluster **LDAP** puissant dont de nombreuses différentes applications peuvent tirer profit.

La version actuelle de Sendmail contient la prise en charge de **LDAP**. Pour étendre le serveur Sendmail utilisant **LDAP**, commencez par exécuter un serveur **LDAP** correctement configuré, tel que **OpenLDAP**. Puis modifiez **/etc/mail/sendmail.mc** de manière à inclure ceci :

```
LDAPROUTE_DOMAIN('yourdomain.com')dnl
FEATURE('ldap_routing')dnl
```



NOTE

Ceci est uniquement destiné à une configuration très basique de Sendmail avec **LDAP**. La configuration peut être largement différente en fonction de l'implémentation de **LDAP**, particulièrement lors de la configuration de plusieurs machines Sendmail pour utiliser un serveur **LDAP** commun.

Veuillez consulter **/usr/share/sendmail-cf/README** pour des instructions et exemples de configuration du routage **LDAP**.

Ensuite, veuillez recréer le fichier **/etc/mail/sendmail.cf** en exécutant le macro processeur **m4** et redémarrez Sendmail. Veuillez consulter la Section 13.3.2.3, « Changements communs de la configuration Sendmail » pour obtenir des instructions.

Pour plus d'informations sur **LDAP**, voir OpenLDAP dans le guide *System-Level Authentication Guide*.

13.3.3. Fetchmail

Fetchmail est un MTA qui récupère le courrier électronique se trouvant sur des serveurs distants et le remet au MTA local. De nombreux utilisateurs apprécient la capacité de séparer le processus de téléchargement de leurs messages se trouvant sur un serveur distant du processus de lecture et d'organisation du courrier électronique dans un MUA. Conçu à l'origine pour répondre aux besoins des utilisateurs accédant à internet par ligne téléphonique commutée, Fetchmail se connecte et télécharge rapidement tous les messages électroniques sur le fichier spool du courrier en utilisant un certain nombre de protocoles, y compris **POP3** et **IMAP**. Il peut également transférer les messages électroniques sur un serveur **SMTP** si nécessaire.

NOTE

Pour utiliser **Fetchmail**, veuillez vous assurer que le paquet fetchmail est installé sur votre système en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install fetchmail
```

Pour obtenir davantage d'informations sur l'installation de paquets avec Yum, veuillez consulter la Section 8.2.4, « Installation de paquets ».

Fetchmail est configuré pour chaque utilisateur à travers l'utilisation d'un fichier **.fetchmailrc** dams le répertoire personnel de l'utilisateur. S'il n'existe pas déjà, veuillez créer le fichier **.fetchmailrc** dans votre répertoire personnel

En utilisant les préférences dans le fichier **.fetchmailrc**, Fetchmail vérifie le courrier sur un serveur distant et le télécharge. Il le livre ensuite sur le port **25** de la machine locale, en utilisant le MTA local pour placer le courrier dans le fichier spool du bon utilisateur. Si Procmail est disponible, il est lancé pour filtrer le courrier électronique et le placer dans une boîte aux lettres afin qu'il soit lu par un MUA.

13.3.3.1. Options de Configuration Fetchmail

Malgré qu'il soit possible de passer toutes les options nécessaires sur la ligne de commande pour vérifier le courrier électronique sur un serveur distant pendant l'exécution de Fetchmail, l'utilisation d'un fichier **.fetchmailrc** est bien plus facile. Placez toutes les options de configuration souhaitées dans le fichier **.fetchmailrc** afin qu'elles soient utilisées chaque fois que la commande **fetchmail** est exécutée. Il est possible d'outrepasser ces commandes pendant l'exécution de Fetchmail en spécifiant cette option sur la ligne de commande.

Le fichier **.fetchmailrc** d'un utilisateur contient trois classes d'option de configuration :

- *options globales* — celles-ci donnent à Fetchmail des instructions contrôlant l'opération du programme ou fournissent des paramètres pour chaque connexion vérifiant le courrrier électronique.
- *options du serveur* — elles spécifient les informations nécessaires sur le serveur en cours d'interrogation, comme le nom d'hôte, ainsi que les préférences de serveurs de courrier électronique particuliers, comme le port à vérifier ou le nombre de secondes à attendre avant expiration. Ces options affectent tous les utilisateurs utilisant ce serveur.
- *options d'utilisateur* — contient des informations nécessaires à l'authentification et à la vérification du courrier électronique, telles que le nom d'utilisateur et le mot de passe, en utilisant un serveur de courrier électronique spécifié.

Des options globales apparaissent en haut du fichier **.fetchmailrc**, suivies par une ou plusieurs options de serveur, qui désigne chacune un différent serveur de courrier électronique que Fetchmail devrait vérifier. Les options d'utilisateur suivent les options de serveur pour chaque compte utilisateur vérifiant ce serveur de courrier électronique. Tout comme les options de serveur, des options de multiples utilisateurs peuvent être spécifiées pour une utilisation avec un serveur particulier, ainsi que pour vérifier de multiples comptes utilisateur sur le même serveur.

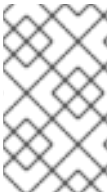
Des options de serveur sont activées dans le fichier **.fetchmailrc** par l'utilisation d'un verbe spéciale option, **poll** (interroger) ou **skip** (ignorer), qui précède les informations serveur. L'action **poll** (interroger) ordonne à Fetchmail d'utiliser cette option de serveur lors de son exécution. Elle vérifie le courrier électronique en utilisant les options spécifiées par l'utilisateur. Cependant, toute option de serveur située après une action **skip** (ignorer), ne sera pas vérifiée, à moins que le nom d'hôte de ce

serveur soit spécifié lorsque Fetchmail est invoqué. L'option **skip** est utile pendant des tests de configuration dans le fichier **.fetchmailrc** car elle ne vérifie que les serveurs ignorés si invoquée, et n'affecte aucune configuration fonctionnant actuellement.

Ci-dessous figure un exemple de fichier **.fetchmailrc** :

```
set postmaster "user1"
set bouncemail

poll pop.domain.com proto pop3
    user 'user1' there with password 'secret' is user1 here

poll mail.domain2.com
    user 'user5' there with password 'secret2' is user1 here
    user 'user7' there with password 'secret3' is user1 here
```

Dans cet exemple, les options globales indiquent que l'utilisateur reçoit du courrier électronique en dernier recours (option **postmaster**) et toutes les erreurs de courrier électronique sont envoyées au «postmaster » au lieu de l'envoyeur (option **bouncemail**). L'action **set** transmet à Fetchmail que cette ligne contient une option globale. Puis, deux serveurs de courrier sont spécifiés, l'un est paramétré pour vérifier en utilisant **POP3**, et l'autre pour tenter divers protocoles dans le but d'en trouver un qui fonctionne. Deux utilisateurs sont vérifés en utilisant la seconde option du serveur, mais tous le courrier trouvé pour un utilisateur quelconque est envoyé dans le spool du courrier de l'utilisateur **user1**. Ceci permet à de multiples boîtes aux lettres d'être vérifiées sur de multiples serveurs, tout en apparaissant comme n'étant qu'une seule boîte aux lettres MUA. Les informations spécifiques de chaque utilisateur commencent par l'action **user**.



NOTE

Les utilisateurs ne sont pas obligés de placer leur mot de passe dans le fichier **.fetchmailrc**. L'omission de la section **with password '*password*'** cause à Fetchmail de demander un mot de passe lorsqu'il est lancé.

Fetchmail possède de nombreuses options globales, de serveur, et locales. Nombre de ces options sont rarement utilisées ou ne s'appliquent qu'à des situations très spécifiques. La page man **fetchmail** explique chaque option en détail, mais les plus communes sont répertoriées dans les trois sections suivantes.

13.3.3.2. Options globales

Chaque option globale doit être placée sur une seule ligne après une action **set**.

- **daemon *seconds*** — spécifie le mode du démon, où Fetchmail reste en arrière-plan. Veuillez remplacer *seconds* par le nombre de secondes pendant lesquelles Fetchmail doit patienter avant d'interroger le serveur.
- **postmaster** — Spécifie un utilisateur local à qui envoyer le courrier en cas de problème de remise du courrier électronique.
- **syslog** — Spécifie le fichier journal pour les messages d'erreur et de statut. Par défaut, ce fichier est **/var/log/maillog**.

13.3.3.3. Options de serveur

Les options de serveur doivent être placées sur leur propre ligne dans **.fetchmailrc**, après une action **poll** ou **skip**.

- **auth *auth-type*** — remplace *auth-type* par le type d'authentification à utiliser. Par défaut, l'authentification **password** est utilisée, mais certains protocoles prennent en charge d'autres types d'authentification, y compris **kerberos_v5**, **kerberos_v4**, et **ssh**. Si le type d'authentification **any** est utilisé, Fetchmail tentera d'abord des méthodes qui ne requièrent pas de mot de passe, puis des méthodes qui masquent le mot de passe, et tentera finalement d'envoyer le mot de passe non chiffré pour s'authentifier sur le serveur.

- **interval *number*** — interroge le serveur spécifié toutes les ***number*** fois qu'il vérifie le courrier électronique sur tous les serveurs configurés. Cette option est généralement utilisée pour les serveurs de courrier sur lesquels l'utilisateur reçoit rarement de messages.

- **port *port-number*** — remplace *port-number* par le numéro du port. Cette valeur remplace le numéro de port par défaut pour le protocole spécifié.

- **proto *protocol*** — remplace *protocol* par le protocole à utiliser, comme **pop3** ou **imap**, pendant les vérifications de messages sur le serveur

- **timeout *seconds*** — remplace *seconds* par le nombre de secondes d'inactivité du serveur après lesquelles Fetchmail abandonnera une tentative de connexion. Si cette valeur n'est pas définie, la valeur par défaut de **300** sera utilisée.

13.3.3.4. Options d'utilisateur

Les options d'utilisateur peuvent être placées sur leurs propres lignes sous une option de serveur ou sur la même ligne que l'option de serveur. Dans les deux cas, les options définies doivent suivre l'option **user** (définie ci-dessous).

- **fetchall** — ordonne à Fetchmail de télécharger tous les messages de la file, y compris les messages qui ont déjà été lus. Par défaut, Fetchmail ne télécharge que les nouveaux messages.

- **fetchlimit *number*** — remplace *number* par le nombre de messages à récupérer avant d'arrêter.

- **flush** — supprime tous les messages déjà vus qui se trouvent dans la file avant de récupérer les nouveaux messages.

- **limit *max-number-bytes*** — remplace *max-number-bytes* par la taille de message maximale autorisée en octets lorsque les messages sont récupérés par Fetchmail. Cette option est utile avec les liens réseau lents, lorsqu'un message de grande taille prend trop de temps à télécharger.

- **password '*password*'** — remplace *password* par le mot de passe de l'utilisateur.

- **preconnect "*command*"** — remplace *command* par la commande à exécuter avant de récupérer les messages pour l'utilisateur.

- **postconnect "*command*"** — remplace *command* par la commande à exécuter après avoir récupéré les messages pour l'utilisateur.

- **ssl** — active l'encodage. Au moment de la rédaction de cet ouvrage, l'action par défaut est d'utiliser soit **SSL2**, **SSL3**, **SSL23**, **TLS1**, **TLS1.1** ou **TLS1.2** suivant ce qu'il y a de mieux de

disponible. Notez que **SSL2** est considéré comme étant obsolète à cause de POODLE: Vulnérabilité SSLv3 (CVE-2014-3566), **SSLv3** ne doit pas être utilisé. Cependant, il n'est pas utile de forcer l'utilisation de TLS1 ou version plus récente, donc, veillez bien à ce que le serveur de messagerie connecté soit configuré de façon à ne **pas** utiliser **SSLv2** ou **SSLv3**. Utiliser **stunnel** quand le serveur ne peut pas être configuré à ne **pas** utiliser **SSLv2** ou **SSLv3**.

- **sslproto** — définit les protocoles SSL ou TLS autorisés. Les valeurs possibles sont **SSL2**, **SSL3**, **SSL23** et **TLS1**. La valeur par défaut, si **sslproto** est omis, désactivé ou défini à une valeur non valide, est **SSL23**. L'action par défaut consiste à utiliser soit **SSLv2**, **SSLv3**, **TLSv1**, **TLS1.1** ou **TLS1.2**. Notez que la définition de toute autre valeur pour SSL ou TLS désactive tous les autres protocoles. À cause de POODLE:ulnérabilité SSLv3 (CVE-2014-3566), il est recommandé d'omettre cette option, ou de la définir à **SSLv23**, et de configurer le serveur de messagerie correspondant à ne **pas** utiliser **SSLv2** et **SSLv3**. Utilisez **stunnel** où le serveur ne peut pas être configuré à ne **pas** utiliser **SSLv2** et **SSLv3**.

- **user "*username*"** — Veuillez remplacer *username* par le nom d'utilisateur utilisé par Fetchmail pour récupérer les messages. *Cette option doit précéder toutes les autres options d'utilisateur.*

13.3.3.5. Options de commande Fetchmail

La plupart des options Fetchmail utilisées sur la ligne de commande lors de l'exécution de la commande **fetchmail** reflètent les options de configuration **.fetchmailrc**. Ainsi, Fetchmail peut être utilisé avec ou sans fichier de configuration. Ces options ne sont pas utilisées sur la ligne de commande par la plupart des utilisateurs car il est plus facile de les laisser dans le fichier **.fetchmailrc**.

Parfois, il est souhaitable d'exécuter la commande **fetchmail** avec d'autres options dans un but particulier. Il est possible d'exécuter des options de commande pour outrepasser de manière temporaire un paramètre **.fetchmailrc** qui causerait une erreur, comme toute option spécifiée sur la ligne de commande outrepasse les options du fichier de configuration.

13.3.3.6. Options de débogage ou à caractère informatif

Certaines options utilisées après la commande **fetchmail** peuvent fournir d'importantes informations.

- **--configdump** — affiche toutes les options possibles en se basant sur les informations de **.fetchmailrc** et sur les valeurs par défaut de Fetchmail. Aucun courrier n'est récupéré pour un utilisateur lorsque cette option est utilisée.

- **-s** — exécute Fetchmail en mode silence, empêchant tout message autre que des messages d'erreur d'apparaître après la commande **fetchmail**.

- **-v** — exécute Fetchmail en mode détaillé, affichant toutes les communications entre Fetchmail et les serveurs de courrier distants.

- **-V** — affiche des informations détaillées sur la version, répertorie ses options globales, et affiche les paramètres à utiliser avec chaque utilisateur, y compris le protocole du courrier et la méthode d'authentification. Aucun courrier n'est récupéré pour un utilisateur lorsque cette option est utilisée.

13.3.3.7. Options spéciales

Ces options sont parfois utiles pour remplacer les valeurs par défaut souvent trouvées dans le fichier **.fetchmailrc**.

- **-a** — Fetchmail télécharge tous les messages du serveur de courrier distant, qu'ils soient nouveaux ou qu'ils aient déjà été vus. Par défaut, Fetchmail télécharge uniquement les nouveaux messages.
- **-k** — Fetchmail laisse les messages sur le serveur de courrier distant après les avoir téléchargés. Cette option outrepasse le comportement par défaut qui consiste à supprimer les messages après les avoir téléchargés.
- **-l *max-number-bytes*** — Fetchmail ne télécharge aucun message au-delà d'une taille particulière et les laisse sur le serveur de courrier distant.
- **--quit** — quitte le processus du démon Fetchmail.

Davantage de commandes et d'options **.fetchmailrc** se trouvent sur la page man de **fetchmail**.

13.3.4. Configuration de l'agent de transport de courrier (« Mail Transport Agent », ou MTA)

Un MTA (« *Mail Transport Agent* ») est essentiel pour envoyer un courrier électronique. Un MUA (« *Mail User Agent* »), tel que **Evolution** ou **Mutt**, est utilisé pour lire et écrire des courriers électroniques. Lorsqu'un utilisateur envoie un courrier à partir d'un MUA, le message est remis au MTA, qui envoie le message à travers une série de MTA jusqu'à ce qu'il atteigne sa destination.

Même si un utilisateur ne planifie pas d'envoyer de courrier à partir du système, certaines tâches automatisées ou programmes du système peuvent devoir utiliser la commande **mail** pour envoyer du courrier contenant des messages de journalisation à l'utilisateur **root** du système local.

Red Hat Enterprise Linux 7 fournit deux MTA : Postfix et Sendmail. Si les deux sont installés, Postfix est le MTA par défaut. Veuillez remarquer que Sendmail est déconseillé dans Red Hat Enterprise Linux 7.

13.4. AGENTS DE REMISE DE COURRIER (« MAIL DELIVERY AGENTS »)

Red Hat Enterprise Linux inclut deux MDA principaux, Procmail et **mail**. Ces deux applications sont considérées comme des LDA et déplacent le courrier depuis le fichier spool du MTA dans la boîte aux lettre de l'utilisateur. Cependant, Procmail offre un système de filtrage robuste.

Cette section détaille uniquement Procmail. Pour obtenir des informations sur la commande **mail**, veuillez consulter sa page man (**man mail**).

Procmail remet et filtre le courrier car il se trouve dans le fichier spool du courrier de l'hôte local. Il est puissant, léger pour les ressources système, et est largement utilisé. Procmail peut jouer un rôle critique dans la remise du courrier pour lecture par les applications client de courrier.

Procmail peut être invoqué de différentes manières. Lorsqu'un MTA place un courrier dans le fichier spool du courrier, Procmail est lancé. Puis, Procmail filtre et archive le courrier pour le MUA et quitte. Sinon, le MUA peut être configuré pour exécuter Procmail à chaque fois qu'un message est reçu de façon à ce que les messages soient déplacés dans la boîte aux lettres qui convient. Par défaut, la présence d'un fichier **/etc/procmailrc** ou **~/.procmailrc** (également appelé un fichier *rc*) dans le répertoire personnel de l'utilisateur invoque Procmail chaque fois qu'un MTA reçoit un nouveau message.

Par défaut, aucun fichier **rc** global n'existe dans le répertoire **/etc** et aucun fichier **.procmailrc** n'existe dans les répertoires de base d'aucun utilisateur. Ainsi, pour utiliser Procmail, chaque utilisateur doit construire un fichier **.procmailrc** avec des variables et des règles d'environnement spécifiques.

La réaction de Procmail à la réception d'un courrier électronique dépend de si le message correspond à un ensemble spécifié de conditions ou de recettes (« *recipes* ») dans le fichier **rc**. Si un message correspond à une recette, alors le courrier est placé dans un fichier spécifié, supprimé, ou traité autrement.

Lorsque Procmail démarre, il lit le courrier et sépare le corps des informations de l'en-tête. Puis, Procmail recherche des variables et recettes d'environnement Procmail globales par défaut dans le fichier **/etc/procmailrc** et les fichiers **rc** du répertoire **/etc/procmailrcs/**. Procmail recherche ensuite un fichier **.procmailrc** dans le répertoire personnel de l'utilisateur. De nombreux utilisateurs créent également des fichiers **rc** supplémentaires pour Procmail auxquels il est fait référence dans le fichier **.procmailrc** de leur répertoire personnel.

13.4.1. Configuration Procmail

Le fichier de configuration Procmail contient des variables d'environnement importantes. Ces variables spécifient certaines choses, comme les messages à trier ou que faire avec les messages qui ne correspondent à aucune recette.

Ces variables d'environnement apparaissent habituellement au début du fichier **~/.procmailrc** sous le format suivant :

```
env-variable="value"
```

Dans cet exemple, ***env-variable*** est le nom de la variable et ***value*** définit la variable.

De nombreuses variables d'environnement n'étaient pas utilisées par la plupart des utilisateurs Procmail et nombre des variables d'environnement plus importantes sont déjà définies par une valeur par défaut. La plupart du temps, les variables suivantes sont utilisées :

- **DEFAULT** — définit la boîte aux lettres par défaut où sont placés les messages ne correspondant à aucune recette.

  La valeur par défaut **DEFAULT** est la même que la valeur **$ORGMAIL**.

- **INCLUDERC** — spécifie des fichiers **rc** supplémentaires contenant davantage de recettes avec lesquelles vérifier les messages. Ceci divise les listes des recettes Procmail en fichiers individuels qui remplissent différents rôles, comme bloquer le courrier indésirable et gérer les listes de courrier électronique, qui peuvent également être activés ou désactivés en utilisant les caractères de mise en commentaire dans le fichier de l'utilisateur **~/.procmailrc**.

  Par exemple, les lignes du fichier **~/.procmailrc** d'un utilisateur pourraient ressembler à celles-ci :

  ```
  MAILDIR=$HOME/Msgs
  INCLUDERC=$MAILDIR/lists.rc
  INCLUDERC=$MAILDIR/spam.rc
  ```

  Pour désactiver le filtrage Procmail des listes de courrier électronique tout en laissant le contrôle du courrier indésirable en place, veuillez mettre en commentaire la première ligne **INCLUDERC** avec le caractère dièse (#). Remarquez que des chemins relatifs au répertoire actuel sont utilisés.

- **LOCKSLEEP** — définit la durée, en secondes, qui doit s'écouler entre chaque tentative d'utilisation d'un fichier lockfile particulier par Procmail. La valeur par défaut s'élève à **8** secondes.

- **LOCKTIMEOUT** — définit la durée, en secondes, qui doit s'écouler après la modification d'un fichier lockfile avant que Procmail considère que ce fichier lockfileest trop ancien et doit être supprimé. La valeur par défaut s'élève à **1024** secondes.

- **LOGFILE** — fichier sur lequel toutes les informations et tous les messages d'erreur Procmail sont écrits.

- **MAILDIR** — définit le répertoire de travail actuel de Procmail. Si défini, tous les autres chemins Procmail seront relatifs à ce répertoire.

- **ORGMAIL** — spécifie la boîte aux lettres d'origine, ou un autre endroit où placer les messages s'ils ne peuvent pas être mis dans l'emplacement par défaut ou l'emplacement requis par la recette.

  Par défaut, une valeur de **/var/spool/mail/$LOGNAME** est utilisée.

- **SUSPEND** — définit la durée, en secondes, pendant laquelle Procmail fait une pause si une ressource importante, telle que l'espace swap, n'est pas disponible.

- **SWITCHRC** — permet à un utilisateur de spécifier un fichier externe contenant des recettes Procmail supplémentaires similaires à l'option **INCLUDERC**, sauf que la vérification de recette s'arrête sur le fichier de configuration référant et seules les recettes se trouvant sur le fichier spécifié par **SWITCHRC** sont utilisées.

- **VERBOSE** — amène Procmail à journaliser davantage d'information. Cette option est utile pour le débogage.

D'autres variables d'environnement importantes sont récupérées du shell, comme **LOGNAME**, le nom de connexion ; **HOME**, l'emplacement du répertoire personnel ; et **SHELL**, le shell par défaut.

Une explication complète de toutes les variables d'environnement et de leurs valeurs par défaut est disponible sur la page man de **procmailrc**.

13.4.2. Recettes Procmail

Les nouveaux utilisateurs trouvent fréquemment que la construction de recettes est la partie la plus difficile lors de l'apprentissage de l'utilisation de Procmail. Cette difficulté est souvent attribuée aux recettes qui font correspondre des messages en utilisant des *expressions régulières* utilisées pour spécifier des critères de correspondance de chaîne. Cependant, les expressions régulières ne sont pas très difficiles à construire, et encore moins difficiles à comprendre lorsqu'elles sont lues. En outre, la cohérence d'écriture des recettes Procmail, peu importe les expressions régulières , facilite l'apprentissage à l'aide d'exemples. Pour afficher des exemples de recettes Procmail, veuillez consulter la Section 13.4.2.5, « Exemples de recettes ».

Les recettes Procmail sont sous la forme suivante :

```
:0 [flags] [: lockfile-name ]
* [ condition_1_special-condition-character condition_1_regular_expression
]
* [ condition_2_special-condition-character condition-2_regular_expression
]
```

```
* [ condition_N_special-condition-character condition-N_regular_expression
]
        special-action-character
        action-to-perform
```

Les deux premiers caractères d'une recette Procmail sont deux points suivis d'un zéro. Divers marqueurs peuvent être placés après le zéro pour contrôler la manière par laquelle Procmail traite la recette. Les deux points suivant la section ***flags*** indiquent qu'un fichier lockfile est créé pour ce message. Si un ficier lockfile est créé, le nom peut être spécifié en remplaçant ***lockfile-name***.

Une recette peut contenir plusieurs conditions pour correspondre aux messages. S'il n'y a aucune condition, chaque message correspond à la recette. Les expressions régulières sont placées dans certaines conditions pour faciliter la correspondance de messages. Si de multiples conditions sont utilisées, elles doivent toutes correspondre pour que l'action soit appliquée. Les conditions sont vérifiées en se basant sur les marqueurs paramétrés dans la première ligne du destinataire. Des caractères spéciaux optionnels sont placés après l'astérisque (*****).

L'argument ***action-to-perform*** spécifie l'action effectuée lorsque le message correspond à l'une des conditions. Il ne peut y avoir qu'une seule action par recette. Dans de nombreux cas, le nom d'une boîte aux lettres est utilisé pour diriger les messages correspondants vers ce fichier, triant ainsi le courrier électronique. Les caractères d'action spéciaux peuvent également être utilisés avant que l'action ne soit spécifiée. Veuillez consulter la Section 13.4.2.4, « Conditions et actions spéciales » pour obtenir davantage d'informations.

13.4.2.1. Recettes de remise vs. Recettes de non-remise

L'action utilisée si la recette correspond à un message particulier détermine s'il s'agit d'une recette de remise (« *delivering* recipe ») ou d'une recette de non-remise (« *non-delivering* »). Une recette de remise contient une action qui écrit le message sur un fichier, envoie le message à un autre programme, ou transfère le message sur une autre adresse électronique. Une recette de non-remise couvre toutes les autres actions, comme un bloc imbriqué (« *nesting block* »). Un bloc imbriqué est un ensemble d'actions, entre crochets **{ }**, qui sont effectuées sur les messages correspondants aux conditions de la recette. Les blocs imbriqués peuvent être imbriqués à l'intérieur les uns des autres, offrant ainsi un meilleur contrôle pour identifier et effectuer des actions sur les messages.

Lorsque des messages correspondent à une recette de remise, Procmail effectue l'action spécifiée et arrête de comparer le message à d'autres recettes. Les messages qui correspondent aux recettes de non-remise continuent d'être comparées aux autres recettes.

13.4.2.2. Marqueurs

Les marqueurs sont essentiels pour déterminer si les conditions d'une recette sont comparées à un message et de quelle manière cela est fait. L'utilitaire **egrep** est utilisé de manière interne pour faire correspondre les conditions. Les marqueurs suivants sont couramment utilisés :

- **A** — spécifie que cette recette est uniquement utilisée si la recette précédente sans marqueur **A** ou **a** correspondait également à ce message.
- **a** — spécifie que cette recette est uniquement utilisée si la recette précédente avec un marqueur **A** ou **a** correspondait également à ce message *et* a été appliquée.
- **B** — analyse le corps du message et recherche des conditions correpondantes.
- **b** — utilise le corps dans toute action résultante, comme l'écriture du message sur un fichier ou son transfert. Il s'agit du comportement par défaut.

- **c** — génère une copie carbone du courrier électronique. Ceci est utile avec les recettes de remise, car l'action requise peut être effectuée sur le message et une copie du message peut toujours être en cours de traitement dans les fichiers **rc**.
- **D** — fait respecter la casse à la comparaison **egrep**. Par défaut, le processus de comparaison ne respecte pas la casse.
- **E** — malgré des similarités avec le marqueur **A**, les conditions de la recette sont uniquement comparées au message si la recette la précédant immédiatement sans marqeur **E** ne correspondait pas. Ceci est comparable à une action *else*.
- **e** — la recette est comparée au message uniquement si l'action spécifiée dans la recette la précédant immédiatement échoue.
- **f** — utilise le tube (« pipe ») en tant que filtre.
- **H** — analyse l'en-tête du message et recherche des conditions correspondantes. Ceci est le comportement par défaut.
- **h** — utilise l'en-tête dans une action résultante. Ceci est le comportement par défaut.
- **w** — ordonne à Procmail d'attendre que le filtre ou programme spécifié se termine, et rapporte si celui-ci a réussi ou non avant de considérer le message comme filtré.
- **W** — est identique à **w**, sauf que les messages « Échec du programme » sont supprimés.

Pour une liste détaillée de marqueurs supplémentaires, veuillez consulter la page man de **procmailrc**.

13.4.2.3. Spécifier un fichier lockfile local

Les fichiers lockfiles sont très utiles avec Procmail pour vous assurer que plusieurs processus ne tentent pas d'altérer un message simultanément. Spécifiez un lockfile local en plaçant un caractère Deux-points (**:**) après tout marqueur se trouvant sur la première ligne d'une recette. Ceci crée un fichier lockfile local basé sur le nom du fichier destinataire plus ce qui a été défini dans la variable globale d'environnement **LOCKEXT**.

Alternativement, veuillez spécifier le nom du fichier lockfile local à utiliser avec cette recette après le caractère des deux-points.

13.4.2.4. Conditions et actions spéciales

Les caractères spéciaux utilisés avant les conditions et actions des recettes Procmail modifient la manière par laquelle celles-ci sont interprétées.

Les caractères suivants peuvent être utilisés après un astérisque (*****) au début de la ligne de condition d'une recette :

- **!** — dans la ligne de condition, ce caractère inverse la condition. Ainsi, une correspondance ne se produira que si la condition ne correspond pas au message.
- **<** — vérifie si le message fait moins qu'un nombre d'octets spécifié.
- **>** — vérifie si le message fait plus qu'un nombre d'octets spécifié.

Les caractères suivants sont utilisés pour effectuer des actions spéciales :

- **!** — dans la ligne de l'action, ce caractère ordonne à Procmail de transférer le message vers les adresses électroniques spécifiées.
- **$** — fait référence à une variable précédemment définie dans le fichier **rc**. Souvent utilisé pour définir une boîte aux lettres commune à laquelle diverses recettes font référence.
- **|** — lance un programme spécifié pour traiter le message.
- **{** and **}** — permet de construire un bloc imbriqué, utilisé pour contenir des recettes supplémentaires pour appliquer des messages correspondants.

Si aucun caractère spécial n'est utilisé au début de la ligne de l'action, Procmail supposera que la ligne d'action spécifie la boîte aux lettre dans laquelle écrire le message.

13.4.2.5. Exemples de recettes

Procmail est un programme extrêmement flexible, mais cette flexibilité rend la composition de recettes Procmail difficile pour les nouveaux utilisateurs.

La meilleure manière de développer les compétences requises pour créer des conditions de recettes Procmail vient d'une bonne compréhension des expressions régulières combiné à un regard objectif sur les nombreux exemples produits par d'autres personnes. L'explication détaillée des expressions régulières est au-delà de l'étendue de cette section. La structure des recettes Procmail et des exemples de recettes Procmail utiles peuvent être trouvés sur internet. L'utilisation et l'adaptation correcte d'expressiosn régulière peuvent être dérivées de l'observation minutieuse de ces exemples de recettes. En outre, des informations d'introduction auxr règles des expressions régulières de base sont situées dans la page man de **grep(1)**.

Les simples exemples suivants démontrent la structure de base des recettes Procmail et peuvent servir de fondation pour des constructions plus complexes.

Une recette de base peut même ne pas contenir de conditions, comme illustré dans l'exemple suivant :

```
:0:
new-mail.spool
```

La première ligne spécifie qu'un lockfile local doit être créé mais ne spécifie pas de nom, Procmail utilise donc le nom du fichier destinataire et ajoute la valeur spécifiée dans la variable d'environnement **LOCKEXT**. Aucune condition n'est spécifiée, donc chaque message correspond à cette recette et est placé dans l'unique fichier spool nommé **new-mail.spool**, qui est situé dans un répertoire spécifié par la variable d'environnement **MAILDIR**. Un MUA peut ensuite afficher les messages dans ce fichier.

Une recette de base, comme celle-ci, peut être placée à la fin de tous les fichiers **rc** pour diriger les messages vers un emplacement par défaut.

L'exemple suivant faisait correspondre des messages d'une adresse électronique spécifique puis les jettait.

```
:0
* ^From: spammer@domain.com
/dev/null
```

Dans cet exemple, tout message envoyé par **spammer@domain.com** est envoyé sur le périphérique **/dev/null**, qui les supprime.

AVERTISSEMENT

Assurez-vous que les règles fonctionnent comme prévu avant d'envoyer des messages sur **/dev/null** pour une suppression permanente. Si une recette obtient des messages de manière non-intentionnelle et que ces messages disparaissent, il sera difficile de résoudre le problème de la règle.

Une meilleure solution consisterait à pointer l'action de la recette vers une boîte aux lettres spéciale, qui pourra être vérifiée de temps en temps pour chercher des faux positifs. Une fois que vous serez satisfait qu'aucun message ne sera accidentellement mis en correspondance, veuillez supprimer la boîte aux lettres et dirigez l'action pour que les messages soit envoyés dans **/dev/null**.

La recette suivante récupère le courrier électronique envoyé à partir d'une liste de diffusion particulière et le place dans un dossier spécifié.

```
:0:
* ^(From|Cc|To).*tux-lug
tuxlug
```

Tout message envoyé à partir de la liste de diffusion **tux-lug@domain.com** est automatiquement placé dans la boîte aux lettres **tuxlug** pour le MUA. Veuillez remarquer que la condition dans cet exemple correspond au message si l'adresse de la liste de diffusion se trouve sur la ligne **De** (« From »), **Cc**, ou **À** (« To »).

Veuillez consulter les nombreuses ressources Procmail en ligne, disponibles dans la Section 13.6, « Ressources supplémentaires » pour obtenir des recettes plus détaillées et plus puissantes.

13.4.2.6. Filtres du courrier indésirable

Comme il est appelé par Sendmail, Postfix, et Fetchmail lors de la réception de nouveaux courriers électroniques, Procmail peut être utilisé comme outil puissant pour combattre le courrier indésirable.

Ceci est particulièrement vrai lorsque Procmail est utilisé en conjonction avec SpamAssassin. Lorsqu'utilisées ensemble, ces deux applications peuvent rapidement identifier le courrier indésirable et le trier ou le détruire.

SpamAssassin utilise des analyses d'en-tête, des analyses de texte, des listes noires, une base de données de suivi de courrier indésirable, et une analyse de courrier indésirable bayésienne pour identifier et baliser le courrier indésirable rapidement et précisément.

NOTE

Pour utiliser **SpamAssassin**, veuillez commencer par vous assurer que le paquet spamassassin est installé sur votre système en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install spamassassin
```

Pour obtenir davantage d'informations sur l'installation de paquets avec Yum, veuillez consulter la Section 8.2.4, « Installation de paquets ».

Pour un utilisateur local, la manière la plus simple d'utiliser SpamAssassin consiste à placer la ligne suivante vers le haut du fichier **~/.procmailrc** :

```
INCLUDERC=/etc/mail/spamassassin/spamassassin-default.rc
```

Le fichier **/etc/mail/spamassassin/spamassassin-default.rc** contient une règle Procmail simple qui active SpamAssassin pour le courrier entrant. Si un courrier électronique est déterminé comme étant un courrier indésirable, son en-tête est balisé comme tel et le modèle suivant est ajouté au début de son titre :

```
*****SPAM*****
```

Le contenu du message commence par un décompte des éléments qui le qualifient comme étant un courier indésirable.

Pour archiver le courrier marqué comme indésirable, une règle similaire à la suivante peut être utilisée :

```
:0 Hw * ^X-Spam-Status: Yes spam
```

Cette règle dépose tous les messages dont l'en-tête est balisé comme courrier indésirable dans une boîte aux lettre nommée **spam**.

Comme SpamAssassin est un script Perl, il pourrait être nécessaire d'utiliser le démon binaire SpamAssassin (**spamd**) et l'application cliente (**spamc**) sur des serveurs occupés. Cependant, la configuration de SpamAssassin de cette manière, requiert un accès **root** à l'hôte.

Pour lancer le démon **spamd**, veuillez saisir la commande suivante :

```
~]# systemctl start spamassassin
```

Pour lancer le démon SpamAssassin lorsque le système est démarré, veuillez exécuter :

```
systemctl enable spamassassin.service
```

Veuillez consulter le Chapitre 9, *Gérer les services avec systemd* pour obtenir davantage d'informations sur le lancement et l'arrêt des services.

Pour configurer Procmail de manière à utiliser l'application cliente SpamAssassin au lieu du script Perl, veuillez placer la ligne suivante vers le haut du fichier **~/.procmailrc**. Pour une configuration globale, veuillez la placer dans **/etc/procmailrc** :

```
INCLUDERC=/etc/mail/spamassassin/spamassassin-spamc.rc
```

13.5. MAIL USER AGENTS (MUA)

Red Hat Enterprise Linux offre une variété de programmes de messagerie, des programmes client de messagerie graphique, comme **Evolution**, ainsi que des programme de messagerie basés texte, comme **mutt**.

Le reste de cette section concerne la sécurisation des communications entre un client et un serveur.

13.5.1. Sécuriser les communications

Les MUA populaires inclus avec Red Hat Enterprise Linux, comme **Evolution** et **Mutt** offrent des sessions de messagerie chiffrées avec SSL.

Comme tout autre service exécuté sur un réseau non chiffré, les informations importantes des courrier électroniques, comme les noms d'utilisateurs, les mots de passe et les messages entiers peuvent être interceptés et vus par les utilisateurs sur le réseau. En outre, comme les protocoles standards **POP** et **IMAP** passent des informations d'authentification non chiffrées, il est possible qu'une personne malveillante obtienne accès aux comptes des utilisateurs en récupérant les noms d'utilisateurs et mots de passe quand ils sont passés sur le réseau.

13.5.1.1. Clients de messagerie sécurisés

La plupart des MUA Linux conçus pour vérifier le courrier électronique sur les serveurs distants prennent en charge le chiffrement SSL. Pour utiliser SSL pendant la récupération du courrier, il doit être activé sur le client de messagerie et sur le serveur.

SSL est facile à activer côté client, ce qui est souvent fait simplement en cliquant sur un bouton dans la fenêtre de configuration du MUA ou via une option du fichier de configuration du MUA. Les protocoles sécurisés **IMAP** et **POP** ont des numéros de port (**993** et **995**, respectivement) utilisés par le MUA pour authentifier et télécharger des messages.

13.5.1.2. Sécurisation des communications des clients de messagerie

Offrir le chiffrement SSL aux utilisateurs **IMAP** et **POP** sur le serveur de messagerie est une simple question.

Premièrement, veuillez créer un certificat SSL. Ceci peut être fait de deux manières différentes en appliquant sur un *CA* (*Certificate Authority*) pour obtenir un certificat SSL ou en créant un certificat autosigné.

AVERTISSEMENT

Les certificats autosignés doivent uniquement être utilisés pour des tests. Tout serveur utilisé dans un environnement de production doit utiliser un certificat SSL signé par un CA.

Pour créer un certificat autosigné pour **IMAP** ou **POP**, rendez-vous dans le répertoire **/etc/pki/dovecot/**, modifiez les paramètres du certificat dans le fichier de configuration

/etc/pki/dovecot/dovecot-openssl.cnf selon vos préférences, puis saisissez les commandes suivantes en tant qu'utilisateur **root** :

```
dovecot]# rm -f certs/dovecot.pem private/dovecot.pem
dovecot]# /usr/libexec/dovecot/mkcert.sh
```

Une fois terminé, assurez-vous d'avoir les configurations suivantes dans votre fichier **/etc/dovecot/conf.d/10-ssl.conf** :

```
ssl_cert = </etc/pki/dovecot/certs/dovecot.pem
ssl_key = </etc/pki/dovecot/private/dovecot.pem
```

Veuillez exécuter la commande suivante pour redémarrer le démon **dovecot** :

```
~]# systemctl restart dovecot
```

Alternativement, la commande **stunnel** peut être utilisée comme emballage de chiffrement autour des connexions standards non-sécurisées aux services **IMAP** ou **POP**.

L'utilitaire **stunnel** utilise des bibliothèques OpenSSL externes incluses avec Red Hat Enterprise Linux pour fournir un chiffrement fort et pour protéger les connexions réseau. Il est recommandé d'appliquer sur un CA pour obtenir un certificat SSL, mais il est également possible de créer un certificat autosigné.

Veuillez consulter Utiliser stunnel dans le Guide de sécurité Red Hat Enterprise Linux 7 pour obtenir des instructions sur l'installation de **stunnel** et pour créer sa configuration de base. Pour configurer **stunnel** comme emballage pour **IMAPS** et **POP3S**, veuillez ajouter les lignes suivantes au fichier de configuration **/etc/stunnel/stunnel.conf** :

```
[pop3s]
accept  = 995
connect = 110

[imaps]
accept  = 993
connect = 143
```

Le Guide de sécurité explique également comment lancer et arrêter **stunnel**. Une fois lancé, il est possible d'utiliser un client de messagerie **IMAP** ou **POP** et de se connecter au serveur de messagerie en utilisant le chiffrement SSL.

13.6. RESSOURCES SUPPLÉMENTAIRES

Voici une liste de documents supplémentaires sur les applications de messagerie.

13.6.1. Documentation installée

- Des informations sur la configuration de Sendmail sont incluses avec les paquets sendmail et sendmail-cf.
    - **/usr/share/sendmail-cf/README** — contient des informations sur le processeur macro **m4**, les emplacements des fichiers pour Sendmail, les logiciels de messagerie pris en charge, sur les manières d'accéder aux fonctionnalités améliorées, et bien plus.

En outre, les pages man de **sendmail** et **aliases** contiennent des informations utiles traitant des diverses options Sendmail et de la configuration correcte du fichier Sendmail **/etc/mail/aliases**.

- **/usr/share/doc/postfix-*version-number*/** — contient une grande quantité d'informations sur comment configurer Postfix. Remplacez *version-number* par le numéro de version de Postfix.
- **/usr/share/doc/fetchmail-*version-number*/** — contient une liste complète des fonctionnalités de Fetchmail dans le fichier **FEATURES** ainsi qu'une **FAQ** d'introduction. Remplacez *version-number* par le numéro de version de Fetchmail.
- **/usr/share/doc/procmail-*version-number*/** — contient un fichier **README** qui fournit une vue d'ensemble de Procmail, un fichier **FEATURES** qui explore toutes les fonctionnalités du programme, et un fichier **FAQ** avec les réponses à de nombreuses questions communes sur la configuration. Remplacez *version-number* par le numéro de version de Procmail.

  Lors de l'apprentissage de Procmail et de la création de nouvelles recettes, les pages man Procmail suivantes seront indispensables :

  - **procmail** — fournit une vue d'ensemble du fonctionnement de Procmail et des étapes impliquées dans le filtrage du courrier.
  - **procmailrc** — explique le format du fichier **rc** utilisé pour créer des recettes.
  - **procmailex** — donne un certain nombre d'exemples utiles et réels de recettes Procmail.
  - **procmailsc** — explique la technique de notation pondérée utilisée par Procmail pour faire correspondre une recette particulière à un message.
  - **/usr/share/doc/spamassassin-*version-number*/** — contient une grande quantité d'informations pertinentes à SpamAssassin. Remplacez *version-number* par le numéro de version du paquet spamassassin.

13.6.2. Documentation en ligne

- Comment configurer postfix avec TLS ? — un article de Red Hat Knowledgebase qui décrit comment configurer postfix pour utiliser TLS.
- http://www.sendmail.org/ — offre une décomposition minutieuse des fonctionnalités, de la documentation, et des exemples de configuration de Sendmail.
- http://www.sendmail.com/ — contient des informations, entretiens, et articles concernant Sendmail, y compris une vue étendue des nombreuses options disponibles.
- http://www.postfix.org/ — la page d'accueil du projet Postfix contient une mine d'informations sur Postfix. La liste de diffusion est particulièrement utile pour rechercher des informations.
- http://www.fetchmail.info/fetchmail-FAQ.html — FAQ détaillée sur Fetchmail.
- http://www.procmail.org/ — page d'accueil de Procmail avec des liens vers les listes de diffusion dédiées à Procmail ainsi que divers documents FAQ.
- http://www.uwasa.fi/~ts/info/proctips.html — contient une douzaine de conseils qui permettent une utilisation bien plus facile de Procmail. Y compris des instructions sur la manière de tester les fichiers **.procmailrc** et d'utiliser la notation Procmail pour décider si une action particulière

doit être prise.

- http://www.spamassassin.org/ — site officiel du projet SpamAssassin.

13.6.3. Livres apparentés

- *Sendmail Milters: A Guide for Fighting Spam* de Bryan Costales et Marcia Flynt; Addison-Wesley — un bon guide Sendmail pouvant vous aider à personnaliser vos filtres de messagerie.
- *Sendmail* de Bryan Costales avec Eric Allman et al.; O'Reilly & Associates — une bonne référence Sendmail écrite à l'aide du créateur à l'origine de Delivermail et Sendmail.
- *Removing the Spam: Email Processing and Filtering* de Geoff Mulligan; Addison-Wesley Publishing Company — un volume examinant les diverses méthodes utilisées par les administrateurs de messagerie utilisant des outils établis, tels que Sendmail et Procmail, pour gérer les problèmes de courrier indésirable.
- *Internet Email Protocols: A Developer's Guide* de Kevin Johnson; Addison-Wesley Publishing Company — examine de manière très détaillée les protocoles de messagerie principaux et le niveau de sécurité qu'ils offrent.
- *Managing IMAP* de Dianna Mullet et Kevin Mullet; O'Reilly & Associates — détaille les étapes requises pour configurer un serveur IMAP.

CHAPITRE 14. SERVEURS DE FICHIERS ET D'IMPRESSION

Ce chapitre vous guide à travers l'installation et la configuration de **Samba**, une implémentation Open Source des protocoles *Server Message Block* (ou SMB), *common Internet file system* (ou **CIFS**), et **vsftpd**, le principal serveur FTP fourni avec Red Hat Enterprise Linux. En outre, il explique comment utiliser l'outil **Imprimer les paramètres** pour configurer les imprimantes.

14.1. SAMBA

Samba est une suite de programmes d'intéropérabilité Windows en source libre du protocole *server message block* (**SMB**). **SMB** permet à Microsoft Windows®, Linux, UNIX, et à d'autres systèmes d'exploitation d'accéder à des fichiers et à des imprimantes partagés à partir de serveurs qui prennent en charge ce protocole. L'utilisation de **SMB** par Samba leur permet d'apparaître comme serveur Windows pour les clients Windows.

NOTE

Pour utiliser **Samba**, commencez par vous assurer que le paquet samba est installé sur votre système en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install samba
```

Pour obtenir davantage d'informations sur l'installation de paquets avec Yum, veuillez consulter la Section 8.2.4, « Installation de paquets ».

14.1.1. Introduction à Samba

Samba est un composant clé qui facilite l'intégration des serveurs Linux et des Desktops dans des environnements Active Directory (AD). Peut fonctionner soit en contrôleur de domaine (NT4-style), soit en tant que membre de domaine standard (AD ou NT4-style).

Ce que Samba peut faire :

- Servir des structures de répertoires et des imprimantes à des clients Linux, UNIX, et Windows
- Assister lors de la navigation réseau (avec NetBIOS)
- Authentifier les connexions aux domaines Windows
- Fournir des résolutions de serveur de noms *Windows Internet Name Service* (**WINS**)
- Agir en tant que contrôleur de domaine de sauvegarde (PDC) NT®-style *Primary Domain Controller*
- Agir en tant que contrôleur de domaine de sauvegarde (BDC) *Backup Domain Controller* pour un PDC basé Samba
- Agir en tant que serveur membre d'un domaine Active Directory
- Joindre un serveur Windows NT/2000/2003/2008 PDC/Windows Server 2012

Ce que Samba ne peut pas faire :

- Agir en tant que BDC pour un PDC Windows (et vice-versa)

- Agir en tant que contrôleur de domaine Active Directory

14.1.2. Démons Samba et services connexes

Samba comprend trois démons (**smbd**, **nmbd**, et **winbindd**). Trois services (**smb**, **nmb**, et **winbind**) contrôlent la manière par laquelle les démons sont lancés, arrêtés, ainsi que d'autres fonctionnalités liées aux services. Ces services agissent comme scripts init différents. Chaque démon est répertorié de manière détaillée ci-dessous, ainsi que les services spécifiques qui en possèdent le contrôle.

smbd
Le démon du serveur **smbd** fournit des services d'impression et de partage de fichiers aux clients Windows. De plus, il est responsable pour l'authentification utilisateur, le verrouillage de ressources, et le partage de données via le protocole **SMB**. Les ports par défaut sur lesquels le serveur écoute le trafic **SMB** sont les ports **TCP 139** et **445**.

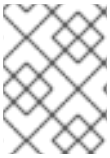
Le démon **smbd** est contrôlé par le service **smb**.

nmbd
Le démon du serveur **nmbd** comprend et répond aux requêtes de service de noms NetBIOS comme celles produites par SMB/CIFS sur des systèmes basés Windows. Ces systèmes incluent des clients Windows 95/98/ME, Windows NT, Windows 2000, Windows XP, et LanManager. Il participte également aux protocoles de navigation qui composent l'affichage du **Voisinage réseau** Windows. Le port par défaut sur lequel le serveur écoute le trafic **NMB** et le port **UDP 137**.

Le démon **nmbd** est contrôlé par le service **nmb**.

winbindd
Le service **windbind** résout les informations des utilisateurs et des groupes reçues d'un serveur exécutant Windows NT, 2000, 2003, Windows Server 2008, ou Windows Server 2012. Ceci rend les informations des utilisateurs et des groupes Windows compréhensibles par les plateformes UNIX. Cela est effectué en utilisant des appels Microsoft RPC, PAM (« *Pluggable Authentication Modules* »), et NSS (« *Name Service Switch* »). Cela permet aux utilisateurs de domaines Windows NT et d'AD (Active Directory) d'apparaître et d'opérer comme s'ils étaient des utilisateurs UNIX sur une machine UNIX. Malgré qu'il soit combiné à la distribution Samba, le service **winbind** est contrôlé séparément à partir du service **smb**.

Le démon **winbind** est contrôlé par le service **winbind** et ne requiert pas que le service **smb** soit lancé pour opérer. **winbind** est également utilisé lorsque Samba est un membre d'Active Directory, et peut aussi être utilisé sur un contrôleur de domaines Samba (pour implémenter des groupes imbriqués et une confiance interdomaines). Comme **winbind** est un service côté client utilisé pour connecter des serveurs Windows basés NT, une discussion approfondie sur **winbind** est au-delà de l'étendue de ce chapitre.



NOTE

Veuillez consulter la Section 14.1.9, « Programmes de distribution Samba » pour voir une liste des utilitaires inclus dans la distribution Samba.

14.1.3. Se connecter à un partage Samba

Vous pouvez utiliser la commande **Nautilus** ou l'utilitaire de ligne de commande pour vous connecter aux partages de Samba disponibles.

Procédure 14.1. Se connecter à un partage Samba via Nautilus

1. Pour afficher une liste des groupes de travail Samba et des domaines sur votre réseau, veuillez sélectionner **Places** → **Réseau** dans le panneau GNOME, puis sélectionner le réseau souhaité. Vous pouvez également saisir **smb:** dans la barre **Fichier** → **Ouvrir l'emplacement** de **Nautilus**.

   Comme indiqué dans la Figure 14.1, « Groupes de travail SMB dans Nautilus », une icône s'affiche pour chaque groupe de travail ou domaine **SMB** sur le réseau.

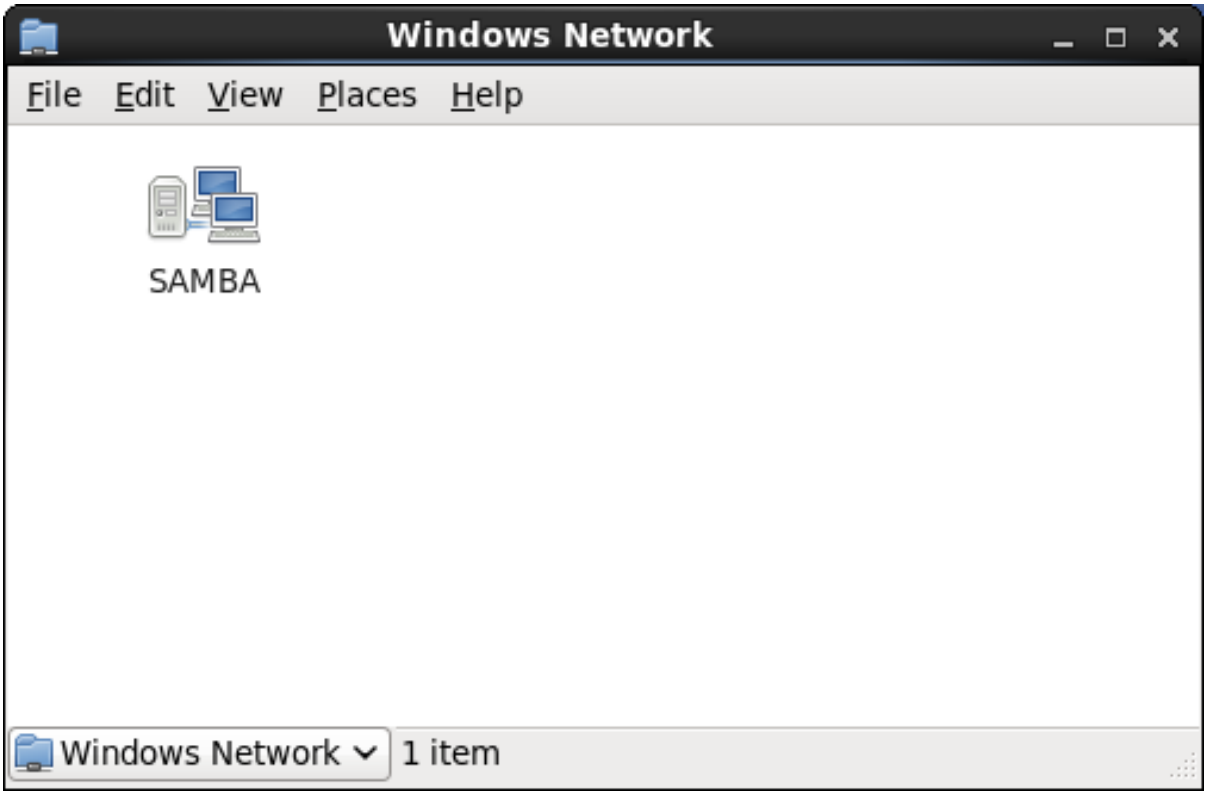


Figure 14.1. Groupes de travail SMB dans Nautilus

2. Faites un double clic sur l'icône de groupe de travail ou de domaine pour afficher la liste des ordinateurs dans ce groupe de travail ou dans ce domaine.

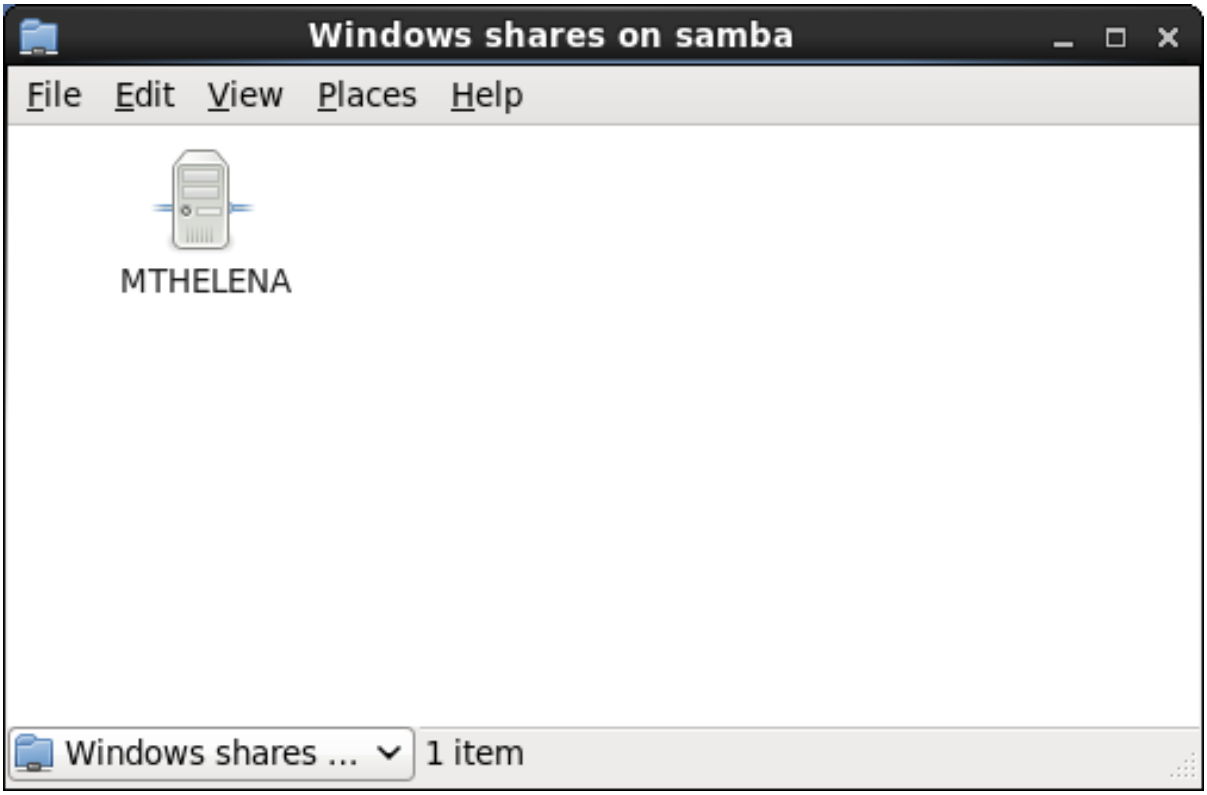


Figure 14.2. Machines SMB dans Nautilus

3. Comme indiqué dans la Figure 14.2, « Machines SMB dans Nautilus », une icône existe pour chaque machine dans le groupe de travail. Faites un double clic sur une icône pour afficher les partages Samba sur la machine. Si une combinaison nom d'utilisateur et mot de passe est requise, celle-ci vous sera demandée.

   Alternativement, vous pouvez également spécifier le serveur Samba et le nom du partage dans la barre **Location:** de **Nautilus** en utilisant la syntaxe suivante (remplacez *servername* et *sharename* par les valeurs appropriées) :

   ```
   smb://servername/sharename
   ```

Procédure 14.2. Se connecter à un partage Samba par l'interface en ligne de commande

1. Pour se connecter à un partage Samba à partir d'une invite de shell, veuillez saisir la commande suivante :

   ```
   ~]$ smbclient //hostname/sharename -U username
   ```

   Remplacez *hostname* par le nom d'hôte ou par l'adresse **IP** du serveur Samba auquel vous souhaitez vous connecter, *sharename* par le nom du répertoire partagé que vous souhaitez parcourir, et *username* par le nom d'utilisateur Samba du système. Saisissez le mot de passe correct ou appuyez sur **Entrée** si aucun mot de passe n'est requis pour l'utilisateur.

   Si vous voyez l'invite **smb:\>**, cela signifie que vous vous êtes connecté. Une fois connecté, veuillez saisir **help** pour obtenir la liste des commandes. Si vous souhaitez parcourir le contenu de votre répertoire personnel, remplacez *sharename* par votre nom d'utilisateur. Si le commutateur **-U** n'est pas utilisé, le nom d'utilisateur de l'utilisateur actuel sera transmis au serveur Samba.

2. Pour quitter **smbclient**, saisissez **exit** à l'invite **smb:\>**.

14.1.4. Monter le partage

Parfois, il peut être utile de monter un partage Samba sur un répertoire afin que les fichiers puissent être traités comme s'ils faisaient partie du système de fichiers local.

Pour monter un partage Samba sur un répertoire, veuillez créer un répertoire pour le monter dessus (s'il n'existe pas déjà), et exécutez la commande suivante en tant qu'utilisateur **root** :

```
mount -t cifs //nom du serveur/nom partage /mnt/point/ -o username=nom
d'utilisateurmot de passe =mot de passe
```

Cette commande mont *nom partage* à partir de *nom du serveur* dans le répertoire local */mnt/point/*.

For more information about mounting a samba share, see the mount.cifs(8) manual page.

NOTE

L'utilitaire **mount.cifs** est un RPM séparé (indépendant de Samba). Pour utiliser **mount.cifs**, commencez par vous assurer que le paquet cifs-utils soit installé sur votre système en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install cifs-utils
```

Pour obtenir davantage d'informations sur l'installation de paquets avec Yum, veuillez consulter la Section 8.2.4, « Installation de paquets ».

Note that the cifs-utils package also contains the **cifs.upcall** binary called by the kernel in order to perform kerberized CIFS mounts. For more information on **cifs.upcall**, see the cifs.upcall(8) manual page.

AVERTISSEMENT

Certains serveur CIFS requièremt des mots de passe en texte brut pour l'authentification. La prise en charge de l'authentification des mots passe en texte brut peut être activée en utilisant la commande suivante en tant qu'utilisateur **root** :

```
~]# echo 0x37 > /proc/fs/cifs/SecurityFlags
```

AVERTISSEMENT : Cette opération peut exposer les mots de passe en supprimant le chiffrement de mot de passe.

14.1.5. Configurer un serveur Samba

Le fichier de configuration par défaut (**/etc/samba/smb.conf**) permet aux utilisateurs de voir leur répertoire personnel en tant que partage Samba. Il partage également toutes les imprimantes configurées pour le système en tant qu'imprimantes partagées Samba. Vous pouvez attacher une imprimante au système et l'utiliser à partir de machines Windows sur votre réseau.

14.1.5.1. Configuration graphique

Pour configurer Samba en utilisant une interface graphique, veuillez utiliser l'une des interfaces utilisateur graphique Samba disponibles. Une liste des GUI se trouve dans http://www.samba.org/samba/GUI/.

14.1.5.2. Configuration en ligne de commande

Samba utilise **/etc/samba/smb.conf** comme fichier de configuration. Si vous modifiez cela, les changements n'entreront en vigueur que lorsque vous redémarrez le démon Samba avec la commande suivante, en tant qu'utilisateur **root** :

```
~]# systemctl restart smb.service
```

Pour spécifier le groupe de travail Windows et pour faire une brève description du serveur Samba, veuillez modifier les lignes suivantes dans votre fichier **/etc/samba/smb.conf** :

```
workgroup = WORKGROUPNAME
server string = BRIEF COMMENT ABOUT SERVER
```

Remplacez *WORKGROUPNAME* par le nom du groupe de travail Windows auquel cette machine doit appartenir. *BRIEF COMMENT ABOUT SERVER* est optionnel et est utilisé comme commentaire Windows sur le système Samba.

Pour créer un répertoire de partage Samba sur votre système Linux, veuillez ajouter la section suivante à votre fichier **/etc/samba/smb.conf** (après l'avoir modifiée pour refléter vos besoins et ceux de votre système) :

Exemple 14.1. Exemple de configuration de serveur Samba

```
[nom partage]
commentaire = Insérer un commentaire ici
chemin = /home/share/
utilisateurs valides = tfox carole
écriture = yes
créer masque = 0765
```

L'exemple ci-dessus permet aux utilisateurs **tfox** et **carole** de lire et d'écrire sur le répertoire **/home/share/**, situé sur le serveur Samba, à partir d'un client Samba.

14.1.5.3. Mots de passe chiffrés

Les mots de passe chiffrés sont activés par défaut car il est plus sûr de les utiliser. Pour créer un utilisateur avec un mot de passe chiffré, veuillez utiliser la commande **smbpasswd** :

```
smbpasswd -a username
```

14.1.6. Lancer et arrêter Samba

Pour lancer un serveur Samba, veuillez saisir la commande suivante dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl start smb.service
```

IMPORTANT

Pour paramétrer un serveur membre d'un domaine, vous devez tout d'abord joindre le domaine ou Active Directory en utilisant la commande **net join** *avant* de lancer le service **smb**. Il est également recommandé d'exécuter **winbind** avant **smbd**.

Pour arrêter le serveur, veuillez saisir la commande suivante dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl stop smb.service
```

L'option **restart** est une façon rapide d'arrêter, puis de redémarrer Samba. Cette manière est la plus efficace pour que les changements de configuration puissent entrer en vigueur après avoir modifié le fichier de configuration de Samba. Remarquez que l'option de redémarrage lance le démon même s'il n'était pas exécuté à l'origine.

Pour redémarrer le serveur, veuillez saisir la commande suivante dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl restart smb.service
```

L'option **condrestart** (redémarrage conditionnel, « *conditional restart* ») lance **smb** à condition d'être actuellement en cours d'exécution. Cette option est utile pour les scripts, car elle ne lance pas le démon s'il n'est pas en cours d'exécution.

NOTE

Lorsque le fichier **/etc/samba/smb.conf** est modifié, Samba le recharge automatiquement après quelques minutes. Exécuter un redémarrage manuel (« **restart** ») ou un rechargement manuel (« **reload** ») est tout autant efficace.

Pour redémarrer le serveur de manière conditionnelle, veuillez saisir la commande suivante dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl try-restart smb.service
```

Un rechargement manuel du fichier **/etc/samba/smb.conf** peut être utile en cas d'échec du rechargement automatique effectué par le service **smb**. Pour vous assurer que le fichier de configuration du serveur Samba soit rechargé sans redémarrer le service, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl reload smb.service
```

Par défaut, le service **smb** *n'est pas* lancé automatiquement pendant l'initialisation. Pour configurer Samba pour qu'il soit lancé pendant l'initialisation, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl enable smb.service
```

Veuillez consulter le Chapitre 9, *Gérer les services avec systemd* pour obtenir davantage d'informations sur cet outil.

14.1.7. Mode de sécurité de Samba

Il y a seulement deux types de modes de sécurité pour Samba, au *niveau du partage* et au *niveau utilisateur*, collectivement connus en tant que *niveaux de sécurité*. La sécurité au niveau du partage est obsolète et a été suprimée de Samba. Les configurations contenant ce mode doivent être migrées pour utiliser la sécurité niveau utilisateur. La sécurité niveau utilisateur peut être implémentée d'une des trois manières. Les différentes façons d'appliquer un niveau de sécurité s'appellent les *modes de sécurité*

14.1.7.1. Sécurité Niveau utilisateur

La sécurité niveau utilisateur est la valeur par défaut et est recommandée avec Samba. Même si la directive ***security = user*** n'est pas répertoriée dans le fichier **/etc/samba/smb.conf**, elle sera utilisée par Samba. Si le serveur accepte le nom d'utilisateur et mot de passe du client, le client peut alors monter plusieurs partages sans spécifier un mot de passe pour chaque instance. Samba peut également accepter des demandes de noms et mots de passe basés sur une session utilisateur. Le client maintient plusieurs contextes d'authentification en utilisant un UID unique pour chaque ouverture de session.

Dans le fichier **/etc/samba/smb.conf**, la directive ***security = user*** qui détermine la sécurité niveau utilisateur correspond à :

```
[GLOBAL]
...
security = user
...
```

Partages d'invités Samba
Comme expliqué ci-dessus, le mode de sécurité niveau partage est déprécié. Pour configurer un partage d'invité Samba sans utiliser le paramètre ***security = share***, suivre la procédure ci-dessous :

Procédure 14.3. Configuration des partages d'invités Samba

1. Créer un fichier de mise en correspondance des noms d'utilisateur, dans cet exemple, **/etc/samba/smbusers**, et y ajouter la ligne suivante :

   ```
   nobody = guest
   ```

2. Ajouter la directive suivante à la section principale du fichier **/etc/samba/smb.conf**. Ne pas utiliser la directive ***valid users***.

   ```
   [GLOBAL]
   ...
   security = user
   map to guest = Bad User
   username map = /etc/samba/smbusers
   ...
   ```

   La directive ***username map*** vous donne un chemin vers le fichier de mappage des noms utilisateurs dans l'étape précédente.

3. Ajouter la directive suivante à la section partages dans le fichier **/ect/samba/smb.conf**. Ne pas utiliser la directive ***valid users***.

```
[SHARE]
...
guest ok = yes
...
```

Les sections suivantes décrivent d'autres implémentations de sécurité niveau utilisateur.

Mode de sécurité domaine (sécurité niveau utilisateur)
En mode de sécurité de domaine, le serveur Samba dispose d'un compte machine (compte de confiance de sécurité de domaine) et fait que toutes les demandes d'authentification passent par les contrôleurs de domaine. Le serveur Samba est transformé en serveur membre de domaine selon les directives suivantes du fichier **/etc/samba/smb.conf** :

```
[GLOBAL]
...
security = domain
workgroup = MARKETING
...
```

Mode de sécurité Active Directory (sécurité niveau utilisateur)
Si vous avez un environnement Active Directory, il est possible de rejoindre le domaine en tant que membre Active Directory natif. Même si une politique de sécurité restreint l'utilisation de protocoles d'authentification NT-compatible, le serveur Samba peut rejoindre une ADS à l'aide de Kerberos. Samba en mode membre Active Directory peut accepter des tickets Kerberos.

Dans le fichier **/etc/samba/smb.conf**, les directives suivantes font de Samba un membre d'Active Directory.

```
[GLOBAL]
...
security = ADS
realm = EXAMPLE.COM
password server = kerberos.example.com
...
```

14.1.7.2. Sécurité Niveau partage

Avec la sécurité niveau partage, le serveur accepte uniquement un mot de passe sans nom d'utilisateur explicite de la part du client. Le serveur s'attend à un mot de passe pour chaque partage, indépendamment du nom d'utilisateur. Des rapports récents nous montrent que les clients de Microsoft Windows ont des problèmes de compatibilité avec les serveurs de sécurité au niveau du partage. Ce mode est obsolète et a été retiré de Samba. Les configurations contenant ***security = share*** doivent être mises à jour pour utiliser la sécurité au niveau utilisateur. Suivez les étapes de Procédure 14.3, « Configuration des partages d'invités Samba » pour éviter d'utiliser la directive ***security = share***.

14.1.8. Navigation réseau Samba

La *Navigation réseau* permet aux serveurs Windows et Samba d'apparaître dans la fenêtre **Voisinage réseau**. Dans la fenêtre du **Voisinage réseau**, des icônes représentent des serveurs et si ouverts, les partages et imprimantes disponibles du serveur sont affichés.

Les capacités de navigation réseau requièrent l'utilisation de NetBIOS sur **TCP**/**IP**. La mise en réseau basée NetBIOS utilise une messagerie de diffusion (**UDP**) pour accomplir la gestion de liste de parcours. Sans NetBIOS et WINS comme méthode principale de résolution de nom d'hôte **TCP**/**IP**, d'autres méthodes comme les fichiers statiques (**/etc/hosts**) ou **DNS** devront être utilisées.

Un explorateur principal de domaines assemble les listes de parcours des explorateurs principaux locaux sur tous les sous-réseaux afin qu'une navigation puisse être effectuée entre groupes de travail et sous-réseaux. Pour son propre réseau, l'explorateur principal de domaines devrait préférablement être l'explorateur principal local.

14.1.8.1. Exploration de domaines

Par défaut, un PDC de serveur Windows d'un domaine est également l'explorateur principal de domaines de ce domaine. Un serveur Samba *ne doit pas* être paramétré comme serveur principal de domaines dans ce type de situation.

Pour les sous-réseaux qui n'incluent pas le PDC du serveur Windows, un serveur Samba peut être implémenté en tant qu'explorateur principal local. La configuration du fichier **/etc/samba/smb.conf** pour un explorateur principal local (ou sans aucune exploration) dans un environnement de contrôleur de domaine est la même chose qu'une configuration de groupe de travail (veuillez consulter Section 14.1.5, « Configurer un serveur Samba »).

14.1.8.2. WINS (« Windows Internet Name Server »)

Un serveur Samba ou Windows NT peut fonctionner en tant que serveur WINS. Lorsqu'un serveur WINS est utilisé avec NetBIOS activé, les monodiffusions UDP peuvent être acheminées, ce qui permet la résolution de noms à travers les réseaux. Sans un serveur WINS, la diffusion UDP est limitée au sous-réseau local et ne peut donc pas être acheminée vers d'autres sous-réseaux, groupes de travail, ou domaines. Si la réplication WINS est nécessaire, n'utilisez pas Samba comme serveur WINS principal, car actuellement, Samba ne prend pas en charge la réplication WINS.

Dans un environnement Samba et serveur NT/2000/2003/2008 mélangé, il est recommandé d'utiliser des capacités Microsoft WINS. Dans un environnement Samba uniquement, il est recommandé d'utiliser *un seul* serveur Samba pour WINS.

Ci-dessous figure un exemple du fichier **/etc/samba/smb.conf**, dans lequel le serveur Samba sert de serveur WINS :

Exemple 14.2. Exemple de configuration de serveur WINS

```
[global]
wins support = yes
```

NOTE

Tous les serveurs (y compris Samba) doivent se connecter à un serveur WINS pour résoudre les noms NetBIOS. Sans WINS, la navigation n'aura lieu que sur le sous-réseau local. De plus, même si une liste globale du domaine peut être obtenue, les hôtes ne pourront pas être résolus pour le client sans WINS.

14.1.9. Programmes de distribution Samba

net

```
net <protocol> <function> <misc_options> <target_options>
```

L'utilitaire **net** est similaire à l'utilitaire **net** utilisé pour Windows et MS-DOS. Le premier argument est utilisé pour spécifier le protocole à utiliser lors de l'exécution d'une commande. L'option ***protocol*** peut être **ads**, **rap**, ou **rpc** pour spécifier le type de connexion serveur. Active Directory utilise **ads**, Win9x/NT3 utilise **rap** et Windows NT4/2000/2003/2008 utilise **rpc**. Si le protocole est omis, **net** tente de le déterminer automatiquement.

L'exemple suivant affiche une liste des partages disponibles pour un hôte nommé **wakko** :

```
~]$ net -l share -S wakko
Password:
Enumerating shared resources (exports) on remote server:
Share name   Type     Description
----------   ----     -----------
data         Disk     Wakko data share
tmp          Disk     Wakko tmp share
IPC$         IPC      IPC Service (Samba Server)
ADMIN$       IPC      IPC Service (Samba Server)
```

L'exemple suivant affiche une liste d'utilisateurs Samba pour un hôte nommé **wakko** :

```
~]$ net -l user -S wakko
root password:
User name             Comment
-----------------------------
andriusb              Documentation
joe                   Marketing
lisa                  Sales
```

nmblookup

```
nmblookup <options> <netbios_name>
```

Le programme **nmblookup** résout les noms NetBIOS en adresses **IP**. Le programme diffuse sa requête sur le sous-réseau local jusqu'à ce que la machine cible réponde.

L'exemple suivant affiche l'adresse **IP** du nom NetBIOS **trek** :

```
~]$ nmblookup trek
querying trek on 10.1.59.255
10.1.56.45 trek<00>
```

pdbedit

```
pdbedit <options>
```

Le programme **pdbedit** gère les comptes situés dans la base de données SAM. Tous les serveurs principaux sont pris en charge, y compris **smbpasswd**, LDAP, et la bibliothèque de base de données tdb.

Ci-dessous figurent des exemples d'ajout, de suppression, et d'écoute d'utilisateurs :

```
~]$ pdbedit -a kristin
new password:
retype new password:
Unix username:        kristin
NT username:
Account Flags:        [U          ]
User SID:             S-1-5-21-1210235352-3804200048-1474496110-2012
Primary Group SID:    S-1-5-21-1210235352-3804200048-1474496110-2077
Full Name: Home Directory:       \\wakko\kristin
HomeDir Drive:
Logon Script:
Profile Path:         \\wakko\kristin\profile
Domain:               WAKKO
Account desc:
Workstations: Munged
dial:
Logon time:           0
Logoff time:          Mon, 18 Jan 2038 22:14:07 GMT
Kickoff time:         Mon, 18 Jan 2038 22:14:07 GMT
Password last set:    Thu, 29 Jan 2004 08:29:28
GMT Password can change:  Thu, 29 Jan 2004 08:29:28 GMT
Password must change: Mon, 18 Jan 2038 22:14:07 GMT
~]$ pdbedit -v -L kristin
Unix username:        kristin
NT username:
Account Flags:        [U          ]
User SID:             S-1-5-21-1210235352-3804200048-1474496110-2012
Primary Group SID:    S-1-5-21-1210235352-3804200048-1474496110-2077
Full Name:
Home Directory:       \\wakko\kristin
HomeDir Drive:
Logon Script:
Profile Path:         \\wakko\kristin\profile
Domain:               WAKKO
Account desc:
Workstations: Munged
dial:
Logon time:           0
Logoff time:          Mon, 18 Jan 2038 22:14:07 GMT
Kickoff time:         Mon, 18 Jan 2038 22:14:07 GMT
Password last set:    Thu, 29 Jan 2004 08:29:28 GMT
Password can change:  Thu, 29 Jan 2004 08:29:28 GMT
Password must change: Mon, 18 Jan 2038 22:14:07 GMT
~]$ pdbedit -L
andriusb:505:
joe:503:
lisa:504:
kristin:506:
~]$ pdbedit -x joe
~]$ pdbedit -L
andriusb:505: lisa:504: kristin:506:
```

rpcclient

```
rpcclient <server> <options>
```

Le programme **rpcclient** exécute des commandes administratives en utilisant des RPC Microsoft, qui fournissent accès aux interfaces utilisateur graphique (GUI) d'administration Windows pour la gestion des systèmes. Ce programme est souvent utilisé par des utilisateurs de niveau avancé, qui comprennent parfaitement la complexité des RPC Microsoft.

smbcacls

```
smbcacls <//server/share> <filename> <options>
```

Le programme **smbcacls** modifie les ACL Windows sur les fichiers et répertoires partagés par un serveur Samba ou un serveur Windows.

smbclient

```
smbclient <//server/share> <password> <options>
```

Le programme **smbclient** est un client UNIX polyvalent offrant une fonctionnalité similaire à **ftp**.

smbcontrol

```
smbcontrol -i <options>
```

```
smbcontrol <options> <destination> <messagetype> <parameters>
```

Le programme **smbcontrol** envoie des messages de contrôle aux démons en cours d'exécution **smbd**, **nmbd**, ou **winbindd**. L'exécution de **smbcontrol -i** exécute des commandes de manière interactive jusqu'à ce qu'une ligne blanche ou que le caractère *'q'* soit saisi(e).

smbpasswd

```
smbpasswd <options> <username> <password>
```

Le programme **smbpasswd** gère des mots de passe chiffrés. Ce programme peut être utilisé par un super-utilisateur pour modifier le mot de passe de tout autre utilisateur, ainsi que par un utilisateur ordinaire pour qu'il puisse modifier son propre mot de passe Samba.

smbspool

```
smbspool <job> <user> <title> <copies> <options> <filename>
```

Le programme **smbspool** est une interface d'impression compatible avec CUPS sur Samba. Malgré sa conception destinée à une utilisation avec des imprimantes CUPS, **smbspool** peut également fonctionner avec des imprimantes n'utilisant pas CUPS.

smbstatus

```
smbstatus <options>
```

Le programme **smbstatus** affiche le statut des connexions actuelles à un serveur Samba.

smbtar

```
smbtar <options>
```

Le programme **smbcacls** effectue des copies de sauvegarde et des restaurations de fichiers et répertoires de partage basés Windows sur une bande d'archive locale. Malgré des similarités avec la commande **tar**, les deux ne sont pas compatibles.

testparm

```
testparm <options> <filename> <hostname IP_address>
```

Le programme **testparm** vérifie la syntaxe du fichier **/etc/samba/smb.conf**. Si votre fichier **smb.conf** ne se trouve pas dans l'emplacement par défaut (**/etc/samba/smb.conf**), vous n'aurez pas besoin de spécifier l'emplacement. La spécification du nom d'hôte et de l'adresse **IP** sur le programme **testparm** vérifie si les fichiers **hosts.allow** et **host.deny** sont correctement configurés. Le programme **testparm** affiche également un résumé du fichier **smb.conf** et le rôle du serveur (stand-alone, domain, etc.), après avoir effectué des tests. Ceci est utile lors du débogage car les commentaires sont exclus et les informations sont présentées de façon concise pour permettre leur lecture à des administrateurs expérimentés. Exemple :

```
~]$ testparm
Load smb config files from /etc/samba/smb.conf
Processing section "[homes]"
Processing section "[printers]"
Processing section "[tmp]"
Processing section "[html]"
Loaded services file OK.
Server role: ROLE_STANDALONE
Press enter to see a dump of your service definitions
<enter>
# Global parameters
[global]
 workgroup = MYGROUP
 server string = Samba Server
 security = SHARE
 log file = /var/log/samba/%m.log
 max log size = 50
 socket options = TCP_NODELAY SO_RCVBUF=8192 SO_SNDBUF=8192
 dns proxy = no
[homes]
 comment = Home Directories
 read only = no
 browseable = no
[printers]
 comment = All Printers
 path = /var/spool/samba
 printable = yes
 browseable = no
[tmp]
 comment = Wakko tmp
 path = /tmp
 guest only = yes
[html]
 comment = Wakko www
 path = /var/www/html
 force user = andriusb
```

```
force group = users
read only = no
guest only = yes
```

wbinfo

```
wbinfo <options>
```

Le programme **wbinfo** affiche des informations du démon **winbindd**. Le démon **winbindd** doit être en cours d'exécution pour que **wbinfo** fonctionne.

14.1.10. Ressources supplémentaires

Les sections suivantes permettent d'explorer Samba avec davantage de détails.

Documentation installée

- **/usr/share/doc/samba-<*version-number*->/** — comprend tous les fichiers supplémentaires inclus avec la distribution de Samba. Ceux-ci incluent tous les scripts d'aide, tous les exemples de fichiers de configuration, et toute la documentation.
- Veuillez consulter les pages de manuel suivantes pour obtenir des informations détaillées sur les fonctionnalités de **Samba** :
  - smb.conf(5)
  - samba(7)
  - smbd(8)
  - nmbd(8)
  - winbindd(8)

Sites Web utiles

- http://www.samba.org/ — page d'accueil de la distribution de Samba ainsi que de toute la documentation créée par l'équipe de développement Samba. De nombreuses ressources sont disponibles sous les formats HTML et PDF, tandis que d'autres sont uniquement disponibles à l'achat. Même si un grand nombre de ces liens ne sont pas spécifiques à Red Hat Enterprise Linux, certains concepts peuvent s'appliquer.
- https://wiki.samba.org/index.php/User_Documentation — documentation officielle de Samba 4.x
- http://samba.org/samba/archives.html — listes actives de courriers électroniques de la communauté Samba. L'activation du mode de synthèse (« digest mode ») est recommandé à cause du niveau élevé d'activité de la liste.
- Samba newsgroups — des groupes d'informations en threads de Samba, par exemple www.gmane.org, qui utilisent le protocole **NNTP** sont également disponibles. Cela est une alternative à la réception de courriers électroniques de la liste de diffusion.

14.2. FTP

Le protocole de transfert de fichiers (*File Transfer Protocol*, **FTP**) est l'un des protocoles les plus anciens

et des plus couramment utilisés sur Internet de nos jours. Son but est de permettre un transfert fiable entre ordinateurs hôtes sur un réseau sans que l'utilisateur ait besoin de se connecter directement à l'hôte distant ou ne possède de connaissances sur l'utilisation d'un système distant. Il permet aux utilisateurs d'accéder à des fichiers sur des systèmes distants en utilisant un ensemble standard de commandes simples.

Cette section énonce les bases du protocole **FTP** et présente ***vsftpd***, qui est le serveur **FTP** préféré sur Red Hat Enterprise Linux.

14.2.1. Le protocole de transfert de fichiers

FTP utilise une architecture client-serveur pour transférer des fichiers en utilisant le protocole réseau **TCP**. Comme **FTP** est une protocole relativement ancien, l'authentification du nom d'utilisateur et du mot de passe n'est pas chiffrée. Pour cela, FTP est considéré comme un protocole non sécurisé et ne doit pas être utilisé sauf si absolument nécessaire. Cepedant, **FTP** est si répandu sur Internet qu'il est souvent requis pour partager des fichiers avec le public. Ainsi, les administrateurs doivent avoir connaissanece des caractéristiques uniques de **FTP**.

Cette section décrit comment configurer **vsftpd** pour établir des connexions sécurisées par **TLS** et comment sécuriser un serveur **FTP** à l'aide de **SELinux**. Un bon protocole de substitution de **FTP** est **sftp**, de la suite d'outils **OpenSSH**. Pour obtenir des informations sur la configuration d'**OpenSSH** et sur le protocole **SSH** en général, veuillez consulter le Chapitre 10, *OpenSSH*.

Contrairement à la plupart des protocoles utilisés sur internet, **FTP** requiert de multiples ports réseau pour fonctionner correctement. Lorsqu'une application cliente **FTP** initie une connexion vers un serveur **FTP**, celle-ci ouvre le port **21** sur le serveur — qui est également appelé *port des commandes*. Ce port est utilisé pour exécuter toutes les commandes sur le serveur. Toutes les données requises du serveur sont retournées vers le client via un *port de données*. Le numéro de port pour les connexions de données et la manière par laquelle celles-ci sont initialisées dépendent de si le client requiert les données en mode *actif* ou *passif*.

Ce qui suit définit les modes :

mode actif

Le mode actif est la méthode d'origine utilisée par le protocole **FTP** pour transférer des données vers l'application cliente. Lorsqu'un transfert de données en mode actif est initié par le client **FTP**, le serveur ouvre une connexion sur le port **20** du serveur pour l'adresse **IP** ainsi qu'un port aléatoire, non privilégié (supérieur à **1024**), spécifié par le client. Cet arrangement signifie que la machine cliente doit être autorisée à accepter des connexions sur tout port supérieur à **1024**. Avec l'augmentation des réseaux non sécurisés, comme internet, l'utilisation de pare-feux pour protéger les machines clientes s'est répandue. Comme les pare-feux du côté client refusent souvent des connexions entrantes en provenance de serveurs **FTP** en mode actif, un mode passif a été conçu.

mode passif

Le mode passif, tout comme le mode actif, est initié par l'application cliente **FTP**. Lorsque des données sont requises du serveur, le client **FTP** indique qu'il souhaite accéder aux données en mode passif et le serveur fournit l'adresse **IP** ainsi qu'un port aléatoire, non-privilégié (supérieur à **1024**) sur le serveur. Le client se connecte ensuite à ce port sur le serveur pour télécharger les informations requises.

Même si le mode passif résout des problèmes d'interférences du pare-feu côté client avec les connexions de données, il peut également compliquer l'administration du pare-feu côté serveur. Vous pouvez réduire le nombre de ports ouverts sur un serveur en limitant la plage de ports non-privilégiés

sur le serveur **FTP**. Cela simplifie également le processus de configuration des règles du pare-feu pour le serveur.

14.2.2. Serveur vsftpd

Le démon **vsftpd** (*Very Secure FTP Daemon*) a été conçu dès le début pour être rapide, stable, et surtout sécurisé. **vsftpd** est le seul serveur **FTP** autonome distribué avec Red Hat Enterprise Linux, grâce à sa capacité de gestion efficace et sécurisée de grands nombres de connexion.

Le modèle de sécurité utilisé par **vsftpd** offre trois aspects principaux :

- *Une importante séparation entre les processus privilégiés et non-privilégiés* — les processus séparés gèrent différentes tâches, et chacun de ces processus est exécuté avec les privilèges minimaux requis pour la tâche.
- *Les tâches qui requièrent des privilèges élevés sont gérées par des processus avec le minimum des privilèges nécessaires* — en tirant profit des compatibilités trouvées dans la bibliothèque **libcap**, les tâches qui requièrent normalement la totalité des privilèges root peuvent être exécutées de manière plus sécurisée à partir d'un processus moins privilégié.
- *La plupart des processus sont exécutés dans une prison **chroot*** — lorsque c'est possible, les processus sont « change-rooted » sur le répertoire en cours de partage ; ce répertoire est ensuite considéré comme une prison **chroot**. Par exemple, si le répertoire **/var/ftp/** est le répertoire principal partagé, **vsftpd** réassigne **/var/ftp/** au nouveau répertoire root, nommé **/**. Ceci prévient toute activité pirate potentielle d'une personne malveillante sur tous les répertoires qui ne sont pas contenus dans le nouveau répertoire root.

L'utilisation de ces pratiques de sécurité a l'effet suivant sur la manière dont **vsftpd** traite les requêtes :

- *Le processus parent est exécuté avec le minimum de privilèges requis* — le processus parent calcule dynamiquement le niveau de privilèges requis pour minimiser le niveau de risques. Les processus enfants gèrent les interactions directes avec les clients **FTP** et sont exécutés avec aussi peu de privilèges que possible.
- *Toutes les opérations nécessitant une élévation de privilèges sont gérées par un petit processus parent* — tout comme avec le **serveur HTTP Apache**, **vsftpd** lance des processus enfants non privilégiés pour gérer les connexions entrantes. Ceci permet au processsus parent privilégié d'être aussi petit que possible et de gérer un nombre de tâches relativement faible.
- *Le processus parent se méfie de toutes les requêtes des processus enfants non privilégiés* — Les communications avec les processus enfants sont reçues via un socket, et la validité des information en provenance d'un processus enfant est vérifiée avant qu'une action ne soit déclenchée.
- *La plupart des interactions avec des clients **FTP** sont gérées par des processus enfants non privilégiés dans une prison **chroot*** — comme ces processus enfants ne sont pas privilégiés et ont uniquement accès au répertoire partagé, tout processus ayant échoué n'autorisera une personne malveillante à n'accéder qu'aux fichiers partagés.

14.2.2.1. Démarrage et arrêt de vsftpd

Pour démarrer le service **vsftpd** dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl start vsftpd.service
```

Pour arrêter le service dans la session actuelle, saisissez ce qui suit en tant qu'utilisateur **root** :

```
~]# systemctl stop vsftpd.service
```

Pour redémarrer le service **vsftpd**, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl restart vsftpd.service
```

Cette commande arrête puis démarre immédiatement le service **vsftpd**, ce qui est la manière la plus efficace de faire en sorte que les changements entrent en vigueur après avoir modifié le fichier de configuration de ce serveur **FTP**. De manière alternative, vous pouvez utiliser la commande suivante pour redémarrer le service **vsftpd**, uniquement s'il est déjà en cours d'exécution :

```
~]# systemctl try-restart vsftpd.service
```

Par défaut, le service **vsftpd** *n'est pas* lancé automatiquement pendant l'initialisation. Pour configurer le service **vsftpd** pour qu'il soit lancé pendant le démarrage, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
~]# systemctl enable vsftpd.service
Created symlink from /etc/systemd/system/multi-
user.target.wants/vsftpd.service to
/usr/lib/systemd/system/vsftpd.service.
```

Pour obtenir des informations supplémentaires sur la gestion des services système sur Red Hat Enterprise Linux 7, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

14.2.2.2. Lancer de multiples copies de vsftpd

De temps à autres, un ordinateur est utilisé pour servir de multiples domaines **FTP**. Cette technique est nommée *multihoming* (multiréseaux). Une manière d'effectuer un multihome en utilisant **vsftpd** consiste à exécuter de multiples copies du démon, chacune avec son propre fichier de configuration.

Pour faire ceci, commencez par assigner toutes les adresses **IP** à des périphériques réseau ou à des alias de périphériques réseau sur le système. Pour obtenir davantage d'informations sur la configuration des périphériques réseau, des alias de périphériques, ainsi que des informations sur les scripts de configuration réseau, veuillez consulter le Guide de mise en réseau Red Hat Enterprise Linux 7.

Esuite, le serveur *DNS* des domaines **FTP** devra être configuré pour faire référence à la machine qui convient. Pour obtenir des informations sur **BIND**, le protocole d'implémentation **DNS** utilisé sur Red Hat Enterprise Linux, ainsi que sur ses fichiers de configuration, veuillez consulter le Guide de mise en réseau Red Hat Enterprise Linux 7.

Pour que **vsftpd** réponde à des requêtes sur différentes adresses **IP**, de multiples copies du démon doivent être en cours d'exécution. Pour faciliter le lancement de multiples instances du démon **vsftpd**, une unité spéciale du service *systemd* (**vsftpd@.service**) pour lancer **vsftpd** en tant que service instancié est fournie dans le paquet vsftpd.

Afin de pouvoir utiliser cette unité du service, un fichier de configuration **vsftpd** séparé pour chaque instance du serveur **FTP** requise devra être créé et placé dans le répertoire **/etc/vsftpd/**. Remarquez que chacun de ces fichiers de configuration doit posséder un nom unique (tel que

/etc/vsftpd/*vsftpd-site-2*.conf) et doit uniquement être accessible en lecture et écriture par l'utilisateur **root**.

Dans chaque fichier de configuration pour chaque serveur **FTP** écoutant sur un réseau **IPv4**, la directive suivante doit être unique :

```
listen_address=N.N.N.N
```

Remplacez *N.N.N.N* par une *adresse* **IP** unique pour le site **FTP** servi. Si le site utilise **IPv6**, veuillez utiliser la directive **listen_address6** à la place.

Une fois que de multiples fichiers de configuration sont présents dans le répertoire **/etc/vsftpd/**, des instances individuelles du démon **vsftpd** peuvent être lancées par la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl start vsftpd@configuration-file-name.service
```

Dans la commande ci-dessus, remplacez *configuration-file-name* par le nom unique du fichier de configuration du serveur requis, tel que **vsftpd-site-2**. Remarquez que l'extension **.conf** du fichier de configuration ne doit pas être incluse dans la commande.

Si vous souhaitez lancer plusieurs instances du démon **vsftpd** à la fois, vous pouvez utiliser un fichier d'unité cible systemd (**vsftpd.target**), qui est fourni dans le paquet vsftpd. Cette cible systemd entraîne le lancement d'un démon indépendant **vsftpd** pour chaque fichier de configuration **vsftpd** qui se trouve dans le répertoire **/etc/vsftpd/**. Veuillez exécuter la commande suivante en tant qu'utilisateur **root** pour activer la cible :

```
~]# systemctl enable vsftpd.target
Created symlink from /etc/systemd/system/multi-
user.target.wants/vsftpd.target to /usr/lib/systemd/system/vsftpd.target.
```

La commande ci-dessus configure le gestionnaire de services systemd pour qu'il lance le service **vsftpd** (avec les instances configurées du serveur **vsftpd**) pendant l'initialisation. Pour lancer le service immédiatement, sans redémarrer le système, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl start vsftpd.target
```

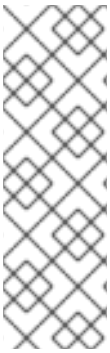
Veuillez consulter la Section 9.3, « Travailler avec des cibles Systemd » pour obtenir des informations supplémentaires sur la manière d'utiliser des cibles systemd pour gérer les services.

Les autres directives pouvant être altérées sur une base « par serveur » incluent :

- **anon_root**
- **local_root**
- **vsftpd_log_file**
- **xferlog_file**

14.2.2.3. Chiffrer des connexions vsftpd en utilisant TLS

Dans le but de contrer la nature non sécurisée inhérente de **FTP**, qui transmet les noms d'utilisateur, mots de passe, et données sans chiffrement par défaut, le démon **vsftpd** peut être configuré pour utiliser le protocole **TLS** afin d'authentifier les connexions et de chiffrer tous les transferts. Remarquez qu'un client **FTP** prenant en charge **TLS** est nécessaire pour communiquer avec **vsftpd** avec **TLS** activé.



NOTE

SSL (« Secure Sockets Layer ») est le nom d'une ancienne implémentation du protocole de sécurité. Les nouvelles versions sont appelées **TLS** (« Transport Layer Security »). Seules les nouvelles versions (**TLS**) doivent être utilisées car **SSL** connaît de graves défaillances de sécurité. La documentation incluse avec le serveur **vsftpd**, ainsi que les directives de configuration utilisées dans le fichier **vsftpd.conf**, utilisent le nom **SSL** lorsqu'elles font référence à des sujets liés à la sécurité, mais **TLS** est pris en charge et est utilisé par défaut lorsque la directive **ssl_enable** est paramétrée sur **YES**.

Veuillez définir la directive de configuration **ssl_enable** du fichier **vsftpd.conf** sur **YES** pour activer la prise en charge **TLS**. Les paramètres par défaut des autres directives liées à **TLS** qui sont automatiquement activés lorsque l'option **ssl_enable** est activée elle-même, fournissent une installation **TLS** relativement bien configurée. Ceci exige, entre autre, d'utiliser le protocole **TLS** v1 pour toutes les connexions (l'utilisation des versions non sécurisées du protocole **SSL** est désactivée par défaut) ou de forcer toutes les connexions non anonymes à utiliser **TLS** pour envoyer des mots de passe et des transferts de données.

Exemple 14.3. Configurer vsftpd pour utiliser TLS

Dans cet exemple, les directives de configuration désactivent explicitement les versions plus anciennes du protocole de sécurité **SSL** dans le fichier **vsftpd.conf** :

```
ssl_enable=YES
ssl_tlsv1=YES
ssl_sslv2=NO
ssl_sslv3=NO
```

Redémarrez le service **vsftpd** après avoir modifié sa configuration :

```
~]# systemctl restart vsftpd.service
```

See the vsftpd.conf(5) manual page for other **TLS**-related configuration directives for fine-tuning the use of **TLS** by **vsftpd**.

14.2.2.4. Politique SELinux pour vsftpd

La politique SELinux gouvernant le démon **vsftpd** (ainsi que d'autres processus **ftpd**) définit un contrôle d'accès obligatoire qui est, par défaut, basé sur le moins d'accès requis. Dans le but d'autoriser le démon **FTP** à accéder à des fichiers ou répertoires spécifiques, des étiquettes appropriées devront leur être assignés.

Par exemple, pour être en mesure de partager des fichiers de manière anonyme, l'étiquette **public_content_t** doit être assignée aux fichiers et répertoires à partager. Vous pouvez effectuer ceci en utilisant la commande **chcon** en tant qu'utilisateur **root** :

```
~]# chcon -R -t public_content_t /path/to/directory
```

Dans la commande ci-dessus, veuillez remplacer */path/to/directory* par le chemin du répertoire sur lequel vous souhaitez assigner l'étiquette. De même, si vous souhaitez paramétrer un répertoire pour téléverser des fichiers, vous devrez assigner l'étiquette **public_content_rw_t** à ce répertoire en particulier. En outre, l'option du booléen SELinux **allow_ftpd_anon_write** doit être paramétrée sur **1**. Veuillez utiliser la commande **setsebool** en tant qu'utilisateur **root** pour cela :

```
~]# setsebool -P allow_ftpd_anon_write=1
```

Si vous souhaitez que les utilisateurs locaux puissent accéder à leur répertoire personnel via **FTP**, ce qui est le paramètre par défaut sur Red Hat Enterprise Linux 7, l'option du booléen **ftp_home_dir** doit être définie sur **1**. Si **vsftpd** est autorisé à être exécuté en mode autonome, ce qui est également le cas par défaut sur Red Hat Enterprise Linux 7, l'option **ftpd_is_daemon** devra également être définie sur **1**.

See the ftpd_selinux(8) manual page for more information, including examples of other useful labels and Boolean options, on how to configure the SELinux policy pertaining to **FTP**. Also, see the Red Hat Enterprise Linux 7 SELinux User's and Administrator's Guide for more detailed information about SELinux in general.

14.2.3. Ressources supplémentaires

Pour obtenir davantage d'informations sur **vsftpd**, veuillez consulter les ressources suivantes.

14.2.3.1. Documentation installée

- **/usr/share/doc/vsftpd-*version-number*/** directory — Remplacez *version-number* par la version installée du paquet vsftpd. Ce répertoire contient un fichier **README** avec des informations de base sur le logiciel. Le fichier **TUNING** contient des conseils de réglage de base des performances et le répertoire **SECURITY/** contient des informations sur le modèle de sécurité employé par **vsftpd**.

- Pages de manuel liées à **vsftpd** — il existe des pages de manuel pour le démon et les fichiers de configuration. Ci-dessous figure une liste des pages de manuel les plus importantes.

  Applications du serveur

  - vsftpd(8) — Describes available command-line options for **vsftpd**.

  Fichiers de configuration

  - vsftpd.conf(5) — Contains a detailed list of options available within the configuration file for **vsftpd**.
  - hosts_access(5) — Describes the format and options available within the **TCP** wrappers configuration files: **hosts.allow** and **hosts.deny**.

  Interaction avec SELinux

  - ftpd_selinux(8) — Contains a description of the *SELinux* policy governing **ftpd** processes as well as an explanation of the way SELinux labels need to be assigned and Booleans set.

14.2.3.2. Documentation en ligne

À propos de vsftpd et de FTP en général

- http://vsftpd.beasts.org/ — la page du projet **vsftpd** est un bon emplacement pour trouver la documentation la plus récente et contacter l'auteur du logiciel.
- http://slacksite.com/other/ftp.html — Ce site web fournit une explication détaillée des différences entre les modes **FTP** actifs et passifs.

Documentation Red Hat Enterprise Linux

- Guide de mise en réseau Red Hat Enterprise Linux 7 — Le *Guide de mise en réseau* de Red Hat Enterprise Linux 7 documente les informations pertinentes à la configuration et à l'administration des interfaces réseau et des services réseau sur ce système. Il fournit une introduction à l'utilitaire **hostnamectl** et explique comment l'utiliser pour afficher et définir des noms d'hôtes sur la ligne de commande localement et à distance.
- Guide de l'utilisateur et de l'administrateur SELinux Red Hat Enterprise Linux 7 — le *Guide de l'utilisateur et de l'administrateur SELinux* Red Hat Enterprise Linux 7 décrit les principes de base de **SELinux** et documente en détails comment configurer et utiliser **SELinux** avec divers services, tels que **Apache HTTP Server**, **Postfix**, **PostgreSQL**, ou **OpenShift**. Celui-ci explique comment configurer les permissions d'accès **SELinux** pour les services système gérés par **systemd**.
- Guide de sécurité Red Hat Enterprise Linux 7 — le *Guide de sécurité* Red Hat Enterprise Linux 7 assiste les utilisateurs et administrateurs dans leur apprentissage des processus et pratiques de sécurisation de leurs stations de travail et serveurs contre des intrusions locales et distantes, les exploitations, et autres activités malicieuses. Celui-ci explique également comment sécuriser des services de système critiques.

Documents RFC pertinents

- *RFC 0959* — *Demande de commentaires* (RFC de l'anglais Request For Comments) d'origine du protocole **FTP** formulée par IETF.
- *RFC 1123* — la petite section concernant **FTP** étend et clarifie la demande RFC 0959.
- *RFC 2228* — extensions de sécurité **FTP**. **vsftpd** implémente le mini sous-ensemble qu'il faut pour prendre en charge les connexions TLS et SSL.
- *RFC 2389* — propose des commandes **FEAT** et **OPTS**.
- *RFC 2428* — prise en charge **IPv6**.

14.3. L'OUTIL « PRINT SETTINGS »

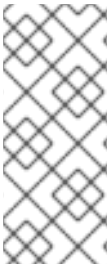
L'outil **Print Settings** est utilisé pour la configuration d'imprimantes, la maintenance des fichiers de configuration d'imprimantes, les répertoires spool et les filtres d'imprimantes, et la gestion des classes d'imprimantes.

L'outil est basé sur le système CUPS (« Common Unix Printing System »). Si vous avez mis à niveau le système à partir d'une version précédente de Red Hat Enterprise Linux qui utilisait CUPS, le processus de mise à niveau préserve les imprimantes configurées.

IMPORTANT

La page man de **cupsd.conf** documente la configuration d'un serveur CUPS. Elle inclut des directives pour activer la prise en charge **SSL**. Cependant, CUPS ne permet pas le contrôle des versions de protocole utilisées. À cause de la vulnérabilité décrite dans la *Résolution sur la vulnérabilité POODLE SSLv3.0 (CVE-2014-3566) pour les composants ne permettant pas à SSLv3 d'être désactivé via des paramètres de configuration*, Red Hat recommande de ne pas se fier à cela pour la sécurité. Il est recommandé d'utiliser **stunnel** pour fournir un tunnel sécurisé et désactiver **SSLv3**. Pour obtenir davantage d'informations sur l'utilisation de **stunnel**, veuillez consulter le Guide de sécurité Red Hat Enterprise Linux 7.

Pour des connexions sécurisées ad hoc à l'outil **Print Settings** d'un système, veuillez utiliser le transfert X11 sur **SSH** comme décrit dans la Section 10.4.1, « Transfert X11 ».



NOTE

Vous pouvez effectuer les mêmes opérations et des opérations supplémentaires sur les imprimantes directement à partir de l'application web CUPS ou à partir de la ligne de commande. Pour accéder à l'application, dans un navigateur web, veuillez vous rendre sur http://localhost:631/. Pour des manuels CUPS, veuillez consulter les liens sur l'onglet d'accueil (« **Home** ») du site web.

14.3.1. Lancer l'outil de configuration « Print Settings »

Avec l'outil de configuration **Print Settings**, vous pouvez effectuer diverses opérations sur les imprimantes existantes et paramétrer de nouvelles imprimantes. Vous pouvez également utiliser CUPS directement (veuillez vous rendre sur http://localhost:631/ pour accéder à l'application web CUPS).

Pour lancer l'outil **Print Settings** à partir de la ligne de commande, veuillez saisir **system-config-printer** à l'invite de shell. L'outil **Print Settings** apparaît. Alternativement, si le bureau GNOME est utilisé, appuyez sur la touche **Super** pour aller sur la « Vue d'ensemble des activités », saisissez **Print Settings** puis appuyez sur **Entrée**. L'outil **Print Settings** apparaît. La touche **Super** peut se trouver sous diverses formes, selon le clavier ou le matériel, mais le plus souvent, il s'agit de la touche Windows ou de la touche de Commande, habituellement à gauche de la **Barre d'espace**.

La fenêtre **Print Settings** décrite dans la Figure 14.3, « Fenêtre « Print Settings » » apparaît.

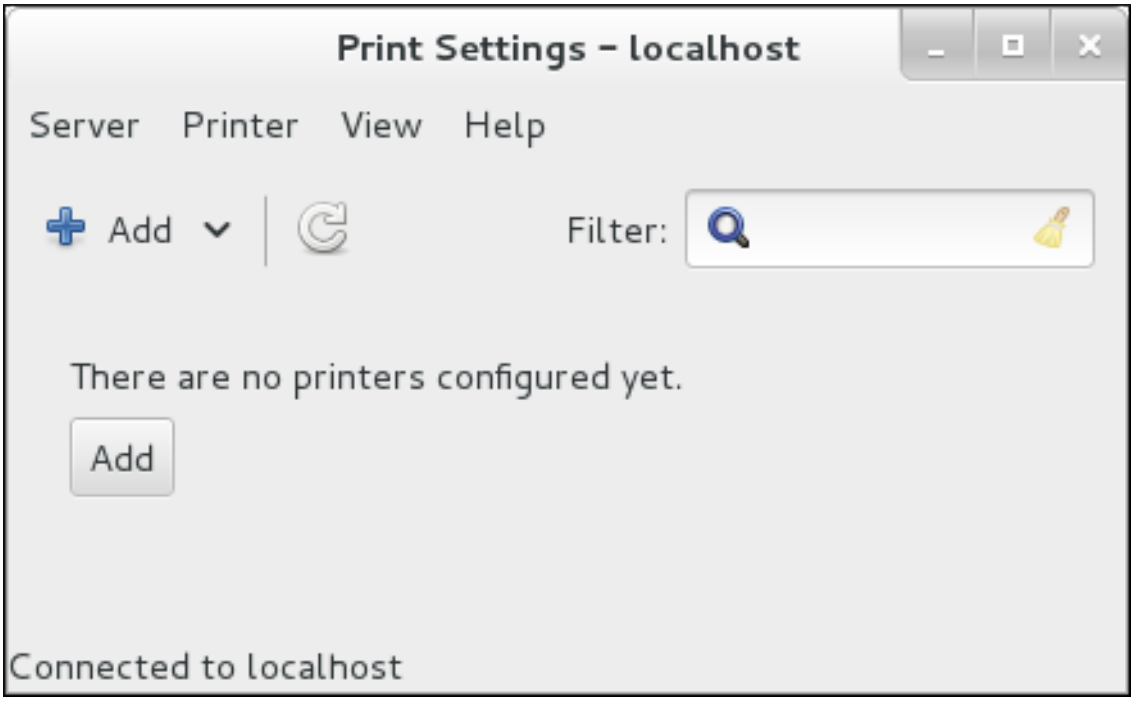


Figure 14.3. Fenêtre « Print Settings »

14.3.2. Lancer l'installation d'une imprimante

Le processus d'installation d'imprimante varie selon le type de file d'attente de l'imprimante.

Si vous paramétrez une imprimante locale connectée par USB, l'imprimante sera découverte et ajoutée automatiquement. Il vous sera demandé de confirmer les paquets à installer et de fournir un mot de passe d'administrateur ou le mot de passe utilisateur **root**. Les imprimantes locales connectées avec d'autres types de port et les imprimantes réseau doivent être installées manuellement.

Veuillez suivre cette procédure pour lancer une installation d'imprimante manuelle :

1. Lancez l'outil « Print Settings » (veuillez consulter la Section 14.3.1, « Lancer l'outil de configuration « Print Settings » »).
2. Rendez-vous sur **Serveur** → **Nouvelle** → **Imprimante**.
3. Dans la boîte de dialogue **Authentification**, veuillez saisir un mot de passe d'administrateur ou d'utilisateur **root**. S'il s'agit de la première fois que vous configurez une imprimante distante, il vous sera demandé d'autoriser un ajustement du pare-feu.
4. Veuillez sélectionner le type de connexion de l'imprimante et fournir ses détails dans la zone se trouvant sur la droite.

14.3.3. Ajouter une imprimante locale

Veuillez suivre cette procédure pour ajouter une imprimante locale connectée à autre chose qu'un port série :

1. Ouvrez la boîte de dialogue **Ajouter** (veuillez consulter la Section 14.3.2, « Lancer l'installation d'une imprimante »).
2. Si le périphérique n'apparaît pas automatiquement, veuillez sélectionner le port sur lequel l'imprimante est connectée dans la liste sur la gauche (tel que **Serial Port #1** ou **LPT #1**).

3. Sur la droite, veuillez saisir les propriétés de connexion :

   pour `Other`

   `URI` (par exemple file:/dev/lp0)

   pour `Serial Port`

   Débit en bauds

   Parité

   Morceaux de données

   Contrôle du flux

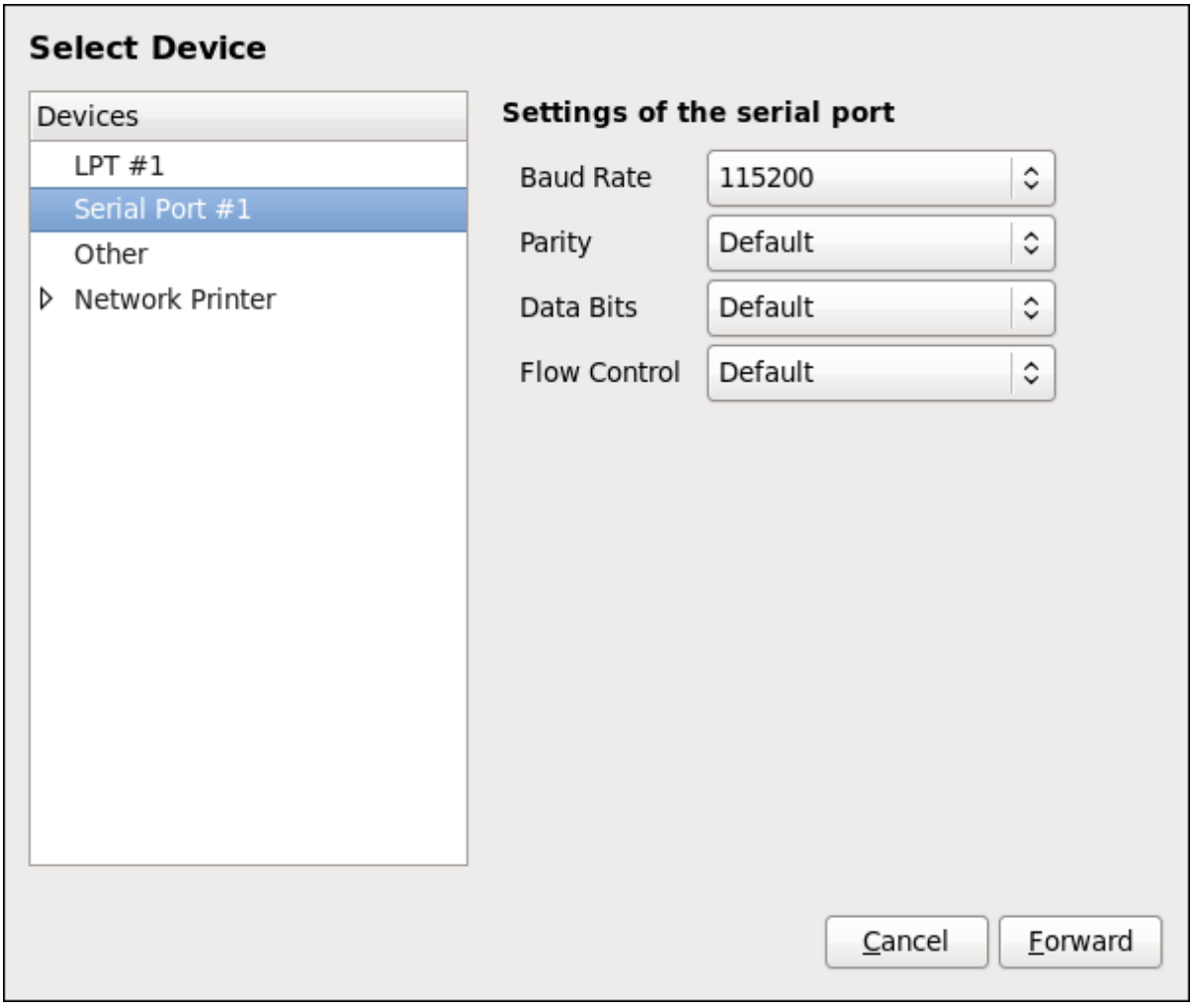


Figure 14.4. Ajouter une imprimante locale

4. Cliquez sur **`Suivant`**.

5. Sélectionnez le modèle de l'imprimante. Veuillez consulter la Section 14.3.8, « Sélectionner le modèle de l'imprimante et terminer » pour obtenir des détails.

14.3.4. Ajouter une imprimante AppSocket/HP JetDirect

Veuillez suivre cette procédure pour ajouter une imprimante AppSocket/HP JetDirect :

1. Ouvrez la boîte de dialogue **Nouvelle imprimante** (veuillez consulter la Section 14.3.1, « Lancer l'outil de configuration « Print Settings » »).

2. Dans la liste sur la gauche, sélectionnez **Imprimante réseau** → **AppSocket/HP JetDirect**.

3. Sur la droite, veuillez saisir les paramètres de connexion :

   Nom d'hôte

   Le nom d'hôte de l'imprimante ou l'adresse **IP**.

   Numéro de port

   Écoute du port de l'imprimante pour les tâches d'impression (Par défaut, **9100**).

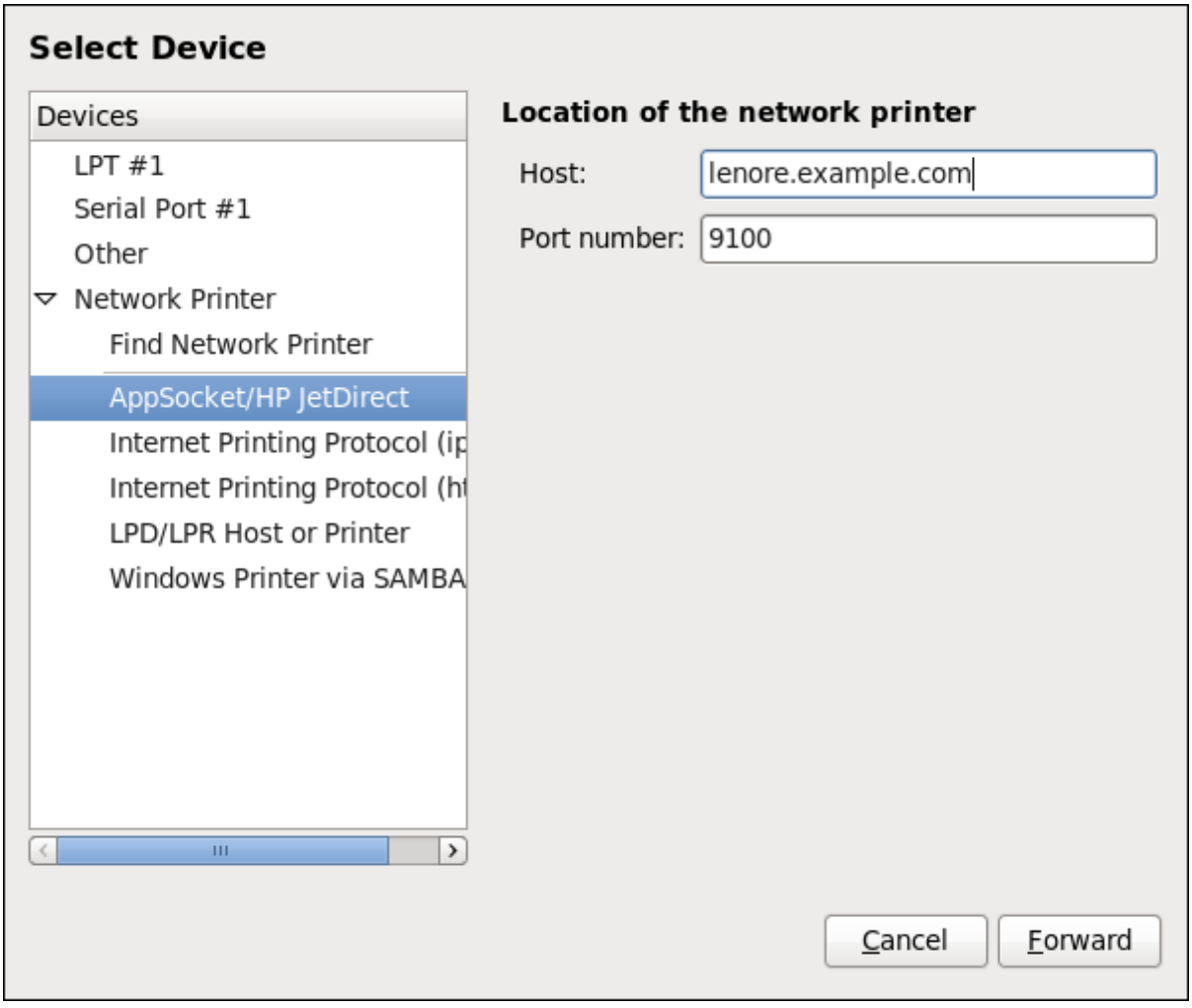


Figure 14.5. Ajouter une imprimante JetDirect

4. Cliquez sur **Suivant**.

5. Sélectionnez le modèle de l'imprimante. Veuillez consulter la Section 14.3.8, « Sélectionner le modèle de l'imprimante et terminer » pour obtenir des détails.

14.3.5. Ajouter une imprimante IPP

Une imprimante **IPP** est une imprimante attachée à un système différent sur le même réseau TCP/IP. Le système auquel cette imprimante est attachée pourrait également exécuter CUPS ou être simplement configuré pour utiliser **IPP**.

Si un pare-feu est activé sur le serveur de l'imprimante, alors le pare-feu doit être configuré pour autoriser les connexions **TCP** entrantes sur le port **631**. Remarquez que le protocole de navigation CUPS autorise les machines clientes à découvrir les files d'attentes CUPS partagées automatiquement. Pour activer cela, le pare-feu sur la machine cliente doit être configuré pour autoriser les paquets **UDP** entrants sur le port **631**.

Veuillez suivre cette procédure pour ajouter une imprimante **IPP** :

1. Ouvrez la boîte de dialogue **Nouvelle imprimante** (veuillez consulter la Section 14.3.2, « Lancer l'installation d'une imprimante »).

2. Dans la liste des périphériques se trouvant sur la gauche, veuillez sélectionner **Imprimante réseau** et **Internet Printing Protocol (ipp)** ou **Internet Printing Protocol (https)**.

3. Sur la droite, veuillez saisir les paramètres de connexion :

   Hôte

   Nom d'hôte de l'imprimante **IPP**.

   File d'attente

   Nom de la file d'attente à donner à la nouvelle file d'attente (si la boîte reste vide, un nom basé sur le nœud du périphérique sera utilisé).

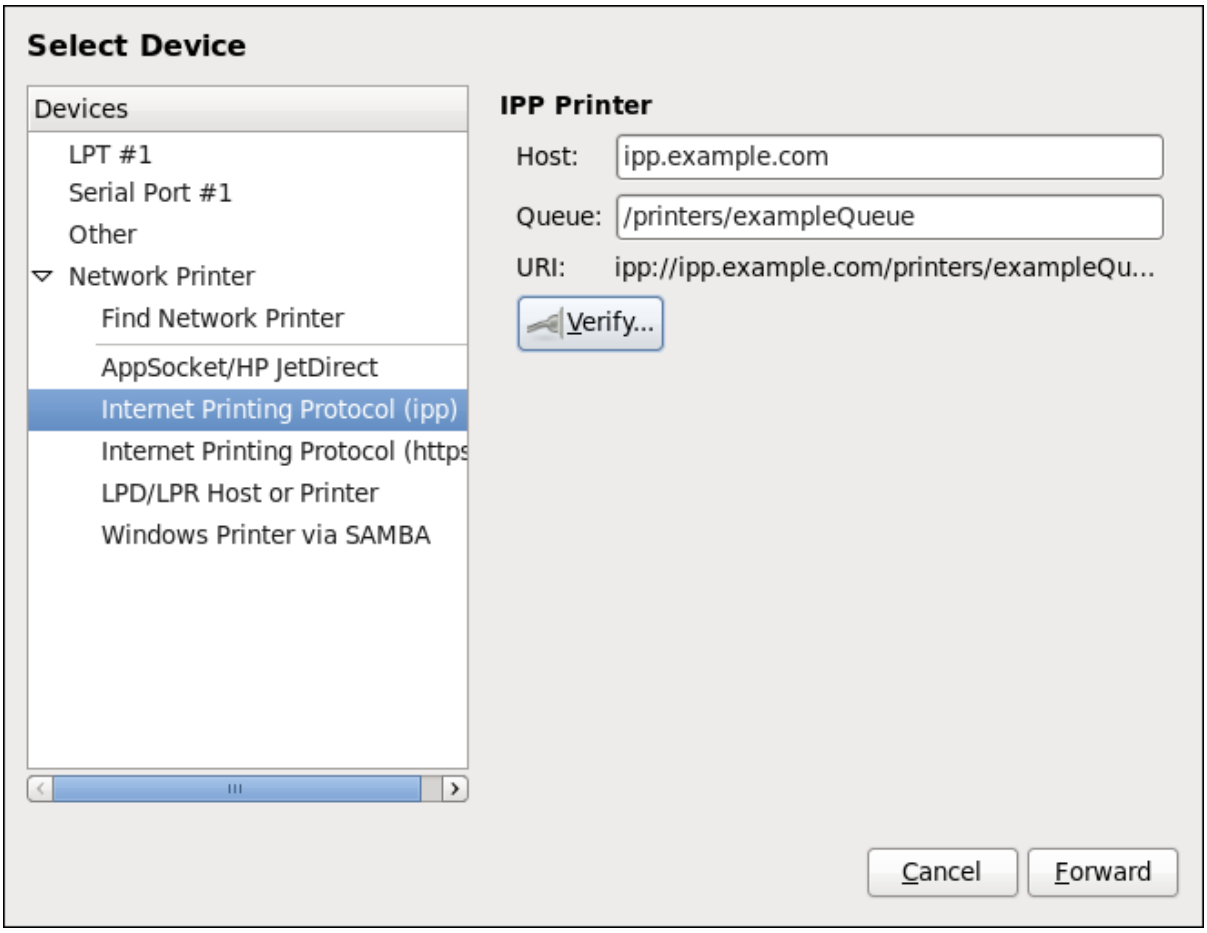


Figure 14.6. Ajouter une imprimante IPP

4. Cliquez sur **Suivant** pour continuer.

5. Sélectionnez le modèle de l'imprimante. Veuillez consulter la Section 14.3.8, « Sélectionner le modèle de l'imprimante et terminer » pour obtenir des détails.

14.3.6. Ajouter un hôte ou une imprimante LPD/LPR

Veuillez suivre cette procédure pour ajouter un hôte ou une imprimante LDP/LPR :

1. Ouvrez la boîte de dialogue **`Nouvelle imprimante`** (veuillez consulter la Section 14.3.2, « Lancer l'installation d'une imprimante »).

2. Dans la liste des périphériques sur la gauche, sélectionnez**Imprimante réseau** → **Hôte ou imprimante LPD/LPR**.

3. Sur la droite, veuillez saisir les paramètres de connexion :

   `Hôte`

   Nom d'hôte de l'imprimante ou de l'hôte LPD/LPR.

   Optionnellement, cliquez sur **`Sonder`** pour trouver les files d'attente situées sur l'hôte LPD.

   `File d'attente`

   Nom de la file d'attente à donner à la nouvelle file d'attente (si la boîte reste vide, un nom basé sur le nœud du périphérique sera utilisé).

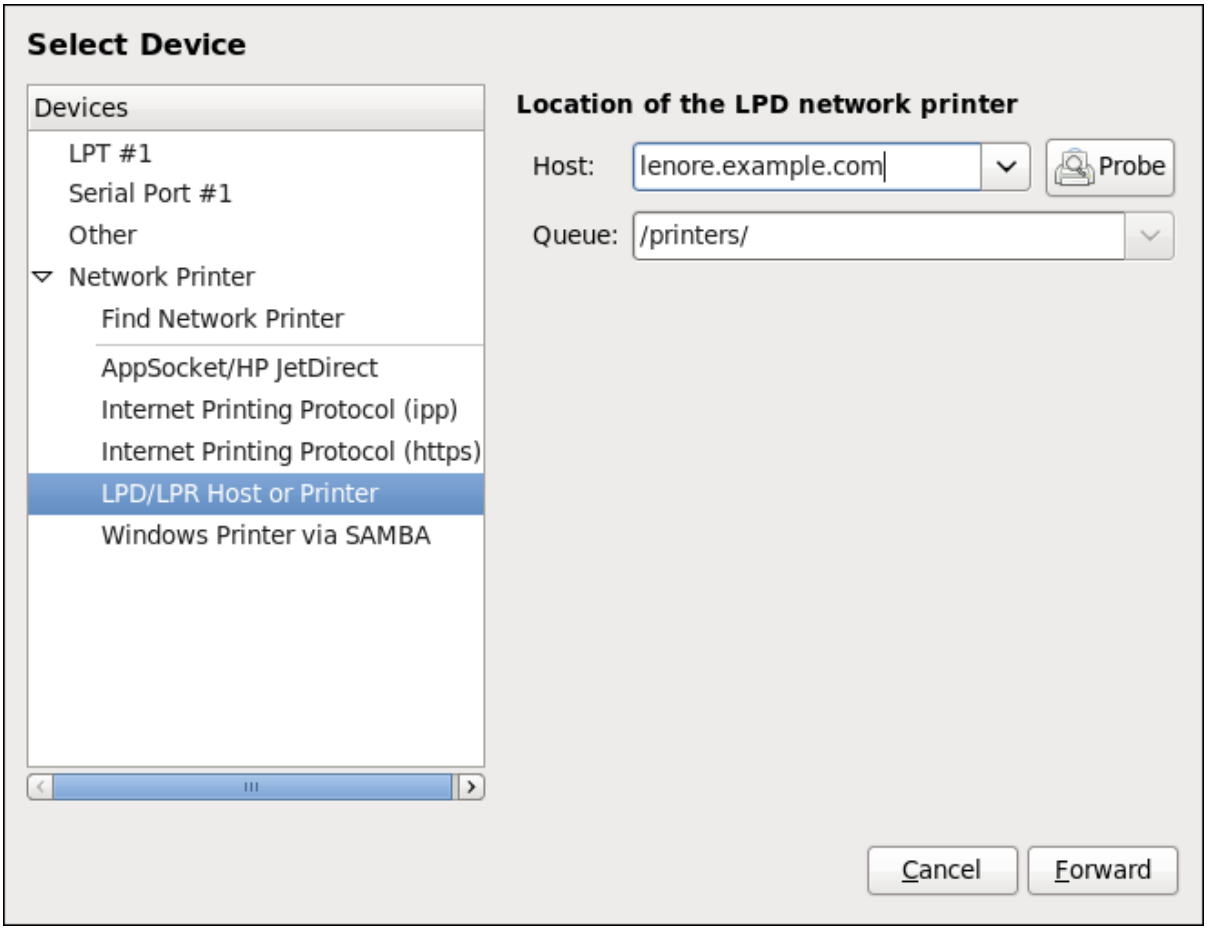


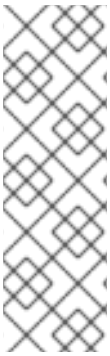
Figure 14.7. Ajouter une imprimante LPD/LPR

4. Cliquez sur **`Suivant`** pour continuer.

5. Sélectionnez le modèle de l'imprimante. Veuillez consulter la Section 14.3.8, « Sélectionner le modèle de l'imprimante et terminer » pour obtenir des détails.

14.3.7. Ajouter une imprimante Samba (SMB)

Veuillez suivre cette procédure pour une imprimante Samba :



NOTE

Remarquez que pour ajouter une imprimante Samba, le paquet samba-client doit être installé. Vous pouvez faire cela en exécutant ce qui suit en tant qu'utilisateur **root** :

```
yum install samba-client
```

Pour obtenir davantage d'informations sur l'installation de paquets avec Yum, veuillez consulter la Section 8.2.4, « Installation de paquets ».

1. Ouvrez la boîte de dialogue **Nouvelle imprimante** (veuillez consulter la Section 14.3.2, « Lancer l'installation d'une imprimante »).

2. Dans la liste sur la gauche, veuillez sélectionner **Imprimante réseau** → **Imprimante Windows via SAMBA**.

3. Veuillez saisir l'adresse SMB dans le champ **smb://**. Utilisez le format *nom d'ordinateur/partage d'imprimante*. Dans la Figure 14.8, « Ajouter une imprimante SMB », le *nom d'ordinateur* est **dellbox** et le *partage d'imprimante* est **r2**.

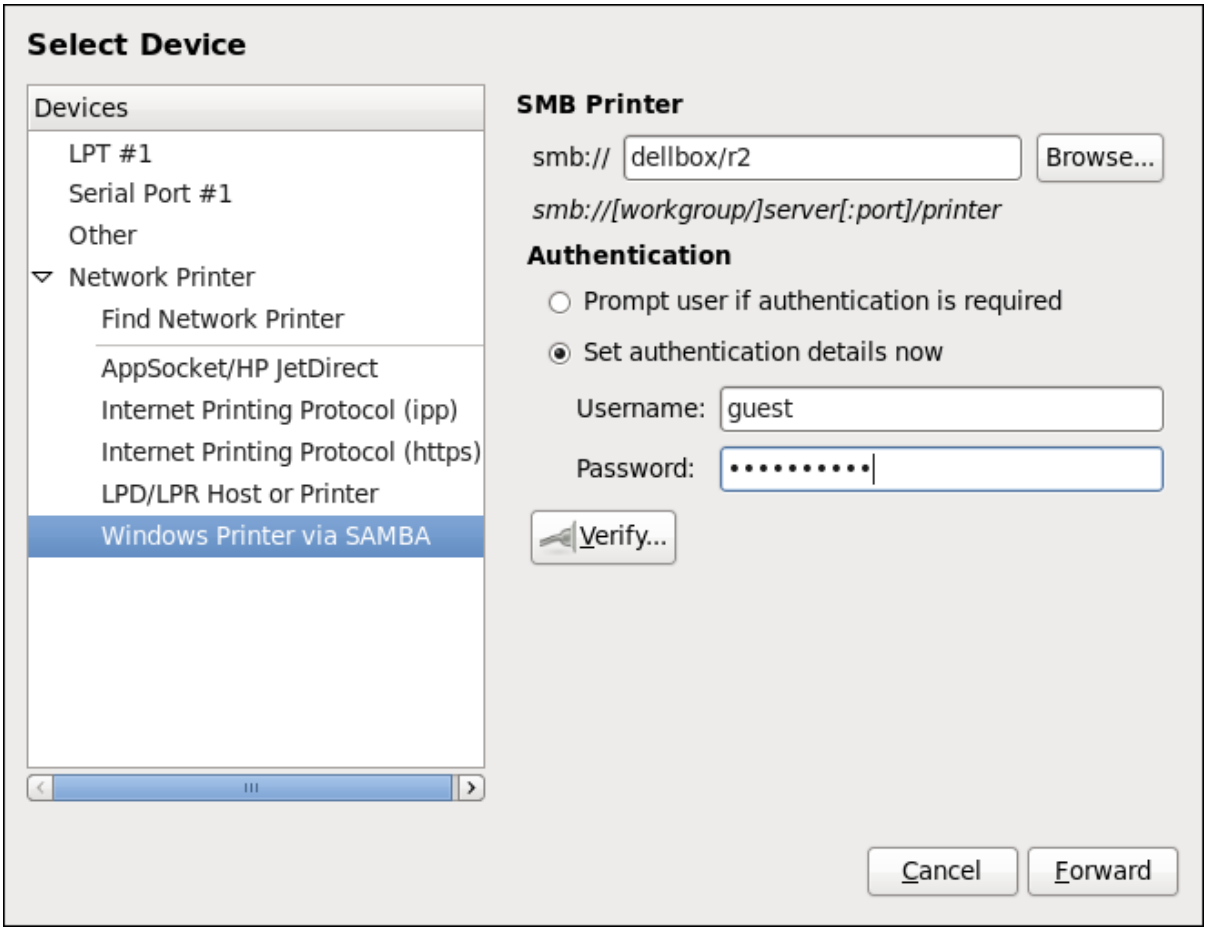


Figure 14.8. Ajouter une imprimante SMB

4. Veuillez cliquer sur **Parcourir** pour afficher les groupes de travail ou domaines disponibles. Pour uniquement afficher les files d'attente d'un hôte particulier, veuillez saisir le nom d'hôte (nom NetBios) et cliquez sur **Parcourir**.

5. Veuillez sélectionner l'une des options :

   - **Demander à l'utilisateur si l'authentification est requise** : le nom d'utilisateur et le mot de passe sont exigés de l'utilisateur lors de l'impression d'un document.

   - **Définir les infos d'authentification maintenant** : permet de fournir les informations d'authentification immédiatment afin qu'elles ne soient pas requises ultérieurement. Dans le champ **Nom d'utilisateur**, veuillez saisir le nom d'utilisateur pour accéder à l'imprimante. Cet utilisateur doit exister sur le système SMB, et l'utilisateur doit avoir la permission d'accéder à l'imprimante. Le nom d'utilisateur par défaut est habituellement « **guest** » (invité) pout les serveurs Windows, ou « **nobody** » pour les serveurs Samba.

6. Saisissez le **Mot de passe** (si requis) de l'utilisateur spécifié dans le champ **Nom d'utilisateur**.

AVERTISSEMENT

Les noms d'utilisateurs et mots de passe Samba sont stockés dans le serveur de l'imprimante en tant que fichiers non chiffrés lisibles par l'utilisateur **root** et le démon d'impression Linux (« Linux Printing Daemon »), **lpd**. Ainsi, les autres utilisateurs ayant un accès **root** au serveur de l'imprimante peuvent afficher le nom d'utilisateur et le mot de passe à utiliser pour accéder à l'imprimante Samba.

Ainsi, lorsque vous choisissez un nom d'utilisateur et un mot de passe pour accéder à une imprimante Samba, il est recommandé de choisir un mot de passe différent de celui utilisé pour accéder à votre système Red Hat Enterprise Linux local.

Si des fichiers sont partagés sur le serveur de l'imprimante Samba, il est également recommandé de choisir un mot de passe différent de celui qui est utilisé par la file d'attente d'impression.

7. Cliquez sur **Vérifier** pour tester la connexion. Lorsque la vérification est réussie, une boîte de dialogue apparaît confirmant l'accessibilité du partage de l'imprimante.

8. Cliquez sur **Suivant**.

9. Sélectionnez le modèle de l'imprimante. Veuillez consulter la Section 14.3.8, « Sélectionner le modèle de l'imprimante et terminer » pour obtenir des détails.

14.3.8. Sélectionner le modèle de l'imprimante et terminer

Une fois que vous aurez correctement sélectionné un type de connexion d'imprimante, le système tentera d'acquérir un pilote. Si le processus échoue, vous pouvez localiser ou rechercher les ressources du pilote manuellement.

Veuillez suivre cette procédure pour fournir le pilote de l'imprimante et terminer l'installation :

1. Dans la fenêtre affichée après l'échec de la détection automatique du pilote, veuillez sélectionner l'une des options suivantes :

   - **Sélectionner une imprimante dans la base de données** — le système choisit un pilote en se basant sur la marque de l'imprimante sélectionnée dans la liste des **Marques**. Si votre imprimante n'est pas répertoriée, veuillez choisir **Générique**.

   - **Fournir le fichier PPD** — le système utilise le fichier « *PostScript Printer Description* » (ou PPD) pour l'installation. Un fichier PPD peut également être remis avec votre imprimante comme s'il était normalement fourni par le constructeur. Si le fichier PPD est disponible, vous pourrez choisir cette option et utiliser la barre de navigation sous la description de l'option pour sélectionner le fichier PPD.

   - **Rechercher un pilote d'imprimante à télécharger** — saisissez la marque et le modèle de l'imprimante dans le champ **Marque et modèle** pour rechercher les paquets appropriés sur OpenPrinting.org.

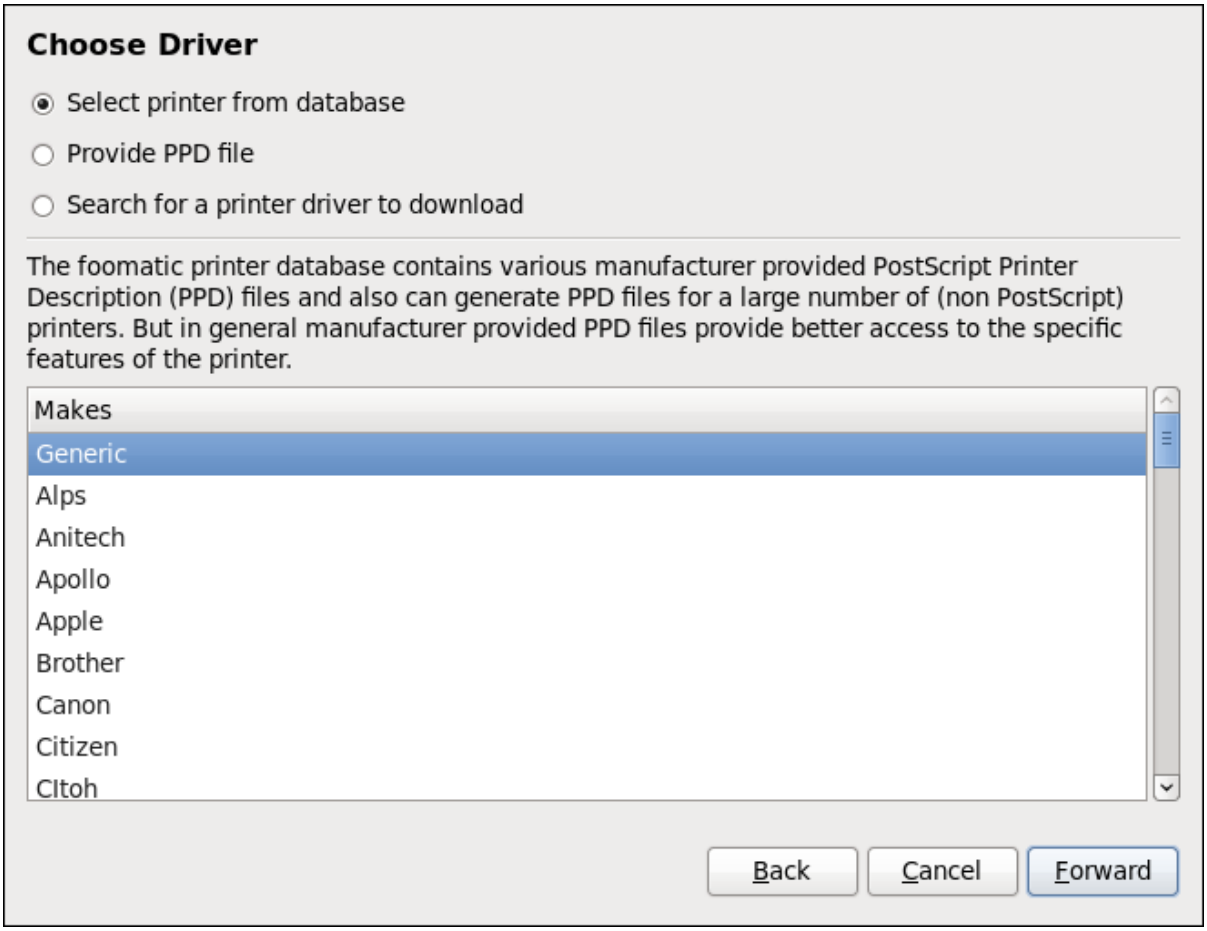


Figure 14.9. Sélectionner une marque d'imprimante

2. Selon votre choix précédent, veuillez fournir des détails dans la zone affichée ci-dessous :
   - Marque de l'imprimante pour l'option **Sélectionner l'imprimante dans la base de données**.
   - Emplacement du fichier PPD pour l'option **Fournir le fichier PPD**.
   - Marque et modèle de l'imprimante pour l'option **Rechercher un pilote d'imprimante à télécharger**.
3. Cliquez sur **Suivant** pour continuer.
4. Si cela est applicable pour votre option, la fenêtre affichée dans la Figure 14.10, « Sélectionner un modèle d'imprimante » apparaît. Choisissez le modèle correspondant dans la colonne **Modèles** sur la gauche.

NOTE

Sur la droite, le pilote de l'imprimante recommandé est automatiquement sélectionné, cependant il est possible de sélectionner un autre pilote disponible. Le pilote de l'imprimante traite les données que vous souhaitez imprimer sous un format que l'imprimante peut comprendre. Comme une imprimante est attachée directement à l'ordinateur, il est nécessaire d'avoit un pilote d'imprimante pour traiter les données envoyées sur l'imprimante.

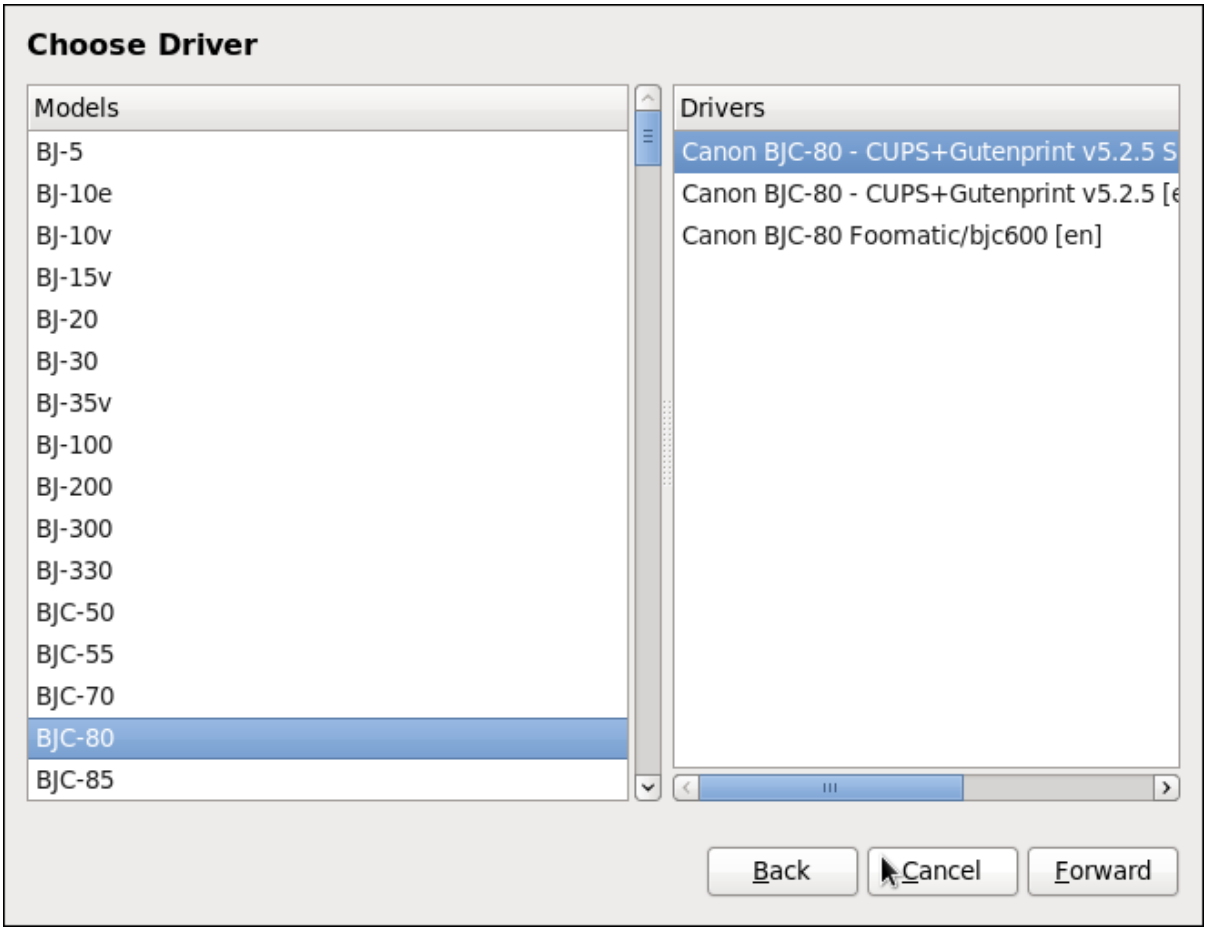


Figure 14.10. Sélectionner un modèle d'imprimante

5. Cliquez sur **Suivant**.

6. Sous **Décrire l'imprimante**, saisissez un nom unique pour l'imprimante dans le champ **Nom de l'imprimante**. Le nom de l'imprimante peut contenir des lettres, des chiffres, des tirets (-), et des traits de soulignement (_) ; il *ne doit pas* contenir d'espaces. Vous pouvez également utiliser les champs **Description** et **Emplacement** pour ajouter des informations supplémentaires sur l'imprimante. Ces champs sont optionnels et peuvent contenir des espaces.

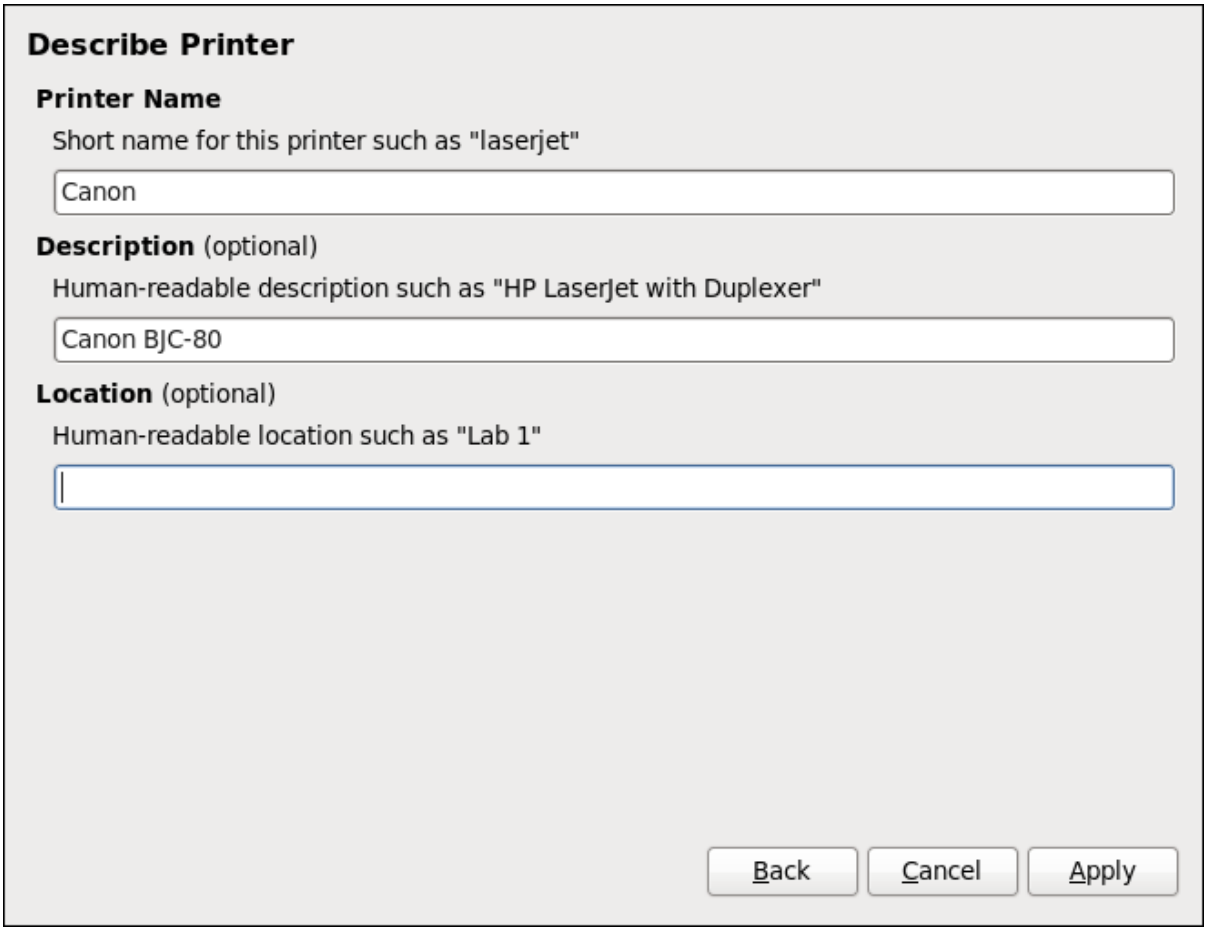


Figure 14.11. Installation de l'imprimante

7. Veuillez cliquer sur **Appliquer** pour confirmer la configuration de l'imprimante et ajouter la file d'attente de l'imprimante si les paramètres sont corrects. Veuillez cliquer sur **Précédent** pour modifier la configuration de l'imprimante.

8. Une fois les changements appliqués, une boîte de dialogue apparaît, vous permettant d'imprimer une page test. Cliquez sur **Oui** pour imprimer la page test maintenant. Alternativement, il est possible d'imprimer une page test ultérieurement, comme décrit dans la Section 14.3.9, « Imprimer une page test ».

14.3.9. Imprimer une page test

Après avoir installé une imprimante ou modifié une configuration d'imprimante, imprimez une page test pour vous assurer que l'imprimante fonctionne correctement :

1. Faites un clic droit dans la fenêtre **Impression** et cliquez sur **Propriétés**.

2. Dans la fenêtre Propriétés, cliquez sur **Paramètres** sur la gauche.

3. Sur l'onglet **Paramètres** affiché, cliquez sur le bouton **Imprimer la page test**.

14.3.10. Modifier les imprimantes existantes

Pour supprimer une imprimante existante, dans la fenêtre **Print Settings**, sélectionnez l'imprimante, puis rendez-vous sur **Imprimante** → **Supprimer**. Confirmez la suppression de l'imprimante. Alternativement, appuyez sur la touche **Delete** (« Suppr. »).

Pour définir l'imprimante par défaut, faites un clic droit sur l'imprimante dans la liste des imprimantes, puis cliquez sur le bouton **Définir par défaut** dans le menu de contexte.

14.3.10.1. Page des paramètres

Pour changer la configuration du pilote de l'imprimante, double-cliquez sur le nom correspondant dans la liste **Imprimante**, puis cliquez sur l'étiquette **Paramètres** sur la gauche pour afficher la page des **Paramètres**.

Vous pouvez modifier les paramètres de l'imprimante, comme la marque et le modèle, imprimer une page test, modifier l'emplacement du périphérique (URI), etc.

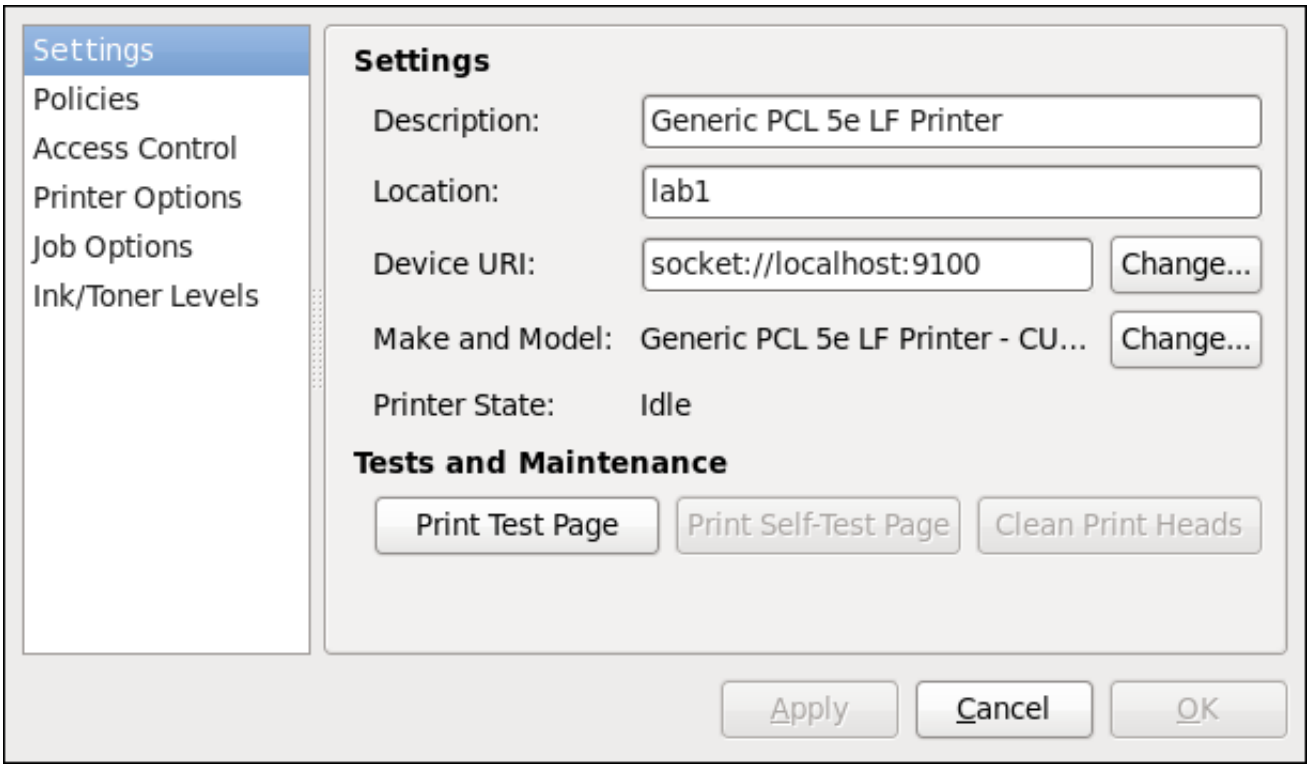


Figure 14.12. Page des paramètres

14.3.10.2. Page des politiques

Veuillez cliquer sur le bouton **Politiques** à gauche pour changer les paramètres de l'état de l'imprimante et de la sortie d'impression.

Vous pouvez sélectionner les états de l'imprimante, configurer la **Politique des erreurs** de l'imprimante (vous pouvez décider d'annuler, de réessayer ou d'arrêter la tâche d'impression lorsqu'une erreur se produit).

Vous pouvez également créer une *page de bannière* (une page décrivant les facettes de la tâche d'impression, comme l'imprimante d'origine, le nom d'utilisateur d'origine de la tâche, et le statut de sécurité du document en cours d'impression) : veuillez cliquer sur le menu déroulant **Lancer la bannière** ou **Arrêter la bannière** et choisir l'option qui décrit le mieux la nature des tâches d'impression (par exemple, **confidentiel**).

14.3.10.2.1. Partager des imprimantes

Sur la page **Comportements**, vous pouvez marquer une imprimante comme partagée : si une imprimante est partagée, les utilisateurs publiés sur le réseau peuvent l'utiliser. Pour autoriser la fonction

de partage pour imprimantes, veuillez vous rendre sur **Serveur** → **Paramètres** et sélectionnez **Publier les imprimantes partagées connectés à ce système**.

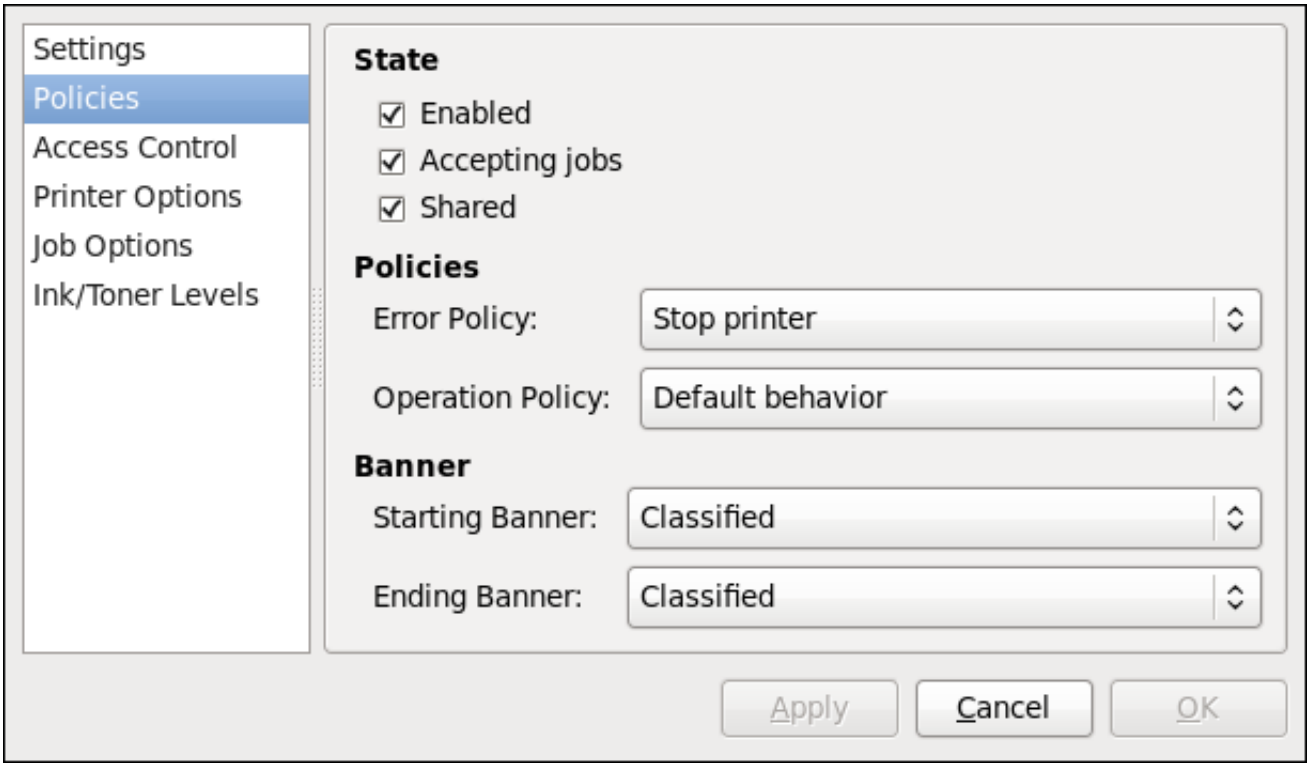


Figure 14.13. Page des comportements

Assurez-vous que le pare-feu autorise les connexions **TCP** entrantes sur le port **631**, le port du protocole **IPP** (« Network Printing Server »). Pour autoriser le trafic **IPP** à travers le pare-feu sur Red Hat Enterprise Linux 7, utilisez le service **firewalld IPP**. Pour cela, veuillez procéder comme suit :

Procédure 14.4. Activer le service IPP dans firewalld

1. Pour lancer l'outil graphique **firewall-config**, veuillez appuyer sur la touche **Super** pour entrer dans la « Vue d'ensemble des activités », saisissez **firewall**, puis appuyez sur la touche **Entrée**. La fenêtre **Configuration du pare-feu** s'ouvrira. Il vous sera alors demandé un mot de passe d'administrateur ou d'utilisateur **root**.

   Alternativement, pour lancer l'outil de configuration du pare-feu graphique en utilisant la ligne de commande, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

   ```
   ~]# firewall-config
   ```

   La fenêtre **Configuration du pare-feu** s'ouvrira.

   Recherchez le mot « Connecté » (« Connected ») dans le coin en bas à gauche. Ceci indique que l'outil **firewall-config** est connecté au démon de l'espace utilisateur **firewalld**.

   Pour immédiatement changer les paramètres actuels du pare-feu, assurez-vous que le menu de sélection déroulant étiquetté **Configuration** est défini sur **Runtime**. Alternativement, pour modifier les paramètres à appliquer lors du prochain démarrage système, ou du prochain rechargement du pare-feu, sélectionnez **Permanent** dans la liste déroulante.

2. Sélectionnez l'onglet **Zones** et sélectionnez la zone du pare-feu à faire correspondre avec l'interface réseau à utiliser. La zone par défaut est la zone « **public** » (publique). L'onglet **Interfaces** affiche quelles interfaces ont été assignées à une zone.

3. Veuillez sélectionner l'onglet **Services** puis sélectionnez le service **ipp** pour activer le partage. Le service **ipp-client** est requis pour accéder aux imprimantes réseau.

4. Fermez l'outil **firewall-config**.

Pour obtenir des informations supplémentaires sur l'ouverture et la fermeture de ports sur **firewalld**, veuillez consulter le Guide de sécurité Red Hat Enterprise Linux 7.

14.3.10.2.2. Page de contrôle des accès

Vous pouvez modifier l'accès niveau utilisateur à l'imprimante configurée sur la page **Contrôle des accès**. Veuillez cliquer sur l'étiquette **Contrôle des accès** sur la gauche pour afficher la page. Sélectionnez **Autoriser l'impression à tout le monde sauf à ces utilisateurs** ou **Refuser l'impression à tout le monde sauf à ces utilisateurs** et définissez l'ensemble des utilisateurs ci-dessous : saisissez le nom d'utilisateur dans la boîte de texte et cliquez sur le bouton **Ajouter** pour ajouter l'utilisateur à l'ensemble des utilisateurs.

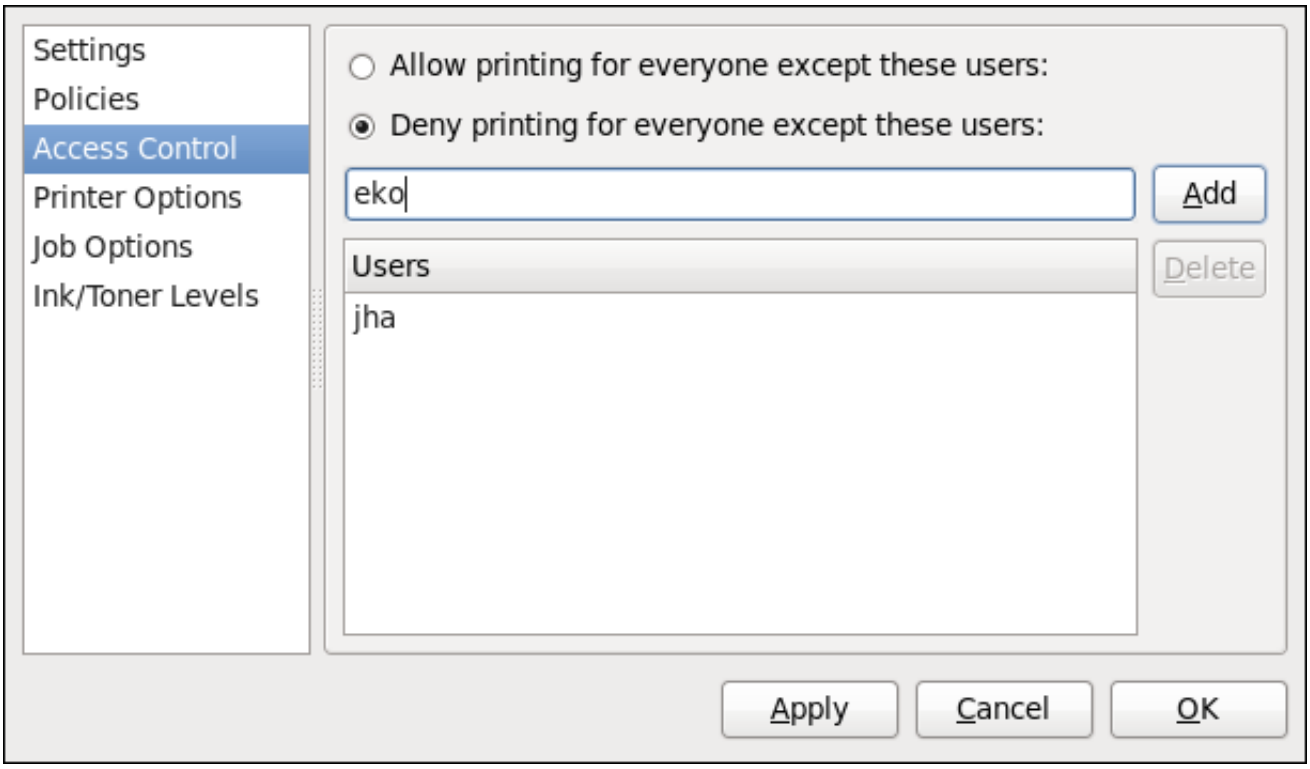


Figure 14.14. Page de contrôle des accès

14.3.10.2.3. Page des options d'imprimante

La page **Options d'imprimante** contient diverses options de configuration pour le support physique et la sortie. Son contenu peut varier d'une imprimante à l'autre. Cette page contient des paramètres généraux sur l'impression, le papier, la qualité et la taille pour l'impression.

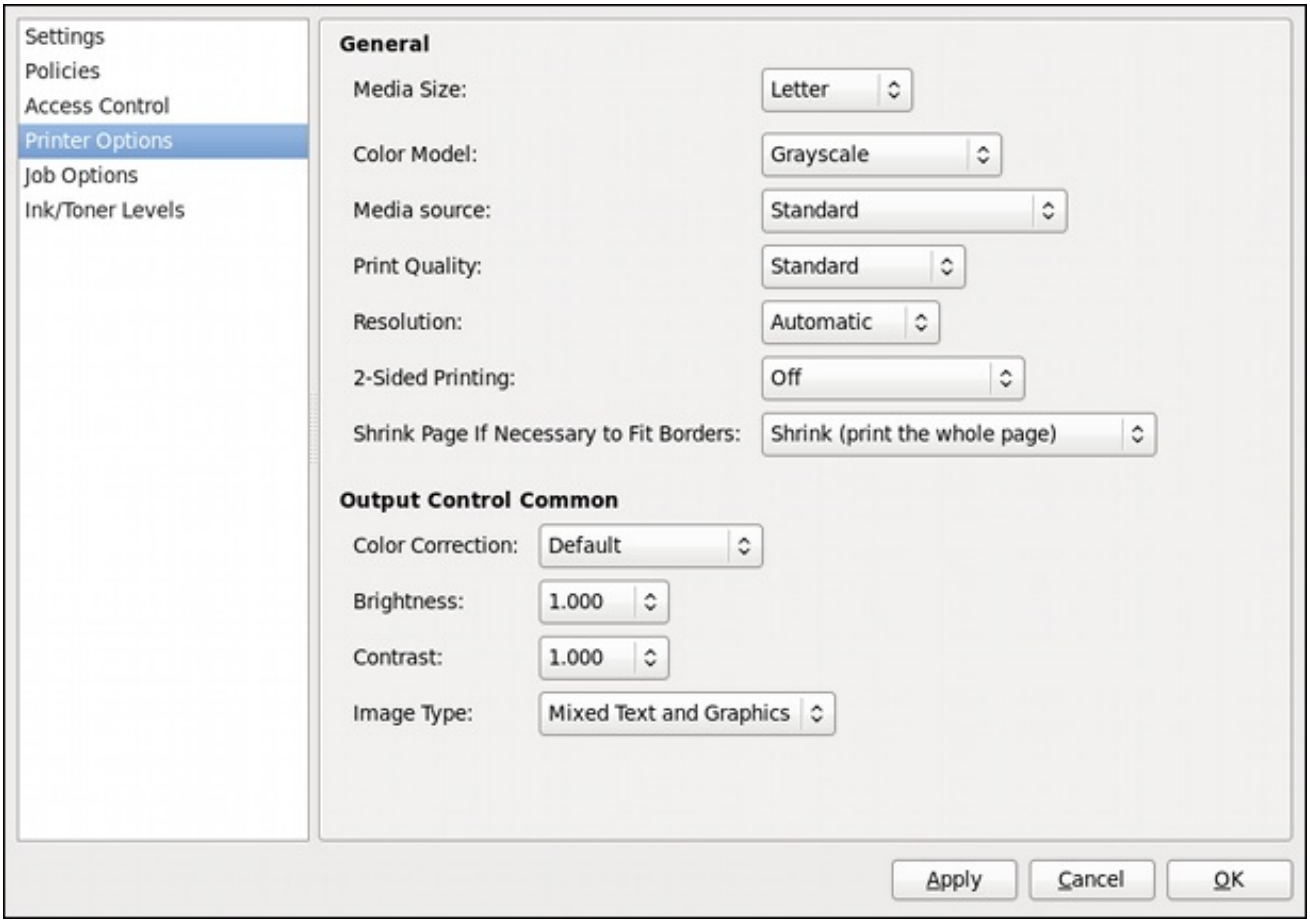


Figure 14.15. Page des options d'imprimante

14.3.10.2.4. Page des options de tâches

Sur la page **`Options de tâches`**, il est possible de détailler les options des tâches de l'imprimante. Cliquez sur l'étiquette **`Options de tâches`** sur la gauche pour afficher la page. Modifiez les paramètres par défaut pour appliquer des options de tâches personnalisées, comme le nombre de copies, l'orientation, les pages par côté, la mise à l'échelle (augmenter ou réduire la taille de la zone imprimable, qui peut être réutilisée pour contenir une zone imprimable trop grande sur un support imprimable physiquement plus petit), des options de texte détaillées, et des options de tâches personnalisées.

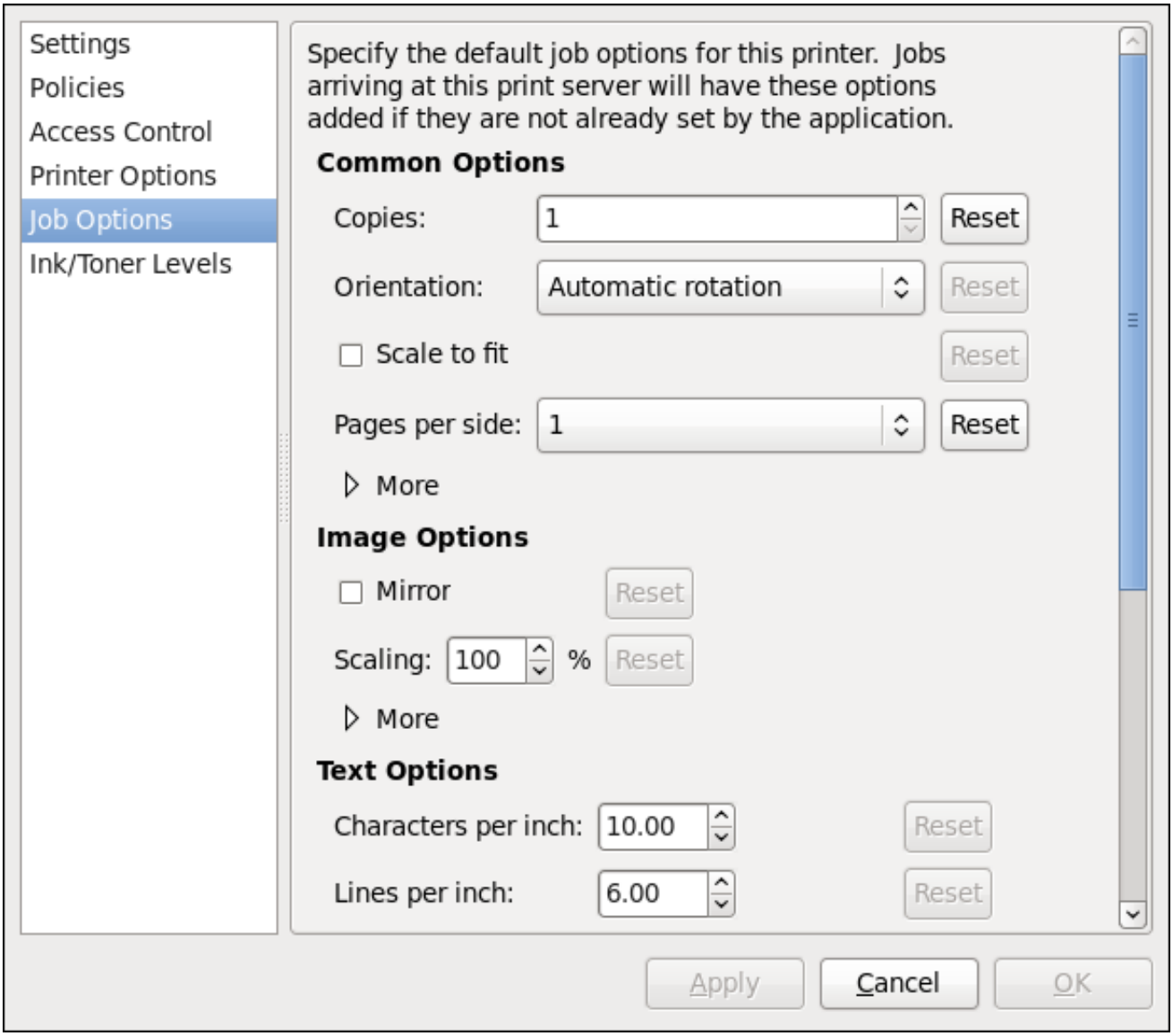


Figure 14.16. Page des options de tâches

14.3.10.2.5. Page des niveaux d'encre / du toner

La page **`Niveaux d'encre / du toner`** contient des détails sur le statut du toner si disponible et des messages sur le statut de l'imprimante. Veuillez cliquer sur l'étiquette **`Niveaux d'encre / du toner`** à gauche pour afficher la page.

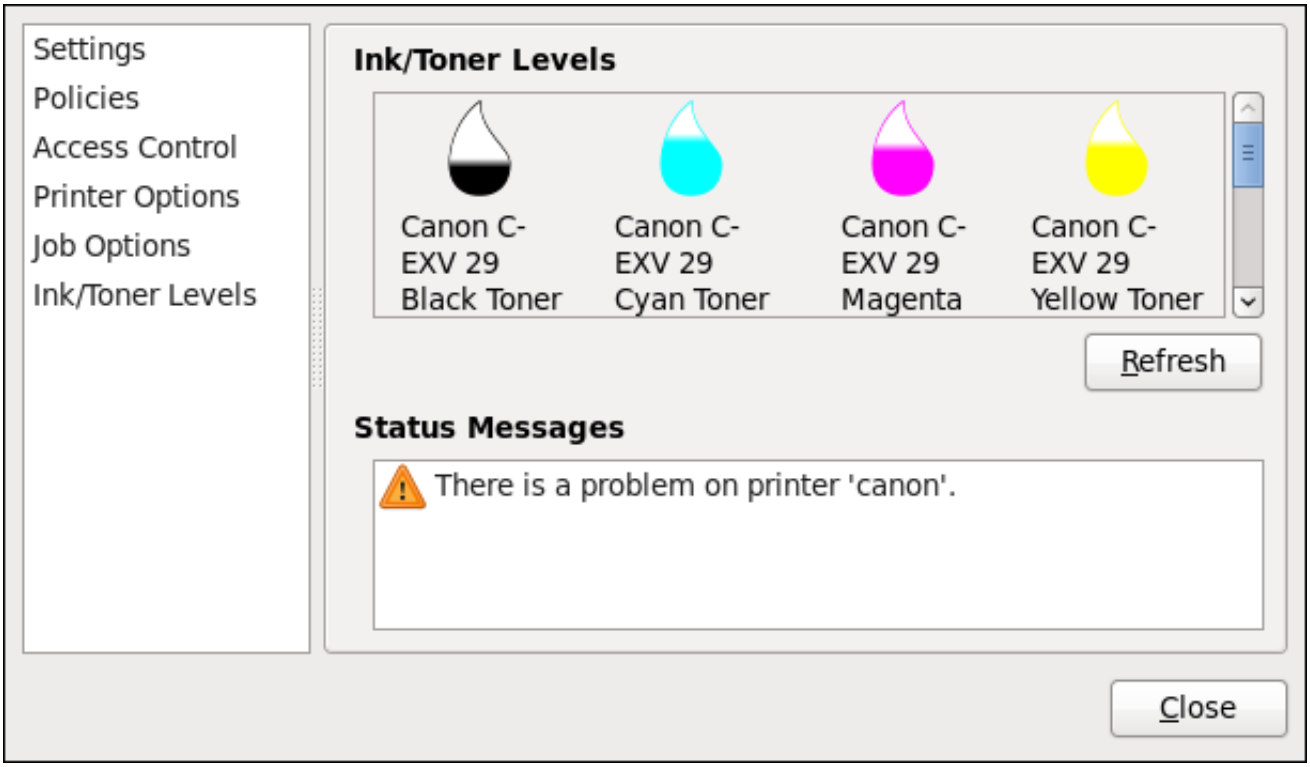


Figure 14.17. Page des niveaux d'encre / du toner

14.3.10.3. Gérer les tâches d'impression

Lorsqu'une tâche est envoyée au démon de l'imprimante, comme pour l'impression d'un fichier texte d'**Emacs** ou l'impression d'une image de **GIMP**, la tâche d'impression est ajoutée à la file d'attente spool d'impression. La file d'attente spool d'impression est une liste de tâches qui ont été envoyées à l'imprimante et d'informations sur chaque requête d'impression, comme le statut de la requête, le numéro de la tâche, etc.

Pendant le processus d'impression, l'icône **`Statut de l'imprimante`** apparaît dans la **Zone de notification** sur le panneau. Pour vérifier le statut d'une tâche d'impression, veuillez cliquer sur **`Statut de l'imprimante`**, qui affiche une fenêtre similaire à la Figure 14.18, « Statut de l'impression GNOME ».

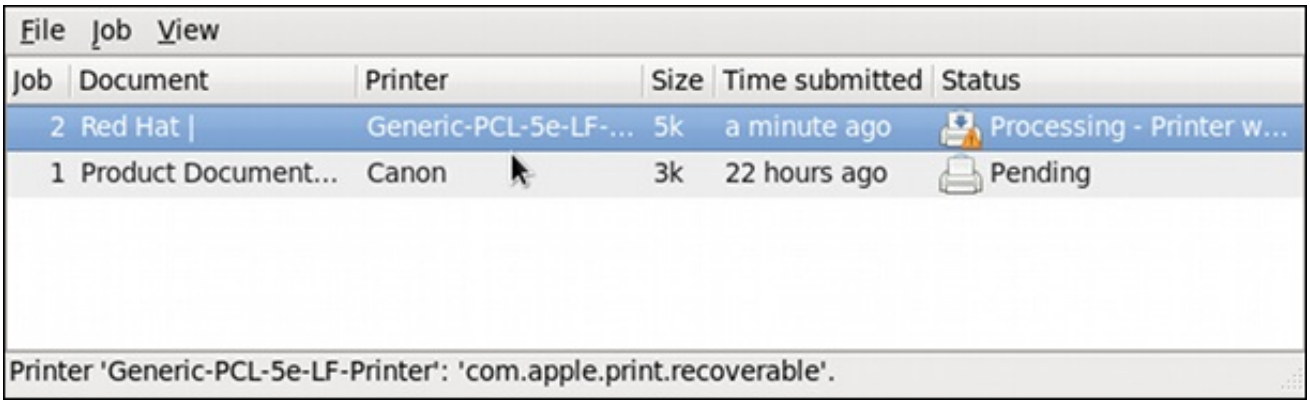


Figure 14.18. Statut de l'impression GNOME

Pour annuler, suspendre, libérer, réimprimer, ou authentifier une tâche d'impression, veuillez sélectionner la tâche dans **Statut d'impression GNOME** et dans le menu de **Tâche**, cliquez sur la commande respective.

Pour afficher une liste des tâches d'impressions dans le spool d'impression à partir d'une invite de shell, veuillez saisir la commande **lpstat -o**. Les quelques dernières lignes seront similaires à ce qui suit :

Exemple 14.4. Exemple de sortie de lpstat -o

```
$ lpstat -o
Charlie-60              twaugh            1024   Tue 08 Feb 2011
16:42:11 GMT
Aaron-61                twaugh            1024   Tue 08 Feb 2011
16:42:44 GMT
Ben-62                  root              1024   Tue 08 Feb 2011
16:45:42 GMT
```

Si vous souhaitez annuler une tâche d'impression, trouvez le numéro de la tâche avec la commande **lpstat -o**, puis utilisez la commande **cancel *numéro de tâche***. Par exemple, **cancel 60** annulera la tâche d'impression dans Exemple 14.4, « Exemple de sortie de **lpstat -o** ». Il n'est pas possible d'annuler des tâches d'impression lancées par d'autres utilisateurs avec la commande **cancel**. Cependant, il est possible de supprimer ces tâches en utilisant la commande **cancel -U root *numéro_de_tâche***. Pour empêcher ce type d'annulation, modifiez la politique d'opération de l'imprimante sur **Authenticated** (« Authentifié ») pour forcer l'authentification **root**.

Vous pouvez également imprimer un fichier directement depuis une invite de shell. Par exemple, la commande **lp sample.txt** imprimera le fichier texte **sample.txt**. Le filtre d'impression détermine le type de fichier dont il s'agit et le convertit sous un format compréhensible par l'imprimante.

14.3.11. Ressources supplémentaires

Pour en savoir plus sur l'impression sur Red Hat Enterprise Linux, veuillez consulter les ressources suivantes.

Documentation installée

- **lp(1)** — page du manuel de la commande **lp**, qui permet d'imprimer des fichiers à partir de la ligne de commande.
- **lpr(1)** — page du manuel de la commande **lpr**, qui permet d'imprimer des fichiers à partir de la ligne de commande.
- **cancel(1)** — page du manuel de l'utilitaire de ligne de commande pour supprimer des tâches de la file d'attente d'impression.
- **mpage(1)** — page du manuel de l'utilitaire de ligne de commande pour imprimer plsieurs pages sur une seule feuille.
- **cupsd(8)** — page du manuel du démon d'imprimante CUPS.
- **cupsd.conf(5)** — page du manuel du fichier de configuration du démon d'imprimante CUPS.
- **classes.conf(5)** — page du manuel pour le fichier de configuration de classe de CUPS.
- **lpstat(1)** — page du manuel de la commande **lpstat**, qui affiche des informations sur le statut des classes, tâches, et imprimantes.

Documentation en ligne

- http://www.linuxprinting.org/ — Le groupe OpenPrinting sur le site web Linux Foundation contient une grande quantité d'informations sur les impressions sur Linux.
- http://www.cups.org/ — Le site web CUPS fournit la documentation, des FAQ, et des groupes d'informations sur CUPS.

CHAPITRE 15. CONFIGURER NTP EN UTILISANT CHRONY SUITE

En informatique, un chronométrage précis est important pour un certain nombre de raisons. Dans les réseaux' par exemple, un horodatage précis des paquets et des journaux est requis. Sur les systèmes Linux, le protocole **NTP** est implémenté par un démon exécuté dans l'espace utilisateur.

Le démon de l'espace utilisateur met à jour l'horloge système exécutée dans le noyau. L'horloge système garde l'heure en utilisant diverses sources horaires. Habituellement, un compteur d'horodatage (« *Time Stamp Counter* », ou TSC) est utilisé. TSC est un enregistrement CPU qui compte le nombre de cycles depuis sa dernière réinitialisation. Ce compteur est très rapide, possède une haute résolution, et aucune interruption ne se produit.

Un choix se produit entre les démons **ntpd** et **chronyd**, qui se trouvent dans les référentiels des paquets ntp et chrony respectivement. Cette section décrit l'utilisation de la suite d'utilitaires **chrony** pour mettre à jour l'horloge système sur les systèmes n'entrant pas dans la catégorie conventionnelle des serveurs dédiés, toujours en cours de fonctionnement et sur le réseau de manière permanente.

15.1. INTRODUCTION À CHRONY SUITE

Chrony est composé de **chronyd**, un démon exécuté dans l'espace utilisateur, et de **chronyc**, un programme de ligne de commande pour effectuer des ajustements sur **chronyd**. Les systèmes qui ne sont pas connectés de manière permanente, ou qui ne sont pas alimentés de manière permanente, prennent un temps relativement long à ajuster leurs horloges système avec **ntpd**. Ceci est dû au fait que de nombreuses petites corrections sont effectuées sur la base des observations de dérive et de décalage des horloges. Les changements de température, qui peuvent être significatives lors du démarrage d'un système, affectent la stabilité des horloges matérielles. Même si les ajustements commencent dès les premières millisecondes du démarrage d'un système, la précision acceptable peut prendre entre zéro et dix secondes pour un redémarrage de type « warm » à plusieurs heures, selon les besoins, l'environnement d'exploitation, et le matériel. **chrony** est une implémentation du protocole **NTP** différente de **ntpd**, et peut ajuster l'horloge système plus rapidement.

15.1.1. Différences entre ntpd et chronyd

L'une des différences principales entre **ntpd** et **chronyd** réside dans les algorithmes utilisés pour contrôler l'horloge de l'ordinateur. Ce que **chronyd** peut faire mieux que **ntpd** inclut :

- **chronyd** peut bien fonctionner lorsque les références horaires externes sont accessibles de manière intermittente, tandis que **ntpd** a besoin d'interroger les références horaires régulièrement pour bien fonctionner.
- **chronyd** peut bien fonctionner même lorsque le réseau est encombré pendant de longues périodes.
- **chronyd** peut habituellement synchroniser l'horloge plus rapidement et avec une meilleure précision.
- **chronyd** s'adapte rapidement aux changements soudains de vitesse de l'horloge, par exemple lorsque ceux-ci sont dus à des changements de température de l'oscillateur à cristal, tandis que **ntpd** pourrait nécessiter plus de temps pour se réajuster.
- Dans la configuration par défaut, **chronyd** n'arrête jamais l'heure après la synchronisation de l'horloge pendant le démarrage système, afin de ne pas perturber les autres programmes en cours d'exécution. **ntpd** peut également être configuré de manière à ne jamais arrêter le temps,

mais doit utiliser différents moyens pour ajuster l'horloge, ce qui entraîne un certain nombre d'inconvénients.

- **chronyd** peut ajuster la vitesse de l'horloge sur un système Linux dans une plage plus importante, ce qui lui permet d'opérer même sur des machines ayant une horloge endommagée ou instable. Par exemple, sur certaines machines virtuelles.

Ce que **chronyd** peut faire, que **ntpd** ne peut pas faire :

- **chronyd** fournit la prise en charge des réseaux isolés, où l'unique méthode de correction du temps possible est manuelle, comme lorsque l'administrateur regarde l'horloge. **chronyd** peut examiner les erreurs corrigées lors de différentes mises à jour pour estimer la vitesse à laquelle l'ordinateur gagne ou perd du temps, et utilise ces estimations pour ajuster l'horloge système conséquemment.

- **chronyd** fournit la prise en charge pour calculer la quantité de gains ou pertes de temps de l'horloge temps réel, l'horloge matérielle, et maintient l'heure lorsque l'ordinateur est éteint. Ces données peuvent être utilisées lorsque le système démarre pour définir l'heure système à l'aide d'une valeur ajustée de l'heure prise de l'horloge temps réel. Cette fonctionnalité, au moment de la rédaction de ce guide, est uniquement disponible sur Linux.

Ce que **ntpd** peut faire, que **chronyd** ne peut pas faire :

- **ntpd** prend totalement en charge **NTP** version 4 (*RFC 5905*), y compris la diffusion, multidiffusion, et le Manycast des clients et serveurs, ainsi que le mode Orphelin. Il prend également en charge des schémas d'authentification supplémentaires basés sur chiffrement de clé publique (*RFC 5906*). **chronyd** utilise **NTP** version 3 (*RFC 1305*), qui est compatible avec la version 4.

- **ntpd** inclut des pilotes pour de nombreuses horloges de référence, tandis que **chronyd** repose sur d'autres programmes, comme par exemple **gpsd** pour accéder aux données des horloges de référence.

15.1.2. Choisir les démons NTP

- **Chrony** devrait être pris en considération pour tous les systèmes qui sont fréquemment suspendus ou déconnectés de manière intermittente puis reconnectés à un réseau, comme des systèmes virtuels et mobiles.

- Le démon **NTP** (**ntpd**) doit être considéré pour les systèmes habituellement en cours d'exécution de manière permanente. Les systèmes qui requièrent l'utilisation d'une adresse **IP** de diffusion ou de multidiffusion, ou qui effectuent l'authentification de paquets avec le protocole **Autokey** devraient envisager d'utiliser **ntpd**. **Chrony** prend uniquement en charge l'authentification de clés symétriques à l'aide d'un code d'authentification de message (MAC) avec MD5, SHA1 ou avec des fonctions de hachage plus robustes, tandis que **ntpd** prend également en charge le protocole d'authentification **Autokey**, qui peut utiliser le système PKI. **Autokey** est décrit dans *RFC 5906*.

15.2. COMPRENDRE CHRONY ET SA CONFIGURATION

15.2.1. Comprendre Chronyd

Le démon **chrony**, **chronyd**, exécuté dans l'espace utilisateur, ajuste l'horloge système exécutée dans le noyau. Ces ajustements sont effectués en consultant des sources de temps externes, en utilisant le

protocole **NTP**, chaque fois que l'accès réseau le permet. Lorsqu'aucune référence externe n'est disponible, **chronyd** utilisera la dernière dérive stockée dans le fichier de dérive. On peut également lui demander de faire des corrections manuelles, via **chronyc**.

15.2.2. Comprendre Chronyc

Le démon **chrony**, **chronyd**, peut être contrôlé par un utilitaire de ligne de commande, **chronyc**. Cet utilitaire fournit une invite de commande qui permet de saisir un nombre de commandes pour effectuer des changements sur **chronyd**. La configuration par défaut fait que **chronyd** accepte les commandes d'une instance locale de **chronyc**, mais **chronyc** peut être utilisé pour altérer la configuration, ainsi **chronyd** autorisera un contrôle externe. **chronyc** peut être exécuté à distance après avoir configuré **chronyd** pour accepter les connexions à distance. Les adresses **IP** autorisées à se connecter à **chronyd** doivent être minutieusement contrôlées.

15.2.3. Comprendre les commandes de configuration Chrony

Le fichier de configuration par défaut de **chronyd** est **/etc/chrony.conf**. L'option **-f** peut être utilisée pour spécifier un autre chemin de fichier de configuration. Veuillez consulter la page man **chronyd** pour obtenir des options supplémentaires. Pour obtenir une liste complète des directives pouvant être utilisées, veuillez consulter *http://chrony.tuxfamily.org/manual.html#Configuration-file*. Voici une sélection des options de configuration :

Commentaires

Les commentaires doivent être précédés de #, %, ; ou de !

autoriser

Optionnellement, spécifiez un hôte, un sous-réseau, ou un réseau à partir duquel autoriser les connexions **NTP** à une machine qui puisse agir en tant que serveur **NTP**. Par défaut, les connexions ne sont pas autorisées.

Exemples :

1. ```
   allow server1.example.com
   ```

   Veuillez utiliser ce format pour spécifier un hôte particulier, par son nom d'hôte, afin d'être autorisé à y accéder.

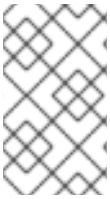
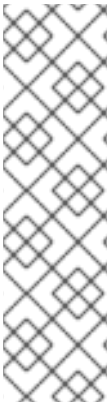
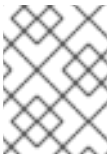
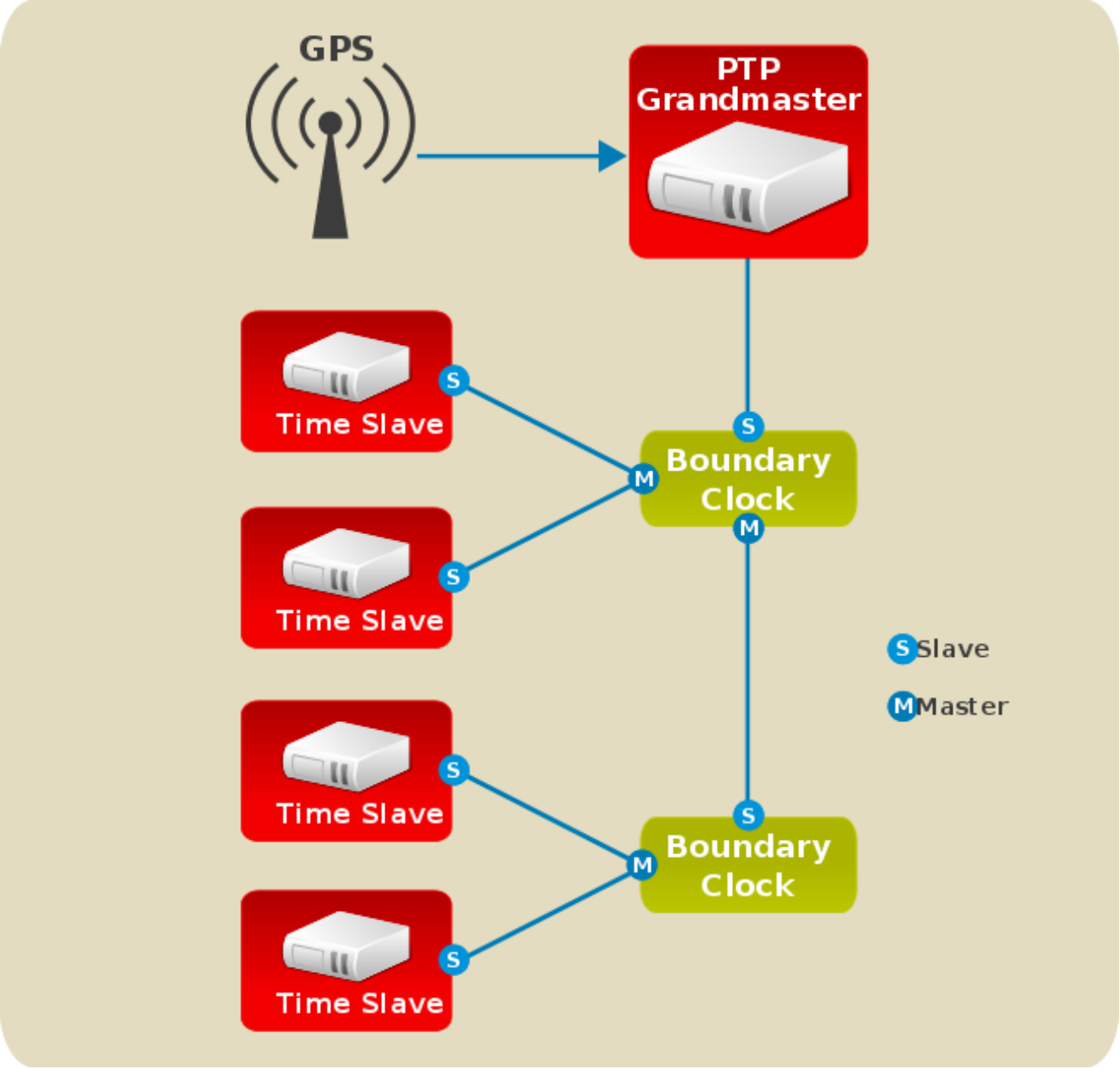
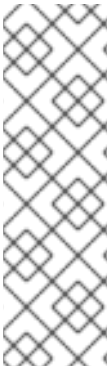
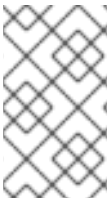
2. ```
   allow 192.0.2.0/24
   ```

   Veuillez utiliser ce format pour spécifier un réseau particulier afin d'être autorisé à y accéder.

3. ```
   allow 2001:db8::/32
   ```

   Veuillez utiliser ce format pour spécifier une adresse **IPv6** afin d'être autorisé à y accéder.

cmdallow

Similaire à la directive **allow** (veuillez consulter la section **allow**), sauf que l'accès de contrôle est autorisé (plutôt que l'accès client **NTP**) sur un sous-réseau ou un hôte particulier. (« accès de contrôle » signifie que **chronyc** peut être exécuté sur ces hôtes et connecté avec succès à **chronyd** sur cet ordinateur.) La syntaxe est identique. Il existe également une directive **cmddeny all** avec un comportement similaire à la directive **cmdallow all**.
   ```

dumpdir

Chemin vers le répertoire pour enregistrer l'historique des mesures à travers les redémarrages de **chronyd** (en supposant qu'aucun changement n'ait été appliqué au comportement de l'horloge système pendant qu'elle n'était pas en cours d'exécution). Si cette capacité doit être utilisée (via la commande **dumponexit** dans le fichier de configuration, ou la commande **dump** dans **chronyc**), la commande **dumpdir** doit être utilisée pour définir le répertoire dans lequel les historiques des mesures sont enregistrés.

dumponexit

Si cette commande est présente, elle infique que **chronyd** devrait enregistrer l'historique des mesure de chacune de ses sources horaire lorsque le programme se ferme. (Veuillez consulter la commande **dumpdir** ci-dessus).

local

Le mot-clé **local** est utilisé pour que **chronyd** puisse apparaître synchronisé avec le temps réel du point de vue des interrogations des clients, même s'il ne possède pas de source de synchronisation actuellement. Cette option est normalement utilisée sur l'ordinateur « maître » dans un réseau isolé, où il est requis que plusieurs ordinateurs soient synchronisés les uns aux autres, et où le « maître » est aligné au temps réel par saisie manuelle.

Voici un exemple de la commande :

```
local stratum 10
```

Une valeur élevée de 10 indique que l'horloge se trouve si loin de l'horloge de référence que son heure n'est pas fiable. Si l'ordinateur a accès à un autre ordinateur qui lui est synchronisé à une horloge de référence, celui-ci se trouvera à un stratum de moins de 10. Ainsi, le choix d'une valeur élevée, comme 10 pour la commande **local**, empêche à l'heure de la machine d'être confondue avec le temps réel, si celle-ci devait un jour se propager aux clients qui peuvent voir des serveurs réels.

log

La commande **log** indique que certaines informations doivent être journalisées. La commande accepte les options suivantes :

measurements

Cette option journalise les mesures **NTP** brutes et les informations les concernants dans un fichier nommé **measurements.log**.

statistics

Cette option journalise les informations concernant le traitement de régression sur un fichier nommé **statistics.log**.

tracking

Cette option journalise les changements apportés à l'estimation des taux de gains ou pertes du système, sur un fichier nommé **tracking.log**.

rtc

Cette option journalise les informations sur l'horloge temps réel du système.

refclocks

Cette option journalise les mesures brutes et filtrées des horloges de référence sur un fichier nommé **refclocks.log**.

tempcomp

Cette option journalise les mesures de température et les compensations de vitesse du système sur un fichier nommé **tempcomp.log**.

Les fichiers journaux sont écrits sur le répertoire spécifié par la commande **logdir**. Ci-dessous figure un exemple de la commande :

```
log measurements statistics tracking
```

logdir

Cette directive permet de spécifier le répertoire où les fichiers journaux sont écrits. Ci-dessous figure un exemple d'utilisation de cette directive :

```
logdir /var/log/chrony
```

makestep

Normalement, **chronyd** fera que le système corrigera peu à peu tout décalage horaire, en ralentissant ou en accélérant l'horloge selon les besoins. Dans certaines situations, l'horloge système peut avoir dérivé au point où le processus de correction risque de prendre un long moment avant de corriger l'horloge système. Cette directive force **chronyd** à modifier l'horloge système si cet ajustement est supérieur à la valeur maximale, mais seulement s'il n'y a pas eu de mise à jour de l'horloge depuis que **chronyd** a été lancé dans la limite spécifiée (une valeur négative peut être utilisée pour désactiver la limite). Ceci est particulièrement utile lors de l'utilisation d'horloges de référence, car la directive **initstepslew** fonctionne uniquement avec des sources **NTP**.

Ci-dessous figure un exemple d'utilisation de la directive :

```
makestep 1000 10
```

Cette directive modifierait l'horloge système si l'ajustement était plus important que 1000 secondes, mais seulement lors des dix premières mises à jour horaires.

maxchange

Cette directive paramètre le décalage maximal autorisé corrigé sur une mise à jour de l'horloge. La vérification est effectuée uniquement après le nombre indiqué de mises à jour afin de permettre un ajustement initial important de l'horloge système. Lorsqu'un décalage supérieur au maximum spécifié se produit, il sera ignoré pour le nombre de fois indiqué, puis **chronyd** abandonnera et quittera (une valeur négative peut être utilisée afin de ne jamais quitter). Dans les deux cas, un message sera envoyé à syslog.

Ci-dessous figure un exemple d'utilisation de la directive :

```
maxchange 1000 1 2
```

Après la première mise à jour de l'horloge, **chronyd** vérifiera le décalage sur chaque mise à jour de l'horloge, ignorera deux ajustements de plus de 1000 secondes et quittera lors d'un ajustement ultérieur.

maxupdateskew

L'une des tâches de **chronyd** consiste à trouver à quelle vitesse l'horloge de l'ordinateur fonctionne comparé à ses sources de référence. En outre, une estimation des limites de l'erreur est calculée autour de l'estimation. Si la marge d'erreurs est trop importante, cela indique que les mesures n'ont pas encore été paramétrées et que la vitesse estimée de gain ou de perte n'est pas très fiable. Le paramètre **maxupdateskew** correspond à la limite pour déterminer si une estimation n'est pas assez fiable pour être utilisée. Par défaut, la limite est de 1000 ppm. Le format de la syntaxe est comme suit :

```
maxupdateskew skew-in-ppm
```

Des valeurs typiques de *skew-in-ppm* peuvent s'élever à 100 pour une connexion téléphonique de serveurs à travers une ligne téléphonique, et à 5 ou 10 pour un ordinateur sur un réseau LAN. Remarquez qu'il ne s'agit pas de l'unique moyen de protection contre l'utilisation d'estimations non fiables. À tout moment, **chronyd** conserve la trace des vitesses estimées de gain ou de perte, et de la limite d'erreur de l'estimation. Lorsqu'une nouvelle estimation est générée suivant une autre mesure de l'une des sources, un algorithme de combinaison pondéré est utilisé pour mettre à jour l'estimation maître. Ainsi. si **chronyd** possède une estimation maître très fiable et qu'une nouvelle estimation est générée avec une marge d'erreurs supérieure, l'esimation maître existante prévaudra sur la nouvelle estimation maître.

noclientlog

Cette directive, qui ne reçoit aucun argument, indique que les accès client ne doivent pas être journalisés. Ceux-ci sont normalement journalisés, permettant ainsi aux statistiques d'être rapportées en utilisant la commande client dans **chronyc**.

reselectdist

Lorsque **chronyd** sélectionne une source de synchronisation parmi les sources disponibles, une source avec une distance de synchronisation minimale est préférable. Cependant, pour éviter une resélection fréquente lorsqu'il se trouve plusieurs sources à une distance similaire, une distance fixe est ajoutée à la distance des sources qui ne sont pas sélectionnées. Cette distance peut être définie avec l'option **reselectdist**. Par défaut, la distance s'élève à 100 microsecondes.

Le format de la syntaxe est comme suit :

```
reselectdist dist-in-seconds
```

stratumweight

La directive **stratumweight** définit la distance devant être ajoutée par stratum à la distance de synchronisation lorsque **chronyd** sélectionne la source de synchronisation parmi les sources disponibles.

Le format de la syntaxe est comme suit :

```
stratumweight dist-in-seconds
```

Par défaut, la valeur de *dist-in-seconds* s'élève à 1 seconde. Cela signifie que les sources possédant un stratum plus bas sont normalement préférées aux sources avec un stratum plus élevé, même lorsque leur distance est significativement plus élevée. Définir **stratumweight** sur 0 cause à **chronyd** d'ignorer le stratum lors de la sélection de la source.

rtcfile

La directive **rtcfile** définit le nom du fichier dans lequel **chronyd** peut enregistrer les paramètres associés au suivi de la précision de l'horloge temps réel (RTC) du système. Le format de la syntaxe est comme suit :

```
rtcfile /var/lib/chrony/rtc
```

chronyd enregistre les informations dans ce fichier lorsqu'il se ferme et quand la commande **writertc** est exécutée dans **chronyc**. Les informations enregistrées contiennent l'erreur de l'horloge RTC à une certaine époque, cette époque (en secondes depuis le 1er janvier 1970), et la vitesse à laquelle l'horloge RTC gagne ou perd du temps. Toutes les horloges RTC ne sont pas prises en charge car leur code est spécifique au système. Remarquez que si cette directive est utilisée, alors l'horloge RTC ne devrait pas être ajustée manuellement car cela interfère avec le besoin de **chrony** de mesurer la vitesse à laquelle l'horloge dérive si elle est ajustée à des intervalles aléatoires.

rtcsync

La directive **rtcsync** est présente dans le fichier **/etc/chrony.conf** par défaut. Cela informera le noyau que l'horloge système est synchronisée et que le noyau mettra à jour l'horloge RTC toutes les 11 minutes.

15.2.4. Sécurité avec Chronyc

Comme l'accès à **chronyc** permet de modifier **chronyd** de la même manière qu'on puisse le faire par une modifcation des fichiers de configuration, l'accès à **chronyc** doit être limité. Des mots de passe écrits en ASCII ou HEX peuvent être spécifiés dans le fichier clé pour restreindre l'utilisation de **chronyc**. L'une des entrées est utilisée pour restreindre l'utilisation de commandes opérationnelles et est désignée comme étant la commande clé. Dans la configuration par défaut, une clé de commande aléatoire est générée automatiquement lors du démarrage. Il ne devrait pas être nécessaire de la spécifier ou de l'altérer manuellement.

D'autres entrées dans le fichier clé peuvent être utilisées comme clés **NTP** pour authentifier les paquets reçus de serveurs ou de pairs **NTP** distants. Les deux côtés doivent partager une clé avec un ID, un type de hachage et un mot de passe identiques dans leurs fichier clé. Cela nécessite la création manuelle des clés, ainsi que de les copier au moyen d'un support sécurisé, tel que **SSH**. Ainsi, si l'ID clé était de 10, alors les systèmes agissant en tant que clients doivent avoir une ligne dans leurs fichiers de configuration sous le format suivant :

```
server w.x.y.z key 10
peer w.x.y.z key 10
```

L'emplacement du fichier clé est spécifié dans le fichier **/etc/chrony.conf**. L'entrée par défaut dans le fichier de configuration est comme suit :

```
keyfile /etc/chrony.keys
```

Le numéro de la clé de commande est spécifié dans **/etc/chrony.conf** à l'aide de la directive **commandkey**, il s'agit de la clé que **chronyd** utilisera pour l'authentification des commandes utilisateurs. La directive du fichier de configuration prend le format suivant :

```
commandkey 1
```

Un exemple du format de l'entrée par défaut dans le fichier clé, **/etc/chrony.keys**, pour la clé de commande est comme suit :

```
1 SHA1 HEX:A6CFC50C9C93AB6E5A19754C246242FC5471BCDF
```

Où **1** est l'ID clé, SHA1 est la fonction de hachage à utiliser, **HEX** est le format de la clé, et **A6CFC50C9C93AB6E5A19754C246242FC5471BCDF** est la clé générée de manière aléatoire lorsque **chronyd** a été lancé pour la première fois. La clé peut être donnée sous un format hexadécimal ou ASCII (par défaut).

Une entrée manuelle dans le fichier clé, utilisée pour authentifier les paquets de certains serveurs ou pairs **NTP**, peut être aussi simple que ce qui suit :

```
20 foobar
```

Où **20** est l'ID clé et **foobar** est la clé d'authentification secrète, le hachage par défaut est MD5, et ASCII est le format par défaut de la clé.

Par défaut, **chronyd** est configuré pour écouter uniquement les commandes en provenance de **localhost** (**127.0.0.1** et **::1**) sur le port **323**. Pour accéder à **chronyd** à distance avec **chronyc**, toutes les directives **bindcmdaddress** devront être supprimées du fichier **/etc/chrony.conf** afin de permettre l'écoute sur toutes les interfaces et la directive **cmdallow** devra être utilisée pour autoriser les commandes en provenance de l'adresse **IP** distante, ou du réseau ou sous-réseau distant. En outre, le port **323** doit être ouvert dans le pare-feu pour se connecter à partir un système distant. Remarquez que la directive **allow** est utilisée pour l'accès **NTP** tandis que la directive **cmdallow** sert à activer la réception de commandes distantes. Il est possible d'effectuer ces changements de manière temporaire en utilisant **chronyc** localement. Pour rendre les changements persistants, veuillez modifier le fichier de configuration.

Les communications entre **chronyc** et **chronyd** sont effectuées via **UDP**, qui doit donc être autorisé avant de pouvoir exécuter des commandes opérationnelles. Pour autoriser son utilisation, veuillez exécuter les commandes **authhash** et **password** comme suit :

```
chronyc> authhash SHA1
chronyc> password HEX:A6CFC50C9C93AB6E5A19754C246242FC5471BCDF
200 OK
```

Si **chronyc** est utilisé pour configurer le démon local **chronyd**, l'option **-a** exécutera les commandes **authhash** et **password** automatiquement.

Seules les commandes suivantes peuvent être utilisées sans mot de passe : **activity**, **authhash**, **dns**, **exit**, **help**, **password**, **quit**, **rtcdata**, **sources**, **sourcestats**, **tracking**, **waitsync** .

15.3. UTILISER CHRONY

15.3.1. Installer Chrony

La suite **chrony** est installée par défaut sur certaines versions de Red Hat Enterprise Linux 7. Si requis, exécutez la commande suivante en tant qu'utilisateur **root** pour vous assurer qu'elle soit effectivement installée :

```
~]# yum install chrony
```

L'emplacement par défaut du démon**chrony** est **/usr/sbin/chronyd**. L'utilitaire de ligne de commande sera installé sur **/usr/bin/chronyc**.

15.3.2. Vérifier le statut de Chronyd

Pour vérifier le statut de **chronyd**, veuillez exécuter la commande suivante :

```
~]$ systemctl status chronyd
chronyd.service - NTP client/server
   Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled)
   Active: active (running) since Wed 2013-06-12 22:23:16 CEST; 11h ago
```

15.3.3. Lancer Chronyd

Pour lancer **chronyd**, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl start chronyd
```

Pour s'assurer que **chronyd** soit lancé automatiquement lors du démarrage système, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl enable chronyd
```

15.3.4. Arrêter Chronyd

Pour arrêter **chronyd**, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl stop chronyd
```

Pour empêcher le lancement automatique de **chronyd** pendant le démarrage système, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl disable chronyd
```

15.3.5. Vérifier si Chrony est synchronisé

Pour vérifier si **chrony** est synchronisé, veuillez utiliser les commandes **tracking**, **sources**, et **sourcestats**.

15.3.5.1. Vérifier le suivi Chrony

Pour vérifier le suivi **chrony**, veuillez exécuter la commande suivante :

```
~]$ chronyc tracking
Reference ID    : 1.2.3.4 (a.b.c)
Stratum         : 3
Ref time (UTC)  : Fri Feb  3 15:00:29 2012
System time     : 0.000001501 seconds slow of NTP time
Last offset     : -0.000001632 seconds
RMS offset      : 0.000002360 seconds
Frequency       : 331.898 ppm fast
```

```
Residual freq   : 0.004 ppm
Skew            : 0.154 ppm
Root delay      : 0.373169 seconds
Root dispersion : 0.024780 seconds
Update interval : 64.2 seconds
Leap status     : Normal
```

Les champs sont comme suit :

Reference ID

Il s'agit de l'ID de référence et du nom (ou de l'adresse **IP**), si disponible, du serveur avec lequel l'ordinateur est actuellement synchronisé. S'il s'agit de **127.127.1.1**, cela signifie que l'ordinateur n'est synchronisé à aucune source externe et que le mode « local » est en cours d'exécution (via la commande locale dans **chronyc**, ou la directive **local** dans le fichier **/etc/chrony.conf** (veuillez consulter la section **local**)).

Stratum

Le stratum indique à combien de sauts d'un ordinateur avec une horloge de référence attachée nous nous trouvons. Ce type d'ordinateur est un ordinateur stratum-1, ainsi l'ordinateur de l'exemple se trouve à deux sauts (c'est-à-dire, a.b.c est un stratum-2 et est synchronisé à partir d'un stratum-1).

Ref time

Il s'agit de l'heure (UTC) à laquelle la dernière mesure en provenance de la source de référence a été traitée.

System time

Lors d'une opération normale, **chronyd** ne réajuste jamais l'horloge système, car tout saut dans le calendrier peut avoir des conséquences adverses pour certains programmes d'application. Au lieu de cela, toute erreur dans l'horloge système est corrigée en l'accélérant ou en la ralentissant légèrement jusqu'à ce que l'erreur soit résolue, puis celle-ci retourne à la vitesse normale. Une conséquence de cela est qu'il y aura une période pendant laquelle l'horloge système (comme lue par d'autres programmes utilisant l'appel système **gettimeofday()**, ou par la commande de date dans le shell) sera différente de l'estimation de l'heure actuelle réelle de **chronyd** (qui est rapportée sur les clients **NTP** lors de son exécution en mode serveur). La valeur rapportée sur cette ligne est la différence causée par cet effet.

Last offset

Décalage local estimé lors de la dernière mise à jour de l'horloge.

RMS offset

Moyenne à long terme de la valeur de décalage.

Frequency

La valeur « frequency » est la vitesse à laquelle l'horloge système serait erronée si **chronyd** ne la corrigeait pas. Celle-ci est exprimée en ppm (parties par million). Par exemple, une valeur de 1ppm signifie que lorsque l'horloge système est une seconde en avance, celle-ci a réellement avancé de 1,000001 secondes, de manière relative à l'heure réelle.

Residual freq

Ceci affiche la « fréquence résiduelle » de la source de référence actuellement sélectionnée. Celle-ci reflète la différence entre ce que la fréquence devrait être selon les indications des mesures de la

source de référence et la fréquence réellement utilisée. Le résultat ne correspond pas toujours zéro car une procédure d'ajustement est appliquée à la fréquence. Chaque fois qu'une mesure de la source de référence est obtenue et qu'une nouvelle fréquence résiduelle est calculée, la précision estimée de ce résidu est comparée à la précision estimée (consultez **skew**) de la valeur de la fréquence. Une moyenne pondérée est calculée pour la nouvelle fréquence, et les coefficients dépendent de la précision. Si les mesures de la source de référence suivent une tendance cohérente, le résidu se rapprochera de zéro petit à petit.

Skew

Estimation de la limite de l'erreur sur la fréquence.

Root delay

Total des délais de chemins réseau vers l'ordinateur stratum-1 à partir duquel l'ordinateur est finalement synchronisé. Dans certaines situations extrêmes, cette valeur peut être négative. (Cela peut se produire dans un arrangement de pairs symétriques dans lequel les fréquences des ordinateurs n'effectuent pas de suivi les unes des autres et où le délai réseau est très court relativement à la durée du délai d'exécution de chaque ordinateur.)

Root dispersion

Dispersion totale accumulée à travers tous les ordinateurs en retournant à l'ordinateur stratum-1 à partir duquel l'ordinateur est finalement synchronisé. La dispersion est dûe à la résolution de l'horloge système, aux variations des mesures des statistiques, etc.

Leap status

Statut de l'intercalaire, qui peut être « Normal », « Insert second » (insérer une seconde), « Delete second » (supprimer une seconde) ou « Not synchronized » (non synchronisé).

15.3.5.2. Vérifier les sources Chrony

La commande des sources affiche des informations sur les sources horaires actuelles accédées par **chronyd**. L'argument optionnel -v, qui signifie verbosité, peut être spécifié. Dans ce cas, des lignes de légende supplémentaires sont affichées comme rappel de la signification des colonnes.

```
~]$ chronyc sources
 210 Number of sources = 3
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
=====
#* GPS0                          0   4   377    11   -479ns[ -621ns] +/-
134ns
^? a.b.c                       2   6   377    23   -923us[ -924us] +/-
43ms
^+ d.e.f                       1   6   377    21  -2629us[-2619us] +/-
86ms
```

Les colonnes sont comme suit :

M

Ceci indique le mode la source. **^** signifie un serveur, **=** signifie un pair, et **#** indique une horloge de référence connectée localement.

S

Cette colonne indique l'état des sources. « * » indique la source avec laquelle **chronyd** est actuellement synchronisé. « + » indique les sources acceptables qui sont combinées avec la source sélectionnée. « - » indique les sources acceptables qui sont exclues de l'algorithme de combinaison. « ? » indique les sources dont la connectivité a été perdue ou dont les paquets n'ont pas passé tous les tests. « x » indique une horloge que **chronyd** croit être de type *falseticker* (son heure est incohérente avec la majorité des autres sources). « ~ » indique une source dont la variabilité de l'heure semble trop importante. La condition « ? » est également affichée lors du démarrage, jusqu'à ce qu'au moins 3 échantillons en soient prélevés.

Name/IP address

Affiche le nom ou l'adresse **IP** de la source, ou l'ID de référence pour les horloges de référence.

Stratum

Affiche le stratum de la source ainais noté dans l'échantillon reçu le plus récemment. Stratum 1 indique un ordinateur avec une horloge de référence attachée localement. Un ordinateur synchronisé avec un ordinateur stratum 1 se trouve au stratum 2. Un ordinateur synchronisé avec un ordinateur stratum 2 se trouve au stratum 3, et ainsi de suite.

Poll

Affiche la vitesse à laquelle la source est sondée, en tant que logarithme de base 2 de l'intervalle en secondes. Ainsi, une valeur de 6 indiquerait qu'une mesure est prise toutes les 64 secondes. **chronyd** fait varier automatiquement la vitesse de sondage en réponse aux conditions actuelles.

Reach

Affiche l'enregistrement de la portée de la source imprimé en tant que nombre octal. L'enregistrement compte 8 bits et est mis à jour chaque fois qu'un paquet est reçu ou n'a pas été reçu de la source. Une valeur de 377 indique qu'une réponse valide a été reçue pour chacune des huit dernières transmissions.

LastRx

Cette colonne affiche le temps écoulé depuis que le dernier échantillon a été reçu de la source. Cette valeur est généralement exprimée en secondes. Les lettres **m**, **h**, **d** ou **y** indiquent les minutes, heures, jours ou années. Une valeur de 10 ans indique qu'aucun échantillon n'a encore été reçu de cette source.

Last sample

Cette colonne affiche le décalage entre l'horloge locale et la source lors de la dernière mesure. Le chiffre entre les crochets montre le décalage mesuré réel. Celui-ci peut être suivi du suffixe **ns** (indiquant des nanosecondes), **us** (indiquant des microsecondes), **ms** (indiquant des millisecondes), ou **s** (indiquant des secondes). Le nombre à gauche des crochets affiche la mesure d'origine ajustée pour permettre tout changement appliqué à l'horloge locale. Le nombre suivant qui se trouve après le marqueur **+/-** affiche la marge d'erreur de la mesure. Des décalages positifs indiquent que l'horloge locale est en avance comparé à la source.

15.3.5.3. Vérifier les statistiques des sources Chrony

La commande **sourcestats** affiche des informations sur le taux de dérive et sur le processus d'estimation de décalage de chacune des sources actuellement examinées par **chronyd**. On peut spécifier l'argument optionnel **-v** qui indique une sortie verbeuse. Dans ce cas, des lignes de légende

supplémentaires sont affichées pour rappeler ce que les colonnes signifient.

```
~]$ chronyc sourcestats
210 Number of sources = 1
Name/IP Address            NP  NR  Span  Frequency  Freq Skew  Offset
Std Dev
==============================================================================
=====
abc.def.ghi                11   5   46m     -0.001      0.045      1us
25us
```

Les colonnes sont comme suit :

Name/IP address

Nom ou adresse **IP** du serveur (ou pair) **NTP** ou de l'ID de référence de l'horloge de référence à laquelle le reste de la ligne se réfère.

NP

Nombre de points des échantillons actuellement conservés pour le serveur. Le taux de dérive et décalage actuel sont estimés en effectuant une régression linéaire à travers ces points.

NR

Nombre de courses des résidus possédant le même signe suivant la dernière régression. Si ce nombre devient trop bas par rapport au nombre d'échantillons, cela indique qu'une ligne droite ne conviendra plus aux données. Si le nombre de courses est trop bas, **chronyd** abandonnera les échantillons plus anciens et ré-exécutera la régression jusqu'à ce que le nombre de courses redevienne acceptable.

Span

Intervalle entre les échantillons les plus anciens et les plus récents. Si aucune unité n'est affichée, la valeur est en secondes. Dans l'exemple, l'intervalle est égale à 46 minutes.

Frequency

Fréquence résiduelle estimée du serveur, en parties par million. Dans ce cas, il est estimé que l'horloge de l'ordinateur exécute 1 partie sur 10^9 plus lentement que le serveur.

Freq Skew

Limites des erreurs estimées sur Freq (en partie par millon).

Offset

Estimation du décalage de la source.

Std Dev

Estimation de la déviation standard de l'échantillon

15.3.6. Ajuster l'horloge système manuellement

Pour mettre à jour, modifier, l'horloge système immédiatement, en outrepassant tout ajustement en cours, veuillez exécuter les commandes suivantes en tant qu'utilisateur **root** :

-

```
~]# chronyc
chrony> password commandkey-password
200 OK
chrony> makestep
200 OK
```

Où *commandkey-password* est la clé de commande ou le mot de passe stocké(e) dans le fichier clé.

Si la directive **rtcfile** est utilisée, l'horloge temps réel ne devrait pas être ajustée manuellement. Des ajustements aléatoires interféreront avec le besoin de **chrony** de mesurer le taux auquel l'horloge temps réel dérive.

Si **chronyc** est utilisé pour configurer le démon **chronyd** local, **-a** exécutera les commandes **authhash** et **password** automatiquement. Cela signifie que la session interactive illustrée ci-dessus peut être remplacée par :

```
chronyc -a makestep
```

15.4. PARAMÉTRER CHRONY POUR DIFFÉRENTS ENVIRONNEMENTS

15.4.1. Paramétrer Chrony pour un système rarement connecté

Cet exemple est conçu pour les systèmes utilisant des connexions à la demande. La configuration normale devrait être suffisante pour les périphériques mobiles et virtuels qui se connectent de manière intermittente. Premièrement, veuillez examiner et confirmer que les paramètres par défaut du fichier **/etc/chrony.conf** sont similaires à :

```
driftfile /var/lib/chrony/drift
commandkey 1
keyfile /etc/chrony.keys
```

L'ID de la clé de commande est générée pendant l'installation et doit correspondre à la valeur **commandkey** dans le fichier des clés, **/etc/chrony.keys**.

En utilisant votre éditeur en tant qu'utilisateur **root**, veuillez ajouter les adresses de quatre serveurs **NTP** comme suit :

```
server 0.pool.ntp.org offline
server 1.pool.ntp.org offline
server 2.pool.ntp.org offline
server 3.pool.ntp.org offline
```

L'option **offline** peut être utile pour empêcher les systèmes de tenter d'activer des connexions. Le démon **chrony** attendra que **chronyc** l'informe si le système est connecté au réseau ou à Internet.

15.4.2. Paramétrer Chrony pour un système dans un réseau isolé

Pour un réseau qui n'est jamais connecté à Internet, un ordinateur est sélectionné pour être le serveur de temps maître. Les autres ordinateurs sont des clients directs du maître, ou des clients des clients. Sur le maître, le fichier de dérive doit être paramétré manuellement avec le taux moyen de dérive de l'horloge système. Si le maître est réinitialisé, il obtiendra l'heure depuis les systèmes environnants et

utilisera une moyenne pour paramétrer son horloge système. Puis il recommencera à appliquer les ajustements basés sur le fichier de dérive. Le fichier de dérive sera mis à jour automatiquement lorsque la commande **settime** est utilisée.

Sur le système sélectionné, pour être le maître, en utilisant un éditeur de texte exécuté en tant qu'utilisateur **root**, veuillez modifier le fichier **/etc/chrony.conf** comme suit :

```
driftfile /var/lib/chrony/drift
commandkey 1
keyfile /etc/chrony.keys
initstepslew 10 client1 client3 client6
local stratum 8
manual
allow 192.0.2.0
```

Quand **192.0.2.0** est l'adresse du réseau ou du sous-réseau à partir de laquelle les clients sont autorisés à se connecter.

Sur les systèmes sélectionnés pour être des clients directs du maître, en utilisant un éditeur de texte exécuté en tant qu'utilisateur **root**, veuillez modifier le fichier **/etc/chrony.conf** comme suit :

```
server master
driftfile /var/lib/chrony/drift
logdir /var/log/chrony
log measurements statistics tracking
keyfile /etc/chrony.keys
commandkey 24
local stratum 10
initstepslew 20 master
allow 192.0.2.123
```

Quand **192.0.2.123** est l'adresse du maître, et **master** est le nom d'hôte du maître. Les clients avec cette configuration resynchroniseront le maître s'il redémarre.

Sur les systèmes client qui ne sont pas des clients directs du maître, le fichier **/etc/chrony.conf** devrait être le même, sauf que les directives **local** et **allow** doivent être omises.

15.5. UTILISER CHRONYC

15.5.1. Utiliser Chronyc pour contrôler Chronyd

Pour effectuer des changements sur l'instance locale de **chronyd** en utilisant l'utilitaire de ligne de commande **chronyc** en mode interactif, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# chronyc -a
```

Si certaines commandes restreintes sont utilisées, **chronyc** devra être exécuté par un utilisateur **root**. L'option **-a** est utilisée pour l'authentification automatique à l'aide de clés locales lors de la configuration de **chronyd** sur le système local. Veuillez consulter la Section 15.2.4, « Sécurité avec Chronyc » pour obtenir davantage d'informations.

L'invite de commande **chronyc** sera affichée comme suit :

```
chronyc>
```

Vous pouvez saisir **help** pour répertorier toutes les commandes.

L'utilitaire peut également être invoqué dans un mode de commande non-interactif si appelé avec une commande comme suit :

```
chronyc command
```



NOTE

Les changements effectués à l'aide de **chronyc** ne sont pas permanents et seront perdus après un redémarrage de **chronyd**. Pour appliquer des changements de manière permanente, veuillez modifier le fichier **/etc/chrony.conf**.

15.5.2. Utiliser Chronyc pour administrer à distance

Pour configurer **chrony** pour se connecter à une instance distante de **chronyd**, veuillez exécuter une commande sous le format suivant :

```
~]$ chronyc -h hostname
```

Quand *hostname* est le nom d'hôte auquel se connecter. Se connecte par défaut au démon local.

Pour configurer **chrony** pour se connecter à une instance distante de **chronyd** sur un port qui n'est pas celui par défaut, veuillez exécuter une commande sous le format suivant :

```
~]$ chronyc -h hostname -p port
```

Quand *port* est le port en cours d'utilisation pour le contrôle et la surveillance effectué par l'instance distante de **chronyd**.

Remarques que les commandes exécutées dans l'invite de commande **chronyc** ne sont pas persistantes. Seules les commandes dans le fichier de configuration sont persistantes.

La première commande doit être la commande **password** sur l'invite de commande **chronyc** comme suit :

```
chronyc> password password
200 OK
```

Le mot de passe ne doit pas contenir d'espaces.

Si le mot de passe n'est pas un hachage MD5, le mot de passe haché doit être précédé par la commande **authhash** comme suit :

```
chronyc> authhash SHA1
chronyc> password HEX:A6CFC50C9C93AB6E5A19754C246242FC5471BCDF
200 OK
```

Le mot de passe ou code de hachage associé à la clé de commande pour un système distant s'obtient mieux avec **SSH**. Une connexion **SSH** devrait être établie sur la machine distante et les ID de la clé de commande du fichier **/etc/chrony.conf** et de la clé de commande dans le fichier

`/etc/chrony.keys` seront mémorisés ou stockés pendant la durée de la session.

15.6. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes offrent des ressources supplémentaires concernant **chrony**.

15.6.1. Documentation installée

- Page man `chrony(1)` — présente le démon **chrony** et l'outil de l'interface de ligne de commande.
- Page man `chronyc(1)` — décrit l'outil de ligne de commande **chronyc** y compris les commandes et options de commandes.
- Page man `chronyd(1)` — décrit le démon **chronyd** y compris les commandes et options de commandes.
- Page man `chrony.conf(5)` — décrit le fichier de configuration **chrony**.
- `/usr/share/doc/chrony*/chrony.txt` — Guide de l'utilisateur de la suite **chrony**.

15.6.2. Documentation en ligne

http://chrony.tuxfamily.org/manual.html

Guide de l'utilisateur **chrony** en ligne.

CHAPITRE 16. CONFIGURER NTP À L'AIDE DE NTPD

16.1. INTRODUCTION À NTP

Le protocole d'heure réseau (de l'anglais *Network Time Protocol*, ou NTP) permet la dissémination précise d'informations sur l'heure et la date afin que les horloges système d'ordinateurs en réseau soient synchronisés avec une référence commune sur un réseau ou sur l'Internet. De nombreux organismes de standards de par le monde possèdent des horloges atomiques pouvant être utilisées comme références. Les satellites composant le système de positionnement global (« Global Position System », ou GPS) contiennent plus d'une horloge atomique, rendant ainsi leurs signaux très précis. Ces signaux peuvent être délibérément dégradés pour des raisons militaires. Une situation idéale serait que chaque site possède un serveur, avec sa propre horloge de référence attachée, pour agir en tant que serveur de temps pour l'ensemble du site. Il existe de nombreux périphériques qui obtiennent l'heure et la date via des transmissions radio à basses fréquences ou via GPS. Cependant dans la plupart des situations, un certain nombre de serveurs de temps publiquement accessibles connectés à l'Internet et géographiquement dispersés peuvent être utilisés. Ces serveurs **NTP** fournissent le temps universel coordonné « *Coordinated Universal Time* » (UTC). Des informations sur ces serveurs de temps se trouvent sur *www.pool.ntp.org*.

En informatique, un chronométrage précis est important pour un certain nombre de raisons. Dans les réseaux, par exemple, un horodatage précis des paquets et des journaux est requis. Les journaux sont utilisés pour examiner les problèmes de service et de sécurité, ainsi des horodatages effectués sur différents systèmes doivent être effectués par des horloges synchronisées pour réellement avoir toute valeur. Comme les systèmes et les réseaux sont de plus en plus rapides, il existe un besoin grandissant pour des horloges offrant une plus grande précision et une résolution plus importante. Dans certains pays, il existe des obligations légales pour que les horloges soient minutieusement synchronisées. Veuillez consulter *www.ntp.org* pour obtenir davantage d'informations. Sur les systèmes Linux, **NTP** est implémenté par un démon exécuté dans l'espace utilisateur. Le démon de l'espace utilisateur **NTP** par défaut dans Red Hat Enterprise Linux 7 est nommé **chronyd**. Il doit être désactivé si vous souhaitez utiliser le démon **ntpd**. Veuillez consulter le Chapitre 15, *Configurer NTP en utilisant Chrony Suite* pour obtenir davantage d'informations sur **chrony**.

Le démon de l'espace utilisateur met à jour l'horloge système, qui est une horloge logicielle exécutée dans le noyau. Linux utilise une horloge logicielle comme horloge système pour obtenir une meilleure résolution que l'horloge matérielle habituellement intégrée, aussi appelée horloge temps réel (de l'anglais, « *Real Time Clock* », ou RTC). Veuillez consulter les pages man **rtc(4)** et **hwclock(8)** pour obtenir des informations sur les horloges matériel. L'horloge système peut rester à l'heure en utilisant diverses sources horaires. Habituellement, le *Time Stamp Counter* (TSC) est utilisé. Le TSC est un enregistrement du processeur qui compte le nombre de cycles depuis la dernière réinitialisation. Celui-ci est très rapide, possède une haute résolution, et aucune interruption ne se produit. Lors du démarrage système, l'horloge système lit l'heure et la date à partir de l'heure RTC. L'horloge RTC peut se décaler de l'heure réelle jusqu'à 5 minutes par mois à cause des variations de température. D'où le besoin pour l'horloge système d'être constamment synchronisée avec des références horaires externes. Lorsque l'horloge système est synchronisée par **ntpd**, le noyau met automatiquement à jour l'heure RTC toutes les 11 minutes.

16.2. STRATUM NTP

Les serveurs **NTP** sont classés en fonction de leur distance de synchronisation avec les horloges atomiques qui sont la source des signaux horaires. Les serveurs sont arrangés en couches, ou stratums, du stratum 1 en haut, jusqu'au stratum 15 en bas. Ainsi le mot stratum est utilisé pour faire référence à une couche particulière. Les horloges atomiques sont appelées Stratum 0 car elles sont la source, mais aucun paquet Stratum 0 n'est envoyé sur l'Internet, toutes les horloges atomiques stratum 0 sont attachées à un serveur appelé stratum 1. Ces serveurs envoient des paquets marqués Stratum 1.

Un serveur synchronisé au moyen de paquets marqués stratum **n** appartient au stratum suivant, plus bas, et marquera ses paquets en tant que stratum **n+1**. Les serveurs du même stratum peuvent échanger des paquets mais sont toujours désignés comme appartenant à un seul stratum, le stratum se trouvant sous la meilleure référence de synchronisation. L'appellation Stratum 16 est utilisée pour indiquer que le serveur n'est pas actuellement synchronisé à une source horaire fiable.

Remarquez que, par défaut, les clients **NTP** agissent en tant que serveurs dans le stratum se trouvant en-dessous.

Ci-dessous figure le sommaire des stratums **NTP** :

Stratum 0 :

Les horloges atomiques et leurs signaux sont diffusés via Radio et GPS

- GPS (« Global Positioning System »)
- Systèmes de téléphonie mobile
- Transmissions Radio à basses fréquences WWVB (Colorado, USA.), JJY-40 et JJY-60 (Japon), DCF77 (Allemagne), et MSF (Royaume-Uni)

Ces signaux peuvent être reçus par des périphériques dédiés et sont habituellement connectés par RS-232 à un système utilisé comme serveur d'heure organisationnel ou pour l'ensemble du site.

Stratum 1 :

Ordinateur avec une horloge radio, une horloge GPS, ou une horloge atomique attachée

Stratum 2 :

Lit le contenu du stratum 1, et le sert au stratum en-dessous

Stratum 3 :

Lit le contenu du stratum 2, et le sert au stratum en-dessous

Stratum *n+1* :

Lit le contenu du stratum *n*, et le sert au stratum en-dessous

Stratum 15 :

Lit le contenu du stratum 14, il s'agit du stratum le plus bas.

Ce processus continue jusqu'au Stratum 15, qui est le stratum valide le plus bas. L'étiquette Stratum 16 est utilisée pour indiquer un état non synchronisé.

16.3. COMPRENDRE NTP

La version de **NTP** utilisée par Red Hat Enterprise Linux est comme celle qui est décrite dans la *RFC 1305 Network Time Protocol (Version 3) Specification, Implementation and Analysis* et *RFC 5905 Network Time Protocol Version 4: Protocol and Algorithms Specification*

Cette implémentation de **NTP** permet d'atteindre une précision à la fraction de seconde. Sur Internet, une précision au dixième de seconde est normale. Sur un réseau LAN, une précision au millième de seconde est possible dans des conditions idéales. Ceci est dû au fait que la dérive horaire est désormais prise en

charge et corrigée, ce qui n'était pas le cas sur les systèmes de protocoles de temps plus anciens, qui étaient plus simples. Une résolution de 233 picosecondes est offerte avec l'utilisation d'horodatages de 64 bits. Les premiers 32 bits de l'horodatage sont utilisés pour les secondes, et les derniers 32 bits sont utilisés pour les fractions de seconde.

NTP représente l'heure en tant que total du nombre de secondes depuis 00:00 (minuit) le 1er janvier, 1900 GMT. Comme 32 bits sont utilisés pour compter les secondes, cela signifie que l'heure devra être « basculée » en 2036. Cependant, **NTP** fonctionne avec les différences entre horodatages, ceci ne présente donc pas le même niveau de problème que les autres implémentations de protocole de temps ont pu offrir. Si une horloge matérielle se trouvant à 68 ans de l'heure correcte est disponible au moment du démarrage, alors **NTP** interprétera correctement la date actuelle. La spécification **NTP4** fournit un numéro d'ère (« « Era Number » ») et un décalage d'ère (« « Era Offset » ») qui peuvent être utilisés pour rendre le logiciel plus robuste lorsqu'il doit gérer des durées de plus de 68 ans. Remarque : veuillez ne pas confondre ceci avec le problème Unix de l'an 2038.

Le protocole **NTP** fournit des informations supplémentaires pour améliorere la précision. Quatre horodatages sont utilisés pour permettre le calcul du temps d'un aller-retour et du temps de réponse du serveur. Dans son rôle de client **NTP**, pour qu'un système puisse se synchroniser avec un serveur de temps référence, un paquet est envoyé avec un « horodatage d'origine ». Lorsque le paquet arrive, le serveur de temps ajoute un « horodatage de réception ». Après avoir traité la requête d'informations sur l'heure et la date et juste avant de retourner le paquet, un « horodatage de transmission » est ajouté. Lorsque le paquet retourné arrive sur le client **NTP**, un « horodatage de réception » est généré. Le client peut désormais calculer le temps d'aller-retour total et dériver le temps de transfert réel en soustrayant le temps de traitement. En supposant que les temps de transfert aller et retour soient égaux, le délai d'un seul transfert lors de la réception des données **NTP** est calculé. L'algorithme **NTP** est bien plus complexe que ce qui est présenté ici.

Lorsqu'un paquet contenant des informations sur le temps est reçu, il n'est pas immédiatement pris en charge, il est tout d'abord sujet à des vérifications de validation, puis il est traité avec d'autres échantillons de temps pour arriver à une estimation de l'heure. Celle-ci est ensuite comparée à l'horloge système pour déterminer le décalage horaire, la différence entre l'heure de l'horloge système et l'heure que **ntpd** a déterminé. L'horloge système est ajustée lentement, au maximum à une vitesse de 0,5 ms par seconde, afin de réduire ce décalage en changeant la fréquence du compteur utilisé. Ainsi, un minimum de 2000 secondes est nécessaire pour ajuster l'horloge d'une seconde en utilisant cette méthode. Ce lent changement est appelé « Slewing » (changement d'angle d'orientation) et ne peut pas être inversé. Si le décalage de l'horloge est supérieur à 128 ms (paramètre par défaut), **ntpd** peut « incrémenter » l'horloge plus haut ou plus bas. Si le décalage est supérieur à 1000 seconde, alors l'utilisateur ou un script d'installation devra effectuer un ajustement manuel. Veuillez consulter le Chapitre 2, *Configurer l'heure et la date*. Avec l'option **-g** ajoutée à commande **ntpd** (utilisée par défaut), tout décalage lors du démarrage système sera corrigé, mais pendant une opération normale, seuls les décalages de plus de 1000 secondes seront corrigés.

Certains logiciels peuvent échouer ou produire une erreur si l'heure est modifiée en arrière. Pour les systèmes sensibles aux changements d'heure, la limite peut être changée sur 600 s. au lieu de 128 ms., en utilisant l'option **-x** (qui n'est pas liée à l'option **-g**). L'utilisation de l'option **-x** pour augmenter la limite de réajustement 0,128 s. à 600 s. présente un inconvénient car une méthode différente de contrôle de l'horloge doit être utilisée. Celle-ci désactive la discipline de l'horloge du noyau et peut avoir un impact négatif sur la précision de l'horloge. L'option **-x** peut être ajoutée au fichier de configuration **/etc/sysconfig/ntpd**.

16.4. COMPRENDRE LE FICHIER DE DÉRIVE

Le fichier de dérive est utilisé pour stocker le décalage de fréquence entre l'horloge système exécutée à sa fréquence nominale et la fréquence requise pour rester en synchronisation avec l'heure UTC. Si présente, la valeur contenue dans le fichier de dérive est lue pendant le démarrage système et est

utilisée pour corriger la source horaire. L'utilisation du fichier de dérive réduit le temps requis pour atteindre une heure stable et précise. La valeur est calculée et le fichier de dérive est remplacé une fois par heure par **ntpd**. Le fichier de dérive est remplacé plutôt que simplement mis à jour, et pour cette raison, le fichier de dérive se trouve dans un répertoire sur lequel **ntpd** possède des permissions d'écriture.

16.5. UTC, FUSEAUX HORAIRES, ET HEURE D'ÉTÉ

Comme **NTP** est entièrement en temps universel coordonné (UTC), les fuseaux horaires et l'heure d'été sont appliqués localement par le système. Le fichier **/etc/localtime** est une copie, ou un lien symbolique du fichier d'informations sur le fuseau **/usr/share/zoneinfo**. L'horloge temps réel (ou RTC) peut se trouver en heure locale ou en UTC, comme spécifié par la troisième ligne de **/etc/adjtime**, qui sera LOCAL ou UTC pour indiquer comment l'horloge RTC a été paramétrée. Les utilisateurs peuvent facilement modifier ce paramètre en cochant la case **L'horloge système utilise UTC** dans l'outil de configuration graphique **Date et heure**. Veuillez consulter le Chapitre 2, *Configurer l'heure et la date* pour obtenir des informations sur la manière d'utiliser cet outil. Il est recommandé d'exécutier l'horloge RTC avec le temps UTC pour éviter divers problèmes lorsque l'heure d'été change.

L'opération de **ntpd** est expliquée en détails sur la page man de **ntpd(8)**. La section des ressources répertorie des sources d'informations utiles. Veuillez consulter la Section 16.20, « Ressources supplémentaires ».

16.6. OPTIONS D'AUTHENTIFICATION NTP

NTPv4 ajoute la prise en charge de l'architecture de sécurité Autokey (« Autokey Security Architecture »), qui est basée sur un chiffrement asymétrique public tout en conservant la prise en charge du chiffrement de clés symétriques. L'architecture de sécurité Autokey est décrite dans la page *RFC 5906 Network Time Protocol Version 4: Autokey Specification*. La page man de **ntp_auth(5)** décrit les options et commandes d'authentification de **ntpd**.

Une personne malveillante peut tenter de perturber un service en envoyant des paquets **NTP** avec des informations horaires incorrectes. Sur les systèmes utilisant le pool public des serveurs **NTP**, ce risque peut être réduit en ayant plus de trois serveurs **NTP** dans la liste des serveurs **NTP** publics située dans **/etc/ntp.conf**. Si une seule source horaire est compromise ou usurpée, **ntpd** ignorera cette source. Vous devriez effectuer une évaluation des risques et prendre en considération l'impact de l'heure incorrecte sur vos applications et votre organisation. Si vous possédez des sources horaires internes, vous devriez envisager de prendre des mesures pour protéger le réseau sur lequel les paquets **NTP** sont distribués. Si vous effectuez une évaluation des risques et concluez que le risque est acceptable, et que l'impact sur vos applications est minime, alors vous pouvez choisir de ne pas utiliser l'authentification.

Les modes de diffusion et de multidiffusion requièrent l'authentification par défaut. Si vous avez décidé de faire confiance au réseau, vous pouvez désactiver l'authentification en utilisant la directive **disable auth** dans le fichier **ntp.conf**. Alternativement, l'authentification doit être configurée en utilisant des clés symétriques SHA1 ou MD5, ou avec un chiffrement à clés (asymétriques) publiques, en utilisant le schéma Autokey. Le schéma Autokey pour un chiffrement asymétrique est expliqué sur la page man **ntp_auth(8)** et la génération des clés est expliquée sur **ntp-keygen(8)**. Pour implémenter le chiffrement de clés symétriques, veuillez consulter la Section 16.17.12, « Configurer l'authentification symétrique en utilisant une clé » pour une explication de l'option **key**.

16.7. GÉRER LE TEMPS SUR DES MACHINES VIRTUELLES

Les machines virtuelles ne peuvent pas accéder à une horloge matérielle réelle, et une horloge virtuelle

n'est pas assez stable car la stabilité est dépendante de la charge de travail des systèmes hôtes. Ainsi, des horloges paravirtualisées doivent être fournies par l'application en cours d'utilisation. Sur Red Hat Enterprise Linux avec **KVM**, la source de l'horloge par défaut est **kvm-clock**. Veuillez consulter le chapitre *Gestion du temps des invités KVM* du *Guide d'administration et de déploiement de la virtualisation Red Hat Enterprise Linux 7*.

16.8. COMPRENDRE LES SECONDES INTERCALAIRES

L'heure moyenne de Greenwich, ou GMT (« Greenwich Mean Time »), est basée sur une mesure du jour solaire, qui dépend de la vitesse de rotation de la Terre. Lorsque le horloges atomiques ont été créées, le besoin potentiel de définir l'heure de manière plus précise s'est concrétisé. En 1958, le Temps Atomique International (TAI) a été introduit, basé sur des horloges atomiques plus précises et très stables. Une heure astronomique encore plus précise, UT1 (« Universal Time 1 »), a également été introduite pour remplacer l'heure GMT. Les horloges atomiques sont en fait bien plus stables que la vitesse de rotation de la Terre, ainsi les deux heures ont commencé à s'éloigner l'une de l'autre. C'est pourquoi l'heure UTC sert de mesure pratique. Celle-ci est décalée de moins d'une seconde d'UT1, mais pour éviter de devoir effectuer de nombreux changements triviaux, le concept de *seconde intercalaire* fut adopté, permettant ainsi de réconcilier cette différence de manière gérable. La différence entre UT1 et UTC est contrôlée dans la limite d'une dérive d'une demi-seconde. Si tel est le cas, il faudra utiliser une seconde d'ajustement de plus ou de moins. À cause de la nature erratique de la vitesse de rotation de la Terre, le besoin d'ajustement ne peut pas être prédit longtemps à l'avance. La décision quant au moment auquel effectuer un ajustement est prise par l'IERS *International Earth Rotation and Reference Systems Service (IERS)* (« Service international de la rotation terrestre et des systèmes de référence »). Cependant, ces communiqués sont uniquement importants pour les administrateurs des serveurs Stratum 1 car **NTP** transmet des informations sur les secondes intercalaires en attente et les applique automatiquement.

16.9. COMPRENDRE LE FICHIER DE CONFIGURATION NTPD

Le démon **ntpd** lit le fichier de configuration au démarrage système ou lorsque le service est redémarré. L'emplacement par défaut du fichier est **/etc/ntp.conf** et le fichier peut être affiché en saisissant la commande suivante :

```
~]$ less /etc/ntp.conf
```

Les commandes de configuration sont brièvement expliquées plus loin dans ce chapitre, veuillez consulter la Section 16.17, « Configurer NTP », et la page man de **ntp.conf(5)** pour davantage de détails.

Ci-dessous figure une brève explication du contenu du fichier de configuration par défaut :

L'entrée driftfile

Un chemin vers le fichier de dérive (« drift file ») est spécifié, l'entrée par défaut de Red Hat Enterprise Linux est :

```
driftfile /var/lib/ntp/drift
```

Si vous changez ceci, assurez-vous que le répertoire soit accessible en écriture par **ntpd**. Le fichier contient une valeur utilisée pour ajuster la fréquence de l'horloge système après chaque démarrage du système ou du service. Veuillez consulter Comprendre le fichier de dérive pour obtenir davantage d'informations.

Entrées de contrôle d'accès

La ligne suivante définit les restrictions de contrôle d'accès par défaut :

```
restrict default nomodify notrap nopeer noquery
```

- Les options **nomodify** empêchent tout changement de configuration.
- L'option **notrap** empêche les interruptions de protocole de messages de contrôle**ntpdc**.
- L'option **nopeer** empêche la formation d'association de pairs.
- L'option **noquery** empêche de répondre aux requêtes **ntpq** et **ntpdc**, mais n'empêche pas de répondre aux requêtes de temps.

IMPORTANT

Les requêtes **ntpq** et **ntpdc** peuvent être utilisées dans les attaques d'amplification, veuillez donc ne pas supprimer l'option **noquery** de la commande **restrict default** sur des systèmes accessibles publiquement.

Veuillez consulter *CVE-2013-5211* pour obtenir davantage de détails.

Les adresses situées dans la plage **127.0.0.0/8** sont parfois requises par divers processus ou applications. Comme la ligne « restrict default » (restreindre par défaut) ci-dessus prévient l'accès à tout ce qui n'est pas explicitement autorisé, l'accès à l'adresse de bouclage **IPv4** et **IPv6** est autorisé au moyen des lignes suivantes :

```
# the administrative functions.
restrict 127.0.0.1
restrict ::1
```

Les adresses peuvent être ajoutées ci-dessous si elles sont spécifiquement requises par une autre application.

Les hôtes sur le réseau local ne sont pas autorisés à cause de la ligne ci-dessus « restrict default » (restreindre par défaut). Pour changer cela, par exemple pour autoriser les hôtes du réseau **192.0.2.0/24** à demander l'heure et les statistiques et rien de plus, une ligne au format suivant est requise :

```
restrict 192.0.2.0 mask 255.255.255.0 nomodify notrap nopeer
```

Pour autoriser un accès non limité à partir d'un hôte spécifique, par exemple à partir de **192.0.2.250/32**, une ligne au format suivant est requise :

```
restrict 192.0.2.250
```

Un masque de **255.255.255.255** est appliqué si aucun masque n'est spécifié.

Les commandes de restriction sont expliquées dans la page man de **ntp_acc(5)**.

Entrées de serveurs publics

Par défaut, le fichier **ntp.conf** contient quatre entrées de serveurs publics :

```
server 0.rhel.pool.ntp.org iburst
```

```
server 1.rhel.pool.ntp.org iburst
server 2.rhel.pool.ntp.org iburst
server 3.rhel.pool.ntp.org iburst
```

Entrées de serveurs de diffusion et de multidiffusion

Par défaut, le fichier **ntp.conf** contient quelques exemples mis en commentaire. Ces exemples sont principalement explicites. Veuillez consulter la Section 16.17, « Configurer NTP » pour une explication des commandes spécifiques. Si requis, ajoutez vos commandes sous les exemples.

NOTE

Lorsque le programme client de **DHCP**, **dhclient** reçoit une liste de serveurs **NTP** du serveur **DHCP**, il les ajoute à **ntp.conf** et redémarre le service. Pour désactiver cette fonctionnalité, veuillez ajouter **PEERNTP=no** à **/etc/sysconfig/network**.

16.10. COMPRENDRE LE FICHIER SYSCONFIG NTPD

Le fichier sera lu par le script init **ntpd** au démarrage du service. Le contenu par défaut est comme suit :

```
# Command line options for ntpd
OPTIONS="-g"
```

L'option **-g** active **ntpd** pour ignorer la limite de décalage de 1000s et tente de synchroniser l'heure même si le décalage est plus important que 1000s, mais uniquement lors du démarrage système. Sans cette option, **ntpd** se fermera si le décalage dépasse 1000s. Il se fermera également après un démarrage système si le service est redémarré et que le décalage fait plus de 1000s même avec l'option **-g**.

16.11. DÉSACTIVER CHRONY

Pour utiliser **ntpd**, le démon de l'espace utilisateur par défaut, **chronyd** doit être arrêté et désactivé. Veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl stop chronyd
```

Pour l'empêcher de redémarrer pendant le démarrage système, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl disable chronyd
```

Pour vérifier le statut de **chronyd**, veuillez exécuter la commande suivante :

```
~]$ systemctl status chronyd
```

16.12. VÉRIFIER SI LE DÉMON NTP EST INSTALLÉ

Pour vérifier si **ntpd** est installé, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install ntp
```

NTP est implémenté par le démon ou le service **ntpd**, qui se trouve dans le paquet ntp.

16.13. INSTALLER LE DÉMON NTP (NTPD)

Pour installer **ntpd**, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install ntp
```

Pour activer **ntpd** lors du démarrage système, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl enable ntpd
```

16.14. VÉRIFIER LE STATUT DE NTP

Pour vérifier si **ntpd** est en cours d'exécution et configuré pour être exécuté lors du démarrage système, veuillez exécuter la commande suivante :

```
~]$ systemctl status ntpd
```

Pour obtenir un bref rapport sur le statut de **ntpd**, veuillez exécuter la commande suivante :

```
~]$ ntpstat
unsynchronised
  time server re-starting
   polling server every 64 s
```

```
~]$ ntpstat
synchronised to NTP server (10.5.26.10) at stratum 2
   time correct to within 52 ms
   polling server every 1024 s
```

16.15. CONFIGURER LE PARE-FEU POUR AUTORISER LES PAQUETS NTP ENTRANTS

Le trafic **NTP** consiste en paquets **UDP** sur le port **123** et doit être autorisé sur le réseau et sur les pare-feux basés hôte pour que **NTP** puisse fonctionner.

Vérifiez si le pare-feu est configuré de manière à autoriser le trafic **NTP** entrant pour les clients utilisant l'outil de configuration du pare-feu graphique **Firewall Configuration**.

Pour lancer l'outil graphique **firewall-config**, appuyez sur la touche **Super** pour vous rendre sur la « Vue d'ensemble des activités » (Activities Overview), saisissez **firewall** puis appuyez sur **Entrée**. La fenêtre « **Configuration du pare-feu** » s'ouvre. Le mot de passe de l'utilisateur vous sera demandé.

Pour lancer l'outil graphique de configuration du pare-feu en utilisant la ligne de commande, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# firewall-config
```

La fenêtre **Configuration du pare-feu** s'ouvre. Remarquez que cette commande peut être exécutée par un utilisateur normal mais le mot de passe de l'utilisateur **root** vous sera quand même demandé de temps à autre.

Recherchez le mot « Connecté » (« Connected ») dans le coin en bas à gauche. Ceci indique que l'outil **firewall-config** est connecté au démon de l'espace utilisateur **firewalld**.

16.15.1. Modifier les paramètres du pare-feu

Pour immédiatement changer les paramètres actuels du pare-feu, assurez-vous que le menu de sélection déroulant étiqueté **Configuration** soit défini sur **Runtime**. Alternativement, pour modifier les paramètres à appliquer lors du prochain démarrage système, ou du prochain rechargement du pare-feu, sélectionnez **Permanent** dans la liste déroulante.



NOTE

Lorsque des changements des paramètres du pare-feu sont effectués, en mode **Runtime**, votre sélection entrera en vigueur dès lors que vous cochez ou décochez la case associée au service. N'oubliez pas ceci lorsque vous travaillez sur un système qui pourrait être en cours d'utilisation par d'autres utilisateurs.

Lorsque des changements des paramètres du pare-feu en mode **Permanent** sont effectués, votre sélection entrera en vigueur dès lors que vous rechargez le pare-feu ou que le système redémarre. Pour recharger le pare-feu, veuillez sélectionner le menu **Options** et sélectionnez **Recharger le pare-feu**.

16.15.2. Ouvrir des ports du pare-feu pour les paquets NTP

Pour autoriser le trafic à travers le pare-feu vers un certain port, lancez l'outil **firewall-config** et sélectionnez la zone réseau dont vous souhaitez changer les paramètres. Sélectionnez l'onglet **Ports** et cliquez sur le bouton **Ajouter**. La fenêtre **Port et Protocole** s'ouvrira.

Saisissez le numéro de port **123** et sélectionnez **udp** dans la liste déroulante.

16.16. CONFIGURER LES SERVEURS NTPDATE

Le but du service **ntpdate** est de régler l'horloge pendant le démarrage système. C'était utilisé pour veiller à ce que les services lancés après **ntpdate** aient l'heure correcte sans « sauts » d'horloge. L'utilisation de **ntpdate** et de la liste des « step-tickers » sont déconseillés et Red Hat Enterprise Linux 7 utilise désormais l'option **-g** sur la commande **ntpd** et non plus **ntpdate** par défaut.

Le service **ntpdate** de Red Hat Enterprise Linux 7 est plus utile en autonome, sans **ntpd**. Avec **systemd**, qui lance des services en parallèle, l'activation du service **ntpdate** ne garantit pas que d'autres services lancés par la suite auront l'heure correcte, à moins qu'ils ne spécifient une dépendance de classement sur **time-sync.target**, qui est fournie par le service **ntpdate**. Pour vous assurer qu'un service soit lancé avec l'heure correcte, ajoutez **After=time-sync.target** au service et activez l'un des services qui fournit la cible (**ntpdate** ou **sntp**). Cette dépendance est incluse par défaut sur certains services Red Hat Enterprise Linux 7 (par exemple, **dhcpd**, **dhcpd6**, et **crond**).

Pour vérifier si le service **ntpdate** est activé pour s'exécuter pendant le démarrage système, veuillez exécuter la commande suivante :

```
~]$ systemctl status ntpdate
```

Pour activer le service pour qu'il soit exécuté au moment du démarrage système, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl enable ntpdate
```

Dans Red Hat Enterprise Linux 7, le fichier **/etc/ntp/step-tickers** par défaut contient **0.rhel.pool.ntp.org**. Pour configurer des serveurs **ntpdate** supplémentaires, veuillez utiliser un éditeur de texte en tant qu'utilisateur **root** et modifiez **/etc/ntp/step-tickers**. Le nombre de serveurs répertoriés n'est pas très important car **ntpdate** utilisera uniquement ceci pour obtenir des informations sur la date une fois que le système aura démarré. Si vous possédez un serveur de temps interne, alors utilisez ce nom d'hôte pour la première ligne. Un hôte supplémentaire sur la seconde ligne en tant que copie de sauvegarde est une bonne idée. La sélection des serveurs de sauvegarde et le choix d'un second hôte interne ou externe dépend de votre évaluation des risques. Par exemple, quelles sont les chances qu'un problème affectant le premier serveur puissent également affecter le second serveur ? Est-ce que la connectivité d'un serveur externe sera plus facilement disponible que la connectivité aux serveurs internes en cas de panne réseau perturbant l'accès au premier serveur ?

16.17. CONFIGURER NTP

Pour modifier la configuration par défaut du service **NTP**, veuillez utiliser un éditeur de texte exécuté en tant qu'utilisateur **root** pour modifier le fichier **/etc/ntp.conf**. Ce fichier est installé avec **ntpd** et est configuré pour utiliser des serveurs de temps du pool Red Hat par défaut. La page man **ntp.conf(5)** décrit les options de commande pouvant être utilisées dans le fichier de configuration, mises à part les commandes d'accès et de limitation du taux, qui sont expliquées dans la page man de **ntp_acc(5)**.

16.17.1. Configurer le contrôle d'accès à un service NTP

Pour limiter ou contrôler l'accès au service **NTP** exécuté sur un système, veuillez utiliser la commande **restrict** dans le fichier **ntp.conf**. Voir l'exemple en commentaire :

```
# Hosts on local network are less restricted.
#restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap
```

La commande **restrict** prend la forme suivante :

```
restrict option
```

où *option* représente soit :

- **ignore** — Tous les paquets seront ignorés, y compris les requêtes **ntpq** ou **ntpdc** ou une combinaison de ces options.
- **kod** — un paquet « Kiss-o'-death » » (KoD) sera envoyé pour réduire les requêtes indésirables.
- **limited** — ne pas répondre aux requêtes de service de temps si le paquet transgresse les valeurs par défaut de la limite de taux ou les valeurs spécifiées par la commande **discard**. Les requêtes **ntpq** et **ntpdc** ne sont pas affectées. Pour obtenir davantage d'informations sur la commande **discard** et sur les valeurs par défaut, veuillez consulter la Section 16.17.2, « Configurer le taux limite d'accès à un service NTP ».
- **lowpriotrap** — interruptions définies comme étant de basse priorité par les hôtes correspondants.

- **nomodify** — empêche tout changement de configuration.
- **noquery** — empêche de répondre aux requêtes **ntpq** et **ntpdc**, mais n'empêche pas de répondre aux requêtes de temps.
- **nopeer** — empêche la formation d'association de pairs.
- **noserve** — refuse tous les paquets, sauf les requêtes **ntpq** et **ntpdc**.
- **notrap** — empêche les interruptions de protocole de messages de contrôle**ntpdc**.
- **notrust** — refuse les paquets dont le chiffrement n'est pas authentifié.
- **ntpport** — modifie l'algorithme de correspondance pour appliquer la restriction uniquement si le port source est le port standard **NTP UDP 123**.
- **version** — refuse les paquets qui ne correspondent pas à la version **NTP** actuelle.

Pour configurer l'accès limitant le taux pour ne pas dutout répondre à une requête, la commande **restrict** respective doit inclure l'option **limited**. Si **ntpd** doit répondre avec un paquet **KoD**, la commande **restrict** devra inclure les options **limited** et **kod**.

Les requêtes **ntpq** et **ntpdc** peuvent être utilisées dans les attaques d'amplification (consultez *CVE-2013-5211* pour plus de détails), veuillez donc ne pas supprimer l'option **noquery** de la commande **restrict default** sur des systèmes accessibles publiquement.

16.17.2. Configurer le taux limite d'accès à un service NTP

Pour activer le taux limite d'accès à un service **NTP** exécuté sur un système, veuillez ajouter l'option **limited** à la commande **restrict** comme expliqué dans Section 16.17.1, « Configurer le contrôle d'accès à un service NTP ». Si vous ne souhaitez pas utiliser les paramètres d'abandon par défaut, alors veuillez également utiliser la commande **discard**, comme décrit ici.

La commande **discard** prend la forme suivante :

```
discard [average value] [minimum value] [monitor value]
```

- **average** — spécifie l'espacement de paquets moyen minimum autorisé, un argument est également accepté en $\log_2$ secondes. La valeur par défaut est de 3 (2^3 équivaut à 8 secondes).
- **minimum** — spécifie l'espacement de paquets minimum autorisé, un argument est accepté en $\log_2$ secondes. La valeur par défaut est de 1 (2^1 équivaut à 2 secondes).
- **monitor** — spécifie la probabilité d'abandon de paquet une fois que les limites de taux autorisées ont été dépassées. La valeur par défaut est de 3000 secondes. Cette option est conçue pour les serveurs recevant 1000 requêtes ou plus par seconde.

Examples of the **discard** command are as follows:

```
discard average 4
```

```
discard average 4 minimum 2
```

16.17.3. Ajouter l'adresse d'un pair

Pour ajouter l'adresse d'un pair, c'est-à-dire l'adresse d'un serveur exécutant un service **NTP** du même stratum, veuillez utiliser la commande **peer** dans le fichier **ntp.conf**.

La commande **peer** prend la forme suivante :

```
peer address
```

où *address* est une adresse de monodiffusion **IP** ou un nom **DNS** résolvable. L'adresse doit uniquement être celle d'un système connu comme étant membre du même stratum. Les homologues (peer) doivent avoir au moins une source de temps différente l'un par rapport à l'autre. Habituellement, les homologues sont des systèmes sous le même contrôle administratif.

16.17.4. Ajouter une adresse de serveur

Pour ajouter l'adresse d'un serveur, c'est-à-dire l'adresse d'un serveur exécutant un service **NTP** d'un stratum plus élevé, veuillez utiliser la commande **server** dans le fichier **ntp.conf**.

La commande **server** prend la forme suivante :

```
server address
```

où *address* est une adresse de monodiffusion **IP** ou un nom **DNS** résolvable. Adresse d'un serveur de référence distant ou d'une horloge de référence locale à partir de laquelle les paquets seront reçus.

16.17.5. Ajouter une adresse de serveur de diffusion ou de multidiffusion

Pour ajouter une adresse de diffusion ou de multidiffusion sur laquelle envoyer les paquets **NTP** de diffusion ou de multidiffusion, utilisez la commande **broadcast** dans le fichier **ntp.conf**.

Les modes de diffusion et de multidiffusion requièrent une authentification par défaut. Veuillez consulter la Section 16.6, « Options d'authentification NTP ».

La commande **broadcast** prend la forme suivante :

```
broadcast address
```

où *address* est une adresse de diffusion ou de multidiffusion **IP** sur laquelle les paquets sont envoyés.

Cette commande configure un système pour agir en tant que serveur de diffusion **NTP**. L'adresse utilisée doit être une adresse de diffusion ou de multidiffusion. Une adresse de diffusion implique l'adresse **IPv4 255.255.255.255**. Par défaut, les routeurs ne transmettent pas de messages de diffusion. L'adresse de multidiffusion peut être une adresse **IPv4** de class D, ou une adresse **IPv6**. L'IANA a assigné l'adresse de multidiffusion **IPv4 224.0.1.1** et l'adresse **IPv6 FF05::101** (locale au site) à **NTP**. Les adresses de multidiffusion **IPv4** dans un cadre administratif peuvent également être utilisées, comme décrit dans la page *RFC 2365 Administratively Scoped IP Multicast*.

16.17.6. Ajouter une adresse de client Manycast

Pour ajouter une adresse de client manycast, autrement dit, pour configurer une adresse de multidiffusion pour une utilisation de type découverte de serveur **NTP**, utiliser la commande **manycastclient** dans le fichier **ntp.conf**.

La commande **manycastclient** prend la forme suivante :

```
manycastclient address
```

où *address* est une adresse de multidiffusion **IP** à partir de laquelle les paquets seront reçus. Le client enverra une requête à l'adresse et sélectionnera les meilleurs serveurs à partir des réponses et ignorera les autres serveurs. La communication **NTP** utilise ensuite des associations de monodiffusion, comme si les serveurs **NTP** découverts étaient répertoriés dans **ntp.conf**.

Cette commande configure un système pour agir en tant que client **NTP**. Les systèmes peuvent être client et serveur à la fois.

16.17.7. Ajouter une adresse de client de diffusion

Pour ajouter une adresse de client de diffusion, autrement dit, pour configurer une adresse de diffusion pour que les paquets de diffusion **NTP** y soient vérifiés, assurez-vous de bien utiliser la commande **broadcastclient** dans le fichier **ntp.conf**.

La commande **broadcastclient** prend la forme suivante :

```
broadcastclient
```

Autorise la réception de messages de diffusion. Requiert une authentification par défaut. Veuillez consulter la Section 16.6, « Options d'authentification NTP ».

Cette commande configure un système pour agir en tant que client **NTP**. Les systèmes peuvent être client et serveur à la fois.

16.17.8. Ajouter une adresse de serveur Manycast

Pour ajouter une adresse de serveur Manycast, autrement dit, pour configurer une adresse afin d'autoriser les clients à découvrir le serveur en effectuant une multidiffusion de paquets **NTP**, utiliser la commande **manycastserver** dans le fichier **ntp.conf**.

La commande **manycastserver** prend la forme suivante :

```
manycastserver address
```

Autorise l'envoi de messages de multidiffusion. Où *address* est l'adresse sur laquelle effectuer la multidiffusion. Ceci devrait être utilisé en conjonction avec l'authentification afin d'empêcher toute perturbation du service.

Cette commande configure un système pour agir en tant que serveur **NTP**. Les systèmes peuvent être client et serveur à la fois.

16.17.9. Ajouter une adresse de client de multidiffusion

Pour ajouter une adresse de client de multidiffusion, autrement dit, pour configurer une adresse de multidiffusion pour que les paquets de multidiffusion **NTP** y soient vérifiés, assurez-vous de bien utiliser la commande **multicastclient** dans le fichier **ntp.conf**.

La commande **multicastclient** prend la forme suivante :

```
multicastclient address
```

Autorise la réception de messages de multidiffusion. Où *address* est l'adresse à laquelle souscrire. Ceci devrait être utilisé en conjonction avec l'authentification afin d'empêcher toute perturbation du service.

Cette commande configure un système pour agir en tant que client **NTP**. Les systèmes peuvent être client et serveur à la fois.

16.17.10. Configurer l'option Burst

Utiliser l'option **burst** sur un serveur public est considéré comme un abus. N'utilisez pas cette option sur des serveurs **NTP** publics. Ne l'utilisez que pour les applications qui appartiennent à votre organisation.

Pour améliorer la qualité moyenne des statistiques de décalage horaire, veuillez ajouter l'option suivante à la fin d'une commande serveur :

```
burst
```

À chaque intervalle d'interrogation, lorsque le serveur répond, le système enverra une rafale allant jusqu'à huit paquets au lieu d'un seul paquet habituel. À utiliser avec la commande **server** pour améliorer la qualité moyenne des calculs de décalage horaire.

16.17.11. Configurer l'option iburst

Pour améliorer le temps pris pour la synchronisation initiale, veuillez ajouter l'option suivante à la fin d'une commande serveur :

```
iburst
```

Lorsque le serveur ne répond pas, on envoie une rafale de huit paquets au lieu d'un habituel. Les paquets sont normalement envoyés à intervalles de 2 s ; mais l'espacement peut être changé entre l'envoi du premier et du second paquet par la commande **calldelay** pour permettre un temps supplémentaire pour l'appel ISDN ou modem. À utiliser avec la commande **server** pour réduire le temps pris pour une synchronisation initiale. Celle-ci est désormais une option par défaut dans le fichier de configuration.

16.17.12. Configurer l'authentification symétrique en utilisant une clé

Pour configurer l'authentification symétrique en utilisant une clé, veuillez ajouter l'option suivante à la fin d'une commande serveur ou pair :

```
key number
```

où le *nombre* se trouve dans une plage de **1** à **65534** inclus. Cette option autorise l'utilisation d'un code d'authentification de message (*message authentication code*, ou MAC) dans les paquets. Cette option est conçue pour une utilisation avec les commandes **peer**, **server**, **broadcast**, et **manycastclient**.

L'option peut être utilisée dans le fichier **/etc/ntp.conf** comme suit :

```
server 192.168.1.1 key 10
broadcast 192.168.1.255 key 20
manycastclient 239.255.254.254 key 30
```

–

Veuillez également consulter la Section 16.6, « Options d'authentification NTP ».

16.17.13. Configurer l'intervalle d'interrogation

Pour modifier l'intervalle d'interrogation par défaut, veuillez ajouter les options suivantes à la fin d'une commande serveur ou pair :

```
minpoll value and maxpoll value
```

Options pour changer l'intervalle d'interrogation par défaut, où l'intervalle en secondes sera calculé en utilisant 2 à la puissance *valeur*, autrement dit, l'intervalle est exprimé en $\log_2$ secondes. La valeur **minpoll** par défaut est 6, 2^6 équivaut à 64s. La valeur par défaut de **maxpoll** est de 10, ce qui équivaut à 1024s. Les valeurs autorisées se trouvent entre 3 et 17 inclus, ce qui signifie entre 8s et 36.4h respectivement. Ces options sont pour une utilisation avec la commande **peer** ou **server**. Définir une valeur **maxpoll** plus basse peut améliorer la précision de l'horloge.

16.17.14. Configurer les préférences du serveur

Pour spécifier qu'un serveur en particulier est préférable à d'autres possédant des qualités statistiques similaires, veuillez ajouter l'option suivante à la fin d'une commande serveur ou pair :

```
prefer
```

Veuillez utiliser ce serveur pour une synchronisation préférable à d'autres serveurs possédant des qualités statistiques similaires. Cette option est conçue pour une utilisation avec les commandes **peer** ou **server**.

16.17.15. Configurer la valeur de durée de vie (« Time-to-Live ») des paquets NTP

Pour spécifier une valeur de durée de vie particulière (« *time-to-live* », ou TTL) plutôt qu'utiliser celle par défaut, ajoutez l'option suivante à la fin d'une commande serveur ou pair :

```
ttl value
```

Spécifiez la valeur TTL à utiliser dans les paquets envoyés par les serveurs de diffusion et de multidiffusion **NTP**. Spécifiez la valeur TTL maximale à utiliser pour la recherche « « expanding ring search » » par un client Manycast. La valeur par défaut est de **127**.

16.17.16. Configurer la version NTP à utiliser

Pour spécifier une version particulière de **NTP** à utiliser à la place de celle par défaut, veuillez ajouter l'option suivante à la fin d'une commande serveur ou pair :

```
version value
```

Spécifiez la version de **NTP** définie dans les paquets **NTP** créés. La valeur peut se trouver dans une gamme allant de **1** à **4**. La valeur par défaut est **4**.

16.18. CONFIGURER LA MISE À JOUR DE L'HORLOGE MATÉRIELLE

L'horloge système peut être également utilisée pour mettre à jour l'horloge matérielle, aussi appelée horloge temps réel (« real-time clock », ou RTC). Cette section vous montre trois façons d'aborder cette tâche :

Mise à jour ponctuelle instantanée

Pour effectuer une mise à jour ponctuelle instantanée de l'horloge matérielle, exécutez cette commande en tant qu'utilisateur root :

```
~]# hwclock --systohc
```

Mise à jour à chaque démarrage

Pour effectuer la mise à jour de l'horloge matérielle à chaque démarrage, après avoir exécuté l'utilitaire de synchronisation suivant **ntpdate**, procédez ainsi :

1. Ajouter la ligne suivante au fichier **/etc/sysconfig/ntpdate** :

   ```
   SYNC_HWCLOCK=yes
   ```

2. Activee le service **ntpdate** en tant qu'utilisateur root :

   ```
   ~]# systemctl enable ntpdate.service
   ```

Notez que le service **ntpdate** utilise les serveurs NTP définis dans le fichier **/etc/ntp/step-tickers**.



NOTE

Dans les machines virtuelles, l'horloge matérielle sera mise à jour lors du prochain démarrage de la machine hôte, et non pas de la machine virtuelle.

Mettre à jour via NTP

Pour effectuer la mise à jour de l'horloge matérielle à chaque fois que l'horloge système est mise à jour par le service **ntpd** ou **chronyd**, démarrez le service **ntpdate** en tant qu'utilisateur root :

```
~]# systemctl start ntpdate.service
```

Pour que le comportement soit consistant à travers les démarrages, rendez le service automatique au démarage :

```
~]# systemctl enable ntpdate.service
```

Après avoir activé le service **ntpdate**, à chaque fois que l'horloge système sera synchronisée par **ntpd** ou **chronyd**, le noyau mettra à jour l'horloge matérielle automatiquement toutes les 11 minutes.

AVERTISSEMENT

Cette approche risque de ne pas toujours fonctionner car ce mode 11-minutes n'est pas toujours activé, donc la mise à jour de l'horloge matérielle n'a pas forcément lieu au même moment que la mise à jour de l'horloge système.

16.19. CONFIGURER LES SOURCES DES HORLOGES

Pour répertorier les sources d'horloge disponibles sur votre système, veuillez exécuter les commandes suivantes :

```
~]$ cd /sys/devices/system/clocksource/clocksource0/
clocksource0]$ cat available_clocksource
kvm-clock tsc hpet acpi_pm
clocksource0]$ cat current_clocksource
kvm-clock
```

Dans l'exemple ci-dessus, le noyau utilise **kvm-clock**. Ceci a été sélectionné pendant le démarrage car il s'agit d'une machine virtuelle. Notez que la source d'horloge varie en fonction de l'architecture.

Pour outrepasser la source d'horloge par défaut, veuillez ajouter la directive **`clocksource`** à la ligne GRUB du noyau. Utiliser l'outil **`grubby`** pour effectuer ce changement. Ainsi, pour forcer le noyau par défaut sur un système pour qu'il utilise le source d'horloge **`tsc`**, saisir la commande suivante :

```
~]# grubby --args=clocksource=tsc --update-kernel=DEFAULT
```

Le paramètre **`--update-kernel`** accepte également le mot clé **`ALL`**, ou une liste de numéros d'indexes de noyaux séparée par des virgules.

Voir Chapitre 24, *Utiliser le chargeur de démarrage GRUB 2* pour obtenir plus d'informations sur la façon d'apporter des changements au menu GRUB.

16.20. RESSOURCES SUPPLÉMENTAIRES

Les sources d'information suivantes offrent des ressources supplémentaires concernant **`NTP`** et **`ntpd`**.

16.20.1. Documentation installée

- Page man **`ntpd(8)`** — décrit **`ntpd`** en détails, y compris les options de ligne de commandes.
- Page man **`ntp.conf(5)`** — contient des informations sur la manière de configurer des associations avec des serveurs et des pairs.
- Page man **`ntpq(8)`** — décrit l'utilitaire de requêtes **`NTP`** pour surveiller et effectuer des requêtes sur un serveur **`NTP`**.
- Page man **`ntpdc(8)`** — décrit l'utilitaire **`ntpd`** pour effectuer des requêtes et changer l'état de **`ntpd`**.

- Page man **ntp_auth(5)** — décrit les options d'authentification, commandes, et la gestion des clés de **ntpd**.
- Page man **ntp_keygen(8)** — décrit la génération de clés publiques et privées pour **ntpd**.
- Page man **ntp_acc(5)** — décrit les options de contrôle d'accès en utilisant la commande **restrict**.
- Page man **ntp_mon(5)** — décrit les options de surveillance pour la collecte de statistiques.
- Page man **ntp_clock(5)** — décrit les commandes pour configurer les horloges de référence.
- Page man **ntp_misc(5)** — décrit les options diverses.
- Page man **ntp_decode(5)** — répertorie le mots de statut, messages d'événements et codes d'erreur utilisés pour les rapports et la surveillance **ntpd**.
- Page man **ntpstat(8)** — décrit un utilitaire pour rapporter l'état de synchronisation du démon **NTP** exécuté sur la machine locale.
- Page man **ntptime(8)** — décrit un utilitaire pour lire et paramétrer les variables de temps du noyau.
- Page man **tickadj(8)** — décrit un utilitaire pour lire, et optionnellement paramétrer la longueur du tic.

16.20.2. Sites Web utiles

http://doc.ntp.org/

Archive de la documentation NTP

http://www.eecis.udel.edu/~mills/ntp.html

Projet de recherche sur la synchronisation du temps réseau

http://www.eecis.udel.edu/~mills/ntp/html/manyopt.html

Informations sur le découverte automatique de serveurs sur **NTPv4**.

CHAPITRE 17. CONFIGURER PTP EN UTILISANT PTP4L

17.1. INTRODUCTION À PTP

Precision Time Protocol (PTP) est un protocole utilisé pour synchroniser des horloges dans un réseau. Lorsqu'utilisé en conjonction avec une prise en charge du matériel, le **PTP** est capable d'atteindre une précision inférieure à la microseconde, ce qui est bien mieux que les résultats normalement obtenus avec **NTP**. La prise en charge de **PTP** est divisée entre le noyau et l'espace utilisateur. Le noyau dans Red Hat Enterprise Linux inclut la prise en charge des horloges **PTP**, qui sont fournies par les pilotes réseaux. L'implémentation du protocole est connue sous le nom **linuxptp**, une implémentation **PTPv2** selon le standard IEEE 1588 pour Linux.

Le paquet linuxptp inclut les programmes **ptp4l** et **phc2sys** pour la synchronisation des horloges. Le programme **ptp4l** implémente la « Boundary lock » **PTP** et l'« Ordinary Clock ». Avec l'horodatage matériel, ce paquet est utilisé pour synchroniser l'horloge matérielle **PTP** à l'horloge maître, et avec l'horodatage logiciel, il permet de synchroniser l'horloge système à l'horloge maître. Le programme **phc2sys** n'est utilie que pour l'horodatage matériel, pour synchroniser l'horloge système à l'horloge matérielle **PTP** sur la *carte réseau* (NIC).

17.1.1. Comprendre PTP

Les horloges synchronisées par **PTP** sont organisées sous une hiérarchie de type maître-esclave. Les esclaves sont synchronisés à leurs maîtres qui peuvent eux-mêmes être esclaves d'autres maîtres. La hiérarchie est créée et mise à jour automatiquement par l'algorithme « *best master clock* » (BMC), qui est exécuté sur chaque horloge. Lorsqu'une horloge ne possède qu'un seul port, celle-ci peut être maître (« *master* ») ou esclave (« *slave* »), ce type d'horloge est appelé une *Ordinary Clock* (OC). Une horloge avec de multiples ports peut être maître sur un port et esclave sur un autre, ce type d'horloge est appelé une *Boundary Clock* (BC). L'horloge maître se trouvant tout en haut est appelée *Grandmaster Clock*, et peut être synchronisée en utilisant une source de temps GPS (*Global Positioning System*). En utilisant une source de temps basée GPS, des réseaux disparates peuvent être synchronisés avec un haut degré de précision.



Figure 17.1. Horloges PTP grandmaster, horloges boundary, et horloges esclaves

17.1.2. Avantages de PTP

L'un des principaux avantages offerts par **PTP** comparé au protocole *Network Time Protocol* (ou NTP) est la prise en charge matérielle présente dans divers NIC (*Network Interface Controllers*) et interrupteurs réseau. Ce matériel spécialisé permet à **PTP** de tenir compte des délais de transfert de messages, et de largement augmenter la précision de la synchronisation du temps. Même s'il est possible d'utiliser des composants matériel dans le réseau sur lesquels PTP n'est pas activé, cela causera souvent une instabilité ou introduira une asymétrie dans le délai, résultant en inexactitudes de synchronisation, et en une multitude de composants ne reconnaissant pas PTP, utilisés tout au long du chemin de communication. Pour atteindre la meilleure précision possible, il est recommandé que le matériel autorise **PTP** sur tous les composants réseau entre horloges **PTP** . La synchronisation du temps dans les réseaux de plus grandes taille où tout le matériel réseau ne prend pas forcément en charge **PTP** pourrait être plus adéquat pour **NTP**.

Avec la prise en charge matérielle **PTP**, le NIC possède sa propre horloge intégrée, qui est utilisée pour horodater les messages **PTP** reçus et transmis. Cette horloge intégrée est effectivement synchronisée avec l'horloge maître **PTP**, et l'horloge système de l'ordinateur est synchronisée avec l'horloge matérielle **PTP** sur le NIC. Avec la prise en charge **PTP**, l'horloge système est utilisée pour horodater les messages

PTP et est synchronisée avec l'horloge maître **PTP** directement. La prise en charge matérielle **PTP** offre une meilleure précision puisque le NIC peut horodater les paquets **PTP** au moment exact où ils sont envoyés et reçus, tandis que la prise en charge **PTP** logicielle requiert le traitement supplémentaire des paquets **PTP** par le système d'exploitation.

17.2. UTILISER PTP

Pour utiliser **PTP**, le pilote réseau du noyau de l'interface doit offrir des capacités de prise en charge d'horodatage logiciel ou matériel.

17.2.1. Vérifier la prise en charge des pilotes et du matériel

En outre de la prise en charge de l'horodatage matériel présent dans le pilote, le NIC doit également être capable de prendre en charge cette fonctionnalité dans le matériel physique. La meilleure manière de vérifier les capacités d'horodatage d'un pilote particulier et de la carte réseau (NIC) consiste à utiliser l'utilitaire **ethtool** pour interroger l'interface comme suit :

```
~]# ethtool -T eth3
Time stamping parameters for eth3:
Capabilities:
        hardware-transmit     (SOF_TIMESTAMPING_TX_HARDWARE)
        software-transmit     (SOF_TIMESTAMPING_TX_SOFTWARE)
        hardware-receive      (SOF_TIMESTAMPING_RX_HARDWARE)
        software-receive      (SOF_TIMESTAMPING_RX_SOFTWARE)
        software-system-clock (SOF_TIMESTAMPING_SOFTWARE)
        hardware-raw-clock    (SOF_TIMESTAMPING_RAW_HARDWARE)
PTP Hardware Clock: 0
Hardware Transmit Timestamp Modes:
        off                   (HWTSTAMP_TX_OFF)
        on                    (HWTSTAMP_TX_ON)
Hardware Receive Filter Modes:
        none                  (HWTSTAMP_FILTER_NONE)
        all                   (HWTSTAMP_FILTER_ALL)
```

Quand *eth3* est l'interface que vous souhaitez vérifier.

Pour la prise en charge de l'horodatage logiciel, la liste des paramètres doit inclure :

- **SOF_TIMESTAMPING_SOFTWARE**
- **SOF_TIMESTAMPING_TX_SOFTWARE**
- **SOF_TIMESTAMPING_RX_SOFTWARE**

Pour la prise en charge de l'horodatage matériel, la liste des paramètres doit inclure :

- **SOF_TIMESTAMPING_RAW_HARDWARE**
- **SOF_TIMESTAMPING_TX_HARDWARE**
- **SOF_TIMESTAMPING_RX_HARDWARE**

17.2.2. Installer PTP

Le noyau dans Red Hat Enterprise Linux inclut la prise en charge de **PTP**. La prise en charge de l'espace utilisateur est fournie par les outils du paquet **linuxptp**. Pour installer **linuxptp**, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install linuxptp
```

Ceci installera **ptp4l** et **phc2sys**.

Veuillez ne pas exécuter plus d'un service à la fois pour paramétrer l'horloge système. Si vous avez l'intention de servir l'heure **PTP** en utilisant **NTP**, veuillez consulter la Section 17.8, « Servir le temps PTP avec NTP ».

17.2.3. Lancer ptp4l

Le programme **ptp4l** peut être lancé à partir de la ligne de commande ou en tant que service. Lorsqu'il est exécuté en tant que service, les options sont spécifiées dans le fichier **/etc/sysconfig/ptp4l**. Les options requises pour une utilisation par le service et sur la ligne de commande doivent être spécifiées dans le fichier **/etc/ptp4l.conf**. Le fichier **/etc/sysconfig/ptp4l** inclut l'option de ligne de commande **-f /etc/ptp4l.conf**, qui cause au programme **ptp4l** de lire le fichier **/etc/ptp4l.conf** et de traiter les options contenues. L'utilisation du fichier **/etc/ptp4l.conf** est expliquée dans la Section 17.4, « Spécifier un fichier de configuration ». Des informations supplémentaires sur les différentes options **ptp4l** et sur les paramètres du fichier de configuration se trouvent sur la page man **ptp4l(8)**.

Lancer ptp4l en tant que service

Pour lancer **ptp4l** en tant que service, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl start ptp4l
```

Pour obtenir des informations supplémentaires sur la gestion des services système sur Red Hat Enterprise Linux 7, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

Utiliser ptp4l dans la ligne de commande

Le programme **ptp4l** tente d'utiliser l'horodatage matériel par défaut. Pour utiliser **ptp4l** avec des pilotes et NIC ayant des capacités d'horodatage matériel, vous devrez fournir l'interface réseau à utiliser avec l'option **-i**. Veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# ptp4l -i eth3 -m
```

Quand *eth3* est l'interface que vous souhaitez configurer. Ci-dessous figure un exemple de sortie de **ptp4l** lorsque l'horloge **PTP** du NIC est synchronisée avec une horloge maître :

```
~]# ptp4l -i eth3 -m
selected eth3 as PTP clock
port 1: INITIALIZING to LISTENING on INITIALIZE
port 0: INITIALIZING to LISTENING on INITIALIZE
port 1: new foreign master 00a069.fffe.0b552d-1
selected best master clock 00a069.fffe.0b552d
port 1: LISTENING to UNCALIBRATED on RS_SLAVE
master offset -23947 s0 freq +0 path delay       11350
master offset -28867 s0 freq +0 path delay       11236
master offset -32801 s0 freq +0 path delay       10841
master offset -37203 s1 freq +0 path delay       10583
```

```
master offset  -7275 s2 freq -30575 path delay   10583
port 1: UNCALIBRATED to SLAVE on MASTER_CLOCK_SELECTED
master offset  -4552 s2 freq -30035 path delay   10385
```

La valeur du décalage maître est le décalage mesuré de l'horloge maître en nanosecondes. Les chaînes **s0**, **s1**, **s2** indiquent différents états de l'horloge servo : **s0** est déverrouillé, **s1** est un step (ou décalage) d'horloge et **s2** est verrouillé. Une fois le servo dans un état verrouillé (**s2**), l'horloge ne sera pas décalée (mais uniquement légèrement ajustée), à moins que l'option **pi_offset_const** ne soit définie sur une valeur positive dans le fichier de configuration (décrit sur la page man **ptp4l(8)**). La valeur **adj** est l'ajustement de la fréquence de l'horloge en ppb (« parts per billion », ou parties par milliard). La valeur du délai du chemin correspond au délai estimé des messages de synchronisation envoyés depuis l'horloge maître en nanosecondes. Le port 0 est un socket de domaines Unix utilisé pour la gestion **PTP** locale. Le port 1 est l'interface **eth3** (basé sur l'exemple ci-dessus.) Les états du port possibles incluent INITIALIZING, LISTENING, UNCALIBRATED et SLAVE, ceux-ci peuvent changer lors des événements INITIALIZE, RS_SLAVE, MASTER_CLOCK_SELECTED. Dans le dernier message de changement d'état, l'état du port est passé de UNCALIBRATED à SLAVE, ce qui indique une synchronisation réussie avec une horloge **PTP** maître.

Journaliser des messages à partir de ptp4l

Par défaut , les messages sont envoyés sur **/var/log/messages**. Cependant, spécifier l'option **-m** active la journalisation sur la sortie standard, ce qui peut être utile pour des raisons de débogage.

Pour activer l'horodatage logiciel, l'option **-S** doit être utilisée comme suit :

```
~]# ptp4l -i eth3 -m -S
```

17.2.3.1. Sélectionner un mécanisme de mesure du délai

Il existe deux mécanismes de mesure de délai différents qui peuvent être sélectionnés en ajoutant une option à la commande **ptp4l** :

-P

L'option **-P** permet de sélectionner le mécanisme de mesure de délai pair à pair (*peer-to-peer*, ou P2P).

Le mécanisme P2P est préféré car il réagit aux changements de la topologie réseau plus rapidement, et peut être plus précis dans la mesure du délai que les autres mécanismes. Le mécanisme P2P peut uniquement être utilisé sur les topologies où chaque port échanges des messages PTP avec au moins un autre port P2P. Il doit être pris en charge et utilisé par tout le matériel sur le chemin de communication, y compris les horloges transparentes.

-E

L'option **-E** sélectionne le mécanisme de mesure de délai *end-to-end* (E2E). Ce mécanisme est utilisé par défaut.

Le mécanisme E2E est également appelé mécanisme « request-response » (requête-réponse) du délai.

-A

L'option **-A** active la sélection automatique du mécanisme de mesure du délai.

L'option automatique lance **ptp4l** en mode E2E. Celle-ci passera en mode P2P si une requête de délai est reçue.



NOTE

Toutes les horloges sur un seul chemin de communication **PTP** doivent utiliser le même mécanisme de mesure de délai. Des avertissements seront imprimés dans les circonstances suivantes :

- Lorsqu'une requête de délai est reçue sur un port utilisant le mécanisme E2E.
- Lorsqu'une requête de délai E2E est reçue sur un port utilisant le mécanisme P2P.

17.3. UTILISER PTP EN INTERFACES MULTIPLES

Quand vous utilisez PTP en interfaces multiples à travers divers réseaux, il vous faudra passer du mode *reverse path forwarding* (transmission chemin inversé) au mode loose (moins strict). Red Hat Enterprise Linux 7 utilise par défaut *Strict Reverse Path Forwarding*, puis Strict Reverse Path recommendé par RFC 3704, Ingress Filtering for Multihomed Networks. Voir la section Reverse Path Forwarding du guide Red Hat Enterprise Linux 7 Security Guide pour obtenir plus d'informations.

L'utilitaire **sysctl** a l'habitude de lire et d'écrire des valeurs dans le noyau. Vous pouvez effectuer des modifications à un système en cours d'exécution par les commandes **sysctl** directement en ligne de commandes, et des changements permanents en ajoutant des lignes au fichier **/etc/sysctl.conf**.

- Pour passer au mode loose (non strict) de filtrage global, saisir les commandes suivantes en tant qu'utilisateur **root** :

```
~]# sysctl -w net.ipv4.conf.default.rp_filter=2
sysctl -w net.ipv4.conf.all.rp_filter=2
```

- Pour changer le mode « Reverse path filtering » (filtrage chemin inversé) pour chaque interface de réseau, exécuter la commande **net.ipv4.*interface*.rp_filter** sur toutes les interfaces PTP. Ainsi, pour une interface ayant comme nom de périphérique **em1** :

```
~]# sysctl -w net.ipv4.conf.em1.rp_filter=2
```

Pour rendre ces configurations persistentes au démarrage, modifier le fichier **/etc/sysctl.conf**. Ainsi, pour changer le mode sur toutes les interfaces, ouvrir le fichier **/etc/sysctl.conf** à l'aide d'un éditeur en tant qu'utilisateur **root** et ajouter la ligne suivante :

```
net.ipv4.conf.all.rp_filter=2
```

Pour modifier certaines interfaces, saisir plusieurs lignes suivant le format :

```
net.ipv4.conf.interface.rp_filter=2
```

17.4. SPÉCIFIER UN FICHIER DE CONFIGURATION

Les options de la ligne de commandes ainsi que d'autres options qui ne peuvent pas être définies sur la ligne de commande, peuvent être définies dans un fichier de configuration optionnel.

Aucun fichier de configuration n'est lu par défaut, ce fichier doit donc être spécifié pendant le runtime avec l'option **-f**. Par exemple :

```
~]# ptp4l -f /etc/ptp4l.conf
```

Un fichier de configuration équivalent aux options **-i eth3 -m -S** affichées ci-dessus peut ressembler à ceci :

```
~]# cat /etc/ptp4l.conf
[global]
verbose               1
time_stamping         software
[eth3]
```

17.5. UTILISER LE CLIENT DE GESTION PTP

Le client de gestion **PTP**, **pmc**, peut être utilisé pour obtenir des informations supplémentaires de **ptp4l** comme suit :

```
~]# pmc -u -b 0 'GET CURRENT_DATA_SET'
sending: GET CURRENT_DATA_SET
        90e2ba.fffe.20c7f8-0 seq 0 RESPONSE MANAGMENT CURRENT_DATA_SET
                stepsRemoved        1
                offsetFromMaster  -142.0
                meanPathDelay     9310.0
```

```
~]# pmc -u -b 0 'GET TIME_STATUS_NP'
sending: GET TIME_STATUS_NP
        90e2ba.fffe.20c7f8-0 seq 0 RESPONSE MANAGMENT TIME_STATUS_NP
                master_offset              310
                ingress_time               1361545089345029441
                cumulativeScaledRateOffset   +1.000000000
                scaledLastGmPhaseChange    0
                gmTimeBaseIndicator        0
                lastGmPhaseChange          0x0000'0000000000000000.0000
                gmPresent                  true
                gmIdentity                 00a069.fffe.0b552d
```

Définir l'option **-b** sur **zéro** limite la valeur « Boundary » à l'instance **ptp4l** exécutée localement. Une valeur « Boundary » plus importante récupérera également les informations des nœuds **PTP** plus éloignés de l'horloge locale. Les informations récupérables incluent :

- **stepsRemoved** est le nombre de chemins de communication vers le « Grandmaster Clock ».
- **offsetFromMaster** et master_offset est le dernier décalage mesuré de l'horloge depuis l'horloge maître en nanosecondes.
- **meanPathDelay** est le délai estimé des messages de synchronisation envoyés depuis l'horloge maître en nanosecondes.
- Si **gmPresent** est « true », l'horloge **PTP** est synchronisée sur une horloge maitre, l'horloge locale ne sera plus le « Grandmaster Clock ».
- **gmIdentity** est l'identité du « Grandmaster ».

Pour afficher une liste complète des commandes **pmc**, veuillez saisir ce qui suit en tant qu'utilisateur **root** :

```
~]# pmc help
```

Des informations supplémentaires sont disponibles sur la page man **pmc(8)**.

17.6. SYNCHRONISER LES HORLOGES

Le programme **phc2sys** est utilisé pour synchroniser l'horloge système sur le l'horloge matérielle **PTP** (PHC) sur le NIC. Le service **phc2sys** est configuré dans le fichier de configuration **/etc/sysconfig/phc2sys**. Le paramètre par défaut du fichier **/etc/sysconfig/phc2sys** est comme suit :

```
OPTIONS="-a -r"
```

L'option **-a** amène **phc2sys** à lire les horloges devant être synchronisées à partir de l'application **ptp4l**. Il suivra les changements des états des ports **PTP**, ajustant la synchronisation entre horloges matérielles NIC en conséquence. L'horloge système n'est pas synchronsée, à moins que l'option **-r** soit également spécifiée. Si vous souhaitez que l'horloge système soit une source de temps éligible, veuillez spécifier l'option **-r** deux fois.

Après avoir appliqué des changements à **/etc/sysconfig/phc2sys**, redémarrez le service **phc2sys** à partir de la ligne de commande en exécutant une commande en tant qu'utilisateur **root** :

```
~]# systemctl restart phc2sys
```

Sous des circonstances normales, utilisez les commandes **systemctl** pour lancer, arrêter, et redémarrer le service **phc2sys**.

Si vous ne souhaitez pas lancer **phc2sys** en tant que service, vous pouvez le lancer à partir de la ligne de commande. Par exemple, saisissez la commande suivante en tant qu'utilisateur **root** :

```
~]# phc2sys -a -r
```

L'option **-a** amène **phc2sys** à lire les horloges devant être synchronisées à partir de l'application **ptp4l**. Si vous souhaitez que l'horloge système puisse être une source de temps éligible, veuillez spécifier l'option **-r** deux fois.

Alternativement, veuillez utiliser l'option **-s** pour synchroniser l'horloge système avec l'horloge matérielle **PTP** d'une l'interface spécifique. Par exemple :

```
~]# phc2sys -s eth3 -w
```

L'option **-w** attend que l'application **ptp4l** synchronise l'horloge **PTP** et récupère le décalage TAI de l'horloge UTC de **ptp4l**.

Normalement, **PTP** opère sous l'échelle horaire du temps atomique international (*International Atomic Time*, ou TAI), tandis que l'horloge système reste dans le temps universel coordonné (*Coordinated Universal Time*, ou UTC). Le décalage actuel entre les heures TAI et UTC est de 36 secondes. Le décalage change lorsque des secondes intercalaires sont insérées ou supprimées, ce qui se produit habituellement au bout de quelques années régulièrement. L'option **-O** doit être utilisée pour paramétrer ce décalage manuellement lorsque l'option **-w** n'est pas utilisée, comme suit :

```
~]# phc2sys -s eth3 -O -36
```

Une fois le servo **phc2sys** en état verrouillé, l'horloge ne sera pas décalée, à moins que l'option **-S** ne soit utilisée. Cela signifie que le programme **phc2sys** devrait être lancé après que le programme **ptp4l** ait synchronisé l'horloge matérielle **PTP**. Cependant avec **-w**, il n'est pas utile de lancer **phc2sys** après **ptp4l** comme celui-ci attendra que l'horloge soit synchronisée.

Le programme **phc2sys** peut également être lancé en tant que service en exécutant :

```
~]# systemctl start phc2sys
```

Lors d'une exécution en tant que service, les options sont spécifiées dans le fichier **/etc/sysconfig/phc2sys**. On peut trouver davantage d'informations sur les différentes options **phc2sys** dans la page man **phc2sys(8)**.

Remarquez que les examples dans cette section supposent que la commande soit exécutée sur un système esclave ou sur un port esclave.

17.7. VÉRIFIER LA SYNCHRONISATION DU TEMPS

Lorsque la synchronisation du temps **PTP** fonctionne correctement, de nouveaux messages avec des décalages et des ajustements de fréquence seront imprimés de manière périodique sur les sorties **ptp4l** et **phc2sys** (si l'horodatage matériel est utilisé). Ces valeurs convergeront éventuellement après une courte période. On peut voir ces messages dans le fichier **/var/log/messages**. En voici un exemple :

```
ptp4l[352.359]: selected /dev/ptp0 as PTP clock
ptp4l[352.361]: port 1: INITIALIZING to LISTENING on INITIALIZE
ptp4l[352.361]: port 0: INITIALIZING to LISTENING on INITIALIZE
ptp4l[353.210]: port 1: new foreign master 00a069.fffe.0b552d-1
ptp4l[357.214]: selected best master clock 00a069.fffe.0b552d
ptp4l[357.214]: port 1: LISTENING to UNCALIBRATED on RS_SLAVE
ptp4l[359.224]: master offset       3304 s0 freq      +0 path delay
9202
ptp4l[360.224]: master offset       3708 s1 freq  -29492 path delay
9202
ptp4l[361.224]: master offset      -3145 s2 freq  -32637 path delay
9202
ptp4l[361.224]: port 1: UNCALIBRATED to SLAVE on MASTER_CLOCK_SELECTED
ptp4l[362.223]: master offset       -145 s2 freq  -30580 path delay
9202
ptp4l[363.223]: master offset       1043 s2 freq  -29436 path delay
8972
ptp4l[364.223]: master offset        266 s2 freq  -29900 path delay
9153
ptp4l[365.223]: master offset        430 s2 freq  -29656 path delay
9153
ptp4l[366.223]: master offset        615 s2 freq  -29342 path delay
9169
ptp4l[367.222]: master offset       -191 s2 freq  -29964 path delay
9169
ptp4l[368.223]: master offset        466 s2 freq  -29364 path delay
9170
ptp4l[369.235]: master offset         24 s2 freq  -29666 path delay
9196
```

```
ptp4l[370.235]: master offset       -375 s2 freq  -30058 path delay
9238
ptp4l[371.235]: master offset        285 s2 freq  -29511 path delay
9199
ptp4l[372.235]: master offset        -78 s2 freq  -29788 path delay
9204
```

Ci-dessous figure un exemple de la sortie de **phc2sys** :

```
phc2sys[526.527]: Waiting for ptp4l...
phc2sys[527.528]: Waiting for ptp4l...
phc2sys[528.528]: phc offset     55341 s0 freq      +0 delay   2729
phc2sys[529.528]: phc offset     54658 s1 freq  -37690 delay   2725
phc2sys[530.528]: phc offset       888 s2 freq  -36802 delay   2756
phc2sys[531.528]: phc offset      1156 s2 freq  -36268 delay   2766
phc2sys[532.528]: phc offset       411 s2 freq  -36666 delay   2738
phc2sys[533.528]: phc offset       -73 s2 freq  -37026 delay   2764
phc2sys[534.528]: phc offset        39 s2 freq  -36936 delay   2746
phc2sys[535.529]: phc offset        95 s2 freq  -36869 delay   2733
phc2sys[536.529]: phc offset      -359 s2 freq  -37294 delay   2738
phc2sys[537.529]: phc offset      -257 s2 freq  -37300 delay   2753
phc2sys[538.529]: phc offset       119 s2 freq  -37001 delay   2745
phc2sys[539.529]: phc offset       288 s2 freq  -36796 delay   2766
phc2sys[540.529]: phc offset      -149 s2 freq  -37147 delay   2760
phc2sys[541.529]: phc offset      -352 s2 freq  -37395 delay   2771
phc2sys[542.529]: phc offset       166 s2 freq  -36982 delay   2748
phc2sys[543.529]: phc offset        50 s2 freq  -37048 delay   2756
phc2sys[544.530]: phc offset       -31 s2 freq  -37114 delay   2748
phc2sys[545.530]: phc offset      -333 s2 freq  -37426 delay   2747
phc2sys[546.530]: phc offset       194 s2 freq  -36999 delay   2749
```

Il existe également une directive pour **ptp4l**, **`summary_interval`**, permettant de réduire les sorties et de n'imprimer que les statistiques, car normalement un message sera imprimé à peu près à chaque seconde. Ainsi, pour réduire les sorties à une toutes les **`1024`** secondes, veuillez ajouter la ligne suivante au fichier **`/etc/ptp4l.conf`** :

```
summary_interval 10
```

Ci-dessous figure un exemple de la sortie de **ptp4l** avec **`summary_interval 6`** :

```
ptp4l: [615.253] selected /dev/ptp0 as PTP clock
ptp4l: [615.255] port 1: INITIALIZING to LISTENING on INITIALIZE
ptp4l: [615.255] port 0: INITIALIZING to LISTENING on INITIALIZE
ptp4l: [615.564] port 1: new foreign master 00a069.fffe.0b552d-1
ptp4l: [619.574] selected best master clock 00a069.fffe.0b552d
ptp4l: [619.574] port 1: LISTENING to UNCALIBRATED on RS_SLAVE
ptp4l: [623.573] port 1: UNCALIBRATED to SLAVE on MASTER_CLOCK_SELECTED
ptp4l: [684.649] rms  669 max 3691 freq -29383 ± 3735 delay  9232 ± 122
ptp4l: [748.724] rms  253 max  588 freq -29787 ± 221 delay  9219 ± 158
ptp4l: [812.793] rms  287 max  673 freq -29802 ± 248 delay  9211 ± 183
ptp4l: [876.853] rms  226 max  534 freq -29795 ± 197 delay  9221 ± 138
ptp4l: [940.925] rms  250 max  562 freq -29801 ± 218 delay  9199 ± 148
ptp4l: [1004.988] rms  226 max  525 freq -29802 ± 196 delay  9228 ± 143
```

```
ptp4l: [1069.065] rms  300 max  646 freq -29802 ± 259 delay  9214 ± 176
ptp4l: [1133.125] rms  226 max  505 freq -29792 ± 197 delay  9225 ± 159
ptp4l: [1197.185] rms  244 max  688 freq -29790 ± 211 delay  9201 ± 162
```

Pour réduire la sortie de **phc2sys**, on peut ajouter l'option **-u** comme suit :

```
~]# phc2sys -u summary-updates
```

Quand *summary-updates* est le nombre de mises à jour d'horloge à inclure dans le résumé des statistiques. Voir l'exemple ci-dessous :

```
~]# phc2sys -s eth3 -w -m -u 60
phc2sys[700.948]: rms 1837 max 10123 freq -36474 ± 4752 delay  2752 ±  16
phc2sys[760.954]: rms  194 max  457 freq -37084 ± 174 delay  2753 ±  12
phc2sys[820.963]: rms  211 max  487 freq -37085 ± 185 delay  2750 ±  19
phc2sys[880.968]: rms  183 max  440 freq -37102 ± 164 delay  2734 ±  91
phc2sys[940.973]: rms  244 max  584 freq -37095 ± 216 delay  2748 ±  16
phc2sys[1000.979]: rms  220 max  573 freq -36666 ± 182 delay  2747 ±  43
phc2sys[1060.984]: rms  266 max  675 freq -36759 ± 234 delay  2753 ±  17
```

17.8. SERVIR LE TEMPS PTP AVEC NTP

Le démon **ntpd** peut être configuré pour distribuer le temps à partir de l'horloge système synchronisée par **ptp4l** ou **phc2sys** en utilisant le pilote de l'horloge de référence LOCAL. Pour empêcher **ntpd** d'ajuster l'horloge système, le fichier **ntp.conf** ne doit pas spécifier de serveur **NTP**. Voici un petit exemple de **ntp.conf** :

```
~]# cat /etc/ntp.conf
server   127.127.1.0
fudge    127.127.1.0 stratum 0
```



NOTE

Lorsque le programme du client **DHCP**, **dhclient**, reçoit une liste de serveurs **NTP** du serveur **DHCP**, il les ajoute au fichier **ntp.conf** et redémarre le service. Pour désactiver cette fonctionnalité, veuillez ajouter **PEERNTP=no** à **/etc/sysconfig/network**.

17.9. SERVIR LE TEMPS NTP AVEC PTP

La synchronisation **NTP** avec **PTP** est également possible dans le sens contraire. Lorsque **ntpd** est utilisé pour synchroniser l'horloge système, **ptp4l** peut être configuré avec l'option **priority1** (ou avec d'autres options d'horloge incluses dans le meilleur algorithme d'horloge maître) pour être l'horloge « Grandmaster » et distribuer le temps depuis l'horloge système via **PTP** :

```
~]# cat /etc/ptp4l.conf
[global]
priority1 127
[eth3]
# ptp4l -f /etc/ptp4l.conf
```

Avec l'horodatage matériel, **phc2sys** doit être utilisé pour synchroniser l'horloge matérielle **PTP** avec l'horloge système. Si **phc2sys** est exécuté en tant que service, modifiez le fichier de configuration

/etc/sysconfig/phc2sys. Le paramètre par défaut dans le fichier **/etc/sysconfig/phc2sys** est comme suit :

```
OPTIONS="-a -r"
```

En tant qu'utilisateur **root**, modifiez cette ligne comme suit :

```
~]# vi /etc/sysconfig/phc2sys
OPTIONS="-a -r -r"
```

L'option **-r** est utilisée deux fois pour permettre la synchronisation de l'horloge matérielle **PTP** sur le NIC à partir de l'horloge système. Veuillez redémarrer le service **phc2sys** pour que les changements puissent entrer en vigueur :

```
~]# systemctl restart phc2sys
```

Pour empêcher des changements rapides de fréquence de l'horloge **PTP**, la synchronisation de l'horloge système peut être relâchée en utilisant des constantes **P** (proportionelle) et **I** (intégrale) plus basses pour le servo PI :

```
~]# phc2sys -a -r -r -P 0.01 -I 0.0001
```

17.10. SYNCHRONISER AVEC LE TEMPS PTP OU NTP EN UTILISANT TIMEMASTER

Lorsque de multiples domaines **PTP** sont disponibles sur le réseau, ou lorsqu'un basculement sur **NTP** est nécessaire, le programme **timemaster** peut être utilisé pour synchroniser l'horloge système sur toutes les sources de temps disponibles. Le temps **PTP** est fourni par **phc2sys** et **ptp4l** via des horloges de référence de pilotes de mémoire partagée *shared memory driver* (SHM) sur **chronyd** ou **ntpd** (selon le démon **NTP** qui a été configuré sur le système). Le démon **NTP** peut ensuite comparer toutes les sources de temps, **PTP** et **NTP**, et utiliser les meilleures sources pour synchroniser l'horloge système.

Lors du démarrage, **timemaster** lit un fichier de configuration qui spécifie les sources de temps **NTP** et **PTP**, vérifie quelles interfaces réseau possèdent la leur ou partagent une horloge matérielle **PTP** (PHC), génère des fichiers de configuration pour **ptp4l** et **chronyd** ou pour **ntpd**, et lance les processus **ptp4l**, **phc2sys**, et **chronyd** ou **ntpd** selon les besoins. Il supprimera également les fichiers de configuration générés lors de la fermeture. Il écrit des fichiers de configuration pour **chronyd**, **ntpd**, et **ptp4l** sur **/var/run/timemaster/**.

17.10.1. Lancer timemaster en tant que service

Pour lancer **timemaster** en tant que service, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl start timemaster
```

Ceci lira les options dans **/etc/timemaster.conf**. Pour obtenir des informations supplémentaires sur la gestion des services système sur Red Hat Enterprise Linux 7, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

17.10.2. Comprendre le fichier de configuration timemaster

Red Hat Enterprise Linux fournit un fichier **/etc/timemaster.conf** par défaut avec un certain nombre de sections contenant les options par défaut. Les en-têtes de section sont entre crochets.

Pour afficher la configuration par défaut, veuillez exécuter une commande comme suit :

```
~]$ less /etc/timemaster.conf
# Configuration file for timemaster

#[ntp_server ntp-server.local]
#minpoll 4
#maxpoll 4

#[ptp_domain 0]
#interfaces eth0

[timemaster]
ntp_program chronyd

[chrony.conf]
include /etc/chrony.conf

[ntp.conf]
includefile /etc/ntp.conf

[ptp4l.conf]

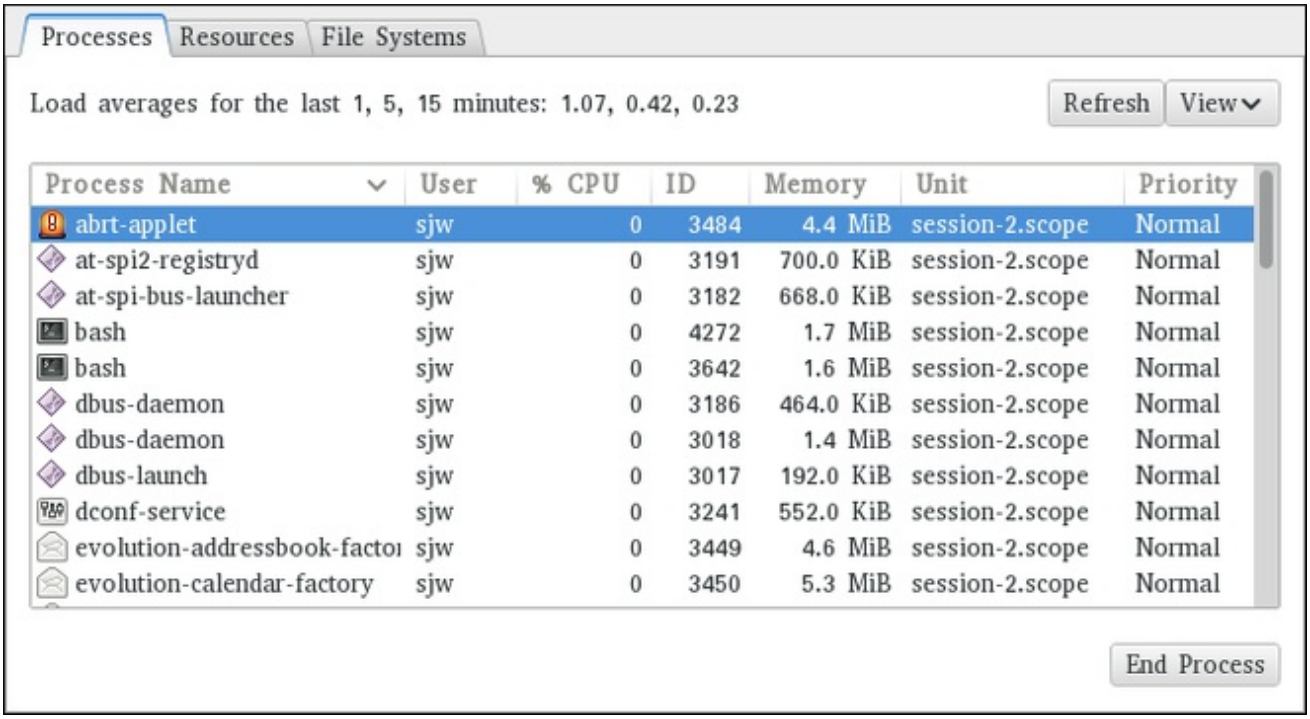
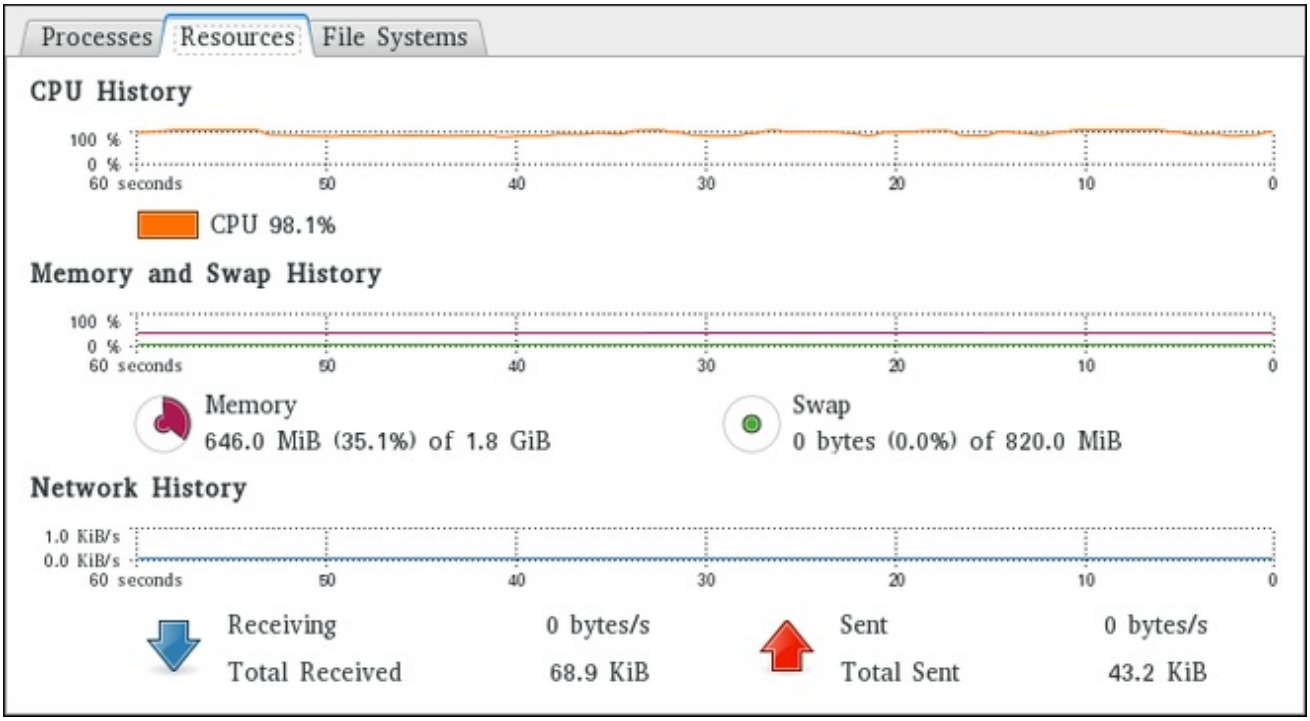
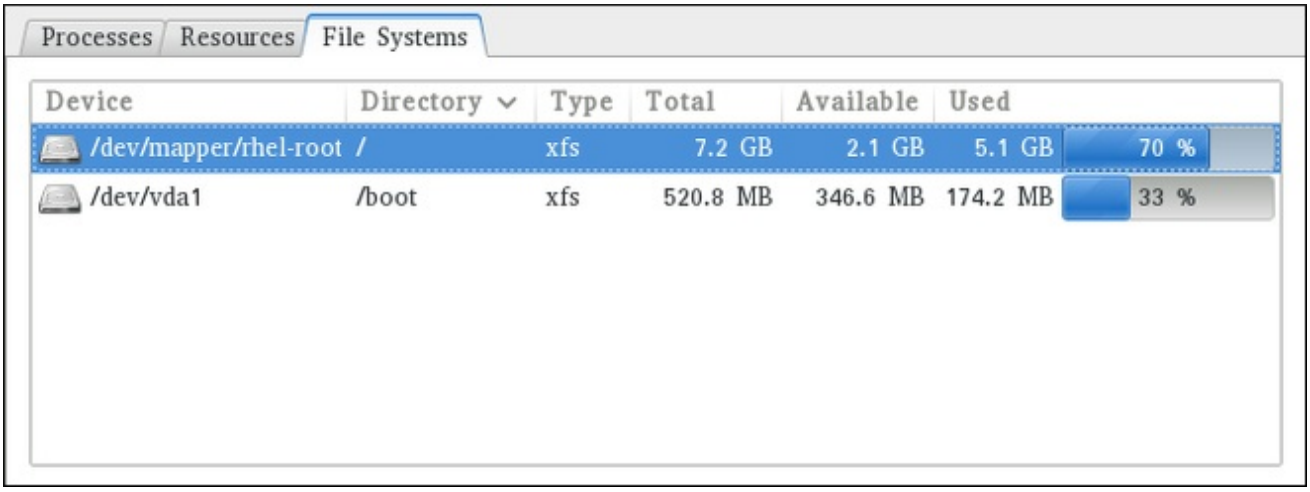
[chronyd]
path /usr/sbin/chronyd
options -u chrony

[ntpd]
path /usr/sbin/ntpd
options -u ntp:ntp -g

[phc2sys]
path /usr/sbin/phc2sys

[ptp4l]
path /usr/sbin/ptp4l
```

Remarquez la section suivante :

```
[ntp_server address]
```

Ceci est un exemple de section de serveur **NTP**, « ntp-server.local » est un exemple de nom d'hôte pour un serveur **NTP** sur le réseau LAN local. Ajoutez davantage de sections selon les besoins en utilisant un nom d'hôte ou une adresse **IP** faisant partie du nom de la section. Remarquez que les valeurs d'interrogation courtes dans cet exemple de section ne sont pas convenables pour un serveur public, veuillez consulter le Chapitre 16, *Configurer NTP à l'aide de ntpd* pour une explication sur des valeurs **minpoll** et **maxpoll** qui conviennent.

Remarquez la section suivante :

```
[ptp_domain number]
```

Un « PTP domain » est un groupe d'une ou plusieurs horloges **PTP** synchronisées. Elles peuvent être ou ne pas être synchronisées avec des horloges dans un autre domaine. Les horloges configurées avec le même numéro de domaine compose celui-ci. Cela inclut une horloge **PTP** « Grandmaster » . Le numéro de domaine dans chaque section « PTP domain » doit correspondre à l'un des domaines **PTP** configuré sur le réseau.

Une instance de **ptp4l** est lancée pour chaque interface possédant sa propre horloge **PTP** et l'horodatage matériel est activé automatiquement. Les interfaces qui prennent en charge l'horodatage matériel possèdent une horloge **PTP** (PHC) attachée, cependant il est possible pour un groupe d'interfaces sur un NIC de partager une horloge PHC. Une instance **ptp4l** séparée sera lancée pour chaque groupe d'interfaces partageant le même PHC et pour chaque interface qui prend en charge l'horodatage logiciel uniquement. Toutes les instances **ptp4l** sont configurées pour être exécutées en tant qu'esclaves. Si une interface avec horodatage matériel est spécifiée dans plus d'un domaine **PTP**, alors seule l'instance **ptp4l** créée aura l'horodatage matériel activé.

Remarquez la section suivante :

```
[timemaster]
```

La configuration **timemaster** par défaut inclut le **ntpd** du système et la configuration chrony (**/etc/ntp.conf** ou **/etc/chronyd.conf**) afin d'inclure la configuration des restrictions d'accès les clés d'authentification. Cela signifie que tout serveur **NTP** spécifié sera également utilisé avec **timemaster**.

Les en-têtes de section sont comme suit :

- **[ntp_server ntp-server.local]** — spécifie les intervalles d'interrogation de ce serveur. Crée des sections supplémentaires selon les besoins. Inclut le nom d'hôte ou l'adresse **IP** dans l'en-tête de section.
- **[ptp_domain 0]** — spécifier les interfaces sur lesquelle les horloges **PTP** sont configurées pour ce domaine. Crée des sections supplémentaires avec le numéro de domaine approprié, selon les besoins.
- **[timemaster]** — spécifie le démon **NTP** à utiliser. Les valeurs possibles sont **chronyd** et **ntpd**.
- **[chrony.conf]** — spécifie tout paramètre supplémentaire à copier dans le fichier de configuration généré pour **chronyd**.
- **[ntp.conf]** — spécifie tout paramètre supplémentaire à copier dans le fichier de configuration généré pour **ntpd**.
- **[ptp4l.conf]** — spécifie les options à copier dans le fichier de configuration généré pour **ptp4l**.
- **[chronyd]** — spécifie tout paramètre supplémentaire à passer sur la ligne de commande pour **chronyd**.
- **[ntpd]** — spécifie tout paramètre supplémentaire à passer sur la ligne de commande pour **ntpd**.
- **[phc2sys]** — spécifie tout paramètre supplémentaire à passer sur la ligne de commande pour **phc2sys**.

- **[ptp4l]** — spécifie tout paramètre supplémentaire à passer sur la ligne de commande pour toutes les instances de **ptp4l**.

Les en-têtes de section et le contenu sont expliqués en détails dans la page man **timemaster(8)**.

17.10.3. Configurer les options timemaster

Procédure 17.1. Modifier le fichier de configuration timemaster

1. Pour modifier la configuration par défaut, veuillez ouvrir le fichier **/etc/timemaster.conf** pour effectuer des modifications en tant qu'utilisateur **root** :

```
~]# vi /etc/timemaster.conf
```

2. Pour chaque serveur **NTP** que vous souhaitez contrôler en utilisant **timemaster**, veuillez créer des sections **[ntp_server *address*]**. Remarquez que les valeurs d'interrogation courtes dans la section exemple ne conviennent pas à un serveur public, veuillez consulter le Chapitre 16, *Configurer NTP à l'aide de ntpd* pour une explication sur des valeurs **minpoll** et **maxpoll** qui conviennent.

3. Pour ajouter des interfaces qui doivent être utilisées dans un domaine, veuillez modifier la section **#[ptp_domain 0]** et ajouter les interfaces. Créer des domaines supplémentaires selon les besoins. Par exemple :

```
[ptp_domain 0]
       interfaces eth0

       [ptp_domain 1]
       interfaces eth1
```

4. Si vous devez d'utiliser **ntpd** comme démon **NTP** sur ce système, veuillez modifier l'entrée par défaut dans la section **[timemaster]** de **chronyd** à **ntpd**. Veuillez consulter le Chapitre 15, *Configurer NTP en utilisant Chrony Suite* pour obtenir des informations sur les différences entre ntpd et chronyd.

5. Si vous utilisez **chronyd** comme serveur **NTP** sur ce système, ajoutez toute option supplémentaire sous l'entrée par défaut **include /etc/chrony.conf** dans la section **[chrony.conf]**. Modifiez l'entrée par défaut **include** si le chemin vers **/etc/chrony.conf** a changé.

6. Si vous utilisez **ntpd** comme serveur **NTP** sur ce système, ajoutez toute option supplémentaire sous l'entrée par défaut **include /etc/ntp.conf** dans la section **[ntp.conf]**. Modifiez l'entréer par défaut **include** si le chemin vers **/etc/ntp.conf** a changé.

7. Dans la section **[ptp4l.conf]**, ajoutez les options à copier au fichier de configuration généré pour **ptp4l**. Ce chapitre documente les options communes et davantage d'informations sont disponibles sur la page man de **ptp4l(8)**.

8. Dans la section **[chronyd]**, veuillez ajouter les options de ligne de commande à passer sur **chronyd** lorsque appelé par **timemaster**. Veuillez consulter le Chapitre 15, *Configurer NTP en utilisant Chrony Suite* pour obtenir des informations sur l'utilisation de **chronyd**.

9. Dans la section **[ntpd]**, veuillez ajouter les options de ligne de commande à passer sur **ntpd** lorsque appelé par **timemaster**. Veuillez consulter le Chapitre 16, *Configurer NTP à l'aide de ntpd* pour obtenir des informations sur l'utilisation de **ntpd**.

10. Dans la section **[phc2sys]**, ajoutez les options de ligne de commande à passer sur **phc2sys** lorsque appelé par **timemaster**. Ce chapitre documente les options communes et davantage d'informations sont disponibles sur la page man de **phy2sys(8)**.

11. Dans la section **[ptp4l]**, ajoutez les options de ligne de commande à passer sur **ptp4l** lorsque appelé par **timemaster**. Ce chapitre documente les options communes et davantage d'informations sont disponibles sur la page man de **ptp4l(8)**.

12. Enregistrez le fichier de configuration et redémarrez **timemaster** en saisissant la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl restart timemaster
```

17.11. AMÉLIORER LA PRÉCISION

Précédemment, les résultats de tests indiquaient que la désactivation de la capacité de noyau sans tic pouvait fortement améliorer la stabilité de l'horloge du système, améliorant la précision de la synchronisation **PTP** par la même occasion (en contrepartie d'une augmentation de la consommation de l'alimentation). Le mode sans tic du noyau peut être désactivé en ajoutant **nohz=off** aux paramètres d'option de démarrage du noyau. Cependant, de récentes améliorations appliquées à **kernel-3.10.0-197.el7** ont fortement amélioré la stabilité de l'horloge système et la différence de stabilité de l'horloge avec et sans **nohz=off** devrait désormais être bien moindre pour la plupart des utilisateurs.

Les applications **ptp4l** et **phc2sys** peuvent être configurées pour utiliser un nouveau servo adaptatif. L'avantage offert par rapport au servo PI est qu'il ne requiert pas de configurer les constantes PI pour bien fonctionner. Pour utiliser ceci pour **ptp4l**, veuillez ajouter la ligne suivante au fichier **/etc/ptp4l.conf** :

```
clock_servo linreg
```

. Après avoir effectué des changements sur **/etc/ptp4l.conf**, redémarrez le service **ptp4l** à partir de la ligne de commande en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl restart ptp4l
```

Pour utiliser ceci pour **phc2sys**, veuillez ajouter la ligne suivante au fichier **/etc/sysconfig/phc2sys** :

```
-E linreg
```

Après avoir effectué des changements à **/etc/sysconfig/phc2sys**, redémarrez le service **phc2sys** à partir de la ligne de commande en passant la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl restart phc2sys
```

17.12. RESSOURCES SUPPLÉMENTAIRES

Les sources d'informations suivantes offrent des ressources supplémentaires concernant **PTP** et les outils **ptp4l**.

17.12.1. Documentation installée

- Page man **`ptp4l(8)`** — décrit les options **ptp4l**, y compris le format du fichier de configuration.
- Page man **`pmc(8)`** — décrit le client de gestion **PTP** et ses options de commande.
- Page man **`phc2sys(8)`** — décrit un outil pour synchroniser l'horloge système avec une horloge matérielle **PTP** (PHC).
- Page man **`timemaster(8)`** — décrit un programme qui utilise **ptp4l** et **phc2sys** pour synchroniser l'horloge système à l'aide de **`chronyd`** ou **`ntpd`**.

17.12.2. Sites Web utiles

http://www.nist.gov/el/isd/ieee/ieee1588.cfm

Standard IEEE 1588.

PARTIE VI. SURVEILLANCE ET AUTOMATISATION

Cette partie décrit les différents outils permettant aux administrateurs système de surveiller les performances du système, d'automatiser des tâches système et de rapporter les bogues.

CHAPITRE 18. OUTILS DE SURVEILLANCE DU SYSTÈME

Pour configurer le système, les administrateurs système ont souvent besoin de déterminer la quantité de mémoire libre, la quantité d'espace disque libre, la manière par laquelle le disque dur est partitionné, ou quels sont les processus en cours d'exécution.

18.1. AFFICHER LES PROCESSUS SYSTÈME

18.1.1. Utiliser la commande ps

La commande **ps** permet d'afficher des informations sur l'exécution de processus. Elle produit une liste statique, ou instantané, de ce qui est cours d'exécution au moment où la commande est exécutée. Si vous souhaitez afficher une liste des processus en cours d'exécution constamment mise à jour, veuillez utiliser la commande **top** ou l'application de surveillance système « **System Monitor** » à la place.

Pour répertorier tous les processus actuellement en cours d'exécution sur le système, y compris les processus appartenant à d'autres utilisateurs, veuillez saisir ce qui suit à l'invite de shell :

```
ps ax
```

Pour chaque processus répertorié, la commande **ps ax** affiche l'ID du processus (**PID**), le terminal qui y est associé (**TTY**), le statut actuel (**STAT**), le temps CPU accumulé (**TIME**), et le nom du fichier exécutable (**COMMAND**). Exemple :

```
~]$ ps ax
PID TTY      STAT   TIME COMMAND
  1 ?        Ss     0:01 /usr/lib/systemd/systemd --switched-root --system
--deserialize 23
  2 ?        S      0:00 [kthreadd]
  3 ?        S      0:00 [ksoftirqd/0]
  5 ?        S>     0:00 [kworker/0:0H][sortie tronquée]
```

Pour afficher le propriétaire à côté de chaque processus, veuillez utiliser la commande suivante :

```
ps aux
```

À l'exception des informations fournies par la commande **ps ax**, **ps aux** affiche le nom d'utilisateur du propriétaire du processus (**USER**), le pourcentage d'utilisation du CPU (**%CPU**) et de la mémoire (**%MEM**), la taille de la mémoire virtuelle en kilo-octets (**VSZ**), la taille de la mémoire physique non swap en kilo-octets (**RSS**), et l'heure ou la date à laquelle le processus a été lancé. Exemple :

```
~]$ ps aux
USER  PID %CPU %MEM    VSZ   RSS TTY   STAT START   TIME COMMAND
root    1  0.3  0.3 134776  6840 ?     Ss   09:28   0:01
/usr/lib/systemd/systemd --switched-root --system --d
root    2  0.0  0.0      0     0 ?     S    09:28   0:00 [kthreadd]
root    3  0.0  0.0      0     0 ?     S    09:28   0:00 [ksoftirqd/0]
root    5  0.0  0.0      0     0 ?     S>   09:28   0:00
[kworker/0:0H][sortie tronquée]
```

Il est aussi possible d'utiliser la commande **ps** en conjonction avec **grep** pour voir si un processus particulier est en cours d'utilisation. Par exemple, pour déterminer si **Emacs** est en cours d'utilisation, veuillez saisir :

```
~]$ ps ax | grep emacs
12056 pts/3    S+     0:00 emacs
12060 pts/2    S+     0:00 grep --color=auto emacs
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **ps**(1).

18.1.2. Utiliser la commande top

La commande **top** affiche une liste en temps réel des processus exécutés sur le système. Elle affiche également des informations supplémentaires sur le temps d'activité du système, l'utilisation actuelle du CPU et de la mémoire, ou le nombre total de processus en cours d'exécution, et vous permet d'effectuer des actions comme le tri de la liste ou l'arrêt d'un processus.

Pour exécuter la commande **top**, veuillez saisir ce qui suit dans une invite de shell :

```
top
```

Pour chaque processus répertorié, la commande **top** affiche l'ID du processus (**PID**), le nom du propriétaire du processus (**USER**), la priorité (**PR**), la valeur nice (**NI**), la quantité de mémoire virtuelle utilisée par le processus (**VIRT**), la quantité de mémoire physique non swap qu'il utilise (**RES**), la quantité de mémoire partagée qu'il utilise (**SHR**), le champ de statut du processus **S**), le pourcentage d'utilisation du CPU (**%CPU**) et de la mémoire (**%MEM**), le temps CPU cummulé (**TIME+**), et le nom du fichier exécutable (**COMMAND**). Exemple :

```
~]$ top
top - 16:42:12 up 13 min,  2 users,  load average: 0.67, 0.31, 0.19
Tasks: 165 total,   2 running, 163 sleeping,   0 stopped,   0 zombie
%Cpu(s): 37.5 us,  3.0 sy,  0.0 ni, 59.5 id,  0.0 wa,  0.0 hi,  0.0 si,
0.0 st
KiB Mem :  1016800 total,    77368 free,   728936 used,   210496
buff/cache
KiB Swap:   839676 total,   776796 free,    62880 used.   122628 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+
COMMAND
 3168 sjw       20   0 1454628 143240  15016 S 20.3 14.1   0:22.53 gnome-
shell
 4006 sjw       20   0 1367832 298876  27856 S 13.0 29.4   0:15.58 firefox
 1683 root      20   0  242204  50464   4268 S  6.0  5.0   0:07.76 Xorg
 4125 sjw       20   0  555148  19820  12644 S  1.3  1.9   0:00.48 gnome-
terminal-
   10 root      20   0       0      0      0 S  0.3  0.0   0:00.39
rcu_sched
 3091 sjw       20   0   37000   1468    904 S  0.3  0.1   0:00.31 dbus-
daemon
 3096 sjw       20   0  129688   2164   1492 S  0.3  0.2   0:00.14 at-
spi2-registr
 3925 root      20   0       0      0      0 S  0.3  0.0   0:00.05
kworker/0:0
```

```
    1 root      20   0  126568   3884   1052 S  0.0  0.4   0:01.61
systemd
    2 root      20   0       0      0      0 S  0.0  0.0   0:00.00
kthreadd
    3 root      20   0       0      0      0 S  0.0  0.0   0:00.00
ksoftirqd/0
    6 root      20   0       0      0      0 S  0.0  0.0   0:00.07
kworker/u2:0[sortie tronquée]
```

Tableau 18.1, « Commandes top interactives » contient des commandes interactives utiles que vous pouvez utiliser avec **top**. Pour obtenir des informations supplémentaires, veuillez consulter la page man de **top**(1).

Tableau 18.1. Commandes top interactives

Commande	Description
Entrée, **Espace**	Réactualise immédiatement l'affichage.
h	Affiche un écran d'aide pour les commandes interactives.
h, **?**	Affiche un écran d'aide pour les fenêtres et les groupes de champs.
k	Arrête un processus. Il vous sera demandé l'ID du processus et le signal à envoyer.
n	Modifie le nombre de processus affichés. Ce nombre vous sera demandé.
u	Trie la liste par utilisateur.
M	Trie la liste selon l'utilisation mémoire.
P	Trie la liste selon l'utilisation du CPU.
q	Termine l'utilitaire et retourne à l'invite du shell.

18.1.3. Utiliser l'outil de surveillance du système « System Monitor »

L'onglet **Processus** de l'outil de surveillance système « **System Monitor** » vous permet d'afficher de rechercher, de modifier la priorité, et d'arrêter des processus à partir de l'interface utilisateur graphique.

Pour lancer l'outil **System Monitor** à partir de la ligne de commande, veuillez saisir **gnome-system-monitor** à l'invite de shell. L'outil **System Monitor** apparaît. Alternativement, si le bureau GNOME est utilisé, appuyez sur la touche **Super** pour aller sur la « Vue d'ensemble des activités », saisissez **System Monitor** puis appuyez sur **Entrée**. L'outil **System Monitor** apparaît. La touche **Super** peut se trouver sous diverses formes selon le clavier ou du matériel, mais le plus souvent, il s'agit de la touche Windows ou de la touche de Commande, habituellement à gauche de la **Barre d'espace**.

Cliquez sur l'onglet **Processus** pour afficher la liste des processus en cours d'exécution.



Figure 18.1. System Monitor — Processus

Pour chaque processus répertorié, l'outil **System Monitor** affiche son nom (**`Process Name`**), statut actuel (**`Status`**), le pourcentage d'utilisation du CPU (**`% CPU`**), la valeur nice (**`Nice`**), L'ID du processus (**`ID`**), l'utilisation de la mémoire (**`Memory`**), le canal dans lequel le processus attend (**`Waiting Channel`**), ainsi que des détails supplémentaires sur la session (**`Session`**). Pour trier les informations sur une colonne spécifique par ordre croissant, veuillez cliquer sur le nom de cette colonne. Cliquez à nouveau sur le nom de la colonne pour basculer le tri entre un ordre croissant et décroissant.

Par défaut, l'outil **System Monitor** affiche une liste de processus appartenant à l'utilisateur actuel. La sélection de diverses options dans le menu **Affichage** permet :

- d'afficher les processus actifs uniquement,
- d'afficher tous les processus,
- d'afficher vos processus,
- d'afficher les dépendances des processus,

En outre, deux boutons permettent de :

- Réactualiser la liste des processus,
- terminer un processus en le sélectionnant dans la liste et en cliquant sur le bouton **`Terminer le processus`**.

18.2. AFFICHER L'UTILISATION DE LA MÉMOIRE

18.2.1. Utiliser la commande free

La commande **`free`** permet d'afficher la quantité de mémoire libre et utilisée sur le système. Pour faire cela, veuillez saisir ce qui suit dans une invite de shell :

```
free
```

La commande **free** fournit à la fois des informations sur la mémoire physique (**Mem**) et sur l'espace swap (**Swap**). Elle affiche le montant de mémoire total (**total**), ainsi que le montant de mémoire utilisée (**used**), la mémoire libre (**free**), la mémoire partagée (**shared**), les mémoires tampon et cache ajoutées ensemble (**buff/cache**), et ce qui reste de disponible (**available**). Exemple :

```
~]$ free
              total        used        free      shared  buff/cache
available
Mem:        1016800      727300       84684        3500      204816
124068
Swap:        839676       66920      772756
```

Par défaut, **free** affiche les valeurs en kilo-octets. Pour afficher les valeurs en méga-octets, veuillez ajouter l'option de ligne de commande **-m** :

```
free -m
```

Par exemple :

```
~]$ free -m
              total        used        free      shared  buff/cache
available
Mem:            992         711          81           3         200
120
Swap:           819          65         754
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **free**(1).

18.2.2. Utiliser l'outil de surveillance du système « System Monitor »

L'onglet **Ressources** de l'outil **System Monitor** permet d'afficher la quantité de mémoire libre et utilisée sur le système.

Pour lancer l'outil **System Monitor** à partir de la ligne de commande, veuillez saisir **gnome-system-monitor** à l'invite de shell. L'outil **System Monitor** apparaît. Alternativement, si le bureau GNOME est utilisé, appuyez sur la touche **Super** pour aller sur la « Vue d'ensemble des activités », saisissez **System Monitor** puis appuyez sur **Entrée**. L'outil **System Monitor** apparaît. La touche **Super** peut se trouver sous diverses formes selon le clavier ou du matériel, mais le plus souvent, il s'agit de la touche Windows ou de la touche de Commande, habituellement à gauche de la **Barre d'espace**.

Cliquez sur l'onglet **Ressources** pour afficher l'utilisation de la mémoire système.



Figure 18.2. System Monitor — Ressources

Dans la section **Historique de la mémoire et de l'espace swap**, l'outil **System Monitor** affiche un représentation graphique de l'historique d'utilisation de la mémoire et de l'espace swap, ainsi que le montant total de mémoire physique (**Memory**) et d'espace swap (**Swap**), et la quantité en cours d'utilisation.

18.3. AFFICHER L'UTILISATION DU CPU

18.3.1. Utiliser l'outil de surveillance du système « System Monitor »

L'onglet **Ressources** de l'outil **System Monitor** permet d'afficher l'utilisation actuelle du CPU sur le système.

Pour lancer l'outil **System Monitor** à partir de la ligne de commande, veuillez saisir **gnome-system-monitor** à l'invite de shell. L'outil **System Monitor** apparaît. Alternativement, si le bureau GNOME est utilisé, appuyez sur la touche **Super** pour aller sur la « Vue d'ensemble des activités », saisissez **System Monitor** puis appuyez sur **Entrée**. L'outil **System Monitor** apparaît. La touche **Super** peut se trouver sous diverses formes selon le clavier ou du matériel, mais le plus souvent, il s'agit de la touche Windows ou de la touche de Commande, habituellement à gauche de la **Barre d'espace**.

Cliquez sur l'onglet **Ressources** pour afficher l'utilisation du CPU par le système.

Dans la section **Historique du CPU**, l'outil **System Monitor** affiche un représentation graphique de l'historique d'utilisation du CPU et affiche le pourcentage du CPU actuellement en cours d'utilisation.

18.4. AFFICHER LES PÉRIPHÉRIQUES BLOC ET LES SYSTÈMES DE FICHIERS

18.4.1. Utiliser la commande lsblk

La commande **lsblk** permet d'afficher une liste des périphériques bloc disponibles. Elle fournit des

informations et un meilleur contrôle sur le formatage des sortie que la commande **blkid**. Elle lit les informations d'**udev**, et est donc utilisable par les utilisateurs non **root**. Pour afficher une liste des périphériques bloc, veuillez saisir ce qui suit dans une invite de shell :

```
lsblk
```

Pour chaque périphérique bloc répertorié, la commande **lsblk** affiche le nom du périphérique (**NAME**), le numéro majeur et mineur du périphérique (**MAJ:MIN**), si le périphérique peut être déplacé (**RM**), sa taille (**SIZE**), si le périphérique est accessible en lecture seule (**RO**), son type (**TYPE**), et sur quel emplacement il est monté (**MOUNTPOINT**). Exemple :

```
~]$ lsblk
NAME                      MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sr0                        11:0    1  1024M  0 rom
vda                       252:0    0    20G  0 rom
|-vda1                    252:1    0   500M  0 part /boot
`-vda2                    252:2    0  19.5G  0 part
  |-vg_kvm-lv_root (dm-0) 253:0    0    18G  0 lvm  /
  `-vg_kvm-lv_swap (dm-1) 253:1    0   1.5G  0 lvm  [SWAP]
```

Par défaut, **lsblk** répertorie les périphériques bloc sous un format qui ressemble à une arborescence. Pour afficher les informations dans une liste ordinaire, veuillez ajouter l'option de ligne de commande **-l** :

```
lsblk -l
```

Par exemple :

```
~]$ lsblk -l
NAME                    MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sr0                      11:0    1  1024M  0 rom
vda                     252:0    0    20G  0 rom
vda1                    252:1    0   500M  0 part /boot
vda2                    252:2    0  19.5G  0 part
vg_kvm-lv_root (dm-0) 253:0    0    18G  0 lvm  /
vg_kvm-lv_swap (dm-1) 253:1    0   1.5G  0 lvm  [SWAP]
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **lsblk**(8).

18.4.2. Utiliser la commande blkid

La commande **blkid** permet d'afficher des informations de bas niveau sur les périphériques bloc. Elle exige des privilèges **root**, donc des utilisateurs non **root** doivent utiliser la commande **lsblk**. Pour cela, veuillez saisir ce qui suit à l'invite de shell en tant qu'utilisateur **root** :

```
blkid
```

Pour chaque périphérique bloc répertorié, la commande **blkid** affiche les attributs disponibles, comme son *identifiant unique universel* (**UUID**), le type de système de fichiers (**TYPE**), ou l'étiquette de volume (**LABEL**). Exemple :

```
~]# blkid
/dev/vda1: UUID="7fa9c421-0054-4555-b0ca-b470a97a3d84" TYPE="ext4"
/dev/vda2: UUID="7IvYzk-TnnK-oPjf-ipdD-cofz-DXaJ-gPdgBW"
TYPE="LVM2_member"
/dev/mapper/vg_kvm-lv_root: UUID="a07b967c-71a0-4925-ab02-aebcad2ae824"
TYPE="ext4"
/dev/mapper/vg_kvm-lv_swap: UUID="d7ef54ca-9c41-4de4-ac1b-4193b0c1ddb6"
TYPE="swap"
```

Par défaut, la commande **blkid** répertorie tous les périphériques bloc disponibles. Pour n'afficher des informations que sur un périphérique en particulier, veuillez spécifier le nom du périphérique sur la ligne de commande :

```
blkid device_name
```

Par exemple, pour afficher les informations sur **/dev/vda1**, veuillez saisir en tant qu'utilisateur **root** :

```
~]# blkid /dev/vda1
/dev/vda1: UUID="7fa9c421-0054-4555-b0ca-b470a97a3d84" TYPE="ext4"
```

Vous pouvez également utiliser la commande ci-dessus avec les options de ligne de commande **-p** et **-o udev** pour obtenir des informations plus détaillées. Remarquez que des privilèges **root** sont requis pour exécuter cette commande :

```
blkid -po udev device_name
```

Par exemple :

```
~]# blkid -po udev /dev/vda1
ID_FS_UUID=7fa9c421-0054-4555-b0ca-b470a97a3d84
ID_FS_UUID_ENC=7fa9c421-0054-4555-b0ca-b470a97a3d84
ID_FS_VERSION=1.0
ID_FS_TYPE=ext4
ID_FS_USAGE=filesystem
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **blkid**(8).

18.4.3. Utiliser la commande findmnt

La commande **findmnt** permet d'afficher une liste de systèmes de fichiers actuellement montés. Pour faire cela, veuillez saisir ce qui suit dans une invite de shell :

```
findmnt
```

Pour chaque système de fichiers répertorié, la commande **findmnt** affiche le point de montage cible (**TARGET**), le périphérique source (**SOURCE**), le type de système de fichiers (**FSTYPE**), et les options de montage connexes (**OPTIONS**). Exemple :

```
~]$ findmnt
TARGET                          SOURCE             FSTYPE
OPTIONS
```

```
/                              /dev/mapper/rhel-root
                                                    xfs
rw,relatime,seclabel,attr2,inode64,noquota
├─/proc                        proc                 proc
rw,nosuid,nodev,noexec,relatime
│ ├─/proc/sys/fs/binfmt_misc   systemd-1            autofs
rw,relatime,fd=32,pgrp=1,timeout=300,minproto=5,maxproto=5,direct
│ └─/proc/fs/nfsd              sunrpc               nfsd
rw,relatime
├─/sys                         sysfs                sysfs
rw,nosuid,nodev,noexec,relatime,seclabel
│ ├─/sys/kernel/security       securityfs           securityfs
rw,nosuid,nodev,noexec,relatime
│ ├─/sys/fs/cgroup             tmpfs                tmpfs
rw,nosuid,nodev,noexec,seclabel,mode=755[sortie tronquée]
```

Par défaut, **findmnt** répertorie les systèmes de fichiers sous un format qui ressemble à une arborescence. Pour afficher les informations dans une liste ordinaire, veuillez ajouter l'option de ligne de commande **-l** :

```
findmnt -l
```

Par exemple :

```
~]$ findmnt -l
TARGET                     SOURCE               FSTYPE          OPTIONS
/proc                      proc                 proc
rw,nosuid,nodev,noexec,relatime
/sys                       sysfs                sysfs
rw,nosuid,nodev,noexec,relatime,seclabel
/dev                       devtmpfs             devtmpfs
rw,nosuid,seclabel,size=933372k,nr_inodes=233343,mode=755
/sys/kernel/security       securityfs           securityfs
rw,nosuid,nodev,noexec,relatime
/dev/shm                   tmpfs                tmpfs
rw,nosuid,nodev,seclabel
/dev/pts                   devpts               devpts
rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000
/run                       tmpfs                tmpfs
rw,nosuid,nodev,seclabel,mode=755
/sys/fs/cgroup             tmpfs                tmpfs
rw,nosuid,nodev,noexec,seclabel,mode=755[sortie tronquée]
```

Vous pouvez également choisir de répertorier les systèmes de fichiers d'un type particulier uniquement. Pour faire cela, veuillez ajouter l'option de ligne de commande **-t** suivie d'un type de système de fichiers :

```
findmnt -t type
```

Par exemple, pour répertorier tous les systèmes de fichiers **xfs**, veuillez saisir :

```
~]$ findmnt -t xfs
TARGET  SOURCE                FSTYPE OPTIONS
/       /dev/mapper/rhel-root xfs
```

```
rw,relatime,seclabel,attr2,inode64,noquota
└─/boot /dev/vda1             xfs
rw,relatime,seclabel,attr2,inode64,noquota
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **findmnt**(8).

18.4.4. Utiliser la commande df

La commande **df** permet d'afficher un rapport détaillé sur l'utilisation de l'espace disque du système. Pour faire cela, veuillez saisir ce qui suit dans une invite de shell :

```
df
```

Pour chaque système de fichiers répertorié, la commande **df** affiche son nom (**Filesystem**), sa taille (**1K-blocks** ou **Size**), combien d'espace est utilisé (**Used**), combien d'espace reste disponible (**Available**), le pourcentage d'utilisation de l'espace (**Use%**), et l'endroit où le système de fichiers est monté (**Mounted on**). Exemple :

```
~]$ df
Filesystem                  1K-blocks      Used Available Use% Mounted on
/dev/mapper/vg_kvm-lv_root  18618236   4357360  13315112  25% /
tmpfs                         380376       288    380088   1% /dev/shm
/dev/vda1                     495844     77029    393215  17% /boot
```

Par défaut, la commande **df** affiche la taille de la partition en bloc de 1 kilo-octet et la quantité d'espace disque utilisé et disponible en kilo-octets. Pour afficher ces informations en méga-octets et giga-octets, veuillez ajouter l'option de ligne de commande **-h**, ce qui amène **df** à afficher les valeurs sous un format lisible :

```
df -h
```

Par exemple :

```
~]$ df -h
Filesystem                  Size  Used Avail Use% Mounted on
/dev/mapper/vg_kvm-lv_root   18G  4.2G   13G  25% /
tmpfs                       372M  288K  372M   1% /dev/shm
/dev/vda1                   485M   76M  384M  17% /boot
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **df**(1).

18.4.5. Utiliser la commande du

La commande **du** permet d'afficher la quantité d'espace utilisée par des fichiers dans un répertoire. Pour afficher l'utilisation du disque de chaque sous-répertoire dans le répertoire de travail actuel, veuillez exécuter la commande sans aucune option de ligne de commande supplémentaire :

```
du
```

Par exemple :

```
~]$ du
14972   ./Downloads
4       ./.mozilla/extensions
4       ./.mozilla/plugins
12      ./.mozilla
15004   .
```

Par défaut, la commande **du** affiche l'usage du disque kilo-octets. Pour afficher ces informations en méga-octets et giga-octets, veuillez ajouter l'option de ligne de commande **-h**, ce qui mène l'utilitaire à afficher les valeurs sous un format lisible :

```
du -h
```

Par exemple :

```
~]$ du -h
15M     ./Downloads
4.0K    ./.mozilla/extensions
4.0K    ./.mozilla/plugins
12K     ./.mozilla
15M     .
```

À la fin de la liste, la commande **du** affiche toujours le total du répertoire actuel. Pour afficher cette information uniquement, veuillez ajouter l'option de ligne de commande **-s** :

```
du -sh
```

Par exemple :

```
~]$ du -sh
15M     .
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **du**(1).

18.4.6. Utiliser l'outil de surveillance du système « System Monitor »

L'onglet **Systèmes de fichiers** de l'outil **System Monitor** permet d'afficher les systèmes de fichiers et l'utilisation de l'espace disque dans l'interface utilisateur graphique.

Pour lancer l'outil **System Monitor** à partir de la ligne de commande, veuillez saisir **gnome-system-monitor** à l'invite de shell. L'outil **System Monitor** apparaît. Alternativement, si le bureau GNOME est utilisé, appuyez sur la touche **Super** pour aller sur la « Vue d'ensemble des activités », saisissez **System Monitor** puis appuyez sur **Entrée**. L'outil **System Monitor** apparaît. La touche **Super** peut se trouver sous diverses formes selon le clavier ou du matériel, mais le plus souvent, il s'agit de la touche Windows ou de la touche de Commande, habituellement à gauche de la **Barre d'espace**.

Cliquez sur l'onglet **Systèmes de fichiers** pour afficher une liste des systèmes de fichiers.



Figure 18.3. System Monitor — Systèmes de fichiers

Pour chaque système de fichiers répertorié, l'outil **System Monitor** affiche le périphérique source (**`Device`**), le point de montage cible (**`Directory`**), et le type de système de fichiers (**`Type`**), ainsi que sa taille (**`Total`**), et la quantité d'espace disponible (**`Available`**), et utilisée (**`Used`**).

18.5. AFFICHER LES INFORMATIONS MATÉRIEL

18.5.1. Utiliser la commande lspci

La commande **`free`** permet d'afficher des informations sur les bus PCI et les périphériques qui y sont attachés. Pour faire répertorier tous les périphériques PCI sur le système, veuillez saisir ce qui suit dans une invite de shell :

```
lspci
```

Cette commande affiche une simple liste de périphériques, comme par exemple :

```
~]$ lspci
00:00.0 Host bridge: Intel Corporation 82X38/X48 Express DRAM Controller
00:01.0 PCI bridge: Intel Corporation 82X38/X48 Express Host-Primary PCI
Express Bridge
00:1a.0 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI
Controller #4 (rev 02)
00:1a.1 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI
Controller #5 (rev 02)
00:1a.2 USB Controller: Intel Corporation 82801I (ICH9 Family) USB UHCI
Controller #6 (rev 02)[sortie tronquée]
```

Vous pouvez également utiliser l'option de ligne de commande **`-v`** pour afficher une sortie plus détaillée, ou **`-vv`** pour une sortie très détaillée :

```
lspci -v|-vv
```

Par exemple, pour déterminer le constructeur, le modèle, et la taille de mémoire de la carte vidéo d'un système, veuillez saisir :

```
~]$ lspci -v[sortie tronquée]
```

```
01:00.0 VGA compatible controller: nVidia Corporation G84 [Quadro FX 370]
(rev a1) (prog-if 00 [VGA controller])
        Subsystem: nVidia Corporation Device 0491
        Physical Slot: 2
        Flags: bus master, fast devsel, latency 0, IRQ 16
        Memory at f2000000 (32-bit, non-prefetchable) [size=16M]
        Memory at e0000000 (64-bit, prefetchable) [size=256M]
        Memory at f0000000 (64-bit, non-prefetchable) [size=32M]
        I/O ports at 1100 [size=128]
        Expansion ROM at <unassigned> [disabled]
        Capabilities: <access denied>
        Kernel driver in use: nouveau
        Kernel modules: nouveau, nvidiafb
[sortie tronquée]
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **lspci**(8).

18.5.2. Utiliser la commande lsusb

La commande **lsusb** permet d'afficher des informations sur les bus USB et les périphériques qui y sont attachés. Pour répertorier tous les périphériques USB du système, veuillez saisir ce qui suit dans une invite de shell :

```
lsusb
```

Cette commande affiche une simple liste de périphériques, comme par exemple :

```
~]$ lsusb
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub[sortie
tronquée]
Bus 001 Device 002: ID 0bda:0151 Realtek Semiconductor Corp. Mass Storage
Device (Multicard Reader)
Bus 008 Device 002: ID 03f0:2c24 Hewlett-Packard Logitech M-UAL-96 Mouse
Bus 008 Device 003: ID 04b3:3025 IBM Corp.
```

Vous pouvez également utiliser l'option de ligne de commande **-v** pour afficher une sortie plus détaillée :

```
lsusb -v
```

Par exemple :

```
~]$ lsusb -v[sortie tronquée]

Bus 008 Device 002: ID 03f0:2c24 Hewlett-Packard Logitech M-UAL-96 Mouse
Device Descriptor:
  bLength                18
  bDescriptorType         1
  bcdUSB               2.00
  bDeviceClass            0 (Defined at Interface level)
  bDeviceSubClass         0
  bDeviceProtocol         0
  bMaxPacketSize0         8
```

```
  idVendor           0x03f0 Hewlett-Packard
  idProduct          0x2c24 Logitech M-UAL-96 Mouse
  bcdDevice           31.00
  iManufacturer           1
  iProduct                2
  iSerial                 0
  bNumConfigurations      1
  Configuration Descriptor:
    bLength                 9
    bDescriptorType         2[sortie tronquée]
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **lsusb**(8).

18.5.3. Utiliser la commande lscpu

La commande **lscpu** permet de répertorier des informations sur les CPU présents sur le système, y compris le nombre de CPU, leur architecture, fournisseur, famille, modèle, cache de CPU, etc. Pour cela, veuillez saisir ce qui suit dans une invite de shell :

```
lscpu
```

Par exemple :

```
~]$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                4
On-line CPU(s) list:   0-3
Thread(s) per core:    1
Core(s) per socket:    4
Socket(s):             1
NUMA node(s):          1
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 23
Stepping:              7
CPU MHz:               1998.000
BogoMIPS:              4999.98
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              3072K
NUMA node0 CPU(s):     0-3
```

Pour obtenir une liste complète des options de ligne de commande disponibles, veuillez consulter la page man de **lscpu**(1).

18.6. VÉRIFICATION DES ERREURS MATÉRIEL

Red Hat Enterprise Linux 7 présente un nouveau mécanisme pour le suivi des événements concernant le matériel ou *Hardware Event Report Mechanism* (HERM). Ce mécanisme collecte les erreurs rapportées par le mécanisme *Error Detection And Correction* (EDAC) pour les modules «dual in-line

memory » (DIMMs) et les rapporte dans l'espace utilisateur. Le démon d'espace-utilisateur **rasdaemon**, collecte et gère tous les événements d'erreurs en matière de *fiabilité, disponibilité et facilité de gestion* qui arrivent dans le mécanisme de suivi du noyau (RAS en anglais pour Reliability, Availability, and Serviceability). Les fonctions qui étaient fournies par **edac-utils** auparavant sont maintenant remplacées par **rasdaemon**.

Pour installer **rasdaemon**, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install rasdaemon
```

Démarrer le service ainsi :

```
~]# systemctl start rasdaemon
```

Saisir la commande suivante pour afficher une liste d'options de commande :

```
~]$ rasdaemon --help
Usage: rasdaemon [OPTION...] <options>
RAS daemon to log the RAS events.

  -d, --disable              disable RAS events and exit
  -e, --enable               enable RAS events and exit
  -f, --foreground           run foreground, not daemonize
  -r, --record               record events via sqlite3sortie tronquée
```

Ces commandes sont également décrites dans la page man de **rasdaemon(8)**.

L'utilitaire **ras-mc-ctl** nous donne un moyen de travailler avec les pilotes EDAC . Saisir la commande suivante pour afficher une liste des options de commande :

```
~]$ ras-mc-ctl --help
Usage: ras-mc-ctl [OPTIONS...]
 --quiet            Quiet operation.
 --mainboard        Print mainboard vendor and model for this hardware.
 --status           Print status of EDAC drivers.sortie tronquée
```

Ces commandes sont également décrites dans la page man **ras-mc-ctl(8)**.

18.7. SURVEILLER LES PERFORMANCES AVEC NET-SNMP

Red Hat Enterprise Linux 7 inclut la suite de logiciels **Net-SNMP**, qui offre un agent flexible et extensible ou *simple network management protocol* (SNMP). Cet agent et ses utilitaires associés peuvent être utilisés pour fournir des données de performance à partir d'un grand nombre de systèmes sur une variété d'outils qui prennent en charge les interrogations via le protocole **SNMP**.

Cette section fournit des informations sur la configuration de l'agent Net-SNMP pour fournir des données de performance sur le réseau de manière sécurisée, sur la récupération de données en utilisant le protocole SNMP, et sur l'extension de l'agent SNMP pour fournir des indicateurs de performance personnalisés.

18.7.1. Installer Net-SNMP

La suite de logiciels Net-SNMP est disponible en tant qu'ensemble de paquets RPM dans la distribution de logiciels Red Hat Enterprise Linux. Tableau 18.2, « Paquets Net-SNMP disponibles » résume chacun des paquets et son contenu.

Tableau 18.2. Paquets Net-SNMP disponibles

Paquet	Fournit
net-snmp	Le démon de l'agent SNMP et la documentation. Ce paquet est requis pour exporter des données de performance.
net-snmp-libs	La bibliothèque **netsnmp** et les *management information bases* (MIB) groupése. Ce paquet est requis pour exporter des données de performance.
net-snmp-utils	Les clients SNMP tels que **snmpget** et **snmpwalk**. Ce paquet est requis pour effectuer des requêtes de données de performance d'un système sur SNMP.
net-snmp-perl	L'utilitaire **mib2c** et le module Perl **NetSNMP**. Remarquez que ce paquet est fournit par le canal optionnel « Optional ». Veuillez consulter la Section 8.5.7, « Ajouter les référentiels « Optional » (Optionnel) et « Supplementary » (Supplémentaire) » pour obtenir davantage d'informations sur les canaux supplémentaires de Red Hat.
net-snmp-python	La bibliothèque cliente SNMP de Python. Remarquez que ce paquet est fourni par le canal « Optional ». Veuillez consulter la Section 8.5.7, « Ajouter les référentiels « Optional » (Optionnel) et « Supplementary » (Supplémentaire) » pour obtenir davantage d'informations sur les canaux supplémentaires Red Hat.

Pour installer ces paquets, utiliser la commande **yum** sous la forme suivante :

```
yum install package…
```

Par exemple, pour installer le démon de l'agent SNMP et les clients SNMP utilisés dans le reste de cette section, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
~]# yum install net-snmp net-snmp-libs net-snmp-utils
```

Pour obtenir davantage d'informations sur la manière d'installer de nouveaux paquets sur Red Hat Enterprise Linux, veuillez consulter la Section 8.2.4, « Installation de paquets ».

18.7.2. Exécuter le démon Net-SNMP

Le paquet net-snmp contient **snmpd**, le démon de l'agent SNMP. Cette section fournit des informations sur la manière de lancer, arrêter, et redémarrer le service **snmpd**. Pour obtenir davantage d'informations sur la gestion des services systèmes dans Red Hat Enterprise Linux 7, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

18.7.2.1. Lancer le service

Pour exécuter le service **snmpd** dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
systemctl start snmpd.service
```

Pour configurer le service afin qu'il soit automatiquement lancé lors du démarrage, veuillez utiliser la commande suivante :

```
systemctl enable snmpd.service
```

18.7.2.2. Arrêter le service

Pour arrêter le service en cours d'exécution **snmpd**, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl stop snmpd.service
```

Pour désactiver le lancement du service lors du démarrage, veuillez utiliser la commande suivante :

```
systemctl disable snmpd.service
```

18.7.2.3. Redémarrer le service

Pour redémarrer le service **snmpd**, saisissez ce qui suit dans l'invite du shell :

```
systemctl restart snmpd.service
```

Cette commande arrête le service et le lance à nouveau en une succession rapide. Pour uniquement recharger la configuration sans arrêter le service, veuillez exécuter la commande suivante à la place :

```
systemctl reload snmpd.service
```

Cela amène le service en cours d'exécution **snmpd** à recharger sa configuration.

18.7.3. Configurer Net-SNMP

Pour changer la configuration du démon de l'agent Net-SNMP, veuillez modifier le fichier de configuration **/etc/snmp/snmpd.conf**. Le fichier par défaut **snmpd.conf** inclus avec Red Hat Enterprise Linux 7 contient beaucoup de commentaires et peut servir de bon point de départ pour la configuration de l'agent.

Cette section traite de deux tâches courantes : la définition des informations système et la configuration de l'authentification. Pour obtenir davantage d'informations sur les directives de configuration disponibles, veuillez consulter la page man de **snmpd.conf**(5). En outre, il existe un utilitaire dans le paquet net-snmp nommé **snmpconf**, qui peut être utilisé de manière interactive pour générer une configuration d'agent valide.

Remarquez que le paquet net-snmp-utils doit être installé pour pouvoir utiliser l'utilitaire **snmpwalk** décrit dans cette section.

NOTE

Pour que tout changement apporté au fichier de configuration puisse entrer en vigueur, veuillez forcer le service **snmpd** à relire la configuration en exécutant la commande suivante en tant qu'utilisateur **root** :

```
systemctl reload snmpd.service
```

18.7.3.1. Définir les informations système

Net-SNMP fournit certaines informations système rudimentaires via l'arborescence **system**. Par exemple, la commande **snmpwalk** suivante montre l'arborescence **system** avec une configuration d'agent par défaut.

```
~]# snmpwalk -v2c -c public localhost system
SNMPv2-MIB::sysDescr.0 = STRING: Linux localhost.localdomain 3.10.0-
123.el7.x86_64 #1 SMP Mon May 5 11:16:57 EDT 2014 x86_64
SNMPv2-MIB::sysObjectID.0 = OID: NET-SNMP-MIB::netSnmpAgentOIDs.10
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (464) 0:00:04.64
SNMPv2-MIB::sysContact.0 = STRING: Root <root@localhost> (configure
/etc/snmp/snmp.local.conf)[sortie tronquée]
```

Par défaut, l'objet **sysName** est défini sur le nom d'hôte. Les objets **sysLocation** et **sysContact** peuvent être configurés dans le fichier **/etc/snmp/snmpd.conf** en modifiant la valeur des directives **syslocation** et **syscontact**. Exemple :

```
syslocation Datacenter, Row 4, Rack 3
syscontact UNIX Admin <admin@example.com>
```

Après avoir apporté des changements au fichier de configuration, rechargez la configuration et testez-la en exécutant la commande **snmpwalk** à nouveau :

```
~]# systemctl reload snmp.service
~]# snmpwalk -v2c -c public localhost system
SNMPv2-MIB::sysDescr.0 = STRING: Linux localhost.localdomain 3.10.0-
123.el7.x86_64 #1 SMP Mon May 5 11:16:57 EDT 2014 x86_64
SNMPv2-MIB::sysObjectID.0 = OID: NET-SNMP-MIB::netSnmpAgentOIDs.10
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (35424) 0:05:54.24
SNMPv2-MIB::sysContact.0 = STRING: UNIX Admin <admin@example.com>
SNMPv2-MIB::sysName.0 = STRING: localhost.localdomain
SNMPv2-MIB::sysLocation.0 = STRING: Datacenter, Row 4, Rack 3[sortie
tronquée]
```

18.7.3.2. Configurer l'authentification

Le démon de l'agent Net-SNMP prend en charge les trois versions du protocole SNMP. Les deux première versions (1 et 2c) fournissent une simple authentification en utilisant une *chaîne de communauté*. Cette chaîne est un secret partagé entre l'agent et tout utilitaire client. Cependant, la chaîne est transmise en texte clair sur le réseau, et n'est donc pas considérée comme sécurisée. La version 3 du protocole SNMP prend en charge l'authentification d'utilisateur et le chiffrement de messages en utilisant tout un ensemble de protocoles. L'agent Net-SNMP prend également en charge la mise sous tunnel avec SSH, l'authentification TLS avec certificats X.509, et l'authentification Kerberos.

Configurer une communauté SNMP Version 2c
Pour configurer une **communauté SNMP version 2c**, veuillez utiliser la directive **rocommunity** ou **rwcommunity** dans le fichier de configuration **/etc/snmp/snmpd.conf**. Le format des directives est comme suit :

```
directive community [source [OID]]
```

… avec *community* correspondant à la chaîne de communauté à utiliser, *source* est une adresse IP ou un sous-réseau, et *OID* est l'arborescence SNMP à laquelle l'accès doit être fourni. Ainsi, la directive suivante fournit un accès en lecture seule à l'arborescence **system** à un client utilisant la chaîne de communité « redhat » sur la machine locale :

```
rocommunity redhat 127.0.0.1 .1.3.6.1.2.1.1
```

Pour tester la configuration, veuillez utiliser la commande **snmpwalk** avec les options **-v** et **-c**.

```
~]# snmpwalk -v2c -c redhat localhost system
SNMPv2-MIB::sysDescr.0 = STRING: Linux localhost.localdomain 3.10.0-
123.el7.x86_64 #1 SMP Mon May 5 11:16:57 EDT 2014 x86_64
SNMPv2-MIB::sysObjectID.0 = OID: NET-SNMP-MIB::netSnmpAgentOIDs.10
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (101376) 0:16:53.76
SNMPv2-MIB::sysContact.0 = STRING: UNIX Admin <admin@example.com>
SNMPv2-MIB::sysName.0 = STRING: localhost.localdomain
SNMPv2-MIB::sysLocation.0 = STRING: Datacenter, Row 4, Rack 3[sortie
tronquée]
```

Configurer un utilisateur SNMP Version 3
Pour configurer un **utilisateur SNMP version 3**, veuillez utiliser la commande **net-snmp-create-v3-user**. Cette command ajoute des entrées aux fichiers **/var/lib/net-snmp/snmpd.conf** et **/etc/snmp/snmpd.conf** qui créent l'utilisateur et offrent accès à l'utilisateur. Remarque que la commande **net-snmp-create-v3-user** peut uniquement être exécutée lorsque l'agent n'est pas en cours d'exécution. L'exemple suivant crée l'utilisateur « admin » avec le mot de passe « redhatsnmp » :

```
~]# systemctl stop snmpd.service
~]# net-snmp-create-v3-user
Enter a SNMPv3 user name to create:
admin
Enter authentication pass-phrase:
redhatsnmp
Enter encryption pass-phrase:
  [press return to reuse the authentication pass-phrase]

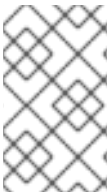
adding the following line to /var/lib/net-snmp/snmpd.conf:
   createUser admin MD5 "redhatsnmp" DES
adding the following line to /etc/snmp/snmpd.conf:
   rwuser admin
~]# systemctl start snmpd.service
```

La directive **rwuser** (ou **rouser** lorsque l'option de ligne de commande **-ro** est fournie) ajoutée par **net-snmp-create-v3-user** à **/etc/snmp/snmpd.conf** possède un format similaire aux directives **rwcommunity** et **rocommunity** :

```
directive user [noauth|auth|priv] [OID]
```

… quand *user* est un nom d'utilisateur et *OID* est l'arborescence SNMP à laquelle l'accès doit être fournit . Par défaut, le démon de l'agent Net-SNMP autorise uniquement les requêtes authentifiées (l'option **auth**). L'option **noauth** vous permet d'autoriser des requêtes non authentifiées, et l'option **priv** applique l'utilisation du chiffrement. L'option **authpriv** spécifie que les requêtes doivent être authentifiées et que les réponses doivent être chiffrées.

Ainsi, la ligne suivante offre à l'utilisateur « admin » un accès en lecture et écriture à la totalité de l'arborescence :

```
rwuser admin authpriv .1
```

Pour tester la configuration, veuillez créer un répertoire **.snmp/** dans le répertoire personnel de l'utilisateur, ainsi qu'un fichier de configuration nommé **snmp.conf** dans ce répertoire (**~/.snmp/snmp.conf**) avec les lignes suivantes :

```
defVersion 3
defSecurityLevel authPriv
defSecurityName admin
defPassphrase redhatsnmp
```

La commande **snmpwalk** utilisera ces paramètres d'authentification lorsque des requêtes sont effectuées sur l'agent :

```
~]$ snmpwalk -v3 localhost system
SNMPv2-MIB::sysDescr.0 = STRING: Linux localhost.localdomain 3.10.0-
123.el7.x86_64 #1 SMP Mon May 5 11:16:57 EDT 2014 x86_64[sortie tronquée]
```

18.7.4. Récupérer des données de performance sur SNMP

L'agent Net-SNMP sur Red Hat Enterprise Linux fournit un large éventail d'informations sur les performances via le protocole SNMP. De plus, des requêtes peuvent être effectués sur l'agent pour obtenir une liste des paquets RPM installés sur le système, une liste des processus actuellement en cours d'exécution sur le système, ou la configuration réseau du système.

Cette section fournit un aperçu des OID liés aux réglages des performances disponibles sur SNMP. Elle suppose que le paquet net-snmp-utils soit installé et que l'utilisateur ait accès à l'arborescence SNMP comme décrit dans la Section 18.7.3.2, « Configurer l'authentification ».

18.7.4.1. Configuration du matériel

Le MIB des ressources d'hôte, « **Host Resources MIB** », inclus avec Net-SNMP présente des informations sur la configuration du matériel et des logiciels d'un hôte à un utilitaire client. Tableau 18.3, « OID disponibles » résume les différents OID disponibles sous ce MIB.

Tableau 18.3. OID disponibles

OID	Description
HOST-RESOURCES-MIB::hrSystem	Contient des informations système générales telles que le temps d'activité, le nombre d'utilisateurs, et le nombre de processus en cours d'utilisation.

OID	Description
HOST-RESOURCES-MIB::hrStorage	Contient des données sur l'utilisation de la mémoire et des systèmes de fichiers.
HOST-RESOURCES-MIB::hrDevices	Contient une liste des processeurs, périphériques réseau, et systèmes de fichiers.
HOST-RESOURCES-MIB::hrSWRun	Contient une liste de tous les processus en cours d'utilisation.
HOST-RESOURCES-MIB::hrSWRunPerf	Contient des statistiques sur la mémoire et le CPU sur la table de processus de HOST-RESOURCES-MIB::hrSWRun.
HOST-RESOURCES-MIB::hrSWInstalled	Contient une liste de la base de données RPM.

Un certain nombre de tables SNMP sont également disponibles dans le MIB des ressources de l'hôte, pouvant être utilisées pour récupérer un résumé des informations disponibles. L'exemple suivant afficher **HOST-RESOURCES-MIB::hrFSTable** :

```
~]$ snmptable -Cb localhost HOST-RESOURCES-MIB::hrFSTable
SNMP table: HOST-RESOURCES-MIB::hrFSTable

 Index MountPoint RemoteMountPoint                                  Type
    Access Bootable StorageIndex LastFullBackupDate LastPartialBackupDate
     1        "/"              "" HOST-RESOURCES-TYPES::hrFSLinuxExt2
 readWrite     true           31      0-1-1,0:0:0.0         0-1-1,0:0:0.0
     5 "/dev/shm"              ""     HOST-RESOURCES-TYPES::hrFSOther
 readWrite    false           35      0-1-1,0:0:0.0         0-1-1,0:0:0.0
     6    "/boot"              "" HOST-RESOURCES-TYPES::hrFSLinuxExt2
 readWrite    false           36      0-1-1,0:0:0.0         0-1-1,0:0:0.0
```

Pour obtenir davantage d'informations sur **HOST-RESOURCES-MIB**, veuillez consulter le fichier **/usr/share/snmp/mibs/HOST-RESOURCES-MIB.txt**.

18.7.4.2. Informations mémoire et CPU

La plupart des données de performances sont disponibles sur le MIB SNMP UCS « **UCD SNMP MIB** ». L'OID **systemStats** fournit un certain nombre de compteurs autour de l'utilisation du processeur :

```
~]$ snmpwalk localhost UCD-SNMP-MIB::systemStats
UCD-SNMP-MIB::ssIndex.0 = INTEGER: 1
UCD-SNMP-MIB::ssErrorName.0 = STRING: systemStats
UCD-SNMP-MIB::ssSwapIn.0 = INTEGER: 0 kB
UCD-SNMP-MIB::ssSwapOut.0 = INTEGER: 0 kB
UCD-SNMP-MIB::ssIOSent.0 = INTEGER: 0 blocks/s
UCD-SNMP-MIB::ssIOReceive.0 = INTEGER: 0 blocks/s
UCD-SNMP-MIB::ssSysInterrupts.0 = INTEGER: 29 interrupts/s
UCD-SNMP-MIB::ssSysContext.0 = INTEGER: 18 switches/s
UCD-SNMP-MIB::ssCpuUser.0 = INTEGER: 0
UCD-SNMP-MIB::ssCpuSystem.0 = INTEGER: 0
```

```
UCD-SNMP-MIB::ssCpuIdle.0 = INTEGER: 99
UCD-SNMP-MIB::ssCpuRawUser.0 = Counter32: 2278
UCD-SNMP-MIB::ssCpuRawNice.0 = Counter32: 1395
UCD-SNMP-MIB::ssCpuRawSystem.0 = Counter32: 6826
UCD-SNMP-MIB::ssCpuRawIdle.0 = Counter32: 3383736
UCD-SNMP-MIB::ssCpuRawWait.0 = Counter32: 7629
UCD-SNMP-MIB::ssCpuRawKernel.0 = Counter32: 0
UCD-SNMP-MIB::ssCpuRawInterrupt.0 = Counter32: 434
UCD-SNMP-MIB::ssIORawSent.0 = Counter32: 266770
UCD-SNMP-MIB::ssIORawReceived.0 = Counter32: 427302
UCD-SNMP-MIB::ssRawInterrupts.0 = Counter32: 743442
UCD-SNMP-MIB::ssRawContexts.0 = Counter32: 718557
UCD-SNMP-MIB::ssCpuRawSoftIRQ.0 = Counter32: 128
UCD-SNMP-MIB::ssRawSwapIn.0 = Counter32: 0
UCD-SNMP-MIB::ssRawSwapOut.0 = Counter32: 0
```

En particulier, les OID **ssCpuRawUser**, **ssCpuRawSystem**, **ssCpuRawWait**, et **ssCpuRawIdle** fournissent des compteurs qui sont utiles pour déterminer si un système passe la plupart de son temps de traitement dans l'espace du noyau, l'espace utilisateur, ou les E/S. **ssRawSwapIn** et **ssRawSwapOut** peuvent être utiles pour déterminer si un système souffre d'épuisement de mémoire.

Davantage d'informations mémoire sont disponibles sous l'OID **UCD-SNMP-MIB::memory**, qui fournit des données similaires à la commande **free** :

```
~]$ snmpwalk localhost UCD-SNMP-MIB::memory
UCD-SNMP-MIB::memIndex.0 = INTEGER: 0
UCD-SNMP-MIB::memErrorName.0 = STRING: swap
UCD-SNMP-MIB::memTotalSwap.0 = INTEGER: 1023992 kB
UCD-SNMP-MIB::memAvailSwap.0 = INTEGER: 1023992 kB
UCD-SNMP-MIB::memTotalReal.0 = INTEGER: 1021588 kB
UCD-SNMP-MIB::memAvailReal.0 = INTEGER: 634260 kB
UCD-SNMP-MIB::memTotalFree.0 = INTEGER: 1658252 kB
UCD-SNMP-MIB::memMinimumSwap.0 = INTEGER: 16000 kB
UCD-SNMP-MIB::memBuffer.0 = INTEGER: 30760 kB
UCD-SNMP-MIB::memCached.0 = INTEGER: 216200 kB
UCD-SNMP-MIB::memSwapError.0 = INTEGER: noError(0)
UCD-SNMP-MIB::memSwapErrorMsg.0 = STRING:
```

Les moyennes des charges sont également disponibles dans le MIB SNMP UCD « **UCD SNMP MIB** ». La table SNMP **UCD-SNMP-MIB::laTable** possède une liste des moyennes de charges de 1, 5, et 15 minutes :

```
~]$ snmptable localhost UCD-SNMP-MIB::laTable
SNMP table: UCD-SNMP-MIB::laTable

 laIndex laNames laLoad laConfig laLoadInt laLoadFloat laErrorFlag
laErrMessage
       1  Load-1   0.00    12.00         0    0.000000     noError
       2  Load-5   0.00    12.00         0    0.000000     noError
       3 Load-15   0.00    12.00         0    0.000000     noError
```

18.7.4.3. Informations sur les systèmes de fichiers et les disques

Le MIB des ressources hôte « **Host Resources MIB** » fournit des informations sur la taille et l'utilisation du système de fichiers. Chaque système de fichiers (ainsi que chaque pool de mémoire) possède une entrée dans la table **HOST-RESOURCES-MIB::hrStorageTable** :

```
~]$ snmptable -Cb localhost HOST-RESOURCES-MIB::hrStorageTable
SNMP table: HOST-RESOURCES-MIB::hrStorageTable

 Index                                       Type           Descr
AllocationUnits    Size   Used AllocationFailures
     1           HOST-RESOURCES-TYPES::hrStorageRam Physical memory
1024 Bytes 1021588 388064                  ?
     3 HOST-RESOURCES-TYPES::hrStorageVirtualMemory  Virtual memory
1024 Bytes 2045580 388064                  ?
     6         HOST-RESOURCES-TYPES::hrStorageOther  Memory buffers
1024 Bytes 1021588  31048                  ?
     7         HOST-RESOURCES-TYPES::hrStorageOther   Cached memory
1024 Bytes  216604 216604                  ?
    10 HOST-RESOURCES-TYPES::hrStorageVirtualMemory      Swap space
1024 Bytes 1023992      0                  ?
    31     HOST-RESOURCES-TYPES::hrStorageFixedDisk              /
4096 Bytes 2277614 250391                  ?
    35     HOST-RESOURCES-TYPES::hrStorageFixedDisk        /dev/shm
4096 Bytes  127698      0                  ?
    36     HOST-RESOURCES-TYPES::hrStorageFixedDisk           /boot
1024 Bytes  198337  26694                  ?
```

Les OID sous **HOST-RESOURCES-MIB::hrStorageSize** et **HOST-RESOURCES-MIB::hrStorageUsed** peuvent être utilisés pour calculer la capacité restante de chaque système de fichiers monté.

Des données d'E/S sont disponibles sur **UCD-SNMP-MIB::systemStats** (**ssIORawSent.0** et **ssIORawRecieved.0**) et sur **UCD-DISKIO-MIB::diskIOTable**. Ce dernier fournit des données plus granulaires. Sous cette table, se trouvent des OID pour **diskIONReadX** et **diskIONWrittenX**, qui fournissent des compteurs pour le nombre d'octets lus et écrits sur le périphérique bloc en question depuis le démarrage système :

```
~]$ snmptable -Cb localhost UCD-DISKIO-MIB::diskIOTable
SNMP table: UCD-DISKIO-MIB::diskIOTable

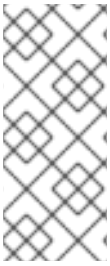
 Index Device     NRead  NWritten Reads Writes LA1 LA5 LA15    NReadX
NWrittenX
...
    25    sda 216886272 139109376 16409   4894   ?   ?    ? 216886272
139109376
    26   sda1   2455552      5120   613      2   ?   ?    ?   2455552
5120
    27   sda2   1486848         0   332      0   ?   ?    ?   1486848
0
    28   sda3 212321280 139104256 15312   4871   ?   ?    ? 212321280
139104256
```

18.7.4.4. Informations réseau

Le MIB des interfaces, « **Interfaces MIB** », fournit des informations sur les périphériques réseau.

IF-MIB::ifTable fournit une table SNMP avec une entrée pour chaque interface sur le système, la configuration de l'interface, et divers compteurs de paquets pour l'interface. L'exemple suivant affiche les premières colonnes d'**ifTable** sur un système avec deux interfaces réseau physiques :

```
~]$ snmptable -Cb localhost IF-MIB::ifTable
SNMP table: IF-MIB::ifTable

 Index Descr             Type   Mtu    Speed      PhysAddress AdminStatus
     1    lo softwareLoopback 16436 10000000
up
     2  eth0   ethernetCsmacd  1500        0 52:54:0:c7:69:58          up
     3  eth1   ethernetCsmacd  1500        0 52:54:0:a7:a3:24        down
```

Le trafic réseau est disponibles sous les OID **IF-MIB::ifOutOctets** et **IF-MIB::ifInOctets**. Les requêtes SNMP suivantes récupéreront le trafic réseau pour chacune des interfaces sur ce système :

```
~]$ snmpwalk localhost IF-MIB::ifDescr
IF-MIB::ifDescr.1 = STRING: lo
IF-MIB::ifDescr.2 = STRING: eth0
IF-MIB::ifDescr.3 = STRING: eth1
~]$ snmpwalk localhost IF-MIB::ifOutOctets
IF-MIB::ifOutOctets.1 = Counter32: 10060699
IF-MIB::ifOutOctets.2 = Counter32: 650
IF-MIB::ifOutOctets.3 = Counter32: 0
~]$ snmpwalk localhost IF-MIB::ifInOctets
IF-MIB::ifInOctets.1 = Counter32: 10060699
IF-MIB::ifInOctets.2 = Counter32: 78650
IF-MIB::ifInOctets.3 = Counter32: 0
```

18.7.5. Étendre Net-SNMP

L'agent Net-SNMP peut être étendu pour fournir des indicateurs d'applications en plus des indicateurs de systèmes bruts. Cela permet la planification des capacités, ainsi que la résolution des problèmes de performance. Ainsi, il peut être utile de savoir qu'un système de courrier électronique possède une charge moyenne de 5 minutes de 15 lorsque testé, mais il est encore plus utile de savoir que le système de courrier électronique possède une charge moyenne de 15 lors du traitement de 80 000 messages par seconde. Lorsque les indicateurs d'applications sont disponibles via la même interface que les indicateurs du système, cela permet également la visualisation de l'impact des différent scénarios de charge sur les performances système (par exemple, la charge moyenne augmentera de manière linéaire, chaque fois qu'il y a 10 000 messages supplémentaires, jusqu'à ce que l'on atteigne 100,000).

Un certain nombre d'applications incluses dans Red Hat Enterprise Linux étendent l'agent Net-SNMP pour fournir des indicateurs d'applications sur SNMP. Il existe également plusieurs manières d'étendre l'agent pour des applications personnalisées. Cette section décrit l'extension de l'agent avec des scripts shell et des greffons Perl à partir du canal « Optional ». Elle suppose que les paquets net-snmp-utils et net-snmp-perl soient installés et que l'utilisateur ait accès à l'arborescence SNMP, comme décrit dans la Section 18.7.3.2, « Configurer l'authentification ».

18.7.5.1. Étendre Net-SNMP avec des scripts Shell

L'agent Net-SNMP fournit une extension MIB (**NET-SNMP-EXTEND-MIB**) qui peut être utilisée pour effectuer des requêtes de scripts shell arbitraires. Pour indiquer quel script shell exécuter, veuillez utiliser la directive **extend** dans le fichier **/etc/snmp/snmpd.conf**. Une fois défini, l'agent fournira le

code de sortie et toute sortie de la commande sur SNMP. L'exemple ci-dessous fait une démonstration de ce mécanisme avec un script qui détermine le nombre de processus **httpd** dans la table des processus.



NOTE

L'agent Net-SNMP fournit également un mécasnisme intégré de vérification de la table des processus via la directive **proc**. Veuillez consulter la page man de **snmpd.conf**(5) pour obtenir davantage d'informations.

Le code de sortie du script shell suivant est le nombre de processus **httpd** exécutés sur le système à un moment donné :

```
#!/bin/sh

NUMPIDS=`pgrep httpd | wc -l`

exit $NUMPIDS
```

Pour que ce script soit disponible sur SNMP, copiez le script sur un emplacement sur le chemin système, définissez le bit exécutable, et ajoutez une directive **extend** au fichier **/etc/snmp/snmpd.conf**. Le format de la directive **extend** est comme suit :

```
extend name prog args
```

… où *name* est une chaîne d'identification pour l'extension, *prog* est le programme à exécuter, et *args* sont les arguments à donner au programme. Par exemple, si le script shell ci-dessus est copié sur **/usr/local/bin/check_apache.sh**, la directive suivante ajoutera le script à l'arborescence SNMP :

```
extend httpd_pids /bin/sh /usr/local/bin/check_apache.sh
```

Des requêtes peuvent ensuite être effectuées sur le script sur **NET-SNMP-EXTEND-MIB::nsExtendObjects** :

```
~]$ snmpwalk localhost NET-SNMP-EXTEND-MIB::nsExtendObjects
NET-SNMP-EXTEND-MIB::nsExtendNumEntries.0 = INTEGER: 1
NET-SNMP-EXTEND-MIB::nsExtendCommand."httpd_pids" = STRING: /bin/sh
NET-SNMP-EXTEND-MIB::nsExtendArgs."httpd_pids" = STRING:
/usr/local/bin/check_apache.sh
NET-SNMP-EXTEND-MIB::nsExtendInput."httpd_pids" = STRING:
NET-SNMP-EXTEND-MIB::nsExtendCacheTime."httpd_pids" = INTEGER: 5
NET-SNMP-EXTEND-MIB::nsExtendExecType."httpd_pids" = INTEGER: exec(1)
NET-SNMP-EXTEND-MIB::nsExtendRunType."httpd_pids" = INTEGER: run-on-
read(1)
NET-SNMP-EXTEND-MIB::nsExtendStorage."httpd_pids" = INTEGER: permanent(4)
NET-SNMP-EXTEND-MIB::nsExtendStatus."httpd_pids" = INTEGER: active(1)
NET-SNMP-EXTEND-MIB::nsExtendOutput1Line."httpd_pids" = STRING:
NET-SNMP-EXTEND-MIB::nsExtendOutputFull."httpd_pids" = STRING:
NET-SNMP-EXTEND-MIB::nsExtendOutNumLines."httpd_pids" = INTEGER: 1
NET-SNMP-EXTEND-MIB::nsExtendResult."httpd_pids" = INTEGER: 8
NET-SNMP-EXTEND-MIB::nsExtendOutLine."httpd_pids".1 = STRING:
```

Remarquez que le code de sortie (« 8 » dans cet exemple) fourni est de type INTEGER (entier) et toute

sortie fournie es de type STRING (chaîne). Pour exposer de multiples indicateurs en tant qu'entiers, veuillez fournir différents arguments au script en utilisant la directive **extend**. Par exemple, le script shell suivant peut être utilisé pour déterminer le nombre de processus correspondants à une chaîne arbitraire, et fera également sortir une chaîne de texte donnant le nombre de processus :

```
#!/bin/sh

PATTERN=$1
NUMPIDS=`pgrep $PATTERN | wc -l`

echo "There are $NUMPIDS $PATTERN processes."
exit $NUMPIDS
```

Les directives suivantes **/etc/snmp/snmpd.conf** donneront le nombre de PID **httpd** ainsi que le nombre de PID **snmpd** lorsque le script ci-dessus est copié sur **/usr/local/bin/check_proc.sh** :

```
extend httpd_pids /bin/sh /usr/local/bin/check_proc.sh httpd
extend snmpd_pids /bin/sh /usr/local/bin/check_proc.sh snmpd
```

L'exemple suivant affiche la sortie **snmpwalk** de l'OID **nsExtendObjects** :

```
~]$ snmpwalk localhost NET-SNMP-EXTEND-MIB::nsExtendObjects
NET-SNMP-EXTEND-MIB::nsExtendNumEntries.0 = INTEGER: 2
NET-SNMP-EXTEND-MIB::nsExtendCommand."httpd_pids" = STRING: /bin/sh
NET-SNMP-EXTEND-MIB::nsExtendCommand."snmpd_pids" = STRING: /bin/sh
NET-SNMP-EXTEND-MIB::nsExtendArgs."httpd_pids" = STRING:
/usr/local/bin/check_proc.sh httpd
NET-SNMP-EXTEND-MIB::nsExtendArgs."snmpd_pids" = STRING:
/usr/local/bin/check_proc.sh snmpd
NET-SNMP-EXTEND-MIB::nsExtendInput."httpd_pids" = STRING:
NET-SNMP-EXTEND-MIB::nsExtendInput."snmpd_pids" = STRING:
...
NET-SNMP-EXTEND-MIB::nsExtendResult."httpd_pids" = INTEGER: 8
NET-SNMP-EXTEND-MIB::nsExtendResult."snmpd_pids" = INTEGER: 1
NET-SNMP-EXTEND-MIB::nsExtendOutLine."httpd_pids".1 = STRING: There are 8
httpd processes.
NET-SNMP-EXTEND-MIB::nsExtendOutLine."snmpd_pids".1 = STRING: There are 1
snmpd processes.
```

AVERTISSEMENT

L'éventail des codes de sortie qui sont des entiers va de 0 à 255. Pour les valeurs qui dépasseront probablement 256, veuillez utiliser la sortie standard du script (qui sera saisie en tant que chaîne) ou une méthode différente d'étendre l'agent.

Ce dernier exemple montre une requête de mémoire libre du système et le nombre de processus **httpd**. Cette requête pourrait être utilisée pendant un test de performance pour déterminer l'impact du nombre de processus sur la pression mémoire :

```
~]$ snmpget localhost \
    'NET-SNMP-EXTEND-MIB::nsExtendResult."httpd_pids"' \
    UCD-SNMP-MIB::memAvailReal.0
NET-SNMP-EXTEND-MIB::nsExtendResult."httpd_pids" = INTEGER: 8
UCD-SNMP-MIB::memAvailReal.0 = INTEGER: 799664 kB
```

18.7.5.2. Extension de Net-SNMP avec Perl

L'exécution de scripts shell en utilisant la directive **extend** est une méthode assez limitée pour exposer des indicateurs d'application personnalisée sur SNMP. L'agent Net-SNMP fournit également une interface Perl intégrée pour exposer des objets personnalisés. Le paquet net-snmp-perl dans le canal « Optional » fournit le module Perl **NetSNMP::agent**, qui est utilisé pour écrire des greffons Perl intégrés dans Red Hat Enterprise Linux.



NOTE

Avant de vous abonner aux canaux « Optional » et « Supplementary », veuillez consulter les Détails de l'étendue de la couverture. Si vous décidez d'installer des paquets à partir de ces canaux, veuillez suivre les étapes documentées dans l'article nommé Comment accéder aux canaux « Optional » et « Supplementary » et aux paquets -devel en utilisant Red Hat Subscription Manager (RHSM) ? sur le Portail Client Red Hat.

Le module Perl **NetSNMP::agent** fournit un objet **agent** utilisé pour gérer les requêtes d'une partie de l'arborescence OID de l'agent. Le constructeur de l'objet **agent** offre des options pour exécuter l'agent en tant que sous-agent de **snmpd** ou en tant qu'agent autonome. Aucun argument n'est nécessaire pour créer un agent intégré :

```
use NetSNMP::agent (':all');

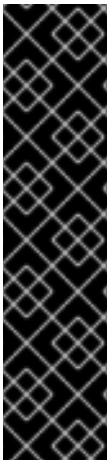
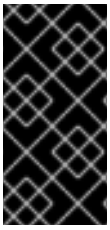
my $agent = new NetSNMP::agent();
```

L'objet **agent** offre une méthode **register** utilisée pour enregistrer une fonction de rappel avec un OID particulier. La fonction **register** prend un nom, un OID, et un pointeur sur la fonction de rappel. L'exemple suivant enregistrera une fonction de rappel nommée **hello_handler** avec l'agent SNM, qui gérera les requêtes sous l'OID **.1.3.6.1.4.1.8072.9999.9999** :

```
$agent->register("hello_world", ".1.3.6.1.4.1.8072.9999.9999",
                 \&hello_handler);
```

NOTE

En général, l'OID **.1.3.6.1.4.1.8072.9999.9999** (**NET-SNMP-MIB::netSnmpPlaypen**) est utilisé pour effectuer des démonstrations uniquement. Si votre organisation ne possède pas déjà un OID root, vous pouvez en obtenir un en contactant une autorité d'enregistrement de nom ISO (appelé ANSI aux États-Unis).

La fonction de gestionnaire sera appelée avec quatre paramètres, ***HANDLER***, ***REGISTRATION_INFO***, ***REQUEST_INFO***, et ***REQUESTS***. Le paramètre ***REQUESTS*** contient une liste de requêtes dans l'appel actuel et devrait être itéré et rempli avec des données. Les objets **request** dans la liste possèdent des méthodes get et set qui permettent de manipuler l' OID et la valeur value de la requête. Par exemple, l'appel suivant définira la valeur d'un objet de requête sur la chaîne « hello world » :

```
$request->setValue(ASN_OCTET_STR, "hello world");
```

La fonction de gestionnaire devrait répondre à deux types de requêtes SNMP : la requête GET et la requête GETNEXT. Le type de requête est déterminé en appelant la méthode **getMode** sur l'objet **request_info** passé en tant que troisième paramètre à la fonction de gestionnaire. Si la requête est une requête GET, l'appelant s'attendra à ce que le gestionnaire définisse la valeur value de l'objet **request**, selon l'OID de la requête. Si la requête est une requête GETNEXT request, l'appelant devra également s'attendre à ce que le gestionnaire définisse l'OID de la requête sur le prochain OID disponible dans l'arborescence. Ceci est illustré dans l'exemple de code suivant :

```
my $request;
my $string_value = "hello world";
my $integer_value = "8675309";

for($request = $requests; $request; $request = $request->next()) {
  my $oid = $request->getOID();
  if ($request_info->getMode() == MODE_GET) {
    if ($oid == new NetSNMP::OID(".1.3.6.1.4.1.8072.9999.9999.1.0")) {
      $request->setValue(ASN_OCTET_STR, $string_value);
    }
    elsif ($oid == new NetSNMP::OID(".1.3.6.1.4.1.8072.9999.9999.1.1")) {
      $request->setValue(ASN_INTEGER, $integer_value);
    }
  } elsif ($request_info->getMode() == MODE_GETNEXT) {
    if ($oid == new NetSNMP::OID(".1.3.6.1.4.1.8072.9999.9999.1.0")) {
      $request->setOID(".1.3.6.1.4.1.8072.9999.9999.1.1");
      $request->setValue(ASN_INTEGER, $integer_value);
    }
    elsif ($oid < new NetSNMP::OID(".1.3.6.1.4.1.8072.9999.9999.1.0")) {
      $request->setOID(".1.3.6.1.4.1.8072.9999.9999.1.0");
      $request->setValue(ASN_OCTET_STR, $string_value);
    }
  }
}
```

Lorsque **getMode** retourne **MODE_GET**, le gestionnaire analyse la valeur de l'appel **getOID** sur l'objet **request**. La valeur value de **request** est définie sur **string_value** si l'OID se termine par « .1.0 », ou sur **integer_value** si l'OID se termine par « .1.1 ». Si **getMode** retourne **MODE_GETNEXT**, le gestionnaire détermine si l'OID de la requête est « .1.0 », puis définit l'OID et la valeur pour « .1.1 ». Si la requête est plus élevée que « .1.0 » sur l'arborescence, l'OID et la valeur de « .1.0 » est définie. Ceci retourne la valeur « next » dans l'arborescence afin qu'un programme comme **snmpwalk** puisse traverser l'arborescence sans connaître la structure au préalable.

Le type de variable est défini en utilisant des constantes de **NetSNMP::ASN**. Veuillez consulter le **perldoc** de **NetSNMP::ASN** pour une liste complète des constantes disponibles.

La liste du code entier de cet exemple de greffon Perl est comme suit :

```
#!/usr/bin/perl

use NetSNMP::agent (':all');
use NetSNMP::ASN qw(ASN_OCTET_STR ASN_INTEGER);

sub hello_handler {
  my ($handler, $registration_info, $request_info, $requests) = @_;
```

```
  my $request;
  my $string_value = "hello world";
  my $integer_value = "8675309";

  for($request = $requests; $request; $request = $request->next()) {
    my $oid = $request->getOID();
    if ($request_info->getMode() == MODE_GET) {
      if ($oid == new NetSNMP::OID(".1.3.6.1.4.1.8072.9999.9999.1.0")) {
        $request->setValue(ASN_OCTET_STR, $string_value);
      }
      elsif ($oid == new NetSNMP::OID(".1.3.6.1.4.1.8072.9999.9999.1.1"))
{
        $request->setValue(ASN_INTEGER, $integer_value);
      }
    } elsif ($request_info->getMode() == MODE_GETNEXT) {
      if ($oid == new NetSNMP::OID(".1.3.6.1.4.1.8072.9999.9999.1.0")) {
        $request->setOID(".1.3.6.1.4.1.8072.9999.9999.1.1");
        $request->setValue(ASN_INTEGER, $integer_value);
      }
      elsif ($oid < new NetSNMP::OID(".1.3.6.1.4.1.8072.9999.9999.1.0")) {
        $request->setOID(".1.3.6.1.4.1.8072.9999.9999.1.0");
        $request->setValue(ASN_OCTET_STR, $string_value);
      }
    }
  }
}

my $agent = new NetSNMP::agent();
$agent->register("hello_world", ".1.3.6.1.4.1.8072.9999.9999",
                 \&hello_handler);
```

Pour tester le greffon, copiez le programme ci-dessus sur **/usr/share/snmp/hello_world.pl** et ajoutez la ligne suivante au fichier de configuration **/etc/snmp/snmpd.conf** :

```
perl do "/usr/share/snmp/hello_world.pl"
```

Le démon de l'agent SNMP devra être redémarré pour charger le nouveau greffon Perl. Une fois redémarré, **snmpwalk** devrait retourner les nouvelles données :

```
~]$ snmpwalk localhost NET-SNMP-MIB::netSnmpPlaypen
NET-SNMP-MIB::netSnmpPlaypen.1.0 = STRING: "hello world"
NET-SNMP-MIB::netSnmpPlaypen.1.1 = INTEGER: 8675309
```

snmpget doit également être utilisé pour exercer l'autre mode du gestionnaire :

```
~]$ snmpget localhost \
    NET-SNMP-MIB::netSnmpPlaypen.1.0 \
    NET-SNMP-MIB::netSnmpPlaypen.1.1
NET-SNMP-MIB::netSnmpPlaypen.1.0 = STRING: "hello world"
NET-SNMP-MIB::netSnmpPlaypen.1.1 = INTEGER: 8675309
```

18.8. RESSOURCES SUPPLÉMENTAIRES

Pour en savoir plus sur la collecte d'informations système, veuillez consulter les ressources suivantes.

18.8.1. Documentation installée

- **lscpu**(1) — page du manuel pour la commande `lscpu`.
- **lsusb**(8) — page du manuel pour la commande `lsusb`.
- **findmnt**(8) — page du manuel pour la commande `findmnt`.
- **blkid**(8) — page du manuel pour la commande `blkid`.
- **lsblk**(8) — page du manuel pour la commande `lsblk`.
- **ps**(1) — page du manuel pour la commande `ps`.
- **top**(1) — page du manuel pour la commande `top`.
- **free**(1) — page du manuel pour la commande `free`.
- **df**(1) — page du manuel pour la commande `df`.
- **du**(1) — page du manuel pour la commande `du`.
- **lspci**(8) — page du manuel pour la commande `lspci`.
- **snmpd**(8) — page du manuel du service `snmpd`.
- **snmpd.conf**(5) — page du manuel du fichier `/etc/snmp/snmpd.conf` contenant la documentation complète des directives de configuration disponibles.

CHAPITRE 19. OPENLMI

L'infrastructure de gestion **Open Linux Management Infrastructure**, communément appelée **OpenLMI**, est une infrastructure commune de gestion de systèmes Linux. Elle est créée au-dessus des outils existants et sert de couche d'abstraction afin de cacher aux administrateurs système la complexité du système sous-jacent. OpenLMI est distribué avec un ensemble de services auxquels on peut accéder localement ou à distance et fournissent de multiples liaisons de langage, des API standard, et des interfaces de script standard pouvant être utilisées pour gérer et surveiller le matériel, les systèmes d'exploitation, et les services système.

19.1. OPENLMI

OpenLMI est conçu pour fournir une interface de gestion commune aux serveurs de production exécutant le système Red Hat Enterprise Linux sur les machines physiques et virtuelles. OpenLMI est constitué de trois composants :

1. *Agents de gestion de système* — ces agents sont installés sur un système géré et implémentent un modèle d'objet qui est présenté à un courtier entre objets standard. Les agents initiaux implémentés dans OpenLMI incluent la configuration du stockage et la configuration du réseau ; un travail ultérieur offrira des éléments supplémentaires pour la gestion de systèmes. Les agents de gestion de systèmes sont couramment appelés des *fournisseurs CIM* (*Common Information Model*).

2. *Courtier d'objets standard* — le courtier d'objets gère les agents de gestion et leur fournit une interface. Le courtier d'objets standard est aussi appelé un *CIMOM* (de l'anglais, *CIM Object Monitor*).

3. *Applications et scripts clients* — les applications et scripts clients appellent les agents de gestion du système via le courtier d'objets standard.

Le projet OpenLMI complémente les initiatives de gestion existantes en fournissant une interface de bas niveau qui peut être utilisée par des scripts ou des consoles de gestion de système. Les interfaces distribuées avec OpenLMI incluent C, C++, Python, Java, un client de ligne de commande interactif, et tous offrent le même accès complet aux capacités implémentées sur chaque agent. Ceci permet de vous assurer que vous aurez toujours accès aux mêmes capacités quelle que soit l'interface de programmation que vous décidez d'utiliser.

19.1.1. Fonctionnalités principales

Ci-dessous figurent les avantages principaux résultant de l'installation et de l'utilisation d'OpenLMI sur votre système :

- OpenLMI fournit une interface standard pour la configuration, la gestion, et la surveillance de vos systèmes locaux et distants.

- OpenLMI vous permet de configurer, gérer, et surveiller des serveurs de production exécutés sur des machines physiques et virtuelles.

- OpenLMI est distribué avec un ensemble de fournisseurs CIM qui vous permettent de configurer, gérer, et surveiller des périphériques de stockage et des réseaux complexes.

- OpenLMI vous permet d'appeler des fonctions de gestion de systèmes à partir de programmes C, C++, Python, et Java, et inclut LMIShell, qui fournit une interface en ligne de commande.

- OpenLMI est un logiciel gratuit basé sur les standards du secteur du logiciel libre.

19.1.2. Capacités de gestion

Les capacités principales d'OpenLMI incluent la gestion des périphériques de stockage, des réseaux, des services systèmes, des comptes utilisateurs, de la configuration matérielle et logicielle, de la gestion de l'alimentation, et des interactions avec Active Directory. Pour afficher une liste complète des fournisseurs CIM distribués avec Red Hat Enterprise Linux 7, veuillez consulter la Tableau 19.1, « Fournisseurs CIM disponibles ».

Tableau 19.1. Fournisseurs CIM disponibles

Nom du paquet	Description
openlmi-account	Un fournisseur CIM pour la gestion des comptes utilisateurs.
openlmi-logicalfile	Un fournisseur CIM pour la lecture des fichiers et répertoires.
openlmi-networking	Un fournisseur CIM pour la gestion des réseaux.
openlmi-powermanagement	Un fournisseur CIM pour la gestion de l'alimentation.
openlmi-service	Un fournisseur CIM pour la gestion des systèmes service.
openlmi-storage	Un fournisseur CIM pour la gestion du stockage.
openlmi-fan	Un fournisseur CIM pour contrôler les ventilateurs de l'ordinateur.
openlmi-hardware	Un fournisseur CIM pour récupérer les informations du matériel.
openlmi-realmd	Un fournisseur CIM pour configurer realmd.
openlmi-software[a]	Un fournisseur CIM pour la gestion de logiciels.

[a] In Red Hat Enterprise Linux 7, the OpenLMI Software provider is included as a Technology Preview. This provider is fully functional, but has a known performance scaling issue where listing large numbers of software packages may consume excessive amount of memory and time. To work around this issue, adjust package searches to return as few packages as possible.

19.2. INSTALLER OPENLMI

OpenLMI est distribué sous forme d'un ensemble de paquets RPM incluant le CIMOM, les fournisseurs CIM individuels, et les applications clientes. Ceci permet de distinguer un système client d'un système géré et d'installer les composants nécessaires uniquement.

19.2.1. Installer OpenLMI sur un système géré

Un *système géré* est un système que vous comptez surveiller et gérer en utilisant des outils client OpenLMI. Pour installer OpenLMI sur un système géré, veuillez observer les étapes suivantes :

1. Veuillez installer le paquet tog-pegasus en saisissant ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
yum install tog-pegasus
```

Cette commande installe le CIMOM OpenPegasus et toutes ses dépendances sur le système et crée un compte utilisateur pour l'utilisateur **pegasus**.

2. Veuillez installer les fournisseurs CIM requis en exécutant la commande suivante en tant qu'utilisateur **root** :

```
yum install openlmi-
{storage,networking,service,account,powermanagement}
```

Cette commande installe les fournisseurs CIM pour le stockage, le réseau, le service, le compte et la gestion de l'alimentation. Pour une liste complète des fournisseurs CIM distribués avec Red Hat Enterprise Linux 7, veuillez consulter la Tableau 19.1, « Fournisseurs CIM disponibles ».

3. Modifiez le fichier de configuration **/etc/Pegasus/access.conf** pour personnaliser la liste des utilisateurs autorisés à se connecter au CIMOM OpenPegasus. Par défaut, seul l'utilisateur **pegasus** est autorisé à accéder au CIMOM à distance et localement. Pour activer ce compte utilisateur, veuillez exécuter la commande suivante en tant qu'utilisateur **root** pour définir le mot de passe de l'utilisateur :

```
passwd pegasus
```

4. Lancez le CIMOM OpenPegasus en activant l'unité **tog-pegasus.service**. Pour activer l'unité **tog-pegasus.service** dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
systemctl start tog-pegasus.service
```

Pour configurer l'unité **tog-pegasus.service** pour qu'elle soit lancée automatiquement pendant le démarrage, veuillez saisir ceci en tant qu'utilisateur **root** :

```
systemctl enable tog-pegasus.service
```

5. Si vous souhaitez interagir avec le système géré à partir d'une machine distante, veuillez activer les communications TCP sur le port **5989** (**wbem-https**). Pour ouvrir ce port dans la session actuelle, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
firewall-cmd --add-port 5989/tcp
```

Pour ouvrir le port **5989** pour les communications TCP de manière permanente, veuillez saisir ceci en tant qu'utilisateur **root** :

```
firewall-cmd --permanent --add-port 5989/tcp
```

Vous pouvez désormais connecter le système géré et interagir avec en utilisant les outils client OpenLMI comme décrit dans la Section 19.4, « Utiliser LMIShell ». Si vous souhaitez effectuer des opérations OpenLMI directement sur le système géré, veuillez également procéder en suivant les étapes décrites dans la Section 19.2.2, « Installer OpenLMI sur un système client ».

19.2.2. Installer OpenLMI sur un système client

Un *système client* est un système à partir duquel vous comptez interagir avec le système géré. Dans un scénario typique, le système client et le système géré sont installés sur deux machines différentes, mais vous pouvez également installer les outils clients sur le système géré et interagir directement avec celui-ci.

Pour installer OpenLMI sur un système client, veuillez effectuer les étapes suivantes :

1. Veuillez installer le paquet openlmi-tools en saisissant ce qui suit à l'invite de shell en tant qu'utilisateur **root** :

   ```
   yum install openlmi-tools
   ```

   Cette commande installe LMIShell, un client et interprète interactif pour accéder aux objets CIM fournis par OpenPegasus, et toutes ses dépendances au système.

2. Veuillez configurer les certificats SSL pour OpenPegasus comme décrit dans la Section 19.3, « Configurer des certificats SSL pour OpenPegasus ».

Vous pouvez désormais utiliser le client LMIShell pour interagir avec le système géré comme décrit dans la Section 19.4, « Utiliser LMIShell ».

19.3. CONFIGURER DES CERTIFICATS SSL POUR OPENPEGASUS

OpenLMI utilise le protocole de gestion d'entreprise basée web WBEM (« Web-Based Enterprise Management ») qui fonctionne sur une couche de transport HTTP. Une authentification de base HTTP standard est effectuée dans ce protocole, ce qui signifie que le nom d'utilisateur et le mot de passe sont transmis avec les requêtes.

Configurer le CIMOM OpenPegasus à utiliser HTTPS pour les communications est nécessaire pour vous assurer une authentification sécurisée. Un certificat « Secure Sockets Layer » (SSL) ou « Transport Layer Security » (TLS) est requis sur le système géré pour établir un canal chiffré.

Il existe deux manières de gérer les certificats SSL/TLS sur un système :

- Les certificats auto-signés requièrent une moindre utilisation de l'infrastructure, mais sont plus difficiles à déployer sur des clients et à gérer de manière sécurisée.
- Les certificats signés par une autorité sont plus faciles à déployer sur des clients une fois qu'ils sont paramétrés, mais ils nécessitent un investissement initial plus élevé.

Lors de l'utilisation d'un certificat signé par autorité, il est nécessaire de configurer une autorité de certificats de confiance sur les systèmes clients. Cette autorité peut ensuite être utilisée pour signer tous les certificats CIMOM des systèmes gérés. Les certificats peuvent également faire partie d'une chaîne de certificats, de manière à ce que le certificat utilisé pour signer les certificats des systèmes gérés puissent ensuite être signés par un autre avec une autorité plus élevée (comme Verisign, CAcert, RSA et bien d'autres).

Les emplacements par défaut des certificats et du trust store sur le système de fichiers sont indiqués dans Tableau 19.2, « Emplacements des certificats et du trust store ».

Tableau 19.2. Emplacements des certificats et du trust store

Option de configuration	Emplacement	Description
sslCertificateFilePath	**/etc/Pegasus/server.pem**	Certificat public du CIMOM.
sslKeyFilePath	**/etc/Pegasus/file.pem**	Clé privée uniquement connue par le CIMOM.
sslTrustStore	**/etc/Pegasus/client.pem**	Fichier ou répertoire fournissant la liste des autorités de certificats de confiance.



IMPORTANT

Si vous modifiez l'un des fichiers mentionnés dans la Tableau 19.2, « Emplacements des certificats et du trust store », redémarrez le service **tog-pegasus** pour vous assurer qu'il reconnaisse les nouveaux certificats. Pour redémarrer le service, veuillez saisir ce qui suit à l'invite de shell en tant qu'utilisateur **root** :

```
systemctl restart tog-pegasus.service
```

Pour obtenir davantage d'informations sur la manière de gérer les services système sur Red Hat Enterprise Linux 7, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

19.3.1. Gérer les certificats auto-signés

Un certificat auto-signé utilise sa propre clé privée pour se signer lui-même et n'est connecté à aucune chaîne de confiance. Sur un système géré, si les certificats n'ont pas été fournis par l'administrateur avant que le service **tog-pegasus** ait été lancé une première fois, un ensemble de certificats auto-signés seront automatiquement générés en utilisant le nom d'hôte principal du système comme sujet du certificat.



IMPORTANT

Les certificats auto-signés générés automatiquement sont valides par défaut pendant 10 ans, mais ils n'ont pas la capacité de se renouveller automatiquement. Toute modification apportée à ces certificats nécessitera de créer des nouveaux certificats manuellement en suivant les directives fournies par la documentation OpenSSL ou Mozilla NSS.

Pour configurer les systèmes client pour qu'ils fassent confiance au certificat auto-signé, veuillez effectuer les étapes suivantes :

1. Veuillez copier le certificat **/etc/Pegasus/server.pem** du système géré au répertoire **/etc/pki/ca-trust/source/anchors/** sur le système client. Pour cela, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

   ```
   scp root@hostname:/etc/Pegasus/server.pem /etc/pki/ca-trust/source/anchors/pegasus-hostname.pem
   ```

   Veuillez remplacer *hostname* par le nom d'hôte du système géré. Remarquez que cette commande fonctionne uniquement si le service **sshd** est en cours d'exécution sur le système

géré et est configuré pour autoriser l'utilisateur **root** à se connecter au système via le protocole SSH. Pour obtenir des informations supplémentaires sur la manière d'installer et de configurer le service **sshd**, ainsi que sur la commande **scp** pour transférer des fichiers sur le protocole SSH, veuillez consulter le Chapitre 10, *OpenSSH*.

2. Veuillez vérifier l'intégrité du certificat sur le système client en comparant son checksum avec celui du fichier d'origine. Pour calculer le checksum du fichier **/etc/Pegasus/server.pem** sur le système géré, veuillez exécuter la commande suivante en tant qu'utilisateur **root** sur ce système :

```
sha1sum /etc/Pegasus/server.pem
```

Pour calculer le checksum du fichier **/etc/pki/ca-trust/source/anchors/pegasus-*hostname*.pem** sur le système client, veuillez exécuter la commande suivante sur ce système :

```
sha1sum /etc/pki/ca-trust/source/anchors/pegasus-hostname.pem
```

Remplacez *hostname* par le nom d'hôte du système géré.

3. Mettez à jour le trust store sur le système client en exécutant la commande suivante en tant qu'utilisateur **root** :

```
update-ca-trust extract
```

19.3.2. Gestion de certificats signés par des autorités avec Identity Management (recommandé)

La fonctionnalité Identity Management sur Red Hat Enterprise Linux fournit un contrôleur de domaines qui simplifie la gestion des certificats SSL dans les systèmes joints au domaine. Entre autres, le serveur Identity Management offre une autorité de certificat (CA) intégrée. Veuillez consulter le Guide des politiques, de l'authentification et des identités de domaines Linux Red Hat Enterprise Linux 7 ou la documentation FreeIPA pour obtenir des informations sur la manière de joindre le client et les systèmes gérés sur le domaine.

Il est nécessaire d'enregistrer le système géré sur Identity Management ; pour les systèmes clients, l'enregistrement est optionnel.

Les étapes suivantes sont requises sur le système géré :

1. Veuillez installer le paquet ipa-client et enregistrer le système sur Identity Management, comme décrit dans le Guide des politiques, de l'authentification et des identités de domaines Linux Red Hat Enterprise Linux 7.

2. Veuillez copier le certificat de signature Identity Management dans le trust store en saisissant la commande suivante en tant qu'utilisateur **root** :

```
cp /etc/ipa/ca.crt /etc/pki/ca-trust/source/anchors/ipa.crt
```

3. Mettre à jour le trust store en exécutant la commande suivante en tant qu'utilisateur **root** :

```
update-ca-trust extract
```

4. Enregistrer Pegasus en tant que service dans le domaine Identity Management en exécutant la commande suivante en tant qu'utilisateur de domaine privilégié :

```
ipa service-add CIMOM/hostname
```

Remplacez *hostname* par le nom d'hôte du système géré.

Cette commande peut être exécutée à partir de n'importe quel système dans le domaine Identity Management sur lequel le paquet ipa-admintools a été installé. Une entrée de service est créée dans Identity Management, celle-ci peut être utilisée pour générer des certificats SSL.

5. Effectuez une copie de sauvegarde des fichiers PEM situés dans le répertoire **/etc/Pegasus/** (recommandé).

6. Récupérer le certificat signé en exécutant la commande suivante en tant qu'utilisateur **root** :

```
ipa-getcert request -f /etc/Pegasus/server.pem -k
/etc/Pegasus/file.pem -N CN=hostname -K CIMOM/hostname
```

Remplacez *hostname* par le nom d'hôte du système géré.

Le certificat et les fichiers clé sont désormais conservés dans des emplacements corrects. Le démon **certmonger**, installé sur le système géré par le script **ipa-client-install** garantit que le certificat soit à jour et renouvellé comme nécessaire.

Pour obtenir davantage d'informations, veuillez consulter le Guide des politiques, de l'authentification et des identités de domaines Linux Red Hat Enterprise Linux 7.

Pour enregistrer le système client et mettre à jour le trust store, veuillez suivre les étapes ci-dessous :

1. Veuillez installer le paquet ipa-client et enregistrer le système sur Identity Management, comme décrit dans le Guide des politiques, de l'authentification et des identités de domaines Linux Red Hat Enterprise Linux 7.

2. Veuillez copier le certificat de signature Identity Management dans le trust store en saisissant la commande suivante en tant qu'utilisateur **root** :

```
cp /etc/ipa/ca.crt /etc/pki/ca-trust/source/anchors/ipa.crt
```

3. Mettre à jour le trust store en exécutant la commande suivante en tant qu'utilisateur **root** :

```
update-ca-trust extract
```

Si le système client n'est pas censé être enregistré sur Identity Management, veuillez effectuer les étapes suivantes pour mettre à jour le trust store.

1. Veuillez copier le fichier **/etc/ipa/ca.crt** de manière sécurisée depuis tout autre système joint au même domaine Identity Management sur le répertoire du magasin des confiances **/etc/pki/ca-trust/source/anchors/** en tant qu'utilisateur **root**.

2. Mettre à jour le trust store en exécutant la commande suivante en tant qu'utilisateur **root** :

```
update-ca-trust extract
```

19.3.3. Gérer manuellement les certificats signés par des autorités

Gérer des certificats signés par des autorités avec d'autres mécanismes qu'Identity Management requiert une configuration manuelle plus importante.

Il est nécessaire de s'assurer que tous les clients fassent effectivement confiance au certificat de l'autorité qui signera les certificats du système géré :

- Si une autorité de certificats est de confiance par défaut, il n'est pas nécessaire d'effectuer des étapes particulières pour accomplir ceci.
- Si l'autorité du certificat n'est pas de confiance par défaut, le certificat devra être importé sur le client et sur les systèmes gérés.

  1. Copiez le certificat sur le magasin des confiances en saisissant la commande suivante en tant qu'utilisateur **root** :

     ```
     cp /path/to/ca.crt /etc/pki/ca-trust/source/anchors/ca.crt
     ```

  2. Mettre à jour le trust store en exécutant la commande suivante en tant qu'utilisateur **root** :

     ```
     update-ca-trust extract
     ```

Sur le système géré, veuillez effectuer les étapes suivantes :

1. Créer le fichier de configuration SSL **/etc/Pegasus/ssl.cnf** pour stocker des informations sur le certificat. Le contentu de ce fichier doit être similaire à l'exemple suivante :

   ```
   [ req ]
   distinguished_name     = req_distinguished_name
   prompt                 = no
   [ req_distinguished_name ]
   C                      = US
   ST                     = Massachusetts
   L                      = Westford
   O                      = Fedora
   OU                     = Fedora OpenLMI
   CN                     = hostname
   ```

   Remplacez *hostname* avec le nom de domaine complet du système géré.

2. Générez une clé privée sur le système géré en utilisant la commande suivante en tant qu'utilisateur **root** :

   ```
   openssl genrsa -out /etc/Pegasus/file.pem 1024
   ```

3. Veuillez générer une requête de signature de certificat (CSR) en exécutant la commande suivante en tant qu'utilisateur **root** :

   ```
   openssl req -config /etc/Pegasus/ssl.cnf -new -key
   /etc/Pegasus/file.pem -out /etc/Pegasus/server.csr
   ```

4. Veuillez envoyer le fichier **/etc/Pegasus/server.csr** sur l'autorité du certificat pour la signature. La procédure détaillée de soumission du fichier dépend de l'autorité du certificat en question.

5. Lorsque le certificat signé est reçu de l'autorité du certificat, enregistrez-le sous **/etc/Pegasus/server.pem**.

6. Veuillez copier le certificat de l'autorité de confiance sur le magasin des confiances Pegasus afin de vous assurer que Pegasus est capable de croire en son propre certificat en exécutant la commande suivante en tant qu'utilisateur **root** :

```
cp /path/to/ca.crt /etc/Pegasus/client.pem
```

Après avoir accompli les étapes décrites ci-dessus, les clients faisant confiance en l'autorité de signature seront en mesure de communiquer avec succès avec le CIMOM du serveur géré.

IMPORTANT

Contrairement à la solution Identity Management, si le certificat expire et doit être renouvellé, toutes les étapes manuelles décrites devront être effectuées à nouveau. Il est recommandé de renouveller les certificats avant leur expiration.

19.4. UTILISER LMISHELL

LMIShell est un client interactif et un interprète non-interactif pouvant être utilisé pour accéder aux objets CIM fournis par le CIMOM OpenPegasus. Il est basé sur l'interprète Python, mais implémente également des fonctions et des classes supplémentaires pour interagir avec les objets CIM.

19.4.1. Lancer, utiliser, et quitter LMIShell

De même qu'avec l'interprète Python, il est possible d'utiliser LMIShell soit en tant que client interactif, ou en tant qu'inteprète non-interactif pour les scripts LMIShell.

Lancer le LMIShell en mode interactif

Pour lancer l'interprète LMIShell en mode interactif, veuillez exécuter la commande **lmishell** sans argument supplémentaire :

```
lmishell
```

Par défaut, lorsque LMIShell tente d'établie une connexion avec un CIMOM, il valide un certificat côté serveur avec le magasin des confiances des autorités de certification (CA). Pour désactiver cette validation, exécutez la commande **lmishell** avec l'option de ligne de commande **--noverify** ou **-n** :

```
lmishell --noverify
```

Utiliser la saisie semi-automatique avec la touche « Tab »

Lors de l'exécution en mode interactif, l'interprète LMIShell vous permet d'appuyer sur la touche **Tab** pour compléter des structures de programmation et d'objets CIM de base, y compris les espaces de noms, les classes, les méthodes, et les propriétés d'objets.

Historique de navigation

Par défaut, LMIShell stocke toutes les commandes saisies sur l'invite interactive dans le fichier **~/.lmishell_history**. Cela vous permet de parcourir l'historique des commandes et de réutiliser les

lignes déjà saisies en mode interactif sans avoir à les réécrire dans l'invite. Pour reculer dans l'historique, veuillez appuyer sur la **Flèche vers le haut** ou sur la combinaison de touches **Ctrl**+**p**. Pour avancer dans l'historique des commandes, veuillez appuyer sur la touche **Flèche vers le bas** ou sur la combinaison de touches **Ctrl**+**n**.

LMIShell prend également en charge les recherches inversées incrémentielles. Pour rechercher une ligne en particulier dans l'historique des commandes, veuillez appuyer sur **Ctrl**+**r**,puis commencez à saisir n'importe quelle partie de la commande. Exemple :

```
> (reverse-i-search)`connect': c = connect("server.example.com",
"pegasus")
```

Pour supprimer l'historique des commandes, veuillez utiliser la fonction **clear_history()** comme suit :

```
clear_history()
```

Vous pouvez configurer le nombre de lignes stockées dans l'historique des commandes en modifiant la valeur de l'option **history_length** dans le fichier de configuration **~/.lmishellrc**. En outre, vous pouvez modifier l'emplacement du fichier de l'historique en changeant la valeur de l'option **history_file** dans ce fichier de configuration. Par exemple, pour paramétrer l'emplacement du fichier de l'historique sur **~/.lmishell_history** et pour configurer LMIShell pour qu'un maximum de **1000** lignes y soient stockées, veuillez ajouter les lignes suivantes au fichier **~/.lmishellrc** :

```
history_file = "~/.lmishell_history"
history_length = 1000
```

Gestion des exceptions

Par défaut, l'interprète LMIShell gère toutes les exceptions et utilise de valeurs de retour. Pour désactiver ce comportement de manière à gérer toutes les exceptions dans le code, veuillez utiliser la fonction **use_exceptions()** comme suit :

```
use_exceptions()
```

Pour réactiver la gestion d'exceptions automatique, veuillez utiliser :

```
use_exception(False)
```

Vous pouvez désactiver la gestion des exceptions de manière permanente en modifiant la valeur de l'option **use_exceptions** dans le fichier de configuration **~/.lmishellrc** sur **True** :

```
use_exceptions = True
```

Configurer un cache temporaire

Avec la configuration par défaut, les objets de connexion LMIShell utilisent un cache temporaire pour stocker les noms de classe CIM et les classes CIM afin de réduire les communications réseaux. Pour supprimer ce cache temporaire, veuillez utiliser la méthode **clear_cache()** comme suit :

```
object_name.clear_cache()
```

Veuillez remplacer *object_name* par le nom d'un objet de connexion.

Pour désactiver le cache temporaire d'un objet de connexion en particulier, veuillez utiliser la méthode **use_cache()** comme suit :

```
object_name.use_cache(False)
```

Pour le réactiver à nouveau, veuillez utiliser :

```
object_name.use_cache(True)
```

Il est possible de désactiver de manière permanente le cache temporaire pour les objets de connexion en modifiant la valeur de l'option **use_cache** dans le fichier de configuration **~/.lmishellrc** sur **False** :

```
use_cache = False
```

Quitter LMIShell

Pour quitter l'interprète LMIShell et retourner à l'invite de shell, veuillez appuyer sur la combinaison de touches **Ctrl**+**d** ou passez la fonction **quit()** comme suit :

```
> quit()
~]$
```

Exécuter un script LMIShell

Pour exécuter un script LMIShell, veuillez exécuter la commande **lmishell** comme suit :

```
lmishell file_name
```

Veuillez remplacer *file_name* par le nom du script. Pour inspecter un script LMIShell après son exécution, veuillez également spécifier l'option de ligne de commande **--interact** ou **-i** :

```
lmishell --interact file_name
```

L'extension de fichier préférée des scripts LMIShell est **.lmi**.

19.4.2. Connexion à un CIMOM

LMIShell vous permet de vous connecter à un CIMOM en cours d'exécution sur le même système local, ou sur une machine distante accessible à travers le réseau.

Connexion à un CIMOM distant

Pour accéder à des objets CIM fournis par un CIMOM distant, veuillez créer un objet de connexion en utilisant la fonction **connect()** comme suit :

```
connect(host_name, user_name[, password])
```

Veuillez remplacer *host_name* par le nom d'hôte du système géré, *user_name* par le nom d'un utilisateur autorisé à se connecter au CIMOM OpenPegasus exécuté sur ce système, et *password* par le mot de passe de l'utilisateur. Si le mot de passe est oublié, LMIShell demandera à l'utilisateur de le saisir. La fonction retourne un objet **LMIConnection**.

Exemple 19.1. Connexion à un CIMOM distant

Pour se connecter au CIMOM OpenPegasus en cours d'exécution sur **server.example.com** en tant qu'utilisateur **pegasus**, veuillez saisir ce qui suit dans l'invite interactive :

```
> c = connect("server.example.com", "pegasus")
password:
>
```

Connexion à un CIMOM local

LMIShell vous permet de vous connecter à un CIMOM local en utilisant un socket UNIX. Pour ce type de connexion, vous devrez exécuter l'interprète LMIShell en tant qu'utilisateur **root** et le socket **/var/run/tog-pegasus/cimxml.socket** doit exister.

Pour accéder à des objets CIM fournis par un CIMOM local, veuillez créer un objet de connexion en utilisant la fonction **connect()** comme suit :

```
connect(host_name)
```

Veuillez remplacer *host_name* par **localhost**, **127.0.0.1**, ou **::1**. la fonction retourne un objet **LMIConnection** ou **None**.

Exemple 19.2. Connexion à un CIMOM local

Pour se connecter au CIMOM OpenPegasus en cours d'exécution sur **localhost** en tant qu'utilisateur **root**, veuillez saisir ce qui suit dans l'invite interactive :

```
> c = connect("localhost")
>
```

Vérifier une connexion sur un CIMOM

La fonction **connect()** retourne soit un objet **LMIConnection**, ou **None** si la connexion n'a pas pu être établie. En outre, lorsque la fonction **connect()** ne parvient pas à établir une connexion, un message d'erreur est imprimé sur la sortie d'erreur standard.

Pour vérifier qu'une connexion à un CIMOM a bien été établie, veuillez utiliser la fonction **isinstance()** comme suit :

```
isinstance(object_name, LMIConnection)
```

Veuillez remplacer *object_name* par le nom de l'objet de connexion. Cette fonction retourne **True** si *object_name* est un objet **LMIConnection**, sinon elle retournera **False**.

Exemple 19.3. Vérifier une connexion sur un CIMOM

Pour vérifier si la variable **c**, créée dans l'Exemple 19.1, « Connexion à un CIMOM distant », contient un objet **LMIConnection**, veuillez saisir ce qui suit dans l'invite interactive :

```
> isinstance(c, LMIConnection)
True
>
```

De manière alternative, vous pouvez vérifier si **c** n'est pas égal à **None** :

```
> c is None
False
>
```

19.4.3. Utiliser des espaces de noms

Les espaces de noms LMIShell offre une manière naturelle d'organiser les classes disponibles et servent de point d'accès hiérarchique à d'autres espaces de noms et classes. L'espace de noms **root** est le premier point d'entrée d'un objet de connexion.

Répertorier les espaces de noms disponibles

Pour répertorier tous les espaces de noms disponibles, veuillez utiliser la méthode **print_namespaces()**, comme suit :

```
object_name.print_namespaces()
```

Veuillez remplacer *object_name* par le nom de l'objet à inspecter. Cette méthode imprime les espaces de noms disponibles sur la sortie standard.

Pour obtenir une liste des espaces de noms disponibles, veuillez accéder à l'attribut d'objet **namespaces** :

```
object_name.namespaces
```

Une liste de chaînes est retournée.

Exemple 19.4. Répertorier les espaces de noms disponibles

Pour inspecter l'objet de l'espace de noms **root** de l'objet de connexion **c** créé dans l'Exemple 19.1, « Connexion à un CIMOM distant » et pour répertorier les espaces de noms disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> c.root.print_namespaces()
cimv2
interop
PG_InterOp
PG_Internal
>
```

Pour assigner une liste de ces espaces de noms à une variable nommée **root_namespaces**, veuillez saisir :

```
> root_namespaces = c.root.namespaces
>
```

Accéder aux objets d'espaces de noms

Pour accéder à un objet d'espace de noms particulier, veuillez utiliser la syntaxe suivante :

```
object_name.namespace_name
```

Veuillez remplacer *object_name* par le nom de l'objet à inspecter et *namespace_name* par le nom de l'espace de noms à accéder. Un objet **LMINamespace** sera retourné.

Exemple 19.5. Accéder aux objets d'espaces de noms

Pour accéder à l'espace de noms **cimv2** de l'objet de connexion **c** créé dans l'Exemple 19.1, « Connexion à un CIMOM distant » et l'assigner à une variable nommée **ns**, veuillez saisir ce qui suit dans l'invite interactive :

```
> ns = c.root.cimv2
>
```

19.4.4. Utiliser des classes

Les classes LMIShell représentent des classes fournies par un CIMOM. Vous pouvez accéder et répertorier leurs propriétés, méthodes, instances, noms d'instances, et propriétés ValueMap, imprimer les chaînes de leur documentation et créer de nouvelles instances et de nouveaux noms d'instances.

Répertorier les classes disponibles

Pour répertorier toutes les classes disponibles, veuillez utiliser la méthode **print_classes()** comme suit :

```
namespace_object.print_classes()
```

Veuillez remplacer *namespace_object* par le nom de l'objet d'espace de noms à inspecter. Cette méthode imprime les classes disponibles sur la sortie standard.

Pour obtenir une liste des classes disponibles, veuillez utiliser la méthode **classes()** :

```
namespace_object.classes()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.6. Répertorier les classes disponibles

Pour inspecter l'objet de l'espace de noms **ns** de l'objet de connexion créé dans l'Exemple 19.5, « Accéder aux objets d'espaces de noms » et pour répertorier les classes disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> ns.print_classes()
CIM_CollectionInSystem
CIM_ConcreteIdentity
CIM_ControlledBy
CIM_DeviceSAPImplementation
CIM_MemberOfStatusCollection
...
>
```

Pour assigner une liste de ces classes à une variable nommée **cimv2_classes**, veuillez saisir :

```
> cimv2_classes = ns.classes()
>
```

Accéder aux objets de classe

Pour accéder à un objet de classe particulier qui est fourni par le CIMOM, veuillez utiliser la syntaxe suivante :

```
namespace_object.class_name
```

Veuillez remplacer *namespace_object* par le nom de l'objet d'espace de noms à inspecter et *class_name* par le nom de la classe à laquelle accéder.

Exemple 19.7. Accéder aux objets de classe

Pour accéder à la classe **LMI_IPNetworkConnection** de l'objet de l'espace de noms **ns** créé dans l'Exemple 19.5, « Accéder aux objets d'espaces de noms » et l'assigner à une variable nommée **cls**, veuillez saisir ce qui suit dans l'invite interactive :

```
> cls = ns.LMI_IPNetworkConnection
>
```

Examiner les objets de classe

Tous les objets de classe stockent des informations sur leurs noms et sur l'espace de noms auquel ils appartiennent, ainsi qu'une documentation de classe détaillée. Pour obtenir le nom d'un objet de classe particulier, veuillez utiliser la syntaxe suivante :

```
class_object.classname
```

Veuillez remplacer *class_object* par le nom de l'objet de classe à inspecter. Une représentation par chaîne du nom de l'objet sera retournée.

Pour obtenir des informations sur l'espace de noms auquel un objet de classe appartient, veuillez utiliser :

```
class_object.namespace
```

Une représentation par chaîne de l'espace de noms sera retournée.

Pour afficher la documentation détaillée d'une classe, veuillez utiliser la méthode **doc()** comme suit :

```
class_object.doc()
```

Exemple 19.8. Examiner les objets de classe

Pour inspecter l'objet de classe **cls** créé dans l'Exemple 19.7, « Accéder aux objets de classe » et pour afficher son nom et son espace de noms correspondant, veuillez saisir ce qui suit dans l'invite interactive :

```
> cls.classname
'LMI_IPNetworkConnection'
```

```
> cls.namespace
'root/cimv2'
>
```

Pour accéder à la documentation des classes, veuillez saisir :

```
> cls.doc()
Class: LMI_IPNetworkConnection
    SuperClass: CIM_IPNetworkConnection
    [qualifier] string UMLPackagePath: 'CIM::Network::IP'

    [qualifier] string Version: '0.1.0'
...
```

Répertorier les méthodes disponibles

Pour répertorier toutes les méthodes disponibles pour un objet de classe particulier, veuillez utiliser la méthode **`print_methods()`**, comme suit :

```
class_object.print_methods()
```

Veuillez remplacer *class_object* par le nom de l'objet de classe à inspecter. Cette méthode imprime les méthodes disponibles sur la sortie standard.

Pour obtenir une liste des méthodes disponibles, veuillez utiliser la méthode **`methods()`** :

```
class_object.methods()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.9. Répertorier les méthodes disponibles

Pour inspecter l'objet de classe **`cls`** créé dans l'Exemple 19.7, « Accéder aux objets de classe » et pour répertorier toutes les méthodes disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> cls.print_methods()
RequestStateChange
>
```

Pour assigner une liste de ces méthodes à une variable nommée **`service_methods`**, veuillez saisir :

```
> service_methods = cls.methods()
>
```

Répertorier les propriétés disponibles

Pour répertorier toutes les propriétés disponibles pour un objet de classe particulier, veuillez utiliser la méthode **`print_properties()`** comme suit :

```
class_object.print_properties()
```

Veuillez remplacer *class_object* par le nom de l'objet de classe à inspecter. Cette méthode imprime les propriétés disponibles dans la sortie standard.

Pour obtenir une liste des propriétés disponibles, veuillez utiliser la méthode **`properties()`** :

```
class_object.properties()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.10. Répertorier les propriétés disponibles

Pour inspecter l'objet de classe **`cls`** créé dans l'Exemple 19.7, « Accéder aux objets de classe » et pour répertorier toutes les propriétés disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> cls.print_properties()
RequestedState
HealthState
StatusDescriptions
TransitioningToState
Generation
...
>
```

Pour assigner une liste de ces classes à une variable nommée **`service_properties`**, veuillez saisir :

```
> service_properties = cls.properties()
>
```

Répertorier et afficher les propriétés ValueMap

Les classes CIM peuvent contenir des *propriétés ValueMap* dans leur définition MOF (« Managed Object Format »). Les propriétés ValueMap contiennent des valeurs constantes, qui peuvent être utiles lorsque des méthodes sont appelées ou pendant la vérification des valeurs retournées.

Pour répertorier toutes les propriétés ValueMap disponibles pour un objet de classe particulier, veuillez utiliser la méthode **`print_valuemap_properties()`**, comme suit :

```
class_object.print_valuemap_properties()
```

Veuillez remplacer *class_object* par le nom de l'objet de classe à inspecter. Cette méthode imprime les propriétés ValueMap disponibles dans la sortie standard.

Pour obtenir une liste des propriétés ValueMap disponibles, veuillez utiliser la méthode **`valuemap_properties()`** :

```
class_object.valuemap_properties()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.11. Répertorier les propriétés ValueMap

Pour inspecter l'objet de classe **cls** créé dans l'Exemple 19.7, « Accéder aux objets de classe » et pour répertorier toutes les propriétés ValueMap disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> cls.print_valuemap_properties()
RequestedState
HealthState
TransitioningToState
DetailedStatus
OperationalStatus
...
>
```

Pour assigner une liste de ces propriétés ValueMap à une variable nommée **service_valuemap_properties**, veuillez saisir :

```
> service_valuemap_properties = cls.valuemap_properties()
>
```

Pour accéder à une propriété ValueMap particulière, veuillez utiliser la syntaxe suivante :

```
class_object.valuemap_propertyValues
```

Veuillez remplacer *valuemap_property* par le nom de la propriété ValueMap à accéder.

Pour répertorier toutes les valeurs constantes, veuillez utiliser la méthode **print_values()**, comme suit :

```
class_object.valuemap_propertyValues.print_values()
```

Cette méthode imprime sur la sortie standard les valeurs constantes nommées disponibles. Vous pouvez également obtenir une liste des valeurs constantes disponibles en utilisant la méthode **values()** :

```
class_object.valuemap_propertyValues.values()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.12. Accéder aux propriétés ValueMap

Exemple 19.11, « Répertorier les propriétés ValueMap » mentione une propriété ValueMap nommée **RequestedState**. Pour inspecter cette propriété et répertorier les valeurs constantes disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> cls.RequestedStateValues.print_values()
Reset
NoChange
NotApplicable
Quiesce
Unknown
...
>
```

Pour assigner une liste de ces valeurs constantes à une variable nommée **requested_state_values**, veuillez saisir :

```
> requested_state_values = cls.RequestedStateValues.values()
>
```

Pour accéder à une valeur constante en particulier, veuillez utiliser la syntaxe suivante :

```
class_object.valuemap_propertyValues.constant_value_name
```

Veuillez remplacer *constant_value_name* par le nom de la valeur constante. Alternativement, vous pouvez utiliser la méthode **value()** comme suit :

```
class_object.valuemap_propertyValues.value("constant_value_name")
```

Pour déterminer le nom d'une valeur constante particulière, veuillez utiliser la méthode **value_name()** :

```
class_object.valuemap_propertyValues.value_name("constant_value")
```

Cette méthode retourne une chaîne.

Exemple 19.13. Accéder à des valeurs constantes

L'Exemple 19.12, « Accéder aux propriétés ValueMap » montre que la propriété **RequestedState** fournit une valeur constante nommée **Reset**. Pour accéder à cette valeur constante nommée, veuillez saisir ce qui suit dans l'invite interactive :

```
> cls.RequestedStateValues.Reset
11
> cls.RequestedStateValues.value("Reset")
11
>
```

Pour déterminer le nom de cette valeur constante, veuillez saisir :

```
> cls.RequestedStateValues.value_name(11)
u'Reset'
>
```

Rechercher un objet CIMClass

De nombreuses méthodes de classe ne requièrent pas d'accès à un objet **CIMClass**, ce qui explique pourquoi LMIShell recherche uniquement cet objet dans le CIMOM lorsqu'une méthode appelée en a besoin. Pour rechercher l'objet **CIMClass** manuellement, veuillez utiliser la méthode **fetch()**, comme suit :

```
class_object.fetch()
```

Veuillez remplacer *class_object* par le nom de l'objet de classe. Remarquez que les méthodes qui requièrent un accès à l'objet **CIMClass** le recherchent automatiquement.

19.4.5. Utiliser des instances

Les instances LMIShell représentent des instances fournies par un CIMOM. Vous pouvez obtenir et définir leurs propriétés, les répertorier et appeler leurs méthodes, imprimer les chaînes de leur documentation, obtenir une liste d'objets associés ou d'objets d'association, envoyer les objets modifiés sur le CIMOM, et supprimer des instances individuelles à partir du CIMOM.

Accéder à des instances

Pour obtenir toutes les instances disponibles d'un objet de classe particulier, veuillez utiliser la méthode **`instances()`** comme suit :

```
class_object.instances()
```

Veuillez remplacer *class_object* par le nom de l'objet de classe à inspecter. Cette méthode retourne une liste d'objets **`LMIInstance`**.

Pour accéder à la première instance d'un objet de classe, veuillez utiliser la méthode **`first_instance()`** :

```
class_object.first_instance()
```

Cette méthode retourne un objet **`LMIInstance`**.

En plus de répertorier toutes les instances ou de retourner la première, **`instances()`** et **`first_instance()`** prennent en charge un argument optionnel vous permettant de filtrer les résultats :

```
class_object.instances(criteria)
```

```
class_object.first_instance(criteria)
```

Veuillez remplacer *criteria* par un dictionnaire consistant de paires de clés-valeurs, où les clés représentent les propriétés de l'instance et les valeurs représentent les valeurs de ces propriétés.

Exemple 19.14. Accéder à des instances

Pour trouver la première instance de l'objet de classe **`cls`** créé dans l'Exemple 19.7, « Accéder aux objets de classe » dont la propriété **`ElementName`** est égale à **`eth0`** et pour l'assigner à une variable nommée **`device`**, veuillez saisir ce qui suit dans l'invite interactive :

```
> device = cls.first_instance({"ElementName": "eth0"})
>
```

Examiner des instances

Tous les objets d'instance stockent des informations sur leur nom de classe et sur l'espace de nom auquel ils appartiennent, ainsi qu'une documentation détaillée sur leurs propriétés et valeurs. En outre, les objets d'instance vous permettent de récupérer un objet d'identification unique.

Pour obtenir le nom de classe d'un objet d'instance particulier, veuillez utiliser la syntaxe suivante :

```
instance_object.classname
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à inspecter. Une représentation par chaîne du nom de la classe sera retournée.

Pour obtenir des informations sur l'espace de nom auquel un objet d'instance appartient, veuillez utiliser :

```
instance_object.namespace
```

Une représentation par chaîne de l'espace de noms sera retournée.

Pour récupérer un objet d'identification unique pour une objet d'instance, veuillez utiliser :

```
instance_object.path
```

Ceci retourne un objet **LMIInstanceName**.

Finalement, pour afficher la documentation détaillée, veuillez utiliser la méthode **doc()** comme suit :

```
instance_object.doc()
```

Exemple 19.15. Examiner des instances

Pour inspecter l'objet d'instance **device** créé dans l'Exemple 19.14, « Accéder à des instances » et pour afficher son nom de classe et son espace de nom correspondant, veuillez saisir ce qui suit dans l'invite interactive :

```
> device.classname
u'LMI_IPNetworkConnection'
> device.namespace
'root/cimv2'
>
```

Pour accéder à la documentation d'un objet d'instance, veuillez saisir :

```
> device.doc()
Instance of LMI_IPNetworkConnection
    [property] uint16 RequestedState = '12'

    [property] uint16 HealthState

    [property array] string [] StatusDescriptions
...
```

Créer de nouvelles instances

Certains fournisseurs de CIM autorisent la création de nouvelles instances d'objets de classe spécifiques. Pour créer une nouvelle instance d'un objet de classe, veuillez utiliser la méthode **create_instance()** comme suit :

```
class_object.create_instance(properties)
```

Veuillez remplacer *class_object* par le nom de l'objet de classe et *properties* par un dictionnaire consistant de paires clés-valeurs, où les clés représentent les propriétés de l'instance et les valeurs représentent les valeurs de ces propriétés. Cette méthode retourne un objet **LMIInstance**.

Exemple 19.16. Créer de nouvelles instances

La classe **LMI_Group** représente les groupes de système et la classe **LMI_Account** représente les comptes utilisateurs sur le système géré. Pour utiliser l'objet d'espace de noms **ns** créé dans l'Exemple 19.5, « Accéder aux objets d'espaces de noms », veuillez créer des instances de ces deux classes pour le groupe de systèmes nommé **pegasus** et l'utilisateur nommé **lmishell-user**, et assignez-les aux variables nommées **group** et **user**, puis saisissez ce qui suit dans l'invite interactive :

```
> group = ns.LMI_Group.first_instance({"Name" : "pegasus"})
> user = ns.LMI_Account.first_instance({"Name" : "lmishell-user"})
>
```

Pour obtenir une instance de la classe **LMI_Identity** pour l'utilisateur **lmishell-user**, veuillez utiliser :

```
> identity = user.first_associator(ResultClass="LMI_Identity")
>
```

La classe **LMI_MemberOfGroup** représente l'adhésion au groupe de systèmes. Pour utiliser la classe **LMI_MemberOfGroup** pour ajouter **lmishell-user** au groupe **pegasus**, veuillez créer une nouvelle instance de cette classe comme suit :

```
> ns.LMI_MemberOfGroup.create_instance({
...     "Member" : identity.path,
...     "Collection" : group.path})
LMIInstance(classname="LMI_MemberOfGroup", ...)
>
```

Supprimer des instances individuelles

Pour supprimer une instance particulière du CIMOM, veuillez utiliser la méthode **delete()**, comme suit :

```
instance_object.delete()
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à supprimer. Cette méthode retourne un booléen. Remarquez qu'après la suppression d'une instance, ses propriétés et méthodes seront inaccessibles.

Exemple 19.17. Supprimer des instances individuelles

La classe **LMI_Account** représente les comptes utilisateurs sur le système géré. Pour utiliser l'objet d'espace de noms **ns** créé dans l'Exemple 19.5, « Accéder aux objets d'espaces de noms », veuillez créer une instance de la classe **LMI_Account** pour l'utilisateur nommé **lmishell-user**, et assignez-la à une variable nommée **user**, puis saisissez ce qui suit dans l'invite interactive :

```
> user = ns.LMI_Account.first_instance({"Name" : "lmishell-user"})
>
```

Pour supprimer cette instance et pour supprimer **lmishell-user** du système, veuillez saisir :

```
> user.delete()
True
>
```

Répertorier et accéder aux propriétés disponibles

Pour répertorier toutes les propriétés disponibles pour un objet d'instance particulier, veuillez utiliser la méthode **print_properties()**, comme suit :

```
instance_object.print_properties()
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à inspecter. Cette méthode imprime les propriétés disponibles dans la sortie standard.

Pour obtenir une liste des propriétés disponibles, veuillez utiliser la méthode **properties()** :

```
instance_object.properties()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.18. Répertorier les propriétés disponibles

Pour inspecter l'objet d'instance **device** créé dans l'Exemple 19.14, « Accéder à des instances » et pour répertorier toutes les propriétés disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> device.print_properties()
RequestedState
HealthState
StatusDescriptions
TransitioningToState
Generation
...
>
```

Pour assigner une liste de ces propriétés à une variable nommée **device_properties**, veuillez saisir :

```
> device_properties = device.properties()
>
```

Pour obtenir la valeur actuelle d'une propriété particulière, veuillez utiliser la syntaxe suivante :

```
instance_object.property_name
```

Veuillez remplacer *property_name* par le nom de la propriété à accéder.

Pour modifier la valeur d'une propriété en particulier, assignez-lui une valeur comme suit :

```
instance_object.property_name = value
```

Veuillez remplacer *value* par la nouvelle valeur de la propriété. Remarquez qu'afin de propager le changement sur le CIMOM, vous devrez également exécuter la méthode **push()** :

```
instance_object.push()
```

Cette méthode retourne un tuple de trois éléments consistant d'une valeur de retour, de paramètres de valeur de retour, et d'une chaîne d'erreur.

Exemple 19.19. Accéder aux propriétés individuelles

Pour inspecter l'objet d'instance **device** créé dans l'Exemple 19.14, « Accéder à des instances » et pour afficher la valeur de la propriété nommée **SystemName**, veuillez saisir ce qui suit dans l'invite interactive :

```
> device.SystemName
u'server.example.com'
>
```

Répertorier et utiliser les méthodes disponibles

Pour répertorier toutes les méthodes disponibles pour un objet d'instance particulier, veuillez utiliser la méthode **print_methods()**, comme suit :

```
instance_object.print_methods()
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à inspecter. Cette méthode imprime les méthodes disponibles sur la sortie standard.

Pour obtenir une liste des méthodes disponibles, veuillez utiliser la méthode **methods()** :

```
instance_object.methods()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.20. Répertorier les méthodes disponibles

Pour inspecter l'objet d'instance **device** créé dans l'Exemple 19.14, « Accéder à des instances » et pour répertorier toutes les méthodes disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> device.print_methods()
RequestStateChange
>
```

Pour assigner une liste de ces méthodes à une variable nommée **network_device_methods**, veuillez saisir :

```
> network_device_methods = device.methods()
>
```

Pour appeler une méthode particulière, veuillez utiliser la syntaxe suivante :

```
instance_object.method_name(
    parameter=value,
    ...)
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à utiliser, *method_name* par le nom de la méthode à appeler, *parameter* par le nom du paramètre à définir, et *value* par la valeur de ce paramètre. Les méthodes retournent un tuple à trois éléments consistant d'une valeur de retour, de paramètres de retour, et d'une chaîne d'erreur.

IMPORTANT

Les objets **LMIInstance** ne réactualisent **pas** automatiquement leur contenu (propriétés, méthodes, qualificateurs, et ainsi de suite). Pour cela, veuillez utiliser la méthode **refresh()** comme décrit ci-dessous.

Exemple 19.21. Utiliser des méthodes

La classe **PG_ComputerSystem** représente le système. Pour créer une instance de cette classe en utilisant l'objet d'espace de noms **ns** créé dans l'Exemple 19.5, « Accéder aux objets d'espaces de noms » et pour l'assigner à une variable nommée **sys**, veuillez saisir ce qui suit dans l'invite interactive :

```
> sys = ns.PG_ComputerSystem.first_instance()
>
```

La classe **LMI_AccountManagementService** implémente les méthodes qui vous permettent de gérer des utilisateurs et des groupes dans le système. Pour créer une instance de cette classe et l'assigner à une variable nommée **acc**, veuillez saisir :

```
> acc = ns.LMI_AccountManagementService.first_instance()
>
```

Pour créer un nouvel utilisateur nommé **lmishell-user** dans le système, veuillez utiliser la méthode **CreateAccount()**, comme suit :

```
> acc.CreateAccount(Name="lmishell-user", System=sys)
LMIReturnValue(rval=0, rparams=NocaseDict({u'Account':
LMIInstanceName(classname="LMI_Account"...), u'Identities':
[LMIInstanceName(classname="LMI_Identity"...),
LMIInstanceName(classname="LMI_Identity"...)]}), errorstr='')
```

LMIShell prend en charge les appels de méthodes synchrones : lorsqu'une méthode synchrone est utilisée, LMIShell attend que l'objet de la tâche correspondante change son état sur « finished », puis qu'il retourne les paramètres de retour de cette tâche. LMIShell est capable d'effectuer un appel de méthode synchrone si la méthode donnée retourne un objet de l'une des classes suivantes :

- **LMI_StorageJob**
- **LMI_SoftwareInstallationJob**

- **`LMI_NetworkJob`**

LMIShell tente avant tout d'utiliser des indications comme la méthode d'attente. Si cela échoue, une méthode d'interrogation est utilisée à la place.

Pour effectuer un appel de méthode synchrone, veuillez utiliser la syntaxe suivante :

```
instance_object.Syncmethod_name(
    parameter=value,
    ...)
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à utiliser, *method_name* par le nom de la méthode à appeler, *parameter* par le nom du paramètre à définir, et *value* par la valeur de ce paramètre. Toutes les méthodes synchrones contiennent le préfixe **Sync** dans leur nom et retournent un tuple à trois éléments consistant de la valeur de retour de la tâche, des paramètres de retour de la tâche, et de la chaîne d'erreur de celle-ci.

Vous pouvez également forcer LMIShell à utiliser une méthode d'interrogation uniquement. Pour faire cela, veuillez spécifier le paramètre **`PreferPolling`**, comme suit :

```
instance_object.Syncmethod_name(
    PreferPolling=True
    parameter=value,
    ...)
```

Répertorier et afficher les paramètres ValueMap

Les méthodes CIM peuvent contenir des *paramètres ValueMap* dans leur définition MOF (« Managed Object Format »). Les paramètres ValueMap contiennent des valeurs constantes.

Pour répertorier tous les paramètres ValueMap disponibles d'une méthode particulière, veuillez utiliser la méthode **`print_valuemap_parameters()`**, comme suit :

```
instance_object.method_name.print_valuemap_parameters()
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance et *method_name* par le nom de la méthode à inspecter. Cette méthode imprime les paramètres ValueMap disponibles sur la sortie standard.

Pour obtenir une liste des paramètres ValueMap disponibles, veuillez utiliser la méthode **`valuemap_parameters()`** :

```
instance_object.method_name.valuemap_parameters()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.22. Répertorier les paramètres ValueMap

Pour inspecter l'objet d'instance **`acc`** créé dans l'Exemple 19.21, « Utiliser des méthodes » et pour répertorier tous les paramètres ValueMap disponibles de la méthode **`CreateAccount()`**, veuillez saisir ce qui suit dans l'invite interactive :

```
> acc.CreateAccount.print_valuemap_parameters()
CreateAccount
>
```

Pour assigner une liste de ces paramètres ValueMap à une variable nommée **create_account_parameters**, veuillez saisir :

```
> create_account_parameters = acc.CreateAccount.valuemap_parameters()
>
```

Pour accéder à un paramètre ValueMap particulier, veuillez utiliser la syntaxe suivante :

```
instance_object.method_name.valuemap_parameterValues
```

Veuillez remplacer *valuemap_parameter* par le nom du paramètre ValueMap à accéder.

Pour répertorier toutes les valeurs constantes, veuillez utiliser la méthode **print_values()**, comme suit :

```
instance_object.method_name.valuemap_parameterValues.print_values()
```

Cette méthode imprime sur la sortie standard les valeurs constantes nommées disponibles. Vous pouvez également obtenir une liste des valeurs constantes disponibles en utilisant la méthode **values()** :

```
instance_object.method_name.valuemap_parameterValues.values()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.23. Accéder aux paramètres ValueMap

L'Exemple 19.22, « Répertorier les paramètres ValueMap » mentionne un paramètre ValueMap nommé **CreateAccount**. Pour inspecter ce paramètre et répertorier les valeurs constantes disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> acc.CreateAccount.CreateAccountValues.print_values()
Operationunsupported
Failed
Unabletosetpasswordusercreated
Unabletocreatehomedirectoryusercreatedandpasswordset
Operationcompletedsuccessfully
>
```

Pour assigner une liste de ces valeurs constantes à une variable nommée **create_account_values**, veuillez saisir :

```
> create_account_values = acc.CreateAccount.CreateAccountValues.values()
>
```

Pour accéder à une valeur constante en particulier, veuillez utiliser la syntaxe suivante :

```
instance_object.method_name.valuemap_parameterValues.constant_value_name
```

Veuillez remplacer *constant_value_name* par le nom de la valeur constante. Alternativement, vous pouvez utiliser la méthode **value()** comme suit :

```
instance_object.method_name.valuemap_parameterValues.value("constant_value_name")
```

Pour déterminer le nom d'une valeur constante particulière, veuillez utiliser la méthode **value_name()** :

```
instance_object.method_name.valuemap_parameterValues.value_name("constant_value")
```

Cette méthode retourne une chaîne.

Exemple 19.24. Accéder à des valeurs constantes

L'Exemple 19.23, « Accéder aux paramètres ValueMap » montre que le paramètre ValueMap **CreateAccount** fournit une valeur constante nommée **Failed**. Pour accéder à cette valeur constante nommée, veuillez saisir ce qui suit dans l'invite interactive :

```
> acc.CreateAccount.CreateAccountValues.Failed
2
> acc.CreateAccount.CreateAccountValues.value("Failed")
2
>
```

Pour déterminer le nom de cette valeur constante, veuillez saisir :

```
> acc.CreateAccount.CreateAccountValues.value_name(2)
u'Failed'
>
```

Actualiser les objets d'instances

Les objets locaux utilisés par LMIShell, qui représentent des objets CIM du côté CIMOM, peuvent devenir obsolètes, si ceux-ci changent pendant leur utilisation avec les objets de LMIShell. Pour mettre à jour les propriétés et les méthodes d'un objet d'instance en particulier, veuillez utiliser la méthode **refresh()**, comme suit :

```
instance_object.refresh()
```

Veuillez remplacer *instance_object* par le nom de l'objet à actualiser. Cette méthode retourne un tuple de trois éléments consistant d'une valeur de retour, d'un paramètre de valeur de retour, et d'une chaîne d'erreur.

Exemple 19.25. Actualiser les objets d'instances

Pour mettre à jour les propriétés et méthodes de l'objet d'instance **device** créé dans l'Exemple 19.14, « Accéder à des instances », veuillez saisir ce qui suit dans l'invite interactive :

```
> device.refresh()
LMIReturnValue(rval=True, rparams=NocaseDict({}), errorstr='')
>
```

Afficher la représentation MOF

Pour afficher la représentation MOF (Format d'objet géré, « Managed Object Format ») d'un objet d'instance, veuillez utiliser la méthode **`tomof()`** comme suit :

```
instance_object.tomof()
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à inspecter. Cette méthode imprime la représentation MOF de l'objet sur la sortie standard.

Exemple 19.26. Afficher la représentation MOF

Pour afficher la représentation MOF de l'objet d'instance **`device`** créé dans l'Exemple 19.14, « Accéder à des instances », veuillez saisir ce qui suit dans l'invite interactive :

```
> device.tomof()
instance of LMI_IPNetworkConnection {
        RequestedState = 12;
        HealthState = NULL;
        StatusDescriptions = NULL;
        TransitioningToState = 12;
...
```

19.4.6. Utiliser des noms d'instance

Les noms d'instance LMIShell sont des objets contenant un ensemble de clés principales et leurs valeurs. Ce type d'objet identifie une instance de manière exacte.

Accéder aux noms d'instances

Les objets **`CIMInstance`** sont identifiés par des objets **`CIMInstanceName`**. Pour obtenir une liste de tous les objets des noms d'instances disponibles, veuillez utiliser la méthode **`instance_names()`**, comme suit :

```
class_object.instance_names()
```

Veuillez remplacer *class_object* par le nom de l'objet de classe à inspecter. Cette méthode retourne une liste d'objets **`LMIInstanceName`**.

Pour accéder au premier objet du nom d'instance d'un objet de classe, veuillez utiliser la méthode **`first_instance_name()`** :

```
class_object.first_instance_name()
```

Cette méthode retourne un objet **`LMIInstanceName`**.

En plus de répertorier tous les objets de noms d'instances ou de retourner le premier, **`instances_names()`** et **`first_instance_name()`** prennent en charge un argument optionnel vous permettant de filtrer les résultats :

```
class_object.instance_names(criteria)
```

```
class_object.first_instance_name(criteria)
```

Veuillez remplacer *criteria* par un dictionnaire consistant de paires de clés-valeurs, où les clés représentent les propriétés clés et les valeurs représentent les valeurs de ces propriétés clés.

Exemple 19.27. Accéder aux noms d'instances

Pour trouver le premier nom d'instance de l'objet de classe **`cls`** créé dans l'Exemple 19.7, « Accéder aux objets de classe » dont la propriété clé **`Name`** est égale à **`eth0`** et pour l'assigner à une variable nommée **`device_name`**, veuillez saisir ce qui suit dans l'invite interactive :

```
> device_name = cls.first_instance_name({"Name": "eth0"})
>
```

Examiner des noms d'instance

Tous les objets de noms d'instances stockent des informations sur leurs noms de classe et sur l'espace de nom auquel ils appartiennent.

Pour obtenir le nom de classe d'un objet de nom d'instance particulier, veuillez utiliser la syntaxe suivante :

```
instance_name_object.classname
```

Veuillez remplacer *instance_name_object* par l'objet du nom d'instance à inspecter. Une représentation par chaîne du nom de la classe sera retournée.

Pour obtenir des informations sur l'espace de noms auquel un objet de nom d'instance appartient, veuillez utiliser :

```
instance_name_object.namespace
```

Une représentation par chaîne de l'espace de noms sera retournée.

Exemple 19.28. Examiner des noms d'instance

Pour inspecter l'objet du nom d'instance **`device_name`** créé dans l'Exemple 19.27, « Accéder aux noms d'instances » et pour afficher son nom de classe et son espace de nom correspondant, veuillez saisir ce qui suit dans l'invite interactive :

```
> device_name.classname
u'LMI_IPNetworkConnection'
> device_name.namespace
'root/cimv2'
>
```

Créer de nouveaux noms d'instance

LMIShell permet de créer un nouvel objet **`CIMInstanceName`** encapsulé (« wrapped ») si vous connaissez toutes les clés principales d'un objet distant. Cet objet de nom d'instance peut ensuite être utilisé pour récupérer l'objet d'instance entier.

Pour créer un nouveau nom d'instance d'un objet de classe, veuillez utiliser la méthode **`new_instance_name()`**, comme suit :

```
class_object.new_instance_name(key_properties)
```

Veuillez remplacer *class_object* par le nom de l'objet de classe et *key_properties* par un dictionnaire consistant de paires clés-valeurs, où les clés représentent les propriétés clés et les valeurs représentent les valeurs de ces propriétés clés. Cette méthode retourne un objet **LMIInstanceName**.

Exemple 19.29. Créer de nouveaux noms d'instance

La classe **LMI_Account** représente les comptes utilisateurs sur le système géré. Pour utiliser l'objet d'espace de noms **ns** créé dans l'Exemple 19.5, « Accéder aux objets d'espaces de noms », et pour créer une nouvelle instance de la classe **LMI_Account** représentant l'utilisateur **lmishell-user** sur le système géré, veuillez saisir ce qui suit dans l'invite interactive :

```
> instance_name = ns.LMI_Account.new_instance_name({
...     "CreationClassName" : "LMI_Account",
...     "Name" : "lmishell-user",
...     "SystemCreationClassName" : "PG_ComputerSystem",
...     "SystemName" : "server"})
>
```

Répertorier et accéder aux propriétés clés

Pour répertorier toutes les propriétés clés disponibles pour un objet de nom d'instance particulier, veuillez utiliser la méthode **print_key_properties()**, comme suit :

```
instance_name_object.print_key_properties()
```

Veuillez remplacer *instance_name_object* par le nom de l'objet du nom d'instance à inspecter. Cette méthode imprime les propriétés clés dans la sortie standard.

Pour obtenir une liste des propriétés clés disponibles, veuillez utiliser la méthode **key_properties()** :

```
instance_name_object.key_properties()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.30. Répertorier les propriétés clés disponibles

Pour inspecter l'objet du nom d'instance **device_name** créé dans l'Exemple 19.27, « Accéder aux noms d'instances » et pour répertorier toutes les propriétés clés disponibles, veuillez saisir ce qui suit dans l'invite interactive :

```
> device_name.print_key_properties()
CreationClassName
SystemName
Name
SystemCreationClassName
>
```

Pour assigner une liste de ces propriétés clés à une variable nommée **device_name_properties**, veuillez saisir :

```
> device_name_properties = device_name.key_properties()
```

```
>
```

Pour obtenir la valeur actuelle d'une propriété clé particulière, veuillez utiliser la syntaxe suivante :

```
instance_name_object.key_property_name
```

Veuillez remplacer *key_property_name* par le nom de la propriété clé à accéder.

Exemple 19.31. Accéder aux propriétés clés individuelles

Pour inspecter l'objet du nom d'instance **device_name** créé dans l'Exemple 19.27, « Accéder aux noms d'instances » et pour afficher la valeur de la propriété clé nommée **SystemName**, veuillez saisir ce qui suit dans l'invite interactive :

```
> device_name.SystemName
u'server.example.com'
>
```

Convertir des noms d'instances en instances

Chaque nom d'instance peut être converti en instance. Pour faire cela, veuillez utiliser la méthode **to_instance()** comme suit :

```
instance_name_object.to_instance()
```

Remplacer *instance_name_object* par l'objet du nom d'instance à convertir. Cette méthode retourne un objet **LMIInstance**.

Exemple 19.32. Convertir des noms d'instances en instances

Pour convertir l'objet du nom d'instance **device_name** créé dans l'Exemple 19.27, « Accéder aux noms d'instances » en objet d'instance et pour l'assigner à une variable nommée **device**, veuillez saisir ce qui suit dans l'invite interactive :

```
> device = device_name.to_instance()
>
```

19.4.7. Utiliser des objets associés

Le CIM (« Common Information Model », ou modèle d'informations communes) définit une relation d'association entre objets gérés.

Accéder à des instances associées

Pour obtenir une liste de tous les objets associés avec un objet d'instance particulier, veuillez utiliser la méthode **associators()** comme suit :

```
instance_object.associators(
    AssocClass=class_name,
    ResultClass=class_name,
    ResultRole=role,
```

```
    IncludeQualifiers=include_qualifiers,
    IncludeClassOrigin=include_class_origin,
    PropertyList=property_list)
```

Pour accéder au premier objet associé à un objet d'instance particulier, veuillez utiliser la méthode **`first_associator()`** :

```
instance_object.first_associator(
    AssocClass=class_name,
    ResultClass=class_name,
    ResultRole=role,
    IncludeQualifiers=include_qualifiers,
    IncludeClassOrigin=include_class_origin,
    PropertyList=property_list)
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à inspecter. Il est possible de filtrer les résultats en spécifiant les paramètres suivants :

- **`AssocClass`** — chaque objet retourné doit être associé à l'objet source à travers une instance de cette classe ou de l'une de ses sous-classes. La valeur par défaut est **`None`**.
- **`ResultClass`** — chaque objet retourné doit être une instance de cette classe ou de l'une de ses sous-classes, ou doit être cette classe ou l'une de ses sous-classes. La valeur par défaut est **`None`**.
- **`Role`** — chaque objet retourné doit être associé à l'objet source à travers une association dans laquelle l'objet source joue le rôle spécifié. Le nom de la propriété dans la classe d'association faisant référence à l'objet source doit correspondre à la valeur de ce paramètre. La valeur par défaut est **`None`**.
- **`ResultRole`** — chaque objet retourné doit être associé à l'objet source à travers une association pour laquelle l'objet retourné joue le rôle spécifié. Le nom de la propriété dans la classe d'association faisant référence à l'objet retourné doit correspondre à la valeur de ce paramètre. La valeur par défaut est **`None`**.

Les paramètres restants font référence à :

- **`IncludeQualifiers`** — un booléen indiquant si les qualificateurs de chaque objet (y compris les qualificateurs de l'objet et de toute propriété retournée) doivent être inclus en tant qu'éléments « QUALIFIER » dans la réponse. La valeur par défaut est **`False`**.
- **`IncludeClassOrigin`** — un booléen indiquant si l'attribut CLASSORIGIN doit être présent sur tous les éléments correspondants de chaque objet retourné. La valeur par défaut est **`False`**.
- **`PropertyList`** — les membres de cette liste définissent un ou plusieurs noms de propriétés. Les objets retournés n'incluront pas d'éléments pour toute propriété manquante à cette liste. Si **`PropertyList`** est une liste vide, aucune propriété ne sera incluse dans les objets retournés. Si **`None`**, aucun filtre supplémentaire ne sera défini. La valeur par défaut est **`None`**.

Exemple 19.33. Accéder à des instances associées

La classe **`LMI_StorageExtent`** représente les périphériques blocs disponibles sur le système. Pour utiliser l'objet d'espace de nom **`ns`** créé dans l'Exemple 19.5, « Accéder aux objets d'espaces de noms », veuillez créer une instance de la classe **`LMI_StorageExtent`** pour le périphérique bloc

nommé **/dev/vda**, et assignez-la à une variable nommée **vda**, puis saisissez ce qui suit dans l'invite interactive :

```
> vda = ns.LMI_StorageExtent.first_instance({
...     "DeviceID" : "/dev/vda"})
>
```

Pour obtenir une liste de toutes les partitions de disque de ce périphérique bloc et l'assigner à une variable nommée **vda_partitions**, veuillez utiliser la méthode **associators()** comme suit :

```
> vda_partitions = vda.associators(ResultClass="LMI_DiskPartition")
>
```

Accéder aux noms d'instances associés

Pour obtenir une liste des tous les noms d'instances associés d'un objet d'instance particulier, veuillez utiliser la méthode **associator_names()** comme suit :

```
instance_object.associator_names(
    AssocClass=class_name,
    ResultClass=class_name,
    Role=role,
    ResultRole=role)
```

Pour accéder au premier nom d'instance associé d'un objet d'instance particulier, veuillez utiliser la méthode **first_associator_name()** :

```
instance_object.first_associator_name(
    AssocClass=class_object,
    ResultClass=class_object,
    Role=role,
    ResultRole=role)
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à inspecter. Il est possible de filtrer les résultats en spécifiant les paramètres suivants :

- **AssocClass** — chaque nom retourné identifie un objet devant être associé à l'objet source à travers une instance de cette classe ou de l'une de ses sous-classes. La valeur par défaut est **None**.
- **ResultClass** — chaque nom retourné identifie un objet qui est une instance de cette classe ou l'une de ses sous-classes, ou qui doit être cette classe ou l'une de ses sous-classes. La valeur par défaut est **None**.
- **Role** — chaque nom retourné identifie un objet devant être associé à l'objet source à travers une association dans laquelle l'objet source joue le rôle spécifié. Le nom de la propriété dans la classe d'association faisant référence à l'objet source doit correspondre à la valeur de ce paramètre. La valeur par défaut est **None**.
- **ResultRole** — chaque nom retourné identifie un objet devant être associé à l'objet source à travers une association dans laquelle l'objet nommé retourné joue le rôle spécifié. Le nom de la propriété dans la classe d'association faisant référence à l'objet retourné doit correspondre à la valeur de ce paramètre. La valeur par défaut est **None**.

Exemple 19.34. Accéder aux noms d'instances associés

Pour utiliser l'objet d'instance **vda** créé dans l'Exemple 19.33, « Accéder à des instances associées », veuillez obtenir une liste de ses noms d'instances associés, puis assignez-la à une variable nommée **vda_partitions**, saisissez :

```
> vda_partitions = vda.associator_names(ResultClass="LMI_DiskPartition")
>
```

19.4.8. Utiliser des objets d'association

Le CIM (« Common Information Model », ou modèle d'informations communes) définit une relation d'association entre objets gérés. Les objets d'association définissent la relation entre deux objets.

Accéder à des instances d'association

Pour obtenir une liste d'objets d'association faisant référence à un objet cible particulier, veuillez utiliser la méthode **references()**, comme suit :

```
instance_object.references(
    ResultClass=class_name,
    Role=role,
    IncludeQualifiers=include_qualifiers,
    IncludeClassOrigin=include_class_origin,
    PropertyList=property_list)
```

Pour accéder au premier objet d'association faisant référence à un objet cible particulier, veuillez utiliser la méthode **first_reference()** :

```
instance_object.first_reference(
...     ResultClass=class_name,
...     Role=role,
...     IncludeQualifiers=include_qualifiers,
...     IncludeClassOrigin=include_class_origin,
...     PropertyList=property_list)
>
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à inspecter. Il est possible de filtrer les résultats en spécifiant les paramètres suivants :

- **ResultClass** — chaque objet retourné doit être une instance de cette classe ou de l'une de ses sous-classes, ou doit être cette classe ou l'une de ses sous-classes. La valeur par défaut est **None**.
- **Role** — chaque objet retourné doit faire référence à l'objet cible à travers une propriété avec un nom correspondant à la valeur de ce paramètre. La valeur par défaut est **None**.

Les paramètres restants font référence à :

- **IncludeQualifiers** — un booléen indiquant si chaque objet (y compris les qualificateurs de l'objet et de toute propriété retournée) doit être inclus en tant qu'élément « QUALIFIER » dans la réponse. La valeur par défaut est **False**.
- **IncludeClassOrigin** — un booléen indiquant si l'attribut CLASSORIGIN doit être présent sur

tous les éléments correspondants de chaque objet retourné. La valeur par défaut est **`False`**.

- **`PropertyList`** — les membres de cette liste définissent un ou plusieurs noms de propriété. Les objets retournés n'incluront pas d'éléments pour toute propriété manquante de cette liste. Si **`PropertyList`** est une liste vide, aucune propriété ne sera incluse dans les objets retournés. Si **`None`**, aucun filtre supplémentaire ne sera défini. La valeur par défaut est **`None`**.

Exemple 19.35. Accéder à des instances d'association

La classe **`LMI_LANEndpoint`** représente un point de terminaison de communication associé à un certain périphérique d'interface réseau. Pour utiliser l'objet d'espace de noms **`ns`** créé dans l'Exemple 19.5, « Accéder aux objets d'espaces de noms », veuillez créer une instance de la classe **`LMI_LANEndpoint`** pour le périphérique d'interface réseau nommé eth0, et assignez-la à une variable nommée **`lan_endpoint`**, puis saisissez ce qui suit dans l'invite interactive :

```
> lan_endpoint = ns.LMI_LANEndpoint.first_instance({
...     "Name" : "eth0"})
>
```

Pour accéder au premier objet d'association faisant référence à un objet **`LMI_BindsToLANEndpoint`** et pour l'assigner à une variable nommée **`bind`**, veuillez saisir :

```
> bind = lan_endpoint.first_reference(
...     ResultClass="LMI_BindsToLANEndpoint")
>
```

Vous pouvez désormais utiliser la propriété **`Dependent`** pour accéder à la classe dépendante **`LMI_IPProtocolEndpoint`** qui représente l'adresse IP du périphérique de l'interface réseau correspondant :

```
> ip = bind.Dependent.to_instance()
> print ip.IPv4Address
192.168.122.1
>
```

Accéder aux noms d'instances d'associations

Pour obtenir une liste des noms d'instances d'associations d'un objet particulier, utiliser la méthode **`reference_names()`** comme suit :

```
instance_object.reference_names(
    ResultClass=class_name,
    Role=role)
```

Pour accéder au premier nom d'instance associé d'un objet d'instance particulier, veuillez utiliser la méthode **`first_reference_name()`** :

```
instance_object.first_reference_name(
    ResultClass=class_name,
    Role=role)
```

Veuillez remplacer *instance_object* par le nom de l'objet d'instance à inspecter. Il est possible de filtrer les résultats en spécifiant les paramètres suivants :

- **`ResultClass`** — chaque objet retourné doit identifier une instance de cette classe ou de l'une de ses sous-classes, ou cette classe ou l'une de ses sous-classes. La valeur par défaut est **`None`**.

- **`Role`** — chaque objet retourné doit identifier un objet ui se réfère à une instance cible à travers une propriété avec un nom correspondant à la valeur de ce paramètre. La valeur par défaut est **`None`**.

Exemple 19.36. Accéder aux noms d'instances d'associations

Pour utiliser l'objet d'instance **`lan_endpoint`** créé dans l'Exemple 19.35, « Accéder à des instances d'association », accéder au premier nom d'instance d'association faisant référence à un objet **`LMI_BindsToLANEndpoint`**, et l'assigner à une variable nommée **`bind`**, veuillez saisir :

```
> bind = lan_endpoint.first_reference_name(
...     ResultClass="LMI_BindsToLANEndpoint")
```

Vous pouvez désormais utiliser la propriété **`Dependent`** pour accéder à la classe dépendante **`LMI_IPProtocolEndpoint`** qui représente l'adresse IP du périphérique de l'interface réseau correspondant :

```
> ip = bind.Dependent.to_instance()
> print ip.IPv4Address
192.168.122.1
>
```

19.4.9. Travailler avec des indications

Une indication est une réaction à un événement particulier qui se produit en réponse à un changement précis parmi les données. LMIShell peut être abonné à une indication pour recevoir des réponses pour de tels événements.

S'abonner à des indications

Pour s'abonner à une indication, veuillez utiliser la méthode **`subscribe_indication()`**, comme suit :

```
connection_object.subscribe_indication(
    QueryLanguage="WQL",
    Query='SELECT * FROM CIM_InstModification',
    Name="cpu",
    CreationNamespace="root/interop",
    SubscriptionCreationClassName="CIM_IndicationSubscription",
    FilterCreationClassName="CIM_IndicationFilter",
    FilterSystemCreationClassName="CIM_ComputerSystem",
    FilterSourceNamespace="root/cimv2",
    HandlerCreationClassName="CIM_IndicationHandlerCIMXML",
    HandlerSystemCreationClassName="CIM_ComputerSystem",
    Destination="http://host_name:5988")
```

Alternativement, vous pouvez utiliser une version plus courte de l'appel de méthode, comme suit :

```
connection_object.subscribe_indication(
    Query='SELECT * FROM CIM_InstModification',
```

```
    Name="cpu",
    Destination="http://host_name:5988")
```

Veuillez remplacer *connection_object* par un objet de connexion et un *host_name* ayant le nom d'hôte du système auquel vous souhaitez fournir les indications.

Par défaut, tous les abonnements créés par l'interprète LMIShell sont automatiquement supprimés lorsque l'interprète se ferme. Pour modifier ce comportement, transmettez le paramètre **`permanente = True`** à l'appel de méthode **`subscribe_indication()`**. Cela empêchera LMIShell de supprimer l'abonnement.

Exemple 19.37. S'abonner à des indications

Pour utiliser l'objet de connection **c** créé dans Exemple 19.1, « Connexion à un CIMOM distant » et pour vous abonner à une indication nommée **cpu**, veuillez saisir ce qui suit dans l'invite interactive :

```
> c.subscribe_indication(
...     QueryLanguage="WQL",
...     Query='SELECT * FROM CIM_InstModification',
...     Name="cpu",
...     CreationNamespace="root/interop",
...     SubscriptionCreationClassName="CIM_IndicationSubscription",
...     FilterCreationClassName="CIM_IndicationFilter",
...     FilterSystemCreationClassName="CIM_ComputerSystem",
...     FilterSourceNamespace="root/cimv2",
...     HandlerCreationClassName="CIM_IndicationHandlerCIMXML",
...     HandlerSystemCreationClassName="CIM_ComputerSystem",
...     Destination="http://server.example.com:5988")
LMIReturnValue(rval=True, rparams=NocaseDict({}), errorstr='')
>
```

Répertorier les indications abonnées

Pour répertorier toutes les indications auxquelles vous êtes abonnées, utiliser la méthode **`print_subscribed_indications()`**, comme suit :

```
connection_object.print_subscribed_indications()
```

Veuillez remplacer *connection_object* par le nom de l'objet de connexion à inspecter. Cette méthode affichera les indications auquelles vous êtes abonnées sur la sortie standard.

Pour obtenir une liste de toutes les indications auxquelles vous êtes abonnées, veuillez utiliser la méthode **`subscribed_indications()`** comme suit :

```
connection_object.subscribed_indications()
```

Cette méthode retourne une liste de chaînes.

Exemple 19.38. Répertorier les indications abonnées

Pour inspecter l'objet de connexion **c** créé dans l'Exemple 19.1, « Connexion à un CIMOM distant » et pour répertorier les indications abonnées, veuillez saisir ce qui suit dans l'invite interactive :

```
> c.print_subscribed_indications()
```

```
>
```

Pour assigner une liste de ces indications à une variable nommée **indications**, veuillez saisir :

```
> indications = c.subscribed_indications()
>
```

Se désabonner des indications

Par défaut, tous les abonnements créés par l'interprète LMIShell sont automatiquement supprimés lorsque l'interprète se ferme. Pour supprimer un abonnement individuel plus tôt, utilisez la méthode **unsubscribe_indication()** comme suit :

```
connection_object.unsubscribe_indication(indication_name)
```

Veuillez remplace *connection_object* par le nom de l'objet de connexion et *indication_name* par le nom de l'indication à supprimer.

Pour supprimer tous les abonnements, veuillez utiliser la méthode **unsubscribe_all_indications()** :

```
connection_object.unsubscribe_all_indications()
```

Exemple 19.39. Se désabonner des indications

Pour utiliser l'objet de connection **c** créé dans Exemple 19.1, « Connexion à un CIMOM distant » et pour vous désabonner de l'indication créée dans l' Exemple 19.37, « S'abonner à des indications », veuillez saisir ce qui suit dans l'invite interactive :

```
> c.unsubscribe_indication('cpu')
LMIReturnValue(rval=True, rparams=NocaseDict({}), errorstr='')
>
```

Implémenter un gestionnaire d'indications (« Indication Handler »)

La méthode **subscribe_indication()** vous permet de spécifier le nom d'hôte du système sur lequel vous souhaitez remettre les indications. L'exemple ci-dessous montre comment implémenter un gestionnaire d'indications :

```
> def handler(ind, arg1, arg2, **kwargs):
...     exported_objects = ind.exported_objects()
...     do_something_with(exported_objects)
> listener = LmiIndicationListener("0.0.0.0", listening_port)
> listener.add_handler("indication-name-XXXXXXXX", handler, arg1, arg2,
**kwargs)
> listener.start()
>
```

Le premier argument du gestionnaire est un objet **LmiIndication**, qui contient une liste de méthodes et d'objets exportés par l'indication. D'autres paramètres sont spécifiques à l'utilisateur : ces arguments devront être spécifiés lors de l'ajout d'un gestionnaire au listener.

Dans l'exemple ci-dessus, l'appel de la méthode **add_handler()** utilise une chaîne spéciale avec huit

caractères « X ». Ces caractères sont remplacés par une chaîne aléatoire générée par des listeners afin d'éviter une collision possible de noms de gestionnaire. Pour utiliser la chaîne aléatoire, commencez par lancer le listener d'indications, puis abonnez-vous à une indication afin que la propriété **`Destination`** de l'objet du gestionnaire contienne la valeur suivante : ***`schema://host_name/random_string`***.

Exemple 19.40. Implémenter un gestionnaire d'indications (« Indication Handler »)

Le script suivant illustre comment écrire un gestionnaire qui surveille un système géré se trouvant sur **`192.168.122.1`** et qui appelle la fonction **`indication_callback()`** lorsqu'un nouveau compte utilisateur est créé :

```
#!/usr/bin/lmishell

import sys
from time import sleep
from lmi.shell.LMIUtil import LMIPassByRef
from lmi.shell.LMIIndicationListener import LMIIndicationListener

# These are passed by reference to indication_callback
var1 = LMIPassByRef("some_value")
var2 = LMIPassByRef("some_other_value")

def indication_callback(ind, var1, var2):
    # Do something with ind, var1 and var2
    print ind.exported_objects()
    print var1.value
    print var2.value

c = connect("hostname", "username", "password")

listener = LMIIndicationListener("0.0.0.0", 65500)
unique_name = listener.add_handler(
    "demo-XXXXXXXX",     # Creates a unique name for me
    indication_callback, # Callback to be called
    var1,                # Variable passed by ref
    var2                 # Variable passed by ref
)

listener.start()

print c.subscribe_indication(
    Name=unique_name,
    Query="SELECT * FROM LMI_AccountInstanceCreationIndication WHERE
SOURCEINSTANCE ISA LMI_Account",
    Destination="192.168.122.1:65500"
)

try:
    while True:
        sleep(60)
except KeyboardInterrupt:
    sys.exit(0)
```

19.4.10. Exemple d'utilisation

Cette section fournit un certain nombre d'exemples de divers fournisseurs CIM distribués avec les paquets OpenLMI. Tous les exemples de cette section utilisent les deux définitions de variables suivantes :

```
c = connect("host_name", "user_name", "password")
ns = c.root.cimv2
```

Veuillez remplacer *host_name* par le nom d'hôte du système géré, *user_name* par le nom d'un utilisateur autorisé à se connecter au CIMOM OpenPegasus exécuté sur ce système, et *password* avec le mot de passe de l'utilisateur.

Utiliser le fournisseur du service OpenLMI

Le paquet openlmi-service installe un fournisseur CIM pour gérer les services système. Les exemples ci-dessous illustrent comment utiliser ce fournisseur CIM pour répertorier les services système disponibles et comment les lancer, les arrêter, les activer, et les désactiver.

Exemple 19.41. Répertorier les services disponibles

Pour répertorier tous les services disponibles sur la machine gérée, ainsi que pour des informations indiquant si le service a été lancé (**TRUE**) ou est arrêté (**FALSE**), et la chaîne de statut, veuillez utiliser l'extrait de code suivant :

```
for service in ns.LMI_Service.instances():
    print "%s:\t%s" % (service.Name, service.Status)
```

Pour uniquement répertorier les services activés par défaut, veuillez utiliser l'extrait de code suivant :

```
cls = ns.LMI_Service
for service in cls.instances():
    if service.EnabledDefault == cls.EnabledDefaultValues.Enabled:
        print service.Name
```

Remarquez que la valeur de la propriété **`EnabledDefault`** est égale à **2** pour les services activés et à **3** pour les services désactivés.

Pour afficher les informations concernant le service **cups**, veuillez utiliser :

```
cups = ns.LMI_Service.first_instance({"Name": "cups.service"})
cups.doc()
```

Exemple 19.42. Lancer et arrêter des services

Pour lancer et arrêter le service **cups** et pour afficher son statut actuel, veuillez utiliser l'extrait de code suivant :

```
cups = ns.LMI_Service.first_instance({"Name": "cups.service"})
cups.StartService()
print cups.Status
cups.StopService()
print cups.Status
```

Exemple 19.43. Activer et désactiver des services

Pour activer et désactiver le service **cups** et pour afficher sa propriété **EnabledDefault**, veuillez utiliser l'extrait de code suivant :

```
cups = ns.LMI_Service.first_instance({"Name": "cups.service"})
cups.TurnServiceOff()
print cups.EnabledDefault
cups.TurnServiceOn()
print cups.EnabledDefault
```

Utiliser le fournisseur réseau OpenLMI

Le paquet openlmi-networking installe un fournisseur CIM pour la mise en réseau. Les exemples ci-dessous illustrent comment utiliser ce fournisseur CIM pour répertorier les adresses IP associées à un certain numéro de port, créer une nouvelle connexion, configurer une adresse IP statique, et activer une connexion.

Exemple 19.44. Répertorier les adresses IP associées à un numéro de port donné

Pour répertorier toutes les adresses associées à l'interface réseau eth0, veuillez utiliser l'extrait de code suivant :

```
device = ns.LMI_IPNetworkConnection.first_instance({'ElementName':
'eth0'})
for endpoint in
device.associators(AssocClass="LMI_NetworkSAPSAPDependency",
ResultClass="LMI_IPProtocolEndpoint"):
    if endpoint.ProtocolIFType ==
ns.LMI_IPProtocolEndpoint.ProtocolIFTypeValues.IPv4:
        print "IPv4: %s/%s" % (endpoint.IPv4Address,
endpoint.SubnetMask)
    elif endpoint.ProtocolIFType ==
ns.LMI_IPProtocolEndpoint.ProtocolIFTypeValues.IPv6:
        print "IPv6: %s/%d" % (endpoint.IPv6Address,
endpoint.IPv6SubnetPrefixLength)
```

Cet extrait de code utilise la classe **LMI_IPProtocolEndpoint** associée à une classe **LMI_IPNetworkConnection** donnée.

Pour afficher la passerelle par défaut, veuillez utiliser l'extrait de code suivant :

```
for rsap in
device.associators(AssocClass="LMI_NetworkRemoteAccessAvailableToElement
", ResultClass="LMI_NetworkRemoteServiceAccessPoint"):
    if rsap.AccessContext ==
ns.LMI_NetworkRemoteServiceAccessPoint.AccessContextValues.DefaultGatewa
y:
        print "Default Gateway: %s" % rsap.AccessInfo
```

La passerelle par défaut est représentée par une instance **LMI_NetworkRemoteServiceAccessPoint** dont la propriété **AccessContext** est égale à **DefaultGateway**.

Pour obtenir une liste de serveurs DNS, le modèle de l'objet doit être traversé comme suit :

1. Obtenez les instances **LMI_IPProtocolEndpoint** associées avec **LMI_IPNetworkConnection** en utilisant **LMI_NetworkSAPSAPDependency**.

2. Utilisez la même association pour les instances **LMI_DNSProtocolEndpoint**.

Les instances **LMI_NetworkRemoteServiceAccessPoint** dont la propriété **AccessContext** est égale au serveur DNS associé à travers **LMI_NetworkRemoteAccessAvailableToElement** possèdent l'adresse du serveur DNS dans la propriété **AccessInfo**.

Il peut y avoir davantage de chemins possibles vers **RemoteServiceAccessPath** et les entrées peuvent être dupliquées. L'extrait de code suivant utilise la fonction **set()** pour supprimer les entrées dupliquées de la liste des serveurs DNS :

```
dnsservers = set()
for ipendpoint in
device.associators(AssocClass="LMI_NetworkSAPSAPDependency",
ResultClass="LMI_IPProtocolEndpoint"):
    for dnsedpoint in
ipendpoint.associators(AssocClass="LMI_NetworkSAPSAPDependency",
ResultClass="LMI_DNSProtocolEndpoint"):
        for rsap in
dnsedpoint.associators(AssocClass="LMI_NetworkRemoteAccessAvailableToEle
ment", ResultClass="LMI_NetworkRemoteServiceAccessPoint"):
            if rsap.AccessContext ==
ns.LMI_NetworkRemoteServiceAccessPoint.AccessContextValues.DNSServer:
                dnsservers.add(rsap.AccessInfo)
print "DNS:", ", ".join(dnsservers)
```

Exemple 19.45. Créer une nouvelle connexion et configurer une adresse IP statique

Pour créer un nouveau paramètre avec une configuration IPv4 statique et IPv6 stateless pour l'interface réseau eth0, veuillez utiliser l'extrait de code suivant :

```
capability = ns.LMI_IPNetworkConnectionCapabilities.first_instance({
'ElementName': 'eth0' })
result = capability.LMI_CreateIPSetting(Caption='eth0 Static',
        IPv4Type=capability.LMI_CreateIPSetting.IPv4TypeValues.Static,

IPv6Type=capability.LMI_CreateIPSetting.IPv6TypeValues.Stateless)
setting = result.rparams["SettingData"].to_instance()
for settingData in
setting.associators(AssocClass="LMI_OrderedIPAssignmentComponent"):
    if setting.ProtocolIFType ==
ns.LMI_IPAssignmentSettingData.ProtocolIFTypeValues.IPv4:
        # Set static IPv4 address
        settingData.IPAddresses = ["192.168.1.100"]
```

```
        settingData.SubnetMasks = ["255.255.0.0"]
        settingData.GatewayAddresses = ["192.168.1.1"]
        settingData.push()
```

Cet extrait de code crée un nouveau paramètre en appelant la méthode **LMI_CreateIPSetting()** sur l'instance de **LMI_IPNetworkConnectionCapabilities**, qui est associée avec **LMI_IPNetworkConnection** à travers **LMI_IPNetworkConnectionElementCapabilities**. Il utilise également la méthode **push()** pour modifier le paramètre.

Exemple 19.46. Activer une connexion

Pour appliquer un paramètre à l'interface réseau, veuillez appeler la méthode **ApplySettingToIPNetworkConnection()** de la classe **LMI_IPConfigurationService**. Cette méthode est asynchrone et retourne une tâche. Les extraits de code suivants illustrent comment appeler cette méthode de manière synchrone :

```
setting = ns.LMI_IPAssignmentSettingData.first_instance({ "Caption":
"eth0 Static" })
port = ns.LMI_IPNetworkConnection.first_instance({ 'ElementName': 'ens8'
})
service = ns.LMI_IPConfigurationService.first_instance()
service.SyncApplySettingToIPNetworkConnection(SettingData=setting,
IPNetworkConnection=port, Mode=32768)
```

Le paramètre **Mode** affecte la manière dont le paramètre est appliqué. Les valeurs les plus couramment utilisées de ce paramètre sont comme suit :

- **1** — applique le paramètre immédiatement et le rend automatiquement actif.
- **2** — rend le paramètre automatiquement actif mais ne l'applique pas immédiatement.
- **4** — déconnecte et désactive l'activation automatique.
- **5** — ne modifie pas l'état du paramètre, désactive uniquement l'activation automatique.
- **32768** — applique le paramètre.
- **32769** — déconnexion.

Utiliser le fournisseur de stockage OpenLMI

Le paquet openlmi-storage installe un fournisseur CIM pour la gestion du stockage. Les exemples ci-dessous illustrent comment utiliser ce fournisseur CIM pour créer un groupe de volumes, un volume logique, un système de fichiers, pour monter un système de fichiers, et pour répertorier les périphériques blocs connus par le système.

En plus des variables **c** et **ns**, ces exemples utilisent les définitions de variables suivantes :

```
MEGABYTE = 1024*1024
storage_service = ns.LMI_StorageConfigurationService.first_instance()
filesystem_service =
ns.LMI_FileSystemConfigurationService.first_instance()
```

Exemple 19.47. Créer un groupe de volumes

Pour créer un nouveau groupe de volumes se trouvant dans **/dev/myGroup/** qui possède trois membres et une taille d'extension par défaut de 4 Mo, veuillez utiliser l'extrait de code suivant :

```
# Find the devices to add to the volume group
# (filtering the CIM_StorageExtent.instances()
# call would be faster, but this is easier to read):
sda1 = ns.CIM_StorageExtent.first_instance({"Name": "/dev/sda1"})
sdb1 = ns.CIM_StorageExtent.first_instance({"Name": "/dev/sdb1"})
sdc1 = ns.CIM_StorageExtent.first_instance({"Name": "/dev/sdc1"})

# Create a new volume group:
(ret, outparams, err) = storage_service.SyncCreateOrModifyVG(
        ElementName="myGroup",
        InExtents=[sda1, sdb1, sdc1])
vg = outparams['Pool'].to_instance()
print "VG", vg.PoolID, \
        "with extent size", vg.ExtentSize, \
        "and",  vg.RemainingExtents, "free extents created."
```

Exemple 19.48. Créer un volume logique

Pour créer deux volumes logiques avec une taille de 100 Mo, veuillez utiliser cet extrait de code :

```
# Find the volume group:
vg = ns.LMI_VGStoragePool.first_instance({"Name":
"/dev/mapper/myGroup"})

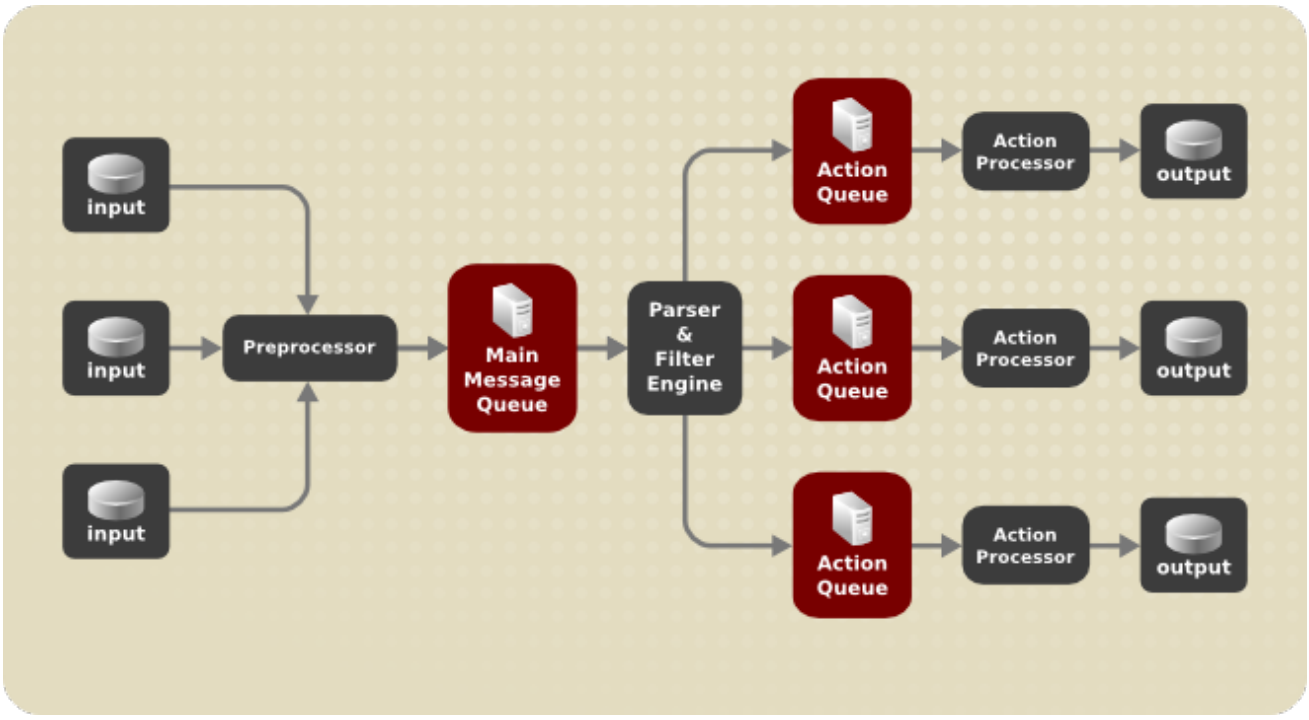
# Create the first logical volume:
(ret, outparams, err) = storage_service.SyncCreateOrModifyLV(
        ElementName="Vol1",
        InPool=vg,
        Size=100 * MEGABYTE)
lv = outparams['TheElement'].to_instance()
print "LV", lv.DeviceID, \
        "with", lv.BlockSize * lv.NumberOfBlocks,\
        "bytes created."

# Create the second logical volume:
(ret, outparams, err) = storage_service.SyncCreateOrModifyLV(
        ElementName="Vol2",
        InPool=vg,
        Size=100 * MEGABYTE)
lv = outparams['TheElement'].to_instance()
print "LV", lv.DeviceID, \
        "with", lv.BlockSize * lv.NumberOfBlocks, \
        "bytes created."
```

Exemple 19.49. Créer un système de fichiers

Pour créer un système de fichiers **ext3** sur un volume logique **lv** selon l'Exemple 19.48, « Créer un volume logique », veuillez utiliser l'extrait de code suivant :

```
(ret, outparams, err) = filesystem_service.SyncLMI_CreateFileSystem(

FileSystemType=filesystem_service.LMI_CreateFileSystem.FileSystemTypeVal
ues.EXT3,
        InExtents=[lv])
```

Exemple 19.50. Monter un système de fichiers

Pour monter le système de fichier créé dans l'Exemple 19.49, « Créer un système de fichiers », veuillez utiliser l'extrait de code suivant :

```
# Find the file system on the logical volume:
fs = lv.first_associator(ResultClass="LMI_LocalFileSystem")

mount_service = ns.LMI_MountConfigurationService.first_instance()
(rc, out, err) = mount_service.SyncCreateMount(
     FileSystemType='ext3',
     Mode=32768, # just mount
     FileSystem=fs,
     MountPoint='/mnt/test',
     FileSystemSpec=lv.Name)
```

Exemple 19.51. Répertorier les périphériques blocs

Pour répertorier tous les périphériques blocs connus par le système, veuillez utiliser l'extrait de code suivant :

```
devices = ns.CIM_StorageExtent.instances()
for device in devices:
    if lmi_isinstance(device, ns.CIM_Memory):
        # Memory and CPU caches are StorageExtents too, do not print
them
        continue
    print device.classname,
    print device.DeviceID,
    print device.Name,
    print device.BlockSize*device.NumberOfBlocks
```

Utiliser le fournisseur de matériel OpenLMI

Le paquet openlmi-hardware installe un fournisseur CIM pour surveiller le matériel. Les exemples ci-dessous illustrent comment utiliser ce fournisseur CIM pour récupérer des informations sur le CPU, les modules de mémoire, les périphériques PCI, et sur le constructeur et le modèle de la machine.

Exemple 19.52. Afficher les informations du CPU

Pour afficher les informations de base du CPU, comme le nom du CPU, le nombre de cœurs du processeur, et le nombre de threads de matériel, veuillez utiliser l'extrait de code suivant :

```
cpu = ns.LMI_Processor.first_instance()
cpu_cap = cpu.associators(ResultClass="LMI_ProcessorCapabilities")[0]
print cpu.Name
print cpu_cap.NumberOfProcessorCores
print cpu_cap.NumberOfHardwareThreads
```

Exemple 19.53. Afficher les informations de mémoire

Pour afficher des informations de base sur les modules de mémoire, comme leur taille individuelle, veuillez utiliser l'extrait de code suivant :

```
mem = ns.LMI_Memory.first_instance()
for i in mem.associators(ResultClass="LMI_PhysicalMemory"):
    print i.Name
```

Exemple 19.54. Afficher les informations du châssis

Pour afficher des informations de base sur la machine, comme son fabricant ou son modèle, veuillez utiliser l'extrait de code suivant :

```
chassis = ns.LMI_Chassis.first_instance()
print chassis.Manufacturer
print chassis.Model
```

Exemple 19.55. Répertorier les périphériques PCI

Pour répertorier tous les périphériques PCI connus par le système, veuillez utiliser l'extrait de code suivant :

```
for pci in ns.LMI_PCIDevice.instances():
    print pci.Name
```

19.5. UTILISER OPENLMI SCRIPTS

L'interprète est construit au-dessus de modules Python pouvant être utilisés pour développer des outils de gestion personnalisés. Le projet OpenLMI Scripts fournit un certain nombre de bibliothèques Python permettant d'interagir avec des fournisseurs OpenLMI. En outre, il est distribué avec **lmi**, un utilitaire extensible pouvant être utilisé pour interagir avec ces bibliothèques à partir de la ligne de commande.

Pour installer OpenLMI Scripts sur votre système, veuillez saisir ce qui suit dans une invite de shell :

```
easy_install --user openlmi-scripts
```

Cette commande installe les modules Python et l'utilitaire **lmi** dans le répertoire **~/.local/**. Pour étendre la fonctionnalité de l'utilitaire **lmi**, veuillez installer des modules OpenLMI supplémentaires en utilisant la commande suivante :

```
easy_install --user package_name
```

Pour obtenir une liste complète des modules disponibles, veuillez consulter le Site web Python. Pour obtenir des informations sur OpenLMI Scripts, veuillez consulter la documentation officelle d'OpenLMI Scripts.

19.6. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur OpenLMI et sur la gestion du système en général, veuillez consulter les ressources répertoriées ci-dessous.

Documentation installée

- lmishell(1) — The manual page for the **lmishell** client and interpreter provides detailed information about its execution and usage.

Documentation en ligne

- Guide de mise en réseau Red Hat Enterprise Linux 7 — Le *Guide de mise en réseau* de Red Hat Enterprise Linux 7 documente les informations pertinentes à la configuration et à l'administration des interfaces réseau et des services réseau sur le système.
- Guide d'administration du stockage Red Hat Enterprise Linux 7 — Le *Guide d'administration du stockage* de Red Hat Enterprise Linux 7 fournit des instructions sur la manière de gérer les périphériques de stockage et les systèmes de fichiers sur le système.
- Guide de gestion de l'alimentation Red Hat Enterprise Linux 7 — Le *Guide de gestion de l'alimentation* de Red Hat Enterprise Linux 7 explique comment gérer la consommation électrique du système de manière efficace. Il traite de différentes techniques permettant de réduire la consommation électrique des serveurs et des ordinateurs portables, et explique comment chaque technique affecte les performances générales du système.
- Guide de la politique, de l'authentification, et de l'identité de domaines Linux Red Hat Enterprise Linux 7 — Le *Guide de la politique, de l'authentification, et de l'identité de domaines Linux* de Red Hat Enterprise Linux 7 couvre tous les aspects de l'installation, de la configuration et de la gestion des domaines IPA, y compris les serveurs et les clients. Ce guide est conçu pour les administrateurs systèmes et administrateurs informatique.
- Documentation FreeIPA — La *Documentation FreeIPA* sert de documentation utilisateur principale pour l'utilisation du projet de gestion des identités FreeIPA (« FreeIPA Identity Management »).
- Page d'accueil OpenSSL — La page d'accueil d'*OpenSSL* fournit une vue d'ensemble du projet OpenSSL.
- Documentation Mozilla NSS — La *Documentation Mozilla NSS* sert de documentation utilisateur principale pour utiliser le projet Mozilla NSS.

Voir aussi

- Le Chapitre 3, *Gérer les utilisateurs et les groupes* documente comment gérer les groupes et utilisateurs système dans l'interface utilisateur graphique et sur la ligne de commande.
- Le Chapitre 8, *Yum* décrit comment utiliser le gestionnaire de paquets **Yum** pour rechercher, installer, mettre à jour, et désinstaller des paquets sur la ligne de commande.

- Le Chapitre 9, *Gérer les services avec systemd* offre une introduction à **systemd** et documente comment utiliser la commande **systemctl** pour gérer des services système, configurer des cibles systemd, et exécuter des commandes de gestion de l'alimentation.
- Le Chapitre 10, *OpenSSH* décrit comment configurer un serveur SSH et comment utiliser les utilitaires client **ssh**, **scp**, et **sftp** pour y accéder.

CHAPITRE 20. AFFICHER ET GÉRER DES FICHIERS JOURNAUX

Les *fichiers journaux* sont des fichiers qui contiennent des messages sur le système, y compris sur le noyau, les services, et les applications qui sont exécutées dessus. Il existe plusieurs types de fichiers journaux permettant de stocker diverses informations. Par exemple, il existe un fichier journal système par défaut, un fichier journal pour les messages de sécurité uniquement, et un fichier journal pour les tâches cron.

Les fichiers journaux peuvent être très utiles dans de nombreuses situations, par exemple, pour résoudre un problème avec le système, lors des tentatives de charger un pilote de noyau, ou pour vérifier les tentatives de connexion non autorisées sur le système. Ce chapitre nous explique où trouver les fichiers de journalisation, comment les trouver et ce qu'il faut chercher à l'intérieur.

Certains fichiers journaux sont contrôlés par un démon nommé **rsyslogd**. Le démon **rsyslogd** est un remplacement amélioré de **sysklogd**, et fournit un filtrage étendu, une redirection de messages protégée par un cryptage, diverses options de configuration, des modules d'entrée et de sortie, et la prise en charge du transport via les protocoles **TCP** ou **UDP**. Notez que **rsyslog** est compatible avec **sysklogd**.

Les fichiers journaux peuvent également être gérés par le démon **journald** – un composant de **systemd**. Le démon **journald** capture les messages Syslog, les messages du journal du noyau, les messages du disque RAM initial et du début du démarrage, ainsi que les messages inscrits sur la sortie standard et la sortie d'erreur standard de tous les services, puis il les indexe et rend ceci disponibles à l'utilisateur. Le format du fichier journal natif, qui est un fichier binaire structuré et indexé, améliore les recherches et permet une opération plus rapide, celui-ci stocke également des informations de métadonnées, comme l'horodatage ou les ID d'utilisateurs. Les fichiers journaux produits par **journald** sont par défaut non persistants, les fichiers journaux sont uniquement stockés en mémoire ou dans une petite mémoire tampon en anneau dans le répertoire **/run/log/journal/**. La quantité de données journalisées dépend de la mémoire libre, lorsque la capacité limite est atteinte, les entrées les plus anciennes sont supprimées. Cependant, ce paramètre peut être altéré – veuillez consulter la Section 20.10.5, « Activer le stockage persistant ». Pour obtenir davantage d' informations sur le Journal, veuillez consulter la Section 20.10, « Utiliser le Journal ».

Par défaut, ces deux outils de journalisation coexistent sur votre système. Le démon **journald** est l'outil principal de résolution de problèmes. Il fournit également les données supplémentaires nécessaires à la création de messages journaux structurés. Les données acquises par **journald** sont transférées sur le socket **/run/systemd/journal/syslog** pouvant être utilisé par **rsyslogd** pour traiter davantage les données. Cependant, **rsyslog** effectue l'intégration par défaut via le module d'entrée **imjournal**, évitant ainsi le socket susmentionné. Il est également possible de transférer les données dans la direction opposée, depuis **rsyslogd** vers **journald** par le module **omjournal**. Veuillez consulter la Section 20.7, « Interaction de Rsyslog et de Journal » pour obtenir des informations supplémentaires. L'intégration permet de maintenir des journaux basés texte sous un format cohérent afin d'assurer la compatibilité avec de possibles applications ou configurations dépendants de **rsyslogd**. Vous pouvez également maintenir les messages rsyslog sous un format structuré (veuillez consulter la Section 20.8, « Journalisation structurée avec Rsyslog »).

20.1. LOCALISER LES FICHIERS JOURNAUX

On peut trouver une liste de fichiers de journalisation maintenue par **rsyslogd** dans le fichier de configuration **/etc/rsyslog.conf**. La plupart des fichiers journaux se trouvent dans le répertoire **/var/log/**. Certaines applications, comme **httpd** et **samba**, peuvent avoir un répertoire dans **/var/log/** pour leurs fichiers journaux.

Vous remarquerez peut-être de multiples fichiers dans le répertoire **/var/log/** suivis par des chiffres (par exemple, **cron-20100906**). Ces chiffres représentent un horodatage ajouté à un fichier journal rotatif. Les fichiers journaux sont en rotation afin que leur taille ne devienne pas trop importante. Le paquet **logrotate** contient une tâche faisant pivoter les fichiers journaux automatiquement, en fonction du fichier de configuration **/etc/logrotate.conf** et des fichiers de configuration situés dans le répertoire **/etc/logrotate.d/**.

20.2. CONFIGURATION DE BASE DE RSYSLOG

Le fichier de configuration principal de **rsyslog** est **/etc/rsyslog.conf**. Vous pouvez y spécifier des *directives globales*, des *modules*, et des *règles* consistant en une partie « *filter* » (filtre) et une partie « *action* ». Vous pouvez également ajouter des commentaires sous la forme d'un texte suivant un caractère dièse (#).

20.2.1. Filtres

Une règle est spécifiée par une partie *filtre*, qui sélectionne un sous-ensemble de messages syslog, et une partie *action*, qui spécifie quoi faire avec les messages sélectionnés. Pour définir une règle dans votre fichier de configuration **/etc/rsyslog.conf**, définissez un filtre et une action, un par ligne, et séparez-les par un ou plusieurs espaces ou par une ou plusieurs tabulations.

rsyslog offre diverses manières de filtrer les messages syslog en fonction des propriétés sélectionnées. Les méthodes de filtrage disponibles peuvent être divisées en filtres *basés Facility/Priorité*, *basés Propriété*, et *basés Expression*.

Filtres basés Facility/Priorité

La manière la plus utilisée et la plus connue de filtrer des messages syslog consiste à utiliser les filtres basés facility/priorité (type/priorité), qui filtrent les messages syslog en se basant sur deux conditions : *facility* (ou type) et *priorité* séparés par un point. Pour créer un sélecteur, veuillez utiliser la syntaxe suivante :

```
FACILITY.PRIORITY
```

où :

- *FACILITY* spécifie le sous-système produisant un message syslog particulier. Par exemple, le sous-système **mail** gère tous les messages syslog concernant le courrier. *FACILITY* peut être représenté par l'un des mots-clés suivants (ou par un code numérique) : **kern** (0), **user** (1), **mail** (2), **daemon** (3), **auth** (4), **syslog** (5), **lpr** (6), **news** (7), **uucp** (8), **cron** (9), **authpriv** (10), **ftp** (11), et **local0** via **local7** (16 - 23).

- *PRIORITY* spécifie la priorité d'un message syslog. *PRIORITY* peut être représenté par l'un des mots-clés suivants (ou par un numéro) : **debug** (7), **info** (6), **notice** (5), **warning** (4), **err** (3), **crit** (2), **alert** (1), et **emerg** (0).

  La syntaxe susmentionnée sélectionne les messages syslog avec la priorité définie ou avec une priorité *supérieure*. En faisant précéder tout mot-clé de priorité par le caractère égal (**=**), vous spécifiez que seuls les messages syslog avec la priorité spécifiée seront sélectionnés. Toutes les autres priorités seront ignorées. De même, faire précéder un mot-clé de priorité par un point d'exclamation (**!**) sélectionne tous les messages syslog sauf ceux dont la priorité est définie.

En plus des mots-clés spécifiés ci-dessus, vous pouvez également utiliser un astérisque (*****) pour définir tous les types (« Facilities ») ou priorités (selon l'endroit où vous placez l'astérisque, avant ou

après la virgule). Spécifier le mot-clé de priorité **none** (aucun) sert aux types sans priorité donnée. Les conditions du type et de la priorité ne respectent pas la casse.

Pour définir de multiples types et priorités, veuillez les séparer par une virgule (`,`). Pour définir de multiples sélecteurs sur une seule ligne, séparez-les avec un point-virgule (`;`). Remarquez que chaque sélecteur dans le champ des sélecteurs est capable d'outrepasser les précédents, ce qui peut exclure certaines priorités du modèle.

Exemple 20.1. Filtres basés Facility/Priorité

Ci-dessous figurent quelques exemples de filtres basés facility/priorité simples pouvant être spécifiés dans **`/etc/rsyslog.conf`**. Pour sélectionner tous les messages syslog du noyau avec n'importe quelle priorité, ajoutez le texte suivant dans le fichier de configuration.

```
kern.*
```

Pour sélectionner tous les messages syslog de courrier avec une priorité **`crit`** ou supérieure, veuillez utiliser le format ci-dessous :

```
mail.crit
```

Pour sélectionner tous les messages syslog cron sauf ceux avec la priorité **`info`** ou **debug**, paramétrez la configuration sous le format suivant :

```
cron.!info,!debug
```

Filtres basés Propriété

Les filtres basés propriété vous permettent de filtrer les messages syslog par toute propriété, comme ***`timegenerated`*** ou ***`syslogtag`***. Pour obtenir davantage d'informations sur les propriétés, veuillez consulter la section intitulée « Propriétés ». Vous pouvez comparer chacune des propriétés spécifiées avec une valeur particulière en utilisant l'une des opérations de comparaison répertoriées dans la Tableau 20.1, « Opération de comparaison basée propriété ». Les noms des propriétés et les opérations de comparaison respectent la casse.

Un filtre basé propriété doit commencer par un caractère des deux-points (`:`). Pour définir le filtre, veuillez utiliser la syntaxe suivante :

```
:PROPERTY, [!]COMPARE_OPERATION, "STRING"
```

où :

- L'attribut *PROPERTY* spécifie la propriété souhaitée.
- Le point d'exclamation optionnel (`!`) ignore la sortie de l'opération de comparaison. D'autres opérateurs booléens ne sont pas actuellement pris en charge par les filtres basés propriété.
- L'attribut *COMPARE_OPERATION* spécifie l'une des opérations de comparaison répertoriée dans la Tableau 20.1, « Opération de comparaison basée propriété ».
- L'attribut *STRING* spécifie la valeur à laquelle est comparé le texte fourni par la propriété. Cette valeur doit se trouver entre guillemets. Pour échapper certains caractères se trouvant dans la chaîne (par exemple un guillemet (`"`)), veuillez utiliser le caractère de la barre

oblique inversée (\).

Tableau 20.1. Opération de comparaison basée propriété

Opération de comparaison	Description
contains	Vérifie si la chaîne fournie correspond à toute partie du texte fourni par la propriété. Pour effectuer des comparaisons ne respectant pas la casse, veuillez utiliser ***contains_i***.
isequal	Compare la chaîne fournie avec tout le texte de la propriété. Ces deux valeurs doivent être exactement pareilles pour correspondre.
startswith	Vérifie si la chaîne fournie est trouvée précisément au début du texte fourni par la propriété. Pour effectuer des comparaisons ne respectant pas la casse, veuillez utiliser ***startswith_i***.
regex	Compare l'expression BRE (« Basic Regular Expression ») POSIX fournie avec le texte fourni par la propriété.
ereregex	Compare l'expression régulière ERE (« Extended Regular Expression ») POSIX fournie avec le texte fourni par la propriété.
isempty	Vérifie si la propriété est vide. La valeur est abandonnée. Ceci est particulièrement utile lors de l'utilisation de données normalisées, avec lesquelles certains champs peuvent être remplis en se basant sur le résultat de la normalisation.

Exemple 20.2. Filtres basés Propriété

Ci-dessous figurent quelques exemples de filtres basés propriété pouvant être spécifiés dans **/etc/rsyslog.conf**. Pour sélectionner les messages syslog qui contiennent la chaîne **error** dans leur texte, veuillez utiliser :

```
:msg, contains, "error"
```

Le filtre suivant sélectionne les messages syslog reçus du nom d'hôte **host1** :

```
:hostname, isequal, "host1"
```

Pour sélectionner les messages syslog qui ne mentionnent pas les mots **fatal** et **error** avec ou sans texte entre ces mots (par exemple, **fatal lib error**), veuillez saisir :

```
:msg, !regex, "fatal .* error"
```

Filtres basés Expression

Les filtres basés Expression sélectionnent des messages syslog en fonction de l'arithmétique définie, ou des opérations des booléens et des chaînes. Les filtres basés Expression utilisent un langage propre à **rsyslog**, nommé *RainerScript*, pour créer des filtres complexes.

La syntaxe de base des filtres basés expression ressemble à ceci :

```
if EXPRESSION then ACTION else ACTION
```

où :

- L'attribut *EXPRESSION* représente une expression à évaluer, par exemple : **`$msg startswith 'DEVNAME'`** ou **`$syslogfacility-text == 'local0'`**. Il est possible de spécifier plus d'une expression dans un seul filtre en utilisant les opérateurs **`and`** et **`or`**.
- L'attribut *ACTION* représente une action à effectuer si l'expression retourne la valeur **`true`**. Ceci peut être une action unique, ou un script complexe arbitraire se trouvant entre des accolades.
- Les filtres basés Expression sont indiqués par le mot-clé *if* au début d'une nouvelle ligne. Le mot-clé *then* sépare l'*EXPRESSION* de l'*ACTION*. Optionnellement, il est possible d'utiliser le mot-clé *else* pour spécifier l'action à effectuer au cas où la condition n'est pas remplie.

Avec les filtres basés expression, il est possible d'imbriquer les conditions en utilisant un script se trouvant entre des accolades, comme dans l'Exemple 20.3, « Filtres basés Expression ». Le script vous permet d'utiliser des filtres *basés facility/priorité* dans l'expression. D'autre part, les filtres *basés propriété* ne sont pas recommandés ici. RainerScript prend en charge les expressions régulières avec les fonctions spécialisées **`re_match()`** et **`re_extract()`**.

Exemple 20.3. Filtres basés Expression

L'expression suivante contient deux conditions imbriquées. Les fichiers journaux créés par un programme nommé *prog1* sont divisés en deux fichiers basés sur la présence de la chaîne « test » dans le message.

```
if $programname == 'prog1' then {
   action(type="omfile" file="/var/log/prog1.log")
   if $msg contains 'test' then
     action(type="omfile" file="/var/log/prog1test.log")
   else
     action(type="omfile" file="/var/log/prog1notest.log")
}
```

Voir la section intitulée « Documentation en ligne » pour obtenir d'autres exemples de filtres basés expression. RainerScriptr est la base du nouveau format de configuration de **rsyslog**, voir Section 20.3, « Utiliser le nouveau format de configuration »

20.2.2. Actions

Les actions spécifient ce qui doit être fait avec les messages filtrés par un sélecteur défini au préalable. Ci-dessous figurent certaines des actions pouvant être définies dans la règle :

Enregistrer des messages syslog dans les fichiers journaux

La majorité des actions spécifient le fichier journal sur lequel un message syslog est enregistré. Ceci peut être effectué en spécifiant un chemin de fichier après le sélecteur défini au préalable :

```
FILTER PATH
```

où *FILTER* correspond au sélecteur spécifié par l'utilisateur et *PATH* est le chemin d'un fichier cible.

Par exemple, la règle suivante comprend un sélecteur, qui sélectionne tous les messages syslog **cron** et une action qui les enregistre sans le fichier journal **/var/log/cron.log** :

```
cron.* /var/log/cron.log
```

Par défaut, le fichier journal est synchronisé chaque fois qu'un message syslog est généré. Utilisez un tiret (**-**) comme préfixe du chemin du fichier que vous spécifiez pour omettre la synchronisation :

```
FILTER -PATH
```

Remarquez que vous pourriez perdre des informations si le système quitte juste après une tentative d'écriture. Cependant ce paramètre peut améliorer les performances, particulièrement si vous exécutez des programmes produisant des messages de journaux très détaillés.

Le chemin du fichier spécifié peut être *statique* ou *dynamique*. Les fichiers statiques sont représentés par un chemin de fichier fixe, comme décrit dans l'exemple ci-dessus. Des chemins de fichier dynamiques peuvent être différents en fonction du message reçu. Les chemins de fichiers dynamiques sont représentés par un modèle et un point d'interrogation (**?**) comme préfixe :

```
FILTER ?DynamicFile
```

où *DynamicFile* est le nom d'un modèle prédéfini qui modifie les chemins de sortie. Il est possible d'utiliser un tiret (**-**) comme préfixe pour désactiver la synchronisation. Il est également possible d'utiliser de multiples modèles séparés par un point-virgule (**;**). Pour obtenir davantage d'informations sur les modèles, veuillez consulter la section intitulée « Générer des noms de fichiers dynamiques ».

Pendant l'utilisation du système X Window, si le fichier spécifié est un **terminal** ou un périphérique **/dev/console**, des messages syslog seront envoyés respectivement dans la sortie standard (à l'aide d'une gestion spéciale du **terminal**), ou dans la console (à l'aide d'une gestion spéciale de **/dev/console**).

Envoyer des messages syslog sur le réseau

rsyslog permet d'envoyer et de recevoir des messages syslog sur le réseau. Cette fonctionnalité permet d'administrer des messages syslog d'hôtes multiples sur une seule machine. Pour transférer des messages syslog sur une machine distante, veuillez utiliser la syntaxe suivante :

```
@[(zNUMBER)]HOST:[PORT]
```

où :

- Le caractère arobase (@) indique que les messages syslog sont transférés sur un hôte utilisant le protocole **UDP**. Pour utiliser le protocole **TCP**, utilisez deux caractères arobase sans les espacer (@@).
- Le paramètre optionnel ***zNUMBER*** active la compression **zlib** des messages syslog. L'attribut

NUMBER spécifie le niveau de compression (de 1 – le plus bas à 9 – maximum). Le gain de compression est automatiquement vérifié par **rsyslogd**, les messages sont uniquement compressés s'il y a un gain de compression et les messages faisant moins de 60 octets ne sont jamais compressés.

- L'attribut *HOST* spécifie l'hôte qui reçoit les messages syslog sélectionnés.
- L'attribut *PORT* spécifie le port de la machine hôte.

Lors de la spécification d'une adresse **IPv6** en tant qu'hôte, veuillez ajouter l'adresse entre crochets (**[**, **]**).

Exemple 20.4. Envoyer des messages syslog sur le réseau

Ci-dessous figurent quelques exemples d'actions qui transfèrent des messages syslog sur le réseau (remarquez que toutes les actions sont précédées d'un sélecteur qui sélectionne tous les messages prioritaires). Pour transférer des messages sur **192.168.0.1** via le protocole **UDP**, veuillez saisir :

```
*.* @192.168.0.1
```

Pour transférer des messages sur « example.com » en utilisant le port 6514 et le protocole **TCP**, veuillez utiliser :

```
*.* @@example.com:6514
```

La commande ci-dessous compresse les messages avec **zlib** (compression de niveau 9) et les transfère vers **2001:db8::1** en utilisant le protocole **UDP**

```
*.* @(z9)[2001:db8::1]
```

Canaux de sortie

Les canaux de sortie sont principalement utilisés pour spécifier la taille maximum qu'un fichier journal peut faire. Ceci est particulièrement utile lors des rotations de fichiers journaux (pour obtenir davantage d'informations, veuillez consulter la Section 20.2.5, « Rotation de journaux »). Un canal de sortie est une collection d'informations sur l'action de sortie. Les canaux de sortie sont définis par la directive **$outchannel**. Pour définir un canal de sortie dans **/etc/rsyslog.conf**, veuillez utiliser la syntaxe suivante :

```
$outchannel NAME, FILE_NAME, MAX_SIZE, ACTION
```

où :

- L'attribut *NAME* spécifie le nom du canal de sortie.
- L'attribut *FILE_NAME* spécifie le nom du fichier de sortie. Les canaux de sortie peuvent uniquement écrire dans des fichiers, et non dans les tubes, dans un terminal ou dans tout autre type de sortie.
- L'attribut *MAX_SIZE* représente la taille maximale que le fichier spécifié (dans *FILE_NAME*) peut faire. Cette valeur est spécifiée en *octets*.

- L'attribut *ACTION* spécifie l'action qui est prise lorsque la taille maximale, définie dans *MAX_SIZE*, est atteinte.

Pour utiliser la canal de sortie défini comme action à l'intérieur d'une règle, veuillez saisir :

```
FILTER :omfile:$NAME
```

Exemple 20.5. Rotation du journal d'un canal de sortie

La sortie suivante affiche une simple rotation de journal à travers l'utilisation d'un canal de sortie. Avant tout, le canal est défini via la directive ***$outchannel*** :

```
 $outchannel log_rotation, /var/log/test_log.log, 104857600,
/home/joe/log_rotation_script
```

puis, il est utilisé dans une règle qui sélectionne chaque message syslog avec une priorité, et exécute le canal de sortie défini au préalable sur les messages syslog acquis :

```
*.* :omfile:$log_rotation
```

Une fois la limite atteinte (dans l'exemple **100 Mo**), **/home/joe/log_rotation_script** est exécuté. Ce script peut contenir une instruction souhaitée, que ce soit de déplacer le fichier dans un autre répertoire, modifier un contenu particulier, ou simplement le supprimer.

Envoyer des messages syslog à des utilisateurs particuliers

rsyslog peut envoyer des messages syslog à des utilisateurs particuliers en spécifiant le nom d'utilisateur de l'utilisateur auquel vous souhaitez envoyer les messages (comme décrit dans l'Exemple 20.7, « Spécifier de multiples actions »). Pour spécifier plus d'un utilisateur, séparez chaque nom d'utilisateur par une virgule (**,**). Pour envoyer des messages à chaque utilisateur actuellement connecté, veuillez ajouter un astérisque (*****).

Exécuter un programme

rsyslog permet d'exécuter un programme pour les messages syslog sélectionnés et utilise l'appel **system()** pour exécuter le programme dans un shell. Pour spécifier l'exécution d'un programme, ajoutez un préfixe avec le caractère caret (**^**). Par conséquent, spécifiez un modèle qui formate le message reçu et le passe à l'exécutable spécifié en tant que paramètre sur une seule ligne (pour obtenir davantage d'informations sur les modèles, veuillez consulter la Section 20.2.3, « Modèles »).

```
FILTER ^EXECUTABLE; TEMPLATE
```

Ici, une sortie de la condition *FILTER* est traitée par un programme représenté par *EXECUTABLE*. Ce programme peut être n'importe quel exécutable valide. Remplacez *TEMPLATE* par le nom du modèle de formatage.

Exemple 20.6. Exécuter un programme

Dans l'exemple suivante, tout message syslog avec une priorité est sélectionné, formaté avec le modèle ***template*** et passé en tant que paramètre sur le programme **test-program**, qui est ensuite exécuté avec le paramètre fourni :

```
*.* ^test-program;template
```

AVERTISSEMENT

Lorsque vous acceptez des messages d'un hôte, et que vous utilisez l'action « shell execute », vous pouvez être vulnérable à une injection de commande. Un attaquant peut tenter d'injecter et d'exécuter des commandes dans le programme dont vous avez spécifié l'exécution dans votre action. Pour éviter toute menace de sécurité possible, considérez bien l'utilisation de l'action « shell execute ».

Stocker les messages syslog dans une base de données

Les messages syslog sélectionnés peuvent être écrits directement dans une table de base de données en utilisant l'action d'écriture sur base de données « *database writer* ». L'écriture sur base de données utilise la syntaxe suivante :

```
:PLUGIN:DB_HOST,DB_NAME,DB_USER,DB_PASSWORD;[TEMPLATE]
```

où :

- *PLUGIN* appelle le greffon spécifié qui gère l'écriture sur la base de données (par exemple, le greffon **ommysql**).
- L'attribut *DB_HOST* spécifie le nom d'hôte de la base de données.
- L'attribut *DB_NAME* spécifie le nom de la base de données.
- L'attribut *DB_USER* spécifie l'utilisateur de la base de données.
- L'attribut *DB_PASSWORD* spécifie le mot de passe utilisé avec l'utilisateur de la base de données susmentionné.
- L'attribut *TEMPLATE* spécifie une utilisation optionnelle d'un modèle qui modifie le message syslog. Pour obtenir davantage d'informations sur les modèles, veuillez consulter la Section 20.2.3, « Modèles ».

IMPORTANT

Actuellement, **rsyslog** fournit uniquement la prise en charge des bases de données **MySQL** et **PostgreSQL**. Pour utiliser les fonctionnalités d'écriture sur base de données de **MySQL** et **PostgreSQL**, veuillez installer les paquets rsyslog-mysql et rsyslog-pgsql, respectivement. Veuillez également vous assurez de charger les modules appropriés dans votre fichier de configuration **/etc/rsyslog.conf** :

```
$ModLoad ommysql    # Output module for MySQL support
$ModLoad ompgsql    # Output module for PostgreSQL support
```

Pour obtenir davantage d'informations sur les modules **rsyslog**, veuillez consulter la Section 20.6, « Utiliser des modules Rsyslog ».

Alternativement, vous pouvez utiliser une interface de base de données générique fournie par le module **omlibdb** (prise en charge : Firebird/Interbase, MS SQL, Sybase, SQLLite, Ingres, Oracle, mSQL).

Abandonner des messages syslog

Pour abandonner les messages sélectionnés, veuillez utiliser le caractère tilde (**~**).

```
FILTER ~
```

L'action « discard » est principalement utilisée pour filtrer les messages avant de continuer à les traiter. Celle-ci peut être efficace si vous souhaitez omettre certains message répétitifs qui auraient rempli les fichiers journaux. Les résultats de l'action discard dépendent de l'endroit où elle se trouve dans le fichier de configuration. Pour de meilleurs résultats, veuillez placer ces actions tout en haut de la liste des actions. Veuillez remarquer qu'une fois qu'un message a été abandonné, il est impossible de le récupérer dans les lignes suivantes du fichier de configuration.

Par exemple, la règle suivante abandonne tout message syslog cron :

```
cron.* ~
```

Spécifier de multiples actions

Pour chaque sélecteur, vous êtes autorisé à spécifier de multiples actions. Pour spécifier de multiples actions pour un sélecteur, écrivez chaque action sur un ligne différente et faites-la précéder d'un caractère perluète (&) :

```
FILTER ACTION
& ACTION
& ACTION
```

Spécifier de multiples actions améliore les performances générales du résultat souhaité car le sélecteur spécifié ne doit être évalué qu'une seule fois.

Exemple 20.7. Spécifier de multiples actions

Dans l'exemple suivant, tous les messages syslog du noyau avec la priorité critique (***crit***) sont envoyés à l'utilisateur **user1**, traités par le modèle ***temp*** et passés sur l'exécutable ***test-program***, puis transférés sur **192.168.0.1** via le protocole **UDP**.

```
kern.=crit user1
& ^test-program;temp
& @192.168.0.1
```

Toute action peut être suivie d'un modèle qui formate le message. Pour spécifier un modèle, ajoutez un suffixe à l'action avec un point-virgule (`;`) et spécifiez le nom du modèle. Pour obtenir davantage d'informations sur les modèles, veuillez consulter la Section 20.2.3, « Modèles ».

AVERTISSEMENT

Un modèle doit être défini avant d'être utilisé dans une action, sinon il sera ignoré. Autrement dit, les définitions de modèles doivent toujours précéder les définitions de règles dans **`/etc/rsyslog.conf`**.

20.2.3. Modèles

Toute sortie générée par **rsyslog** peut être modifiée et formatée selon vos besoins en utilisant des *modèles*. Pour créer un modèle, veuillez utiliser la syntaxe suivante dans **`/etc/rsyslog.conf`** :

```
$template TEMPLATE_NAME,"text %PROPERTY% more text", [OPTION]
```

où :

- ***`$template`*** est la directive du modèle indiquant que le texte la suivant définit un modèle.
- ***`TEMPLATE_NAME`*** est le nom du modèle. Utilisez ce nom pour faire référence au modèle.
- Tout ce qui se trouve entre deux guillemets (**`"`**…**`"`**) fait partie du texte du modèle. Avec ce texte, des caractères spéciaux, tels que **`\n`** pour une nouvelle ligne, ou **`\r`** pour un retour chariot, peuvent être utilisés. d'autres caractères, tels que **`%`** ou **`"`**, doivent être échappés si vous souhaitez utiliser ces caractères de manière litérale.
- Tout texte se trouvant entre deux caractères de pourcentage (**`%`**) indique une *propriété* vous permettant d'accéder au contenu particulier d'un message syslog. Pour obtenir davantage d'informations sur les propriétés, veuillez consulter la section intitulée « Propriétés ».
- L'attribut ***`OPTION`*** spécifie une option qui modifie la fonctionnalité du modèle. Les options de modèle actuellement prises en charge incluent ***`sql`*** et ***`stdsql`***, qui sont utilisées pour formater le texte en tant que requête SQL.

NOTE

Remarquez que le programme d'écriture sur base de données vérifie si les options ***sql*** ou ***stdsql*** sont spécifiées dans le modèle. Si ce n'est pas le cas, le programme d'écriture sur base de données n'effectuera aucune action. Cela permet d'empêcher toute menace de sécurité, comme les injections SQL.

Veuillez consulter la section *Stocker des messages syslog dans une base de donneés* dans Section 20.2.2, « Actions » pour obtenir davantage d'informations.

Générer des noms de fichiers dynamiques

Les modèles peuvent être utilisés pour générer des noms de fichiers dynamiques. En spécifiant une propriété faisant partie de ce chemin de fichier, un nouveau fichier sera créé pour chaque propriété unique, ce qui est une manière pratique de classifier les messages syslog.

Par exemple, utilisez la propriété ***timegenerated***, qui extrait un horodatage du message pour générer un nom de fichier unique pour chaque message syslog :

```
$template DynamicFile,"/var/log/test_logs/%timegenerated%-test.log"
```

N'oubliez pas que la directive ***$template*** indique uniquement le modèle. Vous devez l'utiliser dans une règle pour qu'elle puisse entrer en vigueur. Dans le fichier **/etc/rsyslog.conf**, utilisez le point d'interrogation (**?**) dans une définition d'action pour marquer le modèle du nom de fichier dynamique :

```
*.* ?DynamicFile
```

Propriétés

Les propriétés définies dans un modèle (entre deux caractères de pourcentage (**%**) permettent l'accès à divers contenus d'un message syslog par l'utilisation d'un remplaçant de propriété (*property replacer*). Pour définir une propriété située dans un modèle (entre les deux guillemets (**"…"**), veuillez utiliser la syntaxe suivante :

```
%PROPERTY_NAME[:FROM_CHAR:TO_CHAR:OPTION]%
```

où :

- L'attribut *PROPERTY_NAME* spécifie le nom de la propriété. Une liste de toutes les propriétés disponibles et leur description détaillée peut être trouvée dans la page man de **rsyslog.conf(5)** sous la section *Propriétés disponibles*.
- Les attributs *FROM_CHAR* et *TO_CHAR* dénotent une gamme de caractères qui déclencheront une réaction de la part de la propriété spécifiée. Alternativement, des expressions régulières peuvent être utilisées pour spécifier une gamme de caractères. Pour effectuer cela, paramétrez la lettre **R** comme attribut *FROM_CHAR* et spécifiez l'expression régulière souhaitée comme attribut *TO_CHAR*.
- L'attribut *OPTION* spécifie toute option de propriété, comme l'option **lowercase**, qui permet de convertir l'entrée en minuscules. Une liste de toutes les options des propriétés disponibles et leur description détaillée se trouve dans la page man **rsyslog.conf(5)**, sous la section *Options des propriétés*.

Ci-dessous figurent quelques exemples de propriétés simples :

- La propriété ci-dessous obtient le texte du message complet d'un message syslog :

```
%msg%
```

- La propriété ci-dessous obtient les deux premiers caractères du texte d'un message syslog :

```
%msg:1:2%
```

- La propriété ci-dessous obtient le texte du message complet d'un message syslog et abandonne le dernier caractère de saut de ligne :

```
%msg:::drop-last-lf%
```

- La propriété suivante obtient les 10 premiers caractères de l'horodatage généré lorsque le message syslog est reçu et le formate en fonction du standard de date *RFC 3999*.

```
%timegenerated:1:10:date-rfc3339%
```

Exemples de modèles

Cette section présente quelques exemples de modèles **rsyslog**.

L'Exemple 20.8, « Un modèle de message syslog détaillé » affiche un modèle qui formate un message syslog de manière à inclure dans sa sortie la sévérité du message, son type, l'horodatage du moment auquel il a été reçu, le nom de l'hôte, la balise du message, son texte, le tout se terminant par une nouvelle ligne.

Exemple 20.8. Un modèle de message syslog détaillé

```
$template verbose, "%syslogseverity%, %syslogfacility%, %timegenerated%,
%HOSTNAME%, %syslogtag%, %msg%\n"
```

L'Exemple 20.9, « Modèle de message « wall » » affiche un modèle qui ressemble à un message « wall » traditionnel (un message envoyé à tous les utilisateurs connectés dont la permission ***mesg(1)*** est paramétrée sur ***yes***). Ce modèle inclut le texte du message dans sa sortie, ainsi qu'un nom d'hôte, une balise de message et un horodatage sur une nouvelle ligne (en utilisant **\r** et **\n**, et émet un son (en utilisant **\7**).

Exemple 20.9. Modèle de message « wall »

```
$template wallmsg,"\r\n\7Message from syslogd@%HOSTNAME% at
%timegenerated% ...\r\n %syslogtag% %msg%\n\r"
```

L'Exemple 20.10, « Modèle de message formaté base de données » affiche un modèle qui formate un message syslog de manière à pouvoir l'utiliser en tant que requête de base de données. Remarquez l'utilisation de l'option ***sql*** à la fin du modèle spécifié en tant qu'option de modèle. Elle indique au programme d'écriture sur base de données de formater le message en tant que requête **SQL** MySQL.

Exemple 20.10. Modèle de message formaté base de données

```
$template dbFormat,"insert into SystemEvents (Message, Facility,
FromHost, Priority, DeviceReportedTime, ReceivedAt, InfoUnitID,
```

```
SysLogTag) values ('%msg%', %syslogfacility%, '%HOSTNAME%',
%syslogpriority%, '%timereported:::date-mysql%', '%timegenerated:::date-
mysql%', %iut%, '%syslogtag%')", sql
```

rsyslog contient également un ensemble de modèles prédéfinis identifiés par le préfixe **RSYSLOG_**. Ceux-ci sont réservés à l'utilisation de syslog et il est recommandé de ne pas créer de modèle en utilisant ce préfixe afin d'éviter les conflits. Ci-dessous figure une liste de ces modèles prédéfinis ainsi que leurs définitions.

RSYSLOG_DebugFormat

Format spécial utilisé pour résoudre les problèmes de propriétés.

```
"Debug line with all properties:\nFROMHOST: '%FROMHOST%', fromhost-ip:
'%fromhost-ip%', HOSTNAME: '%HOSTNAME%', PRI: %PRI%,\nsyslogtag
'%syslogtag%', programname: '%programname%', APP-NAME: '%APP-NAME%',
PROCID: '%PROCID%', MSGID: '%MSGID%',\nTIMESTAMP: '%TIMESTAMP%',
STRUCTURED-DATA: '%STRUCTURED-DATA%',\nmsg: '%msg%'\nescaped msg:
'%msg:::drop-cc%'\nrawmsg: '%rawmsg%'\n\n\"
```

RSYSLOG_SyslogProtocol23Format

Format spécifié sur IETF internet-draft ietf-syslog-protocol-23, qui est supposé devenir le nouveau standard RFC de syslog.

```
"%PRI%1 %TIMESTAMP:::date-rfc3339% %HOSTNAME% %APP-NAME% %PROCID%
%MSGID% %STRUCTURED-DATA% %msg%\n\"
```

RSYSLOG_FileFormat

Format logfile moderne similaire à TraditionalFileFormat, mais avec un horodatage de haute précision et des informations sur les fuseaux horaires.

```
"%TIMESTAMP:::date-rfc3339% %HOSTNAME% %syslogtag%%msg:::sp-if-no-1st-
sp%%msg:::drop-last-lf%\n\"
```

RSYSLOG_TraditionalFileFormat

Ancien format de fichier journal par défaut avec un horodatage peu précis.

```
"%TIMESTAMP% %HOSTNAME% %syslogtag%%msg:::sp-if-no-1st-sp%%msg:::drop-
last-lf%\n\"
```

RSYSLOG_ForwardFormat

Format de transfert avec horodatage de haute précision et informations sur les fuseaux horaires.

```
"%PRI%%TIMESTAMP:::date-rfc3339% %HOSTNAME% %syslogtag:1:32%%msg:::sp-
if-no-1st-sp%%msg%\"
```

RSYSLOG_TraditionalForwardFormat

Format de transfert traditionnel avec un horodatage peu précis.

```
"%PRI%%TIMESTAMP% %HOSTNAME% %syslogtag:1:32%%msg:::sp-if-no-1st-
sp%%msg%\"
```

20.2.4. Directives globales

Les directives globales sont des options de configuration qui s'appliquent au démon **rsyslogd**. Elles indiquent habituellement la valeur d'une variable prédéfinie qui affecte le comportement du démon **rsyslogd** ou une règle qui suit. Toutes les directives globales doivent commencer par le caractère du dollar (**$**). Une seule directive peut être spécifiée sur chaque ligne. Ci-dessous figure un exemple de directive globale qui spécifie la taille maximale de la file d'attente des messages syslog :

```
$MainMsgQueueSize 50000
```

La taille par défaut définie pour cette directive (**10,000** messages) peut être outrepassée en spécifiant une valeur différente (comme montré dans l'exemple ci-dessus).

Vous pouvez définir plusieurs directives dans le fichier de configuration **/etc/rsyslog.conf**. Une directive affecte le comportement de toutes les options de configuration jusqu'à ce qu'une autre occurrence de cette même directive soit détectée. Les directives globales peuvent être utilisées pour configurer des actions, des files d'attente, et pour le débogage. Une liste complète des toutes les directives de configuration disponibles se trouve dans la section intitulée « Documentation en ligne ». Actuellement, un nouveau format de configuration a été développé pour remplacer la syntaxe basée $ (veuillez consulter la Section 20.3, « Utiliser le nouveau format de configuration »). Cependant, les directives globales classiques sont toujours prises en charge comme format hérité.

20.2.5. Rotation de journaux

Ci-dessous figure un exemple du fichier de configuration **/etc/logrotate.conf** :

```
# rotate log files weekly
weekly
# keep 4 weeks worth of backlogs
rotate 4
# uncomment this if you want your log files compressed
compress
```

Toutes les lignes dans l'exemple de fichier de configuration définissent des options globales qui s'appliquent à tous les fichiers journaux. Dans cet exemple, les fichiers journaux sont mis en rotation chaque semaine, les fichiers journaux en rotation sont gardés pendant quatre semaines, et tous les fichiers journaux rotatifs sont compressés par **gzip** sous le format **.gz**. Toute ligne commençant par le caractère dièse (#) est un commentaire et ne sera donc pas traité.

Vous pouvez définir les options de configuration d'un fichier journal spécifique et le placer sous les options globales. Cependant, il est recommandé de créer un fichier de configuration séparé pour tout fichier journal spécifique dans le répertoire **/etc/logrotate.d/** et d'y définir les options de configuration souhaitées.

Ci-dessous figure un exemple de fichier de configuration placé dans le répertoire **/etc/logrotate.d/** :

```
/var/log/messages {
    rotate 5
    weekly
```

```
    postrotate
    /usr/bin/killall -HUP syslogd
    endscript
}
```

Les options de configuration dans ce fichier sont spécifiques au fichier journal **/var/log/messages**. Les paramètres spécifiés ici outrepassent les paramètres globaux lorsque possible. Ainsi, le fichier journal rotatif **/var/log/messages** sera conservé pendant cinq semaines au lieu des quatre semaines définies dans les options globales.

Ci-dessous figure une liste de quelques directives que vous pouvez spécifier dans le fichier de configuration **logrotate** :

- ***weekly*** — spécifie la rotation hebdomadaire des fichiers journaux à effectuer. Les directives similaires incluent :
  - ***daily***
  - ***monthly***
  - ***yearly***
- ***compress*** — active la compression des fichiers journaux dont la rotation a été effectuée. Les directives similaires incluent :
  - ***nocompress***
  - ***compresscmd*** — spécifie la commande à utiliser pour la compression.
  - ***uncompresscmd***
  - ***compressext*** — spécifie quelle extension doit être utilisée pour la compression.
  - ***compressoptions*** — permet de spécifier toute option pouvant être passée au programme de compression utilisé.
  - ***delaycompress*** — reporte la compression des fichiers journaux jusqu'à la prochaine rotation des fichiers journaux.
- ***rotate INTEGER*** — spécifie le nombre de rotations qu'un fichier journal effectue avant d'être supprimé ou envoyé à une adresse spécifique. Si la valeur **0** est spécifiée, les anciens fichiers journaux sont supprimés au lieu d'être mis en rotation.
- ***mail ADDRESS*** — cette option active l'envoi vers une adresse spécifiée pour les fichiers journaux qui ont été mis en rotation le nombre de fois défini par la directive ***rotate***. Directives similaires :
  - ***nomail***
  - ***mailfirst*** — indique que seuls les fichiers journaux qui viennent d'effectuer une rotation doivent être envoyés, au lieu d'envoyer les fichiers journaux sur le point d'expirer.
  - ***maillast*** — spécifie que les fichiers journaux sur le point d'expirer doivent être envoyés, au lieu d'envoyer les fichiers journaux qui viennent d'effectuer une rotation. Cette option est utilisée par défaut lorsque ***mail*** est activé.

Pour obtenir la liste complète des directives et des diverses options de configuration, veuillez consulter la page man de **logrotate(5)**.

20.3. UTILISER LE NOUVEAU FORMAT DE CONFIGURATION

Sur la version 7 de **rsyslog**, installée par défaut pour Red Hat Enterprise Linux 7 dans le paquet rsyslog, une nouvelle syntaxe de configuration est présentée. Ce nouveau format de configuration vise à être plus puissant, plus intuitif, et à empêcher que des erreurs communes se produisent en interdisant certained constructions invalides. Les améliorations de la syntaxe sont autorisées par le nouveau processeur de configuration qui repose sur RainerScript. Le format hérité est toujours pris en charge et est utilisé par défaut dans le fichier de configuration **/etc/rsyslog.conf**.

RainerScript est un langage de script conçu pour traiter des événements réseau et pour configurer des processeurs d'événements tels que **rsyslog**. RainerScript était tout d'abord utilisé pour définir des filtres basés sur expressions, veuillez consulter l'Exemple 20.3, « Filtres basés Expression ». La version RainerScript en rsyslog version 7 implémente les déclarations **input()** et **ruleset()**, qui permettent au fichier de configuration **/etc/rsyslog.conf** d'être écrit sous le nouveau style uniquement. La différence avec la nouvelle syntaxe, c'est qu'elle est plus structurée ; les paramètres sont passés sous forme d'arguments aux déclarations, telles que les entrées, les actions, les modèles et les chargements de modules. L'étendue des options est réduite en blocs. Cela améliore la lisibilité et diminue le nombre de bogues causés par les problèmes de configuration. Il y a également une amélioration de performance importante. Certaines fonctionnalités sont dans les deux syntaxes, certaines dans une seule.

Vous allez pouvoir effectuer une comparaison avec la configuration écrite avec les paramètres utilisant le style hérité :

```
$InputFileName /tmp/inputfile
$InputFileTag tag1:
$InputFileStateFile inputfile-state
$InputRunFileMonitor
```

et la même configuration utilisant le nouveau format de déclaration :

```
input(type="imfile" file="/tmp/inputfile" tag="tag1:"
statefile="inputfile-state")
```

Cela réduit de manière significative le nombre de paramètres utilisés dans la configuration, améliore la lisibilité, et offre également une vitesse d'exécution plus rapide. Pour obtenir davantage d'informations sur les déclarations et paramètres RainerScript, veuillez consulter la section intitulée « Documentation en ligne ».

20.3.1. Rulesets

Laissant les directives spéciales de côté, **rsyslog** gère les messages comme défini par des *règles* consistant d'une condition de filtre et d'une action à effectuer si la condition est « true ». Avec un fichier **/etc/rsyslog.conf** écrit de manière traditionnelle, toutes les règles sont évaluées dans leur ordre d'apparition pour chaque message d'entrée. Ce processus est lancé avec la première règle et continue jusqu'à ce que toutes les règles soient traitées ou jusqu'à ce que le message soit rejeté par l'une des règles.

Cependant les règles peuvent être regroupées en séquences appellées des *rulesets*. Avec des rulesets, il est possible de limiter l'effet de certaines règles uniquement à certaines entrées sélectionnées ou d'améliorer les performances de **rsyslog** en définissant un ensemble d'actions limitées à une entrée

spécifique. Autrement dit, les conditions du filtre qui seront inévitablement évaluées comme fausses pour certains types de messages peuvent être ignorées. La définition du ruleset dans **/etc/rsyslog.conf** ressemblera à ceci :

```
$RuleSet rulesetname
rule
rule2
```

La règle se termine dès qu'une autre règle est définie, ou bien, quand le ruleset par défaut est actionné ainsi :

```
$RuleSet RSYSLOG_DefaultRuleset
```

Avec le nouveau format de configuration de rsyslog 7, les déclarations **input()** et **ruleset()** sont réservées à cette opération. Le nouveau format de la définition du ruleset de **/etc/rsyslog.conf** ressemble à ceci :

```
ruleset(name="rulesetname") {
      rule
      rule2
      call rulesetname2
      …
}
```

Remplacez *rulesetname* par un identifiant pour le ruleset. Le nom du ruleset ne peut pas commencer par **RSYSLOG_** car l'utilisation de cet espace nom est réservé à **rsyslog**. **RSYSLOG_DefaultRuleset** définit ensuite un ensemble de règles à effectuer si aucun autre ruleset n'est assigné au message. Avec *rule* et *rule2*, vous pouvez définir les règles sous le format de filtrage d'actions mentionné ci-dessus. Avec le paramètre ***call***, vous pouvez imbriquer des rulesets en les appelant à partir d'autres blocs de rulesets.

Après avoir créé un ruleset, vous devrez spécifier l'entrée à laquelle celui-ci sera appliqué :

```
input(type="input_type" port="port_num" ruleset="rulesetname");
```

Il est possible d'identifier ici un message d'entrée par *input_type*, qui est un module d'entrée collectant le message, ou par *port_num* – le numéro du port. D'autres paramètres, tels que *file* ou *tag* peuvent être spécifiés pour **input()**. Remplacez *rulesetname* par le nom du ruleset devant être évalué avec le message. Au cas où un message d'entrée n'est pas explicitement limité à un ruleset, le ruleset par défaut est déclenché.

Il est également possible d'utiliser le format hérité pour définir des rulesets. Pour obtenir davantage d'information, veuillez consulter la section intitulée « Documentation en ligne ».

Exemple 20.11. Utiliser des rulesets

Les rulesets suivants assurent une gestion différente des messages distants en provenance de différents ports. Ajoutez ce qui suit dans **/etc/rsyslog.conf** :

```
ruleset(name="remote-6514") {
    action(type="omfile" file="/var/log/remote-6514")
}

ruleset(name="remote-601") {
```

```
    cron.* action(type="omfile" file="/var/log/remote-601-cron")
    mail.* action(type="omfile" file="/var/log/remote-601-mail")
}

input(type="imtcp" port="6514" ruleset="remote-6514");
input(type="imtcp" port="601" ruleset="remote-601");
```

Les rulesets affichés dans l'exemple ci-dessus définissent les destinations des journaux pour l'entrée distante de deux ports. Dans le cas du port **601**, les messages sont triés selon la fonctionnalité. Puis, l'entrée TCP est activée et elle est limitée aux rulesets. Remarquez que vous devez charger les modules requis (imtcp) pour que cette configuration fonctionne.

20.3.2. Compatibilité avec sysklogd

Le mode de compatibilité spécifié via l'option **-c** existe en **rsyslog** version 5, mais pas en version 7. Les options de ligne de commande de style sysklogd sont dépréciées et la configuration de **rsyslog** à l'aide de ces options de ligne de commande doit être évitée. Cependant, vous pouvez utiliser plusieurs modèles et directives pour configurer **rsyslogd** afin d'émuler un comportement similaire à sysklogd.

Pour obtenir davantage d'informations sur les diverses options **rsyslogd**, veuillez consulter la page man de **rsyslogd(8)**.

20.4. UTILISER DES FILES D'ATTENTE DANS RSYSLOG

Les files sont utilisées pour transférer des contenus, principalement des messages syslog, entre composants de **rsyslog**. Avec les files, rsyslog est capable de traiter de multiples messages simultanément et d'appliquer plusieurs actions à un seul message à la fois. Le flux de données dans **rsyslog** peut être illustré comme ceci :



Figure 20.1. Flux de messages dans Rsyslog

Lorsque **rsyslog** reçoit un message, celui-ci est transféré vers le pré-processeur, puis il est placé dans la file des messages principaux (« *main message queue* »). Les messages attendent ainsi d'être retirés de cette file et transférés vers le processeur de règles (« *rule processor* »).

Le processeur de règles (« *rule processor* ») est un moteur d'analyse syntaxique et de filtrage. Ici, les règles définies dans **`/etc/rsyslog.conf`** sont appliquées. Basé sur ces règles, le processeur de règles évalue quelles sont les actions devant être effectuées. Chaque action possède sa propre file d'attente d'actions. Les messages sont transférés à travers cette file vers le processeur d'action respectif, qui créera la sortie finale. Remarquez qu'à ce moment, plusieurs actions peuvent être exécutées simultanément sur un seul message. Dans ce but précis, un message est dupliqué et transféré vers de multiples processeurs d'actions.

Seule une file d'attente par action est possible. Selon la configuration, les messages peuvent être envoyés directement vers le processeur d'actions sans file d'attente d'actions. Ce comportement est celui des *files directes* (veuillez voir ci-dessous). Au cas où l'action sortie échoue, le processeur d'actions notifie la file d'actions, qui reprendra ensuite un élément non traité, et après un certain temps, l'action sera tentée à nouveau.

Pour résumer, il existe deux positions dans lesquelles les files sont utilisées dans **rsyslog**: soit en face du processeur de règles en tant que *file principale de messages* unique, ou en face de divers types d'actions de sortie en tant que *files d'action*. Les files fournissent deux avantages principaux, menant tous deux à une amélioration des performances du traitement des messages :

- elles servent de tampon, *découplant* les producteurs et consommateurs dans la structure de **rsyslog**
- elles permettent la *parallélisation* des actions effectués sur des messages

À part cela, les files peuvent être configurées avec plusieurs directives pour fournir des performances optimales à votre système. Il est question de ces options de configuration dans les sections suivantes.

AVERTISSEMENT

Si un greffon de sortie est incapable de remettre un message, il est stocké dans la file de messages précédente. Si la file est remplie, les entrées sont bloquées jusqu'à ce qu'elle ne soit plus remplie. Cela empêchera les nouveaux messages d'être journalisés via la file bloquée. En l'absence de files d'actions séparées, ceci peut avoir de sérieuses conséquences, comme la prévention de la journalisation **SSH**, ce qui peut ensuite interdire l'accès **SSH**. Il est donc recommandé d'utiliser des files d'actions dédiées pour les sorties transférées à travers un réseau ou une base de données.

20.4.1. Définir des files d'attente

Basé sur l'emplacement de stockage des messages, il existe plusieurs types de files d'attente : les files les plus couramment utilisées sont nommées *direct* (file directe), *in-memory* (file en mémoire), *disk* (file de disque), et *disk-assisted in-memory* (file en mémoire assistée par disques). Il est possible de choisir l'un de ces types comme file principale et pour les files d'actions. Ajoutez ce qui suit au fichier **`/etc/rsyslog.conf`** :

```
$objectQueueType queue_type
```

-

Ici, vous pouvez appliquer le paramètre de la file principale des messages (remplacez *object* par **MainMsg**) ou pour une file d'actions (remplacez *object* par **Action**). Remplacez *queue_type* par **direct**, **linkedlist** ou **fixedarray** (qui sont des files en mémoire), ou par **disk**.

Le paramètre par défaut pour une file de message principale est la file FixedArray avec une limite de 10,000 messages. Les files d'actions sont définies par défaut comme des files « Direct » (directes).

Files « Direct »

Pour de nombreuses opérations simples, comme lors de l'écriture de la sortie sur un fichier local, la création d'une file devant chaque action n'est pas nécessaire. Pour éviter la mise en file d'attente, veuillez utiliser :

```
$objectQueueType Direct
```

Remplacez *object* par **MainMsg** ou par **Action** pour utiliser cette option sur la file de messages principale ou pour une file d'actions, respectivement. Avec une file « Direct », les messages sont immédiatement et directement transférés du producteur au consommateur.

Files « Disk »

Les files « Disk » stockent les messages strictement sur un disque dur, ce qui les rend très fiables, mais ce mode est également le plus lent des modes de files. Ce mode peut être utilisé pour empêcher la perte de données de journaux extrêmement importants. Cependant, les files de disques ne sont pas recommandées dans la plupart des cas. Pour définir une file « Disk », veuillez saisir ce qui suit dans le fichier **/etc/rsyslog.conf** :

```
$objectQueueType Disk
```

Veuillez remplacer *object* par **MainMsg** ou par **Action** pour utiliser cette option pour la file de messages principale ou pour une file d'actions, respectivement. Les files « Disk » sont écrites sur différentes parties, avec une taille par défaut de 10 Mo. Cette taille par défaut peut être modifiée avec la directive de configuration suivante :

```
$objectQueueMaxFileSize size
```

où *size* représente la taille spécifiée de la partie de la file « Disk ». La limite de taille définie n'est pas restrictive, **rsyslog** écrit toujours une entrée de file complète, même si elle transgresse la limite de taille. Chaque partie d'une file de disque correspond à un fichier individuel. La directive du nom de ces fichiers est similaire à ceci :

```
$objectQueueFilename name
```

Cela définit un préfixe *name* pour le fichier, suivi d'un numéro à 7 chiffres commençant par le chiffre un et incrémenté pour chaque fichier.

Files « In-memory »

Avec les files en mémoire, les messages mis en file d'attente sont conservés dans la mémoire, ce qui rend le processus très rapide. Les données mises en file seront perdues si l'ordinateur est éteint ou redémarré. Cependant, il est possible d'utiliser le paramètre **$ActionQueueSaveOnShutdown** pour enregistrer les données avant la fermeture. Il existe deux types de files en mémoire :

- La file *FixedArray* — mode par défaut de la file de messages principale, avec une limite de 10,000 éléments. Ce type de file utilise une matrice fixée, pré-allouée qui contient des pointeurs vers des éléments de files d'attente. À cause de ces pointeurs, même si la file est vide, une

bonne quantité de mémoire est utilisée. Cependant, FixedArray offre des meilleures performances de temps d'activité et est optimal lorsque vous vous attendez à un nombre relativement faible de messages en file d'attente et à des performances élevées.

- File *LinkedList* — ici, toutes les structures sont allouées dynamiquement dans une liste liée. Ainsi, la mémoire est uniquement allouée lorsque nécessaire. Les files LinkedList gèrent également très bien les rafales occasionnelles de messages.

En général, veuillez utiliser les files LinkedList lorsque vous avez des doutes. Comparé à FixedArray, ces files consomment moins de mémoire et réduisent la charge de traitement.

Veuillez utiliser la syntaxe suivante pour configurer les files en mémoire :

```
$objectQueueType LinkedList
```

```
$objectQueueType FixedArray
```

Remplacez *object* par **MainMsg** ou par **Action** pour utiliser cette option sur la file de messages principale ou la file d'actions, respectivement.

Files « Disk-Assisted In-memory »

Les files de disque et en mémoire ont chacune leurs avantages, et **rsyslog** permet de les combiner en *files en mémoire assistées par disque*. Pour faire cela, veuillez configurer une file en mémoire normale, puis ajoutez la directive **$objectQueueFileName** pour définir un nom de fichier pour l'assistance disque. Cette file deviendra ainsi une file *assistée par disque*, ce qui signifie qu'une file en mémoire est combinée à une file de disque et celles-ci fonctionneront en tandem.

La file de disque est activée si la file en mémoire est pleine ou doit persister après une fermeture du système. Avec une file assistée par disque, il est possible de définir des paramètres de configuration spécifiques aux files de disque et spécifiques aux files en mémoire. Ce type de file est probablement le plus couramment utilisé, il est particulièrement utile pour les actions potentiellement longues et peu fiables.

Pour spécifier le fonctionnement d'une file en mémoire assistée par disque, veuillez utiliser les *filigranes* :

```
$objectQueueHighWatermark number
```

```
$objectQueueLowWatermark number
```

Veuillez remplacer *object* par **MainMsg** ou par **Action** pour utiliser cette option pour la file de messages principale ou pour une file d'actions, respectivement. Remplacez *number* par le nombre de messages mis en file d'attente. Lorsqu'une file en mémoire atteint le nombre défini par le filigrane du haut, les messages commencent à être écrits sur le disque et cela continue jusqu'à ce que la taille de la file en mémoire passe sous le nombre défini par le filigrane du bas. La définition correcte des filigranes minimise les écritures sur disque non nécessaires, mais cela laisse également de l'espace mémoire pour les rafales de messages, puisque l'écriture sur fichiers de disque est relativement longue. Ainsi, le filigrane du haut doit être plus bas que la totalité de la capacité de la file définie avec *$objectQueueSize*. La différence entre le filigrane du haut et la taille de la file en général est un tampon de mémoire supplémentaire réservé aux rafales de message. D'autre part, définir le filigrane du haut trop bas activera l'assistance par disque trop souvent.

Exemple 20.12. Transférer des messages journaux sur un serveur de manière fiable

Rsyslog est souvent utilisé pour maintenir un système de journalisation centralisé, où les messages sont transférés sur un serveur à travers le réseau. Pour éviter que des pertes de messages se produisent lorsque le serveur est indisponible, il est recommandé de configurer une file d'actions pour l'action de transfert. Ainsi, les messages dont l'envoi a échoué sont stockés localement jusqu'à ce que le serveur soit à nouveau joignable. Remarquez que de telles files ne sont pas configurables pour les connexions utilisant le protocole **UDP**.

Procédure 20.1. Effectuer un transfert sur un seul serveur

Supposez que la tâche consiste à transférer les messages de journalisation d'un système à un serveur avec le nom d'hôte *example.com*, et à configurer une file d'actions pour mettre les messages dans un tampon en cas de panne du serveur. Pour cela, veuillez effectuer les étapes suivantes :

- Veuillez utiliser la configuration suivante dans **/etc/rsyslog.conf** ou créez un fichier avec le contenu suivant dans le répertoire **/etc/rsyslog.d/** :

```
$ActionQueueType LinkedList
$ActionQueueFileName example_fwd
$ActionResumeRetryCount -1
$ActionQueueSaveOnShutdown on
*.*       @@example.com:6514
```

  Où :

  - **$ActionQueueType** active une file en mémoire LinkedList,
  - **$ActionFileName** définit un stockage sur disque. Dans ce cas, les fichiers de sauvegarde sont créés dans le répertoire **/var/lib/rsyslog/** avec le préfixe *example_fwd*,
  - le paramètre **$ActionResumeRetryCount -1** empêche rsyslog d'abandonner des messages lorsqu'il essaie de se connecter à nouveau si le serveur ne répond pas,
  - **$ActionQueueSaveOnShutdown** activé enregistre les données en mémoire si rsyslog s'éteint,
  - la dernière ligne transfère tous les messages reçus au serveur d'enregistrement, la spécification de port est optionnelle.

  Avec la configuration ci-dessus, rsyslog conserve les messages en mémoire si le serveur distant n'est pas joignable. Un fichier sur disque est uniquement créé si rsyslog ne possède plus d'espace configuré de file d'attente en mémoire ou s'il doit s'éteindre, ce qui profite aux performances système.

Procédure 20.2. Effectuer un transfert vers plusieurs serveurs

Le processus de transfert de messages de journalisation vers des serveurs multiples est similaire à la procédure précédente :

- Chaque serveur destinataire requiert une règle de transfert séparée, une spécification de file d'actions séparée, et un fichier de sauvegarde sur disque séparé. Par exemple, veuillez utiliser la configuration suivante dans **/etc/rsyslog.conf** ou créez un fichier avec le contenu suivant dans le répertoire **/etc/rsyslog.d/** :

```
$ActionQueueType LinkedList
$ActionQueueFileName example_fwd1
```

```
$ActionResumeRetryCount -1
$ActionQueueSaveOnShutdown on
*.*       @@example1.com

$ActionQueueType LinkedList
$ActionQueueFileName example_fwd2
$ActionResumeRetryCount -1
$ActionQueueSaveOnShutdown on
*.*       @@example2.com
```

20.4.2. Créez un nouveau répertoire pour les fichiers de journalisation rsyslog

Rsyslog exécute en tant que démon **syslogd** et est géré par SELinux. De ce fait, tous les fichiers dans lesquels rsyslog écrit doivent posséder le contexte de fichier SELinux qui convient.

Procédure 20.3. Création d'un nouveau répertoire de travail

1. Si vous avez besoin d'un répertoire différent pour stocker les fichiers de travail, créer un répertoire comme suit :

   ```
   ~]# mkdir /rsyslog
   ```

2. Installer les utilitaires requis pour la Stratégie SELinux :

   ```
   ~]# yum install policycoreutils-python
   ```

3. Définir le contexte de répertoire SELinux pour qu'il corresponde à celui du répertoire **/var/lib/rsyslog/** :

   ```
   ~]# semanage fcontext -a -t syslogd_var_lib_t /rsyslog
   ```

4. Appliquer le contexte SELinux :

   ```
   ~]# restorecon -R -v /rsyslog
   restorecon reset /rsyslog context
   unconfined_u:object_r:default_t:s0-
   >unconfined_u:object_r:syslogd_var_lib_t:s0
   ```

5. Si nécessaire, vérifiez le contexte SELinux comme suit :

   ```
   ~]# ls -Zd /rsyslog
   drwxr-xr-x. root root system_u:object_r:syslogd_var_lib_t:s0
   /rsyslog
   ```

6. Créer des sous-répertoires selon les besoins. Exemple :

   ```
   ~]# mkdir /rsyslog/work/
   ```

   Les sous-répertoires seront créés avec le même contexte SELinux que le répertoire parent.

7. Ajouter la ligne suivante dans **/etc/rsyslog.conf** juste avant qu'elle puisse entrer en vigueur :

```
$WorkDirectory /rsyslog/work
```

Ce paramétre demeurera valide jusqu'à ce que l'on rencontre une nouvelle directive de **WorkDirectory** lors de l'analyse des fichiers de configuration.

20.4.3. Gestion des files

Tous les types de files peuvent être configurés davantage, afin de correspondre à vos besoins. Il est possible d'utiliser plusieurs directives pour modifier les files d'actions et la file de messages principale. Actuellement, il existe plus de 20 paramètres de file disponibles, veuillez consulter la section intitulée « Documentation en ligne ». Certains de ces paramètres sont couramment utilisés, tandis que d'autres, comme la gestion de threads de travail (« worker thread management »), fournissent un contrôle plus précis du comportement des files et sont réservés aux utilisateurs avancés. Avec des paramètres avancés, il est possible d'optimiser les performances de **rsyslog**, de planifier la mise en file d'attente, ou de modifier le comportement d'une file pendant la fermeture système.

Limiter la taille de file d'attente

Vous pouvez limiter le nombre de messages contenus dans une file d'attente avec le paramètre suivant :

```
$objectQueueHighWatermark number
```

Remplacez *objet* par **MainMsg** ou par **Action** pour utiliser cette option pour la file de messages principale ou pour une file d'actions, respectivement. Remplacez *nombre* par le nombre de messages mis en file d'attente. Vous pouvez uniquement définir la taille de la file en tant que nombre de messages, et non en tant que taille de mémoire. La taille de la file par défaut s'élève à 10,000 messages pour la file de messages principale et les files ruleset, et à 1000 pour les files d'actions.

Les files assistées par disque sont illimitées par défaut et ne peuvent pas être limitées par cette directive, mais vous pouvez leur réserver de l'espace physique en octets avec les paramètres suivants :

```
$objectQueueMaxDiscSpace number
```

Remplacez *objet* par **MainMsg** ou par **Action**. Lorsque la limite de la taille spécifiée par le *nombre* est atteinte, les messages sont abandonnés jusqu'à ce que suffisamment d'espace soit libéré par les messages sortis de la file.

Abandonner des messages

Lorsqu'une file atteint un certain nombre de messages, vous pouvez abandonner les messages moins importants afin d'économiser de l'espace dans la file pour les entrées de plus haute priorité. La limite qui lance le processus d'abandon peut être définie avec l'indication *discard* :

```
$objectQueueDiscardMark number
```

Remplacez *objet* par **MainMsg** ou par **Action** pour utiliser cette option sur la file de messages principale ou sur une file d'actions, respectivement. Le *nombre* correspond au nombre de messages qui doit se trouver dans la file pour lancer le processus d'abandon. Pour définir quels messages abandonner, veuillez utiliser :

```
$objectQueueDiscardSeverity priority
```

Remplacez *priorité* par l'un des mots-clés suivants (ou par un numéro) : **debug** (7), **info** (6), **notice**

(5), **warning** (4), **err** (3), **crit** (2), **alert** (1), et **emerg** (0). Avec ce paramètre, les messages entrants et les messages se trouvant déjà en file d'attente avec une priorité plus basse que celle qui est définie, sont effacés de la file dès que la marque d'abandon aura été atteinte.

Utiliser des délais

Il est possible de configurer **rsyslog** pour traiter des files d'attente pendant une période spécifique. Par exemple, avec cette option, il est possible de transférer une partie du traitement pendant les heures creuses. Pour paramétrer un délai, veuillez utiliser la syntaxe suivante :

```
$objectQueueDequeueTimeBegin hour
```

```
$objectQueueDequeueTimeEnd hour
```

Avec *heure*, vous pouvez spécifier les heures qui limitent votre délai. Veuillez utiliser le format 24 heures sans les minutes.

Configurer les threads de travail

Un *thread de travail* effectue une action spécifiée sur le message mis en file d'attente. Par exemple, dans la file de messages principale, une tâche de travail consiste à appliquer une logique de filtrage sur chaque message puis de l'ajouter à la file d'action correspondante. Lorsqu'un message arrive, un thread de travail est lancé automatiquement. Lorsque le nombre de messages atteint un certain niveau, un autre thread de travail est allumé. Pour spécifier ce nombre, veuillez utiliser :

```
$objectQueueWorkerThreadMinimumMessages number
```

Remplacez *nombre* par le nombre de messages qui déclenchera un thread de travail supplémentaire. Par exemple, si le *nombre* est paramétré sur 100, un nouveau thread de travail sera lancé lorsque plus de 100 messages seront reçus. Lorsque plus de 200 seront reçus, un troisième thread de travail sera lancé, et ainsi de suite. Cependant, trop de threads de travail fonctionnant en parallèle deviendront inefficaces, vous pouvez donc limiter le nombre maximum en utilisant :

```
$objectQueueWorkerThreads number
```

où le *nombre* correspond au nombre maximal de threads de travail pouvant être exécutés en parallèle. Pour la file de messages principale, la limite par défaut est fixée à 1 thread. Une fois qu'un thread de travail a été lancé, il restera en cours d'exécution jusqu'à ce qu'un délai d'inactivité apparaisse. Pour définir la longueur du délai, veuillez saisir :

```
$objectQueueWorkerTimeoutThreadShutdown time
```

Remplacez *durée* par la durée en millisecondes. Sans ce paramètre, un délai d'expiration fixé à zéro sera appliqué, et un thread de travail sera terminé dès qu'il ne contiendra plus de messages. Si vous spécifiez *durée* avec **-1**, aucun thread ne sera fermé.

Retirer un lot de la file

Pour améliorer les performances, vous pouvez configurer **rsyslog** pour retirer de multiples messages de la file à la fois. Pour définir la limite supérieure de ce type de suppression de la file :

```
$objectQueueDequeueBatchSize number
```

Remplacez *nombre* par le nombre maximal de messages pouvant être supprimés de la file à la fois. Remarquez qu'un paramètre plus élevé combiné à un nombre plus élevé de threads de travail autorisés se traduit par une consommation de mémoire plus importante.

Terminer des files d'attente

Lorsqu'une file qui contient encore des messages est terminée, vous pouvez minimiser la perte de données en spécifiant un intervalle pour que les threads de travail puisse finir le traitement de la file :

```
$objectQueueTimeoutShutdown time
```

Spécifiez *time* en millisecondes. Si certains messages sont toujours en file d'attente après cette période, les threads de travail finiront l'élément de données actuel et se termineront. Des messages non traités seront donc perdus. Un autre intervalle peut également être paramétré pour que les threads de travail puissent traiter l'élément final :

```
$objectQueueTimeoutActionCompletion time
```

Au cas où le délai d'expiration se termine, tout thread de travail restant sera éteint. Pour enregistrer les données pendant la fermeture, veuillez utiliser :

```
$objectQueueTimeoutSaveOnShutdown time
```

Si définis, tous les éléments de la file seront enregistrés sur le disque avant que **rsyslog** ne se termine.

20.4.4. Utilisation d'une nouvelle syntaxe pour les files d'attente rsyslog

Dans la nouvelle syntaxe qui se trouve dans rsyslog 7, les files d'attente se trouvent dans l'objet **`action()`** qui peut être utilisé à la fois séparemment ou à l'intérieur d'un ruleset dans **`/etc/rsyslog.conf`**. Le format d'une file d'attente d'action ressemble à ce qui suit :

```
action(type="action_type" queue.size="queue_size" queue.type="queue_type"
queue.filename="file_name")
```

Remplacer *action_type* par le nom du module qui doit effectuer l'action et remplacer *queue_size* par le nombre maximum de messages que la file d'attente puisse contenir. Pour *queue_type*, sélectionner **`disk`** ou bien, parmi l'une des files d'attente en mémoire : **`direct`**, **`linkedlist`** ou **`fixedarray`**. Pour *file_name* ne spécifier qu'un nom de fichier, et non pas un chemin. Notez que si vous créez une nouveau répertoire contenant les fichiers de journalisation, le context SELinux devra être défini. Voir un exemple dans Section 20.4.2, « Créez un nouveau répertoire pour les fichiers de journalisation rsyslog ».

Exemple 20.13. Définir une File d'attente d'action

Pour configurer l'action de sortie avec une file d'attente d'actions asynchrones basées sur une liste chaînée, qui peut contenir un maximum de 10 000 messages, saisir la commande suivante :

```
action(type="omfile" queue.size="10000" queue.type="linkedlist"
queue.filename="logfile")
```

La syntaxe rsyslog 7 de files d'attentes d'actions directes ressemble à ceci :

```
*.* action(type="omfile" file="/var/lib/rsyslog/log_file
     )
```

La syntaxe rsyslog 7 d'une file d'attente ayant plusieurs paramètres peut s'écrire ainsi :

```
*.* action(type="omfile"
```

```
            queue.filename="log_file"
            queue.type="linkedlist"
            queue.size="10000"
    )
```

Le répertoire de travail par défaut, ou le dernier répertoire de travail à définir, sera utilisé. Si l'on doit utiliser un autre répertoire de travail, ajouter une ligne comme suit devant la file d'attente d'action :

```
global(workDirectory="/directory")
```

Exemple 20.14. Redirection vers un Serveur unique avec la nouvelle Syntaxe

L'exemple suivant est basé sur la procédure Procédure 20.1, « Effectuer un transfert sur un seul serveur » afin de montrer la différence entre la syntaxe traditionnelle et la syntaxe rsyslog 7. Le greffon **omfwd** est utilisé pour le transfert via **UDP** ou **TCP**. La valeur par défaut est **UDP**. Comme le greffon est intégré, il n'a pas besoin d'être téléchargé.

Veuillez utiliser la configuration suivante dans **/etc/rsyslog.conf** ou créez un fichier avec le contenu suivant dans le répertoire **/etc/rsyslog.d/** :

```
*.* action(type="omfwd"
      queue.type="linkedlist"
      queue.filename="example_fwd"
      action.resumeRetryCount="-1"
      queue.saveOnShutdown="on"
      target="example.com" port="6514" protocol="tcp"
     )
```

Où :

- **queue.type="linkedlist"** permet d'avoir une file d'attente LinkedList en mémoire,
- **queue.filename** définit un stockage sur disque. Les fichiers de sauvegarde sont créés avec le préfixe *example_fwd*, dans le répertoire de travail spécifié par la directive **workDirectory** globale qui précède,
- le paramètre **action.resumeRetryCount -1** empêche rsyslog d'abandonner des messages lorsqu'il essaie de se connecter à nouveau si le serveur ne répond pas,
- **queue.saveOnShutdown="on"** activé enregistre les données en mémoire si rsyslog s'éteint,
- la dernière ligne transfère tous les messages reçus au serveur d'enregistrement, la spécification de port est optionnelle.

20.5. CONFIGURER RSYSLOG SUR UN SERVEUR D'ENREGISTREMENT

Le service **rsyslog** fournit une installation pour exécuter un serveur d'enregistrement et pour configurer des systèmes individuels pour qu'ils envoient leurs fichiers journaux sur le serveur d'enregistrement. Veuillez consulter Exemple 20.12, « Transférer des messages journaux sur un serveur de manière fiable » pour obtenir des informations sur la configuration **rsyslog** du client.

Le service **rsyslog** doit être installé sur le système que vous comptiez utiliser en tant que serveur d'enregistrement et sur tous les systèmes qui seront configurés pour y envoyer leurs journaux. Rsyslog est installé par défaut sur Red Hat Enterprise Linux 7. Si requis, pour vous assurer que c'est bien le cas, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install rsyslog
```

Le protocole et le port par défaut du trafic syslog est **UDP** et **514**, comme indiqué dans le fichier **/etc/services**. Cependant, **rsyslog** utilise par défaut **TCP** sur le port **514**. Dans le fichier de configuration, **/etc/rsyslog.conf**, **TCP** est indiqué par @@.

D'autres ports sont parfois utilisés en exemples, mais SELinux n'est configuré que pour permettre d'envoyer ou de recevoir sur les ports par défaut suivants :

```
~]# semanage port -l | grep syslog
syslogd_port_t                 tcp      6514, 601
syslogd_port_t                 udp      514, 6514, 601
```

L'utilitaire **semanage** est fourni dans le cadre du paquet policycoreutils-python. Si nécessaire, installez le package comme suit :

```
~]# yum install policycoreutils-python
```

De plus, le type SELinux de **rsyslog**, **rsyslogd_t** par défaut est configuré de façon à permettre d'envoyer et de recevoir dans le port de shell distant (**rsh**) ayant comme type de SELinux **rsh_port_t**, avec par défaut **TCP** sur le port **514**. De ce fait, vous n'avez pas besoin d'utiliser **semanage** pour permettre explicitement **TCP** sur le port **514**. Ainsi, pour vérifier ce sur quoi SELinux est défini pour autoriser l'accès au port **514**, saisir la commande suivante :

```
~]# semanage port -l | grep 514sortie omise
rsh_port_t                     tcp      514
syslogd_port_t                 tcp      6514, 601
syslogd_port_t                 udp      514, 6514, 601
```

Pour obtenir plus d'informations sur SELinux, voir *Red Hat Enterprise Linux 7 SELinux User's and Administrator's Guide*.

Effectuez les étapes décrites dans cette procédure dans le système que vous comptez utiliser comme serveur de journalisation. Toutes les étapes de cette procédure doivent être effectuées en tant qu'utilisateur **root** :

Procédure 20.4. Configurez SELinux pour autoriser le trafic rsyslog sur un port

Si vous avez besoin d'utiliser un nouveau port pour le trafic **rsyslog**, suivez cette procédure sur le serveur de journalisation et sur les clients. Ainsi, pour recevoir et envoyer le trafic **TCP** sur le port **10514**, procédez ainsi :

1. ```
   ~]# semanage port -a -t syslogd_port_t -p tcp 10514
   ```
2. Vérifiez les ports SELinux en saisissant la commande suivante :

   ```
   ~]# semanage port -l | grep syslog
   ```
   ```

3. Si le nouveau port était déjà configuré dans **/etc/rsyslog.conf**, redémarrez **rsyslog** maintenant pour que le changement puisse avoir lieu :

```
~]# service rsyslog restart
```

4. Vérifiez sur quels ports **rsyslog** écoute :

```
~]# netstat -tnlp | grep rsyslog
tcp        0      0 0.0.0.0:10514           0.0.0.0:*   LISTEN
2528/rsyslogd
tcp        0      0 :::10514                ::::*       LISTEN
2528/rsyslogd
```

Voir la page man de **semanage-port(8)** pour obtenir plus d'informations sur la commande **semanage port**.

Procédure 20.5. Configurer firewalld

Configurer **firewalld** pour autoriser le trafic **rsyslog**. Ainsi, pour autoriser le trafic **TCP** sur le port **10514**, procédez ainsi :

1.
```
~]# firewall-cmd --zone=zone --add-port=10514/tcp
success
```

   Quand *zone* correspond à la zone d'interface à utiliser. Veuillez noter que ces changements ne persisteront pas lors d'un nouveau démarrage. Pour rendre les changements au parefeu permanents, répétez la commande an ajoutant l'option **--permanent**. Pour obtenir davantage d'informations sur l'ouverture et la fermeture des ports de **firewalld**, veuillez consulter le Guide de sécurité Red Hat Enterprise Linux 7 .

2. Pour vérifier les paramètres ci-dessus, veuillez utiliser une commande comme suit :

```
~]# firewall-cmd --list-all
public (default, active)
  interfaces: eth0
  sources:
  services: dhcpv6-client ssh
  ports: 10514/tcp
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:
```

Procédure 20.6. Configurer rsyslog pour Recevoir et Trier les Messages de journalisation distants

1. Ouvrez le fichier **/etc/rsyslog.conf** dans un éditeur de texte puis procédez comme suit :

   a. Ajoutez ces lignes sous la section des modules mais au-dessus de la section « **Provides UDP syslog reception** » (Fournit la réception syslog UDP) :

```
# Define templates before the rules that use them

### Per-Host Templates for Remote Systems ###
```

```
$template TmplAuthpriv,
"/var/log/remote/auth/%HOSTNAME%/%PROGRAMNAME:::secpath-
replace%.log"
$template TmplMsg,
"/var/log/remote/msg/%HOSTNAME%/%PROGRAMNAME:::secpath-
replace%.log"
```

b. Remplacez la section par défaut « **Provides TCP syslog reception** » par ce qui suit :

```
# Provides TCP syslog reception
$ModLoad imtcp
# Adding this ruleset to process remote messages
$RuleSet remote1
authpriv.*   ?TmplAuthpriv
*.info;mail.none;authpriv.none;cron.none   ?TmplMsg
$RuleSet RSYSLOG_DefaultRuleset   #End the rule set by switching
back to the default rule set
$InputTCPServerBindRuleset remote1  #Define a new input and bind
it to the "remote1" rule set
$InputTCPServerRun 10514
```

Enregistrez les changements sur le fichier **/etc/rsyslog.conf**.

2. Le service **rsyslog** doit être en cours d'exécution sur le serveur d'enregistrement et sur les systèmes tentant de s'y connecter.

   a. Veuillez utiliser la commande **systemctl** pour lancer le service **rsyslog**.

   ```
   ~]# systemctl start rsyslog
   ```

   b. Pour vous assurer que le service **rsyslog** démarre automatiquement dans le futur, veuillez saisir la commande suivante en tant qu'utilisateur root :

   ```
   ~]# systemctl enable rsyslog
   ```

Votre serveur d'enregistrement est désormais configuré pour recevoir et stocker des fichiers journaux en provenance des autres systèmes de votre environnement.

20.5.1. Utilisation d'une nouvelle Syntaxe pour les Files d'attente rsyslog

Rsyslog 7 possède un certain nombre de styles de modèles. Le modèle sous le format de string est celui qui ressemble le plus à l'ancien format. Voici à quoi ressemble les reproductions de modèles de l'exemple ci-dessus avec le format de string :

```
template(name="TmplAuthpriv" type="string"
         string="/var/log/remote/auth/%HOSTNAME%/%PROGRAMNAME:::secpath-
replace%.log"
        )

template(name="TmplMsg" type="string"
         string="/var/log/remote/msg/%HOSTNAME%/%PROGRAMNAME:::secpath-
replace%.log"
        )
```

Ces modèles peuvent également être écrits sous un format de liste comme suit :

```
template(name="TmplAuthpriv" type="list") {
    constant(value="/var/log/remote/auth/")
    property(name="hostname")
    constant(value="/")
    property(name="programname" SecurePath="replace")
    constant(value=".log")
    }

template(name="TmplMsg" type="list") {
    constant(value="/var/log/remote/msg/")
    property(name="hostname")
    constant(value="/")
    property(name="programname" SecurePath="replace")
    constant(value=".log")
    }
```

Ce modèle de format texte est sans doute plus facile à lire pour ceux qui ne sont pas très familiers avec rsyslog, et peut donc peut mieux s'adapter en cas de changement de conditions.

Pour compléter un changement à la nouvelle syntaxe, nous devons reproduire la commande de chargement de module, ajouter un ensemble de règles, puis relier l'ensemble de règles à un protocole, un port et un ruleset :

```
module(load="imtcp")

ruleset(name="remote1"){
     authpriv.*   action(type="omfile" DynaFile="TmplAuthpriv")
      *.info;mail.none;authpriv.none;cron.none action(type="omfile"
DynaFile="TmplMsg")
}

input(type="imtcp" port="10514" ruleset="remote1")
```

20.6. UTILISER DES MODULES RSYSLOG

En raison de son design modulaire, **rsyslog** offre une variété de *modules* qui fournissent des fonctionnalités supplémentaires. Remarquez que ces modules peuvent être écrits par des parties tierces. La plupart des modules fournissent des entrées supplémentaires (consultez les *Modules d'entrée* ci-dessous) ou des sorties (consultez les *Modules de sortie* ci-dessous). D'autres modules fournissent des fonctionnalités spéciales spécifiques à chaque module. Les modules peuvent également offrir des directives de configuration supplémentaires qui deviennent disponibles après le chargement d'un module. Pour charger un module, veuillez utiliser la syntaxe suivante :

```
$ModLoad MODULE
```

où **$ModLoad** est la directive globale qui charge le module spécifié et *MODULE* représente le module souhaité. Par exemple, si vous souhaitez charger le module « Text File Input » (**imfile**) qui permet à **rsyslog** de convertir tout fichier texte standard en message syslog, spécifiez la ligne suivante dans le fichier de configuration **/etc/rsyslog.conf** :

```
$ModLoad imfile
```

rsyslog offre un certain nombre de modules qui sont divisés selon les catégories principales suivantes :

- Modules d'entrée — les modules d'entrée collectent des messages provenant de différentes sources. Le nom d'un module d'entrée commence toujours par le préfixe **im**, comme **imfile** et **imjournal**.
- Modules de sortie — les modules de sortie offrent la possibilité de remettre un message à différentes cibles, par exemple en l'envoyant à travers un réseau, en le stockant dans une base de données, ou le chiffrant. Le nom d'un module de sortie commence toujours par le préfixe **om**, comme **omsnmp**, **omrelp**, etc.
- Modules d'analyse — ces modules sont utiles à la création de règles d'analyse personnalisées ou pour analyser des messages mal formés. Avec des connaissances modérées du langage de programmation C, il est possible de créer votre propre analyseur de messages. Le nom d'un module d'analyse commence toujours par le préfixe **pm**, comme **pmrfc5424**, **pmrfc3164**, et ainsi de suite.
- Modules de modification de messages — les modules de modification de messages changent le contenu des messages syslog. Les noms de ces modules commencent par le préfixe **mm**. Les modules de modification de messages comme **mmanon**, **mmnormalize**, ou **mmjsonparse** sont utilisée pour l'anonymisation ou la normalisation des messages.
- Modules générateurs de chaînes — les modules générateurs de chaînes génèrent des chaînes basées sur le contenu du message et coopèrent bien avec la fonctionnalité de modèle fournie par **rsyslog**. Pour obtenir davantage d'informations sur les modèles, veuillez consulter la Section 20.2.3, « Modèles ». Le nom d'un module générateur de chaînes commence toujours par le préfixe **sm**, comme **smfile** ou **smtradfile**.
- Modules de bibliothèques — les modules de bibliothèques fournissent des fonctionnalités pour d'autres modules chargeables. Ces modules sont chargés automatiquement par **rsyslog** lorsque nécessaire et ne peuvent pas être configurés par l'utilisateur.

Une liste complète de tous les modules disponibles ainsi que leur description détaillée se trouve sur http://www.rsyslog.com/doc/rsyslog_conf_modules.html.

AVERTISSEMENT

Remarquez que lorsque **rsyslog** charge des modules, il leur fournit accès à certaines de ses fonctions et données. Cela peut poser un problème de sécurité. Pour minimiser les risques de sécurité, veuillez utiliser les modules de confiance uniquement.

20.6.1. Importer des fichiers texte

Le module d'entrée de fichiers texte « Text File Input », abbrévié **imfile**, permet à **rsyslog** de convertir tout fichier texte en flux de messages syslog. Vous pouvez utiliser **imfile** pour importer des messages journaux d'applications qui créent leurs propres journaux de fichiers texte. Pour charger **imfile**, ajoutez ce qui suit au fichier **/etc/rsyslog.conf** :

```
$ModLoad imfile
$InputFilePollInterval int
```

Charger **imfile** une seule fois est suffisant, même pendant l'importation de multiples fichiers. La directive globale *$InputFilePollInterval* spécifie à quelle fréquence **rsyslog** recherche des changements dans les fichiers texte connectés. L'intervalle par défaut s'élève à 10 secondes, et pour le modifier, remplacez *int* par un intervalle spécifié en secondes.

Pour identifier les fichiers texte à importer, veuillez utiliser la syntaxe suivante dans le fichier **/etc/rsyslog.conf** :

```
# File 1
$InputFileName path_to_file
$InputFileTag tag:
$InputFileStateFile state_file_name
$InputFileSeverity severity
$InputFileFacility facility
$InputRunFileMonitor

# File 2
$InputFileName path_to_file2
...
```

Quatre paramètres sont requis pour spécifier un fichier texte d'entrée :

- remplacez *path_to_file* par un chemin vers le fichier texte.
- remplacez *tag:* par un nom de balise pour ce message.
- remplacez *state_file_name* par un nom unique pour le *fichier d'état*. Les *fichiers d'état*, qui sont stockés dans le répertoire de travail de rsyslog, conservent les curseurs des fichiers surveillés et marquent less partitions déjà traitées. Si vous les supprimer, les fichiers entiers seront lus à nouveau. Assurez-vous de spécifier un nom qui n'existe pas déjà.
- ajoutez la directive *$InputRunFileMonitor* qui active la surveillance de fichiers. Sans ce paramètre, le fichier texte sera ignoré.

À part les directives requises, plusieurs autres paramètres peuvent être appliqués à l'entrée texte. Paramétrez la sévérité des messages importés en remplaçant *severity* par un mot-clé approprié. Remplacez *facility*par un mot-clé permettant de définir le sous-système qui a produit le message. Le mots-clés pour la sévérité et le type sont les mêmes que ceux utilisés pour les filtres basés « facility/priorité », veuillez consulter la Section 20.2.1, « Filtres ».

Exemple 20.15. Importer des fichiers texte

Le serveur HTTP Apache crée les fichiers journaux en format texte. Pour appliquer les capacités de traitement de **rsyslog** aux messages d'erreur apache, commencez par utiliser le module **imfile** pour importer les messages. Ajoutez ce qui suit au fichier **/etc/rsyslog.conf** :

```
$ModLoad imfile

$InputFileName /var/log/httpd/error_log
$InputFileTag apache-error:
$InputFileStateFile state-apache-error
$InputRunFileMonitor
```

20.6.2. Exporter des messages sur une base de données

Le traitement de données de journal peut être plus rapide et plus convenable lorsqu'effectué dans une base de données, plutôt qu'avec des fichiers texte. Selon le type de DBMS utiliseé, choisissez l'un des divers modules de sortie, tel que **ommysql**, **ompgsql**, **omoracle**, ou **ommongodb**. Alternativement, utilisez le module de sortie générique **omlibdbi** qui repose sur la bibliothèque **libdbi**. Le module **omlibdbi** prend en charge les systèmes des bases de données Firebird/Interbase, MS SQL, Sybase, SQLite, Ingres, Oracle, mSQL, MySQL, et PostgreSQL.

Exemple 20.16. Exporter des messages Rsyslog sur une base de données

Pour stocker des messages rsyslog dans une base de données MySQL, ajoutez ce qui suit dans le fichier **/etc/rsyslog.conf** :

```
$ModLoad ommysql

$ActionOmmysqlServerPort 1234
*.* :ommysql:database-server,database-name,database-userid,database-
password
```

Premièrement, le module de sortie est chargé, puis le port des communications est spécifié. Des informations supplémentaires, comme le nom du serveur et la base de données, ainsi que les données d'authentification sont spécifiées sur la dernière ligne de l'exemple ci-dessus.

20.6.3. Enabling Encrypted Transport

La confidentialité et l'intégrité des transmissions réseau peut être fournie par le protocole de chiffrement *TLS* ou *GSSAPI*.

Le protocole de chiffrement TLS (« *Transport Layer Security* ») est conçu pour sécuriser les communications sur un réseau. Lors de l'utilisation de TLS, les messages rsyslog sont chiffrés avant l'envoi, et une authentification mutuelle existe entre l'envoyeur et le destinataire.

GSSAPI (« *Generic Security Service API* ») est une interface de programmation d'application permettant aux programmes d'accéder aux services de sécurité. Pour l'utiliser en conjonction avec **rsyslog**, vous devez avoir un environnement **Kerberos** fonctionnant correctement.

TLS

Pour faciliter le transport codifié via TLS, vous devez configurer le serveur et le client. Tout d'abord, vous devez créer une clé publique, une clé privée, et un fichier de certificat, consultez Section 12.1.11, « Générer une nouvelle clé et un nouveau certificat »

Du côté *serveur*, configurez les options suivantes :

1. Définir le pilote gtls netstream comme pilote par défaut :

   ```
   $DefaultNetstreamDriver gtls
   ```

2. Fournissez les chemins menant aux fichiers de certificat :

```
$DefaultNetstreamDriverCAFile path_ca.pem
$DefaultNetstreamDriverCertFile path_cert.pem
$DefaultNetstreamDriverKeyFile path_key.pem
```

Remplacer *path_ca.pem* par le chemin qui mène à votre clé publique, *path_cert.pem* par le chemin qui mène au fichier de certificat, et *path_key.pem* par le chemin qui mène à la clé privée.

3. Charger le module imtcp :

```
$ModLoad imtcp
```

4. Démarrer le serveur et définir les options de pilote :

```
$InputTCPServerStreamDriverMode number
$InputTCPServerStreamDriverAuthMode anon
$InputTCPServerRun port
```

Vous pourrez, ici, définir le mode du pilote en remplaçant le *nombre*, inscrire **1** pour activer le mode TCP-only. Le paramètre *anon* signifie que le client n'est pas authentifié. Remplacer *port* pour démarrer un listener sur le port requis.

Côté *client*, veuillez utiliser la configuration suivante :

1. Télécharger la clé publique :

```
$DefaultNetstreamDriverCAFile path_ca.pem
```

Remplacer *path_ca.pem* par le chemin qui mène à la clé publique :

2. Définir le pilote gtls netstream comme pilote par défaut :

```
$DefaultNetstreamDriver gtls
```

3. Configurez le pilote et spécifier l'action à effectuer :

```
$InputTCPServerStreamDriverMode number
$InputTCPServerStreamDriverAuthMode anon
*.* @@server.net:10514
```

Remplacer *nombre* et *anon* selon les configurations correspondantes du serveur. Sur la dernière ligne, un exemple d'action transfert les messages du serveur vers un port TCP indiqué.

Utiliser GSSAPI

Dans **rsyslog**, l'interaction avec GSSAPI est fournie par le module *imgssapi*. Pour activer le mode de transfert GSSAPI, utiliser la configuration suivante dans **/etc/rsyslog.conf** :

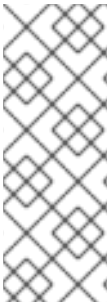
```
$ModLoad imgssapi
```

Cette directive charge le module imgssapi. Après cela, vous pourrez indiquer ce qui suit :

```
$InputGSSServerServiceName name
$InputGSSServerPermitPlainTCP on/off
$InputGSSServerMaxSessions number
```

```
$InputGSSServerRun port
```

Vous pouvez définir un nom de serveur GSS en remplaçant le *nom*. Activer le paramètre *$InputGSSServerPermitPlainTCP* pour permettre au serveur de recevoir également des messages TCP sur le même port. Cela n'est pas permis par défaut. Remplacer le *nombre* pour définir le nombre maximum de sessions prises en charge. Par défaut, ce nombre n'est pas limité. Remplacer *port* par un port que vous aurez sélectionné, et sur lequel vous souhaitez démarrer un serveur GSS.



NOTE

Le module **imgssapi** sera initié aussitôt que le lecteur de fichier de configuration rencontrera la directive $InputGSSServerRun dans le fichier de configuration **/etc/rsyslog.conf**. Les options supplémentaires configurées après $InputGSSServerRun sont donc ignorées. Pour que la configuration puisse avoir lieu, toutes les options de configuration d'imgssapi doivent être mises avant $InputGSSServerRun.

Exemple 20.17. Utiliser GSSAPI

La configuration suivante autorise un serveur GSS sur le port 1514, et permet également de recevoir des messages syslog tcp en texte brut sur le même port.

```
$ModLoad imgssapi
$InputGSSServerPermitPlainTCP on
$InputGSSServerRun 1514
```

20.6.4. Utiliser RELP

RELP (« *Reliable Event Logging Protocol* ») est un protocole réseau pour la journalisation de données dans les réseaux d'ordinateurs. Il est conçu pour fournir une remise fiable des messages d'événements, ce qui le rend utile dans les environnements où la perte de messages n'est pas acceptable.

De même que pour la configuration TLS de base, vous devez passer par trois étapes : créer des certificats, configurer le serveur et le client.

1. Commencer par créer une clé publique, une clé privée et un fichier de certificat, voir Section 12.1.11, « Générer une nouvelle clé et un nouveau certificat »

2. Pour configurer le client, télécharger les modules suivants :

   ```
   module(load="imuxsock")
   module(load="omrelp")
   module(load="imtcp")
   ```

   Configurer l'entrée TCP ainsi :

   ```
   input(type="imtcp" port="port")
   ```

   Remplacer *port* pour démarrer un listener sur le port demandé.

   Avec le nouveau format de configuration, toutes les configurations de transport sont définies comme paramètres d'une action :

```
action(type="omrelp" target="target_IP" port="target_port" tls="on"
tls.caCert="path_ca.pem"
tls.myCert="path_cert.pem"
tls.myPrivKey="path_key.pem"
tls.authmode="mode"
tls.permittedpeer=["peer_name"]
)
```

Ici, *target_IP* et *target_port* sont utilisés pour identifier le serveur cible, le paramètre *tls="on"* active le protocole TLS. Passez les chemins de fichiers de certification par *path_ca.pem*, *path_cert.pem*, et *path_key.pem*. Pour définir le mode d'authentification de la transaction, remplacer le *mode* par *"name"* ou *"fingerprint"*. Avec *tls.permittedpeer*, vous pourrez limiter la connexion à un groupe de pairs sélectionnés. Remplacer *peer_name* par une empreinte de certificat de pair autorisé.

3. La configuration de serveur démarre par le chargement de module :

```
module(load="imuxsock")
module(load="imrelp" ruleset="relp")
```

Configurer l'entrée TCP sur le modèle de la configuration client :

```
input(type="imrelp" port="target_port" tls="on"
tls.caCert="path_ca.pem"
tls.myCert="path_cert.pem"
tls.myPrivKey="path_key.pem"
tls.authmode="name"
tls.permittedpeer=["peer_name","peer_name1","peer_name2"]
)
```

Ces paramètres d'entrée ont la même signification que les options de configuration du client qui sont expliquées dans la deuxième étape. Dans la dernière étape, configurez les règles et sélectionnez une action. Dans l'exemple suivant, les messages se trouvent à l'emplacement suivant :

```
ruleset (name="relp") {
action(type="omfile" file="log_path")
}
```

Remplacer ici *log_path* par un chemin menant à un répertoire où vous souhaitez stocker vos fichiers de journalisation.

20.7. INTERACTION DE RSYSLOG ET DE JOURNAL

Comme mentionné ci-dessus, **Rsyslog** et **Journal**, les deux applications de journalisation présentes sur votre système possèdent des fonctionnalités distinctes les rendant convenables dans certains cas d'utilisation particuliers. Dans de nombreuses situations, il est utile de combiner leurs capacités, par exemple de créer des message structurés et les stocker dans une base de données de fichiers (veuillez consulter la Section 20.8, « Journalisation structurée avec Rsyslog »). Une interface de communication nécessaire à cette coopération est fournie par des modules d'entrée et de sortie à côté de **Rsyslog** et par le socket de communication **Journal**.

Par défaut, **rsyslogd** utilise le module **imjournal** comme mode d'entrée par défaut pour les fichiers journaux. Avec ce module, vous importez non seulement les messages, mais également les données

structurées fournies par **journald**. Aussi, des données plus anciennes peuvent être importées de **journald** (sauf si interdit avec la directive **$ImjournalIgnorePreviousMessages**). Veuillez consulter la Section 20.8.1, « Importer des données de Journal » pour voir une configuration de base de **imjournal**.

Alternativement, veuillez configurer **rsyslogd** pour effectuer la lecture à partir du socket fourni par **journal** en tant que sortie pour des applications basées syslog. Le chemin vers le socket est le suivant : **/run/systemd/journal/syslog**. Veuillez utiliser cette option lorsque vous souhaitez maintenir des messages syslog simples. Comparé à **imjournal**, l'entrée du socket offre actuellement davantage de fonctionnalités, comme la liaison ou le filtrage de rulesets. Pour importer des données **Journal** à travers le socket, veuillez utiliser la configuration suivante dans le fichier **/etc/rsyslog.conf** :

```
$ModLoad imuxsock
$OmitLocalLogging off
```

La syntaxe ci-dessus charge le module **imuxsock** et éteint la directive **$OmitLocalLogging**, qui permet l'importation à travers le socket du système. Le chemin vers ce socket est spécifié séparément, dans le fichier **/etc/rsyslog.d/listen.conf** comme suit :

```
$SystemLogSocketName /run/systemd/journal/syslog
```

Vous pouvez également faire sortir des messages de **Rsyslog** sur **Journal** avec le module **omjournal**. Configurez la sortie dans le fichier **/etc/rsyslog.conf** comme suit :

```
$ModLoad omjournal

*.* :omjournal:
```

Par exemple , la configuration suivante transfère tous les messages reçus sur le port TCP 10514 au Journal :

```
$ModLoad imtcp
$ModLoad omjournal

$RuleSet remote
*.* :omjournal:

$InputTCPServerBindRuleset remote
$InputTCPServerRun 10514
```

20.8. JOURNALISATION STRUCTURÉE AVEC RSYSLOG

Sur les systèmes produisant de grandes quantités de données de journalisation, il est plus pratique de maintenir les messages journaux sous un *format structuré*. Avec des messages structurés, il est plus facile de rechercher des informations particulières, de produire des statistiques et de gérer les changements et incohérences dans la structure du message. **Rsyslog** utilise le format *JSON* (« JavaScript Object Notation ») pour la structure des messages journaux.

Comparez le message journal non structuré suivant :

```
Oct 25 10:20:37 localhost anacron[1395]: Jobs will be executed
sequentially
```

Avec un message structuré :

```
{"timestamp":"2013-10-25T10:20:37", "host":"localhost",
"program":"anacron", "pid":"1395", "msg":"Jobs will be executed
sequentially"}
```

Rechercher des données structurées en utilisant des paires clé-valeur est plus rapide et plus précis qu'effectuer des recherches de fichiers texte avec des expressions régulières. La structure permet également de rechercher la même entrée dans des messages produits par différentes applications. Les fichiers JSON peuvent aussi être stockés dans une base de données de documents, comme MongoDB, qui fournit de meilleures performances et capacités d'analyses. D'autre part, un message structuré requiert davantage d'espace disque qu'un message non structuré.

Dans **rsyslog**, les messages journaux avec des métadonnées sont extraits de **Journal** à l'aide du module **`imjournal`**. Avec le module **`mmjsonparse`**, vous pouvez analyser des données importées de **Journal** et d'autres sources, puis les traiter davantage, par exemple en tant que sortie de base de données. Pour que l'analyse réussisse, **`mmjsonparse`** requiert que les messages d'entrée soient structurés d'une manière définie par le projet **Lumberjack**.

Le projet **Lumberjack** vise à ajouter la journalisation structurée à **rsyslog** de manière rétro-compatible. Pour identifier un message structuré, **Lumberjack** spécifie la chaîne **@cee:** qui ajoute la structure JSON au début. **Lumberjack** définit également la liste des noms de champ standard devant être utilisés pour les entités dans la chaîne JSON. Pour obtenir davantage d'informations sur **Lumberjack**, veuillez consulter la section intitulée « Documentation en ligne ».

Ci-dessous figure un exemple de message formaté lumberjack :

```
 @cee: {"pid":17055, "uid":1000, "gid":1000, "appname":"logger",
"msg":"Message text."}
```

Pour créer cette structure dans **Rsyslog**, un modèle est utilisé, veuillez consulter la Section 20.8.2, « Filtrer des messages structurés ». Les applications et serveurs peuvent utiliser la bibliothèque **`libumberlog`** pour générer des messages sous un format compatible avec lumberjack. Pour obtenir davantage d'informations sur **`libumberlog`**, veuillez consulter la section intitulée « Documentation en ligne ».

20.8.1. Importer des données de Journal

Le module **`imjournal`** est le module d'entrée de **Rsyslog** permettant de lire les fichiers journaux de manière native (consultez la Section 20.7, « Interaction de Rsyslog et de Journal »). Les messages Journal sont ensuite journalisés en format texte comme les autres messages rsyslog. Cependant, avec un traitement supplémentaire, il est possible de traduire les métadonnées fournies par **Journal** en message structuré.

Pour importer les données de **Journal** sur **Rsyslog**, veuillez utiliser la configuration suivante dans le fichier **`/etc/rsyslog.conf`** :

```
$ModLoad imjournal

$imjournalPersistStateInterval number_of_messages
$imjournalStateFile path
$imjournalRatelimitInterval seconds
$imjournalRatelimitBurst burst_number
$ImjournalIgnorePreviousMessages off/on
```

- Avec *number_of_messages*, il est possible de spécifier la fréquence à laquelle les données Journal doivent être enregistrées. Ceci se produit chaque fois que le nombre spécifié de messages est atteint.
- Remplacez *path* par un chemin vers le fichier d'état. Ce fichier surveille la dernière entrée de journal traitée.
- Avec *seconds*, il est possible de définir la longueur de l'intervalle de limite du taux. Le nombre de messages traités pendant cet intervalle ne peut pas dépasser la valeur spécifiée dans *burst_number*. Le paramètre par défaut s'élève à 20,000 messages pour 600 secondes. Rsyslog abandonne les messages arrivant après la rafale maximale dans le délai spécifié.
- Avec **`$ImjournalIgnorePreviousMessages`**, vous pouvez ignorer les messages se trouvant actuellement dans le Journal et importer les nouveaux messages uniquement, ce qui est utilisé lors qu'aucun fichier d'état n'est spécifié. Le paramètre par défaut est « **`off`** ». Veuillez remarquer que si ce paramètre est inactif et qu'il n'y a pas de fichier d'état, tous les messages dans le Journal sont traités, même s'ils ont déjà été traités dans une session rsyslog précédente.

NOTE

Vous pouvez utiliser **`imjournal`** simultanément avec le module **`imuxsock`** qui est l'entrée de journal système traditionnelle. Cependant, pour éviter la duplication de messages, vous devez empêcher **`imuxsock`** de lire le socket du système du Journal. Pour cela, veuillez utiliser la directive **`$OmitLocalLogging`** :

```
$ModLoad imuxsock
$ModLoad imjournal

$OmitLocalLogging on
$AddUnixListenSocket /run/systemd/journal/syslog
```

Vous pouvez traduire toutes les données et métadonnées stockées par le **Journal** en messages structurés. Certaines entrées de métadonnées sont répertoriées dans l'Exemple 20.19, « Sortie détaillée de journalctl ». Pour une liste complète des champs de Journal, veuillez consulter la page man de **`systemd.journal-fields(7)`**. Par exemple, il est possible de se concentrer sur les *champs de journal du noyau*, qui sont utilisés par des messages provenant du noyau.

20.8.2. Filtrer des messages structurés

Pour créer un message formaté lumberjack requis par le module d'analyse **rsyslog**, utilisez le modèle suivant :

```
template(name="CEETemplate" type="string" string="%TIMESTAMP% %HOSTNAME%
%syslogtag% @cee: %$!all-json%\n")
```

Ce modèle ajoute la chaîne **`@cee:`** au début de la chaîne JSON et peut être appliqué, par exemple, pendant la création d'un fichier de sortie avec le module **`omfile`**. Pour accéder aux noms de champ JSON, veuillez utiliser le préfixe **`$!`**. Ainsi, la condition de filtre suivante recherche des messages avec un *nom d'hôte* et un *UID* spécifique :

```
($!hostname == "hostname" && $!UID== "UID")
```

20.8.3. Analyser JSON

Le module **mmjsonparse** est utilisé pour analyser les messages structurés. Ces messages peuvent provenir de **Journal** ou d'autres sources d'entrées, et doivent être formatés d'une manière définie par le projet **Lumberjack**. Ces messages sont identifiés par la présence de la chaîne **@cee:**. **mmjsonparse** vérifie si la structure JSON est valide, puis le message est analysé.

Pour analyser des messages JSON formatés lumberjack avec **mmjsonparse**, veuillez utiliser la configuration suivante dans le fichier **/etc/rsyslog.conf** :

```
$ModLoad mmjsonparse

*.* :mmjsonparse:
```

Dans cet exemple, le module **mmjsonparse** est chargé sur la première ligne, puis tous les messages y sont transférés. Aucun paramètre de configuration n'est actuellement disponible pour **mmjsonparse**.

20.8.4. Stocker des messages dans MongoDB

Rsyslog prend en charge le stockage des journaux JSON dans la base de données de documents MongoDB à travers le module de sortie *ommongodb*.

Pour transférer des messages journaux dans MongoDB, veuillez utiliser la syntaxe suivante dans le fichier **/etc/rsyslog.conf** (les paramètres de configuration de *ommongodb* sont uniquement disponibles sous le nouveau format de configuration ; veuillez consulter la Section 20.3, « Utiliser le nouveau format de configuration ») :

```
$ModLoad ommongodb

*.* action(type="ommongodb" server="DB_server" serverport="port"
db="DB_name" collection="collection_name" uid="UID" pwd="password")
```

- Remplacez *DB_server* par le nom ou l'adresse du serveur MongoDB. Spécifiez *port* pour sélectionner un port non standard du serveur MongoDB. La valeur *port* par défaut est **0**, et il n'est pas habituellement nécessaire de modifier ce paramètre.
- Avec *DB_name*, il est possible d'identifier la base de données sur le serveur MongoDB sur laquelle vous souhaitez diriger la sortie. Remplacez *collection_name* par le nom d'une collection dans cette base de données. Dans MongoDB, une collection est un groupe de documents, l'équivalent d'une table RDBMS.
- Vous pouvez définir vos détails de connexion en remplaçant l'*UID* et le *mot de passe*.

Vous pouvez façonner la forme de la sortie finale de la base de données en utilisant des modèles. Par défaut, **rsyslog** utilise un modèle basé sur les noms de champs **lumberjack** standard.

20.9. DÉBOGUER RSYSLOG

Pour exécuter **rsyslogd** en mode de débogage, veuillez utiliser la commande suivante :

```
rsyslogd -dn
```

Avec cette commande, **rsyslogd** produit des informations de débogage et les imprime sur la sortie standard. L'option **-n** correspond à « no fork ». Vous pouvez modifier le débogage avec des variables

d'environnement. Par exemple, vous pouvez stocker la sortie du débogage dans un fichier journal. Avant de lancer **rsyslogd**, veuillez saisir ce qui suit sur la ligne de commande :

```
export RSYSLOG_DEBUGLOG="path"
export RSYSLOG_DEBUG="Debug"
```

Remplacez *path* par l'emplacement souhaité du fichier où les informations de débogage seront journalisées. Pour une liste complète des options disponibles pour la variable RSYSLOG_DEBUG, veuillez consulter la section connexe dans la page man de **rsyslogd(8)**.

Pour vérifier si la syntaxe utilisée dans le fichier **/etc/rsyslog.conf** est valide, veuillez utiliser :

```
rsyslogd -N 1
```

Où **1** représente le niveau de détails du message de sortie. Ceci est une option de compatibilité de transfert car un seul niveau est actuellement fourni. Cependant, vous devez ajouter cet argument pour exécuter la validation.

20.10. UTILISER LE JOURNAL

Journal est un composant de **systemd** responsable de l'affichage et de la gestion des fichiers journaux. Il peut être utilisé en parallèle ou à la place d'un démon syslog traditionnel, tel que **rsyslogd**. Journal a été développé pour répondre aux problèmes liés aux connexions traditionnelles. Il est étroitement intégré avec le reste du système, prend en charge diverses technologies de connexion et la gestion des accès pour les fichiers journaux.

Les données de journalisation sont collectées, stockées, et traitées par le service **journald** de Journal. Il crée et maintient des fichiers binaires nommés des *journaux* basés sur des informations de journalisation reçues du noyau, des processus utilisateur, de la sortie standard, et de la sortie d'erreurs standard des services système ou via son API native. Ces journaux sont structurés et indexés, ce qui fournit un temps de recherche relativement rapide. Les entrées de journaux peuvent comporter un identifiant unique. Le service **journald** collecte de nombreux champs de métadonnées pour chaque message de journal. Les fichiers journaux sont sécurisés et ne peuvent donc pas être modifiés manuellement.

20.10.1. Afficher les fichiers journaux

Pour accéder aux enregistrements de Journal, veuillez utiliser l'outil **journalctl**. Pour un affichage de base des journaux, veuillez saisir en tant qu'utilisateur **root** :

```
journalctl
```

Une sortie de cette commande est une liste de tous les fichiers journaux générés sur le système, y compris des messages générés par des composants système et par des utilisateurs. La structure de cette sortie est similaire à celle utilisée dans **/var/log/messages/** mais elle offre quelques améliorations :

- la priorité des entrées est marquée visuellement. Des lignes de priorités d'erreurs et des priorités plus élevées sont surlignées en rouge et des caractère en gras sont utilisés pour les lignes avec une notification et une priorité d'avertissement
- les horodatages sont convertis au fuseau horaire local de votre système
- toutes les données journalisées sont affichées, y compris les journaux rotatifs

- le début d'un démarrage est marqué d'une ligne spéciale

Exemple 20.18. Exemple de sortie journalctl

Ci-dessous figure une exemple de sortie fourni par l'outil **journalctl**. Lorsqu'appelé sans paramètres, les entrées répertoriées commencent par un horodatage, puis le nom d'hôte et l'application qui ont effectué l'opération sont mentionnés suivis du message. Cet exemple montre les trois premières entrées de l'enregistrement du journal :

```
# journalctl
-- Logs begin at Thu 2013-08-01 15:42:12 CEST, end at Thu 2013-08-01
15:48:48 CEST. --
Aug 01 15:42:12 localhost systemd-journal[54]: Allowing runtime journal
files to grow to 49.7M.
Aug 01 15:42:12 localhost kernel: Initializing cgroup subsys cpuset
Aug 01 15:42:12 localhost kernel: Initializing cgroup subsys cpu

[...]
```

Dans de nombreux cas, seules les dernières entrées de l'enregistrement du journal sont pertinentes. La manière la plus simple de réduire la sortie **journalctl** consiste à utiliser l'option **-n** qui répertorie uniquement le nombre spécifié des entrées les plus récentes du journal :

```
journalctl -n Number
```

Remplacez *Number* par le nombre de lignes à afficher. Lorsqu'aucun nombre n'est spécifié, **journalctl** affiche les dix entrées les plus récentes.

La commande **journalctl** permet de contrôler le format de la sortie avec la syntaxe suivante :

```
journalctl -o form
```

Remplacez *form* par un mot-clé spécifiant la forme de sortie souhaitée. Il existe plusieurs options, comme **verbose**, qui retourne des éléments d'entrée complètement structurés avec tous les champs, **export**, qui crée un courant binaire convenable aux sauvegardes et transferts réseau, et **json**, qui formate les entrées en tant que structures de données JSON. Pour obtenir la liste complète des mots-clés, veuillez consulter la page man de **journalctl(1)**.

Exemple 20.19. Sortie détaillée de journalctl

Pour afficher la totalité des métadonnées concernant toutes les entrées, veuillez saisir :

```
# journalctl -o verbose
[...]

Fri 2013-08-02 14:41:22 CEST
[s=e1021ca1b81e4fc688fad6a3ea21d35b;i=55c;b=78c81449c920439da57da7bd5c56
a770;m=27cc
        _BOOT_ID=78c81449c920439da57da7bd5c56a770
        PRIORITY=5
        SYSLOG_FACILITY=3
        _TRANSPORT=syslog
        _MACHINE_ID=69d27b356a94476da859461d3a3bc6fd
```

```
        _HOSTNAME=localhost.localdomain
        _PID=562
        _COMM=dbus-daemon
        _EXE=/usr/bin/dbus-daemon
        _CMDLINE=/bin/dbus-daemon --system --address=systemd: --nofork
--nopidfile --systemd-activation
        _SYSTEMD_CGROUP=/system/dbus.service
        _SYSTEMD_UNIT=dbus.service
        SYSLOG_IDENTIFIER=dbus
        SYSLOG_PID=562
        _UID=81
        _GID=81
        _SELINUX_CONTEXT=system_u:system_r:system_dbusd_t:s0-
s0:c0.c1023
        MESSAGE=[system] Successfully activated service
'net.reactivated.Fprint'
        _SOURCE_REALTIME_TIMESTAMP=1375447282839181

[...]
```

Cet exemple répertorie des champs qui identifient une entrée de journal unique. Ces métadonnées peuvent être utilisées pour le filtrage de messages, comme indiqué dans la section intitulée « Filtrage avancé ». Pour une description complète de tous les champs possibles, veuillez consulter la page man de **`systemd.journal-fields(7)`**.

20.10.2. Contrôle des accès

Par défaut, les utilisateurs de **Journal** sans privilèges **`root`** peuvent uniquement voir les fichiers journaux qu'ils ont générés. L'administrateur systèmes peut ajouter les utilisateurs sélectionnés au groupe *adm*, qui leur offre accès aux fichiers journaux complets. Pour effectuer ceci, veuillez saisir en tant qu'utilisateur **`root`** :

```
usermod -a -G adm username
```

Remplacez *username* par un nom d'utilisateur à ajouter au groupe *adm*. Cet utilisateur reçoit ensuite la même sortie de la commande **`journalctl`** que l'utilisateur root. Remarquez que le contrôle des accès fonctionne uniquement lorsque le stockage persistant est activé pour **Journal**.

20.10.3. Utiliser l'affichage Live

Lorsqu'appelé sans paramètres, **`journalctl`** affiche la liste complète des entrées, en commençant par l'entrée collectée la plus ancienne. Avec l'affichage Live, vous pouvez superviser les messages journaux en temps réel pendant l'impression continuelle des nouvelles entrées au fur et à mesure qu'elles apparaissent. Pour lancer **journalctl** en mode d'affichage Live, veuillez saisir :

```
journalctl -f
```

Cette commande retourne une liste des dix lignes de journal les plus actuelles. L'utilitaire **journalctl** continue ensuite de s'exécuter et attend que de nouveaux changements se produisent pour les afficher immédiatement.

20.10.4. Filtrer les messages

La sortie de la commande **journalctl** exécutée sans paramètre est souvent extensive. Ainsi, vous pouvez utiliser différentes méthodes de filtrage pour extraire des informations qui correspondent à vos besoins.

Filtrer par priorité

Les messages journaux sont souvent utilisés pour suivre des comportements erronés sur le système. Pour afficher les entrées avec une priorité sélectionnée ou plus élevée, veuillez utiliser la syntaxe suivante :

```
journalctl -p priority
```

Remplacez *priority* par l'un des mots-clés suivants (ou par un chiffre) : **debug** (7), **info** (6), **notice** (5), **warning** (4), **err** (3), **crit** (2), **alert** (1), et **emerg** (0).

Exemple 20.20. Filtrer par priorité

Pour uniquement afficher les entrées avec une *erreur* ou une priorité plus élevée, veuillez utiliser :

```
journalctl -p err
```

Filtrer par heure

Pour uniquement afficher les entrées de journal du démarrage actuel, veuillez saisir :

```
journalctl -b
```

Si vous redémarrez votre système uniquement de manière occasionnelle, l'option **-b** ne réduira pas significativement la sortie de **journalctl**. Dans de tels cas, le filtrage basé heure sera plus utile :

```
journalctl --since=value --until=value
```

Avec **--since** et **--until**, vous pouvez uniquement afficher les messages journaux créés dans un intervalle spécifié. Vous pouvez attribuer des *valeurs* à ces options sous la forme de date ou d'heure, ou les deux, comme indiqué dans l'exemple ci-dessous.

Exemple 20.21. Filtrer par heure et par priorité

Les options de filtrage peuvent être combinées pour réduire l'ensemble des résultats selon les requêtes spécifiques. Par exemple, pour afficher les messages d'*avertissement* ou avec une priorité plus élevée à partir d'un certain moment, veuillez saisir :

```
journalctl -p warning --since="2013-3-16 23:59:59"
```

Filtrage avancé

Exemple 20.19, « Sortie détaillée de journalctl » répertorie un ensemble de champs qui spécifient une entrée de journal et peuvent tous être utilisés pour le filtrage. Pour une description complète des métadonnées pouvant être stockées par **systemd**, veuillez consulter la page man de **systemd.journal-fields(7)**. Ces métadonnées sont collectées pour chaque message journal, sans intervention de la part de l'utilisateur. Les valeurs sont habituellement basées texte, mais elles peuvent également prendre des valeurs de binaire et de grande taille. Les champs peuvent se voir assigner des multiples valeurs, même si cela n'est pas très commun.

Pour afficher une liste de valeurs uniques se produisant dans un champ spécifique, veuillez utiliser la syntaxe suivante :

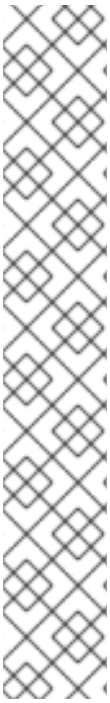
```
journalctl -F fieldname
```

Remplacez *fieldname* par le nom du champ qui vous intéresse.

Pour uniquement afficher les entrées de journal correspondant à une condition particulière, veuillez utiliser la syntaxe suivante :

```
journalctl fieldname=value
```

Remplacez *fieldname* par un nom de champ et *value* par une valeur spécifique contenue dans ce champ. Par conséquent, seules les lignes correspondantes à cette condition seront retournées.



NOTE

Comme le nombre de champs de métadonnées stockés par **systemd** est assez important, il est facile d'oublier le nom exact du champ d'intérêt. En cas d'incertitude, veuillez saisir :

```
journalctl
```

et appuyez sur la touche **Tab** deux fois. Cela affichera une liste de noms de champs disponibles. La complétion **Tab** basée sur le contexte fonctionne sur les noms de champ, et vous pouvez ainsi saisir un ensemble distinctif de lettres d'un nom de champ et appuyer sur **Tab** pour compléter le nom automatiquement. Similairement, vous pouvez répertorier des valeurs uniques à partir d'un champ. Saisissez :

```
journalctl fieldname=
```

et appuyez sur la touche **Tab** deux fois. Ceci sert d'alternative à **journalctl -F** *fieldname*.

Vous pouvez spécifier de multiples valeurs pour un champ :

```
journalctl fieldname=value1 fieldname=value2 ...
```

Spécifier deux correspondances pour le même champ résulte en une combinaison **OR** des deux correspondances. Les entrées qui correspondent à *value1* ou *value2* sont affichées.

Aussi, vous pouvez spécifier de multiples paires champ/valeur pour réduire encore plus l'ensemble de la sortie :

```
journalctl fieldname1=value fieldname2=value ...
```

Si deux correspondances pour différents noms de champs sont spécifiées, elles seront combinées avec un **AND** logique. Les entrées doivent correspondre aux deux conditions pour être affichées.

Avec l'utilisation du symbole *+*, vous pouvez définir une combinaison **OR** logique des correspondances pour plusieurs champs :

```
journalctl fieldname1=value + fieldname2=value ...
```

Cette commande retourne des entrées qui correspondent à au moins une des conditions, et non seulement celles qui correspondent aux deux.

Exemple 20.22. Filtrage avancé

Pour afficher les entrées créées par **avahi-daemon.service** ou **crond.service** sous un utilisateur avec l'UID 70, veuillez utiliser la commande suivante :

```
journalctl _UID=70 _SYSTEMD_UNIT=avahi-daemon.service
_SYSTEMD_UNIT=crond.service
```

Comme il existe deux ensembles de valeurs pour le champ **_SYSTEMD_UNIT**, les deux résultats seront affichés, mais uniquement lorsqu'ils correspondent à la condition **_UID=70**. Ceci peut être exprimé simplement comme suit : (UID=70 and (avahi or cron)).

Vous pouvez également appliquer le filtrage susmentionné dans le mode d'affichage Live pour suivre les derniers changements dans le groupe sélectionné d'entrées de journal :

```
journalctl -f fieldname=value ...
```

20.10.5. Activer le stockage persistant

Par défaut, **Journal** stocke uniquement les fichiers journaux dans la mémoire ou dans une mémoire tampon en anneau dans le répertoire **/run/log/journal/**. Cela est suffisant pour afficher l'historique récent des journaux avec **journalctl**. Ce répertoire est volatile, les données de journal ne sont pas enregistrées de manière permanente. Avec la configuration par défaut, syslog lit les enregistrements de journal et les stocke dans le répertoire **/var/log/**. Avec la journalisation persistante activée, les fichiers journaux sont stockés dans **/var/log/journal**, ce qui signifie qu'ils persisteront après un redémarrage. Journal peut ensuite remplacer **rsyslog** pour certains utilisateurs (veuillez consulter l'introduction du chapitre).

Un stockage persistant activé offre les avantages suivant

- Des données enrichies sont enregistrées pour la résolution de problèmes pendant une longue période
- Pour une résolution de problème immédiate, des données enrichies seront disponibles après un redémarrage
- Actuellement, la console du serveur lit les données à partir du journal, et non à partir des fichiers journaux

Le stockage persistant présente également certains inconvénients :

- Même avec un stockage persistant, la quantité de données stockée dépend de la mémoire disponible, il n'y a pas de garantie de couverture d'une période spécifique
- Davantage d'espace disque est nécessaire pour les journaux

Pour activer le stockage persistant pour Journal, créez le répertoire journal manuellement comme indiqué dans l'exemple suivant. En tant qu'utilisateur **root**, veuillez saisir :

```
mkdir -p /var/log/journal/
```

Puis redémarrez **journald** pour appliquer le changement :

```
systemctl restart systemd-journald
```

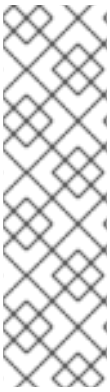
20.11. GÉRER DES FICHIERS JOURNAUX DANS UN ENVIRONNEMENT GRAPHIQUE

Pour fournir une alternative aux utilitaires de ligne de commande susmentionnés, Red Hat Enterprise Linux 7 offre une interface utilisateur graphique accessible pour gérer les messages journaux.

20.11.1. Afficher les fichiers journaux

La plupart des fichiers journaux sont stockés sous un format en texte clair. Il est possible de les afficher avec n'importe quel éditeur de texte, tel que **Vi** ou **Emacs**. Certains fichiers journaux sont lisibles par tous les utilisateurs sur le système. Cependant, des privilèges root sont requis pour lire la plupart des fichiers journaux.

Pour afficher les fichiers journaux du système dans une application interactive en temps réel, veuillez utiliser le journal système « **System Log** ».



NOTE

Pour utiliser le journal système « **System Log** », commencez par vous assurer que le paquet >gnome-system-log soit installé sur votre système en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install gnome-system-log
```

Pour obtenir davantage d'informations sur l'installation de paquets avec Yum, veuillez consulter la Section 8.2.4, « Installation de paquets ».

Après avoir installé le paquet gnome-system-log, ouvrez le journal système « **System Log** » en cliquant sur **Applications** → **Outils système** → **Journal système**, ou saisissez la commande suivante dans l'invite shell :

```
~]$ gnome-system-log
```

L'application affiche uniquement les fichiers journaux qui existent. Ainsi, la liste peut être différente de celle affichée dans la Figure 20.2, « System Log ».

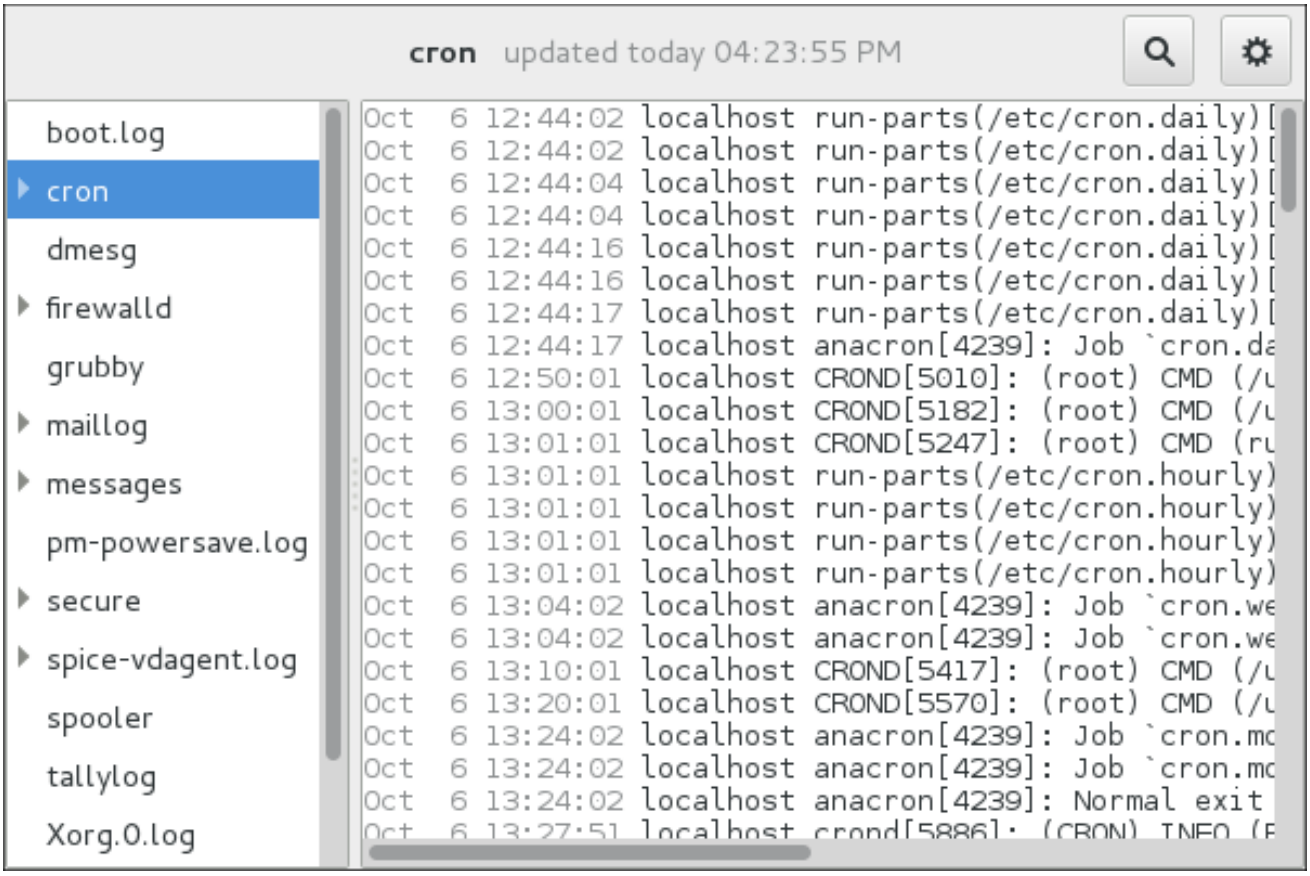


Figure 20.2. System Log

L'application **System Log** permet de filtrer tout fichier journal existant. Cliquez sur le bouton marqué d'un symbole de vitesse pour afficher le menu. Sélectionnez **Filtres** → **Gérer les filtres** pour définir ou modifier le filtre souhaité.

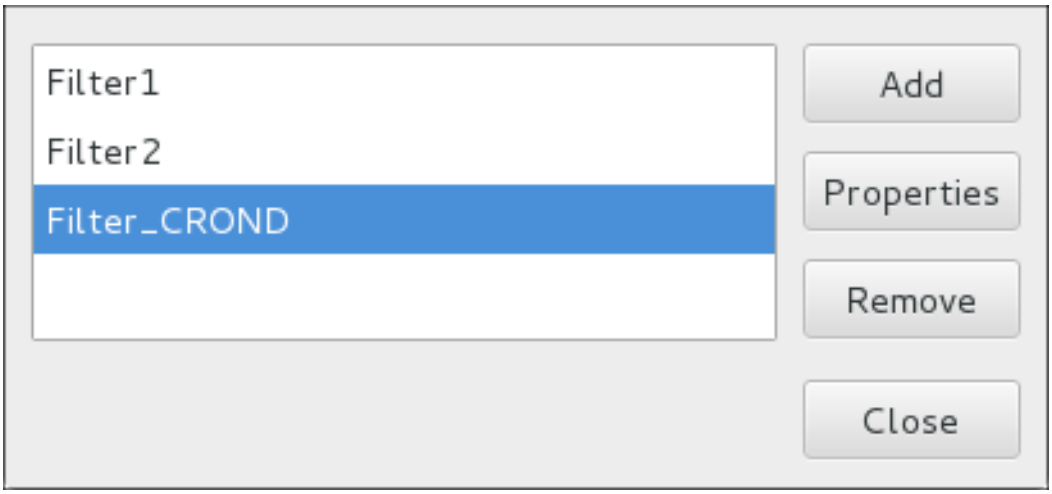


Figure 20.3. System Log - Filtres

Ajouter ou modifier un filtre permet de définir ses paramètres, comme indiqué dans la Figure 20.4, « System Log - définir un filtre ».

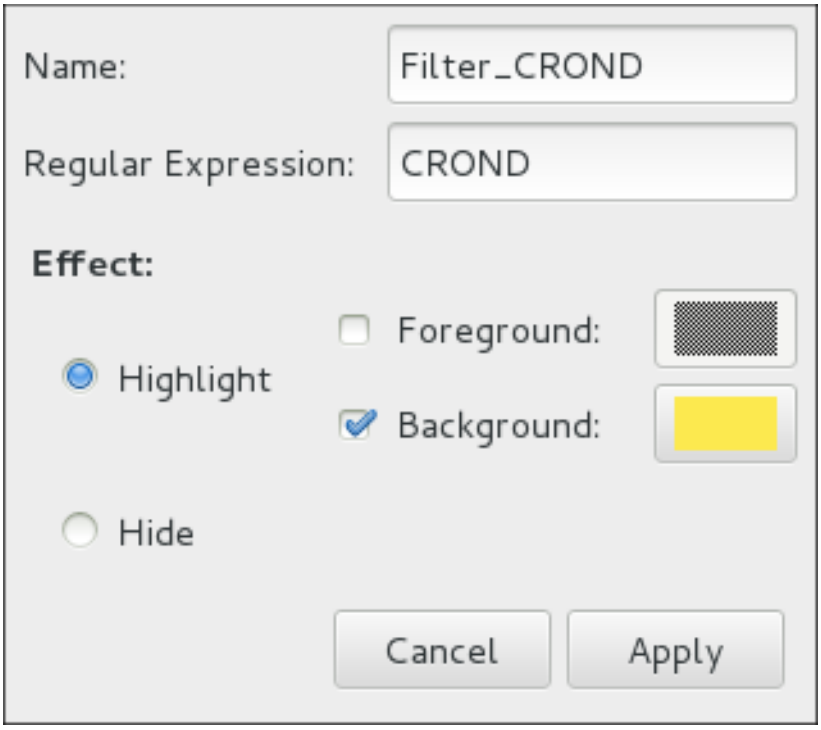


Figure 20.4. System Log - définir un filtre

Lors de la définition d'un filtre, les paramètres suivants peuvent être modifiés :

- **`Name`** — nom du filtre.
- **`Regular Expression`** — expression régulière qui sera appliquée au fichier journal et qui tentera de correspondre à toutes les chaînes possibles de celui-ci.
- **`Effect`**
  - **`Highlight`** — si sélectionné, les résultats trouvés seront surlignés avec la couleur choisie. Vous pouvez également sélectionner si vous souhaitez surligner l'arrière-plan ou l'avant-plan du texte.
  - **`Hide`** — si sélectionné, les résultats trouvés seront cachés du fichier journal affiché.

Lorsque vous aurez défini au moins un filtre, il peut être sélectionné à partir du menu **`Filtres`** et il cherchera automatiquement les chaînes que vous avez définies dans le filtre et surlignera ou cachera chaque résultat dans le fichier journal actuellement affiché.

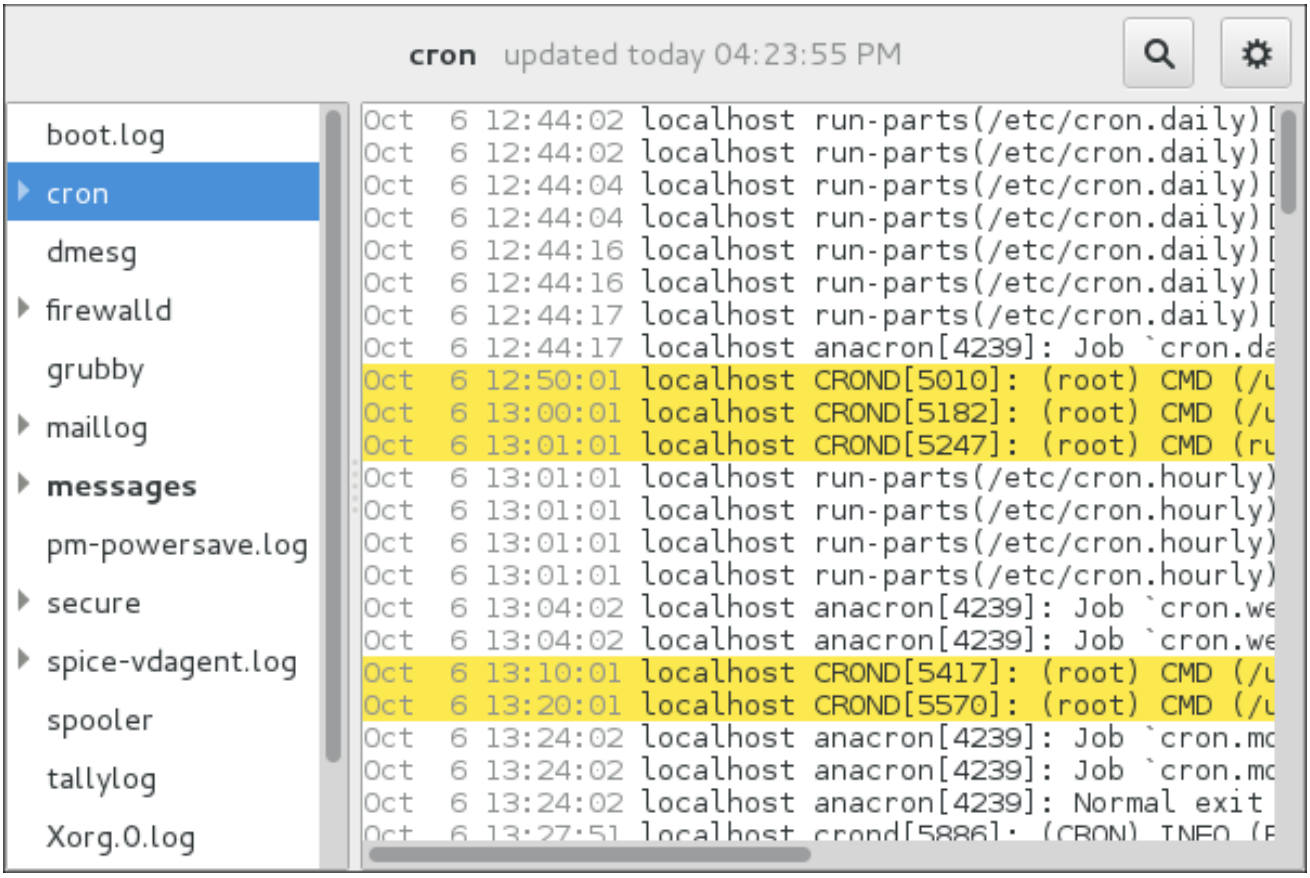


Figure 20.5. System Log - activer un filtre

Lorsque vous sélectionnez l'option **`Uniquement afficher les résultats`**, seules les chaînes avec une correspondance seront affichées dans le fichier journal actuellement affiché.

20.11.2. Ajouter un fichier journal

Pour ajouter un fichier journal que vous souhaitez voir dans la liste, sélectionnez **Fichier** → **Ouvrir**. Cela affichera la fenêtre **`Ouvrir le journal`** dans laquelle vous pouvez sélectionner le nom du répertoire et du fichier journal que vous souhaitez afficher. La Figure 20.6, « System Log - ajouter un fichier journal » illustre la fenêtre **Ouvrir le journal**.

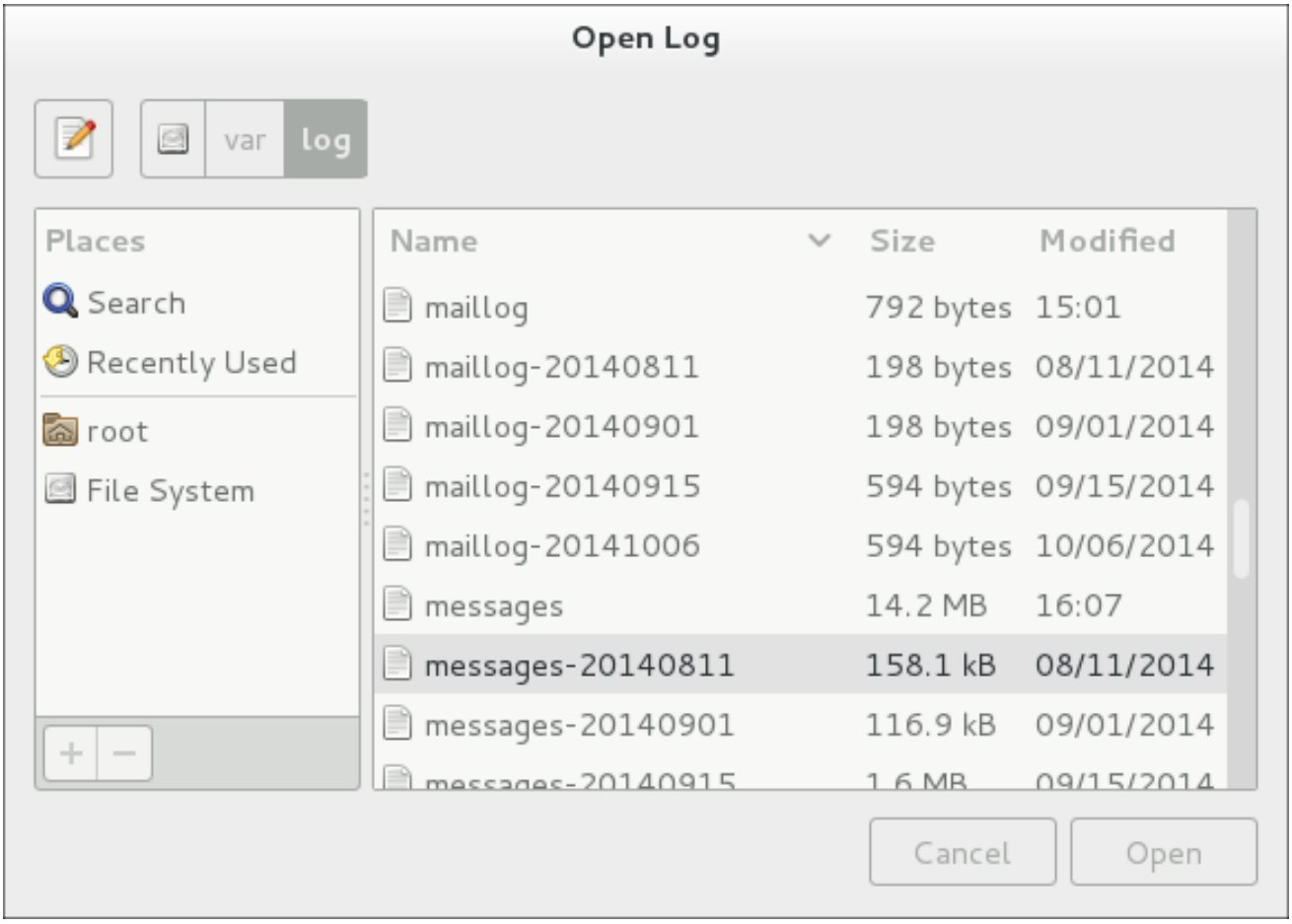


Figure 20.6. System Log - ajouter un fichier journal

Cliquez sur le bouton **Ouvrir** pour ouvrir le fichier. Le fichier est immédiatement ajouté à la liste d'affichage, liste dans laquelle vous pouvez le sélectionner et afficher son contenu.

NOTE

System Log permet également d'ouvrir des fichiers journaux compressés sous le format **.gz**.

20.11.3. Surveiller des fichiers journaux

System Log contrôle tous les journaux ouverts par défaut. Si une nouvelle ligne est ajoutée à un fichier journal contrôlé, le nom du journal apparaît en gras dans la liste des journaux. Si le fichier journal est sélectionné ou affiché, les nouvelles lignes apparaissent en gras en bas du fichier journal. La Figure 20.7, « System Log - nouvelle alerte de journal » illustre une nouvelle alerte dans le fichier journal **cron** et dans le fichier journal **messages**. Cliquer sur le fichier journal **messages** affiche le fichier journal avec les nouvelles lignes en gras.

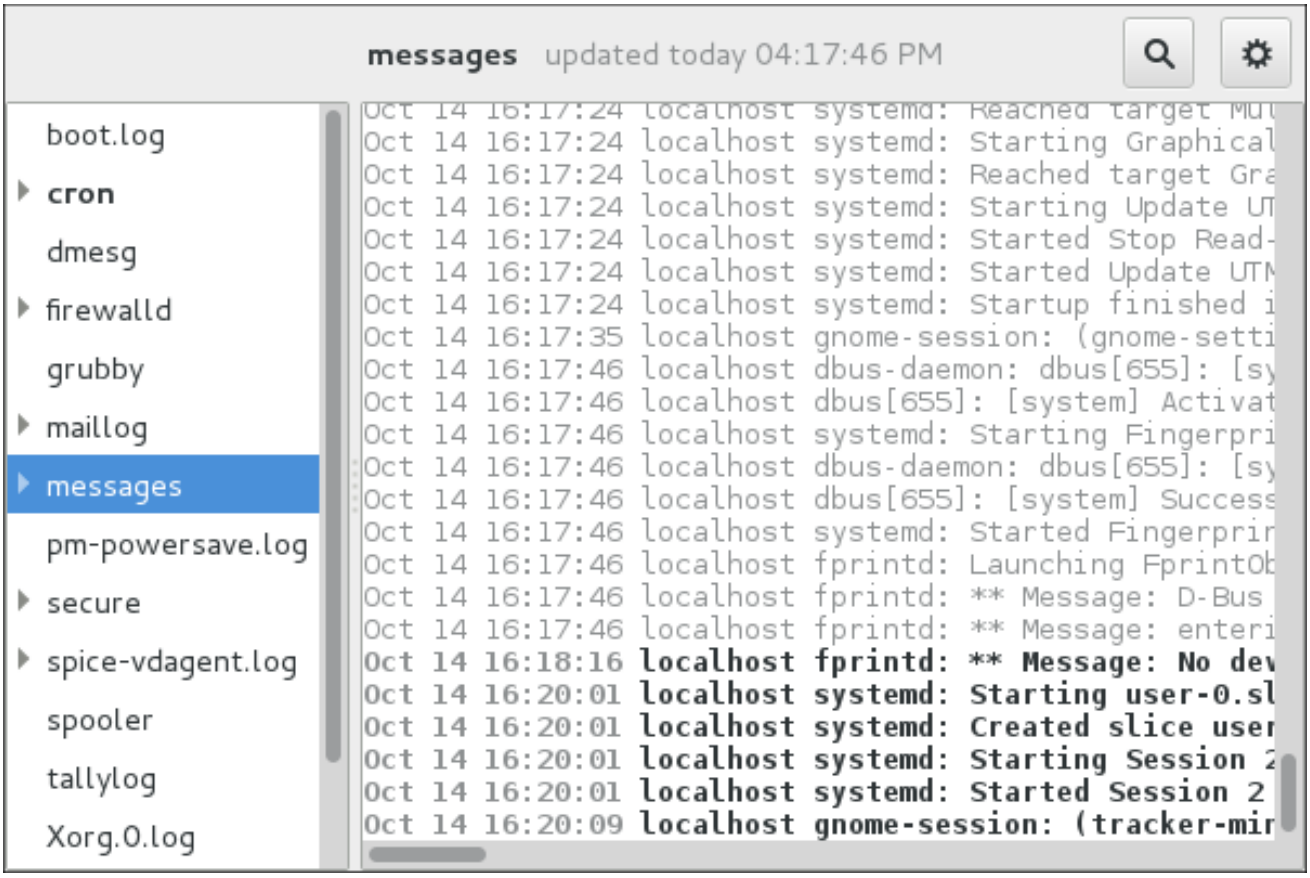


Figure 20.7. System Log - nouvelle alerte de journal

20.12. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur la manière de configurer le démon **rsyslog** et sur la manière de localiser, d'afficher, et de contrôler les fichiers journaux, veuillez consulter les ressources répertoriées ci-dessous.

Documentation installée

- **rsyslogd**(8) — la page du manuel du démon **rsyslogd** documente son utilisation.
- **rsyslog.conf**(5) — la page du manuel nommée **rsyslog.conf** documente ses options de configuration disponibles.
- **logrotate**(8) — la page du manuel de l'utilitaire **logrotate** explique des manière très détaillée comment le configurer et l'utiliser.
- **journalctl**(1) — la page du manuel du démon **journalctl** documente son utilisation.
- **journald.conf**(5) — cette page du manuel documente les options de configuration disponibles.
- **systemd.journal-fields**(7) — cette page du manuel répertorie les champs spéciaux de **Journal**.

Documentation installable

/usr/share/doc/rsyslog*version*/html/index.html — ce fichier, fourni par le paquet rsyslog-doc en provenance du canal « Optional », contient des informations sur **rsyslog**. Voir Section 8.5.7, « Ajouter les référentiels « Optional » (Optionnel) et « Supplementary » (Supplémentaire) » pour obtenir

des informations sur les canaux supplémentaires Red Hat. Avant d'accéder à la documentation, vous devez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install rsyslog-doc
```

Documentation en ligne

La page d'accueil **rsyslog** fournit des documents supplémentaires, des exemples de configuration, et des tutoriels vidéo. Assurez-vous de consulter les documents pertinents à la version que vous utilisez :

- Documentation *RainerScript* de la page d'accueil rsyslog — résumé commenté des types de données, des expressions, et des fonctions disponibles dans *RainerScript*.
- Documentation rsyslog version 7 sur la page d'accueil rsyslog — la version 7 de **rsyslog** est disponible pour Red Hat Enterprise Linux 7 dans le paquet rsyslog.
- Description of *files d'attente* sur la Page d'accueil rsyslog — informations générales sur les divers types de files d'attente de messages et sur leur utilisation.

Voir aussi

- Le Chapitre 5, *Obtention de privilèges* documente la façon d'obtenir des privilèges administratifs en utilisant les commandes **su** et **sudo**.
- Le Chapitre 9, *Gérer les services avec systemd* fournit des informations supplémentaires sur systemd et documente comment utiliser la commande **systemctl** pour gérer les services système.

CHAPITRE 21. AUTOMATISER LES TÂCHES SYSTÈME

Les tâches, aussi appelées *jobs*, peuvent être configurées pour être exécutées automatiquement dans un laps de temps spécifié, à une date spécifiée, ou lorsque la charge moyenne du système passe en-dessous de 0.8.

Red Hat Enterprise Linux est pré-configuré pour exécuter des tâches système importantes afin que le système reste à jour. Par exemple, la base de données slocate utilisée par la commande **locate** est mise à jour quotidiennement. Un administrateur systèmes peut utiliser des tâches automatisées pour effectuer des copies de sauvegarde périodiques, surveiller le système, exécuter des scripts personnalisés, et ainsi de suite.

Red Hat Enterprise Linux est fourni avec les utilitaires de tâches automatisées suivants : **cron**, **anacron**, **at**, et **batch**.

Chaque utilitaire est conçu pour planifier un type de tâche différent : tandis que Cron et Anacron planifient des tâches récurrentes, At et Batch planifient des tâches uniques (veuillez consulter Section 21.1, « Cron et Anacron » et Section 21.2, « « At » et « Batch » »).

Red Hat Enterprise Linux 7 prend en charge l'utilisation de **systemd.timer** pour exécuter une tâche à un moment spécifique. Veuillez consulter la page man **systemd.timer(5)** pour obtenir davantage d'informations.

21.1. CRON ET ANACRON

Cron et Anacron sont des démons pouvant planifier l'exécution de tâches récurrentes à un certain moment, défini par une heure exacte, un jour du mois, un jour de la semaine et par une semaine.

Les tâches Cron peuvent être exécutées chaque minute. Cependant, l'utilitaire suppose que le système soit en cours d'exécution de manière continuelle et si le système n'est pas exécuté lorsqu'une tâche est planifiée, la tâche ne sera pas exécutée.

D'autre part, Anacron se souvient des tâches planifiées si le système n'est pas en cours d'exécution au moment où la tâche est planifiée. La tâche est exécutée aussitôt que le système se trouve en cours d'exécution. Cependant, Anacron ne peut exécuter qu'une seule tâche par jour.

21.1.1. Installer Cron et Anacron

Pour installer Cron et Anacron, vous devrez installer le paquet cronie avec Cron et le paquet cronie-anacron avec Anacron (cronie-anacron est un sous-paquet de cronie).

Pour déterminer si les paquets sont déjà installés sur votre système, veuillez utiliser la commande suivante :

```
rpm -q cronie cronie-anacron
```

La commande retourne les noms complets des paquets cronie et cronie-anacron s'ils sont déjà installés ou vous notifie que les paquets ne sont pas disponibles.

Pour installer ces paquets, veuillez utiliser la commande **yum** sous le format suivant en tant qu'utilisateur **root** :

```
yum install package
```

Par exemple, pour installer Cron et Anacron, veuillez saisir ce qui suit dans une invite de shell :

```
~]# yum install cronie cronie-anacron
```

Pour obtenir davantage d'informations sur la manière d'installer de nouveaux paquets sur Red Hat Enterprise Linux, veuillez consulter la Section 8.2.4, « Installation de paquets ».

21.1.2. Exécuter le service Crond

Les tâches cron et anacron sont effectuées par le service **crond**. Cette section fournit des informations sur la manière de démarrer, arrêter, et de redémarrer le service **crond**, et montre comment le configurer afin qu'il soit lancé automatiquement lors du démarrage. Pour obtenir davantage d'informations sur la manière de gérer des services système sur Red Hat Enterprise Linux 7 en général, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

21.1.2.1. Démarrer et arrêter le service Cron

Pour déterminer si le service est en cours d'exécution, veuillez utiliser la commande suivante :

```
systemctl status crond.service
```

Pour exécuter le service **crond** dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
systemctl start crond.service
```

Pour configurer le service de manière à ce qu'il soit lancé automatiquement lors du démarrage système, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

```
systemctl enable crond.service
```

21.1.2.2. Arrêter le service Cron

Pour arrêter le service **crond** dans sa session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
systemctl stop crond.service
```

Pour empêcher le service d'être lancé automatiquement au démarrage, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

```
systemctl disable crond.service
```

21.1.2.3. Redémarrer le service Cron

Pour redémarrer le service **crond**, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl restart crond.service
```

Cette commande arrête le service et le lance à nouveau en une succession rapide.

21.1.3. Configurer les tâches Anacron

Le fichier de configuration principal pour planifier les tâches est le fichier **/etc/anacrontab**, auquel seul l'utilisateur **root** peut accéder. Le fichier contient ce qui suit :

```
SHELL=/bin/sh
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
# the maximal random delay added to the base delay of the jobs
RANDOM_DELAY=45
# the jobs will be started during the following hours only
START_HOURS_RANGE=3-22

#period in days   delay in minutes   job-identifier   command
1         5     cron.daily    nice run-parts /etc/cron.daily
7         25    cron.weekly   nice run-parts /etc/cron.weekly
@monthly  45    cron.monthly  nice run-parts /etc/cron.monthly
```

Les trois premières lignes définissent les variables qui configurent l'environnement dans lequel les tâches Anacron sont exécutées :

- **SHELL** — environnement shell utilisé pour exécuter des tâches (dans l'exemple, le shell Bash)
- **PATH** — chemins d'accès des programmes exécutables
- **MAILTO** — nom d'utilisateur de l'utilisateur destinataire de la sortie des tâches anacron par courrier électronique

  Si la variable **MAILTO** n'est pas définie (**MAILTO=**), alors le courrier électronique n'est pas envoyé.

Les deux variables suivantes modifient l'heure planifiée des tâches définies :

- **RANDOM_DELAY** — nombre de minutes maximum pouvant être ajoutées à la variable **delay in minutes** (« délai en minutes »), qui est spécifiée pour chaque tâche

  La valeur de délai minimum est définie par défaut sur 6 minutes.

  Par exemple, si **RANDOM_DELAY** est défini sur **12**, alors 6 à 12 minutes sont ajoutées à la variable **delay in minutes** pour chaque tâche dans cet anacrontab particulier. **RANDOM_DELAY** peut également être défini avec une valeur plus basse que **6**, y compris **0**. Si défini sur **0**, aucun délai aléatoire n'est ajouté. Ceci se révèle utile lorsque, par exemple, davantage d'ordinateurs qui partagent une connexion réseau doivent téléverser les mêmes données chaque jour.
- **START_HOURS_RANGE** — intervalle en heures définissant à quel moment les tâches planifiées peuvent être exécutées

  Dans le cas où l'intervalle est manqué, à cause d'une panne de l'alimentation par exemple, les tâches planifiées ne seront pas exécutées ce jour.

Les lignes restantes dans le fichier **/etc/anacrontab** représentent les tâches planifiées et suivent ce format :

```
period in days   delay in minutes   job-identifier   command
```

- **period in days** — fréquence d'exécution des tâches, en jours

  La valeur de la propriété peut être définie en tant qu'entier ou macro (**@daily**, **@weekly**, **@monthly**), où **@daily** possède la même valeur qu'un entier de 1, **@weekly** possède la même que 7, et **@monthly** spécifie que la tâche est exécutée une fois par mois, quelle que soit la longueur du mois.

- **delay in minutes** — nombre de minutes qu'Anacron attend avant d'exécuter la tâche

  La valeur de la propriété est définie en tant qu'entier. Si la valeur est définie sur **0**, aucun délai ne s'appliquera.

- **job-identifier** — nom unique faisant référence à une tâche particulière utilisée dans les fichiers journaux

- **command** — commande devant être exécutée

  La commande peut être une commande telle que **ls /proc >> /tmp/proc** ou une commande qui exécute un script personnalisé.

Toute ligne commençant par le signe dièse (#) correspond à un commentaire et n'est pas traitée.

21.1.3.1. Exemples de tâches Anacron

L'exemple suivant montre un fichier **/etc/anacrontab** simple :

```
SHELL=/bin/sh
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root

# the maximal random delay added to the base delay of the jobs
RANDOM_DELAY=30
# the jobs will be started during the following hours only
START_HOURS_RANGE=16-20

#period in days   delay in minutes   job-identifier   command
1         20    dailyjob      nice run-parts /etc/cron.daily
7         25    weeklyjob     /etc/weeklyjob.bash
@monthly  45    monthlyjob    ls /proc >> /tmp/proc
```

Toutes les tâches définies dans ce fichier **anacrontab** sont retardées de 6 à 30 minutes et peuvent être exécutées entre 16h00 et 20h00.

La première tâche définie est déclenchée quotidiennement entre 16h26 et 16h50 (RANDOM_DELAY se trouve entre 6 et 30 minutes ; la propriété delay in minutes ajoute 20 minutes). La commande spécifiée pour cette tâche exécute tous les programmes présents dans le répertoire **/etc/cron.daily/** à l'aide du script **run-parts** (les scripts **run-parts** acceptent un répertoire en tant qu'argument de ligne de commande et exécutent séquentiellement tous les programmes dans le répertoire). Veuillez consulter la page man **run-parts** pour obtenir davantage d'informations sur le script **run-parts**.

La seconde tâche exécute le script **weeklyjob.bash** dans le répertoire **/etc** une fois par semaine.

La troisième tâche exécute une commande, qui écrit le contenu de **/proc** sur le fichier **/tmp/proc** (**ls /proc >> /tmp/proc**) une fois par mois.

21.1.4. Configuration des tâches Cron

Le fichier de configuration pour les tâches cron est **/etc/crontab**, et peut uniquement être modifié par l'utilitasteur **root**. Le fichier contient ce qui suit :

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/
# For details see man 4 crontabs
# Example of job definition:
# .---------------- minute (0 - 59)
# | .------------- hour (0 - 23)
# | | .---------- day of month (1 - 31)
# | | | .------- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR
sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed
```

Les trois premières lignes contiennent les mêmes définitions de variables qu'un fichier **anacrontab** : **SHELL**, **PATH**, et **MAILTO**. Pour obtenit davantage d'informations sur ces variables, veuillez consulter la Section 21.1.3, « Configurer les tâches Anacron ».

En outre, le fichier peut définir la variable **HOME**. La variable **HOME** définit le répertoire, qui sera utilisé comme répertoire personnel lorsque des commandes ou scripts seront exécutés par la tâche.

Les lignes restantes dans le fichier **/etc/crontab** représentent les tâches planifiées et se trouvent sous le format suivant :

```
minute   hour   day   month   day of week   username   command
```

Ce qui suit permet de définir le moment auquel la tâche doit être exécutée :

- **minute** — un entier de 0 à 59
- **hour** — un entier de 0 à 23
- **day** — un entier de 1 à 31 (doit être un jour valide si le mois est spécifié)
- **month** — un entier de 1 à 12 (ou l'abbrévation du nom en anglais, par exemple « jan » ou « feb »)
- **day of week** — un entier de 0 à 7, où soit 0, soit 7, correspond au dimanche (ou l'abbréviation du nom en anglais du jour de la semaine, par exemple « sun » ou « mon »)

Ce qui suit permet de définir les autres propriétés des tâches :

- **username** — indique l'utilisateur sous lequel les tâches doivent être exécutées.
- **command** — la commande devant être exécutée.

  La commande peut être une commande telle que **ls /proc /tmp/proc**, ou une commande qui exécute un script personnalisé.

Pour toutes les valeurs ci-dessus, un astérisque (« * ») peut être utilisé pour indiquer toutes les valeurs valides. Par exemple, si vous définissez la valeur du mois avec un astérisque, la tâche sera exécutée chaque mois selon les autres valeurs.

Un tiret (« - ») entre nombres entiers indique une plage d'entiers. Par exemple, **1-4** signifie les entiers 1, 2, 3, et 4.

Des valeurs séparées par des virgules (« , ») indiquent une liste. Par exemple, **3,4,6,8** indique exactement ces quatre entiers.

La barre oblique (« / ») peut être utilisée pour spécifier des valeurs étapes. La valeur d'un entier sera ignorée à l'intérieur d'une plage suivant la plage avec les ***/entier***. Par exemple, la valeur minute définie par **0-59/2** indique chaque seconde minute du champ minute. Des valeurs étapes peuvent également être utilisées avec un astérisque. Par exemple, si la valeur du mois est définie par ***/3**, la tâche sera exécutée tous les trois mois.

Toute ligne commençant par le signe dièse (#) correspond à un commentaire et n'est pas traitée.

Les utilisateurs autres que l'utilisateur **root** peuvent configurer les tâches Cron avec l'utilitaire **crontab**. Les crontabs définis utilisateur sont stockés dans le répertoire **/var/spool/cron/** et exécutés comme s'ils étaient exécutés par les utilisateurs qui les ont créés.

Pour créer un crontab en tant qu'utilisateur spécifique, connectez-vous avec ce nom d'utilisateur et saisissez la commande **crontab -e** pour modifier le crontab de l'utilisateur avec l'éditeur spécifié dans la variable d'environnement **VISUAL** ou **EDITOR**. Le fichier utilise le même format que **/etc/crontab**. Lorsque les changements crontab sont enregistrés, le crontab est stocké selon le nom d'utilisateur, puis écrit sur le fichier **/var/spool/cron/*username***. Pour répertorier le contenu du fichier crontab de l'utilisateur, veuillez utiliser la commande **crontab -l**.

Le répertoire **/etc/cron.d/** contient des fichiers qui possèdent la même syntaxe que le fichier **/etc/crontab**. Seul l'utilisateur **root** est autorisé à créer et à modifier des fichiers dans ce répertoire.

NOTE

Le démon cron vérifie si des modifications se produisent dans le fichier **/etc/anacrontab**, le fichier **/etc/crontab**, le répertoire **/etc/cron.d/**, et le répertoire **/var/spool/cron/** chaque minute et tout changement détecté est chargé en mémoire. Ainsi, il n'est pas utile de redémarrer le démon après le changement d'un fichier anacrontab ou crontab.

21.1.5. Contrôle de l'accès à cron

Pour limiter l'accès à Cron, vous pouvez utiliser les fichiers **/etc/cron.allow** et **/etc/cron.deny**. Ces fichiers de contrôle d'accès utilisent le même format avec un nom d'utilisateur sur chaque ligne. N'oubliez pas que les caractères d'espace ne sont pas autorisés dans ces fichiers.

Si le fichier **cron.allow** existe, seuls les utilisateurs répertoriés dans celui-ci seront autorisés à utiliser cron, et le fichier **cron.deny** sera ignoré.

Si le fichier **cron.allow** n'existe pas, les utilisateurs répertoriés dans le fichier **cron.deny** ne seront pas autorisés à utiliser Cron.

Le démon Cron (**crond**) n'a pas besoin d'être redémarré si les fichiers de contrôle d'accès sont modifiés. Les fichiers de contrôle d'accès sont vérifiés chaque fois qu'un utilisateur tente d'ajouter ou de supprimer une tâche Cron.

L'utilisateur **root** peut toujours utiliser Cron, peu importe les noms d'utilisateurs répertoriés dans les fichiers de contrôle d'accès.

Vous pouvez contrôler les accès via les modules PAM (« Pluggable Authentication Modules »). Les paramètres sont stockés dans le fichier **/etc/security/access.conf**. Par exemple, après avoir ajouté la ligne suivante au fichier, aucun utilisateur autre que l'utilisateur **root** ne pourra créer de Crontabs :

```
-:ALL EXCEPT root :cron
```

Les tâches interdites sont journalisées dans un fichier journal approprié ou, lorsque la commande **crontab -e** est utilisée, retournées vers la sortie standard. Pour obtenir davantage d'informations, veuillez consulter la page du manuel **access.conf.5**.

21.1.6. Mettre des tâches Cron sur liste noire et sur liste blanche

La mise sur liste noire et sur liste blanche des tâches est utilisée pour définir les parties d'une tâche qui n'ont pas besoin d'être exécutées. Ceci est utile lors de l'appel d'un script **run-parts** sur un répertoire Cron, tel que **/etc/cron.daily/** : si l'utilisateur ajoute des programmes se trouvant dans le répertoire à la liste noire des tâches, le script **run-parts** n'exécutera pas ces programmes.

Pour définir une liste noire, veuillez créer un fichier **jobs.deny** dans le répertoire à partir duquel les scripts **run-parts** sont exécutés. Par exemple, si vous devez éviter un programme particulier dans **/etc/cron.daily/**, veuillez créer le fichier **/etc/cron.daily/jobs.deny**. Dans ce fichier, spécifiez les noms des programmes à supprimer de l'exécution (seuls les programmes se trouvant dans le même répertoire peuvent être inscrits). Si une tâche exécute une commande qui exécute les programmes du répertoire **/etc/cron.daily/**, comme **run-parts /etc/cron.daily**, les programmes définis dans le fichier **jobs.deny** ne seront pas exécutés.

Pour définir une liste blanche, veuillez créer un fichier **jobs.allow**.

Les principes de **jobs.deny** et **jobs.allow** sont les mêmes que ceux de **cron.deny** et **cron.allow**, ils sont décrits dans la section Section 21.1.5, « Contrôle de l'accès à cron ».

21.2. « AT » ET « BATCH »

Pendant que Cron est utilisé pour planifier des tâches récurrentes, l'utilitaire **At** est utilisé pour planifier une tâche ponctuelle à un moment spécifique et l'utilitaire **Batch** est utilisé pour planifier une tâche ponctuelle à exécuter lorsque la charge moyenne du système tombe sous 0.8.

21.2.1. Installer « At » et « Batch »

Pour déterminer si le paquet at est déjà installé sur votre système, veuillez exécuter la commande suivante :

```
rpm -q at
```

La commande retourne le nom complet du paquet at s'il est déjà installé ou vous notifie que le paquet n'est pas disponible.

Pour installer les paquets, veuillez utiliser la commande **yum** sous le format suivant en tant qu'utilisateur **root** :

```
yum install package
```

Par exemple, pour installer « At » et « Batch », veuillez saisir ce qui suit dans une invite de shell :

```
~]# yum install at
```

Pour obtenir davantage d'informations sur la manière d'installer de nouveaux paquets sur Red Hat Enterprise Linux, veuillez consulter la Section 8.2.4, « Installation de paquets ».

21.2.2. Exécuter le service « At »

Les tâches « At » et « Batch » sont prélevées par le service **atd**. Cette section fournit des informations sur la manière de démarrer, arrêter, et de redémarrer le service **atd**, et montre comment le configurer afin qu'il soit lancé automatiquement lors du démarrage. Pour obtenir davantage d'informations sur la manière de gérer des services système sur Red Hat Enterprise Linux 7 en général, veuillez consulter le Chapitre 9, *Gérer les services avec systemd*.

21.2.2.1. Démarrer et arrêter le service « At »

Pour déterminer si le service est en cours d'exécution, veuillez utiliser la commande suivante :

```
systemctl status atd.service
```

Pour exécuter le service **atd** dans la session actuelle, veuillez saisir ce qui suit dans l'invite de shell en tant qu'utilisateur **root** :

```
systemctl start atd.service
```

Pour configurer le service de manière à ce qu'il soit lancé automatiquement lors du démarrage système, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

```
systemctl enable atd.service
```

NOTE

Il est recommandé de configurer votre système de manière à lancer automatiquement le service **atd** lors du démarrage.

21.2.2.2. Arrêter le service « At »

Pour arrêter le service **atd**, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl stop atd.service
```

Pour empêcher le service d'être lancé automatiquement au démarrage, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

```
systemctl disable atd.service
```

21.2.2.3. Redémarrer le service « At »

Pour redémarrer le service **atd**, veuillez saisir ce qui suit dans une invite de shell en tant qu'utilisateur **root** :

```
systemctl restart atd.service
```

Cette commande arrête le service et le lance à nouveau en une succession rapide.

21.2.3. Configurer une tâche « At »

Pour planifier une tâche ponctuelle à un moment particulier avec l'utilitaire **At**, veuillez procéder comme suit :

1. Sur la ligne de commande, veuillez saisir la commande **at *TIME***, où ***TIME*** représente l'heure à laquelle la commande doit être exécutée.

   L'argument *TIME* peut être défini à l'aide de l'un des formats suivants :

   - ***HH*:*MM*** indique l'heure et la minute exacte ; par exemple, **04:00** signifie 04h00.
   - **midnight** signifie 00h00.
   - **noon** signifie 12h00.
   - **teatime** signifie 16h00.
   - Format ***MONTHDAYYEAR*** ; par exemple, **January 15 2012** signifie le 15ème jour du mois de janvier de l'année 2012. La valeur de l'année est optionnelle.
   - Formats ***MMDDYY***, ***MM/DD/YY***, ou ***MM.DD.YY*** ; par exemple, **011512** pour le 15ème jour du mois de janvier de l'année 2012.
   - **now + *TIME*** où *TIME* est défini en tant que nombre entier et le type de valeur : minutes (« minutes »), heures (« hours »), jour (« days »), ou semaines (« weeks »). Par exemple, **now + 5 days** indique que la commande sera exécutée à la même heure dans cinq jours.

     L'heure doit être spécifiée en premier, suivie de la date optionnelle. Pour obtenir davantage d'informations sur le format de l'heure, veuillez consulter le fichier texte **/usr/share/doc/at-<*version*>/timespec**.

   Si l'heure spécifiée est déjà passée, la tâche sera exécutée au même moment le jour suivant.

2. Définissez les commandes de la tâche dans l'invite **at>** affichée :

   - Veuillez saisir la commande que la tâche devra exécuter et appuyez sur **Entrée**. Optionnellement, répétez l'étape pour fournir des commandes multiples.
   - Saisissez un script shell à l'invite et appuyez sur **Entrée** après chaque ligne du script.

     La tâche utilisera l'ensemble du shell dans l'environnement **SHELL** de l'utilisateur, le shell de connexion de l'utilisateur, ou **/bin/sh** (en fonction de ce qui est trouvé en premier).

3. Une fois terminé, veuillez appuyer sur **Ctrl**+**D** sur une ligne vide pour sortir de l'invite de commande.

Si l'ensemble des commande ou le script tente d'afficher des informations sur la sortie standard, la sortie sera envoyée par courrier électronique à l'utilisateur.

Pour afficher la liste des tâches en attente, utilisez la commande **atq**. Veuillez consulter la Section 21.2.5, « Afficher les tâches en attente » pour obtenir davantage d'informations.

Vous pouvez également limiter l'utilisation de la commande **at**. Pour obtenir davantage d'informations, veuillez consulter la Section 21.2.7, « Contrôle de l'accès à « At » et « Batch » ».

21.2.4. Configurer une tâche « Batch »

L'application **Batch** exécute des tâches ponctuelles définies lorsque la charge moyenne du système passe en-dessous de 0.8.

Pour définir une tâche « Batch », veuillez procéder comme suit :

1. Sur la ligne de commande, saisir la commande **batch**.
2. Définissez les commandes de la tâche dans l'invite **at>** affichée :
   - Veuillez saisir la commande que la tâche devra exécuter et appuyez sur **Entrée**. Optionnellement, répétez l'étape pour fournir des commandes multiples.
   - Saisissez un script shell à l'invite et appuyez sur **Entrée** après chaque ligne du script.

     Si un script est saisi, la tâche utilise l'ensemble du shell dans l'environnement **SHELL** de l'utilisateur, le shell de connexion de l'utilisateur, ou **/bin/sh** (en fonction de ce qui est trouvé en premier).
3. Une fois terminé, veuillez appuyer sur **Ctrl**+**D** sur une ligne vide pour sortir de l'invite de commande.

Si l'ensemble des commande ou le script tente d'afficher des informations sur la sortie standard, la sortie sera envoyée par courrier électronique à l'utilisateur.

Pour afficher la liste des tâches en attente, utilisez la commande **atq**. Veuillez consulter la Section 21.2.5, « Afficher les tâches en attente » pour obtenir davantage d'informations.

Vous pouvez également limiter l'utilisation de la commande **batch**. Pour obtenir davantage d'informations, veuillez consulter la Section 21.2.7, « Contrôle de l'accès à « At » et « Batch » ».

21.2.5. Afficher les tâches en attente

Pour afficher les tâches **At** et **Batch** en attente, exécutez la commande **atq**. La commande **atq** affiche une liste des tâches en attente, avec chaque tâche sur une ligne séparée. Chaque ligne suit le numéro de la tâche, la date, l'heure, la classe de la tâche, et le format du nom d'utilisateur. Les utilisateur peuvent uniquement afficher leurs propres tâches. Si l'utilisateur **root** exécute la commande **atq**, toutes les tâches de tous les utilisateurs seront affichées.

21.2.6. Options de ligne de commande supplémentaires

Les options de ligne de commande supplémentaires des commandes **at** et **batch** incluent :

Tableau 21.1. Options de ligne de commande at et batch

Option	Description
-f	Lit les commandes ou le script shell depuis un fichier au lieu de les spécifier à l'invite de commande.
-m	Envoie un courrier électronique à l'utilisateur une fois la tâche accomplie.
-v	Affiche l'heure à laquelle la tâche sera exécutée.

21.2.7. Contrôle de l'accès à « At » et « Batch »

Vous pouvez restreindre l'accès aux commandes **at** et **batch** à l'aide des fichiers **/etc/at.allow** et **/etc/at.deny**. Ces fichiers de contrôle d'accès utilisent le même format en définissant un nom d'utilisateur sur chaque ligne. N'oubliez pas que les espaces ne sont pas autorisés dans ces fichiers.

Si le fichier **at.allow** existe, seuls les utilisateurs répertoriés dans celui-ci seront autorisés à utiliser **at** ou **batch**, et le fichier **at.deny** sera ignoré.

Si le fichier **at.allow** n'existe pas, les utilisateurs répertoriés dans le fichier **at.deny** ne seront pas autorisés à utiliser **at** ou **batch**.

Le démon **at** (**atd**) n'a pas besoin d'être redémarré si les fichiers de contrôle d'accès sont modifiés. Les fichiers de contrôle d'accès sont lus chaque fois qu'un utilisateur tente d'exécuter les commandes **at** ou **batch**.

L'utilisateur **root** peut toujours exécuter les commandes **at** et **batch**, indépendamment du contenu des fichiers de contrôle d'accès.

21.3. RESSOURCES SUPPLÉMENTAIRES

Pour en savoir plus sur la configuration de tâches automatisées, veuillez consulter la documentation installée suivante :

- La page man de **cron(8)** offre une vue d'ensemble de Cron.
- Les pages man de **crontab** dans les sections 1 et 5 :
  - La page du manuel dans la section 1 contient une vue d'ensemble du fichier **crontab**.
  - La page man dans la section 5 contient le format du fichier et quelques exemples d'entrées.
- La page man de **anacron(8)** contient une vue d'ensemble d'Anacron.
- La page man de **anacrontab(5)** contient une vue d'ensemble du fichier **anacrontab**.
- La page man de **run-parts(4)** contient une vue d'ensemble du script **run-parts**.
- **/usr/share/doc/at-*version*/timespec** contient des informations détaillées sur les valeurs de temps pouvant être utilisées dans les définitions de tâches Cron.
- La page man de **at** contient des descriptions des commandes **at** et **batch** et de leurs options de ligne de commande.

CHAPITRE 22. ABRT (AUTOMATIC BUG REPORTING TOOL)

22.1. INTRODUCTION À ABRT

L'outil **Automatic Bug Reporting Tool**, généralement abrégé par ses initiales, **ABRT**, est un ensemble d'outils conçus pour aider les utilisateurs à détecter et rapporter les incidents d'application. Son but principal est de faciliter le processus de rapport de problèmes et de trouver des solutions. Dans ce contexte, la solution peut être un ticket Bugzilla, un article de la base de connaissances, ou une suggestion pour mettre à jour un paquet sous une version contenant un correctif.

ABRT est composé du démon `abrtd` et d'un nombre de services et utilitaires système pour le traitement, l'analyse et le rapport des problèmes détectés. Le démon est exécuté silencieusement en arrière-plan la plupart du temps et entre en action lorsqu'un incident d'application survient ou lorsqu'un oops de noyau se produit. Le démon collecte ensuite les données problématiques correspondantes, telles que le fichier cœur s'il en existe un, les paramètres de ligne de commande de l'application tombée en panne, ainsi que d'autres données utiles.

Actuellement, **ABRT** prend en charge la détection d'incidents dans les applications écrites dans les langages de programmation C, C++, Java, Python, et Ruby, ainsi que les incidents X.Org, les oops de noyau et les paniques de noyau. Veuillez consulter la Section 22.4, « Détection de problèmes logiciels » pour obtenir des informations plus détaillées sur les types d'échecs et d'incidents pris en charge, ainsi que sur la manière par laquelle les différents types d'incidents sont détectés.

Les problèmes identifiés peuvent être rapportés sur un outil de suivi de problèmes distant, et les rapports peuvent être configurés de manière à se produite automatiquement lorsqu'un problème est détecté. Les données problématiques peuvent également être stockées localement ou sur un système dédié, puis vérifiées, rapportées, et supprimées manuellement par l'utilisateur. Les outils de rapport peuvent envoyer les données problématiques sur la base de données Bugzilla ou sur le site web du support technique de Red Hat, « Red Hat Technical Support » (RHTSupport). Ces outils peuvent également les téléverser à l'aide de **FTP** ou de **SCP**, les envoyer en tant que courrier électronique, ou les écrire sur un fichier.

Le composant **ABRT** qui gère les données problématiques existantes (contrairement à la création de nouvelles données problématiques par exemple) fait partie d'un projet séparé, nommé **libreport**. La bibliothèque **libreport** fournit un mécanisme générique pour l'analyse et le rapport des problèmes, et est également utilisée par d'autres applications que **ABRT**. Cependant, l'opération et la configuration d'**ABRT** et de **libreport** sont étroitement intégrées. Il est traité de celles-ci comme d'une seule entité dans ce document.

22.2. INSTALLER ABRT ET LANCER SES SERVICES

Dans le but d'utiliser **ABRT**, assurez-vous que le paquet abrt-desktop ou abrt-cli soit installé sur votre système. Le paquet abrt-desktop fournit une interface utilisateur graphique pour **ABRT**, tandis que le paquet abrt-cli contient un outil pour utiliser **ABRT** sur la ligne de commande. Vous pouvez également installer les deux. Les flux de travail de l'interface utilisateur graphique et de l'outil en ligne de commande **ABRT** sont similaires quant aux procédures et le même schéma est suivi.

AVERTISSEMENT

Veuillez remarquer que l'installation des paquets **ABRT** remplace le fichier **/proc/sys/kernel/core_pattern**, qui contient un modèle utilisé pour nommer les fichiers core-dump. Le contenu de ce fichier sera remplacé sur :

```
|/usr/libexec/abrt-hook-ccpp %s %c %p %u %g %t e
```

Veuillez consulter la Section 8.2.4, « Installation de paquets » pour obtenir des informations générales sur l'installation de paquets avec le gestionnaire de paquets **Yum**.

22.2.1. Installer l'interface utilisateur graphique ABRT

ABRT - *interface utilisateur graphique* - offre une interface frontale facile à utiliser pour travailler sur un environnement de bureau. Vous pouvez installer le paquet requis en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install abrt-desktop
```

Lors de l'installation, la mini-application de notification **ABRT** est configurée pour démarrer automatiquement lorsque votre session de bureau graphique est lancée. Vous pouvez vérifier si la mini-application **ABRT** est en cours d'exécution en passant la commande suivante dans une fenêtre de terminal :

```
~]$ ps -el | grep abrt-applet
0 S   500  2036  1824  0  80   0 - 61604 poll_s ?        00:00:00 abrt-
applet
```

Si la mini-application n'est pas en cours d'exécution, vous pouvez la lancer manuellement sur votre session de bureau en cours en exécutant le programme **abrt-applet** :

```
~]$ abrt-applet &
[1] 2261
```

22.2.2. Installer ABRT pour l'interface en ligne de commande

L'*interface de ligne de commande* est utile sur les machines sans affichage, les systèmes distants connectés par réseau, ou avec les scripts. Vous pouvez installer le paquet requis en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install abrt-cli
```

22.2.3. Installer les outils ABRT supplémentaires

Pour recevoir des notifications par courrier électronique sur les incidents détectés par **ABRT**, le paquet libreport-plugin-mailx devra être installé. Vous pouvez l'installer en exécutant la commande suivante en tant qu'utilisateur **root** :

```
~]# yum install libreport-plugin-mailx
```

Par défaut les notifications sont envoyées à l'utilisateur **root** de la machine locale. La destination du courrier électronique peut être configurée dans le fichier **/etc/libreport/plugins/mailx.conf**.

Pour que les notifications soient affichées dans votre console au moment de la connexion, veuillez également installer le paquet abrt-console-notification.

ABRT peut détecter analyser et rapporter divers types d'échecs de logiciels. Par défaut, **ABRT** est installé avec la prise en charge des types d'échecs les plus courants, tel que les échecs des applications C et C++. La prise en charge d'autres types d'échecs est fournie par des paquets indépendants. Par exemple, pour installer la prise en charge de la détection d'exceptions dans les applications écrites à l'aide de Java, veuillez exécuter la commande en tant qu'utilisateur **root** :

```
~]# yum install abrt-java-connector
```

Veuillez consulter Section 22.4, « Détection de problèmes logiciels » pour obtenir une liste des langages et des projets logiciels pris en charge par **ABRT**. La section inclut également une liste de tous les paquets correspondants qui permettent la détection des divers types d'échecs.

22.2.4. Lancer les services ABRT

Le démon **abrtd** est configuré pour être lancé lors du démarrage. Vous pouvez utiliser la commande suivante pour vérifier son statut actuel :

```
~]$ systemctl is-active abrtd.service
active
```

Si la commande **systemctl** retourne **inactive** (inactif) ou **unknown** (inconnu), alors le démon n'est pas en cours d'exécution. Vous pouvez le démarrer pour la session en cours en saisissant la commande suivante en tant qu'utilisateur **root** :

```
~]# systemctl start abrtd.service
```

Similairement, vous pouvez suivre les mêmes étapes pour vérifier le statut et pour démarrer les services qui gèrent les divers types d'échecs. Par exemple, assurez-vous que le service **abrt-ccpp** est en cours d'exécution si vous souhaitez qu'**ABRT** détecte les incidents C ou C++. Veuillez consulter la Section 22.4, « Détection de problèmes logiciels » pour obtenir une liste de tous les services de détection **ABRT** disponibles et de leurs paquets respectifs.

À l'exception des services **abrt-vmcore** et **abrt-pstoreoops**, qui sont uniquement lancés lorsqu'une panique ou un oops de noyau se produit, tous les services **ABRT** sont automatiquement activés et lancés au moment du démarrage lorsque leurs paquets respectifs sont installés. Vous pouvez activer ou désactiver un service **ABRT** en utilisant l'utilitaire **systemctl** comme décrit dans le Chapitre 9, *Gérer les services avec systemd*.

22.2.5. Tester la détection d'incidents ABRT

Pour tester si **ABRT** fonctionne correctement, veuillez utiliser la commande **kill** pour envoyer le signal SEGV pour arrêter un processus. Par exemple, lancez un processus **sleep** et arrêtez-le avec la commande **kill** de la manière suivante :

```
~]$ sleep 100 &
[1] 2823
~]$ kill -s SIGSEGV 2823
```

ABRT détecte un incident peu après avoir exécuté la commande **`kill`**, et si une session graphique est en cours d'exécution, l'utilisateur sera notifié du problème détecté par la mini-application de notification de la GUI. Dans l'environnement de ligne de commande, vous pouvez vérifier si l'incident a été détecté en exécutant la commande **`abrt-cli list`** ou en examinant le vidage du noyau créé dans le répertoire **`/var/tmp/abrt/`**. Veuillez consulter la Section 22.5, « Gestion des problèmes détectés » pour obtenir davantage d'informations sur les manières de travailler avec les incidents détectés.

22.3. CONFIGURER ABRT

Le cycle de vie d'un problème est dirigé par des *événements* dans **ABRT**. Par exemple :

- Événement #1 — un répertoire de données des problèmes (« Problem-data directory ») est créé.
- Événement #2 — les données des problèmes sont analysées.
- Événement #3 — le problème est rapporté sur Bugzilla.

Lorsqu'un problème est détecté, **ABRT** le compare avec toutes les données des problèmes existants et détermine si le même problème a déjà été enregistré. Si c'est le cas, les données existantes du problème sont mises à jour, et le problème le plus récent (dupliqué) n'est pas ré-enregistré. Si le problème n'est pas reconnu par **ABRT**, un **répertoire de données des problèmes** est créé. Un répertoire de données des problèmes consiste habituellement en fichiers : **`analyzer`**, **`architecture`**, **`coredump`**, **`cmdline`**, **`executable`**, **`kernel`**, **`os_release`**, **`reason`**, **`time`**, et **`uid`**.

D'autres fichiers, tels que **`backtrace`**, peuvent être créés pendant l'analyse du problème, selon la méthode de l'analyseur utilisée et ses paramètres de configuration. Chacun de ces fichiers contient des informations spécifiques sur le système et le problème. Par exemple, le fichier **`kernel`** enregistre la version du noyau tombé en panne.

Une fois que le répertoire « Problem-data » est créé et que les données des problèmes sont collectées, le problème peut être traité en utilisant l'interface utilisateur graphique **ABRT**, ou l'utilitaire **abrt-cli** sur la ligne de commande. Veuillez consulter la Section 22.5, « Gestion des problèmes détectés » pour obtenir des informations supplémentaires sur les outils **ABRT** fournis pour travailler avec les problèmes enregistrés.

22.3.1. Configurer des événements

Les événements **ABRT** utilisent des *greffons* pour effectuer les opérations de rapport. Les greffons sont des utilitaires compacts appelés par les événements pour traiter le contenu des répertoires de données des problèmes. À l'aide des greffons, **ABRT** est capable de rapporter des problèmes sur diverses destinations, et presque toutes les destinations de rapports requièrent une certaine configuration. Par exemple, Bugzilla requiert un nom d'utilisateur, un mot de passe et un URL pointant vers une instance du service Bugzilla.

Certains détails de configuration peuvent avoir des valeurs par défaut (comme l'URL de Bugzilla), mais ce n'est pas le cas pour tous (par exemple un nom d'utilisateur). **ABRT** recherche ces paramètres dans les fichiers de configuration, tels que **`report_Bugzilla.conf`**, dans les répertoires **`/etc/libreport/events/`** ou **`$HOME/.cache/abrt/events/`** pour les paramètres système globaux ou les paramètres spécifiques à l'utilisateur, respectivement. Les fichiers de configuration contiennent des paires de directives et de valeurs.

Ces fichiers correspondent au minimum absolu nécessaire à l'exécution d'événements et au traitement des répertoires de données des problèmes. Les outils **gnome-abrt** et **abrt-cli** lisent les données de configuration de ces fichiers et les passent aux événements qu'ils exécutent.

Des informations supplémentaires sur les événements (comme les descriptions, les noms, les types de paramètres pouvant être passés en tant que variables d'environnements et autres propriétés) sont stockées dans les fichiers ***event_name*.xml** du répertoire **/usr/share/libreport/events/**. Ces fichiers sont utilisés par **gnome-abrt** et **abrt-cli** pour rendre l'interface utilisateur plus conviviale. Ne modifiez pas ces fichiers à moins de souhaiter modifier l'installation standard. Si vous comptez modifier celle-ci, veuillez copier le fichier à modifier sur le répertoire **/etc/libreport/events/** et modifiez le nouveau fichier. Ces fichiers peuvent contenir les informations suivantes :

- un nom d'événement convivial et une description (Bugzilla, rapport au suivi de bogues Bugzilla),
- une liste d'éléments d'un répertoire de données de problèmes dont on a besoin pour que l'événement réussisse,
- une sélection par défaut et une sélection obligatoire d'éléments à envoyer ou à ne pas envoyer,
- si la GUI doit demander de vérifier les données,
- quelles options de configuration existent, leurs types (chaîne, booléen, etc.), valeur par défaut, chaîne d'invite, etc. ; ceci permet à la GUI de créer les boîtes de dialogue appropriées.

Par exemple, l'événement **report_Logger** accepte un nom de fichier de sortie en tant que paramètre. En utilisant le fichier ***event_name*.xml** respectif, l'interface utilisateur graphique **ABRT** détermine quels paramètres peuvent être spécifiés pour un événement sélectionné et permet à l'utilisateur de définir les valeurs de ces paramètres. Les valeurs sont enregistrées par l'interface utilisateur graphique **ABRT** et réutilisées lors des invocations ultérieures de ces événements. Remarquez que l'interface utilisateur graphique **ABRT** enregistre les options de configuration en utilisant l'outil **GNOME Keyring**, et en les passant aux événements, remplace les données provenant des fichiers de configuration en texte.

Pour ouvrir la fenêtre de **Configuration** graphique, veuillez choisir **Automatic Bug Reporting Tool** → **Préférences** à partir d'une instance en cours d'exécution de l'application **gnome-abrt**. La fenêtre affiche une liste d'événements qui peuvent être sélectionnés pendant le processus de rapport lors de l'utilisation de la GUI. Lorsque vous sélectionnez l'un des événements configurables, vous pourrez cliquer sur le bouton **Configurer** et modifier les paramètres de cet événement.

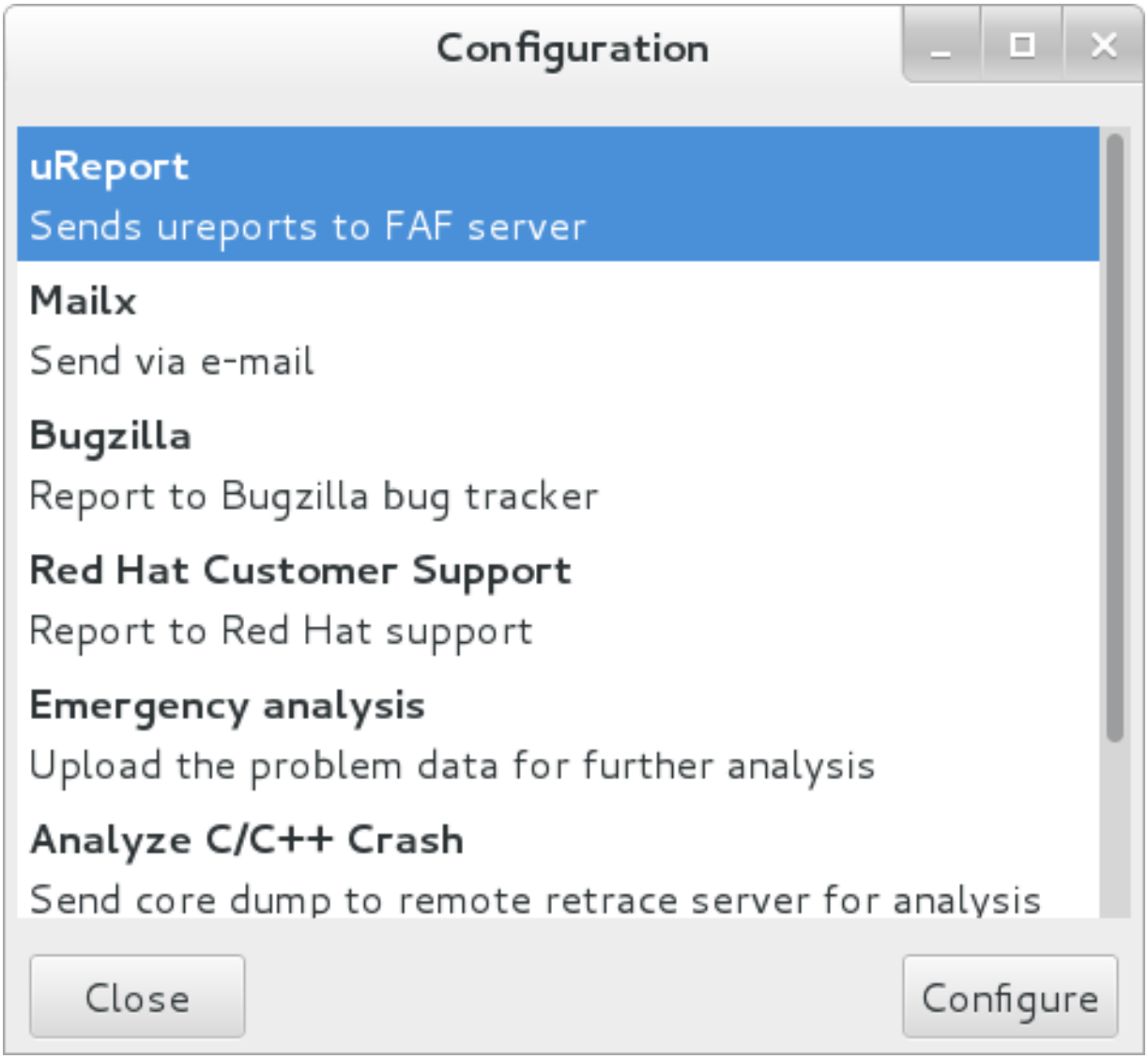
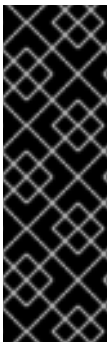


Figure 22.1. Configurer des événements ABRT



IMPORTANT

Tous les fichiers dans la hiérarchie du répertoire **`/etc/libreport/`** sont lisibles globalement et sont conçus pour être utilisés en tant que paramètres globaux. Ainsi, il n'est pas recommandé de stocker les noms d'utilisateurs, mots de passe, ou toutes autres données sensibles dedans. Les paramètres propres à l'utilisateur (paramétrés dans l'application de la GUI et uniquement lisible par le propriétaire de **`$HOME`**) sont stockés de manière sécurisée dans **GNOME Keyring** ; il peuvent également être stockés dans un fichier de configuration texte dans **`$HOME/.abrt/`** pour une utilisation avec **`abrt-cli`**.

Le tableau suivant affiche une sélection des événements d'analyse, de collecte et de rapport par défaut offerts par l'installation standard d'**ABRT**. Le tableau répertorie chaque nom, identificateur et fichier de configuration d'événement du répertoire **`/etc/libreport/events.d/`**, ainsi qu'une brève description. Remarquez qu'alors que les fichiers de configuration utilisent des identificateurs d'événements, l'interface utilisateur graphique **ABRT** fait référence aux événements individuels en utilisant leurs noms. Remarquez également que tous les événements ne peuvent pas être paramétrés à l'aide de la GUI. Pour obtenir des informations sur la manière de définit un événement personnalisé, veuillez consulter la Section 22.3.2, « Créer des événements personnalisés ».

Tableau 22.1. Événements ABRT standard

Nom	Identificateur et fichier de configuration	Description
uReport	report_uReport	Téléverse un µReport sur un serveur **FAF**.
Mailx	report_Mailx **mailx_event.conf**	Envoie le rapport problématique via l'utilitaire **Mailx** vers une adresse électronique spécifiée.
Bugzilla	report_Bugzilla **bugzilla_event.conf**	Rapporte le problème à l'installation spécifiée de l'outil de suivi de bogues **Bugzilla**.
Red Hat Customer Support	report_RHTSupport **rhtsupport_event.conf**	Rapporte le problème au système de Support technique de Red Hat.
Analyse d'un incident C ou C++	analyze_CCpp **ccpp_event.conf**	Envoie le vidage du noyau dans un serveur retrace distant pour analyses, ou effectue une analyse locale si celle à distance échoue.
Téléverseur de rapports	report_Uploader **uploader_event.conf**	Téléverse une archive tarball (**.tar.gz**) avec des données problématiques sur la destination choisie, en utilisant le protocole **FTP** ou **SCP**.
Analyser le vidage mémoire	analyze_VMcore **vmcore_event.conf**	Exécute **GDB** (le débogueur GNU) sur les données problématiques d'un oops de noyau et génère un **backtrace** du noyau.
Débogueur GNU local	analyze_LocalGDB **ccpp_event.conf**	Exécute **GDB** (le débogueur GNU) sur les données problématiques d'une application et génère un **backtrace** du programme.
Collect .xsession-errors	analyze_xsession_errors **ccpp_event.conf**	Enregistre les lignes utiles du fichier **~/.xsession-errors** sur le rapport de problème.
Logger	report_Logger **print_event.conf**	Crée un rapport sur le problème et l'enregistre sur un fichier local spécifié.
Kerneloops.org	report_Kerneloops **koops_event.conf**	Envoie un problème de noyau sur l'outil de suivi des oops sur kerneloops.org.

22.3.2. Créer des événements personnalisés

Chaque événement est défini par une structure de règle dans un fichier de configuration respectif. Les fichiers de configuration sont habituellement stockés dans le répertoire **/etc/libreport/events.d/**. Ces fichiers de configuration sont chargés par le fichier de configuration principal, **/etc/libreport/report_event.conf**. Vous n'avez nul besoin de modifier les fichiers de

configuration par défaut car abr exécutera des scripts qui se trouvent dans **/etc/libreport/events.d/**. Ce fichier accepte les métacaractères shell (*, $, ?, etc.) et interprète les chemins relatifs en fonction de son emplacement.

Chaque *règle* commence par une ligne débutant par un caractère autre qu'un espace, et toute ligne suivante commençant par le caractère **espace** ou par le caractère de **Tabulation** est considérée comme faisant partie de cette règle. Chaque *règle* consiste de deux parties, la partie *condition* et la partie *programme*. La partie condition contient des conditions sous l'une des formes suivantes :

- *VAR*=*VAL*
- *VAR*!=*VAL*
- *VAL*~=*REGEX*

où :

- *VAR* est soit le mot clé **EVENT** (événement) ou le nom d'un élément de répertoire de données problématiques (tel qu'**executable**, **package**, **hostname**, etc.),
- *VAL* est soit le nom d'un événement ou d'un élément de données problématiques, et
- *REGEX* est une expression rationnelle.

La partie du programme est composée des noms de programmes et d'un code interprétable par shell. Si toutes les conditions de la partie Conditions sont valides, le programme sera exécuté dans le shell. Ci-dessous figure un exemple d'*événement* :

```
EVENT=post-create date > /tmp/dt
        echo $HOSTNAME `uname -r`
```

Cet événement remplacera le contenu du fichier **/tmp/dt** par l'heure et la date actuelles et imprimera le nom d'hôte de la machine et sa version de noyau dans la sortie standard.

Voici un exemple d'événement plus complexe, qui se trouve être l'un des événements prédéfinis. Celui-ci enregistre les lignes utiles du fichier **~/.xsession-errors** sur le rapport de problème d'un problème pour lequel le service **abrt-ccpp** a été utilisé, à condition que des bibliothèques X11 aient été chargées au moment où l'application a échoué :

```
EVENT=analyze_xsession_errors analyzer=CCpp dso_list~=.*/libX11.*
        test -f ~/.xsession-errors || { echo "No ~/.xsession-errors";
exit 1; }
        test -r ~/.xsession-errors || { echo "Can't read ~/.xsession-
errors"; exit 1; }
        executable=`cat executable` &&
        base_executable=${executable##*/} &&
        grep -F -e "$base_executable" ~/.xsession-errors | tail -999
>xsession_errors &&
        echo "Element 'xsession_errors' saved"
```

L'ensemble d'événements possibles n'est pas encore définitif. Les administrateurs systèmes peuvent ajouter des événements en fonction de leurs besoins dans le répertoire **/etc/libreport/events.d/**.

Actuellement, les noms d'événements suivants sont fournis avec les installations standards d'**ABRT** et de **libreport** :

post-create

Cet événement est exécuté par **abrtd** pour traiter les nouveaux répertoires de données problématiques créés. Lorsque l'événement **post-create** est exécuté, **abrtd** vérifie si les nouvelles données problématiques correspondent à celles des répertoires de problèmes déjà existants. Si un tel répertoire de problèmes existe, les nouvelles données problématiques seront supprimées. Notez que si le script qui se trouve dans la définition de l'événement **post-create** existe avec une valeur non nulle, **abrtd** interrompera le processus et abandonnera les données du problème.

notify, **notify-dup**

L'événement **notify** est exécuté une fois que **post-create** est terminé. Lorsque l'événement est exécuté, l'utilisateur sait que le problème requiert son attention. **notify-dup** est similaire, sauf qu'il est utilisé pour des instances dupliquées d'un même problème.

analyze_*name_suffix*

où *name_suffix* est la partie remplaçable du nom de l'événement. Cet événement est utilisé pour traiter les données collectées. Par exemple, l'événement **analyze_LocalGDB** utilise GNU Debugger (**GDB**) pour traiter l'image mémoire d'une application et produire un backtrace de l'incident.

collect_*name_suffix*

…où *name_suffix* est la partie ajustable du nom de l'événement. Cet événement est utilisé pour collecter des informations supplémentaires sur les problèmes.

report_*name_suffix*

…où *name_suffix* est la partie ajustable du nom de l'événement. Cet événement est utilisé pour rapporter un problème.

22.3.3. Paramétrer les rapports automatiques

ABRT peut être configuré pour envoyer des rapports initiaux anonymes, ou *µReports*, de toute panne ou problème détecté automatiquement, sans interaction de la part de l'utilisateur. Lorsque les rapports automatiques sont activés, les µReport, qui sont normalement envoyés au début du processus de rapport d'incident, sont envoyés immédiatement après la détection d'un incident. Ceci permet d'éviter les cas de support dupliqués, basés sur des incidents identiques. Pour activer la fonctionnalité de rapport automatique, veuillez passer la commande suivante en tant qu'utilisateur **root** :

```
~]# abrt-auto-reporting enabled
```

La commande ci-dessus définit la directive **AutoreportingEnabled** (rapports automatiques activés) du fichier de configuration **/etc/abrt/abrt.conf** sur « **yes** ». Ce paramètre global s'applique à tous les utilisateurs du système. Remarquez qu'en activant cette option, les rapports automatiques seront également activés dans l'environnement de bureau graphique. Pour activer les rapport automatiques dans l'interface utilisateur graphique **ABRT** uniquement, veuillez faire basculer l'option **Envoyer des uReport automatiquement** sur **YES** dans la fenêtre **Configuration des rapports de problèmes**. Pour ouvrir cette fenêtre, veuillez choisir **Automatic Bug Reporting Tool** → **ABRT Configuration** à partir d'une instance de l'application **gnome-abrt** en cours d'exécution. Pour lancer l'application, veuillez vous rendre sur **Applications** → **Sundry** → **Automatic Bug Reporting Tool**.

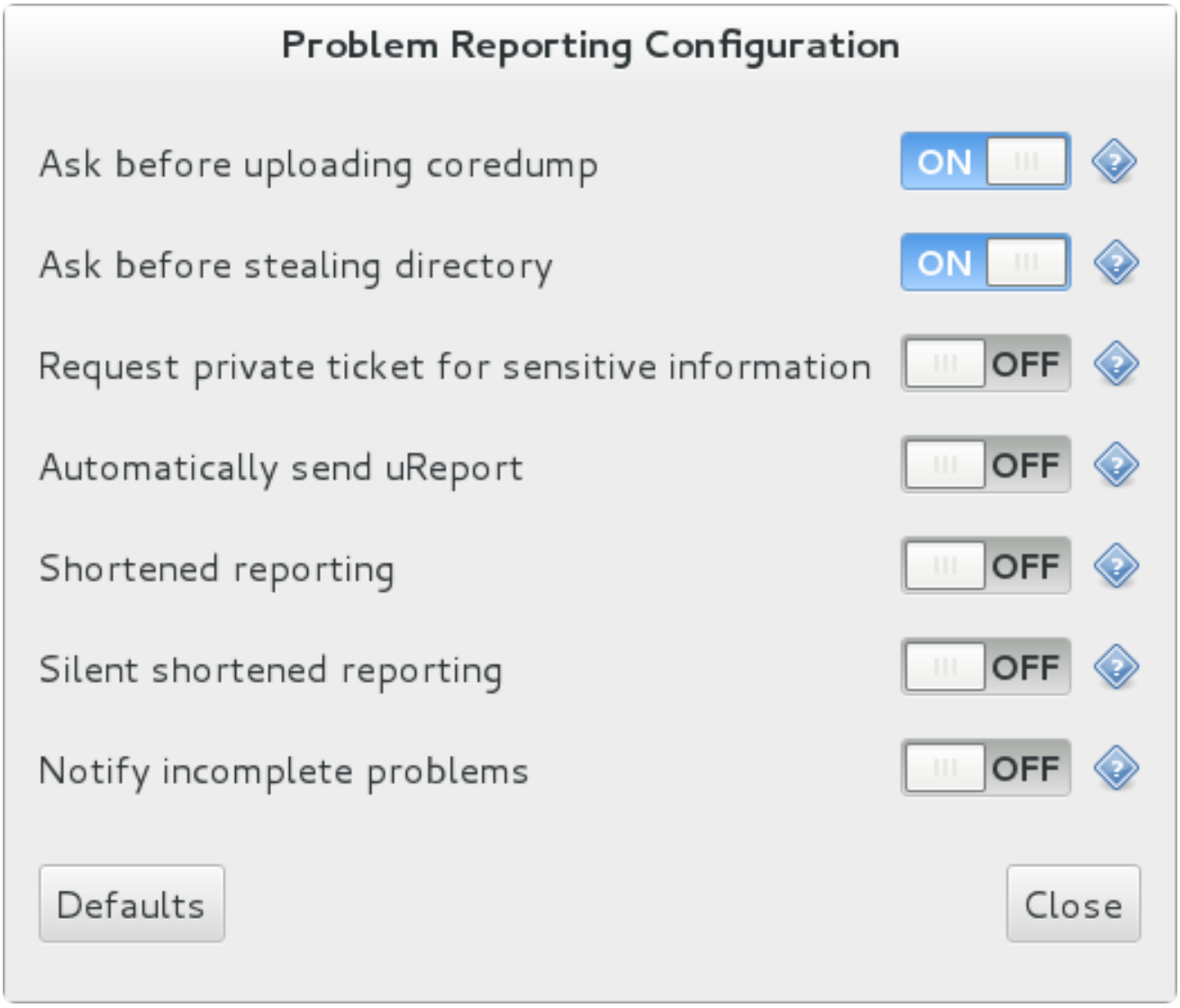
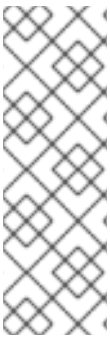


Figure 22.2. Configuration des rapports de problèmes ABRT

Lors de la détection d'un incident, par défaut, **ABRT** soumet un µReport avec des informations de base sur le problème au serveur **ABRT** de Red Hat. Le serveur détermine si le problème est connu et offre une courte description du problème ainsi qu'un URL du cas rapporté si celui-ci est connu, ou invite l'utilisateur à le rapporter s'il n'est pas connu.



NOTE

Un µReport (microrapport) est un objet JSON représentant un problème, tel qu'un incident binaire ou un oops de noyau. Ces rapports sont conçus de manière à être brefs, lisibles en machine, et complètement anonymes, ce qui explique pourquoi ils peuvent être utilisés pour des rapports automatisés. Les µReports permettent de se tenir au fait des incidences de bogues, mais ne fournissent pas suffisamment d'informations pour que les ingénieurs puissent corriger le bogue. Il fut un rapport de bogue complet pour pouvoir ouvrir un dossier de support technique.

Pour changer le comportement par défaut de l'outil de rapports automatique afin de ne plus envoyer de rapports µReport, modifiez la valeur de la directive **`AutoreportingEvent`** du fichier de configuration **`/etc/abrt/abrt.conf`** de manière à pointer vers un événement **ABRT** différent. Veuillez consulter le Tableau 22.1, « Événements ABRT standard » pour un aperçu des événements standard.

22.4. DÉTECTION DE PROBLÈMES LOGICIELS

ABRT est capable de détecter, analyser et traiter les incidents d'applications écrites dans divers langages de programmation. De nombreux paquets offrant la prise en charge de détection d'incidents divers sont installés automatiquement lorsque l'un des paquets principaux **ABRT** (abrt-desktop, abrt-cli) est installé. Veuillez consulter la Section 22.2, « Installer ABRT et lancer ses services » pour obtenir des instructions sur la manière d'installer **ABRT**. Veuillez consulter le tableau ci-dessous pour obtenir une liste des types d'incidents pris en charge et de leurs paquets respectifs.

Tableau 22.2. Langages de programmation et projets logiciels pris en charge

Langage/projet	Paquet
C ou C++	abrt-addon-ccpp
Python	abrt-addon-python
Ruby	rubygem-abrt
Java	abrt-java-connector
X.Org	abrt-addon-xorg
Linux (oops de noyau)	abrt-addon-kerneloops
Linux (panique de noyau)	abrt-addon-vmcore
Linux (stockage persistant)	abrt-addon-pstoreoops

22.4.1. Détection d'incidents C et C++

Le service **abrt-ccpp** installe son propre gestionnaire core-dump, qui, lorsque démarré, remplace la valeur par défaut de la variable **core_pattern** du noyau, afin que les incidents C et C++ soient gérés par **abrtd**. Si vous arrêtez le service **abrt-ccpp**, l'ancienne valeur de **core_pattern** sera réintégrée.

Par défaut, le fichier **/proc/sys/kernel/core_pattern** contient la chaîne **core**, ce qui signifie que le noyau produit des fichiers avec le préfixe **core.** dans le répertoire actuel du processus en panne. Le service **abrt-ccpp** remplace le fichier **core_pattern** afin qu'il contienne la commande suivante :

```
|/usr/libexec/abrt-hook-ccpp %s %c %p %u %g %t e
```

This command instructs the kernel to pipe the core dump to the **abrt-hook-ccpp** program, which stores it in **ABRT**'s dump location and notifies the **abrtd** daemon of the new crash. It also stores the following files from the **/proc/*PID*/** directory (where *PID* is the ID of the crashed process) for debugging purposes: **maps**, **limits**, **cgroup**, **status**. See proc(5) for a description of the format and the meaning of these files.

22.4.2. Détection des exceptions Python

Le paquet abrt-addon-python installe un gestionnaire d'exceptions personnalisé pour les applications Python. L'interprète Python importe ensuite automatiquement le fichier **abrt.pth** installé dans

/usr/lib64/python2.7/site-packages/, qui importera à son tour le fichier **abrt_exception_handler.py**. Ceci remplace le fichier Python par défaut **sys.excepthook** par un gestionnaire personnalisé, qui transfère les exceptions non gérées au démon **abrtd** via son API Socket.

Pour désactiver l'importation automatique de modules spécifiques aux sites, et empêcher ainsi le gestionnaire d'exceptions personnalisé **ABRT** d'être utilisé lors de l'exécution d'une application Python, veuillez passer l'option **-S** à l'interprète Python :

```
~]$ python -S file.py
```

Dans la commande ci-dessus, remplacez *file.py* par le nom du script Python que vous souhaitez exécuter sans utilisation de modules spécifiques aux sites.

22.4.3. Détection des exceptions Ruby

Le paquet rubygem-abrt enregistre un gestionnaire personnalisé en utilisant la fonctionnalité **at_exit**, qui est exécutée lorsqu'un programme se termine. Ceci permet de vérifier les exceptions non gérées possibles. Chaque fois qu'une exception non gérée est capturée, le gestionnaire **ABRT** prépare un rapport de bogue, qui pourra être soumis à Red Hat Bugzilla à l'aide des outils **ABRT** standard.

22.4.4. Détection des exceptions Java

Le Connector Java **ABRT** est un agent JVM qui rapporte les exceptions Java non interceptées au démon **abrtd**. L'agent enregistre plusieurs rappels d'événement JVMTI et doit être chargé dans JVM à l'aide du paramètre de ligne de commande **-agentlib**. Remarquez que le traitement des rappels enregistrés affecte négativement les performances de l'application. Pour intercepter les exceptions **ABRT** d'une classe Java, veuillez utiliser la commande suivante :

```
~]$ java -agentlib:abrt-java-connector[=abrt=on] $MyClass -
platform.jvmtiSupported true
```

Dans la commande ci-dessus, remplacez *$MyClass* par le nom de la classe Java que vous souhaitez tester. En passant l'option **abrt=on** au connecteur, vous vous assurez que les exceptions sont gérées par **abrtd**. Dans le cas où vous souhaiteriez faire en sorte que le connecteur fasse sortir les exceptions sur la sortie standard, ne pas utiliser cette option.

22.4.5. Détection d'incidents X.Org

Le service **abrt-xorg** collecte et traite les informations sur les incidents du serveur **X.Org server** qui proviennent du fichier **/var/log/Xorg.0.log**. Remarquez qu'aucun rapport n'est généré si un module **X.org** sur liste noire est chargé. À la place, un fichier **not-reportable** est créé dans le répertoire de données problématiques avec une explication appropriée. Vous trouverez la liste des modules fautifs dans le fichier **/etc/abrt/plugins/xorg.conf**. Seuls les modules de pilotes graphiques propriétaires sont mis sur liste noire par défaut.

22.4.6. Détection d'oops et de paniques de noyau

En vérifiant la sortie des journaux de noyau, **ABRT** est capable d'intercepter et de traiter les oops de noyau (« Kernel Oops ») — des déviations non fatales du comportement par défaut du noyau Linux. Cette functionalité est fournie par le service **abrt-oops**.

ABRT peut également détecter et traiter les paniques de noyau — des erreurs fatales, et non

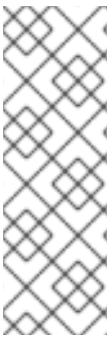
récupérables qui nécessitent un redémarrage à l'aide du service **abrt-vmcore**. Le service démarre uniquement lorsqu'un fichier **vmcore** (un fichier d'image mémoire noyau) apparaît dans le répertoire **/var/crash/**. Lorsqu'un fichier d'image mémoire est trouvé, **abrt-vmcore** crée un nouveau **répertoire de données problématiques** dans le répertoire **/var/tmp/abrt/** et déplace le fichier de l'image mémoire sur le nouveau répertoire de données problématiques créé. Une fois la recherche du répertoire **/var/crash/** est terminée, le service est arrêté.

Pour qu'**ABRT** soit capable de détecter une panique de noyau, le service **kdump** doit être activé sur le système. La quantité de mémoire réservée au noyau kdump doit être définie correctement. Cela peut être effectué en utilisant l'outil graphique **system-config-kdump** ou en spécifiant le paramètre **crashkernel** dans la liste des options de noyau dans le menu GRUB. Pour obtenir des détails sur la manière d'activer et de configurer **kdump**, veuillez consulter le Guide de vidage sur incident noyau Red Hat Enterprise Linux 7. Pour obtenir des informations sur la façon d'apporter des modifications au menu GRUB, voir Chapitre 24, *Utiliser le chargeur de démarrage GRUB 2*.

En utilisant le service **abrt-pstoreoops**, **ABRT** est capable de collecter et de traiter des informations sur les paniques de noyau, qui, sur les systèmes prenant en charge *pstore*, sont stockées dans le répertoire monté automatiquement **/sys/fs/pstore/**. L'interface pstore (stockage persistant) dépendante de la plateforme, fournit un mécanisme de stockage de données à travers les redémarrages système, permettant ainsi de préserver les informations sur les paniques de noyau. Le service démarre automatiquement lorsque les fichiers de vidage sur incident du noyau apparaissent dans le répertoire **/sys/fs/pstore/**.

22.5. GESTION DES PROBLÈMES DÉTECTÉS

Les données des problèmes enregistrées par **abrtd** peuvent être affichées, rapportées, et supprimées en utilisant l'outil de ligne de commande **abrt-cli**, ou l'outil graphique **gnome-abrt**.



NOTE

Remarquez qu'**ABRT** identifie les problèmes dupliqués en comparant les nouveaux problèmes avec tous les problèmes enregistrés localement. Pour répéter un incident, **ABRT** ne requiert qu'une action. Cependant, si vous supprimez le vidage sur incident de ce problème, la prochaine fois que ce problème particulier se produira, **ABRT** le traitera comme s'il s'agissait d'un nouvel incident : **ABRT** vous alertera, et vous demandera de remplir une description et de le rapporter. Pour éviter qu'**ABRT** vous notifie au sujet d'un problème récurrent, veuillez ne pas supprimer les données du problème.

22.5.1. Utiliser l'outil de ligne de commande

Dans l'environnement en ligne de commande, l'utilisateur est notifié de nouveaux incidents lors de la connexion, à condition que le paquet abrt-console-notification soit installé. Les notifications de la console ressemblent à ce qui suit :

```
ABRT has detected 1 problem(s). For more info run: abrt-cli list --since
1398783164
```

Pour afficher les problèmes détectés, veuillez saisir la commande **abrt-cli list** :

```
~]$ abrt-cli list
id 6734c6f1a1ed169500a7bfc8bd62aabaf039f9aa
Directory:      /var/tmp/abrt/ccpp-2014-04-21-09:47:51-3430
count:          1
```

```
executable:     /usr/bin/sleep
package:        coreutils-8.22-11.el7
time:           Mon 21 Apr 2014 09:47:51 AM EDT
uid:            1000
Run 'abrt-cli report /var/tmp/abrt/ccpp-2014-04-21-09:47:51-3430' for
creating a case in Red Hat Customer Portal
```

Chaque incident répertorié dans la sortie de la commande **abrt-cli list** possède un identifiant unique et un répertoire qui peut être utilisé pour des manipulations supplémentaires en utilisant **abrt-cli**.

Pour afficher les informations sur un problème en particulier, veuillez utiliser la commande **abrt-cli info** :

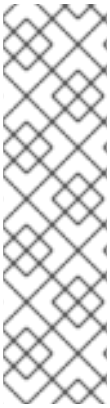
```
abrt-cli info [-d] directory_or_id
```

Pour augmenter la quantité d'informations affichées lors de l'utilisation des sous-commandes **list** et **info**, veuillez leur passer l'option **-d** (**--detailed**), qui affiche toutes les informations stockées sur les problèmes répertoriés, y compris les fichiers **backtrace** respectifs si ceux-ci ont déjà été générés.

Pour analyser et rapporter un problème particulier, veuillez utiliser la commande **abrt-cli report** :

```
abrt-cli report directory_or_id
```

Lors de l'invocation de la commande ci-dessus, il vous sera demandé de fournir les informations d'identification pour l'ouverture d'un dossier de support technique avec le service client de Red Hat. Ensuite, **abrt-cli** ouvre un éditeur de texte avec le contenu du rapport. Vous pouvez voir ce qui est rapporté et vous pouvez inclure des instructions sur la manière de reproduire l'incident, ainsi qu'ajouter d'autres commentaires. Vous pouvez également vérifier le backtrace car celui-ci peut être envoyé sur un serveur public et vu par tout le monde, selon les paramètres de l'événement rapport-problème.



NOTE

Vous pouvez choisir quel éditeur de texte sera utilisé pour vérifier les rapports. **abrt-cli** utilise l'éditeur défini dans la variable d'environnement **ABRT_EDITOR**. Si la variable n'est pas définie, les variables **VISUAL** et **EDITOR** seront vérifiées. Si aucune de ces variables n'est définie, alors l'éditeur **vi** sera utilisé. Vour pouvez définir l'éditeur préféré dans votre fichier de configuration **.bashrc**. Par exemple, si vous préférez **GNU Emacs**, veuillez ajouter la ligne suivante au fichier :

```
export VISUAL=emacs
```

Lorsque vous aurez terminé le rapport, enregistrez vos changements et fermez l'éditeur. Si vous avez rapporté le problème sur la base de données du Support client de Red Hat, un dossier de problème sera créé dans la base de données. Désormais, vous serez informé sur la progression de la résolution du problème via courrier électronique à l'adresse fournie pendant le processus de rapport. Vous pouvez également suivre le dossier du problème à l'aide de l'URL fourni lors de la création du dossier ou via les courriers électroniques reçus du Support de Red Hat.

Si vous êtes certain de ne pas souhaiter rapporter un problème en particulier, vous pouvez le supprimer. Pour supprimer un problème de manière à ce qu'**ABRT** ne conserve pas d'informations à son sujet, veuillez utiliser la commande :

```
abrt-cli rm directory_or_id
```

Pour afficher de l'aide sur une commande **abrt-cli** en particulier, veuillez utiliser l'option **--help** :

```
abrt-cli command --help
```

22.5.2. Utilisation de la GUI

Le démon **ABRT** diffuse un message **D-Bus** chaque fois qu'un rapport de problème est créé. Si la miniapplication **ABRT** est en cours d'exécution dans un environnement de bureau graphique, elle recevra ce message et affichera une boîte de dialogue de notification sur le bureau. Vous pouvez ouvrir la GUI **ABRT** en utilisant cette boîte de dialogue en cliquant sur le bouton **Rapport**. Vous pouvez également ouvrir la GUI **ABRT** en sélectionnant l'élément de menu **Applications** → **Sundry** → **Automatic Bug Reporting Tool**.

Alternativement, vous pouvez exécuter la GUI **ABRT** à partir de la ligne de commande comme suit :

```
~]$ gnome-abrt &
```

La fenêtre de la GUI **ABRT** affiche une liste des problèmes détectés. Chaque entrée de problème comprend le nom de l'application en panne, la raison pour laquelle l'application est en panne, et la date de la dernière incidence du problème.

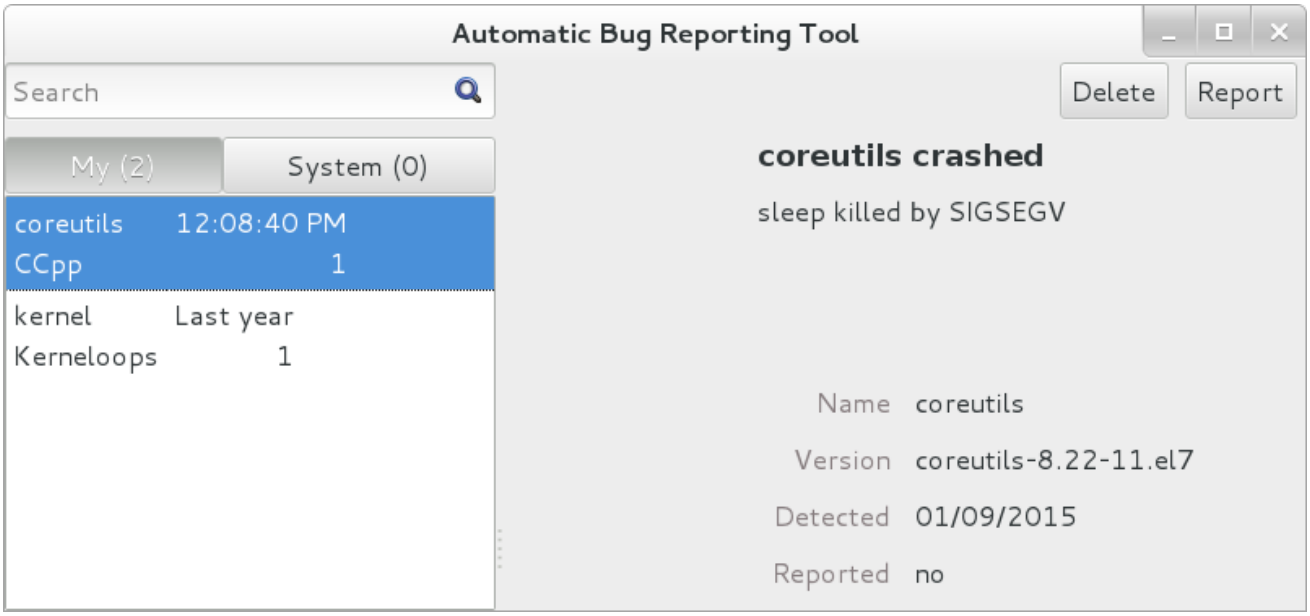


Figure 22.3. GUI ABRT

Pour accéder à la description détaillée d'un problème, double-cliquez sur la ligne du rapport de problème ou cliquez sur le bouton **Rapport** lorsque la ligne du problème correspondant est sélectionnée. Vous pourrez ensuite suivre les instructions pour continuer avec le processus de description du problème, déterminant ainsi comment il doit être analysé, et où il doit être rapporté. Pour abandonner un problème, cliquez sur le bouton **Supprimer**.

22.6. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir des informations supplémentaires sur **ABRT** et sur des sujets proches, veuillez consulter les ressources répertoriées ci-dessous.

Documentation installée

- abrtd(8) — The manual page for the **abrtd** daemon provides information about options that can be used with the daemon.
- abrt_event.conf(5) — The manual page for the **abrt_event.conf** configuration file describes the format of its directives and rules and provides reference information about event meta-data configuration in XML files.

Documentation en ligne

- Guide de mise en réseau Red Hat Enterprise Linux 7 — Le *Guide de mise en réseau* de Red Hat Enterprise Linux 7 documente les informations pertinentes à la configuration et à l'administration des interfaces réseau et des services réseau sur ce système.
- Guide de vidage sur incident noyau Red Hat Enterprise Linux 7 Kernel Crash Dump Guide— Le *Guide de vidage sur incident noyau* de Red Hat Enterprise Linux 7 documente comment configurer, tester, et utiliser le service de récupération sur incident **kdump** et fournit un bref aperçu sur la manière d'analyser l'image mémoire à l'aide de l'utilitaire de débogage **crash**.

Voir aussi

- Chapitre 20, *Afficher et gérer des fichiers journaux* décrit la configuration du démon **rsyslog** et du journal systemd et explique comment localiser, afficher et surveiller les journaux système.
- Chapitre 8, *Yum* décrit comment utiliser le gestionnaire de paquets **Yum** pour rechercher, installer, mettre à jour, et désinstaller des paquets sur la ligne de commande.
- Chapitre 9, *Gérer les services avec systemd* offre une introduction à **systemd** et documente comment utiliser la commande **systemctl** pour gérer des services système, configurer des cibles systemd, et exécuter des commandes de gestion de l'alimentation.

CHAPITRE 23. OPROFILE

OProfile est un outil de surveillance des performances système globales à faible charge. Il utilise le matériel de surveillance des performances sur le processeur pour récupérer des informations sur le noyau et les exécutables se trouvant sur le système, comme lorsque la mémoire est référencée, le nombre de requêtes de cache L2, et le nombre d'interruptions de matériel reçues. Sur un système Red Hat Enterprise Linux, le paquet **oprofile** doit être installé pour utiliser cet outil.

De nombreux processeurs incluent un matériel de surveillance des performances dédié. Ce matériel rend la détection de certains événements possible (comme lorsque les données requises ne se trouvent pas dans le cache). Le matériel prend normalement la forme d'un ou de plusieurs *compteurs* qui sont incrémentés chaque fois qu'un événement se produit. Lorsque la valeur du compteur est incrémentée, une interruption est générée, ce qui rend possible la surveillance des détails (et donc, la charge) donnés par le contrôle de la performance.

OProfile utilise ce matériel (ou un substitut basé compteur dans les cas où aucun matériel de surveillance des performances n'est présent) pour collecter des échantillons (*samples*) de données connexes aux performances chaque fois qu'un compteur génère une interruption. Ces échantillons sont périodiquement écrits sur le disque. Ultérieurement, les données contenues dans ces échantillons pourront être utilisées pour générer des rapports sur les performances au niveau du système et au niveau de l'application.

Limitations lors de l'utilisation d'OProfile :

- *Utilisation des bibliothèques partagées* — les échantillons de code dans les bibliothèques partagées ne sont pas attribués à l'application en particulier, à moins que l'option **--separate=library** soit utilisée.

- *Les échantillons de surveillance des performances sont inexacts* — lorsque la surveillance des performances déclenche un échantillon, la gestion de l'interruption n'est pas précise comme une division par une exception zéro. À cause de l'exécution désordonnée des instructions par le processeur, l'échantillon peut être enregistré sur une instruction aux alentours.

- ***opreport** n'associe pas les échantillons pour les fonctions « inline » correctement* — **opreport** utilise un simple mécanisme de gamme d'adresse pour déterminer dans quelle fonction se trouve une adresse. Les échantillons de fonctions « Inline » ne sont pas attribués à la fonction « inline » mais plutôt à la fonction dans laquelle la fonction « inline » a été insérée.

- *OProfile accumule les données de multiples exécutions* — OProfile est un profileur global et s'attend à ce que les processus démarrent et s'arrêtent à plusieurs reprises. Ainsi, les échantillons des multiples exécutions s'accumulent. Veuillez utiliser la commande **opcontrol --reset** pour supprimer les échantillons des exécutions précédentes.

- *Les compteurs de performance du matériel ne fonctionnent pas sur les machines virtuelles invitées* — comme les compteurs de performance du matériel ne sont pas disponibles sur les systèmes virtuels, vous devrez utiliser le mode **timer**. Saisissez la commande **opcontrol --deinit**, puis exécutez **modprobe oprofile timer=1** pour activer le mode **timer**.

- *Problèmes de performances non limités au CPU* — OProfile est conçu pour trouver des problèmes liés aux processus limités par les CPU. OProfile n'identifie pas les processus endormis car ils attendent qu'un verrouillage ou qu'un autre type d'événement se produise (par exemple qu'un périphérique d'E/S termine une opération).

23.1. APERÇU DES OUTILS

La Tableau 23.1, « Commandes OProfile » offre un bref aperçu des outils les plus couramment utilisés avec le paquet **oprofile**.

Tableau 23.1. Commandes OProfile

Commande	Description
ophelp	Affiche les événements disponibles pour le processeur du système avec une brève description de chacun d'entre eux.
opimport	Convertit les fichiers de la base de données d'un format binaire étranger au format natif du système. Utilisez cette option uniquement lors de l'analyse d'un échantillon de base de données provenant d'une architecture différente.
opannotate	Crée une source annotée pour un exécutable si l'application a été compilée avec des symboles de débogage. Veuillez consulter la Section 23.6.4, « Utiliser **opannotate** » pour voir des détails.
opcontrol	Configure les données à collecter. Veuillez consulter la Section 23.3, « Configurer OProfile en utilisant le mode hérité » pour voir des détails.
operf	Outil recommandé à la place de **opcontrol** pour le profilage. Veuillez consulter la Section 23.2, « Utiliser operf » pour voir des détails. Pour voir les différences entre **operf** et **opcontrol**, veuillez consulter la Section 23.1.1, « operf vs. opcontrol ».
opreport	Récupère les données du profil. Veuillez consulter la Section 23.6.1, « Utiliser **opreport** » pour voir des détails.
oprofiled	S'exécute en tant que démon pour écrire des données échantillon sur le disque de manière périodique.

23.1.1. operf vs. opcontrol

Il existe deux méthodes mutuellement exclusives pour récupérer des données de profilage avec OProfile. Vous pouvez utiliser la méthode la plus récente, qui est également préférable, **operf**, ou l'outil **opcontrol**.

operf

Mode recommandé pour le profilage. L'outil **operf** utilise le sous-système « Linux Performance Events Subsystem », et ne requiert ainsi pas le pilote du noyau *oprofile*. L'outil **operf** vous permet de cibler votre profilage de manière plus précise, en tant que processus unique ou global, il permet également à OProfile de mieux co-exister avec les autres outils utilisant le matériel de surveillance des performances sur votre système. Contrairement à **opcontrol**, celui-ci peut être utilisé sans les privilèges d'utilisateur **root**. Cependant, **operf** est également capable d'effectuer des opérations globales en utilisant l'option **--system-wide**, lorsque les privilèges root sont requis.

Avec **operf**, aucun paramétrage initial n'est nécessaire. Vous pouvez invoquer **operf** avec des options de ligne de commande pour spécifier vos paramètres de profilage. Après cela, vous pourrez exécuter les outils OProfile post-traitement comme décrit dans la Section 23.6, « Analyser les données ». Veuillez consulter la Section 23.2, « Utiliser operf » pour obtenir davantage d'informations.

opcontrol

Ce mode comprend le script shell **opcontrol**, le démon **oprofiled**, et plusieurs outils post-traitement. La commande **opcontrol** est utilisée pour configurer, lancer et arrêter une session de profilage. Un pilote de noyau OProfile, habituellement créé en tant que module de noyau, est utilisé pour collecter les échantillons, qui sont ensuite enregistrés dans des fichiers d'échantillon par **oprofiled**. Vous pouvez utiliser le mode hérité uniquement si vous possédez les privilèges **root**. Dans certains cas, comme lorsque vous devez échantillonner des zones sur lesquelles IRQ (« Interrupt Request ») est désactivé, cette alternative est la meilleure.

Avant qu'OProfile puisse être exécuté en mode hérité, il doit être configuré comme décrit dans la Section 23.3, « Configurer OProfile en utilisant le mode hérité ». Ces paramètres sont ensuite appliqués lors du démarrage d'OProfile (Section 23.4, « Lancer et arrêter OProfile en utilisant le mode hérité »).

23.2. UTILISER OPERF

operf est le mode de profilage recommandé et ne requiert pas de paramétrage initial avant d'être lancé. Tous les paramètres sont spécifiés comme options de ligne de commande et il n'y a pas de commande séparée pour lancer le processus de profilage. Pour arrêter **operf**, appuyez sur Ctrl+C. La syntaxe de la commande **operf** habituelle est comme suit :

```
operf options range command args
```

Remplacez *options* par les options de ligne de commande souhaitées pour spécifier vos paramètres de profilage. Des ensembles complets d'options sont décrits dans la page du manuel **operf(1)**. Remplacez *range* par l'une des options suivantes :

--system-wide - ce paramètre permet d'effectuer un profilage global, veuillez consulter Note

--pid=*PID* - sert à profiler une application en cours d'exécution, où *PID* est l'ID du processus que vous souhaitez profiler.

Avec *command* et *args*, vous pouvez définir une commande ou une application spécifique à profiler, ainsi que les arguments d'entrée requis par cette commande ou application. *command*, **--pid** ou **--system-wide** est requis, mais ces options ne peuvent pas être utilisées simultanément.

Lorsque vous invoquez **operf** sur la ligne de commande sans paramétrer l'option *range*, des données seront collectées pour les processus enfants.

NOTE

Pour exécuter **operf --system-wide**, vous devrez être connecté avec les privilèges **root**. À la fin du profilage, vous pourrez arrêter **operf** avec **Ctrl+C**.

Si vous exécutez **operf --system-wide** en tant que processus d'arrière-plan (avec **&**), arrêtez-le de manière contrôlée pour traiter les données de profil collectées. Pour cela, veuillez utiliser :

```
kill -SIGINT operf-PID
```

Lors de l'exécution de **operf --system-wide**, il est recommandé d'utiliser le répertoire **/root** ou un sous-répertoire de **/root** comme répertoire de travail actuel afin que les fichiers de données échantillons ne soient pas stockés dans des emplacements accessibles aux utilisateurs normaux.

23.2.1. Spécifier le noyau

Pour surveiller le noyau, veuillez exécuter la commande suivante :

```
operf --vmlinux=vmlinux_path
```

Avec cette option, vous pouvez spécifier un chemin vers un fichier vmlinux qui correspond au noyau en cours d'exécution. Les échantillons du noyau seront attribués à ce binaire en permettant aux outils post-traitement d'attribuer des échantillons aux symboles du noyau correspondant. Si cette option n'est pas spécifiée, tous les échantillons de noyau seront attribués à un pseudo binaire nommé « no-vmlinux ».

23.2.2. Paramétrer les événements à surveiller

La plupart des processeurs contiennent des compteurs, qui sont utilisés par OProfile pour surveiller des événements spécifiques. Comme affiché dans la Tableau 23.3, « Processeurs et compteurs OProfile », le nombre de compteurs disponibles dépend du processeur.

Les événements de chaque compteur peuvent être configurés via la ligne de commande ou avec une interface graphique. Pour obtenir plus d'informations sur l'interface graphique, veuillez consulter la Section 23.10, « Interface graphique ». Si le compteur ne peut pas être défini sur un événement particulier, un message d'erreur sera affiché.

NOTE

Certains anciens modèles de processeurs ne sont pas pris en charge par le noyau sous-jacent « Linux Performance Events Subsystem » et ne sont donc pas pris en charge par **operf**. Si vous recevez ce message :

```
Your kernel's Performance Events Subsystem does not support your
processor type
```

lorsque vous tentez d'utiliser **operf**, essayez d'effectuer un profilage avec **opcontrol** pour voir si votre type de processeur pourrait être pris en charge par le mode hérité d'OProfile.

NOTE

Comme les compteurs de performance du matériel ne sont pas disponibles sur les machines virtuelles invitées, il faut activer le mode *timer* pour utiliser **operf** sur des systèmes virtuels. Pour cela, veuillez saisir en tant qu'utilisateur **root** :

```
opcontrol --deinit
```

```
modprobe oprofile timer=1
```

Pour paramétrer l'événement pour chaque compteur configurable via la ligne de commande, veuillez utiliser :

```
operf --events=event1,event2…
```

Veuillez saisir une liste séparée par des virgules des spécifications d'événements pour le profilage. Chaque spécification est une liste d'attributs séparés par le caractère des deux-points sous le format suivant :

```
event-name:sample-rate:unit-mask:kernel:user
```

Tableau 23.2, « Spécifications d'événement » résume ces options. Les trois dernières valeurs sont optionnelles. Si vous les omettez, elles seront paramétrées sur leurs valeurs par défaut. Remarquez que certains événements requièrent un masque d'unité.

Tableau 23.2. Spécifications d'événement

Spécification	Description
event-name	Nom de l'événement symbolique exact pris de **ophelp**
sample-rate	Nombre d'événements à attendre avant d'échantillonner à nouveau. Plus le compte est faible, plus les échantillons sont fréquents. Pour les événements qui ne se produisent pas fréquemment, un compte plus faible peut être nécessaire pour capturer un nombre statistiquement significatif d'instances d'événements. D'un autre côté, un échantillonnage trop fréquent peut surcharger le système. Par défaut, OProfile utilise un ensemble d'événements basé temps, ce qui crée un échantillon tous les 100,000 cycles d'horloge par processeur.
unit-mask	Les masques d'unité, qui définissent encore plus l'événement, sont répertoriés dans **ophelp**. Vous pouvez insérer une valeur hexadécimale, commençant par « 0x », ou une chaîne qui correspond au premier mot de la description du masque d'unité dans **ophelp**. La définition par nom est valide pour les masques d'unité comprenant des paramètres « extra: », comme indiqué par la sortie de **ophelp**. Ce type de masque d'unité ne peut pas être défini avec une valeur hexadécimale. Remarquez que sur certaines architectures, il peut y avoir de multiples masques d'unité avec la même valeur hexadécimale. Dans ce cas, ils devront être spécifiés par leurs nom uniquement.
kernel	Indique s'il faut profiler le code du noyau (insérez **0** ou **1** (par défaut))
user	Indique s'il faut profiler le code de l'espace utilisateur (insérez **0** ou **1** (par défaut))

Les événements disponibles varient fortement selon le type du processeur. Lorsqu'aucune spécification n'est donnée, l'événement par défaut du type de processeur en cours d'exécution sera utilisé pour le profilage. Veuillez consulter la Tableau 23.4, « Événements par défaut » pour voir une liste de ces

événements par défaut. Pour déterminer les événements disponibles pour un profilage, veuillez utiliser la commande **ophelp**.

```
ophelp
```

23.2.3. Catégorisation des échantillons

L'option **--separate-thread** catégorise les échantillons par ID de groupe de thread (tgid) et par ID de thread (tid). Ceci est utile pour voir les échantillons par thread dans les applications à threads multiples. Lorsqu'utilisé en conjonction avec l'option **--system-wide**, **--separate-thread** est également utile pour voir les échantillons par processus (par groupe de threads) dans le cas où de multiples processus seraient exécutés par le même programme pendant l'exécution d'un profilage.

L'option **--separate-cpu** catégorises les échantillons par CPU.

23.3. CONFIGURER OPROFILE EN UTILISANT LE MODE HÉRITÉ

Avant qu'OProfile puisse être exécuté en mode hérité, il doit être configuré. Au minimum, il est requis de sélectionner de surveiller le noyau (ou de sélectionner de ne pas le surveiller). Les sections suivantes décrivent comment utiliser l'utilitaire **opcontrol** pour configurer OProfile. Tandis que les commandes **opcontrol** sont exécutées, les options de l'installation sont enregistrées sur le fichier **/root/.oprofile/daemonrc**.

23.3.1. Spécifier le noyau

Veuillez commencer par configurer si OProfile doit surveiller le noyau. Cette option de configuration est la seule qui est requise avant de lancer OProfile. Toutes les autres sont optionnelles.

Pour surveiller le noyau, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
opcontrol --setup --vmlinux=/usr/lib/debug/lib/modules/`uname -r`/vmlinux
```

IMPORTANT

Pour surveiller le noyau, le paquet kernel-debuginfo, qui contient le noyau non compressé, doit être installé. Pour obtenir davantage d'informations sur la manière d'installer ce paquet, veuillez consulter l'article Comment télécharger les paquets debuginfo comme kernel-debuginfo? sur le Portail Client Red Hat.

Pour configurer OProfile à ne pas surveiller le noyau, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
opcontrol --setup --no-vmlinux
```

Cette commande charge également le module **oprofile** du noyau si celui-ci n'est pas déjà chargé, et crée le répertoire **/dev/oprofile/** s'il n'existe pas déjà. Veuillez consulter la Section 23.7, « Comprendre le répertoire /dev/oprofile/ » pour obtenir des détails sur ce répertoire.

Paramétrer si les échantillons doivent être collectés dans le noyau ne change uniquement que les données collectées, et non la façon dont elles le sont, ni où elles seront stockées après leur collecte. Pour générer différents fichiers échantillons pour le noyau et les bibliothèques d'application, veuillez consulter la Section 23.3.3, « Séparer les profils du noyau et de l'espace utilisateur ».

23.3.2. Paramétrer les événements à surveiller

La plupart des processeurs contiennent des *compteurs*, qui sont utilisés par OProfile pour surveiller des événements spécifiques. Comme affiché dans la Tableau 23.3, « Processeurs et compteurs OProfile », le nombre de compteurs disponibles dépend du processeur.

Tableau 23.3. Processeurs et compteurs OProfile

Processeur	`cpu_type`	Nombre de compteurs
AMD64	x86-64/hammer	4
AMD Family 10h	x86-64/family10	4
AMD Family 11h	x86-64/family11	4
AMD Family 12h	x86-64/family12	4
AMD Family 14h	x86-64/family14	4
AMD Family 15h	x86-64/family15	6
Applied Micro X-Gene	arm/armv8-xgene	4
ARM Cortex A53	arm/armv8-ca53	6
ARM Cortex A57	arm/armv8-ca57	6
IBM eServer System i et IBM eServer System p	timer	1
IBM POWER4	ppc64/power4	8
IBM POWER5	ppc64/power5	6
IBM PowerPC 970	ppc64/970	8
IBM PowerPC 970MP	ppc64/970MP	8
IBM POWER5+	ppc64/power5+	6
IBM POWER5++	ppc64/power5++	6
IBM POWER56	ppc64/power6	6
IBM POWER7	ppc64/power7	6
IBM POWER8	ppc64/power7	8

Processeur	cpu_type	Nombre de compteurs
IBM S/390 et IBM System z	timer	1
Intel Core i7	i386/core_i7	4
Intel Nehalem microarchitecture	i386/nehalem	4
Intel Westmere microarchitecture	i386/westmere	4
Intel Haswell microarchitecture (non-hyper-threaded)	i386/haswell	8
Intel Haswell microarchitecture (hyper-threaded)	i386/haswell-ht	4
Intel Ivy Bridge microarchitecture (non-hyper-threaded)	i386/ivybridge	8
Intel Ivy Bridge microarchitecture (hyper-threaded)	i386/ivybridge-ht	4
Intel Sandy Bridge microarchitecture (non-hyper-threaded)	i386/sandybridge	8
Intel Sandy Bridge microarchitecture	i386/sandybridge-ht	4
Intel Broadwell microarchitecture (non-hyper-threaded)	i386/broadwell	8
Intel Broadwell microarchitecture (hyper-threaded)	i386/broadwell-ht	4
Intel Silvermont microarchitecture	i386/silvermont	2
TIMER_INT	timer	1

Veuillez utiliser la Tableau 23.3, « Processeurs et compteurs OProfile » pour déterminer le nombre d'événements pouvant être surveillés simultanément pour votre type de CPU. Si le processeur ne possède pas de matériel de surveillance des performances, le **timer** sera utilisé en tant que type de processeur.

Si **timer** est utilisé, les événements ne pourront être paramétrés pour aucun processeur car il n'offre pas la prise en charge des compteurs de performances du matériel. Au contraire, l'interruption « timer » est utilisée pour le profilage.

Si **timer** n'est pas utilisé comme type de processeur, les événements surveillés pourront être modifiés, et le compteur 0 du processeur est défini sur un événement basé temps par défaut. S'il existe plus d'un compteur sur le processeur, les compteurs autres que 0 ne seront pas définis sur un événement par

défaut. Les événements par défaut surveillés sont affichés dans la Tableau 23.4, « Événements par défaut ».

Tableau 23.4. Événements par défaut

Processeur	Événement par défaut du compteur	Description
AMD Athlon et AMD64	CPU_CLK_UNHALTED	L'horloge du processeur n'est pas à l'arrêt
AMD Family 10h, AMD Family 11h, AMD Family 12h	CPU_CLK_UNHALTED	L'horloge du processeur n'est pas à l'arrêt
AMD Family 14h, AMD Family 15h	CPU_CLK_UNHALTED	L'horloge du processeur n'est pas à l'arrêt
Applied Micro X-Gene	CPU_CYCLES	Cycles de processeur
ARM Cortex A53	CPU_CYCLES	Cycles de processeur
ARM Cortex A57	CPU_CYCLES	Cycles de processeur
IBM POWER4	CYCLES	Cycles de processeur
IBM POWER5	CYCLES	Cycles de processeur
IBM POWER8	CYCLES	Cycles de processeur
IBM PowerPC 970	CYCLES	Cycles de processeur
Intel Core i7	CPU_CLK_UNHALTED	L'horloge du processeur n'est pas à l'arrêt
Intel Nehalem microarchitecture	CPU_CLK_UNHALTED	L'horloge du processeur n'est pas à l'arrêt
Intel Pentium 4 (hyper-threaded et non-hyper-threaded)	GLOBAL_POWER_EVENTS	Période pendant laquelle le processeur n'est pas arrêté
Intel Westmere microarchitecture	CPU_CLK_UNHALTED	L'horloge du processeur n'est pas à l'arrêt
Intel Broadwell microarchitecture	CPU_CLK_UNHALTED	L'horloge du processeur n'est pas à l'arrêt
Intel Silvermont microarchitecture	CPU_CLK_UNHALTED	L'horloge du processeur n'est pas à l'arrêt

Processeur	Événement par défaut du compteur	Description
TIMER_INT	(aucun)	Échantillon pour chaque interruption du timer

Le nombre d'événements pouvant être surveillés à la fois est déterminé par le nombre de compteurs du processeur. Cependant, il ne s'agit pas ici d'une corrélation exacte. Sur certains processeurs, des événements doivent être mappés sur des compteurs spécifiques. Pour déterminer le nombre de compteurs disponibles, veuillez exécuter la commande suivante :

```
ls -d /dev/oprofile/[0-9]*
```

Les événements disponibles varient en fonction du type de processeur. Veuillez utiliser la commande **ophelp** pour déterminer les événements disponibles pour le profilage. La liste est spécifique au type de processeur du système.

```
ophelp
```

NOTE

À moins qu'OProfile soit correctement configuré, **ophelp** échoue avec le message d'erreur suivant :

```
Unable to open cpu_type file for reading
Make sure you have done opcontrol --init
cpu_type 'unset' is not valid
you should upgrade oprofile or force the use of timer mode
```

Pour configurer OProfile, veuillez suivre les instructions dans la Section 23.3, « Configurer OProfile en utilisant le mode hérité ».

Les événements de chaque compteur peuvent être configurés via la ligne de commande ou avec une interface graphique. Pour obtenir plus d'informations sur l'interface graphique, veuillez consulter la Section 23.10, « Interface graphique ». Si le compteur ne peut pas être défini sur un événement particulier, un message d'erreur sera affiché.

Pour définir l'événement pour chaque compteur configurable via la ligne de commande, veuillez utiliser **opcontrol** :

```
opcontrol --event=event-name:sample-rate
```

Veuillez remplacer *event-name* par le nom exact de l'événement de **ophelp**, et remplacez *sample-rate* par le nombre d'événements entre échantillons.

23.3.2.1. Taux d'échantillonnage

Par défaut, un ensemble d'événements basé temps est sélectionné. Il crée un échantillon tous les 100,000 cycles d'horloge par processeur. Si l'interruption timer est utilisée, le timer sera défini sur le taux prévu et ne pourra pas être modifié pas un utilisateur. Si **cpu_type** n'est pas égal à **timer**, chaque

événement aura un *taux d'échantillonnage* paramétré. Le taux d'échantillonnage correspond au nombre d'événements entre chaque capture d'échantillon.

Lors de la définition de l'événement pour le compteur, un taux d'échantillonnage doit également être spécifié :

```
opcontrol --event=event-name:sample-rate
```

Veuillez remplacer *sample-rate* par le nombre d'événements à attendre avant d'échantillonner à nouveau. Plus le compte est faible, plus les échantillons sont fréquents. Pour les événements qui n'ont pas lieu fréquemment, un compte plus faible peut être nécessaire pour capturer les instances des événements.

AVERTISSEMENT

Faites extrêmement attention pendant la définition des taux d'échantillonnage. Un échantillonnage trop fréquent peut surcharger le système, le faisant apparaitre gelé ou le faisant réellement geler.

23.3.2.2. Masques d'unités

Certains événements de surveillance des performances peuvent également nécessiter des masques d'unités pour mieux définir l'événement.

Des masques d'unités pour chaque événement sont répertoriés avec la commande **ophelp**. Les valeurs de chaque masque d'unité sont répertoriées sous un format hexadécimal. Pour spécifier plus d'un masque d'unité, les valeurs hexadécimales doivent être combinées en utilisant une opération bitwise *or*.

```
opcontrol --event=event-name:sample-rate:unit-mask
```

Remarquez que sur certaines architectures, il peut y avoir des masques d'unités multiples comportant la même valeur hexadécimale. Dans ce cas, ils doivent être spécifiés par leur nom uniquement.

23.3.3. Séparer les profils du noyau et de l'espace utilisateur

Par défaut, les informations du mode noyau et du mode utilisateur sont collectées pour chaque événement. Pour configurer OProfile de manière à ignorer les événements en mode noyau pour un compteur spécifique, veuillez exécuter la commande suivante :

```
opcontrol --event=event-name:sample-rate:unit-mask:0
```

Veuillez exécuter la commande suivante pour lancer à nouveau le profilage du mode noyau pour le compteur :

```
opcontrol --event=event-name:sample-rate:unit-mask:1
```

Pour configurer OProfile de manière à ignorer les événements en mode utilisateur pour un compteur spécifique, veuillez exécuter la commande suivante :

```
opcontrol --event=event-name:sample-rate:unit-mask:1:0
```

Veuillez exécuter la commande suivante pour lancer à nouveau le profilage du mode utilisateur pour le compteur :

```
opcontrol --event=event-name:sample-rate:unit-mask:1:1
```

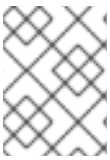
Lorsque le démon OProfile écrit les données du profil sur des fichiers échantillons, il peut séparer les données de profil du noyau et de la bibliothèque en fichiers échantillons séparés. Pour configurer la manière par laquelle le démon écrit sur les fichiers échantillons, veuillez exécuter la commande suivante en tant qu'utilisateur root :

```
opcontrol --separate=choice
```

L'argument *choice* peut prendre l'une des valeurs suivantes :

- **none** — ne pas séparer les profils (valeur par défaut).
- **library** — générer des profils par application pour les bibliothèques.
- **kernel** — générer des profils par application pour le noyau et les modules du noyau.
- **all** — générer des profils par application pour les bibliothèques et des profils par application pour le noyau et les modules du noyau.

Si **--separate=library** est utilisé, le nom du fichier échantillon inclura également le nom de l'exécutable, ainsi que le nom de la bibliothèque.



NOTE

Ces changement de configuration ne rentreont en vigueur que lorsque le profileur OProfile sera redémarré.

23.4. LANCER ET ARRÊTER OPROFILE EN UTILISANT LE MODE HÉRITÉ

Pour lancer la surveillance du système avec OProfile, veuillez exécuter la commande suivante en qu'utilisateur root :

```
opcontrol --start
```

Une sortie similaire à la suivante devrait s'afficher :

```
Using log file /var/lib/oprofile/oprofiled.log Daemon started. Profiler
running.
```

Les paramètres dans **/root/.oprofile/daemonrc** sont utilisés.

Le démon OProfile, **oprofiled**, est lancé. Il écrit de manière périodique les données échantillons sur le répertoire **/var/lib/oprofile/samples/**. Le fichier journal du démon se trouve dans **/var/lib/oprofile/oprofiled.log**.

IMPORTANT

Sur un système Red Hat Enterprise Linux 7, le **nmi_watchdog** est enregistré avec un sous-système **perf**. Pour cette raison, le sous-système **perf** prend contrôle des enregistrements du compteur des performance pendant le démarrage, ce qui empêche à OProfile de fonctionner.

Pour résoudre cela, démarrez avec le paramètre du noyau **nmi_watchdog=0** défini, ou exécutez la commande suivante en tant qu'utilisateur **root** pour désactiver **nmi_watchdog** pendant le démarrage :

```
echo 0 > /proc/sys/kernel/nmi_watchdog
```

Pour réactiver **nmi_watchdog**, veuillez utiliser la commande suivante en tant qu'utilisateur **root** :

```
echo 1 > /proc/sys/kernel/nmi_watchdog
```

Pour arrêter le profileur, veuillez exécuter la commande suivante en tant qu'utilisateur root :

```
opcontrol --shutdown
```

23.5. ENREGISTRER DES DONNÉES EN MODE HÉRITÉ

De temps en temps, il peut être utile d'enregistrer des échantillons à un moment spécifique. Par exemple, lors du profilage d'un exécutable, il peut être utile de rassembler différents échantillons basés sur différents ensembles de données d'entrée. Si le nombre d'événements devant être surveillés excède le nombre de compteurs disponibles au processeur, de multiples exécutions d'OProfile pourront être utilisées pour collecter des données, enregistrant ainsi les données d'échantillons sur différents fichiers à chaque fois.

Pour enregistrer l'ensemble actuel de fichiers échantillons, veuillez exécuter la commande suivante, en remplaçant *name* par un nom descriptif unique pour la session actuelle :

```
opcontrol --save=name
```

La commande crée le répertoire **/var/lib/oprofile/samples/*name*/** et les fichiers d'échantillons actuels y sont copiés.

Pour spécifier le répertoire de la session devant contenir les données d'échantillons, veuillez utiliser l'option **--session-dir**. Si non spécifiées, les données sont enregistrées dans le répertoire **oprofile_data/** se trouvant sur le chemin actuel.

23.6. ANALYSER LES DONNÉES

Les mêmes outils post-traitement d'OProfile sont utilisés lors de la collection du profil avec **operf** ou **opcontrol** en mode hérité.

Par défaut, **operf** stocke les données de profilage dans le répertoire ***current_dir*/oprofile_data/**. Vous pouvez modifier l'emplacement avec l'option **--session-dir**. Les outil d'analyse post-profilage, tels qu'**opreport** et **opannotate**, peuvent être utilisés pour générer des rapports de profil. Ces outils recherchent des échantillons dans le répertoire

***current_dir*/oprofile_data/** en premier. Si ce répertoire n'existe pas, les outils d'analyse utilisent le répertoire de la session standard de **/var/lib/oprofile/**. Des statistiques, comme le total des échantillons reçus et le total des échantillons perdus, sont écrites sur le fichier ***session_dir*/samples/operf.log**.

Pendant l'utilisation du mode hérité, le démon OProfile, **oprofiled** collecte périodiquement les échantillons et les écrits dans le répertoire **/var/lib/oprofile/samples/**. Avant de lire les données, assurez-vous que toutes les données ont bien été écrites sur ce répertoire, en exécutant la commande suivante en tant qu'utilisateur root :

```
opcontrol --dump
```

Chaque nom de fichier d'échantillon est basé sur le nom de l'exécutable. Par exemple, les échantillons de l'événement par défaut sur un processeur Pentium III de **/bin/bash** devient :

```
\{root\}/bin/bash/\{dep\}/\{root\}/bin/bash/CPU_CLK_UNHALTED.100000
```

Les outils suivants sont disponibles pour profiler les données d'échantillons une fois qu'elles ont été collectées :

- **opreport**
- **opannotate**

Veuillez utiliser ces outils, ainsi que les binaires profilés, pour générer des rapports qui pourront être analysés davantage en profondeur.

AVERTISSEMENT

L'exécutable en cours de profilage doit être utilisé avec ces outils pour analyser les données. S'il doit changer après la collection des données, veuillez effectuez une copie de sauvegarder de l'exécutable utilisé pour créer les échantillons ainsi que les fichiers d'échantillons. Remarquez que les noms du fichier d'échantillon et du binaire doivent s'accorder. Vous ne pourrez pas faire de sauvegarde si ces noms ne correspondent pas. De manière alternative, **oparchive** peut être utilisé pour répondre à ce problème.

Les échantillons de chaque exécutable sont écrits sur un seul fichier d'échantillons. Les échantillons de chaque bibliothèque dynamiquement liée sont également écrits sur un seul fichier d'échantillons. Pendant l'exécution d'OProfile, si l'exécutable en cours de surveillance change et qu'un fichier d'échantillons pour l'exécutable existe, le fichier d'échantillons existant sera automatiquement supprimé. Ainsi, si le fichier d'échantillons est nécessaire, il devra être sauvegardé avec l'exécutable utilisé pour le créer avant de remplacer l'exécutable par une nouvelle version. Les outils d'analyse OProfile utilisent le fichier exécutable qui a créé les échantillons pendant l'analyse. Si l'exécutable change, les outils d'analyse seront incapables d'analyser les échantillons associés. Veuillez consulter la Section 23.5, « Enregistrer des données en mode hérité » pour obtenir des détails sur la manière de sauvegarder le fichier d'échantillons.

23.6.1. Utiliser `opreport`

L'outil **opreport** offre une vue d'ensemble sur tous les exécutables en cours de profilage. Voici un extrait d'échantillon de la commande **opreport** :

```
~]$ opreport
Profiling through timer interrupt
TIMER:0|
samples|      %|
------------------
25926 97.5212 no-vmlinux
359  1.3504 pi
65  0.2445 Xorg
62  0.2332 libvte.so.4.4.0
56  0.2106 libc-2.3.4.so
34  0.1279 libglib-2.0.so.0.400.7
19  0.0715 libXft.so.2.1.2
17  0.0639 bash
8  0.0301 ld-2.3.4.so
8  0.0301 libgdk-x11-2.0.so.0.400.13
6  0.0226 libgobject-2.0.so.0.400.7
5  0.0188 oprofiled
4  0.0150 libpthread-2.3.4.so
4  0.0150 libgtk-x11-2.0.so.0.400.13
3  0.0113 libXrender.so.1.2.2
3  0.0113 du
1  0.0038 libcrypto.so.0.9.7a
1  0.0038 libpam.so.0.77
1  0.0038 libtermcap.so.2.0.8
1  0.0038 libX11.so.6.2
1  0.0038 libgthread-2.0.so.0.400.7
1  0.0038 libwnck-1.so.4.9.0
```

Chaque exécutable est répertorié sur sa propre ligne. La première colonne est le nombre d'échantillons enregistrés pour l'exécutable. La seconde colonne est le pourcentage d'échantillons relatif au nombre total d'échantillons. La troisième colonne est le nom de l'exécutable.

Veuillez consulter la page du manuel **opreport(1)** pour afficher une liste des options de ligne de commande disponibles, comme l'option **-r** utilisée pour trier la sortie de l'exécutable avec le plus petit nombre d'échantillons vers celui en ayant le plus grand nombre. Vous pouvez également utiliser l'option **-t** ou **--threshold** pour réduire la sortie de **opcontrol**.

23.6.2. Utiliser opreport sur un seul exécutable

Pour récupérer des informations de profilage plus détaillée sur un exécutable spécifique, veuillez utiliser :

```
opreport mode executable
```

Veuillez remplacer *executable* par le chemin complet de l'exécutable devant être analysé. *mode* correspond à l'une des options suivantes :

-l

Cette option est utilisée pour répertorier les données d'échantillon par symbole. Par exemple, l'exécution de cette commande :

```
~]# opreport -l /usr/lib/tls/libc-version.so
```

produit la sortie suivante :

```
samples % symbol name
12 21.4286 __gconv_transform_utf8_internal
5 8.9286 _int_malloc 4 7.1429 malloc
3 5.3571 __i686.get_pc_thunk.bx
3 5.3571 _dl_mcount_wrapper_check
3 5.3571 mbrtowc
3 5.3571 memcpy
2 3.5714 _int_realloc
2 3.5714 _nl_intern_locale_data
2 3.5714 free
2 3.5714 strcmp
1 1.7857 __ctype_get_mb_cur_max
1 1.7857 __unregister_atfork
1 1.7857 __write_nocancel
1 1.7857 _dl_addr
1 1.7857 _int_free
1 1.7857 _itoa_word
1 1.7857 calc_eclosure_iter
1 1.7857 fopen@@GLIBC_2.1
1 1.7857 getpid
1 1.7857 memmove
1 1.7857 msort_with_tmp
1 1.7857 strcpy
1 1.7857 strlen
1 1.7857 vfprintf
1 1.7857 write
```

La première colonne correspond au nombre d'échantillons pour le symbole, la seconde colonne est le pourcentage d'échantillons de ce symbole relatif au nombre total d'échantillons de l'exécutable, et la troisième colonne est le nom du symbole.

Pour trier la sortie depuis le plus grand nombre d'échantillons au plus petit (l'ordre contraire), veuillez utiliser **-r** en conjonction à l'option **-l**.

-i *symbol-name*

Lister les données d'échantillons spécifiques à un nom de symbole. Par exemple, en exécutant :

```
~]# opreport -l -i __gconv_transform_utf8_internal
/usr/lib/tls/libc-version.so
```

retournera la sortie suivante :

```
samples % symbol name
12 100.000 __gconv_transform_utf8_internal
```

La première ligne est un résumé de la combinaison symbole/exécutable.

La première colonne correspond au nombre d'échantillons pour le symbole de mémoire, la seconde colonne est le pourcentage d'échantillons de l'adresse mémoire relatif au nombre total d'échantillons du symbole, et la troisième colonne est le nom du symbole.

-d

Cette option répertorie les données d'échantillons par symboles avec plus de détails que l'option **-l**. Par exemple, avec la commande suivante :

```
~]# opreport -d -i __gconv_transform_utf8_internal
/usr/lib/tls/libc-version.so
```

la sortie suivante est retournée :

```
vma samples % symbol name
00a98640 12 100.000 __gconv_transform_utf8_internal
00a98640 1 8.3333
00a9868c 2 16.6667
00a9869a 1 8.3333
00a986c1 1 8.3333
00a98720 1 8.3333
00a98749 1 8.3333
00a98753 1 8.3333
00a98789 1 8.3333
00a98864 1 8.3333
00a98869 1 8.3333
00a98b08 1 8.3333
```

Les données sont les mêmes qu'avec l'option **-l** sauf que pour chaque symbole, chaque adresse de mémoire virtuelle est affichée. Pour chaque adresse de mémoire virtuelle, le nombre d'échantillons et le pourcentage d'échantillons relatif au nombre d'échantillons du symbole est affiché.

-e *symbol-name*...

Avec cette option, vous pouvez exclure certains symboles de la sortie. Remplacez *symbol-name* par une liste des symboles que vous souhaitez exclure, séparés par des virgules.

session:*name*

Ici, vous pouvez spécifier le chemin complet de la session, un répertoire relatif au répertoire **/var/lib/oprofile/samples/**, ou si vous utilisez **operf**, un répertoire relatif à **./oprofile_data/samples/**.

23.6.3. Obtenir une sortie plus détaillée sur les modules

OProfile collecte des données sur une base globale pour le code de l'espace utilisateur et de l'espace du noyau exécuté sur la machine. Cependant, une fois qu'un module est chargé dans le noyau, les informations sur l'origine du module du noyau seront perdues. Le module peut provenir du fichier **initrd** pendant le démarrage système, du répertoire avec les divers modules de noyau, ou d'un module de noyau créé localement. Par conséquent, lorsqu'OProfile enregistre des échantillons pour un module, il répertorie les échantillons des modules d'un exécutable dans le répertoire root, mais il est improbable que ce soit l'emplacement où se trouve le code du module. Vous devrez prendre certaines précautions pour vous assurer que les outils d'analyse puissent obtenir le bon exécutable.

Pour obtenir un affichage plus détaillé des actions du module, vous aurez besoin que le module soit « unstripped » (c'est-à-dire installé à partir d'une version personnalisée) ou que le paquet *debuginfo* soit installé pour le noyau.

Découvrez quel noyau est en cours d'exécution avec la commande **uname -a**, obtenez le paquet *debuginfo* approprié et installez-le sur la machine.

Puis, veuillez procéder en supprimant les échantillons des exécutions précédentes avec la commande suivante :

```
opcontrol --reset
```

Par exemple, pour lancer le processus de surveillance sur une machine avec un processeur Westmere, veuillez exécuter la commande suivante :

```
~]# opcontrol --setup --vmlinux=/usr/lib/debug/lib/modules/`uname -
r`/vmlinux --event=CPU_CLK_UNHALTED:500000
```

Puis, les informations détaillées pour, par exemple, le module ext, peuvent être obtenues avec :

```
~]# opreport /ext4 -l --image-path /usr/lib/modules/`uname -r`/kernel
CPU: Intel Westmere microarchitecture, speed 2.667e+06 MHz (estimated)
Counted CPU_CLK_UNHALTED events (Clock cycles when not halted) with a unit
mask of 0x00 (No unit mask) count 500000
warning: could not check that the binary file /lib/modules/2.6.32-
191.el6.x86_64/kernel/fs/ext4/ext4.ko has not been modified since the
profile was taken. Results may be inaccurate.
samples  %        symbol name
1622      9.8381  ext4_iget
1591      9.6500  ext4_find_entry
1231      7.4665  __ext4_get_inode_loc
783       4.7492  ext4_ext_get_blocks
752       4.5612  ext4_check_dir_entry
644       3.9061  ext4_mark_iloc_dirty
583       3.5361  ext4_get_blocks
583       3.5361  ext4_xattr_get
479       2.9053  ext4_htree_store_dirent
469       2.8447  ext4_get_group_desc
414       2.5111  ext4_dx_find_entry
```

23.6.4. Utiliser `opannotate`

L'outil **opannotate** tente de faire correspondre les échantillons de certaines instructions avec les lignes correspondantes du code source. Les fichiers générés en résultant devraient avoir les échantillons des lignes sur le côté gauche. Cet outil met également un commentaire au début de chaque fonction qui répertorie la totalité des échantillons de cette fonction.

Pour que cet utilitaire fonctionne, le paquet *debuginfo* approprié de l'exécutable doit être installé sur le système. Sur Red Hat Enterprise Linux, les paquets *debuginfo* ne sont pas automatiquement installés avec les paquets correspondants qui contiennent l'exécutable. Vous devez les obtenir et les installer séparément.

La syntaxe générale de **opannotate** est comme suit :

```
opannotate --search-dirs src-dir --source executable
```

Ces options de ligne de commande sont obligatoires. Remplacez *src-dir* par un chemin vers le répertoire contenant le code source et spécifiez l'exécutable devant être analysé. Veuillez consulter la page du manuel **opannotate(1)** pour voir une liste des options de ligne de commande supplémentaires.

23.7. COMPRENDRE LE RÉPERTOIRE /DEV/OPROFILE/

Lors de l'utilisation d'OProfile en mode hérité, le répertoire **/dev/oprofile/** est utilisé pour stocker le système de fichiers pour OProfile. D'un autre côté, **operf** ne requiert pas **/dev/oprofile/**. Veuillez utiliser la commande **cat** pour afficher les valeurs des fichiers virtuels dans ce système de fichiers. Par exemple, la commande suivante affiche le type de processeur détecté par OProfile :

```
cat /dev/oprofile/cpu_type
```

Un répertoire existe dans **/dev/oprofile/** pour chaque compteur. Par exemple, s'il y a 2 compteurs, vous verrez les répertoires **/dev/oprofile/0/** et **/dev/oprofile/1/**.

Chaque répertoire de compteur contient les fichiers suivants :

- **count** — l'intervalle entre échantillons.
- **enabled** — si égal à 0, le compteur est éteint et aucun échantillon n'est collecté. Si égal à 1, le compteur est allumé et les échantillonssont collectés.
- **event** — l'événement à surveiller.
- **extra** — utilisé sur les machines avec des processeurs Nehalem pour mieux spécifier l'événement à surveiller.
- **kernel** — si égal à 0, les échantillons ne sont pas collectés pour ce compteur même si le processeur se trouve dans l'espace du noyau. Si égal à 1, les échantillons sont collectés même si le processeur se trouve dans l'espace du noyau.
- **unit_mask** — définit quels masques d'unité sont activés pour le compteur.
- **user** — si égal à 0, les échantillons ne sont pas collectés pour le compteur même si le processeur se trouve dans l'espace utilisateur. Si égal à 1, les échantillons sont collectés même si le processeur se trouve dans l'espace utilisateur.

Les valeurs de ces fichiers peuvent être récupérées par la commande **cat**. Exemple :

```
cat /dev/oprofile/0/count
```

23.8. EXEMPLE D'UTILISATION

Même si OProfile peut être utilisé par des développeurs pour analyser les performances d'applications, il peut également être utilisé par les administrateurs système pour effectuer des analyses système. Exemple :

- *Déterminer les applications et services les plus utilisés sur un système*— **opreport** peut être utilisé pour déterminer combien de temps processeur est utilisé par une application ou un service. Si le système est utilisé pour de multiples services mais qu'il est sous-performant, les services consommant le plus de temps processeur pourront être déplacés sur des systèmes dédiés.

- *Déterminer l'utilisation du processeur*— l'événemement **CPU_CLK_UNHALTED** peut être surveillé pour déterminer la charge du processeur pendant une période donnée. Ces données peuvent ensuite être utilisées pour déterminer si des processeurs supplémentaires ou si un processeur plus rapide pourrait améliorer les performances système.

23.9. PRISE EN CHARGE JAVA D'OPROFILE

OProfile vous permet de profiler dynamiquement le code compilé (également appelé le code JIT, « Just-In-Time ») de la machine virtuelle Java JVM (« Java Virtual Machine »). OProfile sur Red Hat Enterprise Linux 7 inclut la prise en charge intégrée de la bibliothèque d'agents de l'interface d'outils JVMTI (« JVM Tools Interface »), qui prend en charge Java 1.5 et ses versions supérieures.

23.9.1. Profiler le code Java

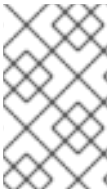
Pour profiler le code JIT de la machine virtuelle Java avec l'agent JVMTI, ajoutez ce qui suit aux paramètres de démarrage JVM :

```
-agentlib:jvmti_oprofile
```

Quand *jvmti_oprofile* est un chemin vers l'agent OProfile. Pour JVM 64 bits, le chemin ressemble à ceci :

```
-agentlib:/usr/lib64/oprofile/libjvmti_oprofile.so
```

Actuellement, vous pouvez ajouter une option de ligne de commande : **--debug**, ce qui active le mode de débogage.



NOTE

Le paquet oprofile-jit doit être installé sur le système pour pouvoir profiler le code JIT avec OProfile. Avec ce paquet, vous serez en mesure d'afficher des informations au niveau de la méthode.

En fonction de la JVM que vous utilisez, vous pourriez devoir installer le paquet *debuginfo* pour la JVM. Pour OpenJDK, ce paquet est requis, il n'y a pas de paquet debuginfo pour Oracle JDK. Pour garder les paquets d'informations debug synchronisés avec leurs paquets non debug respectifs, vous devrez également installer le greffon *yum-plugin-auto-update-debug-info*. Ce greffon recherche les mises à jour correspondantes dans le référentiel des informations debug.

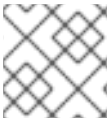
Après une installation réussie, vous pourrez appliquer les outils standard de profilage et d'analyse décrits dans les sections précédentes

Pour en savoir plus sur la prise en charge Java dans OProfile, veuillez consulter le manuel OProfile, qui est relié dans Section 23.12, « Ressources supplémentaires ».

23.10. INTERFACE GRAPHIQUE

Certaines préférences OProfile peuvent être paramétrées avec une interface graphique. Assurez-vous que le paquet **oprofile-gui**, qui fournit l'interface utilisateur graphique OProfile, soit bien installé sur votre système. Pour lancer l'interface, veuillez exécuter la commande **oprof_start** en tant qu'utilisateur root dans l'invite de shell.

Après avoir modifié toute option, veuillez les enregistrer en cliquant sur le bouton **Enregistrer et quitter**. Les préférences sont écrites sur **/root/.oprofile/daemonrc**, et l'application se ferme.



NOTE

Le fait de quitter l'application n'interdira pas à OProfile de prélever des échantillons.

Sur l'onglet **Paramétrage**, pour paramétrer des événements pour les compteurs du processeur comme discuté dans la Section 23.3.2, « Paramétrer les événements à surveiller », veuillez sélectionner le compteur dans le menu déroulant et sélectionnez l'événement dans la liste. Une brève description de l'événement apparaît dans la boîte de texte sous la liste. Seuls les événements disponibles pour le compteur et l'architecture spécifiques sont affichés. L'interface affiche également si le profileur est en cours d'exécution ainsi que de brèves statistiques le concernant.

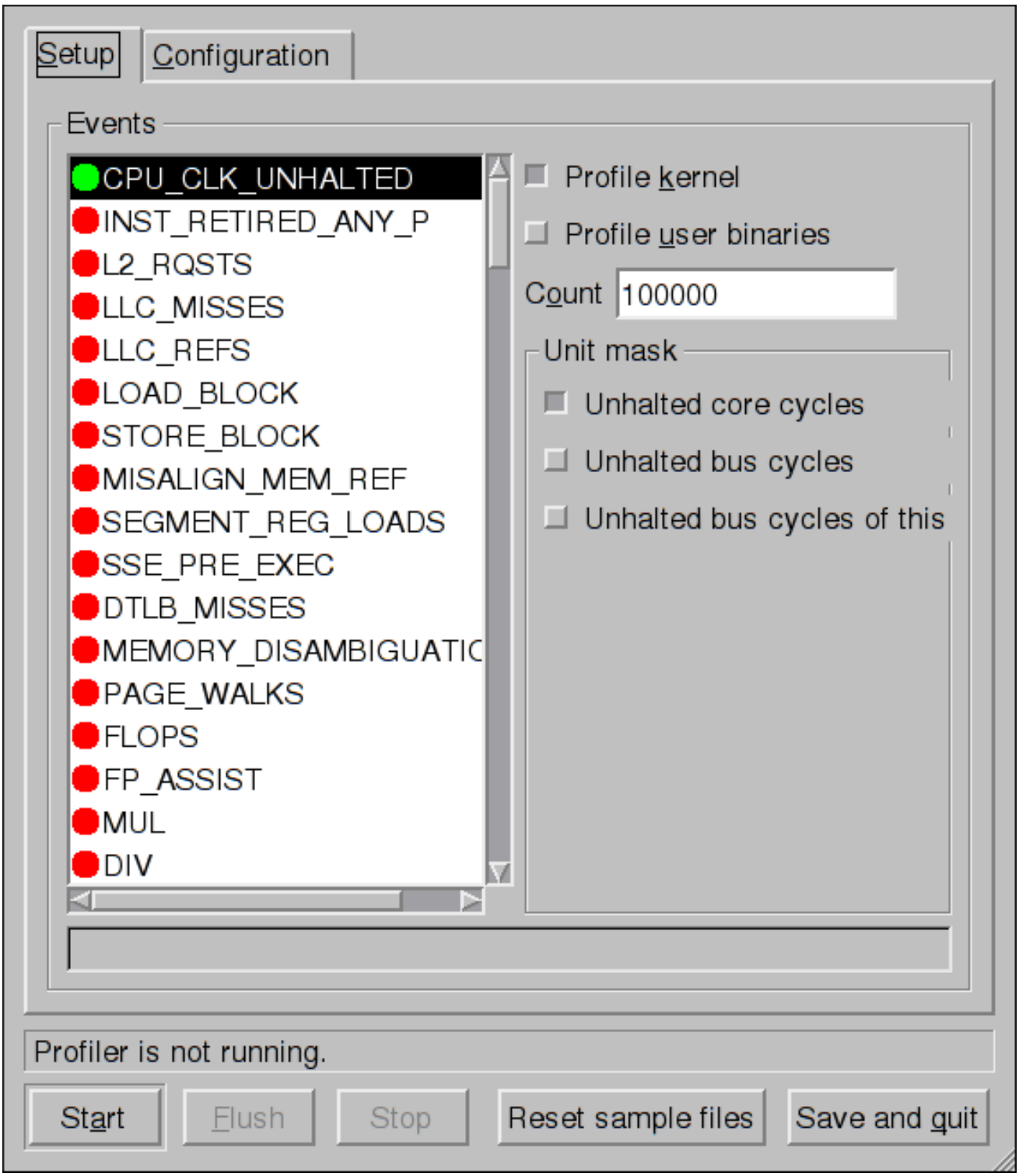


Figure 23.1. Paramétrage d'OProfile

À droite de l'onglet, veuillez sélectionner l'option **Profiler le noyau** pour compter les événements dans le mode du noyau pour l'événement sélectionné, comme discuté dans la Section 23.3.3, « Séparer les profils du noyau et de l'espace utilisateur ». Si cette option n'est pas sélectionnée, aucun échantillon ne sera collecté pour le noyau.

Veuillez sélectionner l'option **Profiler les binaire d'utilisateur** pour compter les événements dans le mode utilisateur pour l'événement actuellement sélectionné, comme discuté dans la Section 23.3.3, « Séparer les profils du noyau et de l'espace utilisateur ». Si cette option n'est pas sélectionnée, aucun échantillon ne sera collecté pour les applications utilisateur.

Veuillez utiliser le champ de texte **Compte** pour définir le taux d'échantillonnage de l'événement actuellement sélectionné, comme discuté dans la Section 23.3.2.1, « Taux d'échantillonnage ».

Si un masque d'unité est disponible pour l'événement actuellement sélectionné, comme on l'explique dans Section 23.3.2.2, « Masques d'unités », il sera affiché dans la zone **Masques d'unités** à droit de l'onglet **Paramétrage**. Veuillez sélectionner la case à cocher à côté du masque d'unité pour l'activer pour l'événement.

Dans l'onglet **Configuration**, pour profiler le noyau, veuillez saisir le nom et l'emplacement du fichier **vmlinux** que le noyau devra surveiller dans le champs de texte **Fichier image du noyau**. Pour configurer OProfile de manière à ne pas surveiller le noyau, veuillez sélectionner **Aucune image de noyau**.

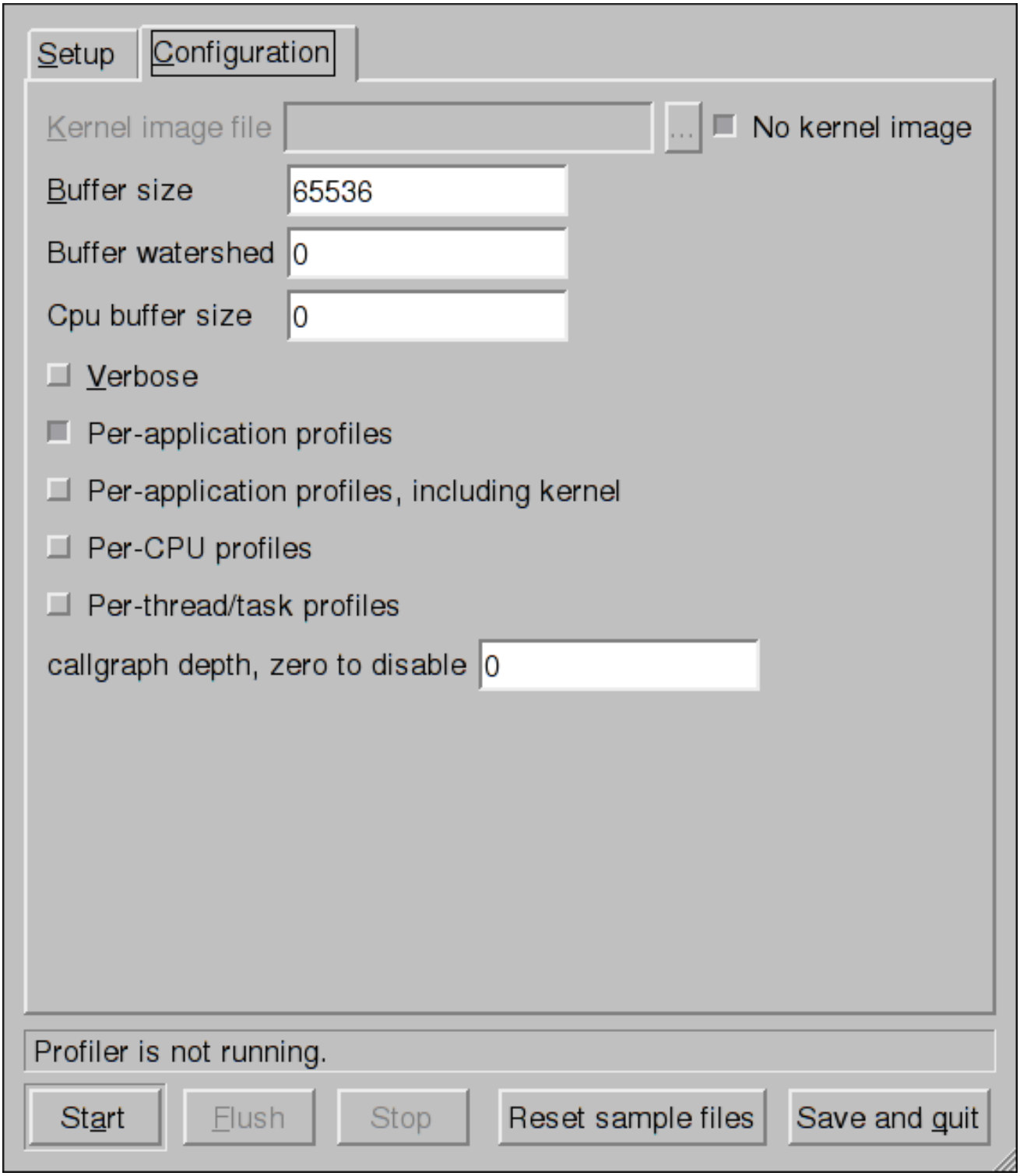


Figure 23.2. Configuration OProfile

Si l'option **Verbose** est sélectionnée, le démon **oprofiled** inclura davantage d'informations détaillées.

Si **Profils par application** est sélectionné, OProfile générera des profils par application pour bibliothèques. Ceci est équivalent à la commande **opcontrol --separate=library**. Si **Profils par application, y compris le noyau** est sélectionné, OProfile générera des profils par application pour le noyau et les modules du noyau comme expliqué dans la Section 23.3.3, « Séparer les profils du noyau et de l'espace utilisateur ». Il n'existe pas d'équivalent à la commande **opcontrol --separate=kernel**.

Pour forcer les données à être écrites sur des fichiers d'échantillons comme expliqué dans la Section 23.6, « Analyser les données », veuillez cliquer sur le bouton **Flush** (« Vider »). Celui-ci est équivalent à la commande **opcontrol --dump**.

Pour lancer OProfile à partir de l'interface graphique, veuillez cliquer sur **Lancer**. Pour arrêter le profileur, cliquez sur **Arrêter**. Quitter l'application n'interdira pas à OProfile de prélever des échantillons.

23.11. OPROFILE ET SYSTEMTAP

SystemTap est un outil de suivi et d'analyse permettant aux utilisateurs d'étudier et de surveiller les activités du système d'exploitation dans les moindres détails. SystemTap fournit des informations similaires à des celles qui sont données par des outils comme **netstat**, **ps**, **top**, et **iostat** ; cependant SystemTap est conçu pour offrir davantage d'options d'analyse et de filtrage pour les informations rassemblées.

Même si l'utilisation d'OProfile est suggérée en cas de collection de données sur où et pourquoi un processeur passe du temps dans une zone de code particulière, il est moins utile lors de la recherche des raisons pour lesquelles un processeur reste inactif.

Vous pourriez souhaiter utiliser SystemTap lors de l'instrumentation d'endroits spécifiques du code. Comme SystemTap vous permet d'exécuter l'instrumentation du code sans avoir à arrêter et redémarrer le code instrumenté, cela est particulièrement utile pour instrumenter le noyau et ses démons.

Pour obtenir des informations supplémentaires sur SystemTap, veuillez consulter le Guide du débutant SystemTap.

23.12. RESSOURCES SUPPLÉMENTAIRES

Pour en savoir plus sur OProfile et pour savoir comment le configurer, veuillez consulter les ressources suivantes.

Documentation installée

- **/usr/share/doc/oprofile-*version*/oprofile.html** — *OProfile Manual*
- Page du manuel **oprofile(1)** — explique **opcontrol**, **opreport**, **opannotate**, et **ophelp**
- Page du manuel **operf(1)**

Documentation en ligne

- http://oprofile.sourceforge.net/ — documentation en amont contenant des documents, des listes de diffusion, des canaux IRC et d'autres informations sur le projet OProfile. Sur Red Hat Enterprise Linux 7, la version 0.9.9. d'OProfile est fournie.

Voir aussi

- Le Guide du débutant SystemTap — fournit des instructions de base sur la manière d'utiliser SystemTap afin de contrôler les différents sous-systèmes de Red Hat Enterprise Linux avec plus de détails.

PARTIE VII. CONFIGURATION DU NOYAU, DE MODULES ET DE PILOTES

Cette partie traite des différents outils assistant les administrateurs dans la personnalisation du noyau.

CHAPITRE 24. UTILISER LE CHARGEUR DE DÉMARRAGE GRUB 2

Red Hat Enterprise Linux 7 est distribué avec la version 2 du chargeur de démarrage GNU GRUB (« GRand Unified Boot loader »), qui permet à l'utilisateur de sélectionner un système d'exploitation ou un noyau à charger pendant le démarrage système. GRUB 2 permet également à l'utilisateur de passer des arguments au noyau.

24.1. INTRODUCTION À GRUB 2

GRUB 2 lit sa configuration dans le fichier **/boot/grub2/grub.cfg** pour les machines traditionnelles basées BIOS et dans le fichier **/boot/efi/EFI/redhat/grub.cfg** pour les machines UEFI. Ce fichier contient des informations de menu.

Le fichier de configuration GRUB 2, **grub.cfg**, est généré en cours d'installation, ou bien, en invoquant l'utilitaire /**usr**/**sbin**/**grub2-mkconfig**, et est mis à jour par **grubby** automatiquement à chaque fois qu'un nouveau noyau est installé. Quand il est regénéré manuellement par **grub2-mkconfig**, le fichier est regénéré selon les fichiers de modèle qui se trouvent dans **/etc/grub.d/**, et les paramètres de configuration personnalisés du fichier **/etc/default/grub**. Les changements apportées à **grub.cfg** seront perdues à chaque fois que **grub2-mkconfig** est utilisé pour regénérer le fichier, donc il faut faire refléter les changements manuels dans le fichier **/etc/default/grub** également.

Les opérations normales de **grub.cfg**, comme la suppression ou le rajout de nouveaux noyaux, doivent être effectuées par l'outil **grubby** et pour les scripts, par l'outil **new-kernel-pkg**. Si vous utilisez **grubby** pour changer le noyau par défaut, les changements seront hérités quand vous installerez des nouveaux noyaux. Pour plus d'informations sur **grubby**, voir Section 24.4, « Effectuer des Changements persistnats à un menu GRUB 2 par l'outil grubby ».

Le fichier **/etc/default/grub** est utilisé par l'outil **grub2-mkconfig**, utilisé lui-même par **anaconda** quand on crée **grub.cfg** au cours du processus d'installation, et il peut être utilisé en cas d'échec du système, comme par exemple, au cas où les configurations du chargeur de démarrage ont besoin d'être créées à nouveau. En général, il n'est pas conseillé de remplacer le fichier **grub.cfg** en exécutant **grub2-mkconfig** manuellement, sauf en cas de dernier recours. Notez que tout changement manuel à **/etc/default/grub** entraîne la création à nouveau du fichier **grub.cfg**.

Entrées de menu dans grub.cfg

Parmi les divers snippets et directives, le fichier de configuration **grub.cfg** contient un ou plusieurs bloc(s) **menuentry**, qui représentent chacun une entrée unique du menu de démarrage GRUB 2. Ces blocs commencent toujours par le mot-clé **menuentry** suivi d'un titre, d'une liste d'options, et d'une accolade d'ouverture, et se terminant avec une accolade de fermeture. Tout ce qui se trouve entre l'accolade d'ouverture et l'accolade de fermeture doit être indenté. Par exemple, ci-dessous figure un exemple de bloc **menuentry** pour Red Hat Enterprise Linux 7 avec le noyau Linux 3.8.0-0.40.el7.x86_64 :

```
menuentry 'Red Hat Enterprise Linux Server' --class red --class gnu-linux
--class gnu --class os $menuentry_id_option 'gnulinux-simple-c60731dc-
9046-4000-9182-64bdcce08616' {
        load_video
        set gfxpayload=keep
        insmod gzio
        insmod part_msdos
        insmod xfs
        set root='hd0,msdos1'
```

```
        if [ x$feature_platform_search_hint = xy ]; then
          search --no-floppy --fs-uuid --set=root --hint-bios=hd0,msdos1 -
-hint-efi=hd0,msdos1 --hint-baremetal=ahci0,msdos1 --hint='hd0,msdos1'
19d9e294-65f8-4e37-8e73-d41d6daa6e58
        else
          search --no-floppy --fs-uuid --set=root 19d9e294-65f8-4e37-8e73-
d41d6daa6e58
        fi
        echo    'Loading Linux 3.8.0-0.40.el7.x86_64 ...'
        linux16   /vmlinuz-3.8.0-0.40.el7.x86_64 root=/dev/mapper/rhel-
root ro rd.md=0 rd.dm=0 rd.lvm.lv=rhel/swap crashkernel=auto rd.luks=0
vconsole.keymap=us rd.lvm.lv=rhel/root rhgb quiet
        echo    'Loading initial ramdisk ...'
        initrd  /initramfs-3.8.0-0.40.el7.x86_64.img
}
```

Chaque bloc **menuentry** qui représente un noyau Linux installé contient **linux** sur 64 bits IBM POWER Series, **linux16** sur systèmes basés BIOS x86_64, et **linuxefi** sur systèmes basés UEFI. Puis les directives **initrd** suivies par le chemin vers le noyau et l'image **initramfs**, respectivement. Si une autre partition **/boot** a été créée, les chemins vers le noyau et l'image **initramfs** sont relatifs à **/boot**. Dans l'exemple ci-dessus, la ligne **initrd /initramfs-3.8.0-0.40.el7.x86_64.img** signifie que l'image **initramfs** est en fait située sur **/boot/initramfs-3.8.0-0.40.el7.x86_64.img** lorsque le système de fichiers **root** est monté, et de même pour le chemin du noyau.

Le numéro de version du noyau donné sur la ligne **linux16 /vmlinuz-kernel_version** doit correspondre au numéro de version de l'image **initramfs** donnée sur la ligne **initrd /initramfs-kernel_version.img** de chaque bloc **menuentry**. Pour obtenir des informations supplémentaires sur la manière de vérifier l'image du disque RAM initial, veuillez consulter la Section 25.5, « Vérifier l'image de disque RAM initial ».

NOTE

Dans les blocs **menuentry**, la directive **initrd** doit indiquer l'emplacement (relatif au répertoire **/boot** s'il se trouve sur une autre partition) du fichier **initramfs** correspondant à la même version du noyau. Cette directive est appellée **initrd** car l'outil précédant, qui avait créé les images de disque RAM initial, **mkinitrd**, créait des fichiers nommés des fichiers **initrd**. La directive **grub.cfg** reste **initrd** afin de rester compatible avec d'autres outils. La convention de dénomination de fichiers des systèmes utilisant l'utilitaire **dracut** pour créer l'image du disque RAM initial est comme suit : **initramfs-*kernel_version*.img**.

Pour obtenir des informations sur l'utilisation de **Dracut**, veuillez consulter la Section 25.5, « Vérifier l'image de disque RAM initial ».

24.2. CONFIGURER LE CHARGEUR DE DÉMARRAGE GRUB 2

Les changements au menu GRUB2 peuvent être effectués de façon temporaire au moment du démarrage (boot) et être rendus persistants dans un système alors que celui-ci est en cours d'exécution, ou dans le cadre d'une création de fichier de configuration GRUB 2.

- Pour effectuer des changements non persistants au menu GRUB 2, voir Section 24.3, « Effectuer des Changements temporaires à un menu GRUB 2 ».

- Pour effectuer des changements persistants à un système en cours d'exécution, voir Section 24.4, « Effectuer des Changements persistnats à un menu GRUB 2 par l'outil grubby ».
- Pour obtenir des informations sur la façon de créer ou de personnaliser un fichier de configuration GRUB 2, voir Section 24.5, « Personnalisation du fichier de configuration GRUB 2 ».

24.3. EFFECTUER DES CHANGEMENTS TEMPORAIRES À UN MENU GRUB 2

Procédure 24.1. Effectuer des Changements temporaires à une Saisie de menu de noyau

Pour changer les paramètres de noyau pendant un processus de démarrage simple, procédez ainsi :

1. Lancez le système, et, sur l'écran de démarrage GRUB 2, déplacez le curseur sur l'entrée de menu que vous souhaitez modifier, et appuyez sur la touche **e** pour effectuer des modifications.
2. Déplacez le curseur sur la ligne de commande de noyau que vous souhaitez. La ligne de commande de noyau commence par **linux** sur IBM Power Series 64 bits, **linux16** sur les systèmes basés BIOS x86-64, ou **linuxefi** sur les systèmes UEFI.
3. Placez le curseur en fin de ligne.

   Appuyez sur **Ctrl**+**a** et **Ctrl**+**e** pour directement passer au début ou à la fin d'une ligne, respectivement. Sur certains systèmes, les touches **Début** et **Fin** peuvent également fonctionner.
4. Modifiez les paramètres de noyau selon les besoins. Par exemple, pour exécuter le système en mode d'urgence, veuilez ajouter le paramètre *emergency* à la fin de la ligne **linux16** :

   ```
   linux16      /vmlinuz-3.10.0-0.rc4.59.el7.x86_64
   root=/dev/mapper/rhel-root ro rd.md=0 rd.dm=0 rd.lvm.lv=rhel/swap
   crashkernel=auto rd.luks=0 vconsole.keymap=us rd.lvm.lv=rhel/root
   rhgb quiet emergency
   ```

   Les paramètres **rhgb** et **quiet** peuvent être supprimés afin d'activer les messages système.

   Ces paramètres de configuration ne sont pas persistants et s'appliquent à un seul démarrage (boot). Pour rendre les modifications à une saisie de menu persistantes, utiliser l'outil suivant **grubby**. Voir la section intitulée « Ajouter ou Supprimer des Arguments d'une entrée de Menu GRUB » pour obtenir plus d'informations sur la façon d'utiliser **grubby**.

24.4. EFFECTUER DES CHANGEMENTS PERSISTNATS À UN MENU GRUB 2 PAR L'OUTIL GRUBBY

L'outil **grubby** peut être utilisé pour lire des informations, et effectuer des modifications permanentes au fichier **grub.cfg**. Il nous permet, par exemple, de modifier les entrées de menu GRUB pour préciser les arguments à faire passer au noyau au démarrage du système, ou changer le noyau par défaut.

Dans Red Hat Enterprise Linux 7, si **grubby** est invoqué manuellement, sans indiquer de fichier de configuration GRUB, il recherchera **/etc/grub2.cfg** par défaut, qui est un lien symbolic du fichier **grub.cfg**, dont l'emplacement est dépendant de l'architecture. S'il ne trouve pas ce fichier, il cherchera une valeur par défaut dépendant de l'architecture.

Recherche du Noyau par défaut

Pour trouver le nom de fichier du noyau par défaut, saisir une commande sur ce modèle :

```
~]# grubby --default-kernel
/boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
```

Pour trouver le numéro d'indexation du noyau par défaut, saisir une commande sur ce modèle :

```
~]# grubby --default-index
0
```

Changer l'entrée de démarrage par défaut

Pour effectuer un changement permanent dans le noyau désigné comme étant le noyau par défaut, utiliser la commande **grubby** sur le modèle suivant :

```
~]# grubby --set-default /boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
```

Visualiser l'entrée de menu GRUB d'un noyau

Pour répertorier toutes les entrées de menu, saisir une commande comme suit :

```
~]$ grubby --info=ALL
```

Dans les systèmes UEFI, toutes les commandes **grubby** doivent être saisies en tant qu'utilisateur **root**.

Pour visualiser l'entrée de menu GRUB d'un noyau en particulier, saisir une commande sur ce modèle :

```
~]$ grubby --info /boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
index=0
kernel=/boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
args="ro rd.lvm.lv=rhel/root crashkernel=auto  rd.lvm.lv=rhel/swap
vconsole.font=latarcyrheb-sun16 vconsole.keymap=us rhgb quiet
LANG=en_US.UTF-8"
root=/dev/mapper/rhel-root
initrd=/boot/initramfs-3.10.0-229.4.2.el7.x86_64.img
title=Red Hat Enterprise Linux Server (3.10.0-229.4.2.el7.x86_64) 7.0
(Maipo)
```

Utiliser la saisie semi-automatique pour voir quels noyaux sont disponibles dans le répertoire **/boot**.

Ajouter ou Supprimer des Arguments d'une entrée de Menu GRUB

L'option **--update-kernel** peut être utilisée pour mettre à jour une entrée de menu si utilisé en conjonction à **--args** pour ajouter de nouveaux arguments et **--remove-arguments** pour supprimer des arguments existants. Ces options acceptent une liste séparée par des espaces avec des guillemets :

```
grubby --remove-args="argX argY" --args="argA argB" --update-kernel
/boot/kernel
```

Pour ajouter ou supprimer des arguments de l'entrée du menu GRUB, utiliser une commande du modèle suivant :

```
~]# grubby --remove-args="rhgb quiet" --args=console=ttyS0,115200 --
update-kernel /boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
```

Cette commande supprime l'argument de démarrage graphique de Red Hat, rend visible le message boot, et ajoute une console série. Comme les arguments de console seront ajoutés en fin de ligne, la nouvelle console aura préséance sur toutes les autres consoles configurées.

Pour vérifier les changements, utiliser l'option de commande **--info** comme suit :

```
~]# grubby --info /boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
index=0
kernel=/boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
args="ro rd.lvm.lv=rhel/root crashkernel=auto  rd.lvm.lv=rhel/swap
vconsole.font=latarcyrheb-sun16 vconsole.keymap=us LANG=en_US.UTF-8
ttyS0,115200"
root=/dev/mapper/rhel-root
initrd=/boot/initramfs-3.10.0-229.4.2.el7.x86_64.img
title=Red Hat Enterprise Linux Server (3.10.0-229.4.2.el7.x86_64) 7.0
(Maipo)
```

Mise à jour de tous les Menus de noyau par les mêmes Arguments

Pour ajouter les mêmes arguments de démarrage de noyau à toutes les entrées de menu de noyau, saisir une commande sur ce modèle :

```
~]# grubby --update-kernel=ALL --args=console=ttyS0,115200
```

Le paramètre **--update-kernel** accepte également DEFAULT, ou une liste de numéros d'indexes de noyaux séparés par des virgules.

Changer un argument de noyau

Pour changer une valeur en argument de noyau existant, spécifiez l'argument à nouveau, en changeant la valeur suivant les besoins. Ainsi, pour changer la taille de la police de console, utiliser la commande suivante :

```
~]# grubby --args=vconsole.font=latarcyrheb-sun32 --update-kernel
/boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
index=0
kernel=/boot/vmlinuz-3.10.0-229.4.2.el7.x86_64
args="ro rd.lvm.lv=rhel/root crashkernel=auto  rd.lvm.lv=rhel/swap
vconsole.font=latarcyrheb-sun32 vconsole.keymap=us LANG=en_US.UTF-8"
root=/dev/mapper/rhel-root
initrd=/boot/initramfs-3.10.0-229.4.2.el7.x86_64.img
title=Red Hat Enterprise Linux Server (3.10.0-229.4.2.el7.x86_64) 7.0
(Maipo)
```

Voir la page man **grubby(8)** pour obtenir d'autres options de commande.

24.5. PERSONNALISATION DU FICHIER DE CONFIGURATION GRUB 2

Les scripts GRUB 2 effectuent des recherches dans l'ordinateur de l'utilisateur et créent un menu de démarrage basé sur les systèmes d'exploitation trouvés par les scripts. Pour refléter les options de démarrage système les plus récentes, le menu de démarrage est automatiquement régénéré lorsque le noyau est mis à jour ou lorsqu'un nouveau noyau est ajouté.

Cependant, les utilisateurs pourraient souhaiter générer un menu contenant des entrées spécifiques ou pour que les entrées se trouvent dans un ordre particulier. GRUB 2 permet d'effectuer une personnalisation de base du menu de démarrage afin de donner aux utilisateurs contrôle de ce qui

s'affiche à l'écran.

GRUB 2 utilise une série de scripts pour générer le menu ; ces scripts se trouvent dans le répertoire **/etc/grub.d/**. Les fichiers suivants sont inclus :

- **00_header**, qui charge les paramètres GRUB 2 du fichier **/etc/default/grub**.
- **01_users**, qui lit le mot de passe du superutilisateur dans le fichier **user.cfg**. Dans Red Hat Enterprise Linux 7.0 et 7.1, ce fichier n'a été créé que lorsque le mot de passe boot a été défini dans le fichier kickstart au moment de l'installation, et il inclut le mot de passe défini en texte brut.
- **10_linux**, qui localise les noyaux dans la partition par défaut de Red Hat Enterprise Linux.
- **30_os-prober**, qui génère des entrées pour les systèmes d'exploitations trouvés sur d'autres partitions.
- **40_custom**, un modèle pouvant être utilisé pour créer des entrées de menu supplémentaires.

Les scripts du répertoire **/etc/grub.d/** sont lus en ordre alphabétique et peuvent ainsi être renommés pour modifier l'ordre de démarrage d'entrées spécifiques du menu.

IMPORTANT

Avec la clé **GRUB_TIMEOUT** définie sur **0** dans le fichier **/etc/default/grub**, GRUB 2 n'affiche pas la liste des noyaux démarrables lorsque le système est démarré. Pour afficher cette liste pendant le démarrage, appuyez continuellement sur n'importe quelle touche alphanumérique lorsque les informations BIOS sont affichées, GRUB 2 vous présentera le menu GRUB.

24.5.1. Changer l'entrée de démarrage par défaut

Par défaut, la clé de la directive **GRUB_DEFAULT** dans le fichier **/etc/default/grub** est le mot « **saved** ». Cela ordonne à GRUB 2 de charger le noyau spécifié par la directive « **saved_entry** » dans le fichier d'environnement GRUB 2, qui se trouve dans **/boot/grub2/grubenv**. Vous pouvez également définir un autre enregistrement GRUB par défaut en utilisant la commande **grub2-set-default**, qui met à jour le fichier d'environnement GRUB 2.

Par défaut, la valeur **saved_entry** est définie avec le nom du noyau installé le plus récent du type de paquet kernel. Ceci est défini dans **/etc/sysconfig/kernel** par les directives **UPDATEDEFAULT** et **DEFAULTKERNEL**. Le fichier peut être affiché par l'utilisateur **root** comme suit :

```
~]# cat /etc/sysconfig/kernel
# UPDATEDEFAULT specifies if new-kernel-pkg should make
# new kernels the default
UPDATEDEFAULT=yes

# DEFAULTKERNEL specifies the default kernel package type
DEFAULTKERNEL=kernel
```

La directive **DEFAULTKERNEL** spécifie quel type de paquet sera utilisé par défaut. Installer un paquet de type kernel-debug ne changera pas le noyau par défaut tant que **DEFAULTKERNEL** est défini sur les paquets de type kernel.

GRUB 2 prend en charge une valeur numérique comme clé pour que la directive **saved_entry** change l'ordre par défaut dans lequel les systèmes d'exploitation sont chargés. Pour spécifier quel système d'exploitation doit être chargé en premier, veuillez inclure son numéro dans la commande **grub2-set-default**. Exemple :

```
~]# grub2-set-default 2
```

Remarquez que la position d'une entrée de menu dans la liste est dénotée par un nombre commençant par zéro. Ainsi, dans l'exemple ci-dessus, la troisième entrée sera chargée. Cette valeur sera remplacée par le nom du prochain noyau devant être installé.

Pour forcer un système à toujours utiliser une entrée de menu particulière, veuillez utiliser le nom de l'entrée du menu comme clé de la directive **GRUB_DEFAULT** dans le fichier **/etc/default/grub**. Pour répertorier les entrées du menu disponibles, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# awk -F\' '$1=="menuentry " {print $2}' /etc/grub2.cfg
```

Le nom de fichier **/etc/grub2.cfg** est un lien symbolic qui mène au fichier **grub.cfg**, dont l'emplacement dépendra de l'architecture. Pour des raisons de fiabilité, le lien symbolic n'est pas utilisé dans d'autres exemples dans ce chapitre. Il vaut mieux utiliser des chemins absolus quand on écrit dans un fichier, surtout quand on le répare.

Les changements apportés à **/etc/default/grub** requièrent de régénérer le fichier **grub.cfg** comme suit :

- Sur les machines basées BIOS, exécutez la commande suivante en tant qu'utilisateur **root** :

  ```
  ~]# grub2-mkconfig -o /boot/grub2/grub.cfg
  ```

- Sur les machines basées UEFI, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

  ```
  ~]# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
  ```

24.5.2. Modifier une entrée de menu

Si vous devez préparer un nouveau fichier GRUB 2 avec des paramètres différents, modifier les valeurs de la clé **GRUB_CMDLINE_LINUX** dans le fichier **/etc/default/grub**. Remarquez que vous pouvez spécifier plusieurs paramètres pour la clé **GRUB_CMDLINE_LINUX**, similairement à l'ajout de paramètres dans le menu de démarrage GRUB 2. Exemple :

```
GRUB_CMDLINE_LINUX="console=tty0 console=ttyS0,9600n8"
```

, où **console=tty0** est le premier terminal virtuel et **console=ttyS0** est le terminal série à utiliser.

Les changements apportés à **/etc/default/grub** requièrent de régénérer le fichier **grub.cfg** comme suit :

- Sur les machines basées BIOS, exécutez la commande suivante en tant qu'utilisateur **root** :

  ```
  ~]# grub2-mkconfig -o /boot/grub2/grub.cfg
  ```

- Sur les machines basées UEFI, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

```
~]# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

24.5.3. Ajouter une nouvelle entrée

Lors de l'exécution de la commande **grub2-mkconfig**, GRUB 2 recherche des noyaux Linux et d'autres systèmes d'exploitation basés sur les fichiers situés dans le répertoire **/etc/grub.d/**. Le script **/etc/grub.d/10_linux** recherche les noyaux Linux installés sur la même partition. Le script **/etc/grub.d/30_os-prober** recherche d'autres systèmes d'exploitation. Des entrées du menu sont également automatiquement ajoutées au menu de démarrage lors de la mise à jour du noyau.

Le fichier **40_custom** situé dans le répertoire **/etc/grub.d/** est un modèle pour les entrées personnalisées et ressemble à ce qui suit :

```
#!/bin/sh
exec tail -n +3 $0
# This file provides an easy way to add custom menu entries.  Simply type
the
# menu entries you want to add after this comment.  Be careful not to
change
# the 'exec tail' line above.
```

Ce fichier peut être modifié ou copié. Remarquez qu'une entrée du menu valide doit inclure au minimum :

```
menuentry "<Title>"{
<Data>
}
```

24.5.4. Créer un menu personnalisé

Si vous ne souhaitez pas que les entrées du menu soient mises à jour automatiquement, vous pouvez créer un menu personnalisé.

IMPORTANT

Avant de continuer, effectuez une copie de sauvegarde du contenu du répertoire **/etc/grub.d/** au cas où vous devriez annuler les changement ultérieurement.

NOTE

Remarquez que la modification du fichier **/etc/default/grub** n'a aucun effet sur la création de menus personnalisés.

1. Sur les machines basées BIOS, copiez le contenu de **/boot/grub2/grub.cfg**, ou le contenu de **/boot/efi/EFI/redhat/grub.cfg** sur les machines UEFI. Ajoutez le contenu de **grub.cfg** dans le fichier **/etc/grub.d/40_custom** sous les lignes d'en-tête. La partie exécutable du script **40_custom** doit être préservée.

2. Dans le contenu intégré au fichier **/etc/grub.d/40_custom**, seuls les blocs **menuentry** sont

nécessaires à la création d'un menu personnalisé. Les fichiers **/boot/grub2/grub.cfg** et **/boot/efi/EFI/redhat/grub.cfg** peuvent contenir des spécifications de fonctions ainsi que d'autres contenus au-dessus et au-dessous des blocs **menuentry**. Si vous avez inclus ces lignes non nécessaires dans le fichier **40_custom** dans l'étape précédente, veuillez les supprimer.

Voici un exemple de script **40_custom** personnalisé :

```
#!/bin/sh
exec tail -n +3 $0
# This file provides an easy way to add custom menu entries.  Simply
type the
# menu entries you want to add after this comment.  Be careful not
to change
# the 'exec tail' line above.

menuentry 'First custom entry' --class red --class gnu-linux --class
gnu --class os $menuentry_id_option 'gnulinux-3.10.0-67.el7.x86_64-
advanced-32782dd0-4b47-4d56-a740-2076ab5e5976' {
        load_video
        set gfxpayload=keep
        insmod gzio
        insmod part_msdos
        insmod xfs
        set root='hd0,msdos1'
        if [ x$feature_platform_search_hint = xy ]; then
          search --no-floppy --fs-uuid --set=root --
hint='hd0,msdos1'  7885bba1-8aa7-4e5d-a7ad-821f4f52170a
        else
          search --no-floppy --fs-uuid --set=root 7885bba1-8aa7-
4e5d-a7ad-821f4f52170a
        fi
        linux16 /vmlinuz-3.10.0-67.el7.x86_64 root=/dev/mapper/rhel-
root ro rd.lvm.lv=rhel/root vconsole.font=latarcyrheb-sun16
rd.lvm.lv=rhel/swap vconsole.keymap=us crashkernel=auto rhgb quiet
LANG=en_US.UTF-8
        initrd16 /initramfs-3.10.0-67.el7.x86_64.img
}
menuentry 'Second custom entry' --class red --class gnu-linux --
class gnu --class os $menuentry_id_option 'gnulinux-0-rescue-
07f43f20a54c4ce8ada8b70d33fd001c-advanced-32782dd0-4b47-4d56-a740-
2076ab5e5976' {
        load_video
        insmod gzio
        insmod part_msdos
        insmod xfs
        set root='hd0,msdos1'
        if [ x$feature_platform_search_hint = xy ]; then
          search --no-floppy --fs-uuid --set=root --
hint='hd0,msdos1'  7885bba1-8aa7-4e5d-a7ad-821f4f52170a
        else
          search --no-floppy --fs-uuid --set=root 7885bba1-8aa7-
4e5d-a7ad-821f4f52170a
        fi
        linux16 /vmlinuz-0-rescue-07f43f20a54c4ce8ada8b70d33fd001c
root=/dev/mapper/rhel-root ro rd.lvm.lv=rhel/root
```

```
vconsole.font=latarcyrheb-sun16 rd.lvm.lv=rhel/swap
vconsole.keymap=us crashkernel=auto rhgb quiet
        initrd16 /initramfs-0-rescue-
07f43f20a54c4ce8ada8b70d33fd001c.img
}
```

3. Supprimez tous les fichier du répertoire **/etc/grub.d/** sauf les fichiers suivants :
   - **00_header**,
   - **40_custom**,
   - **01_users** (s'il existe),
   - et **README**.

   Alternativement, si vous souhaitez conserver les fichiers dans le répertoire **/etc/grub2.d/**, rendez-les non exécutables par la commande **chmod a-x <file_name>**.
4. Modifiez, ajoutez, ou supprimez des entrées de menu dans le fichier **40_custom** comme vous le souhaitez.
5. Créez à nouveau le fichier **grub.cfg** en exécutant la commande **grub2-mkconfig -o** :
   - Sur les machines basées BIOS, exécutez la commande suivante en tant qu'utilisateur **root** :

     ```
     ~]# grub2-mkconfig -o /boot/grub2/grub.cfg
     ```
   - Sur les machines basées UEFI, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

     ```
     ~]# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
     ```

24.6. PROTECTION DE GRUB 2 PAR UN MOT DE PASSE

GRUB 2 donne deux types de protection de mot de passe :

- Un mot de passe est exigé pour pouvoir modifier les entreés de menu *mais pas* pour les entrées de menu boot existantes.
- Un mot de passe est requis pour pouvoir modifier les entrées de menu *et* pour pouvoir démarrer un, plusieurs ou toutes les entrées de menu.

Configurer GRUB2 pour qu'un mot de passe ne soit exigé que pour les saisie de modification.

Pour demander une authentification de mot de passe afin de modifier les entrées GRUB2, suivre les étapes suivantes :

1. Exécutez la commande **grub2-setpassword** en tant qu'utilisateur root :

   ```
   ~]# grub2-setpassword
   ```
2. Saisir et confirmer le mot de passe :

```
Enter password:
Confirm password:
```

En suivant cette procédure, vous créerez un fichier **/boot/grub2/user.cfg** qui contient le hachage du mot de passe. L'utilisateur de ce mot de passe, **root**, est défini dans le fichier **/boot/grub2/grub.cfg**. Avec ce changement, pour modifier une entrée boot au moment de l'amorçage nécessite que vous spécifiez le nom d'utilisateur **root** et votre mot de passe.

Configurer GRUB 2 pour qu'un mot de passe soit exigé pour les saisies de modification et de booting.

Définir un mot de passe avec **grub2-setpassword** protègent les entrées de menu de modifications non autorisées, mais pas de booting non autorisé. Pour demander un mot de passe de booting d'entrée, suivre ces étapes après avoir défini le mot de passe avec **grub2-setpassword** :

AVERTISSEMENT

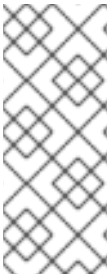
Si vous oubliez le mot de passe GRUB2, il n'y aura pas de booting possible pour les entrées que vous allez reconfigurer dans la procédure suivante.

1. Ouvrir un fichier **/boot/grub2/grub.cfg** .
2. Cherchez l'entrée boot que vous souhaitez protéger avec le mot de passe en cherchant des lignes commençant par **menuentry**.
3. Supprimer le paramètre **--unrestricted** du bloc d'entrée de menu comme dans l'exemple :

```
[file contents truncated]
menuentry 'Red Hat Enterprise Linux Server (3.10.0-
327.18.2.rt56.223.el7_2.x86_64) 7.2 (Maipo)' --class red --class
gnu-linux --class gnu --class os --unrestricted $menuentry_id_option
'gnulinux-3.10.0-327.el7.x86_64-advanced-c109825c-de2f-4340-a0ef-
4f47d19fe4bf' {
        load_video
        set gfxpayload=keep
[file contents truncated]
```

4. Enregistrer et fermer le fichier.

Vous devez maintenant saisir le nom d'utilisateur et le mot de passe **root** pour le booting.



NOTE

Les changements au **/boot/grub2/grub.cfg** persistent quand de nouvelles versions de noyau sont installées, mais sont perdues quand on génère à nouveau **grub.cfg** par la commande **grub2-mkconfig**. Ainsi, pour conserver la protection du mot de passe, il vous faudra utiliser la procédure ci-dessus à chaque fois que vous utiliserez **grub2-mkconfig**.

NOTE

Si vous supprimez le paramètre **--unrestricted** de chaque entrée de menu dans le fichier **/boot/grub2/grub.cfg**, alors tous les noyaux nouvellement installés auront une entrée de menu créée sans **--unrestricted** et donc, héritera automatiquement la protection du mot de passe.

Mots de passe définis avant la mise à jour à Red Hat Enterprise Linux 7.2
L'outil **grub2-setpassword** a été ajouté dans Red Hat Enterprise Linux 7.2 et représente maintenant la méthode de choix pour définir les mots de passe GRUB 2. Ceci est en contraste avec les anciennes versions de Red Hat Enterprise Linux, où les entrées boot avaient besoin d'être spécifiées manuellement dans le fichier **/etc/grub.d/40_custom**, et les superutilisateurs dans le fichier **/etc/grub.d/01_users**.

Utilisateurs GRUB 2 supplémentaires
Les entrées boot sans paramètre **--unrestricted** nécessitent le mot de passe root. Cependant, GRUB 2 permet également de créer des utilisateurs non-root supplémentaires qui peuvent initialiser ces entrés sans mot de passe. Vous aurez quand même besoin d'un mot de passe pour modifier ces entrées. Pour obtenir des informations sur la façon de créer ces utilisateurs, voir le manuel GRUB 2.

24.7. RÉINSTALLER GRUB 2

Réinstaller GRUB 2 est une manière pratique de corriger certains problèmes, habituellement causés par une installation incorrecte de GRUB 2, par des fichiers manquants, ou par un système rompu. D'autres raisons de réinstaller GRUB 2 peuvent inclure :

- Mise à niveau d'une version précédente de GRUB.
- L'utilisateur requiert que le chargeur de démarrage GRUB 2 contrôle les systèmes d'exploitation installés. Cependant, certains systèmes d'exploitation sont installés avec leur propre chargeur de démarrage. Réinstaller GRUB 2 redonne contrôle sur le système d'exploitation souhaité.
- Ajouter les informations de démarrage sur un autre disque.

24.7.1. Réinstaller GRUB 2 sur des machines basées BIOS

Lors de l'utilisation de la commande **grub2-install**, les informations de démarrage sont mises à jour et les fichiers manquants sont restaurés. Remarquez que les fichiers sont uniquement restaurés s'ils ne sont pas corrompus.

Veuillez utiliser la commande **grub2-install *device*** pour réinstaller GRUB 2 si le système fonctionne normalement. Par exemple, si **sda** correspond à votre *périphérique* :

```
~]# grub2-install /dev/sda
```

24.7.2. Réinstaller GRUB 2 sur des machines basées UEFI

Lors de l'utilisation de la commande **yum reinstall grub2-efi shim**, les informations de démarrage sont mises à jour et les fichiers manquants sont restaurés. Notez que les fichiers sont uniquement restaurés s'ils ne sont pas corrompus.

Veuillez utiliser la commande **yum reinstall grub2-efi shim** pour réinstaller GRUB 2 si le système fonctionne normalement. Exemple :

```
~]# yum reinstall grub2-efi shim
```

24.7.3. Reparamétrer et réinstaller GRUB 2

Cette méthode supprime complètement tous les fichiers de configuration et paramètres système de GRUB 2. Appliquez cette méthode pour réinitialiser tous les paramètres de configuration sur leurs valeurs par défaut. La suppression de tous les fichiers de configuration et la réinstallation conséquente de GRUB 2 corrige les échecs causés par les fichiers corrompus et par la configuration incorrecte. Pour faire cela, veuillez effectuer les étapes suivantes en tant qu'utilisateur **root** :

1. Exécutez la commande **rm /etc/grub.d/*** ;
2. Exécutez la commande **rm /etc/sysconfig/grub** ;
3. Pour les systèmes EFI **uniquement**, veuillez exécuter la commande suivante :

   ```
   ~]# yum reinstall grub2-efi shim grub2-tools
   ```

4. Pour les systèmes BIOS et EFI, exécuter cette commande :

   ```
   ~]# yum reinstall grub2-tools
   ```

5. Créez à nouveau le fichier **grub.cfg** en exécutant la commande **grub2-mkconfig -o** :
   - Sur les machines basées BIOS, exécutez la commande suivante en tant qu'utilisateur **root** :

     ```
     ~]# grub2-mkconfig -o /boot/grub2/grub.cfg
     ```

   - Sur les machines basées UEFI, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

     ```
     ~]# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
     ```

6. Puis suivez la procédure indiquée dans Section 24.7, « Réinstaller GRUB 2 » pour restaurer GRUB2 sur la partition **/boot/**.

24.8. GRUB 2 SUR UNE CONSOLE SÉRIE

Si vous utilisez des ordinateurs sans affichage ou sans clavier, il peut être très utile de contrôler les machines à travers des moyens de communication en série.

24.8.1. Configurer le menu GRUB 2

Pour configurer le système de façon à ce qu'il utilise un terminal en série uniquement pendant un processus de démarrage unique, lorsque le menu de démarrage GRUB 2 s'affiche, veuillez déplacer le curseur sur le noyau que vous souhaitez lancer, et appuyez sur la touche **e** pour modifier les paramètres du noyau. Supprimer les paramètres **rhgb** et **quit**, et ajouter les paramètres de console en fin de ligne **linux16** comme suit :

```
linux16      /vmlinuz-3.10.0-0.rc4.59.el7.x86_64 root=/dev/mapper/rhel-
root ro rd.md=0 rd.dm=0 rd.lvm.lv=rhel/swap crashkernel=auto rd.luks=0
vconsole.keymap=us rd.lvm.lv=rhel/root console=ttyS0,115200
```

Ces paramètres ne sont pas persistants et s'appliquent à un seul démarrage.

Pour rendre des changements persistants dans une entrée de menu du système, utiliser l'outil **grubby**. Ainsi, pour mettre à jour l'entrée du noyau par défaut, saisir la commande suivante :

```
~]# grubby --remove-args="rhgb quiet" --args=console=ttyS0,115200 --
update-kernel=DEFAULT
```

Le paramètre **--update-kernel** accepte également le mot clé **ALL**, ou une liste de numéros d'indexes de noyaux séparée par des virgules. Voir la section intitulée « Ajouter ou Supprimer des Arguments d'une entrée de Menu GRUB » pour plus d'informations sur la façon d'utiliser **grubby**.

Si vous avez besoin de construire un nouveau fichier de configuration GRUB 2, ajouter les deux lignes suivantes au fichier **/etc/default/grub** :

```
GRUB_TERMINAL="serial"
GRUB_SERIAL_COMMAND="serial --speed=9600 --unit=0 --word=8 --parity=no --
stop=1"
```

La première ligne désactive le terminal graphique. Remarquez que spécifier la clé **GRUB_TERMINAL** fait qu'elle prend précédent sur les valeurs **GRUB_TERMINAL_INPUT** et **GRUB_TERMINAL_OUTPUT**. Sur la seconde ligne, ajustez le débit en bauds, la parité, et les autres valeurs pour qu'elles correspondre à votre environnement et matériel. Un débit en bauds beaucoup plus élevé, par exemple **115200**, est préférable pour les tâches comme le suivi de fichiers journaux. Une fois les changements effectués dans le fichier **/etc/default/grub**, il sera nécessaire de mettre à jour le fichier de configuration GRUB 2.

Créez à nouveau le fichier **grub.cfg** en exécutant la commande **grub2-mkconfig -o** :

- Sur les machines basées BIOS, exécutez la commande suivante en tant qu'utilisateur **root** :

  ```
  ~]# grub2-mkconfig -o /boot/grub2/grub.cfg
  ```

- Sur les machines basées UEFI, veuillez exécuter la commande suivante en tant qu'utilisateur **root** :

  ```
  ~]# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
  ```

NOTE

Pour accéder au terminal GRUB à travers une connexion en série, une option supplémentaore doit être ajoutée à une définition du noyau afin que ce noyau particulier surveille une connexion en série. Par exemple :

```
console=ttyS0,9600n8
```

, où **console=ttyS0** est le terminal série à utiliser, **9600** est le débit en bauds, **n** signifie pas de parité, et **8** est la longueur de mot en bits. Un débit en bauds beaucoup plus élevé, par exemple **115200**, est préérable pour les tâches comme le suivi de fichiers journaux.

Pour obtenir des informations sur les paramètres de la console série, voir la section intitulée « Documentation installable et externe »

24.8.2. Utiliser « screen » pour se connecter à la console série

L'outil **screen** est utilisé comme terminal série tout à fait capable. Pour l'installer, veuillez exécuter en tant qu'utilisateur **root** :

```
~]# yum install screen
```

Pour vous connecter à votre machine en utilisant la console série, veuillez exécuter le format suivant :

```
screen /dev/console_port baud_rate
```

Par défaut, si aucune option n'est spécifiée, **screen** utilise le débit standard de 9600 bauds. Pour définir un débit de bauds différent, veuillez exécuter :

```
~]$ screen /dev/console_port 115200
```

Quand *console_port* correspond à **ttyS0**, ou **ttyUSB0**, etc.

Pour terminer la session dans **screen**, appuyez sur **Ctrl**+**a**, puis saisissez **:quit** et appuyez sur **Entrée**.

Affichez la page du manuel **screen(1)** pour des options supplémentaires et des informations détaillées.

24.9. MODIFICATION DU MENU DU TERMINAL PENDANT LE DÉMARRAGE

Les entrées du menu peuvent être modifiées et des arguments passés au noyau pendant le démarrage. Ceci est effectué en utilisant l'interface de l'éditeur d'entrées du menu, qui est déclenchée en appuyant sur la touche **e** dans une entrée du menu sélectionnée dans le menu du chargeur de démarrage. La touche **Esc** annule tout changement et recharge l'interface du menu standard. La touche **c** charge l'interface de ligne de commande.

L'interface de ligne de commande est la plus élémentaire des interfaces GRUB, mais c'est celle qui fournit le plus haut niveau de contrôle. La ligne de commande permet de saisir toute commande GRUB pertinente et de l'exécuter en appuyant sur la touche **Entrée**. Cette interface présente certaines fonctionnalités avancées similaires à **shell**, parmi lesquelles figurent la touche **Tab** pour l'achèvement automatique de ligne en fonction du contexte, et **Ctrl**+**a** pour se déplacer au début d'une ligne et **Ctrl**+**e** pour aller directement à la fin d'une ligne. De plus, les flèches, les touches de **flèches**, **Début**, **Fin**, et **Suppr.** fonctionnent de la même façon que sous le shell bash.

24.9.1. Démarrer en mode de secours

Le mode de secours fournit un environnement mono-utilisateur pratique et vous permet de réparer votre système dans des situations où il est impossible d'effectuer un processus de démarrage normal. En mode de secours, le système tente de monter tous les systèmes de fichiers locaux et lancer plusieurs services système importants, mais n'active pas d'interface réseau ou ne permet pas à davantage d'utilisateurs de se connecter au système au même moment. Sur Red Hat Enterprise Linux 7, le mode de secours est équivalent au mode mono-utilisateur et requiert le mot de passe **root**.

1. Pour entrer en mode de secours pendant le démarrage, sur l'écran de démarrage GRUB 2, appuyez sur la touche **e** pour effectuer des modifications.

2. Ajoutez le paramètre suivant à la fin de la ligne **linux** sur IBM Power Series 64 bits, sur la ligne **linux16** sur systèmes basés BIOS x86-64, ou sur la ligne **linuxefi** pour les systèmes UEFI :

```
systemd.unit=rescue.target
```

   Appuyez sur **Ctrl**+**a** et **Ctrl**+**e** pour directement passer au début ou à la fin d'une ligne, respectivement. Sur certains systèmes, les touches **Début** et **Fin** peuvent également fonctionner.

   Remarquez que des paramètres équivalents, ***1***, ***s***, et ***single***, peuvent également être passés au noyau.

3. Appuyez sur **Ctrl**+**x** pour démarrer le système avec le paramètre.

24.9.2. Démarrer en mode d'urgence

Le mode d'urgence fournit l'environnement le plus minimaliste possible et vous permet de réparer votre système même dans des situations où le système est incapable d'entrer en mode de secours. Dans le mode d'urgence, le système monte le système de fichiers **root** uniquement en lecture, il ne tentera pas de monter d'autres systèmes de fichiers locaux, n'activera pas d'interface réseau, et lancera uniquement quelques services essentiels. Dans Red Hat Enterprise Linux 7, le mode d'urgence requiert le mot de passe **root**.

1. Pour entrer en mode d'urgence, sur l'écran de démarrage GRUB 2, appuyez sur la touche **e** pour effectuer des modifications.

2. Ajoutez le paramètre suivant à la fin de la ligne **linux** sur IBM Power Series 64 bits, sur la ligne **linux16** sur systèmes basés BIOS x86-64, ou sur la ligne **linuxefi** pour les systèmes UEFI :

```
systemd.unit=emergency.target
```

   Appuyez sur **Ctrl**+**a** et **Ctrl**+**e** pour directement passer au début ou à la fin d'une ligne, respectivement. Sur certains systèmes, les touches **Début** et **Fin** peuvent également fonctionner.

   Remarquez que des paramètres équivalents, ***emergency*** et ***b*** peuvent également être passés sur le noyau.

3. Appuyez sur **Ctrl**+**x** pour démarrer le système avec le paramètre.

24.9.3. Démarrer le shell de débogage

Le shell de débogage **systemd** fournit un shell tout au début du processus de démarrage, qui peut être utilisé pour faire le diagnostic de problèmes de démarrage liés à **systemd**. Une fois dans le shell de débogage, les commandes de **systemctl** telles que **systemctl list-jobs**, et **systemctl list-units** peuvent être utilisées pour rechercher la cause des problèmes de démarrage. De plus, l'option **debug** peut être ajoutée à la ligne de commande du noyau pour augmenter le nombre de messages de journalisation. Pour **systemd**, l'option de ligne de commande du noyau **debug** est mainteant un raccourci de **systemd.log_level=debug**.

Procédure 24.2. Ajouter une commande de shell de débogage

Pour activer le shell de débogage pour cette session uniquement, procédez ainsi :

1. Sur l'écran de démarrage GRUB 2, déplacez le curseur sur l'entrée de menu que vous souhaitez modifier, et appuyez sur la touche **e** pour effectuer des modifications.

2. Ajoutez le paramètre suivant à la fin de la ligne **linux** sur IBM Power Series 64 bits, sur la ligne **linux16** sur systèmes basés BIOS x86-64, ou sur la ligne**linuxefi** pour les systèmes UEFI :

   ```
   systemd.debug-shell
   ```

   Vous pouvez ajouter l'option **debug**.

   Appuyez sur **Ctrl**+**a** et **Ctrl**+**e** pour directement passer au début ou à la fin d'une ligne, respectivement. Sur certains systèmes, les touches **Début** et **Fin** peuvent également fonctionner.

3. Appuyez sur **Ctrl**+**x** pour démarrer le système avec le paramètre.

Si nécessaire, le shell de débogage peut être configuré pour démarrer à chaque amorçage de système par la commande **systemctl enable debug-shell**. Sinon, l'outil **grubby** peut être utilisé pour faire des changements persistants à la ligne de commande du noyau dans le menu GRUB 2. Voir Section 24.4, « Effectuer des Changements persistnats à un menu GRUB 2 par l'outil grubby » pour obtenir plus d'informations sur la façon d'utiliser **grubby**.

AVERTISSEMENT

Activer le shell de débogage de façon permanente est un risque de sécurité car aucune authentification n'est requise. Le désactiver en fin de session.

Procédure 24.3. Se connecter à un shell de débogage

Lors du processus de démarrage, le **systemd-debug-generator** configurera le shell de débogage TTY9.

1. Appuyer sur **Ctrl**+**Alt**+**F9** pour vous connecter au shell de débogage. Si vous êtes dans une machine virtuelle, cette combinaison de touches devra être prise en charge par l'application de virtualisation. Ainsi, si vous utilisez **Virtual Machine Manager**, sélectionnez **Send Key** → **Ctrl+Alt+F9** dans le menu.

2. Le shell de débogage ne requiert aucune authentification, donc vous verrez sans doute une invite similaire à ce qui suit dans TTY9 : **[root@localhost /]#**

3. Si nécessaire, pour vérifier que vous êtes bien dans le shell de débogage, saisir une commande comme suit :

   ```
   /]# systemctl status $$
   ● debug-shell.service - Early root shell on /dev/tty9 FOR DEBUGGING
   ONLY
      Loaded: loaded (/usr/lib/systemd/system/debug-shell.service;
   disabled; vendor preset: disabled)
      Active: active (running) since Wed 2015-08-05 11:01:48 EDT; 2min
   ago
   ```

```
     Docs: man:sushell(8)
Main PID: 450 (bash)
  CGroup: /system.slice/debug-shell.service
          ├─ 450 /bin/bash
          └─1791 systemctl status 450
```

4. Pour retourner au shell par défaut, si l'amorçage a réussi, appuyez sur **Ctrl**+**Alt**+**F1**.

Pour faire le diagnostique des problèmes de démarrage, certaines unités de **systemd** peuvent être masquées en ajoutant **systemd.mask=*unit_name*** une ou plusieurs fois à la ligne de commande du noyau. Pour démarrer des processus supplémentaires lors du processus d'amorçage, ajouter **systemd.wants=*unit_name*** à la ligne de commande du noyau. La page man **systemd-debug-generator(8)** décrit ces options.

24.9.4. Changer et reconfigurer le mot de passe root

Définir le mot de passe **root** est obligatoire lors de l'installation de Red Hat Enterprise Linux 7. Si vous oubliez ou perdez le mot de passe **root**, il est possible de le réinitialiser. Cependant, les utilisateurs membre du groupe « wheel » peuvent changer le mot de passe **root** comme suit :

```
~]$ sudo passwd root
```

Remarquez que dans GRUB 2, réinitialiser le mot de passe n'est plus effectué en mode mono-utilisateur comme c'était le case avec la version de GRUB incluse dans Red Hat Enterprise Linux 6. Le mot de passe **root** est désormais requis pour opérer en mode mono-utilisateur (« **single-user** »), ainsi qu'en mode d'urgence (« **emergency** »).

Deux procédures pour réinitialiser le mot de passe **root** sont affichées ici :

- La Procédure 24.4, « Réinitialiser le mot de passe root en utilisant un disque d'installation » vous conduit à utiliser une invite shell sans avoir à modifier le menu GRUB. Cette procédure recommandée est la plus courte des deux. Vous pouvez utiliser un disque de démarrage ou un disque d'installation Red Hat Enterprise Linux 7 normal.

- La Procédure 24.5, « Réinitialiser le mot de passe root en utilisant rd.break » utilise **rd.break** pour interrompre le processus de démarrage avant que le contrôle ne passe d'**initramfs** à **systemd**. L'inconvénient de cette méthode est qu'elle requiert davantage d'étapes, y compris la modification du menu GRUB, et implique de devoir choisir entre un ré-étiquetage du fichier SELinux, ou la modification du mode « Enforcing » de SELinux, puis la restauration du contexte de sécurité SELinux pour **/etc/shadow/** lorsque le démarrage est terminé.

Procédure 24.4. Réinitialiser le mot de passe root en utilisant un disque d'installation

1. Démarrez le système et lorsque les informations BIOS sont affichées, sélectionnez l'option pour un menu de démarrage et sélectionnez de démarrer à partir du disque d'installation.

2. Choisissez « **Troubleshooting** » (Résolution de problèmes).

3. Choisissez « **Rescue a Red Hat Enterprise Linux System** » (Secourir un système Red Hat Enterprise Linux).

4. Choisissez **Continue**, qui est l'option par défaut. À ce moment, vous pourrez passer et l'étape suivante, et on vous demandera une phrase de passe si un système de fichiers chiffré est trouvé.

5. Appuyez sur **Valider** pour accepter les informations affichées jusqu'à ce que l'invite shell apparaisse.

6. Modifiez le système de fichiers **root** comme suit :

```
sh-4.2# chroot /mnt/sysimage
```

7. Saisir la commande **passwd** et suivez les instructions affichées sur la ligne de commande pour modifier le mot de passe **root**.

8. Supprimez le fichier **autorelable** pour empêcher le long ré-étiquetage SELinux du disque :

```
sh-4.2# rm -f /.autorelabel
```

9. Saisissez la commande **exit** pour quitter l'environnement **chroot**.

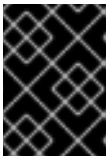
10. Saisissez la commande **exit** à nouveau pour reprendre l'initialisation et terminer le démarrage système.

Procédure 24.5. Réinitialiser le mot de passe root en utilisant rd.break

1. Lancez le système, sur l'écran de démarrage GRUB 2, appuyez sur la touche **e** pour effectuer des modifications.

2. Supprimez les paramètres **rhgb** et **quiet** situés à la fin, ou près de la fin de la ligne **linux16**, ou la ligne **linuxefi** sur les systèmes UEFI.

   Appuyez sur **Ctrl**+**a** et **Ctrl**+**e** pour directement passer au début ou à la fin d'une ligne, respectivement. Sur certains systèmes, les touches **Début** et **Fin** peuvent également fonctionner.



IMPORTANT

Les paramètres **rhgb** et **quiet** doivent être supprimés afin d'activer les messages système.

3. Ajoutez les paramètres suivants à la fin de la ligne **linux** sur IBM Power Series 64 bits, de la ligne **linux16** sur systèmes basés BIOS x86-64, ou de la ligne**linuxefi** sur les systèmes UEFI :

```
rd.break enforcing=0
```

   Ajouter l'option **enforcing=0** permet d'omettre le long processus de ré-étiquetage SELinux.

   initramfs s'arrêtera avant de donner le contrôle au noyau Linux (Linux kernel), vous permettant de travailler en utilisant le système de fichiers **root**.

   Remarquez que l'invite **initramfs** apparaîtra sur la dernière console spécifiée sur la ligne Linux.

4. Appuyez sur **Ctrl**+**x** pour démarrer le système avec les paramètres modifiés.

   Avec un système de fichiers chiffré, un mot de passe est requis. Cependant, l'invite du mot de

passe peut ne pas apparaître et être obscurcie par les messages de journalisation. Vous pouvez appuyer sur la touche **Retour Arrière** (« Backspace ») pour afficher l'invite. Relâchez la touche et saisissez le mot de passe du système de fichiers chiffré tout en ignorant les messages de journalisation.

L'invite **initramfs switch_root** s'affiche.

5. Le système de fichiers est monté en lecture seule sur **/sysroot/**. Vous n'aurez pas le droit de modifier le mot de passe si le système de fichiers n'est pas accessible en écriture.

   Remontez le système de fichier comme étant accessible en écriture :

   ```
   switch_root:/# mount -o remount,rw /sysroot
   ```

6. Le système de fichiers est remonté et est accessible en écriture.

   Modifiez le **root** du système de fichiers comme suit :

   ```
   switch_root:/# chroot /sysroot
   ```

   L'invite est modifée sur **sh-4.2#**.

7. Saisir la commande **passwd** et suivez les instructions affichées sur la ligne de commande pour modifier le mot de passe **root**.

   Remarquez que si le système n'est pas accessible en écriture, l'outil **passwd** échouera avec l'erreur suivante :

   ```
   Erreur de manipulation du jeton d'authentification
   ```

8. Mise à jour des résultats du fichier du mot de passe avec un contexte de sécurité SELinux incorrect. Pour ré-étiqueter tous les fichiers lors du prochain démarrage, veuillez saisir la commande suivante :

   ```
   sh-4.2# touch /.autorelabel
   ```

   Alternativement, pour économiser le temps pris pour ré-étiqueter un disque de grande taille, vous pouvez omettre cette étape, à condition d'avoir inclus l'option **enforcing=0** dans l'étape 3.

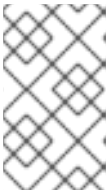
9. Remontez le système de fichier en lecture seule :

   ```
   sh-4.2# mount -o remount,ro /
   ```

10. Saisissez la commande **exit** pour quitter l'environnement **chroot**.

11. Saisissez la commande **exit** à nouveau pour reprendre l'initialisation et terminer le démarrage système.

    Avec un système de fichiers chiffré, un mot de passe ou une phrase de passe est requis(e). Cependant, l'invite du mot de passe peut ne pas apparaître et être obscurcie par les messages de journalisation. Vous pouvez appuyer de façon prolongée sur la touche **Retour Arrière** (« Backspace ») pour afficher l'invite. Relâchez la touche et saisissez le mot de passe du système de fichiers chiffré tout en ignorant les messages de journalisation.



NOTE

Remarquez que le processus de ré-étiquetage SELinux peut prendre longtemps. Un redémarrage système se produira automatiquement lorsque le processus sera terminé.

12. Si vous avez ajouté l'option **`enforcing=0`** dans l'étape 3 et omis la commande **`touch /.autorelabel`** dans l'étape 8, veuillez saisir la commande suivante pour restaurer le contexte de sécurité SELinux du fichier **`/etc/shadow`** :

```
~]# restorecon /etc/shadow
```

Saisissez les commandes suivante pour réactiver l'application de la politique SELinux et vérifier qu'elle soit bien en cours d'exécution :

```
~]# setenforce 1
~]# getenforce
Enforcing
```

24.10. DÉMARRAGE SÉCURISÉ UEFI SECURE BOOT (« UNIFIED EXTENSIBLE FIRMWARE INTERFACE »)

La technologie de démarrage sécurisé « *Unified Extensible Firmware Interface* » (ou UEFI) permet de s'assurer que le microprogramme du système vérifie si le chargeur de démarrage est signé avec une clé de chiffrement autorisée par une base de données de clés publiques contenue dans le microprogramme. Avec la vérification de signatures dans les chargeurs de démarrage et noyaux de future génération, il est possible d'empêcher l'exécution d'un code d'espace de noyau qui n'a pas été signé par une clé de confiance.

Une chaîne de confiance est établie depuis le microprogramme jusqu'aux pilotes signés et aux modules de noyau, comme suit. Le chargeur de démarrage de première étape, **`shim.efi`**, est signé par une clé privée UEFI et authentifié par une clé publique, signé par une autorité de certification (CA), stockée dans la base de données. **`shim.efi`** contient la clé publique de Red Hat, « Red Hat Secure Boot (CA key 1) », qui est utilisée pour authentifier le chargeur de démarrage GRUB 2, **`grubx64.efi`**, et le noyau Red Hat. Le noyau contient des clés publiques pour authentifier les pilotes et modules.

Le démarrage sécurisé (« Secure Boot ») est un composant de validation de chemin de démarrage de la spécification UEFI (« Unified Extensible Firmware Interface »). Cette spécification définit :

- une interface de programmation pour les variables UEFI protégées par chiffrement dans un stockage non volatile,
- la manière dont les certificats root X.509 sont stockés dans des variables UEFI,
- la validation d'applications UEFI comme les chargeurs de démarrage et les pilotes,
- les procédures de révocation des mauvais certificats et hachages d'applications connus.

Le démarrage sécurisé UEFI Secure Boot n'empêche pas l'installation ou la suppression de chargeurs de démarrage de seconde étape, et n'exige pas de confirmation explicite de tels changements de la part de l'utilisateur. Les signatures sont vérifiées pendant le démarrage, pas lorsque le chargeur de démarrage est installé ou mis à jour. Ainsi, le démarrage sécurisé UEFI Secure Boot n'empêche pas les manipulations de chemin de démarrage. Il aide à détecter les changements non autorisés. Un nouveau chargeur de démarrage ou noyau fonctionnera tant qu'il est signé par une clé de confiance du système.

24.10.1. Prise en charge du démarrage sécurisé UEFI Secure Boot sur Red Hat Enterprise Linux 7

Red Hat Enterprise Linux 7 inclut la prise en charge de la fonctionnalité de démarrage sécurisé « UEFI Secure Boot », ce qui signifie que Red Hat Enterprise Linux 7 peut être installé et exécuté sur des systèmes où le démarrage sécurisé « UEFI Secure Boot » est activé. Sur les systèmes basés UEFI avec la technologie Secure Boot activée, tous les pilotes chargés doivent être signés avec une clé de confiance, sinon le système ne les acceptera pas. Tous les pilotes fournis par Red Hat sont signés par l'une des clés privées de Red Hat et authentifiés par la clé publique Red Hat correspondante dans le noyau.

Si vous souhaitez charger des pilotes construits en externe, des pilotes qui ne sont pas fournis sur le DVD de Red Hat Enterprise Linux DVD, vous devez vous assurer que ces pilotes sont également signés.

Des informations sur la signature de pilotes personnalisés sont disponibles dans la Section 26.8, « Signer des modules de noyau pour le démarrage sécurisé « Secure Boot » ».

Restrictions imposées par UEFI Secure Boot

Comme la prise en charge UEFI Secure Boot sur Red Hat Enterprise Linux 7 est conçue pour assurer que le système exécute uniquement un code de mode noyau après l'authentification de sa signature, certaines restrictions existent.

Le chargement de modules GRUB 2 est désactivé car il n'y a pas d'infrastructure pour la signature et vérification des modules GRUB 2, ce qui signifie qu'en leur permettant d'être chargés, cela constituerait une exécution de code non fiable à l'intérieur du périmètre de sécurité défini par Secure Boot. Au lieu de cela, Red Hat fournit un binaire GRUB 2 signé qui contient déjà tous les modules pris en charge sur Red Hat Enterprise Linux 7.

Des informations plus détaillées sont disponibles dans l'article de la base des connaissances Red Hat Restrictions imposées par le démarrage sécurisé « UEFI Secure Boot ».

24.11. RESSOURCES SUPPLÉMENTAIRES

Veuillez consulter les ressources suivantes pour obtenir davantage d'informations sur le chargeur de démarrage GRUB 2 :

Documentation installée

- **`/usr/share/doc/grub2-tools-version-number/`** — Ce répertoire contient des informations sur l'utilisation et la configuration de GRUB 2. *version-number* correspond à la version du paquet GRUB 2 installé.
- **`info grub2`** — La page d'information de GRUB 2 contient des leçons, ainsi qu'un manuel de référence pour les utilisateurs et les programmeurs et une Foire Aux Questions (FAQ) sur GRUB 2 et son utilisation.
- **`grubby(8)`** — la page man de l'outil en ligne de commande qui sert à configurer GRUB et GRUB 2.
- **`new-kernel-pkg(8)`** — la page man de l'outil de script d'intallation du noyau.

Documentation installable et externe

- **`/usr/share/doc/kernel-doc-kernel_version/Documentation/serial-console.txt`** — Ce fichier, fourni par le paquet kernel-doc, contient des informations sur la console série. Avant d'accéder à la documentation du noyau, vous devez exécuter la commande

suivante en tant qu'utilisateur **root** :

```
~]# yum install kernel-doc
```

- Guide d'installation Red Hat — Le guide d'installation fournit des informations de base sur GRUB 2, par exemple, l'installation, la terminologie, les interfaces, et les commandes.

CHAPITRE 25. METTRE À NIVEAU LE NOYAU MANUELLEMENT

Le noyau Red Hat Enterprise Linux est créé de manière personalisée par l'équipe Kernel Team de Red Hat Enterprise Linux afin de s'assurer de son intégrité et de sa compatibilité avec le matériel pris en charge. Avant que Red Hat ne fasse sortir un noyau, celui-ci doit tout d'abord passer un ensemble de tests d'assurance qualité rigoureux.

Les noyaux Red Hat Enterprise Linux sont mis en paquet sous le format RPM, ils sont ainsi facile à mettre à niveau et à vérifier en utilisant le gestionnaire de paquets **Yum** ou **PackageKit**. **PackageKit** effectue des requêtes automatiques sur les serveurs Red Hat Content Delivery Network et vous informe des paquets et mises à jour disponibles, y compris les paquets du noyau.

Ce chapitre est donc *uniquement* utile aux utilisateurs devant mettre à jour manuellement un paquet noyau en utilisant la commande **`rpm`** au lieu de **`yum`**.

AVERTISSEMENT

Lorsque possible, veuillez utiliser le gestionnaire de paquets **Yum** ou **PackageKit** pour installer un nouveau noyau car ils *installent* toujours un nouveau noyau au lieu de remplacer le noyau actuel, ce qui pourrait empêcher votre système de démarrer.

AVERTISSEMENT

La création d'un noyau personnalisé n'est pas prise en charge par l'équipe Red Hat Global Services Support, cela n'est donc pas abordé dans ce manuel.

Pour obtenir des informations supplémentaires sur l'installation de paquets du noyau par **yum**, voir Section 8.1.2, « Mise à jour de paquets ». Pour obtenir des informations sur Red Hat Content Delivery Network, veuillez consulter Chapitre 6, *Enregistrer le système et Gérer les abonnements*.

25.1. VUE D'ENSEMBLE DES PAQUETS DU NOYAU

Red Hat Enterprise Linux contient les paquets de noyau suivants :

- kernel — contient le noyau pour des systèmes unique, multi-cœurs, et multiprocesseurs.
- kernel-debug — contient un noyau avec de nombreuses options de débogage activées pour le diagnostique du noyau, ce qui entraîne une réduction des performances.
- kernel-devel — contient les en-têtes et makefiles suffisants pour créer des modules avec le paquet kernel.

- kernel-debug-devel — contient la version de développement du noyau avec de nombreuses options de débogage activées pour le diagnostique du noyau, ce qui entraîne une réduction des performances.

- kernel-doc — fichiers de documentation de la source du noyau. Diverses portions du noyau Linux et des pilotes de périphériques envoyés avec sont documenté dans ces fichiers. L'installation de ce paquet fournit une référence sur les options pouvant être passées sur les modules du noyau Linux pendant le chargement.

  Par défaut, ces fichiers sont placés dans le répertoire **`/usr/share/doc/kernel-doc-`*`kernel_version`*`/`**.

- kernel-headers — inclut les fichiers d'en-tête C qui spécifient l'interface entre le noyau Linux, les bibliothèques et programmes de l'espace utilisateur. Les fichiers d'en-tête définissent les structures et les constantes qui servent à la création de la plupart des programmes standard.

- linux-firmware — contient tous les fichiers de microprogramme requis par les différents périphériques pour opérer.

- perf — ce paquet contient l'outil **perf** qui permet la surveillance des performances du noyau Linux.

- kernel-abi-whitelists — contient des informations pertinentes à l'ABI du noyau de Red Hat Enterprise Linux, y compris une liste des symboles de noyau nécessaires aux modules de noyau Linux externes et un greffon yum permettant son application.

- kernel-tools — contient des outils pour manipuler le noyau Linux et la documentation de prise en charge.

25.2. PRÉPARER POUR UNE MISE À NIVEAU

Avant de mettre à niveau le noyau, il est recommandé de prendre quelques précautions.

Premièrement, assurez-vous qu'un support de démarrage fonctionnant existe pour le système, au cas où un problème se produirait. Si le chargeur de démarrage n'est pas configuré correctement pour démarrer le nouveau noyau, vous pourrez utiliser ce support pour démarrer Red Hat Enterprise Linux.

Le support USB est souvent présenté sous la forme d'un périphérique flash, parfois appelé *clés USB*, *clés*, or *lecteurs flash*, ou comme un disque dur connecté de manière externe. Presque tous les supports de ce type sont formatés avec un système de fichiers **`VFAT`**. Vous pouvez créer un support de démarrage USB formatté **`ext2`**, **`ext3`**, **`ext4`**, ou **`VFAT`**.

Vous pouvez transférer un fichier image de distribution ou un fichier image de support de démarrage minimal sur un support USB. Assurez-vous que suffisamment d'espace libre est disponible sur le périphérique. Vous aurez besoin d'environ **`4 Go`** pour une image de DVD de distribution, environ **`700 Mo`** pour une image de CD de distribution, et **`10 Mo`** pour une image de support de démarrage minimal.

Vous devez posséder une copie du fichier **`boot.iso`** provenant d'un DVD d'installation Red Hat Enterprise Linux, ou du CD-ROM #1 d'installation, il vous faudra également un périphérique de stockage USB formaté avec le système de fichiers **`VFAT`**, ainsi que **`16 Mo`** d'espace libre. La procédure suivante n'affectera pas les fichiers existants sur le périphérique de stockage USB à moins qu'ils aient les mêmes noms de chemin que les fichiers copiés dans celui-ci. Pour créer le support de démarrage USB, veuillez exécuter les commandes suivantes en tant qu'utilisateur **`root`** :

1. Veuillez installer le paquet syslinux s'il n'est pas déjà installé sur votre système. Pour cela, veuillez exécuter la commande **yum install syslinux** en tant qu'utilisateur root.

2. Veuillez installer le chargeur de démarrage **SYSLINUX** sur le périphérique de stockage USB :

```
~]# syslinux /dev/sdX1
```

   ... avec *sdX* comme nom de périphérique.

3. Veuillez créer des points de montage pour **boot.iso** et le périphérique de stockage USB :

```
~]# mkdir /mnt/isoboot /mnt/diskboot
```

4. Montez **boot.iso** :

```
~]# mount -o loop boot.iso /mnt/isoboot
```

5. Montez le périphérique de stockage USB :

```
~]# mount /dev/sdX1 /mnt/diskboot
```

6. Copiez les fichiers **ISOLINUX** depuis **boot.iso** sur le périphérique de stockage USB :

```
~]# cp /mnt/isoboot/isolinux/* /mnt/diskboot
```

7. Veuillez utiliser le fichier **isolinux.cfg** de **boot.iso** comme fichier **syslinux.cfg** pour le périphérique USB :

```
~]# grep -v local /mnt/isoboot/isolinux/isolinux.cfg >
/mnt/diskboot/syslinux.cfg
```

8. Démontez **boot.iso** et le périphérique de stockage USB :

```
~]# umount /mnt/isoboot /mnt/diskboot
```

9. Vous devez redémarrer la machine avec le support de démarrage et vérifier que vous êtes en mesure de démarrer avec avant de continuer.

De manière alternative, sur les systèmes avec un lecteur de disquette, il est possible de créer une disquette de démarrage en installant le paquet mkbootdisk et en exécutant la commande **mkbootdisk** en tant qu'utilisateur **root**. Veuillez consulter la page man **man mkbootdisk** après avoir installé le paquet pour les informations d'utilisation.

Pour déterminer quels paquets sont installés, veuillez exécuter la commande **yum list installed "kernel-*"** à l'invite du shell. La sortie comprendra certains ou tous les paquets suivants, selon l'architecture du système, et les numéros de version peuvent différer :

```
~]# yum list installed "kernel-*"
kernel.x86_64                  3.10.0-54.0.1.el7          @rhel7/7.0
kernel-devel.x86_64            3.10.0-54.0.1.el7          @rhel7
kernel-headers.x86_64          3.10.0-54.0.1.el7          @rhel7/7.0
```

Veuillez déterminer, à partir de la sortie, les paquets qui doivent être téléchargés pour la mise à niveau du noyau. Pour un système à processeur unique, le seul paquet requis est le paquet kernel. Veuillez consulter Section 25.1, « Vue d'ensemble des paquets du noyau » pour afficher les descriptions des différents paquets.

25.3. TÉLÉCHARGER LE NOYAU MIS À NIVEAU

Il existe plusieurs manières de déterminer si un noyau mis à jour est disponible pour le système.

- Errata de sécurité — veuillez consulter https://access.redhat.com/site/security/updates/active/ pour obtenir des informations sur les errata de sécurité, y compris les mises à niveau de noyau qui corrigent les problèmes de sécurité.

- The Red Hat Content Delivery Network — pour un système abonné au Red Hat Content Delivery Network, le gestionnaire de paquets **yum** peut télécharger le dernier noyau et mettre à jour le noyau sur le système. L'utilitaire **Dracut** va créer une image de disque RAM initial si nécessaire, et configurer le chargeur de démarrage pour démarrer le nouveau noyau. Pour obtenir davantage d'informations sur la façon d'installer des packages sur le Red Hat Content Delivery Network, consulter Chapitre 8, *Yum*. Pour obtenir plus d'informations sur la façon d'abonner un système au Red Hat Content Delivery Network, voir Chapitre 6, *Enregistrer le système et Gérer les abonnements*.

Si **yum** a été utilisé pour télécharger et installer le noyau mis à jour à partir du Red Hat Network, veuillez suivre les instructions se trouvant dans Section 25.5, « Vérifier l'image de disque RAM initial » et Section 25.6, « Vérifier le chargeur de démarrage » uniquement, mais *ne modifiez pas* le noyau pour qu'il démarre par défaut. Red Hat Network change automatiquement le noyau par défaut à la dernière version. Pour installer le noyau manuellement, veuillez passer à la Section 25.4, « Effectuer la mise à niveau ».

25.4. EFFECTUER LA MISE À NIVEAU

Après avoir récupéré tous les paquets nécessaires, il est temps de mettre à niveau le noyau existant.

IMPORTANT

Il est fortement recommandé de conserver l'ancien noyau au cas où il y aurait des problèmes avec le nouveau noyau.

À l'invite du shell, veuillez modifier le répertoire qui contient les paquets RPM du noyau. Veuillez utiliser l'argument **-i** avec la commande **rpm** pour garder l'ancien noyau. N'utilisez *pas* l'option **-U**, car elle remplacera le noyau actuellement installé, ce qui créera des problèmes avec le chargeur de démarrage. Par exemple :

```
~]# rpm -ivh kernel-kernel_version.arch.rpm
```

La prochaine étape consiste à vérifier que l'image de disque RAM initial a bien été créée . Veuillez consulter la Section 25.5, « Vérifier l'image de disque RAM initial » pour obtenir des détails.

25.5. VÉRIFIER L'IMAGE DE DISQUE RAM INITIAL

Le but de l'image de disque RAM initial consiste à précharger les modules de périphériques blocs, comme pour IDE, SCSI ou RAID, afin que le système de fichiers racine, sur lequel ces modules résident normalement, puissent ensuite être accédés et montés. Sur les systèmes Red Hat Enterprise Linux 7,

lorsqu'un nouveau noyau est installé en utilisant le gestionnaire de paquets **Yum**, **PackageKit**, ou **RPM**, l'utilitaire **Dracut** est toujours appelé par les scripts d'installation pour créer une image de disque RAM initial *initramfs*.

Sur toutes les architectures autres que IBM eServer System i (veuillez consulter la section intitulée « Vérifier l'image de disque RAM initial et le noyau sur IBM eServer System i »), il est possible de créer une image **initramfs** en exécutant la commande **dracut**. Cependant, il n'est habituellement pas nécessaire de créer une image **initramfs** manuellement : cette étape est automatiquement effectuée si le noyau et ses paquets associés sont installés ou mis à niveau à partir des paquets RPM distribués par Red Hat.

Vous pouvez vérifier qu'une image **initramfs** correspondant à votre version du noyau actuel existe et qu'elle est correctement spécifiée dans le fichier de configuration **grub.cfg** en suivant la procédure ci-dessous :

Procédure 25.1. Vérifier l'image de disque RAM initial

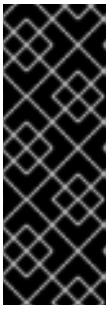
1. En tant qu'utilisateur **root**, répertoriez le contenu du répertoire **/boot** et trouvez le noyau (**vmlinuz-*kernel_version***) et **initramfs-*kernel_version*** avec le numéro de version le plus récent :

   Exemple 25.1. Assurez-vous que les versions du noyau et d'initramfs correspondent bien

   ```
   ~]# ls /boot
   config-3.10.0-67.el7.x86_64
   config-3.10.0-78.el7.x86_64
   efi
   grub
   grub2
   initramfs-0-rescue-07f43f20a54c4ce8ada8b70d33fd001c.img
   initramfs-3.10.0-67.el7.x86_64.img
   initramfs-3.10.0-67.el7.x86_64kdump.img
   initramfs-3.10.0-78.el7.x86_64.img
   initramfs-3.10.0-78.el7.x86_64kdump.img
   initrd-plymouth.img
   symvers-3.10.0-67.el7.x86_64.gz
   symvers-3.10.0-78.el7.x86_64.gz
   System.map-3.10.0-67.el7.x86_64
   System.map-3.10.0-78.el7.x86_64
   vmlinuz-0-rescue-07f43f20a54c4ce8ada8b70d33fd001c
   vmlinuz-3.10.0-67.el7.x86_64
   vmlinuz-3.10.0-78.el7.x86_64
   ```

   L'Exemple 25.1, « Assurez-vous que les versions du noyau et d'initramfs correspondent bien » montre que :

   - trois noyaux sont installés (ou plutôt, trois fichiers noyau sont présents dans le répertoire **/boot/**),
   - le dernier noyau est nommé **vmlinuz-3.10.0-78.el7.x86_64**, et
   - un fichier **initramfs** correspondant à la version du noyau **initramfs-3.10.0-78.el7.x86_64kdump.img** existe également.



IMPORTANT

Dans le répertoire **/boot**, vous trouverez plusieurs fichiers **initramfs-*kernel_version*kdump.img**. Ces fichiers sont des fichiers spéciaux créés par le mécanisme **Kdump** à des fins de débogage de noyau, ils ne sont pas utilisés pour démarrer le système, et peuvent être ignorés en toute sécurité. Pour obtenir davantage d'informations sur **kdump**, veuillez consulter le Guide de vidage sur incident de noyau Red Hat Enterprise Linux 7.

2. Si le fichier **initramfs-*kernel_version*** ne correspond pas à la version du noyau le plus récent du fichier **/boot**, ou dans d'autres situations, si vous deviez générer un fichier **initramfs** avec l'utilitaire **Dracut**, veuillez simplement invoquer **dracut** en tant qu'utilisateur **root** sans lui faire générer de fichier **initramfs** dans le répertoire **/boot/** pour obtenir le noyau le plus récent présent dans ce répertoire :

```
~]# dracut
```

Vous devez utiliser l'option **-f**, **--force** si vous souhaitez que **dracut** remplace le fichier **initramfs** existant (par exemple, si **initramfs** a été corrompu). Sinon, **dracut** refusera de remplacer le fichier **initramfs** existant :

```
~]# dracut
Ne remplacera pas le fichier initramfs existant (/boot/initramfs-
3.10.0-78.el7.x86_64.img) sans --force
```

You can create an initramfs in the current directory by calling **dracut *initramfs_name kernel_version***:

```
~]# dracut "initramfs-$(uname -r).img" $(uname -r)
```

If you need to specify specific kernel modules to be preloaded, add the names of those modules (minus any file name suffixes such as **.ko**) inside the parentheses of the **add_dracutmodules+="*module* [*more_modules*]"** directive of the **/etc/dracut.conf** configuration file. You can list the file contents of an **initramfs** image file created by dracut by using the **lsinitrd *initramfs_file*** command:

```
~]# lsinitrd /boot/initramfs-3.10.0-78.el7.x86_64.img
Image: /boot/initramfs-3.10.0-78.el7.x86_64.img: 11M
================================================================
====
dracut-033-68.el7
================================================================
====

drwxr-xr-x  12 root     root            0 Feb  5 06:35 .
drwxr-xr-x   2 root     root            0 Feb  5 06:35 proc
lrwxrwxrwx   1 root     root           24 Feb  5 06:35 init ->
/usr/lib/systemd/systemd
drwxr-xr-x  10 root     root            0 Feb  5 06:35 etc
drwxr-xr-x   2 root     root            0 Feb  5 06:35
usr/lib/modprobe.d[sortie tronquée]
```

Veuillez consulter **man dracut** et **man dracut.conf** pour obtenir davantage d'informations sur les options et l'utilisation.

3. Examinez le fichier de configuration **/boot/grub2/grub.cfg** pour vous assurer qu'un fichier **initramfs-*kernel_version*.img** existe bien pour la version du noyau que vous démarrez. Par exemple :

```
~]# grep initramfs /boot/grub2/grub.cfg
 initrd16 /initramfs-3.10.0-123.el7.x86_64.img
 initrd16 /initramfs-0-rescue-6d547dbfd01c46f6a4c1baa8c4743f57.img
```

Veuillez consulter Section 25.6, « Vérifier le chargeur de démarrage » pour obtenir davantage d'informations.

Vérifier l'image de disque RAM initial et le noyau sur IBM eServer System i

Sur les machines IBM eServer System i, les fichiers de l'image de disque RAM initial et du noyau sont combinés en un fichier unique, qui est créé par la commande **addRamDisk**. Cela est effectué automatiquement si le noyau et ses paquets associés sont installés ou mis à niveau à partir des paquets RPM distribués par Red Hat ; ainsi, il n'est pas nécessaire de l'exécuter manuellement. Pour vérifier qu'il a bien été créé, veuillez exécuter la commande suivante en tant qu'utilisateur **root** pour vous assurer que le fichier **/boot/vmlinitrd-*kernel_version*** existe au préalable :

```
ls -l /boot/
```

kernel_version doit correspondre à la version du noyau qui vient d'être installée.

Annuler les changements faits à l'image de disque RAM initial

Dans certains cas, comme par exemple, quand vous faîtes une erreur de configuration du système, et qu'il ne démarre plus, vous devrez sans doute annuler les changements faits à l'image de disque RAM initiale, en suivant la procédure suivante :

Procédure 25.2. Annuler des changements faits à l'image de disque RAM initial

1. Redémarrez le système en sélectionnant le noyau de secours dans le menu GRUB.
2. Changez le paramètre de configuration qui a amené **initramfs** à mal-fonctionner.
3. Recréer **initramfs** avec les paramètres qui conviennent en exécutant la commande suivante en tant qu'utilisateur root :

```
~]# dracut --kver kernel_version --force
```

La procédure ci-dessus peut vous être utile si, par exemple, vous définissez les **vm.nr_hugepages** dans le fichier **sysctl.conf**. Comme le fichier **sysctl.conf** est inclus dans **initramfs**, la nouvelle configuration de **vm.nr_hugepages** sera appliquée à **initramfs** et **initramfs** sera reconstruit. Cependant, comme la configuration est erronée, le nouvel **initramfs** est endommagé et le nouveau noyau ne démarre pas, ce qui nécessite une correction par la procédure ci-dessus.

Répertorier le contenu de l'image de disque RAM initial

Pour répertorier les fichiers inclus dans **initramfs**, exécutez la commande suivante en tant qu'utilisateur root :

```
~]# lsinitrd
```

Pour ne répertorier que les fichiers qui se trouvent dans le répertoire **/etc**, utilisez la commande suivante :

```
~]# lsinitrd | grep etc/
```

Pour obtenir la sortie de contenu d'un fichier spécifique qui se situe dans **initramfs** pour le noyau actuel, utiliser l'option **-f** :

```
~]# lsinitrd -f filename
```

Par exemple, pour obtenir une sortie de contenu de **sysctl.conf**, utilisez la commande suivante :

```
~]# lsinitrd -f /etc/sysctl.conf
```

Pour spécifier une version de noyau quelconque, utilisez l'option **--kver** :

```
~]# lsinitrd --kver kernel_version -f /etc/sysctl.conf
```

Par exemple, pour répertorier les informations sur une version de noyau 3.10.0-327.10.1.el7.x86_64, utilisez la commande suivante :

```
~]# lsinitrd --kver 3.10.0-327.10.1.el7.x86_64 -f /etc/sysctl.conf
```

25.6. VÉRIFIER LE CHARGEUR DE DÉMARRAGE

Lorsque vous installez un noyau à l'aide de **rpm**, le paquet du noyau crée une entrée dans le fichier de configuration du chargeur de démarrage de ce nouveau noyau. Cependant, **rpm** ne configure *pas* le nouveau noyau pour démarrer en tant que noyau par défaut. Cela doit être effectué manuellement pendant l'installation d'un nouveau noyau avec **rpm**.

Il est toujours recommandé de vérifier une seconde fois le fichier de configuration du chargeur de démarrage après avoir installé un nouveau noyau avec **rpm** afin de vous assurer que la configuration soit correcte. Autrement, le système pourrait ne pas être en mesure de démarrer dans Red Hat Enterprise Linux correctement. Si cela se produit, veuillez démarrer le système avec le support de démarrage créé précédemment et configurez à nouveau le chargeur de démarrage.

CHAPITRE 26. UTILISER DES MODULES DE NOYAU

Le noyau Linux est modulaire, ce qui signifie qu'il peut étendre ses capacités par l'utilisation de *modules de noyau* chargés dynamiquement. Un module de noyau peut fournir :

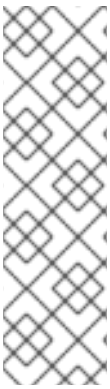
- un pilote de périphérique qui ajoute la prise en charge de nouveau matériel ; ou,
- la prise en charge d'un système de fichiers tel que **btrfs** ou **NFS**.

Comme le noyau lui-même, les modules peuvent prendre des paramètres permettant de personnaliser leur comportement, même si les paramètres par défaut fonctionnent bien dans la plupart des cas. Les outils de l'espace utilisateur peuvent répertorier les modules actuellement chargés dans un noyau d'exécution ; interroger tous les modules disponibles pour les paramètres disponibles et des informations spécifiques au module ; et charger ou décharger des modules (remove) dynamiquement vers ou depuis un noyau en cours d'exécution. Bon nombre de ces utilitaires, qui sont fournis dans le paquet kmod, prennent les dépendances de module en compte lorsque vous effectuez des opérations, ce qui fait que les suivis de dépendance sont rarement nécessaires.

Sur les systèmes modernes, les modules de noyau sont automatiquement chargés par divers mécanismes lorsque les conditions l'exigent. Cependant, il peut y avoir des occasions où il est nécessaire de charger ou décharger des modules manuellement, comme lorsqu'un module est préférable à un autre même si les deux fournissent la même fonctionnalité de base, ou lorsqu'un module se comporte anormalement.

Ce chapitre explique comment :

- utiliser les utilitaires **kmod** d'espace utilisateur pour afficher, effectuer des requêtes, charger et décharger des modules de noyau et leurs dépendances ;
- définir des paramètres de modules sur la ligne de commande de manière dynamique et permanente afin de pouvoir personnaliser le comportement de vos modules de noyau ; et,
- charger les modules pendant le démarrage.



NOTE

Pour utiliser les utilitaires de modules de noyau décrits dans ce chapitre, commencez par vous assurer que le paquet kmod est installé sur votre système en exécutant la commande suivante en tant qu'utilisateur root :

```
~]# yum install kmod
```

Pour obtenir davantage d'informations sur l'installation de paquets avec Yum, veuillez consulter la Section 8.2.4, « Installation de paquets ».

26.1. RÉPERTORIER LES MODULES ACTUELLEMENT CHARGÉS

Vous pouvez répertorier tous les modules de noyau actuellement chargés dans le noyau en exécutant la commande **lsmod**, exemple :

```
~]$ lsmod
Module                  Size  Used by
tcp_lp                 12663  0
bnep                   19704  2
bluetooth             372662  7 bnep
```

```
rfkill                 26536  3 bluetooth
fuse                   87661  3
ip6t_rpfilter          12546  1
ip6t_REJECT            12939  2
ipt_REJECT             12541  2
xt_conntrack           12760  7
ebtable_nat            12807  0
ebtable_broute         12731  0
bridge                110196  1 ebtable_broute
stp                    12976  1 bridge
llc                    14552  2 stp,bridge
ebtable_filter         12827  0
ebtables               30913  3 ebtable_broute,ebtable_nat,ebtable_filter
ip6table_nat           13015  1
nf_conntrack_ipv6      18738  5
nf_defrag_ipv6         34651  1 nf_conntrack_ipv6
nf_nat_ipv6            13279  1 ip6table_nat
ip6table_mangle        12700  1
ip6table_security      12710  1
ip6table_raw           12683  1
ip6table_filter        12815  1
ip6_tables             27025  5
ip6table_filter,ip6table_mangle,ip6table_security,ip6table_nat,ip6table_ra
w
iptable_nat            13011  1
nf_conntrack_ipv4      14862  4
nf_defrag_ipv4         12729  1 nf_conntrack_ipv4
nf_nat_ipv4            13263  1 iptable_nat
nf_nat                 21798  4
nf_nat_ipv4,nf_nat_ipv6,ip6table_nat,iptable_nat[sortie tronquée]
```

Chaque ligne de la sortie **lsmod** spécifie :

- le nom d'un module de noyau actuellement chargé en mémoire ;
- la quantité de mémoire utilisée ;
- la somme totale des processus utilisant le module et des autres modules qui en dépendent, suivie d'une liste des noms de ces modules, s'ils existent. À l'aide de cette liste, vous pouvez décharger tous les modules selon le module que vous souhaitez décharger. Pour obtenir davantage d'informations, veuillez consulter la Section 26.4, « Décharger un module ».

Finalement, veuillez remarquer que la sortie **lsmod** est moins verbeuse et considérablement plus facile à lire que le contenu du pseudo-fichier **/proc/modules**.

26.2. AFFICHER DES INFORMATIONS SUR UN MODULE

Vous pouvez afficher des informations détaillées sur un module de noyau en utilisant la commande **modinfo *module_name***.

NOTE

Lorsqu'un nom de module de noyau est saisi en tant qu'argument de l'un des utilitaires **kmod**, veuillez ne pas ajouter d'extension **.ko** à la fin du nom. Les noms de module de noyau n'ont pas d'extensions. mais leurs fichiers correspondants en ont.

Exemple 26.1. Répertorier des informations sur un module de noyau avec lsmod

Pour afficher des informations sur le module **e1000e**, qui est le pilote réseau Intel PRO/1000, veuillez saisir la commande suivante en tant qu'utilisateur **root** :

```
~]# modinfo e1000e
filename:       /lib/modules/3.10.0-
121.el7.x86_64/kernel/drivers/net/ethernet/intel/e1000e/e1000e.ko
version:        2.3.2-k
license:        GPL
description:    Intel(R) PRO/1000 Network Driver
author:         Intel Corporation, <linux.nics@intel.com>
srcversion:     E9F7E754F6F3A1AD906634C
alias:          pci:v00008086d000015A3sv*sd*bc*sc*i*
alias:          pci:v00008086d000015A2sv*sd*bc*sc*i*[certaines lignes
alias ont été omises]
alias:          pci:v00008086d0000105Esv*sd*bc*sc*i*
depends:        ptp
intree:         Y
vermagic:       3.10.0-121.el7.x86_64 SMP mod_unload modversions
signer:         Red Hat Enterprise Linux kernel signing key
sig_key:
42:49:68:9E:EF:C7:7E:95:88:0B:13:DF:E4:67:EB:1B:7A:91:D1:08
sig_hashalgo:   sha256
parm:           debug:Debug level (0=none,...,16=all) (int)
parm:           copybreak:Maximum size of packet that is copied to a
new buffer on receive (uint)
parm:           TxIntDelay:Transmit Interrupt Delay (array of int)
parm:           TxAbsIntDelay:Transmit Absolute Interrupt Delay (array
of int)
parm:           RxIntDelay:Receive Interrupt Delay (array of int)
parm:           RxAbsIntDelay:Receive Absolute Interrupt Delay (array
of int)
parm:           InterruptThrottleRate:Interrupt Throttling Rate (array
of int)
parm:           IntMode:Interrupt Mode (array of int)
parm:           SmartPowerDownEnable:Enable PHY smart power down (array
of int)
parm:           KumeranLockLoss:Enable Kumeran lock loss workaround
(array of int)
parm:           WriteProtectNVM:Write-protect NVM [WARNING: disabling
this can lead to corrupted NVM] (array of int)
parm:           CrcStripping:Enable CRC Stripping, disable if your BMC
needs the CRC (array of int)
```

Ci-dessous figurent des descriptions de quelques champs dans la sortie de **modinfo** :

filename

Chemin absolu vers le fichier objet du noyau **.ko**. Vous pouvez utiliser **modinfo -n** en tant que raccourci de commande pour imprimer le nom de fichier **filename** uniquement.

description

Courte description du module. Vous pouvez utiliser **modinfo -d** comme raccourci de commande pour imprimer le champ de la description uniquement.

alias

Le champ **alias** apparaît autant de fois qu'il existe d'alias pour un module, ou il est entièrment omis s'il n'y en a pas.

depends

Ce champ contient une liste de tous les modules séparés par des virgules dont ce module dépend.

NOTE

Si un module ne possède aucune dépendance, le champ **depends** peut être omis de la sortie.

parm

Chaque champ **parm** présente un paramètre de module sous le format ***parameter_name*:*description***, où :

- *parameter_name* est la syntaxe exacte à utiliser lors d'une utilisation comme paramètre de module sur la ligne de commande, ou dans une ligne d'option dans un fichier **.conf** dans le répertoire **/etc/modprobe.d/** ;
- *description* est une brève explication de ce que le paramètre fait, ainsi qu'une attente quant au type de valeur acceptée entre parenthèses (comme int, unit ou array of int).

Exemple 26.2. Répertorier les paramètres de modules

Vous pouvez répertorier tous les paramètres pris en charge par le module en utilisant l'option **-p**. Cependant, comme des informations utiles concernant le type de valeur sont omises de la sortie **modinfo -p**, il est plus utile d'exécuter :

```
~]# modinfo e1000e | grep "^parm" | sort
parm:           copybreak:Maximum size of packet that is copied to a
new buffer on receive (uint)
parm:           CrcStripping:Enable CRC Stripping, disable if your
BMC needs the CRC (array of int)
parm:           debug:Debug level (0=none,...,16=all) (int)
parm:           InterruptThrottleRate:Interrupt Throttling Rate
(array of int)
parm:           IntMode:Interrupt Mode (array of int)
parm:           KumeranLockLoss:Enable Kumeran lock loss workaround
(array of int)
parm:           RxAbsIntDelay:Receive Absolute Interrupt Delay (array
of int)
parm:           RxIntDelay:Receive Interrupt Delay (array of int)
parm:           SmartPowerDownEnable:Enable PHY smart power down
(array of int)
parm:           TxAbsIntDelay:Transmit Absolute Interrupt Delay
(array of int)
```

```
parm:           TxIntDelay:Transmit Interrupt Delay (array of int)
parm:           WriteProtectNVM:Write-protect NVM [WARNING: disabling
this can lead to corrupted NVM] (array of int)
```

26.3. CHARGER UN MODULE

Pour charger un module de noyau, veuillez exécuter **modprobe *module_name*** en tant qu'utilisateur **root**. Par exemple, pour charger le module **wacom**, veuillez exécuter :

```
~]# modprobe wacom
```

Par défaut, **modprobe** tente de charger le module de **/lib/modules/*kernel_version*/kernel/drivers/**. Dans ce répertoire, chaque type de module possède son propre sous-répertoire, comme **net/** et **scsi/**, pour les pilotes des interfaces réseau et SCSI, respectivement.

Certains modules ont des dépendances qui sont d'autres modules de noyau devant être chargés avant que le module en question puisse être chargé. La commande **modprobe** prend toujours les dépendances en compte lorsqu'elle effectue des opérations. Lorsque vous demandez à **modprobe** de charger un module de noyau spécifique, il examine avant tout les dépendances de ce module, s'il y en a, et les charge si elles ne sont pas déjà chargées dans le noyau. **modprobe** résoud les dépendances de manière récursive : les dépendances des dépendances seront chargées, et ainsi de suite si nécessaire, assurant ainsi que toutes les dépendances soient bien résolues.

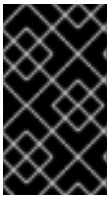
Vous pouvez utiliser l'option **-v** (ou **--verbose**) pour faire en sorte que **modprobe** affiche des informations détaillées sur ce qui est fait, ce qui peut inclure le chargement de dépendances de modules.

Exemple 26.3. modprobe -v affiche les dépendances de module au fur et à mesure de leur chargement

Vous pouvez charger le module **Fibre Channel over Ethernet** de manière détaillée en saisissant ce qui suit dans l'invite shell :

```
~]# modprobe -v fcoe
insmod /lib/modules/3.10.0-
121.el7.x86_64/kernel/drivers/scsi/scsi_tgt.ko
insmod /lib/modules/3.10.0-
121.el7.x86_64/kernel/drivers/scsi/scsi_transport_fc.ko
insmod /lib/modules/3.10.0-
121.el7.x86_64/kernel/drivers/scsi/libfc/libfc.ko
insmod /lib/modules/3.10.0-
121.el7.x86_64/kernel/drivers/scsi/fcoe/libfcoe.ko
insmod /lib/modules/3.10.0-
121.el7.x86_64/kernel/drivers/scsi/fcoe/fcoe.ko
```

Dans cet exemple, vous pouvez voir que **modprobe** a chargé les modules **scsi_tgt**, **scsi_transport_fc**, **libfc** et **libfcoe** en tant que dépendances avant de finalement charger **fcoe**. Remarquez également que **modprobe** a utilisé une commande plus primitive, **insmod**, pour insérer les modules dans le noyau en cours d'exécution.



IMPORTANT

Même si la commande **insmod** peut également être utilisée pour charger des modules de noyau, elle ne résoud pas de dépendances. Par conséquent, vous devriez *toujours* charger les modules à l'aide de **modprobe**.

26.4. DÉCHARGER UN MODULE

Vous pouvez décharger un module de noyau en exécutant **modprobe -r *module_name*** en tant qu'utilisateur **root**. Par exemple, si le module **wacom** est déjà chargé dans le noyau, vous pouvez le décharger en exécutant :

```
~]# modprobe -r wacom
```

Cependant, cette commande échouera si un processus utilise :

- le module **wacom** ;
- un module dont **wacom** dépend directement ; ou
- tout module dont **wacom** dépend indirectement à travers l'arborescence des dépendances.

Veuillez consulter la Section 26.1, « Répertorier les modules actuellement chargés » pour obtenir davantage d'informations sur l'utilisation de **lsmod** pour obtenir les noms des modules qui vous empêchent de décharger un certain module.

Exemple 26.4. Décharger un module de noyau

Par exemple, si vous souhaitez décharger le module **firewire_ohci**, votre session de terminal pourrait ressembler à ceci :

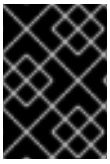
```
~]# modinfo -F depends firewire_ohci
firewire-core
~]# modinfo -F depends firewire_core
crc-itu-t
~]# modinfo -F depends crc-itu-t
```

Vous avez compris le fonctionnement de l'arborescence des dépendances (aucune branche ne se trouve dans cet exemple) pour les modules Firewire chargés : **firewire_ohci** dépend de **firewire_core**, qui dépend de **crc-itu-t**.

Vous pouvez décharger **firewire_ohci** en utilisant la commande **modprobe -v -r *module_name*** , où **-r** est un raccourci pour **--remove** et **-v** pour **--verbose** :

```
~]# modprobe -r -v firewire_ohci
rmmod firewire_ohci
rmmod firewire_core
rmmod crc_itu_t
```

La sortie montre que les modules sont déchargés dans l'ordre inverse de celui dans lequel ils ont été chargés, à condition qu'aucun processus ne dépende de l'un des modules en cours de déchargement.



IMPORTANT

Même si la commande **rmmod** peut être utilisée pour décharger des modules de noyau, il est recommandé d'utiliser **modprobe -r** à la place.

26.5. DÉFINIR LES PARAMÈTRES DE MODULE

Tout comme le noyau, les modules peuvent également prendre des paramètres qui changeront leur comportement. La plupart du temps, les paramètres par défaut fonctionnent correctement. Mais occasionnellement, il peut être nécessaire ou souhaitable de définir des paramètres personnalisés pour un module. Comme les paramètres ne peuvent pas être définis de manière dynamique pour un module déjà chargé dans un noyau en cours d'exécution, deux méthodes différentes de les définir s'offrent à vous.

1. Vous pouvez décharger toutes les dépendances du module dont vous souhaitez définir les paramètres, déchargez le module en utilisant **modprobe -r**, puis chargez-le avec **modprobe**, ainsi qu'avec une liste de paramètres personnalisés. Cette méthode, couverte dans cette section, est souvent utilisée lorsque le module ne possède que peu de dépendances, ou pour tester différentes combinaisons de paramètres sans les rendre persistants.

2. Alternativement, vous pouvez répertorier les nouveaux paramètres dans un fichier nouveau ou existant dans le répertoire **/etc/modprobe.d/**. Cette méthode rend les paramètres du module persistants en assurant qu'ils soient définis chaque fois que le module est chargé, comme après chaque redémarrage, ou après la commande **modprobe**. Même si les informations suivantes sont des conditions préalables, cette méthode est couverte dans la Section 26.6, « Chargement de modules persistants ».

Exemple 26.5. Fournir des paramètres optionnels lors du chargement d'un module de noyau

Vous pouvez utiliser **modprobe** pour charger un module de noyau avec des paramètres personnalisés en utilisant le format de ligne de commande suivant :

```
~]# modprobe module_name [parameter=value]
```

Lors du chargement d'un module avec des paramètres personnalisés sur la ligne de commande, n'oubliez pas que :

- Vous pouvez saisir plusieurs paramètres et valeurs en les séparant par des espaces.
- Certains paramètres de module s'attendent à une liste de valeurs séparées par des virgules comme arguments. Lorsque la liste des valeurs est saisie, n'insérez *pas* d'espace après les virgules, ou **modprobe** n'interprétera pas correctement les valeurs qui suivent les espaces en tant que paramètres supplémentaires.
- La commande **modprobe** fonctionne silencieusement avec un statut de sortie de **0** si :
  - le module est bien chargé, *ou*
  - le module est *déjà* chargé dans le noyau.

  Ainsi, vous devez vous assurer que le module n'est pas déjà chargé avant de tenter de le charger avec des paramètres personnalisés. La commande **modprobe** ne recharge pas automatiquement le module, ou ne vous alerte pas automatiquement qu'il est déjà chargé.

Voici les étapes recommandées pour définir des paramètres personnalisés, puis charger un module de noyau. Cette procédure illustre les étapes utilisant le module **e1000e**, qui est le pilote réseau des adaptateurs réseau Intel PRO/1000 :

Procédure 26.1. Charger un module de noyau avec des paramètres personnalisés

1. Veuillez commencer par vous assurer que le module n'est pas chargé dans le noyau :

```
~]# lsmod |grep e1000e
~]#
```

   La sortie indiquera que le module est déjà chargé dans le noyau, dans lequel cas vous devrez tout d'abord le décharger avant de continuer. Veuillez consulter la Section 26.4, « Décharger un module » pour obtenir des instructions sur la manière de le décharger en toute sécurité.

2. Chargez le module et répertoriez tous les paramètres personnalisés après le nom du module. Par exemple, si vous souhaitez charger le pilote réseau Intel PRO/1000 avec un taux d'accélération d'interruptions défini sur 3000 interruptions par seconde pour la première, seconde et troisième instance du pilote, et activer le débogage, vous devrez exécuter en tant qu'utilisateur **root** :

```
~]# modprobe e1000e InterruptThrottleRate=3000,3000,3000 debug=1
```

   Cet exemple illustre le transfert de plusieurs valeurs sur un paramètre unique en les séparant avec des virgules et en omettant tout espace les séparant.

26.6. CHARGEMENT DE MODULES PERSISTANTS

Comme affiché dans l'Exemple 26.1, « Répertorier des informations sur un module de noyau avec lsmod », de nombreux modules de noyau sont chargés automatiquement pendant le démarrage. Vous pouvez spécifier des modules supplémentaires devant être chargés par le démon **systemd-modules-load.service** en créant le fichier ***program*.conf** dans le répertoire **/etc/modules-load.d/**, où *program* est un nom descriptif de votre choix. Les fichiers dans **/etc/modules-load.d/** sont des fichiers texte qui répertorient les modules à charger, un par ligne.

Exemple 26.6. Un fichier texte pour charger un module

Pour créer un fichier pour charger le module **virtio-net.ko**, créez un fichier **/etc/modules-load.d/virtio-net.conf** avec le contenu suivant :

```
# Load virtio-net.ko at boot
virtio-net
```

Veuillez consulter les pages man de **modules-load.d(5)** et **systemd-modules-load.service(8)** pour obtenir davantage d'informations.

26.7. INSTALLATION DE MODULES À PARTIR D'UN DISQUE DE MISE À JOUR DE PILOTE

Les modules de pilotes de matériel sont parfois fournis sous la forme d'un *disque de mise à jour de pilote* (DUD). Le disque de mise à jour du pilote, ou bien une image ISO, sont normalement utilisés au moment

de l'installation pour charger et installer les modules dont le matériel utilisé a besoin, et ce processus est décrit dans le guide Red Hat Enterprise Linux 7 Installation Guide. Cependant, si on a besoin de nouveaux modules après l'installation, utiliser la procédure suivante. Si vous avez déjà des fichiers RPM, procédez directement à l'étape **5**.

Procédure 26.2. Installation de nouveaux modules à partir d'un disque de mise à jour de pilote

Suivre cette procédure de post installation pour installer des nouveaux modules de pilotes à partir d'un disque de mise à jour de pilote (DUD).

1. Installer le disque de mise à jour de pilote.

2. Créer un point de montage et monter le DUD. Ainsi, en tant qu'utilisateur **root** :

```
~]# mkdir /run/OEMDRV
~]# mount -r -t iso9660 /dev/sr0 /run/OEMDRV
```

3. Afficher les contenus de DUD. Exemple :

```
~]# ls /run/OEMDRV/
rhdd3  rpms  src
```

4. Rendez vous dans le répertoire qui correspond à l'architecture de votre système, qui se trouve dans **rpms/**, et listez-en le contenu. Exemple :

```
~]# cd /run/OEMDRV/rpms/x86_64/
~]# ls
kmod-bnx2x-1.710.51-3.el7_0.x86_64.rpm  kmod-bnx2x-firmware-
1.710.51-3.el7_0.x86_64.rpm  repodata
```

   Dans la sortie ci-dessus, la version du package est **1.710.51** et la version est **3.el7_0**.

5. Installer les fichiers RPM simultanément. Exemple :

```
~]# yum install kmod-bnx2x-1.710.51-3.el7_0.x86_64.rpm kmod-bnx2x-
firmware-1.710.51-3.el7_0.x86_64.rpm
Loaded plugins: product-id, subscription-manager
This system is not registered to Red Hat Subscription Management.
You can use subscription-manager to register.
Examining kmod-bnx2x-1.710.51-3.el7_0.x86_64.rpm: kmod-bnx2x-
1.710.51-3.el7_0.x86_64
Marking kmod-bnx2x-1.710.51-3.el7_0.x86_64.rpm to be installed
Examining kmod-bnx2x-firmware-1.710.51-3.el7_0.x86_64.rpm: kmod-
bnx2x-firmware-1.710.51-3.el7_0.x86_64
Marking kmod-bnx2x-firmware-1.710.51-3.el7_0.x86_64.rpm to be
installed
Resolving Dependencies
--> Running transaction check
---> Package kmod-bnx2x.x86_64 0:1.710.51-3.el7_0 will be installed
---> Package kmod-bnx2x-firmware.x86_64 0:1.710.51-3.el7_0 will be
installed
--> Finished Dependency Resolution

Dependencies Resolved
```

```
================================================================================
 Package              Arch    Version           Repository
================================================================================
Installing:
 kmod-bnx2x           x86_64  1.710.51-3.el7_0  /kmod-bnx2x-
1.710.51-3.el7_0.x8
 kmod-bnx2x-firmware  x86_64  1.710.51-3.el7_0  /kmod-bnx2x-
firmware-1.710.51-3

Transaction Summary
================================================================================
Install  2 Packages

Total size: 1.6 M
Installed size: 1.6 M
Is this ok [y/d/N]:
```

6. Saisir la commande suivante pour que **depmod** puisse interroger tous les modules et mettre à jour la liste des dépendances :

```
~]# depmod -a
```

7. Faire une copie de sauvegarde du *système de fichiers RAM initial*, en saisissant la commande suivante :

```
~]# cp /boot/initramfs-$(uname -r).img /boot/initramfs-$(uname -
r).img.$(date +%m-%d-%H%M%S).bak
```

8. Construire à nouveau le système de fichiers RAM initial :

```
~]# dracut -f -v
```

9. Pour faire la liste de contenu du fichier sur une image de système de fichiers RAM initial créé par dracut, saisir la commande suivante :

```
~]# lsinitrd /boot/initramfs-3.10.0-229.el7.x86_64.img
```

La sortie de commande est très longue, filtrer par la barre verticale cette sortie avec **less** ou **grep** pour trouver le module que vous êtes en train de mettre à jour. Exemple :

```
~# lsinitrd /boot/initramfs-3.10.0-229.el7.x86_64.img | grep bnx
drwxr-xr-x 2 root  root        0 Jun  9 11:25 usr/lib/firmware/bnx2x
-rw-r--r-- 1 root  root   164392 Nov 25  2014
usr/lib/firmware/bnx2x/bnx2x-e1-7.10.51.0.fw
-rw-r--r-- 1 root  root   173016 Nov 25  2014
usr/lib/firmware/bnx2x/bnx2x-e1h-7.10.51.0.fw
-rw-r--r-- 1 root  root   321456 Nov 25  2014
usr/lib/firmware/bnx2x/bnx2x-e2-7.10.51.0.fw
drwxr-xr-x 2 root  root        0 Jun  9 11:25
usr/lib/modules/3.10.0-
229.el7.x86_64/kernel/drivers/net/ethernet/broadcom/bnx2x
```

```
-rw-r--r-- 1 root  root  1034553 Jan 29 19:11
usr/lib/modules/3.10.0-
229.el7.x86_64/kernel/drivers/net/ethernet/broadcom/bnx2x/bnx2x.ko
```

10. Le système doit être redémarré pour que les changements entrent en vigueur.

Si nécessaire, pour afficher le pilote in-kernel en cours, utiliser la commade **modinfo *driver_name*** comme suit :

```
~]# modinfo bnx2x
filename:       /lib/modules/3.10.0-
229.el7.x86_64/kernel/drivers/net/ethernet/broadcom/bnx2x/bnx2x.ko
firmware:       bnx2x/bnx2x-e2-7.10.51.0.fw
firmware:       bnx2x/bnx2x-e1h-7.10.51.0.fw
firmware:       bnx2x/bnx2x-e1-7.10.51.0.fw
version:        1.710.51-0
license:        GPL
description:    Broadcom NetXtreme II
BCM57710/57711/57711E/57712/57712_MF/57800/57800_MF/57810/57810_MF/57840/5
7840_MF Driver
author:         Eliezer Tamir
rhelversion:    7.1
```

26.8. SIGNER DES MODULES DE NOYAU POUR LE DÉMARRAGE SÉCURISÉ « SECURE BOOT »

Red Hat Enterprise Linux 7 inclut la prise en charge de la fonctionnalité de démarrage sécurisé « UEFI Secure Boot », ce qui signifie que Red Hat Enterprise Linux 7 peut être installé et exécuté sur des systèmes où le démarrage sécurisé « UEFI Secure Boot » est activé. Notez que Red Hat Enterprise Linux 7 ne requiert pas l'utilisation de « UEFI Secure Boot » sur les systèems UEFI.

Lorsque Secure Boot est activé, les chargeurs de démarrage des systèmes d'exploitation EFI, le noyau Red Hat Enterprise Linux, et tous les modules de noyau doivent être signés avec une clé privée et authentifiés avec la clé publique correspondante. La distribution Red Hat Enterprise Linux 7 inclut des chargeurs de démarrage signés et des modules de noyau signés. En outre, le chargeur de démarrage signé de la première étape et le noyau signé incluent des clés publiques Red Hat intégrées. Ces binaires exécutables signés et clés intégrées permettent à Red Hat Enterprise Linux 7 d'installer, de démarrer, et d'exécuter avec les clés de l'autorité de certification Microsoft UEFI Secure Boot fournies par le microprogramme UEFI sur les systèmes qui prennent en charge les démarrages UEFI Secure Boot. Notez que tous les systèmes basés UEFI n'incluent pas la prise en charge de Secure Boot.

Les informations fournies dans les sections suivantes décrivent les étapes nécessaires vous permettant d'auto-signer des modules de noyau créés de manière privée pour une utilisation avec Red Hat Enterprise Linux 7 sur des systèmes basés UEFI où Secure Boot est activé. Ces sections fournissent également un aperçu des options disponibles pour obtenir votre clé publique sur le système cible sur lequel vous souhaitez déployer le module de noyau.

26.8.1. Conditions préalables

Pour permettre la signature de modules créés à l'externe, il est requis d'installer les outils répertoriés ci-dessous sur le système.

Tableau 26.1. Outils requis

Outil	Fourni par le paquet	Utilisé sur	But
openssl	openssl	Système de génération	Génère une paire de clés X.509 publique et privée
sign-file	kernel-devel	Système de génération	Script Perl utilisé pour signer les modules de noyau
perl	perl	Système de génération	Interprète Perl utilisé pour exécuter le script de signature
mokutil	mokutil	Système cible	Outil optionnel utilisé pour inscrire la clé publique manuellement
keyctl	keyutils	Système cible	Outil optionnel utilisé pour afficher des clés publiques dans l'anneau de clés du système

NOTE

Remarquez que le système de génération avec lequel vous créez et signez votre module de noyau ne nécessite pas qu'UEFI Secure Boot soit activé et ne nécessite même pas d'être un système basé UEFI.

26.8.2. Authentification du module de noyau

Dans Red Hat Enterprise Linux 7, lorsqu'un module de noyau est chargé, la signature du module est vérifiée à l'aide de clés X.509 publiques sur l'anneau de clés du système du noyau, à l'exception des clé se trouvant sur l'anneau des clés sur liste noire du système du noyau.

26.8.2.1. Sources de clés publiques utilisées pour authentifier des modules de noyau

Pendant le démarrage, le noyau charge les clés X.509 dans l'anneau de clés du système dans l'anneau des clés sur liste noire du système à partir d'un ensemble de magasins de clés persistantes, comme décrit dans la Tableau 26.2, « Sources pour les anneaux de clés système »

Tableau 26.2. Sources pour les anneaux de clés système

Source des clés X.509	Capacité de l'utilisateur à ajouter des clés	État UEFI Secure Boot	Clés chargées pendant le démarrage
Intégré au noyau	Non	-	**.system_keyring**
UEFI Secure Boot "db"	Limité	Non activé	Non

Source des clés X.509	Capacité de l'utilisateur à ajouter des clés	État UEFI Secure Boot	Clés chargées pendant le démarrage
		Activé	**.system_keyring**
UEFI Secure Boot "dbx"	Limité	Non activé	Non
		Activé	**.system_keyring**
Intégré au chargeur de démarrage **shim.efi**	Non	Non activé	Non
		Activé	**.system_keyring**
Liste MOK (« Machine Owner Key »)	Oui	Non activé	Non
		Activé	**.system_keyring**

Remarquez que si le système n'est pas basé UEFI ou si UEFI Secure Boot n'est pas activé, alors seules les clés intégrées au noyau seront chargées dans l'anneau des clés du système et vous n'aurez pas la capacité d'augmenter cet ensemble de clés sans regénérer le noyau. L'anneau des clés sur liste noire du système est une liste de clés X.509 qui ont été révoquées. Si votre module est signé par une clé sur la liste noire, alors son authentification échouera même si votre clé publique se trouve dans l'anneau des clés du système.

Vous pouvez afficher des informations concernant les clés sur les anneaux des clés du système en utilisant l'utilitaire **keyctl**. Ci-dessous figure un exemple abbrévié d'un système Red Hat Enterprise Linux 7 sur lequel UEFI Secure Boot n'est pas activé.

```
~]# keyctl list %:.system_keyring
3 keys in keyring:
...asymmetric: Red Hat Enterprise Linux Driver Update Program (key 3):
bf57f3e87...
...asymmetric: Red Hat Enterprise Linux kernel signing key:
4249689eefc77e95880b...
...asymmetric: Red Hat Enterprise Linux kpatch signing key:
4d38fd864ebe18c5f0b7...
```

Ci-dessous figure un exemple de sortie abbréviée d'un système Red Hat Enterprise Linux 7 sur lequel UEFI Secure Boot est activé.

```
~]# keyctl list %:.system_keyring
6 keys in keyring:
...asymmetric: Red Hat Enterprise Linux Driver Update Program (key 3):
bf57f3e87...
...asymmetric: Red Hat Secure Boot (CA key 1):
4016841644ce3a810408050766e8f8a29...
...asymmetric: Microsoft Corporation UEFI CA 2011:
13adbf4309bd82709c8cd54f316ed...
...asymmetric: Microsoft Windows Production PCA 2011:
a92902398e16c49778cd90f99e...
```

```
...asymmetric: Red Hat Enterprise Linux kernel signing key:
4249689eefc77e95880b...
...asymmetric: Red Hat Enterprise Linux kpatch signing key:
4d38fd864ebe18c5f0b7...
```

La sortie ci-dessus montre l'ajout de deux clés provenant des clés UEFI Secure Boot "db" plus **`Red Hat Secure Boot (CA key 1)`**, qui est intégrée au chargeur de démarrage **`shim.efi`**. Vous pouvez également chercher les messages de la console du noyau qui identifient les clés avec une source liée à UEFI Secure Boot, c'est-à-dire UEFI Secure Boot db, un shim intégré, et une liste MOK.

```
~]# dmesg | grep 'EFI: Loaded cert'
[5.160660] EFI: Loaded cert 'Microsoft Windows Production PCA 2011:
a9290239...
[5.160674] EFI: Loaded cert 'Microsoft Corporation UEFI CA 2011:
13adbf4309b...
[5.165794] EFI: Loaded cert 'Red Hat Secure Boot (CA key 1):
4016841644ce3a8...
```

26.8.2.2. Conditions préalables à l'authentification de modules de noyau

Si UEFI Secure Boot est activé ou si le paramètre de noyau **`module.sig_enforce`** a été spécifié, alors seuls les modules de noyau signés qui sont authentifiés à l'aide d'une clé sur l'anneau des clés du système pourront être chargés, à condition que la clé publique ne se trouve pas sur l'anneau des clés sur liste noire du système. Si UEFI Secure Boot est désactivé et si le paramètre de noyau **`module.sig_enforce`** n'a pas été spécifié, alors les modules de noyau non-signés et les modules de noyau signés sans clé publique pourront être chargés. Ceci est résumé dans la Tableau 26.3, « Conditions préalables à l'authentification de modules de noyau pour effectuer un chargement ».

Tableau 26.3. Conditions préalables à l'authentification de modules de noyau pour effectuer un chargement

Module signé	Clé publique trouvée et signature valide	État UEFI Secure Boot	module.sig_enforce	Charge du module	Noyau avarié
Non signé	-	Non activé	Non activé	Réussi	Oui
		Non activé	Activé	Échoué	
		Activé	-	Échoué	-
Signé	Non	Non activé	Non activé	Réussi	Oui
		Non activé	Activé	Échoué	-
		Activé	-	Échoué	-
Signé	Oui	Non activé	Non activé	Réussi	Non
		Non activé	Activé	Réussi	Non

Module signé	Clé publique trouvée et signature valide	État UEFI Secure Boot	module.sig_enforce	Charge du module	Noyau avarié
		Activé	-	Réussi	Non

Les sections suivantes décrivent comment générer une paire de clés X.509 publique et privée, comment utiliser la clé privée pour signer un module de noyau, et comment inscrire la clé publique dans une source pour l' anneau des clés du système.

26.8.3. Générer une paire de clés X.509 publique et privée

Vous devez générer une paire de clés X.509 publique et privée qui sera utilisée pour signer un module de noyau une fois qu'il aura été créé. La clé publique correspondante sera utilisée pour authentifier le module du noyau après son chargement.

1. L'outil **openssl** peut être utilisé pour générer une paire de clés qui satisfait les conditions préalables à la signature d'un module de noyau sur Red Hat Enterprise Linux 7. Certains des paramètres de cette requêtre de génération de clé sont mieux spécifiés à l'aide d'un fichier de configuration ; veuillez suivre l'exemple ci-dessous pour créer votre propre fichier de configuration.

```
~]# cat << EOF > configuration_file.config
[ req ]
default_bits = 4096
distinguished_name = req_distinguished_name
prompt = no
string_mask = utf8only
x509_extensions = myexts

[ req_distinguished_name ]
O = Organization
CN = Organization signing key
emailAddress = E-mail address

[ myexts ]
basicConstraints=critical,CA:FALSE
keyUsage=digitalSignature
subjectKeyIdentifier=hash
authorityKeyIdentifier=keyid
EOF
```

2. Après avoir créé le fichier de configuration, vous pouvez créer une paire de clés X.509 publique et privée. La clé publique sera écrite sur le fichier ***public_key*.der** et la clé privée sera écrite sur le fichier ***private_key*.priv**.

```
~]# openssl req -x509 -new -nodes -utf8 -sha256 -days 36500 \ -batch
-config configuration_file.config -outform DER \ -out public_key.der
\ -keyout private_key.priv
```

3. Veuillez inscrire votre clé publique sur tous les systèmes sur lesquels vous souhaitez authentifier et charger votre module de noyau.

AVERTISSEMENT

Prenez soin de bien garder privé le contenu de votre clé privée. Dans de mauvaises mains, cette clé pourrait être utilisée pour compromettre tout système qui possède votre clé publique.

26.8.4. Inscrire une clé publique sur un système cible

Lorsque Red Hat Enterprise Linux 7 démarre sur un système bas UEFI avec Secure Boot activé, toutes les clés se trouvant dans la base de données de clés Secure Boot db, mais pas dans la base de données dbx des clés révoquées, sont chargées dans l'anneau des clés du système par le noyau. L'anneau des clés du système est utilisé pour authentifier des modules de noyau.

26.8.4.1. Image du microprogramme d'usine, y compris la clé publique

Pour faciliter l'authentification de modules de noyau sur vos systèmes, envisagez de demander à votre fournisseur de systèmes d'incorporer votre clé publique dans la base de données de clés UEFI Secure Boot sous leur image du microprogramme d'usine.

26.8.4.2. Image d'inscription de clé exécutable ajoutant une clé publique

Il est possible d'ajouter une clé à une base de données Secure Boot remplie et active. Ceci peut être effectué en écrivant et en fournissant une image d'*inscription* exécutable EFI. Une telle image d'inscription contient une requête correctement formée d'ajout d'une clé à la base de données de clés Secure Boot. Cette requête doit inclure des données correctement signées par la clé privée qui correspond à une clé publique se trouvant déjà dans la base de données KEK (« Key Exchange Key ») de Secure Boot. En outre, cette image EFI doit être signée par une clé privée qui correspond à une clé publique se trouvant déjà dans la base de données des clés.

Il est également possible d'écrire une image d'inscription exécutée sous Red Hat Enterprise Linux 7. Cependant, l'image Red Hat Enterprise Linux 7 doit être correctement signée par une clé privée correspondant à une clé publique se trouvant déjà dans la base de données KEK.

La construction de chaque type d'image d'inscription de clé requiert de l'aide de la part du fournisseur de plateforme.

26.8.4.3. Administrateur systèmes ajoutant manuellement la clé publique à la liste MOK

La fonctionnalité MOK (« Machine Owner Key ») est prise en charge par Red Hat Enterprise Linux 7 et peut être utilisée pour élargir la base de données de clés UEFI Secure Boot. Lorsque Red Hat Enterprise Linux 7 démarre sur un système activé UEFI avec Secure Boot activé, les clés de la liste MOK sont également ajoutées à l'anneau des clés du système en plus d'être ajoutées aux clés de la base de données des clés. Les clés de la liste MOK sont également stockées de manière persistante et sécurisée de la même manière que les clés de la base de données de clés Secure Boot, mais il s'agit de deux installations différentes. L'installation MOK est prise en charge par shim.efi, MokManager.efi, grubx64.efi, et par l'utilitaire Red Hat Enterprise Linux 7 **`mokutil`**.

La capacité principale fournie par l'installation MOK consiste à ajouter des clés publiques à la liste MOK sans avoir besoin de récupérer la chaîne de la clé sur une autre clé déjà présente dans la base de données KEK. Cependant, l'inscription d'une clé MOK requiert une interaction manuelle de la part d'un

utilisateur *présent physiquement* sur la console système UEFI sur chaque système cible. Néanmoins, l'installation MOK offre une excellente méthode pour tester les nouvelles paires de clés générées et pour tester les modules de noyau signés avec celles-ci.

Veuillez suivre ces étapes pour ajouter votre clé publique à la liste MOK :

1. Demandez l'ajout de votre clé publique à la liste MOK en utilisant un utilitaire d'espace utilisateur Red Hat Enterprise Linux 7 :

   ```
   ~]# mokutil --import my_signing_key_pub.der
   ```

   Il vous sera demandé de saisir et de confirmer un mot de passe pour cette requête d'inscription MOK.

2. Redémarrez la machine.

3. La requête d'inscription de clé MOK en attente sera remarquée par **shim.efi** et lancera **MokManager.efi** afin de vous permettre de terminer l'inscription à partir de la console UEFI. Vous devrez saisir le mot de passe précédemment associé à cette requête et confirmer l'inscription. Votre clé publique est ajoutée à la liste MOK, qui est persistante.

Une fois qu'une clé est sur la liste MOK, elle sera automatiquement propagée sur l'anneau des clés du système et pendant les démarrages suivants, lorsqu'UEFI Secure Boot est activé.

26.8.5. Signer un module de noyau avec la clé privée

Il n'y a pas d'étape supplémentaire requise pour préparer votre module de noyau à une signature. Vous pouvez générer votre module de noyau normalement. En supposant qu'un Makefile approprié et ses sources correspondantes soient effectivement présents, veuillez suivre ces étapes pour générer votre module et le signer :

1. Générez votre module **my_module.ko** de manière standard :

   ```
   ~]# make -C /usr/src/kernels/$(uname -r) M=$PWD modules
   ```

2. Signez votre module de noyau avec votre clé privée. Ceci peut être fait avec un script Perl. Remarquez que le script requiert que les fichiers contenant la clé privée et la clé publique soient fournis, ainsi que le fichier du module de noyau que vous souhaitez signer.

   ```
   ~]# perl /usr/src/kernels/$(uname -r)/scripts/sign-file \ sha256 \
   my_signing_key.priv \ my_signing_key_pub.der \ my_module.ko
   ```

Votre module se trouve sous le format d'image ELF et ce script calcule et ajoute la signature directement sur l'image ELF dans votre fichier **my_module.ko**. L'utilitaire **modinfo** peut être utilisé pour afficher des informations sur la signature du module du noyau, si elle est présente. Pour obtenir des informations sur l'utilisation de l'utilitaire, veuillez consulter la Section 26.2, « Afficher des informations sur un module ».

Remarquez que cette signature ajoutée n'est pas contenue dans une section d'image ELF et n'est pas une partie formelle de l'image ELF. Ainsi, des outils tels que **readelf** ne seront pas en mesure d'afficher la signature sur votre module de noyau.

Votre module de noyau est désormais prêt à être chargé. Remarquez que votre module de noyau signé est également chargeable sur des systèmes où UEFI Secure Boot est désactivé ou sur un système non-UEFI. Cela signifie que vous ne devrez pas fournir une version signée et une version non-signée à la fois

de votre module de noyau.

26.8.6. Charger un module de noyau signé

Une fois que votre clé publique est inscrite et se trouve dans l'anneau de clés du système, les mécanismes normaux de chargement de modules de noyau fonctionneront de manière transparente. Dans l'exemple suivant, vous utiliserez **mokutil** pour ajouter votre clé publique à la liste MOK et vous chargerez votre module de noyau manuellement avec **modprobe**.

1. Optionnellement, vous pouvez vérifier que votre module de noyau ne se charge pas avant d'avoir inscrit la clé publique. Premièrement, veuillez vérifier quelles clés ont été ajoutées à l'anneau des clés du système lors du démarrage actuel en exécutant la commande **keyctl list %:.system_keyring** en tant qu'utilisateur root. Comme votre clé publique n'a pas encore été inscrite, elle ne sera pas affichée dans la sortie de la commande.

2. Demander l'inscription de votre clé publique.

```
~]# mokutil --import my_signing_key_pub.der
```

3. Redémarrez et terminez l'inscription sur la console UEFI.

```
~]# reboot
```

4. Après les redémarrages système, veuillez vérifier à nouveau les clés sur l'anneau des clés du système.

```
~]# keyctl list %:.system_keyring
```

5. Vous devriez désormais être en mesure de charger votre module de noyau.

```
~]# modprobe -v my_module
insmod /lib/modules/3.10.0-123.el7.x86_64/extra/my_module.ko
~]# lsmod | grep my_module
my_module 12425 0
```

26.9. RESSOURCES SUPPLÉMENTAIRES

Pour obtenir davantage d'informations sur les modules de noyau et sur leurs utilitaires, veuillez consulter les ressources suivantes.

Documentation de la page du manuel

- **lsmod(8)** — page man de la commande **lsmod**.
- **modinfo(8)** — page man de la commande **modinfo**.
- **modprobe(8)** — page man de la commande **modprobe**.
- **rmmod(8)** — page man de la commande **rmmod**.
- **ethtool(8)** — page man de la commande **ethtool**.
- **mii-tool(8)** — page man de la commande **mii-tool**.

Documentation installable et externe

- **`/usr/share/doc/kernel-doc-`*`kernel_version`*`/Documentation/`** — Ce répertoire, fourni par le paquet kernel-doc, contient des informations sur le noyau sur les modules de noyau, et sur leurs paramètres respectifs. Avant d'accéder à la documentation du noyau, vous devez exécuter la commande suivante en tant qu'utilisateur **`root`** :

```
~]# yum install kernel-doc
```

- Linux Loadable Kernel Module HOWTO — « *Linux Loadable Kernel Module HOWTO* », du projet « Linux Documentation Project » contient davantage d'informations sur l'utilisation des modules de noyau.

PARTIE VIII. SAUVEGARDE ET RESTAURATION DU SYSTÈME

Cette partie décrit comment utiliser l'utilitaire de recouvrement et de migration de système ReaR (Relax-and-Recover).

CHAPITRE 27. RELAX-AND-RECOVER (REAR)

Lors des échecs système causés par des logiciels ou du matériel, l'administrateur systèmes doit procéder aux trois tâches suivantes pour restaurer le système totalement sur le nouvel environnement matériel :

1. démarrer l'ordinateur en « Mode de Secours » (Rescue Mode)
2. reproduire la configuration de stockage d'origine
3. restaurer les fichiers d'utilisateur et de système

La plupart des logiciels de sauvegarde ne résolvent que le troisième problème. Pour résoudre le premier et le deuxième problème, utiliser *Relax-and-Recover (ReaR)*, un utilitaire de recouvrement et de migration de système.

Le logiciel de sauvegarde crée des sauvegardes. ReaR complémente le logiciel de sauvegarde en créant un *système de secours*. L'initialisation du système de sauvetage sur un nouveau matériel vous permet d'émettre la commande **`rear recovery`**, qui démarre le processus de récupération. Au cours de ce processus, ReaR reproduit le schéma de partition et les systèmes de fichiers, et lance des invites pour restaurer les fichiers système et utilisateur de la sauvegarde créée par le logiciel de sauvegarde, et enfin, installe le boot loader. Par défaut, le système de sauvetage créé par ReaR restaure uniquement le schéma de stockage et le chargeur de démarrage, mais pas l'utilisateur réel et les fichiers système.

Ce chapitre décrit comment utiliser ReaR.

27.1. BASIC REAR USAGE

27.1.1. Installer ReaR

Installer la paquet rear et ses dépendances en exécutant la commande suivante en tant que root :

```
~]# yum install rear genisoimage syslinux
```

27.1.2. Configurer ReaR

ReaR est configuré dans le fichier **`/etc/rear/local.conf`**. Spécifier la configuration du système de secours en ajoutant les lignes suivantes :

```
OUTPUT=output format
OUTPUT_URL=output location
```

Remplacer *output format* par le format de secours système, par exemple, **`ISO`** pour un disque ou une image ISO, et **`USB`** pour une clé USB enfichable.

Rempalcer *output location* par le nom de l'emplacement, **`file:///mnt/rescue_system/`** par un répertoire de système de fichiers local ou **`sftp://backup:password@192.168.0.0/`** pour un répertoire SFTP.

Exemple 27.1. Configurer l'emplacement et le format du système de secours système

Pour configurer ReaR pour qu'il transfère le système de secours en image ISO dans le répertoire **`/mnt/rescue_system/`**, ajouter les lignes suivantes au fichier **`/etc/rear/local.conf`** :

```
OUTPUT=ISO
OUTPUT_URL=file:///mnt/rescue_system/
```

Voir la section "Rescue Image Configuration" de la page man sur rear(8) pour obtenir une liste de toutes les options.

27.1.3. Créer un système de secours

L'exemple suivant vous explique comment créer un système de secours avec une sortie en mode détaillé :

```
~]# rear -v mkrescue
Relax-and-Recover 1.17.2 / Git
Using log file: /var/log/rear/rear-rhel7.log
mkdir: created directory '/var/lib/rear/output'
Creating disk layout
Creating root filesystem layout
TIP: To login as root via ssh you need to set up
/root/.ssh/authorized_keys or SSH_ROOT_PASSWORD in your configuration file
Copying files and directories
Copying binaries and libraries
Copying kernel modules
Creating initramfs
Making ISO image
Wrote ISO image: /var/lib/rear/output/rear-rhel7.iso (124M)
Copying resulting files to file location
```

Avec la configuration de Exemple 27.1, « Configurer l'emplacement et le format du système de secours système », ReaR affiche la sortie suivante. Les deux dernières lignes confirment que le système de secours a bien été créé, et copié dans l'emplacement de sauvegarde configuré **/mnt/rescue_system/**. Comme le nom d'hôte du sysème est **rhel7**, l'emplacement de sauvegarde contient maintenant le répertoire **rhel7/** avec le système de secours et les fichiers auxiliaires :

```
~]# ls -lh /mnt/rescue_system/rhel7/
total 124M
-rw-------. 1 root root  202 Jun 10 15:27 README
-rw-------. 1 root root 166K Jun 10 15:27 rear.log
-rw-------. 1 root root 124M Jun 10 15:27 rear-rhel7.iso
-rw-------. 1 root root  274 Jun 10 15:27 VERSION
```

Transférer le système de secours vers un medium externe afin de ne rien perdre en cas de désastre.

27.1.4. Programmer ReaR

Pour configurer ReaR afin qu'il crée régulièrement un système de secours en utilisant **cron**, ajouter la ligne suivante au fichier **/etc/crontab** :

```
minute hour day_of_month month day_of_week root /usr/sbin/rear mkrescue
```

Remplacer la commande ci-dessus par la spécification de durée cron (décrite en détail dans Section 21.1.4, « Configuration des tâches Cron »).

Exemple 27.2. Programmer ReaR

Pour que ReaR crée un système de secours à 22:00 tous les jours de la semaine, ajouter cette ligne au fichier **`/etc/crontab`** :

```
0 22 * * 1-5 root /usr/sbin/rear mkrescue
```

27.1.5. Dépanner un système

Pour effectuer une restauration ou une migration :

1. Démarrer le système de secours sur le nouveau matériel. Par exemple, graver l'image ISO sur un DVD et démarrez à partir du DVD.
2. Dans l'interface de la console, sélectionner l'option « Recover » :

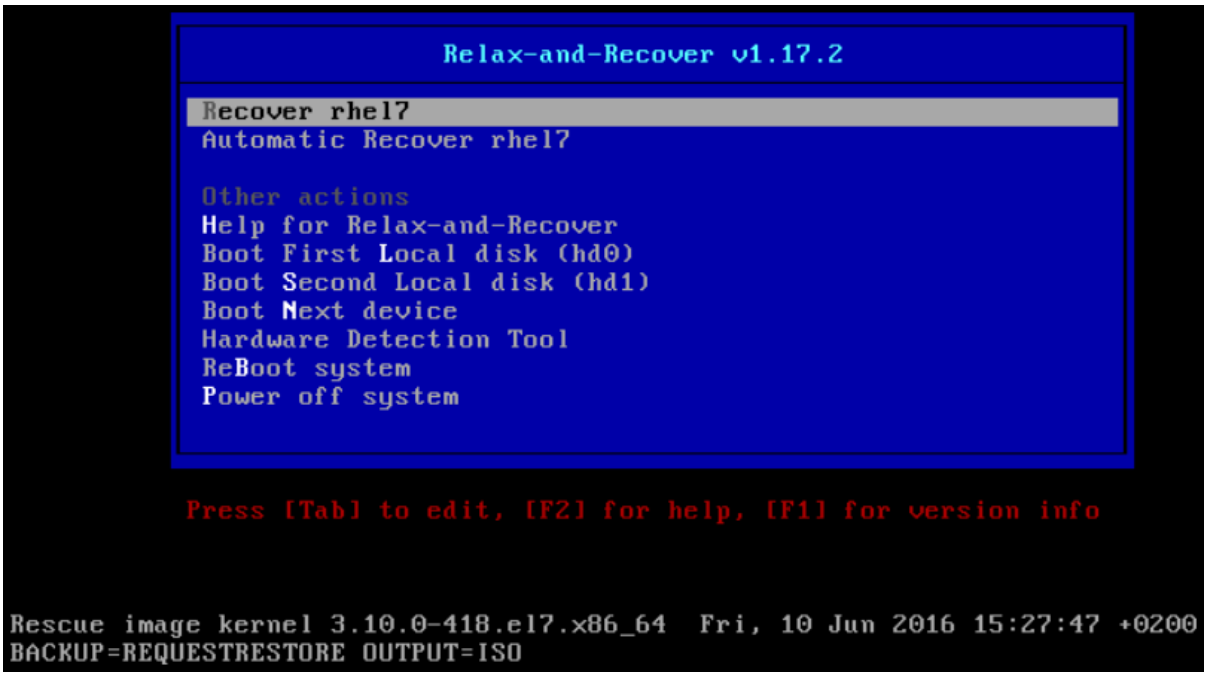


Figure 27.1. Rescue system: menu

3. Invite :

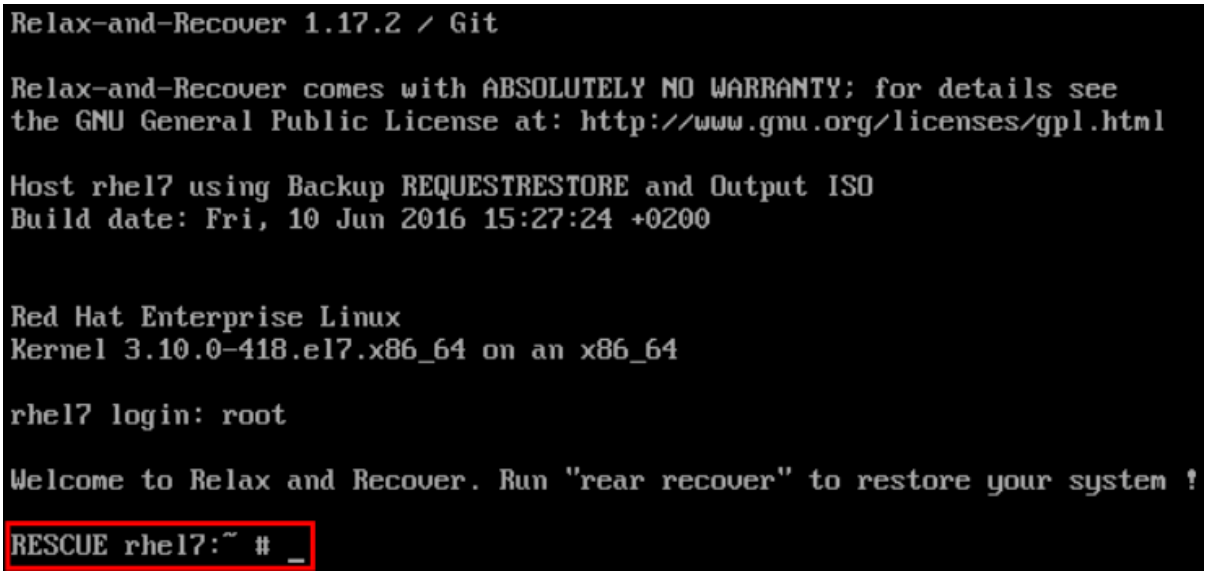


Figure 27.2. Rescue system: invite

AVERTISSEMENT

Une fois que vous aurez démarré le processus de restauration, vous ne pourrez sans doute pas revenir en arrière, et vous risquez de perdre ce que vous avez stocké dans les disques physiques du système.

4. Exécuter la commande **`rear recover`** pour effectuer la restauration ou la migration. Le système de migration pourra alors créer les partitions et les systèmes de fichiers :

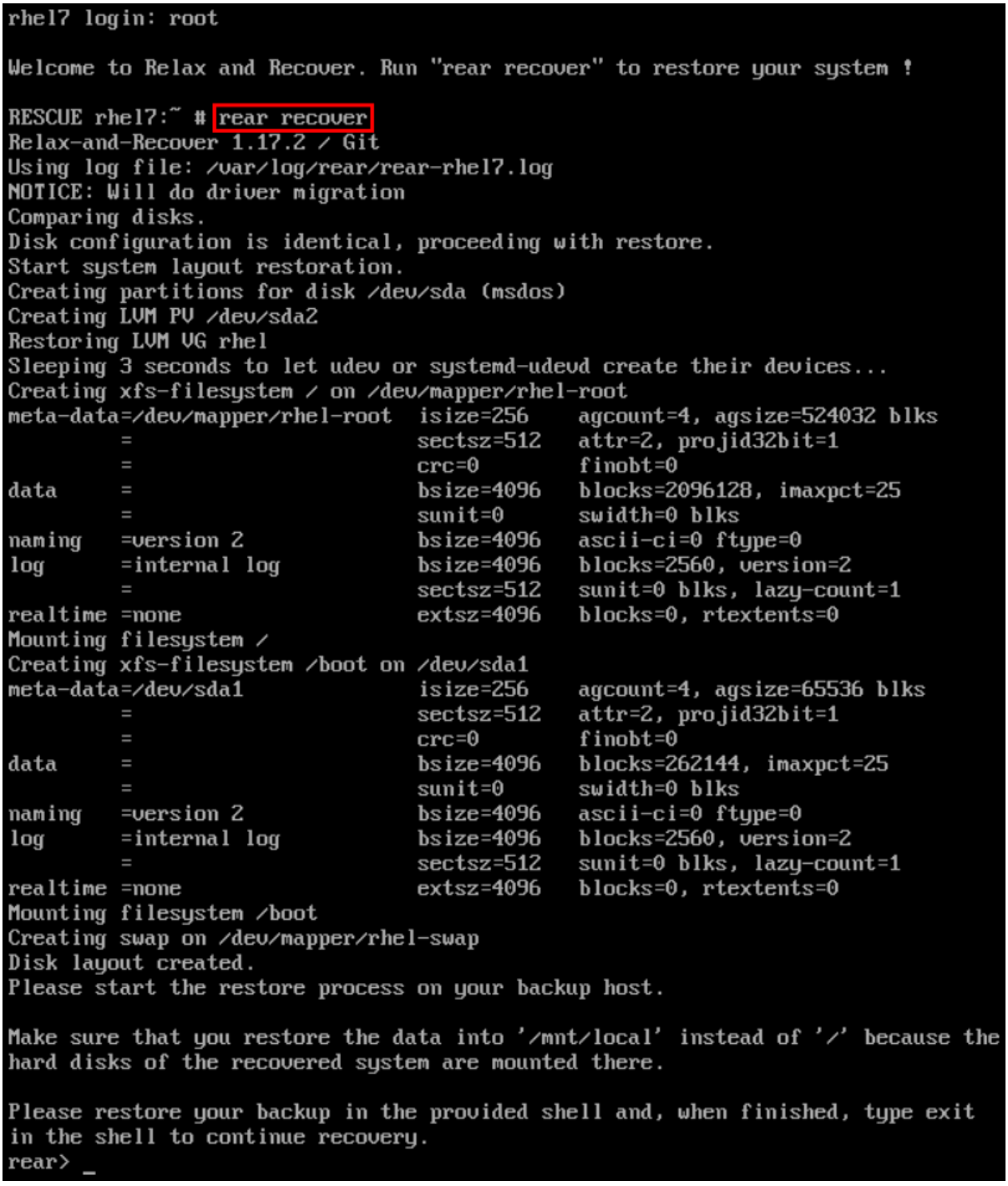


Figure 27.3. Rescue system: exécuter « rear recover »

5. Restaurer les fichiers utilisateur et système à partir de la sauvegarde dans le répertoire **`/mnt/local/`**.

> **Exemple 27.3. Restaurer les fichiers d'utilisateur et de système**
>
> Dans cet exemple, le fichier de sauvegarde est une archive **tar** créée selon les instructions qui se trouvent dans Section 27.2.1.1, « Configurer la méthode de sauvegarde interne ». Veuillez, tout d'abord, copier l'archive de son stockage, puis dépaqueter les fichiers dans **`/mnt/local/`**, et finalement, supprimer l'archive :
>
> ```
> ~]# scp root@192.168.122.7:/srv/backup/rhel7/backup.tar.gz
> /mnt/local/
> ~]# tar xf /mnt/local/backup.tar.gz -C /mnt/local/
> ```

```
~]# rm -f /mnt/local/backup.tar.gz
```

Le nouveau stockage doit avoir sufisamment d'espace pour l'archive et les fichiers extraits à la fois.

6. Vérifiez bien que les fichiers ont bien été restaurés :

```
~]# ls /mnt/local/
```

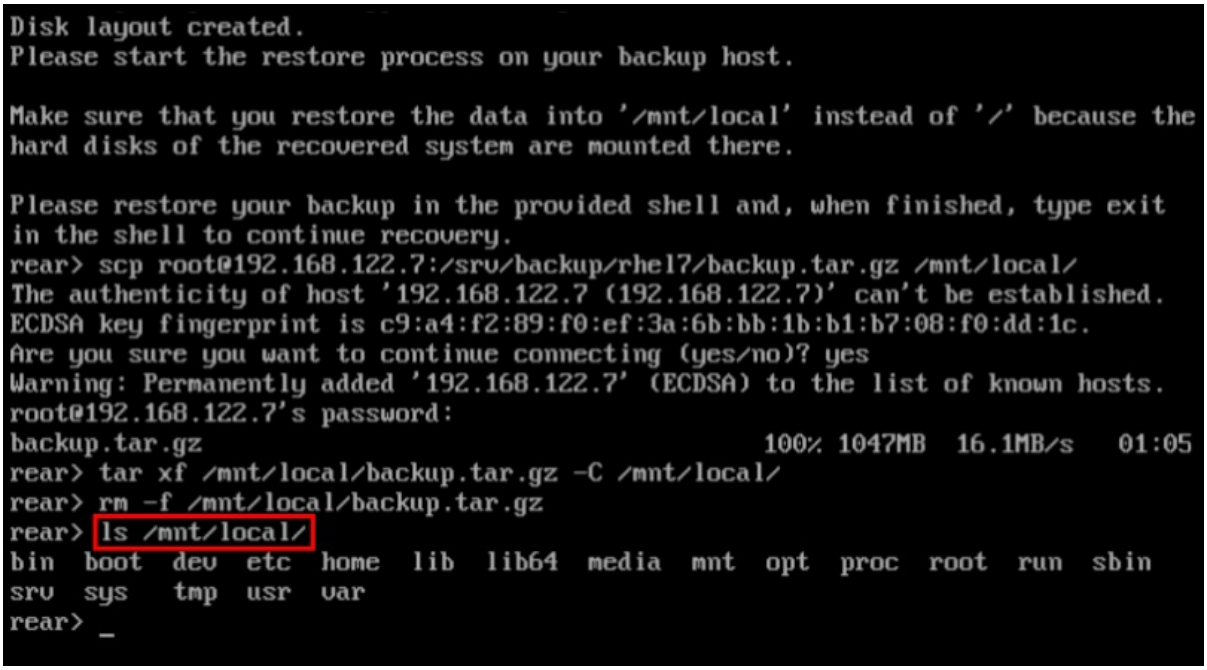


Figure 27.4. Rescue system: restaurer les fichiers d'utilisateur et de système à partir de la copie de sauvegarde

7. Veuillez à ce que SELinux renomme les fichiers lors du nouveau démarrage :

```
~]# touch /mnt/local/.autorelabel
```

Sinon, vous risquez de ne pas pouvoir vous connecter au système, car le fichier **/etc/passwd** pourrait avoir un contexte SELinux incorrect.

8. Terminer le processus de recouvrement en saisissant la commande **exit**. ReaR installera alors le boot loader. Ensuite, redémarrez le système :

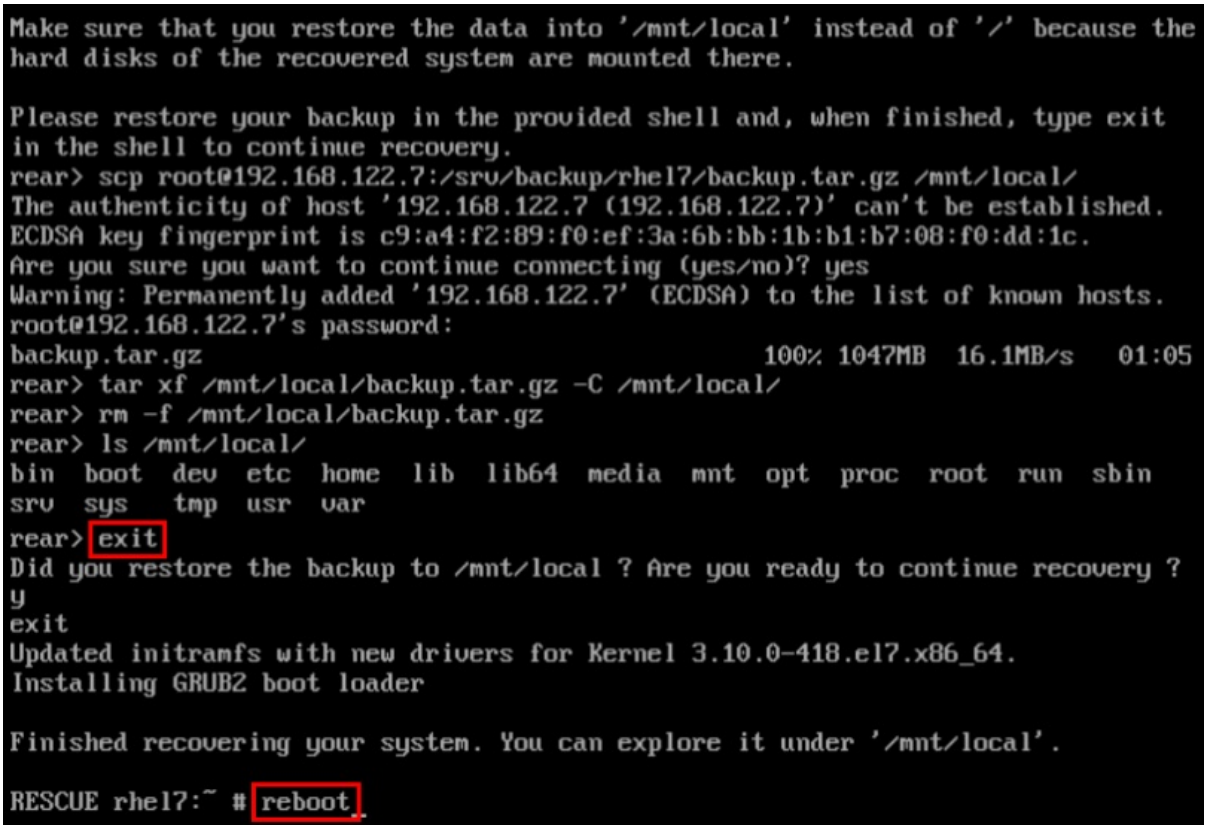


Figure 27.5. Rescue system: terminer la restauration

Lors du second démarrage, SELinux renomme le système de fichiers dans son ensemble. Vous pourrez, ensuite, vous connecter au système restauré.

27.2. INTÉGRER REAR AU LOGICIEL DE SAUVEGARDE

Le principal but de ReaR est de produire un système de secours, mais il peut également être intégré au logiciel de sauvegarde. L'intégration a un sens différent pour les méthodes de sauvegarde intégrées, prises en charge et non prises en charge.

27.2.1. La méthode de sauvegarde intégrée

ReaR est livré avec une méthode de sauvegarde interne ou intégrée. Cette méthode est totalement intégrée dans ReaR, et comporte les avantages suivants :

- un système de secours et une sauvegarde de système totale peuvent être créés en un coup par la commande **`rear mkbackup`**
- le système de secours restaure les fichiers de la sauvegarde automatiquement

De ce fait, ReaR peut couvrir tout le processus de création du système de secours et de sauvegarde du système dans sa totalité.

27.2.1.1. Configurer la méthode de sauvegarde interne

Pour que ReaR utilise sa méthode de sauvegarde interne, ajouter les lignes suivantes au fichier **`/etc/rear/local.conf`** :

```
BACKUP=NETFS
BACKUP_URL=backup location
```

Ces lignes configurent ReaR pour créer une archive avec une sauvegarde complète du système à l'aide de la commande "**tar**". Remplacez l'*emplacement de sauvegarde* par l'une des options de la section « Backup Software Integration » de la page man de rear(8). Assurez-vous que l'emplacement de sauvegarde possède assez d'espace.

Exemple 27.4. Ajouter des Sauvegardes tar

Pour étendre l'exemple dans Section 27.1, « Basic ReaR Usage », configurez ReaR pour pouvoir créer en sortie une sauvegarde **tar** du système dans son entier dans le répertoire **/srv/backup/** :

```
OUTPUT=ISO
OUTPUT_URL=file:///mnt/rescue_system/
BACKUP=NETFS
BACKUP_URL=file:///srv/backup/
```

La méthode de sauvegarde interne permet des configurations supplémentaires.

- Pour conserver d'anciennes archives de sauvegarde quand vous en créez de nouvelles, ajouter la ligne suivante :

  ```
  NETFS_KEEP_OLD_BACKUP_COPY=y
  ```

- Par défaut, ReaR crée une sauvegarde totale pour chaque exécution. Pour rendre les sauvegardes incrémentales, c'est à dire, pour que les fichiers modifiés soient sauvegardés à chaque exécution, ajouter la ligne suivante :

  ```
  BACKUP_TYPE=incremental
  ```

  Ceci définit **NETFS_KEEP_OLD_BACKUP_COPY** automatiquement à **y**.

- Pour qu'une sauvegarde complète soit faîte régulièrement, en plus des sauvegardes incrémentielles, ajouter la ligne suivante :

  ```
  FULLBACKUPDAY="Day"
  ```

  Remplacer *"Jour"* par un parmi "Lun", "Mar", "Mer", "Jeu". "Ven", "Sam", "Dim".

- ReaR peut également inclure le système de secours et la sauvegarde de l'image ISO à la fois. Pour cela, définir **BACKUP_URL** à **iso:///backup/**:

  ```
  BACKUP_URL=iso:///backup/
  ```

  C'est la manière la plus simple d'effectuer une sauvegarde complète d'un système, car le système de secours n'a pas besoin que l'utilisateur aille chercher la sauvegarde lors du processus de recouvrement. Cependant, cela nécessite plus de stockage. Aussi, les sauvegardes d'ISO simples ne peuvent pas être incrémentales.

NOTE

ReaR crée actuellement deux copies de l'image ISO, et utilise donc deux fois plus de stockage. Pour plus d'informations, voir le chapitre *ReaR crée deux images ISO au lieu d'une* qui se trouve dans les Notes de sortie de Red Hat Enterprise Linux 6.

Exemple 27.5. Configurer un Système de secours ISO-Single et les Sauvegardes

Cette configuration crée un système de secours et un fichier de sauvegarde sous forme d'une image ISO et la met dans le répertoire **/srv/backup/** :

```
OUTPUT=ISO
OUTPUT_URL=file:///srv/backup/
BACKUP=NETFS
BACKUP_URL=iso:///backup/
```

- Pour utiliser **rsync** à la place de **tar**, ajouter la ligne suivante :

```
BACKUP_PROG=rsync
```

Notes que les sauvegardes incrémentielles ne sont pas prisses en charge quand on utilise la commande **tar**.

27.2.1.2. Créer une sauvegarde par la Méthode de sauvegarde interne

Avec **BACKUP=NETFS** défini, ReaR peut soit créer un système de secours, soit un fichier de sauvegarde, ou les deux.

- Pour créer *un système de secours uniquement*, exécutez :

```
rear mkrescue
```

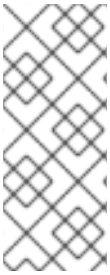
- Pour créer *une sauvegarde uniquement*, exécutez :

```
rear mkbackuponly
```

- Pour créer *un système de secours et une sauvegarde*, exécutez :

```
rear mkbackup
```

Notez que vous ne pouvez initier de sauvagarde ReaR que si vous utilisez la méthode NETFS. ReaR ne peut pas déclencher d'autres méthodes de sauvegarde.



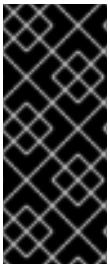
NOTE

En cours de restauration, le système de secours créé avec le paramètre **BACKUP=NETFS** s'attend à ce que la sauvegarde existe avant d'exécuter **`rear recover`**. Ainsi, une fois que le système de secours démarre, copier le fichier de sauvegarde dans le répertoire de sauvegarde indiqué dans **BACKUP_URL**, à moins que vous utilisiez une seule image ISO. Seulement après cela, exécutez **`rear recover`**.

Pour éviter de recréer le système de secours sans bonne raison, vérifiez si la disposition du stockage a été modifiée depuis la dernière fois que le système a été secouru, par les commandes suivantes :

```
~]# rear checklayout
~]# echo $?
```

Un statut non-zéro indique un changement de disposition de disque. Un statut non-zéro est retourné également si la configuration de ReaR a changé.



IMPORTANT

La commande **`rear checklayout`** ne vérifie pas si le système de secours est actuellement présent dans la sortie, et peut retourner 0 même s'il n'est pas présent. Il n'y a donc aucune garantie que le système de secours soit disponible, cela indique uniquement un changement de disposition depuis la dernière création de système de secours.

Exemple 27.6. La commande rear checklayout

Pour créer un système de secours, mais uniquement si la disposiiton a changé, utiliser la commande :

```
~]# rear checklayout || rear mkrescue
```

27.2.2. Méthodes de sauvegarde prises en charge

En plus de la méthode de sauvegarde interne de NETFS, ReaR prend en charge plusieurs méthodes de sauvegarde externes. Cela signifie que le système de secours restaure des fichers de la sauvegarde automatiquement, mais la création de la sauvegarde ne peut pas être initiée par ReaR.

Pour obtenir une liste des options de configuration des méthodes de sauvegarde externes prises en charge, voir la section « Intégration du logiciel de sauvegarde » de la page man de rear(8).

27.2.3. Méthodes de sauvegarde non prises en charge

Pour les méthodes de sauvegarde non prises en charge, voici deux options :

1. Le système de secours invite l'utilisateur à restaurer les fichiers manuellement. Ce sécnario est décrit dans "Utilisation de ReaR de base", sauf pour le format de fichier de sauvegarde, qui peut prendre un forme différente que sous forme de **tar**.

2. ReaR exécute les commandes personnalisées fournies par l'utilisateur. Pour configurer cela, définir la directive **BACKUP** à **EXTERNAL**. Puis, spécifier les commandes à exécuter lors de la

sauvegarde et la restauration, en utilisant les directives **EXTERNAL_BACKUP** et **EXTERNAL_RESTORE**. Spécifiez, en option, les directives **EXTERNAL_IGNORE_ERRORS** et **EXTERNAL_CHECK** également. Voir **/usr/share/rear/conf/default.conf** pour obtenir un exemple de configuration.

ANNEXE A. RPM

Le système de gestion des paquets *RPM Package Manager* (**RPM**) est un système qui exécute sur Red Hat Enterprise Linux ainsi que sur d'autres systémes Linux ou UNIX. Red Hat et le Projet Fedora encouragent d'autres vendeurs à utiliser **RPM** pour leurs produits. **RPM** est distribué sous les termes et conditions de la licence *GPL* (*GNU General Public License*).

Le **RPM Package Manager** fonctionne uniquement avec les paquets créés sous le *format RPM*. **RPM** est fourni dans le paquet rpm pré-installé. Pour l'utilisateur final, **RPM** facilite la mise à jour des systèmes. L'installation, la désinstallation et la mise à niveau des paquets **RPM** est rendue possible grâce à quelques commandes. **RPM** maintient une base de données des paquets installés et leurs fichiers. Ainsi, vous pouvez faire des requêtes et des contrôles puissants à travers votre système. Il existe plusieurs applications, comme **Yum** ou **PackageKit**, qui peuvent faciliter davantage le travail avec les paquets au format **RPM**.

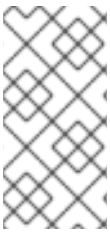
AVERTISSEMENT

Pour la plupart des tâches de gestion de paquets, le gestionnaire de paquets **Yum** offre égale et souvent une meilleure fonctionnalité et c'est un meilleur outil que **RPM**. **Yum** est également en mesure d'effectuer et d'assurer le suivi de résolutions de dépendances systèmes plus compliquées. **Yum** maintient l'intégrité du système et oblige un contrôle d'intégrité de système si des paquets sont installés ou supprimés par une autre application, comme **RPM** à la place de **Yum**. Pour ces raisons, il est fortement recommandé d'utiliser **Yum** à la place de **RPM**, lorsque c'est possible, afin d'effectuer des tâches de gestion des paquets. Voir Chapitre 8, *Yum*.

Si vous préférez une interface graphique, vous pouvez utiliser l'application GUI **PackageKit**, qui utilise **Yum** en arrière plan, pour gérer les paquets du système.

Lors des mises à niveau, **RPM** gère les fichiers de configuration avec soin, de façon à ne jamais perdre vos personnalisation, ce qui est une chose difficile à réaliser avec les fichiers `.tar.gz` habituels.

Pour les développeurs, **RPM** permet au code source du logiciel d'être présent dans les paquets sources et binaires pour les utilisateurs finaux. Ce processus est très simple et dérive d'un seul fichier et de correctifs optionnels que vous créez. Cette délimitation claire entre les sources intactes et vos correctifs ainsi que les instructions de création facilitent la maintenance du paquet quand de nouvelles versions du logiciel sont disponibles.



NOTE

Comme **RPM** peut apporter des modifications au système lui-même, effectuer des opérations comme l'installation, la mise à niveau, le déclassement, ou désinstaller des paquets binaires sur tout le système nécessite le niveau de privilège `root` dans la plupart des cas.

A.1. OBJECTIFS DE LA CONCEPTION RPM

Pour comprendre comment utiliser **RPM**, il est bon de comprendre le but de l'utilisation de **RPM**:

Upgradability

Avec **RPM**, vous pouvez mettre à jour différents composants de votre système sans besoin de réinstallation complète. Quand vous obtenez une nouvelle version du système d'exploitation basé sur **RPM**, tels que Red Hat Enterprise Linux, vous n'avez pas besoin de réinstaller une nouvelle copie du système d'exploitation sur votre machine (ce que vous devrez peut-être effectuer avec des systèmes d'exploitation basés sur d'autres systèmes de packaging). **RPM** permet des mises à jour intelligentes, entièrement automatisées, sur place, dans votre système. De plus, les paquets, les fichiers de configuration des paquets sont conservés à travers les mises à jour, ce qui vous permet de conserver vos personnalisations. Il n'y a aucun fichier de mise à niveau spécial nécessaire pour mettre à jour un paquet parce qu'un même fichier **RPM** est utilisé à la fois pour installer et mettre à niveau le paquet sur votre système.

Powerful Querying

RPM est conçu pour fournir des options de recherche puissantes. Vous pouvez effectuer des recherches de paquets ou même simplement des fichiers sur votre copie de la base de données. Vous pouvez également facilement trouver à quel paquet appartient un fichier et d'où le paquet vient. Les fichiers contenus par un paquet **RPM** sont dans une archive compressée, avec un en-tête binaire personnalisé contenant des informations utiles sur l'emballage et son contenu, vous permettant de chercher des paquets un par un, rapidement et facilement.

System Verification

Une autre fonctionnalité puissante de **RPM** est sa capacité de vérifier les paquets. RPM vous permet de vérifier que les fichiers installés sur le système sont les mêmes que ceux qui sont fournis par un paquet donné. Si une incohérence est détectée, **RPM** vous informe, et vous pouvez réinstaller le paquet si nécessaire. Les fichiers de configuration que vous avez modifiés sont préservés lors de la réinstallation.

Pristine Sources

Un objectif de conception crucial consistait à permettre l'utilisation de sources *intactes* de logiciels, distribuées par les auteurs d'origine du logiciel. Avec **RPM**, vous avez les sources intactes, ainsi que tous les correctifs qui ont été utilisés, en plus des instructions de build. Il s'agit d'un avantage important pour plusieurs raisons. Par exemple, si une nouvelle version de programme est lancée, vous n'avez pas forcément besoin de tout recommencer du début pour le compiler. Vous pouvez regarder le correctif pour voir ce que vous *pourriez* faire. Tous les défauts compilés et toutes les modifications qui ont été faites pour que le logiciel puisse compiler correctement sont bien visibles avec cette technique.

Le but de conserver les sources pristines ne peut sembler important qu'aux développeurs, mais cela amène à de meilleurs résultats de qualité de logiciels pour les utilisateurs finaux.

A.2. UTILISATION DE RPM

RPM has five basic modes of operation (excluding package building): installing, uninstalling, upgrading, querying, and verifying. This section contains an overview of each mode. For complete details and options, try **`rpm --help`** or see rpm(8). Also, see Section A.5, « Ressources supplémentaires » for more information on **RPM**.

A.2.1. Installation et mise à niveau des paquets

Les paquets **RPM** ont des noms de fichier de la forme suivante :

```
package_name-version-release-operating_system-CPU_architecture.rpm
```

–

Ainsi, le nom de fichier **tree-1.6.0-10.el7.x86_64.rpm** inclut le nom de paquet (**tree**), la version (**1.6.0**), la distribution (**10**), la version majeure de système d'exploitation (**el7**) et l'architecture de CPU (**x86_64**).

IMPORTANT

Lorsque vous installez un paquet, veillez à ce qu'il soit compatible avec votre système d'exploitation et avec l'architecture de processeur. Ceci peut être normalement déterminé en vérifiant le nom du paquet. Par exemple, le nom de fichier d'un paquet **RPM** compilé pour les architectures d'ordinateurs AMD64/Intel 64 se termine par **x86_64.rpm**.

L'option **-U** (ou **--upgrade**) possède deux fonctions, elle peut être utilisée pour :

- mettre un programme existant à jour sur le sytème d'une nouvelle version, ou
- installer un paquet si une ancienne version n'est pas encore installée.

La commande **rpm -U *package.rpm*** est soit capable, soit de *mettre à niveau* ou d'*installer*, selon la présence d'une ancienne version de *package.rpm* sur le système.

Assumons que le paquet **tree-1.6.0-10.el7.x86_64.rpm** soit dans le répertoire actif, connectez-vous en tant qu'utilisateur **root** et tapez la commande suivante à l'invite du shell pour mettre à niveau ou installer le paquet tree :

```
~]# rpm -Uvh tree-1.6.0-10.el7.x86_64.rpm
```

Les options **-v** et **-h** (qui sont combinées à **-U**) amènent le **rpm** à afficher des sorties plus détaillées et une jauge de progression utilisant des signes de hachage. Si la mise à niveau ou l'installation réussissent, la sortie suivante s'affichera :

```
Preparing...                ################################# [100%]
Updating / installing...
   1:tree-1.6.0-10.el7      ################################# [100%]
```

AVERTISSEMENT

Le **rpm**fournit deux options différentes pour installer les paquets : l'option **-U**, qui est historiquement utilisée pour les mise à niveau (de l'anglais *upgrade*), et l'option **-i**, qui est historiquement utilisée pour les installations (de l'anglais *install*). Comme l'option **-U** inclut à la fois les fonctions d'installation et de mise à niveau, l'utilisation de la commande **rpm -Uvh** pour tous les paquets, **exceptés** les paquets de noyau, est conseillée.

Vous devez toujours utiliser l'option **-i** pour *installer* un nouveau paquet de noyau au lieu de le mettre à niveau. C'est parce que l'option **-U** de mise à jour d'un paquet de noyau supprime le paquet de noyau précédent (plus ancien), ce qui pourrait rendre le système incapable de démarrer s'il y a un problème avec le nouveau noyau. Par conséquent, utiliser la commande **rpm -i *kernel_package*** pour installer un nouveau noyau *sans avoir à remplacer tous les paquets de noyau plus anciens*. Pour plus d'informations sur l'installation des paquets de noyau, voir Chapitre 25, *Mettre à niveau le noyau manuellement*.

La signature d'un paquet est vérifiée automatiquement lors de l'installation ou de la mise à niveau d'un paquet. La signature confirme que le paquet a été signé par un utilisateur autorisé. Si la vérification de la signature échoue, un message d'erreur apparaîtra à l'écran :

Si vous ne possédez pas la clé qui convient intallée pour vérifier la signature, le message contiendra le mot **NOKEY**:

```
warning: tree-1.6.0-10.el7.x86_64.rpm: Header V3 RSA/SHA256 Signature, key
ID 431d51: NOKEY
```

Voir Section A.3.2, « Vérification des signatures de paquets » pour obtenir plus d'informations sur la vérification des signatures de paquets.

A.2.1.1. Remplacer les paquets déjà installés

Si un paquet de même nom et version est déjà installé, la sortie suivante s'affichera :

```
Preparing...                ########################################### 
[100%]
        package tree-1.6.0-10.el7.x86_64 is already installed
```

Pour installer le paquet quand même, utiliser l'option **--replacepkgs**, qui indique au **RPM** d'ignorer l'erreur :

```
~]# rpm -Uvh --replacepkgs tree-1.6.0-10.el7.x86_64.rpm
```

Cette option est utile si les fichers du paquet installés sont été supprimés ou si vous souhaitez que les fichiers de configuration d'origine soient installés.

Si vous souhaitez une mise à niveau d'une *ancienne* version d'un paquet (c'est à dire, si une nouvelle version du paquet est déjà installée), le **RPM** vous informe qu'une nouvelle version est déjà installée. Pour forcer le **RPM** à appliquer un passage à une version antérieure, utiliser l'option **--oldpackage** :

```
rpm -Uvh --oldpackage older_package.rpm
```

A.2.1.2. Résolution des conflits de fichiers

Si vous essayez d'installer un paquet qui contient un fichier qui a déjà été installé par un autre paquet, un message de conflit s'affiche. Pour que le **RPM** ignore cette erreur, utilisez l'option **--replacefiles** :

```
rpm -Uvh --replacefiles package.rpm
```

A.2.1.3. Résoudre les dépendances manquantes

Les paquets **RPM** dépendent parfois d'autres packages, ce qui veut dire qu'ils sont dépendants de la bonne installation d'autres paquets. Si vous essayez d'installer un paquet possédant une dépendance non résolue, un message de dépendance ayant échoué s'affichera.

Cherchez le ou les paquet(s) suggérés dans le media d'installation Red Hat Enterprise Linux ou dans l'un des miroirs Red Hat Enterprise Linux actifs, et ajoutez-le à la commande d'installation. Pour déterminer quel paquet contient le fichier requis, utilisez l'option **--whatprovides** :

```
rpm -q --whatprovides "required_file"
```

Si le paquet qui contient *required_file* est dans la base de données **RPM**, le nom du paquet s'affichera.

AVERTISSEMENT

Bien que vous puissiez *forcer* **RPM** à installer un paquet ayant une dépendance non résolue (à l'aide de l'option **--nodeps**), ce n'est *pas* recommandé et cela entraînera généralement la suspension du logiciel installé. L'installation de paquets avec l'option **--nodeps** peut provoquer des troubles de comportements d'applications et des interruptions inattendues. Cela peut également provoquer des problèmes graves de gestion de paquets ou de défaillance du système. Pour ces raisons, tenez compte des avertissements à propos des dépendances manquantes. Le gestionnaire de paquets **Yum** effectue la résolution de dépendance automatique et récupère les dépendances de référentiels en ligne.

A.2.1.4. Préservation des changements dans les fichiers de configuration

Comme **RPM** procède à des mises à niveau intelligentes de paquets ayant des fichiers de configuration, vous verrez sans doute apparaître le message suivant :

```
sauvegarder /etc/configuration_file.conf en tant que
/etc/configuration_file.conf.rpmsave
```

Ce message signifie que les modifications apportées au fichier de configuration ne peuvent pas être *en compatibilité ascendante* avec le nouveau fichier de configuration du paquet, donc **RPM** a dû enregistrer votre fichier d'origine et en a installé un nouveau. Vous devriez étudier les différences entre les fichiers pour les deux configurations et les résoudre dès que possible pour que votre système continue de fonctionner correctement.

Sinon **RPM** peut enregistrer le *nouveau* fichier de configuration du package comme, par exemple, ***configuration_file.conf*.rpmnew** et laisser le fichier de configuration que vous venez de modifier intact. Il vous reste à résoudre tous les conflits entre votre fichier de configuration modifié et le nouveau, en procédant normalement à la fusion des modifications de l'ancien dans le nouveau, à l'aide du programme **diff**, par exemple.

A.2.2. Désinstaller les paquets

La désinstallation d'un paquetage est aussi simple que son installation. Entrez simplement la commande suivante en tant qu'utilisateur **root** à l'invite du shell :

```
rpm -e package
```

NOTE

Notez que la commande s'attend uniquement au *nom* du paquet, et non pas au nom de *fichier* du paquet d'origine. Si vous essayez de désinstaller un package à l'aide de la commande **rpm-e** et que vous fournissez le nom complet du fichier d'origine, vous recevez une erreur de nom de paquet.

Vous pouvez rencontrer des erreurs de dépendance lors de la désinstallation d'un paquet si un autre paquet installé dépend de celui que vous voulez supprimer. Par exemple :

```
~]# rpm -e ghostscript
error: Failed dependencies:
        ghostscript is needed by (installed) ghostscript-cups-9.07-
16.el7.x86_64
        ghostscript is needed by (installed) foomatic-4.0.9-6.el7.x86_64
        libgs.so.9()(64bit) is needed by (installed) libspectre-0.2.7-
4.el7.x86_64
        libijs-0.35.so()(64bit) is needed by (installed) gutenprint-5.2.9-
15.el7.x86_64
        libijs-0.35.so()(64bit) is needed by (installed) cups-filters-
1.0.35-15.el7.x86_64
```

AVERTISSEMENT

Bien que vous puissiez *forcer* le **rpm** à désinstaller un package dont les dépendances ne sont pas résolues (à l'aide de l'option **--nodeps**), ce n'est *pas* recommandé. La suppression de paquets avec l'option **--nodeps** peut provoquer des problèmes de comportement ou des interruptions inattendues de la part des paquets dont les dépendances sont retirées. Il peut également provoquer des problèmes graves de gestion de paquets, voire de défaillance du système. Pour ces raisons, tenez compte des avertissements de dépendances ayant échoué.

A.2.3. Rafraîchissement de paquets

Le rafraîchissement est similaire à la mise à niveau, sauf que seuls les paquets installés sont mis à niveau. Tapez la commande suivante à une invite du shell en tant qu'utilisateur **root** :

```
rpm -Fvh package.rpm
```

L'option **-F** (ou **--freshen**) compare les versions des paquets nommés sur la ligne de commande par rapport aux versions des paquets qui sont déjà installés sur le système. Lorsqu'une nouvelle version d'un paquet déjà installé est traitée par l'option **--freshen**, elle est mis à niveau à une version plus récente. Cependant, l'option **--freshen** n'installe pas un paquet si aucun paquet précédemment installée du même nom existe déjà. Cela diffère de la mise à niveau régulière, car une mise à jour installe tous les paquets spécifiés indépendamment de savoir si oui ou non les anciennes versions des paquets sont déjà installées.

Comment rafraîchir des travaux de paquets simples ou de groupes de paquets. Par exemple, l'action rafraîchir peut aider si vous téléchargez un grand nombre de paquets différents, et que vous souhaitez uniquement mettre à jour les paquets déjà installés sur le système. Dans ce cas, exécutez la commande suivante avec l'expression globale ***.rpm** :

```
~]# rpm -Fvh *.rpm
```

RPM met automatiquement à niveau les paquets déjà installés uniquement.

A.2.4. Recherche de paquets

La base de données de **RPM** stocke des informations sur tous les paquets **RPM** installés sur votre système. C'est stocké dans le répertoire **/var/lib/rpm/** et est utilisé pour beaucoup de choses, y compris pour savoir quels paquets sont installés, quelle est la version de chaque paquet, et pour le calcul des modifications apportées aux fichiers en paquets depuis leur installation. Pour interroger cette base de données, utilisez la commande **rpm** avec l'option **-q** (ou **--query**) :

```
rpm -q package_name
```

Cette commande affiche le nom du paquet, sa version et le numéro de publication du paquet installé *package_name*. Exemple :

```
~]$ rpm -q tree
tree-1.6.0-10.el7.x86_64
```

-

See the **Package Selection Options** subheading in the rpm(8) manual page for a list of options that can be used to further refine or qualify your query. Use options listed below the **Package Query Options** subheading to specify what information to display about the queried packages.

A.2.5. Vérification des paquets

Vérifier un paquet revient à comparer les informations des fichiers sur le système installé à partir d'un paquet ayant les mêmes informations que celles du paquet d'origine. Entre autres paramètres, vérifier un paquet compare la taille du fichier, MD5 sum, les permissions, le type, le propriétaire et le groupe de chaque fichier

Utiliser la commande **rpm** avec l'option **-V** (or **--verify**) pour vérifier les paquets. Par exemple :

```
~]$ rpm -V tree
```

See the **Package Selection Options** subheading in the rpm(8) manual page for a list of options that can be used to further refine or qualify your query. Use options listed below the **Verify Options** subheading to specify what characteristics to verify in the queried packages.

Si tout est vérifié correctement, il n'y aura aucune sortie. S'il y a des différences, elles s'afficheront. La sortie consiste en lignes qui ressemblent à ceci :

```
~]# rpm -V abrt
S.5....T.  c /etc/abrt/abrt.conf
.M.......    /var/spool/abrt-upload
```

Le format de sortie de string consiste en neuf caractères suivis par un marqueur d'attribut en option et du nom du fichier traité.

Les neuf premiers caractères sont les résultats de tests effectués sur le fichier. Chaque test est une comparaison d'un attribut du fichier par rapport à la valeur de l'attribut tel qu'il est enregistré dans la base de données **RPM**. Un point unique (**.**) signifie que le test a pu être passé, et le caractère point d'interrogation (**?**) signifie que le test ne peut pas être effectué. Le tableau suivant répertorie les symboles qui indiquent certaines incohérences :

Tableau A.1. Symboles de vérification RPM

Symbole	Description
S	la taille des fichiers diffère
M	le mode diffère (inclut les permissions et le type de fichier)
5	digest (anciennement MD5 sum) diffère
D	non correspondance du numéro de version mineur/majeur de la machine
L	readLink(2) path mismatch
U	appartenance utilisateur diffère

Symbole	Description
G	appartenance groupe diffère
T	mtime diffère
P	les capacités diffèrent

Le marqueur d'attribut, si présent, décrit le but du fichier donné. Le tableau suivant dresse une liste des marqueurs d'attributs disponibles :

Tableau A.2. Symboles de vérification RPM

Marqueurs	Description
c	fichier de configuration
d	fichier de documentation
l	fichier de licence
r	fichier readme

Si vous voyez un résultat affiché, essayez de déterminer s'il est préférable de supprimer, de réinstaller le paquet, ou de résoudre le problème autrement.

A.3. RECHERCHE ET VÉRIFICATION DE PAQUETS RPM

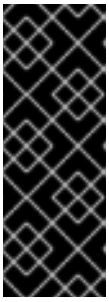
Avant d'utiliser des paquets **RPM**, vous devez savoir où les trouver et être à même de vérifier si vous pouvez leur faire confiance.

A.3.1. Recherche de paquets RPM

Bien qu'il existe de nombreux référentiels **RPM** sur Internet, pour des raisons de compatibilité et de sécurité, vous devriez envisager d'installer uniquement les paquets de RPM officiels fournis par Red Hat. Ce qui suit est une liste de sources de paquets **RPM** :

- Official Red Hat Enterprise Linux installation media.
- Official **RPM** repositories provided with the **Yum** package manager. See Chapitre 8, *Yum* for details on how to use the official Red Hat Enterprise Linux package repositories.
- La page Red Hat Errata se trouve sur le Portail Client à l'adresse suivante https://rhn.redhat.com/rhn/errata/RelevantErrata.do.
- Extra Packages for Enterprise Linux (EPEL) is a community effort to provide a repository with high-quality add-on packages for Red Hat Enterprise Linux. See http://fedoraproject.org/wiki/EPEL for details on EPEL **RPM** packages.

- De façon non officielle, les référentiels de tierce partie non affiliés à Red Hat fournissent également des paquets RPM.



IMPORTANT

Quand on examine les référentiels de tierce partie qui puissent être utilisés avec votre système Red Hat Enterprise Linux, observer attentivement le site web du référentiel en ce qui concerne la compatibilité de paquet avant d'ajouter le référentiel comme paquet source. D'autres référentiels de paquets peuvent offrir des versions différentes et incompatibles du même logiciel, y compris des paquets déjà inclus dans les référentiels Red Hat Enterprise Linux.

A.3.2. Vérification des signatures de paquets

Les paquets **RPM** peuvent être signés par **GNU Privacy Guard** (**GPG**), pour vous assurer que les paquets téléchargés sont dignes de confiance. **GPG** est un outil de communication sécurisé. Avec **GPG**, vous pouvez authentifier la validité des documents et encoder ou décoder des données.

Pour vérifier qu'un package n'est pas encore corrompu ou altéré, vérifier sa signature **GPG** en utilisant la commande **rpmkeys** avec l'option **-K** (ou **--checksig**) :

```
rpmkeys -K package.rpm
```

Notez que le gestionnaire de paquets **Yum** effectue la vérification automatique des signatures **GPG** lors des installations et des mises à niveau.

GPG est installé par défaut, accompagné d'un ensemble de clés de Red Hat qui servent à vérifier les paquets. Pour importer des clés supplémentaires pour une utilisation **RPM**, voir Section A.3.2.1, « Import de clés GPG ».

A.3.2.1. Import de clés GPG

Pour vérifier les paquets de Red Hat, une clé **GPG** de Red Hat doit être installée. Un jeu de clés de base est installé par défaut. Pour afficher une liste des clés installées, exécutez la commande suivante à l'invite du shell :

```
~]$ rpm -qa gpg-pubkey*
```

Pour afficher des détails sur une clé spécifique, utilisez **rpm -qi** suivie de la sortie de la commande précédente. Par exemple :

```
~]$ rpm -qi gpg-pubkey-fd431d51-4ae0493b
```

Utiliser la commande **rpmkeys** avec l'option **--import** pour installer une nouvelle clé à utiliser avec **RPM**. L'emplacement par défaut pour stocker les clés GPG **RPM** est le répertoire **/etc/pki/rpm-gpg/**. Pour importer de nouvelles clés, utiliser une commande semblable à celle-ci en tant qu'utilisateur **root** :

```
~]# rpmkeys --import /etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
```

Voir l'article Product Signing (GPG) Keys sur le portail clients de Red Hat pour plus d'informations sur les pratiques de signature de paquets de Red Hat.

A.4. EXEMPLES COMMUNS DE L'UTILISATION DE RPM

RPM est un outil utile pour gérer votre système, diagnostiquer et résoudre les problèmes. Voir les exemples suivants pour avoir un aperçu de quelques-unes des options les plus utilisées.

- Pour vérifier dans tout votre système quels sont les fichiers manquants, lancez la commande suivante en tant qu'utilisateur **root**:

  ```
  rpm -Va
  ```

  Si certains fichiers ont disparu ou semblent avoir été corrompus, il vaut mieux réinstaller les paquets qui conviennent.

- Pour savoir à quel paquet un fichier appartient, saisir :

  ```
  rpm -qf file
  ```

- Pour vérifier l'appartenance d'un fichier particulier à un paquet, saisir la commande suivante en tant qu'utilisateur **root** :

  ```
  rpm -Vf file
  ```

- Pour trouver l'emplacement de fichiers de documentation qui font partie d'un paquet auquel un fichier appartient, saisir :

  ```
  rpm -qdf file
  ```

- Pour trouver des informations sur un fichier de paquet (non installé), utiliser la commande suivante :

  ```
  rpm -qip package.rpm
  ```

- Pour afficher des fichiers contenus dans un paquet, utiliser :

  ```
  rpm -qlp package.rpm
  ```

See the rpm(8) manual page for more options.

A.5. RESSOURCES SUPPLÉMENTAIRES

RPM est un programme utilitaire très complexe, doté de nombreuses options et méthodes de recherche, d'installation, de mise à jour et de désinstallation de paquets. Consultez les sources d'informations suivantes pour en savoir plus sur **RPM**.

Documentation installée

- **rpm --help** — cette commande afficher une référence rapide de paramètres **RPM**.
- rpm(8) — The **RPM** manual page offers an overview of all available **RPM** parameters.

Documentation en ligne

- Red Hat Enterprise Linux 7 Security Guide — Le guide de sécurité *Security Guide* de Red Hat Enterprise Linux 7 documente la façon de conserver votre système à jour grâce au gestionnaire de paquets **Yum** et comment vérifier et installer les paquets téléchargés.
- Le site web **RPM** — http://www.rpm.org/
- La liste de diffusion **RPM** — http://lists.rpm.org/mailman/listinfo/rpm-list

Voir aussi

- Chapitre 8, *Yum* décrit comment utiliser le gestionnaire de paquets **Yum** pour chercher, installer, mettre à jour et installer des paquets par la ligne de commande.

ANNEXE B. HISTORIQUE DES VERSIONS

Version 0.14-8.4 **Thu May 31 2018** **Red Hat**
Fichiers de traduction synchronisés avec les sources XML 0.14-8

Version 0.14-8.3 **Tue Jul 4 2017** **Terry Chuang**
Fichiers de traduction synchronisés avec les sources XML 0.14-8

Version 0.14-8.2 **Mon Apr 3 2017** **Terry Chuang**
Fichiers de traduction synchronisés avec les sources XML 0.14-8

Version 0.14-8.1 **Mon Feb 6 2017** **Terry Chuang**
Fichiers de traduction synchronisés avec les sources XML 0.14-8

Version 0.14-8 **Mon Nov 3 2016** **Maxim Svistunov**
Version pour la distribution GA 7.3.

Version 0.14-7 **Mon Jun 20 2016** **Maxim Svistunov**
Ajout de *Relax-and-Recover (ReaR)*; légères améliorations.

Version 0.14-6 **Thu Mar 10 2016** **Maxim Svistunov**
Corrections mineures et mises à jour.

Version 0.14-5 **Thu Jan 21 2016** **Lenka Špačková**
Mises à jour factuelles mineures.

Version 0.14-3 **Wed Nov 11 2015** **Jana Heves**
Distribution 7.2 GA.

Version 0.14-1 **Mon Nov 9 2015** **Jana Heves**
Changements mineurs, lien au cours de formation RH ajouté.

Version 0.14-0.3 **Fri Apr 3 2015** **Stephen Wadeley**
Ajouté *Enregistrer le système et gérer les abonnements* , et *Accéder au support en utilisant l'outil de prise en charge Red Hat Support Tool*, mise à jour de *Afficher et gérer les fichiers journaux*.

Version 0.13-2 **Tue Feb 24 2015** **Stephen Wadeley**
Version pour la sortie 7.1 GA.

Version 0.12-0.6 **Tue Nov 18 2014** **Stephen Wadeley**
Amélioration de *TigerVNC*.

Version 0.12-0.4 **Mon Nov 10 2014** **Stephen Wadeley**
Amélioration de *Yum*, *Gérer les services avec systemd*, *OpenLDAP*, *Afficher et gérer des fichiers journaux* , *OProfile*, et *Utiliser le chargeur de démarrage GRUB 2*.

Version 0.12-0 **Tue 19 Aug 2014** **Stephen Wadeley**
Sortie du Guide de l'administrateur systèmes Red Hat Enterprise Linux 7.0 GA .

INDEX

Symboles

A

B

C

D

E

F

G

L

M

N

O

P

R

S

T

U

V

W

Y