

Rejoignez les explorateurs, les bâtisseurs et tous ceux qui ont le courage de proposer des solutions nouvelles à des problèmes anciens. Dans le domaine de l'open source, l'innovation dépend entièrement des personnes qui y travaillent.



Red Hat Training and Certification

MANUEL D'EXERCICES (ROLE)

Red Hat Enterprise Linux 8.0 RH124

RED HAT SYSTEM ADMINISTRATION I

Édition 1



RED HAT SYSTEM ADMINISTRATION I



Red Hat Enterprise Linux 8.0 RH124
Red Hat System Administration I
Édition 120190531
Date de publication 20190531

Auteurs: Fiona Allen, Marc Kesler, Saumik Paul, Snehangshu Karmakar,

Victor Costea

Éditeur: Steven Bonneville, Ralph Rodriguez, David Sacco, Heather Charles,
David O'Brien, Seth Kenlon

Copyright © 2019 Red Hat, Inc.

The contents of this course and all its modules and related materials, including handouts to audience members, are
Copyright © 2019 Red Hat, Inc.

No part of this publication may be stored in a retrieval system, transmitted or reproduced in any way, including, but
not limited to, photocopy, photograph, magnetic, electronic or other record, without the prior written permission of
Red Hat, Inc.

This instructional program, including all material provided herein, is supplied without any guarantees from Red Hat,
Inc. Red Hat, Inc. assumes no liability for damages or legal action arising from the use or misuse of contents or details
contained herein.

If you believe Red Hat training materials are being used, copied, or otherwise improperly distributed please e-mail
training@redhat.com or phone toll-free (USA) +1 (866) 626-2994 or +1 (919) 754-3700.

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

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

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

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

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

All other trademarks are the property of their respective owners.

Collaborateurs : Achyut Madhusudan, Rudolf Kastl, Rob Locke, Dallas Spohn, Michael Phillips

Conventions de la documentation	ix
Introduction	xi
Red Hat System Administration I	xi
Organisation de l'environnement de formation	xii
Internationalisation	xvi
1. Prise en main de Red Hat Enterprise Linux	1
Qu'est-ce que Linux ?	2
Quiz: Prise en main de Red Hat Enterprise Linux	10
Résumé	12
2. Accès à la ligne de commande	13
Accès à la ligne de commande	14
Quiz: Accès à la ligne de commande	20
Accès à la ligne de commande depuis le bureau	24
Exercice guidé: Accès à la ligne de commande depuis le bureau	30
Exécution de commandes à l'aide du shell bash	32
Quiz: Exécution de commandes à l'aide du shell bash	39
Open Lab: Accès à la ligne de commande	43
Résumé	49
3. Gestion de fichiers à partir de la ligne de commande	51
Description des concepts de hiérarchie du système de fichiers Linux	52
Quiz: Description des concepts de hiérarchie du système de fichiers Linux	55
Spécification des fichiers par nom	59
Quiz: Spécification des fichiers par nom	65
Gestion des fichiers à l'aide des outils de ligne de commande	69
Exercice guidé: Gestion des fichiers à l'aide des outils de ligne de commande	75
Création de liens entre des fichiers	81
Exercice guidé: Création de liens entre des fichiers	85
Correspondance des noms de fichiers à l'aide des extensions par le shell	87
Quiz: Correspondance des noms de fichiers à l'aide des extensions par le shell	92
Open Lab: Gestion de fichiers à partir de la ligne de commande	96
Résumé	106
4. Aide dans Red Hat Enterprise Linux	107
Lecture des pages de manuel	108
Exercice guidé: Lecture des pages de manuel	112
Lecture de la documentation Info	116
Exercice guidé: Lecture de la documentation Info	120
Open Lab: Aide dans Red Hat Enterprise Linux	123
Résumé	130
5. Crédit, affichage et modification de fichiers texte	131
Redirection de la sortie vers un fichier ou un programme	132
Quiz: Redirection de la sortie vers un fichier ou un programme	138
Modification de fichiers texte à partir de l'invite du shell	142
Exercice guidé: Modification de fichiers texte à partir de l'invite du shell	146
Modification de l'environnement shell	148
Exercice guidé: Modification de l'environnement shell	154
Open Lab: Crédit, affichage et modification de fichiers texte	157
Résumé	165
6. Gestion des utilisateurs et des groupes locaux	167
Description des concepts d'utilisateur et de groupe	168
Quiz: Description des concepts d'utilisateur et de groupe	172
Accès en tant que super utilisateur	176

Exercice guidé: Accès en tant que super utilisateur	182
Gestion des comptes d'utilisateur locaux	187
Exercice guidé: Gestion des comptes d'utilisateur locaux	191
Gestion des comptes de groupes locaux	194
Exercice guidé: Gestion des comptes de groupes locaux	197
Gestion des mots de passe des utilisateurs	200
Exercice guidé: Gestion des mots de passe des utilisateurs	204
Open Lab: Gestion des utilisateurs et des groupes locaux	208
Résumé	213
7. Contrôle de l'accès aux fichiers	215
Interprétation des permissions du système de fichiers Linux	216
Quiz: Interprétation des permissions du système de fichiers Linux	221
Gestion des permissions du système de fichiers à partir de la ligne de commande	225
Exercice guidé: Gestion des permissions du système de fichiers à partir de la ligne de commande	229
Gestion de l'accès aux fichiers et des permissions par défaut	233
Exercice guidé: Gestion de l'accès aux fichiers et des permissions par défaut	238
Open Lab: Contrôle de l'accès aux fichiers	243
Résumé	250
8. Contrôle et gestion des processus Linux	251
Création d'une liste de processus	252
Quiz: Création d'une liste de processus	258
Contrôle des tâches	260
Exercice guidé: Contrôle des tâches	263
Suppression de processus	268
Exercice guidé: Suppression de processus	274
Contrôle de l'activité des processus	279
Exercice guidé: Contrôle de l'activité des processus	283
Open Lab: Contrôle et gestion des processus Linux	288
Résumé	299
9. Contrôle des services et des démons	301
Identification des processus système démarrés automatiquement	302
Exercice guidé: Identification des processus système démarrés automatiquement	308
Contrôle des services du système	312
Exercice guidé: Contrôle des services du système	316
Open Lab: Contrôle des services et des démons	320
Résumé	325
10. Configuration et sécurisation de SSH	327
Accès en ligne de commande distante via SSH	328
Exercice guidé: Accès à la ligne de commande distante	332
Configuration de l'authentification par clé SSH	336
Exercice guidé: Configuration de l'authentification par clé SSH	341
Personnalisation de la configuration du service OpenSSH	347
Exercice guidé: Personnalisation de la configuration du service OpenSSH	349
Open Lab: Configuration et sécurisation de SSH	355
Résumé	362
11. Analyse et stockage des journaux	363
Description de l'architecture du journal du système	364
Quiz: Description de l'architecture du journal du système	366
Examen des fichiers Syslog	370
Exercice guidé: Examen des fichiers Syslog	374
Analyse des entrées du journal système	377

Exercice guidé: Analyse des entrées du journal système	383
Conservation du journal de système	387
Exercice guidé: Conservation du journal de système	390
Gestion précise de l'heure	393
Exercice guidé: Gestion précise de l'heure	397
Open Lab: Analyse et stockage des journaux	402
Résumé	408
12. Gestion de réseaux	409
Description des concepts réseau	410
Quiz: Description des concepts réseau	422
Validation de la configuration réseau	426
Exercice guidé: Validation de la configuration réseau	432
Configuration de la mise en réseau à partir de la ligne de commande	435
Exercice guidé: Configuration de la mise en réseau à partir de la ligne de commande	441
Modification des fichiers de configuration réseau	447
Exercice guidé: Modification des fichiers de configuration réseau	451
Configuration de noms d'hôte et résolution de noms	456
Exercice guidé: Configuration de noms d'hôte et résolution de noms	459
Open Lab: Gestion de réseaux	463
Résumé	468
13. Archivage et transfert de fichiers	469
Gestion des archives tar compressées	470
Exercice guidé: Gestion des archives tar compressées	477
Transfert sécurisé de fichiers entre systèmes	480
Exercice guidé: Transfert sécurisé de fichiers entre systèmes	482
Synchronisation de fichiers sécurisée entre des systèmes	485
Exercice guidé: Synchronisation de fichiers sécurisée entre des systèmes	489
Open Lab: Archivage et transfert de fichiers	491
Résumé	497
14. Installation et mise à jour de paquetages logiciels	499
Enregistrement de systèmes pour le support Red Hat	500
Quiz: Enregistrement de systèmes pour le support Red Hat	504
Explication et analyse des paquetages logiciels RPM	506
Exercice guidé: Explication et analyse des paquetages logiciels RPM	512
Installation et mise à jour de paquetages logiciels avec Yum	515
Exercice guidé: Installation et mise à jour de paquetages logiciels avec Yum	522
Activation de référentiels logiciels Yum	527
Exercice guidé: Activation de référentiels logiciels Yum	531
Gestion des flux de modules de paquetages	535
Exercice guidé: Gestion des flux de modules de paquetages	542
Open Lab: Installation et mise à jour de paquetages logiciels	547
Résumé	554
15. Accès aux systèmes de fichiers Linux	555
Identification des systèmes de fichiers et des périphériques	556
Quiz: Identification des systèmes de fichiers et des périphériques	560
Montage et démontage de systèmes de fichiers	562
Exercice guidé: Montage et démontage de systèmes de fichiers	566
Localisation de fichiers dans le système	569
Exercice guidé: Localisation de fichiers dans le système	576
Open Lab: Accès aux systèmes de fichiers Linux	579
Résumé	584
16. Analyser les serveurs et obtenir une assistance	585

Analyse et gestion de serveurs distants	586
Exercice guidé: Analyse et gestion de serveurs distants	600
Obtenir de l'aide auprès du portail client Red Hat	605
Exercice guidé: Obtenir de l'aide auprès du portail client Red Hat	615
Détection et résolution des problèmes avec Red Hat Insights	617
Quiz: Détection et résolution des problèmes avec Red Hat Insights	626
Résumé	628
17. Révision complète	629
Révision complète	630
Open Lab: Gestion de fichiers à partir de la ligne de commande	635
Open Lab: Gestion des utilisateurs et des groupes, des autorisations et des processus ..	643
Open Lab: Configuration et gestion d'un serveur	650
Open Lab: Gestion de réseaux	658
Open Lab: Montage de systèmes de fichiers et recherche de fichiers	665

CONVENTIONS DE LA DOCUMENTATION



RÉFÉRENCES

Les « références » indiquent où trouver de la documentation externe se rapportant à un sujet.



NOTE

Une « remarque » est un conseil, un raccourci ou une approche alternative pour la tâche considérée. Le fait d'ignorer une remarque ne devrait pas entraîner de conséquences négatives, mais vous pourriez passer à côté d'une astuce qui vous simplifierait la vie.



IMPORTANT

Les cadres « Important » détaillent des éléments qui pourraient aisément être négligés : des changements de configuration qui ne s'appliquent qu'à la session en cours ou des services qui doivent être redémarrés pour qu'une mise à jour soit appliquée. Ignorer un cadre « Important » ne vous fera perdre aucune donnée, mais cela pourrait être source de frustration et d'irritation.



MISE EN GARDE

Un « avertissement » ne doit pas être ignoré. Le fait d'ignorer un avertissement risque fortement d'entraîner une perte de données.

INTRODUCTION

RED HAT SYSTEM ADMINISTRATION I

Red Hat System Administration I (RH124) est conçu pour les professionnels de l'informatique sans expérience préalable dans l'administration système sous Linux. Ce cours permet aux stagiaires d'acquérir des « compétences de survie » en administration Linux grâce à une analyse des principales tâches administratives. Le cours *Red Hat System Administration I* fournit également une base pour les stagiaires qui envisagent de devenir administrateurs de systèmes Linux à temps complet en présentant les concepts clés de ligne de commande et les outils à usage professionnel. Ces concepts sont davantage développés dans le cours suivant, *Red Hat System Administration II* (RH134).

OBJECTIFS DU COURS

- Acquérir des compétences suffisantes pour effectuer les principales tâches d'administration système sous Red Hat Enterprise Linux.
- Commencer à acquérir les compétences que doit avoir un administrateur système Red Hat Enterprise Linux certifié RHCSA.

PUBLIC

- Les professionnels de l'informatique d'un large éventail de disciplines qui doivent effectuer des tâches d'administration Linux essentielles, y compris l'installation, la mise en œuvre de la connectivité réseau, la gestion du stockage physique et l'administration de base de la sécurité.

CONDITIONS PRÉALABLES

- Il n'existe aucune condition préalable formelle pour ce cours ; toutefois, une expérience préalable dans l'administration système sous d'autres systèmes d'exploitation est fortement conseillée.

ORGANISATION DE L'ENVIRONNEMENT DE FORMATION

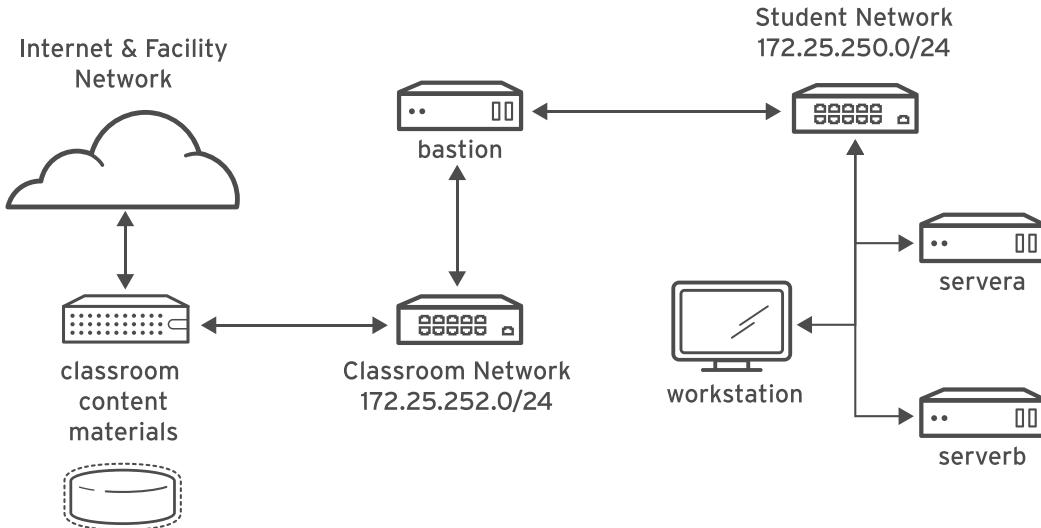


Figure 0.1: Environnement de formation

Dans ce cours, le système informatique principal utilisé pour les travaux pratiques est **workstation**. Deux autres machines sont également utilisées par les stagiaires pour ces activités : **servera** et **serverb**. Ces trois systèmes se trouvent dans le domaine DNS `lab.example.com`.

Tous les systèmes informatiques des stagiaires possèdent un compte d'utilisateur standard, **student**, protégé par le mot de passe **student**. Le mot de passe **root** est **redhat** sur tous les systèmes des stagiaires.

Machines de la salle de classe

NOM DE LA MACHINE	ADRESSES IP	RÔLE
bastion.lab.example.com	172.25.250.254	Système passerelle pour connecter le réseau privé des stagiaires au serveur de la salle de classe (doit toujours être en cours d'exécution)
workstation.lab.example.com	172.25.250.9	Poste de travail graphique utilisé pour l'administration du système
servera.lab.example.com	172.25.250.10	Premier serveur
serverb.lab.example.com	172.25.250.11	Second serveur

La fonction principale de **bastion** est de servir de routeur entre le réseau sur lequel sont connectées les machines des stagiaires et le réseau de la salle de classe. Si le poste de travail

Introduction

bastion est arrêté, les autres machines de stagiaires peuvent uniquement accéder aux systèmes qui se trouvent sur le réseau des stagiaires.

Plusieurs systèmes dans la salle de classe proposent des services d'assistance. Deux serveurs, `content.example.com` et `materials.example.com`, hébergent les logiciels et les supports d'atelier utilisés pour les activités pratiques. Les informations relatives à l'utilisation de ces serveurs sont fournies dans les instructions de ces activités, notamment par la machine virtuelle `classroom.example.com`. Les machines `classroom` et `bastion` doivent toujours être en cours d'exécution pour une utilisation correcte de l'environnement d'atelier.



NOTE

Lorsque vous vous connectez à `servera` ou `serverb`, il se peut que vous voyiez un message concernant l'activation de `cockpit`. Le message peut être ignoré.

```
[student@workstation ~]$ ssh student@serverb
Warning: Permanently added 'serverb,172.25.250.11' (ECDSA) to the list of
known hosts.
Activate the web console with: systemctl enable --now cockpit.socket

[student@serverb ~]$
```

CONTRÔLE DE VOS SYSTÈMES

Les stagiaires se voient attribuer des ordinateurs distants dans une salle de classe de formation en ligne Red Hat. L'accès s'effectue par le biais d'une application Web hébergée à l'adresse suivante : `rol.redhat.com` [`http://rol.redhat.com`]. Pour se connecter à ce site, les stagiaires doivent utiliser leurs informations d'identification du Portail client Red Hat.

Contrôle des machines virtuelles

Les machines virtuelles de votre environnement de formation sont contrôlées sur une Page Web. L'état de chaque machine virtuelle de la salle de classe est affiché sur la page de l'onglet Online Lab.

États de la machine

ÉTAT DE LA MACHINE VIRTUELLE	DESCRIPTION
STARTING	La machine virtuelle est en cours de démarrage.
STARTED	La machine virtuelle est en cours d'exécution et disponible (ou, pendant le démarrage, le sera bientôt.)
STOPPING	La machine virtuelle est en cours d'arrêt.
STOPPED	La machine virtuelle est complètement arrêtée. Au démarrage, la machine virtuelle affiche le même état que lors de son arrêt (le disque est préservé).
PUBLISHING	La création initiale de la machine virtuelle est en cours.

ÉTAT DE LA MACHINE VIRTUELLE	DESCRIPTION
WAITING_TO_START	La machine virtuelle est en attente du démarrage d'autres machines virtuelles.

Selon l'état de la machine, une sélection des actions suivantes est disponible.

Actions de machine/salle de classe

BOUTON OU ACTION	DESCRIPTION
PROVISION LAB	Permet de créer la salle de classe ROL. Crée toutes les machines virtuelles nécessaires pour la salle de classe et les démarre. Cette procédure peut prendre plusieurs minutes.
DELETE LAB	Permet de supprimer la salle de classe ROL. Détruit toutes les machines virtuelles de la salle de classe. Attention : tous les travaux générés sur les disques seront perdus.
START LAB	Permet de démarrer toutes les machines virtuelles de la salle de classe.
SHUTDOWN LAB	Permet d'arrêter toutes les machines virtuelles de la salle de classe.
OPEN CONSOLE	Permet d'ouvrir un nouvel onglet dans le navigateur et de se connecter à la console de la machine virtuelle. Les stagiaires peuvent se connecter directement à la machine virtuelle et exécuter des commandes. Dans la plupart des cas, les stagiaires doivent se connecter à la machine virtuelle de la station de travail et utiliser ssh pour se connecter aux autres machines virtuelles.
ACTION → Start	Permet de démarrer (allumer) la machine virtuelle.
ACTION → Shutdown	Permet d'éteindre correctement la machine virtuelle, en conservant le contenu sur son disque.
ACTION → Power Off	Force l'arrêt de la machine virtuelle, en conservant le contenu du disque. Cela équivaut à couper l'alimentation d'une machine physique.
ACTION → Reset	Force l'arrêt de la machine virtuelle et réinitialise le disque à son état initial. Attention : tous les travaux générés sur le disque seront perdus.

Au début d'un exercice, si vous êtes invité à réinitialiser un seul nœud de machine virtuelle, cliquez sur ACTION → Reset pour la machine virtuelle concernée.

Au début d'un exercice, si vous êtes invité à réinitialiser l'ensemble des machines virtuelles, cliquez sur ACTION → Reset

Si vous souhaitez que l'environnement de formation retourne à son état d'origine du début du cours, vous pouvez cliquer sur DELETE LAB pour supprimer l'ensemble de l'environnement de

formation. Une fois l'exercice pratique supprimé, vous pouvez cliquer sur PROVISION LAB pour déployer un nouvel ensemble de systèmes de salle de classe.



MISE EN GARDE

L'opération DELETE LAB ne peut pas être annulée. Tous les travaux que vous aurez terminés jusqu'ici dans l'environnement de formation seront perdus.

Minuterie d'arrêt automatique

L'inscription à la formation en ligne Red Hat (ROL) confère aux stagiaires le droit d'utiliser l'ordinateur pendant un temps donné. Afin de les aider à conserver le temps d'utilisation de l'ordinateur qui leur est alloué, une minuterie d'arrêt automatique est associée à la salle de classe ROL. Celle-ci ferme l'environnement de formation à l'expiration du temps prévu.

Pour régler la minuterie, cliquez sur MODIFY afin que la boîte de dialogue New Autostop Time s'affiche. Définissez le nombre d'heures jusqu'à l'arrêt automatique de la classe. Cliquez sur ADJUST TIME pour appliquer cette modification aux paramètres de la minuterie.

INTERNATIONALISATION

SÉLECTION DE LA LANGUE PAR UTILISATEUR

Il se peut que vos utilisateurs veuillent utiliser, pour leur environnement de bureau, une langue différente de celle utilisée par l'ensemble du système. Il se peut aussi qu'ils veuillent utiliser une autre disposition de clavier ou une autre méthode de saisie pour leur compte.

Paramètres linguistiques

Dans l'environnement de bureau GNOME, l'utilisateur peut être invité, lors de sa première connexion, à configurer la langue et la méthode de son choix. Dans le cas contraire, la manière la plus simple pour un utilisateur d'ajuster les réglages de langue et de méthode de saisie est d'utiliser l'application Region & Language.

Vous pouvez démarrer cette application de deux manières. Vous pouvez exécuter la commande **gnome-control-center region** depuis la fenêtre de terminal ou sur la barre du haut, à partir du menu système situé dans le coin droit, sélectionnez le bouton des paramètres (dont l'icône ressemble à un tournevis croisé et une clé) en bas à gauche du menu.

Dans la fenêtre qui s'ouvre, sélectionnez Region & Language. Cliquez sur la case Language et sélectionnez la langue souhaitée dans la liste qui s'affiche. Cela met également à jour le réglage Formats pour qu'il corresponde aux réglages par défaut pour cette langue. Ces modifications entreront en vigueur la prochaine fois que vous vous connectez.

Ces paramètres affectent l'environnement de bureau GNOME et toutes les applications qui y sont lancées, telles que **gnome-terminal**. Toutefois, par défaut, ils ne s'appliquent pas à ce compte si l'accès a été réalisé via une connexion **ssh** à partir d'un système distant ou d'une connexion texte sur une console virtuelle (ex. : **tty5**).



NOTE

Vous pouvez faire en sorte que votre environnement de shell utilise le même paramètre **LANG** que votre environnement graphique, même lorsque vous vous connectez par l'intermédiaire d'une console virtuelle en mode texte ou par **ssh**. Pour ce faire, vous pouvez placer le code suivant ou son équivalent dans votre fichier **~/.bashrc**. Ce code fourni en exemple règle la langue utilisée pour une connexion en mode texte pour qu'elle corresponde à celle configurée pour l'environnement de bureau GNOME :

```
i=$(grep 'Language=' /var/lib/AccountsService/users/${USER} \
| sed 's/Language=/"/')
if [ "$i" != "" ]; then
    export LANG=$i
fi
```

Le japonais, le coréen, le chinois et d'autres langues à jeu de caractères autre que le latin peuvent ne pas s'afficher correctement sur les consoles virtuelles en mode texte.

On peut obliger chaque commande à utiliser une autre langue, en réglant la variable LANG depuis la ligne de commande :

```
[user@host ~]$ LANG=fr_FR.utf8 date  
jeu. avril 25 17:55:01 CET 2019
```

Les commandes suivantes continuent d'utiliser la langue par défaut du système. La commande **locale** peut être utilisée pour déterminer la valeur actuelle de LANG, ainsi que d'autres variables d'environnement connexes.

Paramètres de la méthode de saisie

Dans Red Hat Enterprise Linux 7 ou version ultérieure, GNOME 3 utilise automatiquement le système de sélection de méthode de saisie IBus qui permet de changer facilement et rapidement la disposition du clavier et les méthodes de saisie.

L'application Region & Language peut aussi servir à activer d'autres méthodes de saisie. Dans la fenêtre de l'application Region & Language, le cadre Input Sources présente les méthodes de saisie actuellement disponibles. Par défaut, English (US) peut être la seule méthode disponible. Sélectionnez English (US), puis cliquez sur l'icône du clavier pour afficher la disposition actuelle du clavier.

Pour ajouter une nouvelle méthode de saisie, cliquez sur le bouton + dans le coin inférieur gauche de la fenêtre Input Sources. Une fenêtre Add an Input Source s'ouvre. Sélectionnez votre langue, puis la méthode de saisie ou la disposition de clavier souhaitée.

Lorsque plusieurs méthodes de saisie ont été configurées. L'utilisateur peut passer rapidement de l'une à l'autre en saisissant **Super+Space** (parfois appelé **Windows+Space**). Par ailleurs, un *indicateur d'état* s'affiche dans la barre supérieure de l'environnement GNOME. Celui-ci a deux fonctions : il indique la méthode de saisie active et joue le rôle de menu vous permettant de passer d'une méthode de saisie à l'autre ou de sélectionner les fonctions avancées de méthodes de saisie plus complexes.

Certaines des méthodes sont marquées par des engrenages, qui indiquent qu'elles ont des options de configuration et des possibilités avancées. Par exemple, la méthode de saisie japonaise Japonais (Kana Kanji) permet à l'utilisateur de préparer un texte en caractères latins et d'utiliser les touches **Flèche vers le bas** et **Flèche vers le haut** pour sélectionner les caractères à utiliser.

Les anglophones américains peuvent également trouver cette méthode utile. Pour English (United States) par exemple, la disposition de clavier est English (international avec touches mortes en AltGr), qui considère la touche **AltGr** (ou la touche **Alt**) de droite) sur un clavier PC à 104-105 touches comme une touche de modification « Maj secondaire » et comme touche d'activation des touches mortes pour la saisie des caractères supplémentaires. Le Dvorak et d'autres dispositions sont également proposées.



NOTE

Si vous connaissez le point de code Unicode du caractère, vous pouvez le saisir dans l'environnement de bureau GNOME. Appuyez sur **Ctrl+Maj+U**, suivi du point de code. Après avoir appuyé sur les touches **Ctrl+Maj+U**, un **u** souligné s'affiche pour indiquer que le système attend la saisie du code du caractère.

Par exemple, la lettre minuscule grecque lambda a pour point de code U+03BB et peut être saisie en appuyant sur **Ctrl+Maj+U**, puis **03BB**, et ensuite **Entrée**.

PARAMÈTRES LINGUISTIQUES PAR DÉFAUT POUR L'ENSEMBLE DU SYSTÈME

La langue par défaut du système est configurée sur US English (Anglais des États-Unis), avec le jeu de caractères Unicode UTF-8 (**en_US.utf8**), mais cela peut être changé pendant ou après l'installation.

Depuis la ligne de commande, l'utilisateur **root** peut modifier les paramètres linguistiques à l'échelle du système à l'aide de la commande **localectl**. Si la commande **localectl** est exécutée sans argument, elle affiche les paramètres linguistiques à l'échelle du système.

Pour définir une langue par défaut au niveau du système, exécutez la commande **localectl set-locale LANG=locale**, où *locale* est la valeur appropriée pour la variable d'environnement LANG correspondante décrite dans le tableau « Référence des codes de langue » du présent chapitre. Les changements seront pris en compte lors de la prochaine connexion de l'utilisateur et seront stockés dans le fichier **/etc/locale.conf**.

```
[root@host ~]# localectl set-locale LANG=fr_FR.utf8
```

Dans GNOME, les administrateurs peuvent modifier ce paramètre dans Region & Language en cliquant sur le bouton Login Screen dans le coin supérieur droit de la fenêtre. La modification de la langue (Language) de l'écran de connexion graphique ajustera également le paramètre linguistique pour l'ensemble du système, stocké dans le fichier de configuration **/etc/locale.conf**.



IMPORTANT

Les consoles virtuelles en mode texte telles que **tty4** sont plus limitées en ce qui concerne les polices qu'elles peuvent afficher que les terminaux d'une console virtuelle fonctionnant sous un environnement graphique, ou les pseudoterminal pour les sessions **ssh**. Par exemple, les caractères japonais, coréens et chinois peuvent ne pas s'afficher correctement dans une console virtuelle en mode texte. Pour cette raison, vous devriez envisager d'utiliser l'anglais ou une autre langue avec un jeu de caractères latins pour la langue par défaut pour l'ensemble du système.

De même, les consoles virtuelles en mode texte reconnaissent moins de méthodes de saisie. Ce point est géré séparément de l'environnement graphique du bureau. On peut configurer les paramètres de saisie globaux par l'intermédiaire de **localectl**, à la fois pour les consoles virtuelles en mode texte et pour l'environnement graphique. Voir les pages du manuel **localectl(1)** et **vconsole.conf(5)** pour plus d'informations.

MODULES LINGUISTIQUES

Des paquetages RPM spéciaux appelés *langpacks* installent des paquetages de langue qui prennent en charge des langues spécifiques. Ces langpacks utilisent des dépendances pour installer automatiquement des paquetages RPM supplémentaires contenant des localisations, des dictionnaires et des traductions pour les autres paquetages logiciels de votre système.

Pour lister les langpacks installés et susceptibles d'être installés, utilisez **yum list langpacks-* :**

```
[root@host ~]# yum list langpacks-*  
Updating Subscription Management repositories.  
Updating Subscription Management repositories.  
Installed Packages  
langpacks-en.noarch      1.0-12.el8        @AppStream  
Available Packages  
langpacks-af.noarch       1.0-12.el8        rhel-8-for-x86_64-appstream-rpms  
langpacks-am.noarch       1.0-12.el8        rhel-8-for-x86_64-appstream-rpms  
langpacks-ar.noarch       1.0-12.el8        rhel-8-for-x86_64-appstream-rpms  
langpacks-as.noarch       1.0-12.el8        rhel-8-for-x86_64-appstream-rpms  
langpacks-ast.noarch      1.0-12.el8        rhel-8-for-x86_64-appstream-rpms  
...output omitted...
```

Pour ajouter une prise en charge linguistique, installez le paquetage langpacks approprié. Par exemple, la commande suivante ajoute la prise en charge du français :

```
[root@host ~]# yum install langpacks-fr
```

Utilisez **yum repoquery --what supplements** pour déterminer quels paquetages RPM peuvent être installés par un langpack :

```
[root@host ~]# yum repoquery --what supplements langpacks-fr  
Updating Subscription Management repositories.  
Updating Subscription Management repositories.  
Last metadata expiration check: 0:01:33 ago on Wed 06 Feb 2019 10:47:24 AM CST.  
glibc-langpack-fr-0:2.28-18.el8.x86_64  
gnome-getting-started-docs-fr-0:3.28.2-1.el8.noarch  
hunspell-fr-0:6.2-1.el8.noarch  
hyphen-fr-0:3.0-1.el8.noarch  
libreoffice-langpack-fr-1:6.0.6.1-9.el8.x86_64  
man-pages-fr-0:3.70-16.el8.noarch  
mythes-fr-0:2.3-10.el8.noarch
```



IMPORTANT

Les paquetages langpacks utilisent les *dépendances faibles* RPM afin d'installer des paquetages supplémentaires uniquement lorsque le paquetage principal qui en a besoin est également installé.

Par exemple, lors de l'installation de *langpacks-fr* comme le montrent les exemples précédents, le paquetage *mythes-fr* ne sera installé que si le dictionnaire des synonymes *mythes* est également installé sur le système.

Si *mythes* est ensuite installé sur ce système, le paquetage *mythes-fr* sera également automatiquement installé en raison de la faible dépendance du paquetage *langpacks-fr* déjà installé.



RÉFÉRENCES

Pages du manuel **locale(7)**, **localectl(1)**, **locale.conf(5)**, **vconsole.conf(5)**, **unicode(7)** et **utf-8(7)**

Les conversions entre le nom des présentations X11 de l'environnement graphique de bureau et leur nom dans **localectl** se trouvent dans le fichier **/usr/share/X11/xkb/rules/base.lst**.

RÉFÉRENCE DES CODES DE LANGUE



NOTE

Ce tableau peut ne pas refléter tous les langpacks disponibles sur votre système. Utilisez **yum info langpacks-SUFFIX** pour obtenir plus d'informations sur un paquetage particulier de langpacks.

Codes de langue

LANGUE	SUFFIXE LANGPACKS	VALEUR \$LANG
Anglais (États-Unis)	en	en_US.utf8
Assamais	comme	as_IN.utf8
Bengali	bn	bn_IN.utf8
Chinois (simplifié)	zh_CN	zh_CN.utf8
Chinois (traditionnel)	zh_TW	zh_TW.utf8
Français	FR	fr_FR.utf8
Allemand	de	de_DE.utf8
Gujarati	gu	gu_IN.utf8
Hindi	hi	hi_IN.utf8
Italien	it	it_IT.utf8
Japonais	ja	ja_JP.utf8
Kannada	kn	kn_IN.utf8
Coréen	ko	ko_KR.utf8
Malayalam	ml	ml_IN.utf8
Marathi	mr	mr_IN.utf8
Odia	ou	or_IN.utf8
Portugais (brésilien)	pt_BR	pt_BR.utf8

LANGUE	SUFFIXE LANGPACKS	VALEUR \$LANG
Pendjabi	pa	pa_IN.utf8
Russe	ru	ru_RU.utf8
Espagnol	es	es_ES.utf8
Tamoul	ta	ta_IN.utf8
Télougou	te	te_IN.utf8

CHAPITRE 1

PRISE EN MAIN DE RED HAT ENTERPRISE LINUX

PROJET

Décrire et définir Open Source, Linux, les distributions Linux et Red Hat Enterprise Linux.

OBJECTIFS

- Décrire et expliquer l'objet de Linux, d'Open Source, des distributions Linux et de Red Hat Enterprise Linux.

SECTIONS

- Qu'est-ce que Linux ?

INTERROGATION

Prise en main de Red Hat Enterprise Linux

QU'EST-CE QUE LINUX ?

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir définir et expliquer l'objet de Linux, d'Open Source, des distributions Linux et de Red Hat Enterprise Linux.

QUEL INTÉRÊT L'APPRENTISSAGE DE LINUX PRÉSENTE-T-IL POUR VOUS ?

Linux est une technologie essentielle que les professionnels de l'informatique doivent connaître.

En effet, Linux est très répandu, et si vous utilisez Internet, vous êtes probablement déjà en interaction avec les systèmes Linux dans votre vie quotidienne. Par exemple, vous interagissez avec les systèmes Linux lorsque vous naviguez sur le Web et que vous achetez et vendez des produits sur des sites de commerce électronique.

Cependant, l'utilisation de Linux est beaucoup plus vaste que cela. Linux gère des systèmes de points de vente et les marchés boursiers mondiaux, et équipe également les téléviseurs intelligents et systèmes de divertissement à bord. Il est installé sur la plupart des 500 premiers supercalculateurs du monde. Linux fournit les technologies de base qui alimentent la révolution du cloud et les outils utilisés pour construire la prochaine génération d'applications de microservices en conteneur, les technologies de stockage logicielles et les solutions de gros volumes de données.

Linux et Microsoft Windows sont les principaux acteurs du datacenter moderne, et dans ce contexte, Linux constitue un segment en pleine croissance. L'apprentissage de Linux se justifie à plusieurs titres :

- Un utilisateur Windows doit interagir avec Linux.
- Dans le développement d'applications, Linux héberge l'application ou son environnement d'exécution.
- Dans le contexte du cloud computing, les instances de cloud dans l'environnement de cloud privé ou public utilisent Linux comme système d'exploitation.
- Avec les applications mobiles ou l'Internet des objets (IoT), il y a de fortes chances que le système d'exploitation de votre appareil utilise Linux.
- Si vous recherchez de nouvelles opportunités dans l'informatique, les compétences en Linux sont très demandées.

QUELS SONT LES ATOUTS MAJEURS DE LINUX ?

Les réponses sont nombreuses ; toutefois, nous pouvons mentionner en particulier :

- Linux est un logiciel Open Source.

Open Source n'implique pas seulement que vous pouvez voir comment fonctionne le système. Vous pouvez également expérimenter des modifications et les partager librement pour que les autres les utilisent. Le modèle Open Source signifie que les améliorations sont plus faciles à réaliser, ce qui permet d'innover plus rapidement.

- Linux fournit un accès facile à une puissante *interface de ligne de commande (CLI)* permettant de créer des scripts.

Linux repose sur la philosophie de conception basique selon laquelle les utilisateurs peuvent effectuer toutes les tâches d'administration à partir de l'interface de ligne de commande.

Il facilite l'automatisation, le déploiement et le provisioning, et simplifie l'administration des systèmes locaux et distants. Contrairement à d'autres systèmes d'exploitation, ces fonctionnalités ont été intégrées depuis le début. De plus, le principe de base a toujours été de permettre l'activation de ces fonctionnalités importantes.

- Linux est un système d'exploitation modulaire qui vous permet de remplacer ou de supprimer facilement des composants.

Les composants du système peuvent être mis à niveau et mis à jour si nécessaire. Un système Linux peut être un poste de travail de développement à usage général ou une appliance logicielle extrêmement simplifiée.

EN QUOI CONSISTE UN LOGICIEL OPEN SOURCE ?

Un *logiciel Open Source* est un logiciel comprenant du code source que tout le monde peut utiliser, étudier, modifier et partager.

Le *code source* est l'ensemble d'instructions lisibles par l'homme utilisées pour créer un programme. Il peut être interprété comme un script ou compilé dans un fichier exécutable binaire qui est directement exécuté par l'ordinateur. Lors de la création du code source, celui-ci est protégé par le droit d'auteur et le détenteur du droit d'auteur contrôle les conditions dans lesquelles le logiciel peut être copié, adapté et distribué. Les utilisateurs peuvent utiliser ce logiciel sous licence de logiciel.

Certains logiciels contiennent du code source qui n'est visible, modifiable et distribuable que par les personnes, l'équipe ou l'organisation qui l'ont créé. Ce logiciel est parfois appelé logiciel « propriétaire » ou « à code source fermé ». En règle générale, la licence permet uniquement à l'utilisateur final d'exécuter le programme, et ne fournit aucun accès à la source, ou un accès très limité.

Le logiciel Open Source est différent. Lorsque le détenteur du droit d'auteur fournit un logiciel sous une licence Open Source, il accorde à l'utilisateur le droit d'exécuter le programme et de visualiser, modifier, compiler et redistribuer le code source, sans redevance.

L'Open Source favorise la collaboration, le partage, la transparence et l'innovation rapide, car il encourage les utilisateurs qui ne sont pas les développeurs d'origine à apporter des modifications et des améliorations au logiciel et à partager ce dernier avec d'autres.

Le fait que le logiciel soit Open Source n'empêche absolument pas de l'utiliser ou de le fournir à des fins commerciales. L'Open Source constitue une part essentielle des opérations commerciales de nombreuses organisations. Certaines licences Open Source permettent de réutiliser le code dans des produits propriétaires. On peut vendre du code Open Source, mais les conditions d'utilisation des vraies licences Open Source permettent généralement au client de redistribuer le code source. Le plus souvent, des fournisseurs tels que Red Hat fournissent une aide commerciale pour le déploiement, l'assistance et l'extension de solutions basées sur des produits Open Source.

L'Open Source présente de nombreux avantages pour l'utilisateur :

- *Contrôle* : voyez ce que le code fait et changez-le pour l'améliorer.
- *Formation* : tirez parti du code issu de situations réelles et développez des applications plus efficaces.

CHAPITRE 1 | Prise en main de Red Hat Enterprise Linux

- **Sécurité** : inspectez le code sensible, corrigez-le avec ou sans l'aide des développeurs d'origine.
- **Stabilité** : l'existence du code n'est pas impactée en cas de changement de développeur ou de distributeur d'origine.

En fin de compte, l'Open Source permet de créer de meilleurs logiciels avec un retour sur investissement plus élevé grâce à la collaboration.

TYPES DE LICENCES OPEN SOURCE

Il existe plusieurs façons de fournir des logiciels Open Source. Les conditions d'utilisation de la licence de logiciel déterminent la manière dont le code source peut être combiné avec un autre code ou réutilisé. Des centaines de licences Open Source différentes existent. Cependant, pour être Open Source, les licences doivent permettre d'utiliser, de visualiser, de modifier, de compiler et de distribuer le code librement.

Il existe deux grandes catégories de licences Open Source qui sont particulièrement importantes :

- Les licences *copyleft* sont conçues pour encourager le maintien du code Open Source.
- Les licences *permissives* sont conçues pour optimiser la réutilisation du code.

Les licences copyleft ou « partageables » exigent que quiconque distribuant le code source, avec ou sans modification, transmette également la liberté de copier, modifier et distribuer le code. Le principal avantage de ces licences est lié au fait qu'elles permettent de conserver le code existant et que les améliorations apportées à ce code permettent d'ouvrir et d'augmenter la quantité de code Open Source disponible. Les licences copyleft courantes incluent la licence *GNU General Public License (GPL)* et la licence *GNU Lesser Public License (LGPL)*.

Les licences permissives sont conçues pour optimiser la réutilisation du code source. Le code source peut être utilisé à n'importe quelle fin, tant que les déclarations de droits d'auteur et de licence sont préservées, notamment sous des licences plus restrictives, voire propriétaires. Cela rend très facile la réutilisation de ce code, mais au risque d'encourager des améliorations uniquement propriétaires. Plusieurs licences Open Source permissives communément utilisées incluent la licence *MIT/X11*, la licence *BSD simplifiée* et la licence *logicielle Apache 2.0*.

QUI DÉVELOPPE UN LOGICIEL OPEN SOURCE ?

C'est une idée fausse de penser que l'Open Source est uniquement développé par une « armée de volontaires » ou même par une armée d'individus, en plus de Red Hat. Le développement Open Source est aujourd'hui extrêmement professionnel. De nombreux développeurs sont mandatés par leurs organisations pour travailler sur des projets Open Source afin de construire et d'apporter les améliorations dont leurs clients et eux-mêmes ont besoin.

Les bénévoles et la communauté universitaire jouent un rôle important et peuvent apporter une contribution essentielle, en particulier dans les domaines des nouvelles technologies. La combinaison du développement formel et informel crée un environnement hautement dynamique et productif.

QUI EST RED HAT ?

Red Hat est le premier fournisseur mondial de solutions logicielles Open Source, s'appuyant sur une approche communautaire pour fournir des technologies de cloud, de virtualisation, de stockage, de middleware et Linux fiables et très performantes. Red Hat a pour mission de catalyser les communautés de clients, de collaborateurs et de partenaires qui optent pour l'Open Source pour créer des technologies innovantes.

Red Hat a pour rôle d'aider les clients à entrer en relation avec la communauté Open Source et les partenaires pour utiliser efficacement les solutions logicielles Open Source. Red Hat participe activement à la communauté Open Source et la soutient. De nombreuses années d'expérience ont convaincu l'entreprise de l'importance de l'Open Source pour l'avenir du secteur des technologies de l'information.

Red Hat est célèbre pour sa participation à la communauté Linux et la distribution Red Hat Enterprise Linux. Cependant, Red Hat est également très actif dans d'autres communautés Open Source, notamment dans les projets de middleware centrés sur la communauté de développeurs JBoss, les solutions de virtualisation, les technologies cloud comme OpenStack et OpenShift, ainsi que les projets de stockage logiciel Ceph et Gluster, entre autres.

QU'EST-CE QU'UNE DISTRIBUTION LINUX ?

Une *distribution Linux* est un système d'exploitation installable, construit à partir d'un noyau Linux et prenant en charge les bibliothèques et les programmes utilisateur. Un système d'exploitation Linux complet n'est pas développé par une seule organisation, mais par un ensemble de communautés de développement Open Source indépendantes travaillant avec des composants logiciels individuels. Une distribution offre aux utilisateurs un moyen simple d'installer et de gérer un système Linux opérationnel.

En 1991, un jeune étudiant en informatique nommé Linus Torvalds a développé un noyau de type Unix qu'il a appelé *Linux* et concédé en tant que logiciel Open Source sous la licence GPL. Le noyau est le composant principal du système d'exploitation qui gère le matériel, la mémoire et la planification des programmes d'exécution. Ce noyau Linux pouvait ensuite être complété par d'autres logiciels Open Source, tels que des utilitaires et des programmes du projet GNU, l'interface graphique du système *X Window* du MIT et de nombreux autres composants Open Source, tels que le serveur de messagerie Sendmail ou le serveur Web HTTP Apache, afin de créer un système d'exploitation de type Unix complet et Open Source.

Cependant, l'un des défis pour les utilisateurs Linux était d'assembler tous ces composants provenant de nombreuses sources différentes. Très tôt, les développeurs Linux se sont employés à fournir une distribution d'outils prédéfinis et testés que les utilisateurs pouvaient télécharger et utiliser pour configurer rapidement leurs systèmes Linux.

Il existe de nombreuses distributions Linux différentes, avec des objectifs et des critères divers de sélection et de prise en charge du logiciel fourni par leur distribution. Cependant, les distributions ont généralement de nombreuses caractéristiques communes :

- Les distributions consistent en un noyau Linux et la prise en charge de programmes d'espace utilisateur.
- Les distributions peuvent être petites et spécialisées ou inclure des milliers de programmes Open Source.
- Les distributions doivent fournir un moyen d'installer et de mettre à jour la distribution et ses composants.
- Le fournisseur de la distribution doit prendre en charge ce logiciel et, idéalement, participer directement à la communauté qui développe ce logiciel.

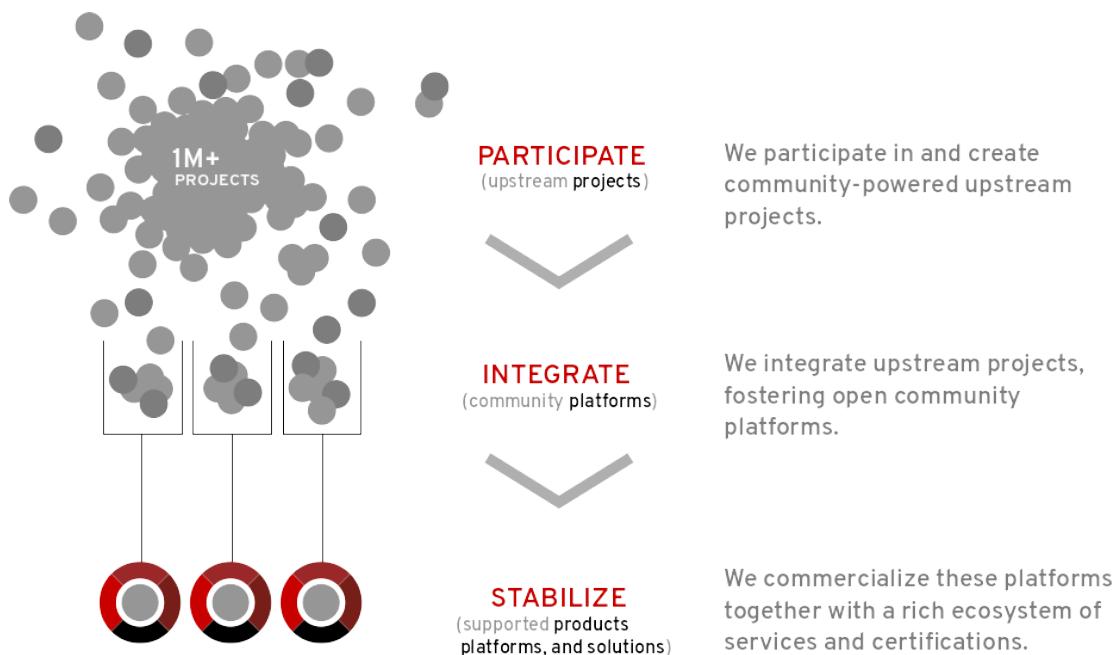
Red Hat Enterprise Linux est la distribution Linux commercialisée de Red Hat.

RED HAT ENTERPRISE LINUX

Développement de Red Hat Enterprise Linux

Red Hat développe et intègre des logiciels Open Source dans RHEL via un processus à plusieurs étapes.

- Red Hat *participe* en soutenant des projets individuels Open Source. Il fournit du code, du temps de développement, des ressources et d'autres types de support, en collaborant souvent avec des développeurs d'autres distributions Linux. Cela contribue à améliorer la qualité générale des logiciels pour tous.
- Red Hat parraine et *intègre* des projets Open Source dans une distribution Linux communautaire, *Fedora*. Fedora fournit un environnement de travail gratuit qui peut servir de laboratoire de développement et de terrain d'essai pour les fonctionnalités intégrées aux produits commercialisés.
- Red Hat *stabilise* le logiciel afin de garantir qu'il est prêt pour une prise en charge et une normalisation à long terme, et l'intègre dans une distribution destinée aux entreprises, RHEL.



Fedora

Fedora est un projet communautaire qui produit et publie une version complète et gratuite du système d'exploitation Linux. Red Hat parraine la communauté et collabore avec des représentants de la communauté pour intégrer les derniers logiciels en amont dans une distribution sécurisée et à évolution rapide. Le projet Fedora apporte sa contribution au monde Open Source libre, et tout le monde peut participer.

Toutefois, Fedora se concentre sur l'innovation et l'excellence, et non sur la stabilité à long terme. De nouvelles mises à jour majeures ont lieu tous les six mois et peuvent apporter des changements significatifs. Fedora ne prend en charge les versions que pendant environ un an (deux mises à jour majeures), ce qui le rend moins approprié pour une utilisation en entreprise.

Red Hat Enterprise Linux

Red Hat Enterprise Linux (RHEL) est la distribution Linux commercialisée pour les entreprises de Red Hat. Il s'agit de la plateforme leader de l'informatique Open Source, et pas seulement d'un ensemble de projets Open Source éprouvés. RHEL est soumis à de nombreux tests et s'appuie sur un vaste écosystème de partenaires, de certifications de matériel et de logiciels, de services de consulting, de formation et de garanties d'assistance et de maintenance pluriannuelles.

Red Hat base ses principales versions de RHEL sur Fedora. Cependant, Red Hat peut ensuite choisir les paquetages à inclure et apporter d'autres améliorations (retour de contribution aux projets en amont et Fedora), et prendre des décisions de configuration qui répondent aux besoins des clients. Red Hat aide les fournisseurs et les clients à dialoguer avec la communauté Open Source et à travailler avec le développement en amont pour développer des solutions et résoudre des problèmes.

Red Hat Enterprise Linux utilise un modèle de distribution sur abonnement. Comme il s'agit d'un logiciel Open Source, cela n'est pas un droit de licence. Il s'agit d'une contribution pour l'assistance, la maintenance, les mises à jour, les correctifs de sécurité, l'accès à la base de connaissances sur le Portail Client Red Hat (<http://access.redhat.com/>), les certifications, etc. Le client paie pour une prise en charge et une expertise à long terme, un engagement et une assistance quand il en a besoin.

Lorsque les mises à jour majeures deviennent disponibles, les clients peuvent y accéder quand bon leur semble sans frais supplémentaires. Cela simplifie la gestion des aspects financiers et pratiques des mises à jour du système.

CentOS

CentOS est une distribution Linux communautaire dérivée d'une grande partie du code base et autres sources de l'Open Source Red Hat Enterprise Linux. Elle est gratuite, facile à installer, dispose d'un personnel et est assistée par une communauté active d'utilisateurs composée de bénévoles qui travaillent indépendamment de Red Hat.

Le tableau suivant répertorie les différences clés entre CentOS et Red Hat Enterprise Linux.

CENTOS	RED HAT ENTERPRISE LINUX
Assistance en libre-service uniquement.	Plusieurs niveaux d'assistance sont disponibles, notamment l'assistance standard pendant les heures ouvrables, l'assistance Premium 24x7 pour les problèmes critiques et l'aide pour l'abonnement d'entrée de gamme. Différents niveaux de service de l'application peuvent être combinés et mis en correspondance dans un environnement.
La génération d'errata se produit lorsqu'une version officielle d'errata RHEL est disponible.	Une réponse rapide aux problèmes de la part des développeurs internes et des correctifs peuvent être disponibles avant le lancement officiel de l'errata RHEL.

CENTOS	RED HAT ENTERPRISE LINUX
Les mises à jour de paquets sont fournies pour la version mineure la plus récente jusqu'à la fin de la phase 2 de l'assistance de maintenance de RHEL.	Les mises à jour sont disponibles pour les versions mineures plus anciennes dans le cadre du programme Extended Update Support (EUS) et pendant des années après la fin de l'assistance de maintenance 2 via le programme Extended Lifecycle Support (ELS).
Elles ne sont généralement pas certifiées par des éditeurs de logiciels tels que SAS, SAP et Oracle en tant que plateforme prise en charge.	Des milliers d'applications certifiées proviennent de centaines d'éditeurs de logiciels indépendants.
Des ressources d'aide et de documentation sont disponibles grâce aux forums, aux listes de diffusion, aux discussions en ligne, au site Web du projet CentOS et son wiki et à d'autres sources communautaires.	De la documentation, des architectures de référence, des études de cas et des articles de la base de connaissances sont disponibles via le Portail Client Red Hat. Accédez aux ateliers du Portail Client Red Hat afin de pouvoir utiliser un ensemble d'outils pour améliorer les performances, identifier les problèmes de sécurité ou vous aider à résoudre les problèmes. Profitez de l'analyse proactive de système optionnelle de Red Hat Insights, un outil SaaS pour fournir une évaluation en temps réel des risques liés aux performances, à la disponibilité, à la stabilité et à la sécurité.

COMMENT ESSAYER RED HAT ENTERPRISE LINUX ?

Il existe plusieurs façons d'essayer Red Hat Enterprise Linux. L'une des méthodes consiste à télécharger une copie d'évaluation à partir du site Web à l'adresse suivante : <https://access.redhat.com/products/red-hat-enterprise-linux/evaluation>. Cette page contient des liens vers des informations supplémentaires.

Red Hat propose également des abonnements gratuits à un certain nombre de produits à des fins de développement par le biais du programme des développeurs de Red Hat à l'adresse suivante : <https://developer.redhat.com>. Ces abonnements permettent aux développeurs de développer rapidement, de créer des prototypes, de tester et de démontrer que leurs logiciels sont déployés sur ces produits d'entreprise.

Une autre approche consiste à déployer une instance de Red Hat Enterprise Linux mise à disposition via un fournisseur de cloud. Par exemple, Red Hat dispose d'AMI officiels disponibles pour Red Hat Entreprise Linux sur le marché Amazon AWS.

Pour plus d'informations, visitez la page « Prendre en main Red Hat Enterprise Linux », indiquée à la fin de cette section.



RÉFÉRENCES

Prendre en main Red Hat Enterprise Linux

<https://access.redhat.com/products/red-hat-enterprise-linux#getstarted>

La méthode Open Source

<https://opensource.com/open-source-way>

► QUIZ

PRISE EN MAIN DE RED HAT ENTERPRISE LINUX

Choisissez les éléments appropriés en réponse aux questions suivantes :

► 1. Parmi les avantages suivants, lesquels correspondent à ceux que les logiciels Open Source présentent à l'utilisateur ? (Choisissez-en deux.)

- a. L'existence du code n'est pas impactée en cas de changement de développeur ou de distributeur d'origine.
- b. Les parties de code sensibles sont protégées et disponibles uniquement pour le développeur d'origine.
- c. Vous pouvez tirer parti du code issu de situations réelles et développer des applications plus efficaces.
- d. Le code reste ouvert tant qu'il se trouve dans un référentiel public, mais la licence peut changer si elle est incluse dans un logiciel source propriétaire.

► 2. Parmi les moyens suivants, lesquels Red Hat utilise-t-il pour développer ses produits pour l'avenir et interagir avec la communauté ? (Choisissez-en deux.)

- a. Parrainage et intégration de projets Open Source dans le projet communautaire Fedora.
- b. Développement d'outils d'intégration spécifiques uniquement disponibles dans les distributions Red Hat.
- c. Participation à des projets en amont.
- d. Reconditionnement des produits communautaires et création de nouvelles licences pour ces derniers.

► 3. Parmi les déclarations suivantes, lesquelles décrivent les avantages de Linux ? (Choisissez-en deux.)

- a. Linux est entièrement développé par des bénévoles, ce qui en fait un système d'exploitation à faible coût.
- b. Linux est modulaire et peut être configuré comme un bureau graphique complet ou une petite appliance.
- c. Linux est verrouillé dans un état connu pendant au moins un an pour chaque version, ce qui facilite le développement de logiciels personnalisés.
- d. Linux comprend une puissante interface de ligne de commande permettant de créer des scripts et qui facilite l'automatisation et le provisioning.

► SOLUTION

PRISE EN MAIN DE RED HAT ENTERPRISE LINUX

Choisissez les éléments appropriés en réponse aux questions suivantes :

► 1. Parmi les avantages suivants, lesquels correspondent à ceux que les logiciels Open Source présentent à l'utilisateur ? (Choisissez-en deux.)

- a. L'existence du code n'est pas impactée en cas de changement de développeur ou de distributeur d'origine.
- b. Les parties de code sensibles sont protégées et disponibles uniquement pour le développeur d'origine.
- c. Vous pouvez tirer parti du code issu de situations réelles et développer des applications plus efficaces.
- d. Le code reste ouvert tant qu'il se trouve dans un référentiel public, mais la licence peut changer si elle est incluse dans un logiciel source propriétaire.

► 2. Parmi les moyens suivants, lesquels Red Hat utilise-t-il pour développer ses produits pour l'avenir et interagir avec la communauté ? (Choisissez-en deux.)

- a. Parrainage et intégration de projets Open Source dans le projet communautaire Fedora.
- b. Développement d'outils d'intégration spécifiques uniquement disponibles dans les distributions Red Hat.
- c. Participation à des projets en amont.
- d. Reconditionnement des produits communautaires et création de nouvelles licences pour ces derniers.

► 3. Parmi les déclarations suivantes, lesquelles décrivent les avantages de Linux ? (Choisissez-en deux.)

- a. Linux est entièrement développé par des bénévoles, ce qui en fait un système d'exploitation à faible coût.
- b. Linux est modulaire et peut être configuré comme un bureau graphique complet ou une petite appliance.
- c. Linux est verrouillé dans un état connu pendant au moins un an pour chaque version, ce qui facilite le développement de logiciels personnalisés.
- d. Linux comprend une puissante interface de ligne de commande permettant de créer des scripts et qui facilite l'automatisation et le provisioning.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Un logiciel Open Source est un logiciel comprenant du code source que tout le monde peut librement utiliser, étudier, modifier et partager.
- Une distribution Linux est un système d'exploitation installable, construit à partir d'un noyau Linux et prenant en charge les bibliothèques et les programmes utilisateur.
- Red Hat participe à la prise en charge et à la contribution de code aux projets Open Source, parraine et intègre le logiciel de projet dans des distributions communautaires, et le stabilise pour le proposer comme produit pris en charge pour des entreprises.
- Red Hat Enterprise Linux est la distribution Open Source Linux commercialisée pour les entreprises de Red Hat.

CHAPITRE 2

ACCÈS À LA LIGNE DE COMMANDE

PROJET

Se connecter à un système Linux et exécuter des commandes simples à l'aide du shell.

OBJECTIFS

- Se connecter à un système Linux sur une console locale et exécuter des commandes simples à l'aide du shell.
- Se connecter à un système Linux en utilisant l'environnement de bureau GNOME 3 et exécuter des commandes depuis l'invite du shell dans un programme de terminal.
- Gagner du temps en utilisant la saisie semi-automatique par tabulation, l'historique des commandes et les raccourcis de modification de commande pour exécuter des commandes dans le shell bash.

SECTIONS

- Accès à la ligne de commande (et quiz)
- Accès à la ligne de commande depuis le bureau (et exercice guidé)
- Exécution de commandes à l'aide du shell bash (et quiz)

ATELIER

Accès à la ligne de commande

ACCÈS À LA LIGNE DE COMMANDE

OBJECTIFS

Après avoir terminé cette section, les stagiaires doivent pouvoir se connecter à un système Linux et exécuter des commandes simples en utilisant le shell.

PRÉSENTATION DU SHELL BASH

Une *ligne de commande* est une interface en mode texte qui permet de saisir des instructions dans un système informatique. La ligne de commande de Linux est fournie par un programme appelé *shell* (interpréteur de commandes). Diverses options du programme shell ont été développées au fil du temps et différents utilisateurs peuvent être configurés pour utiliser différents shells. Cependant, la plupart des utilisateurs conservent les paramètres par défaut.

Le shell par défaut des utilisateurs de Red Hat Enterprise Linux est le shell GNU Bourne-Again (**bash**). Bash est une version améliorée de l'un des shells les plus populaires sur les systèmes UNIX, le shell Bourne (**sh**).

Lorsqu'un shell est utilisé de manière interactive, il affiche une chaîne lorsqu'il attend une commande de la part de l'utilisateur. On l'appelle *l'invite du shell*. Lorsqu'un utilisateur standard lance un shell, l'invite par défaut se termine par le caractère **\$**, comme indiqué ci-dessous.

```
[user@host ~]$
```

Le caractère **\$** est remplacé par un **#** si le shell est exécuté par le super utilisateur, **root**. Cela souligne bien qu'il s'agit du shell d'un super utilisateur, ce qui permet d'éviter les accidents et les erreurs susceptibles d'entraîner un effet sur tout le système. L'invite du super utilisateur est indiquée ci-dessous.

```
[root@host ~]#
```

bash peut s'avérer puissant pour l'exécution de commandes. Le shell **bash** fournit un langage de script qui permet l'automatisation des tâches. Le shell offre des fonctionnalités supplémentaires qui peuvent simplifier ou rendre possibles des opérations difficiles à accomplir efficacement avec des outils graphiques.



NOTE

Le shell **bash** repose sur le même concept que l'interpréteur de ligne de commande que l'on trouve dans les dernières versions de **cmd.exe** de Microsoft Windows, bien que le shell **bash** offre un langage de script plus évolué. Il s'apparente également au Windows PowerShell de Windows 7 et Windows Server 2008 R2 et version ultérieure. Les administrateurs Apple Mac qui font appel à l'utilitaire Terminal auront le plaisir de constater que **bash** est le shell par défaut sous MacOS.

CONCEPTS ÉLÉMENTAIRES DU SHELL

Les commandes saisies dans l'invite du shell comportent trois parties principales :

CHAPITRE 2 | Accès à la ligne de commande

- la *commande* à exécuter ;
- les *options* qui ajustent le comportement de la commande ;
- les *arguments*, qui sont, le plus souvent, les cibles de la commande.

La *commande* est le nom du programme à exécuter. Elle peut être suivie d'une ou plusieurs *options* qui affinent le comportement de la commande ou ce qu'elle va faire. Les options commencent normalement par un ou deux tirets (**-a** ou **--all**, par exemple) pour les distinguer des arguments. Les commandes peuvent également être suivies d'un ou plusieurs *arguments* qui indiquent souvent une cible sur laquelle la commande doit s'exécuter.

Par exemple, la commande **usermod -L user01** a une commande (**usermod**), une option (**-L**) et un argument (**user01**). Cette commande a pour effet de verrouiller le mot de passe du compte de l'utilisateur **user01**.

CONNEXION À UN ORDINATEUR LOCAL

Pour exécuter le shell, vous devez vous connecter à l'ordinateur sur un *terminal*. Un terminal est une interface textuelle utilisée pour entrer des commandes dans un système informatique et imprimer les résultats. Il y a plusieurs manières de procéder.

L'ordinateur pourrait disposer d'un clavier matériel et d'un écran pour les entrées et les sorties qui y sont directement connectées. Il s'agit de la *console physique* de la machine Linux. La console physique prend en charge plusieurs *consoles virtuelles* qui peuvent exécuter des terminaux distincts. Chaque console virtuelle prend en charge une session connectée indépendante. Vous pouvez passer de l'une à l'autre en appuyant simultanément sur **Ctrl+Alt** et une touche de fonction (de **F1** à **F6**). La plupart de ces consoles virtuelles exécutent un terminal fournissant une invite de connexion texte. Si vous entrez votre nom d'utilisateur et votre mot de passe correctement, vous vous connectez et une invite du shell s'affiche.

L'ordinateur pourrait fournir une invite de connexion graphique sur l'une des consoles virtuelles. Vous pouvez l'utiliser pour vous connecter à un *environnement graphique*. L'environnement graphique s'exécute également sur une console virtuelle. Pour obtenir une invite du shell, vous devez démarrer un programme de terminal dans l'environnement graphique. L'invite du shell est fournie dans une fenêtre d'application du programme graphique de terminal.



NOTE

De nombreux administrateurs système choisissent de ne pas exécuter d'environnement graphique sur leurs serveurs. Les ressources qui seraient utilisées par l'environnement graphique sont alors utilisées par les services du serveur.

Dans Red Hat Enterprise Linux 8, si l'environnement graphique est disponible, il s'exécute sur la première console virtuelle, nommée **tty1**. Cinq invites de connexion texte supplémentaires sont disponibles sur les consoles virtuelles deux à six.

Si vous vous connectez à l'aide de l'écran de connexion graphique, votre environnement graphique démarre sur la première console virtuelle qui n'est actuellement pas utilisée par une session de connexion. Normalement, votre session graphique remplace l'invite de connexion sur la deuxième console virtuelle. (**tty2**). Cependant, si cette console est utilisée par une session de connexion texte active (pas seulement une invite de connexion), la console virtuelle libre suivante est utilisée à la place.

L'écran de connexion graphique continue de s'exécuter sur la première console virtuelle. (**tty1**). Si vous êtes déjà connecté à une session graphique et que vous vous connectez sous l'identité

CHAPITRE 2 | Accès à la ligne de commande

d'un autre utilisateur sur l'écran de connexion graphique ou que vous utilisez l'élément de menu Switch User (qui permet de changer d'utilisateur dans l'environnement graphique sans se déconnecter), un autre environnement graphique est démarré pour cet utilisateur sur la console virtuelle libre suivante.

Lorsque vous vous déconnectez d'un environnement graphique, celui-ci se ferme et la console physique revient automatiquement à l'écran de connexion graphique de la première console virtuelle.



NOTE

Dans Red Hat Enterprise Linux 6 et 7, l'écran de connexion graphique s'exécute sur la première console virtuelle, mais lorsque vous vous connectez, votre environnement graphique initial remplace l'écran de connexion sur la première console virtuelle au lieu de démarrer sur une nouvelle console virtuelle.

Dans Red Hat Enterprise Linux 5 et versions antérieures, les six premières consoles virtuelles fournissaient toujours des invites de connexion texte. Lorsque l'environnement graphique est lancé, il s'exécute sur la console virtuelle sept (accessible via **Ctrl+Alt+F7**).

Un serveur administré à distance est dépourvu de clavier et d'écran. Un datacenter peut contenir de nombreux racks de serveurs administrés à distance, et le fait ne pas les équiper d'un clavier et d'un écran permet d'économiser de l'espace et de l'argent. Pour permettre aux administrateurs de se connecter, un serveur administré à distance peut disposer d'une invite de connexion fournie par sa *console de série*, s'exécutant sur un port série connecté à un serveur de console en réseau pour l'accès à distance de la console de série.

La console de série est normalement utilisée pour réparer le serveur si sa propre carte réseau est mal configurée et que la connexion via sa propre connexion réseau devient impossible. Cependant, la plupart du temps, les serveurs administrés à distance sont accessibles via d'autres moyens sur le réseau.

CONNEXION SUR LE RÉSEAU

Les utilisateurs et les administrateurs Linux ont souvent besoin d'obtenir un accès shell à un système distant en se connectant à ce dernier via le réseau. Dans un environnement informatique moderne, de nombreux serveurs administrés à distance sont en fait des machines virtuelles ou qui s'exécutent en tant qu'instances de cloud public ou privé. Ces systèmes ne sont pas physiques et ne disposent pas de consoles matérielles réelles. Ils pourraient même ne pas donner accès à leur console de série ou leur console physique (simulée).

Sous Linux, le moyen le plus courant d'obtenir une invite du shell sur un système distant consiste à utiliser SSH (Secure Shell). La plupart des systèmes Linux (y compris Red Hat Enterprise Linux) et MacOS fournissent le programme **ssh** de ligne de commande OpenSSH dans ce but.

Dans cet exemple, un utilisateur avec une invite du shell sur la machine hôte utilise **ssh** pour se connecter au système Linux distant `remotehost` en tant qu'utilisateur `remoteuser` :

```
[user@host ~]$ ssh remoteuser@remotehost
remoteuser@remotehost's password: password
[remoteuser@remotehost ~]$
```

La commande **ssh** chiffre la connexion pour sécuriser la communication contre l'interception ou le piratage des mots de passe et du contenu.

Pour plus de sécurité, certains systèmes (tels que les nouvelles instances de cloud) n'autorisent pas les utilisateurs à se servir d'un mot de passe pour se connecter avec **ssh**. Une autre façon de s'authentifier sur un ordinateur distant sans entrer de mot de passe consiste à utiliser l'*authentification par clé publique*.

Avec cette méthode d'authentification, les utilisateurs ont un fichier d'identité spécial contenant une *clé privée* qui équivaut à un mot de passe, et qu'ils gardent secret. Leur compte sur le serveur est configuré avec une *clé publique* correspondante qui ne doit pas forcément être secrète. Lors de la connexion, les utilisateurs peuvent configurer **ssh** en vue de fournir la clé privée, et si la clé publique correspondante est installée dans ce compte sur ce serveur distant, les utilisateurs sont connectés sans demander de mot de passe.

Dans l'exemple suivant, un utilisateur avec une invite du shell sur la machine **hôte** se connecte à **remotehost** en tant que **remoteuser** à l'aide de **ssh**, en utilisant l'authentification par clé publique. L'option **-i** est utilisée pour spécifier le fichier de clé privée de l'utilisateur qui est **mylab.pem**. La clé publique correspondante est déjà configurée en tant que clé autorisée dans le compte **remoteuser**.

```
[user@host ~]$ ssh -i mylab.pem remoteuser@remotehost  
[remoteuser@remotehost ~]$
```

Pour que cela fonctionne, le fichier de clé privée ne doit être lisible que par l'utilisateur qui en est propriétaire. Dans l'exemple précédent, où la clé privée est dans le fichier **mylab.pem**, la commande **chmod 600 mylab.pem** peut être utilisée à cette fin. La définition des permissions de fichiers est décrite plus en détail dans un chapitre ultérieur.

Les utilisateurs peuvent également disposer de clés privées configurées qui sont essayées automatiquement, mais ce sujet sort du cadre de cette section. Les références répertoriées à la fin de cette section contiennent des liens vers de plus amples informations sur ce sujet.

**NOTE**

La première fois que vous vous connectez à une nouvelle machine, un message provenant de **ssh** vous avertit que l'authentification de l'hôte ne peut pas être établie :

```
[user@host ~]$ ssh -i mylab.pem remoteuser@remotehost
The authenticity of host 'remotehost (192.0.2.42)' can't be established.
ECDSA key fingerprint is 47:bf:82:cd:fa:68:06:ee:d8:83:03:1a:bb:29:14:a3.
Are you sure you want to continue connecting (yes/no)? yes
[remoteuser@remotehost ~]$
```

Chaque fois que vous vous connectez à un hôte distant avec **ssh**, l'hôte distant envoie à **ssh** sa clé d'hôte pour s'authentifier et mettre en place une communication chiffrée. La commande **ssh** compare celle-ci à une liste de clés d'hôte enregistrées pour s'assurer qu'elle n'a pas changé. Si la clé d'hôte a changé, cela peut indiquer que quelqu'un essaie de prétendre être cet hôte pour pirater la connexion. Cette situation est connue sous le nom d'attaque d'intercepteur (man-in-the-middle). Dans SSH, les clés d'hôte protègent contre les attaques d'intercepteurs. Ces clés d'hôte sont uniques pour chaque serveur. Elles doivent être changées régulièrement, et chaque fois qu'un problème est suspecté.

Vous recevez un message d'avertissement si votre ordinateur local ne dispose pas de clé d'hôte enregistrée pour l'hôte distant. Si vous entrez **yes**, la clé d'hôte envoyée par l'hôte distant est acceptée et enregistrée pour référence ultérieure. La connexion se poursuit et vous ne devriez plus voir ce message lorsque vous vous connectez à cet hôte. Si vous entrez **no**, la clé d'hôte est rejetée et la connexion arrêtée.

Si l'ordinateur local ne dispose pas d'une clé d'hôte enregistrée et qu'elle ne correspond pas à celle réellement envoyée par l'hôte distant, la connexion est automatiquement arrêtée avec un avertissement.

DÉCONNEXION

Lorsque vous avez terminé d'utiliser le shell et que vous souhaitez quitter, vous avez le choix entre plusieurs méthodes pour mettre fin à la session. Vous pouvez saisir la commande **exit** pour mettre fin à la session shell en cours. Vous pouvez également terminer une session en appuyant sur **Ctrl+D**.

Voici un exemple d'utilisateur qui se déconnecte d'une session SSH :

```
[remoteuser@remotehost ~]$ exit
logout
Connection to remotehost closed.
[user@host ~]$
```



RÉFÉRENCES

Pages du manuel **intro(1)**, **bash(1)**, **console(4)**, **pts(4)**, **ssh(1)** et **ssh-keygen(1)**

Remarque : certains détails de la page de manuel **console(4)** relatifs à **init(8)** et **inittab(5)** sont obsolètes.

Pour plus d'informations sur OpenSSH et l'authentification par clé publique, reportez-vous au chapitre *Using secure communications between two systems with OpenSSH* du manuel *Securing networks* de Red Hat Enterprise Linux 8 à l'adresse https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/securing_networks/assembly_using-secure-communications-with-openssh-securing-networks



NOTE

Des instructions sur la façon de lire les pages **man** et d'autres documents d'aide en ligne sont inclus à la fin de la section suivante.

► QUIZ

ACCÈS À LA LIGNE DE COMMANDE

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. **Quel terme décrit le programme qui interprète et exécute les commandes saisies sous forme de chaînes de caractères ?**
 - a. Commande
 - b. Console
 - c. Shell
 - d. Terminal

- ▶ 2. **Quel terme décrit le signe visuel qui indique qu'un shell interactif attend que l'utilisateur saisisse une commande ?**
 - a. Argument
 - b. Commande
 - c. Option
 - d. Invite

- ▶ 3. **Quel terme décrit le nom d'un programme à exécuter ?**
 - a. Argument
 - b. Commande
 - c. Option
 - d. Invite

- ▶ 4. **Quel terme décrit la partie de la ligne de commande qui ajuste le comportement de la commande ?**
 - a. Argument
 - b. Commande
 - c. Option
 - d. Invite

- ▶ 5. **Quel terme décrit la partie de la ligne de commande qui spécifie la cible sur laquelle la commande doit s'exécuter ?**
 - a. Argument
 - b. Commande
 - c. Option
 - d. Invite

► 6. Quel terme décrit l'écran et le clavier matériels utilisés pour interagir avec un système ?

- a. Console physique
- b. Console virtuelle
- c. Shell
- d. Terminal

► 7. Quel terme décrit l'une des multiples consoles logiques qui peuvent prendre en charge une session de connexion indépendante ?

- a. Console physique
- b. Console virtuelle
- c. Shell
- d. Terminal

► 8. Quel terme décrit l'interface qui fournit un affichage pour la sortie et un clavier pour l'entrée d'une session shell ?

- a. Console
- b. Console virtuelle
- c. Shell
- d. Terminal

► SOLUTION

ACCÈS À LA LIGNE DE COMMANDE

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. **Quel terme décrit le programme qui interprète et exécute les commandes saisies sous forme de chaînes de caractères ?**
 - a. Commande
 - b. Console
 - c. Shell
 - d. Terminal

- ▶ 2. **Quel terme décrit le signe visuel qui indique qu'un shell interactif attend que l'utilisateur saisisse une commande ?**
 - a. Argument
 - b. Commande
 - c. Option
 - d. Invite

- ▶ 3. **Quel terme décrit le nom d'un programme à exécuter ?**
 - a. Argument
 - b. Commande
 - c. Option
 - d. Invite

- ▶ 4. **Quel terme décrit la partie de la ligne de commande qui ajuste le comportement de la commande ?**
 - a. Argument
 - b. Commande
 - c. Option
 - d. Invite

- ▶ 5. **Quel terme décrit la partie de la ligne de commande qui spécifie la cible sur laquelle la commande doit s'exécuter ?**
 - a. Argument
 - b. Commande
 - c. Option
 - d. Invite

► **6. Quel terme décrit l'écran et le clavier matériels utilisés pour interagir avec un système ?**

- a. Console physique
- b. Console virtuelle
- c. Shell
- d. Terminal

► **7. Quel terme décrit l'une des multiples consoles logiques qui peuvent prendre en charge une session de connexion indépendante ?**

- a. Console physique
- b. Console virtuelle
- c. Shell
- d. Terminal

► **8. Quel terme décrit l'interface qui fournit un affichage pour la sortie et un clavier pour l'entrée d'une session shell ?**

- a. Console
- b. Console virtuelle
- c. Shell
- d. Terminal

ACCÈS À LA LIGNE DE COMMANDE DEPUIS LE BUREAU

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir vous connecter au système Linux en utilisant l'environnement de bureau GNOME 3 pour exécuter des commandes depuis l'invite du shell dans un programme de terminal.

INTRODUCTION À L'ENVIRONNEMENT DE BUREAU GNOME

L'*environnement de bureau* est l'interface graphique de l'utilisateur d'un système Linux. L'environnement de bureau par défaut dans Red Hat Enterprise Linux 8 est fourni par GNOME 3. Il offre un bureau intégré aux utilisateurs ainsi qu'une plateforme de développement unifiée basée sur l'environnement graphique offert par Wayland (par défaut) ou le système X Window hérité.

Le shell GNOME fournit les fonctions de base de l'interface utilisateur de l'environnement de bureau GNOME. L'application GNOME Shell est hautement personnalisable. Red Hat Entreprise Linux 8 permet d'adapter l'apparence de GNOME Shell au thème « Standard » qui est utilisé dans cette section. Red Hat Entreprise Linux 7 est adapté à un thème alternatif nommé « Classique » qui est plus proche de l'apparence des anciennes versions de GNOME. L'un ou l'autre thème sont accessibles de manière persistante à la connexion en cliquant sur l'icône en forme d'engrenage en regard du bouton Sign In disponible après avoir sélectionné le compte, mais avant de saisir le mot de passe.

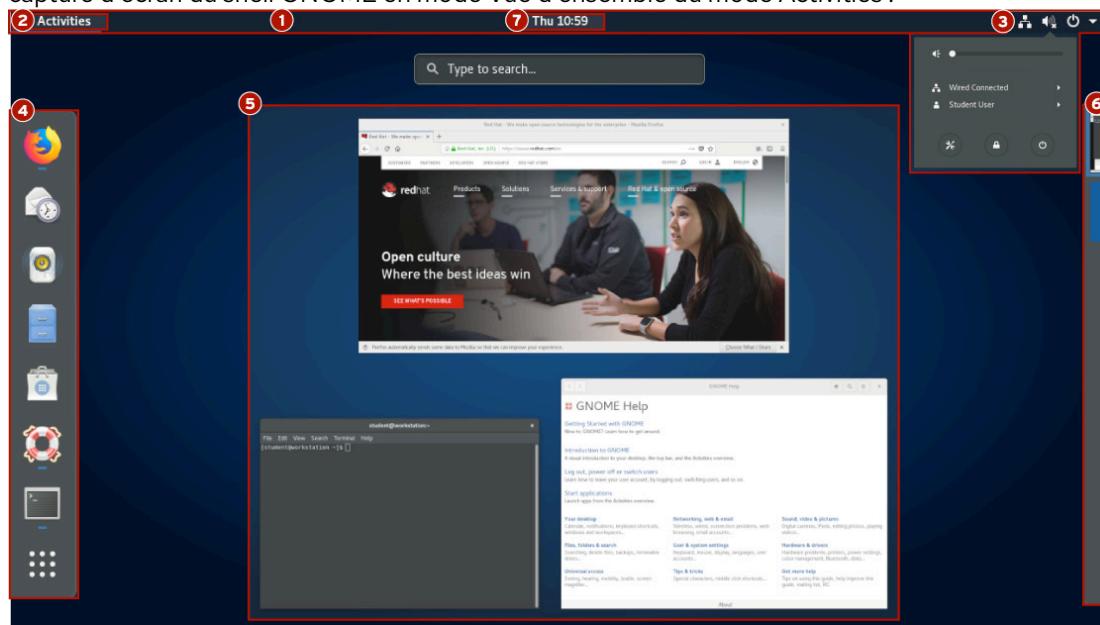


Figure 2.1: Un bureau GNOME 3 vide

À la première connexion d'un utilisateur, un programme de configuration initiale se lance pour aider à configurer les réglages de base du compte. Une fois cette opération effectuée, l'application GNOME Help démarre sur l'écran Getting Started with GNOME. Cet écran comprend des vidéos et de la documentation pour orienter les nouveaux utilisateurs vers l'environnement GNOME 3. Vous pouvez rapidement lancer l'Aide GNOME en cliquant sur le bouton Activities situé à gauche de la barre supérieure et dans le lanceur d'applications qui apparaît à gauche de l'écran, en cliquant sur l'icône en forme de bouée de sauvetage pour la lancer.

Parties du shell GNOME

Les éléments du shell GNOME comprennent les éléments suivants, comme illustré par cette capture d'écran du shell GNOME en mode Vue d'ensemble du mode Activities :



- ➊ *Barre supérieure* : barre qui longe tout le haut de l'écran. Elle s'affiche dans Vue d'ensemble du mode Activities et dans les espaces de travail. La barre supérieure contient le bouton Activities et les contrôles de volume, du réseau, d'accès au calendrier et de sélection entre les modes de saisie du clavier (si plusieurs modes sont configurés).
- ➋ *Vue d'ensemble du mode Activities* : il s'agit d'un mode spécial qui permet à l'utilisateur d'organiser les fenêtres et de lancer des applications. La Vue d'ensemble du mode Activities peut être saisie en cliquant sur le bouton Activities dans le coin supérieur gauche de la barre supérieure ou en appuyant sur la touche **Super**. La touche **Super** (parfois appelée touche **Windows** ou touche de **commande**) se trouve à proximité du coin inférieur gauche d'un clavier de type IBM PC à 104/105 touches ou Apple. Les trois grandes zones de la vue d'ensemble Activities sont le lanceur d'applications sur le côté gauche de l'écran, la vue d'ensemble des fenêtres au milieu de l'écran et le sélecteur d'espace de travail sur le côté droit de l'écran.
- ➌ *Menu système* : le menu est situé dans l'angle supérieur droit de la barre supérieure permet de régler la luminosité de l'écran et d'activer ou de désactiver les connexions réseau. Sous le menu du nom d'utilisateur se trouvent les options permettant d'ajuster les paramètres de compte et de se déconnecter du système. Le menu système propose également des boutons pour ouvrir la fenêtre Settings, verrouiller l'écran ou arrêter le système.
- ➍ *Lanceur d'applications* : liste configurable des icônes des applications favorites de l'utilisateur et des applications en cours d'exécution, complétée par un bouton grille au bas du lanceur d'applications pouvant servir à sélectionner d'autres applications. Les applications peuvent être lancées en cliquant sur l'une des icônes, ou en utilisant le bouton grille pour trouver une application moins utilisée. Ce « lanceur d'applications » est également parfois appelé le dock.
- ➎ *Vue d'ensemble des fenêtres* : zone au centre de la vue d'ensemble des activités affichant les vignettes de toutes les fenêtres actives dans l'espace de travail actuel. Cela permet de faire apparaître plus facilement les fenêtres au premier plan dans un espace de travail encombré ou de les déplacer vers un autre espace de travail.
- ➏ *Sélecteur d'espace de travail* : zone à droite de la vue d'ensemble Activities qui affiche les vignettes de tous les espaces de travail actifs et permet de sélectionner des espaces de travail ainsi que de déplacer des fenêtres d'un espace de travail à un autre.

CHAPITRE 2 | Accès à la ligne de commande

- 7 Corbeille à messages : offre un moyen de passer en revue les notifications envoyées à GNOME par les applications ou des composants du système. Si une notification est envoyée, elle s'affiche brièvement sous la forme d'une simple ligne en haut de l'écran, et un indicateur apparaît au milieu de la barre supérieure, à côté de l'horloge, pour informer l'utilisateur des notifications reçues récemment. Il est possible d'ouvrir la corbeille à messages pour examiner ces notifications en cliquant sur l'horloge dans la barre supérieure ou en appuyant sur **Super+M**. Pour fermer la barre des notifications, cliquez sur l'horloge dans la barre supérieure ou en appuyant sur **Échap** ou à nouveau sur **Super+M**.

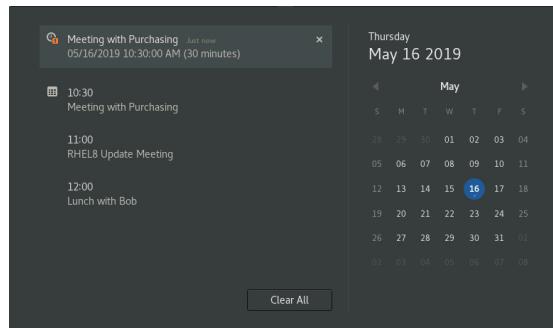


Figure 2.2: Gros plan sur une corbeille à messages ouverte

Vous pouvez afficher et modifier les raccourcis clavier GNOME utilisés par votre compte. Ouvrez le menu système sur le côté droit de la barre supérieure. Cliquez sur le bouton Settings en bas du menu à gauche. Dans la fenêtre de l'application qui s'ouvre, sélectionnez Devices → Keyboard à partir du volet de gauche. Le volet de droite affiche vos paramètres de raccourci actuels.

**NOTE**

Il peut s'avérer difficile d'envoyer certains raccourcis clavier, tels que les touches de fonction ou la touche **Super**, à une machine virtuelle. Cela est dû au fait que les frappes de touche spéciales utilisées par ces raccourcis peuvent être capturées par votre système d'exploitation local ou par l'application que vous utilisez pour accéder au bureau graphique de votre machine virtuelle.



IMPORTANT

L'utilisation de la touche **Super** n'est pas toujours simple dans les environnements de formation adaptée au rythme de chacun et de formation virtuelle actuels Red Hat. Vous ne pouvez probablement pas simplement utiliser la touche **Super** du clavier car il n'est généralement pas transmis à la machine virtuelle dans l'environnement de formation par votre navigateur Web.

Dans la barre Ravello Systems en haut de la fenêtre de votre navigateur, une icône de clavier devrait apparaître à droite. Si vous cliquez dessus, un clavier s'affiche à l'écran. Cliquez à nouveau pour le fermer.

Le clavier Ravello à l'écran traite la touche **Super** comme une touche de modification qui est souvent maintenue enfoncée tout en appuyant sur une autre touche. Si vous cliquez dessus une fois, elle devient jaune pour indiquer que la touche est maintenue enfoncée. Alors, pour saisir **Super+M** dans le clavier Ravello à l'écran, cliquez sur **Super** puis sur **M**.

Si vous voulez simplement appuyer sur la touche **Super** et la relâcher sur le clavier Ravello à l'écran, cliquez deux fois dessus. Le premier clic permet de maintenir la touche **Super** et le deuxième clic la libère.

Les autres touches traitées comme des touches de modification (comme **Super**) par le clavier Ravello à l'écran sont **Maj**, **Ctrl**, **Alt** et **Caps**. Les touche **Échap** et **Menu** sont traitées comme des touches normales et *non* comme des touches de modification.

ESPACES DE TRAVAIL

Les espaces de travail sont des écrans de bureau séparés qui contiennent différentes fenêtres d'application. Ils peuvent servir à organiser l'environnement de travail en regroupant par tâche les fenêtres d'applications ouvertes. Par exemple, les fenêtres utilisées pour effectuer une activité particulière de maintenance du système (comme la configuration d'un nouveau serveur distant) peuvent être regroupées dans un espace de travail unique, tandis que les applications de messagerie et autres applications de communication peuvent être regroupées dans un autre espace de travail.

Il existe deux méthodes simples pour passer d'un espace de travail à l'autre. Une méthode, sûrement plus rapide, consiste à appuyer sur **Ctrl+Alt+Flèche vers le haut** ou **Ctrl+Alt+Flèche vers le bas** pour changer d'espace de travail de manière séquentielle. La deuxième méthode consiste à basculer vers la vue d'ensemble Activities et à cliquer sur l'espace de travail souhaité.

L'utilisation de la vue d'ensemble Activities présente un avantage : on peut cliquer sur les fenêtres et les faire glisser entre les espaces de travail, en utilisant le sélecteur d'espace de travail sur le côté droit de l'écran et la vue d'ensemble des fenêtres au centre de l'écran.

**IMPORTANT**

Comme la touche **Super** dans les environnements de formation adaptée au rythme de chacun et de formation virtuelle actuels Red Hat, les combinaisons de touches **Ctrl+Alt** ne sont généralement pas transmises à la machine virtuelle dans l'environnement de formation par votre navigateur Web.

Vous pouvez saisir ces combinaisons de touches pour changer d'espace de travail à l'aide du clavier Ravello à l'écran. Au moins deux espaces de travail doivent être utilisés. Ouvrez le clavier Ravello à l'écran et cliquez sur **Ctrl, Alt**, puis soit sur la **Flèche vers le haut** soit sur la **Flèche vers le bas**.

Cependant, dans ces environnements de formation, il est généralement plus simple d'éviter d'utiliser des raccourcis clavier et le clavier Ravello à l'écran. Changez d'espace de travail en cliquant sur le bouton **Activities** puis, dans le sélecteur d'espace de travail situé à droite de la vue d'ensemble des activités, cliquez sur l'espace de travail vers lequel vous souhaitez basculer.

LANCLEMENT D'UN TERMINAL

Pour lancer une invite de shell dans GNOME, démarrez une application de terminal graphique telle que GNOME Terminal. Il y a plusieurs manières de procéder. Les deux méthodes les plus couramment utilisées sont énumérées ci-dessous :

- Dans la vue d'ensemble Activities, sélectionnez Terminal depuis le lanceur d'applications (soit depuis la zone des favoris, soit en le recherchant avec le bouton grille (dans le regroupement Utilities) ou le champ de recherche en haut de la vue d'ensemble des fenêtres).
- Appuyez sur la combinaison de touches **Alt+F2** pour ouvrir la zone Enter a Command et entrez **gnome-terminal**.

Lorsqu'une fenêtre de terminal est ouverte, une invite de shell s'affiche pour l'utilisateur qui a lancé le programme de terminal graphique. L'invite de shell et la barre de titre de la fenêtre de terminal indiquent le nom de l'utilisateur, le nom de l'hôte et le répertoire de travail actuels.

VERROUILLAGE DE L'ÉCRAN OU DÉCONNEXION

Le verrouillage de l'écran ou la déconnexion peuvent être effectués à partir du menu système, à l'extrême droite de la barre supérieure.

Pour verrouiller l'écran, à partir du menu système situé dans l'angle supérieur droit, cliquez sur le bouton de verrouillage en bas du menu ou appuyez sur **Super+L** (ce qui pourrait peut-être plus facile à retenir que **Windows+L**). L'écran se verrouille également si une session graphique reste inactive pendant quelques minutes.

Un rideau de verrouillage d'écran apparaît et affiche l'heure du système et le nom de l'utilisateur connecté. Pour déverrouiller l'écran, appuyez sur **Entrée** ou **Espace** pour lever le rideau de verrouillage d'écran, puis entrez le mot de passe de l'utilisateur sur l'écran de verrouillage.

Pour vous déconnecter et fermer la session graphique actuelle, sélectionnez le menu système dans l'angle supérieur droit de la barre supérieure et sélectionnez (User) → Log Out. Une fenêtre s'affiche, offrant la possibilité d'annuler (Cancel) ou de confirmer l'action Log Out.

ARRÊT ET REDÉMARRAGE DU SYSTÈME

Pour arrêter le système, dans le menu système situé dans l'angle supérieur droit, cliquez sur le bouton d'alimentation en bas du menu ou appuyez sur **Ctrl+Alt+Suppr**. Dans la boîte de dialogue qui s'affiche, vous pouvez choisir d'éteindre (Power Off) la machine, de la redémarrer (Restart) ou d'annuler (Cancel) l'opération. Si vous n'effectuez aucun choix, le système s'arrête automatiquement au bout de 60 secondes.



RÉFÉRENCES

Aide de GNOME

- **yelp**
Aide de GNOME : *Getting Started with GNOME*
- **yelp help:gnome-help/getting-started**

► EXERCICE GUIDÉ

ACCÈS À LA LIGNE DE COMMANDE DEPUIS LE BUREAU

Dans cet atelier, vous allez vous connecter au moyen du gestionnaire d'affichage graphique en tant qu'utilisateur standard pour vous familiariser avec l'environnement de bureau GNOME Standard fourni par GNOME 3.

RÉSULTATS

Vous devez pouvoir vous connecter à un système Linux en utilisant l'environnement de bureau GNOME 3 et exécuter des commandes depuis l'invite du shell dans un programme de terminal.

AVANT DE COMMENCER

Assurez-vous que la machine virtuelle **workstation** est en cours d'exécution. Effectuez les tâches suivantes sur **workstation**.

- ▶ 1. Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.
 - 1.1. Sur **workstation**, dans l'écran de connexion GNOME, cliquez sur le compte d'utilisateur **student**. Saisissez **student** lorsque le mot de passe vous est demandé.
 - 1.2. Cliquez sur **Sign In**.
- ▶ 2. Remplacez le mot de passe **student** de l'utilisateur **student** par **55TurnK3y**.



IMPORTANT

Le script **finish** réinitialise le mot de passe de l'utilisateur **student** en **student**.
Le script doit être exécuté à la fin de l'exercice.

- 2.1. L'approche la plus simple consiste à ouvrir une fenêtre Terminal et à utiliser la commande **passwd** à l'invite du shell.
Dans l'environnement d'apprentissage virtuel avec clavier visuel, appuyez deux fois sur la touche **Super** pour ouvrir la vue d'ensemble Activities. Tapez **Terminal** et appuyez sur **Entrée** pour lancer Terminal.
- 2.2. Dans la fenêtre de terminal qui s'affiche, tapez **passwd** à l'invite du shell. Remplacez le mot de passe **student** de l'utilisateur **student** par **55TurnK3y**.

```
[student@workstation ~]$ passwd
Changing password for user student.
Current password: student
New password: 55TurnK3y
Retype new password: 55TurnK3y
passwd: all authentication tokens updated successfully.
```

CHAPITRE 2 | Accès à la ligne de commande

- 3. Déconnectez-vous et reconnectez-vous en tant que **student** avec le mot de passe **55TurnK3y** pour vérifier le mot de passe modifié.
- 3.1. Cliquez sur le menu système dans l'angle supérieur droit.
 - 3.2. Sélectionnez Student User → Log Out.
 - 3.3. Cliquez sur Log Out dans la boîte de dialogue de confirmation qui s'affiche.
 - 3.4. Dans l'écran de connexion GNOME, cliquez sur le compte d'utilisateur **student**. Saisissez **55TurnK3y** lorsque le mot de passe vous est demandé.
 - 3.5. Cliquez sur Sign In.
- 4. Verrouillez l'écran.
- 4.1. Depuis le menu système dans l'angle supérieur droit, appuyez sur le bouton de verrouillage de l'écran en bas du menu.
- 5. Déverrouillez l'écran.
- 5.1. Appuyez sur **Entrée** pour lever le rideau de verrouillage de l'écran.
 - 5.2. Dans le champ Password, saisissez **55TurnK3y** comme mot de passe.
 - 5.3. Cliquez sur **Unlock**.
- 6. Choisissez la manière d'arrêter **workstation** à partir de l'interface graphique, mais annulez (Cancel) l'opération sans arrêter le système.
- 6.1. Depuis le menu système dans l'angle supérieur droit, cliquez sur le bouton d'alimentation en bas du menu. Une boîte de dialogue s'affiche avec les options permettant de redémarrer (Restart) ou d'éteindre (Power Off) la machine.
 - 6.2. Cliquez sur **Cancel** dans la boîte de dialogue qui s'affiche.

Fin

Sur **workstation**, exécutez le script **lab cli-desktop finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab cli-desktop finish
```

L'exercice guidé est maintenant terminé.

EXÉCUTION DE COMMANDES À L'AIDE DU SHELL BASH

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir gagner du temps lors de l'exécution de commandes depuis l'invite du shell, grâce aux raccourcis de bash.

SYNTAXE DES COMMANDES DE BASE

Le GNU Bourne-Again Shell (**bash**) est un programme qui interprète les commandes saisies par l'utilisateur. Chaque chaîne saisie dans le shell peut comporter trois parties au maximum : la commande, les options (qui commencent généralement par - ou --) et les arguments. Chaque mot saisi dans le shell est séparé des autres par des espaces. Les commandes correspondent aux noms des programmes installés sur le système. Chaque commande possède ses propres options et arguments.

Lorsque vous êtes prêt à exécuter une commande, appuyez sur la touche **Entrée**. Tapez chaque commande sur une ligne séparée. Le résultat de la commande est affiché avant l'invite du shell suivante.

```
[user@host]$ whoami  
user  
[user@host]$
```

Si vous souhaitez saisir plusieurs commandes sur une seule ligne, utilisez un point-virgule (;) pour les séparer. Le point-virgule fait partie de la classe des caractères nommés *métacaractères* qui ont une signification spéciale pour **bash**. Dans ce cas, le résultat des deux commandes est affiché avant l'invite du shell suivante.

L'exemple suivant montre comment combiner deux commandes (**command1** et **command2**) sur la ligne de commande.

```
[user@host]$ command1;command2
```

EXEMPLES DE COMMANDES SIMPLES

La commande **date** sert à afficher la date et l'heure actuelles. Elle peut également être utilisée par le super utilisateur pour régler l'horloge du système. Un argument qui commence par un signe plus (+) spécifie un gabarit pour la commande date.

```
[user@host ~]$ date  
Sat Jan 26 08:13:50 IST 2019  
[user@host ~]$ date +%R  
08:13  
[user@host ~]$ date +%x  
01/26/2019
```

CHAPITRE 2 | Accès à la ligne de commande

La commande **passwd** modifie le mot de passe d'un utilisateur. Le mot de passe d'origine du compte doit être spécifié avant que le changement soit autorisé. Par défaut, **passwd** est configuré pour exiger un mot de passe fort, composé de minuscules, de majuscules, de chiffres et de symboles, et qui n'est pas basé sur un mot du dictionnaire. Le super utilisateur peut se servir de la commande **passwd** pour changer le mot de passe des autres utilisateurs.

```
[user@host ~]$ passwd
Changing password for user user.
Current password: old_password
New password: new_password
Retype new password: new_password
passwd: all authentication tokens updated successfully.
```

Linux ne nécessite pas d'extensions de nom de fichier pour classifier les fichiers par type. La commande **file** analyse le début du contenu d'un fichier et affiche son type. Les fichiers à classer sont transmis à la commande en tant qu'arguments.

```
[user@host ~]$ file /etc/passwd
/etc/passwd: ASCII text
[user@host ~]$ file /bin/passwd
/bin/passwd: setuid ELF 64-bit LSB shared object, x86-64, version 1
(SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2,
for GNU/Linux 3.2.0, BuildID[sha1]=a3637110e27e9a48dc9f38b4ae43388d32d0e4,
stripped
[user@host ~]$ file /home
/home: directory
```

AFFICHAGE DU CONTENU DES FICHIERS

L'une des commandes les plus simples et les plus utilisées sous Linux est **cat**. La commande **cat** vous permet de créer un ou plusieurs fichiers, d'afficher le contenu des fichiers, de concaténer le contenu de plusieurs fichiers et de rediriger le contenu du fichier vers un terminal ou des fichiers.

L'exemple montre comment afficher le contenu du fichier **/etc/passwd**.

```
[user@host ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
...output omitted...
```

Utilisez la commande suivante pour afficher le contenu de plusieurs fichiers.

```
[user@host ~]$ cat file1 file2
Hello World!!
Introduction to Linux commands.
```

Certains fichiers sont très longs et leur affichage peut nécessiter davantage d'espace que le terminal. La commande **cat** n'affiche pas le contenu d'un fichier sous forme de pages. La commande **less** affiche une page d'un fichier à la fois et vous permet de la faire défiler à votre guise.

CHAPITRE 2 | Accès à la ligne de commande

La commande **less** permet de faire défiler vers le bas ou vers le haut des fichiers plus longs que le contenu affichable dans une fenêtre de terminal. Utilisez les touches **Flèche vers le haut** et **Flèche vers le bas** pour faire défiler vers le haut ou vers le bas. Appuyez sur **q** pour quitter la commande.

Les commandes **head** et **tail** affichent le début et la fin d'un fichier, respectivement. Par défaut, ces commandes affichent 10 lignes du fichier, mais elles disposent toutes deux d'une option **-n** qui permet de spécifier un nombre de lignes différent. Le fichier à afficher est transmis à ces commandes en tant qu'argument.

```
[user@host ~]$ head /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
[user@host ~]$ tail -n 3 /etc/passwd
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:977:977::/run/gnome-initial-setup/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
```

La commande **wc** compte les lignes, mots et caractères d'un fichier. Elle accepte une option **-l**, **-w** ou **-c** pour n'afficher respectivement que le nombre de lignes, de mots ou de caractères.

```
[user@host ~]$ wc /etc/passwd
45 102 2480 /etc/passwd
[user@host ~]$ wc -l /etc/passwd ; wc -l /etc/group
45 /etc/passwd
70 /etc/group
[user@host ~]$ wc -c /etc/group /etc/hosts
966 /etc/group
516 /etc/hosts
1482 total
```

SAISIE SEMI-AUTOMATIQUE PAR TABULATION

La *saisie semi-automatique par tabulation* vous permet de compléter rapidement les commandes et les noms de fichiers après que vous avez entré le nombre de caractères qui réduit les possibilités à une seule. Si les caractères saisis ne sont pas uniques, appuyez deux fois sur la touche de **tabulation** pour afficher toutes les commandes qui commencent par les caractères déjà saisis.

```
[user@host ~]$ pas①Tab+Tab
passwd      paste      pasususpend
[user@host ~]$ pass②Tab
[user@host ~]$ passwd
Changing password for user user.
Current password:
```

CHAPITRE 2 | Accès à la ligne de commande

- ➊ Appuyez deux fois sur la touche de **tabulation**.
- ➋ Appuyez une fois sur la touche de **tabulation**.

La saisie semi-automatique par tabulation peut servir à compléter les noms de fichiers quand on les saisit comme arguments de commandes. Quand on appuie sur la touche de **tabulation**, celle-ci complète le nom du fichier autant qu'il est possible. Une seconde pression sur la touche de **tabulation** entraîne l'affichage par le shell d'une liste de tous les fichiers qui correspondent au schéma courant. Tapez des caractères supplémentaires jusqu'à ce que le nom soit unique, puis utilisez la saisie semi-automatique par tabulation pour compléter la commande.

```
[user@host ~]$ ls /etc/pas❶Tab  
[user@host ~]$ ls /etc/passwd❷Tab  
passwd  passwd-
```

- ➊ ➋ Appuyez une fois sur la touche de **tabulation**.

La saisie semi-automatique par tabulation peut être utilisée avec les arguments et les options de nombreuses commandes. La commande **useradd** sert au super utilisateur, **root**, pour créer d'autres utilisateurs dans le système. Elle dispose de nombreuses options qui peuvent servir à contrôler le comportement de la commande. On peut utiliser la saisie semi-automatique par tabulation après une option partielle pour la compléter sans avoir à taper trop de texte.

```
[root@host ~]# useradd --❶Tab+Tab  
--base-dir      --groups          --no-log-init    --shell  
--comment       --help            --non-unique     --skel  
--create-home   --home-dir        --no-user-group --system  
--defaults      --inactive        --password      --uid  
--expiredate   --key             --root          --user-group  
--gid           --no-create-home --selinux-user  
[root@host ~]# useradd --
```

- ➊ Appuyez deux fois sur la touche de **tabulation**.

AFFICHAGE D'UNE LONGUE COMMANDE SUR PLUSIEURS LIGNES

Les commandes comportant de nombreuses options et arguments peuvent rapidement devenir longues et sont automatiquement entourées par la fenêtre de commande lorsque le curseur atteint la marge de droite. Pour faciliter la lecture des commandes, vous pouvez taper une commande longue en utilisant plusieurs lignes.

Pour ce faire, utilisez une barre oblique inverse (\), aussi appelée caractère d'*échappement*, pour ignorer la signification du caractère qui suit immédiatement la barre oblique inverse. Vous avez appris que la saisie d'un saut de ligne, en appuyant sur la touche **Entrée**, indique au shell que la saisie de la commande est terminée et qu'elle doit être exécutée. En échappant le saut de ligne, le shell est invité à passer à une nouvelle ligne de commande sans exécuter la commande. Le shell accuse réception de la demande en affichant une invite de continuation, appelée *invite secondaire*, en utilisant le caractère supérieur à (>) par défaut, sur une nouvelle ligne vide. Les commandes peuvent être saisies sur plusieurs lignes.

```
[user@host]$ head -n 3 \  
> /usr/share/dict/words \  
> /usr/share/dict/linux.words  
==> /usr/share/dict/words ==
```

CHAPITRE 2 | Accès à la ligne de commande

```
1080
10-point
10th

==> /usr/share/dict/linux.words <=
1080
10-point
10th
[user@host ~]$
```



IMPORTANT

L'exemple d'écran précédent montre comment une commande continue apparaît pour un utilisateur typique. Cependant, la présentation en conditions réelles dans des supports d'apprentissage, tels que ce manuel, est généralement source de confusion. Les nouveaux apprenants peuvent insérer par erreur le caractère supérieur à dans le cadre de la commande tapée. Le shell interprète la saisie d'un caractère supérieur à comme une *redirection de processus* non prévue par l'utilisateur. La redirection de processus est décrite dans un chapitre ultérieur.

Pour éviter cette confusion, ce manuel de cours ne montre pas les invites secondaires dans une sortie écran. Un utilisateur voit toujours l'invite secondaire dans la fenêtre shell, mais le contenu du cours n'affiche intentionnellement que les caractères à saisir, comme illustré dans l'exemple ci-dessous. Comparez avec l'exemple d'écran précédent.

```
[user@host]$ head -n 3 \
/usr/share/dict/words \
/usr/share/dict/linux.words
==> /usr/share/dict/words <=
1080
10-point
10th

==> /usr/share/dict/linux.words <=
1080
10-point
10th
[user@host ~]$
```

HISTORIQUE DES COMMANDES

La commande **history** affiche la liste des commandes précédemment exécutées, précédées d'un numéro de commande.

Le point d'exclamation (!) est un métacaractère qui sert à rappeler les commandes précédentes sans avoir à les retaper. La commande **!number** rappelle la commande qui correspond au nombre spécifié. La commande **!string** rappelle la commande la plus récente qui commence par la chaîne spécifiée.

```
[user@host ~]$ history
...output omitted...
23 clear
```

CHAPITRE 2 | Accès à la ligne de commande

```

24 who
25 pwd
26 ls /etc
27 uptime
28 ls -l
29 date
30 history
[user@host ~]$ !ls
ls -l
total 0
drwxr-xr-x. 2 user user 6 Mar 29 21:16 Desktop
...output omitted...
[user@host ~]$ !26
ls /etc
abrt hosts pulse
adjtime hosts.allow purple
aliases hosts.deny qemu-ga
...output omitted...

```

Les touches fléchées peuvent être utilisées pour naviguer dans les commandes précédentes de l'historique du shell. La **Flèche vers le haut** modifie la commande précédente dans la liste de l'historique. La **Flèche vers le bas** modifie la commande suivante dans la liste d'historique. La **Flèche gauche** et la **Flèche droite** déplacent le curseur vers la gauche et la droite dans la commande en cours à partir de la liste de l'historique afin de vous permettre de la modifier avant de l'exécuter.

Vous pouvez utiliser les combinaisons de touches **Échap+.** ou **Alt+.** pour insérer le dernier mot de la commande précédente à l'emplacement actuel du curseur. L'utilisation répétée de la combinaison de touches remplacera ce texte par le dernier mot de commandes plus antérieures dans l'historique. La combinaison de touches **Alt+.** est particulièrement pratique, car vous pouvez maintenir **Alt** enfonce et appuyer sur **.** à plusieurs reprises pour parcourir facilement les commandes précédentes et plus antérieures.

MODIFICATION DE LA LIGNE DE COMMANDE

Quand on l'utilise de manière interactive, **bash** offre des fonctions de modification de la ligne de commande. À l'aide des commandes d'éditeur de texte, l'utilisateur peut se déplacer dans les commandes en cours de saisie et les modifier. L'utilisation des touches de direction pour se déplacer dans la commande courante et pour parcourir l'historique des commandes a été présentée plus tôt au cours de cette session. Des commandes d'édition plus puissantes sont présentées dans le tableau suivant.

Raccourcis utiles de modification de ligne de commande

RACCOURCI	DESCRIPTION
Ctrl+A	Passe au début de la ligne de commande.
Ctrl+E	Passe à la fin de la ligne de commande.
Ctrl+U	Efface le texte entre le début de la ligne de commande et le curseur.
Ctrl+K	Efface le texte entre le curseur et la fin de la ligne de commande.

RACCOURCI	DESCRIPTION
Ctrl +Flèche gauche	Passe au début du mot précédent sur la ligne de commande.
Ctrl+Flèche droite	Passe à la fin du mot suivant sur la ligne de commande.
Ctrl+R	Recherche une chaîne de caractères dans l'historique des commandes.

D'autres commandes de modification de la ligne de commande sont disponibles, mais celles-ci sont les plus utiles pour les nouveaux utilisateurs. Les autres commandes sont décrites dans la page de manuel **bash(1)**.



RÉFÉRENCES

Pages de manuel **bash(1)**, **date(1)**, **file(1)**, **cat(1)**, **more(1)**, **less(1)**, **head(1)**, **passwd(1)**, **tail(1)** et **wc(1)**

► QUIZ

EXÉCUTION DE COMMANDES À L'AIDE DU SHELL BASH

Répondez aux questions suivantes en sélectionnant un ou plusieurs éléments :

- ▶ 1. Quel raccourci ou commande bash permet de revenir au début du mot précédent sur la ligne de commande ?
 - a. Combinaison de touches **Ctrl+Flèche gauche**
 - b. Combinaison de touches **Ctrl+K**
 - c. Combinaison de touches **Ctrl+A**
 - d. **!string**
 - e. **!number**

- ▶ 2. Quel raccourci ou commande bash sépare les commandes sur la même ligne ?
 - a. Touche de **tabulation**
 - b. **history**
 - c. ;
 - d. **!string**
 - e. Combinaison de touches **Échap+**.

- ▶ 3. Quel raccourci ou commande bash est utilisé pour effacer les caractères du curseur jusqu'à la fin de la ligne de commande ?
 - a. Combinaison de touches **Ctrl+Flèche gauche**
 - b. Combinaison de touches **Ctrl+K**
 - c. Combinaison de touches **Ctrl+A**
 - d. ;
 - e. Combinaison de touches **Échap+**.

- ▶ 4. Quel raccourci ou commande bash est utilisé pour exécuter à nouveau une commande récente en faisant correspondre le nom de la commande ?
 - a. Touche de **tabulation**
 - b. **!number**
 - c. **!string**
 - d. **history**
 - e. Combinaison de touches **Échap+**.

- 5. Quel raccourci ou commande bash permet de compléter les commandes, les noms de fichiers et les options ?
- a. ;
 - b. **!number**
 - c. **history**
 - d. Touche de **tabulation**
 - e. Combinaison de touches **Échap+**.
- 6. Quel raccourci ou commande bash exécute à nouveau une commande spécifique dans la liste de l'historique ?
- a. Touche de **tabulation**
 - b. **!number**
 - c. **!string**
 - d. **history**
 - e. Combinaison de touches **Échap+**.
- 7. Quel raccourci ou commande bash permet de passer au début de la ligne de commande ?
- a. **!number**
 - b. **!string**
 - c. Combinaison de touches **Ctrl+Flèche gauche**
 - d. Combinaison de touches **Ctrl+K**
 - e. Combinaison de touches **Ctrl+A**
- 8. Quel raccourci ou commande bash affiche la liste des commandes précédentes ?
- a. Touche de **tabulation**
 - b. **!string**
 - c. **!number**
 - d. **history**
 - e. Combinaison de touches **Échap+**.
- 9. Quel raccourci ou commande bash copie le dernier argument des commandes précédentes ?
- a. Combinaison de touches **Ctrl+K**
 - b. Combinaison de touches **Ctrl+A**
 - c. **!number**
 - d. Combinaison de touches **Échap+**.

► SOLUTION

EXÉCUTION DE COMMANDES À L'AIDE DU SHELL BASH

Répondez aux questions suivantes en sélectionnant un ou plusieurs éléments :

- ▶ 1. Quel raccourci ou commande bash permet de revenir au début du mot précédent sur la ligne de commande ?
 - a. Combinaison de touches **Ctrl+Flèche gauche**
 - b. Combinaison de touches **Ctrl+K**
 - c. Combinaison de touches **Ctrl+A**
 - d. **!string**
 - e. **!number**
- ▶ 2. Quel raccourci ou commande bash sépare les commandes sur la même ligne ?
 - a. Touche de **tabulation**
 - b. **history**
 - c. ;
 - d. **!string**
 - e. Combinaison de touches **Échap+**.
- ▶ 3. Quel raccourci ou commande bash est utilisé pour effacer les caractères du curseur jusqu'à la fin de la ligne de commande ?
 - a. Combinaison de touches **Ctrl+Flèche gauche**
 - b. Combinaison de touches **Ctrl+K**
 - c. Combinaison de touches **Ctrl+A**
 - d. ;
 - e. Combinaison de touches **Échap+**.
- ▶ 4. Quel raccourci ou commande bash est utilisé pour exécuter à nouveau une commande récente en faisant correspondre le nom de la commande ?
 - a. Touche de **tabulation**
 - b. **!number**
 - c. **!string**
 - d. **history**
 - e. Combinaison de touches **Échap+**.

- 5. Quel raccourci ou commande bash permet de compléter les commandes, les noms de fichiers et les options ?
- a. ;
 - b. *!number*
 - c. history
 - d. Touche de **tabulation**
 - e. Combinaison de touches Échap+.
- 6. Quel raccourci ou commande bash exécute à nouveau une commande spécifique dans la liste de l'historique ?
- a. Touche de **tabulation**
 - b. *!number*
 - c. *!string*
 - d. history
 - e. Combinaison de touches Échap+.
- 7. Quel raccourci ou commande bash permet de passer au début de la ligne de commande ?
- a. *!number*
 - b. *!string*
 - c. Combinaison de touches **Ctrl+F**lèche gauche
 - d. Combinaison de touches **Ctrl+K**
 - e. Combinaison de touches **Ctrl+A**
- 8. Quel raccourci ou commande bash affiche la liste des commandes précédentes ?
- a. Touche de **tabulation**
 - b. *!string*
 - c. *!number*
 - d. history
 - e. Combinaison de touches Échap+.
- 9. Quel raccourci ou commande bash copie le dernier argument des commandes précédentes ?
- a. Combinaison de touches **Ctrl+K**
 - b. Combinaison de touches **Ctrl+A**
 - c. *!number*
 - d. Combinaison de touches Échap+.

► OPEN LAB

ACCÈS À LA LIGNE DE COMMANDE

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez utiliser le shell bash pour exécuter les commandes.

RÉSULTATS

- Exécutez des programmes simples à l'aide de la ligne de commande du shell bash.
- Exécutez les commandes utilisées pour identifier les types de fichiers et afficher des portions de fichiers texte.
- Entraînez-vous à utiliser certains « raccourcis » de l'historique des commandes bash pour répéter plus efficacement des commandes ou des parties de commandes.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez le script **lab cli-review start** pour configurer un nouvel environnement d'atelier. Le script copie le fichier **zcat** dans le répertoire personnel de l'utilisateur **student**.

```
[student@workstation ~]$ lab cli-review start
```

1. Utilisez la commande **date** pour afficher la date et l'heure actuelles.
2. Affichez l'heure actuelle sur une horloge de douze heures (par exemple, 11:42:11 AM). Conseil : la chaîne de format qui affiche cette sortie est **%r**.
3. Quel est le type du fichier **/home/student/zcat** ? Est-il lisible par des humains ?
4. Utilisez la commande **wc** et les raccourcis bash pour afficher la taille de **zcat**.
5. Affichez les 10 premières lignes de **zcat**.
6. Affichez les 10 dernières lignes du fichier **zcat**.
7. Répétez la commande précédente avec exactement trois frappes de touches ou moins.
8. Répétez la commande précédente, mais utilisez l'option **-n 20** pour afficher les 20 dernières lignes du fichier. Utilisez la modification de ligne de commande pour accomplir cela avec un nombre minimal de frappes de touches.
9. Utilisez l'historique du shell pour exécuter à nouveau la commande **date +%r**.

Évaluation

À partir de **workstation**, exécutez le script **lab cli-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab cli-review grade
```

Fin

Sur workstation, exéutez le script **lab cli-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab cli-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

ACCÈS À LA LIGNE DE COMMANDE

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez utiliser le shell bash pour exécuter les commandes.

RÉSULTATS

- Exécutez des programmes simples à l'aide de la ligne de commande du shell bash.
- Exécutez les commandes utilisées pour identifier les types de fichiers et afficher des portions de fichiers texte.
- Entraînez-vous à utiliser certains « raccourcis » de l'historique des commandes bash pour répéter plus efficacement des commandes ou des parties de commandes.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez le script `lab cli-review start` pour configurer un nouvel environnement d'atelier. Le script copie le fichier `zcat` dans le répertoire personnel de l'utilisateur `student`.

```
[student@workstation ~]$ lab cli-review start
```

- Utilisez la commande `date` pour afficher la date et l'heure actuelles.

```
[student@workstation ~]$ date
Thu Jan 22 10:13:04 PDT 2019
```

- Affichez l'heure actuelle sur une horloge de douze heures (par exemple, 11:42:11 AM). Conseil : la chaîne de format qui affiche cette sortie est `%r`. Utilisez l'argument `+%r` avec la commande `date` pour afficher l'heure actuelle sur une horloge de douze heures.

```
[student@workstation ~]$ date +%r
10:14:07 AM
```

- Quel est le type du fichier `/home/student/zcat`? Est-il lisible par des humains ? Utilisez la commande `file` pour déterminer son type.

```
[student@workstation ~]$ file zcat
zcat: POSIX shell script, ASCII text executable
```

- Utilisez la commande `wc` et les raccourcis bash pour afficher la taille de `zcat`.

CHAPITRE 2 | Accès à la ligne de commande

La commande **wc** peut être utilisée pour afficher le nombre de lignes, de mots et d'octets dans le script **zcat**. Au lieu de retaper le nom du fichier, utilisez le raccourci de l'historique bash **Échap+**. (la combinaison de touches **Échap** et **.**) pour réutiliser l'argument de la commande précédente.

```
[student@workstation ~]$ wc Esc+.
[student@workstation ~]$ wc zcat
 51  299 1983 zcat
```

5. Affichez les 10 premières lignes de **zcat**.

La commande **head** affiche le début du fichier. Essayez d'utiliser à nouveau le raccourci **Échap+..**

```
[student@workstation ~]$ head Esc+.
[student@workstation ~]$ head zcat
#!/bin/sh
# Uncompress files to standard output.

# Copyright (C) 2007, 2010-2018 Free Software Foundation, Inc.

# This program is free software; you can redistribute it and/or modify
# it under the terms of the GNU General Public License as published by
# the Free Software Foundation; either version 3 of the License, or
# (at your option) any later version.
```

6. Affichez les 10 dernières lignes du fichier **zcat**.

Utilisez la commande **tail** pour afficher les 10 dernières lignes du fichier **zcat**.

```
[student@workstation ~]$ tail Esc+.
[student@workstation ~]$ tail zcat
With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>.

case $1 in
--help)    printf '%s\n' "$usage"  || exit 1;;
--version) printf '%s\n' "$version" || exit 1;;
esac

exec gzip -cd "$@"
```

7. Répétez la commande précédente avec exactement trois frappes de touches ou moins.

Répétez la commande précédente exactement. Appuyez une fois sur la touche **Flèche vers le haut** pour revenir d'une commande en arrière dans l'historique des commandes et appuyez sur **Entrée** (utilisation de deux frappes de touches), ou entrez la commande de raccourci **!!** et appuyez sur **Entrée** (utilisation de trois frappes de touches) pour exécuter la commande la plus récente de l'historique des commandes. (Essayez les deux.)

```
[student@workstation]$ !!
tail zcat
With no FILE, or when FILE is -, read standard input.
```

CHAPITRE 2 | Accès à la ligne de commande

```
Report bugs to <bug-gzip@gnu.org>.

case $1 in
--help)   printf '%s\n' "$usage"  || exit 1;;
--version) printf '%s\n' "$version" || exit 1;;
esac

exec gzip -cd "$@"

```

- 8.** Répétez la commande précédente, mais utilisez l'option **-n 20** pour afficher les 20 dernières lignes du fichier. Utilisez la modification de ligne de commande pour accomplir cela avec un nombre minimal de frappes de touches.
- La **Flèche vers le haut** affiche la commande précédente. **Ctrl+A** place le curseur au début de la ligne. **Ctrl+Flèche droite** passe au mot suivant, puis ajoutez l'option **-n 20** et appuyez sur **Entrée** pour exécuter la commande.

```
[student@workstation ~]$ tail -n 20 zcat
-1, --list      list compressed file contents
-q, --quiet     suppress all warnings
-r, --recursive operate recursively on directories
-S, --suffix=SUF use suffix SUF on compressed files
    --synchronous synchronous output (safer if system crashes, but slower)
-t, --test       test compressed file integrity
-v, --verbose    verbose mode
    --help        display this help and exit
    --version     display version information and exit

With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>.

case $1 in
--help)   printf '%s\n' "$usage"  || exit 1; exit;;
--version) printf '%s\n' "$version" || exit 1; exit;;
esac

exec gzip -cd "$@"

```

- 9.** Utilisez l'historique du shell pour exécuter à nouveau la commande **date +%r**.
- Utilisez la commande **history** pour afficher la liste des commandes précédentes et pour identifier la commande **date** spécifique à exécuter. Utilisez **!number** pour exécuter la commande, où *number* est le numéro de commande à utiliser de la sortie de la commande **history**.
- Notez que l'historique de votre shell peut être différent de l'exemple suivant. Déterminez le numéro de commande à utiliser en fonction de la sortie de votre propre commande **history**.

```
[student@workstation ~]$ history
1  date
2  date +%r
3  file zcat
4  wc zcat
5  head zcat
```

```
6 tail zcat
7 tail -n 20 zcat
8 history
[student@workstation ~]$ !2
date +%r
10:49:56 AM
```

Évaluation

À partir de **workstation**, exécutez le script lab cli-review grade pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab cli-review grade
```

Fin

Sur workstation, exécutez le script **lab cli-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab cli-review finish
```

L'atelier est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Le shell bash est un interpréteur de commandes qui invite les utilisateurs interactifs à spécifier des commandes Linux.
- De nombreuses commandes ont une option **--help** qui affiche un message d'utilisation ou un écran.
- L'utilisation d'espaces de travail facilite l'organisation de plusieurs fenêtres d'application.
- Le bouton Activities situé dans l'angle supérieur gauche de la barre supérieure fournit une vue d'ensemble qui permet à l'utilisateur d'organiser les fenêtres et de démarrer les applications.
- La commande **file** analyse le début du contenu d'un fichier et affiche son type.
- Les commandes **head** et **tail** affichent le début et la fin d'un fichier, respectivement.
- Vous pouvez utiliser la saisie semi-automatique par **tabulation** pour compléter les noms de fichiers quand vous les saisissez comme arguments de commandes.

CHAPITRE 3

GESTION DE FICHIERS À PARTIR DE LA LIGNE DE COMMANDE

PROJET

Copier, déplacer, créer, supprimer et organiser les fichiers depuis le shell bash.

OBJECTIFS

- Décrire comment Linux organise les fichiers, et l'objet des divers répertoires dans la hiérarchie du système de fichiers.
- Spécifier l'emplacement des fichiers par rapport au répertoire de travail actuel et par emplacement absolu, déterminer et modifier votre répertoire de travail et lister le contenu des répertoires.
- Créer, copier, déplacer et supprimer des fichiers et des répertoires.
- Faire en sorte que plusieurs noms de fichiers référencent le même fichier en utilisant des liens fixes et symboliques.
- Exécuter efficacement les commandes qui affectent de nombreux fichiers en utilisant les fonctionnalités de filtrage par motif du shell bash.

SECTIONS

- Description des concepts de hiérarchie du système de fichiers Linux (et quiz)
- Spécification des fichiers par nom (et quiz)
- Gestion des fichiers avec les outils de ligne de commande (et exercice guidé)
- Création de liens entre fichiers (et exercice guidé)
- Correspondance des noms de fichiers à l'aide des extensions par le shell (et quiz)

ATELIER

Gestion de fichiers à partir de la ligne de commande

DESCRIPTION DES CONCEPTS DE HIÉRARCHIE DU SYSTÈME DE FICHIERS LINUX

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir décrire comment Linux organise les fichiers, et l'objet des divers répertoires dans la hiérarchie du système de fichiers.

HIÉRARCHIE DU SYSTÈME DE FICHIERS

Tous les fichiers d'un système Linux sont stockés sur des systèmes de fichiers organisés dans une arborescence de répertoires inversée unique appelée *hiérarchie du système de fichiers*. On dit que cette arborescence est inversée, car sa racine se trouve en haut de la hiérarchie, et les branches des répertoires et sous-répertoires s'étendent sous la racine.

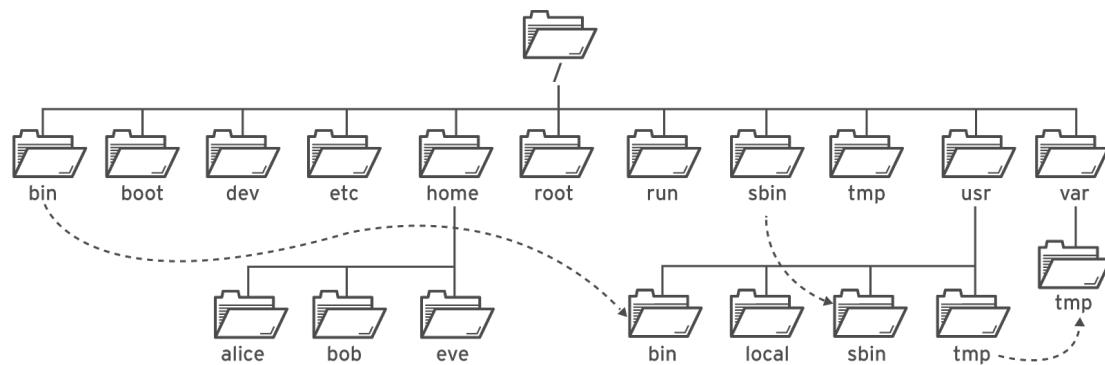


Figure 3.1: Répertoires significatifs d'un système de fichiers dans Red Hat Enterprise Linux 8

Le répertoire `/` est le répertoire racine situé au sommet de la hiérarchie du système de fichiers. Le caractère `/` est également utilisé comme séparateur de répertoires dans les noms de fichiers. Par exemple, si `etc` est un sous-répertoire du répertoire `/`, vous pouvez faire référence à ce fichier sous le nom `/etc`. De même, si le répertoire `/etc` contient un fichier nommé `issue`, vous pouvez faire référence à ce fichier sous le nom `/etc/issue`.

Les sous-répertoires de `/` sont utilisés à des fins de standardisation, pour organiser les fichiers par type et par utilisation. Cela facilite la recherche des fichiers. Par exemple, dans le répertoire racine (root), le sous-répertoire `/boot` est utilisé pour le stockage des fichiers nécessaires pour démarrer le système.

**NOTE**

Les termes suivants aident à décrire le contenu du répertoire du système de fichiers :

- *statique* : le contenu reste inchangé jusqu'à ce qu'il soit explicitement modifié ou reconfiguré.
- *dynamique* ou *variable* : le contenu est généralement modifié ou complété par des processus actifs.
- *persistent* : contenu qui persiste après un redémarrage, en particulier les paramètres de configuration.
- *exécution* : le contenu est un contenu spécifique au processus ou au système supprimé lors d'un redémarrage.

Le tableau ci-dessous répertorie certains des répertoires les plus importants du système par nom et par fonction.

Répertoires importants de Red Hat Enterprise Linux

EMPLACEMENT	OBJECTIF
/usr	Logiciels installés, bibliothèques partagées, y compris les fichiers, et données des programmes, en lecture seule. Parmi les sous-répertoires importants figurent : <ul style="list-style-type: none"> • /usr/bin : commandes utilisateur. • /usr/sbin : commandes d'administration système. • /usr/local : logiciels personnalisés localement.
/etc	Fichiers de configuration spécifiques à ce système.
/var	Données variables spécifiques à ce système, qui doivent persister d'un démarrage à l'autre. Les fichiers qui changent de manière dynamique, tels que les bases de données, les répertoires cache, les fichiers journaux, les documents du spool d'impression, le contenu des sites Web se trouvent dans /var .
/run	Données d'exécution des processus démarrés depuis le dernier démarrage. Cela comprend les fichiers d'identification et de verrouillage des processus, entre autres. Le contenu de ce répertoire est recréé au redémarrage. Ce répertoire consolide /var/run et /var/lock des versions antérieures de Red Hat Enterprise Linux.
/home	Répertoires personnels où les utilisateurs standard stockent leurs données personnelles et leurs fichiers de configuration.
/root	Répertoire personnel du super utilisateur administratif, root .

EMPLACEMENT	OBJECTIF
/tmp	Espace ouvert à tous pour les fichiers temporaires. Les fichiers qui n'ont pas été ouverts, changés ou modifiés depuis 10 jours sont automatiquement supprimés de ce répertoire. Il existe un autre répertoire temporaire, /var/tmp , dans lequel les fichiers qui n'ont pas été ouverts, changés ou modifiés depuis plus de 30 jours sont automatiquement supprimés.
/boot	Fichiers nécessaires au lancement du processus de démarrage.
/dev	Contient les <i>fichiers de périphériques</i> spéciaux, utilisés par le système pour accéder au matériel.



IMPORTANT

Dans Red Hat Enterprise Linux 7, quatre anciens répertoires de **/** ont désormais un contenu identique à celui de leurs homologues de **/usr** :

- **/bin** et **/usr/bin**
- **/sbin** et **/usr/sbin**
- **/lib** et **/usr/lib**
- **/lib64** et **/usr/lib64**

Dans les anciennes versions de Red Hat Enterprise Linux, il s'agissait de répertoires distincts qui contenaient des ensembles de fichiers différents.

Dans Red Hat Enterprise Linux 7 et versions ultérieures, les répertoires dans **/** sont des liens symboliques vers les répertoires correspondants dans **/usr**.



RÉFÉRENCES

Page de manuel **hier(7)**

La page de fonctionnalité UsrMove de Fedora 17

<https://fedoraproject.org/wiki/Features/UsrMove>

► QUIZ

DESCRIPTION DES CONCEPTS DE HIÉRARCHIE DU SYSTÈME DE FICHIERS LINUX

Choisissez les réponses aux questions suivantes :

- ▶ 1. Quel répertoire contient des données de configuration persistantes et spécifiques au système ?
 - a. /etc
 - b. /root
 - c. /run
 - d. /usr

- ▶ 2. Quel répertoire est le plus haut de la hiérarchie du système de fichiers du système ?
 - a. /etc
 - b. /
 - c. /home/root
 - d. /root

- ▶ 3. Quel répertoire contient les répertoires personnels des utilisateurs ?
 - a. /
 - b. /home
 - c. /root
 - d. /user

- ▶ 4. Quel répertoire contient des fichiers temporaires ?
 - a. /tmp
 - b. /trash
 - c. /run
 - d. /var

- ▶ 5. Quel répertoire contient des données dynamiques, tels que pour les bases de données et les sites Web ?
 - a. /etc
 - b. /run
 - c. /usr
 - d. /var

► **6. Quel répertoire est le répertoire personnel du super utilisateur administratif ?**

- a. /etc
- b. /
- c. /home/root
- d. /root

► **7. Quel répertoire contient les commandes et les utilitaires habituels ?**

- a. /commands
- b. /run
- c. /usr/bin
- d. /usr/sbin

► **8. Quel répertoire contient des données d'exécution de processus non persistantes ?**

- a. /tmp
- b. /etc
- c. /run
- d. /var

► **9. Quel répertoire contient les programmes et les bibliothèques des logiciels installés ?**

- a. /etc
- b. /lib
- c. /usr
- d. /var

► SOLUTION

DESCRIPTION DES CONCEPTS DE HIÉRARCHIE DU SYSTÈME DE FICHIERS LINUX

Choisissez les réponses aux questions suivantes :

- ▶ 1. Quel répertoire contient des données de configuration persistantes et spécifiques au système ?
 - a. /etc
 - b. /root
 - c. /run
 - d. /usr

- ▶ 2. Quel répertoire est le plus haut de la hiérarchie du système de fichiers du système ?
 - a. /etc
 - b. /
 - c. /home/root
 - d. /root

- ▶ 3. Quel répertoire contient les répertoires personnels des utilisateurs ?
 - a. /
 - b. /home
 - c. /root
 - d. /user

- ▶ 4. Quel répertoire contient des fichiers temporaires ?
 - a. /tmp
 - b. /trash
 - c. /run
 - d. /var

- ▶ 5. Quel répertoire contient des données dynamiques, tels que pour les bases de données et les sites Web ?
 - a. /etc
 - b. /run
 - c. /usr
 - d. /var

► **6. Quel répertoire est le répertoire personnel du super utilisateur administratif ?**

- a. /etc
- b. /
- c. /home/root
- d. /root

► **7. Quel répertoire contient les commandes et les utilitaires habituels ?**

- a. /commands
- b. /run
- c. /usr/bin
- d. /usr/sbin

► **8. Quel répertoire contient des données d'exécution de processus non persistantes ?**

- a. /tmp
- b. /etc
- c. /run
- d. /var

► **9. Quel répertoire contient les programmes et les bibliothèques des logiciels installés ?**

- a. /etc
- b. /lib
- c. /usr
- d. /var

SPÉCIFICATION DES FICHIERS PAR NOM

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir spécifier l'emplacement des fichiers par rapport au répertoire de travail actuel et par emplacement absolu, déterminer et modifier le répertoire de travail et répertorier le contenu des répertoires.

CHEMINS ABSOLUS ET RELATIFS

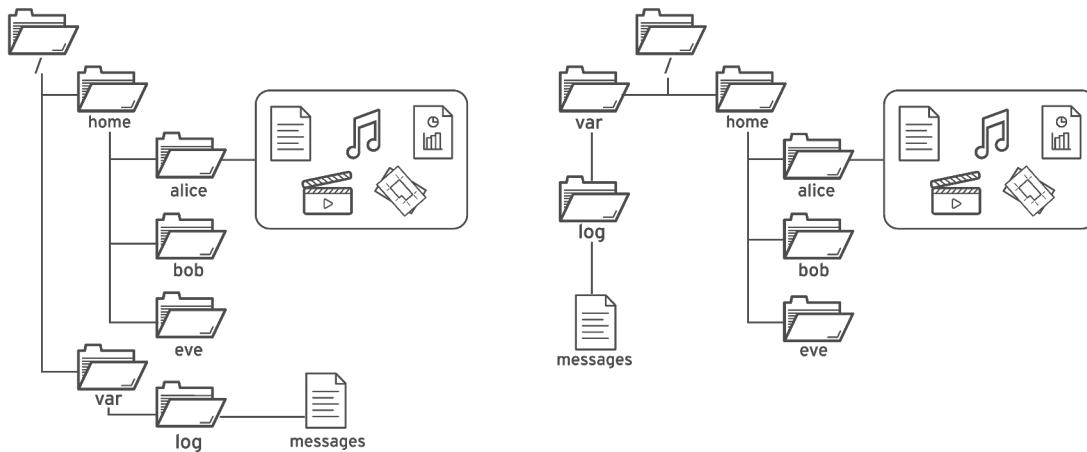


Figure 3.2: La vue classique du navigateur de fichiers (à gauche) est équivalente à la vue de haut en bas (à droite).

Le *chemin* d'un fichier ou d'un répertoire spécifie son emplacement unique au sein d'un système de fichiers. En suivant un chemin de fichier, on traverse un ou plusieurs sous-répertoires nommés, délimités par des barres obliques (/). Les répertoires, également appelés *dossiers*, contiennent d'autres fichiers et sous-répertoires. Ils peuvent être référencés de la même manière que les fichiers.



IMPORTANT

Un caractère d'espacement est acceptable en tant que partie d'un nom de fichier Linux. Cependant, le shell utilise également des espaces pour séparer les options et les arguments sur la ligne de commande. Si vous entrez une commande qui inclut un fichier dont le nom contient un espace, le shell peut mal interpréter la commande et supposer que vous souhaitez démarrer un nouveau nom de fichier ou un autre argument sur cet espace.

Il est possible d'éviter cela en mettant les noms de fichiers entre guillemets. Toutefois, si vous n'avez pas besoin d'utiliser des espaces dans les noms de fichiers, vous pouvez simplement éviter de les utiliser.

Chemins absolus

Un *chemin absolu* est un nom *complet*, spécifiant l'emplacement exact des fichiers dans la hiérarchie du système de fichiers. Il commence au répertoire root (/) et spécifie chaque sous-

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

répertoire à parcourir pour atteindre le fichier spécifique. Chaque fichier d'un système de fichiers possède un nom de chemin absolu unique, reconnaissable à une règle simple : un nom de chemin qui commence par une barre oblique (/) est un nom de chemin absolu. Par exemple, le nom de chemin absolu pour le fichier du journal des messages du système est **/var/log/messages**. Comme les noms de chemins absolus peuvent être longs à taper, on peut aussi localiser les fichiers *par rapport* au répertoire de travail actuel de votre invite du shell.

Répertoire de travail actuel et chemins relatifs

Lorsqu'un utilisateur se connecte et ouvre une fenêtre de commande, l'emplacement initial est normalement son répertoire personnel. Les processus système ont également un répertoire initial. Les utilisateurs et les processus parcourront les répertoires au besoin ; les termes *répertoire de travail* ou *répertoire de travail actuel* font référence à leur emplacement *actuel*.

Tout comme un chemin absolu, un *chemin relatif* identifie un fichier unique, ne spécifiant que le chemin nécessaire pour atteindre le fichier depuis le répertoire de travail. L'identification des noms de chemins relatifs suit une règle simple : un nom de chemin qui commence par un caractère quelconque, *autre que* la barre oblique, est un nom de chemin relatif. Un utilisateur dans le répertoire **/var** pourrait se référer au fichier journal des messages de cette façon : **log/messages**.

Les systèmes de fichiers Linux, y compris mais sans s'y limiter, ext4, XFS, BTRFS, GFS2 et GlusterFS, sont sensibles à la casse. Créer **FileCase.txt** et **filecase.txt** dans le même répertoire génère deux fichiers uniques.

Les systèmes de fichiers non-Linux peuvent fonctionner différemment. Par exemple, VFAT, NTFS de Microsoft et HFS+ d'Apple présentent un comportement de *conservation de la casse*. Bien que ces systèmes de fichiers ne soient *pas* sensibles à la casse, ils affichent les noms de fichiers avec la majuscule d'origine utilisée lors de la création du fichier. Par conséquent, si vous avez essayé de créer les fichiers de l'exemple précédent sur un système de fichiers VFAT, les deux noms seraient traités comme pointant vers le même fichier au lieu de deux fichiers différents.

CHEMINS DE NAVIGATION

La commande **pwd** affiche le nom de chemin complet du répertoire de travail actuel pour ce shell. Cela peut vous aider à déterminer la syntaxe pour accéder aux fichiers en utilisant des noms de chemin relatifs. La commande **ls** affiche la liste du contenu du répertoire spécifié ou, si aucun répertoire n'est indiqué, celle du répertoire actuel.

```
[user@host ~]$ pwd  
/home/user  
[user@host ~]$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos  
[user@host ~]$
```

Utilisez la commande **cd** pour changer le répertoire de travail actuel de votre shell. Si vous ne spécifiez aucun argument pour la commande, elle sera remplacée par votre répertoire de base.

Dans l'exemple suivant, un mélange de chemins absolus et relatifs est utilisé avec la commande **cd** pour modifier le répertoire de travail actuel du shell.

```
[user@host ~]$ pwd  
/home/user  
[user@host ~]$ cd Videos  
[user@host Videos]$ pwd
```

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

```
/home/user/Videos  
[user@host Videos]$ cd /home/user/Documents  
[user@host Documents]$ pwd  
/home/user/Documents  
[user@host Documents]$ cd  
[user@host ~]$ pwd  
/home/user  
[user@host ~]$
```

Comme vous pouvez le voir dans l'exemple précédent, l'invite par défaut de shell affiche également le dernier composant du chemin absolu vers le répertoire de travail actuel. Par exemple, pour **/home/user/Videos**, seul **Videos** est affiché. L'invite affiche le caractère tilde (~) lorsque votre répertoire de travail actuel est votre répertoire personnel.

Normalement, la commande **touch** met à jour l'horodatage d'un fichier à la date et à l'heure actuelles, sans rien y modifier. Cette commande est utile pour créer des fichiers vides, qui peuvent être utilisés pour des exercices pratiques, puisque l'exécution de la commande touch sur un nom de fichier qui n'existe pas entraîne la création de celui-ci. Dans l'exemple suivant, la commande **touch** crée des fichiers d'exercice pratique dans les sous-répertoires **Documents** et **Videos**.

```
[user@host ~]$ touch Videos/blockbuster1.ogg  
[user@host ~]$ touch Videos/blockbuster2.ogg  
[user@host ~]$ touch Documents/thesis_chapter1.odf  
[user@host ~]$ touch Documents/thesis_chapter2.odf  
[user@host ~]$
```

La commande **ls** comporte plusieurs options pour afficher les attributs des fichiers. Les options les plus courantes et les plus utiles sont **-l** (format de liste long), **-a** (tous les fichiers, y compris les fichiers cachés), et **-R** (récuratif, pour inclure le contenu de tous les sous-répertoires).

```
[user@host ~]$ ls -l  
total 15  
drwxr-xr-x. 2 user user 4096 Feb 7 14:02 Desktop  
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Documents  
drwxr-xr-x. 3 user user 4096 Jan 9 15:00 Downloads  
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Music  
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Pictures  
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Public  
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Templates  
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Videos  
[user@host ~]$ ls -la  
total 15  
drwx----- 16 user user 4096 Feb 8 16:15 .  
drwxr-xr-x. 6 root root 4096 Feb 8 16:13 ..  
-rw----- 1 user user 22664 Feb 8 00:37 .bash_history  
-rw-r--r-- 1 user user 18 Jul 9 2013 .bash_logout  
-rw-r--r-- 1 user user 176 Jul 9 2013 .bash_profile  
-rw-r--r-- 1 user user 124 Jul 9 2013 .bashrc  
drwxr-xr-x. 4 user user 4096 Jan 20 14:02 .cache  
drwxr-xr-x. 8 user user 4096 Feb 5 11:45 .config  
drwxr-xr-x. 2 user user 4096 Feb 7 14:02 Desktop  
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Documents  
drwxr-xr-x. 3 user user 4096 Jan 25 20:48 Downloads  
drwxr-xr-x. 11 user user 4096 Feb 6 13:07 .gnome2
```

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

```
drwx----- 2 user user 4096 Jan 20 14:02 .gnome2_private  
-rw----- 1 user user 15190 Feb 8 09:49 .ICEauthority  
drwxr-xr-x 3 user user 4096 Jan 9 15:00 .local  
drwxr-xr-x 2 user user 4096 Jan 9 15:00 Music  
drwxr-xr-x 2 user user 4096 Jan 9 15:00 Pictures  
drwxr-xr-x 2 user user 4096 Jan 9 15:00 Public  
drwxr-xr-x 2 user user 4096 Jan 9 15:00 Templates  
drwxr-xr-x 2 user user 4096 Jan 9 15:00 Videos  
[user@host ~]$
```

Les deux répertoires spéciaux en début de liste font référence au répertoire courant (.) et au répertoire parent (...). Ces répertoires spéciaux existent dans chaque répertoire du système. Vous découvrirez leur utilité lorsque vous commencerez à utiliser les commandes de gestion de fichiers.

**IMPORTANT**

Les noms de fichiers qui commencent par un point (.) désignent des fichiers *cachés* de la vue normale à l'aide de **ls** et d'autres commandes. Il ne s'agit pas d'une fonction de sécurité. Les fichiers cachés évitent que les fichiers de configuration nécessaires à l'utilisateur n'encombrent les répertoires personnels. De nombreuses commandes ne traitent les fichiers cachés qu'avec des options de ligne de commande, ce qui évite la copie accidentelle de la configuration d'un utilisateur vers d'autres répertoires ou utilisateurs.

Pour protéger la consultation indue du *contenu* d'un fichier, il faut recourir aux *permissions de fichier*.

```
[user@host ~]$ ls -R  
. :  
Desktop Documents Downloads Music Pictures Public Templates Videos  
  
. /Desktop:  
  
. /Documents:  
thesis_chapter1.odf thesis_chapter2.odf  
  
. /Downloads:  
  
. /Music:  
  
. /Pictures:  
  
. /Public:  
  
. /Templates:  
  
. /Videos:  
blockbuster1.ogg blockbuster2.ogg  
[user@host ~]$
```

La commande **cd** comporte de nombreuses options. Quelques-unes sont suffisamment utiles pour qu'il vaille la peine de les maîtriser rapidement et les utiliser souvent. La commande **cd -** sert à passer du répertoire où se trouvait l'utilisateur *avant* au répertoire actuel. L'exemple suivant

illustre ce comportement, en passant d'un répertoire à l'autre, ce qui est utile lors du traitement d'une série de tâches similaires.

```
[user@host ~]$ cd Videos
[user@host Videos]$ pwd
/home/user/Videos
[user@host Videos]$ cd /home/user/Documents
[user@host Documents]$ pwd
/home/user/Documents
[user@host Documents]$ cd -
[user@host Videos]$ pwd
/home/user/Videos
[user@host Videos]$ cd -
[user@host Documents]$ pwd
/home/user/Documents
[user@host Documents]$ cd -
[user@host Videos]$ pwd
/home/user/Videos
[user@host Videos]$ cd
[user@host ~]$
```

La commande **cd ..** utilise le répertoire caché .. pour passer au niveau supérieur, *répertoire parent*, sans avoir à connaître le nom exact du parent. L'autre répertoire caché (.) spécifie le *répertoire courant* pour les commandes dans lesquelles l'emplacement actuel constitue l'argument source ou de destination, ce qui évite d'avoir à saisir le nom de chemin absolu du répertoire.

```
[user@host Videos]$ pwd
/home/user/Videos
[user@host Videos]$ cd .
[user@host Videos]$ pwd
/home/user/Videos
[user@host Videos]$ cd ..
[user@host ~]$ pwd
/home/user
[user@host ~]$ cd ..
[user@host home]$ pwd
/home
[user@host home]$ cd ..
[user@host /]$ pwd
/
[user@host /]$ cd
[user@host ~]$ pwd
/home/user
[user@host ~]$
```



RÉFÉRENCES

info libc 'file name resolution' (*Manuel de référence de la bibliothèque GNU C Library*)

- Section 11.2.2 : résolution des noms de fichier

Pages du manuel **bash(1)**, **cd(1)**, **ls(1)**, **pwd(1)**, **unicode(7)** et **utf-8(7)**

UTF-8 et Unicode

<http://www.utf-8.com/>

► QUIZ

SPÉCIFICATION DES FICHIERS PAR NOM

Choisissez les réponses aux questions suivantes :

- ▶ 1. Quelle commande est utilisée pour revenir au répertoire personnel de l'utilisateur actuel, en supposant que le répertoire de travail actuel est /tmp et son répertoire personnel est /home/user ?
 - a. **cd**
 - b. **cd ..**
 - c. **cd .**
 - d. **cd ***
 - e. **cd /home**

- ▶ 2. Quelle commande affiche le nom de chemin absolu de l'emplacement actuel ?
 - a. **cd**
 - b. **pwd**
 - c. **ls ~**
 - d. **ls -d**

- ▶ 3. Quelle commande vous ramènera toujours au répertoire de travail utilisé avant le répertoire de travail actuel ?
 - a. **cd -**
 - b. **cd -p**
 - c. **cd ~**
 - d. **cd ..**

- ▶ 4. Quelle commande modifiera toujours le répertoire de travail de deux niveaux par rapport à l'emplacement actuel ?
 - a. **cd ~**
 - b. **cd ../**
 - c. **cd ../../..**
 - d. **cd -u2**

- ▶ 5. Quelle commande liste les fichiers dans l'emplacement actuel, en utilisant un format long et en incluant les fichiers cachés ?
 - a. **llong ~**
 - b. **ls -a**
 - c. **ls -l**
 - d. **ls -al**

- 6. Quelle commande modifiera toujours le répertoire de travail en /bin?
- a. **cd bin**
 - b. **cd /bin**
 - c. **cd ~bin**
 - d. **cd -bin**
- 7. Quelle commande modifiera toujours le répertoire de travail en répertoire parent de l'emplacement actuel ?
- a. **cd ~**
 - b. **cd ..**
 - c. **cd ../..**
 - d. **cd -u1**
- 8. Quelle commande modifiera le répertoire de travail en /tmp si le répertoire de travail actuel est /home/student ?
- a. **cd tmp**
 - b. **cd ..**
 - c. **cd ../../tmp**
 - d. **cd ~tmp**

► SOLUTION

SPÉCIFICATION DES FICHIERS PAR NOM

Choisissez les réponses aux questions suivantes :

- ▶ 1. Quelle commande est utilisée pour revenir au répertoire personnel de l'utilisateur actuel, en supposant que le répertoire de travail actuel est /tmp et son répertoire personnel est /home/user ?
 - a. cd
 - b. cd ..
 - c. cd .
 - d. cd *
 - e. cd /home

- ▶ 2. Quelle commande affiche le nom de chemin absolu de l'emplacement actuel ?
 - a. cd
 - b. pwd
 - c. ls ~
 - d. ls -d

- ▶ 3. Quelle commande vous ramènera toujours au répertoire de travail utilisé avant le répertoire de travail actuel ?
 - a. cd -
 - b. cd -p
 - c. cd ~
 - d. cd ..

- ▶ 4. Quelle commande modifiera toujours le répertoire de travail de deux niveaux par rapport à l'emplacement actuel ?
 - a. cd ~
 - b. cd ../
 - c. cd .../..
 - d. cd -u2

- ▶ 5. Quelle commande liste les fichiers dans l'emplacement actuel, en utilisant un format long et en incluant les fichiers cachés ?
 - a. llong ~
 - b. ls -a
 - c. ls -l
 - d. ls -al

- 6. Quelle commande modifiera toujours le répertoire de travail en /bin?
- a. cd bin
 - b. **cd /bin**
 - c. cd ~bin
 - d. cd -bin
- 7. Quelle commande modifiera toujours le répertoire de travail en répertoire parent de l'emplacement actuel ?
- a. cd ~
 - b. **cd ..**
 - c. cd ../..
 - d. cd -u1
- 8. Quelle commande modifiera le répertoire de travail en /tmp si le répertoire de travail actuel est /home/student ?
- a. cd tmp
 - b. cd ..
 - c. **cd ../../tmp**
 - d. cd ~tmp

GESTION DES FICHIERS À L'AIDE DES OUTILS DE LIGNE DE COMMANDE

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir créer, copier, déplacer et supprimer des fichiers et des sous-répertoires.

GESTION DES FICHIERS EN LIGNE DE COMMANDE

Pour gérer les fichiers, vous devez pouvoir les créer, les supprimer, les copier et les déplacer. Vous devez également les organiser logiquement dans des répertoires que vous devez également pouvoir créer, supprimer, copier et déplacer.

Le tableau suivant récapitule certaines des commandes de gestion de fichiers les plus courantes. Le reste de cette section traitera plus en détails des manières d'utiliser ces commandes.

Commandes de gestion de fichiers courantes

ACTIVITÉ	SYNTAXE DES COMMANDES
Créer un répertoire	<code>mkdir directory</code>
Copier un fichier	<code>cp file new-file</code>
Copier un répertoire et son contenu	<code>cp -r directory new-directory</code>
Déplacer ou renommer un fichier ou un répertoire.	<code>mv file new-file</code>
Supprimer un fichier	<code>rm file</code>
Supprimer un répertoire contenant des fichiers	<code>rm -r directory</code>
Supprimer un répertoire vide	<code>rmdir directory</code>

Création de répertoires

La commande `mkdir` crée un ou plusieurs répertoires ou sous-répertoires. Elle prend comme arguments une liste de chemins d'accès aux répertoires que vous voulez créer.

La commande `mkdir` échouera avec une erreur si le répertoire existe déjà, ou si vous essayez de créer un sous-répertoire dans un répertoire qui n'existe pas. L'option `-p (parent)` crée les répertoires parents manquants pour la destination demandée. Utilisez la commande `mkdir -p` avec prudence, car des fautes de frappe accidentelles peuvent créer des répertoires non souhaités, sans pour autant générer de messages d'erreur.

Dans l'exemple suivant, supposons que vous essayez de créer un répertoire dans le répertoire **Videos** nommé **Watched**, mais que vous avez accidentellement oublié la lettre "s" dans **Videos** dans votre commande `mkdir`.

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

```
[user@host ~]$ mkdir Video/Watched  
mkdir: cannot create directory `Video/Watched': No such file or directory
```

La commande **mkdir** a échoué parce que le nom **Videos** a été mal orthographié et le répertoire **Video** n'existe pas. Si vous aviez utilisé la commande **mkdir** avec l'option **-p**, le répertoire **Video** serait créé, ce qui n'était pas votre intention, et le sous-répertoire **Watched** serait créé dans ce répertoire incorrect.

Une fois le répertoire parent **Videos** correctement orthographié, la création du sous-répertoire **Watched** doit aboutir.

```
[user@host ~]$ mkdir Videos/Watched  
[user@host ~]$ ls -R Videos  
Videos/:  
blockbuster1.ogg blockbuster2.ogg Watched  
  
Videos/Watched:
```

Dans l'exemple suivant, les fichiers et les répertoires sont organisés sous le répertoire **/home/user/Documents**. Utilisez la commande **mkdir** et une liste de noms de répertoire séparés par des espaces pour créer plusieurs répertoires.

```
[user@host ~]$ cd Documents  
[user@host Documents]$ mkdir ProjectX ProjectY  
[user@host Documents]$ ls  
ProjectX ProjectY
```

Utilisez la commande **mkdir -p** et des chemins relatifs de noms de sous-répertoires séparés par des espaces pour créer plusieurs répertoires parents contenant des sous-répertoires.

```
[user@host Documents]$ mkdir -p Thesis/Chapter1 Thesis/Chapter2 Thesis/Chapter3  
[user@host Documents]$ cd  
[user@host ~]$ ls -R Videos Documents  
Documents:  
ProjectX ProjectY Thesis  
  
Documents/ProjectX:  
  
Documents/ProjectY:  
  
Documents/Thesis:  
Chapter1 Chapter2 Chapter3  
  
Documents/Thesis/Chapter1:  
  
Documents/Thesis/Chapter2:  
  
Documents/Thesis/Chapter3:  
  
Videos:  
blockbuster1.ogg blockbuster2.ogg Watched
```

Videos/Watched:

La dernière commande **mkdir** a créé trois sous-répertoires ChapterN avec une seule commande. L'option **-p** a créé le répertoire parent manquant **Thesis**.

Copie de fichiers

La commande **cp** copie un fichier et crée un fichier dans le répertoire actuel ou dans un répertoire spécifié. Elle peut également copier plusieurs fichiers dans un répertoire.



MISE EN GARDE

Si le fichier de destination existe déjà, la commande **cp** remplace le fichier.

```
[user@host ~]$ cd Videos
[user@host Videos]$ cp blockbuster1.ogg blockbuster3.ogg
[user@host Videos]$ ls -l
total 0
-rw-rw-r-- 1 user user    0 Feb  8 16:23 blockbuster1.ogg
-rw-rw-r-- 1 user user    0 Feb  8 16:24 blockbuster2.ogg
-rw-rw-r-- 1 user user    0 Feb  8 16:34 blockbuster3.ogg
drwxrwxr-x. 2 user user 4096 Feb  8 16:05 Watched
[user@host Videos]$
```

Lors de la copie de plusieurs fichiers avec une seule commande, le dernier argument doit spécifier un répertoire. Les fichiers copiés conservent leur nom d'origine dans le nouveau répertoire. Si un fichier portant le même nom existe dans le répertoire cible, le fichier existant est remplacé. Par défaut, la commande **cp** ne copie pas les répertoires ; elle les ignore.

Dans l'exemple suivant, deux répertoires sont listés, **Thesis** et **ProjectX**. Seul le dernier argument, **ProjectX**, est valide en tant que destination. Le répertoire **Thesis** est ignoré.

```
[user@host Videos]$ cd ../Documents
[user@host Documents]$ cp thesis_chapter1.odf thesis_chapter2.odf Thesis ProjectX
cp: omitting directory `Thesis'
[user@host Documents]$ ls Thesis ProjectX
ProjectX:
thesis_chapter1.odf  thesis_chapter2.odf

Thesis:
Chapter1  Chapter2  Chapter3
```

Dans la première commande **cp**, la copie du répertoire **Thesis** a échoué, mais la copie des fichiers **thesis_chapter1.odf** et **thesis_chapter2.odf** a fonctionné.

Si vous souhaitez copier un fichier dans le répertoire de travail actuel, vous pouvez utiliser le répertoire spécial **.** :

```
[user@host ~]$ cp /etc/hostname .
[user@host ~]$ cat hostname
host.example.com
[user@host ~]$
```

Utilisez la commande de copie avec l'option **-r** (récuratif) pour copier le répertoire **Thesis** et son contenu dans le répertoire **ProjectX**.

```
[user@host Documents]$ cp -r Thesis ProjectX
[user@host Documents]$ ls -R ProjectX
ProjectX:
Thesis  thesis_chapter1.odf  thesis_chapter2.odf

ProjectX/Thesis:
Chapter1  Chapter2  Chapter3

ProjectX/Thesis/Chapter1:

ProjectX/Thesis/Chapter2:
thesis_chapter2.odf

ProjectX/Thesis/Chapter3:
```

Déplacement des fichiers

La commande **mv** déplace les fichiers d'un emplacement à un autre. Si vous considérez le chemin absolu d'un fichier comme son nom complet, déplacer un fichier équivaut en fait à renommer un fichier. Le contenu du fichier reste inchangé.

Utilisez la commande **mv** pour renommer un fichier.

```
[user@host Videos]$ cd ..\Documents
[user@host Documents]$ ls -l thesis*
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter1.odf
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter2.odf
[user@host Documents]$ mv thesis_chapter2.odf thesis_chapter2_reviewed.odf
[user@host Documents]$ ls -l thesis*
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter1.odf
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter2_reviewed.odf
```

Utilisez la commande **mv** pour déplacer un fichier dans un répertoire différent.

```
[user@host Documents]$ ls Thesis/Chapter1
[user@host Documents]$
[user@host Documents]$ mv thesis_chapter1.odf Thesis/Chapter1
[user@host Documents]$ ls Thesis/Chapter1
thesis_chapter1.odf
[user@host Documents]$ ls -l thesis*
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter2_reviewed.odf
```

Suppression de fichiers et de répertoires

La commande **rm** supprime les fichiers. Par défaut, **rm** ne supprimera pas les répertoires contenant des fichiers, sauf si vous ajoutez l'option **-r** ou **--recursive**.



IMPORTANT

Il n'existe aucune fonction de ligne de commande pour annuler la suppression, ni de "corbeille" à partir de laquelle vous pouvez restaurer les fichiers supprimés.

Il est judicieux de vérifier votre répertoire de travail actuel avant de supprimer un fichier ou un répertoire.

```
[user@host Documents]$ pwd
/home/student/Documents
```

Utilisez la commande **rm** pour supprimer un seul fichier de votre répertoire de travail.

```
[user@host Documents]$ ls -l thesis*
-rw-rw-r-- 1 user user 0 Feb  6 21:16 thesis_chapter2_reviewed.odf
[user@host Documents]$ rm thesis_chapter2_reviewed.odf
[user@host Documents]$ ls -l thesis*
ls: cannot access 'thesis*': No such file or directory
```

Si vous essayez d'utiliser la commande **rm** pour supprimer un répertoire sans utiliser l'option **-r**, la commande échouera.

```
[user@host Documents]$ rm Thesis/Chapter1
rm: cannot remove `Thesis/Chapter1': Is a directory
```

Utilisez la commande **rm -r** pour supprimer un sous-répertoire et son contenu.

```
[user@host Documents]$ ls -R Thesis
Thesis/:
Chapter1 Chapter2 Chapter3

Thesis/Chapter1:
thesis_chapter1.odf

Thesis/Chapter2:
thesis_chapter2.odf

Thesis/Chapter3:
[user@host Documents]$ rm -r Thesis/Chapter1
[user@host Documents]$ ls -l Thesis
total 8
drwxrwxr-x. 2 user user 4096 Feb 11 12:47 Chapter2
drwxrwxr-x. 2 user user 4096 Feb 11 12:48 Chapter3
```

La commande **rm -r** parcourt d'abord chaque sous-répertoire, en supprimant individuellement leurs fichiers avant de supprimer chaque répertoire. Vous pouvez utiliser la commande **rm -ri**

pour demander de manière interactive une confirmation avant suppression. C'est essentiellement le contraire de l'option **-f** qui force la suppression sans demander de confirmation à l'utilisateur.

```
[user@host Documents]$ rm -ri Thesis
rm: descend into directory `Thesis'? y
rm: descend into directory `Thesis/Chapter2'? y
rm: remove regular empty file `Thesis/Chapter2/thesis_chapter2.odf'? y
rm: remove directory `Thesis/Chapter2'? y
rm: remove directory `Thesis/Chapter3'? y
rm: remove directory `Thesis'? y
[user@host Documents]$
```



MISE EN GARDE

Si vous spécifiez à la fois l'option **-i** et **-f**, l'option **-f** est prioritaire et aucune confirmation ne vous sera demandée avant que **rm** supprime les fichiers.

Dans l'exemple suivant, la commande **rmdir** supprime uniquement le répertoire vide. Comme dans l'exemple précédent, vous devez utiliser la commande **rm -r** pour supprimer un répertoire comprenant du contenu.

```
[user@host Documents]$ pwd
/home/student/Documents
[user@host Documents]$ rmdir ProjectY
[user@host Documents]$ rmdir ProjectX
rmdir: failed to remove `ProjectX': Directory not empty
[user@host Documents]$ rm -r ProjectX
[user@host Documents]$ ls -lR
.:
total 0
[user@host Documents]$
```



NOTE

La commande **rm** sans options ne peut pas supprimer un répertoire vide. Vous devez utiliser la commande **rmdir**, **rm -d** (qui équivaut à **rmdir**), ou **rm -r**.



RÉFÉRENCES

Pages du manuel **cp(1)**, **mkdir(1)**, **mv(1)**, **rm(1)** et **rmdir(1)**

► EXERCICE GUIDÉ

GESTION DES FICHIERS À L'AIDE DES OUTILS DE LIGNE DE COMMANDE

Dans cet exercice, vous allez créer, organiser, copier et supprimer des fichiers et des répertoires.

RÉSULTATS

Vous devez pouvoir créer, organiser, copier et supprimer des fichiers et des répertoires.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

Sur workstation, exécutez la commande **lab files-manage start**. Cette commande exécute un script de démarrage qui détermine si la machine servera est accessible sur le réseau.

```
[student@workstation ~]$ lab files-manage start
```

- 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Dans le répertoire personnel de l'utilisateur student, utilisez la commande **mkdir** pour créer trois sous-répertoires : **Music**, **Pictures** et **Videos**.

```
[student@servera ~]$ mkdir Music Pictures Videos
```

- 3. Toujours dans le répertoire personnel de l'utilisateur student, utilisez la commande **touch** pour créer des ensembles de fichiers d'exercices pratiques vides à utiliser pendant cet atelier.
- Créez six fichiers avec des noms du type **songX.mp3**.
 - Créez six fichiers avec des noms du type **snapX.jpg**.
 - Créez six fichiers avec des noms du type **filmX.avi**.

Dans chaque ensemble, remplacez X par les chiffres de 1 à 6.

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

```
[student@servera ~]$ touch song1.mp3 song2.mp3 song3.mp3 song4.mp3 \
song5.mp3 song6.mp3
[student@servera ~]$ touch snap1.jpg snap2.jpg snap3.jpg snap4.jpg \
snap5.jpg snap6.jpg
[student@servera ~]$ touch film1.avi film2.avi film3.avi film4.avi \
film5.avi film6.avi
[student@servera ~]$ ls -l
total 0
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film1.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film2.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film3.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film4.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film5.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film6.avi
drwxrwxr-x. 2 student student 6 Feb  4 18:23 Music
drwxrwxr-x. 2 student student 6 Feb  4 18:23 Pictures
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap1.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap2.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap3.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap4.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap5.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap6.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song1.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song2.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song3.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song4.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song5.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song6.mp3
drwxrwxr-x. 2 student student 6 Feb  4 18:23 Videos
```

- 4. Toujours dans le répertoire personnel de l'utilisateur **student**, déplacez les fichiers de musique dans le sous-répertoire **Music**, les fichiers de captures d'écran dans le sous-répertoire **Pictures** et les fichiers de films dans le sous-répertoire **Videos**.

Lorsque vous organisez des fichiers depuis un emplacement unique vers plusieurs emplacements, commencez par ouvrir le répertoire qui contient les fichiers sources. Utilisez la syntaxe de chemin la plus simple, absolue ou relative, pour atteindre la destination de chaque tâche de gestion de fichiers.

```
[student@servera ~]$ mv song1.mp3 song2.mp3 song3.mp3 song4.mp3 \
song5.mp3 song6.mp3 Music
[student@servera ~]$ mv snap1.jpg snap2.jpg snap3.jpg snap4.jpg \
snap5.jpg snap6.jpg Pictures
[student@servera ~]$ mv film1.avi film2.avi film3.avi film4.avi \
film5.avi film6.avi Videos
[student@servera ~]$ ls -l Music Pictures Videos
Music:
total 0
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song1.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song2.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song3.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song4.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song5.mp3
```

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

```
-rw-rw-r-- 1 student student 0 Feb  4 18:23 song6.mp3

Pictures:
total 0
-rw-rw-r-- 1 student student 0 Feb  4 18:23 snap1.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:23 snap2.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:23 snap3.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:23 snap4.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:23 snap5.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:23 snap6.jpg

Videos:
total 0
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film1.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film2.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film3.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film4.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film5.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film6.avi
```

- 5. Dans le répertoire personnel de l'utilisateur **student**, créez trois sous-répertoires pour organiser vos fichiers en projets. Nommez ces sous-répertoires **friends**, **family** et **work**. Utilisez une seule commande pour créer les trois sous-répertoires en même temps.
Vous utiliserez ces répertoires pour réorganiser vos fichiers en projets.

```
[student@servera ~]$ mkdir friends family work
[student@servera ~]$ ls -l
total 0
drwxrwxr-x. 2 student student   6 Feb  4 18:38 family
drwxrwxr-x. 2 student student   6 Feb  4 18:38 friends
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Music
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Pictures
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Videos
drwxrwxr-x. 2 student student   6 Feb  4 18:38 work
```

- 6. Copiez une sélection de nouveaux fichiers dans les répertoires de projets **family** et **friends**. Utilisez autant de commandes que nécessaire. Il ne vous est pas demandé d'utiliser une commande unique comme dans l'exemple. Pour chaque projet, commencez par ouvrir le répertoire du projet, puis copiez les fichiers sources dans ce répertoire. N'oubliez pas que vous effectuez des copies. Par conséquent, les fichiers d'origine resteront dans leur emplacement d'origine une fois les fichiers copiés dans les répertoires de projets.
- Copiez les fichiers (tous types) contenant les numéros 1 et 2 dans le sous-répertoire **friends**.
 - Copiez les fichiers (tous types) contenant les numéros 3 et 4 dans le sous-répertoire **family**.

Lorsque vous regroupez des fichiers provenant de plusieurs emplacements dans un emplacement unique, Red Hat vous recommande de vous placer dans le répertoire avant

d'y copier les fichiers. Utilisez la syntaxe de chemin la plus simple, absolue ou relative, pour atteindre la source de chaque tâche de gestion de fichiers.

```
[student@servera ~]$ cd friends
[student@servera friends]$ cp ~/Music/song1.mp3 ~/Music/song2.mp3 \
~/Pictures/snap1.jpg ~/Pictures/snap2.jpg ~/Videos/film1.avi \
~/Videos/film2.avi .
[student@servera friends]$ ls -l
total 0
-rw-rw-r--. 1 student student 0 Feb  4 18:42 film1.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:42 film2.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:42 snap1.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:42 snap2.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:42 song1.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:42 song2.mp3
[student@servera friends]$ cd ../family
[student@servera family]$ cp ~/Music/song3.mp3 ~/Music/song4.mp3 \
~/Pictures/snap3.jpg ~/Pictures/snap4.jpg ~/Videos/film3.avi \
~/Videos/film4.avi .
[student@servera family]$ ls -l
total 0
-rw-rw-r--. 1 student student 0 Feb  4 18:44 film3.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:44 film4.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:44 snap3.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:44 snap4.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:44 song3.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:44 song4.mp3
```

- ▶ 7. Pour votre projet « work », créez des copies supplémentaires.

```
[student@servera family]$ cd ../work
[student@servera work]$ cp ~/Music/song5.mp3 ~/Music/song6.mp3 \
~/Pictures/snap5.jpg ~/Pictures/snap6.jpg \
~/Videos/film5.avi ~/Videos/film6.avi .
[student@servera work]$ ls -l
total 0
-rw-rw-r--. 1 student student 0 Feb  4 18:48 film5.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:48 film6.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:48 snap5.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:48 snap6.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:48 song5.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:48 song6.mp3
```

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

- 8. Vos tâches de projets sont maintenant terminées et il est temps de nettoyer les projets. Ouvrez le répertoire personnel de l'utilisateur **student**. Essayez de supprimer les projets **family** et **friends** avec une seule commande **rmdir**.

```
[student@servera work]$ cd  
[student@servera ~]$ rm -r family friends  
rmdir: failed to remove 'family': Directory not empty  
rmdir: failed to remove 'friends': Directory not empty
```

L'utilisation de la commande **rmdir** doit échouer, car les deux sous-répertoires contiennent des fichiers.

- 9. Utilisez la commande **rm -r** pour supprimer de manière récursive les sous-répertoires **family** et **friends**, et leur contenu.

```
[student@servera ~]$ rm -r family friends  
[student@servera ~]$ ls -l  
total 0  
drwxrwxr-x. 2 student student 108 Feb 4 18:36 Music  
drwxrwxr-x. 2 student student 108 Feb 4 18:36 Pictures  
drwxrwxr-x. 2 student student 108 Feb 4 18:36 Videos  
drwxrwxr-x. 2 student student 108 Feb 4 18:48 work
```

- 10. Supprimez tous les fichiers du projet « **work** », mais ne supprimez pas le répertoire « **work** ».

```
[student@servera ~]$ cd work  
[student@servera work]$ rm song5.mp3 song6.mp3 snap5.jpg snap6.jpg \  
film5.avi film6.avi  
[student@servera work]$ ls -l  
total 0
```

- 11. Enfin, depuis le répertoire personnel de l'utilisateur **student**, utilisez la commande **rmdir** pour supprimer le répertoire **work**. La commande devrait s'exécuter correctement, maintenant que le répertoire est vide.

```
[student@servera work]$ cd  
[student@servera ~]$ rm -r work  
[student@servera ~]$ ls -l  
total 0  
drwxrwxr-x. 2 student student 108 Feb 4 18:36 Music  
drwxrwxr-x. 2 student student 108 Feb 4 18:36 Pictures  
drwxrwxr-x. 2 student student 108 Feb 4 18:36 Videos
```

- 12. Quittez **servera**.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Fin

Sur workstation, exéutez le script **lab files-manage finish** pour mettre fin à l'exercice. Ce script supprime tous les fichiers et répertoires créés pendant l'exercice.

```
[student@workstation ~]$ lab files-manage finish
```

L'exercice guidé est maintenant terminé.

CRÉATION DE LIENS ENTRE DES FICHIERS

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir faire en sorte que plusieurs noms de fichiers référencent le même fichier en utilisant des liens fixes et symboliques.

GESTION DES LIENS ENTRE DES FICHIERS

Liens matériels et liens symboliques

Il est possible de créer plusieurs noms qui pointent vers le même fichier. Il y a deux façons de le faire : en créant un *lien matériel* vers le fichier, ou en créant un *lien symbolique* (parfois appelé *lien logique*) au fichier. Chacun a ses avantages et ses inconvénients.

Création de liens matériels

Chaque fichier commence par un seul lien matériel, de son nom initial aux données du système de fichiers. Lorsque vous créez un nouveau lien matériel vers un fichier, vous créez un autre nom qui pointe vers ces mêmes données. Le nouveau lien matériel agit exactement comme le nom du fichier d'origine. Une fois créé, vous ne pouvez pas faire la différence entre le nouveau lien matériel et le nom original du fichier.

Vous pouvez savoir si un fichier a plusieurs liens matériels avec la commande **ls -l**. L'une des choses qu'il rapporte est le *nombre de liens* de chaque fichier, le nombre de liens matériels qu'il contient.

```
[user@host ~]$ pwd
/home/user
[user@host ~]$ ls -l newfile.txt
-rw-r--r-- 1 user user 0 Mar 11 19:19 newfile.txt
```

Dans l'exemple précédent, le nombre de liens de **newfile.txt** est 1, il a exactement un chemin absolu, qui est **/home/user/newfile.txt**.

Vous pouvez utiliser la commande **ln** pour créer un nouveau lien matériel (un autre nom) qui pointe vers un fichier existant. La commande nécessite au moins deux arguments, un chemin vers le fichier existant et le chemin vers le lien matériel que vous souhaitez créer.

L'exemple suivant crée un lien matériel nommé **newfile-link2.txt** pour le fichier existant **newfile.txt** dans le répertoire **/tmp**.

```
[user@host ~]$ ln newfile.txt /tmp/newfile-hlink2.txt
[user@host ~]$ ls -l newfile.txt /tmp/newfile-hlink2.txt
-rw-rw-r-- 2 user user 12 Mar 11 19:19 newfile.txt
-rw-rw-r-- 2 user user 12 Mar 11 19:19 /tmp/newfile-hlink2.txt
```

Si vous voulez savoir si deux fichiers sont des liens matériels l'un de l'autre, vous pouvez utiliser l'option **-i** avec la commande **ls** pour lister le *numéro d'inode* des fichiers. Si les fichiers sont sur

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

le même système de fichiers (sujet traité dans un moment) et que leurs numéros d'inodes sont les mêmes, les fichiers sont des liens matériels pointant vers les mêmes données.

```
[user@host ~]$ ls -il newfile.txt /tmp/newfile-hlink2.txt
8924107 -rw-rw-r--. 2 user user 12 Mar 11 19:19 newfile.txt
8924107 -rw-rw-r--. 2 user user 12 Mar 11 19:19 /tmp/newfile-hlink2.txt
```



IMPORTANT

Tous les liens matériels qui font référence au même fichier auront le même nombre de liens, les mêmes droits de d'accès, les mêmes propriétaires d'utilisateur et de groupe, les mêmes horodatages et le même contenu de fichier. Si l'une de ces informations est modifiée dans l'un des liens matériels, tous les autres liens matériels qui pointent vers le même fichier présenteront eux aussi la nouvelle information. En effet, chaque lien matériel pointe vers les mêmes données sur le périphérique de stockage.

Même si le fichier d'origine est supprimé, le contenu du fichier reste disponible tant qu'il existe encore au moins un lien matériel. Les données ne sont supprimées de la mémoire que lorsque le dernier lien matériel est supprimé.

```
[user@host ~]$ rm -f newfile.txt
[user@host ~]$ ls -l /tmp/newfile-hlink2.txt
-rw-rw-r--. 1 user user 12 Mar 11 19:19 /tmp/newfile-hlink2.txt
[user@host ~]$ cat /tmp/newfile-hlink2.txt
Hello World
```

Limitations des liens matériels

Les liens matériels ont des limites. Premièrement, les liens matériels ne peuvent être utilisés qu'avec des fichiers normaux. Vous ne pouvez pas utiliser **ln** pour créer un lien matériel vers un répertoire ou un fichier spécial.

Deuxièmement, les liens matériels ne peuvent être utilisés que si les deux fichiers sont sur le même *système de fichiers*. La hiérarchie du système de fichiers peut être composée de plusieurs périphériques de stockage. Selon la configuration de votre système, lorsque vous passez à un nouveau répertoire, ce répertoire et son contenu peuvent être stockés sur un système de fichiers différent.

Vous pouvez utiliser la commande **df** pour répertorier les répertoires situés sur différents systèmes de fichiers. Par exemple, vous pouvez voir un résultat comme suit :

```
[user@host ~]$ df
Filesystem      1K-blocks   Used Available Use% Mounted on
devtmpfs          886788     0   886788   0% /dev
tmpfs            902108     0   902108   0% /dev/shm
tmpfs            902108   8696   893412   1% /run
tmpfs            902108     0   902108   0% /sys/fs/cgroup
/dev/mapper/rhel_rhel8--root 10258432 1630460   8627972 16% /
/dev/sda1        1038336 167128   871208 17% /boot
tmpfs           180420     0   180420   0% /run/user/1000
[user@host ~]$
```

Les fichiers de deux répertoires « Mounted on » différents et leurs sous-répertoires se trouvent sur des systèmes de fichiers différents. (La correspondance la plus spécifique gagne.) Ainsi, le système dans cet exemple, vous pouvez créer un lien matériel entre **/var/tmp/link1** et **/home/user/file** parce qu'ils sont les deux sous-répertoires de **/** mais pas n'importe quel autre répertoire de la liste. Mais vous ne pouvez pas créer un lien dur entre **/boot/test/badlink** et **/home/user/file** parce que le premier fichier est dans un sous-répertoire de **/boot** (sur la liste "Mounted on") et le second fichier ne l'est pas.

Création de liens symboliques

La commande **ln -s** crée un lien symbolique, également appelé « lien logique ». Un lien symbolique n'est pas un fichier normal, mais un type de fichier spécial qui pointe vers un fichier ou un répertoire existant.

Les liens symboliques présentent certains avantages par rapport aux liens matériels :

- Ils peuvent lier deux fichiers sur des systèmes de fichiers différents.
- Ils peuvent pointer vers un répertoire ou un fichier spécial, pas seulement un fichier normal.

Dans l'exemple suivant, la commande **ln -s** est utilisée pour créer un nouveau lien symbolique pour le fichier **/home/user/newfile-link2.txt** existant qui sera nommé **/tmp/newfile-symlink.txt**.

```
[user@host ~]$ ln -s /home/user/newfile-link2.txt /tmp/newfile-symlink.txt
[user@host ~]$ ls -l newfile-link2.txt /tmp/newfile-symlink.txt
-rw-rw-r--. 1 user user 12 Mar 11 19:19 newfile-link2.txt
lrwxrwxrwx. 1 user user 11 Mar 11 20:59 /tmp/newfile-symlink.txt -> /home/user/
newfile-link2.txt
[user@host ~]$ cat /tmp/newfile-symlink.txt
Soft Hello World
```

Dans l'exemple précédent, le premier caractère de la longue liste de **/tmp/newfile-symlink.txt** est **l** au lieu de **-**. Cela indique que le fichier est un lien symbolique et non un fichier normal. (Un **d** indiquerait que le fichier est un répertoire.)

Lorsque le fichier d'origine est supprimé, le lien symbolique pointe toujours vers ce fichier, mais la cible n'existe plus. Un lien symbolique qui pointe vers un fichier manquant est appelé « lien symbolique non résolu ».

```
[user@host ~]$ rm -f newfile-link2.txt
[user@host ~]$ ls -l /tmp/newfile-symlink.txt
lrwxrwxrwx. 1 user user 11 Mar 11 20:59 /tmp/newfile-symlink.txt -> /home/user/
newfile-link2.txt
[user@host ~]$ cat /tmp/newfile-symlink.txt
cat: /tmp/newfile-symlink.txt: No such file or directory
```



IMPORTANT

Un des effets secondaires du lien symbolique non résolu dans l'exemple précédent est que si vous créez ultérieurement un nouveau fichier avec le même nom que le fichier supprimé (**/home/user/newfile-link2.txt**), le lien symbolique ne sera plus "non résolu" et pointera vers le nouveau fichier.

Les liens matériels ne fonctionnent pas comme ça. Si vous supprimez un lien matériel puis utilisez des outils normaux (plutôt que **ln**) pour créer un nouveau fichier avec le même nom, le nouveau fichier ne sera pas lié à l'ancien fichier.

Une façon de comparer les liens matériels et les liens symboliques qui pourraient vous aider à comprendre leur fonctionnement :

- Un lien matériel pointe un nom vers des données sur un périphérique de stockage
- Un lien symbolique pointe un nom vers un autre nom, qui pointe vers des données sur un périphérique de stockage

Un lien symbolique peut pointer vers un dossier. Le lien symbolique se comporte alors comme un répertoire. Le passage au lien symbolique avec **cd** fera du répertoire de travail actuel le répertoire lié. Certains outils peuvent garder une trace du fait que vous avez suivi un lien symbolique pour y accéder. Par exemple, par défaut **cd** mettra à jour votre répertoire de travail actuel en utilisant le nom du lien symbolique plutôt que le nom du répertoire actuel. (Il y a une option, **-P**, qui le mettra à jour avec le nom du répertoire actuel.)

Dans l'exemple suivant, un lien symbolique nommé **/home/user/configfiles** est créé qui pointe vers le répertoire **/etc**.

```
[user@host ~]$ ln -s /etc /home/user/configfiles
[user@host ~]$ cd /home/user/configfiles
[user@host configfiles]$ pwd
/home/user/configfiles
```



RÉFÉRENCES

Page de manuel (1)**ln**

info ln ('*ln*' : créer des liens entre des fichiers)

► EXERCICE GUIDÉ

CRÉATION DE LIENS ENTRE DES FICHIERS

Dans cet exercice, vous allez créer des liens fixes et des liens symboliques, et comparer les résultats.

RÉSULTATS

Vous devez pouvoir créer des liens fixes et des liens logiques entre des fichiers.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

Sur workstation, exécutez la commande **lab files-make start**. Cette commande exécute un script de démarrage qui détermine si l'hôte servera est accessible sur le réseau et crée les fichiers et les répertoires de travail sur servera.

```
[student@workstation ~]$ lab files-make start
```

- 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Créez un lien fixe nommé **/home/student/backups/source.backup** pour le fichier existant, **/home/student/files/source.file**.

- 2.1. Affichez le nombre de liens pour le fichier, **/home/student/files/source.file**.

```
[student@servera ~]$ ls -l files/source.file
total 4
-rw-r--r--. 1 student student 11 Mar 5 21:19 source.file
```

- 2.2. Créez un lien fixe nommé **/home/student/backups/source.backup**. Reliez-le au fichier, **/home/student/files/source.file**.

```
[student@servera ~]$ ln /home/student/files/source.file \
/home/student/backups/source.backup
```

- 2.3. Vérifiez le nombre de liens pour le fichier **/home/student/files/source.file** d'origine et le nouveau fichier lié, **/home/student/backups/source.backup**. Le nombre de liens devrait être **2** pour les deux fichiers.

```
[student@servera ~]$ ls -l /home/student/files/  
-rw-r--r--. 2 student student 11 Mar 5 21:19 source.file  
[student@servera ~]$ ls -l /home/student/backups/  
-rw-r--r--. 2 student student 11 Mar 5 21:19 source.backup
```

- 3. Créez un lien logique nommé **/home/student/tempdir** qui pointe vers le répertoire **/tmp** sur servera.

- 3.1. Créez un lien logique nommé **/home/student/tempdir** et liez-le à **/tmp**.

```
[student@servera ~]$ ln -s /tmp /home/student/tempdir
```

- 3.2. Utilisez la commande **ls -l** pour vérifier le lien logique nouvellement créé.

```
[student@servera ~]$ ls -l /home/student/tempdir  
lrwxrwxrwx. 1 student student 4 Mar 5 22:04 /home/student/tempdir -> /tmp
```

- 4. Quittez servera.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Fin

Sur workstation, exéutez le script **lab files-make finish** pour mettre fin à l'exercice. Ce script supprime tous les fichiers et répertoires créés sur servera pendant l'exercice.

```
[student@workstation ~]$ lab files-make finish
```

L'exercice guidé est maintenant terminé.

CORRESPONDANCE DES NOMS DE FICHIERS À L'AIDE DES EXTENSIONS PAR LE SHELL

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir exécuter efficacement les commandes affectant de nombreux fichiers en utilisant les fonctionnalités de filtrage par motif du shell bash.

EXTENSIONS DE LIGNE DE COMMANDE

Le shell bash dispose de plusieurs moyens pour étendre une ligne de commande, notamment le *filtrage par motif*, l'extension du répertoire personnel, l'extension de la chaîne et la substitution de variable. La fonction de recherche de chemins d'accès, historiquement appelée *globbing*, ou globalisation, est certainement la plus puissante d'entre eux. La fonction de globalisation de bash, appelée parfois « caractères génériques », facilite la gestion d'un grand nombre de fichiers. Grâce aux métacaractères, qui s'« étendent » pour correspondre aux noms de fichiers et de chemins recherchés, les commandes s'exécutent en une fois sur tout un ensemble ciblé de fichiers.

Filtrage par motif

La globalisation est une opération d'analyse de commande de shell qui étend un motif de caractères génériques en une liste de noms de chemin correspondants. Les métacaractères de la ligne de commande sont remplacés par la liste correspondante avant l'exécution de la commande. Les motifs qui ne renvoient pas de résultat affichent le motif original recherché sous forme de texte littéral. Les classes de motifs et les métacaractères ci-dessous sont couramment utilisés.

Tableau des métacaractères et des correspondances

MOTIF	CORRESPONDANCE
*	Toute chaîne de zéro caractère ou plus.
?	Tout caractère unique.
[abc...]	N'importe quel caractère de la classe entre crochets.
[!abc...]	Tout caractère <i>non</i> compris dans la classe entre crochets.
[^abc...]	Tout caractère <i>non</i> compris dans la classe entre crochets.
[:alpha:]	Tout caractère alphabétique.
[:lower:]	Tout caractère minuscule.
[:upper:]	Tout caractère majuscule.
[:alnum:]	Tout caractère alphabétique ou numérique.
[:punct:]	Tout caractère imprimeable, sauf un espace ou un caractère alphanumérique.

MOTIF	CORRESPONDANCE
<code>[:digit:]</code>	Tout chiffre unique compris entre 0 et 9.
<code>[:space:]</code>	Tout caractère d'espacement unique. Cela peut inclure des tabulations, des sauts de ligne, des retours chariot, des sauts de page ou des espaces.

Pour les exemples suivants, supposez que vous avez exécuté les commandes suivantes pour créer des exemples de fichiers.

```
[user@host ~]$ mkdir glob; cd glob
[user@host glob]$ touch alfa bravo charlie delta echo able baker cast dog easy
[user@host glob]$ ls
able alfa baker bravo cast charlie delta dog easy echo
[user@host glob]$
```

Le premier exemple utilisera des correspondances de modèle simples avec l'astérisque (*) et le point d'interrogation (?), et une classe de caractères, pour correspondre à certains de ces noms de fichiers.

```
[user@host glob]$ ls a*
able alfa
[user@host glob]$ ls *a*
able alfa baker bravo cast charlie delta easy
[user@host glob]$ ls [ac]*
able alfa cast charlie
[user@host glob]$ ls ???
able alfa cast easy echo
[user@host glob]$ ls ****
baker bravo delta
[user@host glob]$
```

Extension par tilde

Le caractère tilde (~) correspond au répertoire personnel de l'utilisateur courant. S'il démarre une chaîne de caractères autre qu'une barre oblique (/), le shell interprétera la chaîne jusqu'à cette barre oblique comme un nom d'utilisateur, le cas échéant, et remplacera la chaîne par le chemin absolu vers le répertoire personnel de cet utilisateur. Si aucun nom d'utilisateur ne correspond, un tilde réel suivi de la chaîne de caractères sera alors utilisé à la place.

Dans l'exemple suivant, la commande **echo** est utilisée pour afficher la valeur du caractère tilde. La commande **echo** peut également être utilisée pour afficher les valeurs des caractères d'extension par accolades, variable et autres.

```
[user@host glob]$ ls ~root
/root
[user@host glob]$ ls ~user
/home/user
[user@host glob]$ ls ~/glob
able alfa baker bravo cast charlie delta dog easy echo
```

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

```
[user@host glob]$ echo ~/glob  
/home/user/glob  
[user@host glob]$
```

Extension par accolades

L'extension par accolades permet de générer des chaînes de caractères discrétionnaires. Les accolades contiennent une liste de chaînes séparées par des virgules ou une expression séquentielle. Le résultat inclut le texte qui précède ou qui suit la définition de l'accolade. Les extensions par accolades peuvent être imbriquées les unes dans les autres. La syntaxe double point (...) s'étend en une séquence tel que **{m..p}** s'étend jusqu'en **m n o p**.

```
[user@host glob]$ echo {Sunday,Monday,Tuesday,Wednesday}.log  
Sunday.log Monday.log Tuesday.log Wednesday.log  
[user@host glob]$ echo file{1..3}.txt  
file1.txt file2.txt file3.txt  
[user@host glob]$ echo file{a..c}.txt  
filea.txt fileb.txt filec.txt  
[user@host glob]$ echo file{a,b}{1,2}.txt  
filea1.txt filea2.txt fileb1.txt fileb2.txt  
[user@host glob]$ echo file{a{1,2},b,c}.txt  
filea1.txt filea2.txt fileb.txt filec.txt  
[user@host glob]$
```

Une utilisation pratique de l'extension d'accolade est de créer rapidement un certain nombre de fichiers ou de répertoires.

```
[user@host glob]$ mkdir ../{RHEL{6,7,8}}  
[user@host glob]$ ls ../{RHEL*}  
RHEL6 RHEL7 RHEL8  
[user@host glob]$
```

Extension par variable

Une variable agit comme un conteneur nommé pouvant stocker une valeur en mémoire. Les variables facilitent l'accès aux données stockées et leur modification à partir de la ligne de commande ou dans un script shell.

Vous pouvez affecter des données sous forme de valeur à une variable via la syntaxe suivante :

```
[user@host ~]$ VARIABLENAME=value
```

Vous pouvez utiliser l'extension par variable pour convertir le nom de la variable en sa valeur sur la ligne de commande. Si une chaîne commence par un signe dollar (\$), alors le shell essaiera d'utiliser le reste de cette chaîne en tant que nom de variable et de la remplacer par la valeur de la variable.

```
[user@host ~]$ USERNAME=operator  
[user@host ~]$ echo $USERNAME  
operator
```

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

Pour éviter les erreurs dues à d'autres extensions du shell, vous pouvez mettre le nom de la variable entre accolades, par exemple \${VARIABLENAME}.

```
[user@host ~]$ USERNAME=operator  
[user@host ~]$ echo ${USERNAME}  
operator
```

Les variables du shell et les manières de les utiliser seront traitées plus en détail plus loin dans ce cours.

Substitution des commandes

La substitution de commande permet à la sortie d'une commande de remplacer la commande elle-même sur la ligne de commande. La substitution de commande se produit lorsqu'une commande est entre parenthèses et précédée d'un signe dollar (\$). La forme \$(**commande**) permet d'imbriquer plusieurs extensions de commandes.

```
[user@host glob]$ echo Today is $(date +%A).  
Today is Wednesday.  
[user@host glob]$ echo The time is $(date +%M) minutes past $(date +%l%p).  
The time is 26 minutes past 11AM.  
[user@host glob]$
```



NOTE

Une forme plus ancienne de substitution de commande utilise des apostrophes inversées : `**commande**` . La forme composée d'apostrophes inversées présente deux inconvénients : 1) il est facile de confondre les apostrophes inversées avec les apostrophes normales et 2) on ne peut pas imbriquer les apostrophes inversées.

Protection des arguments contre l'extension

Dans le shell bash, de nombreux caractères ont une signification particulière. Pour empêcher le shell d'effectuer des extensions de shell sur certaines parties de votre ligne de commande, vous pouvez citer et échapper des caractères et chaînes.

La barre oblique inversée (\) est un caractère d'échappement dans le shell Bash. Cela protégera le caractère qui le suit de son extension.

```
[user@host glob]$ echo The value of $HOME is your home directory.  
The value of /home/user is your home directory.  
[user@host glob]$ echo The value of \$HOME is your home directory.  
The value of $HOME is your home directory.  
[user@host glob]$
```

Dans l'exemple précédent, pour protéger le signe dollar de l'extension, Bash l'a traité comme un caractère régulier et il n'a pas effectué d'extension variable sur \$HOME.

Pour protéger les chaînes de caractères plus longues, on utilise des guillemets simples ('') ou doubles (") pour entourer ces chaînes. Ils ont des effets légèrement différents. Les guillemets simples arrêtent toute extension du shell. Les guillemets doubles arrêtent *la plupart* des extensions shell.

Utilisez les guillemets doubles pour supprimer la globalisation et l'extension par le shell, tout en permettant la substitution de commandes et de variables.

```
[user@host glob]$ myhost=$(hostname -s); echo $myhost
host
[user@host glob]$ echo "***** hostname is ${myhost} *****"
***** hostname is host *****
[user@host glob]$
```

Utilisez des guillemets simples pour interpréter *tout* le texte littéralement.

```
[user@host glob]$ echo "Will variable $myhost evaluate to $(hostname -s)?"
Will variable myhost evaluate to host?
[user@host glob]$ echo 'Will variable $myhost evaluate to $(hostname -s)?'
Will variable $myhost evaluate to $(hostname -s)?
[user@host glob]$
```



IMPORTANT

Les guillemets simples ('') et l'accent grave de substitution de commande (`) peuvent être faciles à confondre, tant à l'écran que sur le clavier. L'utilisation de l'un lorsque vous voulez utiliser l'autre entraîne un comportement inattendu du shell.



RÉFÉRENCES

Pages de manuel **bash(1)**, **cd(1)**, **glob(7)**, **isalpha(3)**, **ls(1)**, **path_resolution(7)** et **pwd(1)**

► QUIZ

CORRESPONDANCE DES NOMS DE FICHIERS À L'AIDE DES EXTENSIONS PAR LE SHELL

Choisissez les éléments appropriés en réponse aux questions suivantes :

► 1. Quel motif correspond uniquement aux noms de fichiers se terminant par « b » ?

- a. **b***
- b. ***b**
- c. ***b***
- d. **[!b]***

► 2. Quel motif correspond uniquement aux noms de fichiers commençant par « b » ?

- a. **b***
- b. ***b**
- c. ***b***
- d. **[!b]***

► 3. Quel motif correspond uniquement aux noms de fichiers dont la première lettre est différente de « b » ?

- a. **b***
- b. ***b**
- c. ***b***
- d. **[!b]***

► 4. Quel motif correspond à tous les noms de fichiers contenant la lettre « b » ?

- a. **b***
- b. ***b**
- c. ***b***
- d. **[!b]***

► 5. Quel motif correspond uniquement aux noms de fichiers contenant un chiffre ?

- a. ***#***
- b. ***[[:digit:]]***
- c. ***[digit]***
- d. **[0-9]**

► **6. Quel motif correspond uniquement aux noms de fichiers commençant par une lettre majuscule ?**

- a. ^?*
- b. ^*
- c. [upper]*
- d. [[:upper:]]*
- e. [[CAP]]*

► **7. Quel motif correspond uniquement aux noms de fichiers d'au moins trois caractères ?**

- a. ???*
- b. ???
- c. \3*
- d. +***
- e. . . .*

► SOLUTION

CORRESPONDANCE DES NOMS DE FICHIERS À L'AIDE DES EXTENSIONS PAR LE SHELL

Choisissez les éléments appropriés en réponse aux questions suivantes :

► 1. Quel motif correspond uniquement aux noms de fichiers se terminant par « b » ?

- a. b*
- b. *b
- c. *b*
- d. [!b]*

► 2. Quel motif correspond uniquement aux noms de fichiers commençant par « b » ?

- a. b*
- b. *b
- c. *b*
- d. [!b]*

► 3. Quel motif correspond uniquement aux noms de fichiers dont la première lettre est différente de « b » ?

- a. b*
- b. *b
- c. *b*
- d. [!b]*

► 4. Quel motif correspond à tous les noms de fichiers contenant la lettre « b » ?

- a. b*
- b. *b
- c. *b*
- d. [!b]*

► 5. Quel motif correspond uniquement aux noms de fichiers contenant un chiffre ?

- a. *#*
- b. *[[:digit:]]*
- c. *[digit]*
- d. [0-9]

► **6. Quel motif correspond uniquement aux noms de fichiers commençant par une lettre majuscule ?**

- a. ^?*
- b. ^*
- c. [upper]*
- d. [[:upper:]]*
- e. [[CAP]]*

► **7. Quel motif correspond uniquement aux noms de fichiers d'au moins trois caractères ?**

- a. ???*
- b. ???
- c. \3*
- d. +++*
- e. . . .*

► OPEN LAB

GESTION DE FICHIERS À PARTIR DE LA LIGNE DE COMMANDE

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez créer, déplacer et supprimer des fichiers et des répertoires de manière efficace en utilisant le shell et diverses techniques de correspondance de noms de fichiers.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Utiliser des caractères génériques pour localiser et manipuler des fichiers.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez la commande `lab files-review start`. Cette commande exécute un script de démarrage qui détermine si la machine `serverb` est accessible sur le réseau.

```
[student@workstation ~]$ lab files-review start
```

1. Utilisez la commande `ssh` pour vous connecter à `serverb` en tant qu'utilisateur `student`. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

2. Avant de créer des fichiers de projet, utilisez la commande `mkdir` avec extension par accolades pour créer des documents de planification de projet vides dans le répertoire `/home/student/Documents/project_plans`. (Conseil : si `~/Documents` n'existe pas, l'option `-p` pour la commande `mkdir` le crée.)
Créez deux fichiers vides dans le répertoire `~/Documents/project_plans` : `season1_project_plan.odf` et `season2_project_plan.odf`.
3. Créez des ensembles de fichiers vides à utiliser au cours de l'atelier. Si vous ne reconnaisssez pas immédiatement le raccourci d'extension par le shell prévu, vous devez utiliser la solution pour apprendre et vous exercer. Utilisez la saisie semi-automatique par tabulation du shell pour trouver facilement le nom du chemin d'accès aux fichiers.
Créez un total de 12 fichiers appelés `tv_seasonX_episodeY.ogg`. Remplacez `X` par le numéro de la saison et `Y` par le numéro de l'épisode, pour deux saisons de six épisodes chacune.

4. En tant qu'auteur d'une série de romans policiers à succès, les chapitres de votre prochain bestseller sont en cours d'édition avant publication. Créez un total de huit fichiers appelés **mystery_chapterX.odf**. Remplacez X par les chiffres de 1 à 8.
5. Utilisez la commande unique pour créer deux sous-répertoires nommés **season1** et **season2** dans le répertoire **Videos** pour organiser les épisodes télévisés.
6. Déplacez les épisodes télévisés appropriés vers les sous-répertoires des saisons. N'utilisez que deux commandes qui spécifient les destinations avec une syntaxe relative.
7. Créez une hiérarchie de répertoires à deux niveaux avec une seule commande pour organiser les chapitres du roman policier. Créez **my_bestseller** sous le répertoire **Documents**, et **chapters** sous le nouveau répertoire **my_bestseller**.
8. Créez trois sous-répertoires supplémentaires directement dans le répertoire **my_bestseller** en utilisant une seule commande. Nommez ces sous-répertoires **editor**, **changes** et **vacation**. L'option **-p** (create parents) est inutile, puisque le répertoire parent **my_bestseller** existe déjà.
9. Choisissez le répertoire **chapters**. En utilisant le raccourci vers le répertoire personnel tilde (~) pour spécifier les fichiers sources, déplacez tous les chapitres du livre dans le répertoire **chapters**, qui est désormais votre répertoire courant. Quelle est la syntaxe la plus simple pour spécifier le répertoire de destination ?
10. Vous avez envoyé les deux premiers chapitres à l'éditeur pour révision. Déplacez uniquement ces deux chapitres dans le répertoire **editor** pour éviter de les modifier lors de la révision. À partir du sous-répertoire **chapters**, utilisez l'extension par accolades avec une plage pour spécifier les noms de fichiers de chapitres à déplacer et un chemin relatif pour le répertoire de destination.
11. Lors de vos congés, vous avez l'intention d'écrire les chapitres 7 et 8. Utilisez une seule commande pour déplacer les fichiers du répertoire **chapters** dans le répertoire **vacation**. Spécifiez les noms de fichiers de chapitres en utilisant l'extension par accolades avec une liste de chaînes et sans utiliser de caractères génériques.
12. Choisissez le répertoire de travail **~/Videos/season2**, puis copiez le premier épisode de la saison dans le répertoire **vacation**.
13. Utilisez la commande **cd** unique pour changer votre répertoire de travail par le répertoire **~/Documents/my_bestseller/vacation**. Listez ses fichiers. Utilisez l'argument du répertoire de travail précédent pour revenir au répertoire **season2**. (Cette action fonctionnera si le dernier changement de répertoire avec la commande **cd** a été accompli avec une commande plutôt que plusieurs commandes **cd**.) À partir du répertoire **season2**, copiez le fichier de l'épisode 2 dans le répertoire **vacation**. Utilisez à nouveau le raccourci pour revenir au répertoire **vacation**.
14. Les auteurs des chapitres 5 et 6 veulent expérimenter d'éventuels changements. Copiez les deux fichiers du répertoire **~/Documents/my_bestseller/chapters** dans le répertoire **~/Documents/my_bestseller/changes** pour éviter que ces changements ne modifient les fichiers d'origine. Accédez au répertoire **~/Documents/my_bestseller**. Utilisez le filtrage par motif entre crochets pour spécifier les numéros de chapitre à rechercher dans l'argument de nom de fichier de la commande **cp**.
15. Changez le répertoire courant par le répertoire **changes**. Utilisez la commande **date +%F** avec substitution de commande pour copier **mystery_chapter5.odf** dans un nouveau fichier qui inclut la date complète. Le nom doit avoir la forme **mystery_chapter5_YYYY-MM-DD.odf**. Effectuez une autre copie de **mystery_chapter5.odf** et ajoutez l'horodatage actuel (sous la forme du nombre de secondes écoulées depuis l'époque, 1970-01-01 00:00 UTC) pour

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

garantir un nom de fichier unique. Utilisez la substitution de commande avec la commande **date +%s** pour accomplir cela.

16. Après un examen plus approfondi, vous décidez que les modifications de l'intrigue ne sont pas nécessaires. Supprimez le répertoire **changes**.
Si nécessaire, accédez au répertoire **changes** et supprimez tous les fichiers du répertoire. Vous ne pouvez pas supprimer un répertoire tant qu'il s'agit du répertoire de travail actuel. Choisissez le répertoire parent du répertoire **changes**. Essayez de supprimer le répertoire vide en utilisant la commande **rm** sans l'option récursive **-r**. Cette tentative devrait échouer. Enfin, utilisez la commande **rmdir** pour supprimer le répertoire vide, ce qui doit fonctionner.
17. Une fois les vacances terminées, le répertoire **vacation** devient inutile. Supprimez-le en utilisant la commande **rm** avec l'option récursive.
Une fois l'opération terminée, revenez au répertoire personnel de l'utilisateur **student**.
18. Créez un lien fixe vers le fichier **~/Documents/project_plans/season2_project_plan.odf** nommé **~/Documents/backups/season2_project_plan.back**. Un lien fixe protège contre la suppression accidentelle du fichier d'origine et met le fichier de sauvegarde à jour au fil des modifications apportées au fichier d'origine.
19. Quittez **serverb**.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

Évaluation

Sur **workstation**, exécutez le script **lab files-review grade** pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab files-review grade
```

Finish (Terminer)

Sur **workstation**, exécutez le script **lab files-review finish** pour mettre fin à l'atelier. Ce script supprime tous les fichiers et répertoires créés sur **serverb** pendant cet exercice pratique.

```
[student@workstation ~]$ lab files-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

GESTION DE FICHIERS À PARTIR DE LA LIGNE DE COMMANDE

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez créer, déplacer et supprimer des fichiers et des répertoires de manière efficace en utilisant le shell et diverses techniques de correspondance de noms de fichiers.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Utiliser des caractères génériques pour localiser et manipuler des fichiers.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab files-review start**. Cette commande exécute un script de démarrage qui détermine si la machine **serverb** est accessible sur le réseau.

```
[student@workstation ~]$ lab files-review start
```

1. Utilisez la commande **ssh** pour vous connecter à **serverb** en tant qu'utilisateur **student**. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

2. Avant de créer des fichiers de projet, utilisez la commande **mkdir** avec extension par accolades pour créer des documents de planification de projet vides dans le répertoire **/home/student/Documents/project_plans**. (Conseil : si **~/Documents** n'existe pas, l'option **-p** pour la commande **mkdir** le crée.)

Créez deux fichiers vides dans le répertoire **~/Documents/project_plans** : **season1_project_plan.odf** et **season2_project_plan.odf**.

```
[student@serverb ~]$ mkdir -p ~/Documents/project_plans
[student@serverb ~]$ touch \
~/Documents/project_plans/{season1,season2}_project_plan.odf
[student@serverb ~]$ ls -lR Documents/
Documents/:
total 0
```

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

```
drwxrwxr-x. 2 student student 70 Jan 31 18:20 project_plans
```

Documents/project_plans:

```
total 0  
-rw-rw-r--. 1 student student 0 Jan 31 18:20 season1_project_plan.odf  
-rw-rw-r--. 1 student student 0 Jan 31 18:20 season2_project_plan.odf
```

3. Créez des ensembles de fichiers vides à utiliser au cours de l'atelier. Si vous ne reconnaissiez pas immédiatement le raccourci d'extension par le shell prévu, vous devez utiliser la solution pour apprendre et vous exercer. Utilisez la saisie semi-automatique par tabulation du shell pour trouver facilement le nom du chemin d'accès aux fichiers.

Créez un total de 12 fichiers appelés **tv_seasonX_episodeY.ogg**. Remplacez X par le numéro de la saison et Y par le numéro de l'épisode, pour deux saisons de six épisodes chacune.

```
[student@serverb ~]$ touch tv_season{1..2}_episode{1..6}.ogg  
[student@serverb ~]$ ls tv*  
tv_season1_episode1.ogg  tv_season1_episode5.ogg  tv_season2_episode3.ogg  
tv_season1_episode2.ogg  tv_season1_episode6.ogg  tv_season2_episode4.ogg  
tv_season1_episode3.ogg  tv_season2_episode1.ogg  tv_season2_episode5.ogg  
tv_season1_episode4.ogg  tv_season2_episode2.ogg  tv_season2_episode6.ogg
```

4. En tant qu'auteur d'une série de romans policiers à succès, les chapitres de votre prochain bestseller sont en cours d'édition avant publication. Créez un total de huit fichiers appelés **mystery_chapterX.odf**. Remplacez X par les chiffres de 1 à 8.

```
[student@serverb ~]$ touch mystery_chapter{1..8}.odf  
[student@serverb ~]$ ls mys*  
mystery_chapter1.odf  mystery_chapter4.odf  mystery_chapter7.odf  
mystery_chapter2.odf  mystery_chapter5.odf  mystery_chapter8.odf  
mystery_chapter3.odf  mystery_chapter6.odf
```

5. Utilisez la commande unique pour créer deux sous-répertoires nommés **season1** et **season2** dans le répertoire **Videos** pour organiser les épisodes télévisés.

```
[student@serverb ~]$ mkdir -p Videos/season{1..2}  
[student@serverb ~]$ ls Videos  
season1  season2
```

6. Déplacez les épisodes télévisés appropriés vers les sous-répertoires des saisons. N'utilisez que deux commandes qui spécifient les destinations avec une syntaxe relative.

```
[student@serverb ~]$ mv tv_season1* Videos/season1  
[student@serverb ~]$ mv tv_season2* Videos/season2  
[student@serverb ~]$ ls -R Videos  
Videos:  
season1  season2  
  
Videos/season1:  
tv_season1_episode1.ogg  tv_season1_episode3.ogg  tv_season1_episode5.ogg  
tv_season1_episode2.ogg  tv_season1_episode4.ogg  tv_season1_episode6.ogg
```

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

Videos/season2:

tv_season2_episode1.ogg tv_season2_episode3.ogg tv_season2_episode5.ogg
tv_season2_episode2.ogg tv_season2_episode4.ogg tv_season2_episode6.ogg

7. Créez une hiérarchie de répertoires à deux niveaux avec une seule commande pour organiser les chapitres du roman policier. Créez **my_bestseller** sous le répertoire **Documents**, et **chapters** sous le nouveau répertoire **my_bestseller**.

```
[student@serverb ~]$ mkdir -p Documents/my_bestseller/chapters  
[student@serverb ~]$ ls -R Documents  
Documents:  
my_bestseller project_plans
```

```
Documents/my_bestseller:  
chapters
```

```
Documents/my_bestseller/chapters:
```

```
Documents/project_plans:  
season1_project_plan.odf season2_project_plan.odf
```

8. Créez trois sous-répertoires supplémentaires directement dans le répertoire **my_bestseller** en utilisant une seule commande. Nommez ces sous-répertoires **editor**, **changes** et **vacation**. L'option **-p** (create parents) est inutile, puisque le répertoire parent **my_bestseller** existe déjà.

```
[student@serverb ~]$ mkdir Documents/my_bestseller/{editor,changes,vacation}  
[student@serverb ~]$ ls -R Documents
```

```
Documents:
```

```
my_bestseller project_plans
```

```
Documents/my_bestseller:  
changes chapters editor vacation
```

```
Documents/my_bestseller/changes:
```

```
Documents/my_bestseller/chapters:
```

```
Documents/my_bestseller/editor:
```

```
Documents/my_bestseller/vacation:
```

```
Documents/project_plans:
```

```
season1_project_plan.odf season2_project_plan.odf
```

9. Choisissez le répertoire **chapters**. En utilisant le raccourci vers le répertoire personnel tilde (~) pour spécifier les fichiers sources, déplacez tous les chapitres du livre dans le répertoire

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

chapters, qui est désormais votre répertoire courant. Quelle est la syntaxe la plus simple pour spécifier le répertoire de destination ?

```
[student@serverb ~]$ cd Documents/my_bestseller/chapters
[student@serverb chapters]$ mv ~/mystery_chapter* .
[student@serverb chapters]$ ls
mystery_chapter1.odf mystery_chapter4.odf mystery_chapter7.odf
mystery_chapter2.odf mystery_chapter5.odf mystery_chapter8.odf
mystery_chapter3.odf mystery_chapter6.odf
```

10. Vous avez envoyé les deux premiers chapitres à l'éditeur pour révision. Déplacez uniquement ces deux chapitres dans le répertoire **editor** pour éviter de les modifier lors de la révision. À partir du sous-répertoire chapters, utilisez l'extension par accolades avec une plage pour spécifier les noms de fichiers de chapitres à déplacer et un chemin relatif pour le répertoire de destination.

```
[student@serverb chapters]$ mv mystery_chapter{1..2}.odf ../editor
[student@serverb chapters]$ ls
mystery_chapter3.odf mystery_chapter5.odf mystery_chapter7.odf
mystery_chapter4.odf mystery_chapter6.odf mystery_chapter8.odf
[student@serverb chapters]$ ls ../editor
mystery_chapter1.odf mystery_chapter2.odf
```

11. Lors de vos congés, vous avez l'intention d'écrire les chapitres 7 et 8. Utilisez une seule commande pour déplacer les fichiers du répertoire **chapters** dans le répertoire **vacation**. Spécifiez les noms de fichiers de chapitres en utilisant l'extension par accolades avec une liste de chaînes et sans utiliser de caractères génériques.

```
[student@serverb chapters]$ mv mystery_chapter{7,8}.odf ../vacation
[student@serverb chapters]$ ls
mystery_chapter3.odf mystery_chapter5.odf
mystery_chapter4.odf mystery_chapter6.odf
[student@serverb chapters]$ ls ../vacation
mystery_chapter7.odf mystery_chapter8.odf
```

12. Choisissez le répertoire de travail **~/Videos/season2**, puis copiez le premier épisode de la saison dans le répertoire **vacation**.

```
[student@serverb chapters]$ cd ~/Videos/season2
[student@serverb season2]$ cp *episode1.ogg ~/Documents/my_bestseller/vacation
```

13. Utilisez la commande **cd** unique pour changer votre répertoire de travail par le répertoire **~/Documents/my_bestseller/vacation**. Listez ses fichiers. Utilisez l'argument *du répertoire de travail précédent* pour revenir au répertoire **season2**. (Cette action fonctionnera si le dernier changement de répertoire avec la commande **cd** a été accompli avec une commande plutôt que plusieurs commandes **cd**.) À partir du répertoire **season2**, copiez le fichier de l'épisode 2 dans le répertoire **vacation**. Utilisez à nouveau le raccourci pour revenir au répertoire **vacation**.

```
[student@serverb season2]$ cd ~/Documents/my_bestseller/vacation
[student@serverb vacation]$ ls
mystery_chapter7.odf mystery_chapter8.odf tv_season2_episode1.ogg
[student@serverb vacation]$ cd -
```

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

```
/home/ec2-user/Videos/season2  
[student@serverb season2]$ cp *episode2.ogg ~/Documents/my_bestseller/vacation  
[student@serverb vacation]$ cd -  
/home/ec2-user/Documents/my_bestseller/vacation  
[student@serverb vacation]$ ls  
mystery_chapter7.odf tv_season2_episode1.ogg  
mystery_chapter8.odf tv_season2_episode2.ogg
```

14. Les auteurs des chapitres 5 et 6 veulent expérimenter d'éventuels changements. Copiez les deux fichiers du répertoire **~/Documents/my_bestseller/chapters** dans le répertoire **~/Documents/my_bestseller/changes** pour éviter que ces changements ne modifient les fichiers d'origine. Accédez au répertoire **~/Documents/my_bestseller**. Utilisez le filtrage par motif entre crochets pour spécifier les numéros de chapitre à rechercher dans l'argument de nom de fichier de la commande **cp**.

```
[student@serverb vacation]$ cd ~/Documents/my_bestseller  
[student@serverb my_bestseller]$ cp chapters/mystery_chapter[56].odf changes  
[student@serverb my_bestseller]$ ls chapters  
mystery_chapter3.odf mystery_chapter5.odf  
mystery_chapter4.odf mystery_chapter6.odf  
[student@serverb my_bestseller]$ ls changes  
mystery_chapter5.odf mystery_chapter6.odf
```

15. Changez le répertoire courant par le répertoire **changes**.

Utilisez la commande **date +%F** avec substitution de commande pour copier **mystery_chapter5.odf** dans un nouveau fichier qui inclut la date complète. Le nom doit avoir la forme **mystery_chapter5_YYYY-MM-DD.odf**.

Effectuez une autre copie de **mystery_chapter5.odf** et ajoutez l'horodatage actuel (sous la forme du nombre de secondes écoulées depuis l'époque, 1970-01-01 00:00 UTC) pour garantir un nom de fichier unique. Utilisez la substitution de commande avec la commande **date +%s** pour accomplir cela.

```
[student@serverb my_bestseller]$ cd changes  
[student@serverb changes]$ cp mystery_chapter5.odf \  
mystery_chapter5_$(date +%F).odf  
[student@serverb changes]$ cp mystery_chapter5.odf \  
mystery_chapter5_$(date +%s).odf  
[student@serverb changes]$ ls  
mystery_chapter5_1492545076.odf mystery_chapter5.odf  
mystery_chapter5_2017-04-18.odf mystery_chapter6.odf
```

16. Après un examen plus approfondi, vous décidez que les modifications de l'intrigue ne sont pas nécessaires. Supprimez le répertoire **changes**.

Si nécessaire, accédez au répertoire **changes** et supprimez tous les fichiers du répertoire. Vous ne pouvez pas supprimer un répertoire tant qu'il s'agit du répertoire de travail actuel. Choisissez le répertoire parent du répertoire **changes**. Essayez de supprimer le répertoire

CHAPITRE 3 | Gestion de fichiers à partir de la ligne de commande

vide en utilisant la commande **rm** sans l'option récursive **-r**. Cette tentative devrait échouer. Enfin, utilisez la commande **rmdir** pour supprimer le répertoire vide, ce qui doit fonctionner.

```
[student@serverb changes]$ rm mystery*
[student@serverb changes]$ cd ..
[student@serverb my_bestseller]$ rm changes
rm: cannot remove 'changes': Is a directory
[student@serverb my_bestseller]$ rmdir changes
[student@serverb my_bestseller]$ ls
chapters editor vacation
```

17. Une fois les vacances terminées, le répertoire **vacation** devient inutile. Supprimez-le en utilisant la commande **rm** avec l'option **récursive**.

Une fois l'opération terminée, revenez au répertoire personnel de l'utilisateur **student**.

```
[student@serverb my_bestseller]$ rm -r vacation
[student@serverb my_bestseller]$ ls
chapters editor
[student@serverb my_bestseller]$ cd
[student@serverb ~]$
```

18. Créez un lien fixe vers le fichier **~/Documents/project_plans/season2_project_plan.odf** nommé **~/Documents/backups/season2_project_plan.odf.back**. Un lien fixe protège contre la suppression accidentelle du fichier d'origine et met le fichier de sauvegarde à jour au fil des modifications apportées au fichier d'origine.

Notez que le nombre de liens est **2** pour les deux fichiers **season2_project_plan.odf.back** et **season2_project_plan.odf**.

```
[student@serverb ~]$ mkdir ~/Documents/backups
[student@serverb ~]$ ln ~/Documents/project_plans/season2_project_plan.odf \
~/Documents/backups/season2_project_plan.odf.back
[student@serverb ~]$ ls -lR ~/Documents/
/home/student/Documents/:
total 0
drwxrwxr-x. 2 student student 43 Jan 31 18:59 backups
drwxrwxr-x. 4 student student 36 Jan 31 19:42 my_bestseller
drwxrwxr-x. 2 student student 70 Jan 31 18:20 project_plans

/home/student/Documents/backups:
total 4
-rw-rw-r--. 2 student student 0 Jan 31 19:05 season2_project_plan.odf.back

/home/student/Documents/my_bestseller:
total 0
drwxrwxr-x. 2 student student 118 Jan 31 19:39 chapters
drwxrwxr-x. 2 student student 62 Jan 31 19:38 editor

/home/student/Documents/my_bestseller/chapters:
total 0
-rw-rw-r--. 1 student student 0 Jan 31 19:18 mystery_chapter3.odf
-rw-rw-r--. 1 student student 0 Jan 31 19:18 mystery_chapter4.odf
-rw-rw-r--. 1 student student 0 Jan 31 19:18 mystery_chapter5.odf
```

```
-rw-rw-r-- 1 student student 0 Jan 31 19:18 mystery_chapter6.odf

/home/student/Documents/my_bestseller/editor:
total 0
-rw-rw-r-- 1 student student 0 Jan 31 19:18 mystery_chapter1.odf
-rw-rw-r-- 1 student student 0 Jan 31 19:18 mystery_chapter2.odf

/home/student/Documents/project_plans:
total 4
-rw-rw-r-- 1 student student 0 Jan 31 18:20 season1_project_plan.odf
-rw-rw-r-- 2 student student 0 Jan 31 19:05 season2_project_plan.odf
```

19. Quittez serverb.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

Évaluation

Sur workstation, exécutez le script **lab files-review grade** pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab files-review grade
```

Finish (Terminer)

Sur workstation, exécutez le script **lab files-review finish** pour mettre fin à l'atelier. Ce script supprime tous les fichiers et répertoires créés sur serverb pendant cet exercice pratique.

```
[student@workstation ~]$ lab files-review finish
```

L'atelier est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Les fichiers d'un système Linux sont organisés dans une arborescence de répertoires inversée unique appelée hiérarchie du système de fichiers.
- Les chemins absous commencent par un / et indiquent l'emplacement d'un fichier dans la hiérarchie du système de fichiers.
- Les chemins relatifs ne commencent pas par un / et indiquent l'emplacement d'un fichier par rapport au répertoire de travail actuel.
- Cinq commandes clés sont utilisées pour gérer les fichiers : **mkdir**, **rmdir**, **cp**, **mv**, et **rm**.
- Les liens matériels et les liens symboliques sont différents moyens de faire pointer plusieurs noms de fichiers vers les mêmes données.
- Le shell Bash fournit des fonctionnalités de filtrage par motif, d'extension et de substitution pour vous aider à exécuter efficacement des commandes.

CHAPITRE 4

AIDE DANS RED HAT ENTERPRISE LINUX

PROJET

Résoudre les problèmes en utilisant les systèmes d'aide en local.

OBJECTIFS

- Rechercher des informations dans les pages de manuel du système Linux local.
- Rechercher des informations dans la documentation locale de GNU Info.

SECTIONS

- Lecture de pages de manuel (et exercice guidé)
- Lecture de la documentation Info (et exercice guidé)

ATELIER

Aide dans Red Hat Enterprise Linux

LECTURE DES PAGES DE MANUEL

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir trouver des informations dans le manuel du système Linux local.

PRÉSENTATION DE LA COMMANDE DU MANUEL

Parmi les sources de documentation généralement disponibles sur le système local figurent les pages de manuel du système ou *pages de manuel*. Ces pages font partie des packages logiciels pour lesquels elles fournissent de la documentation. Vous pouvez y accéder à partir de la ligne de commande à l'aide de la commande **man**.

Le manuel historique du programmeur Linux, duquel proviennent les pages de manuel, était suffisamment épais pour constituer plusieurs parties imprimées. Chaque section contient des informations sur un sujet particulier.

Sections courantes du manuel Linux

SECTION	TYPE DE CONTENU
1	Commandes de l'utilisateur (<i>programmes exécutables et shell</i>)
2	Appels au système (<i>routines du noyau appelées depuis l'espace utilisateur</i>)
3	Fonctions des bibliothèques (<i>fournies par les bibliothèques des programmes</i>)
4	Fichiers spéciaux (<i>tels que les fichiers de périphérique</i>)
5	Formats de fichier (<i>pour de nombreux fichiers et structures de configuration</i>)
6	Jeux (<i>section historique pour les programmes de divertissement</i>)
7	Conventions, normes et divers (<i>protocoles, systèmes de fichiers</i>)
8	Administration du système et commandes privilégiées (<i>tâches de maintenance</i>)
9	API du noyau Linux (<i>appels internes au noyau</i>)

Afin de distinguer des noms de rubriques identiques dans des sections différentes, les références de pages de manuel sont suivies du numéro de section entre parenthèses après la rubrique. Par exemple, **passwd(1)** décrit la commande pour modifier les mots de passe, tandis que **passwd(5)** explique le format du fichier **/etc/passwd** qui stocke les comptes d'utilisateurs locaux.

Pour afficher des pages de manuel spécifiques, utilisez la commande **man topic**. Le contenu s'affiche écran par écran. La commande **man** recherche les sections du manuel par ordre alphanumérique. Par exemple, la commande **man passwd** affiche **passwd(1)** par défaut. Pour afficher la rubrique de la page de manuel d'une section spécifique, ajoutez en argument le numéro de section : **man 5 passwd** affiche **passwd(5)**.

NAVIGATION ET RECHERCHE DANS LES PAGES DE MANUEL

La capacité à rechercher efficacement des rubriques et à naviguer dans les pages du manuel est une compétence administrative essentielle. Les outils d'interface graphique facilitent la configuration des ressources système courantes, mais l'interface de ligne de commande est encore plus efficace. Pour naviguer efficacement dans la ligne de commande, vous devez pouvoir trouver les informations dont vous avez besoin dans les pages de manuel.

Le tableau suivant liste les commandes de navigation basiques lors de la visualisation des pages de manuel :

Navigation dans les pages de manuel

COMMANDÉ	RÉSULTAT
Barre d'espace	Faire défiler d'un écran (vers le bas)
Page suivante	Faire défiler d'un écran (vers le bas)
Page précédente	Faire défiler d'un écran (vers le haut)
Flèche vers le bas	Faire défiler d'une ligne (vers le bas)
Flèche vers le haut	Revenir à la ligne précédente (vers le haut)
D	Faire défiler d'un demi-écran (vers le bas)
U	Revenir au demi-écran précédent (vers le haut)
/chaîne de caractères	Rechercher (vers le bas) la <i>chaîne de caractères</i> suivante dans la page de manuel
N	Répéter la recherche précédente vers l'avant (vers le bas) dans la page de manuel
Maj+N	Répéter la recherche précédente vers l'arrière (vers le haut) de la page de manuel
G	Aller au début de la page de manuel
Maj+G	Aller à la fin de la page de manuel
Q	Quitter man et revenir à l'invite du shell de commande

**IMPORTANT**

Pendant les recherches, la chaîne de caractères autorise la syntaxe des expressions régulières. Alors qu'un texte simple (tel que **passwd**) fonctionne comme on pourrait s'y attendre, les expressions régulières utilisent des métacaractères (comme \$, *, . et ^) pour un filtrage par motif plus sophistiqué. Par conséquent, la recherche de chaînes de caractères qui peuvent inclure des métacaractères d'expressions de programme, comme **make \$\$\$**, peut renvoyer des résultats inattendus.

Les expressions régulières et leur syntaxe sont traitées dans *Red Hat System Administration II* et à la rubrique **regex(7)** du manuel.

Lecture des pages de manuel

Chaque rubrique est séparée en plusieurs parties. La plupart des rubriques partagent les mêmes titres et sont présentées dans le même ordre. Généralement, une rubrique ne comporte pas tous les titres, car tous ne s'appliquent pas à toutes les rubriques.

Les titres courants sont :

Titres

TITRE	DESCRIPTION
NAME	Nom du sujet. Généralement, une commande ou un nom de fichier. Description très brève.
SYNOPSIS	Résumé de la syntaxe de la commande.
DESCRIPTION	Description détaillée permettant d'offrir une connaissance basique du sujet.
OPTIONS	Explication des options d'exécution de la commande.
EXAMPLES	Exemples d'utilisation de la commande, de la fonction ou du fichier.
FILES	Liste de fichiers et de répertoires liés à la page de manuel.
SEE ALSO	Informations connexes, généralement d'autres rubriques de pages de manuel.
BUGS	Bogues connus dans le logiciel.
AUTHOR	Informations sur les personnes ayant contribué au développement de la rubrique.

RECHERCHE DANS LES PAGES DE MANUEL PAR MOT-CLÉ

On lance une recherche par mot-clé dans les pages de manuel avec **man -k keyword** qui affiche une liste de rubriques de pages de manuel correspondant au mot-clé avec leurs numéros de section.

```
[student@desktopX ~]$ man -k passwd
checkPasswdAccess (3) - query the SELinux policy database in the kernel.
chpasswd (8)           - update passwords in batch mode
```

```

ckpasswd (8)           - nnrpd password authenticator
fgetpwent_r (3)        - get passwd file entry reentrantly
getpwent_r (3)         - get passwd file entry reentrantly
...
passwd (1)             - update user's authentication tokens
sslpasswd (1ssl)       - compute password hashes
passwd (5)              - password file
passwd.nntp (5)         - Passwords for connecting to remote NNTP servers
passwd2des (3)          - RFS password encryption
...

```

Les rubriques d'administration du système les plus populaires se trouvent dans les sections 1 (commandes utilisateur), 5 (formats de fichiers) et 8 (commandes administratives). Les administrateurs qui utilisent certains outils de résolution de problème utilisent également la section 2 (appels système). Les sections restantes sont généralement des références pour les programmeurs ou pour une administration avancée.



NOTE

Les recherches par mot-clé s'appuient sur un index généré par la commande **mandb(8)** qui doit être exécutée en tant que **root**. La commande est lancée quotidiennement via le fichier **cron.daily** ou par **anacrontab** dans l'heure qui suit le démarrage si le fichier est obsolète.



IMPORTANT

L'option **-K** (majuscule) de la commande **man** effectue une recherche par page en texte intégral, et non pas sur les seuls titres et descriptions comme l'option **-k**. Une recherche en texte intégral utilise plus de ressources système et prend plus de temps.



RÉFÉRENCES

Pages de manuel **man(1)**, **mandb(8)**, **man-pages(7)**, **less(1)**, **intro(1)**, **intro(2)**, **intro(5)**, **intro(7)** et **intro(8)**

► EXERCICE GUIDÉ

LECTURE DES PAGES DE MANUEL

Au cours de cet exercice, vous allez vous entraîner à rechercher des informations à l'aide des options et arguments de **man**.

RÉSULTATS

Vous devez pouvoir utiliser le système manuel Linux **man** et trouver des informations utiles par la recherche et la navigation.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab help-manual start**. Elle crée un fichier appelé **manual**.

```
[student@workstation ~]$ lab help-manual start
```

- 1. Sur **workstation**, affichez la page de manuel **gedit**. Affichez les options permettant de modifier un fichier spécifique en utilisant **gedit** à partir de la ligne de commande. Utilisez l'une des options de la page de manuel **gedit** pour ouvrir le fichier **/home/student/manual** en utilisant **gedit** avec le curseur à la fin du fichier.

- 1.1. Affichez la page de manuel **gedit**.

```
[student@workstation ~]$ man gedit
```

```
GEDIT(1)      General Commands Manual      GEDIT(1)
NAME
       gedit - text editor for the GNOME Desktop

SYNOPSIS
       gedit [OPTION...] [FILE...] [+LINE[:COLUMN]]
       gedit [OPTION...] -
...output omitted...
```

- 1.2. Dans la page de manuel **gedit**, découvrez les options permettant de modifier un fichier spécifique à partir de la ligne de commande.

```
...output omitted...
FILE Specifies the file to open when gedit starts.
...output omitted...
+LINE For the first file, go to the line specified by LINE (do not insert
a space between the "+" sign and the number). If LINE is missing, go to the last
line.
...output omitted...
```

Appuyez sur **q** pour quitter la page de manuel.

- 1.3. Utilisez la commande **gedit +** pour ouvrir le fichier **manual**. Le numéro de ligne manquant à côté de l'option **+** ouvre un fichier passé en tant qu'argument avec le curseur à la fin de la dernière ligne.

```
[student@workstation ~]$ gedit + manual
```

```
the quick brown fox just came over to greet the lazy poodle!
```

Confirmez que le fichier est ouvert avec le curseur à la fin du fichier de la dernière ligne du fichier. Appuyez sur **Ctrl+q** pour fermer l'application.

▶ 2. Lisez la page de manuel **su(1)**.

Quand **user** est omis, la commande **su** suppose que l'utilisateur est **root**. Si la commande **su** est suivie d'un tiret unique (-), il commence un shell de connexion enfant. Sans le tiret simple, un shell enfant sans connexion est créé, correspondant à l'environnement actuel de l'utilisateur.

```
[student@workstation ~]$ man 1 su
```

```
SU(1) User Commands SU(1)
NAME
    su - run a command with substitute user and group ID

SYNOPSIS
    su [options] [-] [user [argument...]]

DESCRIPTION
    su allows to run commands with a substitute user and group ID.
    When called without arguments, su defaults to running an interactive
    shell as root.
...output omitted...
OPTIONS
...output omitted...
-, -l, --login
    Start the shell as a login shell with an environment similar to a real login
...output omitted...
```

**NOTE**

Notez que les options séparées par des virgules sur une seule ligne, telles que **-**, **-1** et **--login**, affichent le même comportement.

Appuyez sur **q** pour quitter la page de manuel.

- 3. La commande **man** dispose également de ses propres pages de manuel.

```
[student@workstation ~]$ man man
MAN(1)           Manual pager utils                               MAN(1)

NAME
    man - an interface to the on-line reference manuals
...output omitted...

DESCRIPTION
    man is the system's manual pager. Each page argument given to man is
    normally the name of a program, utility or function. The manual page
    associated with each of these arguments is then found and displayed.
    A section, if provided, will direct man to look only in that section
    of the manual.
...output omitted...
```

Appuyez sur **q** pour quitter la page de manuel.

- 4. Toutes les pages de manuel sont situées dans **/usr/share/man**. Localisez les pages binaires, source et de manuel situées dans le répertoire **/usr/share/man** à l'aide de la commande **whereis**.

```
[student@workstation ~]$ whereis passwd
passwd: /usr/bin/passwd /etc/passwd /usr/share/man/man1/passwd.1.gz /usr/share/
man/man5/passwd.5.gz
```

- 5. Utilisez la commande **man -k zip** pour lister des informations détaillées sur une archive Zip.

```
[student@workstation ~]$ man -k zip
...output omitted...
zipinfo (1)      - list detailed information about a ZIP archive
zipnote (1)      - write the comments in zipfile to stdout, edit comments and
    rename files in zipfile
zipsplit (1)     - split a zipfile into smaller zipfiles
```

- 6. Utilisez la commande **man -k boot** pour lister la page de manuel contenant une liste des paramètres qui peuvent être transmis au noyau lors du démarrage.

```
[student@workstation ~]$ man -k boot
...output omitted...
bootctl (1)           - Control the firmware and boot manager settings
bootparam (7)          - introduction to boot time parameters of the Linux kernel
bootup (7)             - System bootup process
...output omitted...
```

- 7. Utilisez **man -k ext4** pour rechercher la commande utilisée pour ajuster les paramètres du système de fichiers ext4.

```
[student@workstation ~]$ man -k ext4
...output omitted...
resize2fs (8)           - ext2/ext3/ext4 file system resizer
tune2fs (8)             - adjust tunable filesystem parameters on ext2/ext3/ext4
filesystems
```

Fin

Sur workstation, exécutez le script **lab help-manual finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab help-manual finish
```

L'exercice guidé est maintenant terminé.

LECTURE DE LA DOCUMENTATION INFO

OBJECTIFS

À la fin de cette section, les étudiants doivent pouvoir trouver des informations dans la documentation locale dans GNU Info.

PRÉSENTATION DE GNU INFO

Les pages de manuel présentent un format utile pour rechercher les références de commande, mais sont moins efficaces comme documentation générale. À cette fin, le projet GNU a développé un système de documentation en ligne différent, connu sous le nom de *GNU Info*. Les documents Info constituent une ressource importante d'informations sur un système Red Hat Enterprise Linux, parce que beaucoup de composants et d'utilitaires fondamentaux, comme le paquetage *coreutils* et les bibliothèques standards *glibc*, sont développés par le projet GNU, ou utilisent au moins le système de documents Info.



IMPORTANT

Vous vous demandez peut-être pourquoi il existe deux systèmes de documentation locaux, des pages de manuel et des documents Info. Certaines des raisons à cela sont d'ordre pratique, et d'autres sont liées à la manière dont Linux et ses applications ont été développés par diverses communautés open source au fil des ans.

Les pages de manuel ont un format beaucoup plus formel et documentent généralement une commande ou une fonction spécifique à partir d'un progiciel, et sont structurées sous forme de fichiers texte individuels. Les documents d'information couvrent généralement des progiciels particuliers dans leur ensemble, ont tendance à contenir des exemples plus pratiques d'utilisation du logiciel et sont structurés sous la forme de documents hypertextes.

Vous devez vous familiariser avec les deux systèmes afin de tirer le meilleur parti des informations qui vous sont fournies par le système.

Lecture de la documentation Info

Pour lancer la visionneuse de documents Info, utilisez la commande **pinfo**. **pinfo** ouvre le répertoire supérieur.

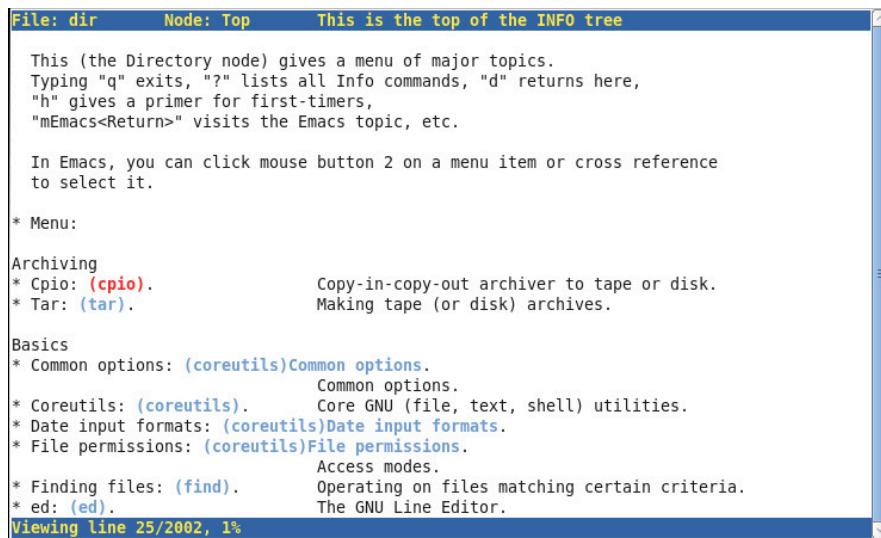


Figure 4.1: La visionneuse de documents Info pinfo, répertoire supérieur

La documentation Info est complète et contient des liens hypertexte. Il est possible de générer des pages Info dans plusieurs formats. En revanche, les pages de manuel sont optimisées pour la sortie imprimée. Le format Info est plus souple que celui des pages de manuel, ce qui permet une discussion plus complète des commandes et concepts complexes. Tout comme les pages de manuel, les nœuds Info sont lus depuis la ligne de commande à l'aide de la commande **pinfo**.

Une page de manuel classique contient une petite quantité de contenu qui se concentre sur un sujet, une commande, un outil ou un fichier en particulier. La documentation Info est un document complet. Elle apporte les améliorations suivantes :

- un seul document pour un grand système contenant toutes les informations nécessaires à ce système ;
- des liens hypertexte ;
- un index complet des documents consultables ;
- une recherche en texte intégral du document entier.

Certains utilitaires et commandes contiennent aussi bien des pages **man** qu'une documentation Info. En règle générale, la documentation Info est plus complète. Comparez les différences dans la documentation de **tar** en utilisant **man** et **pinfo** :

```
[user@host ~]$ man tar
[user@host ~]$ pinfo tar
```

La visionneuse **pinfo** est plus avancée que la commande **info** d'origine. Pour parcourir une rubrique spécifique, utilisez la commande **pinfo topic**. La commande **pinfo** sans argument ouvre le répertoire principal. La nouvelle documentation est disponible dans **pinfo** lorsque les progiciels correspondants sont installés.



NOTE

Si aucune rubrique Info n'existe dans le système pour une entrée particulière que vous avez demandée, Info recherchera une page de manuel correspondante et l'affichera à la place.

COMPARAISON DE LA NAVIGATION DANS GNU INFO ET DANS LES PAGES DE MANUEL

La commande **pinfo** et la commande **man** utilisent des touches de navigation légèrement différentes. Le tableau suivant compare les combinaisons de touches de navigation pour les deux commandes :

pinfo et man, comparaison des raccourcis clavier

NAVIGATION	PINFO	MAN
Faire défiler d'un écran (vers le bas)	Page suivante ou Espace	Page suivante ou Espace
Faire défiler d'un écran (vers le haut)	Page précédente ou b	Page précédente ou b
Afficher le répertoire des rubriques	D	-
Faire défiler d'un demi-écran (vers le bas)	-	D
Afficher le nœud parent d'une rubrique	U	-
Afficher le début d'une rubrique (haut)	ORIGINE	G
Revenir au demi-écran précédent (vers le haut)	-	U
Passer au lien hypertexte suivant (vers le bas)	Flèche vers le bas	-
Ouvrir la rubrique à l'emplacement du curseur	Entrée	-
Faire défiler d'une ligne (vers le bas)	-	Flèche vers le bas ou Entrée
Revenir à l'hyperlien précédent (vers le haut)	Flèche vers le haut	-
Revenir à la ligne précédente (vers le haut)	-	Flèche vers le haut
Permet de rechercher un modèle	<i>/chaîne de caractères</i>	<i>/chaîne de caractères</i>
Afficher le prochain nœud (chapitre) de la rubrique	N	-
Répéter la recherche précédente (vers le bas)	/ puis Entrée	n
Afficher le nœud précédent (chapitre) de la rubrique	P	-
Répéter la recherche précédente en arrière (vers le haut)	-	MajN

NAVIGATION	PINFO	MAN
Quitter le programme	Q	Q



RÉFÉRENCES

pinfo info (*Info: An Introduction*)

pinfo pinfo (*Documentation for pinfo*)

Le projet GNU

<http://www.gnu.org/gnu/thegnuproject.html>

Pages de manuel **pinfo(1)** and **info(1)**

► EXERCICE GUIDÉ

LECTURE DE LA DOCUMENTATION INFO

Dans cet exercice, vous allez rechercher des informations stockées dans des documents GNU Info en les parcourant avec des outils de ligne de commande.

RÉSULTATS

Vous devez pouvoir naviguer dans la documentation GNU Info avec les outils de ligne de commande.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez la commande `lab help-info start`.

```
[student@workstation ~]$ lab help-info start
```

- 1. Sur `workstation` lancez `pinfo` sans aucun argument.

```
[student@workstation ~]$ pinfo
```

- 2. Rendez-vous à la rubrique `Common options`.

Utilisez les touches **Flèche vers le haut** ou **Flèche vers le bas** jusqu'à ce que (`coreutils`) `Common options` soit en surbrillance.

`Basics`

```
* Bash: (bash). The GNU Bourne-Again SHeLL.
* Common options: (coreutils)Common options.
```

Figure 4.2: Documentation bash

- 3. Appuyez sur la touche **Entrée** pour afficher cette rubrique.

```
File: coreutils.info, Node: Common options, Next: Output of entire files, Prev: Introduction, Up: Top

2 Common options
*****  

Certain options are available in all of these programs. Rather than writing identical descriptions for each of the programs, they are described here. (In fact, every GNU program accepts (or should accept) these options.)  

  Normally options and operands can appear in any order, and programs act as if all the options appear before any operands. For example, 'sort -r passwd -t :' acts like 'sort -r -t : passwd', since ':' is an option-argument of '-t'. However, if the 'POSIXLY_CORRECT' environment variable is set, options must appear before operands, unless otherwise specified for a particular command.
```

Figure 4.3: Rubrique Info sur les options courantes

- ▶ 4. Parcourez cette rubrique Info. Voyez si les options longues peuvent être abrégées. Utilisez les touches **Page précédente** et **Page suivante** pour naviguer dans la rubrique. Oui, de nombreux programmes permettent d'abréger les options longues.
- ▶ 5. Déterminez ce que signifient les symboles `--` lorsqu'ils sont utilisés en tant qu'argument de commande.
- Les symboles `--` signifient la fin des options d'une commande *options* et le début des *arguments* pour les commandes complexes où l'analyseur de ligne de commande du shell pourrait avoir des difficultés à les distinguer.
- ▶ 6. Sans quitter **pinfo**, remontez au nœud **GNU Coreutils**. Appuyez sur **u** pour revenir au nœud supérieur de la rubrique.
- ▶ 7. Revenez à la rubrique de niveau supérieur. Appuyez de nouveau sur **u**. Observez que lorsque vous êtes au sommet du nœud d'une rubrique, le fait de remonter vous ramène au répertoire des rubriques. L'autre solution consiste à taper **d** pour revenir directement au répertoire des rubriques, depuis n'importe quel niveau ou n'importe quelle rubrique.
- ▶ 8. Recherchez le motif **coreutils** et sélectionnez cette rubrique. Tapez `/` suivi du motif de recherche « coreutils ». Une fois la rubrique mise en surbrillance, tapez **Entrée**.

```
* Coreutils: (coreutils).      Core GNU (file, text, shell) utilities.
```

Figure 4.4: Résultat de la recherche

- ▶ 9. Dans le menu supérieur, localisez et sélectionnez **Output of entire files** en appuyant **n**. Parcourez la rubrique. Utilisez **Entrée** pour sélectionner **cat invocation**. Utilisez les touches de direction pour parcourir la rubrique.
- ▶ 10. Remontez de deux niveaux pour revenir à **GNU Coreutils**. Accédez à **Summarizing files**. Appuyez sur **Entrée** pour sélectionner la rubrique, puis parcourez-la.
- ▶ 11. Appuyez sur **q** pour quitter **pinfo**.
- ▶ 12. Utilisez à nouveau la commande **pinfo** en spécifiant **coreutils** comme rubrique de destination à partir de la ligne de commande.

```
[student@workstation ~]$ pinfo coreutils
```

- ▶ 13. Sélectionnez la rubrique **Disk usage**. Appuyez sur la touche **Flèche vers le bas** pour mettre en surbrillance **Disk usage** et appuyez sur **Entrée** pour sélectionner cette rubrique.
- ▶ 14. Lisez les sous-rubriques **df invocation** et **du invocation**. Utilisez les touches fléchées pour mettre une rubrique en surbrillance, **Page précédente** et **Page suivante** pour parcourir le texte, et **u** pour remonter d'un niveau. Appuyez sur la touche **q** pour quitter lorsque vous avez terminé.

Fin

Sur workstation, exéutez le script **lab help-info finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab help-info finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

AIDE DANS RED HAT ENTERPRISE LINUX

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez rechercher des informations qui vous aideront à effectuer des tâches dans les pages de manuel et les documents GNU Info.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Localiser les commandes pertinentes en recherchant les pages de manuel et les nœuds Info.
- Apprendre de nouvelles options pour les commandes de documentation classiques.
- Utiliser des outils appropriés pour afficher et imprimer de la documentation et d'autres fichiers au format non textuel.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab help-review start**.

```
[student@workstation ~]$ lab help-review start
```

1. Sur **workstation**, déterminez comment préparer une page de manuel à l'impression. Plus précisément, recherchez le format ou la langue de rendu utilisé pour l'impression.
2. Créez un fichier de sortie formaté de la page de manuel **passwd**. Nommez le fichier **passwd.ps**. Déterminez le format du contenu du fichier. Inspectez le contenu du fichier **passwd.ps**.



NOTE

Créez une sortie formatée de la page de manuel **passwd** en utilisant la commande suivante :

```
[student@workstation ~]$ man -t passwd > passwd.ps
```

Le symbole **>** redirige le contenu de la page de manuel dans le fichier **passwd.ps**. Cette commande est enseignée plus en détail dans un prochain chapitre.

3. Avec **man**, étudiez les commandes qui servent à afficher et imprimer des fichiers PostScript.
4. Apprenez à utiliser la visionneuse **evince(1)** en mode aperçu. Déterminez aussi comment ouvrir un document à partir d'une page spécifique.

CHAPITRE 4 | Aide dans Red Hat Enterprise Linux

5. Affichez votre fichier PostScript à l'aide des diverses options **evince** que vous avez recherchées. Fermez le document quand vous avez terminé.
6. Utilisez la commande **man** pour rechercher **lp(1)** et déterminer comment imprimer un document quelconque à partir d'une page spécifique. Sans vraiment entrer de commande (puisque'il n'y a pas d'imprimante), apprenez la syntaxe, en une commande, pour imprimer uniquement les pages 2 et 3 de votre fichier PostScript.
7. Utilisez **pinfo** pour rechercher de la documentation GNU Info sur la visionneuse **evince**.
8. Utilisez Firefox pour ouvrir le répertoire de documentation sur les paquetages du système et naviguez jusqu'au sous-répertoire du paquetage **man-db**. Affichez les manuels fournis.
9. Utilisez le navigateur Firefox pour localiser le sous-répertoire du paquetage **initscripts**. Affichez le fichier **sysconfig.txt** qui décrit les options de configuration importantes du système stockées dans le répertoire **/etc/sysconfig**.

Évaluation

Sur workstation, exécutez **lab help-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab help-review grade
```

Fin

Sur workstation, exécutez le script **lab help-review finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab help-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

AIDE DANS RED HAT ENTERPRISE LINUX

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez rechercher des informations qui vous aideront à effectuer des tâches dans les pages de manuel et les documents GNU Info.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Localiser les commandes pertinentes en recherchant les pages de manuel et les nœuds Info.
- Apprendre de nouvelles options pour les commandes de documentation classiques.
- Utiliser des outils appropriés pour afficher et imprimer de la documentation et d'autres fichiers au format non textuel.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab help-review start**.

```
[student@workstation ~]$ lab help-review start
```

1. Sur **workstation**, déterminez comment préparer une page de manuel à l'impression. Plus précisément, recherchez le format ou la langue de rendu utilisé pour l'impression.
 - 1.1. Utilisez la commande **man man** pour déterminer comment préparer une page de manuel à l'impression.

```
[student@worksation ~]$ man man
...output omitted...
```

Appuyez sur la touche **q** pour quitter la page de manuel.



NOTE

man utilise **-t** pour préparer une page de manuel à l'impression à l'aide de PostScript.

2. Créez un fichier de sortie formaté de la page de manuel **passwd**. Nommez le fichier **passwd.ps**. Déterminez le format du contenu du fichier. Inspectez le contenu du fichier **passwd.ps**.

**NOTE**

Créez une sortie formatée de la page de manuel **passwd** en utilisant la commande suivante :

```
[student@workstation $]$ man -t passwd > passwd.ps
```

Le symbole **>** redirige le contenu de la page de manuel dans le fichier **passwd.ps**. Cette commande est enseignée plus en détail dans un prochain chapitre.

- 2.1. Utilisez la commande **man -t** pour créer un fichier formaté de la page de manuel **passwd**.

```
[student@workstation ~]$ man -t passwd > passwd.ps
[student@workstation ~]$ ls -al
...output omitted...
-rw-rw-r--. 1 student student 19947 Feb 26 11:14 passwd.ps
...output omitted...
```

- 2.2. Utilisez la commande **file** pour déterminer le format du contenu du fichier.

```
[student@workstation ~]$ file /home/student/passwd.ps
passwd.ps: PostScript document text conforming DSC level 3.0
```

- 2.3. Utilisez la commande **less** pour afficher le fichier **/home/student/passwd.ps**.

```
[student@workstation ~]$ less /home/student/passwd.ps
%!PS-Adobe-3.0
%%Creator: groff version 1.22.3
%%CreationDate: Tue Feb 26 11:14:40 2019
%%DocumentNeededResources: font Times-Roman
%%+ font Times-Bold
%%+ font Times-Italic
%%+ font Symbol
%%DocumentSuppliedResources: procset grops 1.22 3
...output omitted...
```

**NOTE**

La sortie de **file** affirme que le fichier est au format PostScript et que vous avez confirmé ce point en consultant son contenu. Remarquez les informations relatives à PostScript dans les lignes d'en-tête. Utilisez **q** pour quitter la commande **less**.

3. Avec **man**, étudiez les commandes qui servent à afficher et imprimer des fichiers PostScript.

- 3.1. Avec **man**, étudiez les commandes qui servent à afficher et imprimer des fichiers PostScript.

```
[student@workstation ~]# man -k postscript viewer
evince (1)           - GNOME document viewer
evince-previewer (1) - show a printing preview of PostScript and PDF documents
```

```

evince-thumbnailer (1) - create png thumbnails from PostScript and PDF documents
gcm-viewer (1)          - GNOME Color Manager Profile Viewer Tool
gnome-logs (1)          - log viewer for the systemd journal
grops (1)               - PostScript driver for groff
pango-view (1)          - Pango text viewer
pluginviewer (8)         - list loadable SASL plugins and their properties

```

**NOTE**

Utiliser plusieurs mots avec l'option **-k** permet de trouver les pages de manuel qui correspondent à *l'un ou l'autre* de ces mots, ici celles dont la description contient « postscript » ou « viewer ». Remarquez les commandes **evince(1)** dans le résultat.

4. Apprenez à utiliser la visionneuse **evince(1)** en mode aperçu. Déterminez aussi comment ouvrir un document à partir d'une page spécifique.
 - 4.1. Utilisez la commande **man evince** pour apprendre à utiliser la visionneuse en mode aperçu.

```
[student@workstation ~]$ man evince
...output omitted...
```

Appuyez sur la touche **q** pour quitter la page de manuel.

**NOTE**

L'option **-w** (ou **--preview**) permet d'ouvrir **evince** en mode aperçu. L'option **-i** sert à spécifier une page de début.

5. Affichez votre fichier PostScript à l'aide des diverses options **evince** que vous avez recherchées. Fermez le document quand vous avez terminé.
 - 5.1. Utilisez la commande **evince** pour ouvrir **/home/student/passwd.ps**.

```
[student@workstation ~]$ evince /home/student/passwd.ps
```

- 5.2. Utilisez la commande **evince -w /home/student/passwd.ps** pour ouvrir le fichier en mode aperçu.

```
[student@workstation ~]$ evince -w /home/student/passwd.ps
```

- 5.3. Utilisez la commande **evince -i 3 /home/student/passwd.ps** pour ouvrir le fichier à la page 3.

```
[student@workstation ~]$ evince -i 3 /home/student/passwd.ps
```

**NOTE**

Alors que le mode **evince** normal autorise un affichage plein écran et des styles de présentation, le mode aperçu **evince** est utile pour une navigation et une impression rapides. Remarquez l'icône d'impression au sommet.

CHAPITRE 4 | Aide dans Red Hat Enterprise Linux

6. Utilisez la commande **man** pour rechercher **lp(1)** et déterminer comment imprimer un document quelconque à partir d'une page spécifique. Sans vraiment entrer de commande (puisque'il n'y a pas d'imprimante), apprenez la syntaxe, en une commande, pour imprimer uniquement les pages 2 et 3 de votre fichier PostScript.

- 6.1. Utilisez la commande **man lp** pour déterminer comment imprimer des pages spécifiques d'un document.

```
[student@workstation ~]$ man lp  
...output omitted...
```

Appuyez sur la touche **q** pour quitter la page de manuel.

**NOTE**

Dans **lp(1)**, vous apprendrez que l'option **-P** permet de spécifier les pages. La commande **lp** n'envoie au spool de l'imprimante *par défaut* que l'extrait qui commence à la page 2 et finit à la page 3. Par conséquent, une réponse correcte est **lp passwd.ps -P 2-3**.

7. Utilisez **pinfo** pour rechercher de la documentation GNU Info sur la visionneuse **evince**.

- 7.1. Utilisez **pinfo command** pour rechercher de la documentation GNU Info sur la visionneuse **evince**.

```
[student@workstation ~]$ pinfo evince
```

**NOTE**

Remarquez que la page de manuel **evince(1)** s'affiche à sa place. Quand il n'existe aucun nœud de documentation GNU pour le sujet demandé, la visionneuse de documentation **pinfo** recherche une page de manuel pertinente. Appuyez sur la touche **q** pour quitter.

8. Utilisez Firefox pour ouvrir le répertoire de documentation sur les paquetages du système et naviguez jusqu'au sous-répertoire du paquetage **man-db**. Affichez les manuels fournis.

- 8.1. Utilisez **firefox /usr/share/doc** pour afficher la documentation système. Naviguez jusqu'au sous-répertoire **man-db**. Cliquez sur les manuels pour les afficher.

```
[student@workstation ~]$ firefox /usr/share/doc
```

**NOTE**

Des signets peuvent être créés pour tout répertoire fréquemment utilisé. Après avoir navigué jusqu'au répertoire **man-db**, cliquez pour ouvrir et afficher la version texte du manuel, puis refermez-la. Cliquez sur la version PostScript pour l'ouvrir. Comme nous l'avons vu précédemment, **evince** est la visionneuse par défaut du système pour les documents PDF et PostScript. Vous voudrez peut-être revenir à ces documents plus tard pour en apprendre plus sur **man**. Quand vous aurez terminé, fermez la visionneuse **evince**.

Index of file:///usr/share/doc/man-db/		
Up to higher level directory		
Name	Size	Last Modified
File: ChangeLog	51 KB	12/12/16 1:44:30 PM GMT+1
File: NEWS	60 KB	11/7/18 4:46:16 PM GMT+1
File: README	12 KB	12/11/16 12:44:45 AM GMT+1
man-db-manual.ps	129 KB	11/7/18 4:47:06 PM GMT+1
man-db-manual.txt	70 KB	11/7/18 4:47:01 PM GMT+1

9. Utilisez le navigateur Firefox pour localiser le sous-répertoire du paquetage **initscripts**. Affichez le fichier **sysconfig.txt** qui décrit les options de configuration importantes du système stockées dans le répertoire **/etc/sysconfig**.
- 9.1. Dans le navigateur Firefox, localisez le sous-répertoire du paquetage **initscripts**. Remarquez combien un navigateur peut être utile pour localiser et afficher la documentation locale du système. Fermez le document et Firefox quand vous avez terminé.

Évaluation

Sur **workstation**, exécutez **lab help-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab help-review grade
```

Fin

Sur **workstation**, exécutez le script **lab help-review finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab help-review finish
```

L'atelier est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Les pages de manuel sont visualisées avec la commande **man** et fournissent des informations sur les composants d'un système Linux, tels que les fichiers, les commandes et les fonctions.
- Par principe, lorsque vous vous référez à une page de manuel, le nom d'une page est suivi de son numéro de section entre parenthèses.
- Les documents Info sont visualisés avec la commande **pinfo** et sont constitués d'un ensemble de nœuds hypertextes, fournissant des informations sur l'ensemble des progiciels.
- Les touches de navigation utilisées par **man** et **pinfo** sont légèrement différentes.

CHAPITRE 5

CRÉATION, AFFICHAGE ET MODIFICATION DE FICHIERS TEXTE

PROJET

Créer, afficher et modifier des fichiers texte à partir de la sortie d'une commande ou dans un éditeur de texte.

OBJECTIFS

- Enregistrer la sortie de la commande ou les erreurs dans un fichier avec la redirection du shell et traiter la sortie de la commande via plusieurs programmes de ligne de commande avec des pipes.
- Créer et modifier des fichiers texte en utilisant l'éditeur **vim**.
- Utiliser des variables shell pour vous aider à exécuter des commandes et modifier les scripts de démarrage bash pour définir des variables shell et d'environnement afin de modifier le comportement du shell et des programmes exécutés à partir de celui-ci.

SECTIONS

- Redirection de la sortie vers un fichier ou un programme (et quiz)
- Modification de fichiers texte à partir de l'invite du shell (et exercice guidé)
- Modification de l'environnement shell (et exercice guidé)

ATELIER

Création, affichage et modification de fichiers texte

REDIRECTION DE LA SORTIE VERS UN FICHIER OU UN PROGRAMME

OBJECTIFS

Après avoir terminé cette section, vous devriez savoir enregistrer la sortie de la commande ou les erreurs dans un fichier avec la redirection du shell et traiter la sortie de la commande via plusieurs programmes de ligne de commande avec des pipes.

ENTRÉE STANDARD, SORTIE STANDARD ET ERREUR STANDARD

Un programme d'exécution, ou *processus*, doit lire l'entrée depuis un emplacement et écrire la sortie quelque part. Une commande exécutée depuis l'invite du shell lit habituellement son entrée depuis le clavier et envoie sa sortie vers sa fenêtre de terminal.

Un processus utilise des canaux numérotés appelés *descripteurs de fichiers* pour obtenir une entrée et envoyer une sortie. Tous les processus commencent par au moins trois descripteurs de fichier. *Entrée standard* (canal 0) lit les entrées du clavier. *Sortie standard* (canal 1) envoie une sortie normale au terminal. *Erreur standard* (canal 2) envoie des messages d'erreur au terminal. Si un programme ouvre des connexions séparées avec d'autres fichiers, il peut utiliser des descripteurs de fichiers portant des numéros plus élevés.

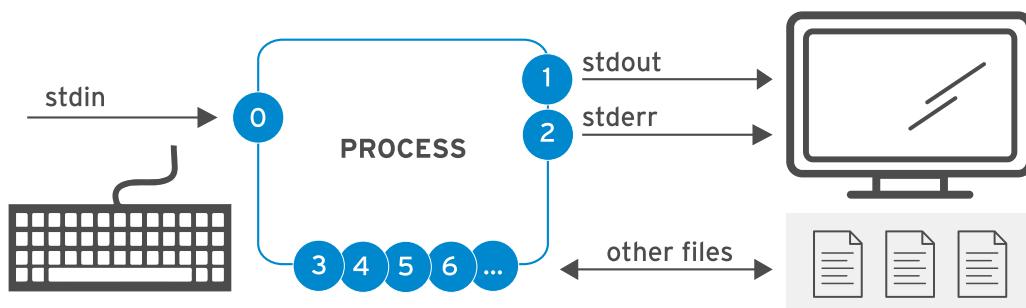


Figure 5.1: Traiter des canaux d'E/S (descripteurs de fichiers)

Canaux (Descripteurs de fichiers)

NUMÉRO	NOM DU CANAL	DESCRIPTION	CONNEXION PAR DÉFAUT	UTILISATION
0	stdin	Entrée standard	Clavier	lecture seule
1	stdout	Sortie standard	Terminal	écriture seule
2	stderr	Erreur standard	Terminal	écriture seule
3+	nom de fichier	Autres fichiers	aucun	lecture et/ou écriture

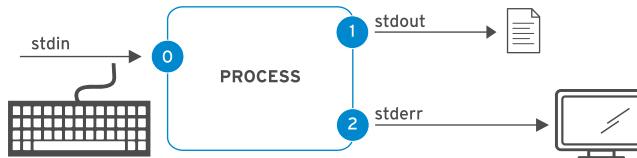
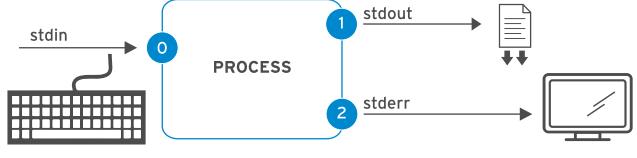
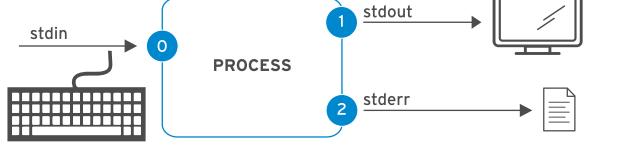
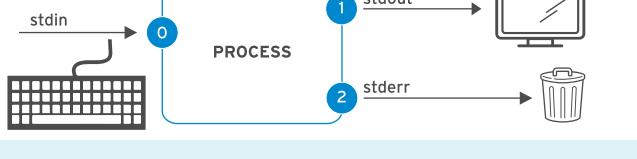
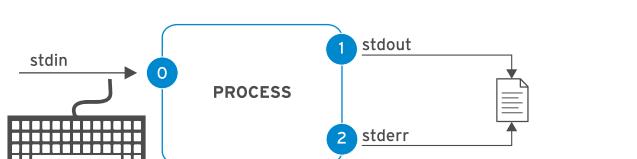
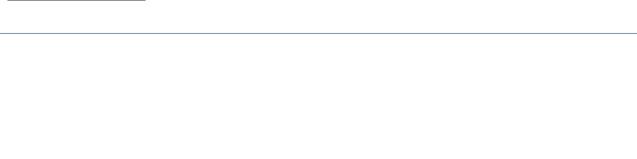
REDIRECTION DE LA SORTIE VERS UN FICHIER

La *redirection* des E/S change la façon dont le processus obtient son entrée ou sa sortie. Au lieu d'obtenir une entrée du clavier ou d'envoyer une sortie et des erreurs au terminal, le processus lit ou écrit dans des fichiers. La redirection vous permet de sauvegarder des messages dans un fichier normalement envoyé à la fenêtre du terminal. Vous pouvez également utiliser la redirection pour supprimer les sorties ou les erreurs, afin qu'elles ne soient ni affichées sur le terminal ni enregistrées.

La redirection de **stdout** empêche l'affichage de la sortie du processus sur le terminal. Comme indiqué dans le tableau suivant, la redirection de **stdout** seulement n'empêche pas l'affichage des messages d'erreur **stderr** sur le terminal. Si le fichier n'existe pas, il sera créé. Si le fichier existe et que la redirection ne s'ajoute pas au fichier, le contenu du fichier sera écrasé.

Le fichier spécial **/dev/null** élimine discrètement la sortie de canal redirigée vers ce fichier et est toujours un fichier vide.

Opérateurs de redirection de sortie

UTILISATION	EXPLICATION	AIDE VISUELLE
<code>> fichier</code>	redirige stdout pour écraser un fichier	
<code>>> fichier</code>	redirige stdout pour l'ajouter à un fichier	
<code>2> fichier</code>	redirige stderr pour écraser un fichier	
<code>2> /dev/null</code>	supprime les messages d'erreur stderr en les redirigeant vers /dev/null	
<code>> fichier 2>&1</code>	redirige stdout et stderr pour écraser le même fichier	
<code>&> fichier</code>	redirige stdout et stderr pour écraser le même fichier	

UTILISATION	EXPLICATION	AIDE VISUELLE
<code>>> fichier 2>&1</code>	redirige stdout et stderr pour les ajouter au même fichier	
<code>&>> fichier</code>		<pre> graph LR K[Keyboard] -- "stdin" --> N0((0)) N0 --> N1((1)) N1 -- "stdout" --> F1[File] N1 <--> N2((2)) N2 -- "stderr" --> F1 </pre>

**IMPORTANT**

L'ordre des opérations de redirection est important. La séquence suivante redirige la sortie standard vers **file** et redirige l'erreur standard vers le même emplacement que la sortie standard (**file**).

```
> file 2>&1
```

Toutefois, la séquence suivante exécute la redirection dans l'ordre inverse. Cette action redirige l'erreur standard vers l'emplacement par défaut pour la sortie standard (la fenêtre de terminal, donc aucune modification), *puis* redirige uniquement la sortie standard vers **file**.

```
2>&1 > file
```

Pour cette raison, certaines personnes préfèrent utiliser les opérateurs de redirection associés :

&>fichier	au lieu	>fichier 2>&1
de		

&>>fichier	au lieu	>>fichier 2>&1 (dans Bash 4 / RHEL 6 et versions ultérieures)
de		

Toutefois, les autres administrateurs système et programmeurs qui utilisent également d'autres shells associés à **bash** (shells compatibles Bourne) pour créer des scripts de commande considèrent que les nouveaux opérateurs de redirection associés doivent être évités, car ils ne sont pas standardisés ni mis en œuvre dans tous ces shells et sont soumis à d'autres limites.

Les auteurs de ce cours adoptent une position neutre sur ce sujet, et les deux syntaxes sont susceptibles de figurer dans le champ.

Exemples de redirection de sortie

De nombreuses tâches d'administration de routine sont simplifiées grâce à la redirection. Utilisez le tableau précédent pour vous aider à déchiffrer les exemples ci-dessous :

- Enregistrer un horodatage pour référence ultérieure.

```
[user@host ~]$ date > /tmp/saved-timestamp
```

- Copier les 100 dernières lignes d'un fichier journal vers un autre fichier.

```
[user@host ~]$ tail -n 100 /var/log/dmesg > /tmp/last-100-boot-messages
```

- Concaténer quatre fichiers en un.

```
[user@host ~]$ cat file1 file2 file3 file4 > /tmp/all-four-in-one
```

- Répertorier le nom des fichiers masqués et standard du répertoire personnel dans un fichier.

```
[user@host ~]$ ls -a > /tmp/my-file-names
```

- Ajouter la sortie à un fichier existant.

```
[user@host ~]$ echo "new line of information" >> /tmp/many-lines-of-information
[user@host ~]$ diff previous-file current-file >> /tmp/tracking-changes-made
```

- Les quelques commandes suivantes génèrent des messages d'erreur, car certains répertoires système sont inaccessibles aux utilisateurs normaux. Observez lorsque les messages d'erreur sont redirigés. Redirigez les erreurs vers un fichier tout en affichant la sortie de la commande normale sur le terminal.

```
[user@host ~]$ find /etc -name passwd 2> /tmp/errors
```

- Enregistrer la sortie des processus et les messages d'erreur dans des fichiers distincts.

```
[user@host ~]$ find /etc -name passwd > /tmp/output 2> /tmp/errors
```

- Ignorer et éliminer les messages d'erreur.

```
[user@host ~]$ find /etc -name passwd > /tmp/output 2> /dev/null
```

- Stocker ensemble la sortie et les erreurs générées.

```
[user@host ~]$ find /etc -name passwd &> /tmp/save-both
```

- Ajouter la sortie et les erreurs générées à un fichier existant.

```
[user@host ~]$ find /etc -name passwd >> /tmp/save-both 2>&1
```

CONSTRUCTION DE PIPELINES

Un *pipeline* est une séquence d'une ou plusieurs commandes séparées par le caractère *pipe* (`|`). Un pipe connecte la sortie standard de la première commande à l'entrée standard de la commande suivante.

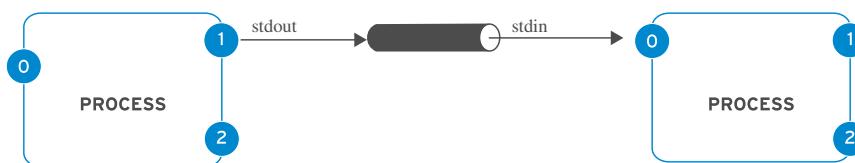


Figure 5.8: Traitement des pipelines d'E/S

Les pipelines permettent la manipulation et le formatage de la sortie d'un processus par d'autres processus avant sa transmission vers le terminal. Une image mentale utile consiste à imaginer que les données « circulent » dans le pipeline d'un processus à un autre, et sont légèrement modifiées par chaque commande du pipeline par lequel elles transitent.

**NOTE**

Les pipelines et la redirection des E/S manipulent la sortie standard et l'entrée standard. La redirection envoie la sortie standard vers des fichiers ou obtient l'entrée standard depuis des fichiers. Les pipes envoient la sortie standard depuis un processus vers l'entrée standard d'un autre processus.

Exemples de pipelines

Dans cet exemple, la sortie de la commande **ls** est affichée sur un écran du terminal à la fois en utilisant **less**.

```
[user@host ~]$ ls -l /usr/bin | less
```

La sortie de la commande **ls** est transmise par le pipe à **wc -l**, qui compte le nombre de lignes reçues de **ls** et l'imprime sur le terminal.

```
[user@host ~]$ ls | wc -l
```

Dans ce pipeline, **head** sort les 10 premières lignes de **ls -t**, avec le résultat final redirigé vers un fichier.

```
[user@host ~]$ ls -t | head -n 10 > /tmp/ten-last-changed-files
```

Pipelines, redirection et commande tee

Lorsque la redirection est associée à un pipeline, le shell configure d'abord l'intégralité du pipeline, puis redirige l'entrée/la sortie. Cela signifie que si une redirection de sortie est utilisée au milieu d'un pipeline, la sortie accède au fichier et non à la commande suivante du pipeline.

Dans cet exemple, la sortie de la commande **ls** accède au fichier et **less** n'affiche rien sur le terminal.

```
[user@host ~]$ ls > /tmp/saved-output | less
```

La commande **tee** surmonte cette limitation. Dans un pipeline, **tee** copie son entrée standard vers sa sortie standard et redirige également sa sortie standard vers les fichiers nommés comme arguments de la commande. En imaginant que les données sont comme de l'eau circulant à travers

un pipeline, **tee** peut être visualisé comme un raccord en « T » dans le tuyau qui dirige la sortie dans deux directions.

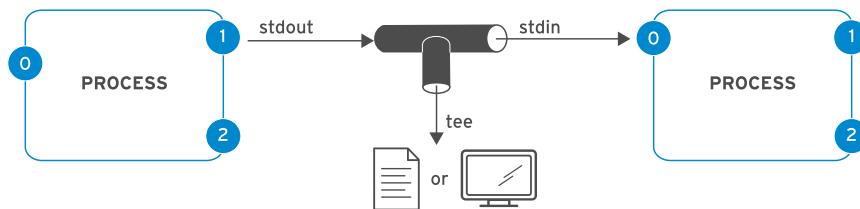


Figure 5.9: Traitement des E/S par pipeline avec la commande tee

Exemples de pipeline utilisant la commande tee

Cet exemple redirige la sortie de la commande **ls** vers le fichier et la transmet à **less** pour l'afficher sur un écran du terminal à la fois.

```
[user@host ~]$ ls -l | tee /tmp/saved-output | less
```

Si **tee** est utilisé à la fin d'un pipeline, la sortie finale d'une commande peut être enregistrée et envoyée vers le terminal en même temps.

```
[user@host ~]$ ls -t | head -n 10 | tee /tmp/ten-last-changed-files
```



IMPORTANT

L'erreur standard peut être redirigée via un pipe, mais les opérateurs de redirection associés (**&>** et **&>>**) ne peuvent pas être utilisés pour cela.

Ci-dessous est présentée la méthode appropriée pour rediriger la sortie standard et l'erreur standard via un pipe :

```
[user@host ~]$ find -name /passwd 2>&1 | less
```



RÉFÉRENCES

info bash (*Le manuel de référence GNU Bash*)

- Section 3.2.2 : pipelines
- Section 3.6 : redirections

info coreutils 'tee invocation' (*Le manuel GNU coreutils*)

- Section 17.1 : Redirect output to multiple files or processes (Rediriger la sortie vers plusieurs fichiers ou processus)

Pages de manuel **bash(1)**, **cat(1)**, **head(1)**, **less(1)**, **mail(1)**, **tee(1)**, **tty(1)**, **wc(1)**

► QUIZ

REDIRECTION DE LA SORTIE VERS UN FICHIER OU UN PROGRAMME

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. **Quelle réponse affiche la sortie sur un terminal et ignore toutes les erreurs ?**
 - a. &>fichier
 - b. 2>>fichier
 - c. 2>/dev/null
 - d. 1>/dev/null

- ▶ 2. **Quelle réponse envoie la sortie vers un fichier et envoie les erreurs vers un autre fichier ?**
 - a. >fichier 2>fichier2
 - b. >fichier 1>fichier2
 - c. >fichier &2>fichier2
 - d. | tee fichier

- ▶ 3. **Quelle réponse envoie à la fois la sortie et les erreurs dans un fichier, le créant ou écrasant son contenu?**
 - a. | tee fichier
 - b. 2 &>fichier
 - c. 1&>fichier
 - d. &>fichier

- ▶ 4. **Quelle réponse envoie la sortie et les erreurs au même fichier en s'assurant que le contenu du fichier existant est préservé ?**
 - a. >fichier 2>fichier2
 - b. &>fichier
 - c. >>fichier 2>&1
 - d. >>fichier 1>&1

- ▶ 5. **Quelle réponse élimine tous les messages normalement envoyés au terminal?**
 - a. >fichier 2>fichier2
 - b. &>/dev/null
 - c. &>/dev/null 2>fichier
 - d. &>fichier

► **6. Quelle réponse envoie la sortie en même temps sur l'écran et dans un fichier ?**

- a. &>/dev/null
- b. >fichier 2>fichier2
- c. | tee fichier
- d. |< fichier

► **7. Quelle réponse enregistre la sortie vers un fichier et élimine les messages d'erreur ?**

- a. &>fichier
- b. | tee fichier 2> /dev/null
- c. > fichier 1> /dev/null
- d. > fichier 2> /dev/null

► SOLUTION

REDIRECTION DE LA SORTIE VERS UN FICHIER OU UN PROGRAMME

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. **Quelle réponse affiche la sortie sur un terminal et ignore toutes les erreurs ?**
 - a. &>fichier
 - b. 2> &>fichier
 - c. 2>/dev/null
 - d. 1>/dev/null

- ▶ 2. **Quelle réponse envoie la sortie vers un fichier et envoie les erreurs vers un autre fichier ?**
 - a. >fichier 2>fichier2
 - b. >fichier 1>fichier2
 - c. >fichier &2>fichier2
 - d. | tee fichier

- ▶ 3. **Quelle réponse envoie à la fois la sortie et les erreurs dans un fichier, le créant ou écrasant son contenu?**
 - a. | tee fichier
 - b. 2 &>fichier
 - c. 1&>fichier
 - d. &>fichier

- ▶ 4. **Quelle réponse envoie la sortie et les erreurs au même fichier en s'assurant que le contenu du fichier existant est préservé ?**
 - a. >fichier 2>fichier2
 - b. &>fichier
 - c. >>fichier 2>&1
 - d. >>fichier 1>&1

- ▶ 5. **Quelle réponse élimine tous les messages normalement envoyés au terminal?**
 - a. >fichier 2>fichier2
 - b. &>/dev/null
 - c. &>/dev/null 2>fichier
 - d. &>fichier

► **6. Quelle réponse envoie la sortie en même temps sur l'écran et dans un fichier ?**

- a. &>/dev/null
- b. >fichier 2>fichier2
- c. | tee fichier
- d. | < fichier

► **7. Quelle réponse enregistre la sortie vers un fichier et élimine les messages d'erreur ?**

- a. &>fichier
- b. | tee fichier 2>/dev/null
- c. > fichier 1>/dev/null
- d. > fichier 2>/dev/null

MODIFICATION DE FICHIERS TEXTE À PARTIR DE L'INVITE DU SHELL

OBJECTIFS

Après avoir terminé cette section, vous devriez pouvoir créer et éditer des fichiers texte à partir de la ligne de commande en utilisant l'éditeur **vim**.

MODIFICATION DE FICHIERS AVEC VIM

L'un des principes clés à la base de la conception de Linux repose sur le fait que les informations et les paramètres de configuration sont stockés dans des fichiers texte. Ces fichiers peuvent être structurés de différentes manières, sous forme de listes de paramètres, de formats de type INI, en tant que fichier XML ou YAML structuré, etc. Cependant, l'avantage des fichiers texte est qu'ils peuvent être visualisés et édités à l'aide de n'importe quel éditeur de texte.

Vim est une version améliorée de l'éditeur **vi** fourni avec les systèmes Linux et UNIX. Vim est hautement configurable et efficace pour les utilisateurs avertis. Il comprend des fonctionnalités comme la division de l'écran d'édition, le formatage par couleurs et la mise en surbrillance du texte en cours d'édition.

Pourquoi apprendre Vim ?

Vous devez savoir utiliser au moins un éditeur de texte pouvant être utilisé depuis une invite shell texte uniquement. Si c'est le cas, vous pouvez éditer des fichiers texte de configuration à partir d'une fenêtre de terminal ou à partir de connexions distantes via **ssh** ou la console Web. Ensuite, vous n'avez pas besoin d'accéder à un bureau graphique pour éditer des fichiers sur un serveur. En fait, il est possible que ce serveur n'ait pas besoin d'un environnement de bureau graphique du tout.

Mais alors, pourquoi apprendre Vim au lieu des autres options possibles? La raison principale est que Vim est presque toujours installé sur un serveur, si un éditeur de texte est présent. Ceci est dû au fait que **vi** a été spécifié par le standard POSIX auquel Linux et beaucoup d'autres systèmes d'exploitation UNIX sont en grande partie conformes.

De plus, Vim est souvent utilisé comme mise en œuvre **vi** sur d'autres distributions ou systèmes d'exploitation courants. Par exemple, macOS comprend actuellement une installation légère de Vim par défaut. Ainsi, les compétences acquises par Vim pour Linux pourraient également vous aider à réaliser d'autres choses.

Commencer Vim

Vim peut être installé sur Red Hat Enterprise Linux de deux manières différentes. Cela peut avoir un impact sur les fonctions et les commandes Vim disponibles.

Il est possible que seul le paquet *vim-minimal* soit installé sur votre serveur. Il s'agit d'une installation très légère comprenant uniquement les fonctions principales et les commandes **vi** de base. Dans ce cas, vous pouvez ouvrir un fichier pour le modifier avec **vi filename** et toutes les fonctions principales abordées dans cette section seront disponibles.

Si non, le paquet *vim-enhanced* peut être installé sur votre serveur. Il fournit un ensemble de fonctions beaucoup plus complet, un système d'aide en ligne et un programme de tutoriel. Pour lancer Vim dans ce mode amélioré, vous utilisez la commande **vim**.

```
[user@host ~]$ vim filename
```

Dans les deux cas, les fonctions principales dont nous parlerons dans cette section fonctionneront avec les deux commandes.

NOTE

Si *vim-enhanced* est installé, les utilisateurs réguliers auront un alias de shell défini, de sorte que s'ils exécutent la commande **vi**, ils obtiendront automatiquement la commande **vim** à la place. Cela ne s'applique pas aux utilisateurs **root** et aux autres utilisateurs avec des UID inférieurs à 200 (qui sont utilisés par les services système).

Si vous éditez des fichiers en tant qu'utilisateur **root** et que vous pensez que **vi** s'exécute en mode amélioré, cela peut vous surprendre. De même, si *vim-enhanced* est installé et qu'un utilisateur régulier veut la commande **vi** simple pour une raison quelconque, il pourrait avoir besoin d'utiliser **\vi** pour remplacer temporairement l'alias.

Les utilisateurs avancés peuvent utiliser **\vi --version** et **vim --version** pour comparer les jeux de fonctions des deux commandes.

Modes de fonctionnement de Vim

Une caractéristique inhabituelle de Vim est qu'il propose plusieurs *modes de fonctionnement*, y compris le *mode commande*, le *mode commande étendu*, le *mode édition* et le *mode visuel*. Selon le mode, vous pouvez émettre des commandes, modifier du texte ou utiliser des blocs de texte. En tant que nouvel utilisateur de Vim, vous devez toujours être conscient de votre mode actuel, car les raccourcis clavier ont des effets différents selon les modes.

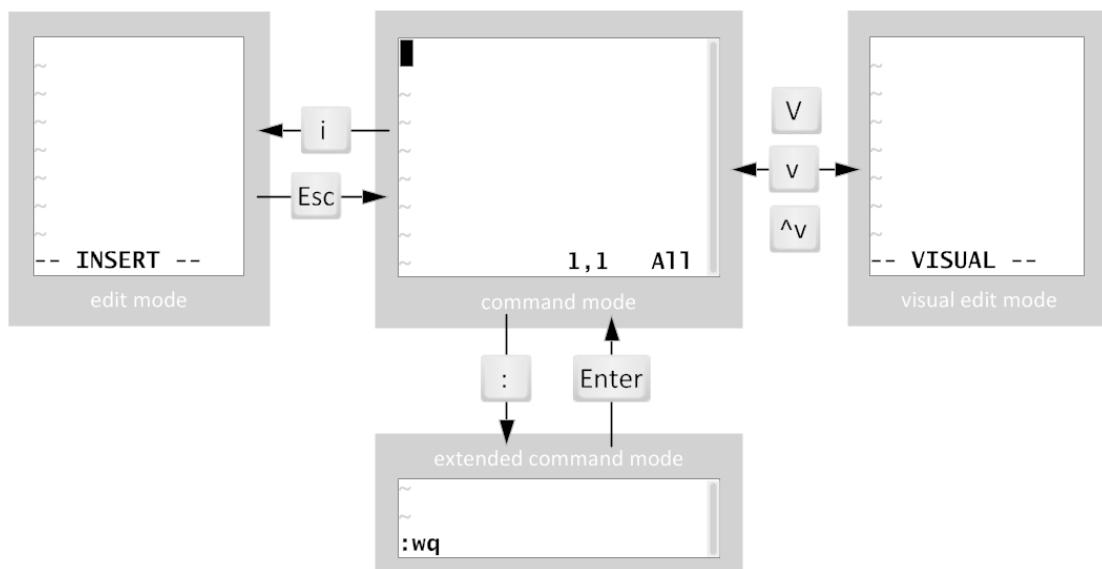


Figure 5.10: Permutation entre les modes Vim

Lorsque vous ouvrez Vim pour la première fois, il démarre en *mode commande* qui est utilisé pour la navigation, les couper-coller et d'autres manipulations de texte. Entrez dans chacun des

CHAPITRE 5 | Création, affichage et modification de fichiers texte

autres modes avec un raccourci clavier à une touche pour accéder à des fonctionnalités d'édition spécifiques :

- Une pression sur **i** ouvre le *mode d'insertion*, dans lequel tout le texte saisi s'ajoute au contenu du fichier. Appuyez sur **Échap** pour revenir au mode de commande.
- Une pression sur **v** ouvre le *mode visuel*, dans lequel il est possible de sélectionner des caractères multiples pour manipuler le texte. Utilisez **Maj+V** pour sélectionner plusieurs lignes et **Ctrl+V** pour sélectionner un bloc. Le même raccourci (**v**, **Maj+V** ou **Ctrl+V**) est utilisé pour passer en mode visuel et pour en sortir.
- La touche **:** ouvre le *mode commande étendu* pour effectuer des tâches comme écrire un fichier (pour l'enregistrer), ou encore quitter l'éditeur Vim.



NOTE

Si vous n'êtes pas sûr du mode dans lequel se trouve Vim, vous pouvez appuyer sur **Échap** quelques fois pour revenir en mode commande. Appuyer sur **Échap** en mode commande ne présente aucun risque, il n'est donc pas gênant d'appuyer quelques fois supplémentaires sur la touche.

Workflow de base et minimal de Vim

Vim offre des combinaisons de touches efficaces et coordonnées pour effectuer des tâches d'édition avancées. Bien qu'elles deviennent utiles avec de la pratique, les possibilités de Vim peuvent désorienter les nouveaux utilisateurs.

La touche **i** place Vim en mode insertion. Tout le texte saisi par la suite est traité comme du contenu de fichier jusqu'à ce que vous quittiez le mode insertion. La touche **Échap** permet de quitter le mode insertion et replace Vim en mode commande. La touche **u** annulera l'édition la plus récente. Appuyez sur la touche **x** pour supprimer un seul caractère. La commande **:w** écrit (enregistre) le fichier et permet de rester en mode commande afin de continuer l'édition. La commande **:wq** écrit (enregistre) le fichier et permet de quitter Vim. La commande **:q!** permet de quitter Vim en annulant toutes les modifications apportées au fichier depuis la dernière écriture. L'utilisateur Vim doit apprendre ces commandes pour accomplir toute tâche d'édition.

Réorganisation de texte existant

Dans Vim, le copier-coller s'appelle *yank and put* (arracher et mettre) et recourt aux caractères de commande **y** et **p**. Commencez par placer le curseur sur le premier caractère à sélectionner, puis passez en mode visuel. Utilisez les touches de direction pour étendre la sélection visuelle. Une fois la sélection terminée, appuyez sur **y** pour extraire (*yank*) la sélection en mémoire. Positionnez le curseur sur le nouvel emplacement et appuyez sur **p** pour *placer* la sélection à l'emplacement du curseur.

Mode visuel dans Vim

Le mode visuel est un excellent moyen de mettre en surbrillance et de manipuler du texte. Il existe trois raccourcis clavier :

- Mode caractère : **v**
- Mode ligne : **Maj+v**
- Mode bloc : **Ctrl+v**

Le mode caractère met en surbrillance les phrases dans un bloc de texte. Le mot **VISUEL** apparaîtra en bas de l'écran. Appuyez sur **v** pour passer en mode caractère visuel. **Maj+v** permet de passer en mode ligne. **LIGNE VISUELLE** apparaîtra en bas de l'écran.

Le mode bloc visuel est parfait pour manipuler des fichiers de données. Depuis le curseur, appuyez sur **Ctrl+v** pour entrer dans le bloc visuel. **BLOC VISUEL** apparaîtra en bas de l'écran. Utilisez les touches fléchées pour mettre en surbrillance la section à modifier.



NOTE

Vim dispose de nombreuses fonctionnalités, mais vous devez d'abord maîtriser le workflow de base. Vous n'avez pas besoin de connaître rapidement tout l'éditeur et ses fonctionnalités. Familiarisez-vous avec ces bases en pratiquant, puis enrichissez votre vocabulaire Vim en apprenant d'autres commandes Vim (raccourcis clavier).

L'exercice de cette section vous présentera la commande **vimtutor**. Ce tutoriel, fourni avec le paquet *vim-enhanced*, est un excellent moyen d'apprendre les fonctions principales de Vim.



RÉFÉRENCES

Page de manuel (1)**vim**

La commande :**help** dans **vim**(si le paquet *vim-enhanced* est installé).

L'éditeur Vim

<http://www.vim.org/>

Démarrer avec le mode visuel de Vim

<https://opensource.com/article/19/2/getting-started-vim-visual-mode>

► EXERCICE GUIDÉ

MODIFICATION DE FICHIERS TEXTE À PARTIR DE L'INVITE DU SHELL

Dans cet exercice, vous utiliserez **vimtutor** pour mettre en pratique des techniques d'édition basiques dans l'éditeur vim.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Modifier les fichiers en utilisant Vim.
- Acquérir des compétences dans Vim à l'aide de **vimtutor**.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab edit-vim start**. Ce script vérifie que le serveur cible est en cours d'exécution.

```
[student@workstation ~]$ lab edit-vim start
```

- 1. Utilisez la commande **ssh** pour vous connecter à **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Ouvrez **vimtutor**. Lisez l'écran d'accueil et lancez la *Leçon 1.1*.

```
[student@servera ~]$ vimtutor
```

Dans la présentation, les touches de direction du clavier ont été utilisées pour naviguer. Quand **vi** a été développé pour la première fois, les utilisateurs ne pouvaient pas compter sur les touches de direction ni sur les mappages de clavier actifs pour déplacer le curseur. Par conséquent, **vi** a d'abord été conçu pour déplacer le curseur à l'aide de commandes utilisant des touches de clavier standard, telles que les touches **H**, **J**, **K** et **L** opportunément groupées.

Voici un moyen de les mémoriser :

hang back, jump down, kick up, leap forward.

- 3. Dans la fenêtre **vimtutor**, suivez la *Leçon 1.2*.

Cette première leçon apprend aux utilisateurs à quitter vim sans conserver une modification non souhaitée. Tous les changements sont perdus. Parfois, il vaut mieux adopter cette solution plutôt que de laisser un fichier essentiel dans un état incorrect.

► 4. Dans la fenêtre **vimtutor**, suivez la *Leçon 1.3*.

vim propose des combinaisons de touches plus efficaces pour supprimer un nombre exact de mots, de lignes, de phrases et de paragraphes. Cependant, tout travail de modification peut être accompli en utilisant **x** pour la suppression d'un caractère.

► 5. Dans la fenêtre **vimtutor**, suivez la *Leçon 1.4*.

Pour la plupart des tâches d'édition, la première touche utilisée est le **i**.

► 6. Dans la fenêtre **vimtutor**, suivez la *Leçon 1.5*.

Dans ce cours, seule la commande **i** (*insérer*) est enseignée. C'est la touche qui sert à passer en mode édition. Cette leçon **vimtutor** démontre que d'autres touches sont disponibles pour changer la position du curseur en mode insertion. En mode insertion, tout texte tapé est du contenu de fichier.

► 7. Dans la fenêtre **vimtutor**, suivez la *Leçon 1.6*.

Tapez :**wq** pour enregistrer le fichier et quitter l'éditeur.

► 8. Dans la fenêtre **vimtutor**, lisez le *résumé de la Leçon 1*.

La commande **vimtutor** inclut six autres leçons contenant plusieurs étapes. Ces leçons ne font pas partie de ce cours, mais n'hésitez pas à les explorer par vous-même pour en apprendre davantage.

► 9. Quittez **servera**.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Fin

Sur **workstation**, exécutez le script **lab edit-vim finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab edit-vim finish
```

L'exercice guidé est maintenant terminé.

MODIFICATION DE L'ENVIRONNEMENT SHELL

OBJECTIFS

Après avoir terminé cette section, vous devriez être capable de définir des variables shell pour vous aider à exécuter des commandes et modifier les scripts de démarrage bash pour définir des variables shell et d'environnement afin de modifier le comportement du shell et des programmes exécutés à partir de celui-ci.

UTILISATION DE VARIABLES SHELL

Le shell bash vous permet de définir des *variables shell* que vous pouvez utiliser pour aider à exécuter des commandes ou pour modifier le comportement du shell. Vous pouvez également exporter des variables shell en tant que *variables d'environnement* qui sont automatiquement copiées dans les programmes exécutés à partir de ce shell au démarrage. Vous pouvez utiliser des variables pour faciliter l'exécution d'une commande avec un argument long ou pour appliquer un paramètre commun aux commandes exécutées à partir de ce shell.

Les variables shell sont uniques à une session shell particulière. Si vous avez deux fenêtres de terminal ouvertes ou deux sessions de connexion indépendantes sur le même serveur distant, vous exéutez deux shells. Chaque shell a son propre ensemble de valeurs pour ses variables shell.

Attribution de valeurs aux variables

Attribuez une valeur à une variable shell à l'aide de la syntaxe suivante :

```
VARIABLENAME=value
```

Les noms de variables peuvent contenir des lettres majuscules ou minuscules, des chiffres et le caractère de soulignement (_). Par exemple, les commandes suivantes définissent des variables shell :

```
[user@host ~]$ COUNT=40
[user@host ~]$ first_name=John
[user@host ~]$ file1=/tmp/abc
[user@host ~]$ _ID=RH123
```

N'oubliez pas que cette modification affecte uniquement le shell dans lequel vous exécutez la commande, et non pas les autres shells que vous exécutez sur ce serveur.

Vous pouvez utiliser la commande **set** pour lister toutes les variables shell actuellement définies. (Elle liste également toutes les fonctions du shell que vous pouvez ignorer.) Cette liste est suffisamment longue pour que vous souhaitiez canaliser la sortie dans la commande **less** afin de l'afficher page par page.

```
[user@host ~]$ set | less
BASH=/usr/bin/bash
BASHOPTS=checkwinsize:cmdhist:complete_fullquote:expand_aliases:extglob:extquote:
force_fignore:histappend:interactive_comments:progcomp:promptvars:sourcepath
BASHRCSSOURCED=Y
...output omitted...
```

Récupération de valeurs avec une extension de variable

Vous pouvez utiliser l'*extension de variable* pour faire référence à la valeur d'une variable que vous avez définie. Pour ce faire, faites précédé le nom de la variable d'un signe dollar (\$). Dans l'exemple suivant, la commande **echo** affiche le reste de la ligne de commande saisie, mais une fois l'extension de variable effectuée.

Par exemple, la commande suivante définit la variable shell COUNT sur **40**.

```
[user@host ~]$ COUNT=40
```

Si vous entrez la commande **echo COUNT**, cela affichera la chaîne **COMPTER**.

```
[user@host ~]$ echo COUNT
COUNT
```

Mais si vous entrez la commande **echo \$COUNT**, cela affichera la valeur de la variable COUNT.

```
[user@host ~]$ echo $COUNT
40
```

Un exemple plus pratique serait d'utiliser une variable pour désigner un nom de fichier long pour plusieurs commandes.

```
[user@host ~]$ file1=/tmp/tmp.z9pXW0HqcC
[user@host ~]$ ls -l $file1
-rw----- 1 student student 1452 Jan 22 14:39 /tmp/tmp.z9pXW0HqcC
[user@host ~]$ rm $file1
[user@host ~]$ ls -l $file1
total 0
```

**IMPORTANT**

S'il existe des caractères de fin adjacents au nom de la variable, vous devrez peut-être protéger le nom de la variable par des accolades. Vous pouvez toujours utiliser des accolades dans les extensions de variables, mais vous verrez également de nombreux exemples dans lesquels ils ne sont pas nécessaires et sont omis.

Dans l'exemple suivant, la première commande **echo** tente d'étendre la variable inexistante **COUNTx**, ce qui ne provoque pas d'erreur, mais ne renvoie rien.

```
[user@host ~]$ echo Repeat $COUNTx
Repeat
[user@host ~]$ echo Repeat ${COUNT}x
Repeat 40x
```

Configuration de bash avec des variables shell

Certaines variables du shell sont définies au démarrage de bash mais peuvent être modifiées pour ajuster le comportement du shell.

Par exemple, deux variables shell qui affectent l'historique du shell et la commande **history** sont **HISTFILE** et **HISTFILESIZE**. Si **HISTFILE** est définie, elle spécifie l'emplacement d'un fichier dans lequel enregistrer l'historique du shell à la fermeture de ce dernier. Par défaut, il s'agit du fichier de l'utilisateur **~/.bash_history**. La variable **HISTFILESIZE** spécifie le nombre de commandes devant être enregistrées dans ce fichier à partir de l'historique.

Un autre exemple est **PS1**, qui est une variable shell contrôlant l'apparence de l'invite du shell. Si vous modifiez cette valeur, l'apparence de votre invite de shell sera différente. Un certain nombre d'extensions de caractères spéciaux prises en charge par l'invite sont listés dans la section « PROMPTING » de la page de manuel **bash(1)**.

```
[user@host ~]$ PS1="bash\$ "
bash$ PS1="[\u@\h \w]\$ "
[user@host ~]$
```

Deux éléments à noter à propos de l'exemple ci-dessus : premièrement, comme la valeur définie par **PS1** est une invite, il est pratiquement toujours souhaitable de terminer l'invite par un espace de fin. Deuxièmement, chaque fois que la valeur d'une variable contient un espace sous n'importe quelle forme, à savoir un espace, une tabulation ou un retour, la valeur doit être encadrée par des guillemets simples ou doubles. Ce n'est pas optionnel. Des résultats inattendus se produisent si les guillemets sont omis. Examinez l'exemple **PS1** ci-dessus et notez qu'il est conforme à la recommandation (espace de fin) et la règle (guillemets).

CONFIGURATION DE PROGRAMMES AVEC DES VARIABLES D'ENVIRONNEMENT

Le shell fournit un *environnement* aux programmes que vous exécutez à partir de ce shell. Entre autres choses, cet environnement inclut des informations sur le répertoire de travail en cours du système de fichiers, les options de ligne de commande transmises au programme, et les valeurs des *variables d'environnement*. Les programmes peuvent utiliser ces variables d'environnement pour modifier leur comportement ou leurs paramètres par défaut.

CHAPITRE 5 | Création, affichage et modification de fichiers texte

Les variables shell qui ne sont pas des variables d'environnement ne peuvent être utilisées que par le shell. Les variables d'environnement peuvent être utilisées par le shell et par des programmes exécutés à partir de ce shell.



NOTE

Les variables `HISTFILE`, `HISTFILESIZE` et `PS1` que vous avez étudiées dans la section précédente n'ont pas besoin d'être exportées en tant que variables d'environnement, car elles ne sont utilisées que par le shell lui-même et non par les programmes que vous exécutez à partir du shell.

Vous pouvez transformer toute variable définie dans le shell en une variable d'environnement en la marquant pour exportation avec la commande `export`.

```
[user@host ~]$ EDITOR=vim  
[user@host ~]$ export EDITOR
```

Vous pouvez définir et exporter une variable en une seule étape :

```
[user@host ~]$ export EDITOR=vim
```

Les applications et les sessions utilisent ces variables pour déterminer leur comportement. Par exemple, le shell définit automatiquement la variable `HOME` avec le nom de fichier du répertoire personnel de l'utilisateur au démarrage. Ceci peut être utilisé pour aider les programmes à déterminer l'emplacement d'enregistrement des fichiers.

Un autre exemple est `LANG`, qui définit les paramètres régionaux. Cette variable ajuste la langue préférée de la sortie du programme ; le jeu de caractères ; le formatage des dates, des chiffres et de la devise ; et l'ordre de tri des programmes. S'il est réglé sur `en_US.UTF-8`, le paramètre régional utilise l'anglais américain avec l'encodage de caractères Unicode UTF-8. S'il est réglé sur autre chose, par exemple `fr_FR.UTF-8`, il utilisera l'encodage de caractères Unicode UTF-8 français.

```
[user@host ~]$ date  
Tue Jan 22 16:37:45 CST 2019  
[user@host ~]$ export LANG=fr_FR.UTF-8  
[user@host ~]$ date  
mar. janv. 22 16:38:14 CST 2019
```

Une autre variable d'environnement importante est `PATH`. La variable `PATH` contient une liste de répertoires séparés des deux-points contenant des programmes :

```
[user@host ~]$ echo $PATH  
/home/user/.local/bin:/home/user/bin:/usr/share/Modules/bin:/usr/local/bin:/usr/  
bin:/usr/local/sbin:/usr/sbin
```

Lorsque vous exécutez une commande telle que `ls`, le shell recherche le fichier exécutable `ls` dans chacun de ces répertoires dans l'ordre, et exécute le premier fichier correspondant qu'il trouve. (Sur un système classique, il s'agit de `/usr/bin/ls`.)

Vous pouvez facilement ajouter des répertoires supplémentaires à la fin de votre `PATH`. Par exemple, vous avez peut-être des programmes ou des scripts exécutables à exécuter comme des

CHAPITRE 5 | Création, affichage et modification de fichiers texte

commandes normales dans **/home/user/sbin**. Vous pouvez ajouter **/home/user/sbin** à la fin de votre PATH pour la session en cours comme ceci :

```
[user@host ~]$ export PATH=${PATH}:/home/user/sbin
```

Pour lister toutes les variables d'environnement d'un shell particulier, exécutez la commande **env** :

```
[user@host ~]$ env  
...output omitted...  
LANG=en_US.UTF-8  
HISTCONTROL=ignoredups  
HOSTNAME=host.example.com  
XDG_SESSION_ID=4  
...output omitted...
```

Configuration de l'éditeur de texte par défaut

La variable d'environnement **EDITOR** spécifie le programme que vous souhaitez utiliser comme éditeur de texte par défaut pour les programmes de ligne de commande. De nombreux programmes utilisent **vi** ou **vim** s'il n'est pas spécifié, mais vous pouvez remplacer cette préférence si nécessaire :

```
[user@host ~]$ export EDITOR=nano
```



IMPORTANT

Par convention, les variables d'environnement et les variables shell qui sont automatiquement définies par le shell ont des noms qui utilisent toutes les lettres majuscules. Si vous définissez vos propres variables, vous pouvez utiliser des noms composés de minuscules pour éviter les conflits de noms.

PARAMÉTRAGE AUTOMATIQUE DES VARIABLES

Si vous souhaitez définir automatiquement des variables d'environnement ou du shell au démarrage de votre shell, vous pouvez modifier les scripts de démarrage bash. Au démarrage de bash, plusieurs fichiers texte contenant des commandes shell sont exécutés pour initialiser l'environnement shell.

Les scripts exacts qui s'exécutent dépendent de la manière dont le shell a été lancé, qu'il s'agisse d'un shell de connexion interactif, d'un shell sans connexion ou d'un script shell.

Avec les fichiers **/etc/profile**, **/etc/bashrc** et **~/.bash_profile** par défaut, si vous souhaitez modifier votre compte d'utilisateur qui affecte toutes les invites de votre shell interactif au démarrage, modifiez votre fichier **~/.bashrc**. Par exemple, vous pouvez définir l'éditeur par défaut de ce compte sur **nano** en modifiant le fichier comme suit :

```
# .bashrc  
  
# Source global definitions  
if [ -f /etc/bashrc ]; then  
    . /etc/bashrc  
fi
```

```
# User specific environment
PATH="$HOME/.local/bin:$HOME/bin:$PATH"
export PATH

# User specific aliases and functions
export EDITOR=nano
```

**NOTE**

Le meilleur moyen d'ajuster les paramètres qui affectent tous les comptes d'utilisateur est d'ajouter un fichier dont le nom se termine par **.sh** contenant les modifications dans le répertoire **/etc/profile.d**. Pour ce faire, vous devez être connecté en tant qu'utilisateur root.

ANNULATION DE LA DÉFINITION ET DE L'EXPORTATION DES VARIABLES

Pour annuler complètement la définition et l'exportation d'une variable, utilisez la commande **unset** :

```
[user@host ~]$ echo $file1
/tmp/tmp.z9pXW0HqcC
[user@host ~]$ unset file1
[user@host ~]$ echo $file1

[user@host ~]$
```

Pour annuler l'exportation d'une variable sans en annuler la définition, utilisez la commande **export -n** :

```
[user@host ~]$ export -n PS1
```

**RÉFÉRENCES**

Pages du manuel **bash(1)**, **env(1)** et **builtins(1)**

► EXERCICE GUIDÉ

MODIFICATION DE L'ENVIRONNEMENT SHELL

Dans cet exercice, vous allez utiliser des variables shell et leur extension pour exécuter des commandes et définir une variable d'environnement pour régler l'éditeur par défaut selon des nouveaux shells.

RÉSULTATS :

Vous devez pouvoir réaliser les tâches suivantes :

- Modifier le profil de l'utilisateur.
- Créer une variable shell.
- Créer une variable d'environnement.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez la commande `lab edit-shell start`. Ce script vérifie que le serveur cible est en cours d'exécution.

```
[student@workstation ~]$ lab edit-shell start
```

- 1. Remplacez la variable shell `PS1` de l'utilisateur `student` par `[\u@\h \t \w]$` (rappelez-vous de mettre la valeur de `PS1` entre guillemets et de mettre un espace de fin après le signe dollar). Cela va ajouter le temps à l'invite.

- 1.1. Sur `workstation`, utilisez la commande `ssh` pour vous connecter à `servera`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 1.2. Utilisez Vim pour éditer le fichier de configuration `~/.bashrc`.

```
[student@servera ~]$ vim ~/.bashrc
```

- 1.3. Ajoutez la variable shell `PS1` et sa valeur au fichier `~/.bashrc`. N'oubliez pas d'inclure, à la fin de la valeur que vous définissez, un espace de fin et de placer la valeur entière entre guillemets, y compris l'espace de fin.

CHAPITRE 5 | Création, affichage et modification de fichiers texte

```
...output omitted...
# User specific environment and startup programs
PATH=$PATH:$HOME/.local/bin:$HOME/bin
PS1='[\u@\h \t \w]$ '
export PATH
```

- 1.4. Quittez servera et reconnectez-vous en utilisant la commande **ssh** pour mettre à jour l'invite de commande.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera 14:45:05 ~]$
```

- ▶ 2. Attribuez une valeur à une variable shell locale. Les noms de variables peuvent contenir des lettres majuscules ou minuscules, des chiffres et le caractère de soulignement. Récupérez la valeur de la variable.
- 2.1. Créez une variable appelée **file** avec la valeur **tmp.zdkei083**. Le fichier **tmp.zdkei083** existe dans le répertoire personnel student.

```
[student@servera 14:47:05 ~]$ file=tmp.zdkei083
```

- 2.2. Récupérez la valeur de la variable **file**.

```
[student@servera 14:48:35 ~]$ echo $file
tmp.zdkei083
```

- 2.3. Utilisez le nom de la variable **file** et la commande **ls -l** pour lister les fichiers **tmp.zdkei083**. Utilisez la commande **rm** et le nom de la variable **file** pour supprimer le fichier **tmp.zdkei083**. Confirmez qu'il a été supprimé.

```
[student@servera 14:59:07 ~]$ ls -l $file
-rw-rw-r-- 1 student student 0 Jan 23 14:59 tmp.zdkei083
[student@servera 14:59:10 ~]$ rm $file
[student@servera 14:59:15 ~]$ ls -l $file
ls: cannot access 'tmp.zdkei083': No such file or directory
```

- ▶ 3. Attribuez une valeur à la variable **editor**. Utilisez une commande pour transformer la variable en variable d'environnement.

```
[student@servera 14:46:40 ~]$ export EDITOR=vim
[student@servera 14:46:55 ~]$ echo $EDITOR
vim
```

- ▶ 4. Quittez servera.

```
[student@servera 14:47:11 ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Fin

Sur workstation, exécutez le script **lab edit-shell finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab edit-shell finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

CRÉATION, AFFICHAGE ET MODIFICATION DE FICHIERS TEXTE

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez modifier un fichier texte à l'aide de l'éditeur **vim**.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Utiliser Vim pour modifier un fichier.
- Utiliser le mode visuel pour simplifier la modification de fichiers.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab edit-review start**.

```
[student@workstation ~]$ lab edit-review start
```

1. Redirigez une longue liste de tous les contenus du répertoire personnel du stagiaire, y compris les répertoires et fichiers cachés, vers un fichier nommé **editing_final_lab.txt**.
2. Modifiez le fichier en utilisant Vim.
3. Supprimez les trois premières lignes. Passez en mode visuel sur ligne avec un **V** majuscule.
4. Supprimer les colonnes sur la première ligne. Passez en mode visuel avec un **v** minuscule. **v** minuscule sélectionne les caractères sur une seule ligne. Les colonnes après **-rw-** devraient être supprimées.
5. Supprimer les colonnes et le point suivant (".") sur les lignes restantes. Utilisez le mode bloc visuel. Passez en mode bloc visuel en utilisant le raccourci-clavier **Ctrl+V**. Utilisez cette séquence de touches pour sélectionner un bloc de caractères sur plusieurs lignes. Les colonnes après **-rw-** devraient être supprimées.
6. Utilisez le mode bloc visuel pour supprimer la quatrième colonne.
7. Utilisez le mode bloc visuel pour supprimer la colonne « time », mais laissez le mois et le jour sur toutes les lignes.
8. Supprimez les lignes **Desktop** et **Public**. Passez en mode visuel avec un **V** majuscule.
9. Utilisez la commande **:wq** pour enregistrer et fermer le fichier. Effectuez une sauvegarde en utilisant la date (en secondes) pour créer un nom de fichier unique.
10. Ajoutez une ligne en pointillé au fichier. La ligne en pointillé doit contenir au moins 12 tirets.

11. Ajoutez une liste de répertoires du répertoire **Documents**. Créez la liste de répertoires sur le terminal et envoyez-la vers le fichier **editing_final_lab.txt** à l'aide d'une ligne de commande.
12. Vérifiez que la liste de répertoires se trouve à la fin du fichier de l'atelier.

Évaluation

Sur workstation, exécutez la commande **lab edit-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab edit-review grade
```

Fin

Sur workstation, exécutez le script **lab edit-review finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab edit-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

CRÉATION, AFFICHAGE ET MODIFICATION DE FICHIERS TEXTE

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez modifier un fichier texte à l'aide de l'éditeur **vim**.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Utiliser Vim pour modifier un fichier.
- Utiliser le mode visuel pour simplifier la modification de fichiers.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab edit-review start**.

```
[student@workstation ~]$ lab edit-review start
```

1. Redirigez une longue liste de tous les contenus du répertoire personnel du stagiaire, y compris les répertoires et fichiers cachés, vers un fichier nommé **editing_final_lab.txt**.



NOTE

Il est possible que la sortie ne corresponde pas exactement aux exemples présentés.

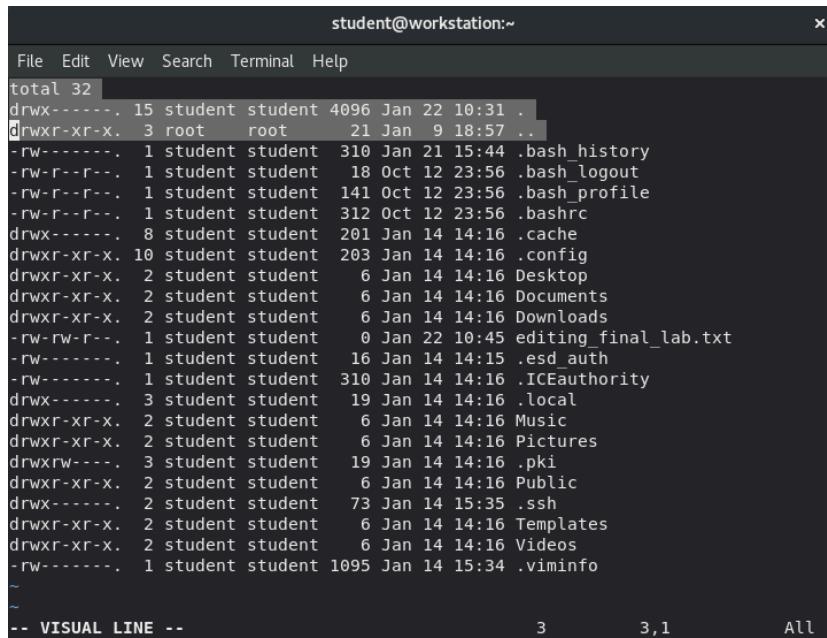
Dans **workstation**, depuis le répertoire personnel **student**, utilisez la commande **ls -al** pour rediriger une longue liste de tout le contenu vers un fichier nommé **editing_final_lab.txt**.

```
[student@workstation ~]$ ls -al > editing_final_lab.txt
```

2. Modifiez le fichier en utilisant Vim.

```
[student@workstation ~]$ vim editing_final_lab.txt
```

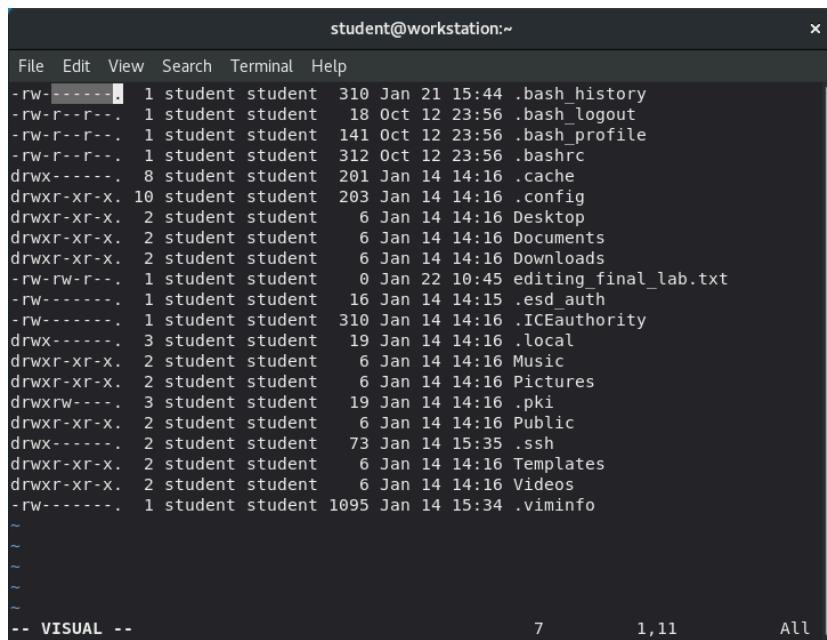
3. Supprimez les trois premières lignes. Passez en mode visuel sur ligne avec un **V** majuscule. Utilisez les touches de direction pour positionner le curseur sur le premier caractère de la première ligne. Passez en mode visuel sur ligne avec **Maj+V**. Déplacez la sélection vers le bas en appuyant deux fois sur la flèche vers le bas afin de sélectionner les trois premières lignes. Supprimez les lignes en utilisant **x**.

CHAPITRE 5 | Création, affichage et modification de fichiers texte


```
student@workstation:~$ total 32
drwx----- 15 student student 4096 Jan 22 10:31 .
d[rw-r--r-- 3 root root 21 Jan 9 18:57 ..
-rw----- 1 student student 310 Jan 21 15:44 .bash_history
-rw-r--r-- 1 student student 18 Oct 12 23:56 .bash_logout
-rw-r--r-- 1 student student 141 Oct 12 23:56 .bash_profile
-rw-r--r-- 1 student student 312 Oct 12 23:56 .bashrc
drwx----- 8 student student 201 Jan 14 14:16 .cache
drwxr-xr-x 10 student student 203 Jan 14 14:16 .config
drwxr-xr-x 2 student student 6 Jan 14 14:16 Desktop
drwxr-xr-x 2 student student 6 Jan 14 14:16 Documents
drwxr-xr-x 2 student student 6 Jan 14 14:16 Downloads
-rw-rw-r-- 1 student student 0 Jan 22 10:45 editing_final_lab.txt
-rw----- 1 student student 16 Jan 14 14:15 .esd_auth
-rw----- 1 student student 310 Jan 14 14:16 .ICEauthority
drwx----- 3 student student 19 Jan 14 14:16 .local
drwxr-xr-x 2 student student 6 Jan 14 14:16 Music
drwxr-xr-x 2 student student 6 Jan 14 14:16 Pictures
drwxr-w----
```

- 4.** Supprimer les colonnes sur la première ligne. Passez en mode visuel avec un **v** minuscule. **v** minuscule sélectionne les caractères sur une seule ligne. Les colonnes après **-rw-** devraient être supprimées.

Utilisez les touches de direction pour positionner le curseur sur le premier caractère. Passez en mode visuel avec un **v** minuscule. Utilisez les touches de direction pour positionner le curseur sur le dernier caractère. Supprimez la sélection en tapant **x**.



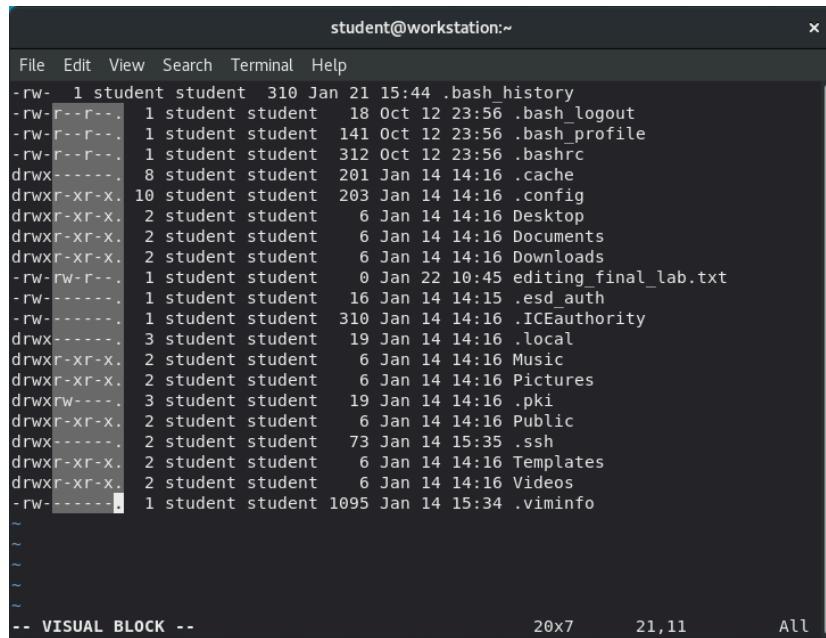
```
student@workstation:~$ total 32
-rw----- 1 student student 310 Jan 21 15:44 .bash_history
-rw-r--r-- 1 student student 18 Oct 12 23:56 .bash_logout
-rw-r--r-- 1 student student 141 Oct 12 23:56 .bash_profile
-rw-r--r-- 1 student student 312 Oct 12 23:56 .bashrc
drwx----- 8 student student 201 Jan 14 14:16 .cache
drwxr-xr-x 10 student student 203 Jan 14 14:16 .config
drwxr-xr-x 2 student student 6 Jan 14 14:16 Desktop
drwxr-xr-x 2 student student 6 Jan 14 14:16 Documents
drwxr-xr-x 2 student student 6 Jan 14 14:16 Downloads
-rw-rw-r-- 1 student student 0 Jan 22 10:45 editing_final_lab.txt
-rw----- 1 student student 16 Jan 14 14:15 .esd_auth
-rw----- 1 student student 310 Jan 14 14:16 .ICEauthority
drwx----- 3 student student 19 Jan 14 14:16 .local
drwxr-xr-x 2 student student 6 Jan 14 14:16 Music
drwxr-xr-x 2 student student 6 Jan 14 14:16 Pictures
drwxr-w----
```

- 5.** Supprimer les colonnes et le point suivant (".") sur les lignes restantes. Utilisez le mode bloc visuel. Passez en mode bloc visuel en utilisant le raccourci-clavier **Ctrl+V**. Utilisez cette séquence de touches pour sélectionner un bloc de caractères sur plusieurs lignes. Les colonnes après **-rw-** devraient être supprimées.

Utilisez les touches de direction pour positionner le curseur sur le premier caractère. Passez en mode visuel en utilisant le raccourci-clavier **Ctrl+V**. Utilisez les touches de direction

CHAPITRE 5 | Création, affichage et modification de fichiers texte

pour positionner le curseur au niveau du dernier caractère de la colonne sur la dernière ligne. Supprimez la sélection avec **X**.



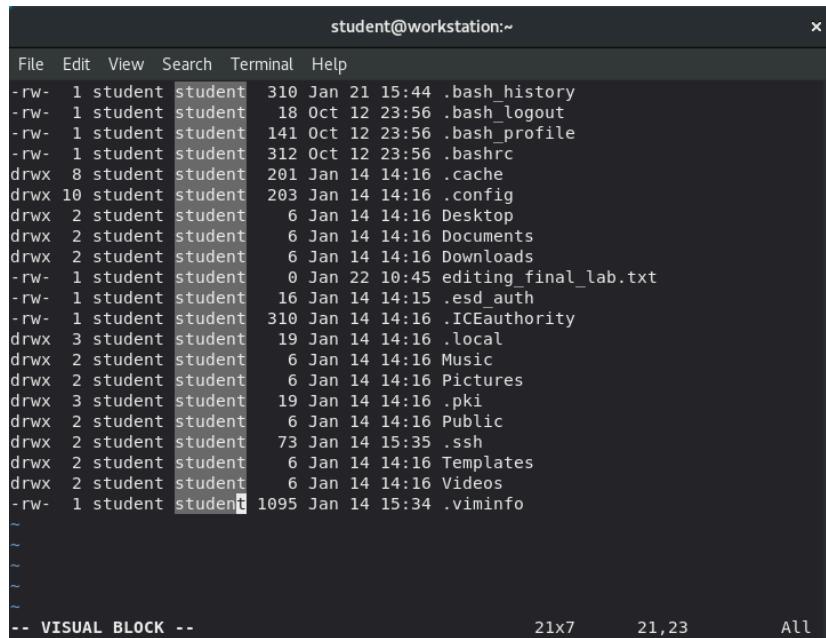
```
student@workstation:~
```

```
File Edit View Search Terminal Help
-rw- 1 student student 310 Jan 21 15:44 .bash_history
-rw-r--r--. 1 student student 18 Oct 12 23:56 .bash_logout
-rw-r--r--. 1 student student 141 Oct 12 23:56 .bash_profile
-rw-r--r--. 1 student student 312 Oct 12 23:56 .bashrc
drwx-----. 8 student student 201 Jan 14 14:16 .cache
drwxr-xr-x. 10 student student 203 Jan 14 14:16 .config
drwxr-xr-x. 2 student student 6 Jan 14 14:16 Desktop
drwxr-xr-x. 2 student student 6 Jan 14 14:16 Documents
drwxr-xr-x. 2 student student 6 Jan 14 14:16 Downloads
-rw-rw-r--. 1 student student 0 Jan 22 10:45 editing_final_lab.txt
-rw-----. 1 student student 16 Jan 14 14:15 .esd_auth
-rw-----. 1 student student 310 Jan 14 14:16 .ICEAuthority
drwx-----. 3 student student 19 Jan 14 14:16 .local
drwxr-xr-x. 2 student student 6 Jan 14 14:16 Music
drwxr-xr-x. 2 student student 6 Jan 14 14:16 Pictures
drwxrw----. 3 student student 19 Jan 14 14:16 .pki
drwxr-xr-x. 2 student student 6 Jan 14 14:16 Public
drwx-----. 2 student student 73 Jan 14 15:35 .ssh
drwxr-xr-x. 2 student student 6 Jan 14 14:16 Templates
drwxr-xr-x. 2 student student 6 Jan 14 14:16 Videos
-rw-----. 1 student student 1095 Jan 14 15:34 .viminfo
~
~
~
~
~
-- VISUAL BLOCK --
```

20x7 21,11 All

- 6.** Utilisez le mode bloc visuel pour supprimer la quatrième colonne.

Utilisez les touches de direction pour positionner le curseur sur le premier caractère de la quatrième colonne. Passez en mode bloc visuel avec **Ctrl+V**. Utilisez les touches de direction pour positionner le curseur sur le dernier caractère de la quatrième colonne. Supprimez la sélection avec **X**.



```
student@workstation:~
```

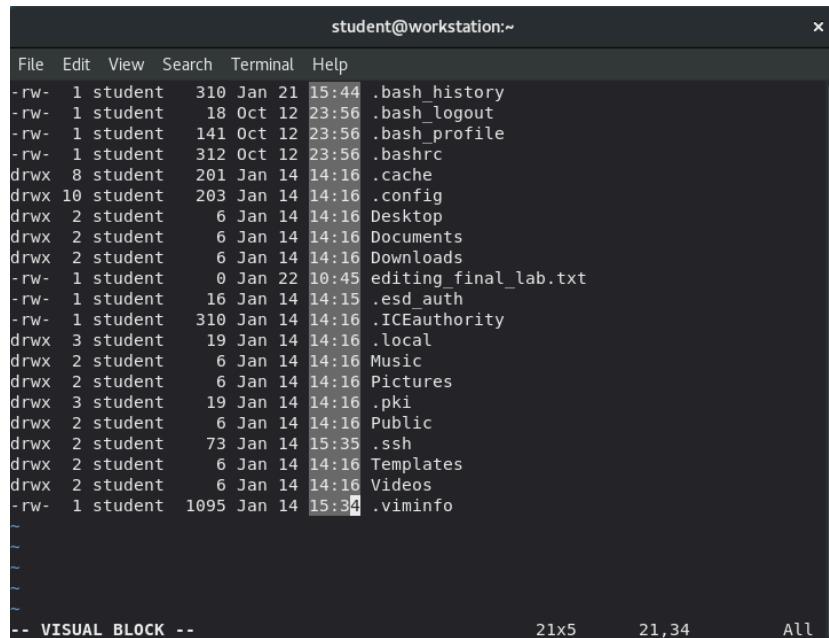
```
File Edit View Search Terminal Help
-rw- 1 student student 310 Jan 21 15:44 .bash_history
-rw- 1 student student 18 Oct 12 23:56 .bash_logout
-rw- 1 student student 141 Oct 12 23:56 .bash_profile
-rw- 1 student student 312 Oct 12 23:56 .bashrc
drwx 8 student student 201 Jan 14 14:16 .cache
drwx 10 student student 203 Jan 14 14:16 .config
drwx 2 student student 6 Jan 14 14:16 Desktop
drwx 2 student student 6 Jan 14 14:16 Documents
drwx 2 student student 6 Jan 14 14:16 Downloads
-rw- 1 student student 0 Jan 22 10:45 editing_final_lab.txt
-rw- 1 student student 16 Jan 14 14:15 .esd_auth
-rw- 1 student student 310 Jan 14 14:16 .ICEAuthority
drwx 3 student student 19 Jan 14 14:16 .local
drwx 2 student student 6 Jan 14 14:16 Music
drwx 2 student student 6 Jan 14 14:16 Pictures
drwx 3 student student 19 Jan 14 14:16 .pki
drwx 2 student student 6 Jan 14 14:16 Public
drwx 2 student student 73 Jan 14 15:35 .ssh
drwx 2 student student 6 Jan 14 14:16 Templates
drwx 2 student student 6 Jan 14 14:16 Videos
-rw- 1 student student 1095 Jan 14 15:34 .viminfo
~
~
~
~
~
-- VISUAL BLOCK --
```

21x7 21,23 All

- 7.** Utilisez le mode bloc visuel pour supprimer la colonne « time », mais laissez le mois et le jour sur toutes les lignes.

Utilisez les touches de direction pour positionner le curseur sur le premier caractère. Passez en mode bloc visuel avec **Ctrl+V**. Utilisez les touches de direction pour positionner le

curseur sur le dernier caractère de la dernière ligne de la colonne « time ». Supprimez la sélection en tapant **x**.



```
student@workstation:~
```

```

File Edit View Search Terminal Help
-rw- 1 student 310 Jan 21 15:44 .bash_history
-rw- 1 student 18 Oct 12 23:56 .bash_logout
-rw- 1 student 141 Oct 12 23:56 .bash_profile
-rw- 1 student 312 Oct 12 23:56 .bashrc
drwx 8 student 201 Jan 14 14:16 .cache
drwx 10 student 203 Jan 14 14:16 .config
drwx 2 student 6 Jan 14 14:16 Desktop
drwx 2 student 6 Jan 14 14:16 Documents
drwx 2 student 6 Jan 14 14:16 Downloads
-rw- 1 student 0 Jan 22 10:45 editing_final_lab.txt
-rw- 1 student 16 Jan 14 14:15 .esd_auth
-rw- 1 student 310 Jan 14 14:16 .ICEauthority
drwx 3 student 19 Jan 14 14:16 .local
drwx 2 student 6 Jan 14 14:16 Music
drwx 2 student 6 Jan 14 14:16 Pictures
drwx 3 student 19 Jan 14 14:16 .pki
drwx 2 student 6 Jan 14 14:16 Public
drwx 2 student 73 Jan 14 15:35 .ssh
drwx 2 student 6 Jan 14 14:16 Templates
drwx 2 student 6 Jan 14 14:16 Videos
-rw- 1 student 1095 Jan 14 15:34 .viminfo
~  

~  

~  

~  

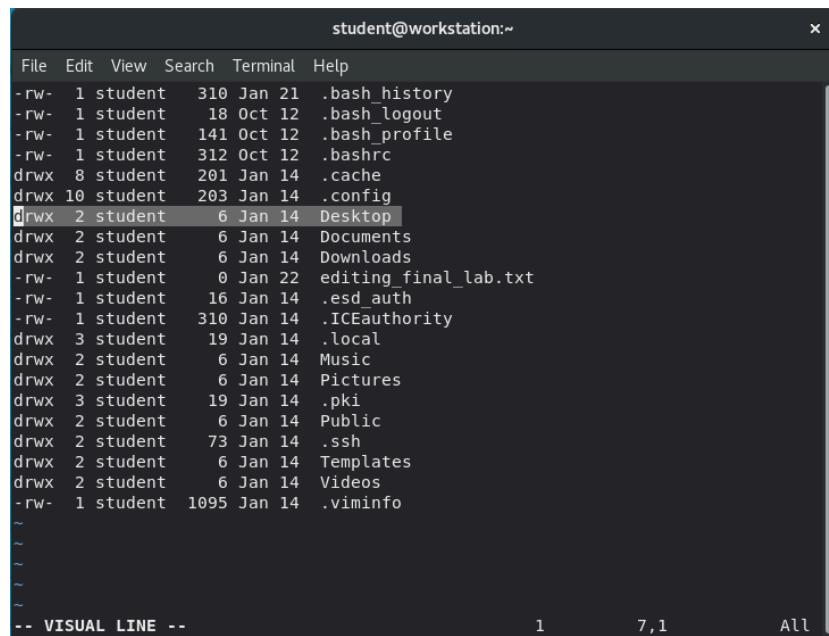
~  

-- VISUAL BLOCK --
```

21x5 21,34 All

- 8.** Supprimez les lignes **Desktop** et **Public**. Passez en mode visuel avec un **V** majuscule.

Utilisez les touches de direction pour placer le curseur sur un caractère quelconque de la ligne **Desktop**. Passez en mode visuel avec un **V** majuscule. La ligne complète est sélectionnée. Supprimez la sélection en tapant **x**. Répétez l'opération pour la ligne **Public**.



```
student@workstation:~
```

```

File Edit View Search Terminal Help
-rw- 1 student 310 Jan 21 .bash_history
-rw- 1 student 18 Oct 12 .bash_logout
-rw- 1 student 141 Oct 12 .bash_profile
-rw- 1 student 312 Oct 12 .bashrc
drwx 8 student 201 Jan 14 .cache
drwx 10 student 203 Jan 14 14:16 .config
drwx 2 student 6 Jan 14 14:16 Desktop
drwx 2 student 6 Jan 14 14:16 Documents
drwx 2 student 6 Jan 14 14:16 Downloads
-rw- 1 student 0 Jan 22 editing_final_lab.txt
-rw- 1 student 16 Jan 14 .esd_auth
-rw- 1 student 310 Jan 14 .ICEauthority
drwx 3 student 19 Jan 14 14:16 .local
drwx 2 student 6 Jan 14 14:16 Music
drwx 2 student 6 Jan 14 14:16 Pictures
drwx 3 student 19 Jan 14 14:16 .pki
drwx 2 student 6 Jan 14 14:16 Public
drwx 2 student 73 Jan 14 15:35 .ssh
drwx 2 student 6 Jan 14 14:16 Templates
drwx 2 student 6 Jan 14 14:16 Videos
-rw- 1 student 1095 Jan 14 15:34 .viminfo
~  

~  

~  

~  

~  

~  

-- VISUAL LINE --
```

1 7,1 All

9. Utilisez la commande :wq pour enregistrer et fermer le fichier. Effectuez une sauvegarde en utilisant la date (en secondes) pour créer un nom de fichier unique.

```
[student@workstation ~]$ cp editing_final_lab.txt \
editing_final_lab_$(date +%-s).txt
```

- 10.** Ajoutez une ligne en pointillé au fichier. La ligne en pointillé doit contenir au moins 12 tirets.

```
[student@workstation ~]$ echo "-----" \
>> editing_final_lab.txt
```

11. Ajoutez une liste de répertoires du répertoire **Documents**. Créez la liste de répertoires sur le terminal et envoyez-la vers le fichier **editing_final_lab.txt** à l'aide d'une ligne de commande.

```
[student@workstation ~]$ ls Documents/ | tee -a editing_final_lab.txt  
lab review.txt
```

- 12.** Vérifiez que la liste de répertoires se trouve à la fin du fichier de l'atelier.

```
[student@workstation ~]$ cat editing_final_lab.txt
-rw- 1 student 310 Jan 21 .bash_history
-rw- 1 student 18 Oct 12 .bash_logout
-rw- 1 student 141 Oct 12 .bash_profile
-rw- 1 student 312 Oct 12 .bashrc
drwx 8 student 201 Jan 14 .cache
drwx 10 student 203 Jan 14 .config
drwx 2 student 6 Jan 14 Documents
drwx 2 student 6 Jan 14 Downloads
-rw- 1 student 0 Jan 22 editing_final_lab.txt
-rw- 1 student 16 Jan 14 .esd_auth
-rw- 1 student 310 Jan 14 .ICEauthority
drwx 3 student 19 Jan 14 .local
```

```
drwx  2 student    6 Jan 14  Music
drwx  2 student    6 Jan 14  Pictures
drwx  3 student    19 Jan 14  .pki
drwx  2 student    73 Jan 14  .ssh
drwx  2 student    6 Jan 14  Templates
drwx  2 student    6 Jan 14  Videos
-rw-  1 student  1095 Jan 14  .viminfo
-----
lab_review.txt
```

Évaluation

Sur workstation, exéutez la commande **lab edit-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab edit-review grade
```

Fin

Sur workstation, exéutez le script **lab edit-review finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab edit-review finish
```

L'atelier est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Les programmes ou processus en cours d'exécution ont trois canaux de communication standard, une entrée standard, une sortie standard et une erreur standard.
- Vous pouvez utiliser redirection des E/S pour lire l'entrée standard d'un fichier ou écrire la sortie ou les erreurs d'un processus dans un fichier.
- Les pipelines peuvent être utilisés pour connecter une sortie standard d'un processus à une entrée standard d'un autre processus, et peuvent également être utilisés pour formater la sortie ou créer des commandes complexes.
- Vous devriez savoir comment utiliser au moins un éditeur de texte de ligne de commande, et Vim est généralement installé.
- Les variables shell peuvent vous aider à exécuter des commandes et sont uniques à une session shell particulière.
- Les variables d'environnement peuvent vous aider à configurer le comportement du shell ou les processus qu'il démarre.

CHAPITRE 6

GESTION DES UTILISATEURS ET DES GROUPES LOCAUX

PROJET

Créer, gérer et supprimer les utilisateurs et groupes locaux, et administrer les politiques locales relatives aux mots de passe.

OBJECTIFS

- Décrire l'objet des utilisateurs et des groupes sur un système Linux.
- Se connecter en tant que super utilisateur pour gérer un système Linux et accorder à d'autres utilisateurs un accès super utilisateur à l'aide de la commande **sudo**.
- Créer, modifier et supprimer des comptes d'utilisateur définis localement.
- Créer, modifier et supprimer des comptes de groupes définis localement.
- Définir une politique de gestion des mots de passe pour les utilisateurs, ainsi que verrouiller et déverrouiller manuellement les comptes d'utilisateur.

SECTIONS

- Descriptions des concepts relatifs aux utilisateurs et aux groupes (et quiz)
- Accès en tant que super utilisateur (et exercice guidé)
- Gestion des comptes d'utilisateur locaux (et exercice guidé)
- Gestion des comptes de groupes locaux (et exercice guidé)
- Gestion des mots de passe des utilisateurs (et exercice guidé)

ATELIER

Gestion des utilisateurs et groupes Linux locaux

DESCRIPTION DES CONCEPTS D'UTILISATEUR ET DE GROUPE

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir décrire le rôle des utilisateurs et des groupes sur un système Linux.

QU'EST-CE QU'UN UTILISATEUR ?

Un compte d'*utilisateur* est utilisé pour fournir des limites de sécurité entre différentes personnes et divers programmes pouvant exécuter des commandes.

Les utilisateurs ont des *noms d'utilisateur* pour les identifier aux utilisateurs humains et faciliter le travail. En interne, le système distingue les comptes d'utilisateurs du numéro d'identification unique qui leur est attribué, l'*identifiant d'utilisateur* ou *UID*. Si un compte d'utilisateur est utilisé par des humains, un *mode de passe* confidentiel lui sera généralement attribué. L'utilisateur s'en servira pour prouver qu'il est bien l'utilisateur autorisé lors de la connexion.

Les comptes d'utilisateurs sont essentiels à la sécurité du système. Chaque processus (programme en cours d'exécution) du système s'exécute avec le nom d'un utilisateur particulier. Chaque fichier est la propriété d'un utilisateur particulier. La propriété des fichiers permet au système d'appliquer le contrôle d'accès aux utilisateurs des fichiers. L'utilisateur auquel un processus en cours d'exécution est associé détermine à quels fichiers et répertoires ce processus peut accéder.

Les trois principaux types de comptes d'utilisateur sont *super utilisateur*, *utilisateur système* et *utilisateur normal*.

- Le compte *super utilisateur* est destiné à l'administration du système. Le nom du super utilisateur est `root` et le compte est associé à l'UID 0. Le super utilisateur dispose d'un accès complet au système.
- Le système a des comptes *utilisateur système* utilisés par des processus fournissant des services de support. Ces processus, ou *démons*, n'ont généralement pas besoin de s'exécuter en tant que super utilisateur. Il s'agit de comptes sans privilège assignés qui leur permettent de sécuriser leurs fichiers et d'autres ressources les uns des autres et des utilisateurs standards du système. Les utilisateurs ne se connectent pas de manière interactive à l'aide d'un compte d'utilisateur système.
- La plupart des utilisateurs ont des comptes *d'utilisateurs standards* qu'ils utilisent pour leur travail quotidien. Comme les utilisateurs système, les utilisateurs standards ont un accès limité au système.

Vous pouvez utiliser la commande `id` pour afficher des informations sur l'utilisateur actuellement connecté.

```
[user01@host ~]$ id
uid=1000(user01) gid=1000(user01) groups=1000(user01)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

CHAPITRE 6 | Gestion des utilisateurs et des groupes locaux

Pour afficher des informations basiques sur un autre utilisateur, transmettez le nom d'utilisateur à la commande **id** comme argument.

```
[user01@host]$ id user02  
uid=1002(user02) gid=1001(user02) groups=1001(user02)  
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Pour afficher le propriétaire d'un fichier, utilisez la commande **ls -l**. Pour afficher le propriétaire d'un répertoire, utilisez la commande **ls -ld**. Dans la sortie suivante, la troisième colonne indique le nom de l'utilisateur.

```
[user01@host ~]$ ls -l file1  
-rw-rw-r--. 1 user01 user01 0 Feb 5 11:10 file1  
[user01@host]$ ls -ld dir1  
drwxrwxr-x. 2 user01 user01 6 Feb 5 11:10 dir1
```

Pour afficher des informations relatives au processus, utilisez la commande **ps**. Par défaut, seul les processus du shell actuel sont affichés. Ajoutez l'option **a** pour afficher tous les processus liés à un terminal. Pour afficher l'utilisateur associé à un processus, ajoutez l'option **u**. Dans la sortie suivante, la première colonne indique le nom de l'utilisateur.

```
[user01@host]$ ps -au  
USER          PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND  
root          777  0.0  0.0 225752  1496 tty1      Ss+  11:03   0:00 /sbin/agetty -o -  
p -- \u --noclear tty1 linux  
root          780  0.0  0.1 225392  2064 ttys0      Ss+  11:03   0:00 /sbin/agetty -o -  
p -- \u --keep-baud 115200,38400,9600  
user01        1207  0.0  0.2 234044  5104 pts/0      Ss   11:09   0:00 -bash  
user01        1319  0.0  0.2 266904  3876 pts/0      R+  11:33   0:00 ps au
```

Le résultat de la commande précédente affiche les utilisateurs par nom, mais en interne, le système d'exploitation les repère à l'aide de leur identifiant utilisateur. La mise en correspondance des noms d'utilisateur et des UID est définie dans les bases de données des informations sur les comptes. Par défaut, les systèmes utilisent le fichier **/etc/passwd** pour stocker les informations concernant les utilisateurs locaux.

Chaque ligne du fichier **/etc/passwd** contient des informations sur un utilisateur. Il est divisé en sept champs séparés par deux-points. Voici un exemple de ligne de **/etc/passwd** :

```
①user01:②x:③1000:④1000:⑤User One:⑥/home/user01:⑦/bin/bash
```

- ① Nom d'utilisateur pour cet utilisateur (**user01**).
- ② Le mot de passe de l'utilisateur était stocké ici au format crypté. Il a été déplacé au fichier **/etc/shadow**, qui sera abordé plus loin. Ce champ doit toujours être x .
- ③ Le numéro UID de ce compte utilisateur (**1000**).
- ④ Le numéro GID du groupe principal de ce compte utilisateur (**1000**). Les groupes seront abordés plus loin dans cette section.
- ⑤ Le vrai nom de cet utilisateur (**User One**).
- ⑥ Le répertoire personnel de cet utilisateur (**/home/user01**). Il s'agit du répertoire de travail initial au démarrage du shell. Il contient les données de l'utilisateur et les paramètres de configuration.

CHAPITRE 6 | Gestion des utilisateurs et des groupes locaux

- 7 Le programme shell par défaut de cet utilisateur, qui s'exécute lors de la connexion (**/bin/bash**). Pour un utilisateur normal, il s'agit généralement du programme qui fournit l'invite de ligne de commande de l'utilisateur. Un utilisateur système peut utiliser **/sbin/nologin** si les connexions interactives ne sont pas autorisées pour cet utilisateur.

QU'EST-CE QU'UN GROUPE ?

Un groupe est un ensemble d'utilisateurs devant partager l'accès aux fichiers et aux autres ressources du système. Les groupes peuvent être utilisés pour accorder l'accès à des fichiers à un ensemble d'utilisateurs plutôt qu'à un seul utilisateur.

Comme les utilisateurs, les groupes ont des *noms de groupe* pour faciliter le travail. En interne, le système distingue les comptes d'utilisateurs du numéro d'identification unique qui leur est attribué, l'*identifiant d'utilisateur ou UID*.

La mise en correspondance des noms de groupe et des GID est définie dans les bases de données des informations sur les comptes de groupe. Par défaut, les systèmes utilisent le fichier **/etc/group** pour stocker les informations concernant les groupes locaux.

Chaque ligne du fichier **/etc/group** contient des informations sur un groupe. Chaque entrée de groupe est divisée en quatre champs séparés par deux-points. Voici un exemple de ligne de **/etc/group**:

```
❶ group01:❷x:❸10000:❹user01,user02,user03
```

- ❶ Nom de ce groupe (group01).
- ❷ Champ de mot de passe de groupe obsolète. Ce champ doit toujours être x .
- ❸ Le numéro GID de ce groupe (10000).
- ❹ Une liste des utilisateurs membres de ce groupe en tant que groupe supplémentaire (user01, user02, user03). Les groupes principaux (ou par défaut) et supplémentaires sont abordés plus loin dans cette section.

Groupes primaires et groupes supplémentaires

Chaque utilisateur a exactement un groupe principal. Pour les utilisateurs locaux, il s'agit du groupe répertorié par numéro GID dans le fichier **/etc/passwd**. Par défaut, il s'agit du groupe qui possédera les nouveaux fichiers créés par l'utilisateur.

Normalement, lorsque vous créez un nouvel utilisateur normal, un nouveau groupe portant le même nom que cet utilisateur est créé. Ce groupe est utilisé comme groupe principal pour le nouvel utilisateur, et cet utilisateur est le seul membre de *Groupe privé d'utilisateurs*. Il s'avère que cela facilite la gestion des autorisations de fichiers, ce que nous verrons plus loin dans ce cours.

Les utilisateurs peuvent également avoir des *groupes supplémentaires*. La composition des groupes supplémentaires est déterminée par le fichier **/etc/group**. Les utilisateurs ont accès aux fichiers en fonction de l'accès de l'un de leurs groupes. Peu importe si le groupe ou les groupes ayant accès sont des groupes principaux ou supplémentaires de l'utilisateur.

Par exemple, si l'utilisateur user01 a un groupe principal user01 et des groupes supplémentaires wheel et webadmin, cet utilisateur pourra alors lire les fichiers lisibles par l'un de ces trois groupes.

La commande **id** peut également être utilisée pour connaître l'appartenance d'un utilisateur à un groupe.

```
[user03@host ~]$ id  
uid=1003(user03) gid=1003(user03) groups=1003(user03),10(wheel),10000(group01)  
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Dans l'exemple précédent, `user03` a le groupe `user03` comme groupe principal (**gid**). L'élément **groupes** liste tous les groupes de cet utilisateur (autres que le groupe principal `user03`). Les groupes `wheel` et `group01` sont des groupes supplémentaires de l'utilisateur.



RÉFÉRENCES

Pages de manuel **id(1)**, **passwd(5)** et **group(5)**

info libc (*Manuel de référence de la bibliothèque C GNU*)

- Section 30 : Utilisateurs et groupes

(Notez que le paquetage `glibc-devel` doit être installé pour que ce nœud info soit disponible.)

► QUIZ

DESCRIPTION DES CONCEPTS D'UTILISATEUR ET DE GROUPE

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. Quel élément désigne le nombre qui identifie l'utilisateur au niveau le plus fondamental ?
 - a. utilisateur principal
 - b. UID
 - c. GID
 - d. nom d'utilisateur

- ▶ 2. Quel élément désigne le programme qui fournit l'invite de ligne de commande de l'utilisateur ?
 - a. shell principal
 - b. répertoire personnel
 - c. shell de connexion
 - d. nom de commande

- ▶ 3. Quel élément ou fichier désigne l'emplacement des informations de groupe local ?
 - a. répertoire personnel
 - b. **/etc/passwd**
 - c. **/etc/GID**
 - d. **/etc/group**

- ▶ 4. Quel élément ou fichier désigne l'emplacement des fichiers personnels de l'utilisateur ?
 - a. répertoire personnel
 - b. shell de connexion
 - c. **/etc/passwd**
 - d. **/etc/group**

- ▶ 5. Quel élément désigne le nombre qui identifie le groupe au niveau le plus fondamental ?
 - a. groupe principal
 - b. UID
 - c. GID
 - d. groupid

► **6. Quel élément ou fichier désigne l'emplacement des informations de compte d'utilisateur local ?**

- a. répertoire personnel
- b. **/etc/passwd**
- c. **/etc/UID**
- d. **/etc/group**

► **7. Quel est le quatrième champ du fichier /etc/passwd ?**

- a. répertoire personnel
- b. UID
- c. shell de connexion
- d. groupe principal

► SOLUTION

DESCRIPTION DES CONCEPTS D'UTILISATEUR ET DE GROUPE

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. Quel élément désigne le nombre qui identifie l'utilisateur au niveau le plus fondamental ?
 - a. utilisateur principal
 - b. UID
 - c. GID
 - d. nom d'utilisateur

- ▶ 2. Quel élément désigne le programme qui fournit l'invite de ligne de commande de l'utilisateur ?
 - a. shell principal
 - b. répertoire personnel
 - c. shell de connexion
 - d. nom de commande

- ▶ 3. Quel élément ou fichier désigne l'emplacement des informations de groupe local ?
 - a. répertoire personnel
 - b. /etc/passwd
 - c. /etc/GID
 - d. /etc/group

- ▶ 4. Quel élément ou fichier désigne l'emplacement des fichiers personnels de l'utilisateur ?
 - a. répertoire personnel
 - b. shell de connexion
 - c. /etc/passwd
 - d. /etc/group

- ▶ 5. Quel élément désigne le nombre qui identifie le groupe au niveau le plus fondamental ?
 - a. groupe principal
 - b. UID
 - c. GID
 - d. groupid

► **6. Quel élément ou fichier désigne l'emplacement des informations de compte d'utilisateur local ?**

- a. répertoire personnel
- b. **/etc/passwd**
- c. **/etc/UID**
- d. **/etc/group**

► **7. Quel est le quatrième champ du fichier /etc/passwd ?**

- a. répertoire personnel
- b. UID
- c. shell de connexion
- d. groupe principal

ACCÈS EN TANT QUE SUPER UTILISATEUR

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir vous connecter en tant que super utilisateur pour gérer un système Linux et accorder à d'autres utilisateurs un accès super utilisateur à l'aide de la commande **sudo**.

LE SUPER UTILISATEUR

La plupart des systèmes d'exploitation ont une sorte de *super utilisateur* qui a tout pouvoir sur le système. Dans Red Hat Enterprise Linux, il s'agit de l'utilisateur **root**. Cet utilisateur a le pouvoir d'outrepasser les priviléges normaux sur le système de fichiers. Il peut gérer et administrer le système. Pour effectuer des tâches comme l'installation ou la suppression de logiciels, et pour gérer des fichiers et des répertoires du système, l'utilisateur doit augmenter ses priviléges au niveau de ceux de l'utilisateur **root**.

L'utilisateur **root** est le seul, par rapport aux utilisateurs normaux, à pouvoir contrôler la plupart des périphériques, à quelques exceptions près. Par exemple, les utilisateurs normaux peuvent contrôler des périphériques amovibles, tels que des périphériques USB. Ainsi, les utilisateurs normaux sont autorisés à ajouter et supprimer des fichiers et à gérer un périphérique amovible, mais par défaut, seul l'utilisateur **root** peut gérer les disques durs « fixes ».

Ce privilège illimité est toutefois assorti de responsabilités. L'utilisateur **root** a une capacité infinie à endommager le système : suppression de fichiers et de répertoires, suppression de comptes d'utilisateur, ajout de portes dérobées, etc. Si la sécurité du compte de l'utilisateur **root** était compromise, quelqu'un d'autre aurait le contrôle de l'administration du système. Tout au long de la présente formation, nous encourageons les administrateurs à se connecter en tant qu'utilisateur normal et à n'augmenter leurs priviléges au niveau de **root** que lorsque cela est nécessaire.

Le compte **root** sous Linux équivaut à peu près au compte local « Administrateur » sous Microsoft Windows. Dans Linux, la plupart des administrateurs système se connectent à l'aide d'un compte d'utilisateur sans privilège et utilisent divers outils pour disposer temporairement des priviléges **root**.



MISE EN GARDE

Sous Microsoft Windows, il était courant par le passé que l'utilisateur **Administrateur** local se connecte directement pour exécuter les tâches d'administration du système. Bien que cela soit possible sous Linux, Red Hat recommande aux administrateurs système de ne pas se connecter directement en tant que **root**. Les administrateurs système doivent plutôt se connecter en tant qu'utilisateur normal et utiliser d'autres mécanismes (**su**, **sudo** ou PolicyKit, par exemple) pour obtenir temporairement des priviléges de super utilisateur.

Lorsque vous vous connectez en tant que super utilisateur, tout l'environnement du bureau s'exécute inutilement avec des priviléges administratifs. Dans cette situation, toute faille de sécurité qui ne compromettrait en temps normal que le compte de l'utilisateur est alors susceptible de compromettre tout le système.

CHANGEMENT DE COMPTE D'UTILISATEUR

La commande **su** permet à un utilisateur de basculer vers le compte d'un autre utilisateur. Si vous exéutez **su** à partir d'un compte d'utilisateur normal, vous serez invité à saisir le mot de passe du compte sur lequel vous souhaitez passer. Quand root exécute **su**, vous n'avez pas besoin d'entrer le mot de passe de l'utilisateur.

```
[user01@host ~]$ su - user02
Password:
[user02@host ~]$
```

Si vous omettez le nom d'utilisateur, la commande **su** ou **su -** tente de basculer vers root par défaut.

```
[user01@host ~]$ su -
Password:
[root@host ~]#
```

La commande **su** démarre un *shell sans connexion*, alors que la commande **su -** (avec l'option de tiret) lance un *shell de connexion*. La principale différence qui existe entre ces deux commandes est que **su -** configure l'environnement shell comme s'il s'agissait d'une nouvelle connexion sous cette identité, alors que **su** lance simplement un shell sous l'identité de cet utilisateur, mais en conservant les paramètres d'environnement courants.

Dans la plupart des cas, les administrateurs doivent exécuter **su -** pour obtenir les paramètres d'environnement normaux de l'utilisateur. Pour plus d'informations, consultez la page de manuel **bash(1)**.



NOTE

La commande **su** est le plus souvent utilisée pour obtenir une interface de ligne de commande (une invite shell) exécutée sous l'identité d'un autre utilisateur, généralement root. Cependant, avec l'option **-c**, on peut l'utiliser comme l'utilitaire **runas** de Windows, pour lancer n'importe quel programme sous une autre identité. Exédez **info su** pour afficher davantage de détails.

EXÉCUTION DE COMMANDES AVEC SUDO

Dans certains cas, le compte d'utilisateur root peut ne pas avoir de mot de passe valide pour des raisons de sécurité. Dans ce cas, les utilisateurs ne peuvent pas se connecter au système en tant que root directement avec un mot de passe, et **su** ne peut pas être utilisé pour obtenir un shell interactif. Dans ce cas, **sudo** peut être utilisé pour accéder à root.

À la différence de **su**, **sudo** nécessite généralement que les utilisateurs entrent leur propre mot de passe pour l'authentification, et non le mot de passe du compte d'utilisateur auquel ils tentent d'accéder. Autrement dit, les utilisateurs qui utilisent **sudo** pour exécuter des commandes en tant que root n'ont pas besoin de connaître le mot de passe root. Au lieu de cela, ils utilisent leurs propres mots de passe pour authentifier l'accès.

En outre, **sudo** peut être configuré pour autoriser des utilisateurs spécifiques à exécuter une commande sous une autre identité, ou uniquement certaines commandes sous l'identité de cet utilisateur.

CHAPITRE 6 | Gestion des utilisateurs et des groupes locaux

Par exemple, lorsque **sudo** est configuré pour autoriser l'utilisateur **user01** à exécuter la commande **usermod** en tant que **root**, **user01** peut exécuter la commande suivante pour verrouiller ou déverrouiller le compte d'un utilisateur :

```
[user01@host ~]$ sudo usermod -L user02
[sudo] password for user01:
[user01@host ~]$ su - user02
Password:
su: Authentication failure
[user01@host ~]$
```

Si un utilisateur essaie d'exécuter une commande sous une autre identité et que la configuration **sudo** ne l'autorise pas, la commande sera bloquée, la tentative sera consignée dans le journal et, par défaut, un email sera envoyé à l'utilisateur **root**.

```
[user02@host ~]$ sudo tail /var/log/secure
[sudo] password for user02:
user02 is not in the sudoers file. This incident will be reported.
[user02@host ~]$
```

Un autre avantage de **sudo** est que toutes les commandes exécutées sont journalisées par défaut dans **/var/log/secure**.

```
[user01@host ~]$ sudo tail /var/log/secure
...output omitted...
Feb 6 20:45:46 host sudo[2577]: user01 : TTY=pts/0 ; PWD=/home/user01 ;
USER=root ; COMMAND=/sbin/usermod -L user02
...output omitted...
```

Dans Red Hat Enterprise Linux 7 et Red Hat Enterprise Linux 8, tous les membres du groupe **wheel** peuvent utiliser **sudo** pour exécuter des commandes sous l'identité de n'importe quel utilisateur, y compris **root**. L'utilisateur sera invité à saisir son propre mot de passe. Ceci est un changement par rapport à Red Hat Enterprise Linux 6 et aux versions ultérieures : les utilisateurs qui étaient membres du groupe **wheel** ne bénéficiaient pas de cet accès administratif par défaut.



MISE EN GARDE

RHEL 6 n'accordait au groupe **wheel** aucun privilège particulier par défaut. Les sites qui utilisent ce groupe à des fins non standards pourraient être surpris de voir RHEL 7 et RHEL 8 accorder automatiquement à tous les membres de **wheel** les priviléges **sudo** complets. Il pourrait en résulter que des utilisateurs non autorisés obtiennent un accès administratif aux systèmes RHEL 7 and RHEL 8.

Historiquement, les systèmes de type UNIX utilisent l'appartenance au groupe **wheel** pour accorder ou contrôler l'accès de super utilisateur.

Obtenir un shell root interactif avec Sudo

Si un compte d'utilisateur non administrateur sur le système peut utiliser **sudo** pour exécuter la commande **su**, vous pouvez exécuter **sudo su** - à partir de ce compte pour obtenir un shell d'utilisateur **root** interactif. Cela fonctionne parce que **sudo** exécutera **su** - en tant que **root** et **root** n'a pas besoin d'entrer un mot de passe pour utiliser **su**.

Une autre façon d'accéder au compte **root** avec **sudo** est d'utiliser la commande **sudo -i**. Cela permet de basculer vers le compte **root** et d'exécuter le shell par défaut de cet utilisateur (généralement **bash**) et les scripts de connexion shell associés. Si vous voulez juste exécuter le shell, vous pouvez utiliser la commande **sudo -s**.

Par exemple, un administrateur peut obtenir un shell interactif en tant que **root** sur une instance AWS EC2 à l'aide de l'authentification SSH par clé publique pour se connecter en tant qu'utilisateur normal **ec2-user**, puis en exécutant **sudo -i** pour obtenir le shell de l'utilisateur **root**.

```
[ec2-user@host ~]$ sudo -i  
[sudo] password for ec2-user:  
[root@host ~]#
```

La commande **sudo su -** et **sudo -i** ne se comportent pas exactement de la même façon. Ce point sera brièvement abordé à la fin de la section.

Configuration de Sudo

Le fichier de configuration principal pour **sudo** et **/etc/sudoers**. Pour éviter les problèmes si plusieurs administrateurs tentent de le modifier en même temps, vous devez le modifier uniquement avec la commande **visudo** spéciale.

Par exemple, la ligne suivante du fichier **/etc/sudoers** active l'accès **sudo** pour les membres du groupe **wheel**.

```
%wheel      ALL=(ALL)      ALL
```

Dans cette ligne, **%wheel** est l'utilisateur ou le groupe auquel la règle s'applique. Un **%** spécifie qu'il s'agit d'un groupe, le groupe **wheel**. L'indication **ALL=(ALL)** spécifie que sur tout hôte pouvant contenir ce fichier, **wheel** peut exécuter n'importe quelle commande. Le **ALL** final indique que **wheel** peut exécuter ces commandes en tant que n'importe quel utilisateur du système.

Par défaut, **/etc/sudoers** comprend également le contenu de tous les fichiers du répertoire **/etc/sudoers.d** dans le fichier de configuration. Cela permet à un administrateur d'ajouter l'accès **sudo** pour un utilisateur simplement en mettant un fichier approprié dans ce répertoire.

NOTE

L'utilisation de fichiers supplémentaires sous le répertoire **/etc/sudoers.d** est pratique et simple. Vous pouvez activer ou désactiver l'accès **sudo** simplement en copiant un fichier dans le répertoire ou en le supprimant.

Dans ce cours, vous créerez et supprimerez des fichiers dans le répertoire **/etc/sudoers.d** pour configurer l'accès **sudo** pour les utilisateurs et les groupes.

Pour activer l'accès **sudo** complet pour l'utilisateur **user01**, vous pouvez créer **/etc/sudoers.d/user01** avec le contenu suivant :

```
user01  ALL=(ALL)  ALL
```

Pour activer l'accès **sudo** complet pour le groupe **group01**, vous pouvez créer **/etc/sudoers.d/group01** avec le contenu suivant :

```
%group01 ALL=(ALL) ALL
```

Il est également possible de configurer **sudo** pour permettre à un utilisateur d'exécuter des commandes sous une autre identité sans entrer son mot de passe :

```
ansible ALL=(ALL) NOPASSWD:ALL
```

Bien que l'octroi de ce niveau d'accès à un utilisateur ou un groupe présente des risques évidents pour la sécurité, il est fréquemment utilisé avec des instances de cloud, des systèmes de provisionnement et des machines virtuelles pour permettre la configuration des serveurs. Le compte avec cet accès doit être soigneusement protégé et peut nécessiter une authentification SSH par clé publique afin qu'un utilisateur sur un système distant puisse y accéder.

Par exemple, l'AMI officielle de Red Hat Enterprise Linux dans Amazon Web Services Marketplace est fournie avec les mots de passe verrouillés des utilisateurs `root` et `ec2-user`. Le compte d'utilisateur `ec2-user` est configuré pour permettre un accès interactif distant via l'authentification SSH par clé publique. L'utilisateur `ec2-user` peut également exécuter une commande en tant que `root` sans mot de passe car la dernière ligne du fichier `/etc/sudoers` de l'AMI est configurée comme suit :

```
ec2-user ALL=(ALL) NOPASSWD: ALL
```

L'obligation de saisir un mot de passe pour **sudo** peut être réactivée ou d'autres modifications peuvent être apportées pour renforcer la sécurité dans le cadre du processus de configuration du système.



NOTE

Dans ce cours, vous verrez fréquemment **sudo su -** utilisé au lieu de **sudo -i**. Les deux commandes fonctionnent, mais il existe des différences subtiles entre elles.

La commande **sudo su -** définit l'environnement root exactement comme une connexion normale parce que la commande **su -** ignore les paramètres définis par **sudo** et configure l'environnement à partir de zéro.

La configuration par défaut de la commande **sudo -i** définit certains détails de l'environnement de l'utilisateur root différemment d'une connexion normale. Par exemple, il définit la variable d'environnement PATH légèrement différemment. Cela a un impact sur les emplacements où le shell cherchera des commandes.

Vous pouvez faire en sorte que **sudo -i** se comporte davantage comme **su -** en modifiant **/etc/sudoers** avec **visudo**. Recherchez la ligne

```
Defaults    secure_path = /sbin:/bin:/usr/sbin:/usr/bin
```

et remplacez-la par les deux lignes suivantes :

```
Defaults    secure_path = /usr/local/bin:/usr/bin  
Defaults>root  secure_path = /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin
```

Dans la plupart des cas, cela ne constitue pas une différence majeure. Cependant, pour la cohérence des paramètres PATH sur les systèmes avec le fichier **/etc/sudoers** par défaut, les auteurs de ce cours utilisent principalement **sudo su -** dans les exemples.



RÉFÉRENCES

Pages de manuel **su(8)**, **sudo(8)**, **visudo(5)** et **sudoers(5)**

info libc persona (*Manuel de référence de la bibliothèque C GNU*)

- Section 30.2 : Identité d'un processus

(Notez que le paquetage *glibc-devel* doit être installé pour que ce nœud info soit disponible.)

► EXERCICE GUIDÉ

ACCÈS EN TANT QUE SUPER UTILISATEUR

Dans cet exercice, vous vous exercerez à basculer vers le compte `root` et à exécuter des commandes en tant que `root`.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Utiliser `sudo` pour basculer vers `root` et accéder au shell interactif en tant que `root` sans connaître le mot de passe du super utilisateur.
- Expliquer comment `su` et `sudo` - peuvent affecter l'environnement shell en exécutant ou non les scripts de connexion.
- Utiliser `sudo` pour exécuter d'autres commandes en tant que `root`.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab users-sudo start` pour démarrer l'exercice. Ce script crée les comptes d'utilisateur nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab users-sudo start
```

- 1. À partir de `workstation`, ouvrez une session SSH sur `servera` en tant que `student`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Explorez l'environnement shell de `student`. Affichez les informations relatives à l'utilisateur et au groupe, puis le répertoire de travail actuel. Affichez également les variables d'environnement qui spécifient le répertoire personnel de l'utilisateur et les emplacements des exécutables de l'utilisateur.

2.1. Exécutez `id` pour afficher les informations sur l'utilisateur et le groupe actuels.

```
[student@servera ~]$ id
uid=1000(student) gid=1000(student) groups=1000(student),10(wheel)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

2.2. Exécutez `pwd` pour afficher le répertoire de travail courant.

```
[student@servera ~]$ pwd
/home/student
```

- 2.3. Imprimez les valeurs des variables HOME et PATH pour déterminer le chemin du répertoire personnel et des exécutables de l'utilisateur, respectivement.

```
[student@servera ~]$ echo $HOME
/home/student
[student@servera ~]$ echo $PATH
/home/student/.local/bin:/home/student/bin:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin
```

- 3. Basculez vers root dans un shell sans connexion et explorez le nouvel environnement shell.

- 3.1. Exécutez sudo su à l'invite du shell pour devenir l'utilisateur root.

```
[student@servera ~]$ sudo su
[sudo] password for student: student
[root@servera student]#
```

- 3.2. Exécutez id pour afficher les informations sur l'utilisateur et le groupe actuels.

```
[root@servera student]# id
uid=0(root) gid=0(root) groups=0(root)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

- 3.3. Exécutez pwd pour afficher le répertoire de travail courant.

```
[root@servera student]# pwd
/home/student
```

- 3.4. Imprimez les valeurs des variables HOME et PATH pour déterminer le chemin du répertoire personnel et des exécutables de l'utilisateur, respectivement.

```
[root@servera student]# echo $HOME
/root
[root@servera student]# echo $PATH
/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/sbin:/usr/local/bin
```

Si vous avez déjà une expérience de Linux et de la commande **su**, vous vous attendiez peut-être à ce que l'utilisation de **su** sans l'option (-) de tiret pour basculer vers l'utilisateur **root** vous permettrait de conserver le PATH actuel de **student**. Ce qui n'a pas été le cas ! Comme vous le verrez à la prochaine étape, il ne s'agit pas non plus du PATH courant de **root**.

Que s'est-il passé ? La différence est que vous n'avez pas exécuté **su** directement. Au lieu de cela, vous avez exécuté **su** en tant que **root** en utilisant **sudo**, car vous ne possédez pas le mot de passe du super utilisateur. La commande **sudo** annule initialement la variable PATH de l'environnement initial pour des raisons de sécurité. Toute commande exécutée après le remplacement initial peut toujours mettre à jour la variable PATH, comme vous le verrez dans les étapes suivantes.

3.5. Quittez le shell de l'utilisateur `root` pour revenir au shell de l'utilisateur `student`.

```
[root@servera student]# exit
exit
[student@servera ~]$
```

► 4. Basculez vers `root` dans un shell de connexion et explorez le nouvel environnement shell.

4.1. Exécutez `sudo su -` à l'invite du shell pour devenir l'utilisateur `root`.

```
[student@servera ~]$ sudo su -
[root@servera ~]#
```

Remarquez la différence entre l'invite du shell et celle de `sudo su` à l'étape précédente.

`sudo` peut ou non vous demander le mot de passe `student`, en fonction de la période d'expiration de `sudo`. La période d'expiration par défaut est de cinq minutes. Si vous vous êtes authentifié auprès de `sudo` dans les cinq dernières minutes, `sudo` ne vous demandera pas le mot de passe. Si cela fait plus de cinq minutes que vous vous êtes authentifié auprès de `sudo`, vous devez entrer `student` comme mot de passe pour être authentifié auprès de `sudo`.

4.2. Exécutez `id` pour afficher les informations sur l'utilisateur et le groupe actuels.

```
[root@servera ~]# id
uid=0(root) gid=0(root) groups=0(root)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

4.3. Exécutez `pwd` pour afficher le répertoire de travail courant.

```
[root@servera ~]# pwd
/root
```

4.4. Imprimez les valeurs des variables `HOME` et `PATH` pour déterminer le chemin du répertoire personnel et des exécutables de l'utilisateur, respectivement.

```
[root@servera ~]# echo $HOME
/root
[root@servera ~]# echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin
```

Comme à l'étape précédente, après que `sudo` a réinitialisé la variable `PATH` des paramètres dans l'environnement shell de l'utilisateur `student`, la commande `su -` a exécuté les scripts de connexion shell pour `root` et a défini une autre valeur pour la variable `PATH`. La commande `su` sans l'option (-) de tiret n'a pas effectué cela.

4.5. Quittez le shell de l'utilisateur `root` pour revenir au shell de l'utilisateur `student`.

```
[root@servera ~]# exit
logout
[student@servera ~]$
```

- 5. Vérifiez que l'utilisateur **operator1** est configuré pour exécuter n'importe quelle commande sous n'importe quelle identité à l'aide de **sudo**.

```
[student@servera ~]$ sudo cat /etc/sudoers.d/operator1  
operator1 ALL=(ALL) ALL
```

- 6. Basculez vers **operator1** et affichez le contenu de **/var/log/messages**. Copiez **/etc/motd** vers **/etc/motdOLD** et supprimez-le (**/etc/motdOLD**). Ces opérations nécessitent des droits d'administration et utilisent donc **sudo** pour exécuter ces commandes en tant que super utilisateur. Ne basculez pas vers root en utilisant **sudo su** ou **sudo su -**. Utilisez **redhat** comme mot de passe pour **operator1**.

6.1. Basculez vers **operator1**.

```
[student@servera ~]$ su - operator1  
Password: redhat  
[operator1@servera ~]$
```

6.2. Essayez d'afficher les cinq dernières lignes de **/var/log/messages** sans utiliser **sudo**. Cette opération doit échouer.

```
[operator1@servera ~]$ tail -5 /var/log/messages  
tail: cannot open '/var/log/messages' for reading: Permission denied
```

6.3. Essayez d'afficher les cinq dernières lignes de **/var/log/messages** avec **sudo**. Cette opération devrait réussir.

```
[operator1@servera ~]$ sudo tail -5 /var/log/messages  
[sudo] password for operator1: redhat  
Jan 23 15:53:36 servera su[2304]: FAILED SU (to operator1) student on pts/1  
Jan 23 15:53:51 servera su[2307]: FAILED SU (to operator1) student on pts/1  
Jan 23 15:53:58 servera su[2310]: FAILED SU (to operator1) student on pts/1  
Jan 23 15:54:12 servera su[2322]: (to operator1) student on pts/1  
Jan 23 15:54:25 servera su[2353]: (to operator1) student on pts/1
```



NOTE

La sortie précédente peut différer sur votre système.

6.4. Essayez de faire une copie de **/etc/motd** comme **/etc/motdOLD** sans utiliser **sudo**. Cette opération doit échouer.

```
[operator1@servera ~]$ cp /etc/motd /etc/motdOLD  
cp: cannot create regular file '/etc/motdOLD': Permission denied
```

6.5. Essayez de faire une copie de **/etc/motd** comme **/etc/motdOLD** avec **sudo**. Cette opération devrait réussir.

```
[operator1@servera ~]$ sudo cp /etc/motd /etc/motdOLD  
[operator1@servera ~]$
```

6.6. Essayez de supprimer **/etc/motdOLD** sans utiliser **sudo**. Cette opération doit échouer.

```
[operator1@servera ~]$ rm /etc/motdOLD  
rm: remove write-protected regular empty file '/etc/motdOLD'? y  
rm: cannot remove '/etc/motdOLD': Permission denied  
[operator1@servera ~]$
```

6.7. Essayez de supprimer **/etc/motdOLD** avec **sudo**. Cette opération devrait réussir.

```
[operator1@servera ~]$ sudo rm /etc/motdOLD  
[operator1@servera ~]$
```

6.8. Quittez le shell de l'utilisateur **operator1** pour revenir au shell de l'utilisateur **student**.

```
[operator1@servera ~]$ exit  
logout  
[student@servera ~]$
```

6.9. Déconnectez-vous de **servera**.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Fin

Sur **workstation**, exéutez **lab users-sudo finish** pour mettre fin à l'exercice. Ce script supprime les comptes d'utilisateur et les fichiers créés au début de l'exercice et permet de nettoyer l'environnement.

```
[student@workstation ~]$ lab users-sudo finish
```

L'exercice guidé est maintenant terminé.

GESTION DES COMPTES D'UTILISATEUR LOCAUX

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir créer, modifier et supprimer les comptes de groupes locaux.

GESTION DES UTILISATEURS LOCAUX

Plusieurs outils de ligne de commande peuvent être utilisés pour gérer des comptes d'utilisateur locaux.

Création d'utilisateurs à partir de la ligne de commande

- La commande **useradd username** crée un nouvel utilisateur appelé `username`. Elle configure le répertoire personnel et les informations de compte de l'utilisateur, et crée un groupe privé pour l'utilisateur appelé `username`. À ce stade, le compte n'a pas de mot de passe valide défini et l'utilisateur ne peut se connecter avant qu'un mot de passe n'ait été défini.
- La commande **useradd --help** affiche les options de base qui peuvent être utilisées pour remplacer les valeurs par défaut. Dans la plupart des cas, on peut utiliser les mêmes options avec la commande **usermod** pour modifier un utilisateur existant.
- Certaines valeurs par défaut, comme la plage des numéros UID valides et les règles de vieillissement des mots de passe, sont lues dans le fichier **/etc/login.defs**. Les valeurs de ce fichier ne sont utilisées que lorsqu'on crée des utilisateurs. Une modification apportée à ce fichier n'affecte pas les utilisateurs existants.

Modification d'utilisateurs existants à partir de la ligne de commande

- La commande **usermod --help** affiche les options de base utilisables pour modifier un compte. On trouve parmi les options courantes :

OPTIONS USERMOD :	UTILISATION
-c, --comment COMMENT	Ajouter le vrai nom de l'utilisateur au champ de commentaire.
-g, --gid GROUP	Spécifier le groupe principal du compte d'utilisateur.
-G, --groups GROUPS	Spécifier une liste de groupes supplémentaires séparés par des virgules pour le compte d'utilisateur.
-a, --append	Utilisé avec l'option -G pour ajouter les groupes supplémentaires à l'ensemble de groupes auxquels l'utilisateur appartient au lieu de remplacer l'ensemble de groupes supplémentaires par un nouvel ensemble.
-d, --home HOME_DIR	Spécifier un répertoire personnel spécifique pour le compte d'utilisateur.

OPTIONS USERMOD :	UTILISATION
-m, --move-home	Déplacer le répertoire personnel de l'utilisateur vers un nouvel emplacement. Doit être utilisé avec l'option -d .
-s, --shell SHELL	Spécifier un shell de connexion spécifique pour le compte d'utilisateur.
-L, --lock	Verrouiller le compte.
-U, --unlock	Déverrouiller le compte.

Suppression d'utilisateurs à partir de la ligne de commande

- La commande **userdel username** supprime les détails de l'utilisateur `username` de **/etc/passwd**, mais laisse le répertoire personnel de l'utilisateur intact.
- La commande **userdel -r username** supprime les détails de l'utilisateur `username` de **/etc/passwd** ainsi que le répertoire personnel de l'utilisateur.



MISE EN GARDE

Quand on supprime un utilisateur avec **userdel** sans spécifier l'option **-r**, le système se retrouve avec des fichiers qui appartiennent à un UID non affecté. Cela peut aussi arriver lorsqu'un fichier, créé et appartenant à un utilisateur supprimé, se retrouve en dehors du répertoire personnel de l'utilisateur. Cette situation peut entraîner des fuites d'informations et d'autres problèmes de sécurité.

Dans Red Hat Enterprise Linux 7 et Red Hat Enterprise Linux 8, la commande **useradd** affecte aux nouveaux utilisateurs le premier UID libre supérieur ou égal à 1000, sauf si vous spécifiez de manière explicite un UID à l'aide de l'option **-u**.

C'est ce qui explique pourquoi la fuite d'informations peut se produire. Si le premier UID libre a déjà été affecté à un compte d'utilisateur qui a depuis été supprimé du système, cet ancien UID est réaffecté au nouvel utilisateur et celui-ci se retrouve propriétaire des fichiers restants de l'ancien compte d'utilisateur.

Le scénario suivant fait la démonstration de cette situation.

```
[root@host ~]# useradd user01
[root@host ~]# ls -l /home
drwx----- 3 user01 user01 74 Feb 4 15:22 user01
[root@host ~]# userdel user01
[root@host ~]# ls -l /home
drwx----- 3 1000 1000 74 Feb 4 15:22 user01
[root@host ~]# useradd user02
[root@host ~]# ls -l /home
drwx----- 3 user02 user02 74 Feb 4 15:23 user02
drwx----- 3 user02 user02 74 Feb 4 15:22 user01
```

Vous pouvez remarquer que **user02** possède maintenant tous les fichiers dont **user01** était auparavant le propriétaire.

Suivant la situation, l'une des solutions à ce problème consiste à supprimer du système tous les fichiers sans propriétaire une fois que l'utilisateur qui les a créés a été supprimé. Une autre solution consiste à affecter manuellement les fichiers sans propriétaire à un autre utilisateur. L'utilisateur **root** peut utiliser la commande **find / -nouser -o -nogroup** pour trouver tous les fichiers et répertoires sans propriétaire.

Définition de mots de passe à partir de la ligne de commande

- La commande **passwd username** définit le mot de passe initial ou modifie le mot de passe existant de l'utilisateur **username**.
- L'utilisateur **root** peut attribuer n'importe quelle valeur à un mot de passe. Un message s'affiche si le mot de passe ne satisfait pas aux critères minimaux recommandés, mais il est suivi d'une invite pour retaper le nouveau mot de passe. Tous les tokens sont ensuite mis à jour correctement.

```
[root@host ~]# passwd user01
Changing password for user user01.
New password: redhat
BAD PASSWORD: The password fails the dictionary check - it is based on a
dictionary word
Retype new password: redhat
passwd: all authentication tokens updated successfully.
[root@host ~]#
```

- Un utilisateur standard doit choisir un mot de passe d'au moins huit caractères, qui ne peut être ni un mot du dictionnaire, ni le nom de l'utilisateur, ni le mot de passe précédent.

Plages d'UID

Red Hat Enterprise Linux utilise des numéros et plages de numéros UID à des fins spécifiques.

- L'*UID 0* est toujours affecté au compte du super utilisateur, **root**.
- *UID 1-200* est une plage « d'utilisateurs système » que Red Hat affecte de manière statique aux processus système.
- *UID 201-999* est une plage « d'utilisateurs système » utilisée par des processus système qui ne sont pas propriétaires de fichiers sur le système de fichiers. En général, ils sont affectés de manière dynamique à partir du pool disponible, lorsque le logiciel qui en a besoin est installé. Les programmes sont exécutés sous l'identité de ces utilisateurs système « sans privilège » afin de limiter leur accès aux seules ressources dont ils ont besoin pour fonctionner.
- *UID 1000+* est la plage disponible pour l'affectation aux utilisateurs standard.



NOTE

Avant RHEL 7, la convention était d'utiliser la plage UID 1-499 pour les utilisateurs système, et la plage 500+ pour les utilisateurs standard. Les plages par défaut utilisées par **useradd** et **groupadd** peuvent être modifiées dans le fichier **/etc/login.defs**.



RÉFÉRENCES

Pages du manuel **useradd(8)**, **usermod(8)**, **userdel(8)**

► EXERCICE GUIDÉ

GESTION DES COMPTES D'UTILISATEUR LOCAUX

Dans cet exercice, vous allez créer plusieurs utilisateurs sur votre système et définir des mots de passe pour ces derniers.

RÉSULTATS

Vous devez pouvoir configurer un système Linux avec des comptes d'utilisateur supplémentaires.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez **lab users-manage start** pour démarrer l'exercice. Ce script garantit que l'environnement est correctement configuré.

```
[student@workstation ~]$ lab users-manage start
```

- 1. À partir de **workstation**, ouvrez une session SSH sur **servera** en tant que **student**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Sur **servera**, basculez vers **root** en utilisant **sudo**, ce qui permet de convertir l'environnement shell de l'utilisateur **root**.

```
[student@servera ~]$ sudo su -
[sudo] password for student: student
[root@servera ~]#
```

- 3. Créez l'utilisateur **operator1** et confirmez qu'il existe dans le système.

```
[root@servera ~]# useradd operator1
[root@servera ~]# tail /etc/passwd
...output omitted...
operator1:x:1001:1001::/home/operator1:/bin/bash
```

- 4. Définissez le mot de passe de l'utilisateur **operator1** sur **redhat**.

CHAPITRE 6 | Gestion des utilisateurs et des groupes locaux

```
[root@servera ~]# passwd operator1
Changing password for user operator1.
New password: redhat
BAD PASSWORD: The password is shorter than 8 characters
Retype new password: redhat
passwd: all authentication tokens updated successfully.
```

- ▶ 5. Créez les utilisateurs supplémentaires operator2 et operator3. Définissez leur mot de passe sur **redhat**.

- 5.1. Ajoutez l'utilisateur operator2. Définissez le mot de passe de l'utilisateur operator2 sur **redhat**.

```
[root@servera ~]# useradd operator2
[root@servera ~]# passwd operator2
Changing password for user operator2.
New password: redhat
BAD PASSWORD: The password is shorter than 8 characters
Retype new password: redhat
passwd: all authentication tokens updated successfully.
```

- 5.2. Ajoutez l'utilisateur operator3. Définissez le mot de passe de l'utilisateur operator3 sur **redhat**.

```
[root@servera ~]# useradd operator3
[root@servera ~]# passwd operator3
Changing password for user operator3.
New password: redhat
BAD PASSWORD: The password is shorter than 8 characters
Retype new password: redhat
passwd: all authentication tokens updated successfully.
```

- ▶ 6. Mettez à jour les comptes d'utilisateur operator1 et operator2 afin d'y inclure les commentaires **Operator One** et **Operator Two**, respectivement. Vérifiez que les commentaires ont été correctement ajoutés.

- 6.1. Exécutez **usermod -c** pour mettre à jour les commentaires du compte d'utilisateur operator1.

```
[root@servera ~]# usermod -c "Operator One" operator1
```

- 6.2. Exécutez **usermod -c** pour mettre à jour les commentaires du compte d'utilisateur operator2.

```
[root@servera ~]# usermod -c "Operator Two" operator2
```

- 6.3. Confirmez que les commentaires pour chacun des utilisateurs operator1 et operator2 sont reflétés dans les enregistrements d'utilisateur.

```
[root@servera ~]# tail /etc/passwd
...output omitted...
operator1:x:1001:1001:Operator One:/home/operator1:/bin/bash
operator2:x:1002:1002:Operator Two:/home/operator2:/bin/bash
operator3:x:1003:1003::/home/operator3:/bin/bash
```

- ▶ 7. Supprimez l'utilisateur operator3 ainsi que toutes ses données personnelles. Confirmez que l'utilisateur a bien été supprimé.

7.1. Supprimez l'utilisateur operator3 du système.

```
[root@servera ~]# userdel -r operator3
```

7.2. Confirmez que l'utilisateur operator3 a bien été supprimé.

```
[root@servera ~]# tail /etc/passwd
...output omitted...
operator1:x:1001:1001:Operator One:/home/operator1:/bin/bash
operator2:x:1002:1002:Operator Two:/home/operator2:/bin/bash
```

Notez que la sortie précédente n'affiche pas les informations de compte d'utilisateur de l'utilisateur operator3.

7.3. Quittez le shell de l'utilisateur root pour revenir au shell de l'utilisateur student.

```
[root@servera ~]# exit
logout
[student@servera ~]$
```

7.4. Déconnectez-vous de servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Fin

Sur workstation, exécutez **lab users-manage finish** pour mettre fin à l'exercice. Ce script garantit le nettoyage de l'environnement.

```
[student@workstation ~]$ lab users-manage finish
```

L'exercice guidé est maintenant terminé.

GESTION DES COMPTES DE GROUPES LOCAUX

OBJECTIFS

Après avoir terminé cette section, les stagiaires doivent pouvoir créer, modifier et supprimer les comptes de groupes locaux.

GESTION DES GROUPES LOCAUX

Un groupe doit exister avant qu'on puisse y ajouter un utilisateur. Plusieurs outils de ligne de commande sont utilisés pour gérer les comptes de groupes locaux.

Création de groupes à partir de la ligne de commande

- Utilisez la commande **groupadd** pour créer des groupes. Sans option, la commande **groupadd** utilise le premier GID disponible dans la plage spécifiée dans le fichier **/etc/login.defs** lors de la création des groupes.
- L'option **-g** spécifie un GID particulier pour le groupe à utiliser.

```
[user01@host ~]$ sudo groupadd -g 10000 group01
[user01@host ~]$ tail /etc/group
...output omitted...
group01:x:10000:
```



NOTE

Les groupes privés d'utilisateurs étant créés automatiquement (GID 1000 et plus), il est généralement recommandé de mettre de côté une plage de GID destinés aux groupes supplémentaires. Une plage plus élevée permettra d'éviter les collisions avec un groupe système (GID 0-999).

- L'option **-r** sert à créer un groupe système avec un GID issu des GID valides du système, listés dans le fichier **/etc/login.defs**. Les éléments de configuration **SYS_GID_MIN** et **SYS_GID_MAX** dans **/etc/login.defs** définissent la plage de GID système.

```
[user01@host ~]$ sudo groupadd -r group02
[user01@host ~]$ tail /etc/group
...output omitted...
group01:x:10000:
group02:x:988:
```

Modification des groupes existants à partir de la ligne de commande

- La commande **groupmod** change les propriétés d'un groupe existant. L'option **-n** spécifie un nouveau nom pour le groupe.

```
[user01@host ~]$ sudo groupmod -n group0022 group02
[user01@host ~]$ tail /etc/group
...output omitted...
group0022:x:988:
```

Notez que le nom du groupe **group02** est mis à jour par **group0022**.

- L'option **-g** sert à spécifier un nouveau GID.

```
[user01@host ~]$ sudo groupmod -g 20000 group0022
[user01@host ~]$ tail /etc/group
...output omitted...
group0022:x:20000:
```

Notez que le GID **988** est mis à jour par **20000**.

Création de groupes à partir de la ligne de commande

- La commande **groupdel** supprime les groupes.

```
[user01@host ~]$ sudo groupdel group0022
```



NOTE

Vous ne pouvez pas supprimer un groupe s'il s'agit du groupe principal d'un utilisateur existant. Tout comme avec la commande **userdel**, vérifiez tous les systèmes de fichiers pour vous assurer qu'il ne reste aucun fichier appartenant au groupe sur le système.

Modification de l'appartenance à un groupe à partir de la ligne de commande

- Les membres d'un groupe sont contrôlés par la gestion des utilisateurs. Utilisez la commande **usermod -g** pour changer le groupe principal d'un utilisateur.

```
[user01@host ~]$ id user02
uid=1006(user02) gid=1008(user02) groups=1008(user02)
[user01@host ~]$ sudo usermod -g group01 user02
[user01@host ~]$ id user02
uid=1006(user02) gid=10000(group01) groups=10000(group01)
```

- Utilisez la commande **usermod -aG** pour ajouter un utilisateur au groupe supplémentaire.

```
[user01@host ~]$ id user03
uid=1007(user03) gid=1009(user03) groups=1009(user03)
[user01@host ~]$ sudo usermod -aG group01 user03
[user01@host ~]$ id user03
uid=1007(user03) gid=1009(user03) groups=1009(user03),10000(group01)
```



IMPORTANT

L'utilisation de l'option **-a** active le mode *append* pour la fonction **usermod**. Sans **-a**, l'utilisateur sera supprimé de l'un des groupes supplémentaires actuels qui ne sont pas inclus dans la liste des options **-G**.



RÉFÉRENCES

Pages de manuel **group(5)**, **groupadd(8)**, **groupdel(8)** et **usermod(8)**

► EXERCICE GUIDÉ

GESTION DES COMPTES DE GROUPES LOCAUX

Dans cet exercice, vous allez créer des groupes, les utiliser en tant que groupes supplémentaires pour certains utilisateurs sans changer leurs groupes principaux, puis configurer l'un des groupes avec un accès sudo pour exécuter des commandes en tant que root.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Créer des groupes et les utiliser comme groupes supplémentaires.
- Configurer un accès sudo complet pour un groupe.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez **lab users-group-manage start** pour mettre fin à l'exercice. Ce script crée les comptes d'utilisateur nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab users-group-manage start
```

- 1. À partir de **workstation**, ouvrez une session SSH sur **servera** en tant que **student**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Sur **servera**, basculez vers **root** en utilisant **sudo**, ce qui permet d'hériter de l'environnement complet de l'utilisateur **root**.

```
[student@servera ~]$ sudo su -
[sudo] password for student: student
[root@servera ~]#
```

- 3. Créez le groupe supplémentaire **operators** avec le GID « 30000 ».

```
[root@servera ~]# groupadd -g 30000 operators
```

- 4. Créez un autre groupe supplémentaire **admin**.

```
[root@servera ~]# groupadd admin
```

- 5. Vérifiez que les deux groupes supplémentaires operators et admin existent.

```
[root@servera ~]# tail /etc/group
...output omitted...
operators:x:30000:
admin:x:30001:
```

- 6. Assurez-vous que les utilisateurs operator1, operator2 et operator3 appartiennent au groupe operators.

6.1. Ajoutez operator1, operator2 et operator3 à operators.

```
[root@servera ~]# usermod -aG operators operator1
[root@servera ~]# usermod -aG operators operator2
[root@servera ~]# usermod -aG operators operator3
```

6.2. Confirmez que les utilisateurs ont bien été ajoutés au groupe.

```
[root@servera ~]# id operator1
uid=1001(operator1) gid=1001(operator1) groups=1001(operator1),30000(operators)
[root@servera ~]# id operator2
uid=1002(operator2) gid=1002(operator2) groups=1002(operator2),30000(operators)
[root@servera ~]# id operator3
uid=1003(operator3) gid=1003(operator3) groups=1003(operator3),30000(operators)
```

- 7. Assurez-vous que les utilisateurs sysadmin1, sysadmin2 et sysadmin3 appartiennent au groupe admin. Activez les droits d'administration pour tous les membres du groupe admin. Vérifiez que tout membre d'admin peut exécuter des commandes administratives.

7.1. Ajoutez sysadmin1, sysadmin2 et sysadmin3 à admin.

```
[root@servera ~]# usermod -aG admin sysadmin1
[root@servera ~]# usermod -aG admin sysadmin2
[root@servera ~]# usermod -aG admin sysadmin3
```

7.2. Confirmez que les utilisateurs ont bien été ajoutés au groupe.

```
[root@servera ~]# id sysadmin1
uid=1004(sysadmin1) gid=1004(sysadmin1) groups=1004(sysadmin1),30001(admin)
[root@servera ~]# id sysadmin2
uid=1005(sysadmin2) gid=1005(sysadmin2) groups=1005(sysadmin2),30001(admin)
[root@servera ~]# id sysadmin3
uid=1006(sysadmin3) gid=1006(sysadmin3) groups=1006(sysadmin3),30001(admin)
```

7.3. Examinez **/etc/group** pour vérifier les membres des groupes supplémentaires.

```
[root@servera ~]# tail /etc/group  
...output omitted...  
operators:x:30000:operator1,operator2,operator3  
admin:x:30001:sysadmin1,sysadmin2,sysadmin3
```

- 7.4. Créez le fichier **/etc/sudoers.d/admin** de telle sorte que les membres **admin** disposent de tous les priviléges administratifs.

```
[root@servera ~]# echo "%admin ALL=(ALL) ALL" >> /etc/sudoers.d/admin
```

- 7.5. Basculez vers **sysadmin1** (un membre d'**admin**) et vérifiez que vous pouvez exécuter une commande **sudo** en tant que **sysadmin1**.

```
[root@servera ~]# su - sysadmin1  
[sysadmin1@servera ~]$ sudo cat /etc/sudoers.d/admin  
[sudo] password for sysadmin1: redhat  
%admin ALL=(ALL) ALL
```

- 7.6. Quittez le shell de l'utilisateur **sysadmin1** pour revenir au shell **root** de l'utilisateur.

```
[sysadmin1@servera ~]$ exit  
logout  
[root@servera ~]#
```

- 7.7. Quittez le shell de l'utilisateur **root** pour revenir au shell de l'utilisateur **student**.

```
[root@servera ~]# exit  
logout  
[student@servera ~]$
```

- 7.8. Déconnectez-vous de **servera**.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Fin

Sur **workstation**, exécutez **lab users-group-manage finish** pour mettre fin à l'exercice. Ce script supprime les comptes d'utilisateur créés au début de l'exercice.

```
[student@workstation ~]$ lab users-group-manage finish
```

L'exercice guidé est maintenant terminé.

GESTION DES MOTS DE PASSE DES UTILISATEURS

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir définir une stratégie de gestion des mots de passe pour les utilisateurs, et verrouiller et déverrouiller manuellement les comptes d'utilisateur.

MOTS DE PASSE FANTÔMES ET POLITIQUE DES MOTS DE PASSE

À un moment donné, les mots de passe chiffrés étaient stockés dans le fichier **/etc/passwd**, lisibles par tout le monde. Cela était considéré comme raisonnablement sûr, jusqu'à ce que les attaques par dictionnaire sur les mots de passe chiffrés se banalisent. À ce stade, les mots de passe chiffrés ont été déplacés vers un fichier **/etc/shadow** séparé, lisible uniquement par **root**. Ce nouveau fichier permettait également de mettre en œuvre des fonctionnalités de vieillissement et d'expiration des mots de passe.

Comme **/etc/passwd**, chaque utilisateur a une ligne dans le fichier **/etc/shadow**. Un exemple de ligne de **/etc/shadow** avec ses neuf champs séparés par des deux-points figure ci-dessous.

```
❶ user03:❷$6$CSSX...output omitted...:❸17933:❹0:❺99999:❻7:❼2:❽18113:❾
```

- ❶ Nom d'utilisateur du compte auquel ce mot de passe appartient.
- ❷ Le *mot de passe chiffré* de l'utilisateur. Le format des mots de passe chiffrés est abordé plus loin dans cette section.
- ❸ Le jour où le mot de passe a été modifié pour la dernière fois. Il est défini en jours depuis 1970-01-01, et est calculé dans le fuseau horaire UTC.
- ❹ Nombre minimal de jours devant s'écouler depuis le dernier changement de mot de passe avant que l'utilisateur ne puisse le modifier à nouveau.
- ❺ Nombre maximal de jours pouvant s'écouler sans modification du mot de passe avant son expiration. Un champ vide signifie qu'il n'expire pas en fonction du temps écoulé depuis le dernier changement.
- ❻ Période d'avertissement. L'utilisateur sera averti de l'expiration du mot de passe lorsqu'il se connectera pendant ce nombre de jours avant la date limite.
- ❼ Période d'inactivité. Une fois que le mot de passe a expiré, il sera toujours accepté pour la connexion pendant autant de jours. Une fois cette période écoulée, le compte sera verrouillé.
- ❽ Le jour où le mot de passe expire. Il est défini en jours depuis 1970-01-01, et est calculé dans le fuseau horaire UTC. Un champ vide signifie qu'il n'expire pas à une date donnée.
- ❾ Ce dernier champ est généralement vide et réservé à un usage ultérieur.

Format d'un mot de passe chiffré

Le champ de mot de passe chiffré stocke trois informations : l'*algorithme de hachage* utilisé, le *sel* et le *hachage chiffré*. Chaque information est délimitée par le signe **\$**.

```
$❶6$❷CSSXcYG1L/4ZfHr/$❸2W6evvJahUfzfHpc9X.45Jc6H30E...output omitted...
```

- ➊ L'algorithme de hachage utilisé pour ce mot de passe. Le chiffre **6** indique qu'il s'agit d'un hachage SHA-512, qui est la valeur par défaut dans Red Hat Enterprise Linux 8. Un **1** indiquerait un hachage MD5 et un **5** du SHA-256.
- ➋ Le sel utilisé pour chiffrer le mot de passe. À l'origine, celui-ci est choisi de manière aléatoire.
- ➌ Le hachage chiffré du mot de passe de l'utilisateur. Le sel et le mot de passe non chiffré sont combinés, puis chiffrés pour générer le hachage de mot de passe chiffré.

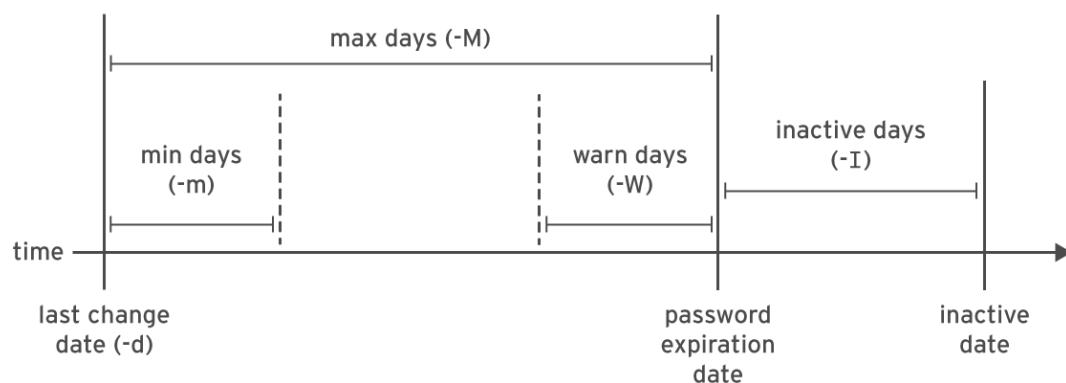
L'utilisation d'un sel empêche deux utilisateurs dotés du même mot de passe d'avoir des entrées identiques dans le fichier **/etc/shadow**. Par exemple, même si user01 et user02 utilisent **redhat** comme mot de passe, leurs mots de passe chiffrés dans **/etc/shadow** seront différents si leur sel est différent.

Vérification du mot de passe

Lorsqu'un utilisateur tente de se connecter, le système recherche l'entrée de cet utilisateur dans **/etc/shadow**, associe le sel de l'utilisateur au mot de passe non chiffré qui a été saisi, puis les chiffre à l'aide de l'algorithme de hachage spécifié. Si le résultat correspond au hachage chiffré, l'utilisateur a saisi le bon mot de passe. Si le résultat ne correspond pas au hachage chiffré, l'utilisateur a saisi le mauvais mot de passe et la tentative de connexion échoue. Cette méthode permet au système de déterminer si l'utilisateur a saisi le bon mot de passe, sans devoir stocker le mot de passe dans un format utilisable pour la connexion.

CONFIGURATION DU VIEILLISSEMENT DU MOT DE PASSE

Le diagramme suivant indique les paramètres relatifs au vieillissement d'un mot de passe qui peuvent être ajustés avec la commande **chage** pour mettre en œuvre une politique de vieillissement de mot de passe.



```
[user01@host ~]$ sudo chage -m 0 -M 90 -W 7 -I 14 user03
```

La commande **chage** précédente utilise les options **-m**, **-M**, **-W** et **-I** permettant de définir l'âge minimal, l'âge maximal, la période d'avertissement et la période d'inactivité du mot de passe de l'utilisateur, respectivement.

La commande **chage -d 0 user03** force l'utilisateur user03 à mettre à jour son mot de passe lors de la prochaine connexion.

La commande **chage -l user03** affiche les détails relatifs au vieillissement du mot de passe de l'utilisateur user03.

La commande **chage -E 2019-08-05 user03** entraîne l'expiration du compte d'utilisateur **user03** le 5 août 2019 (au format AAAA-MM-JJ).

**NOTE**

La commande **date** peut être utilisée pour calculer une date ultérieure. L'option **-u** indique l'heure en UTC.

```
[user01@host ~]$ date -d "+45 days" -u
Thu May 23 17:01:20 UTC 2019
```

Modifiez les éléments de configuration de vieillissement du mot de passe dans le fichier **/etc/login.defs** pour définir les politiques de vieillissement du mot de passe par défaut. La variable **PASS_MAX_DAYS** représente l'âge maximal par défaut du mot de passe. La variable **PASS_MIN_DAYS** représente l'âge minimal par défaut du mot de passe. La variable **PASS_WARN_AGE** représente la période d'avertissement par défaut du mot de passe. Toute modification des politiques de péremption du mot de passe par défaut ne sera effective que pour les nouveaux utilisateurs. Les utilisateurs existants continueront à utiliser les anciens paramètres de péremption du mot de passe plutôt que les nouveaux.

RESTRICTION D'ACCÈS

Vous pouvez utiliser la commande **chage** pour définir les dates d'expiration du compte. À cette date, l'utilisateur ne pourra plus se connecter au système de manière interactive. Utilisée avec l'option **-L**, la commande **usermod** peut verrouiller un compte.

```
[user01@host ~]$ sudo usermod -L user03
[user01@host ~]$ su - user03
Password: redhat
su: Authentication failure
```

Si un utilisateur quitte l'entreprise, l'administrateur peut verrouiller et faire expirer un compte avec une seule commande **usermod**. La date doit être spécifiée avec le nombre de jours depuis le 1er janvier 1970, ou au format AAAA-MM-JJ.

```
[user01@host ~]$ sudo usermod -L -e 2019-10-05 user03
```

La commande **usermod** précédente utilise l'option **-e** pour définir la date d'expiration du compte pour le compte d'utilisateur donné. L'option **-L** verrouille le mot de passe de l'utilisateur.

Le verrouillage du compte empêche l'utilisateur de s'authentifier sur le système à l'aide d'un mot de passe. Il s'agit de la méthode recommandée pour empêcher un employé qui a quitté l'entreprise d'accéder à son compte. En cas de retour de l'employé, le compte peut être déverrouillé avec **usermod -u**. Si le compte a aussi expiré, pensez à changer également la date d'expiration.

Shell nologin

Le shell **nologin** agit en tant que shell de substitution pour les comptes d'utilisateur qui ne sont pas prévus pour se connecter de manière interactive au système. Du point de vue de la sécurité, il est prudent d'empêcher le compte d'utilisateur de se connecter au système lorsque le compte d'utilisateur assume une responsabilité qui n'exige pas que l'utilisateur se connecte au système.

Par exemple, un serveur de messagerie peut exiger un compte pour le stockage du courrier et un mot de passe pour permettre à l'utilisateur de s'authentifier auprès d'un client de messagerie pour récupérer le courrier. Cet utilisateur n'a pas besoin de se connecter directement au système.

La solution courante à cette situation est de définir le shell de connexion de l'utilisateur sur / **sbin/nologin**. Si l'utilisateur tente de se connecter directement au système, le shell **nologin** ferme simplement la connexion.

```
[user01@host ~]$ usermod -s /sbin/nologin user03
[user01@host ~]$ su - user03
Last login: Wed Feb  6 17:03:06 IST 2019 on pts/0
This account is currently not available.
```



IMPORTANT

Le shell **nologin** empêche l'utilisation interactive du système, mais ne bloque pas tous les accès. Les utilisateurs peuvent être en mesure de s'authentifier pour envoyer ou récupérer des fichiers par le biais d'applications comme les navigateurs Web, les programmes de transfert de fichiers ou les programmes de lecture de courrier s'ils utilisent le mot de passe de l'utilisateur pour s'authentifier.



RÉFÉRENCES

Pages de manuel **chage(1)**, **usermod(8)**, **shadow(5)**, **crypt(3)**

► EXERCICE GUIDÉ

GESTION DES MOTS DE PASSE DES UTILISATEURS

Dans cet exercice, vous allez définir des politiques de mot de passe pour plusieurs utilisateurs.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Forcer un changement de mot de passe lorsque l'utilisateur se connecte au système pour la première fois.
- Forcer un changement de mot de passe tous les 90 jours.
- Configurer le compte pour qu'il expire 180 jours à compter du jour actuel.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez **lab users-pw-manage start** pour mettre fin à l'exercice. Ce script crée les comptes d'utilisateur et les fichiers nécessaires pour garantir que l'environnement est correctement configuré.

```
[student@workstation ~]$ lab users-pw-manage start
```

- 1. À partir de **workstation**, ouvrez une session SSH sur **servera** en tant que **student**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Sur **servera**, explorez le verrouillage et le déverrouillage des comptes d'utilisateur en tant que **student**.

- 2.1. En tant que **student**, verrouillez le compte **operator1** en utilisant les droits administratifs.

```
[student@servera ~]$ sudo usermod -L operator1
[sudo] password for student: student
```

- 2.2. Tentez de vous connecter en tant qu'**operator1**. Cette opération doit échouer.

```
[student@servera ~]$ su - operator1
Password: redhat
su: Authentication failure
```

2.3. Déverrouillez le compte operator1.

```
[student@servera ~]$ sudo usermod -U operator1
```

2.4. Tentez de vous connecter à nouveau en tant qu'operator1. Cette opération devrait réussir.

```
[student@servera ~]$ su - operator1
Password: redhat
...output omitted...
[operator1@servera ~]$
```

2.5. Quittez le shell de l'utilisateur operator1 pour revenir au shell de l'utilisateur student.

```
[operator1@servera ~]$ exit
logout
```

- ▶ 3. Changez la politique du mot de passe de l'utilisateur operator1 pour exiger un nouveau mot de passe tous les 90 jours. Confirmez que l'âge du mot de passe est correctement défini.

3.1. Définissez l'âge maximal du mot de passe de l'utilisateur operator1 sur 90 jours.

```
[student@servera ~]$ sudo chage -M 90 operator1
```

3.2. Vérifiez que le mot de passe de l'utilisateur operator1 expire 90 jours après sa modification.

```
[student@servera ~]$ sudo chage -l operator1
Last password change      : Jan 25, 2019
Password expires          : Apr 25, 2019
Password inactive         : never
Account expires           : never
Minimum number of days between password change   : 0
Maximum number of days between password change   : 90
Number of days of warning before password expires : 7
```

- ▶ 4. Obligez l'utilisateur à changer de mot de passe lors de sa première connexion au compte operator1.

```
[student@servera ~]$ sudo chage -d 0 operator1
```

- ▶ 5. Connectez-vous en tant qu'operator1 et remplacez le mot de passe par forsooth123. Après avoir défini le mot de passe, revenez au shell de l'utilisateur student.

CHAPITRE 6 | Gestion des utilisateurs et des groupes locaux

- 5.1. Connectez-vous en tant qu'**operator1** et remplacez le mot de passe par **forsooth123** lorsque vous y êtes invité.

```
[student@servera ~]$ su - operator1
Password: redhat
You are required to change your password immediately (administrator enforced)
Current password: redhat
New password: forsooth123
Retype new password: forsooth123
...output omitted...
[operator1@servera ~]$
```

- 5.2. Quittez le shell de l'utilisateur **operator1** pour revenir au shell de l'utilisateur **student**.

```
[operator1@servera ~]$ exit
logout
```

- ▶ 6. Configurez le compte **operator1** pour qu'il expire 180 jours à compter du jour actuel.
Conseil : la commande **date -d "+180 days"** vous donne la date et l'heure, 180 jours à compter de la date et l'heure actuelles.

- 6.1. Déterminez une date située 180 jours dans le futur. Utilisez le format **%F** avec la commande **date** pour obtenir la valeur exacte.

```
[student@servera ~]$ date -d "+180 days" +%F
2019-07-24
```

Vous pouvez obtenir une valeur différente à utiliser à l'étape suivante en fonction de la date et de l'heure actuelles de votre système.

- 6.2. Définissez le compte pour qu'il expire à la date affichée à l'étape précédente.

```
[student@servera ~]$ sudo chage -E 2019-07-24 operator1
```

- 6.3. Vérifiez que la date d'expiration du compte est correctement définie.

```
[student@servera ~]$ sudo chage -l operator1
Last password change      : Jan 25, 2019
Password expires          : Apr 25, 2019
Password inactive         : never
Account expires           : Jul 24, 2019
Minimum number of days between password change   : 0
Maximum number of days between password change   : 90
Number of days of warning before password expires : 7
```

- ▶ 7. Définissez les mots de passe pour qu'ils expirent 180 jours à compter de la date actuelle pour tous les utilisateurs. Utilisez les droits administratifs pour modifier le fichier de configuration.

- 7.1. Définissez PASS_MAX_DAYS sur **180** dans **/etc/login.defs**. Utilisez les droits administratifs lors de l'ouverture du fichier avec l'éditeur de texte. Vous pouvez utiliser la commande **sudo vim /etc/login.defs** pour effectuer cette étape.

```
...output omitted...
# Password aging controls:
#
#      PASS_MAX_DAYS    Maximum number of days a password may be
#      used.
#      PASS_MIN_DAYS    Minimum number of days allowed between
#      password changes.
#      PASS_MIN_LEN     Minimum acceptable password length.
#      PASS_WARN_AGE    Number of days warning given before a
#      password expires.
#
PASS_MAX_DAYS    180
PASS_MIN_DAYS    0
PASS_MIN_LEN     5
PASS_WARN_AGE    7
...output omitted...
```



IMPORTANT

Les paramètres de mot de passe et d'expiration de compte par défaut s'appliqueront aux nouveaux utilisateurs, mais pas aux utilisateurs existants.

- 7.2. Déconnectez-vous de servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Fin

Sur workstation, exéutez **lab users-pw-manage finish** pour mettre fin à l'exercice. Ce script supprime les comptes d'utilisateur et les fichiers créés au début de l'exercice et permet de nettoyer l'environnement.

```
[student@workstation ~]$ lab users-pw-manage finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

GESTION DES UTILISATEURS ET DES GROUPES LOCAUX

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez définir une politique de mot de passe locale par défaut, créer un groupe supplémentaire pour trois utilisateurs, permettre à ce groupe d'utiliser **sudo** pour exécuter des commandes en tant que **root** et modifier la politique de mot de passe d'un utilisateur.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Définir une politique de péremption du mot de passe par défaut pour le mot de passe de l'utilisateur local.
- Créer un groupe et l'utiliser comme groupe supplémentaire pour les nouveaux utilisateurs.
- Créer trois utilisateurs avec, comme groupe supplémentaire, ce nouveau groupe.
- Configurer les membres du groupe supplémentaire afin qu'ils puissent exécuter toute commande en tant qu'utilisateur à l'aide de **sudo**.
- Définir une politique de péremption de mot de passe propre à l'utilisateur.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez **lab users-review start** pour démarrer l'exercice. Ce script crée les fichiers nécessaires pour garantir que l'environnement est correctement configuré.

```
[student@workstation ~]$ lab users-review start
```

1. À partir de **workstation**, ouvrez une session SSH sur **serverb** en tant que **student**.
2. Sur **serverb**, assurez-vous que les nouveaux utilisateurs disposent d'un mot de passe qui doit être changé tous les 30 jours.
3. Créez le groupe **consultants** avec le GID **35000**.
4. Configurez les droits administratifs de tous les membres du groupe **consultants** afin qu'ils puissent exécuter n'importe quelle commande en tant qu'utilisateur.
5. Créez les utilisateurs **consultant1**, **consultant2** et **consultant3** avec le groupe **consultants** comme groupe supplémentaire.
6. Définissez les comptes **consultant1**, **consultant2** et **consultant3** pour qu'ils expirent dans 90 jours à compter du jour actuel.

7. Changez la politique de mot de passe du compte **consultant2**, pour qu'elle exige un nouveau mot de passe tous les 15 jours.
8. De plus, obligez les utilisateurs **consultant1**, **consultant2** et **consultant3** à changer leurs mots de passe lors de la première connexion.

Évaluation

Sur **workstation**, exécutez la commande **lab users-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab users-review grade
```

Finish (Terminer)

Sur **workstation**, exécutez **lab users-review finish** pour mettre fin à l'atelier. Ce script supprime les comptes d'utilisateur et les fichiers créés au cours de l'atelier et permet de nettoyer l'environnement.

```
[student@workstation ~]$ lab users-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

GESTION DES UTILISATEURS ET DES GROUPES LOCAUX

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez définir une politique de mot de passe locale par défaut, créer un groupe supplémentaire pour trois utilisateurs, permettre à ce groupe d'utiliser **sudo** pour exécuter des commandes en tant que **root** et modifier la politique de mot de passe d'un utilisateur.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Définir une politique de péremption du mot de passe par défaut pour le mot de passe de l'utilisateur local.
- Créer un groupe et l'utiliser comme groupe supplémentaire pour les nouveaux utilisateurs.
- Créer trois utilisateurs avec, comme groupe supplémentaire, ce nouveau groupe.
- Configurer les membres du groupe supplémentaire afin qu'ils puissent exécuter toute commande en tant qu'utilisateur à l'aide de **sudo**.
- Définir une politique de péremption de mot de passe propre à l'utilisateur.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez **lab users-review start** pour démarrer l'exercice. Ce script crée les fichiers nécessaires pour garantir que l'environnement est correctement configuré.

```
[student@workstation ~]$ lab users-review start
```

1. À partir de **workstation**, ouvrez une session SSH sur **serverb** en tant que **student**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

2. Sur **serverb**, assurez-vous que les nouveaux utilisateurs disposent d'un mot de passe qui doit être changé tous les 30 jours.

- 2.1. Définissez **PASS_MAX_DAYS** sur **30** dans **/etc/login.defs**. Utilisez les droits administratifs lors de l'ouverture du fichier avec l'éditeur de texte. Vous pouvez utiliser

CHAPITRE 6 | Gestion des utilisateurs et des groupes locaux

la commande **sudo vim /etc/login.defs** pour effectuer cette étape. Utilisez **student** comme mot de passe lorsque **sudo** vous invite à entrer le mot de passe de l'utilisateur **student**.

```
...output omitted...
# Password aging controls:
#
#      PASS_MAX_DAYS    Maximum number of days a password may be
#      used.
#      PASS_MIN_DAYS    Minimum number of days allowed between
#      password changes.
#      PASS_MIN_LEN     Minimum acceptable password length.
#      PASS_WARN_AGE    Number of days warning given before a
#      password expires.
#
PASS_MAX_DAYS    30
PASS_MIN_DAYS    0
PASS_MIN_LEN     5
PASS_WARN_AGE    7
...output omitted...
```

- Créez le groupe **consultants** avec le GID **35000**.

```
[student@serverb ~]$ sudo groupadd -g 35000 consultants
```

- Configurez les droits administratifs de tous les membres du groupe **consultants** afin qu'ils puissent exécuter n'importe quelle commande en tant qu'utilisateur.
 - Créez le fichier **/etc/sudoers.d/consultants** et ajoutez-y le contenu suivant.
Vous pouvez utiliser la commande **sudo vim /etc/sudoers.d/consultants** pour effectuer cette étape.

```
%consultants  ALL=(ALL) ALL
```

- Créez les utilisateurs **consultant1**, **consultant2** et **consultant3** avec le groupe **consultants** comme groupe supplémentaire.

```
[student@serverb ~]$ sudo useradd -G consultants consultant1
[student@serverb ~]$ sudo useradd -G consultants consultant2
[student@serverb ~]$ sudo useradd -G consultants consultant3
```

- Définissez les comptes **consultant1**, **consultant2** et **consultant3** pour qu'ils expirent dans 90 jours à compter du jour actuel.
 - Déterminez une date située 90 jours dans le futur. Vous pouvez obtenir une valeur différente par rapport à la sortie suivante en fonction de la date et de l'heure actuelles de votre système.

```
[student@serverb ~]$ date -d "+90 days" +%F
2019-04-28
```

- Définissez la date d'expiration des comptes **consultant1**, **consultant2** et **consultant3** avec la même valeur que celle déterminée à l'étape précédente.

```
[student@serverb ~]$ sudo chage -E 2019-04-28 consultant1  
[student@serverb ~]$ sudo chage -E 2019-04-28 consultant2  
[student@serverb ~]$ sudo chage -E 2019-04-28 consultant3
```

7. Changez la politique de mot de passe du compte **consultant2**, pour qu'elle exige un nouveau mot de passe tous les 15 jours.

```
[student@serverb ~]$ sudo chage -M 15 consultant2
```

8. De plus, obligez les utilisateurs **consultant1**, **consultant2** et **consultant3** à changer leurs mots de passe lors de la première connexion.
- 8.1. Définissez le dernier jour du changement de mot de passe sur **0** de sorte que les utilisateurs soient obligés de changer le mot de passe lors de leur première connexion au système.

```
[student@serverb ~]$ sudo chage -d 0 consultant1  
[student@serverb ~]$ sudo chage -d 0 consultant2  
[student@serverb ~]$ sudo chage -d 0 consultant3
```

8.2. Déconnectez-vous de **serverb**.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.
```

Évaluation

Sur **workstation**, exécutez la commande **lab users-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab users-review grade
```

Finish (Terminer)

Sur **workstation**, exécutez **lab users-review finish** pour mettre fin à l'atelier. Ce script supprime les comptes d'utilisateur et les fichiers créés au cours de l'atelier et permet de nettoyer l'environnement.

```
[student@workstation ~]$ lab users-review finish
```

L'atelier est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Les trois principaux types de comptes d'utilisateur sont super utilisateur, utilisateur système et utilisateur normal.
- Un utilisateur doit avoir un groupe principal et peut être membre d'un ou de plusieurs groupes supplémentaires.
- Les trois fichiers critiques contenant des informations sur les utilisateurs et les groupes sont : **/etc/passwd**, **/etc/group** et **/etc/shadow**.
- Les commandes **su** et **sudo** peuvent être utilisées pour exécuter des commandes en tant que super utilisateur.
- Les commandes **useradd**, **usermod** et **userdel** peuvent être utilisées pour gérer les utilisateurs.
- Les commandes **groupadd**, **groupmod** et **groupdel** peuvent être utilisées pour gérer les groupes.
- La commande **chage** peut être utilisée pour configurer et afficher les paramètres d'expiration du mot de passe pour les utilisateurs.

CHAPITRE 7

CONTRÔLE DE L'ACCÈS AUX FICHIERS

PROJET

Définir les permissions de système de fichiers de Linux sur des fichiers et évaluez les effets sur la sécurité de différents paramètres de permissions.

OBJECTIFS

- Lister les permissions du système de fichiers sur les fichiers et les répertoires, et évaluez l'effet de ces permissions sur l'accès des utilisateurs et des groupes.
- Changer les permissions et la propriété des fichiers en utilisant des outils de ligne de commande.
- Contrôler les permissions par défaut des fichiers créés par les utilisateurs, expliquer l'effet des permissions spéciales, et utiliser des permissions spéciales et par défaut pour définir le groupe propriétaire des fichiers créés dans un répertoire particulier.

SECTIONS

- Interprétation des permissions du système de fichiers Linux (et quiz)
- Gestion des permissions du système de fichiers à partir de la ligne de commande (et exercice guidé)
- Gestion des permissions et de l'accès aux fichiers par défaut (et exercice guidé)

ATELIER

Contrôle de l'accès aux fichiers

INTERPRÉTATION DES PERMISSIONS DU SYSTÈME DE FICHIERS LINUX

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir lister les permissions du système de fichiers sur les fichiers et les répertoires, et évaluer l'effet de ces permissions sur l'accès des utilisateurs et des groupes.

PERMISSIONS DU SYSTÈME DE FICHIERS LINUX

Les *permissions de fichier* contrôlent l'accès aux fichiers. Les permissions de fichier Linux sont simples mais flexibles, faciles à comprendre et à appliquer, mais aussi capables de traiter aisément la plupart des cas de permissions classiques.

Les fichiers sont associés à trois catégories d'utilisateurs, auxquelles s'appliquent des permissions. Le fichier est la propriété d'un utilisateur, généralement le créateur du fichier. Le fichier appartient également à un groupe unique. Il s'agit généralement du groupe principal dont dépend l'utilisateur ayant créé le fichier. Toutefois, ce groupe peut être modifié. Des permissions différentes peuvent être définies pour l'utilisateur propriétaire, le groupe propriétaire et tous les autres utilisateurs du système autres que l'utilisateur ou un membre du groupe propriétaire.

Les permissions les plus spécifiques prévalent. Les permissions de l'*utilisateur* prennent sur les permissions du *groupe*, lesquelles prennent sur les permissions des *autres*.

Dans Figure 7.1, joshua est membre des groupes joshua et web, tandis qu'allison fait partie des groupes allison, wheel et web. Quand joshua et allison doivent collaborer, les fichiers doivent être associés au groupe web, et les permissions du groupe doivent autoriser l'accès voulu.

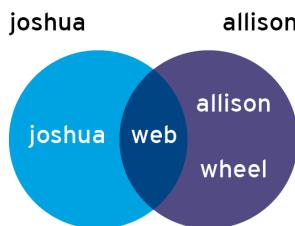


Figure 7.1: Exemple d'appartenance à un groupe pour faciliter la collaboration

Il existe trois catégories de permissions : lecture, écriture et exécution. Le tableau suivant explique comment ces permissions affectent l'accès aux fichiers et répertoires.

Effets des permissions sur les fichiers et les répertoires

PERMISSION	EFFET SUR LES FICHIERS	EFFET SUR LES RÉPERTOIRES
r (lecture)	Le contenu du fichier peut être lu.	Le contenu du répertoire (noms des fichiers) peut être listé.
w (écriture)	Le contenu du fichier peut être modifié.	Des fichiers peuvent être créés dans le répertoire ou supprimés de celui-ci.

PERMISSION	EFFET SUR LES FICHIERS	EFFET SUR LES RÉPERTOIRES
x (exécution)	Les fichiers peuvent être exécutés en tant que commandes.	Le répertoire peut devenir le répertoire de travail courant. (Vous pouvez cd dedans, mais également demander une permission de lecture pour lister les fichiers qui y sont trouvés.)

Les utilisateurs disposent normalement de permissions de lecture et d'exécution sur les répertoires en lecture seule, ce qui leur permet d'en lister le contenu et d'y accéder dans ce mode. Si un utilisateur possède uniquement un accès en lecture au répertoire, il peut lister le nom des fichiers qu'il contient, mais il ne peut pas y accéder, et aucune autre information n'est accessible ou disponible, y compris les permissions ou l'horodatage. Si un utilisateur dispose uniquement d'un accès d'exécution sur un répertoire, il ne peut pas lister les noms de fichiers dans le répertoire. S'il connaît le nom d'un fichier qu'il est autorisé à lire, il peut accéder au contenu de ce fichier depuis l'extérieur du répertoire en spécifiant explicitement le nom de fichier relatif.

Un fichier peut être supprimé par toute personne disposant de permissions d'écriture sur le répertoire dans lequel se trouve le fichier, quelles que soient la propriété ou les permissions du fichier en question. Cette règle peut être contournée au moyen d'une permission spéciale, le *sticky bit*, qui sera abordé plus loin dans ce chapitre.



NOTE

Les permissions de fichier Linux fonctionnent différemment du système de permission utilisé par le système de fichiers NTFS pour Microsoft Windows.

Sous Linux, les permissions s'appliquent uniquement au fichier ou répertoire sur lequel elles sont définies. Ainsi, les sous-répertoires et les fichiers contenus dans un répertoire donné n'héritent donc pas automatiquement des permissions définies pour ce dernier. Toutefois, les permissions sur un répertoire peuvent bloquer l'accès au contenu du répertoire en fonction de leur degré de restriction.

La permission d'accès en lecture sur un répertoire sous Linux équivaut approximativement à l'option Affichage du contenu du dossier sous Windows.

La permission d'accès en écriture sur un répertoire sous Linux est l'équivalent de l'option Modification sous Windows. Elle implique la possibilité de supprimer des fichiers et des sous-répertoires. Sous Linux, si un répertoire possède à la fois la permission d'accès en écriture et le sticky bit, seul le propriétaire du fichier ou du sous-répertoire peut le supprimer. Ce comportement s'apparente à celui de la permission Écriture de Windows.

L'utilisateur root de Linux dispose d'une permission équivalente à celle de Contrôle total sous Windows, sur tous les fichiers. Toutefois, l'utilisateur root peut voir son accès limité par la politique SELinux du système utilisant le contexte de sécurité du processus et des fichiers. SELinux fera l'objet d'un prochain cours.

AFFICHAGE DES PERMISSIONS ET DE LA PROPRIÉTÉ D'UN FICHIER ET D'UN RÉPERTOIRE

L'option **-l** de la commande **ls** affiche des informations détaillées sur les permissions et la propriété :

CHAPITRE 7 | Contrôle de l'accès aux fichiers

```
[user@host~]$ ls -l test  
-rw-rw-r--. 1 student student 0 Feb 8 17:36 test
```

Utilisez l'option **-d** pour afficher les informations détaillées du répertoire lui-même, et non de son contenu.

```
[user@host ~]$ ls -ld /home  
drwxr-xr-x. 5 root root 4096 Jan 31 22:00 /home
```

Le premier caractère de la longue liste est le type de fichier, il doit être interprété comme suite :

- **-** est un fichier normal
- **d** est un répertoire.
- **l** est un lien symbolique.
- Les autres caractères représentent des périphériques matériels (**b** et **c**) ou d'autres fichiers à usage spécifique (**p** et **s**).

Les neuf caractères suivants sont les permissions de fichier. Celles-ci se composent de trois jeux de trois caractères : les permissions qui s'appliquent à l'utilisateur propriétaire du fichier, au groupe propriétaire du fichier et à tous les autres utilisateurs. Si l'ensemble indique **rwx**, cette catégorie possède les trois autorisations : lire, écrire et exécuter. Si une lettre a été remplacée par **-**, alors cette catégorie n'a pas cette permission.

Après le nombre de liens, le premier nom spécifie l'utilisateur à qui appartient le fichier et le deuxième nom, le groupe à qui appartient le fichier.

Ainsi, dans l'exemple ci-dessus, les permissions pour l'utilisateur **student** sont spécifiées par le premier ensemble de trois caractères. L'utilisateur **student** a lu et écrit sur **test**, mais n'a pas exécuté.

Le groupe **student** est spécifié par le deuxième ensemble de trois caractères : il a également lu et écrit sur **test**, mais n'a pas exécuté.

Les permissions des autres utilisateurs sont spécifiées par le troisième ensemble de trois caractères : ils ne disposent que d'une autorisation en lecture sur **test**.

La permission la plus spécifique s'applique. Donc si l'utilisateur **student** a des autorisations différentes de celles du groupe **student** et si l'utilisateur **student** est également membre de ce groupe, alors les permissions de l'utilisateur seront appliquées.

EXEMPLES D'EFFETS DE PERMISSION

Les exemples suivants aideront à illustrer l'interaction des permissions de fichiers. Pour ces exemples, nous avons quatre utilisateurs avec les appartenances aux groupes suivants :

UTILISATEUR	APPARTENANCES AUX GROUPES
operator1	operator1, consultant1
database1	database1, consultant1
database2	database2, operator2
contractor1	contractor1, operator2

CHAPITRE 7 | Contrôle de l'accès aux fichiers

Ces utilisateurs travailleront avec des fichiers dans le répertoire **dir**. Il s'agit d'une longue liste de fichiers dans ce répertoire :

```
[database1@host dir]$ ls -la
total 24
drwxrwxr-x.  2 database1 consultant1  4096 Apr  4 10:23 .
drwxr-xr-x. 10 root      root       4096 Apr  1 17:34 ..
-rw-rw-r--.  1 operator1 operator1   1024 Apr  4 11:02 lfile1
-rw-r--rw-.  1 operator1 consultant1 3144 Apr  4 11:02 lfile2
-rw-rw-r--.  1 database1 consultant1 10234 Apr  4 10:14 rfile1
-rw-r-----. 1 database1 consultant1 2048 Apr  4 10:18 rfile2
```

L'option **-a** affiche les permissions des fichiers cachés, y compris les fichiers spéciaux utilisés pour représenter le répertoire et son parent. Dans cet exemple, **.** reflète les permissions de **dir** lui-même et **..** les autorisations de son répertoire parent.

Quelles sont les permissions de **rfile1**? L'utilisateur qui possède le fichier (**database1**) a lu et écrit, mais pas exécuté. Le groupe qui possède le fichier (**consultant1**) a lu et écrit, mais pas exécuté. Tous les autres utilisateurs ont lu, mais pas écrit ni exécuté.

Le tableau suivant explore certains des effets de cet ensemble de permissions sur ces utilisateurs :

EFFET	POURQUOI EST-CE VRAI ?
L'utilisateur operator1 peut modifier le contenu de rfile1 .	L'utilisateur operator1 est membre du groupe consultant1 , et ce groupe a accès en lecture et en écriture au fichier rfile1 .
L'utilisateur database1 peut voir et modifier le contenu du fichier rfile2 .	L'utilisateur database1 est le propriétaire du fichier, et il a accès à la fois en lecture et en écriture au fichier rfile2 .
L'utilisateur opérateur1 peut voir mais pas modifier le contenu du fichier rfile2 (sans le supprimer ni le recréer).	L'utilisateur operator1 est membre du groupe consultant1 , a accès au fichier rfile2 uniquement en lecture.
Les utilisateurs database2 et contractor1 n'ont aucun accès au contenu du fichier rfile2 .	Les permissions other s'appliquent aux utilisateurs database2 et contractor1 ; elles ne comprennent aucun accès en lecture ou en écriture.
operator1 est le seul utilisateur à pouvoir modifier le contenu du fichier lfile1 (sans le supprimer ni le recréer).	L'utilisateur et le groupe operator1 ont accès au fichier en écriture, ce n'est pas le cas des autres utilisateurs. Mais le seul membre du groupe operator1 est l'utilisateur operator1 .
L'utilisateur database2 peut modifier le contenu du fichier lfile2 .	L'utilisateur database2 n'est pas l'utilisateur propriétaire du fichier et n'est pas membre du groupe consultant1 , donc les permissions d' other s'appliquent. Ceux-ci accordent une permission en écriture.

EFFET	POURQUOI EST-CE VRAI ?
L'utilisateur database1 peut voir le contenu du fichier lfile2 , mais pas modifier le contenu du fichier lfile2 (sans le supprimer ni le recréer).	L'utilisateur database1 est membre du groupe consultant1, qui a accès au fichier lfile2 uniquement en lecture. Bien qu' other dispose d'une permission d'écriture, les permissions du groupe priment.
L'utilisateur database1 peut supprimer les fichiers lfile1 et lfile2 .	L'utilisateur database1 peut accéder en écriture au répertoire contenant les deux fichiers (indiqué par .), et peut donc effacer n'importe quel fichier de ce répertoire. Ceci est vrai même si l'utilisateur database1 n'a pas le droit d'écriture sur le fichier lui-même.



RÉFÉRENCES

Page de manuel **ls(1)**

info coreutils (*GNU Coreutils*)

- Section 13 : Modification des attributs d'un fichier

► QUIZ

INTERPRÉTATION DES PERMISSIONS DU SYSTÈME DE FICHIERS LINUX

Passez en revue les informations suivantes et utilisez-les pour répondre aux questions du quiz.

Le système a quatre utilisateurs affectés aux groupes suivants :

- L'utilisateur **consultant1** est membre des groupes **consultant1** et **database1**
- L'utilisateur **operator1** est membre des groupes **operator1** et **database1**
- L'utilisateur **contractor1** est membre des groupes **contractor1** et **contractor3**
- L'utilisateur **operator2** est membre des groupes **operator2** et **contractor3**

Le répertoire courant (.) contient quatre fichiers avec les informations de permissions suivantes :

```
drwxrwxr-x. operator1 database1 .
-rw-rw-r--. consultant1 consultant1 lfile1
-rw-r--rw-. consultant1 database1 lfile2
-rw-rw-r--. operator1 database1 rfile1
-rw-r-----. operator1 database1 rfile2
```

► 1. Quel fichier appartenant à **operator1** est accessible en lecture par tous les utilisateurs ?

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► 2. Quel fichier peut être modifié par l'utilisateur **contractor1** ?

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► 3. Quel fichier ne peut pas être lu par l'utilisateur **operator2** ?

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► **4. Quel fichier appartient au groupe consultant1 ?**

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► **5. Quels fichiers peuvent être supprimés par l'utilisateur operator1 ?**

- a. **rfile1**
- b. **rfile2**
- c. Toutes les réponses précédentes.
- d. Aucune de ces réponses.

► **6. Quels fichiers peuvent être supprimés par l'utilisateur operator2 ?**

- a. **lfile1**
- b. **lfile2**
- c. Toutes les affirmations précédentes.
- d. Aucune de ces réponses.

► SOLUTION

INTERPRÉTATION DES PERMISSIONS DU SYSTÈME DE FICHIERS LINUX

Passez en revue les informations suivantes et utilisez-les pour répondre aux questions du quiz.

Le système a quatre utilisateurs affectés aux groupes suivants :

- L'utilisateur **consultant1** est membre des groupes **consultant1** et **database1**
- L'utilisateur **operator1** est membre des groupes **operator1** et **database1**
- L'utilisateur **contractor1** est membre des groupes **contractor1** et **contractor3**
- L'utilisateur **operator2** est membre des groupes **operator2** et **contractor3**

Le répertoire courant (.) contient quatre fichiers avec les informations de permissions suivantes :

```
drwxrwxr-x. operator1 database1 .
-rw-rw-r--. consultant1 consultant1 lfile1
-rw-r--rw-. consultant1 database1 lfile2
-rw-rw-r--. operator1 database1 rfile1
-rw-r-----. operator1 database1 rfile2
```

► 1. Quel fichier appartenant à **operator1** est accessible en lecture par tous les utilisateurs ?

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► 2. Quel fichier peut être modifié par l'utilisateur **contractor1** ?

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► 3. Quel fichier ne peut pas être lu par l'utilisateur **operator2** ?

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► **4. Quel fichier appartient au groupe consultant1 ?**

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► **5. Quels fichiers peuvent être supprimés par l'utilisateur operator1 ?**

- a. **rfile1**
- b. **rfile2**
- c. Toutes les réponses précédentes.
- d. Aucune de ces réponses.

► **6. Quels fichiers peuvent être supprimés par l'utilisateur operator2 ?**

- a. **lfile1**
- b. **lfile2**
- c. Toutes les affirmations précédentes.
- d. Aucune de ces réponses.

GESTION DES PERMISSIONS DU SYSTÈME DE FICHIERS À PARTIR DE LA LIGNE DE COMMANDE

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir modifier les permissions et la propriété des fichiers à l'aide d'outils de ligne de commande.

MODIFICATION DES PERMISSIONS DES FICHIERS ET DES RÉPERTOIRES

La commande utilisée pour modifier les permissions depuis la ligne de commande est **chmod**, contraction de l'anglais « change mode » ou « modifier le mode » (les permissions sont aussi appelées le *mode* d'un fichier). La commande **chmod** prend une instruction de permission d'accès, suivie par une liste de fichiers ou de répertoires à modifier. L'instruction de permission peut être soit symbolique (méthode symbolique), soit numérique (méthode numérique).

Modification des permissions à l'aide de la méthode symbolique

```
chmod WhoWhatWhich file|directory
```

- Qui correspond à u, g, o, a (*pour utilisateur, groupe, other [autre], all [tous]*)
- Quoi correspond à +, -, = (*pour ajouter, supprimer, définir exactement*)
- Comment correspond à r, w, x (*pour read [lecture], write [écriture], execute [exécution]*)

La méthode *symbolique* de modification des permissions de fichier utilise des lettres pour représenter les différents groupes de permissions : **u** pour l'utilisateur, **g** pour le groupe, **o** pour les autres (other) et **a** pour tous (all).

Avec la méthode symbolique, il n'est pas nécessaire de définir un nouveau groupe de permissions dans son ensemble. Vous pouvez, au contraire, modifier une ou plusieurs des permissions existantes. Utilisez **+** ou **-** pour ajouter ou supprimer des permissions, respectivement, ou **=** pour remplacer l'ensemble complet pour un groupe de permissions.

Les permissions proprement dites sont représentées par une seule lettre : **r** pour la lecture, **w** pour l'écriture et **x** pour l'exécution. Lorsque la commande **chmod** est utilisée pour modifier les permissions à l'aide de la méthode symbolique, l'utilisation de la lettre X en majuscule comme balise de permission ajoute une permission d'exécution uniquement si le fichier est un répertoire ou si l'exécution est déjà définie pour « user », « group » ou « other ».

**NOTE**

La commande **chmod** prend en charge l'option **-R** pour définir des permissions de manière récursive sur les fichiers de toute une arborescence de répertoires. Lors de l'utilisation de l'option **-R**, il peut être utile de définir les permissions de manière symbolique à l'aide de l'option **X**. Ainsi, la permission d'exécution (recherche) pourra être définie sur les répertoires, de sorte que leur contenu soit accessible sans avoir à modifier les permissions de la plupart des fichiers. L'option **X** requiert de la prudence, car si un fichier est associé à un ensemble de permissions d'exécution, **X** définit la permission d'exécution spécifiée également sur ce fichier. Par exemple, la commande suivante définit de manière récursive un accès en lecture et en écriture sur **demodir**, ainsi que tous ses enfants, pour le propriétaire du groupe, mais applique les permissions d'exécution de groupe uniquement aux répertoires et aux fichiers ayant déjà une exécution définie pour user, group ou other.

```
[root@host opt]# chmod -R g+rwx demodir
```

Exemples

- Supprimez les permissions de lecture et d'écriture sur **file1** pour le groupe et les autres :

```
[user@host ~]$ chmod go-rw file1
```

- Ajoutez la permission d'exécution pour tous sur **file2** :

```
[user@host ~]$ chmod a+x file2
```

Modification des permissions à l'aide de la méthode numérique

Dans l'exemple ci-dessous, le caractère # représente un chiffre.

```
chmod ### file|directory
```

- Chaque chiffre représente un niveau d'accès : utilisateur (user), groupe (group), autre (other).
- Le chiffre est calculé en additionnant les nombres pour chaque permission à ajouter, 4 en lecture, 2 en écriture et 1 en exécution.

Avec la méthode *numérique*, les permissions sont représentées par un nombre *octal* à trois chiffres (ou quatre, pour les permissions avancées). Un seul chiffre octal peut représenter n'importe quelle valeur de 0 à 7.

Dans la représentation octale à trois chiffres (numérique) des permissions, chaque chiffre représente un niveau d'accès, de gauche à droite : utilisateur (user), groupe (group), autre (other). Pour déterminer chaque chiffre:

- Commencez par 0.
- Si l'autorisation de lecture doit être présente pour ce niveau d'accès, ajoutez 4.
- Si l'autorisation d'écriture doit être présente, ajoutez 2.
- Si l'autorisation d'exécution doit être présente, ajoutez 1.

CHAPITRE 7 | Contrôle de l'accès aux fichiers

Examinez les permissions **-rwxr-x---**. Pour l'utilisateur, **rwx** est calculé sous la forme $4+2+1=7$. Pour le groupe, **r-x** est calculé comme $4+0+1=5$, et pour d'autres utilisateurs, **---** est représenté par 0. En réunissant ces trois éléments, la représentation numérique de ces autorisations est 750.

Ce calcul peut aussi être exécuté dans le sens inverse. Examinez les permissions 640. Pour les permissions de l'utilisateur, 6 représente la lecture (4) et l'écriture (2), ce qui apparaît sous la forme **rw-**. Pour le groupe, 4 ne comprend que la lecture (4) et s'affiche sous la forme **r--**. Le 0 appliqué aux autres n'accorde aucune permission (**---**), et l'ensemble final des permissions symboliques pour ce fichier est donc **-rw-r----**.

Les administrateurs expérimentés utilisent souvent les permissions numériques, car elles sont plus courtes à saisir et à énoncer, tout en donnant un contrôle total sur toutes les permissions.

Exemples

- Définissez les permissions de lecture et d'écriture pour l'utilisateur, les permissions de lecture pour le groupe et les autres sur **samplefile** :

```
[user@host ~]$ chmod 644 samplefile
```

- Définissez les permissions de lecture, d'écriture et d'exécution pour l'utilisateur, les permissions de lecture et d'exécution pour le groupe, et aucune permission pour les autres sur **sampledir** :

```
[user@host ~]$ chmod 750 sampledir
```

MODIFICATION DES DROITS DE PROPRIÉTÉ DU GROUPE OU DE L'UTILISATEUR SUR LES FICHIERS ET LES RÉPERTOIRES

Un fichier nouvellement créé est la propriété de l'utilisateur qui l'a créé. Par défaut, les nouveaux fichiers appartiennent à un groupe, qui est le groupe principal de l'utilisateur qui l'a créé. Sur Red Hat Enterprise Linux, le groupe principal de l'utilisateur est généralement un groupe privé qui ne compte qu'un seul membre : cet utilisateur. Pour accorder l'accès en fonction de l'appartenance à un groupe, il peut s'avérer nécessaire de modifier le groupe propriétaire du fichier.

Seul **root** peut modifier l'utilisateur propriétaire d'un fichier. Cependant, la propriété du groupe peut être définie par **root** ou par le propriétaire du fichier. **root** peut accorder la propriété à n'importe quel groupe, mais les utilisateurs normaux peuvent accorder uniquement la propriété des groupes auxquels ils appartiennent.

La propriété d'un fichier peut être modifiée à l'aide de la commande **chown** (modifier le propriétaire). Par exemple, pour accorder la propriété du fichier **test_file** à l'utilisateur **student**, utilisez la commande suivante :

```
[root@host ~]# chown student test_file
```

La commande **chown** peut être utilisée avec l'option **-R** pour modifier de manière récursive la propriété d'une arborescence de répertoires entière. La commande suivante accorde la propriété de **test_dir** et de tous les fichiers et sous-répertoires qui s'y trouvent à **student** :

```
[root@host ~]# chown -R student test_dir
```

On peut aussi utiliser la commande **chown** pour modifier la propriété de groupe d'un fichier en faisant précéder le nom du groupe du caractère « deux-points » (:). Par exemple, la commande suivante fait passer le groupe de **test_dir** à **admins** :

```
[root@host ~]# chown :admins test_dir
```

La commande **chown** peut aussi servir à modifier à la fois le propriétaire et le groupe, avec la syntaxe *owner:group*. Par exemple, pour que **test_dir** devienne la propriété de **visitor** et du groupe **guests**, utilisez la commande suivante :

```
[root@host ~]# chown visitor:guests test_dir
```

Au lieu d'utiliser **chown**, certains utilisateurs changent la propriété du groupe en utilisant la commande **chgrp**. Cette commande fonctionne exactement comme **chown**, sauf qu'elle n'est utilisée que pour changer la propriété du groupe et les deux-points (:) avant que le nom du groupe n'est pas requis.



IMPORTANT

Vous pouvez rencontrer des exemples de commandes **chown** utilisant une syntaxe alternative qui séparent le propriétaire et le groupe avec un point au lieu de deux points :

```
[root@host ~]# chown owner.group filename
```

Vous ne devez pas utiliser cette syntaxe. Utilisez systématiquement les deux-points.

Un point est un caractère valide dans un nom d'utilisateur, mais pas les deux-points. Si l'utilisateur **enoch.root**, l'utilisateur **enoch** et le groupe **root** existent sur le système, le résultat de **chown enoch.root filename** implique que **filename** appartient à l'utilisateur **enoch.root**. Vous avez peut-être essayé de définir la propriété du fichier sur l'utilisateur **enoch** et le groupe **root**. Cela peut être déroutant.

Si vous utilisez toujours la syntaxe **chown** des deux-points lors de la définition simultanée de l'utilisateur et du groupe, les résultats sont toujours faciles à prédire.



RÉFÉRENCES

Pages du manuel **ls(1)**, **chmod(1)**, **chown(1)** et **chgrp(1)**

► EXERCICE GUIDÉ

GESTION DES PERMISSIONS DU SYSTÈME DE FICHIERS À PARTIR DE LA LIGNE DE COMMANDE

Dans cet exercice, vous allez utiliser les permissions du système de fichiers pour créer un répertoire dans lequel tous les membres d'un groupe particulier peuvent ajouter et supprimer des fichiers.

RÉSULTATS

Vous devez pouvoir créer un répertoire de collaboration accessible à tous les membres d'un groupe particulier.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez la commande `lab perms-cli start`. Le script de démarrage crée un groupe appelé `consultants` et deux utilisateurs appelés `consultant1` et `consultant2`.

```
[student@workstation ~]$ lab perms-cli start
```

- 1. À partir de `workstation`, utilisez la commande `ssh` pour vous connecter à `servera` en tant qu'utilisateur `student`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Basculez vers l'utilisateur `root` en utilisant `redhat` comme mot de passe.

```
[student@servera ~]$ su -
Password: redhat
[root@servera ~]#
```

- 3. Utilisez la commande `mkdir` pour créer le répertoire `/home/consultants`.

```
[root@servera ~]# mkdir /home/consultants
```

- 4. Utilisez la commande `chown` pour transférer la propriété du groupe du répertoire `consultants` au groupe `consultants`.

CHAPITRE 7 | Contrôle de l'accès aux fichiers

```
[root@servera ~]# chown :consultants /home/consultants
```

- 5. Vérifiez que les permissions du groupe **consultants** permettent à ses membres de créer et de supprimer des fichiers dans le répertoire **/home/consultants**. Ces permissions doivent empêcher les autres utilisateurs d'accéder aux fichiers.
- 5.1. Utilisez la commande **ls** pour vérifier que les permissions du groupe **consultants** permettent à ses membres de créer et de supprimer des fichiers dans le répertoire **/home/consultants**.

```
[root@servera ~]# ls -ld /home/consultants
drwxr-xr-x. 2 root      consultants       6 Feb  1 12:08 /home/consultants
```

Notez que le groupe **consultants** ne dispose actuellement pas de la permission d'écriture.

- 5.2. Utilisez la commande **chmod** pour ajouter la permission d'écriture au groupe **consultants**.

```
[root@servera ~]# chmod g+w /home/consultants
[root@servera ~]# ls -ld /home/consultants
drwxrwxr-x. 2 root      consultants 6 Feb  1 13:21 /home/consultants
```

- 5.3. Utilisez la commande **chmod** pour interdire aux autres d'accéder aux fichiers du répertoire **/home/consultants**.

```
[root@servera ~]# chmod 770 /home/consultants
[root@servera ~]# ls -ld /home/consultants
drwxrwx---. 2 root      consultants 6 Feb  1 12:08 /home/consultants/
```

- 6. Quittez le shell root et basculez vers l'utilisateur **consultant1**. Le mot de passe est **redhat**.

```
[root@servera ~]# exit
logout
[student@servera ~]$
[student@servera ~]$ su - consultant1
Password: redhat
```

- 7. Accédez au répertoire **/home/consultants** et créez un fichier appelé **consultant1.txt**.

- 7.1. Utilisez la commande **cd** pour accéder au répertoire **/home/consultants**.

```
[consultant1@servera ~]$ cd /home/consultants
```

- 7.2. Utilisez la commande **touch** pour créer un répertoire vide appelé **consultant1.txt**.

```
[consultant1@servera consultants]$ touch consultant1.txt
```

CHAPITRE 7 | Contrôle de l'accès aux fichiers

- 8. Utilisez la commande **ls -l** pour lister la propriété par défaut des utilisateurs et groupes du nouveau fichier et ses permissions.

```
[consultant1@servera consultants]$ ls -l consultant1.txt  
-rw-rw-r-- 1 consultant1 consultant1 0 Feb 1 12:53 consultant1.txt
```

- 9. Veillez à ce que tous les membres du groupe **consultants** puissent modifier le fichier **consultant1.txt**. Transférez la propriété du groupe du fichier **consultant1.txt** au groupe **consultants**.
- 9.1. Utilisez la commande **chown** pour transférer la propriété du groupe du fichier **consultant1.txt** au groupe **consultants**.

```
[consultant1@servera consultants]$ chown :consultants consultant1.txt
```

- 9.2. Utilisez la commande **ls** avec l'option **-l** pour lister la nouvelle propriété du fichier **consultant1.txt**.

```
[consultant1@servera consultants]$ ls -l consultant1.txt  
-rw-rw-r-- 1 consultant1 consultants 0 Feb 1 12:53 consultant1.txt
```

- 10. Quittez le shell root et basculez vers l'utilisateur **consultant2**. Le mot de passe est **redhat**.

```
[consultant1@servera consultants]$ exit  
logout  
[student@servera ~]$ su - consultant2  
Password: redhat  
[consultant2@servera ~]$
```

- 11. Accédez au répertoire **/home/consultants**. Assurez-vous que l'utilisateur **consultant2** puisse ajouter du contenu au fichier **consultant1.txt**. Quittez le shell.

- 11.1. Utilisez la commande **cd** pour accéder au répertoire **/home/consultants**. Utilisez la commande **echo** pour ajouter du **texte** au fichier **consultant1.txt**.

```
[consultant2@servera ~]$ cd /home/consultants/  
[consultant2@servera consultants]$ echo "text" >> consultant1.txt  
[consultant2@servera consultants]$
```

- 11.2. Utilisez la commande **cat** pour vérifier que le texte a été ajouté au fichier **consultant1.txt**.

```
[consultant2@servera consultants]$ cat consultant1.txt  
text  
[consultant2@servera consultants]$
```

- 11.3. Quittez le shell.

```
[consultant2@servera consultants]$ exit  
logout  
[student@servera ~]$
```

- 12. Déconnectez-vous de servera.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Fin

Sur workstation, exéutez le script **lab perms-cli finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab perms-cli finish
```

L'exercice guidé est maintenant terminé.

GESTION DE L'ACCÈS AUX FICHIERS ET DES PERMISSIONS PAR DÉFAUT

OBJECTIFS

Après avoir terminé cette section, les stagiaires doivent pouvoir réaliser les tâches suivantes :

- Contrôler les permissions par défaut des nouveaux fichiers créés par les utilisateurs.
- Expliquer l'effet des permissions spéciales.
- Utiliser des permissions spéciales et des permissions par défaut pour définir le groupe de propriétaires des fichiers créé dans un répertoire particulier.

PERMISSIONS SPÉCIALES

Les *permissions spéciales* constituent un quatrième type de permission en plus des types d'utilisateur, de groupe et autres de base. Comme leur nom l'indique, ces permissions fournissent des fonctions supplémentaires liées à l'accès au-delà des possibilités offertes par les types de permission de base. Cette section détaille l'impact des permissions spéciales, résumées dans le tableau ci-dessous.

Effets des permissions spéciales sur les fichiers et les répertoires

PERMISSION SPÉCIALE	EFFET SUR LES FICHIERS	EFFET SUR LES RÉPERTOIRES
u+s (suid)	Le fichier s'exécute sous l'identité de l'utilisateur qui le possède, et non pour le compte de l'utilisateur qui l'exécute.	Sans effet.
g+s (sgid)	Le fichier s'exécute sous l'identité du groupe qui en est le propriétaire.	Les fichiers nouvellement créés dans le répertoire voient leur groupe propriétaire modifié pour correspondre à celui du répertoire.
o+t (sticky)	Sans effet.	Les utilisateurs qui ont accès en écriture au répertoire ne peuvent supprimer que les fichiers qu'ils possèdent. Ils ne peuvent pas supprimer ni forcer l'enregistrement sur ceux qui appartiennent à d'autres utilisateurs.

La permission *setuid* d'un fichier exécutable indique que les commandes seront exécutées au nom de l'utilisateur propriétaire du fichier, et non pas au nom de l'utilisateur qui a exécuté la commande. La commande **passwd** est un bon exemple :

```
[user@host ~]$ ls -l /usr/bin/passwd
-rwsr-xr-x. 1 root root 35504 Jul 16 2010 /usr/bin/passwd
```

CHAPITRE 7 | Contrôle de l'accès aux fichiers

Dans une liste longue, vous pouvez repérer les permissions setuid au **s** minuscule présent à l'endroit où se trouve normalement le **x** (permissions d'exécution pour le propriétaire). Si le propriétaire ne dispose pas des permissions d'exécution, un **S** majuscule est utilisé à la place.

La permission spéciale *setgid* sur un répertoire indique que les fichiers créés dans ce répertoire héritent de la propriété de groupe de ce répertoire, et non de l'utilisateur qui les a créés. Cette permission sert généralement pour les répertoires de collaboration de groupe, afin d'attribuer automatiquement un fichier au groupe partagé, au lieu du groupe privé par défaut, ou dans le cas où les fichiers d'un répertoire doivent toujours appartenir à un groupe spécifique. Un exemple de ceci est le répertoire **/run/log/journal** :

```
[user@host ~]$ ls -ld /run/log/journal
drwxr-sr-x. 3 root systemd-journal 60 May 18 09:15 /run/log/journal
```

Si *setgid* est défini sur un fichier exécutable, les commandes sont exécutées au nom du groupe propriétaire de ce fichier, et non au nom de l'utilisateur ayant exécuté la commande, un peu comme *setuid*. La commande **locate** est un bon exemple :

```
[user@host ~]$ ls -ld /usr/bin/locate
-rwx--s--x. 1 root slocate 47128 Aug 12 17:17 /usr/bin/locate
```

Dans une liste longue, vous pouvez repérer les permissions *setgid* au **s** minuscule présent à l'emplacement normal du **x** (permissions d'exécution pour le groupe). Si le groupe ne dispose pas des permissions d'exécution, un **S** majuscule est utilisé à la place.

Enfin, le *sticky bit* pour un répertoire définit une restriction spéciale sur la suppression de fichiers. Seul le propriétaire du fichier (et root) peut supprimer des fichiers dans le répertoire. **/tmp** constitue un exemple :

```
[user@host ~]$ ls -ld /tmp
drwxrwxrwt. 39 root root 4096 Feb 8 20:52 /tmp
```

Dans une liste longue, vous pouvez identifier les permissions sticky au **t** minuscule présent à l'endroit où se trouve normalement le **x** (permissions d'exécution pour les autres). Si « other » n'a pas de permission d'exécution, c'est alors la lettre **T** majuscule qui est affichée.

Définition des permissions spéciales

- Symboliquement : *setuid* = **u+s**; *setgid* = **g+s**; *sticky* = **o+t**
- Méthode numérique (quatrième chiffre en préfixe) : *setuid* = 4 ; *setgid* = 2 ; *sticky* = 1

Exemples

- Ajoutez le bit *setgid* à **directory** :

```
[user@host ~]# chmod g+s directory
```

- Définissez le bit *setgid* et ajoutez les permissions en lecture/écriture/exécution pour l'utilisateur et le groupe, sans accès pour les autres, sur **directory** :

```
[user@host ~]# chmod 2770 directory
```

PERMISSIONS DE FICHIERS PAR DÉFAUT

Lorsque vous créez un nouveau fichier ou répertoire, des permissions initiales lui sont attribuées. Deux choses affectent ces permissions initiales. La première est de savoir si vous créez un fichier normal ou un répertoire. La seconde est le *umask* courant.

Si vous créez un nouveau répertoire, le système d'exploitation commence par lui attribuer des autorisations octales 0777 (**drwxrwxrwx**). Si vous créez un nouveau fichier standard, le système d'exploitation lui attribue des autorisations octales 0666 (**-rw-rw-rw-**). Vous devez toujours explicitement ajouter l'autorisation d'exécution à un fichier normal. Cela rend plus difficile pour un attaquant de compromettre un service réseau afin qu'il crée un nouveau fichier et l'exécute immédiatement en tant que programme.

Cependant, la session shell définira également un umask pour restreindre davantage les autorisations initialement définies. Il s'agit d'un masque de bits converti en octal qui sert à bloquer les permissions des nouveaux fichiers et répertoires créés par un processus. Si un bit est défini dans l'umask, il bloque la permission correspondante pour les nouveaux fichiers. Par exemple, l'umask précédent 0002 bloque le bit d'écriture des autres utilisateurs. Les zéros en préfixe indiquent que les permissions spéciales, d'utilisateur et de groupe ne sont pas bloquées. Un umask égal à 0077 bloque toutes les permissions pour le groupe et pour les autres, pour tous les fichiers nouvellement créés.

La commande **umask** sans argument affiche la valeur actuelle de l'umask du shell :

```
[user@host ~]$ umask  
0002
```

Utilisez la commande **umask** avec un seul argument numérique pour modifier l'umask du shell courant. L'argument numérique doit être une valeur octale qui correspond à la nouvelle valeur de l'umask. Vous pouvez omettre les zéros en préfixe dans l'umask.

Les valeurs umask par défaut du système pour les utilisateurs du shell bash sont définies dans les fichiers **/etc/profile** et **/etc/bashrc**. Les utilisateurs peuvent outrepasser les paramètres par défaut du système dans les fichiers **.bash_profile** et **.bashrc** de leurs répertoires personnels.

Exemple d'umask

L'exemple suivant explique comment umask affecte les permissions des fichiers et des répertoires. Examinez les permissions umask par défaut pour les fichiers et les répertoires du shell actuel. Le propriétaire et le groupe ont tous deux la permission de lecture et d'écriture sur les fichiers et « other » est configuré pour la lecture. Le propriétaire et le groupe possèdent des permissions de lecture, d'écriture et d'exécution sur les répertoires. La seule permission pour « other » est la lecture.

```
[user@host ~]$ umask  
0002  
[user@host ~]$ touch default  
[user@host ~]$ ls -l default.txt  
-rw-rw-r--. 1 user user 0 May  9 01:54 default.txt  
[user@host ~]$ mkdir default  
[user@host ~]$ ls -ld default  
drwxrwxr-x. 2 user user 0 May  9 01:54 default
```

CHAPITRE 7 | Contrôle de l'accès aux fichiers

En définissant la valeur umask sur 0, les permissions sur les fichiers pour « other » passent de lecture à lecture et écriture. Les permissions sur les répertoires pour « other » passent de lecture et exécution à lecture, écriture et exécution.

```
[user@host ~]$ umask 0
[user@host ~]$ touch zero.txt
[user@host ~]$ ls -l zero.txt
-rw-rw-rw-. 1 user user 0 May  9 01:54 zero.txt
[user@host ~]$ mkdir zero
[user@host ~]$ ls -ld zero
drwxrwxrwx. 2 user user 0 May  9 01:54 zero
```

Pour masquer toutes les permissions sur les fichiers et les répertoires pour « other », définissez la valeur umask sur 007.

```
[user@host ~]$ umask 007
[user@host ~]$ touch seven.txt
[user@host ~]$ ls -l seven.txt
-rw-rw----. 1 user user 0 May  9 01:55 seven.txt
[user@host ~]$ mkdir seven
[user@host ~]$ ls -ld seven
drwxrwx---. 2 user user 0 May  9 01:54 seven
```

Un umask de 027 garantit que les nouveaux fichiers disposent des permissions de lecture et d'écriture pour l'utilisateur et de lecture pour le groupe. Les nouveaux répertoires ont un accès en lecture et en écriture pour le groupe et aucune autorisation pour les autres.

```
[user@host ~]$ umask 027
[user@host ~]$ touch two-seven.txt
[user@host ~]$ ls -l two-seven.txt
-rw-r-----. 1 user user 0 May  9 01:55 two-seven.txt
[user@host ~]$ mkdir two-seven
[user@host ~]$ ls -ld two-seven
drwxr-x---. 2 user user 0 May  9 01:54 two-seven
```

Le umask par défaut pour les utilisateurs est défini par les scripts de démarrage du shell. Par défaut, si l'UID de votre compte est égal à 200 ou plus et si votre nom d'utilisateur et votre nom de groupe principal sont identiques, un umask de 002 vous sera attribué. Sinon, votre umask sera 022.

En tant que root, vous pouvez changer cela en ajoutant un script de démarrage de shell nommé **/etc/profile.d/local-umask.sh** qui ressemble à la sortie dans cet exemple :

```
[root@host ~]# cat /etc/profile.d/local-umask.sh
# Overrides default umask configuration
if [ $UID -gt 199 ] && [ "`id -gn` = `id -un`" ]; then
    umask 007
else
    umask 022
fi
```

L'exemple précédent définira l'umask sur 007 pour les utilisateurs avec un UID supérieur à 199 et avec un nom d'utilisateur et un nom de groupe principal correspondants, et sur 022 pour tout le monde. Si vous souhaitez simplement définir l'umask pour tout le monde sur 022, vous pouvez créer ce fichier avec uniquement le contenu suivant :

```
# Overrides default umask configuration  
umask 022
```

Pour vous assurer que les modifications globales de l'umask prennent effet, vous devez vous déconnecter du shell et vous y reconnecter. Jusqu'à ce moment, l'umask configuré dans le shell actuel est toujours actif.



RÉFÉRENCES

Pages du manuel **bash(1)**, **ls(1)**, **chmod(1)** et **umask(1)**

► EXERCICE GUIDÉ

GESTION DE L'ACCÈS AUX FICHIERS ET DES PERMISSIONS PAR DÉFAUT

Dans cet exercice, vous contrôlerez les permissions sur les fichiers créés dans un répertoire à l'aide des paramètres umask et de la permission setgid.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Créer un répertoire partagé dans lequel les nouveaux fichiers appartiennent automatiquement au groupe operators.
- Faire des essais avec différents paramètres umask.
- Ajuster les permissions par défaut pour des utilisateurs spécifiques.
- Vérifier que vos ajustements sont corrects.

AVANT DE COMMENCER

Connectez-vous à workstation en tant qu'utilisateur student avec le mot de passe student.

Sur workstation, exécutez la commande **lab perms-default start**. La commande exécute un script de démarrage qui détermine si servera est accessible sur le réseau. Le script crée également le groupe operators et l'utilisateur operator1 sur servera.

```
[student@workstation ~]$ lab perms-default start
```

- 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Utilisez la commande **su** pour basculer vers l'utilisateur operator1 en utilisant redhat comme mot de passe.

```
[student@servera ~]$ su - operator1
Password: redhat
[operator1@servera ~]$
```

- 3. Utilisez la commande **umask** pour lister la valeur umask par défaut de l'utilisateur operator1.

```
[operator1@servera ~]$ umask  
0002
```

- ▶ 4. Créez un répertoire nommé **/tmp/shared**. Dans le répertoire **/tmp/shared**, créez un fichier appelé **defaults**. Examinez les permissions par défaut.

- 4.1. Utilisez la commande **mkdir** pour créer le répertoire **/tmp/shared**. Utilisez la commande **ls -ld** pour lister les permissions du nouveau répertoire.

```
[operator1@servera ~]$ mkdir /tmp/shared  
[operator1@servera ~]$ ls -ld /tmp/shared  
drwxrwxr-x. 2 operator1 operator1 6 Feb 4 14:06 /tmp/shared
```

- 4.2. Utilisez la commande **touch** pour créer un fichier nommé **defaults** dans le répertoire **/tmp/shared**.

```
[operator1@servera ~]$ touch /tmp/shared/defaults
```

- 4.3. Utilisez la commande **ls -l** pour lister les permissions du nouveau fichier.

```
[operator1@servera ~]$ ls -l /tmp/shared/defaults  
-rw-rw-r--. 1 operator1 operator1 0 Feb 4 14:09 /tmp/shared/defaults
```

- ▶ 5. Modifiez la propriété du groupe de **/tmp/shared** en **operators**. Confirmez la nouvelle propriété et les nouvelles permissions.

- 5.1. Utilisez la commande **chown** pour transférer la propriété du groupe du répertoire **/tmp/shared** au groupe **operators**.

```
[operator1@servera ~]$ chown :operators /tmp/shared
```

- 5.2. Utilisez la commande **ls -ld** pour lister les permissions du nouveau répertoire **/tmp/shared**.

```
[operator1@servera ~]$ ls -ld /tmp/shared  
drwxrwxr-x. 2 operator1 operators 22 Feb 4 14:09 /tmp/shared
```

- 5.3. Utilisez la commande **touch** pour créer un fichier nommé **group** dans le répertoire **/tmp/shared**. Utilisez la commande **ls -l** pour lister les permissions du fichier.

```
[operator1@servera ~]$ touch /tmp/shared/group  
[operator1@servera ~]$ ls -l /tmp/shared/group  
-rw-rw-r--. 1 operator1 operator1 0 Feb 4 17:00 /tmp/shared/group
```



NOTE

Le groupe propriétaire du fichier **/tmp/shared/group** n'est pas **operators** mais **operator1**.

CHAPITRE 7 | Contrôle de l'accès aux fichiers

- 6. Assurez-vous que les fichiers créés dans le répertoire **/tmp/shared** appartiennent au groupe operators.

- 6.1. Utilisez la commande **chmod** pour définir l'ID de groupe sur operators pour le répertoire **/tmp/shared**.

```
[operator1@servera ~]$ chmod g+s /tmp/shared
```

- 6.2. Utilisez la commande **touch** pour créer un fichier nommé **operations_database.txt** dans le répertoire **/tmp/shared**.

```
[operator1@servera ~]$ touch /tmp/shared/operations_database.txt
```

- 6.3. Utilisez la commande **ls -l** pour vérifier que le groupe operators est le propriétaire du groupe pour le nouveau fichier.

```
[operator1@servera ~]$ ls -l /tmp/shared/operations_database.txt
-rw-rw-r--. 1 operator1 operators 0 Feb  4 16:11 /tmp/shared/
operations_database.txt
```

- 7. Créez un fichier nommé **operations_network.txt** dans le répertoire **/tmp/shared**.

Enregistrez la propriété et les permissions. Changer l'umask par operator1. Créez un fichier nommé **operations_production.txt**. Enregistrez la propriété et les permissions du fichier **operations_production.txt**.

- 7.1. Utilisez la commande **echo** pour créer un fichier nommé **operations_network.txt** dans le répertoire **/tmp/shared**.

```
[operator1@servera ~]$ echo text >> /tmp/shared/operations_network.txt
```

- 7.2. Utilisez la commande **ls -l** pour lister les permissions du fichier **operations_network.txt**.

```
[operator1@servera ~]$ ls -l /tmp/shared/operations_network.txt
-rw-rw-r--. 1 operator1 operators 5 Feb  4 15:43 /tmp/shared/
operations_network.txt
```

- 7.3. Utilisez la commande **umask** pour modifier l'umask de l'utilisateur operator1 en 027. Utilisez la commande **umask** pour confirmer la modification.

```
[operator1@servera ~]$ umask 027
[operator1@servera ~]$ umask
0027
```

- 7.4. Utilisez la commande **touch** pour créer un fichier nommé **operations_production.txt** dans le répertoire **/tmp/shared/**. Utilisez la commande **ls -l** pour vous assurer que les fichiers sont créés avec un accès en lecture seule pour le groupe operators et sans aucun accès pour les autres utilisateurs.

```
[operator1@servera ~]$ touch /tmp/shared/operations_production.txt  
[operator1@servera ~]$ ls -l /tmp/shared/operations_production.txt  
-rw-r----- 1 operator1 operators 0 Feb 4 15:56 /tmp/shared/  
operations_production.txt
```

- 8. Ouvrez une nouvelle fenêtre de terminal et connectez-vous à servera en tant qu'operator1.

```
[student@workstation ~]$ ssh operator1@servera  
...output omitted...  
[operator1@servera ~]$
```

- 9. Listez la valeur umask pour operator1.

```
[operator1@servera ~]$ umask  
0002
```

- 10. Changez l'umask par défaut pour l'utilisateur operator1. Le nouvel umask empêche tout accès aux utilisateurs qui n'appartiennent pas à son groupe. Confirmez que l'umask a été modifié.

10.1. Utilisez la commande **echo** pour modifier l'umask de l'utilisateur operator1 en 007.

```
[operator1@servera ~]$ echo "umask 007" >> ~/.bashrc  
[operator1@servera ~]$ cat ~/.bashrc  
# .bashrc  
  
# Source global definitions  
if [ -f /etc/bashrc ]; then  
  . /etc/bashrc  
fi  
  
# Uncomment the following line if you don't like systemctl's auto-paging feature:  
# export SYSTEMD_PAGER=  
  
# User specific aliases and functions  
umask 007
```

10.2. Déconnectez-vous et reconnectez-vous en tant qu'utilisateur operator1. Utilisez la commande **umask** pour confirmer que la modification est permanente.

```
[operator1@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$ ssh operator1@servera  
...output omitted...  
[operator1@servera ~]$ umask  
0007
```

- 11. Sur servera, quittez tous les shells utilisateur operator1 et student.



MISE EN GARDE

Quittez tous les shells ouverts par **operator1**. Si vous ne quittez pas tous les shells, le script de fin échoue.

```
[operator1@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Fin

Sur workstation, exécutez le script **lab perms-default finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab perms-default finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

CONTRÔLE DE L'ACCÈS AUX FICHIERS

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez configurer les permissions sur les fichiers et configurer un répertoire que les utilisateurs d'un groupe particulier peuvent utiliser pour partager facilement des fichiers sur le système de fichiers local.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Créer un répertoire dans lequel les utilisateurs peuvent travailler en collaboration sur des fichiers.
- Créer des fichiers auxquels la propriété du groupe est automatiquement attribuée.
- Créer des fichiers qui ne sont pas accessibles en dehors du groupe.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez la commande `lab perms-review start`. La commande exécute un script de démarrage qui détermine si `serverb` est accessible sur le réseau. Le script crée également le groupe `techdocs` et trois utilisateurs nommés `tech1`, `tech2` et `database1`.

```
[student@workstation ~]$ lab perms-review start
```

1. Utilisez la commande `ssh` pour vous connecter à `serverb` en tant qu'utilisateur `student`. Basculez vers l'utilisateur `root` sur `serverb` en utilisant `redhat` comme mot de passe.
2. Créez un répertoire nommé **/home/techdocs**.
3. Transférez la propriété du groupe du répertoire **/home/techdocs** au groupe `techdocs`.
4. Vérifiez que les utilisateurs du groupe `techdocs` peuvent créer et modifier des fichiers dans le répertoire `/home/techdocs`.
5. Définissez les permissions sur le répertoire **/home/techdocs**. Sur le répertoire **/home/techdocs**, configurez `setgid` (2), les permissions en lecture/écriture/exécution (7) pour le propriétaire/l'utilisateur et le groupe, et aucune permission (0) pour les autres utilisateurs.
6. Vérifiez que les permissions sont correctement définies.
7. Confirmez que les utilisateurs du groupe `techdocs` peuvent désormais créer et modifier des fichiers dans le répertoire `/home/techdocs`. Les utilisateurs qui ne sont pas membres du groupe `techdocs` ne peuvent pas modifier ni créer des fichiers dans le répertoire `/home/techdocs`. Les utilisateurs `tech1` et `tech2` sont membres du groupe `techdocs`. Les utilisateurs `database1` ne font pas partie du groupe.

8. Modifiez les scripts de connexion globaux. Le paramétrage de l'umask des utilisateurs normaux doit empêcher les autres utilisateurs d'afficher ou de modifier le contenu des nouveaux fichiers et répertoires.
9. Déconnectez-vous de serverb.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.
```

Évaluation

À partir de workstation, exécutez le script **lab perms-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab perms-review grade
```

Fin

Sur workstation, exécutez le script **lab perms-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab perms-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

CONTRÔLE DE L'ACCÈS AUX FICHIERS

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez configurer les permissions sur les fichiers et configurer un répertoire que les utilisateurs d'un groupe particulier peuvent utiliser pour partager facilement des fichiers sur le système de fichiers local.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Créer un répertoire dans lequel les utilisateurs peuvent travailler en collaboration sur des fichiers.
- Créer des fichiers auxquels la propriété du groupe est automatiquement attribuée.
- Créer des fichiers qui ne sont pas accessibles en dehors du groupe.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab perms-review start**. La commande exécute un script de démarrage qui détermine si **serverb** est accessible sur le réseau. Le script crée également le groupe **techdocs** et trois utilisateurs nommés **tech1**, **tech2** et **database1**.

```
[student@workstation ~]$ lab perms-review start
```

1. Utilisez la commande **ssh** pour vous connecter à **serverb** en tant qu'utilisateur **student**. Basculez vers l'utilisateur **root** sur **serverb** en utilisant **redhat** comme mot de passe.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$ su -
Password: redhat
[root@serverb ~]#
```

2. Créez un répertoire nommé **/home/techdocs**.
 - 2.1. Utilisez la commande **mkdir** pour créer un répertoire appelé **/home/techdocs**.

```
[root@serverb ~]# mkdir /home/techdocs
```

3. Transférez la propriété du groupe du répertoire **/home/techdocs** au groupe **techdocs**.

CHAPITRE 7 | Contrôle de l'accès aux fichiers

- 3.1. Utilisez la commande **chown** pour transférer la propriété du groupe du répertoire **/home/techdocs** au groupe **techdocs**.

```
[root@serverb ~]# chown :techdocs /home/techdocs
```

4. Vérifiez que les utilisateurs du groupe **techdocs** peuvent créer et modifier des fichiers dans le répertoire **/home/techdocs**.

- 4.1. Utilisez la commande **su** pour basculer vers l'utilisateur **tech1**.

```
[root@serverb ~]# su - tech1  
[tech1@serverb ~]$
```

- 4.2. Utilisez la commande **touch** pour créer un fichier nommé **techdoc1.txt** dans le répertoire **/home/techdocs**.

```
[tech1@serverb ~]$ touch /home/techdocs/techdoc1.txt  
touch: cannot touch '/home/techdocs/techdoc1.txt': Permission denied
```

**NOTE**

Notez que même si le répertoire **/home/techdocs** appartient à **techdocs** et que **tech1** fait partie du groupe **techdocs**, il n'est pas possible de créer un fichier dans ce répertoire. C'est parce que le groupe **techdocs** n'a pas de permission d'écriture. Utilisez la commande **ls -ld** pour afficher les permissions.

```
[tech1@serverb ~]$ ls -ld /home/techdocs/  
drwxr-xr-x. 2 root techdocs 6 Feb 5 16:05 /home/techdocs/
```

5. Définissez les permissions sur le répertoire **/home/techdocs**. Sur le répertoire **/home/techdocs**, configurez setgid (2), les permissions en lecture/écriture/exécution (7) pour le propriétaire/l'utilisateur et le groupe, et aucune permission (0) pour les autres utilisateurs.

- 5.1. Quittez le shell de l'utilisateur **tech1**.

```
[tech1@serverb ~]$ exit  
logout  
[root@serverb ~]#
```

- 5.2. Utilisez la commande **chmod** pour définir les permissions de groupe pour le répertoire **/home/techdocs**. Sur le répertoire **/home/techdocs**, configurez setgid (2), les permissions en lecture/écriture/exécution (7) pour le propriétaire/l'utilisateur et le groupe, et aucune permission (0) pour les autres utilisateurs.

```
[root@serverb ~]# chmod 2770 /home/techdocs
```

6. Vérifiez que les permissions sont correctement définies.

```
[root@serverb ~]# ls -ld /home/techdocs  
drwxrws---. 2 root techdocs 6 Feb 4 18:12 /home/techdocs/
```

Notez que le groupe techdocs a maintenant une permission d'écriture.

7. Confirmez que les utilisateurs du groupe techdocs peuvent désormais créer et modifier des fichiers dans le répertoire /home/techdocs. Les utilisateurs qui ne sont pas membres du groupe techdocs ne peuvent pas modifier ni créer des fichiers dans le répertoire /home/techdocs. Les utilisateurs tech1 et tech2 sont membres du groupe techdocs. Les utilisateurs database1 ne font pas partie du groupe.

- 7.1. Basculez vers l'utilisateur tech1. Utilisez la commande **touch** pour créer un fichier nommé **techdoc1.txt** dans le répertoire /home/techdocs. Quittez le shell de l'utilisateur tech1.

```
[root@serverb ~]# su - tech1
[tech1@serverb ~]$ touch /home/techdocs/techdoc1.txt
[tech1@serverb ~]$ ls -l /home/techdocs/techdoc1.txt
-rw-rw-r-- 1 tech1 techdocs 0 Feb  5 16:42 /home/techdocs/techdoc1.txt
[tech1@serverb ~]$ exit
logout
[root@serverb ~]#
```

- 7.2. Basculez vers l'utilisateur tech2. Utilisez la commande **echo** pour ajouter du contenu au fichier **/home/techdocs/techdoc1.txt**. Quittez le shell de l'utilisateur tech2.

```
[root@serverb ~]# su - tech2
[tech2@serverb ~]$ cd /home/techdocs
[tech2@serverb techdocs]$ echo "This is the first tech doc." > techdoc1.txt
[tech2@serverb techdocs]$ exit
logout
[root@serverb ~]#
```

- 7.3. Basculez vers l'utilisateur database1. Utilisez la commande **echo** pour ajouter du contenu au fichier **/home/techdocs/techdoc1.txt**. Notez que vous recevrez le message **Permission denied**. Utilisez la commande **ls -l** pour confirmer que database1 n'a pas accès au fichier. Quittez le shell de l'utilisateur database1.

```
[root@serverb ~]# su - database1
[database1@serverb ~]$ echo "This is the first tech doc." \
>> /home/techdocs/techdoc1.txt
-bash: /home/techdocs/techdoc1.txt: Permission denied
[database1@serverb ~]$ ls -l /home/techdocs/techdoc1.txt
ls: cannot access '/home/techdocs/techdoc1.txt': Permission denied
[database1@serverb ~]$ exit
logout
[root@serverb ~]#
```

8. Modifiez les scripts de connexion globaux. Le paramétrage de l'umask des utilisateurs normaux doit empêcher les autres utilisateurs d'afficher ou de modifier le contenu des nouveaux fichiers et répertoires.
- 8.1. Déterminer l'umask de l'utilisateur student. Utilisez la commande **su - student** pour basculer vers le shell de connexion student. Lorsque vous avez terminé, quittez le shell.

```
[root@serverb ~]# su - student
[student@serverb ~]$ umask
0002
[student@serverb ~]$ exit
logout
[root@serverb ~]#
```

- 8.2. Créez le fichier **/etc/profile.d/local-umask.sh** avec le contenu suivant pour définir l'umask sur **007** pour les utilisateurs avec un UID supérieur à **199** et avec un nom d'utilisateur et un nom de groupe principal correspondants, et sur **022** pour tout le monde :

```
[root@serverb ~]# cat /etc/profile.d/local-umask.sh
# Overrides default umask configuration
if [ $UID -gt 199 ] && [ "`id -gn`" = "`id -un`" ]; then
    umask 007
else
    umask 022
fi
```

- 8.3. Déconnectez-vous du shell et reconnectez-vous en tant que student pour vérifier que l'umask global est désormais défini sur **007**.

```
[root@serverb ~]# exit
logout
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$ umask
0007
```

9. Déconnectez-vous de serverb.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
```

Évaluation

À partir de workstation, exéutez le script **lab perms-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab perms-review grade
```

Fin

Sur workstation, exéutez le script **lab perms-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab perms-review finish
```

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Les fichiers sont associés à trois catégories auxquelles s'appliquent des permissions. Un fichier appartient à un utilisateur, à un seul groupe et à d'autres utilisateurs. La permission la plus spécifique s'applique. Les permissions de l'utilisateur prévalent sur les permissions du groupe, lesquelles priment sur les permissions des autres.
- L'option **-l** de la commande **ls** développe la liste des fichiers pour inclure à la fois les permissions d'un fichier et les droits de propriété.
- La commande **chmod** modifie les permissions de fichier à partir de la ligne de commande. Il existe deux méthodes pour représenter les permissions, symbolique (lettres) et numérique (chiffres).
- La commande **chown** change la propriété du fichier. L'option **-R** modifie de manière récursive la propriété d'une arborescence de répertoires.
- La commande **umask** sans argument affiche la valeur actuelle de l'umask du shell. Chaque processus sur le système a un umask. Les valeurs umask par défaut pour bash sont définies dans les fichiers **/etc/profile** et **/etc/bashrc**.

CHAPITRE 8

CONTRÔLE ET GESTION DES PROCESSUS LINUX

PROJET

Évaluer et contrôler les processus exécutés sur un système Red Hat Enterprise Linux.

OBJECTIFS

- Obtenir des informations sur les programmes en cours d'exécution sur le système afin de pouvoir déterminer leur statut, l'utilisation des ressources et leur propriété pour les contrôler.
- Utiliser le contrôle de tâche Bash pour gérer plusieurs processus démarrés à partir de la même session de terminal.
- Contrôler et terminer les processus qui ne sont pas associés à votre shell et forcer la fin des processus et sessions utilisateur.
- Décrire la charge moyenne et déterminer les processus responsables d'une utilisation intensive des ressources sur un serveur.

SECTIONS

- Création d'une liste de processus (avec quiz)
- Contrôle des tâches (et exercice guidé)
- Suppression de processus (et exercice guidé)
- Contrôle de l'activité des processus (avec exercice pratique)

ATELIER

Contrôle et gestion des processus Linux

CRÉATION D'UNE LISTE DE PROCESSUS

OBJECTIFS

Au terme de cette section, vous devez pouvoir obtenir des informations sur les programmes en cours d'exécution sur un système afin de pouvoir déterminer leur statut, l'utilisation des ressources et leur propriété, de manière à pouvoir les contrôler.

DÉFINITION D'UN PROCESSUS

Un *processus* est une instance en cours d'exécution d'un programme exécutable. Un processus consiste en :

- un espace d'adressage de mémoire allouée ;
- des propriétés de sécurité, dont les informations de propriété et les priviléges ;
- un ou plusieurs fils (threads) d'exécution de code de programmes ;
- l'état du processus.

L'*environnement* d'un processus comprend :

- des variables locales et globales ;
- un contexte de planification actif ;
- des ressources système allouées, telles que des descripteurs de fichier et des ports réseau.

Un processus (*parent*) existant réplique son propre espace d'adressage (**fork**) pour créer une nouvelle structure de processus (*enfant*). Un identifiant de *processus unique* (PID) est affecté à chaque nouveau processus pour le suivi et la sécurité. Le PID et l'*identifiant du processus parent* (PPID) sont des éléments de l'environnement du nouveau processus. Tout processus peut créer un processus enfant. Tous les processus sont des descendants du processus système initial, **systemd** sur un système Red Hat Enterprise Linux 8).

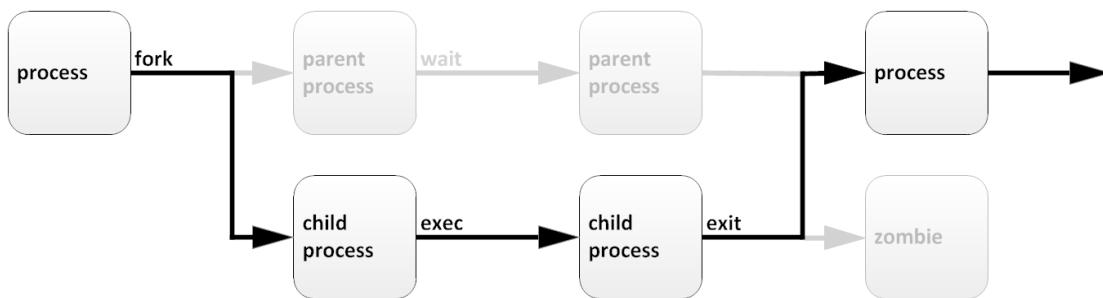


Figure 8.1: Cycle de vie d'un processus

Through the *fork* routine, a child process inherits security identities, previous and current file descriptors, port and resource privileges, environment variables, and program code. Un processus enfant peut ensuite exécuter son propre code de programme. Généralement, un processus père dort (*sleep*) pendant l'exécution du processus fils, et émet une requête (*wait*) pour être averti lorsque le processus fils est terminé. Lorsque le processus enfant s'arrête, il a déjà fermé ou éliminé ses ressources et son environnement. La seule ressource restante, appelée un *zombie*, est

une entrée dans la table de processus. Le parent, signalé comme réveillé à la sortie de l'enfant, supprime l'entrée de l'enfant dans la table de processus, libérant ainsi la dernière ressource du processus enfant. Le processus parent poursuit ensuite l'exécution de son propre code de programme.

DESCRIPTION DES ÉTATS DE PROCESSUS

Dans un système d'exploitation multitâches, chaque processeur (ou cœur de processeur) peut travailler sur un processus à un instant donné. Pendant l'exécution d'un processus, ses besoins immédiats en temps processeur et en ressources évoluent. Un état, qui évolue en fonction des circonstances, est affecté à chaque processus.

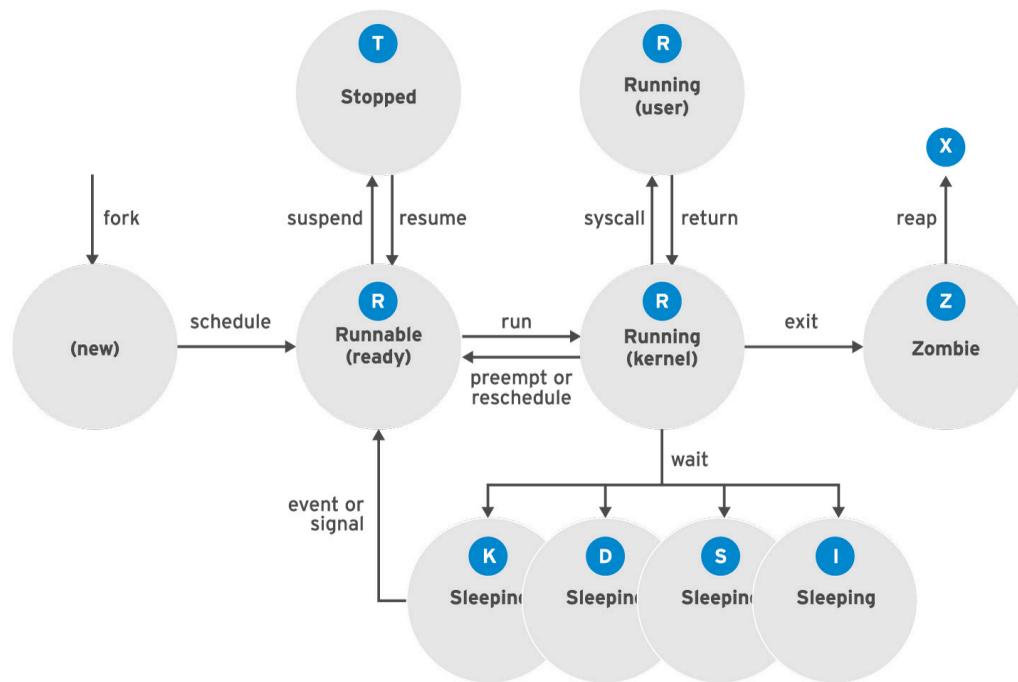


Figure 8.2: États d'un processus Linux

Les états des processus Linux sont illustrés dans le diagramme précédent et décrits dans le tableau suivant :

États d'un processus Linux

NOM	INDICATEUR	NOM ET DESCRIPTION DE L'ÉTAT DÉFINI PAR LE NOYAU
Running	R	TASK_RUNNING : le processus est soit en cours d'exécution sur un processeur, soit en attente d'exécution. Dans l'état <i>Running</i> (ou <i>Runnable</i>), un processus peut être en train d'exécuter les routines d'un utilisateur ou celles du noyau (appels système), ou être en attente et prêt à être exécuté.

NOM	INDICATEUR	NOM ET DESCRIPTION DE L'ÉTAT DÉFINI PAR LE NOYAU
Sleeping	S	TASK_INTERRUPTIBLE : le processus est en train d'attendre qu'une condition soit satisfaite ; une requête de matériel, un accès aux ressources du système ou un signal. Lorsqu'un événement ou un signal satisfait cette condition, le processus revient à <i>Running</i> .
	D	TASK_UNINTERRUPTIBLE : ce processus est également <i>Sleeping</i> mais, contrairement à l'état S, il ne répond pas aux signaux. Cet état ne sert que quand l'interruption d'un processus peut mettre un périphérique dans un état imprévisible.
	K	TASK_KILLABLE : identique à l'état D « uninterruptible », mais modifié pour permettre à une tâche en attente de répondre au signal à éliminer (« killed »). Les utilitaires affichent souvent des processus pouvant être éliminés (« Killable ») comme étant dans l'état D.
	I	TASK_REPORT_IDLE : sous-ensemble de l'état D. Le noyau ne compte pas ces processus lors du calcul de la charge moyenne. Utilisé pour les threads du noyau. Les indicateurs TASK_UNINTERRUPTABLE et TASK_NOLOAD sont définis. Semblable à TASK_KILLABLE, aussi un sous-ensemble de l'état D. Il accepte les signaux fatals.
Stopped	T	TASK_STOPPED : le processus a été arrêté (« Stopped ») (suspendu), généralement suite au signal d'un utilisateur ou d'un autre processus. Le processus peut poursuivre son exécution (reprendre) sur réception d'un autre signal, pour revenir à l'état <i>Running</i> .
	T	TASK_TRACED : un processus en cours de débogage est momentanément arrêté (« Stopped ») et possède le même indicateur d'état T.
Zombie	Z	EXIT_ZOMBIE : un processus fils envoie un signal à son père lorsqu'il s'arrête. Toutes les ressources sont libérées, à l'exception de l'identité du processus (PID).
	X	EXIT_DEAD : lorsque le parent nettoie (élimine) la structure restante du processus fils, le processus est à présent complètement libéré. Cet état ne sera jamais indiqué par les utilitaires d'énumération des processus.

Pourquoi les états de processus sont importants

Lors du dépannage d'un système, il est important de comprendre comment le noyau communique avec les processus et comment les processus communiquent entre eux. Lors de la création du processus, le système attribue un état au processus. La colonne **S** de la commande **top** ou la colonne **STAT** de **ps** indique l'état de chaque processus. Sur un système à processeur unique, un seul processus peut être exécuté à la fois. Il est possible d'afficher plusieurs processus avec l'état **R**. Cependant, ils ne fonctionneront pas tous consécutivement, certains auront le statut *waiting*.

```
[user@host ~]$ top
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
 1 root 20 0 244344 13684 9024 S 0.0 0.7 0:02.46 systemd
 2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
...output omitted...
```

```
[user@host ~]$ ps aux
USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND
...output omitted...
root 2 0.0 0.0 0 0 ? S 11:57 0:00 [kthreadd]
student 3448 0.0 0.2 266904 3836 pts/0 R+ 18:07 0:00 ps aux
...output omitted...
```

Le processus peut être suspendu, arrêté, repris, arrêté et interrompu à l'aide de signaux. Les signaux sont abordés plus en détail ultérieurement dans ce chapitre. Les signaux peuvent être utilisés par d'autres processus, par le noyau lui-même ou par les utilisateurs connectés au système.

CRÉATION D'UNE LISTE DE PROCESSUS

La commande **ps** sert à dresser la liste des processus actifs. Elle peut fournir des informations détaillées sur les processus, telles que :

- le numéro d'identification de l'utilisateur (UID), qui détermine les priviléges des processus ;
- le numéro d'identification unique du processus (PID) ;
- le temps processeur et le temps réel déjà utilisés ;
- la quantité de mémoire allouée par le processus à divers emplacements ;
- l'emplacement du processus **stdout**, appelé *terminal de contrôle* ;
- l'état du processus actif.



IMPORTANT

La version Linux de la commande **ps** prend en charge trois formats d'option :

- les options UNIX (POSIX), qui peuvent être regroupées et doivent être précédées d'un tiret ;
- les options BSD, qui peuvent être regroupées, mais ne doivent pas être utilisées avec un tiret ;
- les options longues GNU, qui sont précédées de deux tirets.

Par exemple, **ps -aux** est différent de **ps aux**.

Peut-être l'ensemble d'options le plus courant, **aux**, affiche tous les processus, y compris les processus sans terminal de contrôle. Une liste longue (options **lax**) fournit plus de détails techniques. Elle peut toutefois s'afficher plus rapidement en évitant les recherches de noms d'utilisateurs. La syntaxe UNIX équivalente utilise les options **-ef** pour afficher tous les processus.

```
[user@host ~]$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START  TIME COMMAND
root      1  0.1  0.1  51648  7504 ?          Ss   17:45  0:03 /usr/lib/systemd/
syst
root      2  0.0  0.0     0     0 ?          S    17:45  0:00 [kthreadd]
root      3  0.0  0.0     0     0 ?          S    17:45  0:00 [ksoftirqd/0]
root      5  0.0  0.0     0     0 ?          S<  17:45  0:00 [kworker/0:0H]
root      7  0.0  0.0     0     0 ?          S    17:45  0:00 [migration/0]
...output omitted...
[user@host ~]$ ps lax
F  UID  PID  PPID PRI  NI    VSZ   RSS WCHAN  STAT TTY      TIME COMMAND
4  0     1     0  20   0  51648  7504 ep_pol Ss   ?      0:03 /usr/lib/
systemd/
1  0     2     0  20   0     0   0 kthrea S    ?      0:00 [kthreadd]
1  0     3     2  20   0     0   0 smpboo S    ?      0:00 [ksoftirqd/0]
1  0     5     2  0 -20   0     0 worker  S<  ?      0:00 [kworker/0:0H]
1  0     7     2 -100  -    0     0 smpboo S    ?      0:00 [migration/0]
...output omitted...
[user@host ~]$ ps -ef
UID      PID  PPID C STIME TTY      TIME CMD
root      1     0  0 17:45 ?      00:00:03 /usr/lib/systemd/systemd --
switched-ro
root      2     0  0 17:45 ?      00:00:00 [kthreadd]
root      3     2  0 17:45 ?      00:00:00 [ksoftirqd/0]
root      5     2  0 17:45 ?      00:00:00 [kworker/0:0H]
root      7     2  0 17:45 ?      00:00:00 [migration/0]
...output omitted...
```

Par défaut, **ps** sans option sélectionne tous les processus avec le même *identifiant d'utilisateur effectif* (EUID) comme utilisateur actif, et qui sont associés au même terminal que celui depuis lequel **ps** a été appelé.

- Les processus entre crochets (généralement affichés en haut de la liste) sont des fils (threads) de noyau planifiés.
- Les zombies sont listés comme **exiting** ou **defunct**.
- La sortie de **ps** s'affiche une fois. Utilisez **top** pour un affichage de processus qui se met à jour dynamiquement.
- **ps** peut s'afficher sous forme d'arborescence afin de pouvoir visualiser les relations entre les processus parents et enfants.
- La sortie par défaut est triée par numéro d'identification de processus. À première vue, cela peut sembler être un ordre chronologique. Cependant, le noyau réutilise les ID de processus, de sorte que l'ordre est moins structuré qu'il n'y paraît. Pour trier, utilisez les options **-o** ou **--sort**. L'ordre d'affichage correspond à celui du tableau des processus système, qui réutilise les lignes du tableau à mesure que les processus meurent et que de nouveaux processus sont créés. La sortie peut sembler être en ordre chronologique, mais ce n'est pas garanti, à moins qu'on utilise les options **-o -** ou **--sort--**.



RÉFÉRENCES

info libc signal (*Manuel de référence de la bibliothèque C GNU*)

- Section 24 : Traitement des signaux

info libc processes (*Manuel de référence de la bibliothèque C GNU*)

- Section 26 : Processus

Pages de manuel **ps(1)** and **signal(7)**

► QUIZ

CRÉATION D'UNE LISTE DE PROCESSUS

Répondez aux questions suivantes en sélectionnant un ou plusieurs éléments :

► 1. Quel état représente un processus qui a été arrêté ou suspendu ?

- a. D
- b. R
- c. S
- d. T
- e. Z

► 2. Quel état représente un processus qui a libéré toutes ses ressources sauf son PID?

- a. D
- b. R
- c. S
- d. T
- e. Z

► 3. Quel processus un parent utilise-t-il pour dupliquer afin de créer un nouveau processus enfant ?

- a. exec
- b. fork
- c. zombie
- d. syscall
- e. reap

► 4. Quel état représente un processus en sommeil jusqu'à ce qu'une condition soit remplie ?

- a. D
- b. R
- c. S
- d. T
- e. Z

► SOLUTION

CRÉATION D'UNE LISTE DE PROCESSUS

Répondez aux questions suivantes en sélectionnant un ou plusieurs éléments :

- ▶ 1. Quel état représente un processus qui a été arrêté ou suspendu ?
 - a. D
 - b. R
 - c. S
 - d. T
 - e. Z

- ▶ 2. Quel état représente un processus qui a libéré toutes ses ressources sauf son PID?
 - a. D
 - b. R
 - c. S
 - d. T
 - e. Z

- ▶ 3. Quel processus un parent utilise-t-il pour dupliquer afin de créer un nouveau processus enfant ?
 - a. exec
 - b. fork
 - c. zombie
 - d. syscall
 - e. reap

- ▶ 4. Quel état représente un processus en sommeil jusqu'à ce qu'une condition soit remplie ?
 - a. D
 - b. R
 - c. S
 - d. T
 - e. Z

Contrôle des tâches

OBJECTIFS

Au terme de cette section, vous devez pouvoir utiliser le contrôle de tâche Bash pour gérer plusieurs processus démarrés à partir de la même session de terminal.

DESCRIPTION DES TRAVAUX ET DES SESSIONS

Le *contrôle des tâches* est une fonctionnalité du shell qui permet à une instance unique du shell d'exécuter et de gérer plusieurs commandes.

Une *tâche* est associée à chaque pipeline saisi à l'invite du shell. Tous les processus de ce pipeline font partie de la tâche et du même *groupe de processus*. Si une seule commande est saisie à l'invite du shell, elle peut être considérée comme le « pipeline » minimal d'une commande, en créant une tâche avec un seul membre.

Une seule tâche peut lire en même temps l'entrée et les signaux générés par le clavier depuis une fenêtre de terminal donnée. Les processus qui font partie de cette tâche sont les processus en *avant-plan* de ce *terminal de contrôle*.

Un processus en *arrière-plan* de ce terminal de contrôle est membre de n'importe quelle tâche associée à ce terminal. Les processus en arrière-plan d'un terminal ne peuvent ni lire ni recevoir les interruptions générées par le clavier du terminal, mais peuvent éventuellement écrire sur le terminal. Une tâche en arrière-plan peut être arrêtée (suspendue) ou en cours d'exécution. Si une tâche en arrière-plan en cours d'exécution tente de lire depuis le terminal, elle sera automatiquement suspendue.

Chaque terminal est sa propre *session* et peut disposer d'un processus en avant-plan et de n'importe quel nombre de processus en arrière-plan indépendants. Une tâche fait partie exactement d'une seule session : celle qui appartient à son terminal de contrôle.

La commande **ps** affiche le nom de périphérique du terminal de contrôle d'un processus dans la colonne **TTY**. Certains processus, tels que les *démons système*, sont initiés par le système, et non par une invite de shell. Ces processus ne sont pas associés à un terminal de contrôle, ne sont pas membres d'une tâche et ne peuvent pas être placés en avant-plan. La commande **ps** affiche un point d'interrogation (?) dans la colonne **TTY** pour ces processus.

EXÉCUTION DE TÂCHES EN ARRIÈRE-PLAN

Une commande ou un pipeline peut être démarré en arrière-plan en ajoutant une esperluette (&) à la fin de la ligne de commande. Le shell Bash affiche un *numéro de tâche* (unique pour la session) et le PID du nouveau processus enfant. Le shell n'attend pas la fin du processus enfant, mais affiche plutôt l'invite shell.

```
[user@host ~]$ sleep 10000 &
[1] 5947
[user@host ~]$
```

**NOTE**

Lorsqu'une ligne de commande contenant un pipe (barre verticale) est envoyée à l'arrière-plan à l'aide d'une esperluette, le PID de la dernière commande du pipeline est utilisé en sortie. Tous les processus du pipeline restent membres de cette tâche.

```
[user@host ~]$ example_command | sort | mail -s "Sort output" &
[1] 5998
```

Vous pouvez afficher la liste des tâches que Bash suit pour une session particulière avec la commande **jobs**.

```
[user@host ~]$ jobs
[1]+  Running                  sleep 10000 &
[user@host ~]$
```

Une tâche en arrière-plan peut être affichée au premier plan à l'aide de la commande **fg** avec son ID de tâche (%*numéro de tâche*).

```
[user@host ~]$ fg %1
sleep 10000
```

Dans l'exemple précédent, la commande **sleep** est exécutée en avant-plan sur le terminal de contrôle. Le shell lui-même repasse en état de veille et attend la fin de ce processus enfant.

Pour envoyer un processus en avant-plan vers l'arrière-plan, appuyez d'abord sur la requête *suspend* générée par le clavier (**Ctrl+z**) dans le terminal.

```
sleep 10000
^Z
[1]+  Stopped                  sleep 10000
[user@host ~]$
```

La tâche est immédiatement placée à l'arrière-plan et est suspendue.

La commande **ps j** affiche les informations relatives aux tâches. Le PID est l'*ID de processus* unique du processus. Le PPID est le PID du *processus parent* de ce processus, le processus qui l'a commencé (scindé). Le PGID est le PID du *leader du groupe de processus*, normalement le premier processus du pipeline de la tâche. Le SID est le PID du *leader de session* qui, pour une tâche, correspond normalement au shell interactif exécuté sur son terminal de contrôle. L'exemple de commande **sleep** étant actuellement suspendu, l'état de son processus est **T**.

```
[user@host ~]$ ps j
  PPID  PID  PGID   SID TTY      TPGID STAT   UID    TIME COMMAND
 2764  2768  2768  2768 pts/0      6377 Ss    1000   0:00 /bin/bash
 2768  5947  5947  2768 pts/0      6377 T     1000   0:00 sleep 10000
 2768  6377  6377  2768 pts/0      6377 R+    1000   0:00 ps j
```

Pour démarrer le processus suspendu exécuté en arrière-plan, utilisez la commande **bg** avec le même ID de tâche.

```
[user@host ~]$ bg %1  
[1]+ sleep 10000 &
```

Le shell informera l'utilisateur qui tente de quitter la fenêtre de terminal (session) que des tâches ont été suspendues. Si l'utilisateur tente à nouveau de quitter immédiatement la session, les tâches suspendues sont éliminées.



NOTE

Notez le signe **+** après le **[1]** dans les exemples ci-dessus. Le signe **+** indique que cette tâche est la tâche actuelle par défaut. C'est-à-dire que, si une commande attend un argument **%numéro de tâche** et qu'un numéro de tâche n'est pas fourni, l'action est prise dans la tâche avec l'indicateur **+**.



RÉFÉRENCES

Page info Bash (*Manuel de référence GNU Bash*)

<https://www.gnu.org/software/bash/manual>

- Section 7 : Contrôle des tâches

Pages de manuel **bash(1)**, **builtins(1)**, **ps(1)**, **sleep(1)**

► EXERCICE GUIDÉ

Contrôle des tâches

Dans cet exercice, vous allez démarrer, suspendre, mettre en arrière-plan et mettre en avant plusieurs processus à l'aide du contrôle des tâches.

RÉSULTATS

Vous devez pouvoir utiliser le contrôle des tâches pour suspendre et redémarrer les processus utilisateur.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab processes-control start**. Ce script s'assure que **servera** est disponible.

```
[student@workstation ~]$ lab processes-control start
```

- 1. Sur **workstation**, ouvrez deux fenêtres de terminal côté à côté. Dans cette section, ces deux terminaux sont appelés *gauche* et *droite*. Sur chaque terminal, utilisez la commande **ssh** pour vous connecter à **servera** en tant qu'utilisateur **student**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Dans la fenêtre *gauche*, créez un nouveau répertoire appelé **/home/student/bin**. Dans le nouveau répertoire, créez un script shell appelé **control**. Rendez le script exécutable.

 - 2.1. Utilisez la commande **mkdir** pour créer un nouveau répertoire appelé **/home/student/bin**.

```
[student@servera ~]$ mkdir /home/student/bin
```

- 2.2. Utilisez la commande **vim** pour créer un script appelé **control** dans le répertoire **/home/student/bin**. Pour accéder au mode interactif Vim, appuyez sur la touche **i**. Utilisez la commande **:wq** pour enregistrer le fichier.

```
[student@servera ~]$ vim /home/student/bin/control
#!/bin/bash
while true; do
    echo -n "$@" >> ~/control_outfile
    sleep 1
done
```

**NOTE**

Le script **control** s'exécute jusqu'à ce qu'il soit terminé. Il ajoute des arguments de ligne de commande au fichier **~/control_outfile** une fois par seconde.

2.3. Utilisez la commande **chmod** pour créer le fichier **control** exécutable.

```
[student@servera ~]$ chmod +x /home/student/bin/control
```

- ▶ 3. Exécutez le script **control**. Le script ajoute en permanence le mot « technical » et un espace au fichier **~/control_outfile** à des intervalles d'une seconde.

**NOTE**

Vous êtes capable d'exécuter votre script **control** parce qu'il est situé dans votre **PATH**, et a été rendu exécutable.

```
[student@servera ~]$ control technical
```

- ▶ 4. Dans le shell du terminal de droite, utilisez la commande **tail** avec l'option **-f** pour vérifier que le nouveau processus est bien en train d'écrire dans le fichier **/home/student/control_outfile**.

```
[student@servera ~]$ tail -f ~/control_outfile
technical technical technical technical
...output omitted...
```

- ▶ 5. Dans le terminal de gauche, appuyez sur **Ctrl+z** pour suspendre le processus en cours. Le shell renvoie l'ID de la tâche entre crochets. Dans la fenêtre de droite, vérifiez que la sortie du processus est arrêtée.

```
^Z
[1]+  Stopped                  control technical
[student@servera ~]$
```

```
technical technical technical technical
...no further output...
```

- ▶ 6. Dans le terminal de gauche, visualisez la liste **jobs**. Rappelez-vous que le signe **+** indique la tâche par défaut. Redémarrez la tâche en arrière-plan. Dans le shell du terminal de droite, vérifiez que la sortie du processus est à nouveau active.

6.1. En utilisant la commande **jobs**, affichez la liste des tâches.

```
[student@servera ~]$ jobs
[1]+  Stopped                  control technical
```

6.2. À l'aide de la commande **bg**, redémarrez la tâche **control** en arrière-plan.

```
[student@servera ~]$ bg
[1]+ control technical &
```

6.3. Utilisez la commande **jobs** pour vérifier que la tâche **control** s'exécute à nouveau.

```
[student@servera ~]$ jobs
[1]+ Running control technical &
```

6.4. Dans le shell du terminal de droite, vérifiez que la commande **tail** génère une sortie.

```
...output omitted...
technical technical technical technical technical technical technical technical
```

- ▶ 7. Dans le shell du terminal de gauche, démarrez deux processus **control** supplémentaires à ajouter au fichier **~/output**. Utilisez l'esperluette (&) pour démarrer les processus en arrière-plan. Remplacez **technical** par **documents**, puis par **database**. Le remplacement des arguments permet de différencier les trois processus.

```
[student@servera ~]$ control documents &
[2] 6579
[student@servera ~]$
[student@servera ~]$ control database &
[3] 6654
```



NOTE

Le numéro de tâche de chaque nouveau processus est imprimé entre crochets. Le deuxième numéro correspond au numéro d'identification de processus (PID) unique à l'échelle du système pour le processus.

- ▶ 8. Dans le shell du terminal de gauche, utilisez la commande **jobs** pour afficher les trois processus en cours. Dans le shell du terminal de droite, vérifiez que les trois processus sont bien en train d'ajouter des éléments au fichier.

```
[student@servera ~]$ jobs
[1] Running control technical &
[2]- Running control documents &
[3]+ Running control database &
```

```
...output omitted...
technical documents database technical documents database technical documents
database technical documents database
...output omitted...
```

- ▶ 9. Suspendez le processus **control technical**. Vérifiez qu'il a été suspendu. Terminez le processus **control documents** et vérifiez qu'il a été terminé.

- 9.1. Dans le shell du terminal de gauche, utilisez la commande **fg** avec l'ID de tâche pour mettre le processus **control technical** à l'avant-plan. Appuyez sur **Ctrl+z** pour

suspendre le processus. Utilisez la commande **jobs** pour confirmer que le processus est suspendu.

```
[student@servera ~]$ fg %1
control technical
^Z
[1]+  Stopped                  control technical
[student@servera ~]$ jobs
[1]+  Stopped                  control technical
[2]   Running                 control documents &
[3]-  Running                 control database &
```

- 9.2. Dans le shell du terminal de droite, vérifiez que le processus **control technical** n'envoie plus de sortie.

```
database documents database documents database
...no further output...
```

- 9.3. Dans le shell du terminal de gauche, utilisez la commande **fg** avec l'ID de tâche pour mettre le processus **control documents** à l'avant-plan. Appuyez sur **Ctrl+C** pour mettre fin au processus. Utilisez la commande **jobs** pour vérifier que le processus est terminé.

```
[student@servera ~]$ fg %2
control documents
^C
[student@servera ~]$ jobs
[1]+  Stopped                  control technical
[3]-  Running                 control database &
```

- 9.4. Dans le shell du terminal de droite, vérifiez que le processus **control documents** n'envoie plus de sortie.

```
...output omitted...
database database database database database database database
...no further output...
```

- 10. Dans la fenêtre de gauche, utilisez la commande **ps** avec l'option **jT** pour afficher les tâches restantes. Les tâches suspendues ont pour état **T**. Les autres tâches en arrière-plan dorment (**S**).

```
[student@servera ~]$ ps jT
  PPID  PID  PGID  SID TTY      TPGID STAT   UID    TIME COMMAND
27277 27278 27278 27278 pts/1    28702 Ss    1000   0:00 -bash
27278 28234 28234 27278 pts/1    28702 T     1000   0:00 /bin/bash /home/student/
bin/control technical
27278 28251 28251 27278 pts/1    28702 S     1000   0:00 /bin/bash /home/student/
bin/control database
28234 28316 28234 27278 pts/1    28702 T     1000   0:00 sleep 1
28251 28701 28251 27278 pts/1    28702 S     1000   0:00 sleep 1
27278 28702 28702 27278 pts/1    28702 R+    1000   0:00 ps jT
```

- 11. Dans la fenêtre de gauche, utilisez la commande **jobs** pour afficher les tâches en cours. Terminez le processus **control database** et vérifiez qu'il a été terminé.

```
[student@servera ~]$ jobs  
[1]+ Stopped control technical  
[3]- Running control database &
```

Utilisez la commande **fg** avec l'ID de tâche pour mettre le processus **control database** à l'avant-plan. Appuyez sur **Ctrl+c** pour mettre fin au processus. Utilisez la commande **jobs** pour vérifier que le processus est terminé.

```
[student@servera ~]$ fg %3  
control database  
^C  
[student@servera ~]$ jobs  
[1]+ Stopped control technical
```

- 12. Dans le shell du terminal de droite, utilisez la commande **Ctrl+c** pour arrêter la commande **tail**. À l'aide de la commande **rm**, supprimez le fichier **~/control_outfile**.

```
...output omitted...  
Ctrl+c  
[student@servera ~]$ rm ~/control_outfile
```

- 13. Déconnectez-vous de **servera** sur les deux terminaux.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.
```

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.
```

Fin

Sur **workstation**, exécutez le script **lab processes-control finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab processes-control finish
```

L'exercice guidé est maintenant terminé.

SUPPRESSION DE PROCESSUS

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir :

- Utiliser des commandes pour supprimer les processus et communiquer avec eux.
- Définir les caractéristiques d'un processus démon.
- Mettre fin à des sessions et des processus utilisateur.

CONTRÔLE DE PROCESSUS À L'AIDE DE SIGNAUX

Un signal est une interruption logicielle envoyée à un processus. Les signaux renvoient des événements à un programme en cours d'exécution. Les événements qui génèrent un signal peuvent être une erreur, un événement externe (une requête d'E/S ou l'expiration d'un chronomètre) ou une utilisation explicite d'une commande d'envoi de signal ou d'une séquence de clavier.

Le tableau suivant répertorie les principaux signaux utilisés régulièrement par les administrateurs système pour la gestion des processus. Les signaux sont indiqués soit par leur nom abrégé (HUP), soit par leur nom complet (SIGHUP).

Signaux fondamentaux de gestion des processus

NUMÉRO DE SIGNAL	NOM ABRÉGÉ	DÉFINITION	OBJET
1	HUP	Suspendre (Hangup)	Sert à signifier l'arrêt du processus de contrôle d'un terminal. Sert aussi à demander la réinitialisation du processus (rechargement de configuration) sans le supprimer.
2	INT	Arrêt (interrupt) au clavier	Provoque l'arrêt du programme. Peut être bloqué ou traité. Envoyé en appuyant sur la séquence de touches INTR (Ctrl+c).
3	QUITTER	Quitter au clavier	Semblable à SIGINT ; ajoute un vidage du processus lors de l'arrêt. Envoyé en appuyant sur la séquence de touches QUIT (Ctrl+\).
9	TUER	Suppression, non blocable	Provoque l'arrêt abrupt du programme. Ne peut être bloqué, ignoré ou traité ; toujours fatal.
15 défaut	TERM	Arrêter	Provoque l'arrêt du programme. Contrairement à SIGKILL, peut être bloqué, ignoré ou traité. Manière « polie » de demander l'arrêt d'un programme ; permet l'auto-nettoyage.

NUMÉRO DE SIGNAL	NOM ABRÉGÉ	DÉFINITION	OBJET
18	CONT	Continuer	Envoyé à un processus pour le faire reprendre, s'il a été interrompu. Ne peut pas être bloqué. Même si traité, fait toujours reprendre le processus.
19	STOP	Arrêter, ne peut pas être bloqué	Suspend le processus. Ne peut être ni bloqué ni traité.
20	TSTP	Arrêt par le clavier	Contrairement à SIGSTOP, peut être bloqué, ignoré ou traité. Envoyé en appuyant sur la séquence de touches SUSP (Ctrl+z).



NOTE

Les numéros des signaux varient selon la plateforme Linux utilisée, mais leur nom et leur signification sont standardisés. Pour l'utilisation avec des commandes, il est conseillé d'utiliser le nom des signaux plutôt que leur numéro. Les numéros abordés dans cette section s'appliquent aux systèmes x86_64.

Chaque signal possède une *action par défaut*, qui est généralement l'une des suivantes :

- **Term** : provoquer la fermeture (quitter ou exit) immédiate d'un programme.
- **Core** : provoquer l'enregistrement d'une image (vidage système ou core dump), puis la fermeture du programme.
- **Stop** : provoquer l'interruption (suspension ou suspend) d'un programme en attendant sa reprise (resume).

On peut préparer les programmes pour qu'ils réagissent à des signaux d'événements attendus pour mettre en œuvre des routines de traitement pour ignorer, remplacer ou étendre l'action par défaut d'un signal.

Commandes pour l'envoi de signaux par requête explicite

Vous pouvez envoyer un signal à leur processus en avant-plan en appuyant sur un raccourci clavier pour suspendre (**Ctrl+z**), supprimer (**Ctrl+c**) ou vider (**Ctrl+**) le processus. Toutefois, vous utiliserez des commandes d'envoi de signaux pour envoyer des signaux aux processus exécutés en arrière-plan ou dans une session différente.

Les signaux peuvent être spécifiés en tant qu'options soit à l'aide de leur nom (par exemple, **-HUP** ou **-SIGHUP**), soit à l'aide de leur numéro (le **-1** associé). Les utilisateurs peuvent supprimer leurs propres processus, mais doivent disposer des priviléges root pour supprimer ceux des autres.

La commande **kill** envoie un signal à un processus à l'aide de son PID. Malgré son nom, la commande **kill** peut être utilisée pour envoyer n'importe quel signal, et pas seulement pour supprimer un programme. Vous pouvez utiliser la commande **kill -l** pour lister les noms et les numéros de tous les signaux disponibles.

```
[user@host ~]$ kill -1
 1) SIGHUP      2) SIGINT      3) SIGQUIT      4) SIGILL      5) SIGTRAP
 6) SIGABRT     7) SIGBUS      8) SIGFPE       9) SIGKILL     10) SIGUSR1
11) SIGSEGV     12) SIGUSR2     13) SIGPIPE     14) SIGALRM     15) SIGTERM
16) SIGSTKFLT   17) SIGCHLD     18) SIGCONT     19) SIGSTOP     20) SIGTSTP
...output omitted...
[user@host ~]$ ps aux | grep job
5194  0.0  0.1 222448  2980 pts/1    S    16:39   0:00 /bin/bash /home/user/bin/
control job1
5199  0.0  0.1 222448  3132 pts/1    S    16:39   0:00 /bin/bash /home/user/bin/
control job2
5205  0.0  0.1 222448  3124 pts/1    S    16:39   0:00 /bin/bash /home/user/bin/
control job3
5430  0.0  0.0 221860  1096 pts/1    S+   16:41   0:00 grep --color=auto job
[user@host ~]$ kill 5194
[user@host ~]$ ps aux | grep job
user  5199  0.0  0.1 222448  3132 pts/1    S    16:39   0:00 /bin/bash /home/
user/bin/control job2
user  5205  0.0  0.1 222448  3124 pts/1    S    16:39   0:00 /bin/bash /home/
user/bin/control job3
user  5783  0.0  0.0 221860   964 pts/1    S+   16:43   0:00 grep --color=auto
job
[1]  Terminated                  control job1
[user@host ~]$ kill -9 5199
[user@host ~]$ ps aux | grep job
user  5205  0.0  0.1 222448  3124 pts/1    S    16:39   0:00 /bin/bash /home/
user/bin/control job3
user  5930  0.0  0.0 221860  1048 pts/1    S+   16:44   0:00 grep --color=auto
job
[2]- Killed                     control job2
[user@host ~]$ kill -SIGTERM 5205
user  5986  0.0  0.0 221860  1048 pts/1    S+   16:45   0:00 grep --color=auto
job
[3]+ Terminated                 control job3
```

La commande **killall** peut signaler plusieurs processus en fonction de leur nom de commande.

```
[user@host ~]$ ps aux | grep job
5194  0.0  0.1 222448  2980 pts/1    S    16:39   0:00 /bin/bash /home/user/bin/
control job1
5199  0.0  0.1 222448  3132 pts/1    S    16:39   0:00 /bin/bash /home/user/bin/
control job2
5205  0.0  0.1 222448  3124 pts/1    S    16:39   0:00 /bin/bash /home/user/bin/
control job3
5430  0.0  0.0 221860  1096 pts/1    S+   16:41   0:00 grep --color=auto job
[user@host ~]$ killall control
[1]  Terminated                  control job1
[2]- Terminated                 control job2
[3]+ Terminated                 control job3
[user@host ~]$
```

Utilisez **pkill** pour envoyer un signal à un ou plusieurs processus correspondant aux critères de sélection. Les critères de sélection peuvent être un nom de commande, un processus appartenant

à un utilisateur spécifique ou des processus dans l'ensemble du système. La commande **pkill** inclut des critères de sélection avancés :

- Command : processus dont le nom de commande correspond à un modèle.
- UID : processus appartenant à un compte utilisateur Linux, effectif ou réel.
- GID : processus appartenant à un compte de groupe Linux, effectif ou réel.
- Parent : processus enfants d'un processus parent spécifique.
- Terminal : processus exécutés sur un terminal de contrôle spécifique.

```
[user@host ~]$ ps aux | grep pkill
user  5992  0.0  0.1 222448  3040 pts/1    S    16:59   0:00 /bin/bash /home/
user/bin/control pkill1
user  5996  0.0  0.1 222448  3048 pts/1    S    16:59   0:00 /bin/bash /home/
user/bin/control pkill2
user  6004  0.0  0.1 222448  3048 pts/1    S    16:59   0:00 /bin/bash /home/
user/bin/control pkill3
[user@host ~]$ pkill control
[1]  Terminated                  control pkill1
[2]- Terminated                  control pkill2
[user@host ~]$ ps aux | grep pkill
user  6219  0.0  0.0 221860  1052 pts/1    S+   17:00   0:00 grep --color=auto
  pkill
[3]+  Terminated                  control pkill3
[user@host ~]$ ps aux | grep test
user  6281  0.0  0.1 222448  3012 pts/1    S    17:04   0:00 /bin/bash /home/
user/bin/control test1
user  6285  0.0  0.1 222448  3128 pts/1    S    17:04   0:00 /bin/bash /home/
user/bin/control test2
user  6292  0.0  0.1 222448  3064 pts/1    S    17:04   0:00 /bin/bash /home/
user/bin/control test3
user  6318  0.0  0.0 221860  1080 pts/1    S+   17:04   0:00 grep --color=auto
  test
[user@host ~]$ pkill -U user
[user@host ~]$ ps aux | grep test
user  6870  0.0  0.0 221860  1048 pts/0    S+   17:07   0:00 grep --color=auto
  test
[user@host ~]$
```

DÉCONNEXION DES UTILISATEURS PAR LES ADMINISTRATEURS

Vous devez peut-être déconnecter d'autres utilisateurs pour diverses raisons. Pour nommer quelques-unes des nombreuses possibilités : l'utilisateur a commis une violation de la sécurité ; l'utilisateur peut avoir utilisé trop de ressources ; l'utilisateur peut avoir un système qui ne répond pas ; ou l'utilisateur dispose d'un accès inappropriate aux documents. Dans ces cas, vous devrez peut-être terminer administrativement leur session en utilisant des signaux.

Pour déconnecter un utilisateur, identifiez d'abord la session de connexion à terminer. Utilisez la commande **w** pour lister les connexions des utilisateurs et les processus en cours. Notez les colonnes **TTY** et **FROM** pour déterminer les sessions à fermer.

Toutes les sessions de connexion sont associées à un périphérique de terminal (TTY). Si le nom du périphérique est de la forme **pts/N**, alors il s'agit d'un *pseudo-terminal* associé à une fenêtre de terminal graphique ou à une session de connexion à distance. S'il est de la forme **ttyN**, l'utilisateur

se trouve sur la console d'un système, une autre console ou un autre périphérique terminal connecté en direct.

```
[user@host ~]$ w
12:43:06 up 27 min, 5 users, load average: 0.03, 0.17, 0.66
USER      TTY      FROM          LOGIN@    IDLE     JCPU    PCPU WHAT
root      tty2
bob       tty3
user      pts/1   desk.example.com 12:41    2.00s   0.03s  0.03s w
[user@host ~]$
```

Découvrez le temps qu'un utilisateur a passé sur le système en affichant l'heure d'ouverture de sa session. Pour chaque session, les ressources processeur consommées par les tâches en cours, y compris les tâches en arrière-plan et les processus enfants, se trouvent dans la colonne **JCPU**. La consommation des ressources processeur pour l'exécution des processus en avant-plan est affichée dans la colonne **PCPU**.

Les signaux peuvent être envoyés aux processus et aux sessions un par un ou par groupes. Pour mettre fin à tous les processus d'un utilisateur, utilisez la commande **pkill**. Parce que le processus initial d'une session de connexion (*leader de session*) est destiné à traiter les demandes de fermeture et à ignorer les signaux involontaires envoyés par le clavier, l'arrêt de tous les processus et des shells de connexion d'un utilisateur requiert l'utilisation du signal SIGKILL.



IMPORTANT

SIGKILL est généralement utilisé trop rapidement par les administrateurs.

Comme le signal SIGKILL ne peut être ni traité ni ignoré, il est toujours fatal. Cependant, il force l'arrêt des processus sans les autoriser à exécuter les routines d'auto-nettoyage. Il est recommandé d'envoyer d'abord le signal SIGTERM, puis d'essayer avec le signal SIGKILL et d'uniquement réessayer avec le signal SIGKILL si aucun ne répond.

Identifiez d'abord les numéros PID à supprimer à l'aide de **pgrep**, qui fonctionne de façon très similaire à **pkill**, y compris en utilisant les mêmes options, sauf que **pgrep** liste des processus plutôt que de les supprimer.

```
[root@host ~]# pgrep -l -u bob
6964 bash
6998 sleep
6999 sleep
7000 sleep
[root@host ~]# pkill -SIGKILL -u bob
[root@host ~]# pgrep -l -u bob
[root@host ~]#
```

Lorsque des processus requérant une attention particulière sont exécutés au cours d'une même session de connexion, il n'est pas forcément nécessaire d'arrêter tous les processus d'un utilisateur. Déterminez le terminal de contrôle de la session à l'aide de la commande **w**, puis supprimez (kill) uniquement les processus qui font référence au même identifiant de terminal. À moins que **SIGKILL** ne soit spécifié, le leader de session (ici, le shell de connexion Bash) traite la demande d'arrêt et y survit, mais il est mis fin à tous les autres processus de la session.

```
[root@host ~]# pgrep -l -u bob
7391 bash
7426 sleep
7427 sleep
7428 sleep
[root@host ~]# w -h -u bob
bob      tty3      18:37    5:04   0.03s  0.03s -bash
[root@host ~]# pkill -t tty3
[root@host ~]# pgrep -l -u bob
7391 bash
[root@host ~]# pkill -SIGKILL -t tty3
[root@host ~]# pgrep -l -u bob
[root@host ~]#
```

La même méthode d'arrêt sélectif peut être appliquée en utilisant les relations entre processus parents et enfants. Utilisez la commande **pstree** pour afficher l'arborescence des processus d'un système ou d'un seul utilisateur. Utilisez le PID des processus parents pour supprimer (kill) tous les processus enfants qu'ils ont créés. Cette fois, le shell de connexion Bash parent survit, car le signal ne s'adresse qu'à ses processus enfants.

```
[root@host ~]# pstree -p bob
bash(8391)---sleep(8425)
          |   ---sleep(8426)
          |   \---sleep(8427)
[root@host ~]# pkill -P 8391
[root@host ~]# pgrep -l -u bob
bash(8391)
[root@host ~]# pkill -SIGKILL -P 8391
[root@host ~]# pgrep -l -u bob
bash(8391)
[root@host ~]#
```



RÉFÉRENCES

info libc signal (*Manuel de référence de la bibliothèque C GNU*)

- Section 24 : Traitement des signaux

info libc processes (*Manuel de référence de la bibliothèque C GNU*)

- Section 26 : Processus

Pages de manuel **kill(1)**, **killall(1)**, **pgrep(1)**, **pkill(1)**, **pstree(1)**, **signal(7)** et **w(1)**

► EXERCICE GUIDÉ

SUPPRESSION DE PROCESSUS

Dans cet exercice, vous utiliserez des signaux pour gérer et arrêter les processus.

RÉSULTATS

Vous devez être capable de démarrer et d'arrêter plusieurs processus shell.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez la commande `lab processes-kill start`. La commande exécute un script de démarrage qui détermine si l'hôte, `servera`, est accessible sur le réseau.

```
[student@workstation ~]$ lab processes-kill start
```

- ▶ 1. Sur `workstation`, ouvrez deux fenêtres de terminal côté à côté. Dans cette section, ces terminaux sont appelés *gauche* et *droite*. Sur chaque terminal, utilisez la commande `ssh` pour vous connecter à `servera` en tant qu'utilisateur `student`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. Dans la fenêtre *gauche*, créez un nouveau répertoire appelé `/home/student/bin`. Dans le nouveau répertoire, créez un script shell appelé `killing`. Rendez le script exécutable.

- 2.1. Utilisez la commande `mkdir` pour créer un nouveau répertoire appelé `/home/student/bin`.

```
[student@servera ~]$ mkdir /home/student/bin
```

- 2.2. Utilisez la commande `vim` pour créer un script appelé `killing` dans le répertoire `/home/student/bin`. Appuyez sur la touche `i` pour accéder au mode interactif Vim. Utilisez la commande `:wq` pour enregistrer le fichier.

```
[student@servera ~]$ vim /home/student/bin/killing
#!/bin/bash
while true; do
    echo -n "$@" >> ~/killing_outfile
    sleep 5
done
```

**NOTE**

Le script **killing** s'exécute jusqu'à ce qu'il soit terminé. Il ajoute des arguments de ligne de commande à **~/killing_outfile** une fois toutes les 5 secondes.

- 2.3. Utilisez la commande **chmod** pour créer le fichier **killing** exécutable.

```
[student@servera ~]$ chmod +x /home/student/bin/killing
```

- 3. Dans le shell du terminal de gauche, utilisez la commande **cd** pour accéder au répertoire **/home/student/bin/**. Démarrez trois processus **killing** avec les arguments **network**, **interface** et **connection**, respectivement. Démarrez trois processus appelés **network**, **interface** et **connection**. Utilisez l'esperluette (**&**) pour démarrer les processus en arrière-plan.

```
[student@servera ~]$ cd /home/student/bin
[student@servera bin]$ killing network &
[1] 3460
[student@servera bin]$ killing interface &
[2] 3482
[student@servera bin]$ killing connection &
[3] 3516
```

Vos processus auront différents numéros de PID.

- 4. Dans le shell du terminal de droite, utilisez la commande **tail** avec l'option **-f** pour vérifier que les trois processus sont bien en train d'ajouter des éléments au fichier **/home/student/killing_outfile**.

```
[student@servera ~]$ tail -f ~/killing_outfile
network interface network connection interface network connection interface
network
...output omitted...
```

- 5. Dans le shell du terminal de gauche, utilisez la commande **jobs** pour lister les tâches.

```
[student@servera bin]$ jobs
[1]  Running                  killing network &
[2]- Running                  killing interface &
[3]+ Running                  killing connection &
```

- 6. Utilisez des signaux pour suspendre le processus **network**. Vérifiez que le processus **network** est arrêté (**Stopped**). Dans le shell du terminal de droite, vérifiez que le processus **network** n'ajoute plus la sortie à **~/killing_output**.

- 6.1. Utilisez la commande **kill** avec l'option **-SIGSTOP** pour arrêter le processus **network**. Exécutez **jobs** pour vérifier qu'il est arrêté.

```
[student@servera bin]$ kill -SIGSTOP %1
[1]+  Stopped                  killing network
[student@servera bin]$ jobs
[1]+  Stopped                  killing network
[2]   Running                 killing interface &
[3]-  Running                 killing connection &
```

- 6.2. Dans le shell du terminal de droite, regardez la sortie de la commande **tail**. Vérifiez que le mot **network** n'est plus ajouté au fichier **~/killing_outfile**.

```
...output omitted...
interface connection interface connection interface connection interface
```

- ▶ 7. Dans le shell du terminal de gauche, terminez le processus **interface** à l'aide de signaux. Vérifiez que le processus **interface** a disparu. Dans le shell du terminal de droite, vérifiez que la sortie du processus **interface** n'est plus ajoutée au fichier **~/killing_outfile**.
- 7.1. Utilisez la commande **kill** avec l'option **-SIGTERM** pour terminer le processus **interface**. Exécutez la commande **jobs** pour vérifier qu'il a été terminé.

```
[student@servera bin]$ kill -SIGTERM %2
[student@servera bin]$ jobs
[1]+  Stopped                  killing network
[2]  Terminated                killing interface
[3]-  Running                 killing connection &
```

- 7.2. Dans le shell du terminal de droite, regardez la sortie de la commande **tail**. Vérifiez que le mot **interface** n'est plus ajouté au fichier **~/killing_outfile**.

```
...output omitted...
connection connection connection connection connection connection connection
connection
```

- ▶ 8. Dans le shell du terminal de gauche, redémarrez le processus **network** à l'aide de signaux. Vérifiez que le processus **network** est en cours d'exécution (**Running**). Dans la fenêtre de droite, vérifiez que la sortie du processus **network** est ajoutée au fichier **~/killing_outfile**.
- 8.1. Utilisez la commande **kill** avec l'option **-SIGCONT** pour redémarrer le processus **network**. Exécutez la commande **jobs** pour vérifier que le processus est en cours d'exécution (**Running**).

```
[student@servera bin]$ kill -SIGCONT %1
[student@servera bin]$ jobs
[1]+  Running                  killing network &
[3]-  Running                 killing connection &
```

- 8.2. Dans le shell du terminal de droite, regardez la sortie de la commande **tail**. Vérifiez que le mot **network** est ajouté au fichier **~/killing_outfile**.

```
...output omitted...
network connection network connection network connection network connection
network connection
```

- 9. Dans le terminal gauche, terminez les deux tâches restantes. Vérifiez qu'il ne reste plus de tâches et que la sortie est arrêtée.

9.1. Utilisez la commande **kill** avec l'option **-SIGTERM** pour terminer le processus **network**. Utilisez la même commande pour terminer le processus **connection**.

```
[student@servera bin]$ kill -SIGTERM %1
[student@servera bin]$ kill -SIGTERM %3
[1]+  Terminated          killing network
[student@servera bin]$ jobs
[3]+  Terminated          killing connection
```

- 10. Dans le shell du terminal de gauche, listez les processus **tail** s'exécutant sur tous les shells de terminal ouverts. Terminez les commandes **tail** en cours d'exécution. Vérifiez que le processus n'est plus en cours d'exécution.

10.1. Utilisez la commande **ps** avec l'option **-ef** pour lister tous les processus **tail** en cours d'exécution. Affinez la recherche en utilisant la commande **grep**.

```
[student@servera bin]$ ps -ef | grep tail
student    4581 31358  0 10:02 pts/0    00:00:00 tail -f killing_outfile
student    4869  2252  0 10:33 pts/1    00:00:00 grep --color=auto tail
```

10.2. Utilisez la commande **pkill** avec l'option **-SIGTERM** pour supprimer le processus **tail**. Utilisez la commande **ps** pour confirmer qu'il n'est plus présent.

```
[student@servera bin]$ pkill -SIGTERM tail
[student@servera bin]$ ps -ef | grep tail
student    4874  2252  0 10:36 pts/1    00:00:00 grep --color=auto tail
```

10.3. Dans le shell du terminal de droite, vérifiez que la commande **tail** n'est plus en cours d'exécution.

```
...output omitted...
network connection network connection network connection Terminated
[student@servera ~]$
```

- 11. Quittez les deux fenêtres du terminal. Si vous ne quittez pas toutes les sessions, le script de fin échoue.

```
[student@servera bin]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Finish (Terminer)

Sur workstation, exéutez le script **lab processes-kill finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab processes-kill finish
```

L'exercice guidé est maintenant terminé.

CONTÔLE DE L'ACTIVITÉ DES PROCESSUS

OBJECTIFS

Au terme de cette section, vous serez en mesure de décrire la charge moyenne et déterminer les processus responsables d'une utilisation intensive des ressources sur un serveur.

DESCRIPTION DE LA CHARGE MOYENNE

La *charge moyenne* est une mesure fournie par le noyau Linux qui constitue un moyen simple de représenter la charge système perçue au fil du temps. Elle peut être utilisée pour déterminer de manière générale le nombre de demandes de ressources système en attente et pour déterminer si la charge du système augmente ou diminue avec le temps.

Toutes les cinq secondes, le noyau collecte l'*indice de charge* actuelle en fonction du nombre de processus se trouvant dans des états exécutables et ne pouvant être interrompus. Ce nombre est accumulé et signalé en tant que moyenne mobile exponentielle de la dernière minute, et des cinq et quinze dernières minutes.

Compréhension du calcul de la charge moyenne sous Linux

La charge moyenne représente la charge du système telle qu'elle est perçue sur une période donnée. Linux la détermine en indiquant le nombre de processus prêts à être exécutés sur un processeur ainsi que le nombre de processus en attente de disque ou de réseau E/S à compléter.

- L'indice de charge se base essentiellement sur le nombre de processus prêts à être exécutés (en cours de traitement **R**) et attendant que E/S se termine (en cours de traitement **D**).
- Certains systèmes UNIX ne tiennent compte que de l'usage du processeur ou de la longueur des files d'attente d'exécution pour indiquer la charge du système. Linux inclut également l'utilisation de disque ou de réseau, car cela peut avoir un impact aussi important sur les performances du système que sur la charge du processeur. Lorsque les charges moyennes sont élevées alors que l'activité des processeurs est minimale, examinez l'activité du disque et du réseau.

La moyenne de charge est une mesure approximative du nombre de processus en attente d'une demande pour pouvoir effectuer quoi que ce soit d'autre. La demande peut concerner le temps nécessaire au processeur pour exécuter le processus. Sinon, la demande pourrait concerner la fin de l'exécution E/S d'un disque critique, et le processus ne peut pas être exécuté sur le processeur tant que la demande n'est pas terminée, même si le processeur est inactif. Dans les deux cas, la charge du système est touchée et le système semble fonctionner plus lentement car des processus sont en attente d'exécution.

Interprétation des valeurs de charge moyenne affichées

La commande **uptime** est un moyen d'afficher la moyenne de charge actuelle. Elle imprime l'heure actuelle, la durée d'utilisation de la machine, le nombre de sessions utilisateur en cours et la charge moyenne actuelle.

```
[user@host ~]$ uptime
15:29:03 up 14 min,  2 users,  load average: 2.92, 4.48, 5.20
```

CHAPITRE 8 | Contrôle et gestion des processus Linux

Les trois valeurs de la charge moyenne représentent la charge au cours de la dernière minute, et des cinq et quinze dernières minutes. Un coup d'œil rapide permet de savoir si la charge du système semble augmenter ou diminuer.

Si la contribution principale à la charge moyenne provient de processus en attente du processeur, vous pouvez calculer la charge moyenne approximative *par processeur* pour déterminer si le système connaît des temps d'attente importants.

La commande **lscpu** peut vous aider à déterminer le nombre de processeurs d'un système.

Dans l'exemple suivant, le système est un système simple socket double cœur avec deux hyperthreads par cœur. Grosso modo, Linux considérera cela comme un système à quatre processeurs à des fins de planification.

```
[user@host ~]$ 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:   2
Core(s) per socket:   2
Socket(s):             1
NUMA node(s):          1
...output omitted...
```

Imaginez pendant un moment que la seule contribution à l'indice de charge provienne de processus nécessitant du temps processeur. Vous pouvez alors diviser les valeurs de charge moyenne affichées par le nombre de processeurs logiques du système. Une valeur inférieure à 1 indique une utilisation satisfaisante des ressources et des temps d'attente minimaux. Une valeur supérieure à 1 indique une saturation des ressources, et un certain temps de traitement.

```
# From lscpu, the system has four logical CPUs, so divide by 4:
#                           load average: 2.92, 4.48, 5.20
#       divide by number of logical CPUs:    4    4    4
#                                         -----
#                           per-CPU load average: 0.73  1.12  1.30
#
# This system's load average appears to be decreasing.
# With a load average of 2.92 on four CPUs, all CPUs were in use ~73% of the time.
# During the last 5 minutes, the system was overloaded by ~12%.
# During the last 15 minutes, the system was overloaded by ~30%.
```

Une file d'attente inactive de processeur possède un indice de charge de 0. Chaque processus en attente de processeur ajoute 1 à l'indice de charge. Si un processus est en cours d'exécution sur un processeur, l'indice de charge est de 1, la ressource (le processeur) est en cours d'utilisation, mais aucune demande n'est en attente. Si ce processus est en cours d'exécution pendant une minute complète, sa contribution à la moyenne de charge d'une minute sera de 1.

Cependant, les processus en attente d'entrées-sorties critiques, mises en veille de manière ininterrompue quand les ressources de disque ou de réseau sont occupées, sont également pris en compte dans le calcul et augmentent la charge moyenne. Bien que cela ne soit pas une indication de l'utilisation d'un processeur, ces processus s'ajoutent au nombre de files d'attente car ils attendent des ressources et ne peuvent pas s'exécuter sur un processeur tant qu'ils ne les

ont pas obtenues. Cela représente toujours la charge du système en raison de ressources limitées empêchant l'exécution des processus.

Tant qu'une ressource n'est pas saturée, la charge moyenne demeure inférieure à 1, puisque les tâches se retrouvent rarement placées en file d'attente. La charge moyenne n'augmente que lorsque la saturation d'une ressource oblige les requêtes à rester en file d'attente, où elles sont prises en compte par la routine de calcul de la charge. Lorsque le taux d'utilisation de la ressource atteint 100 %, chaque requête supplémentaire commence à connaître des délais d'attente de services.

Un certain nombre d'outils supplémentaires indiquent la moyenne de charge, y compris **w** et **top**.

CONTROLE DES PROCESSUS EN TEMPS RÉEL

Le programme **top** est un aperçu dynamique des processus du système, qui affiche un en-tête récapitulatif suivi d'une liste de processus ou de fils d'exécution similaire aux informations de **ps**. À la différence de la sortie statique de **ps**, **top** met à jour les informations en permanence en fonction d'un intervalle configurable, et fournit des fonctionnalités de réorganisation, de tri et de mise en évidence des colonnes. La configuration de l'utilisateur peut être enregistrée et rendue persistante.

Les colonnes de résultat par défaut se distinguent de celles des autres outils dédiés aux ressources :

- L'identifiant du processus (**PID**).
- Le nom d'utilisateur (**USER**) est le propriétaire du processus.
- La mémoire virtuelle (**VIRT**) est la mémoire totale utilisée par le processus, y compris l'ensemble résident, les bibliothèques partagées et toute page de mémoire mappée ou échangées (swap). (Identifiée par **VSZ** dans la commande **ps**.)
- La mémoire résidente (**RES**) est la mémoire physique utilisée par le processus, y compris tout objet partagé résident. (Identifiée par **RSS** dans la commande **ps**.)
- L'état du processus (**S**) est affiché comme suit :
 - **D** = Uninterruptible Sleeping (en veille, ne peut être interrompu)
 - **R** = Running or Runnable (en cours d'exécution ou exécutable)
 - **S** = Sleeping (en veille)
 - **T** = Stopped or Traced (arrêté ou suivi)
 - **Z** = Zombie
- Le temps processeur (**TIME**) est la durée totale du traitement depuis le démarrage du processus. On peut activer l'inclusion du temps cumulé de tous les processus fils précédents.
- Le nom de la commande du processus (**COMMAND**).

Toiles fondamentales dans top

CLÉ	OBJET
? ou h	Aide relative aux frappes interactives.

CLÉ	OBJET
l, t, m	Active les lignes d'en-tête de la charge, des fils d'exécution et de la mémoire.
1	Active, dans l'en-tête, l'affichage par processeur ou un résumé de tous les processeurs.
s⁽¹⁾	Change la fréquence de rafraîchissement (de l'écran) en secondes décimales (par exemple : 0,5, 1, 5).
b	Active ou désactive le surlignage des processus en cours d'exécution , gras uniquement par défaut.
Maj+b	Active l'utilisation des caractères gras dans l'affichage, dans l'en-tête et pour les processus <i>en cours d'exécution</i> .
Maj+h	Affiche ou masque les fils d'exécution ; affiche le récapitulatif des processus ou fil par fil.
u, Maj+u	Filtre tout nom d'utilisateur (effectif, réel).
Maj+m	Trie la liste des processus en fonction de l'utilisation de la mémoire, en ordre décroissant.
Maj+p	Trie la liste des processus en fonction de l'utilisation du processeur, par ordre décroissant.
k⁽¹⁾	Arrête un processus. À l'invite, saisissez PID , puis signal .
r⁽¹⁾	Change la priorité d'un processus. À l'invite, saisissez PID , puis nice_value .
Maj+w	Écrit (enregistre) la configuration d'affichage en cours pour l'utiliser au prochain redémarrage de top .
q	Quitte.
f	Gérez les colonnes en activant ou en désactivant des champs. Cela vous permet également de définir le champ de tri pour top .
Remarque :	(1) Indisponible si top est démarré en mode sécurisé. Voir top(1) .



RÉFÉRENCES

Pages de manuel **ps(1)**, **top(1)**, **uptime(1)** et **w(1)**

► EXERCICE GUIDÉ

CONTRÔLE DE L'ACTIVITÉ DES PROCESSUS

Dans cet exercice, vous allez utiliser la commande **top** pour examiner dynamiquement les processus en cours et les contrôler.

RÉSULTATS

Vous devez être capable de gérer les processus en temps réel.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab processes-monitor start**. La commande exécute un script de démarrage qui détermine si l'hôte, **servera**, est accessible sur le réseau.

```
[student@workstation ~]$ lab processes-monitor start
```

- 1. Sur **workstation**, ouvrez deux fenêtres de terminal côté à côté. Ces terminaux sont appelés *gauche* et *droite*. Sur chaque terminal, utilisez la commande **ssh** pour vous connecter à **servera** en tant qu'utilisateur **student**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Dans le shell du terminal de *gauche*, créez un nouveau répertoire appelé **/home/student/bin**. Dans le nouveau répertoire, créez un script shell appelé **monitor**. Assurez-vous que le script est exécutable.

 - 2.1. Utilisez la commande **mkdir** pour créer un nouveau répertoire appelé **/home/student/bin**.

```
[student@servera ~]$ mkdir /home/student/bin
```

- 2.2. Utilisez la commande **vim** pour créer un script appelé **monitor** dans le répertoire **/home/student/bin**. Appuyez sur la touche **i** pour accéder au mode interactif Vim. Utilisez la commande **:wq** pour enregistrer le fichier.

```
[student@servera ~]$ vim /home/student/bin/monitor
#!/bin/bash
while true; do
    var=1
```

```
while [[ var -lt 50000 ]]; do
    var=$($var+1)
done
sleep 1
done
```

**NOTE**

Le script **monitor** s'exécute jusqu'à ce qu'il soit terminé. Il génère une charge artificielle du processeur en effectuant cinquante mille problèmes d'addition. Il se met en veille ensuite pendant une seconde, réinitialise la variable et se répète.

2.3. Utilisez la commande **chmod** pour créer le fichier **monitor** exécutable.

```
[student@servera ~]$ chmod a+x /home/student/bin/monitor
```

- 3. Dans le shell du terminal de droite, exécutez l'utilitaire **top**. Redimensionnez la fenêtre pour la rendre aussi grande que possible.

```
[student@servera ~]$ top
top - 12:13:03 up 11 days, 58 min, 3 users, load average: 0.00, 0.00, 0.00
Tasks: 113 total, 2 running, 111 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.2 us, 0.0 sy, 0.0 ni, 99.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 1829.4 total, 1377.3 free, 193.9 used, 258.2 buff/cache
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1476.1 avail Mem

PID USER      PR  NI      VIRT      RES      SHR S %CPU %MEM     TIME+ COMMAND
5861 root      20   0          0          0          0 I  0.3  0.0  0:00.71 kworker/1:3-
events
6068 student   20   0  273564  4300  3688 R  0.3  0.2  0:00.01 top
  1 root      20   0  178680 13424  8924 S  0.0  0.7  0:04.03 systemd
  2 root      20   0          0          0          0 S  0.0  0.0  0:00.03 kthreadd
  3 root      20  -20          0          0          0 I  0.0  0.0  0:00.00 rcu_gp
...output omitted...
```

- 4. Dans le shell du terminal de gauche, utilisez la commande **lscpu** pour déterminer le nombre de processeurs logiques présents sur cette machine virtuelle.

```
[student@servera ~]$ lscpu
Architecture:           x86_64
CPU op-mode(s):         32-bit, 64-bit
Byte Order:             Little Endian
CPU(s):                 2
...output omitted...
```

- 5. Dans le shell du terminal de gauche, exécutez une instance unique de l'exécutable **monitor**. Utilisez l'esperluette (&) pour exécuter le processus en arrière-plan.

```
[student@servera ~]$ monitor &
[1] 6071
```

- ▶ 6. Dans le shell du terminal de droite, observez l'affichage **top**. Utilisez les touches uniques **1**, **t** et **m** pour activer les lignes d'en-tête de la charge, des fils d'exécution (threads) et de la mémoire. Après avoir observé ce comportement, assurez-vous que tous les en-têtes sont affichés.
- ▶ 7. Notez l'identifiant de processus (PID) de **monitor**. Affichez le pourcentage processeur du processus, qui devrait osciller aux alentours de 15 à 20 %.

```
[student@servera ~]$ top
PID USER      PR  NI    VIRT      RES      SHR S %CPU %MEM     TIME+ COMMAND
071 student    20   0 222448   2964    2716 S 18.7  0.2  0:27.35 monitor
...output omitted...
```

Affichez les charges moyennes. La charge moyenne calculée sur une minute est actuellement inférieure à 1. La valeur observée peut être affectée par la rétention de ressources de la part d'une autre machine virtuelle ou de l'hôte virtuel.

```
top - 12:23:45 up 11 days,  1:09,  3 users,  load average: 0.21, 0.14, 0.05
```

- ▶ 8. Dans le shell du terminal de gauche, exécutez une deuxième instance de **monitor**. Utilisez l'esperluette (&) pour exécuter le processus en arrière-plan.

```
[student@servera ~]$ monitor &
[2] 6498
```

- ▶ 9. Dans le shell du terminal de droite, notez l'ID de processus (PID) pour le second processus **monitor**. Affichez le pourcentage processeur du processus, qui devrait lui aussi osciller autour de 15 à 20 %.

```
[student@servera ~]$ top
PID USER      PR  NI    VIRT      RES      SHR S %CPU %MEM     TIME+ COMMAND
6071 student    20   0 222448   2964    2716 S 19.0  0.2  1:36.53 monitor
6498 student    20   0 222448   2996    2748 R 15.7  0.2  0:16.34 monitor
...output omitted...
```

Affichez de nouveau la charge moyenne calculée sur une minute, qui est toujours inférieure à 1. Il est important de patienter au moins une minute pour permettre au calcul de s'ajuster à la nouvelle charge de travail.

```
top - 12:27:39 up 11 days,  1:13,  3 users,  load average: 0.36, 0.25, 0.11
```

- ▶ 10. Dans le shell du terminal de gauche, exécutez une troisième instance de **monitor**. Utilisez l'esperluette (&) pour exécuter le processus en arrière-plan.

```
[student@servera ~]$ monitor &
[3] 6881
```

- 11. Dans le shell du terminal de droite, notez l'ID de processus (PID) pour le troisième processus **monitor**. Affichez le pourcentage processeur du processus, qui devrait à nouveau osciller autour de 15 à 20 %.

```
[student@servera ~]$ top
  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM   TIME+ COMMAND
6881 student    20   0 222448  3032  2784 S 18.6  0.2  0:11.48 monitor
6498 student    20   0 222448  2996  2748 S 15.6  0.2  0:47.86 monitor
6071 student    20   0 222448  2964  2716 S 18.1  0.2  2:07.86 monitor
```

Pour pousser la charge moyenne au-dessus de 1, vous devez lancer davantage de processus **monitor**. La configuration de la salle de classe dispose de 2 processeurs, donc 3 processus ne suffisent pas pour créer une contrainte significative. Lancez trois processus **monitor** supplémentaires. Affichez de nouveau la charge moyenne calculée sur une minute, qui est normalement supérieure à 1. Il est important de patienter au moins une minute pour permettre au calcul de s'ajuster à la nouvelle charge de travail.

```
[student@servera ~]$ monitor &
[4] 10708
[student@servera ~]$ monitor &
[5] 11122
[student@servera ~]$ monitor &
[6] 11338
```

```
top - 12:42:32 up 11 days,  1:28,  3 users,  load average: 1.23, 2.50, 1.54
```

- 12. Lorsque vous avez terminé d'observer les valeurs de charge moyenne, arrêtez chacun des processus **monitor** depuis **top**.

12.1. Dans le shell du terminal de droite, appuyez sur **k**. Observez l'invite sous les en-têtes et au-dessus des colonnes.

```
...output omitted...
PID to signal/kill [default pid = 11338]
```

12.2. L'invite a choisi les processus **monitor** en haut de la liste. Appuyez sur **Entrée** pour arrêter le processus.

```
...output omitted...
Send pid 11338 signal [15/sigterm]
```

12.3. Appuyez à nouveau sur **Entrée** pour confirmer le signal SIGTERM 15.

Vérifiez que le processus sélectionné n'est plus visible dans **top**. Si le PID est toujours présent, répétez ces étapes de fermeture des processus en utilisant cette fois le signal SIGKILL 9 lorsque vous y êtes invité.

```

6498 student    20   0  222448   2996   2748 R  22.9   0.2   5:31.47 monitor
6881 student    20   0  222448   3032   2784 R  21.3   0.2   4:54.47 monitor
11122 student    20   0  222448   2984   2736 R  15.3   0.2   2:32.48 monitor
 6071 student    20   0  222448   2964   2716 S  15.0   0.2   6:50.90 monitor
10708 student    20   0  222448   3032   2784 S  14.6   0.2   2:53.46 monitor

```

- ▶ 13. Répétez l'étape suivante pour chaque instance de **monitor** restante. Vérifiez qu'aucun processus **monitor** ne reste dans **top**.
- ▶ 14. Dans le shell du terminal de droite, appuyez sur **q** pour quitter **top**. Quittez **servera** sur les deux fenêtres de terminal.

```

[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$

```

```

[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$

```

Finish (Terminer)

Sur **workstation**, exécutez le script **lab processes-monitor finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab processes-monitor finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

CONTRÔLE ET GESTION DES PROCESSUS LINUX

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous vous entraînerez à localiser et à gérer les processus utilisant le plus de ressources sur un système.

RÉSULTATS

Vous devriez être capable de gérer les processus en utilisant **top** comme outil de gestion de processus.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab processes-review start**. La commande exécute un script de démarrage pour déterminer si l'hôte, **serverb**, est accessible sur le réseau.

```
[student@workstation ~]$ lab processes-review start
```

- Sur **workstation**, ouvrez deux fenêtres de terminal côté à côté. Dans cette section, ces terminaux sont appelés *gauche* et *droite*. Sur chaque fenêtre de terminal, connectez-vous à **serverb** en tant qu'utilisateur **student**. Créez un script appelé **process101**, qui va générer une charge artificielle du processeur. Créez le script dans le répertoire **/home/student/bin**.

```
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 50000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

- Dans la fenêtre de droite, exécutez l'utilitaire **top**.
- Dans le shell du terminal de gauche, déterminez le nombre de processeurs logiques présents sur la machine virtuelle. Exécutez le script **process101** en arrière-plan.
- Dans le shell du terminal de droite, observez l'affichage **top**. Basculez entre la charge, les threads et la mémoire. Notez l'identifiant de processus (PID) de **process101**. Consultez le pourcentage du processeur. Il doit osciller autour de 10 à 15 %. Veillez à ce que **top**

indique l'utilisation du processeur une fois que vous avez visualisé la charge, les threads et la mémoire.

5. Désactivez l'utilisation des caractères gras dans l'affichage. Enregistrez cette configuration pour pouvoir la réutiliser au prochain démarrage de top. Vérifiez que les modifications sont enregistrées.
6. Copiez le script **process101** dans un nouveau fichier appelé **process102**. Editez le script pour créer une plus grande charge artificielle sur le processeur. Augmentez la charge de cinquante mille à cent mille. Démarrez le processus **process102** en arrière-plan.
7. Dans le shell de terminal de droite, vérifiez que le processus est en cours d'exécution et qu'il utilise le plus de ressources possible du processeur. La charge doit osciller entre 25 % et 35 %.
8. La charge moyenne est toujours inférieure à 1. Copiez **process101** vers un nouveau script appelé **process103**. Augmentez le nombre d'addition à huit cent mille. Démarrez **process103** en arrière-plan. Confirmez que la charge moyenne est supérieure à 1. Cela peut prendre quelques minutes pour que la charge moyenne change.
9. Dans le shell du terminal de gauche, devenez l'utilisateur **root**. Suspendez le processus **process101**. Listez les tâches restantes. Observez que l'état du processus pour **process101** est maintenant **T**.
10. Reprenez le processus **process101**.
11. Terminez **process101**, **process102** et **process103** au moyen de la ligne de commande. Vérifiez que les processus ne s'affichent plus dans **top**.
12. Dans le shell du terminal de gauche, déconnectez l'utilisateur **root**. Dans le shell du terminal de droite, arrêtez la commande **top**. Quittez **serverb** sur les deux fenêtres.

Évaluation

À partir de **workstation**, exécutez le script lab processes-review grade pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab processes-review grade
```

Fin

Sur workstation, exécutez le script **lab processes-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab processes-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

CONTRÔLE ET GESTION DES PROCESSUS LINUX

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous vous entraînerez à localiser et à gérer les processus utilisant le plus de ressources sur un système.

RÉSULTATS

Vous devriez être capable de gérer les processus en utilisant **top** comme outil de gestion de processus.

AVANT DE COMMENCER

Connectez-vous à **workstation** en tant qu'utilisateur **student** avec le mot de passe **student**.

Sur **workstation**, exécutez la commande **lab processes-review start**. La commande exécute un script de démarrage pour déterminer si l'hôte, **serverb**, est accessible sur le réseau.

```
[student@workstation ~]$ lab processes-review start
```

- Sur **workstation**, ouvrez deux fenêtres de terminal côté à côté. Dans cette section, ces terminaux sont appelés *gauche* et *droite*. Sur chaque fenêtre de terminal, connectez-vous à **serverb** en tant qu'utilisateur **student**. Créez un script appelé **process101**, qui va générer une charge artificielle du processeur. Créez le script dans le répertoire **/home/student/bin**.

```
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 50000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

- Sur **workstation**, ouvrez deux fenêtres de terminal côté à côté. Sur chaque terminal, utilisez la commande **ssh** pour vous connecter à **serverb** en tant qu'utilisateur **student**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Dans le shell du terminal de gauche, utilisez la commande **mkdir** pour créer le répertoire **/home/student/bin**.

```
[student@serverb ~]$ mkdir /home/student/bin
```

- 1.3. Dans le shell du terminal de gauche, utilisez la commande **vim** pour créer le script **process101**. Appuyez sur la touche **i** pour accéder au mode interactif. Tapez **:wq** pour enregistrer le fichier.

```
[student@serverb ~]$ vim /home/student/bin/process101
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 50000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

- 1.4. Utilisez la commande **chmod** pour créer le fichier **process101** exécutable.

```
[student@serverb ~]$ chmod +x /home/student/bin/process101
```

2. Dans la fenêtre de droite, exécutez l'utilitaire **top**.

- 2.1. Dans la fenêtre de droite, exécutez l'utilitaire **top**. Redimensionnez la fenêtre pour la rendre aussi grande que possible.

```
[student@serverb ~]$ top
top - 13:47:06 up 19 min,  2 users,  load average: 0.00,  0.00,  0.00
Tasks: 110 total,   1 running, 109 sleeping,   0 stopped,   0 zombie
%CPU(s):  0.0 us,  3.1 sy,  0.0 ni, 96.9 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem : 1829.4 total, 1439.1 free,   171.9 used,   218.4 buff/cache
MiB Swap: 1024.0 total, 1024.0 free,      0.0 used. 1499.6 avail Mem

PID USER      PR  NI      VIRT      RES      SHR S %CPU %MEM     TIME COMMAND
 1 root      20   0 178536 13488 8996 S  0.0  0.7  0:01.15 systemd
 2 root      20   0      0      0      0 S  0.0  0.0  0:00.00 kthreadd
 3 root      0 -20      0      0      0 I  0.0  0.0  0:00.00 rcu_gp
 4 root      0 -20      0      0      0 I  0.0  0.0  0:00.00 rcu_par_gp
 6 root      0 -20      0      0      0 I  0.0  0.0  0:00.00 kworker/0:0H-
kblockd
...output omitted...
```

3. Dans le shell du terminal de gauche, déterminez le nombre de processeurs logiques présents sur la machine virtuelle. Exécutez le script **process101** en arrière-plan.

- 3.1. Utilisez la commande **grep** pour déterminer le nombre de processeurs logiques.

```
[student@serverb ~]$ grep "model name" /proc/cpuinfo | wc -l
2
```

CHAPITRE 8 | Contrôle et gestion des processus Linux

3.2. Utilisez la commande **cd** pour accéder au répertoire **/home/student/bin**. Exécutez le script **process101** en arrière-plan.

```
[student@serverb ~]$ cd /home/student/bin  
[student@serverb bin]$ process101 &  
[1] 1180
```

4. Dans le shell du terminal de droite, observez l'affichage **top**. Basculez entre la charge, les threads et la mémoire. Notez l'identifiant de processus (PID) de **process101**. Consultez le pourcentage du processeur. Il doit osciller autour de 10 à 15 %. Veillez à ce que **top** indique l'utilisation du processeur une fois que vous avez visualisé la charge, les threads et la mémoire.

4.1. Appuyez sur **Maj+m**.

```
top - 13:56:24 up 28 min,  2 users,  load average: 0.21, 0.08, 0.02  
Tasks: 112 total,   2 running, 110 sleeping,   0 stopped,   0 zombie  
%Cpu(s):  5.8 us,  1.3 sy,  0.0 ni, 92.8 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st  
MiB Mem : 1829.4 total, 1438.1 free,    172.7 used,   218.6 buff/cache  
MiB Swap: 1024.0 total, 1024.0 free,      0.0 used. 1498.7 avail Mem  
  
 PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND  
 705 root      20   0 409956 34880 33620 S  0.0   1.9  0:00.04 sssd_nss  
 706 root      20   0 454304 34472 14304 S  0.0   1.8  0:00.62 firewalld  
 725 root      20   0 611348 28244 14076 S  0.0   1.5  0:00.27 tuned  
 663 polkitd   20   0 1907312 23876 16040 S  0.0   1.3  0:00.04 polkitd  
 718 root      20   0 600316 17176 14832 S  0.0   0.9  0:00.06 NetworkManager  
...output omitted...
```

**NOTE**

Notez que lorsque top est activé en mode *mémoire*, **process101** n'est plus le premier processus. Vous pouvez appuyer sur **Maj+p** pour revenir à l'utilisation du processeur.

4.2. Appuyez sur **m**.

```
top - 09:32:52 up 20:05,  2 users,  load average: 0.18, 0.10, 0.03  
Tasks: 112 total,   2 running, 110 sleeping,   0 stopped,   0 zombie  
%Cpu(s):  7.8/1.5  9[|||||||||] ]  
MiB Mem : 18.3/1829.4 [||||||||||||||||||] ]  
MiB Swap:  0.0/1024.0 [ ] ]  
  
 PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND  
 705 root      20   0 409956 34880 33620 S  0.0   1.9  0:00.04 sssd_nss  
 706 root      20   0 454304 34472 14304 S  0.0   1.8  0:00.62 firewalld  
 725 root      20   0 611348 28244 14076 S  0.0   1.5  0:00.30 tuned  
 663 polkitd   20   0 1907312 23876 16040 S  0.0   1.3  0:00.04 polkitd  
 718 root      20   0 600316 17176 14832 S  0.0   0.9  0:00.07 NetworkManager  
...output omitted...
```

4.3. Appuyez sur **t**.

```
Tasks: 113 total, 2 running, 111 sleeping, 0 stopped, 0 zombie
%Cpu(s): 7.8/1.5 9[|||||||||]
MiB Mem : 1829.4 total, 1436.7 free, 173.7 used, 219.0 buff/cache
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1497.7 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
1180 student 20 0 222448 3056 2808 S 12.0 0.2 1:59.94 process101
705 root 20 0 409956 34880 33620 S 0.0 1.9 0:00.04 sssd_nss
706 root 20 0 454304 34472 14304 S 0.0 1.8 0:00.62 firewalld
725 root 20 0 611348 28244 14076 S 0.0 1.5 0:00.30 tuned
663 polkitd 20 0 1907312 23876 16040 S 0.0 1.3 0:00.04 polkitd
718 root 20 0 600316 17176 14832 S 0.0 0.9 0:00.07 NetworkManager
...output omitted...
```

4.4. Appuyez sur Maj+p.

```
top - 09:35:48 up 20:08, 2 users, load average: 0.10, 0.10, 0.04
Tasks: 110 total, 4 running, 106 sleeping, 0 stopped, 0 zombie
%Cpu(s): 6.8 us, 1.0 sy, 0.0 ni, 92.2 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 18.3/1829.4 [||||||||||||||||||] ]
MiB Swap: 0.0/1024.0 [ ]]

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
27179 student 20 0 222448 3060 2812 R 15.6 0.2 1:13.47 process101
...output omitted...
```

- Désactivez l'utilisation des caractères gras dans l'affichage. Enregistrez cette configuration pour pouvoir la réutiliser au prochain démarrage de top. Vérifiez que les modifications sont enregistrées.

5.1. Appuyez sur Maj+b pour désactiver l'utilisation du gras.

```
top - 19:40:30 up 6:12, 2 users, load average: 0.11, 0.12, 0.09
Tasks: 112 total, 1 running, 111 sleeping, 0 stopped, 0 zombie
%Cpu(s): 7.6/1.5 9[|||||||||] ]
MiB Mem : 18.2/1829.4 [||||||||||||||||||] ]
MiB Swap: 0.0/1024.0 [ ]]
```

- Appuyez sur Maj+w pour enregistrer cette configuration. La configuration par défaut est stockée dans le **toprc** dans le répertoire **/home/student/.config/procps**. Dans le shell du terminal de gauche, vérifiez que le fichier **toprc** existe.

```
[student@serverb bin]$ ls -l /home/student/.config/procps/toprc
-rw-rw-r-- 1 student student 966 Feb 18 19:45 /home/student/.config/procps/toprc
```

- Dans le shell du terminal de droite, quittez **top**, puis redémarrez-le. Vérifiez que le nouvel affichage utilise la configuration enregistrée.

```
top - 00:58:21 up 43 min,  2 users,  load average: 0.29, 0.28, 0.20
Tasks: 105 total,   1 running, 104 sleeping,   0 stopped,   0 zombie
%Cpu(s): 11.0/1.8   13[||||||||||||||]                                ]
MiB Mem : 18.7/1829.0   [|||||||||||||||||||]                                ]
MiB Swap:  0.0/0.0      [                                         ]
```

6. Copiez le script **process101** dans un nouveau fichier appelé **process102**. Editez le script pour créer une plus grande charge artificielle sur le processeur. Augmentez la charge de cinquante mille à cent mille. Démarrer le processus **process102** en arrière-plan.

- 6.1. Dans le shell du terminal de gauche, utilisez la commande **cp** pour copier **process101** vers **process102**.

```
[student@serverb bin]$ cp process101 process102
```

- 6.2. Utilisez la commande **vim** pour éditer le script **process102**. Augmentez les problèmes d'addition de cinquante mille à cent mille. Entre en mode interactif au moyen du **i**. Tapez **:wq** pour enregistrer le fichier.

```
[student@serverb bin]$ vim process102
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 100000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

- 6.3. Démarrer le processus **process102** en arrière-plan.

```
[student@serverb bin]$ process102 &
[2] 20723
```

- 6.4. Utilisez la commande **jobs** pour vérifier que les deux processus s'exécutent en arrière-plan.

```
[student@serverb bin]$ jobs
[1]-  Running                  process101 &
[2]+  Running                  process102 &
```

7. Dans le shell de terminal de droite, vérifiez que le processus est en cours d'exécution et qu'il utilise le plus de ressources possible du processeur. La charge doit osciller entre 25 % et 35 %.

- 7.1. Dans le shell de terminal de droite, vérifiez que le processus est en cours d'exécution et qu'il utilise le plus de ressources possible du processeur. La charge doit osciller entre 25 % et 35 %.

```
top - 20:14:16 up 6:46, 2 users, load average: 0.58, 0.34, 0.18
Tasks: 112 total, 2 running, 110 sleeping, 0 stopped, 0 zombie
499 %Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0
st
500 MiB Mem : 1829.4 total, 1428.7 free, 179.0 used, 221.8 buff/cache
501 MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1492.1 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
20723 student 20 0 222448 3016 2764 S 24.7 0.2 0:53.28 process102
1180 student 20 0 222448 3056 2808 S 12.0 0.2 58:01.56 process101
...output omitted...
```

**NOTE**

Si vous ne voyez pas **process101** et **process102** en haut de la liste de processus, appuyez sur **Maj+p** pour vous assurer que top est trié en fonction de l'utilisation du processeur.

8. La charge moyenne est toujours inférieure à 1. Copiez **process101** vers un nouveau script appelé **process103**. Augmentez le nombre d'addition à huit cent mille. Démarrerez **process103** en arrière-plan. Confirmez que la charge moyenne est supérieure à 1. Cela peut prendre quelques minutes pour que la charge moyenne change.

- 8.1. Dans le terminal de droite, vérifiez que la charge moyenne est inférieure à 1.

```
top - 20:24:13 up 6:56, 2 users, load average: 0.43, 0.41, 0.29
...output omitted...
```

- 8.2. Dans le shell du terminal de gauche, utilisez la commande **cp** pour copier **process101** vers un nouveau script appelé **process103**.

```
[student@serverb bin]$ cp process101 process103
```

- 8.3. Dans le shell du terminal de gauche, utilisez la commande **vim** pour éditer le script **process103**. Augmentez le nombre d'addition à huit cent mille. Entrez en mode interactif au moyen de la touche **i**. Tapez **:wq** pour enregistrer le fichier.

```
[student@serverb bin]$ vim process103
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 800000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

- 8.4. Démarrerez **process103** en arrière-plan. L'utilisation du processeur oscille entre 60 % et 85 %.

```
[student@serverb bin]$ process103 &
[3] 22751
```

8.5. Vérifiez que les trois tâches sont exécutées en arrière-plan.

```
[student@serverb bin]$ jobs
[1]  Running                  process101 &
[2]- Running                  process102 &
[3]+ Running                  process103 &
```

8.6. Dans la fenêtre du terminal de droite, vérifiez que la charge moyenne est supérieure à 1.

```
top - 20:45:34 up 7:17, 2 users, load average: 1.10, 0.90, 0.64
```

9. Dans le shell du terminal de gauche, devenez l'utilisateur **root**. Suspendez le processus **process101**. Listez les tâches restantes. Observez que l'état du processus pour **process101** est maintenant **T**.

9.1. Exécutez la commande **su -** pour devenir l'utilisateur **root**. Le mot de passe est **redhat**.

```
[student@serverb bin]$ su -
Password: redhat
```

9.2. Utilisez la commande **pkill** avec l'option **-SIGSTOP** pour suspendre le processus **process101**.

```
[root@serverb ~]# pkill -SIGSTOP process101
```

9.3. Dans le shell du terminal de droite, vérifiez que **process101** n'est plus en cours d'exécution.

```
top - 20:52:01 up 7:24, 2 users, load average: 1.19, 1.19, 0.89
Tasks: 112 total, 2 running, 110 sleeping, 0 stopped, 0 zombie
499 %Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
500 MiB Mem : 1829.4 total, 1428.7 free, 179.0 used, 221.8 buff/cache
501 MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1492.1 avail Mem

 PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
24043 student    20    0 222448  2992  2744 R  66.1   0.2   6:59.50 process103
20723 student    20    0 222448  3016  2764 R  29.9   0.2  11:04.84 process102
...output omitted...
```

9.4. Dans le shell du terminal de gauche, utilisez la commande **ps JT** pour afficher les tâches restantes.

```
[root@serverb ~]# ps jT
  PPID  PID  PGID  SID TTY      TPGID STAT   UID    TIME COMMAND
...output omitted...
 27138 1180 1180 27138 pts/0      28558 T     1000   3:06 /bin/bash /home/student/
bin/process101
 27138 20723 20723 27138 pts/0      28558 R     1000   1:23 /bin/bash /home/student/
bin/process102
 27138 24043 24043 27138 pts/0      28558 R     1000   2:35 /bin/bash /home/student/
bin/process103
...output omitted...
```

Notez que **process101** a le statut **T**. Cela indique que le processus est actuellement suspendu.

10. Reprenez le processus **process101**.

- 10.1. Dans le shell du terminal de gauche, utilisez la commande **pkill** avec l'option **-SIGCONT** pour reprendre le processus **process101**.

```
[root@serverb ~]# pkill -SIGCONT process101
```

- 10.2. Dans le shell de terminal de droite, vérifiez que le processus est à nouveau en cours d'exécution.

```
top - 20:57:02 up 7:29, 2 users, load average: 1.14, 1.20, 0.99
Tasks: 112 total, 2 running, 110 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 1829.4 total, 1428.7 free, 179.0 used, 221.8 buff/cache
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1492.1 avail Mem

 PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
24043 student    20    0 222448 2992 2744 R  66.8  0.2  10:40.61 process103
20723 student    20    0 222448 3016 2764 S  24.9  0.2  12:25.10 process102
1180 student    20    0 222448 3056 2808 S  17.9  0.2  64:07.99 process101
```

11. Terminez **process101**, **process102** et **process103** au moyen de la ligne de commande. Vérifiez que les processus ne s'affichent plus dans **top**.

- 11.1. Dans le shell du terminal de gauche, utilisez la commande **pkill** pour terminer **process101**, **process102** et **process103**.

```
[root@serverb ~]# pkill process101
[root@serverb ~]# pkill process102
[root@serverb ~]# pkill process103
```

- 11.2. Dans le shell du terminal de droite, vérifiez que les processus n'apparaissent plus dans **top**.

```
top - 21:05:06 up 7:37, 2 users, load average: 1.26, 1.29, 1.12
Tasks: 112 total, 2 running, 110 sleeping, 0 stopped, 0 zombie
499 %Cpu(s): 0.0 us, 0.0 sy, 0.0 ni, 100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
```

```
500 MiB Mem : 1829.4 total, 1428.7 free, 179.0 used, 221.8 buff/cache
501 MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1492.1 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
 1 root 20 0 178536 13488 8996 S 0.0 0.7 0:01.21 systemd
 2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
 3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
...output omitted...
```

12. Dans le shell du terminal de gauche, déconnectez l'utilisateur root. Dans le shell du terminal de droite, arrêtez la commande **top**. Quittez serverb sur les deux fenêtres.

12.1. Utilisez la commande **exit** pour déconnecter l'utilisateur root.

```
[root@serverb ~]# exit
logout
[1]  Terminated                  process101
[2]  Terminated                  process102
[3]-  Terminated                  process103
```

12.2. Quittez toutes les deux fenêtres de terminal.

```
[student@serverb bin]$ exit
[student@workstation ~]$
```

12.3. Dans le shell du terminal de droite, appuyez sur **q** pour quitter **top**. Utilisez la commande **exit** pour vous déconnecter.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

Évaluation

À partir de **workstation**, exécutez le script lab processes-review grade pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab processes-review grade
```

Fin

Sur workstation, exécutez le script **lab processes-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab processes-review finish
```

L'atelier est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Un processus est une instance en cours d'exécution d'un programme exécutable. Les processus sont assignés à l'un des états suivants : running, sleeping, stopped ou zombie. La commande **ps** permet de lister les processus.
- Chaque terminal est sa propre session et peut disposer d'un processus en avant-plan et de processus en arrière-plan indépendants. La commande **jobs** affiche les processus dans une session de terminal.
- Un signal est une interruption logicielle qui signale des événements à un programme en cours d'exécution. Les commandes **kill**, **pkill** et **killall** utilisent des signaux pour contrôler les processus.
- La moyenne de charge est une estimation de l'occupation du système. Pour afficher les valeurs moyennes de charge, utilisez la commande **top**, **uptime** ou **w**.

CHAPITRE 9

CONTRÔLE DES SERVICES ET DES DÉMONS

PROJET

Contrôler et surveiller les services réseau et les démons système à l'aide de Systemd.

OBJECTIFS

- Répertorier les démons système et les services réseau démarrés par le service `systemd` et les unités de socket.
- Contrôler les démons système et les services réseau, en utilisant `systemctl`.

SECTIONS

- Identification des processus système démarrés automatiquement (et exercice guidé)
- Contrôle des services système (et exercice guidé)

ATELIER

Contrôle des services et des démons

IDENTIFICATION DES PROCESSUS SYSTÈME DÉMARRÉS AUTOMATIQUEMENT

OBJECTIFS

Après avoir terminé cette section, vous devez être en mesure d'afficher la liste des démons système et des services réseau démarrés par les unités de socket et le service `systemd`.

INTRODUCTION À `systemd`

Le démon `systemd` gère le démarrage pour Linux, y compris le démarrage et la gestion des services en général. Il active les ressources du système, les démons du serveur et d'autres processus à la fois lors du démarrage et sur un système en cours d'exécution.

Les démons sont des processus qui attendent ou sont exécutés en arrière-plan pour effectuer différentes tâches. En général, les démons s'exécutent automatiquement au démarrage et poursuivent leur exécution jusqu'à l'extinction ou jusqu'à ce qu'ils soient arrêtés manuellement. Par convention, le nom de nombreux programmes démons se terminent par la lettre **d**.

Un service au sens `systemd` fait souvent référence à un ou plusieurs démons, mais le démarrage ou l'arrêt d'un service peut aussi modifier l'état du système de manière ponctuelle, ce qui n'implique pas qu'un processus démon reste actif par la suite (appelé **Oneshot**).

Dans Red Hat Enterprise Linux, le premier processus qui commence (PID 1) est `systemd`. Quelques-unes des nouvelles fonctionnalités fournies par le service `systemd` incluent :

- des fonctionnalités de parallélisation (démarrage de plusieurs services simultanément), accélérant la vitesse de démarrage d'un système ;
- le démarrage à la demande des démons sans recours à un service distinct ;
- la gestion automatique des dépendances de service, ce qui peut éviter de longs délais d'attente. Par exemple, un service dépendant du réseau ne tentera pas de démarrer tant que le réseau ne sera pas disponible ;
- une méthode de suivi groupé des processus liés à l'aide des groupes de contrôle de Linux.

DESCRIPTION DES UNITÉS DE SERVICE

`systemd` utilise des *unités* pour gérer différents types d'objets. Parmi les types d'unité courants, on trouve notamment :

- Les unités de service présentent l'extension **.service** et représentent des services système. Ce type d'unité sert à démarrer les démons auxquels l'accès est fréquent, comme un serveur Web.
- Les unités de socket portent une extension **.socket** et représentent les sockets de communication inter-processus (IPC) que `systemd` doit contrôler. Si un client se connecte au socket, `systemd` va démarrer un démon et lui transmettre la connexion. Les unités de socket servent à lancer un service au démarrage et à lancer les services moins fréquemment utilisés à la demande.

- Les unités de chemin présentent l'extension **.path** et servent à retarder l'activation d'un service jusqu'à ce qu'une modification spécifique du système de fichiers se produise. Cette stratégie est couramment utilisée pour les services utilisant des répertoires spool comme un système d'impression.

La commande **systemctl** est utilisée pour gérer les unités. Par exemple, affichez les types d'unités disponibles avec la commande **systemctl -t help**.



IMPORTANT

Lorsque vous utilisez **systemctl**, vous pouvez abréger les noms d'unités, les entrées de l'arborescence de processus et les descriptions d'unités.

LISTE DES UNITÉS DE SERVICE

Vous utilisez la commande **systemctl** pour explorer l'état actuel du système. Par exemple, la commande suivante répertorie toutes les unités de service actuellement chargées, paginant la sortie à l'aide de **less**.

```
[root@host ~]# systemctl list-units --type=service
UNIT                           LOAD   ACTIVE SUB      DESCRIPTION
atd.service                     loaded active running Job spooling tools
auditd.service                  loaded active running Security Auditing Service
chronyd.service                 loaded active running NTP client/server
crond.service                   loaded active running Command Scheduler
dbus.service                     loaded active running D-Bus System Message Bus
...output omitted...
```

La sortie ci-dessus limite le type d'unité listé aux unités de service avec l'option **--type=service**. La sortie contient les colonnes suivantes :

Colonnes dans la sortie de commande **systemctl list-units**

UNIT

Le nom de l'unité de service.

LOAD

Indique si **systemd** a correctement analysé la configuration de l'unité et chargé l'unité dans la mémoire.

ACTIVE

État d'activation de haut niveau de l'unité. Cette information indique si l'unité a démarré avec succès ou non.

SUB

État d'activation de bas niveau de l'unité. Ces informations indiquent des informations plus détaillées sur l'unité. Les informations varient en fonction du type d'unité, de son état et de la manière dont l'unité est exécutée.

DESCRIPTION

Courte description de l'unité.

Par défaut, la commande **systemctl list-units --type=service** ne liste que les unités de service avec les états d'activation **active**. L'option **--all** liste toutes les unités de service,

queles que soient les états d'activation. Utilisez l'option **--state=** pour filtrer selon les valeurs des champs **LOAD**, **ACTIVE** ou **SUB**.

```
[root@host ~]# systemctl list-units --type=service --all
UNIT                      LOAD   ACTIVE   SUB      DESCRIPTION
atd.service                loaded  active   running  Job spooling tools
auditd.service              loaded  active   running  Security Auditing ...
auth-rpcgss-module.service loaded  inactive dead     Kernel Module ...
chronyd.service             loaded  active   running  NTP client/server
cpupower.service            loaded  inactive dead     Configure CPU power ...
crond.service               loaded  active   running  Command Scheduler
dbus.service                loaded  active   running  D-Bus System Message Bus
● display-manager.service   not-found inactive dead     display-manager.service
...output omitted...
```

La commande **systemctl** sans argument liste les unités à la fois chargées et actives.

```
[root@host ~]# systemctl
UNIT                      LOAD   ACTIVE   SUB      DESCRIPTION
proc-sys-fs-binfmt_misc.automount    loaded  active waiting  Arbitrary...
sys-devices-....device           loaded  active plugged  Virtio network...
sys-subsystem-net-devices-ens3.device loaded  active plugged  Virtio network...
...
-.mount                     loaded  active mounted  Root Mount
boot.mount                  loaded  active mounted  /boot
...
systemd-ask-password-plymouth.path  loaded  active waiting  Forward Password...
systemd-ask-password-wall.path    loaded  active waiting  Forward Password...
init.scope                  loaded  active running   System and Servi...
session-1.scope              loaded  active running   Session 1 of...
atd.service                 loaded  active running   Job spooling tools
auditd.service               loaded  active running   Security Auditing...
chronyd.service              loaded  active running   NTP client/server
crond.service                loaded  active running   Command Scheduler
...output omitted...
```

La commande **systemctl list-units** affiche les unités que le service **systemd** tente d'analyser et de charger en mémoire ; elle n'affiche pas les services installés mais non activés. Pour voir l'état de tous les fichiers d'unité installés, utilisez la commande **systemctl list-unit-files**. Par exemple :

```
[root@host ~]# systemctl list-unit-files --type=service
UNIT FILE                           STATE
arp-ethers.service                  disabled
atd.service                         enabled
auditd.service                      enabled
auth-rpcgss-module.service          static
autovt@.service                     enabled
blk-availability.service            disabled
...output omitted...
```

À la sortie de la commande **systemctl list-units-files**, les entrées valides pour le champ **STATE** sont **enabled**, **disabled**, **static** et **masked**.

AFFICHAGE DES ÉTATS DE SERVICE

Affichez le statut d'une unité spécifique avec **systemctl status name.type**. Si le type d'unité n'est pas fourni, **systemctl** affichera l'état d'une unité de service, s'il en existe une.

```
[root@host ~]# systemctl status sshd.service
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Thu 2019-02-14 12:07:45 IST; 7h ago
    Main PID: 1073 (sshd)
      CGroup: /system.slice/sshd.service
              └─1073 /usr/sbin/sshd -D ...
Feb 14 11:51:39 host.example.com systemd[1]: Started OpenSSH server daemon.
Feb 14 11:51:39 host.example.com sshd[1073]: Could not load host key: /etc/...
Feb 14 11:51:39 host.example.com sshd[1073]: Server listening on 0.0.0.0 ....
Feb 14 11:51:39 host.example.com sshd[1073]: Server listening on :: port 22.
Feb 14 11:53:21 host.example.com sshd[1270]: error: Could not load host k...
Feb 14 11:53:22 host.example.com sshd[1270]: Accepted password for root f...
...output omitted...
```

Cette commande affiche l'état actuel du service. La signification des champs est la suivante :

Informations sur l'unité de service

CHAMP	DESCRIPTION
Loaded	Indique si l'unité de service est chargée en mémoire.
Active	Indique si l'unité de service est en cours d'exécution et, le cas échéant, depuis combien de temps elle fonctionne.
Main PID	L'ID de processus principal du service, y compris le nom de la commande.
Status	Informations supplémentaires sur le service.

La sortie d'état comporte plusieurs mots-clés indiquant l'état du service :

États de service dans la sortie de systemctl

MOT-CLÉ	DESCRIPTION
loaded	Le fichier de configuration de l'unité a été traité.
active (running)	En cours d'exécution avec un ou plusieurs processus en cours.
active (exited)	Une configuration ponctuelle a été effectuée avec succès.
active (waiting)	En cours d'exécution, mais en attente d'un événement.

MOT-CLÉ	DESCRIPTION
inactive	Pas exécuté.
enabled	Est lancé au démarrage.
disabled	N'est pas défini pour être lancé au démarrage.
static	Ne peut pas être activé automatiquement, mais peut être démarré automatiquement par une unité activée.

**NOTE**

La commande **systemctl status NAME** remplace la commande **service NAME status** utilisée dans la version de Red Hat Enterprise Linux 6 et versions antérieures.

VÉRIFICATION DU STATUT D'UN SERVICE

La commande **systemctl** fournit des méthodes permettant de vérifier les états spécifiques d'un service. Par exemple, utilisez la commande suivante pour vérifier qu'une unité de service est actuellement active (running) :

```
[root@host ~]# systemctl is-active sshd.service
active
```

La commande renvoie l'état de l'unité de service, qui est généralement **active** ou **inactive**.

Exécutez la commande suivante pour vérifier si une unité de service est activée pour démarrer automatiquement au démarrage du système :

```
[root@host ~]# systemctl is-enabled sshd.service
enabled
```

La commande indique si l'unité de service est autorisée à démarrer au démarrage, généralement avec les valeurs **enabled** ou **disabled**.

Pour vérifier si l'unité a échoué au démarrage, exécutez la commande suivante :

```
[root@host ~]# systemctl is-failed sshd.service
active
```

La commande retourne **active** si elle s'exécute ou **failed** si une erreur s'est produite lors du démarrage. Si l'unité est arrêtée, elle retourne **unknown** ou **inactive**.

Pour lister toutes les unités défaillantes, exécutez la commande **systemctl --failed --type=service**.



RÉFÉRENCES

Pages de manuel `systemd(1)`, `systemd.unit(5)`, `systemd.service(5)`,
`systemd.socket(5)` et `systemctl(1)`

Pour plus d'informations, reportez-vous au chapitre *Managing services with systemd* de *Red Hat Enterprise Linux 8.0 Configuring basic system settings* sur
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/managing-services-with-systemd_configuring-basic-system-settings#managing-services-with-systemd_configuring-basic-system-settings

► EXERCICE GUIDÉ

IDENTIFICATION DES PROCESSUS SYSTÈME DÉMARRÉS AUTOMATIQUEMENT

Dans cet exercice, vous allez lister les unités de service installées et identifier les services actuellement activés et actifs sur un serveur.

RÉSULTATS

Vous devez pouvoir lister les unités de service installées et identifier les services actifs et activés sur le système.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab services-identify start**. La commande exécute un script de démarrage qui détermine si l'hôte, servera, est accessible sur le réseau.

```
[student@workstation ~]$ lab services-identify start
```

- 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis pour se connecter à servera.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Listez toutes les unités de service installées sur servera.

```
[student@servera ~]$ systemctl list-units --type=service
UNIT           LOAD   ACTIVE SUB   DESCRIPTION
atd.service    loaded  active running Job spooling tools
auditd.service loaded  active running Security Auditing Service
chronyd.service loaded  active running NTP client/server
crond.service  loaded  active running Command Scheduler
dbus.service   loaded  active running D-Bus System Message Bus
...output omitted...
```

Appuyez sur **q** pour quitter la commande.

- 3. Affichez la liste de toutes les unités de socket, actives et inactives sur servera.

```
[student@servera ~]$ systemctl list-units --type=socket --all
UNIT                  LOAD ACTIVE SUB     DESCRIPTION
dbus.socket           loaded active  running  D-Bus System Message Bus Socket
dm-event.socket       loaded active  listening Device-mapper event daemon FIFOs
lvm2-lvmpoold.socket loaded active  listening LVM2 poll daemon socket
...output omitted...
systemd-udevd-control.socket    loaded active  running  udev Control Socket
systemd-udevd-kernel.socket     loaded active  running  udev Kernel Socket

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.

12 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'.
```

- ▶ 4. Étudiez l'état du service chronyd. Ce service est utilisé pour la synchronisation de l'heure réseau (NTP).

4.1. Affichez l'état du service chronyd. Notez l'ID de processus de tout démon actif.

```
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor
  preset: enabled)
  Active: active (running) since Wed 2019-02-06 12:46:57 IST; 4h 7min ago
    Docs: man:chronyd(8)
          man:chrony.conf(5)
  Process: 684 ExecStartPost=/usr/libexec/chrony-helper update-daemon
            (code=exited, status=0/SUCCESS)
  Process: 673 ExecStart=/usr/sbin/chronyd $OPTIONS (code=exited, status=0/
            SUCCESS)
    Main PID: 680 (chronyd)
      Tasks: 1 (limit: 11406)
     Memory: 1.5M
        CPU: 0.000 CPU(s) (idle)
       CGroup: /system.slice/chronyd.service
                 └─680 /usr/sbin/chronyd

... jegui.ilt.example.com systemd[1]: Starting NTP client/server...
...output omitted...
... jegui.ilt.example.com systemd[1]: Started NTP client/server.
... servera.lab.example.com chronyd[680]: Source 172.25.254.254 offline
... servera.lab.example.com chronyd[680]: Source 172.25.254.254 online
... servera.lab.example.com chronyd[680]: Selected source 172.25.254.254
```

Appuyez sur **q** pour quitter la commande.

- 4.2. Confirmez que le démon répertorié est en cours d'exécution. Dans la commande ci-dessus, la sortie de l'ID de processus associé au service chronyd est 680. L'ID de processus peut différer sur votre système.

```
[student@servera ~]$ ps -p 680
 PID TTY      TIME CMD
 680 ?        00:00:00 chronyd
```

- 5. Étudiez l'état du service sshd. Ce service est utilisé pour la communication sécurisée par chiffrement entre les systèmes.

- 5.1. Déterminez si le service sshd est activé de manière à s'exécuter au démarrage du système.

```
[student@servera ~]$ systemctl is-enabled sshd
enabled
```

- 5.2. Déterminez si le service sshd est actif sans afficher toutes les informations sur son statut.

```
[student@servera ~]$ systemctl is-active sshd
active
```

- 5.3. Affichez l'état du service sshd.

```
[student@servera ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2019-02-06 12:46:58 IST; 4h 21min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
  Main PID: 720 (sshd)
     Tasks: 1 (limit: 11406)
    Memory: 5.8M
   CGroup: /system.slice/sshd.service
           └─720 /usr/sbin/sshd -D -oCiphers=aes256-gcm@openssh.com,
              chacha20-poly1305@openssh.com,aes256-ctr,
              aes256-cbc,aes128-gcm@openssh.com,aes128-ctr,
              aes128-cbc -oMACs= hmac-sha2-256-etm@openssh.com,hmac-sha>

...
... jegui.ilt.example.com systemd[1]: Starting OpenSSH server daemon...
... servera.lab.example.com sshd[720]: Server listening on 0.0.0.0 port 22.
... servera.lab.example.com systemd[1]: Started OpenSSH server daemon.
... servera.lab.example.com sshd[720]: Server listening on :: port 22.
... output omitted...
... servera.lab.example.com sshd[1380]: pam_unix(sshd:session): session opened for
 user student by (uid=0)
```

Appuyez sur **q** pour quitter la commande.

- 6. Affichez la liste des états (actifs ou inactifs) de toutes les unités de service.

```
[student@servera ~]$ systemctl list-unit-files --type=service
UNIT FILE                                STATE
arp-ethers.service                         disabled
atd.service                                enabled
auditd.service                            enabled
auth-rpcgss-module.service                static
autovt@.service                           enabled
blk-availability.service                 disabled
chrony-dnssrv@.service                  static
chrony-wait.service                      disabled
chronyd.service                           enabled
...output omitted...
```

Appuyez sur **q** pour quitter la commande.

► 7. Quittez servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

Fin

Sur workstation, exéutez le script **lab services-identify finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab services-identify finish
```

L'exercice guidé est maintenant terminé.

Contrôle des services du système

OBJECTIFS

Après avoir terminé cette section, vous serez en mesure de contrôler les démons du système et les services de réseau, à l'aide de **systemctl**.

DÉMARRAGE ET ARRÊT DE SERVICES

Les services doivent être arrêtés ou démarrés manuellement pour un certain nombre de raisons : le service doit peut-être être mis à jour ; le fichier de configuration doit peut-être être modifié ; ou un service peut avoir besoin d'être désinstallé ; ou un administrateur peut démarrer manuellement un service rarement utilisé.

Pour démarrer un service, vérifiez d'abord qu'il ne fonctionne pas avec la commande **systemctl status**. Ensuite, utilisez la commande **systemctl start** en tant qu'utilisateur root (en utilisant **sudo** si nécessaire). L'exemple ci-dessous montre comment démarrer le service `sshd.service` :

```
[root@host ~]# systemctl start sshd.service
```

Le service `systemd` recherche des fichiers `.service` pour la gestion de service dans les commandes en l'absence du type de service avec le nom du service. Ainsi, la commande ci-dessus peut être exécutée comme suit :

```
[root@host ~]# systemctl start sshd
```

Pour arrêter un service en cours, utilisez l'argument **stop** avec la commande **systemctl**. L'exemple ci-dessous montre comment arrêter le service `sshd.service` :

```
[root@host ~]# systemctl stop sshd.service
```

REDÉMARRAGE ET RECHARGEMENT DES SERVICES

Lors du redémarrage d'un service en cours d'exécution, le service est arrêté puis démarré. Lors du redémarrage du service, l'ID de processus change et un nouvel ID de processus est associé lors du démarrage. Pour redémarrer un service en cours, utilisez l'argument **restart** avec la commande **systemctl**. L'exemple ci-dessous montre comment redémarrer le service `sshd.service` :

```
[root@host ~]# systemctl restart sshd.service
```

Certains services peuvent recharger leurs fichiers de configuration sans nécessiter de redémarrage. Ce processus s'appelle une *recharge de service*. Le recharge d'un service ne modifie pas l'ID de processus associé aux divers processus de service. Pour recharger un service en cours, utilisez l'argument **reload** avec la commande **systemctl**. L'exemple ci-dessous montre comment recharger le service `sshd.service` après les changements de configuration :

```
[root@host ~]# systemctl reload sshd.service
```

CHAPITRE 9 | Contrôle des services et des démons

Si vous n'êtes pas sûr que le service dispose de la fonctionnalité permettant de recharger les modifications du fichier de configuration, utilisez l'argument **reload-or-restart** avec la commande **systemctl**. La commande recharge les modifications de configuration si la fonctionnalité de rechargement est disponible. Sinon, la commande redémarre le service pour implémenter les nouvelles modifications de configuration :

```
[root@host ~]# systemctl reload-or-restart sshd.service
```

LISTE DES DÉPENDANCES DES UNITÉS

Certains services nécessitent que d'autres services soient exécutés en premier, créant ainsi des dépendances sur les autres services. Les autres services ne sont pas démarrés au démarrage mais plutôt uniquement à la demande. Dans les deux cas, **systemd** et **systemctl** démarrent les services selon vos besoins, que ce soit pour résoudre la dépendance ou pour démarrer un service peu utilisé. Par exemple, si le service d'impression CUPS n'est pas en cours d'exécution et qu'un fichier est placé dans le répertoire de spool d'impression, le système démarre les démons ou les commandes liés au CUPS pour satisfaire la demande d'impression.

```
[root@host ~]# systemctl stop cups.service
Warning: Stopping cups, but it can still be activated by:
  cups.path
  cups.socket
```

Pour interrompre complètement les services d'impression sur un système, vous devez interrompre les trois unités. La désactivation du service désactive les dépendances.

La commande **systemctl list-dependencies** **UNIT** affiche un mappage hiérarchique des dépendances permettant de démarrer l'unité de service. Pour lister les dépendances inverses (les unités qui dépendent de l'unité spécifiée), utilisez l'option **--reverse** avec la commande.

```
[root@host ~]# systemctl list-dependencies sshd.service
sshd.service
• └─system.slice
• └─sshd-keygen.target
•   └─ssh-daemon@ecdsa.service
•   └─ssh-daemon@ed25519.service
•   └─ssh-daemon@rsa.service
•   └─sysinit.target
...output omitted...
```

MASQUAGE ET ANNULATION DU MASQUAGE DES SERVICES

Parfois, un système peut avoir différents services installés qui sont en conflit les uns avec les autres. Par exemple, il existe plusieurs méthodes pour gérer les serveurs de messagerie (**postfix** et **sendmail**, par exemple). Le masquage d'un service empêche un administrateur de démarrer accidentellement un service en conflit avec d'autres. Le masquage crée un lien dans les répertoires de configuration vers le fichier **/dev/null** qui empêche le service de démarrer.

```
[root@host ~]# systemctl mask sendmail.service
Created symlink /etc/systemd/system/sendmail.service → /dev/null.
```

```
[root@host ~]# systemctl list-unit-files --type=service
UNIT FILE STATE
...output omitted...
sendmail.service masked
...output omitted...
```

La tentative de démarrage d'une unité de service masquée échoue avec la sortie suivante :

```
[root@host ~]# systemctl start sendmail.service
Failed to start sendmail.service: Unit sendmail.service is masked.
```

Utilisez la commande **systemctl unmask** pour annuler le masquage de l'unité de service.

```
[root@host ~]# systemctl unmask sendmail
Removed /etc/systemd/system/sendmail.service.
```



IMPORTANT

Un service désactivé peut être démarré manuellement ou par d'autres fichiers d'unité, mais il ne démarre pas automatiquement au démarrage. Un service masqué ne démarre pas manuellement ou automatiquement.

ACTIVATION DE SERVICES DE FAÇON À CE QU'ILS S'EXÉCUTENT OU S'ARRÊTENT AU DÉMARRAGE

Le lancement d'un service sur un système en cours d'exécution ne garantit pas qu'il s'exécute automatiquement au redémarrage du système. De même, l'interruption d'un service sur un système en cours d'exécution ne l'empêche pas de redémarrer en même temps que le système. La création de liens dans les répertoires de configuration `systemd` permet au service de s'exécuter au démarrage. Les commandes **systemctl** créent et suppriment ces liens.

Pour exécuter un service au démarrage, utilisez la commande **systemctl enable**.

```
[root@root ~]# systemctl enable sshd.service
Created symlink /etc/systemd/system/multi-user.target.wants/sshd.service → /usr/
lib/systemd/system/sshd.service.
```

La commande ci-dessus crée un lien symbolique à partir du fichier d'unité de service, généralement dans le répertoire **/usr/lib/systemd/system**, vers l'emplacement sur le disque où `systemd` recherche les fichiers, à savoir dans le répertoire **/etc/systemd/system/TARGETNAME.target.wants**. L'activation d'un service ne démarre pas le service dans la session actuelle. Pour démarrer le service et lui permettre de s'exécuter automatiquement au démarrage, exécutez les commandes **systemctl start** et **systemctl enable**.

Pour désactiver le démarrage automatique du service, utilisez la commande suivante, qui supprime le lien symbolique créé lors de l'activation d'un service. Notez que la désactivation d'un service n'implique pas son arrêt.

```
[root@host ~]# systemctl disable sshd.service
Removed /etc/systemd/system/multi-user.target.wants/sshd.service.
```

Pour vérifier si le service est activé ou désactivé, utilisez la commande **systemctl is-enabled**.

RÉCAPITULATIF DES COMMANDES **systemctl**

Les services peuvent être démarrés et arrêtés sur un système en cours d'exécution, et activés ou désactivés pour être lancés automatiquement au démarrage.

Commandes de gestion de service utiles

TÂCHE	COMMANDÉ
Afficher des informations détaillées concernant l'état d'une unité.	systemctl status UNIT
Interrompre un service sur un système en cours d'exécution.	systemctl stop UNIT
Démarrer un service sur un système en cours d'exécution.	systemctl start UNIT
Redémarrer un service sur un système en cours d'exécution.	systemctl restart UNIT
Recharger le fichier de configuration d'un service en cours d'exécution.	systemctl reload UNIT
Désactiver complètement le lancement d'un service, que ce soit manuellement ou au démarrage.	systemctl mask UNIT
Rendre un service masqué disponible.	systemctl unmask UNIT
Configurer un service pour qu'il soit lancé au démarrage.	systemctl enable UNIT
Désactiver un service pour empêcher qu'il soit lancé au démarrage.	systemctl disable UNIT
Afficher la liste des unités nécessaires et requises par l'unité spécifiée.	systemctl list-dependencies UNIT



RÉFÉRENCES

Pages de manuel **systemd(1)**, **systemd.unit(5)**, **systemd.service(5)**, **systemd.socket(5)** et **systemctl(1)**

Pour plus d'informations, reportez-vous au chapitre *Managing services with systemd* de *Red Hat Enterprise Linux 8.0 Configuring basic system settings* sur https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/managing-services-with-systemd_configuring-basic-system-settings#managing-system-services_managing-services-with-systemd

► EXERCICE GUIDÉ

CONTRÔLE DES SERVICES DU SYSTÈME

Dans cet exercice, vous allez utiliser **systemctl** pour arrêter, démarrer, redémarrer, recharger, activer et désactiver un service géré par systemd.

RÉSULTATS

Vous devez pouvoir utiliser la commande **systemctl** pour contrôler les services gérés par systemd.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab services-control start**. La commande exécute un script de démarrage qui détermine si l'hôte, servera, est accessible sur le réseau. Le script s'assure également que les services sshd et chronyrd sont en cours d'exécution sur servera.

```
[student@workstation ~]$ lab services-control start
```

- ▶ 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. Exécutez les commandes **systemctl restart** et **systemctl reload** sur le service sshd. Observez les différents résultats de l'exécution de ces commandes.
- 2.1. Affichez l'état du service sshd. Notez l'ID de processus du démon sshd.

```
[student@servera ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2019-02-06 23:50:42 EST; 9min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
  Main PID: 759 (sshd)
     Tasks: 1 (limit: 11407)
    Memory: 5.9M
  ...output omitted...
```

CHAPITRE 9 | Contrôle des services et des démons

Appuyez sur **q** pour quitter la commande.

- 2.2. Redémarrez le service `sshd` et affichez son état. L'ID de processus du démon doit changer.

```
[student@servera ~]$ sudo systemctl restart sshd
[sudo] password for student: student
[student@servera ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2019-02-06 23:50:42 EST; 9min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
Main PID: 1132 (sshd)
  Tasks: 1 (limit: 11407)
  Memory: 5.9M
...output omitted...
```

Dans la sortie précédente, notez que l'ID de processus est passé de 759 à 1132 (sur votre système, les chiffres seront probablement différents). Appuyez sur **q** pour quitter la commande.

- 2.3. Rechargez le service `sshd` et affichez son état. L'ID de processus du démon ne doit pas changer et les connexions ne sont pas interrompues.

```
[student@servera ~]$ sudo systemctl reload sshd
[student@servera ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2019-02-06 23:50:42 EST; 9min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
Main PID: 1132 (sshd)
  Tasks: 1 (limit: 11407)
  Memory: 5.9M
...output omitted...
```

Appuyez sur **q** pour quitter la commande.

- 3. Vérifiez que le service `chronyd` est en cours d'exécution.

```
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor
  preset: enabled)
  Active: active (running) since Wed 2019-02-06 23:50:38 EST; 1h 25min ago
    Docs: man:chronyd(8)
...output omitted...
```

Appuyez sur **q** pour quitter la commande.

- 4. Arrêtez le service `chronyd` et affichez son état.

```
[student@servera ~]$ sudo systemctl stop chronyd
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor
  preset: enabled)
    Active: inactive (dead) since Thu 2019-02-07 01:20:34 EST; 44s ago
      ...output omitted...
... servera.lab.example.com chronyd[710]: System clock wrong by 1.349113 seconds,
adjustment started
... servera.lab.example.com systemd[1]: Stopping NTP client/server...
... servera.lab.example.com systemd[1]: Stopped NTP client/server.
```

Appuyez sur **q** pour quitter la commande.

- 5. Déterminez si le service `chronyd` est activé pour s'exécuter au démarrage du système.

```
[student@server ~]$ systemctl is-enabled chronyd
enabled
```

- 6. Redémarrez `servera`, puis affichez l'état du service `chronyd`.

```
[student@servera ~]$ sudo systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

Connectez-vous en tant qu'utilisateur étudiant sur `servera` et affichez le statut du serveur `chronyd`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor
  preset: enabled)
    Active: active (running) since Thu 2019-02-07 01:48:26 EST; 5min ago
      ...output omitted...
```

Appuyez sur **q** pour quitter la commande.

CHAPITRE 9 | Contrôle des services et des démons

- 7. Désactivez le service chronyd de façon à ce qu'il ne s'exécute pas au démarrage du système, puis affichez l'état du service.

```
[student@servera ~]$ sudo systemctl disable chronyd
[sudo] password for student:
Removed /etc/systemd/system/multi-user.target.wants/chronyd.service.
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; disabled; vendor
  preset: enabled)
  Active: active (running) since Thu 2019-02-07 01:48:26 EST; 5min ago
    ...output omitted...
```

Appuyez sur **q** pour quitter la commande.

- 8. Redémarrez servera, puis affichez l'état du service chronyd.

```
[student@servera ~]$ sudo systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

Connectez-vous en tant qu'utilisateur étudiant sur servera et affichez le statut du serveur chronyd.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; disabled; vendor
  preset: enabled)
  Active: inactive (dead)
    Docs: man:chronyd(8)
          man:chrony.conf(5)
```

- 9. Quittez servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

Fin

Sur workstation, exécutez le script **lab services-control finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab services-control finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

CONTRÔLE DES SERVICES ET DES DÉMONS

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez configurer plusieurs services pour qu'ils soient activés ou désactivés, en cours d'exécution ou arrêtés, en fonction d'une spécification qui vous est fournie.

RÉSULTATS

Vous devez pouvoir activer, désactiver, démarrer et arrêter des services.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur **student** à **workstation** avec le mot de passe **student**.

À partir de **workstation**, exécutez la commande **lab services-review start**. La commande exécute un script de démarrage qui détermine si l'hôte, **serverb**, est accessible sur le réseau. Le script garantit également que les services **psacct** et **rsyslog** sont configurés de manière appropriée sur **serverb**.

```
[student@workstation ~]$ lab services-review start
```

1. Sur **serverb**, démarrez le service **psacct**.
2. Configurez le service **psacct** pour qu'il s'exécute au démarrage du système.
3. Arrêtez le service **rsyslog**.
4. Configurez le service **rsylog** de façon à ce qu'il ne s'exécute pas au démarrage du système.
5. Redémarrez **serverb** avant d'évaluer l'atelier.

Évaluation

Sur **workstation**, exécutez le script **lab services-review grade** pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab services-review grade
```

Finish (Terminer)

Sur **workstation**, exécutez le script **lab services-review finish** pour mettre fin à cet atelier.

```
[student@workstation ~]$ lab services-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

CONTRÔLE DES SERVICES ET DES DÉMONS

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez configurer plusieurs services pour qu'ils soient activés ou désactivés, en cours d'exécution ou arrêtés, en fonction d'une spécification qui vous est fournie.

RÉSULTATS

Vous devez pouvoir activer, désactiver, démarrer et arrêter des services.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur **student** à **workstation** avec le mot de passe **student**.

À partir de **workstation**, exécutez la commande **lab services-review start**. La commande exécute un script de démarrage qui détermine si l'hôte, **serverb**, est accessible sur le réseau. Le script garantit également que les services **psacct** et **rsyslog** sont configurés de manière appropriée sur **serverb**.

```
[student@workstation ~]$ lab services-review start
```

- Sur **serverb**, démarrez le service **psacct**.

- Utilisez la commande **ssh** pour vous connecter à **serverb** en tant qu'utilisateur **student**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- Utilisez la commande **systemctl** pour vérifier le statut du service **psacct**. Remarquez que **psacct** est arrêté et désactivé pour démarrer au démarrage.

```
[student@serverb ~]$ systemctl status psacct
● psacct.service - Kernel process accounting
  Loaded: loaded (/usr/lib/systemd/system/psacct.service; disabled; vendor
  preset: disabled)
    Active: inactive (dead)
```

- Démarrez le service **psacct**.

```
[student@serverb ~]$ sudo systemctl start psacct  
[sudo] password for student: student  
[student@serverb ~]$
```

1.4. Vérifiez que le service `psacct` est en cours d'exécution.

```
[student@serverb ~]$ systemctl is-active psacct  
active
```

2. Configurez le service `psacct` pour qu'il s'exécute au démarrage du système.

2.1. Activez le service `psacct` pour qu'il s'exécute au démarrage du système.

```
[student@serverb ~]$ sudo systemctl enable psacct  
Created symlink /etc/systemd/system/multi-user.target.wants/psacct.service → /usr/  
lib/systemd/system/psacct.service.
```

2.2. Vérifiez que le service `psacct` est activé pour s'exécuter au démarrage du système.

```
[student@serverb ~]$ systemctl is-enabled psacct  
enabled
```

3. Arrêtez le service `rsyslog`.

3.1. Utilisez la commande `systemctl` pour vérifier le statut du service `rsyslog`. Notez que `rsyslog` est en cours d'exécution et activé pour démarrer au démarrage.

```
[student@serverb ~]$ systemctl status rsyslog  
● rsyslog.service - System Logging Service  
  Loaded: loaded (/usr/lib/systemd/system/rsyslog.service; enabled; vendor  
  preset: enabled)  
  Active: active (running) since Fri 2019-02-08 10:16:00 IST; 2h 34min ago  
    ...output omitted...
```

Appuyez sur `q` pour quitter la commande.

3.2. Arrêtez le service `rsyslog`.

```
[student@serverb ~]$ sudo systemctl stop rsyslog  
[sudo] password for student: student  
[student@serverb ~]$
```

3.3. Vérifiez que le service `rsyslog` est arrêté.

```
[student@serverb ~]$ systemctl is-active rsyslog  
inactive
```

4. Configurez le service `rsylog` de façon à ce qu'il ne s'exécute pas au démarrage du système.

4.1. Désactivez le service `rsylog` de façon à ce qu'il ne s'exécute pas au démarrage du système.

```
[student@serverb ~]$ sudo systemctl disable rsyslog
Removed /etc/systemd/system/syslog.service.
Removed /etc/systemd/system/multi-user.target.wants/rsyslog.service.
```

4.2. Vérifiez que le service `rsyslog` est désactivé de façon à ce qu'il ne s'exécute pas au démarrage du système.

```
[student@serverb ~]$ systemctl is-enabled rsyslog
disabled
```

5. Redémarrez `serverb` avant d'évaluer l'atelier.

```
[student@serverb ~]$ sudo systemctl reboot
Connection to serverb closed by remote host.
Connection to serverb closed.
[student@workstation ~]$
```

Évaluation

Sur `workstation`, exécutez le script `lab services-review grade` pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab services-review grade
```

Finish (Terminer)

Sur `workstation`, exécutez le script `lab services-review finish` pour mettre fin à cet atelier.

```
[student@workstation ~]$ lab services-review finish
```

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- `systemd` fournit une méthode permettant d'activer les ressources du système, les démons du serveur et d'autres processus à la fois lors du démarrage et sur un système en cours d'exécution.
- Utilisez **`systemctl`** pour démarrer, arrêter, recharger, activer et désactiver des services.
- Utilisez la commande **`systemctl status`** pour déterminer l'état des démons système et des services en réseau démarrés par `systemd`.
- La commande **`systemctl list-dependencies`** liste toutes les unités de service dont dépend une unité de service spécifique.
- `systemd` peut masquer une unité de service afin qu'elle ne s'exécute pas même pour satisfaire les dépendances.

CHAPITRE 10

CONFIGURATION ET SÉCURISATION DE SSH

PROJET

Configurer un service de ligne de commande sécurisé sur les systèmes distants à l'aide d'OpenSSH.

OBJECTIFS

- Se connecter à un système distant et exécuter des commandes à l'aide de **ssh**.
- Configurer une authentification par clé permettant à un compte d'utilisateur de se connecter à des systèmes distants de manière sécurisée, sans mot de passe.
- Restreindre les connexions directes en tant que root et désactiver l'authentification par mot de passe pour le service OpenSSH.

SECTIONS

- Accès à la ligne de commande distante via SSH (avec exercice guidé)
- Configuration de l'authentification par clé SSH (avec exercice guidé)
- Personnalisation de la configuration du service OpenSSH (avec exercice guidé)

ATELIER

Configuration et sécurisation de SSH

ACCÈS EN LIGNE DE COMMANDE DISTANTE VIA SSH

OBJECTIFS

Au terme de cette section, vous devez pouvoir vous connecter à un système distant et exécuter des commandes à l'aide de **ssh**.

QU'EST-CE QU'OPENSSH ?

OpenSSH implémente le protocole Secure Shell ou SSH dans les systèmes Red Hat Enterprise Linux. Le protocole SSH permet aux systèmes de communiquer de manière chiffrée et sécurisée sur un réseau non sécurisé.

Vous pouvez utiliser la commande **ssh** pour créer une connexion sécurisée à un système distant, s'authentifier en tant qu'utilisateur spécifique et obtenir une session shell interactive sur le système distant en tant qu'utilisateur. Vous pouvez également utiliser la commande **ssh** pour exécuter une commande individuelle sur le système distant sans exécuter un shell interactif.

EXEMPLES DE SECURE SHELL

La commande **ssh** suivante vous connecte sur le serveur distant `remotehost` en utilisant le même nom d'utilisateur que l'utilisateur local actuel. Dans cet exemple, le système distant vous invite à vous authentifier avec le mot de passe de cet utilisateur.

```
[user01@host ~]$ ssh remotehost
user01@remotehost's password: redhat
...output omitted...
[user01@remotehost ~]$
```

Utilisez la commande **exit** pour vous déconnecter du système distant.

```
[user01@remotehost ~]$ exit
logout
Connection to remotehost closed.
[user01@host ~]$
```

La prochaine commande **ssh** vous connecte sur le serveur distant `remotehost` en utilisant le nom d'utilisateur `user02`. Encore une fois, le système distant vous invite à vous authentifier avec le mot de passe de cet utilisateur.

```
[user01@host ~]$ ssh user02@remotehost
user02@remotehost's password: shadowman
...output omitted...
[user02@remotehost ~]$
```

Cette commande **ssh** exécuterait la commande **hostname** sur le système distant `remotehost` en tant qu'utilisateur `user02` sans accéder au shell interactif distant.

```
[user01@host ~]$ ssh user02@remotehost hostname
user02@remotehost's password: shadowman
remotehost.lab.example.com
[user01@host ~]$
```

Notez que la commande précédente affichait la sortie dans le terminal du système local.

IDENTIFICATION DES UTILISATEURS DISTANTS

La commande **w** affiche une liste des utilisateurs actuellement connectés à l'ordinateur. Cela s'avère particulièrement utile pour afficher les utilisateurs qui sont connectés à l'aide de **ssh**, à partir de quel emplacement et ce qu'ils font.

```
[user01@host ~]$ ssh user02@remotehost
user02@remotehost's password: redhat
[user01@remotehost ~]$ w
16:13:38 up 36 min, 1 user, load average: 0.00, 0.00, 0.00
USER      TTY      FROM          LOGIN@    IDLE     JCPU    PCPU WHAT
user02    pts/0    172.25.250.10  16:13      7:30    0.01s  0.01s -bash
user01    pts/1    172.25.250.10  16:24      3.00s  0.01s  0.00s w
[user02@remotehost ~]$
```

La sortie précédente montre que l'utilisateur **user02** s'est connecté au système sur le pseudo-terminal **0** à **16:13** aujourd'hui à partir de l'hôte avec l'adresse IP **172.25.250.10** et a été inactif lors d'une invite du shell pendant sept minutes et trente secondes. La sortie précédente montre également que l'utilisateur **user01** s'est connecté au système sur le pseudo-terminal **1** et a été inactif pendant les trois dernières secondes qui ont suivi l'exécution de la commande **w**.

CLÉS D'HÔTE SSH

SSH sécurise les communications par chiffrement à clé publique. Lorsqu'un client SSH se connecte à un serveur SSH, le serveur envoie une copie de sa clé publique au client avant la connexion du client. Cela permet de configurer le chiffrement sécurisé pour le canal de communication et d'authentifier le serveur auprès du client.

Lorsqu'un utilisateur utilise la commande **ssh** afin de se connecter à un serveur SSH, la commande vérifie s'il possède une copie de la clé publique de ce serveur dans ses fichiers hôtes locaux connus. L'administrateur système peut l'avoir préconfiguré dans **/etc/ssh/ssh_known_hosts**, ou l'utilisateur peut avoir un fichier **~/.ssh/known_hosts** dans son répertoire personnel qui contient la clé.

Si le client possède une copie de la clé, **ssh** il comparera la clé des fichiers hôtes connus pour ce serveur à celle qu'il a reçue. Si les clés ne correspondent pas, le client suppose que le trafic réseau vers le serveur pourrait être piraté ou que le serveur a été compromis, et demande à l'utilisateur de confirmer s'il souhaite ou non poursuivre la connexion.



NOTE

Définissez le paramètre **StrictHostKeyChecking** sur **yes** dans le fichier spécifique à l'utilisateur **~/.ssh/config** ou le fichier global **/etc/ssh/ssh_config** pour que la commande **ssh** annule toujours la connexion SSH si les clés publiques ne correspondent pas.

Si le client ne possède pas de copie de la clé publique dans ses fichiers hôtes connus, la commande **ssh** vous demandera si vous souhaitez quand même vous connecter. Dans ce cas, une copie de la clé publique sera sauvegardée dans votre fichier **~/.ssh/known_hosts** afin que l'identité du serveur puisse être automatiquement confirmée à l'avenir.

```
[user01@host ~]$ ssh newhost
The authenticity of host 'remotehost (172.25.250.12)' can't be established.
ECDSA key fingerprint is SHA256:qaS0PToLrqlCo2XGk1A0iY7CaP7aPKimerDoaUkv720.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'newhost,172.25.250.12' (ECDSA) to the list of known
hosts.
user01@newhost's password: redhat
...output omitted...
[user01@newhost ~]$
```

Gestion des clés des hôtes connus SSH

Si la clé publique d'un serveur est modifiée parce que la clé a été perdue en raison d'un dysfonctionnement du disque dur, ou qu'elle a été remplacée pour une raison légitime, vous devez modifier les fichiers hôtes connus pour vous assurer que l'entrée de l'ancienne clé publique est remplacée par une entrée avec la nouvelle clé publique afin de vous connecter sans erreur.

Les clés publiques sont stockées dans le fichier **/etc/ssh/ssh_known_hosts** et dans le fichier **~/.ssh/known_hosts** de chaque utilisateur du client SSH. Chaque clé est sur une ligne. Le premier champ est une liste de noms d'hôtes et d'adresses IP qui partagent cette clé publique. Le deuxième champ est l'algorithme de chiffrement de la clé. Le dernier champ est la clé elle-même.

```
[user01@host ~]$ cat ~/.ssh/known_hosts
remotehost,172.25.250.11 ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAIbm1zdHAyNTYAAABBB0sEi0e+F1aNT6jul8Ag5Nj
+RViZl0yE2w6iYUr+1fPt0IF0Ea0gFZ1LXM37VFTxdgFxHS3D5WhnIfb+68zf8+w=
```

Chaque serveur SSH distant que vous connectez enregistre sa clé publique dans le répertoire **/etc/ssh** dans des fichiers avec l'extension **.pub**.

```
[user01@remotehost ~]$ ls /etc/ssh/*key.pub
/etc/ssh/ssh_host_ecdsa_key.pub  /etc/ssh/ssh_host_ed25519_key.pub  /etc/ssh/
ssh_host_rsa_key.pub
```



NOTE

Il est recommandé d'ajouter des entrées correspondant aux fichiers **ssh_host_*key.pub** d'un serveur à votre fichier **~/.ssh/known_hosts** ou au fichier system-wide **/etc/ssh/ssh_known_hosts**.



RÉFÉRENCES

Pages de manuel `ssh(1)`, `w(1)` et `hostname(1)`

Pour plus d'informations, reportez-vous au chapitre *Using Secure Communications Between Two Systems With OpenSSH* du *Red Hat Enterprise Linux 8.0 Configuring and Managing Security Guide* à l'adresse

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/securing_networks/assembly_using-secure-communications-with-openssh-securing-networks#The_SSH_protocol_configuring-and-managing-security

► EXERCICE GUIDÉ

ACCÈS À LA LIGNE DE COMMANDE DISTANTE

Au cours de cet exercice, vous allez vous connecter à un système distant sous l'identité d'autres utilisateurs et exécuter des commandes.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Vous connecter à un système distant.
- Exécuter les commandes avec le shell sécurisé OpenSSH.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab ssh-access start` pour mettre fin à l'exercice. Ce script garantit que l'environnement est configuré correctement.

```
[student@workstation ~]$ lab ssh-access start
```

- 1. À partir de `workstation`, ouvrez une session SSH sur `servera` en tant que `student`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Ouvrez une session SSH sur `serverb` en tant que `student`. Acceptez la clé de l'hôte. Utilisez `student` comme mot de passe lorsque vous êtes invité à le fournir pour l'utilisateur `student` sur `serverb`.

```
[student@servera ~]$ ssh student@serverb
The authenticity of host 'serverb (172.25.250.11)' can't be established.
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWzA.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'serverb,172.25.250.11' (ECDSA) to the list of known
hosts.
student@serverb's password: student
...output omitted...
[student@serverb ~]$
```

La clé d'hôte est enregistrée dans le fichier `/home/student/.ssh/known_hosts` sur `servera` pour identifier `serverb` parce que l'utilisateur `student` a initié la connexion SSH à partir de `servera`. Si le fichier `/home/student/.ssh/known_hosts` n'existe pas

CHAPITRE 10 | Configuration et sécurisation de SSH

déjà, il est créé en tant que nouveau fichier avec la nouvelle entrée. La commande **ssh** ne s'exécute pas correctement s'il s'avère que l'hôte distant dispose d'une clé différente de la clé enregistrée.

- 3. Exécutez la commande **w** pour afficher les utilisateurs actuellement connectés à **serverb**.

```
[student@serverb ~]$ w
18:49:29 up 2:55, 1 user, load average: 0.00, 0.00, 0.00
USER      TTY      FROM          LOGIN@    IDLE    JCPU   PCPU WHAT
student  pts/0    172.25.250.10    18:33    0.00s  0.01s  0.00s w
```

La sortie précédente indique que l'utilisateur **student** s'est connecté au système à partir de l'hôte avec l'adresse IP **172.25.250.10**, qui correspond à **servera** dans le réseau de la classe.

**NOTE**

L'adresse IP d'un système identifie le système sur un réseau. Vous allez en apprendre davantage sur les adresses IP dans le chapitre suivant.

- 4. Quittez le shell de l'utilisateur **student** sur **serverb**.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@servera ~]$
```

- 5. Ouvrez une session SSH sur **serverb** en tant que **root**. Utilisez **redhat** comme mot de passe de l'utilisateur **root**.

```
[student@servera ~]$ ssh root@serverb
root@serverb's password: redhat
...output omitted...
[root@serverb ~]#
```

Notez que la commande **ssh** précédente ne vous a pas demandé d'accepter la clé hôte car elle a été trouvée parmi les hôtes connus. Si l'identité de **serverb** devait changer à un moment quelconque, OpenSSH vous inviterait à vérifier et à accepter la nouvelle clé d'hôte.

- 6. Exécutez la commande **w** pour afficher les utilisateurs actuellement connectés à **serverb**.

```
[root@serverb ~]# w
19:10:28 up 3:16, 1 user, load average: 0.00, 0.00, 0.00
USER      TTY      FROM          LOGIN@    IDLE    JCPU   PCPU WHAT
root      pts/0    172.25.250.10    19:09    1.00s  0.01s  0.00s w
```

La sortie précédente indique que l'utilisateur **root** s'est connecté au système à partir de l'hôte avec l'adresse IP **172.25.250.10**, qui correspond à **servera** dans le réseau de la classe.

- 7. Quittez le shell de l'utilisateur **root** sur **serverb**.

```
[root@serverb ~]# exit
logout
Connection to serverb closed.
[student@servera ~]$
```

- 8. Supprimez le fichier **/home/student/.ssh/known_hosts** sur **servera**. Ce qui amène **ssh** à perdre les identités enregistrées des systèmes distants.

```
[student@servera ~]$ rm /home/student/.ssh/known_hosts
```

Les clés d'hôte peuvent changer pour des raisons légitimes : la machine distante a peut-être été remplacée en raison d'une panne matérielle ou la machine distante a été réinstallée. On conseille en général de ne supprimer que l'entrée de la clé correspondant à l'hôte concerné dans le fichier **known_hosts**. Étant donné que ce fichier **known_hosts** particulier n'a qu'une seule entrée, vous pouvez supprimer l'ensemble du fichier.

- 9. Ouvrez une session SSH sur **serverb** en tant que **student**. Acceptez la clé d'hôte si un message vous y invite. Utilisez **student** comme mot de passe lorsque vous êtes invité à le fournir pour l'utilisateur **student** sur **serverb**.

```
[student@servera ~]$ ssh student@serverb
The authenticity of host 'serverb (172.25.250.11)' can't be established.
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWzA.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'serverb,172.25.250.11' (ECDSA) to the list of known
hosts.
student@serverb's password: student
...output omitted...
[student@serverb ~]$
```

Notez que la commande **ssh** a demandé votre confirmation pour accepter ou refuser la clé d'hôte, car elle n'a pas pu en trouver une pour l'hôte distant.

- 10. Quittez le shell de l'utilisateur **student** sur **serverb** et vérifiez la présence d'une nouvelle instance de **known_hosts** sur **servera**.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@servera ~]$ ls -l /home/student/.ssh/known_hosts
-rw-r--r--. 1 student student 183 Feb 1 20:26 /home/student/.ssh/known_hosts
```

- 11. Vérifiez que la nouvelle instance du fichier **known_hosts** dispose de la clé d'hôte de serverb.

```
[student@servera ~]$ cat /home/student/.ssh/known_hosts
serverb,172.25.250.11 ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAIBmlzdHAyNTYAAABBI9LEYEhwmu1rNqnbBPukH2Ba0/
QBAu9WbS4m03B3MIhhXWKFFNa/U1NjY8NDpEM+hkJe/GmnkcEYMLbCfd9nMA=
```

La sortie réelle variera.

- 12. Exécutez **hostname** à distance sur serverb sans accéder au shell interactif.

```
[student@servera ~]$ ssh student@serverb hostname
student@serverb's password: student
serverb.lab.example.com
```

La commande précédente affichait le nom d'hôte complet du système distant serverb.

- 13. Quittez le shell de l'utilisateur student sur servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
```

Finish (Terminer)

Sur workstation, exécutez **lab ssh-access finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab ssh-access finish
```

L'exercice guidé est maintenant terminé.

CONFIGURATION DE L'AUTHENTIFICATION PAR CLÉ SSH

OBJECTIFS

Au terme de cette section, vous serez en mesure de configurer un compte d'utilisateur pour utiliser l'authentification par clé afin de vous connecter à des systèmes distants de manière sécurisée, sans mot de passe.

AUTHENTIFICATION PAR CLÉ SSH

Vous pouvez configurer un serveur SSH pour vous permettre de vous authentifier sans mot de passe en utilisant l'authentification par clé. Il s'agit d'un système à clé privée-publique.

Pour ce faire, vous générerez une paire de fichiers de clés cryptographiques correspondants. L'une est une clé privée, l'autre une clé publique correspondante. Le fichier de la clé privée est utilisé comme identifiant d'authentification et, comme un mot de passe, il doit être tenu secret et sécurisé. La clé publique est copiée sur les systèmes auxquels l'utilisateur souhaite se connecter, et sert à vérifier la clé privée. La clé publique n'a pas besoin d'être secrète.

Vous mettez une copie de la clé publique dans votre compte sur le serveur. Lorsque vous essayez de vous connecter, le serveur SSH peut utiliser la clé publique pour lancer un défi qui ne peut être résolu correctement qu'en utilisant la clé privée. En conséquence, votre client **ssh** peut automatiquement authentifier votre connexion au serveur avec votre copie unique de la clé privée. Cela vous permet d'accéder à des systèmes en toute sécurité sans devoir saisir un mot de passe de manière interactive à chaque fois.

Génération de clés SSH

Pour créer une clé privée et une clé publique correspondante pour l'authentification, utilisez la commande **ssh-keygen**. Par défaut, vos clés privées et publiques sont enregistrées dans vos fichiers **~/.ssh/id_rsa** et **~/.ssh/id_rsa.pub**, respectivement.

```
[user@host ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa): Enter
Created directory '/home/user/.ssh'.
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
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:
SHA256:vxutUNPio3QDCyvkYm1oIx35hmMrHpPKWFdIYu3HV+w user@host.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
|           |
|           |
|   o o     o |
|   . = o   o .|
|   o + = S E .|
| ..o o + * + |
```

```
| .% 0 . + B .      |
|=*o0 . . + *      |
|++. . . +.      |
+---[SHA256]-----+
```

Si vous ne spécifiez pas de phrase de passe quand **ssh-keygen** vous y invite, la clé privée générée n'est pas protégée. Dans ce cas, toute personne possédant votre fichier de clé privée pourrait l'utiliser pour l'authentification. Si vous définissez une phrase de passe, vous devrez la saisir lorsque vous utiliserez la clé privée pour l'authentification. (Par conséquent, vous utiliseriez la phrase de passe de la clé privée plutôt que votre mot de passe sur l'hôte distant pour vous authentifier.)

Vous pouvez exécuter un programme d'aide appelé **ssh-agent** qui peut temporairement mettre en cache votre phrase de passe de la clé privée en mémoire au début de votre session pour obtenir une véritable authentification sans mot de passe. Cette question sera traitée ultérieurement dans cette section.

L'exemple suivant de la commande **ssh-keygen** montre la création de la clé privée protégée par phrase de passe de pair avec la clé publique.

```
[user@host ~]$ ssh-keygen -f .ssh/key-with-pass
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in .ssh/key-with-pass.
Your public key has been saved in .ssh/key-with-pass.pub.
The key fingerprint is:
SHA256:w3GGB7EyHURY4a0cNPKmhNKS7d1YsMVLvFZJ77VxAo user@host.lab.example.com
The key's randomart image is:
+---[RSA 2048]----+
|     . + = . o ... |
|     = B XEo o.   |
|     . o O X =.... |
|     = = = B = o.  |
|     = + * * S .  |
|     . + = o + .  |
|     + .           |
|                   |
|                   |
+---[SHA256]-----+
```

L'option **-f** avec la commande **ssh-keygen** détermine les fichiers dans lesquels les clés sont enregistrées. Dans l'exemple précédent, les clés privées et publiques sont enregistrées dans les fichiers **/home/user/.ssh/key-with-pass** et **/home/user/.ssh/key-with-pass.pub**, respectivement.



MISE EN GARDE

Lors de la génération d'autres paires de clés SSH, à moins que vous ne spécifiez un nom de fichier unique, vous êtes invité à donner la permission de remplacer les fichiers existants **id_rsa** et **id_rsa.pub**. Si vous écrasez les fichiers **id_rsa** et **id_rsa.pub** existants, vous devez remplacer l'ancienne clé publique par la nouvelle sur tous les serveurs SSH dotés de votre ancienne clé publique.

Une fois que les clés SSH ont été générées, elles sont stockées par défaut dans le répertoire **.ssh/** du répertoire personnel de l'utilisateur. Les modes de permission de la clé privée doivent être réglés sur 600 et ceux de la clé publique sur 644.

Partage de la clé publique

Avant de pouvoir utiliser l'authentification par clé, vous devez copier la clé publique sur le système cible. La commande **ssh-copy-id** copie la clé publique de la paire de clés SSH vers le système de destination. Si vous omettez le chemin d'accès au fichier de clé publique lors de l'exécution de **ssh-copy-id**, elle utilise le fichier **/home/user/.ssh/id_rsa.pub** par défaut.

```
[user@host ~]$ ssh-copy-id -i .ssh/key-with-pass.pub user@remotehost
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/user/.ssh/
id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys
user@remotehost's password: redhat
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'user@remotehost'"
and check to make sure that only the key(s) you wanted were added.
```

Une fois la clé publique transférée avec succès sur un système distant, vous pouvez vous authentifier auprès du système distant à l'aide de la clé privée correspondante tout en vous connectant au système distant via le protocole SSH. Si vous omettez le chemin d'accès au fichier de clé privée lors de l'exécution de la commande **ssh**, celle-ci utilise le fichier **/home/user/.ssh/
id_rsa** par défaut.

```
[user@host ~]$ ssh -i .ssh/key-with-pass user@remotehost
Enter passphrase for key '.ssh/key-with-pass': redhatpass
...output omitted...
[user@remotehost ~]$ exit
logout
Connection to remotehost closed.
[user@host ~]$
```

Utilisation de ssh-agent pour l'authentification non interactive

Si votre clé privée SSH est protégée par une phrase de passe, vous devez normalement entrer la phrase de passe pour utiliser la clé privée pour l'authentification. Cependant, vous pouvez utiliser un programme appelé **ssh-agent** pour mettre temporairement la phrase de passe en mémoire cache. Ensuite, chaque fois que vous utilisez SSH pour vous connecter à un autre système avec la clé privée, **ssh-agent** fournira automatiquement la phrase de passe pour vous. Ceci est pratique et peut améliorer la sécurité en offrant moins de possibilités à une personne « regardant par-dessus l'épaule » de vous voir taper la phrase de passe.

Selon la configuration de votre système local, si vous vous connectez initialement à l'environnement de travail graphique GNOME, le programme **ssh-agent** peut être démarré automatiquement et configuré pour vous.

CHAPITRE 10 | Configuration et sécurisation de SSH

Si vous vous connectez à une console texte, connectez-vous à l'aide de **ssh**, ou utilisez **sudo** ou **su**, vous aurez probablement besoin de démarrer **ssh-agent** manuellement pour cette session. Vous pouvez le faire avec la commande suivante :

```
[user@host ~]$ eval $(ssh-agent)  
Agent pid 10155  
[user@host ~]$
```



NOTE

Lorsque vous exécutez **ssh-agent**, il affiche certaines commandes shell. Vous devez exécuter ces commandes pour définir les variables d'environnement utilisées par des programmes tels que **ssh-add** pour communiquer avec eux. La commande **eval \$(ssh-agent)** démarre **ssh-agent** et exécute ces commandes pour définir automatiquement ces variables d'environnement pour cette session shell. Il affiche également le PID du processus **ssh-agent**.

Une fois que **ssh-agent** est en cours d'exécution, vous devez lui indiquer la phrase de passe de votre ou vos clés privées. Vous pouvez le faire avec la commande **ssh-add**.

Les commandes **ssh-add** suivantes ajoutent les clés privées à partir de **/home/user/.ssh/id_rsa** (le fichier par défaut) et les fichiers **/home/user/.ssh/key-with-pass**, respectivement.

```
[user@host ~]$ ssh-add  
Identity added: /home/user/.ssh/id_rsa (user@host.lab.example.com)  
[user@host ~]$ ssh-add .ssh/key-with-pass  
Enter passphrase for .ssh/key-with-pass: redhatpass  
Identity added: .ssh/key-with-pass (user@host.lab.example.com)
```

Après avoir ajouté avec succès les clés privées au processus **ssh-agent**, vous pouvez appeler une connexion SSH à l'aide de la commande **ssh**. Si vous utilisez un autre fichier de clé privée que le fichier **/home/user/.ssh/id_rsa** par défaut, alors vous devez utiliser l'option **-i** avec la commande **ssh** pour spécifier le chemin d'accès au fichier de clé privée.

L'exemple suivant de la commande **ssh** utilise le fichier de clé privée par défaut pour l'authentification auprès d'un serveur SSH.

```
[user@host ~]$ ssh user@remotehost  
Last login: Fri Apr  5 10:53:50 2019 from host.example.com  
[user@remotehost ~]$
```

L'exemple suivant de la commande **ssh** utilise le fichier de clé privée **/home/user/.ssh/key-with-pass** (non par défaut) pour l'authentification sur un serveur SSH. La clé privée dans l'exemple suivant a déjà été déchiffrée et ajoutée à son processus parent **ssh-agent**, de sorte que la commande **ssh** ne vous invite pas à déchiffrer la clé privée en saisissant sa phrase de passe de manière interactive.

```
[user@host ~]$ ssh -i .ssh/key-with-pass user@remotehost  
Last login: Mon Apr  8 09:44:20 2019 from host.example.com  
[user@remotehost ~]$
```

Lorsque vous vous déconnectez de la session qui a démarré **ssh-agent**, le processus se termine et vos phrases de passe pour vos clés privées sont effacées de la mémoire.



RÉFÉRENCES

Pages de manuel `ssh-keygen(1)`, `ssh-copy-id(1)`, `ssh-agent(1)`, `ssh-add(1)`

► EXERCICE GUIDÉ

CONFIGURATION DE L'AUTHENTIFICATION PAR CLÉ SSH

Au cours de cet exercice, vous allez configurer un utilisateur pour utiliser une authentification par clé pour SSH.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Générer une paire de clés SSH sans protection par phrase de passe.
- Générer une paire de clés SSH avec protection par phrase de passe.
- Authentifier en utilisant les deux clés SSH sans protection par phrase de passe et avec protection par phrase de passe.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab ssh-configure start` pour mettre fin à l'exercice. Ce script crée les comptes utilisateur nécessaires.

```
[student@workstation ~]$ lab ssh-configure start
```

- 1. À partir de `workstation`, ouvrez une session SSH sur `serverb` en tant que `student`.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 2. Utilisez la commande `su` pour basculer vers l'utilisateur `operator1` sur `serverb`. Utilisez `redhat` comme mot de passe pour `operator1`.

```
[student@serverb ~]$ su - operator1
Password: redhat
[operator1@serverb ~]$
```

- 3. Utilisez la commande `ssh-keygen` pour générer des clés SSH. N'entrez pas de phrase de passe.

```
[operator1@serverb ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/operator1/.ssh/id_rsa): Enter
Created directory '/home/operator1/.ssh'.
```

CHAPITRE 10 | Configuration et sécurisation de SSH

```
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /home/operator1/.ssh/id_rsa.
Your public key has been saved in /home/operator1/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:JainiQdnRosC+xXh0qsJQQLzBNULdb+jJbyrCZQBERI
operator1@serverb.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
|E+*+ooo .      |
|.= 0.0 o .     |
|o... = . . o   |
|+. + * . o    |
|+= X . S +   |
| + @ + = .    |
|. + = o       |
|.o . . . .   |
|o       o..    |
+---[SHA256]---+
```

- ▶ 4. Utilisez la commande **ssh-copy-id** pour envoyer la clé publique de la paire de clés SSH à operator1 sur servera. Utilisez redhat comme mot de passe pour operator1 sur servera.

```
[operator1@serverb ~]$ ssh-copy-id operator1@servera
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/
operator1/.ssh/id_rsa.pub"
The authenticity of host 'servera (172.25.250.10)' can't be established.
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWzA.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys
operator1@servera's password: redhat
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'operator1@servera'"
and check to make sure that only the key(s) you wanted were added.
```

- ▶ 5. Exécutez la commande **hostname** sur servera à distance à l'aide du protocole SSH sans accéder au shell interactif distant.

```
[operator1@serverb ~]$ ssh operator1@servera hostname
servera.lab.example.com
```

Notez que la commande **ssh** précédente ne vous a pas demandé de mot de passe car elle utilisait la clé privée sans phrase de passe sur la base de la clé publique exportée pour s'authentifier comme operator1 sur servera. Cette approche n'est pas sécurisée, car toute personne ayant accès au fichier de clé privée peut se connecter à servera en tant qu'operator1. L'alternative sécurisée consiste à protéger la clé privée avec une phrase de passe, ce qui constitue l'étape suivante.

- ▶ 6. Utilisez la commande **ssh-keygen** pour générer un autre jeu de clés SSH avec protection par phrase de passe. Enregistrez la clé sous le nom de **/home/operator1/.ssh/key2**. Utilisez **redhatpass** comme phrase de passe de la clé privée.



MISE EN GARDE

Si vous ne spécifiez pas le fichier dans lequel la clé est enregistrée, le fichier par défaut (**/home/user/.ssh/id_rsa**) est utilisé. Vous avez déjà utilisé le nom de fichier par défaut lors de la génération de clés SSH à l'étape précédente. Il est donc essentiel de spécifier un fichier autre que par défaut, sinon les clés SSH existantes seront écrasées.

```
[operator1@serverb ~]$ ssh-keygen -f .ssh/key2
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase): redhatpass
Enter same passphrase again: redhatpass
Your identification has been saved in .ssh/key2.
Your public key has been saved in .ssh/key2.pub.
The key fingerprint is:
SHA256:0CtCjfPm5QrbPBgqbEIWCCw5AI4oSlMEbgLrBQ1HWKI
operator1@serverb.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
|O=X*          |
|OB=.          |
|E*o.          |
|Booo .         |
|.= . o S      |
|+.o o          |
|+.oo+ o        |
|+o.0.+         |
|+. . =o.       |
+---[SHA256]---
```

- ▶ 7. Utilisez la commande **ssh-copy-id** pour envoyer la clé publique de la paire de clés protégée par phrase de passe à **operator1** sur **servera**.

```
[operator1@serverb ~]$ ssh-copy-id -i .ssh/key2.pub operator1@servera
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ".ssh/key2.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'operator1@servera'"
and check to make sure that only the key(s) you wanted were added.
```

Notez que la commande **ssh-copy-id** précédente ne vous a pas demandé de mot de passe parce qu'elle utilisait la clé publique de la clé privée sans phrase de passe que vous avez exportée vers **servera** à l'étape précédente.

CHAPITRE 10 | Configuration et sécurisation de SSH

- 8. Exécutez la commande **hostname** sur **servera** à distance avec SSH sans accéder au shell interactif distant. Utilisez **/home/operator1/.ssh/key2** comme fichier d'identité. Spécifiez **redhatpass** en tant que phrase de passe, que vous avez définie pour la clé privée à l'étape précédente.

```
[operator1@serverb ~]$ ssh -i .ssh/key2 operator1@servera hostname  
Enter passphrase for key '.ssh/key2': redhatpass  
servera.lab.example.com
```

Notez que la commande **ssh** précédente vous a demandé la phrase de passe que vous avez utilisée pour protéger la clé privée de la paire de clés SSH. Cette phrase de passe protège la clé privée. Si un attaquant parvient à accéder à la clé privée, il ne peut pas l'utiliser pour accéder à d'autres systèmes car la clé privée elle-même est protégée par une phrase de passe. La commande **ssh** utilise une phrase de passe différente de celle de **operator1** sur **servera**, obligeant les utilisateurs à connaître les deux.

Vous pouvez utiliser **ssh-agent**, comme à l'étape suivante, afin d'éviter la saisie interactive de la phrase de passe lors de la connexion avec SSH. L'utilisation de **ssh-agent** est à la fois plus pratique et plus sûre dans les situations où les administrateurs se connectent régulièrement à des systèmes distants.

- 9. Exécutez **ssh-agent** dans votre shell Bash et ajoutez la clé privée protégée par phrase de passe (**/home/operator1/.ssh/key2**) de la paire de clés SSH à la session shell.

```
[operator1@serverb ~]$ eval $(ssh-agent)  
Agent pid 21032  
[operator1@serverb ~]$ ssh-add .ssh/key2  
Enter passphrase for .ssh/key2: redhatpass  
Identity added: .ssh/key2 (operator1@serverb.lab.example.com)
```

La commande **eval** précédente a démarré **ssh-agent** et configuré cette session shell pour l'utiliser. Vous avez ensuite utilisé **ssh-add** pour fournir la clé privée déverrouillée à **ssh-agent**.

- 10. Exécutez la commande **hostname** sur **servera** à distance sans accéder à un shell interactif distant. Utilisez **/home/operator1/.ssh/key2** comme fichier d'identité.

```
[operator1@serverb ~]$ ssh -i .ssh/key2 operator1@servera hostname  
servera.lab.example.com
```

Notez que la commande **ssh** précédente ne vous a pas demandé de saisir la phrase de passe de manière interactive.

- 11. Ouvrez un autre terminal sur **workstation** et ouvrez une session SSH sur **serverb** en tant que **student**.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

- 12. Sur **serverb**, utilisez la commande **su** pour basculer vers **operator1** et appelez une connexion SSH à **servera**. Utilisez **/home/operator1/.ssh/key2** en tant que fichier d'identité pour vous authentifier à l'aide des clés SSH.

CHAPITRE 10 | Configuration et sécurisation de SSH

- 12.1. Utilisez la commande **su** pour basculer vers operator1. Utilisez redhat comme mot de passe pour operator1.

```
[student@serverb ~]$ su - operator1  
Password: redhat  
[operator1@serverb ~]$
```

- 12.2. Ouvrez une session SSH sur servera en tant que operator1.

```
[operator1@serverb ~]$ ssh -i .ssh/key2 operator1@servera  
Enter passphrase for key '.ssh/key2': redhatpass  
...output omitted...  
[operator1@servera ~]$
```

Notez que la commande **ssh** précédente vous a demandé de saisir la phrase de passe de manière interactive parce que vous n'avez pas appelé la connexion SSH à partir du shell que vous avez utilisé pour démarrer **ssh-agent**.

- 13. Quittez tous les shells que vous utilisez dans le second terminal.

- 13.1. Déconnectez-vous de servera.

```
[operator1@servera ~]$ exit  
logout  
Connection to servera closed.  
[operator1@serverb ~]$
```

- 13.2. Quittez les shells operator1 et student sur serverb pour revenir au shell de l'utilisateur student sur workstation.

```
[operator1@serverb ~]$ exit  
logout  
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

- 13.3. Fermez le deuxième terminal sur workstation.

```
[student@workstation ~]$ exit
```

- 14. Déconnectez-vous de serverb sur le premier terminal et terminez cet exercice.

- 14.1. À partir du premier terminal, quittez tous les shells de l'utilisateur operator1 sur serverb.

```
[operator1@serverb ~]$ exit  
logout  
[student@serverb ~]$
```

La commande **exit** vous a amené à quitter le shell de l'utilisateur **operator1**, mettant fin à la session shell où **ssh-agent** était actif, et à revenir au shell de l'utilisateur **student** sur **serverb**.

- 14.2. Quittez le shell de l'utilisateur **student** sur **serverb** pour revenir au shell de l'utilisateur **student** sur **workstation**.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

Fin

Sur **workstation**, exécutez **lab ssh-configure finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab ssh-configure finish
```

L'exercice guidé est maintenant terminé.

PERSONNALISATION DE LA CONFIGURATION DU SERVICE OPENSSH

OBJECTIFS

Au terme de cette section, vous serez en mesure de restreindre les connexions directes en tant que `root` et de désactiver l'authentification par mot de passe pour le service OpenSSH.

CONFIGURATION DU SERVEUR OPENSSH

Le service OpenSSH est fourni par un démon appelé `sshd`. Son fichier de configuration principal est `/etc/ssh/sshd_config`.

La configuration par défaut du serveur OpenSSH fonctionne bien. Cependant, vous souhaiterez peut-être apporter des modifications pour renforcer la sécurité de votre système. Il est possible que vous souhaitiez apporter deux modifications courantes. Vous voudrez peut-être interdire la connexion directe à distance au compte `root`, et peut-être interdire l'authentification par mot de passe (en faveur de l'authentification par clé privée SSH).

INTERDIRE À L'UTILISATEUR ROOT DE SE CONNECTER À L'AIDE DE SSH

C'est une bonne pratique d'interdire la connexion directe au compte de l'utilisateur `root` à partir des systèmes distants. Certains des risques d'autoriser la connexion directe en tant que `root` incluent :

- Le nom d'utilisateur `root` existe par défaut sur chaque système Linux. Dès lors, il suffit à un intrus potentiel de ne deviner que son mot de passe, au lieu d'une combinaison valide d'un nom d'utilisateur et d'un mot de passe. Cela facilite les choses pour un attaquant.
- L'utilisateur `root` dispose de priviléges illimités, sa compromission peut donc causer un maximum de dommages au système.
- Du point de vue de l'audit, il peut être difficile de savoir quel utilisateur autorisé s'est connecté en tant que `root` et a apporté des modifications. Si les utilisateurs doivent se connecter en tant qu'utilisateur régulier et basculer vers le compte `root`, cela génère un événement de journal qui peut être utilisé pour aider à assurer la responsabilité.

Le serveur OpenSSH utilise les paramètres de configuration **PermitRootLogin** dans fichier de configuration `/etc/ssh/sshd_config` pour autoriser ou interdire des utilisateurs de se connecter au système en tant que `root`.

```
PermitRootLogin yes
```

Avec le paramètre **PermitRootLogin** défini sur **yes**, comme c'est le cas par défaut, les utilisateurs sont autorisés à se connecter en tant que `root`. Pour éviter cela, définissez la valeur sur **no**. Sinon, pour empêcher l'authentification par mot de passe mais autoriser l'authentification par clé privée pour `root`, définissez le paramètre **PermitRootLogin** sur **without-password**.

Le serveur SSH (`sshd`) doit être recharge pour que les modifications prennent effet.

```
[root@host ~]# systemctl reload sshd
```

**IMPORTANT**

L'avantage d'utiliser la commande **systemctl reload sshd** est qu'elle indique à sshd de relire son fichier de configuration plutôt que de redémarrer complètement le service. Une commande **systemctl restart sshd** appliquerait également les modifications, mais arrêterait et redémarrerait également le service, en interrompant toutes les connexions SSH actives vers cet hôte.

INTERDICTION DE L'AUTHENTIFICATION PAR MOT DE PASSE À L'AIDE DE SSH

Le fait de n'autoriser les connexions à la ligne de commande à distance que par clé privée présente divers avantages :

- Les attaquants ne peuvent pas utiliser d'attaques en devinant des mots de passe pour pénétrer à distance dans des comptes connus sur le système.
- Avec les clés privées protégées par phrase de passe, un attaquant a besoin à la fois de la phrase de passe et d'une copie de la clé privée. Avec les mots de passe, un attaquant a juste besoin du mot de passe.
- En utilisant les clés privées protégées par phrase de passe en conjonction avec ssh-agent, la phrase de passe est exposée moins souvent puisqu'elle est saisie moins fréquemment, et la connexion est plus pratique pour l'utilisateur.

Le serveur OpenSSH utilise le paramètre **PasswordAuthentication** dans le fichier de configuration **/etc/ssh/sshd_config** pour contrôler si les utilisateurs peuvent utiliser l'authentification par mot de passe pour se connecter au système.

```
PasswordAuthentication yes
```

La valeur par défaut de **yes** pour le paramètre **PasswordAuthentication** dans le fichier de configuration **/etc/ssh/sshd_config** amène le serveur SSH à autoriser aux utilisateurs d'utiliser l'authentification par mot de passe lorsqu'ils se connectent. La valeur **no** pour **PasswordAuthentication** empêche les utilisateurs d'utiliser l'authentification par mot de passe.

Gardez à l'esprit que chaque fois que vous modifiez le fichier **/etc/ssh/sshd_config**, vous devez recharger le service sshd pour que les modifications prennent effet.

**IMPORTANT**

Rappelez-vous, si vous désactivez l'authentification par mot de passe pour **ssh**, vous devez avoir un moyen de vous assurer que le fichier **~/.ssh/authorized_keys** de l'utilisateur sur le serveur distant est rempli avec sa clé publique, afin qu'il puisse se connecter.

**RÉFÉRENCES**

Pages de manuel **ssh(1)** et **sshd_config(5)**

► EXERCICE GUIDÉ

PERSONNALISATION DE LA CONFIGURATION DU SERVICE OPENSSH

Au cours de cet exercice, vous allez restreindre les connexions directes en tant que `root` et l'authentification par mot de passe pour le service OpenSSH sur l'un de vos serveurs.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Désactiver les connexions directes en tant que `root` via `ssh`.
- Désactiver l'authentification par mot de passe permettant aux utilisateurs distants de se connecter via SSH.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab ssh-customize start` pour mettre fin à l'exercice. Ce script crée les comptes et les fichiers utilisateur nécessaires.

```
[student@workstation ~]$ lab ssh-customize start
```

- 1. À partir de `workstation`, ouvrez une session SSH sur `serverb` en tant que `student`.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 2. Utilisez la commande `su` pour basculer vers `operator2` sur `serverb`. Utilisez `redhat` comme mot de passe pour `operator2`.

```
[student@serverb ~]$ su - operator2
Password: redhat
[operator2@serverb ~]$
```

- 3. Utilisez la commande `ssh-keygen` pour générer des clés SSH. N'entrez aucune phrase de passe pour les clés.

```
[operator2@serverb ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/operator2/.ssh/id_rsa): Enter
```

CHAPITRE 10 | Configuration et sécurisation de SSH

```
Created directory '/home/operator2/.ssh'.
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /home/operator2/.ssh/id_rsa.
Your public key has been saved in /home/operator2/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:JainiQdnRosC+xXh0qsJQQLzBNULdb+jJbyrCZQBERI
operator1@serverb.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
|E+*+000 .      |
|.= o.o o.      |
|o.. = . . o.    |
|+. + * . o.    |
|+= X . S +     |
| + @ + = .     |
|. + = o        |
| .o . . . .    |
|o       o..     |
+---[SHA256]---+
```

- 4. Utilisez la commande **ssh-copy-id** pour envoyer la clé publique de la paire de clés SSH à operator2 sur servera. Utilisez redhat comme mot de passe pour operator2 sur servera.

```
[operator2@serverb ~]$ ssh-copy-id operator2@servera
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/
operator1/.ssh/id_rsa.pub"
The authenticity of host 'servera (172.25.250.10)' can't be established.
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWza.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys
operator2@servera's password: redhat
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'operator2@servera'"
and check to make sure that only the key(s) you wanted were added.
```

- 5. Confirmez que vous pouvez vous connecter à servera en tant que operator2 en utilisant les clés SSH.

- 5.1. Ouvrez une session SSH sur servera en tant que operator2.

```
[operator2@serverb ~]$ ssh operator2@servera
...output omitted...
[operator2@servera ~]$
```

Notez que la commande **ssh** précédente utilisait des clés SSH pour l'authentification.

- 5.2. Déconnectez-vous de servera.

```
[operator2@servera ~]$ exit  
logout  
Connection to servera closed.
```

- 6. Confirmez que vous pouvez vous connecter à **servera** en tant que **root** en utilisant **redhat** comme mot de passe.

- 6.1. Ouvrez une session SSH sur **servera** en tant que **root** en utilisant **redhat** comme mot de passe.

```
[operator2@serverb ~]$ ssh root@servera  
root@servera's password: redhat  
...output omitted...  
[root@servera ~]#
```

Notez que la commande **ssh** précédente utilisait le mot de passe de l'utilisateur **root** pour l'authentification, car les clés SSH n'existent pas pour l'utilisateur **root**.

- 6.2. Déconnectez-vous de **servera**.

```
[root@servera ~]# exit  
logout  
Connection to servera closed.  
[operator2@serverb ~]$
```

- 7. Confirmez que vous pouvez vous connecter à **servera** en tant que **operator3** en utilisant **redhat** comme mot de passe.

- 7.1. Ouvrez une session SSH sur **servera** en tant que **operator3** en utilisant **redhat** comme mot de passe.

```
[operator2@serverb ~]$ ssh operator3@servera  
operator3@servera's password: redhat  
...output omitted...  
[operator3@servera ~]$
```

Notez que la commande **ssh** précédente utilisait le mot de passe de **operator3** pour l'authentification, car les clés SSH n'existent pas pour **operator3**.

- 7.2. Déconnectez-vous de **servera**.

```
[operator3@servera ~]# exit  
logout  
Connection to servera closed.  
[operator2@serverb ~]$
```

- 8. Configurez **sshd** sur **servera** pour empêcher les utilisateurs de se connecter en tant que **root**. Utilisez **redhat** comme mot de passe du superutilisateur lorsque nécessaire.

- 8.1. Ouvrez une session SSH sur **servera** en tant que **operator2** au moyen des clés SSH.

```
[operator2@serverb ~]$ ssh operator2@servera  
...output omitted...  
[operator2@servera ~]$
```

- 8.2. Sur servera, basculez vers root. Utilisez redhat comme mot de passe de l'utilisateur root.

```
[operator2@servera ~]$ su -  
Password: redhat  
[root@servera ~]#
```

- 8.3. Définissez **PermitRootLogin** sur no dans **/etc/ssh/sshd_config** et rechargez sshd. Vous pouvez utiliser **vim /etc/ssh/sshd_config** pour éditer le fichier de configuration de sshd.

```
...output omitted...  
PermitRootLogin no  
...output omitted...  
[root@servera ~]# systemctl reload sshd
```

- 8.4. Ouvrez un autre terminal sur workstation et ouvrez une session SSH sur serverb en tant que operator2. À partir de serverb, essayez de vous connecter à servera en tant que root. Cela devrait échouer parce que vous avez désactivé la connexion de l'utilisateur root via SSH à l'étape précédente.



NOTE

Pour des raisons pratiques, la connexion sans mot de passe est déjà configurée entre workstation et serverb dans l'environnement de formation.

```
[student@workstation ~]$ ssh operator2@serverb  
...output omitted...  
[operator2@serverb ~]$ ssh root@servera  
root@servera's password: redhat  
Permission denied, please try again.  
root@servera's password: redhat  
Permission denied, please try again.  
root@servera's password: redhat  
root@servera: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
```

Par défaut, la commande **ssh** tente de s'authentifier à l'aide de l'authentification par clé d'abord, puis, si cela échoue, à l'aide de l'authentification par mot de passe.

- 9. Configurez sshd sur servera pour permettre aux utilisateurs de s'authentifier en utilisant uniquement des clés SSH, plutôt que leurs mots de passe.

- 9.1. Retournez sur le premier terminal dont le shell de l'utilisateur root est actif sur servera. Définissez **PasswordAuthentication** sur **no** dans **/etc/ssh/sshd_config** et rechargez sshd. Vous pouvez utiliser **vim /etc/ssh/sshd_config** pour éditer le fichier de configuration de sshd.

```
...output omitted...
PasswordAuthentication no
...output omitted...
[root@servera ~]# systemctl reload sshd
```

- 9.2. Accédez au deuxième terminal dont le shell de l'utilisateur **operator2** est actif sur **serverb** et essayez de vous connecter à **servera** en tant que **operator3**. Cela devrait échouer car les clés SSH ne sont pas configurées pour **operator3**, et le service **sshd** sur **servera** n'autorise pas l'utilisation de mots de passe pour l'authentification.

```
[operator2@serverb ~]$ ssh operator3@servera
operator3@servera: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
```



NOTE

Pour plus de précision, vous pouvez utiliser les options explicites **-o PubkeyAuthentication=no** et **-o PasswordAuthentication=yes** avec la commande **ssh**. Cela vous permet de remplacer les valeurs par défaut de la commande **ssh** et de déterminer avec confiance que la commande précédente échoue en fonction des paramètres que vous avez définis dans **/etc/ssh/sshd_config** à l'étape précédente.

- 9.3. Retournez sur le premier terminal dont le shell de l'utilisateur **root** est actif sur **servera**. Vérifiez que **PubkeyAuthentication** est activé dans **/etc/ssh/sshd_config**. Vous pouvez utiliser **vim /etc/ssh/sshd_config** pour afficher le fichier de configuration de **sshd**.

```
...output omitted...
#PubkeyAuthentication yes
...output omitted...
```

Notez que la ligne **PubkeyAuthentication** figure sous forme de commentaires. Toute ligne sous forme de commentaires dans ce fichier utilise la valeur par défaut. Les lignes sous forme de commentaires indiquent les valeurs par défaut d'un paramètre. L'authentification par clé publique de SSH est active par défaut, comme l'indique la ligne en commentaires.

- 9.4. Revenez au deuxième terminal dont le shell de l'utilisateur **operator2** est actif sur **serverb** et essayez de vous connecter à **servera** en tant que **operator2**. Cela devrait réussir car les clés SSH sont configurées pour **operator2** afin de se connecter à **servera** à partir de **serverb**.

```
[operator2@serverb ~]$ ssh operator2@servera
...output omitted...
[operator2@servera ~]$
```

- 9.5. À partir du deuxième terminal, quittez le shell de l'utilisateur **operator2** à la fois sur **servera** et **serverb**.

```
[operator2@servera ~]$ exit  
logout  
Connection to servera closed.  
[operator2@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

9.6. Fermez le deuxième terminal sur workstation.

```
[student@workstation ~]$ exit
```

9.7. À partir du premier terminal, quittez le shell de l'utilisateur **root** sur **servera**.

```
[root@servera ~]# exit  
logout
```

9.8. À partir du premier terminal, quittez le shell de l'utilisateur **operator2** à la fois sur **servera** et **serverb**.

```
[operator2@servera ~]$ exit  
logout  
Connection to servera closed.  
[operator2@serverb ~]$ exit  
logout  
[student@serverb ~]$
```

9.9. Déconnectez-vous de **serverb** et retournez sur le shell de l'utilisateur **student** sur **workstation**.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

Finish (Terminer)

Sur **workstation**, exéutez **lab ssh-customize finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab ssh-customize finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

CONFIGURATION ET SÉCURISATION DE SSH

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez configurer l'authentification par clé pour les utilisateurs, et désactiver la connexion directe en tant que `root` et l'authentification par mot de passe pour le service OpenSSH sur l'un de vos serveurs.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Authentifiez-vous à l'aide de clés SSH.
- Empêchez les utilisateurs de se connecter directement en tant que `root` via `ssh`.
- Empêchez les utilisateurs de se connecter au système à l'aide de l'authentification par mot de passe SSH.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab ssh-review start` pour mettre fin à l'exercice. Ce script crée les comptes et les fichiers utilisateur nécessaires.

```
[student@workstation ~]$ lab ssh-review start
```

1. À partir de `workstation`, ouvrez une session SSH sur `servera` en tant que `student`.
2. Utilisez la commande `su` pour basculer vers `production1` sur `servera`.
3. Utilisez la commande `ssh-keygen` pour générer des clés SSH sans phrase de passe pour `production1` sur `servera`.
4. Utilisez la commande `ssh-copy-id` pour envoyer la clé publique de la paire de clés SSH à `production1` sur `serverb`.
5. Vérifiez que `production1` peut se connecter avec succès à `serverb` au moyen des clés SSH.
6. Configurez `sshd` sur `serverb` pour empêcher les utilisateurs de se connecter en tant que `root`. Utilisez `redhat` comme mot de passe du superutilisateur.
7. Configurez `sshd` sur `serverb` pour permettre aux utilisateurs de s'authentifier en utilisant uniquement des clés SSH, plutôt que leurs mots de passe.

Évaluation

Sur workstation, exécutez la commande **lab ssh-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab ssh-review grade
```

Finish (Terminer)

Sur workstation, exécutez **lab ssh-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab ssh-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

CONFIGURATION ET SÉCURISATION DE SSH

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez configurer l'authentification par clé pour les utilisateurs, et désactiver la connexion directe en tant que `root` et l'authentification par mot de passe pour le service OpenSSH sur l'un de vos serveurs.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Authentifiez-vous à l'aide de clés SSH.
- Empêchez les utilisateurs de se connecter directement en tant que `root` via `ssh`.
- Empêchez les utilisateurs de se connecter au système à l'aide de l'authentification par mot de passe SSH.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab ssh-review start` pour mettre fin à l'exercice. Ce script crée les comptes et les fichiers utilisateur nécessaires.

```
[student@workstation ~]$ lab ssh-review start
```

- À partir de `workstation`, ouvrez une session SSH sur `servera` en tant que `student`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- Utilisez la commande `su` pour basculer vers `production1` sur `servera`.

```
[student@servera ~]$ su - production1
Password: redhat
[production1@servera ~]$
```

- Utilisez la commande `ssh-keygen` pour générer des clés SSH sans phrase de passe pour `production1` sur `servera`.

```
[production1@servera ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/production1/.ssh/id_rsa): Enter
```

CHAPITRE 10 | Configuration et sécurisation de SSH

```
Created directory '/home/production1/.ssh'.
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /home/production1/.ssh/id_rsa.
Your public key has been saved in /home/production1/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:CsWCAmW0g5qaJujLzIAcengNj3u21kbrPP4Ys13PXCA
    production1@servera.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
| ..o          |
| o+ . .        |
| = o . o      |
| .+ o          |
| o... . E .    |
| *o.= . . .    |
| Xo+ +oo.. .   |
| Oo . +==+ + . |
| *o+o=*o. +    |
+---[SHA256]---+
```

4. Utilisez la commande **ssh-copy-id** pour envoyer la clé publique de la paire de clés SSH à **production1** sur **serverb**.

```
[production1@servera ~]$ ssh-copy-id production1@serverb
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/
production1/.ssh/id_rsa.pub"
The authenticity of host 'serverb (172.25.250.11)' can't be established.
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWzA.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys
production1@serverb's password: redhat
Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'production1@serverb'"
and check to make sure that only the key(s) you wanted were added.
```

5. Vérifiez que **production1** peut se connecter avec succès à **serverb** au moyen des clés SSH.

```
[production1@servera ~]$ ssh production1@serverb
...output omitted...
[production1@serverb ~]$
```

6. Configurez **sshd** sur **serverb** pour empêcher les utilisateurs de se connecter en tant que **root**. Utilisez **redhat** comme mot de passe du superutilisateur.

- 6.1. Utilisez la commande **su -** pour basculer vers **root** sur **serverb**.

```
[production1@serverb ~]$ su -
Password: redhat
[root@serverb ~]#
```

- 6.2. Définissez `PermitRootLogin` sur `no` dans `/etc/ssh/sshd_config` et rechargez `sshd`. Vous pouvez utiliser `vim /etc/ssh/sshd_config` pour éditer le fichier de configuration de `sshd`.

```
...output omitted...
PermitRootLogin no
...output omitted...
[root@serverb ~]# systemctl reload sshd.service
```

- 6.3. Ouvrez un autre terminal sur `workstation` et ouvrez une session SSH sur `servera` en tant que `production1`. À partir de `servera`, essayez de vous connecter à `serverb` en tant que `root`. Cela devrait échouer parce que vous avez désactivé la connexion de l'utilisateur `root` via SSH à l'étape précédente.



NOTE

Pour des raisons pratiques, la connexion sans mot de passe est déjà configurée entre `workstation` et `servera` dans l'environnement de formation.

```
[student@workstation ~]$ ssh production1@servera
...output omitted...
[production1@servera ~]$ ssh root@serverb
root@serverb's password: redhat
Permission denied, please try again.
root@serverb's password: redhat
Permission denied, please try again.
root@serverb's password: redhat
root@serverb: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
[production1@servera ~]$
```

La commande `ssh` précédente a été renvoyée après trois tentatives infructueuses de connexion à `servera` en tant que `root`. Par défaut, la commande `ssh` préfère utiliser les clés SSH pour l'authentification, mais si elle ne trouve pas les clés nécessaires de l'utilisateur, elle demande le mot de passe de l'utilisateur pour l'authentification.

7. Configurez `sshd` sur `serverb` pour permettre aux utilisateurs de s'authentifier en utilisant uniquement des clés SSH, plutôt que leurs mots de passe.
- 7.1. Retournez sur le premier terminal dont le shell de l'utilisateur `root` est actif sur `serverb`. Définissez `PasswordAuthentication` sur `no` dans `/etc/ssh/sshd_config` et rechargez `sshd`. Vous pouvez utiliser `vim /etc/ssh/sshd_config` pour éditer le fichier de configuration de `sshd`.

```
...output omitted...
PasswordAuthentication no
...output omitted...
[root@serverb ~]# systemctl reload sshd
```

- 7.2. Accédez au deuxième terminal dont le shell de l'utilisateur `production1` est actif sur `servera` et essayez de vous connecter à `serverb` en tant que `production2`. Cela devrait échouer car les clés SSH ne sont pas configurées pour `production2`, et le service `sshd` sur `serverb` n'autorise pas l'utilisation de mots de passe pour l'authentification.

```
[production1@servera ~]$ ssh production2@serverb  
production2@serverb: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
```



NOTE

Pour plus de précision, vous pouvez utiliser les options explicites `-o` **PubkeyAuthentication=no** et `-o PasswordAuthentication=yes` avec la commande `ssh`. Cela vous permet de remplacer les valeurs par défaut de la commande `ssh` et d'établir avec confiance que la commande précédente échoue en fonction des paramètres que vous avez définis dans `/etc/ssh/sshd_config` à l'étape précédente.

- 7.3. Retournez sur le premier terminal dont le shell de l'utilisateur `root` est actif sur `serverb`. Vérifiez que **PubkeyAuthentication** est activé dans `/etc/ssh/sshd_config`. Vous pouvez utiliser `vim /etc/ssh/sshd_config` pour afficher le fichier de configuration de `sshd`.

```
...output omitted...  
#PubkeyAuthentication yes  
...output omitted...
```

Notez que la ligne `PubkeyAuthentication` figure sous forme de commentaires. Toute ligne sous forme de commentaires dans ce fichier utilise la valeur par défaut. Les lignes sous forme de commentaires indiquent les valeurs par défaut d'un paramètre. L'authentification par clé publique de SSH est active par défaut, comme l'indique la ligne en commentaires.

- 7.4. Retournez vers le deuxième terminal dont le shell de l'utilisateur `production1` est actif sur `servera` et essayez de vous connecter à `serverb` en tant que `production1`. Cela devrait réussir car les clés SSH sont configurées pour `production1` afin de se connecter à `serverb` à partir de `servera`.

```
[production1@servera ~]$ ssh production1@serverb  
...output omitted...  
[production1@serverb ~]$
```

- 7.5. À partir du deuxième terminal, quittez le shell de l'utilisateur `production1` à la fois sur `serverb` et `servera`.

```
[production1@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[production1@servera ~]$ exit  
logout  
[student@workstation ~]$
```

7.6. Fermez le deuxième terminal sur workstation.

```
[student@workstation ~]$ exit
```

7.7. À partir du premier terminal, quittez le shell de l'utilisateur root sur serverb.

```
[root@serverb ~]# exit  
logout
```

7.8. À partir du premier terminal, quittez le shell de l'utilisateur production1 à la fois sur serverb et servera.

```
[production1@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[production1@servera ~]$ exit  
logout  
[student@servera ~]$
```

7.9. Déconnectez-vous de servera et retournez sur le shell de l'utilisateur student sur workstation.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Évaluation

Sur workstation, exécutez la commande **lab ssh-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab ssh-review grade
```

Finish (Terminer)

Sur workstation, exécutez **lab ssh-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab ssh-review finish
```

L'atelier est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- La commande **ssh** permet aux utilisateurs d'accéder en toute sécurité aux systèmes distants à l'aide du protocole SSH.
- Un système client stocke les identités de serveurs distants dans `~/.ssh/known_hosts` et `/etc/ssh/ssh_known_hosts`.
- SSH prend en charge l'authentification par mot de passe et par clé.
- La commande **ssh-keygen** génère une paire de clés SSH pour l'authentification. La commande **ssh-copy-id** exporte la clé publique vers des systèmes distants.
- Le service sshd implémente le protocole SSH dans les systèmes Red Hat Enterprise Linux.
- Il est recommandé de configurer sshd pour désactiver les connexions à distance en tant que `root` et d'exiger une authentification par clé publique plutôt que par mot de passe.

CHAPITRE 11

ANALYSE ET STOCKAGE DES JOURNAUX

PROJET

Localiser et analyser avec précision les journaux d'événements système à des fins de diagnostic.

OBJECTIFS

- Décrire l'architecture de journalisation de base utilisée par Red Hat Enterprise Linux pour enregistrer des événements.
- Interpréter les événements dans les fichiers syslog pertinents pour résoudre des problèmes ou vérifier l'état du système.
- Trouver et interpréter des entrées dans le journal système pour résoudre des problèmes ou vérifier l'état du système.
- Configurer le journal système pour conserver l'enregistrement des événements lorsqu'un serveur est redémarré.
- Maintenir une synchronisation précise de l'horloge à l'aide de NTP et configurer le fuseau horaire pour garantir des horodatages corrects pour les événements enregistrés par le journal système et les journaux.

SECTIONS

- Description de l'architecture du journal du système (avec quiz)
- Examen des fichiers syslog (avec exercice guidé)
- Examen des entrées de journal (avec exercice guidé)
- Préservation du journal système (avec exercice guidé)
- Gestion précise de l'heure (avec exercice guidé)

ATELIER

Analyse et stockage des journaux

DESCRIPTION DE L'ARCHITECTURE DU JOURNAL DU SYSTÈME

OBJECTIFS

Au terme de cette section, vous serez en mesure de décrire l'architecture de journalisation de base utilisée par Red Hat Enterprise Linux pour enregistrer des événements.

JOURNALISATION SYSTÈME

Les processus et le noyau du système d'exploitation consignent les événements qui se produisent dans un journal. Ces journaux permettent de contrôler le système et de résoudre les problèmes.

De nombreux systèmes enregistrent les journaux d'événements dans des fichiers texte stockés dans le répertoire **/var/log**. Ces journaux peuvent être examinés à l'aide d'utilitaires de texte normaux tels que **less** et **tail**.

Un système de journalisation standard basé sur le protocole **syslog** est intégré à Red Hat Enterprise Linux. De nombreux programmes utilisent ce système pour enregistrer les événements et les organiser dans des fichiers journaux. Les services **systemd-journald** et **rsyslog** traitent les messages **syslog** dans Red Hat Enterprise Linux 8.

Le service **systemd-journald** est au centre de l'architecture de journalisation des événements du système d'exploitation. Il collecte les messages d'événements provenant de plusieurs sources incluant le noyau, la sortie des premières étapes du processus de démarrage, la sortie standard et l'erreur standard des démons lors de leur démarrage et de leur exécution, ainsi que les événements **syslog**. Il les restructure ensuite dans un format standard et les écrit dans un journal système structuré et indexé. Par défaut, ce journal est stocké sur un système de fichiers qui ne persiste pas après les redémarrages.

Cependant, le service **rsyslog** lit les messages **syslog** reçus par **systemd-journald** depuis le journal dès qu'ils arrivent. Il traite ensuite les événements **syslog**, en les enregistrant dans ses fichiers journaux ou en les transmettant à d'autres services selon sa propre configuration.

Le service **rsyslog** trie et écrit les messages **syslog** dans les fichiers journaux qui ne persistent pas après les redémarrages dans **/var/log**. Le service **rsyslog** trie les messages de journal dans des fichiers de journal spécifiques en fonction du type de programme qui a envoyé chaque message, ou *fonction*, et de la priorité de chaque message **syslog**.

En plus des fichiers de messages **syslog**, le répertoire **/var/log** contient les fichiers journaux d'autres services du système. Le tableau suivant répertorie certains fichiers utiles du répertoire **/var/log**.

Fichiers journaux sélectionnés du système

FICHIER JOURNAL	TYPE DE MESSAGES STOCKÉS
/var/log/messages	La plupart des messages syslog sont consignés ici. Les messages liés à l'authentification, au traitement des courriers électroniques, à l'exécution de tâches planifiées et ceux qui sont purement liés au débogage constituent des exceptions.

FICHIER JOURNAL	TYPE DE MESSAGES STOCKÉS
/var/log/secure	Messages syslog liés aux événements de sécurité et d'authentification.
/var/log/maillog	Messages syslog liés au serveur de messagerie.
/var/log/cron	Messages syslog liés à l'exécution des tâches planifiées.
/var/log/boot.log	Messages de console non syslog liés au démarrage du système.

**NOTE**

Certaines applications n'utilisent pas syslog pour gérer leurs messages de journal, bien qu'en règle générale, elles placent leurs fichiers de journal dans un sous-répertoire de /var/log. Par exemple, le serveur Web Apache enregistre les messages de journal dans des fichiers d'un sous-répertoire du répertoire **/var/log**.

**RÉFÉRENCES**

Pages de manuel **systemd-journald.service(8)**, **rsyslogd(8)** et **rsyslog.conf(5)**

Pour plus d'informations, reportez-vous à la section *Using the log files to troubleshoot problems* dans le manuel *Red Hat Enterprise Linux 8.0 Configuring basic system settings Guide* à l'adresse https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#Troubleshoot-log-files_getting-started-with-system-administration

► QUIZ

DESCRIPTION DE L'ARCHITECTURE DU JOURNAL DU SYSTÈME

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. Quel fichier journal stocke la plupart des messages syslog, à l'exception de ceux liés à l'authentification, au courrier électronique, aux tâches planifiées et au débogage ?
 - a. /var/log/maillog
 - b. /var/log/boot.log
 - c. /var/log/messages
 - d. /var/log/secure
- ▶ 2. Quel fichier journal stocke les messages syslog liés aux opérations de sécurité et d'authentification dans le système ?
 - a. /var/log/maillog
 - b. /var/log/boot.log
 - c. /var/log/messages
 - d. /var/log/secure
- ▶ 3. Quel service trie et organise les messages syslog en fichiers dans /var/log ?
 - a. rsyslog
 - b. systemd-journald
 - c. auditd
 - d. tuned
- ▶ 4. Quel répertoire héberge les fichiers syslog « lisibles par l'utilisateur » ?
 - a. /sys/kernel/debug
 - b. /var/log/journal
 - c. /run/log/journal
 - d. /var/log
- ▶ 5. Quel fichier stocke les messages syslog liés au serveur de messagerie ?
 - a. /var/log/lastlog
 - b. /var/log/maillog
 - c. /var/log/tallylog
 - d. /var/log/boot.log

► 6. Quel fichier stocke les messages syslog liés aux tâches planifiées ?

- a. `/var/log/cron`
- b. `/var/log/tallylog`
- c. `/var/log/spooler`
- d. `/var/log/secure`

► 7. Quel fichier stocke les messages de console liés au démarrage du système ?

- a. `/var/log/messages`
- b. `/var/log/cron`
- c. `/var/log/boot.log`
- d. `/var/log/secure`

► SOLUTION

DESCRIPTION DE L'ARCHITECTURE DU JOURNAL DU SYSTÈME

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. Quel fichier journal stocke la plupart des messages syslog, à l'exception de ceux liés à l'authentification, au courrier électronique, aux tâches planifiées et au débogage ?
 - a. /var/log/maillog
 - b. /var/log/boot.log
 - c. /var/log/messages
 - d. /var/log/secure
- ▶ 2. Quel fichier journal stocke les messages syslog liés aux opérations de sécurité et d'authentification dans le système ?
 - a. /var/log/maillog
 - b. /var/log/boot.log
 - c. /var/log/messages
 - d. /var/log/secure
- ▶ 3. Quel service trie et organise les messages syslog en fichiers dans /var/log ?
 - a. rsyslog
 - b. systemd-journald
 - c. auditd
 - d. tuned
- ▶ 4. Quel répertoire héberge les fichiers syslog « lisibles par l'utilisateur » ?
 - a. /sys/kernel/debug
 - b. /var/log/journal
 - c. /run/log/journal
 - d. /var/log
- ▶ 5. Quel fichier stocke les messages syslog liés au serveur de messagerie ?
 - a. /var/log/lastlog
 - b. /var/log/maillog
 - c. /var/log/tallylog
 - d. /var/log/boot.log

► 6. Quel fichier stocke les messages syslog liés aux tâches planifiées ?

- a. **/var/log/cron**
- b. `/var/log/tallylog`
- c. `/var/log/spooler`
- d. `/var/log/secure`

► 7. Quel fichier stocke les messages de console liés au démarrage du système ?

- a. `/var/log/messages`
- b. `/var/log/cron`
- c. **`/var/log/boot.log`**
- d. `/var/log/secure`

EXAMEN DES FICHIERS SYSLOG

OBJECTIFS

À la fin de cette section, vous devez être en mesure d'interpréter les événements des fichiers syslog pertinents pour résoudre les problèmes ou examiner l'état du système.

JOURNALISATION DES ÉVÉNEMENTS SUR LE SYSTÈME

De nombreux programmes utilisent le protocole syslog pour consigner les événements dans le système. Tous les messages du journal sont classés par service (type du message) et par priorité (importance du message). Les fonctions disponibles sont expliquées dans la page de manuel `rsyslog.conf(5)`.

Le tableau suivant liste les huit priorités syslog standard, par ordre décroissant.

Présentation des priorités syslog

CODE	PRIORITÉ	GRAVITÉ
0	emerg	Système inutilisable
1	alert	Une action doit être prise immédiatement
2	crit	Condition critique
3	err	Erreur non critique
4	warning	Condition d'avertissement
5	notice	Événement normal mais significatif
6	info	Événement informatif
7	debug	Message de débogage

Le service `rsyslog` détermine comment traiter les messages du journal selon leur fonction et leur niveau de priorité. Il est configuré selon les règles figurant dans le fichier `/etc/rsyslog.conf` et tout fichier du répertoire `/etc/rsyslog.d` dont l'extension de son nom est `.conf`. Les paquetages logiciels peuvent facilement ajouter des règles en installant un fichier approprié dans le répertoire `/etc/rsyslog.d`.

Chaque règle qui contrôle le tri des messages syslog correspond à une ligne dans l'un des fichiers de configuration. La partie gauche de chaque ligne indique la fonction et la gravité des messages syslog correspondant à la règle. La partie droite de chaque ligne indique le fichier où sera enregistré le message du journal (ou un autre endroit pour le distribuer). Un astérisque (*) est un caractère générique qui correspond à toutes les valeurs.

Par exemple, la ligne suivante enregistre les messages envoyés à la fonction `authpriv` à un niveau de priorité quelconque dans le fichier `/var/log/secure` :

authpriv.*	/var/log/secure
------------	-----------------

Les messages de journal correspondent parfois à plusieurs règles dans **rsyslog.conf**. Dans ce cas, un message est stocké dans plusieurs fichiers journaux. Pour limiter les messages stockés, le mot-clé **none** dans le champ de priorité indique qu'aucun message pour la fonction indiquée ne doit être stocké dans le fichier donné.

Au lieu de consigner les messages syslog dans un fichier, ils peuvent également être affichés sur le terminal de tous les utilisateurs connectés. Le fichier **rsyslog.conf** offre un paramètre permettant d'imprimer tous les messages syslog avec la priorité **emerg** sur les terminaux de tous les utilisateurs connectés.

EXEMPLES DE RÈGLES DE RSYSLOG

```
##### RULES #####
# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.*                                         /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none        /var/log/messages

# The authpriv file has restricted access.
authpriv.*                                       /var/log/secure

# Log all the mail messages in one place.
mail.*                                           -/var/log/maillog

# Log cron stuff
cron.*                                           /var/log/cron

# Everybody gets emergency messages
*.emerg                                         :omusrmsg:*

# Save news errors of level crit and higher in a special file.
uucp,news.crit                                    /var/log/spooler

# Save boot messages also to boot.log
local7.*                                         /var/log/boot.log
```



NOTE

Le sous-système syslog comporte de nombreuses autres fonctionnalités qui vont au-delà de la portée de ce cours. Pour ceux qui souhaitent approfondir leurs connaissances, consultez la page de manuel **rsyslog.conf(5)** et la documentation HTML complète dans **/usr/share/doc/rsyslog/html/index.html** figurant dans le paquetage **rsyslog-doc**, disponible dans le référentiel AppStream de Red Hat Enterprise Linux 8.

ROTATION DU FICHIER JOURNAL

L'outil **logrotate** fait tourner les fichiers journaux pour les empêcher de prendre trop de place dans le système de fichiers contenant le répertoire **/var/log**. Lorsqu'un fichier journal tourne, il est renommé avec une extension indiquant sa date de rotation. Par exemple, l'ancien fichier **/var/log/messages** peut devenir **/var/log/messages-20190130** si la rotation est effectuée le 2019-01-30. Après la rotation de l'ancien fichier journal, un nouveau fichier journal est créé, et le service qui y écrit est averti.

Après un certain nombre de rotations, généralement au bout de quatre semaines, le plus ancien fichier journal est supprimé pour libérer de l'espace disque. Une tâche planifiée exécute le programme **logrotate** quotidiennement pour déterminer si une rotation de journal est nécessaire. La majorité des fichiers journaux tourne une fois par semaine, mais **logrotate** effectue la rotation de certains fichiers plus rapidement ou plus lentement, ou lorsqu'ils atteignent une certaine taille.

La configuration de **logrotate** n'est pas traitée au cours de cette formation. Pour plus d'informations, consultez la page de manuel **logrotate(8)**.

ANALYSE D'UNE ENTRÉE SYSLOG

Les messages de journaux commencent par le message le plus ancien et se terminent par le plus récent. Le service **rsyslog** utilise un format standard lors de l'enregistrement des entrées dans les fichiers journaux. L'exemple suivant explique l'anatomie d'un message de journal contenu dans le fichier journal **/var/log/secure**.

```
❶ Feb 11 20:11:48 ❷ localhost ❸ sshd[1433]: ❹ Failed password for student from
172.25.0.10 port 59344 ssh2
```

- ❶ L'horodatage de l'enregistrement de l'entrée de journal
- ❷ L'hôte depuis lequel le message a été envoyé au journal
- ❸ Le programme ou le nom et numéro PID du processus qui a envoyé le message de journal
- ❹ Le message effectivement envoyé

CONTRÔLE DES JOURNAUX

Il est utile de contrôler les événements dans un ou plusieurs fichiers journaux pour reproduire les problèmes. La commande **tail -f /path/to/file** renvoie les 10 dernières lignes du fichier spécifié, et continue à renvoyer de nouvelles lignes dans le fichier au fur et à mesure de leur écriture.

Par exemple, pour surveiller les tentatives de connexion infructueuses, exécutez la commande **tail** dans un terminal, puis dans un autre, exécutez la commande **ssh** en tant qu'utilisateur root pendant qu'un utilisateur tente de se connecter au système.

Dans le premier terminal, exécutez la commande **tail** suivante :

```
[root@host ~]# tail -f /var/log/secure
```

Dans le second terminal, exécutez la commande **ssh** suivante :

```
[root@host ~]# ssh root@localhost
root@localhost's password: redhat
...output omitted...
[root@host ~]#
```

Retournez au premier terminal et examinez les journaux.

```
...output omitted...
Feb 10 09:01:13 host sshd[2712]: Accepted password for root from 172.25.254.254
port 56801 ssh2
Feb 10 09:01:13 host sshd[2712]: pam_unix(sshd:session): session opened for user
root by (uid=0)
```

ENVOI MANUEL DE MESSAGES SYSLOG

La commande **logger** peut envoyer des messages au service **rsyslog**. Par défaut, elle envoie le message à la fonction **user** avec la priorité **notice** (**user.notice**), à moins que l'option **-p** ne spécifie autre chose. Cela est utile pour tester les modifications apportées à la configuration du service **rsyslog**.

Pour envoyer au service **rsyslog** un message qui sera enregistré dans le fichier journal **/var/log/boot.log**, exécutez la commande **logger** :

```
[root@host ~]# logger -p local7.notice "Log entry created on host"
```



RÉFÉRENCES

Pages de manuel **logger(1)**, **tail(1)**, **rsyslog.conf(5)** et **logrotate(8)**

Manuel de **rsyslog**

- **/usr/share/doc/rsyslog/html/index.html** fourni dans le paquetage **rsyslog-doc**

Pour plus d'informations, reportez-vous à la section *Using the log files to troubleshoot problems* dans le manuel *Red Hat Enterprise Linux 8.0 Configuring basic system settings Guide* à l'adresse

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#Troubleshoot-log-files_getting-started-with-system-administration

► EXERCICE GUIDÉ

EXAMEN DES FICHIERS SYSLOG

Au cours de cet exercice, vous allez reconfigurer `rsyslog` pour écrire des messages de journaux spécifiques dans un nouveau fichier.

RÉSULTATS

Vous devez pouvoir configurer le service `rsyslog` pour écrire tous les messages de journaux avec la priorité **debug** dans le fichier journal `/var/log/messages-debug`.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez **lab log-configure start** pour mettre fin à l'exercice. Ce script garantit que l'environnement est configuré correctement.

```
[student@workstation ~]$ lab log-configure start
```

- 1. À partir de `workstation`, ouvrez une session SSH sur `servera` en tant que `student`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Configurez `rsyslog` sur `servera` pour consigner tous les messages avec la priorité **debug**, ou plus élevée, pour tout service dans le nouveau fichier journal `/var/log/messages-debug` en ajoutant le fichier de configuration `rsyslog /etc/rsyslog.d/debug.conf`.

- 2.1. Utilisez la commande **sudo -i** pour basculer vers l'utilisateur `root`. Spécifiez le mot de passe `student` pour l'utilisateur `student` si vous y êtes invité lors de l'exécution de la commande **sudo -i**.

```
[student@servera ~]$ sudo -i
[sudo] password for student: student
[root@servera ~]#
```

- 2.2. Créez le fichier `/etc/rsyslog.d/debug.conf` avec les entrées nécessaires pour rediriger tous les messages du journal avec la priorité **debug** définie sur `/var/log/messages-debug`. Vous pouvez utiliser la commande **vim /etc/rsyslog.d/debug.conf** pour créer le fichier avec le contenu suivant.

```
*.debug /var/log/messages-debug
```

CHAPITRE 11 | Analyse et stockage des journaux

Cette ligne de configuration intercepte les messages syslog ayant n'importe quelle fonction et un niveau de priorité **debug** ou supérieur. Le service **rsyslog** écrit ces messages syslog dans le fichier **/var/log/messages-debug**. Le caractère générique (*) dans les champs **facility** ou **priority** de la ligne de configuration indique une fonction ou une priorité.

2.3. Redémarrez le service **rsyslog**.

```
[root@servera ~]# systemctl restart rsyslog
```

- 3. Vérifiez que tous les messages de journaux avec la priorité **debug** apparaissent dans le fichier **/var/log/messages-debug**.

3.1. Utilisez la commande **logger** avec l'option **-p** pour générer un message de journal avec la fonction **user** et la priorité **debug**.

```
[root@servera ~]# logger -p user.debug "Debug Message Test"
```

3.2. Utilisez la commande **tail** pour afficher les dix derniers messages de journaux du fichier **/var/log/messages-debug** et vérifiez que vous voyez le message **Debug Message Test** parmi les autres messages de journaux.

```
[root@servera ~]# tail /var/log/messages-debug
Feb 13 18:22:38 servera systemd[1]: Stopping System Logging Service...
Feb 13 18:22:38 servera rsyslogd[25176]: [origin software="rsyslogd"
swVersion="8.37.0-9.el8" x-pid="25176" x-info="http://www.rsyslog.com"] exiting
on signal 15.
Feb 13 18:22:38 servera systemd[1]: Stopped System Logging Service.
Feb 13 18:22:38 servera systemd[1]: Starting System Logging Service...
Feb 13 18:22:38 servera rsyslogd[25410]: environment variable TZ is not set, auto
correcting this to TZ=/etc/localtime [v8.37.0-9.el8 try http://www.rsyslog.com/
e/2442 ]
Feb 13 18:22:38 servera systemd[1]: Started System Logging Service.
Feb 13 18:22:38 servera rsyslogd[25410]: [origin software="rsyslogd"
swVersion="8.37.0-9.el8" x-pid="25410" x-info="http://www.rsyslog.com"] start
Feb 13 18:27:58 servera student[25416]: Debug Message Test
```

3.3. Quittez à la fois les shells des utilisateurs **root** et **student** sur **servera** pour revenir au shell de l'utilisateur **student** sur **workstation**.

```
[root@servera ~]# exit
logout
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Finish (Terminer)

Sur **workstation**, exécutez **lab log-configure finish** pour mettre fin à l'exercice. Ce script s'assure que l'environnement est restauré à un état propre.

```
[student@workstation ~]$ lab log-configure finish
```

L'exercice guidé est maintenant terminé.

ANALYSE DES ENTRÉES DU JOURNAL SYSTÈME

OBJECTIFS

Au terme de cette section, vous devez être en mesure de trouver et d'interpréter des entrées dans le journal système pour résoudre des problèmes ou examiner l'état du système.

RECHERCHE D'ÉVÉNEMENTS

Le service `systemd-journald` stocke les données de journalisation dans un fichier binaire structuré et indexé appelé journal. Ces données comprennent des informations supplémentaires sur l'événement consigné. Par exemple, pour les événements `syslog`, cela inclut la fonction et la priorité du message d'origine.



IMPORTANT

Dans Red Hat Enterprise Linux 8, le répertoire `/run/log` stocke le journal système par défaut. Le contenu du répertoire `/run/log` est effacé après un redémarrage. Vous pouvez modifier ce paramètre. La procédure à suivre est décrite plus loin dans ce chapitre.

Pour récupérer les messages du journal, utilisez la commande `journalctl`. Cette commande permet d'afficher tous les messages du journal ou de rechercher des événements spécifiques en fonction de toute une série d'options et de critères. L'exécution de la commande en tant que `root` vous permet d'avoir un accès complet au journal. Les utilisateurs normaux peuvent également utiliser cette commande, mais peuvent ne pas voir certains messages.

```
[root@host ~]# journalctl
...output omitted...
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Stopped target Sockets.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Closed D-Bus User Message Bus
Socket.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Closed Multimedia System.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Reached target Shutdown.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Starting Exit the Session...
Feb 21 17:46:25 host.lab.example.com systemd[24268]: pam_unix(systemd-
user:session): session c>
Feb 21 17:46:25 host.lab.example.com systemd[1]: Stopped User Manager for UID
1001.
Feb 21 17:46:25 host.lab.example.com systemd[1]: user-runtime-dir@1001.service:
Unit not needed.
Feb 21 17:46:25 host.lab.example.com systemd[1]: Stopping /run/user/1001 mount
wrapper...
Feb 21 17:46:25 host.lab.example.com systemd[1]: Removed slice User Slice of UID
1001.
Feb 21 17:46:25 host.lab.example.com systemd[1]: Stopped /run/user/1001 mount
wrapper.
Feb 21 17:46:36 host.lab.example.com sshd[24434]: Accepted publickey for root from
172.25.250.>
```

CHAPITRE 11 | Analyse et stockage des journaux

```
Feb 21 17:46:37 host.lab.example.com systemd[1]: Started Session 20 of user root.
Feb 21 17:46:37 host.lab.example.com systemd-logind[708]: New session 20 of user
root.
Feb 21 17:46:37 host.lab.example.com sshd[24434]: pam_unix(sshd:session): session
opened for u>
Feb 21 18:01:01 host.lab.example.com CROND[24468]: (root) CMD (run-parts /etc/
cron.hourly)
Feb 21 18:01:01 host.lab.example.com run-parts[24471]: (/etc/cron.hourly) starting
@anacron
Feb 21 18:01:01 host.lab.example.com run-parts[24477]: (/etc/cron.hourly) finished
@anacron
lines 1464-1487/1487 (END) q
```

La commande **journalctl** met en évidence les messages importants du journal : les messages avec la priorité **notice** ou **warning** sont en gras tandis que les messages avec la priorité **error** ou plus élevée sont en texte rouge.

La clé pour utiliser correctement le journal lors d'un dépannage ou d'un contrôle consiste à lancer les recherches dans le journal pour afficher uniquement les sorties pertinentes.

Par défaut, la commande **journalctl -n** affiche les 10 dernières entrées de journal. Vous pouvez l'ajuster avec un argument facultatif qui spécifie le nombre d'entrées de journal à afficher. Pour les cinq dernières entrées du journal, exécutez la commande **journalctl** suivante :

```
[root@host ~]# journalctl -n 5
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:01:01 +07.
--
...output omitted...
Feb 21 17:46:37 host.lab.example.com systemd-logind[708]: New session 20 of user
root.
Feb 21 17:46:37 host.lab.example.com sshd[24434]: pam_unix(sshd:session): session
opened for u>
Feb 21 18:01:01 host.lab.example.com CROND[24468]: (root) CMD (run-parts /etc/
cron.hourly)
Feb 21 18:01:01 host.lab.example.com run-parts[24471]: (/etc/cron.hourly) starting
@anacron
Feb 21 18:01:01 host.lab.example.com run-parts[24477]: (/etc/cron.hourly) finished
@anacron
lines 1-6/6 (END) q
```

À l'instar de la commande **tail -f**, la commande **journalctl -f** présente les 10 dernières lignes du journal système et continue d'afficher les nouvelles entrées au fur et à mesure qu'elles apparaissent dans le journal. Pour sortir du processus **journalctl -f**, utilisez la combinaison de touches **Ctrl+C**.

```
[root@host ~]# journalctl -f
-- Logs begin at Wed 2019-02-20 16:01:17 +07. --
...output omitted...
Feb 21 18:01:01 host.lab.example.com run-parts[24477]: (/etc/cron.hourly) finished
@anacron
Feb 21 18:22:42 host.lab.example.com sshd[24437]: Received disconnect from
172.25.250.250 port 48710:11: disconnected by user
Feb 21 18:22:42 host.lab.example.com sshd[24437]: Disconnected from user root
172.25.250.250 port 48710
```

CHAPITRE 11 | Analyse et stockage des journaux

```
Feb 21 18:22:42 host.lab.example.com sshd[24434]: pam_unix(sshd:session): session closed for user root
Feb 21 18:22:42 host.lab.example.com systemd-logind[708]: Session 20 logged out.
  Waiting for processes to exit.
Feb 21 18:22:42 host.lab.example.com systemd-logind[708]: Removed session 20.
Feb 21 18:22:43 host.lab.example.com sshd[24499]: Accepted publickey for root from 172.25.250.250 port 48714 ssh2: RSA
  SHA256:1UGybTe52L2jzEJa1HLVKn9QUCKrTv3ZxxnMjol1Fro
Feb 21 18:22:44 host.lab.example.com systemd-logind[708]: New session 21 of user root.
Feb 21 18:22:44 host.lab.example.com systemd[1]: Started Session 21 of user root.
Feb 21 18:22:44 host.lab.example.com sshd[24499]: pam_unix(sshd:session): session opened for user root by (uid=0)
^C
[root@host ~]#
```

Pour résoudre des problèmes, filtrez la sortie du journal en fonction de la priorité des entrées. La commande **journalctl -p** permet de saisir le nom ou le numéro correspondant au niveau de priorité et d'afficher les entrées de journal de ce niveau et des niveaux supérieurs. La commande **journalctl** comprend les niveaux de priorité **debug, info, notice, warning, err, crit, alert et emerg**.

Exécutez la commande **journalctl** suivante pour lister les entrées de journal avec la priorité **err** ou supérieure :

```
[root@host ~]# journalctl -p err
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:01:01 +07.
--
...output omitted...
Feb 20 16:01:17 host.lab.example.com kernel: Detected CPU family 6 model 13
  stepping 3
Feb 20 16:01:17 host.lab.example.com kernel: Warning: Intel Processor - this
  hardware has not undergone testing by Red Hat and might not be certif>
Feb 20 16:01:20 host.lab.example.com smartd[669]: DEVICESCAN failed: glob(3)
  aborted matching pattern /dev/discs/disc*
Feb 20 16:01:20 host.lab.example.com smartd[669]: In the system's table of devices
  NO devices found to scan
lines 1-5/5 (END) q
```

Pour une recherche d'événements spécifiques, vous pouvez limiter les résultats à un laps de temps spécifique. La commande **journalctl** propose deux options pour limiter les résultats à une période spécifique : **--since** et **--until**. Les deux options prennent un argument de temps au format "AAAA-MM-JJhh:mm:ss" (les guillemets doubles sont nécessaires pour préserver l'espace dans l'option). Si la date est omise, la commande suppose qu'il s'agit du jour présent, et si l'heure est omise, la commande considère qu'il s'agit de toute la journée à partir de 00:00:00. Les deux options acceptent les arguments **yesterday, today** et **tomorrow** en plus du champ de date et d'heure.

Exécutez la commande **journalctl** suivante pour lister toutes les entrées de journal dans les enregistrements d'aujourd'hui.

```
[root@host ~]# journalctl --since today
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:31:14 +07.
--
...output omitted...
Feb 21 18:22:44 host.lab.example.com systemd-logind[708]: New session 21 of user
root.
Feb 21 18:22:44 host.lab.example.com systemd[1]: Started Session 21 of user root.
Feb 21 18:22:44 host.lab.example.com sshd[24499]: pam_unix(sshd:session): session
opened for user root by (uid=0)
Feb 21 18:31:13 host.lab.example.com systemd[1]: Starting dnf makecache...
Feb 21 18:31:14 host.lab.example.com dnf[24533]: Red Hat Enterprise Linux 8.0
AppStream (dvd) 637 kB/s | 2.8 kB 00:00
Feb 21 18:31:14 host.lab.example.com dnf[24533]: Red Hat Enterprise Linux 8.0
BaseOS (dvd) 795 kB/s | 2.7 kB 00:00
Feb 21 18:31:14 host.lab.example.com dnf[24533]: Metadata cache created.
Feb 21 18:31:14 host.lab.example.com systemd[1]: Started dnf makecache.
lines 533-569/569 (END) q
```

Exécutez la commande **journalctl** suivante pour lister toutes les entrées de journal allant de **2019-02-10 20:30:00** à **2019-02-13 12:00:00**.

```
[root@host ~]# journalctl --since "2019-02-10 20:30:00" \
--until "2019-02-13 12:00:00"
...output omitted...
```

Vous pouvez également spécifier toutes les entrées depuis une heure donnée par rapport au présent. Par exemple, pour spécifier toutes les entrées de la dernière heure, vous pouvez utiliser la commande suivante :

```
[root@host ~]# journalctl --since "-1 hour"
...output omitted...
```



NOTE

Vous pouvez utiliser d'autres spécifications de temps plus complexes avec les options **--since** et **--until**. Vous trouverez quelques exemples dans la page de manuel `systemd.time(7)`.

En plus du contenu visible du journal, il existe des champs attachés aux entrées de journal qui ne s'affichent que si la sortie détaillée (verbose) est activée. Tous les champs supplémentaires affichés peuvent servir à filtrer la sortie d'une requête auprès du journal. Cela permet de réduire la sortie de recherches complexes de certains événements du journal.

```
[root@host ~]# journalctl -o verbose
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:31:14 +07.
--
...output omitted...
Thu 2019-02-21 18:31:14.509128 +07...
    _PRIORITY=6
    _BOOT_ID=4409bbf54680496d94e090de9e4a9e23
    _MACHINE_ID=73ab164e278e48be9bf80e80714a8cd5
```

```
SYSLOG_FACILITY=3
SYSLOG_IDENTIFIER=systemd
_UID=0
_GID=0
CODE_FILE=../src/core/job.c
CODE_LINE=826
CODE_FUNC=job_log_status_message
JOB_TYPE=start
JOB_RESULT=done
MESSAGE_ID=39f53479d3a045ac8e11786248231fbf
_TRANSPORT=journal
_PID=1
_COMM=systemd
_EXE=/usr/lib/systemd/systemd
_CMDLINE=/usr/lib/systemd/systemd --switched-root --system --deserialize 18
_CAP_EFFECTIVE=3fffffff
_SELINUX_CONTEXT=system_u:system_r:init_t:s0
_SYSTEMD_CGROUP=/init.scope
_SYSTEMD_UNIT=init.scope
_SYSTEMD_SLICE=--.slice
UNIT=dnf-makecache.service
MESSAGE=Started dnf makecache.
_HOSTNAME=host.lab.example.com
INVOCATION_ID=d6f90184663f4309835a3e8ab647cb0e
_SOURCE_REALTIME_TIMESTAMP=1550748674509128
lines 32239-32275/32275 (END) q
```

La liste suivante présente les champs communs du journal système pouvant être utilisés pour rechercher des lignes pertinentes pour un processus ou un événement particulier.

- `_COMM` correspond au nom de la commande
- `_EXE` correspond au chemin de l'exécutable du processus
- `_PID` correspond au PID du processus
- `_UID` correspond à l'UID de l'utilisateur qui exécute le processus
- `_SYSTEM_UNIT` correspond à l'unité systemd qui a lancé le processus

Plusieurs champs du journal système peuvent être combinés pour former une requête de recherche granulaire avec la commande **journalctl**. Par exemple, la commande **journalctl** suivante affiche toutes les entrées de journal liées à l'unité **sshd.service** systemd d'un processus avec le PID 1182.

```
[root@host ~]# journalctl _SYSTEMD_UNIT=sshd.service _PID=1182
Apr 03 19:34:27 host.lab.example.com sshd[1182]: Accepted password for root
from ::1 port 52778 ssh2
Apr 03 19:34:28 host.lab.example.com sshd[1182]: pam_unix(sshd:session): session
opened for user root by (uid=0)
...output omitted...
```



NOTE

Pour une liste des champs de journal les plus utilisés, consultez la page de manuel **systemd.journal-fields(7)**.



RÉFÉRENCES

Pages de manuel `journalctl(1)`, `systemd.journal-fields(7)` et `systemd.time(7)`

Pour plus d'informations, reportez-vous à la section *Using the log files to troubleshoot problems* dans le manuel *Red Hat Enterprise Linux 8.0 Configuring basic system settings Guide* à l'adresse

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#Troubleshoot-log-files_getting-started-with-system-administration

► EXERCICE GUIDÉ

ANALYSE DES ENTRÉES DU JOURNAL SYSTÈME

Au cours de cet exercice, vous allez rechercher dans le journal système des entrées enregistrant des événements correspondant à des critères spécifiques.

RÉSULTATS

Vous serez en mesure de rechercher dans le journal système des entrées enregistrant des événements en fonction de différents critères.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab log-query start` pour mettre fin à l'exercice. Ce script garantit que l'environnement est configuré correctement.

```
[student@workstation ~]$ lab log-query start
```

- 1. À partir de `workstation`, ouvrez une session SSH sur `servera` en tant que `student`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Utilisez la correspondance `_PID=1` avec la commande `journalctl` pour afficher uniquement les événements du journal résultant du processus `systemd` s'exécutant avec l'identificateur de processus 1 sur `servera`. Pour quitter `journalctl`, appuyez sur `q`.

```
[student@servera ~]$ journalctl _PID=1
...output omitted...
Feb 13 13:21:08 localhost systemd[1]: Found device /dev/disk/by-uuid/
cdf61ded-534c-4bd6-b458-cab18b1a72ea.
Feb 13 13:21:08 localhost systemd[1]: Started dracut initqueue hook.
Feb 13 13:21:08 localhost systemd[1]: Found device /dev/disk/by-
uuid/44330f15-2f9d-4745-ae2e-20844f22762d.
Feb 13 13:21:08 localhost systemd[1]: Reached target Initrd Root Device.
lines 1-5/5 (END) q
[student@servera ~]$
```



NOTE

La commande `journalctl` peut produire une sortie différente sur votre système.

CHAPITRE 11 | Analyse et stockage des journaux

- 3. Utilisez la correspondance **_UID=81** avec la commande **journalctl** pour afficher tous les événements du journal provenant d'un service de système démarré avec l'identificateur d'utilisateur 81 sur **servera**. Pour quitter **journalctl**, appuyez sur **q**.

```
[student@servera ~]$ journalctl _UID=81
...output omitted...
Feb 22 01:29:09 servera.lab.example.com dbus-daemon[672]: [system] Activating via
systemd: service name='org.freedesktop.nm_dispatcher'
Feb 22 01:29:09 servera.lab.example.com dbus-daemon[672]: [system] Successfully
activated service 'org.freedesktop.nm_dispatcher'
lines 1-5/5 (END) q
[student@servera ~]$
```

- 4. Utilisez l'option **-p warning** avec la commande **journalctl** pour afficher les événements du journal avec la priorité **warning** et supérieure sur **servera**. Pour quitter **journalctl**, appuyez sur **q**.

```
[student@servera ~]$ journalctl -p warning
...output omitted...
Feb 13 13:21:07 localhost kernel: Detected CPU family 6 model 13 stepping 3
Feb 13 13:21:07 localhost kernel: Warning: Intel Processor - this hardware has not
undergone testing by Red Hat and might not >
Feb 13 13:21:07 localhost kernel: acpi PNP0A03:00: fail to add MMCONFIG
information, can't access extended PCI configuration s>
Feb 13 13:21:07 localhost rpc.statd[288]: Running as root. chown /var/lib/nfs/
statd to choose different user
Feb 13 13:21:07 localhost rpc.idmapd[293]: Setting log level to 0
...output omitted...
Feb 13 13:21:13 servera.lab.example.com rsyslogd[1172]: environment variable TZ is
not set, auto correcting this to TZ=/etc/lo>
Feb 13 14:51:42 servera.lab.example.com systemd[1]: cgroup compatibility
translation between legacy and unified hierarchy sett>
Feb 13 17:15:37 servera.lab.example.com rsyslogd[25176]: environment variable TZ
is not set, auto correcting this to TZ=/etc/l>
Feb 13 18:22:38 servera.lab.example.com rsyslogd[25410]: environment variable TZ
is not set, auto correcting this to TZ=/etc/l>
Feb 13 18:47:55 servera.lab.example.com rsyslogd[25731]: environment variable TZ
is not set, auto correcting this to TZ=/etc/l>
lines 1-17/17 (END) q
[student@servera ~]$
```

- 5. Affichez tous les événements du journal enregistrés au cours des 10 dernières minutes à partir de l'heure actuelle sur **servera**.

- 5.1. Utilisez l'option **--since** avec la commande **journalctl** pour afficher tous les événements de journal enregistrés au cours des 10 dernières minutes sur **servera**. Pour quitter **journalctl**, appuyez sur **q**.

```
[student@servera ~]$ journalctl --since "-10min"
...output omitted...
Feb 13 22:31:01 servera.lab.example.com CROND[25890]: (root) CMD (run-parts /etc/
cron.hourly)
```

```
Feb 13 22:31:01 servera.lab.example.com run-parts[25893]: (/etc/cron.hourly)
  starting 0anacron
Feb 13 22:31:01 servera.lab.example.com run-parts[25899]: (/etc/cron.hourly)
  finished 0anacron
Feb 13 22:31:41 servera.lab.example.com sshd[25901]: Bad protocol version
  identification 'brain' from 172.25.250.254 port 37450
Feb 13 22:31:42 servera.lab.example.com sshd[25902]: Accepted publickey for root
  from 172.25.250.254 port 37452 ssh2: RSA SHA2>
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Started /run/user/0 mount
  wrapper.
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Created slice User Slice of
  UID 0.
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Starting User Manager for UID
  0...
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Started Session 118 of user
  root.
Feb 13 22:31:42 servera.lab.example.com systemd-logind[712]: New session 118 of
  user root.
Feb 13 22:31:42 servera.lab.example.com systemd[25906]: pam_unix(systemd-
  user:session): session opened for user root by (uid=0)
  ...output omitted...
lines 1-32/84 39% q
[student@servera ~]$
```

- ▶ 6. Utilisez l'option **--since** et la correspondance **_SYSTEMD_UNIT="sshd.service"** avec la commande **journalctl** pour afficher tous les événements du journal provenant du service sshd et enregistrés depuis **09:00:00** ce matin sur servera. Pour quitter **journalctl**, appuyez sur **q**.

```
[student@servera ~]$ journalctl --since 9:00:00 _SYSTEMD_UNIT="sshd.service"
  ...output omitted...
Feb 13 13:21:12 servera.lab.example.com sshd[727]: Server listening on 0.0.0.0
  port 22.
Feb 13 13:21:12 servera.lab.example.com sshd[727]: Server listening on :: port 22.
Feb 13 13:22:07 servera.lab.example.com sshd[1238]: Accepted publickey for student
  from 172.25.250.250 port 50590 ssh2: RSA SH>
Feb 13 13:22:07 servera.lab.example.com sshd[1238]: pam_unix(sshd:session):
  session opened for user student by (uid=0)
Feb 13 13:22:08 servera.lab.example.com sshd[1238]: pam_unix(sshd:session):
  session closed for user student
Feb 13 13:25:47 servera.lab.example.com sshd[1289]: Accepted publickey for root
  from 172.25.250.254 port 37194 ssh2: RSA SHA25>
Feb 13 13:25:47 servera.lab.example.com sshd[1289]: pam_unix(sshd:session):
  session opened for user root by (uid=0)
Feb 13 13:25:47 servera.lab.example.com sshd[1289]: pam_unix(sshd:session):
  session closed for user root
Feb 13 13:25:48 servera.lab.example.com sshd[1316]: Accepted publickey for root
  from 172.25.250.254 port 37196 ssh2: RSA SHA25>
Feb 13 13:25:48 servera.lab.example.com sshd[1316]: pam_unix(sshd:session):
  session opened for user root by (uid=0)
Feb 13 13:25:48 servera.lab.example.com sshd[1316]: pam_unix(sshd:session):
  session closed for user root
Feb 13 13:26:07 servera.lab.example.com sshd[1355]: Accepted publickey for student
  from 172.25.250.254 port 37198 ssh2: RSA SH>
```

```
Feb 13 13:26:07 servera.lab.example.com sshd[1355]: pam_unix(sshd:session):  
    session opened for user student by (uid=0)  
Feb 13 13:52:28 servera.lab.example.com sshd[1473]: Accepted publickey for root  
    from 172.25.250.254 port 37218 ssh2: RSA SHA25>  
Feb 13 13:52:28 servera.lab.example.com sshd[1473]: pam_unix(sshd:session):  
    session opened for user root by (uid=0)  
...output omitted...  
lines 1-32 q  
[student@servera ~]$
```

► 7. Déconnectez-vous de servera.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Finish (Terminer)

Sur workstation, exéutez **lab log-query finish** pour mettre fin à l'exercice. Ce script s'assure que l'environnement est restauré à un état propre.

```
[student@workstation ~]$ lab log-query finish
```

L'exercice guidé est maintenant terminé.

CONSERVATION DU JOURNAL DE SYSTÈME

OBJECTIFS

Au terme de cette section, vous devez être en mesure de configurer le journal système pour conserver l'enregistrement des événements lorsqu'un serveur est redémarré.

STOCKAGE DU JOURNAL SYSTÈME DE MANIÈRE PERMANENTE

Par défaut, les journaux du système sont conservés dans le répertoire **/run/log/journal**, ce qui signifie qu'ils sont effacés lorsque le système redémarre. Vous pouvez modifier les paramètres de configuration du service `systemd-journald` dans le fichier **/etc/systemd/journald.conf** pour que les journaux soient conservés après le redémarrage.

Le paramètre `Storage` dans le fichier **/etc/systemd/journald.conf** définit si les journaux système doivent être stockés de manière volatile ou persistante lors du redémarrage. Définissez ce paramètre sur **persistent**, **volatile** ou **auto**, comme suit :

- **persistent** : stocke les journaux dans le répertoire **/var/log/journal** qui est conservé d'un redémarrage à un autre.

Si le répertoire **/var/log/journal** n'existe pas, le service `systemd-journald` le crée.

- **volatile** : stocke les journaux dans le répertoire **/run/log/journal**.

Comme le système de fichiers **/run** est temporaire et n'existe que dans la mémoire d'exécution, les données qu'il contient, y compris les journaux système, ne sont pas conservés après le redémarrage.

- **auto** : rsyslog détermine s'il faut utiliser un stockage persistant ou volatile. Si le répertoire **/var/log/journal** existe, alors rsyslog utilise un stockage persistant, sinon il utilise un stockage volatile.

C'est l'action par défaut si le paramètre `Storage` n'est pas défini.

L'avantage des journaux système persistants est que les données d'historique sont immédiatement accessibles au démarrage. Cependant, même avec un journal persistant, toutes les données ne sont pas conservées éternellement. Le journal possède un mécanisme intégré de rotation du journal à déclenchement mensuel. De plus, par défaut, les journaux ne sont pas autorisés à dépasser 10 % de la taille du système de fichiers, ou à laisser moins de 15 % du système de fichiers libre. Ces valeurs peuvent être ajustées à la fois pour les journaux d'exécution et les journaux persistants dans **/etc/systemd/journald.conf**. Les limites actuelles concernant la taille du journal sont consignées lorsque le processus `systemd-journald` commence. La sortie de commande suivante affiche les entrées de journal qui reflètent les limites de taille actuelles :

```
[user@host ~]$ journalctl | grep -E 'Runtime|System journal'
Feb 25 13:01:46 localhost systemd-journald[147]: Runtime journal (/run/log/
journal/ae06db7da89142138408d77eafea9229c) is 8.0M, max 91.4M, 83.4M free.
Feb 25 13:01:48 remotehost.lab.example.com systemd-journald[548]: Runtime journal
(/run/log/journal/73ab164e278e48be9bf80e80714a8cd5) is 8.0M, max 91.4M, 83.4M
free.
Feb 25 13:01:48 remotehost.lab.example.com systemd-journald[548]: System journal
(/var/log/journal/73ab164e278e48be9bf80e80714a8cd5) is 8.0M, max 3.7G, 3.7G free.
Feb 25 13:01:48 remotehost.lab.example.com systemd[1]: Starting Tell Plymouth To
Write Out Runtime Data...
Feb 25 13:01:48 remotehost.lab.example.com systemd[1]: Started Tell Plymouth To
Write Out Runtime Data.
```



NOTE

Dans le **grep** ci-dessus, le symbole de pipe (**|**) fait office d'indicateur **or**. En d'autres termes, **grep** correspond à n'importe quelle ligne contenant soit la chaîne **Runtime**, soit la chaîne **System** de la sortie **journalctl**. Ceci récupère les limites de taille actuelles sur le stockage de journal volatile (**Runtime**) ainsi que le stockage de journal (**System**) persistant.

Configuration de journaux système persistants

Pour configurer le service **systemd-journald** permettant de conserver les journaux système de manière persistante entre les redémarrages, définissez **Storage** sur **persistent** dans le fichier **/etc/systemd/journald.conf**. Exécutez l'éditeur de texte de votre choix en tant que superutilisateur pour modifier le fichier **/etc/systemd/journald.conf**.

```
[Journal]
Storage=persistent
...output omitted...
```

Après avoir édité le fichier de configuration, redémarrez le service **systemd-journald** pour appliquer les changements de configuration.

```
[root@host ~]# systemctl restart systemd-journald
```

Si le service **systemd-journald** redémarre avec succès, vous pouvez voir que le répertoire **/var/log/journal** est créé et contient un ou plusieurs sous-répertoires. Ces sous-répertoires présentent des caractères hexadécimaux dans leurs noms longs et contiennent des fichiers ***.journal**. Les fichiers ***.journal** sont les fichiers binaires qui stockent les entrées de journal structurées et indexées.

```
[root@host ~]# ls /var/log/journal
73ab164e278e48be9bf80e80714a8cd5
[root@host ~]# ls /var/log/journal/73ab164e278e48be9bf80e80714a8cd5
system.journal user-1000.journal
```

Bien que les journaux système soient conservés entre les redémarrages, vous obtenez un nombre important d'entrées dans la sortie de la commande **journalctl** qui inclut les entrées du démarrage du système actuel ainsi que les précédentes. Pour limiter la sortie à un démarrage

système spécifique, utilisez l'option **-b** avec la commande **journalctl**. La commande **journalctl** suivante récupère les entrées limitées au premier démarrage du système :

```
[root@host ~]# journalctl -b 1  
...output omitted...
```

La commande **journalctl** suivante récupère les entrées limitées au second démarrage du système. L'argument suivant n'a de sens que si le système a été redémarré plus de deux fois :

```
[root@host ~]# journalctl -b 2
```

La commande **journalctl** suivante récupère les entrées limitées au démarrage actuel du système :

```
[root@host ~]# journalctl -b
```



NOTE

Lors du débogage d'une panne du système à l'aide d'un journal persistant, il est généralement nécessaire de limiter la requête du journal au redémarrage qui a précédé la panne. L'option **-b** peut être accompagnée d'un nombre négatif, qui indique le nombre de démarrages précédents du système que les résultats doivent inclure. Par exemple, **journalctl -b -1** limite les résultats au démarrage précédent.



RÉFÉRENCES

Pages de manuel `systemd-journald.conf(5)`, `systemd-journald(8)`

Pour plus d'informations, reportez-vous à la section *Using the log files to troubleshoot problems* dans le manuel *Red Hat Enterprise Linux 8.0 Configuring basic system settings Guide* à l'adresse

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#Troubleshoot-log-files_getting-started-with-system-administration

► EXERCICE GUIDÉ

CONSERVATION DU JOURNAL DE SYSTÈME

Dans cet exercice, vous allez configurer le journal système pour conserver ses données après un redémarrage.

RÉSULTATS

Vous serez en mesure de configurer le journal système pour conserver ses données après un redémarrage.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab log-preserve start` pour démarrer l'exercice. Ce script garantit que l'environnement est correctement configuré.

```
[student@workstation ~]$ lab log-preserve start
```

- ▶ 1. À partir de `workstation`, ouvrez une session SSH sur `servera` en tant que `student`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. En tant que superutilisateur, vérifiez que le répertoire `/var/log/journal` n'existe pas. Utilisez la commande `ls` pour lister le contenu du répertoire `/var/log/journal`. Utilisez `sudo` pour éléver les priviléges de l'utilisateur `student`. Utilisez `student` comme mot de passe le cas échéant.

```
[student@servera ~]$ sudo ls /var/log/journal
[sudo] password for student: student
ls: cannot access '/var/log/journal': No such file or directory
```

Étant donné que le répertoire `/var/log/journal` n'existe pas, le service `systemd-journald` ne conserve pas ses journaux.

- ▶ 3. Configurez le service `systemd-journald` sur `servera` pour conserver les journaux d'un redémarrage à un autre.
 - 3.1. Supprimez la mise en commentaires pour la ligne `Storage=auto` dans le fichier `/etc/systemd/journald.conf` et définissez `Storage` sur `persistent`. Vous pouvez utiliser la commande `sudo vim /etc/systemd/journald.conf` pour

CHAPITRE 11 | Analyse et stockage des journaux

modifier le fichier de configuration. Tapez `/ Storage=auto` à partir du mode de commande `vim` pour rechercher la ligne `Storage=auto`.

```
...output omitted...
[Journal]
Storage=persistent
...output omitted...
```

3.2. Utilisez la commande `systemctl` pour redémarrer le service `systemd-journald` afin d'appliquer les changements de configuration.

```
[student@servera ~]$ sudo systemctl restart systemd-journald.service
```

- ▶ 4. Vérifiez le service `systemd-journald` sur `servera` conserve ses journaux, de telle sorte que ceux-ci soient conservés d'un redémarrage à un autre.

4.1. Utilisez la commande `systemctl reboot` pour redémarrer `servera`.

```
[student@servera ~]$ sudo systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

Notez que la connexion SSH a été terminée dès que vous avez redémarré le système `servera`.

4.2. Ouvrez une session SSH sur `servera` à nouveau.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 4.3. Utilisez la commande `ls` pour vérifier que le répertoire `/var/log/journal` existe. Le répertoire `/var/log/journal` contient un sous-répertoire ayant un nom hexadécimal long. Les fichiers journaux se trouvent dans ce répertoire. Le nom du sous-répertoire sur votre système sera différent.

```
[student@servera ~]$ sudo ls /var/log/journal
[sudo] password for student: student
73ab164e278e48be9bf80e80714a8cd5
[student@servera ~]$ sudo ls /var/log/journal/73ab164e278e48be9bf80e80714a8cd5
system.journal user-1000.journal
```

4.4. Déconnectez-vous de `servera`.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
```

Finish (Terminer)

Sur workstation, exéutez **lab log-preserve finish** pour mettre fin à l'exercice. Ce script s'assure que l'environnement est restauré à un état propre.

```
[student@workstation ~]$ lab log-preserve finish
```

L'exercice guidé est maintenant terminé.

GESTION PRÉCISE DE L'HEURE

OBJECTIFS

Au terme de cette section, vous serez en mesure de maintenir une synchronisation précise de l'horloge à l'aide de NTP et de configurer le fuseau horaire pour garantir des horodatages corrects pour les événements enregistrés par le journal système et les journaux.

DÉFINITION DES HORLOGES LOCALES ET DES FUSEAUX HORAIRES

La synchronisation correcte de l'heure du système est essentielle pour l'analyse du fichier journal entre de nombreux systèmes. Le protocole de temps du réseau (*NTP* pour *Network Time Protocol*) est un moyen standard pour les machines de fournir et d'obtenir un horodatage correct sur Internet. Une machine peut obtenir ces informations précises sur l'heure depuis des services NTP publics sur Internet, comme le NTP Pool Project. Une horloge logicielle de haute qualité indiquant l'heure exacte aux clients locaux est une autre option.

La commande **timedatectl** affiche un aperçu des paramètres du système relatifs à l'heure actuelle, notamment l'heure actuelle, le fuseau horaire et les paramètres de synchronisation NTP du système.

```
[user@host ~]$ timedatectl
      Local time: Fri 2019-04-05 16:10:29 CDT
      Universal time: Fri 2019-04-05 21:10:29 UTC
            RTC time: Fri 2019-04-05 21:10:29
           Time zone: America/Chicago (CDT, -0500)
System clock synchronized: yes
          NTP service: active
    RTC in local TZ: no
```

Une base de données des fuseaux horaires est disponible et peut être affichée avec la commande **timedatectl list-timezones**.

```
[user@host ~]$ timedatectl list-timezones
Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
Africa/Asmara
Africa/Bamako
...
```

Le nom des fuseaux horaires repose sur la base de données des fuseaux horaires gérée par l'IANA. Les fuseaux horaires sont nommés en fonction du continent ou de l'océan, puis généralement, mais pas obligatoirement, en fonction de la plus grande ville dans la région du fuseau horaire. Par exemple, le fuseau horaire « Rocheuses - É.-U. » correspond en grande partie à « Amérique/Denver ».

CHAPITRE 11 | Analyse et stockage des journaux

La sélection du fuseau horaire peut être plus difficile pour les localités d'un même fuseau horaire qui utilisent un système d'heure d'été différent. Par exemple, aux États-Unis, une grande partie de l'état de l'Arizona (« Rocheuses - É.-U. ») n'applique pas l'heure d'été, et se trouve par conséquent dans le fuseau horaire « Amérique/Phoenix ».

La commande **tzselect** est utile pour identifier les bons fuseaux horaires zoneinfo. Elle pose, en effet, à l'utilisateur des questions sur l'emplacement du système en mode interactif, et affiche ainsi le nom du bon fuseau horaire. Cela ne modifie pas le réglage de fuseau horaire du système.

Le superutilisateur peut modifier le paramètre système pour mettre à jour le fuseau horaire actuel à l'aide de la commande **timedatectl set-timezone**. La commande **timedatectl** suivante met à jour le fuseau horaire actuel sur **America/Phoenix**.

```
[root@host ~]# timedatectl set-timezone America/Phoenix
[root@host ~]# timedatectl
    Local time: Fri 2019-04-05 14:12:39 MST
    Universal time: Fri 2019-04-05 21:12:39 UTC
        RTC time: Fri 2019-04-05 21:12:39
       Time zone: America/Phoenix (MST, -0700)
System clock synchronized: yes
          NTP service: active
    RTC in local TZ: no
```



NOTE

Si vous devez utiliser le temps universel coordonné (UTC) sur un serveur particulier, définissez son fuseau horaire sur UTC. La commande **tzselect** n'inclut pas le nom du fuseau horaire UTC. Utilisez la commande **timedatectl set-timezone UTC** pour définir le fuseau horaire actuel du système sur **UTC**.

Utilisez la commande **timedatectl set-time** pour modifier le fuseau horaire actuel du système. L'heure est indiquée au format « AAAA-MM-JJ hh:mm:ss », où la date ou l'heure peut être omise. La commande **timedatectl** suivante change l'heure sur **09:00:00**.

```
[root@host ~]# timedatectl set-time 9:00:00
[root@host ~]# timedatectl
    Local time: Fri 2019-04-05 09:00:27 MST
    Universal time: Fri 2019-04-05 16:00:27 UTC
        RTC time: Fri 2019-04-05 16:00:27
       Time zone: America/Phoenix (MST, -0700)
System clock synchronized: yes
          NTP service: active
    RTC in local TZ: no
```

La commande **timedatectl set-ntp** active ou désactive la synchronisation NTP pour le réglage automatique de l'heure. Cette option nécessite un argument **true** ou **false** pour être activée ou désactivée. La commande **timedatectl** suivante active la synchronisation NTP.

```
[root@host ~]# timedatectl set-ntp true
```

**NOTE**

Dans Red Hat Enterprise Linux 8, la commande **timedatectl set-ntp** détermine si le service NTP chronyd fonctionne ou non. D'autres distributions Linux peuvent utiliser ce paramètre pour ajuster un service NTP ou SNTP différent.

L'activation ou la désactivation de NTP à l'aide d'autres utilitaires dans Red Hat Enterprise Linux, comme dans l'application graphique des paramètres GNOME, met également à jour ce paramètre.

CONFIGURATION ET CONTRÔLE DE CHRONYD

Le service chronyd veille à la précision de l'horloge matérielle locale (RTC) (manquant habituellement de précision) en la synchronisant sur les serveurs NTP configurés. Si aucune connexion réseau n'est disponible, chronyd calcule la dérive d'horloge RTC calculée, qui est enregistrée dans le **driftfile** spécifié dans le fichier de configuration **/etc/chrony.conf**.

Le service chronyd utilise par défaut les serveurs du NTP Pool Project pour la synchronisation, et n'a pas besoin de configuration supplémentaire. Il peut être utile de modifier les serveurs NTP lorsque la machine en question se trouve sur un réseau isolé.

La **stratum** de la source de temps NTP détermine sa qualité. La valeur stratum détermine le nombre de sauts qui séparent la machine d'une horloge de référence à hautes performances. L'horloge de référence est une source de temps de **stratum 0**. Un serveur NTP directement lié à cette horloge est de **stratum 1**, alors qu'une machine dont l'heure est synchronisée à partir d'un serveur NTP est une source de temps de **stratum 2**.

server et **peer** sont deux catégories de sources temporelles qui peuvent être configurées dans le fichier de configuration **/etc/chrony.conf**. **server** est un niveau de stratum au-dessus de votre serveur NTP local et **peer** est au même niveau de stratum. Plusieurs sources **server** et **peer** peuvent être indiquées, une par ligne.

Le premier argument de la ligne **server** est l'adresse IP ou le nom DNS du serveur NTP. Une série d'options pour le serveur peut être répertoriée après le nom ou l'adresse IP du serveur. Il est recommandé d'utiliser l'option **iburst**, car après le démarrage du serveur, quatre mesures sont prises en un court laps de temps pour assurer une meilleure synchronisation initiale de l'horloge.

La ligne **server classroom.example.com iburst** suivante dans le fichier **/etc/chrony.conf** amène le service chronyd à utiliser la source de temps NTP `classroom.example.com`.

```
# Use public servers from the pool.ntp.org project.
...output omitted...
server classroom.example.com iburst
...output omitted...
```

Après avoir dirigé **chronyd** vers la source de temps locale, `classroom.example.com`, vous devez redémarrer le service.

```
[root@host ~]# systemctl restart chronyd
```

La commande **chronyc** agit comme client du service chronyd. Après avoir configuré la synchronisation NTP, vous devez vérifier que le système local utilise le serveur NTP de manière transparente pour synchroniser l'horloge système à l'aide de la commande **chrony sources**.

Pour un résultat plus détaillé avec des explications supplémentaires sur la sortie, utilisez la commande **chronyc sources -v**.

```
[root@host ~]# chronyc sources -v
210 Number of sources = 1

--- Source mode '^' = server, '=' = peer, '#' = local clock.
/ .- Source state '*' = current synced, '+' = combined , '-' = not combined,
| /   '?' = unreachable, 'x' = time may be in error, '~' = time too variable.
||                               .- xxxx [ yyyy ] +/- zzzz
||                               / xxxx = adjusted offset,
||           Log2(Polling interval) -.          | yyyy = measured offset,
||                               \          | zzzz = estimated error.
||                               |          |
MS Name/IP address      Stratum Poll Reach LastRx Last sample
=====
^* classroom.example.com      8    6    17    23   -497ns[-7000ns] +/-  956us
```

Un caractère * dans le champ **S** (état Source) indique que le serveur `classroom.example.com` a été utilisé comme source temporelle, et qu'il s'agit du serveur NTP avec lequel la machine est actuellement synchronisée.



RÉFÉRENCES

Pages de manuel `timedatectl(1)`, `tzselect(8)`, `chronyd(8)`, `chrony.conf(5)` et `chronyc(1)`

NTP Pool Project

<http://www.pool.ntp.org/>

Base de données des fuseaux horaires

<http://www.iana.org/time-zones>

► EXERCICE GUIDÉ

GESTION PRÉCISE DE L'HEURE

Au cours de cet exercice, vous allez régler le fuseau horaire d'un serveur et vous assurer que son horloge système est synchronisée avec une source de temps NTP.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Changer le fuseau horaire sur un serveur.
- Configurer le serveur pour synchroniser son heure avec une source d'heure NTP.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab log-maintain start` pour mettre fin à l'exercice. Ce script garantit que la synchronisation de l'heure est désactivée sur le système `servera` pour vous permettre de mettre à jour manuellement les paramètres du système et d'activer la synchronisation de l'heure.

```
[student@workstation ~]$ lab log-maintain start
```

- 1. À partir de `workstation`, ouvrez une session SSH sur `servera` en tant que `student`.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Dans le cadre de cette activité, supposez que le système `servera` est déménagé vers Haïti et que vous devez donc mettre à jour le fuseau horaire de manière appropriée. Utilisez `sudo` pour éléver les priviléges de l'utilisateur `student` tout en exécutant la commande `timedatectl` pour mettre à jour le fuseau horaire. Utilisez `student` comme mot de passe le cas échéant.
- 2.1. Utilisez la commande `tzselect` pour déterminer le fuseau horaire approprié pour Haïti.

```
[student@servera ~]$ tzselect
Please identify a location so that time zone rules can be set correctly.
Please select a continent, ocean, "coord", or "TZ".
1) Africa
2) Americas
3) Antarctica
4) Asia
5) Atlantic Ocean
```

```
6) Australia
7) Europe
8) Indian Ocean
9) Pacific Ocean
10) coord - I want to use geographical coordinates.
11) TZ - I want to specify the time zone using the Posix TZ format.
#? 2
Please select a country whose clocks agree with yours.
1) Anguilla          19) Dominican Republic   37) Peru
2) Antigua & Barbuda 20) Ecuador           38) Puerto Rico
3) Argentina         21) El Salvador        39) St Barthelemy
4) Aruba             22) French Guiana      40) St Kitts & Nevis
5) Bahamas           23) Greenland         41) St Lucia
6) Barbados          24) Grenada           42) St Maarten (Dutch)
7) Belize             25) Guadeloupe        43) St Martin (French)
8) Bolivia            26) Guatemala         44) St Pierre & Miquelon
9) Brazil             27) Guyana            45) St Vincent
10) Canada            28) Haiti              46) Suriname
11) Caribbean NL     29) Honduras          47) Trinidad & Tobago
12) Cayman Islands    30) Jamaica           48) Turks & Caicos Is
13) Chile              31) Martinique        49) United States
14) Colombia          32) Mexico             50) Uruguay
15) Costa Rica        33) Montserrat       51) Venezuela
16) Cuba               34) Nicaragua         52) Virgin Islands (UK)
17) Curaçao           35) Panama            53) Virgin Islands (US)
18) Dominica          36) Paraguay
#? 28
The following information has been given:
```

Haiti

Therefore TZ='America/Port-au-Prince' will be used.

Selected time is now: Tue Feb 19 00:51:05 EST 2019.

Universal Time is now: Tue Feb 19 05:51:05 UTC 2019.

Is the above information OK?

- 1) Yes
 - 2) No
- #? 1

You can make this change permanent for yourself by appending the line

TZ='America/Port-au-Prince'; export TZ
to the file '.profile' in your home directory; then log out and log in again.

Here is that TZ value again, this time on standard output so that you
can use the /usr/bin/tzselect command in shell scripts:

America/Port-au-Prince

Notez que la commande **tzselect** précédente affiche le fuseau horaire approprié pour Haïti.

2.2. Utilisez la commande **timedatectl** pour mettre à jour le fuseau horaire de servera sur **America/Port-au-Prince**.

```
[student@servera ~]$ sudo timedatectl set-timezone America/Port-au-Prince
[sudo] password for student:
```

- 2.3. Utilisez la commande **timedatectl** pour vérifier que le fuseau horaire a été mis à jour vers **America/Port-au-Prince**.

```
[student@servera ~]$ timedatectl
    Local time: Tue 2019-02-19 01:16:29 EST
    Universal time: Tue 2019-02-19 06:16:29 UTC
          RTC time: Tue 2019-02-19 06:16:29
        Time zone: America/Port-au-Prince (EST, -0500)
System clock synchronized: no
          NTP service: inactive
      RTC in local TZ: no
```

- 3. Configurez le service chronyd sur servera pour synchroniser l'heure du système avec la source de temps NTP classroom.example.com.
- 3.1. Éditez le fichier **/etc/chrony.conf** pour spécifier le serveur classroom.example.com comme source de temps NTP. Vous pouvez utiliser la commande **sudo vim /etc/chrony.conf** pour éditer le fichier de configuration. La sortie suivante montre la ligne de configuration que vous devez ajouter au fichier de configuration :

```
...output omitted...
server classroom.example.com iburst
...output omitted...
```

La ligne précédente dans le fichier de configuration **/etc/chrony.conf** comprend l'option **iburst** pour accélérer la synchronisation de l'heure initiale.

- 3.2. Utilisez la commande **timedatectl** pour activer la synchronisation de l'heure sur servera.

```
[student@servera ~]$ sudo timedatectl set-ntp yes
```

La commande **timedatectl** précédente active le serveur NTP avec les paramètres modifiés dans le fichier de configuration **/etc/chrony.conf**. La commande **timedatectl** précédente peut activer le service chronyd ou ntpd, selon celui qui est actuellement installé sur le système.

- 4. Vérifiez que les paramètres d'heure sur servera sont actuellement configurés pour se synchroniser avec la source de temps classroom.example.com dans l'environnement de formation.
- 4.1. Utilisez la commande **timedatectl** pour vérifier que la synchronisation de temps est actuellement activée sur servera.

```
[student@servera ~]$ timedatectl
    Local time: Tue 2019-02-19 01:52:17 EST
    Universal time: Tue 2019-02-19 06:52:17 UTC
        RTC time: Tue 2019-02-19 06:52:17
      Time zone: America/Port-au-Prince (EST, -0500)
System clock synchronized: yes
          NTP service: active
    RTC in local TZ: no
```

**NOTE**

Si la sortie précédente indique que l'horloge n'est pas synchronisée, attendez deux secondes, puis exéutez à nouveau la commande **timedatectl**. Il faut quelques secondes pour synchroniser avec succès les paramètres de l'heure avec la source de l'heure.

- 4.2. Utilisez la commande **chronyc** pour vérifier que le système servera synchronise actuellement ses paramètres d'heure avec la source de temps `classroom.example.com`.

```
[student@servera ~]$ chronyc sources -v
210 Number of sources = 1

    --- Source mode  '^' = server, '=' = peer, '#' = local clock.
    / .- Source state '*' = current synced, '+' = combined , '-' = not combined,
| /   '?' = unreachable, 'x' = time may be in error, '~-' = time too variable.
||                               .- xxxx [ yyyy ] +/- zzzz
||       Reachability register (octal) -.           |   xxxx = adjusted offset,
||       Log2(Polling interval) --.          |           |   yyyy = measured offset,
||                           \           |           |   zzzz = estimated error.
||                           |           |
MS Name/IP address          Stratum Poll Reach LastRx Last sample
=====
^* classroom.example.com      2   6   377   62   +105us[ +143us] +/-  14ms
```

Notez que la sortie précédente indique un astérisque (*) dans le champ d'état source (**S**) pour la source de temps NTP `classroom.example.com`. L'astérisque indique que l'heure du système local est actuellement synchronisée correctement avec la source de l'heure NTP.

- 4.3. Déconnectez-vous de servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Finish (Terminer)

Sur workstation, exéutez **lab log-maintain finish** pour mettre fin à l'exercice. Ce script garantit que le fuseau horaire d'origine est restauré avec tous les paramètres d'heure d'origine sur servera.

```
[student@workstation ~]$ lab log-maintain finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

ANALYSE ET STOCKAGE DES JOURNAUX

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez modifier le fuseau horaire d'un serveur existant et configurer un nouveau fichier journal pour tous les événements liés aux échecs d'authentification.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Mettre à jour le fuseau horaire sur un serveur existant.
- Configurer un nouveau fichier journal pour stocker tous les messages liés aux échecs d'authentification.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab log-review start` pour mettre fin à l'exercice. Ce script enregistre le fuseau horaire actuel du système `serverb` et veille à ce que l'environnement soit configuré correctement.

```
[student@workstation ~]$ lab log-review start
```

1. À partir de `workstation`, ouvrez une session SSH sur `serverb` en tant que `student`.
2. Supposez que le système `serverb` est déménagé vers la Jamaïque et que vous devez donc mettre à jour le fuseau horaire de manière appropriée. Utilisez `sudo` pour éléver les priviléges de l'utilisateur `student` de la commande `timedatectl` pour mettre à jour le fuseau horaire. Utilisez `student` comme mot de passe le cas échéant.
3. Affichez les événements de journal enregistrés au cours des 30 dernières minutes sur `serverb`.
4. Créez le fichier `/etc/rsyslog.d/auth-errors.conf` configuré de manière à ce que le service `rsyslog` écrive des messages liés aux problèmes d'authentification et de sécurité dans le nouveau fichier `/var/log/auth-errors`. Utilisez la fonction `authpriv` et la priorité `alert` dans le fichier de configuration.

Évaluation

Sur `workstation`, exécutez la commande `lab log-review grade` pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab log-review grade
```

Finish (Terminer)

Sur workstation, exéutez **lab log-review finish** pour mettre fin à l'atelier. Ce script garantit que le fuseau horaire d'origine est restauré avec tous les paramètres d'heure d'origine sur serverb.

```
[student@workstation ~]$ lab log-review finish
```

L'exercice guidé est maintenant terminé.

► SOLUTION

ANALYSE ET STOCKAGE DES JOURNAUX

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez modifier le fuseau horaire d'un serveur existant et configurer un nouveau fichier journal pour tous les événements liés aux échecs d'authentification.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Mettre à jour le fuseau horaire sur un serveur existant.
- Configurer un nouveau fichier journal pour stocker tous les messages liés aux échecs d'authentification.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab log-review start` pour mettre fin à l'exercice. Ce script enregistre le fuseau horaire actuel du système `serverb` et veille à ce que l'environnement soit configuré correctement.

```
[student@workstation ~]$ lab log-review start
```

1. À partir de `workstation`, ouvrez une session SSH sur `serverb` en tant que `student`.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

2. Supposez que le système `serverb` est déménagé vers la Jamaïque et que vous devez donc mettre à jour le fuseau horaire de manière appropriée. Utilisez `sudo` pour éléver les priviléges de l'utilisateur `student` de la commande `timedatectl` pour mettre à jour le fuseau horaire. Utilisez `student` comme mot de passe le cas échéant.
 - Utilisez la commande `timedatectl` pour afficher les fuseaux horaires disponibles et déterminer le fuseau horaire approprié pour la Jamaïque.

```
[student@serverb ~]$ timedatectl list-timezones | grep Jamaica
America/Jamaica
```

2. Utilisez la commande `timedatectl` pour définir le fuseau horaire du système `serverb` sur **Amérique/Jamaïque**.

```
[student@serverb ~]$ sudo timedatectl set-timezone America/Jamaica
[sudo] password for student:
```

- 2.3. Utilisez la commande **timedatectl** pour vérifier que le fuseau horaire est bien défini sur **Amérique/Jamaïque**.

```
[student@serverb ~]$ timedatectl
    Local time: Tue 2019-02-19 11:12:46 EST
    Universal time: Tue 2019-02-19 16:12:46 UTC
          RTC time: Tue 2019-02-19 16:12:45
        Time zone: America/Jamaica (EST, -0500)
System clock synchronized: yes
          NTP service: active
    RTC in local TZ: no
```

3. Affichez les événements de journal enregistrés au cours des 30 dernières minutes sur serverb.
- 3.1. Utilisez la commande **date** pour déterminer la période nécessaire pour voir les entrées de journal.

```
[student@serverb ~]$ date
Fri Feb 22 07:31:05 EST 2019
[student@serverb ~]$ date -d "-30 minutes"
Fri Feb 22 07:01:31 EST 2019
```

- 3.2. Utilisez les options **--since** et **--until** de la commande **journalctl** pour afficher les événements de journal enregistrés au cours des 30 dernières minutes sur serverb. Pour quitter **journalctl**, appuyez sur **q**.

```
[student@serverb ~]$ journalctl --since 07:01:00 --until 07:31:00
...output omitted...
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Timers.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Paths.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Starting D-Bus User Message Bus Socket.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Listening on D-Bus User Message Bus Socket.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Sockets.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Basic System.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Default.
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Startup finished in 123ms.
Feb 22 07:24:28 serverb.lab.example.com systemd[1]: Started User Manager for UID 1000.
Feb 22 07:24:28 serverb.lab.example.com sshd[1134]: pam_unix(sshd:session): session opened for user student by (uid=0)
Feb 22 07:26:56 serverb.lab.example.com systemd[1138]: Starting Mark boot as successful...
```

CHAPITRE 11 | Analyse et stockage des journaux

```
Feb 22 07:26:56 serverb.lab.example.com systemd[1138]: Started Mark boot as
successful.
lines 1-36/36 (END) q
[student@serverb ~]$
```

4. Créez le fichier **/etc/rsyslog.d/auth-errors.conf** configuré de manière à ce que le service **rsyslog** écrive des messages liés aux problèmes d'authentification et de sécurité dans le nouveau fichier **/var/log/auth-errors**. Utilisez la fonction **authpriv** et la priorité **alert** dans le fichier de configuration.
- 4.1. Créez le fichier **/etc/rsyslog.d/auth-errors.conf** pour spécifier le nouveau fichier **/var/log/auth-errors** comme destination des messages relatifs aux problèmes d'authentification et de sécurité. Vous pouvez utiliser la commande **sudo vim /etc/rsyslog.d/auth-errors.conf** pour créer le fichier de configuration.

```
authpriv.alert  /var/log/auth-errors
```

- 4.2. Redémarrez le service **rsyslog** pour que les modifications apportées au fichier de configuration prennent effet.

```
[student@serverb ~]$ sudo systemctl restart rsyslog
```

- 4.3. Utilisez la commande **logger** pour écrire un nouveau message de journal dans le fichier **/var/log/auth-errors**. Appliquez l'option **-p authpriv.alert** pour générer un message de journal relatif aux problèmes d'authentification et de sécurité.

```
[student@serverb ~]$ logger -p authpriv.alert "Logging test authpriv.alert"
```

- 4.4. Utilisez la commande **tail** pour vérifier que le fichier **/var/log/auth-errors** présente l'entrée de journal contenant le message **Logging test authpriv.alert**.

```
[student@serverb ~]$ sudo tail /var/log/auth-errors
Feb 19 11:56:07 serverb student[6038]: Logging test authpriv.alert
```

- 4.5. Déconnectez-vous de **serverb**.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

Évaluation

Sur **workstation**, exécutez la commande **lab log-review grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab log-review grade
```

Finish (Terminer)

Sur workstation, exéutez **lab log-review finish** pour mettre fin à l'atelier. Ce script garantit que le fuseau horaire d'origine est restauré avec tous les paramètres d'heure d'origine sur serverb.

```
[student@workstation ~]$ lab log-review finish
```

L'exercice guidé est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Les services `systemd-journald` et `rsyslog` capturent et écrivent des messages de journal dans les fichiers appropriés.
- Le répertoire **/var/log** contient les fichiers journaux.
- La rotation périodique des fichiers journaux les empêche de remplir l'espace du système de fichiers.
- Les journaux `systemd` sont temporaires et ne sont pas conservés entre les redémarrages.
- Le service `chrony` permet de synchroniser les paramètres d'heure avec une source de temps.
- Le fuseau horaire du serveur peut être mis à jour en fonction de son emplacement.

CHAPITRE 12

GESTION DE RÉSEAUX

PROJET

Configurer les interfaces réseau et les paramètres sur des serveurs Red Hat Enterprise Linux.

OBJECTIFS

- Décrire les concepts fondamentaux de l'adressage réseau et du routage pour un serveur.
- Tester et inspecter la configuration réseau actuelle avec les utilitaires de ligne de commande.
- Gérer les paramètres réseau et les périphériques à l'aide de **nmcli**.
- Modifier les paramètres réseau en modifiant les fichiers de configuration.
- Configurer le nom d'hôte statique d'un serveur et sa résolution, puis tester les résultats.

SECTIONS

- Description des concepts de mise en réseau (et quiz)
- Validation de la configuration du réseau (avec exercice guidé)
- Configuration de la mise en réseau à partir de la ligne de commande (avec exercice guidé)
- Édition de la configuration du réseau (avec exercice guidé)
- Configuration des noms d'hôte et de la résolution de noms (avec exercice pratique)

ATELIER

Gestion de réseaux

DESCRIPTION DES CONCEPTS RÉSEAU

OBJECTIFS

Au terme de cette section, vous serez en mesure de décrire les concepts fondamentaux de l'adressage réseau et du routage pour un serveur.

MODÈLE DE RÉSEAU TCP/IP

Le *modèle de réseau TCP/IP* est un ensemble, simplifié et comprenant quatre couches, d'abstractions décrivant l'interopérabilité des différents protocoles afin que les ordinateurs puissent envoyer du trafic d'un ordinateur à un autre via Internet. Il est spécifié par la RFC 1122, *Requirements for Internet Hosts -- Communication Layers*. Les quatre couches sont :

- **Application**

Chaque application a ses spécifications de communication, afin que les clients et les serveurs puissent communiquer par l'intermédiaire de plateformes. On trouve parmi les protocoles courants SSH (connexion à distance), HTTPS (réseau sécurisé), NFS ou CIFS (partage de fichiers) et SMTP (envoi de courriers électroniques).

- **Transport**

Les protocoles de transport sont le TCP et l'UDP. TCP est un protocole de communication en ligne fiable, alors qu'UDP est un protocole de *datagrammes sans connexion*. Les protocoles d'application utilisent les ports TCP ou UDP. Vous trouverez une liste des ports connus et enregistrés dans le fichier **/etc/services**.

Lorsqu'un paquetage est envoyé sur le réseau, la combinaison du port du service et de l'adresse IP forme un *socket*. Chaque paquetage possède une socket source et un socket de destination. Ces informations peuvent servir pour la surveillance et le filtrage.

- **Internet**

La couche Internet, ou couche réseau, transporte les données de l'hôte source à l'hôte destinataire. Les protocoles IPv4 et IPv6 sont des protocoles de couche Internet. Chaque hôte possède une adresse IP et un préfixe, utilisés pour déterminer les adresses réseau. On utilise des routeurs pour connecter les réseaux les uns aux autres.

- **Liaison**

La couche liaison, ou « media access », assure la connexion aux médias physiques. Les types de réseaux les plus courants sont les réseaux filaires Ethernet (802.3) et les réseaux sans fil WLAN (802.11). Chaque périphérique physique possède une adresse matérielle (MAC) qui sert à identifier la destination des paquetages sur le segment du réseau local.

DESCRIPTION DES NOMS D'INTERFACES RÉSEAU

Chaque port réseau d'un système a un nom, qui vous permet de configurer et d'identifier celui-ci.

Les anciennes versions de Red Hat Enterprise Linux utilisaient des noms tels que `eth0`, `eth1` et `eth2` pour chaque interface réseau. Le nom `eth0` était le premier port réseau détecté par le système d'exploitation, `eth1` le second, et ainsi de suite. Cependant, au fur et à mesure que

CHAPITRE 12 | Gestion de réseaux

des périphériques étaient ajoutés et supprimés, le mécanisme de détection et de nommage des périphériques pouvait changer l'interface obtenant le nom. De plus, la norme PCIe ne garantit pas l'ordre dans lequel les périphériques PCIe seront détectés au démarrage, ce qui pourrait modifier le nommage des périphériques de manière inattendue en raison de variations lors du démarrage des périphériques ou du système.

Les nouvelles versions de Red Hat Enterprise Linux utilisent un système de nommage différent. Au lieu d'être basés sur l'ordre de détection, les noms des interfaces réseau sont attribués en fonction des informations du microprogramme, de la topologie de bus PCI et du type de périphérique réseau.

Les noms d'interface réseau commencent par le type d'interface :

- Les interfaces Ethernet commencent par `en`.
- Les interfaces WLAN commencent par `wl`.
- Les interfaces WWAN commencent par `ww`.

Le reste du nom de l'interface qui suit le type sera basé sur les informations fournies par le microprogramme du serveur ou déterminées par l'emplacement du périphérique dans la topologie PCI.

- `oN` indique qu'il s'agit d'un périphérique embarqué et que le numéro d'index du périphérique fourni par le microprogramme du serveur est *N*. `eno1` est donc un périphérique Ethernet embarqué 1. De nombreux serveurs ne fourniront pas ces informations.
- `sN` indique que ce périphérique se situe dans l'emplacement enfichable à chaud PCI *N*. `ens3` est donc une carte Ethernet insérée dans l'emplacement enfichable à chaud PCI 3.
- `pMsN` indique qu'il s'agit d'un périphérique PCI sur le bus *M* inséré dans l'emplacement *N*. `w1p4s0` est donc une carte WLAN sur le bus PCI 4 insérée dans l'emplacement 0. Si la carte est un périphérique multifonction dispositif (possible avec une carte Ethernet à ports multiples ou avec des périphériques dotés d'Ethernet et d'autres fonctionnalités), `fN` sera ajouté au nom du périphérique. `enp0s1f0` correspond donc à la fonction 0 de la carte Ethernet sur le bus 0 insérée dans l'emplacement 1. Il peut également y avoir une deuxième interface nommée `enp0s1f1` qui correspond à la fonction 1 de ce même périphérique.

Un nommage persistant signifie qu'une fois que vous connaissez le nom d'une interface réseau sur le système, vous savez également que celui-ci ne changera pas plus tard. En contrepartie, vous ne pouvez pas supposer qu'un système ayant une seule interface nommera cette interface `eth0`.

MISE EN RÉSEAU IPV4

IPv4 est le principal protocole réseau utilisé sur Internet aujourd'hui. Vous devez au moins avoir une connaissance de base du réseau IPv4 afin de gérer les communications réseau de vos serveurs.

Adresses IPv4

Une adresse IPv4 est une adresse de 32 bits, normalement exprimée sous la forme de quatre octets de 8 bits écrits en nombres décimaux, d'une valeur comprise entre 0 et 255 chacun et séparés par des points. L'adresse est divisée en deux parties : la *partie réseau* et la *partie hôte*. Tous les hôtes d'un même sous-réseau, qui peuvent communiquer directement sans passer par un routeur, ont la même partie réseau. Cette partie identifie le sous-réseau. Deux hôtes appartenant au même sous-réseau ne peuvent pas avoir la même partie hôte. La partie hôte identifie un hôte en particulier sur un sous-réseau.

Dans l'Internet moderne, la taille d'un sous-réseau IPv4 est variable. Pour connaître la partie d'une adresse IPv4 qui correspond à la partie réseau et celle qui correspond à la partie hôte, un administrateur doit connaître le *masque de réseau* affecté au sous-réseau. Le masque de réseau indique le nombre de bits de l'adresse IPv4 alloués au sous-réseau. Plus il y a de bits disponibles pour la partie hôte, plus il peut y avoir d'hôtes dans le sous-réseau.

L'adresse la plus basse possible d'un sous-réseau (une partie hôte qui ne comprend que des zéros en binaire) est parfois appelée *adresse réseau*. L'adresse la plus haute possible d'un sous-réseau (une partie hôte qui ne comprend que des 1 en binaire) sert aux messages « broadcast » en IPv4, et est appelée *adresse de diffusion* (ou broadcast).

Le masque de réseau s'exprime de deux manières différentes. L'ancienne syntaxe d'un masque de réseau utilise 24 bits pour la partie réseau et se présente comme suit : 255.255.255.0. Une nouvelle syntaxe, la notation CIDR, définit un *préfixe de réseau* en /24. Ces deux formes véhiculent la même information, à savoir le nombre de bits en début d'adresse IP qui représentent l'adresse réseau.

Les exemples suivants illustrent le lien entre l'adresse IP, le préfixe (masque de réseau), la partie réseau et la partie hôte.

IP Address:

172.17.5.3 = **10101100.00010001.00000101.00000011**

Prefix: /16

Netmask:

255.255.0.0 = **11111111.11111111.00000000.00000000**



IP Address:

192.168.5.3 = **11000000.10101000.00000101.00000011**

Prefix: /24

Netmask:

255.255.255.0 = **11111111.11111111.11111111.00000000**



Figure 12.1: Adresses IPv4 et masques de réseau

Calcul de l'adresse réseau pour 192.168.1.107/24

Adresse d'hôte	192.168.1.107	11000000.10101000.00000001.01101011
Préfixe de réseau	/24 (255.255.255.0)	11111111.11111111.11111111.00000000
Adresse réseau	192.168.1.0	11000000.10101000.00000001.00000000
Adresse de diffusion	192.168.1.255	11000000.10101000.00000001.11111111

Calcul de l'adresse réseau pour 10.1.1.18/8

Adresse d'hôte	10.1.1.18	00001010 . 00000001 . 00000001 . 00010010
Préfixe de réseau	/8 (255.0.0.0)	11111111 . 00000000 . 00000000 . 00000000
Adresse réseau	10.0.0.0	00001010 . 00000000 . 00000000 . 00000000
Adresse de diffusion	10.255.255.255	00001010 . 11111111 . 11111111 . 11111111

Calcul de l'adresse réseau pour 172.16.181.23/19

Adresse d'hôte	172.168.181.23	10101100 . 10101000 . 10110101 . 00010111
Préfixe de réseau	/19 (255.255.224.0)	11111111 . 11111111 . 11100000 . 00000000
Adresse réseau	172.168.160.0	10101100 . 10101000 . 10100000 . 00000000
Adresse de diffusion	172.168.191.255	10101100 . 10101000 . 10111111 . 11111111

L'adresse spéciale 127.0.0.1 pointe toujours vers le système local (« localhost »), et le réseau 127.0.0.0/8 appartient au système local, de manière à pouvoir communiquer avec lui-même en utilisant les protocoles de réseau.

Routage IPv4

Que ce soit avec IPv4 ou IPv6, le trafic réseau doit passer d'un hôte à l'autre et d'un réseau à l'autre. Chaque hôte possède une *table de routage*, qui l'instruit sur la façon de diriger le trafic pour certains réseaux. Une entrée de la table de routage liste le réseau de destination, l'interface à utiliser pour l'envoi du trafic et l'adresse IP de tout routeur intermédiaire nécessaire pour relayer le message vers sa destination finale. L'entrée de la table de routage correspondant à la destination du trafic réseau sert à le diriger, ou « router ». Si deux entrées sont équivalentes, celle avec le plus long préfixe est utilisée.

Si le trafic du réseau ne suit pas une route spécifique, le tableau de routage comprend généralement une entrée *default route* vers l'ensemble de l'Internet IPv4 : 0.0.0.0/0. Cette route par défaut pointe vers un *routeur* d'un sous-réseau accessible (c'est-à-dire un sous-réseau qui possède une route plus spécifique que celle indiquée dans le tableau de routage de l'hôte).

Si un routeur reçoit du trafic qui ne lui est pas destiné, plutôt que de l'ignorer comme un hôte normal, il le *fait suivre* en fonction des informations de sa propre table de routage. Cela peut envoyer le trafic directement vers l'hôte de destination (s'il se trouve que le routeur se trouve sur le sous-réseau de destination), ou le faire suivre à un autre routeur. Ce processus de transfert continue jusqu'à ce que le trafic arrive à sa destination finale.

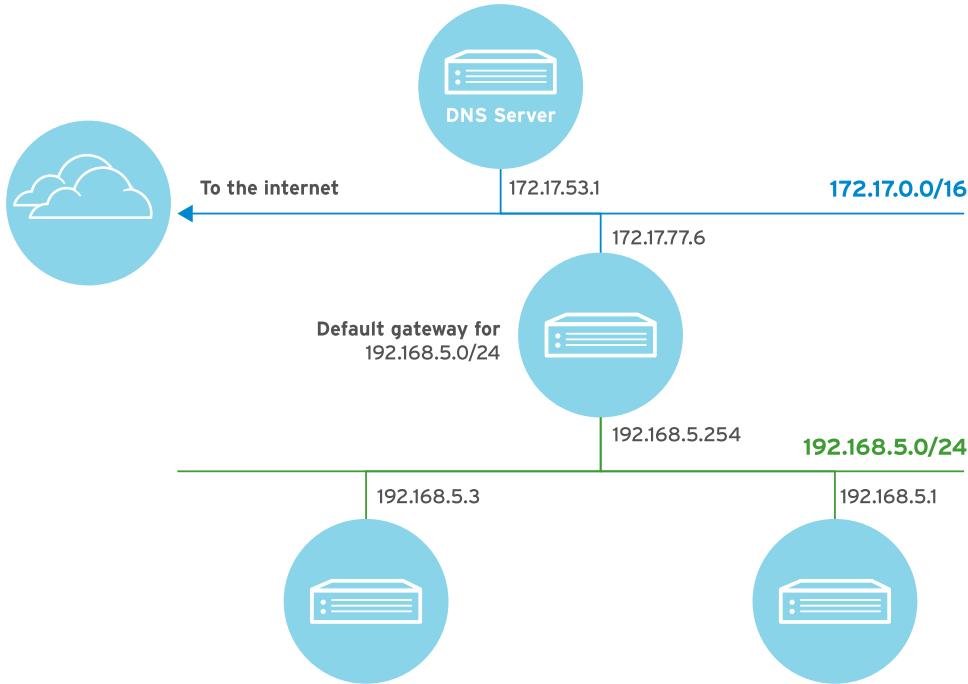


Figure 12.2: Exemple de topologie réseau

Exemple de table de routage

DESTINATION	INTERFACE	ROUTEUR (SI NÉCESSAIRE)
192.0.2.0/24	wlo1	
192.168.5.0/24	enp3s0	
0.0.0.0/0 (par défaut)	enp3s0	192.168.5.254

Dans cet exemple, le trafic entre cet hôte et l'adresse IP 192.0.2.102 est transmis directement vers sa destination via l'interface sans fil wlo1, car c'est celle qui correspond le plus à la route 192.0.2.0/24. Le trafic vers l'adresse IP 192.168.5.3 est transmis directement à cette destination via l'interface Ethernet enp3s0, car c'est celle qui correspond le plus à la route 192.168.5.0/24.

Le trafic destiné à l'adresse IP 10.2.24.1 est transmis de l'interface Ethernet enp3s0 au routeur 192.168.5.254, qui fera suivre ce trafic vers sa destination finale. Ce trafic correspond le plus à la route 0.0.0.0/0 puisqu'il n'y a pas d'itinéraire plus spécifique sur la table de routage de cet hôte. Le routeur utilise sa propre table de routage pour déterminer vers où faire suivre ce trafic par la suite.

Configuration de l'adresse IPv4 et de la route

Un serveur peut configurer automatiquement ses paramètres réseau IPv4 au démarrage à partir d'un serveur DHCP. Un démon de client local interroge le lien pour un serveur et les paramètres réseau, et obtient un bail permettant d'utiliser ces paramètres pendant une durée spécifique.

Si le client ne demande pas un renouvellement du bail périodiquement, il risque de perdre ses paramètres de configuration réseau.

Alternativement, vous pouvez configurer un serveur pour utiliser une configuration réseau *statique*. Dans ce cas, les paramètres réseau sont lus à partir des fichiers de configuration locaux. Vous devez obtenir les paramètres corrects auprès de votre administrateur réseau et les mettre à jour manuellement si nécessaire pour éviter les conflits avec d'autres serveurs.

MISE EN RÉSEAU IPV6

IPv6 a été conçu pour éventuellement remplacer le protocole réseau IPv4. Il faudra que vous compreniez son fonctionnement car de plus en plus de systèmes de production utilisent l'adressage IPv6. Par exemple, de nombreux fournisseurs de services Internet utilisent déjà IPv6 pour les réseaux de communication interne et de gestion de périphériques afin de conserver les rares adresses IPv4 pour les besoins des clients.

IPv6 peut également être utilisé en parallèle avec IPv4 dans un modèle à *double pile*. Dans cette configuration, une interface réseau peut avoir une ou plusieurs adresses IPv6 ainsi que des adresses IPv4. Red Hat Enterprise Linux fonctionne en mode à double pile par défaut.

Adresses IPv6

Une adresse IPv6 est un numéro codé sur 128 bits, normalement exprimé sous forme de 8 groupes de 4 quartets hexadécimaux séparés par des deux-points (demi-octets). Chaque quartet représente quatre bits de l'adresse IPv6, de sorte que chaque groupe représente 16 bits de l'adresse IPv6.

```
2001:0db8:0000:0010:0000:0000:0000:0001
```

Pour faciliter l'écriture des adresses IPv6, les zéros du début d'un groupe séparé par des deux-points sont facultatifs. Cependant, au moins un chiffre hexadécimal doit être écrit dans chaque groupe séparé par des deux-points.

```
2001:db8:0:10:0:0:0:1
```

Étant donné que les adresses composées de longues chaînes de zéros sont courantes, il est possible de combiner un ou plusieurs groupes de zéros consécutifs pour former *exactement un seul bloc* ::.

```
2001:db8:0:10::1
```

Notez que, conformément à ces règles, `2001:db8::0010:0:0:0:1` constitue une méthode moins pratique pour écrire l'exemple d'adresse. Il s'agit toutefois d'une représentation valide de la même adresse, ce qui peut prêter à confusion pour les administrateurs qui n'ont pas l'habitude d'utiliser IPv6. Voici quelques astuces pour écrire des adresses raisonnablement cohérentes :

- Supprimez les zéros non significatifs dans un groupe.
- Utilisez le symbole :: pour raccourcir l'adresse le plus possible.
- Si une adresse contient deux groupes de zéros consécutifs, de longueur égale, il est préférable de raccourcir les groupes de zéros les plus à gauche de :: et les groupes les plus à droite de :: pour chaque groupe.

- Même si c'est autorisé, n'utilisez pas :: pour raccourcir un groupe de zéros. Utilisez :0: à la place, et économisez :: pour des groupes de zéros consécutifs.
- Utilisez toujours des minuscules pour les nombres hexadécimaux a à f.

**IMPORTANT**

Lors de l'insertion d'un port réseau TCP ou UDP à la suite d'une adresse IPv6, placez toujours l'adresse IPv6 entre crochets pour que le numéro de port ne semble pas faire partie de l'adresse.

[2001:db8:0:10::1]:80

Sous-réseau IPv6

Une adresse de monodiffusion IPv6 normale est divisée en deux parties : le *préfixe réseau* et l'*ID d'interface*. Le préfixe réseau identifie le sous-réseau. Deux interfaces sur le même sous-réseau ne peuvent pas avoir le même ID d'interface ; l'ID d'interface identifie une interface spécifique dans le sous-réseau.

Contrairement au protocole IPv4, IPv6 dispose d'un masque de sous-réseau standard utilisé pour la plupart des adresses normales, /64. Dans ce cas, la moitié de l'adresse est le préfixe de réseau et l'autre moitié est l'ID d'interface. Cela signifie qu'un seul sous-réseau peut contenir autant d'hôtes que nécessaire.

En règle générale, l'opérateur réseau *alloue* un préfixe plus court à une organisation, par exemple /48. Cela permet de conserver le reste de la partie dédiée au réseau pour l'attribution de sous-réseaux (toujours avec la longueur /64) à partir de ce préfixe alloué. Pour une allocation /48 par exemple, cela laisse 16 bits pour les sous-réseaux (jusqu'à 65 536 sous-réseaux).

IPv6 address is 2001:db8:0:1::1/64

Allocation from provider is 2001:db8::/48

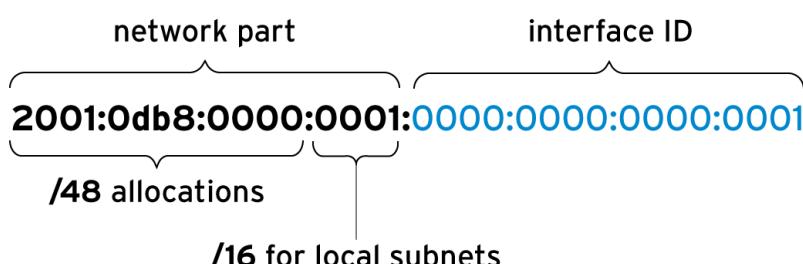


Figure 12.3: Parties d'adresses IPv6 et sous-réseau

Adresses IPv6 courantes et réseaux

ADRESSE IPV6 OU RÉSEAU	OBJET	DESCRIPTION
::1/128	hôte local	L'équivalent IPv6 de 127.0.0.1/8 qui est défini sur l'interface de bouclage.

ADRESSE IPV6 OU RÉSEAU	OBJET	DESCRIPTION
::	adresse non spécifiée	L'équivalent IPv6 de 0 . 0 . 0 . 0. Pour un service réseau, cela peut indiquer qu'il écoute sur toutes les adresses IP configurées.
::/0	acheminement par défaut (Internet IPv6)	L'équivalent IPv6 de 0 . 0 . 0 . 0/0. L'acheminement par défaut dans la table de routage correspond à ce réseau ; le routeur de ce réseau correspond à l'endroit où tout le trafic est envoyé en l'absence d'un meilleur acheminement.
2000::/3	adresses de monodiffusion globales	Les adresses IPv6 « normales » sont actuellement allouées depuis cet espace par IANA. Il s'agit de l'équivalent de tous les réseaux compris entre 2000::/16 et 3fff::/16.
fd00::/8	adresses locales uniques (RFC 4193)	IPv6 n'a aucun équivalent direct de l'espace d'adresse privé RFC 1918, même si celui-ci est proche. Un site peut les utiliser pour s'auto-attribuer un espace d'adresses IP routable privé au sein de l'organisation, mais ces réseaux ne peuvent pas être utilisés sur l'Internet global. Le site doit sélectionner de manière aléatoire un /48 dans cet espace, mais il peut allouer le sous-réseau dans les réseaux /64 normalement.
fe80::/10	Adresses link-local	Chaque interface IPv6 configure automatiquement une adresse de monodiffusion <i>link-local</i> qui fonctionne uniquement sur le lien local du réseau fe80::/64. Cependant, l'ensemble de la plage fe80::/10 est réservé pour une utilisation future par le lien local. Cela sera décrit plus en détails ultérieurement.
ff00::/8	Multidiffusion	L'équivalent IPv6 de 224 . 0 . 0 . 0/4. La multidiffusion permet de transmettre des informations simultanément à plusieurs hôtes et est particulièrement importante pour le protocole IPv6 car celui-ci ne dispose pas d'adresse de diffusion.

**IMPORTANT**

Le tableau ci-dessus liste les *allocations* d'adresses réseau qui sont réservées à des fins spécifiques. Ces allocations peuvent consister en de nombreux réseaux différents. N'oubliez pas que les réseaux IPv6 alloués à partir des espaces de monodiffusion globale et de monodiffusion link-local ont un masque de sous-réseau /64 standard.

Avec le protocole IPv6, une adresse *link-local* est une adresse impossible à acheminer, utilisée uniquement pour communiquer avec les hôtes d'un lien réseau spécifique. Chaque interface réseau du système est automatiquement configurée avec une adresse link-local sur le réseau `fe80::/64`. Pour s'assurer qu'il est unique, l'ID d'interface de l'adresse link-local est construit à partir de l'adresse matérielle Ethernet de l'interface réseau. La procédure habituelle pour convertir l'adresse MAC de 48 bits en un ID d'interface de 64 bits consiste à inverser le bit 7 de l'adresse MAC et à insérer `ff:fe` entre ses deux octets centraux.

- Préfixe réseau : `fe80::/64`
- Adresse MAC : `00:11:22:aa:bb:cc`
- Adresse link-local : `fe80::211:22ff:fea:bbcc/64`

Les adresses link-local d'autres machines peuvent être utilisées comme des adresses normales par les hôtes sur le même lien. Étant donné que chaque lien a un réseau `fe80::/64`, il est impossible d'utiliser la table de routage pour sélectionner correctement l'interface de sortie. Le lien à utiliser lorsqu'il s'agit d'une adresse link-local doit être spécifié par un *identificateur d'étendue* à la fin de l'adresse. L'identificateur d'étendue se compose du symbole % suivi du nom de l'interface réseau.

Par exemple, si l'on veut utiliser `ping6` pour effectuer un test ping de l'adresse link-local `fe80::211:22ff:fea:bbcc` à l'aide du lien connecté à l'interface réseau `ens3`, la syntaxe correcte à utiliser est la suivante :

```
[user@host ~]$ ping6 fe80::211:22ff:fea:bbcc%ens3
```

**NOTE**

Les identificateurs d'étendue ne sont nécessaires que si l'on doit contacter des adresses dont l'étendue est « link ». Les adresses globales normales sont utilisées comme dans le protocole IPv4, et sélectionnent leurs interfaces de sortie dans la table de routage.

La *multidiffusion* permet à un système d'envoyer le trafic vers une adresse IP spéciale reçue par plusieurs systèmes. Elle diffère de la diffusion puisque seuls des systèmes spécifiques du réseau reçoivent le trafic. Elle diffère également de la diffusion dans IPv4, car une partie du trafic multidiffusion peut être acheminée vers d'autres sous-réseaux, en fonction de la configuration de vos routeurs et systèmes.

La multidiffusion joue un rôle plus important sous IPv6 que sous IPv4 car le protocole IPv6 ne comporte aucune adresse de diffusion. La principale adresse de multidiffusion sous IPv6 est `ff02::1`, l'adresse link-local de tous les nœuds. L'exécution d'un test ping de cette adresse envoie le trafic sur tous les nœuds du lien. Les adresses de multidiffusion dont l'étendue est le lien (qui commencent par `ff02::/8`) doivent être spécifiées avec un identificateur d'étendue, tout comme une adresse link-local.

```
[user@host ~]$ ping6 ff02::1%ens3
PING ff02::1%ens3(ff02::1) 56 data bytes
64 bytes from fe80::211:22ff:feaa:bbcc: icmp_seq=1 ttl=64 time=0.072 ms
64 bytes from fe80::200:aaff:fe33:2211: icmp_seq=1 ttl=64 time=102 ms (DUP!)
64 bytes from fe80::bcd:ffff:fea1:b2c3: icmp_seq=1 ttl=64 time=103 ms (DUP!)
64 bytes from fe80::211:22ff:feaa:bbcc: icmp_seq=2 ttl=64 time=0.079 ms
...output omitted...
```

Configuration d'une adresse IPv6

IPv4 propose deux manières de configurer les adresses sur les interfaces réseau. Les adresses réseau peuvent être configurées manuellement sur les interfaces par l'administrateur, ou de manière dynamique à partir du réseau via DHCP. IPv6 prend également en charge la configuration manuelle et les deux méthodes de configuration dynamique, dont l'une d'elles s'appelle DHCPv6.

Les ID d'interface des adresses IPv6 statiques peuvent être sélectionnées à volonté, tout comme pour le protocole IPv4. Avec le protocole IPv4, deux adresses sur un réseau ne pouvaient pas être utilisées : l'adresse la plus basse dans le sous-réseau et l'adresse la plus haute dans le sous-réseau. Avec le protocole IPv6, les ID d'interface suivants sont réservés et ne peuvent pas être utilisés pour une adresse réseau normale sur un hôte.

- L'identificateur ne comportant que des zéros `0000:0000:0000:0000` (« routeur de sous-réseau anycast ») qui est utilisé par tous les routeurs sur le lien. (Pour le réseau `2001:db8::/64`, il s'agit de l'adresse `2001:db8::`)
- Les identificateurs `fdff:ffff:ffff:ff80` à `fdff:ffff:ffff:ffff`.

DHCPv6 fonctionne différemment de DHCP pour IPv4 car il ne comporte pas d'adresse de diffusion. Schématiquement, un hôte envoie une requête DHCPv6 à partir de son adresse link-local vers le port 547/UDP sur `ff02::1:2`, le groupe de multidiffusion link-local de tous les serveurs `dhcp`. En général, le serveur DHCPv6 envoie alors une réponse contenant les informations appropriées au port 546/UDP sur l'adresse link-local du client.

Le paquetage `dhclient` dans Red Hat Enterprise Linux 8 prend en charge un serveur DHCPv6.

Outre IDHCPv6, IPv6 prend également en charge une deuxième méthode de configuration dynamique appelée *autoconfiguration sans état* (*SLAAC, Stateless Autoconfiguration*). Avec la méthode SLAAC, l'hôte active son interface normalement avec une adresse link-local `fe80::/64`. Il envoie ensuite une « sollicitation du routeur » à `ff02::2`, le groupe de multidiffusion link-local de tous les routeurs. Un routeur IPv6 sur le lien local répond à l'adresse link-local de l'hôte avec un préfixe réseau et d'autres informations s'il y en a. L'hôte utilise ensuite ce préfixe réseau avec un ID d'interface qu'il génère normalement de la même manière que les adresses link-local. Le routeur envoie régulièrement des mises à jour de multidiffusion (« annonces de routage ») pour confirmer ou mettre à jour les informations qu'il a préalablement fournies.

Le paquetage `radvd` dans Red Hat Enterprise Linux 8 permet à un routeur IPv6 basé sur Red Hat Enterprise Linux de fournir la méthode SLAAC via les annonces de routage.

**IMPORTANT**

Une machine Red Hat Enterprise Linux 8 type configurée pour obtenir des adresses IPv4 via DHCP est généralement configurée aussi pour utiliser la méthode SLAAC afin d'obtenir des adresses IPv6. Par conséquent, ces machines risquent d'obtenir des adresses IPv6 de manière inattendue lorsqu'un routeur IPv6 est ajouté au réseau.

Certains déploiements IPv6 combinent les méthodes SLAAC et DHCPv6, et utilisent la configuration SLAAC pour fournir uniquement les informations d'adresses réseau et DHCPv6 pour transmettre d'autres informations telles que les serveurs DNS et les domaines de recherche à configurer.

NOMS D'HÔTES ET ADRESSES IP

Il serait peu pratique de toujours utiliser des adresses IP pour contacter vos serveurs. Les humains préfèrent généralement travailler avec des noms plutôt qu'avec des chaînes de nombres longues et difficiles à retenir. Et Linux offre donc un certain nombre de mécanismes permettant de mapper un nom d'hôte à une adresse IP, collectivement appelés *résolution de noms*.

L'une des méthodes consiste à définir une entrée statique pour chaque nom dans le fichier **/etc/hosts** sur chaque système. Cela nécessite de mettre à jour manuellement la copie du fichier de chaque serveur.

Pour la plupart des hôtes, vous pouvez rechercher l'adresse d'un nom d'hôte (ou un nom d'hôte à partir d'une adresse) à partir d'un service réseau appelé *système de noms de domaines (DNS, Domain Name System)*. Le DNS est un réseau distribué de serveurs fournissant des mappages de noms d'hôtes avec des adresses IP. Pour que le service de noms fonctionne, un hôte doit pointer vers un *serveur de noms*. Ce serveur de noms ne doit pas nécessairement se trouver sur le même sous-réseau : l'hôte doit simplement pouvoir y accéder. Ceci est généralement configuré via DHCP ou un paramètre statique dans un fichier appelé **/etc/resolv.conf**. Les sections suivantes de ce chapitre vont expliquer comment configurer la résolution de noms.



RÉFÉRENCES

Pages de manuel `services(5)`, `ping(8)`, `biosdevname(1)` et `udev(7)`

Pour plus d'informations, reportez-vous à la section *Configuring and Managing Networking* dans *Red Hat Enterprise Linux 8.0* à l'adresse
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_and_managing_networking/

Compréhension des noms prévisibles des périphériques réseau de systemd

<https://major.io/2015/08/21/understanding-systemds-predictable-network-device-names/>

Références IETF RFC sélectionnées :

RFC 2460 : Protocole Internet, version 6 (IPv6) – Spécifications

<http://tools.ietf.org/html/rfc2460>

RFC 4291 : Architecture d'adressage IP version 6

<http://tools.ietf.org/html/rfc4291>

RFC 5952 : Recommandation pour la représentation textuelle de l'adressage IPv6

<http://tools.ietf.org/html/rfc5952>

RFC 4862 : Configuration automatique d'adresses sans état IPv6

<http://tools.ietf.org/html/rfc4862>

RFC 3315 : Protocole de configuration d'hôtes dynamique pour IPv6 (DHCPv6)

<http://tools.ietf.org/html/rfc3315>

RFC 3736 : Service de protocole de configuration d'hôte dynamique (DHCP) sans état pour IPv6

<http://tools.ietf.org/html/rfc3736>

RFC 4193 : Adresses de monodiffusion IPv6 locales uniques

<http://tools.ietf.org/html/rfc4193>

► QUIZ

DESCRIPTION DES CONCEPTS RÉSEAU

Répondez aux questions suivantes en sélectionnant une réponse :

► 1. Quel nombre représente la taille, en bits, d'une adresse IPv4 ?

- a. 4
- b. 8
- c. 16
- d. 32
- e. 64
- f. 128

► 2. Quel terme détermine combien de bits principaux dans l'adresse IP contribuent à son adresse réseau ?

- a. netscope
- b. masque de réseau
- c. sous-réseau
- d. multidiffusion
- e. netaddr
- f. réseau

► 3. Quelle adresse représente une adresse IP d'hôte IPv4 valide ?

- a. 192.168.1.188
- b. 192.168.1.0
- c. 192.168.1.255
- d. 192.168.1.256

► 4. Quel nombre représente la taille, en bits, d'une adresse IPv6 ?

- a. 4
- b. 8
- c. 16
- d. 32
- e. 64
- f. 128

► 5. Quelle adresse ne représente pas une adresse IPv6 valide ?

- a. 2000:0000:0000:0000:0000:0000:0000:0001
- b. 2::1
- c. ::
- d. ff02::1:0:0
- e. 2001:3::7:0:2
- f. 2001:db8::7::2
- g. 2000::1

► 6. Quel terme décrit un système qui envoie le trafic vers une adresse IP spéciale reçue par plusieurs systèmes ?

- a. netscope
- b. masque de réseau
- c. sous-réseau
- d. multidiffusion
- e. netaddr
- f. réseau

► SOLUTION

DESCRIPTION DES CONCEPTS RÉSEAU

Répondez aux questions suivantes en sélectionnant une réponse :

► 1. Quel nombre représente la taille, en bits, d'une adresse IPv4 ?

- a. 4
- b. 8
- c. 16
- d. 32
- e. 64
- f. 128

► 2. Quel terme détermine combien de bits principaux dans l'adresse IP contribuent à son adresse réseau ?

- a. netscope
- b. masque de réseau
- c. sous-réseau
- d. multidiffusion
- e. netaddr
- f. réseau

► 3. Quelle adresse représente une adresse IP d'hôte IPv4 valide ?

- a. 192.168.1.188
- b. 192.168.1.0
- c. 192.168.1.255
- d. 192.168.1.256

► 4. Quel nombre représente la taille, en bits, d'une adresse IPv6 ?

- a. 4
- b. 8
- c. 16
- d. 32
- e. 64
- f. 128

► 5. Quelle adresse ne représente pas une adresse IPv6 valide ?

- a. 2000:0000:0000:0000:0000:0000:0000:0001
- b. 2::1
- c. ::
- d. ff02::1:0:0
- e. 2001:3::7:0:2
- f. 2001:db8::7::2
- g. 2000::1

► 6. Quel terme décrit un système qui envoie le trafic vers une adresse IP spéciale reçue par plusieurs systèmes ?

- a. netscope
- b. masque de réseau
- c. sous-réseau
- d. multidiffusion
- e. netaddr
- f. réseau

VALIDATION DE LA CONFIGURATION RÉSEAU

OBJECTIFS

Au terme de cette section, vous serez en mesure de tester et d'inspecter la configuration réseau actuelle à l'aide d'utilitaires de ligne de commande.

COLLECTE D'INFORMATIONS SUR L'INTERFACE RÉSEAU

Identification des interfaces réseau

La commande **ip link** liste toutes les interfaces réseau disponibles sur votre système :

```
[user@host ~]$ ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
    group default qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT
    group default qlen 1000
        link/ether 52:54:00:00:00:0a brd ff:ff:ff:ff:ff:ff
3: ens4: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT
    group default qlen 1000
        link/ether 52:54:00:00:00:1e brd ff:ff:ff:ff:ff:ff
```

Dans l'exemple précédent, le serveur possède trois interfaces réseau : **lo**, qui correspond au périphérique de bouclage connecté au serveur lui-même et deux interfaces Ethernet, **ens3** et **ens4**.

Pour configurer correctement chaque interface réseau, vous devez savoir quelle interface est connectée à quel réseau. Dans de nombreux cas, vous connaîtrez l'adresse MAC de l'interface connectée à chaque réseau, soit parce qu'elle est imprimée physiquement sur la carte ou le serveur, soit parce qu'il s'agit d'une machine virtuelle et vous connaissez sa configuration. L'adresse MAC du périphérique est indiquée après **link/ether** pour chaque interface. Vous savez donc que la carte réseau ayant l'adresse MAC **52:54:00:00:00:0a** est l'interface réseau **ens3**.

Affichage des adresses IP

Utilisez la commande **ip** pour afficher les informations sur le périphérique et son adresse. Une seule interface réseau peut avoir plusieurs adresses IPv4 ou IPv6.

```
[user@host ~]$ ip addr show ens3
2: ens3: <BROADCAST,MULTICAST,①UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
    qlen 1000
        ②link/ether 52:54:00:00:00:0b brd ff:ff:ff:ff:ff:ff
        ③inet 192.0.2.2/24 brd 192.0.2.255 scope global ens3
            valid_lft forever preferred_lft forever
        ④inet6 2001:db8:0:1:5054:ff:fe00:b/64 scope global
```

```
valid_lft forever preferred_lft forever
⑤inet6 fe80::5054:ff:fe00:b/64 scope link
    valid_lft forever preferred_lft forever
```

- ➊ Une interface active est UP.
- ➋ La ligne `link/ether` spécifie l'adresse matérielle (MAC) du périphérique.
- ➌ La ligne `inet` présente une adresse IPv4, la longueur du préfixe et l'étendue de son réseau.
- ➍ La ligne `inet6` présente une adresse IPv6, la longueur du préfixe et l'étendue de son réseau. Cette adresse a une étendue *globale* et est utilisée normalement.
- ➎ Cette ligne `inet6` montre que l'interface a une adresse IPv6 d'étendue *link* qui ne peut être utilisée que pour les communications sur la liaison Ethernet locale.

Affichage des statistiques de performance

La commande `ip` peut également servir à afficher les statistiques relatives à la performance du réseau. Des compteurs pour chaque interface réseau peuvent être utilisés pour identifier la présence de problèmes de réseau. Les compteurs enregistrent des statistiques pour des éléments comme le nombre de paquetages reçus (RX) et transmis (TX), les erreurs de paquetage et les paquetages qui ont été abandonnés.

```
[user@host ~]$ ip -s link show ens3
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 52:54:00:00:00:0a brd ff:ff:ff:ff:ff:ff
        RX: bytes   packets   errors   dropped overrun mcast
            269850      2931       0       0       0       0
        TX: bytes   packets   errors   dropped carrier collsns
            300556      3250       0       0       0       0
```

VÉRIFICATION DE LA CONNECTIVITÉ ENTRE LES HÔTES

La commande `ping` sert à tester la connectivité. La commande continue à s'exécuter jusqu'à ce que l'utilisateur appuie sur **Ctrl+c**, à moins que des options limitent le nombre de paquetages envoyés.

```
[user@host ~]$ ping -c3 192.0.2.254
PING 192.0.2.1 (192.0.2.254) 56(84) bytes of data.
64 bytes from 192.0.2.254: icmp_seq=1 ttl=64 time=4.33 ms
64 bytes from 192.0.2.254: icmp_seq=2 ttl=64 time=3.48 ms
64 bytes from 192.0.2.254: icmp_seq=3 ttl=64 time=6.83 ms

--- 192.0.2.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 3.485/4.885/6.837/1.424 ms
```

La commande `ping6` est la version IPv6 de la commande `ping` dans Red Hat Enterprise Linux. Mis à part le fait qu'elle communique via IPv6 et accepte des adresses IPv6, elle fonctionne comme la commande `ping`.

```
[user@host ~]$ ping6 2001:db8:0:1::1
PING 2001:db8:0:1::1(2001:db8:0:1::1) 56 data bytes
64 bytes from 2001:db8:0:1::1: icmp_seq=1 ttl=64 time=18.4 ms
```

CHAPITRE 12 | Gestion de réseaux

```
64 bytes from 2001:db8:0:1::1: icmp_seq=2 ttl=64 time=0.178 ms
64 bytes from 2001:db8:0:1::1: icmp_seq=3 ttl=64 time=0.180 ms
^C
--- 2001:db8:0:1::1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 0.178/6.272/18.458/8.616 ms
[user@host ~]$
```

Lorsque vous envoyez la commande ping à des adresses link-local et au groupe de multidiffusion de tous les noeuds link-local (`ff02::1`), l'interface réseau à utiliser doit être spécifiée explicitement par un identificateur de zone d'étendue (tel que `ff02::1%eth0`). S'il est omis, l'erreur `connect: Invalid argument` s'affiche alors.

Effectuer un test ping de `ff02::1` peut être utile pour trouver d'autres noeuds IPv6 dans le réseau local.

```
[user@host ~]$ ping6 ff02::1%ens4
PING ff02::1%ens4(ffff::1) 56 data bytes
64 bytes from fe80::78cf:7fff:fed2:f97b: icmp_seq=1 ttl=64 time=22.7 ms
64 bytes from fe80::f482:dbff:fe25:6a9f: icmp_seq=1 ttl=64 time=30.1 ms (DUP!)
64 bytes from fe80::78cf:7fff:fed2:f97b: icmp_seq=2 ttl=64 time=0.183 ms
64 bytes from fe80::f482:dbff:fe25:6a9f: icmp_seq=2 ttl=64 time=0.231 ms (DUP!)
^C
--- ff02::1%ens4 ping statistics ---
2 packets transmitted, 2 received, +2 duplicates, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.183/13.320/30.158/13.374 ms
[user@host ~]$ ping6 -c 1 fe80::f482:dbff:fe25:6a9f%ens4
PING fe80::f482:dbff:fe25:6a9f%ens4(fe80::f482:dbff:fe25:6a9f) 56 data bytes
64 bytes from fe80::f482:dbff:fe25:6a9f: icmp_seq=1 ttl=64 time=22.9 ms

--- fe80::f482:dbff:fe25:6a9f%ens4 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 22.903/22.903/22.903/0.000 ms
```

Rappelez-vous que les adresses link-local IPv6 peuvent être utilisées par d'autres hôtes sur le même lien, tout comme des adresses normales.

```
[user@host ~]$ ssh fe80::f482:dbff:fe25:6a9f%ens4
user@fe80::f482:dbff:fe25:6a9f%ens4's password:
Last login: Thu Jun  5 15:20:10 2014 from host.example.com
[user@server ~]$
```

DÉPANNAGE DU ROUTAGE

Le routage réseau est complexe et parfois le trafic ne se comporte pas comme prévu. Voici quelques outils de diagnostic utiles.

Affichage de la table de routage

Utilisez la commande `ip` avec l'option `route` pour afficher les informations de routage.

```
[user@host ~]$ ip route
default via 192.0.2.254 dev ens3 proto static metric 1024
192.0.2.0/24 dev ens3 proto kernel scope link src 192.0.2.2
10.0.0.0/8 dev ens4 proto kernel scope link src 10.0.0.11
```

Ceci affiche la table de routage IPv4. Tous les paquetages destinés au réseau 10.0.0.0/8 sont envoyés directement à destination via le périphérique ens4. Tous les paquetages destinés au réseau 192.0.2.0/24 sont envoyés directement à destination via le périphérique ens3. Tous les autres paquetages sont envoyés au routeur par défaut situé sur 192.0.2.254, mais également par l'intermédiaire du périphérique ens3.

Ajoutez l'option **-6** pour afficher la table de routage IPv6 :

```
[user@host ~]$ ip -6 route
unreachable ::/96 dev lo metric 1024 error -101
unreachable ::ffff:0.0.0.0/96 dev lo metric 1024 error -101
2001:db8:0:1::/64 dev ens3 proto kernel metric 256
unreachable 2002:a00::/24 dev lo metric 1024 error -101
unreachable 2002:7f00::/24 dev lo metric 1024 error -101
unreachable 2002:a9fe::/32 dev lo metric 1024 error -101
unreachable 2002:ac10::/28 dev lo metric 1024 error -101
unreachable 2002:c0a8::/32 dev lo metric 1024 error -101
unreachable 2002:e000::/19 dev lo metric 1024 error -101
unreachable 3ffe:ffff::/32 dev lo metric 1024 error -101
fe80::/64 dev ens3 proto kernel metric 256
default via 2001:db8:0:1::ffff dev ens3 proto static metric 1024
```

Dans cet exemple, ignorez les routes inaccessibles, qui pointent vers des réseaux inutilisés. Il reste donc trois routes :

1. Le réseau 2001:db8:0:1::/64, en utilisant l'interface ens3 (qui a sûrement une adresse sur ce réseau).
2. Le réseau fe80::/64, en utilisant l'interface ens3, pour l'adresse link-local. Dans un système à plusieurs interfaces, il existe une route vers fe80::/64 sur chaque interface pour chaque adresse link-local.
3. Une route par défaut vers tous les réseaux sur l'Internet IPv6 (le réseau ::/0) qui n'a pas de route plus spécifique dans le système, via le routeur à l'adresse 2001:db8:0:1::ffff qui peut être atteint via le périphérique ens3.

Traçage des routes empruntées par le trafic

Pour tracer le chemin emprunté par le trafic réseau pour atteindre un hôte distant via plusieurs routeurs, utilisez **traceroute** ou **tracepath**. Cela permet d'identifier si l'un de vos routeurs ou un routeur intermédiaire pose un problème. Les deux commandes utilisent des paquetages UDP pour tracer un chemin par défaut. Cependant, de nombreux réseaux bloquent les trafics UDP et ICMP. La commande **traceroute** propose des options pour tracer le chemin avec des paquetages UDP (par défaut), ICMP (-I) ou TCP (-T). Cependant, en règle générale, la commande **traceroute** n'est pas installée par défaut.

```
[user@host ~]$ tracepath access.redhat.com
...output omitted...
4: 71-32-28-145.rcmt.qwest.net          48.853ms asymm 5
```

CHAPITRE 12 | Gestion de réseaux

```

5: dcp-brdr-04.inet.qwest.net          100.732ms asymm 7
6: 206.111.0.153.ptr.us.xo.net        96.245ms asymm 7
7: 207.88.14.162.ptr.us.xo.net        85.270ms asymm 8
8: ae1d0.cir1.atlanta6-ga.us.xo.net   64.160ms asymm 7
9: 216.156.108.98.ptr.us.xo.net      108.652ms
10: bu-ether13.atlngamq46w-bcr00.tbone.rr.com 107.286ms asymm 12
...output omitted...

```

Chaque ligne de la sortie de **tracepath** représente un routeur ou un saut que le paquet traverse entre sa source et sa destination finale. Des informations complémentaires sont fournies quand elles sont disponibles, y compris le *temps aller-retour (RTT)* et tout changement dans la taille de l'*unité de transmission maximale (MTU)*. L'indication **asymm** signifie que le trafic a atteint ce routeur et est revenu de ce routeur en utilisant différentes routes (*asymmetric*). Les routeurs présentés sont ceux utilisés pour le trafic sortant, pas le trafic de retour.

Les commandes **tracepath6** et **traceroute -6** sont l'équivalent des commandes **tracepath** et **traceroute** pour IPv6.

```
[user@host ~]$ tracepath6 2001:db8:0:2::451
1?: [LOCALHOST]                      0.091ms pmtu 1500
1: 2001:db8:0:1::ba                  0.214ms
2: 2001:db8:0:1::1                   0.512ms
3: 2001:db8:0:2::451                0.559ms reached
                                         Resumé: pmtu 1500 hops 3 back 3
```

DÉPANNAGE DES PORTS ET DES SERVICES

Les services TCP utilisent des sockets comme points de terminaison pour les communications. Ils sont constitués d'une adresse IP, d'un protocole et d'un numéro de port. Les services écoutent généralement sur les ports standard, alors que les clients utilisent un port disponible aléatoire. Les noms connus des ports standard sont répertoriés dans le fichier **/etc/services**.

La commande **ss** sert à afficher les statistiques des sockets. La commande **ss** est destinée à remplacer l'ancien outil **netstat**, faisant partie du paquetage *net-tools*, qui peut être mieux connu des administrateurs système, mais n'est pas toujours installé.

```
[user@host ~]$ ss -ta
State     Recv-Q Send-Q      Local Address:Port          Peer Address:Port
LISTEN      0      128          *:sunrpc                 *:*
LISTEN      0      128          ①*:ssh                  *:*
LISTEN      0      100          ②127.0.0.1:smtp       *:*
LISTEN      0      128          *:36889                 *:*
ESTAB       0      0            ③172.25.250.10:ssh    172.25.254.254:59392
LISTEN      0      128          :::sunrpc               :::*
LISTEN      0      128          ④:::ssh                 :::*
LISTEN      0      100          ⑤::1:smtp               :::*
LISTEN      0      128          :::34946                 :::*
```

- ➊ Le port utilisé pour SSH écoute sur toutes les adresses IPv4. Le « * » sert à représenter « tous » lorsqu'il est fait référence à des adresses ou des ports IPv4.
- ➋ Le port utilisé pour SMTP écoute sur l'interface de bouclage IPv4 127.0.0.1.
- ➌ La connexion SSH établie se trouve sur l'interface 172.25.250.10 et provient d'un système dont l'adresse est 172.25.254.254.

- ④ Le port utilisé pour SSH écoute sur toutes les adresses IPv6. La syntaxe « :: » représente toutes les interfaces IPv6.
- ⑤ Le port utilisé pour SMTP écoute sur l'interface de bouclage IPv6 « ::1 ».

Options pour ss et netstat

OPTION	DESCRIPTION
-n	Affiche le numéro des interfaces et des ports plutôt que leur nom.
-t	Affiche les sockets TCP.
-u	Affiche les sockets UDP.
-l	N'affiche que les sockets d'écoute.
-a	Affiche tous les sockets (d'écoute et connectés).
-p	Affiche le processus qui utilise les sockets.
-A inet	Affiche les connexions actives (mais pas les sockets d'écoute) pour la famille d'adresses <code>inet</code> . Cela signifie que la commande ignore les sockets du domaine UNIX local. Pour ss , à la fois les connexions IPv4 et IPv6 sont affichées. Pour netstat , seules les connexions IPv4 sont affichées. (netstat -A inet6 affiche les connexions IPv6, et netstat -46 affiche à la fois les connexions IPv4 et IPv6.)



RÉFÉRENCES

Pages de manuel `ip-link(8)`, `ip-address(8)`, `ip-route(8)`, `ip(8)`, `ping(8)`, `tracepath(8)`, `traceroute(8)`, `ss(8)` et `netstat(8)`

Pour plus d'informations, reportez-vous à la section *Configuring and Managing Networking* dans *Red Hat Enterprise Linux 8.0* à l'adresse
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_and_managing_networking/

► EXERCICE GUIDÉ

VALIDATION DE LA CONFIGURATION RÉSEAU

Au cours de cet exercice, vous allez inspecter la configuration réseau de l'un de vos serveurs.

RÉSULTATS

Identifier les interfaces réseau actives et les adresses réseau de base.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur **student** à **workstation** avec le mot de passe **student**.

À partir de **workstation**, exécutez la commande **lab net-validate start**. La commande exécute un script de démarrage qui détermine si l'hôte, **servera**, est accessible sur le réseau.

```
[student@workstation ~]$ lab net-validate start
```

- ▶ 1. Utilisez la commande **ssh** pour vous connecter à **servera** en tant qu'utilisateur **student**. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification et pour accéder à **servera** sans mot de passe.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```



IMPORTANT

Les noms d'interface réseau sont déterminés par leur type de bus et l'ordre de détection des périphériques lors du démarrage. Les noms de vos interfaces réseau varient en fonction de la plateforme du cours et du matériel utilisés.

Sur votre système, localisez le nom de l'interface (tel que **ens06** ou **en1p2**) associé à l'adresse Ethernet **52:54:00:00:fa:0a**. Utilisez ce nom d'interface pour remplacer le marqueur **enX** utilisé tout au long de l'exercice.

Localisez le nom de l'interface associé à l'adresse Ethernet **52:54:00:00:fa:0a**. Enregistrez ou mémorisez ce nom et utilisez-le pour remplacer le marqueur **enX** dans les commandes suivantes.

```
[student@servera ~]$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
```

- 3. Affichez l'adresse IP active et le masque de sous-réseau de toutes les interfaces.

```
[student@servera ~]$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
            inet 127.0.0.1/8 scope host lo
                valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host
                valid_lft forever preferred_lft forever
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
            inet 172.25.250.10/24 brd 172.25.250.255 scope global noprefixroute ens3
                valid_lft forever preferred_lft forever
            inet6 fe80::3059:5462:198:58b2/64 scope link noprefixroute
                valid_lft forever preferred_lft forever
```

- 4. Affichez les statistiques de l'interface enX.

```
[student@servera ~]$ ip -s link show enX
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode
    DEFAULT group default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
        RX: bytes   packets   errors   dropped overrun mcast
          89014225     168251       0      154418       0       0
        TX: bytes   packets   errors   dropped carrier collsns
          608808      6090       0       0       0       0
```

- 5. Affichez les informations de routage.

```
[student@servera ~]$ ip route
default via 172.25.250.254 dev enX proto static metric 100
172.25.250.0/24 dev enX proto kernel scope link src 172.25.250.10 metric 100
```

- 6. Vérifiez que le routeur est accessible.

```
[student@servera ~]$ ping -c3 172.25.250.254
PING 172.25.250.254 (172.25.250.254) 56(84) bytes of data.
64 bytes from 172.25.250.254: icmp_seq=1 ttl=64 time=0.196 ms
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.436 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.361 ms
```

```
--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 49ms
rtt min/avg/max/mdev = 0.196/0.331/0.436/0.100 ms
```

- 7. Affichez tous les sauts entre le système local et classroom.example.com.

```
[student@servera ~]$ tracepath classroom.example.com
 1?: [LOCALHOST]                                pmtu 1500
 1: workstation.lab.example.com                0.270ms
 1: workstation.lab.example.com                0.167ms
 2: classroom.example.com                     0.473ms reached
    Resume: pmtu 1500 hops 2 back 2
```

- 8. Affichez les sockets TCP qui écoutent sur le système local.

```
[student@servera ~]$ ss -lt
State      Recv-Q  Send-Q      Local Address:Port      Peer Address:Port
LISTEN      0        128          0.0.0.0:sunrpc      0.0.0.0:*
LISTEN      0        128          0.0.0.0:ssh       0.0.0.0:*
LISTEN      0        128          [::]:sunrpc      [::]:*
LISTEN      0        128          [::]:ssh       [::]:*
```

- 9. Quittez servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Fin

Sur workstation, exéutez le script **lab net-validate finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab net-validate finish
```

L'exercice guidé est maintenant terminé.

CONFIGURATION DE LA MISE EN RÉSEAU À PARTIR DE LA LIGNE DE COMMANDE

OBJECTIFS

Au terme de cette section, vous devez être capable de gérer les paramètres et périphériques réseau au moyen de la commande **nmcli**.

DESCRIPTION DES CONCEPTS NETWORKMANAGER

NetworkManager est un démon qui surveille et gère les paramètres du réseau. Outre le démon, une applet pour la zone de notification de GNOME fournit des informations sur l'état du réseau. Les outils graphiques et de ligne de commande communiquent avec NetworkManager et enregistrent les fichiers de configuration dans le répertoire **/etc/sysconfig/network-scripts**.

- Un *périphérique* est une interface réseau.
- Une *connexion* est un ensemble de paramètres qu'il est possible de configurer pour un périphérique.
- Une seule connexion peut être *active* pour un périphérique quelconque à un moment donné. Des connexions multiples peuvent exister en vue d'être utilisées par plusieurs périphériques ou pour permettre de modifier la configuration d'un même périphérique. Si vous devez modifier temporairement les paramètres réseau, au lieu de modifier la configuration d'une connexion, vous pouvez modifier la connexion qui est active pour un périphérique. Par exemple, un périphérique pour une interface réseau sans fil sur un ordinateur portable peut utiliser différentes connexions pour le réseau sans fil sur un site de travail et pour le réseau sans fil à domicile.
- Chaque connexion porte un *nom* ou un ID qui l'identifie.
- L'utilitaire **nmcli** permet de créer et de modifier les fichiers de connexion à partir de la ligne de commande.

AFFICHAGE DES INFORMATIONS RÉSEAU

La commande **nmcli dev status** affiche le statut de tous les périphériques réseau :

```
[user@host ~]$ nmcli dev status
DEVICE  TYPE      STATE       CONNECTION
eno1    ethernet  connected   eno1
ens3    ethernet  connected   static-ens3
eno2    ethernet  disconnected --
lo     loopback  unmanaged   --
```

La commande **nmcli con show** affiche une liste de toutes les connexions. Pour afficher uniquement les connexions actives, ajoutez l'option **--active**.

```
[user@host ~]$ nmcli con show
NAME      UUID                               TYPE      DEVICE
eno2      ff9f7d69-db83-4fed-9f32-939f8b5f81cd 802-3-ethernet --
static-ens3 72ca57a2-f780-40da-b146-99f71c431e2b 802-3-ethernet ens3
eno1      87b53c56-1f5d-4a29-a869-8a7bdaf56dfa 802-3-ethernet eno1
[user@host ~]$ nmcli con show --active
NAME      UUID                               TYPE      DEVICE
static-ens3 72ca57a2-f780-40da-b146-99f71c431e2b 802-3-ethernet ens3
eno1      87b53c56-1f5d-4a29-a869-8a7bdaf56dfa 802-3-ethernet eno1
```

AJOUT D'UNE CONNEXION RÉSEAU

La commande **nmcli con add** permet d'ajouter de nouvelles connexions réseau. Les exemples de commandes **nmcli con add** suivantes supposent que le nom de la connexion réseau en cours d'ajout n'est pas encore utilisé.

La commande suivante ajoute une nouvelle connexion nommée eno2 pour l'interface eno2, qui obtient alors les informations sur le réseau IPv4 en utilisant le protocole DHCP et se connecte automatiquement au démarrage. Elle obtient également les paramètres du réseau IPv6 en écoutant les annonces de routage sur le lien local. Le nom du fichier de configuration est basé sur la valeur de l'option **con-name**, eno2 et est sauvegardé dans le fichier **/etc/sysconfig/network-scripts/ifcfg-eno2**.

```
[root@host ~]# nmcli con add con-name eno2 type ethernet ifname eno2
```

Dans l'exemple suivant, on crée une connexion eno2 pour le périphérique eno2 avec une adresse statique IPv4, en utilisant l'adresse IPv4, le préfixe réseau **192.168.0.5/24** et la passerelle par défaut **192.168.0.254**, mais de sorte qu'elle se connecte automatiquement au démarrage et que son fichier de configuration soit enregistré dans le même fichier. En raison des limitations de taille d'écran, terminez la première ligne avec un signe d'échappement de shell \ et complétez la commande sur la ligne suivante.

```
[root@host ~]# nmcli con add con-name eno2 type ethernet ifname eno2 \
ipv4.address 192.168.0.5/24 ipv4.gateway 192.168.0.254
```

Dans l'exemple suivant, on crée une connexion eno2 pour le périphérique eno2 avec des adresses IPv6 et IPv4 statiques, en utilisant l'adresse IPv6 avec le préfixe réseau **2001:db8:0:1::c000:207/64** et la passerelle IPv6 par défaut **2001:db8:0:1::1**, ainsi que l'adresse IPv4 avec le préfixe réseau **192.0.2.7/24** et la passerelle IPv4 par défaut **192.0.2.1**, mais de sorte qu'elle se connecte toujours automatiquement au démarrage et qu'elle enregistre sa configuration dans **/etc/sysconfig/network-scripts/ifcfg-eno2**. En raison des limitations de taille d'écran, terminez la première ligne avec un signe d'échappement de shell \ et complétez la commande sur la ligne suivante.

```
[root@host ~]# nmcli con add con-name eno2 type ethernet ifname eno2 \
ipv6.address 2001:db8:0:1::c000:207/64 ipv6.gateway 2001:db8:0:1::1 \
ipv4.address 192.0.2.7/24 ipv4.gateway 192.0.2.1
```

Contrôle des connexions réseau

La commande **nmcli con up name** active la connexion *name* sur l'interface réseau à laquelle elle est liée. Notez que la commande prend le nom d'une *connexion*, et non le nom de l'interface

réseau. Rappelez-vous que la commande **nmcli con show** affiche le nom de toutes les connexions disponibles.

```
[root@host ~]# nmcli con up static-ens3
```

La commande **nmcli dev disconnect device** déconnecte l'interface réseau périphérique et l'arrête. Cette commande peut être abrégée en **nmcli dev dis device**:

```
[root@host ~]# nmcli dev dis ens3
```



IMPORTANT

Utilisez **nmcli dev dis device** pour désactiver une interface réseau.

La commande **nmcli con down name** n'est généralement pas le meilleur moyen de désactiver une interface réseau car elle interrompt la connexion. Toutefois, la plupart des connexions système câblées sont configurées par défaut avec l'option **autoconnect** activée. Cette option active la connexion dès que son interface réseau est disponible. Étant donné que l'interface réseau de la connexion est toujours disponible, la commande **nmcli con down name** arrête l'interface, mais NetworkManager la réactive immédiatement après, à moins que la connexion ne soit complètement déconnectée de l'interface.

MODIFICATION DES PARAMÈTRES DE CONNEXION RÉSEAU

Les connexions NetworkManager présentent deux types de paramètres. Il existe des propriétés de connexion *statiques*, configurées par l'administrateur et enregistrées dans les fichiers de configuration dans **/etc/sysconfig/network-scripts/ifcfg-***. Il peut également y avoir les données de connexion *active* que la connexion extrait d'un serveur DHCP, mais qui ne sont pas enregistrées de manière persistante.

Pour afficher la liste des paramètres actuels d'une connexion, exécutez la commande **nmcli con show name**, où *name* est le nom de la connexion. Les paramètres en minuscules sont des propriétés statiques que l'administrateur peut modifier. Les paramètres en majuscules sont les paramètres actifs utilisés temporairement pour l'instance de la connexion.

```
[root@host ~]# nmcli con show static-ens3
connection.id:                      static-ens3
connection.uuid:                     87b53c56-1f5d-4a29-a869-8a7bdaf56dfa
connection.interface-name:           --
connection.type:                      802-3-ethernet
connection.autoconnect:              yes
connection.timestamp:                1401803453
connection.read-only:                no
connection.permissions:              --
connection.zone:                     --
connection.master:                  --
connection.slave-type:               --
connection.secondaries:              --
connection.gateway-ping-timeout:    0
802-3-ethernet.port:                --
```

```

802-3-ethernet.speed:          0
802-3-ethernet.duplex:        --
802-3-ethernet.auto-negotiate: yes
802-3-ethernet.mac-address:    CA:9D:E9:2A:CE:F0
802-3-ethernet.cloned-mac-address: --
802-3-ethernet.mac-address-blacklist:
802-3-ethernet.mtu:           auto
802-3-ethernet.s390-subchannels:
802-3-ethernet.s390-nettype:   --
802-3-ethernet.s390-options:
  ipv4.method:                 manual
  ipv4.dns:                    192.168.0.254
  ipv4.dns-search:             example.com
  ipv4.addresses:              { ip = 192.168.0.2/24, gw =
                                192.168.0.254 }
  ipv4.routes:
    ipv4.ignore-auto-routes:   no
    ipv4.ignore-auto-dns:     no
    ipv4.dhcp-client-id:      --
    ipv4.dhcp-send-hostname:  yes
    ipv4.dhcp-hostname:       --
    ipv4.never-default:       no
    ipv4.may-fail:            yes
  ipv6.method:                 manual
  ipv6.dns:                    2001:4860:4860::8888
  ipv6.dns-search:             example.com
  ipv6.addresses:              { ip = 2001:db8:0:1::7/64, gw =
                                2001:db8:0:1::1 }
  ipv6.routes:
    ipv6.ignore-auto-routes:   no
    ipv6.ignore-auto-dns:     no
    ipv6.never-default:       no
    ipv6.may-fail:            yes
    ipv6.ip6-privacy:         -1 (unknown)
    ipv6.dhcp-hostname:       --
...output omitted...

```

La commande **nmcli con mod name** permet de modifier les paramètres d'une connexion. Ces modifications sont également enregistrées dans le fichier **/etc/sysconfig/network-scripts/ifcfg-name** de la connexion. Les paramètres disponibles sont documentés dans la page de manuel **nm-settings(5)**.

Pour définir l'adresse IPv4 sur 192.0.2.2/24 et la passerelle par défaut sur 192.0.2.254 pour la connexion static-ens3 :

```
[root@host ~]# nmcli con mod static-ens3 ipv4.address 192.0.2.2/24 \
               ipv4.gateway 192.0.2.254
```

Pour définir l'adresse IPv6 sur 2001:db8:0:1::a00:1/64 et la passerelle par défaut sur 2001:db8:0:1::1 pour la connexion static-ens3 :

```
[root@host ~]# nmcli con mod static-ens3 ipv6.address 2001:db8:0:1::a00:1/64 \
               ipv6.gateway 2001:db8:0:1::1
```

**IMPORTANT**

Si une connexion qui obtient ses informations IPv4 d'un serveur DHCPv4 est cours de modification pour les extraire des fichiers de configuration statique uniquement, le paramètre **ipv4.method** doit également passer de `auto` à `manual`.

De même, si une connexion qui obtient ses informations IPv6 via SLAAC ou un serveur DHCPv6 est en cours de modification en vue de les extraire à partir de fichiers de configuration statique uniquement, le paramètre **ipv6.method** doit également passer de `auto` ou `dhcp` à `manual`.

Sinon, la connexion peut être suspendue ou ne pas s'exécuter correctement lorsqu'elle est activée, ou elle risque d'extraire une adresse IPv4 de DHCP ou une adresse IPv6 de DHCPv6 ou SLAAC en plus de l'adresse statique.

Un certain nombre de paramètres peuvent accepter plusieurs valeurs. Il est possible d'ajouter ou de supprimer une valeur spécifique à une liste ou à un paramètre en ajoutant un signe + ou - devant le nom du paramètre.

SUPPRESSION D'UNE CONNEXION RÉSEAU

La commande **nmcli con del name** supprime la connexion appelée *nom* du système, la déconnecte du périphérique et supprime le fichier `/etc/sysconfig/network-scripts/ifcfg-name`.

```
[root@host ~]# nmcli con del static-ens3
```

UTILISATEUR POUVANT MODIFIER LES PARAMÈTRES RÉSEAU

L'utilisateur `root` peut effectuer les modifications de configuration réseau nécessaires avec **nmcli**.

Toutefois, les utilisateurs normaux connectés à la console locale peuvent également apporter de nombreuses modifications de configuration réseau sur le système. Ils doivent se connecter, au niveau du clavier du système, à une console virtuelle basée sur du texte ou à l'environnement de bureau graphique afin d'obtenir ce contrôle. La logique ici est que, si une personne est physiquement présente sur la console de l'ordinateur, la console sera probablement utilisée comme station de travail ou ordinateur portable et elle pourra avoir besoin de configurer, d'activer et de désactiver à volonté les interfaces réseau sans fil ou filaires. En revanche, si le système est un serveur du centre de données, les seuls utilisateurs qui se connectent localement à la machine doivent être des administrateurs.

Les utilisateurs normaux qui se connectent à l'aide de `ssh` ne peuvent pas modifier les autorisations du réseau s'ils ne deviennent pas des utilisateurs `root`.

Vous pouvez utiliser la commande **nmcli gen permissions** pour connaître vos autorisations actuelles.

RÉCAPITULATIF DES COMMANDES

Le tableau ci-dessous répertorie les principales commandes **nmcli** décrites dans cette section.

COMMANDÉ	OBJET
nmcli dev status	Affiche le statut NetworkManager de toutes les interfaces réseau.
nmcli con show	Liste toutes les connexions.
nmcli con show <i>name</i>	Liste les paramètres actuels de la connexion <i>nom</i> .
nmcli con add con-name <i>name</i>	Ajoute une nouvelle connexion appelée <i>nom</i> .
nmcli con mod <i>name</i>	Modifie la connexion <i>nom</i> .
nmcli con reload	Recharge les fichiers de configuration (utile lorsque ceux-ci ont été modifiés manuellement).
nmcli con up <i>name</i>	Active la connexion <i>nom</i> .
nmcli dev dis <i>dev</i>	Désactive et déconnecte la connexion actuelle de l'interface réseau <i>dev</i> .
nmcli con del <i>name</i>	Supprime la connexion <i>nom</i> et son fichier de configuration.



RÉFÉRENCES

Pages de manuel NetworkManager(8), nmcli(1), nmcli-examples(5), nm-settings(5), hostnamectl(1), resolv.conf(5), hostname(5), ip(8) et ip-address(8)

► EXERCICE GUIDÉ

CONFIGURATION DE LA MISE EN RÉSEAU À PARTIR DE LA LIGNE DE COMMANDE

Au cours de cet exercice, vous allez configurer des paramètres réseau avec la commande **nmcli**.

RÉSULTATS

Vous serez en mesure de convertir un système d'une configuration DHCP à une configuration statique.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur **student** à **workstation** avec le mot de passe **student**.

À partir de **workstation**, exécutez la commande **lab net-configure start**. La commande exécute un script de démarrage qui détermine si l'hôte, **servera**, est accessible sur le réseau.

```
[student@workstation ~]$ lab net-configure start
```



NOTE

Si la commande **sudo** demande un mot de passe **student**, saisissez **student** comme mot de passe.

- 1. Utilisez la commande **ssh** pour vous connecter à **servera** en tant qu'utilisateur **student**. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis pour se connecter à **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Localisez les noms d'interfaces réseau.

**IMPORTANT**

Les noms des interfaces réseau sont déterminés par leur type de bus et l'ordre de détection des périphériques lors du démarrage. Les noms de vos interfaces réseau varient en fonction de la plateforme du cours et du matériel utilisés.

Sur votre système, localisez le nom de l'interface (tel que `ens06` ou `en1p2`) associé à l'adresse Ethernet `52:54:00:00:fa:0a`. Utilisez ce nom d'interface pour remplacer le marqueur `enX` utilisé tout au long de l'exercice.

Localisez le nom de l'interface associé à l'adresse Ethernet `52:54:00:00:fa:0a`. Enregistrez ou mémorisez ce nom et utilisez-le pour remplacer le marqueur `enX` dans les commandes suivantes.

```
[student@servera ~]$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
```

► 3. Affichez les paramètres du réseau avec la commande **nmcli**.

3.1. Affichez toutes les connexions.

```
[student@servera ~]$ nmcli con show
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
```

3.2. Affichez uniquement la connexion active.

Le nom de votre interface réseau doit apparaître sous **DEVICE** et le nom de la connexion active de ce périphérique est indiqué sur la même ligne, sous **NAME**. Cet exercice suppose que la connexion active est **Connexion filaire 1**.

Si le nom de la connexion active est différent, utilisez-le au lieu de **Connexion filaire 1** pour le reste de cet exercice.

```
[student@servera ~]$ nmcli con show --active
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
```

3.3. Affichez tous les paramètres de configuration de la connexion active.

```
[student@servera ~]$ nmcli con show "Wired connection 1"
connection.id:            Wired connection 1
connection.uuid:          03da038a-3257-4722-a478-53055cc90128
connection.stable-id:     --
connection.type:          802-3-ethernet
connection.interface-name: --
connection.autoconnect:   yes
```

CHAPITRE 12 | Gestion de réseaux

```
...output omitted...
ipv4.method:           manual
ipv4.dns:              172.25.250.254
ipv4.dns-search:       lab.example.com,example.com
ipv4.dns-options:      ""
ipv4.dns-priority:     0
ipv4.addresses:        172.25.250.10/24
ipv4.gateway:          172.25.250.254
...output omitted...
GENERAL.NAME:          Wired connection 1
GENERAL.UUID:          03da038a-3257-4722-a478-53055cc90128
GENERAL.DEVICES:       enX
GENERAL.STATE:         activated
GENERAL.DEFAULT:       yes
GENERAL.DEFAULT6:      no
GENERAL.SPEC-OBJECT:   --
GENERAL.VPN:           no
GENERAL.DBUS-PATH:     /org/freedesktop/NetworkManager/ActiveConnection/1
GENERAL.CON-PATH:      /org/freedesktop/NetworkManager/Settings/1
GENERAL.ZONE:          --
GENERAL.MASTER-PATH:   --
IP4.ADDRESS[1]:        172.25.250.10/24
IP4.GATEWAY:           172.25.250.254
IP4.ROUTE[1]:          dst = 172.25.250.0/24, nh = 0.0.0.0, mt = 100
IP4.ROUTE[2]:          dst = 0.0.0.0/0, nh = 172.25.250.254, mt = 100
IP4.DNS[1]:             172.25.250.254
IP6.ADDRESS[1]:        fe80::3059:5462:198:58b2/64
IP6.GATEWAY:           --
IP6.ROUTE[1]:          dst = fe80::/64, nh = ::, mt = 100
IP6.ROUTE[2]:          dst = ff00::/8, nh = ::, mt = 256, table=255
```

Appuyez sur **q** pour quitter la commande.

3.4. Affichez le statut des périphériques.

```
[student@servera ~]$ nmcli dev status
DEVICE  TYPE      STATE      CONNECTION
enX     ethernet  connected  Wired connection 1
lo      loopback  unmanaged  --
```

3.5. Affichez les paramètres du périphérique enX.

```
[student@servera ~]$ nmcli dev show enX
GENERAL.DEVICE:           enX
GENERAL.TYPE:             ethernet
GENERAL.HWADDR:           52:54:00:00:FA:0A
GENERAL.MTU:              1500
GENERAL.STATE:            100 (connected)
GENERAL.CONNECTION:       Wired connection 1
GENERAL.CON-PATH:         /org/freedesktop/NetworkManager/ActiveConnection/1
WIRED-PROPERTIES.CARRIER: on
IP4.ADDRESS[1]:           172.25.250.10/24
IP4.GATEWAY:              172.25.250.254
IP4.ROUTE[1]:              dst = 172.25.250.0/24, nh = 0.0.0.0, mt = 100
```

```
IP4.ROUTE[2]:           dst = 0.0.0.0/0, nh = 172.25.250.254, mt = 100
IP4.DNS[1]:              172.25.250.254
IP6.ADDRESS[1]:          fe80::3059:5462:198:58b2/64
IP6.GATEWAY:             --
IP6.ROUTE[1]:            dst = fe80::/64, nh = ::, mt = 100
IP6.ROUTE[2]:            dst = ff00::/8, nh = ::, mt = 256, table=255
```

- ▶ 4. Créez une connexion statique avec la même adresse IPv4, le même préfixe réseau et la même passerelle par défaut. Nommez la nouvelle connexion *static-addr*.



MISE EN GARDE

Comme l'accès à votre machine est fourni par la connexion réseau principale, le réglage de valeurs erronées lors de la configuration du réseau pourrait rendre votre ordinateur inaccessible. Si cela arrive, utilisez le bouton Reset situé au-dessus de ce qui était l'affichage graphique de votre ordinateur, puis recommencez.

```
[student@servera ~]$ sudo nmcli con add con-name "static-addr" ifname enX \
type ethernet ipv4.method manual \
ipv4.address 172.25.250.10/24 ipv4.gateway 172.25.250.254
Connection 'static-addr' (15aa3901-555d-40cb-94c6-cea6f9151634) successfully
added.
```

- ▶ 5. Modifiez la nouvelle connexion pour ajouter le paramètre DNS.

```
[student@servera ~]$ sudo nmcli con mod "static-addr" ipv4.dns 172.25.250.254
```

- ▶ 6. Affichez et activez la nouvelle connexion.

6.1. Affichez toutes les connexions.

```
[student@servera ~]$ nmcli con show
NAME                UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
static-addr         15aa3901-555d-40cb-94c6-cea6f9151634  ethernet  --
```

6.2. Affichez la connexion active.

```
[student@servera ~]$ nmcli con show --active
NAME                UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
```

6.3. Activez la nouvelle connexion *static-addr*.

```
[student@servera ~]$ sudo nmcli con up "static-addr"
Connection successfully activated (D-Bus active path: /org/freedesktop/
NetworkManager/ActiveConnection/2)
```

6.4. Vérifiez la nouvelle connexion active.

```
[student@servera ~]$ nmcli con show --active
NAME           UUID
static-addr   15aa3901-555d-40cb-94c6-cea6f9151634
                           TYPE      DEVICE
                                         ethernet enx
```

- 7. Configurez la connexion d'origine de sorte qu'elle ne soit pas établie au démarrage, et vérifiez que la connexion statique est utilisée au redémarrage du système.

7.1. Désactivez le démarrage automatique de la connexion d'origine à l'amorçage.

```
[student@servera ~]$ sudo nmcli con mod "Wired connection 1" \
connection.autoconnect no
```

7.2. Redémarrez le système.

```
[student@servera ~]$ sudo systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

7.3. Affichez la connexion active.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$ nmcli con show --active
NAME           UUID
static-addr   15aa3901-555d-40cb-94c6-cea6f9151634
                           TYPE      DEVICE
                                         ethernet enx
```

- 8. Testez la connectivité avec les nouvelles adresses réseau.

8.1. Vérifiez l'adresse IP.

```
[student@servera ~]$ ip addr show enx
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
    inet 172.25.250.10/24 brd 172.25.250.255 scope global noprefixroute enX
        valid_lft forever preferred_lft forever
    inet6 fe80::6556:cdd9:ce15:1484/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

8.2. Vérifiez la passerelle par défaut.

```
[student@servera ~]$ ip route
default via 172.25.250.254 dev enX proto static metric 100
172.25.250.0/24 dev enX proto kernel scope link src 172.25.250.10 metric 100
```

8.3. Lancez un ping à l'adresse du DNS.

```
[student@servera ~]$ ping -c3 172.25.250.254
PING 172.25.250.254 (172.25.250.254) 56(84) bytes of data.
64 bytes from 172.25.250.254: icmp_seq=1 ttl=64 time=0.225 ms
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.314 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.472 ms

--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 46ms
rtt min/avg/max/mdev = 0.225/0.337/0.472/0.102 ms
```

8.4. Quittez servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Fin

Sur workstation, exéutez le script **lab net-configure finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab net-configure finish
```

L'exercice guidé est maintenant terminé.

MODIFICATION DES FICHIERS DE CONFIGURATION RÉSEAU

OBJECTIFS

Au terme de cette section, vous devez être capable de modifier la configuration du réseau en modifiant les fichiers de configuration.

DESCRIPTION DES FICHIERS DE CONFIGURATION DE CONNEXION

Par défaut, les modifications effectuées avec la commande `nmcli con mod name` sont automatiquement enregistrées dans `/etc/sysconfig/network-scripts/ifcfg-name`. Ce fichier peut également être modifié manuellement dans un éditeur de texte. Après cela, exécutez la commande `nmcli con reload` afin que NetworkManager lise les modifications de configuration.

Pour des raisons de compatibilité ascendante, les directives enregistrées dans ce fichier portent des noms différents et présentent une syntaxe différente par rapport aux noms utilisés dans `nm-settings(5)`. Le tableau suivant associe les noms de certains paramètres clés aux directives `ifcfg-*`.

Comparaison de nm-settings et des directives ifcfg-*

NMCLI CON MOD	IFCFG-* FILE	EFFET
<code>ipv4.method manual</code>	<code>BOOTPROTO=none</code>	Adresses IPv4 configurées de manière statique.
<code>ipv4.method auto</code>	<code>BOOTPROTO=dhcp</code>	Recherche les paramètres de configuration auprès d'un serveur DHCPv4. Si des adresses statiques sont également définies, elles ne seront pas activées avant d'obtenir les informations par DHCPv4.
<code>ipv4.addresses "192.0.2.1/24 192.0.2.254"</code>	<code>IPADDR0=192.0.2.1 PREFIX0=24 GATEWAY0=192.0.2.254</code>	Définit l'adresse IPv6 statique, le préfixe réseau et la passerelle par défaut. Si plusieurs adresses sont définies pour la connexion, les directives <code>ifcfg-*</code> se terminent par 1, 2, 3, et ainsi de suite au lieu de se terminer par 0.
<code>ipv4.dns 8.8.8.8</code>	<code>DNS0=8.8.8.8</code>	Modifiez <code>/etc/resolv.conf</code> pour utiliser ce serveur de noms .

NMCLI CON MOD	IFCFG-* FILE	EFFET
ipv4.dns-search example.com	DOMAIN=example.com	Modifiez /etc/resolv.conf pour utiliser ce domaine dans la directive search .
ipv4.ignore-auto-dns true	PEERDNS=no	Ignore les informations relatives au serveur DNS transmises par le serveur DHCP.
ipv6.method manual	IPV6_AUTOCONF=no	Adresses IPv6 configurées de manière statique.
ipv6.method auto	IPV6_AUTOCONF=yes	Configure les paramètres réseau en utilisant SLAAC à partir des annonces de routage.
ipv6.method dhcp	IPV6_AUTOCONF=no DHCPV6C=yes	Configure les paramètres réseau en utilisant DHCPv6, mais pas SLAAC.
ipv6.addresses "2001:db8::a/64 2001:db8::1"	IPV6ADDR=2001:db8::a/64 IPV6_DEFAULTGW=2001:db8	Définit l'adresse IPv6 statique, le préfixe réseau et la passerelle par défaut. Si plusieurs adresses sont définies pour la connexion, IPV6_SECONDARIES utilise une liste entre guillemets doubles pour les définitions adresse/préfixe délimitées par un espace.
ipv6.dns ...	DNS0= ...	Modifiez /etc/resolv.conf pour utiliser ce serveur de noms . Exactement comme pour IPv4.
ipv6.dns-search example.com	DOMAIN=example.com	Modifiez /etc/resolv.conf pour utiliser ce domaine dans la directive search . Exactement comme pour IPv4.
ipv6.ignore-auto-dns true	IPV6_PEERDNS=no	Ignore les informations relatives au serveur DNS transmises par le serveur DHCP.
connection.autoconnect yes	ONBOOT=yes	Active automatiquement cette connexion au démarrage.

NMCLI CON MOD	IFCFG-* FILE	EFFET
connection.id ens3	NAME=ens3	Nom de cette connexion.
connection.interface-name ens3	DEVICE=ens3	La connexion est liée à l'interface réseau portant ce nom.
802-3-ethernet.mac-address . . .	HWADDR= . . .	La connexion est liée à l'interface réseau avec cette adresse MAC.

MODIFICATION DE LA CONFIGURATION RÉSEAU

Il est également possible de configurer le réseau en modifiant les fichiers de configuration de la connexion. Les fichiers de configuration de la connexion contrôlent les interfaces logicielles des différents périphériques réseau. Ces fichiers sont en général nommés **/etc/sysconfig/network-scripts/ifcfg-name**, où *name* est le nom du périphérique ou de la connexion que le fichier de configuration contrôle. Le tableau suivant répertorie les variables standard présentes dans le fichier utilisé pour une configuration IPv4 statique ou dynamique.

Options de configuration IPv4 du fichier ifcfg

STATIQUE	DYNAMIQUE	L'UNE OU L'AUTRE
BOOTPROTO=none	BOOTPROTO=dhcp	DEVICE=ens3
IPADDR0=172.25.250.10		NAME="static-ens3"
PREFIX0=24		ONBOOT=yes
GATEWAY0=172.25.250.254		UUID=f3e8(. . .)ad3e
DEFROUTE=yes		USERCTL=yes
DNS1=172.25.254.254		

Dans les paramètres statiques, les variables pour l'adresse IP, le préfixe et la passerelle ont un nombre en suffixe. Cela permet l'attribution de plusieurs ensembles de valeurs à l'interface. La variable DNS comprend également un nombre qui est utilisé pour indiquer l'ordre de recherche lorsque plusieurs serveurs sont spécifiés.

Après avoir modifié les fichiers de configuration, exécutez la commande **nmcli con reload** pour que NetworkManager lise les modifications de la configuration. L'interface doit encore être redémarrée pour que les modifications soient prises en compte.

```
[root@host ~]# nmcli con reload
[root@host ~]# nmcli con down "static-ens3"
[root@host ~]# nmcli con up "static-ens3"
```



RÉFÉRENCES

Page de manuel `nmcli(1)`

Pour plus d'informations, reportez-vous à la section *Configuring and Managing Networking* dans *Red Hat Enterprise Linux 8.0* à l'adresse
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_and_managing_networking/

► EXERCICE GUIDÉ

MODIFICATION DES FICHIERS DE CONFIGURATION RÉSEAU

Au cours de cet exercice, vous allez modifier manuellement les fichiers de configuration réseau et vous assurer que les nouveaux paramètres sont appliqués.

RÉSULTATS

Vous serez en mesure d'ajouter une adresse réseau supplémentaire à chaque système.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur **student** à **workstation** avec le mot de passe **student**.

À partir de **workstation**, exécutez la commande **lab net-edit start**. La commande exécute un script de démarrage qui détermine si les hôtes, **servera** et **serverb**, sont accessibles sur le réseau.

```
[student@workstation ~]$ lab net-edit start
```

- ▶ 1. Utilisez la commande **ssh** pour vous connecter à **servera** en tant qu'utilisateur **student**. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis pour se connecter à **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. Localisez les noms d'interfaces réseau.



IMPORTANT

Les noms des interfaces réseau sont déterminés par leur type de bus et l'ordre de détection des périphériques lors du démarrage. Les noms de vos interfaces réseau varient en fonction de la plateforme du cours et du matériel utilisés.

Sur votre système, localisez le nom de l'interface (tel que **ens06** ou **en1p2**) associé à l'adresse Ethernet **52:54:00:00:fa:0a**. Utilisez ce nom d'interface pour remplacer le marqueur **enX** utilisé tout au long de l'exercice.

Localisez le nom de l'interface associé à l'adresse Ethernet **52:54:00:00:fa:0a**. Enregistrez ou mémorisez ce nom et utilisez-le pour remplacer le marqueur **enX** dans les commandes suivantes. La connexion active est également nommée Connexion filaire 1 (et est donc gérée par le fichier **/etc/sysconfig/network-scripts/ifcfg-Wired_connection_1**).

CHAPITRE 12 | Gestion de réseaux

Si vous avez fait les précédents exercices de ce chapitre et que cela s'appliquait à votre système, il devrait en être de même pour cet exercice.

```
[student@servera ~]$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
[student@servera ~]$ nmcli con show --active
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
[student@servera ~]$ ls /etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
/etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
```

- 3. Modifiez le fichier **/etc/sysconfig/network-scripts/ifcfg-Wired_connection_1** sur servera pour ajouter l'adresse supplémentaire **10.0.1.1/24**.

3.1. Ajoutez une entrée au fichier pour spécifier l'adresse IPv4.

```
[student@servera ~]$ echo "IPADDR1=10.0.1.1" | \
sudo tee -a /etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
[sudo] password for student:
IPADDR1=10.0.1.1
```

3.2. Ajoutez une entrée au fichier pour spécifier le préfixe de réseau.

```
[student@servera ~]$ echo "PREFIX1=24" | \
sudo tee -a /etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
PREFIX1=24
```

- 4. Activez la nouvelle adresse.

4.1. Rechargez les modifications de la configuration.

```
[student@servera ~]$ sudo nmcli con reload
```

4.2. Relancez la connexion avec les nouveaux paramètres.

```
[student@servera ~]$ sudo nmcli con up "Wired connection 1"
Connection successfully activated (D-Bus active path: /org/freedesktop/
NetworkManager/ActiveConnection/3)
```

4.3. Vérifiez la nouvelle adresse IP.

```
[student@servera ~]$ ip addr show enX
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
```

CHAPITRE 12 | Gestion de réseaux

```
inet 172.25.250.10/24 brd 172.25.250.255 scope global noprefixroute enX
    valid_lft forever preferred_lft forever
inet 10.0.1.1/24 brd 10.0.1.255 scope global noprefixroute enX
    valid_lft forever preferred_lft forever
inet6 fe80::4bf3:e1d9:3076:f8d7/64 scope link noprefixroute
    valid_lft forever preferred_lft forever
```

4.4. Quittez **servera** pour revenir à **workstation** en tant qu'utilisateur **student**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

- 5. En tant qu'utilisateur **student** sur **serverb**, modifiez le fichier **/etc/sysconfig/network-scripts/ifcfg-Wired_connection_1** pour ajouter l'adresse supplémentaire **10.0.1.2/24**, puis chargez la nouvelle configuration.

5.1. À partir de **workstation**, utilisez la commande **ssh** pour vous connecter à **serverb** en tant qu'utilisateur **student**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

5.2. Modifiez le fichier **ifcfg-Wired_connection_1** afin d'ajouter la deuxième adresse IPv4 et le préfixe réseau.

```
[student@serverb ~]$ echo "IPADDR2=10.0.1.2" | \
sudo tee -a /etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
[sudo] password for student: student
IPADDR2=10.0.1.2
[student@serverb ~]$ echo "PREFIX2=24" | \
sudo tee -a /etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
PREFIX2=24
```

5.3. Rechargez les modifications de la configuration.

```
[student@serverb ~]$ sudo nmcli con reload
```

5.4. Lancez la connexion à l'aide des nouveaux paramètres.

```
[student@serverb ~]$ sudo nmcli con up "Wired connection 1"
Connection successfully activated (D-Bus active path: /org/freedesktop/
NetworkManager/ActiveConnection/4)
```

5.5. Vérifiez la nouvelle adresse IP.

```
[student@serverb ~]$ ip addr show enx
2: enx: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 52:54:00:00:fa:0b brd ff:ff:ff:ff:ff:ff
    inet 172.25.250.11/24 brd 172.25.250.255 scope global noprefixroute enX
        valid_lft forever preferred_lft forever
    inet 10.0.1.2/24 brd 10.0.1.255 scope global noprefixroute enX
        valid_lft forever preferred_lft forever
    inet6 fe80::74c:3476:4113:463f/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

► 6. Testez la connectivité avec les nouvelles adresses réseau.

- 6.1. À partir de serverb, envoyez un ping vers la nouvelle adresse de servera.

```
[student@serverb ~]$ ping -c3 10.0.1.1
PING 10.0.1.1 (10.0.1.1) 56(84) bytes of data.
64 bytes from 10.0.1.1: icmp_seq=1 ttl=64 time=0.342 ms
64 bytes from 10.0.1.1: icmp_seq=2 ttl=64 time=0.188 ms
64 bytes from 10.0.1.1: icmp_seq=3 ttl=64 time=0.317 ms

--- 10.0.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 35ms
rtt min/avg/max/mdev = 0.188/0.282/0.342/0.068 ms
```

- 6.2. Quittez serverb pour revenir à workstation.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

- 6.3. À partir de workstation, utilisez la commande **ssh** pour accéder à servera en tant qu'utilisateur student afin d'envoyer un ping vers la nouvelle adresse de serverb .

```
[student@workstation ~]$ ssh student@servera ping -c3 10.0.1.2
PING 10.0.1.2 (10.0.1.2) 56(84) bytes of data.
64 bytes from 10.0.1.2: icmp_seq=1 ttl=64 time=0.269 ms
64 bytes from 10.0.1.2: icmp_seq=2 ttl=64 time=0.338 ms
64 bytes from 10.0.1.2: icmp_seq=3 ttl=64 time=0.361 ms

--- 10.0.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 48ms
rtt min/avg/max/mdev = 0.269/0.322/0.361/0.044 ms
```

Fin

Sur workstation, exécutez le script **lab net-edit finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab net-edit finish
```

L'exercice guidé est maintenant terminé.

CONFIGURATION DE NOMS D'HÔTE ET RÉSOLUTION DE NOMS

OBJECTIFS

Au terme de cette section, vous serez capable de configurer le nom d'hôte statique d'un serveur et sa résolution de noms, et de tester les résultats.

MODIFICATION DU NOM D'HÔTE DU SYSTÈME

La commande **hostname** affiche ou modifie temporairement le nom d'hôte entièrement qualifié du système.

```
[root@host ~]# hostname
host@example.com
```

Un nom d'hôte statique peut être spécifié dans le fichier **/etc/hostname**. La commande **hostnamectl** sert à modifier ce fichier et peut être utilisée pour afficher l'état du nom d'hôte entièrement qualifié du système. Si ce fichier n'existe pas, le nom d'hôte est défini par une requête DNS inverse une fois qu'une adresse IP a été attribuée à l'interface.

```
[root@host ~]# hostnamectl set-hostname host@example.com
[root@host ~]# hostnamectl status
  Static hostname: host.example.com
    Icon name: computer-vm
      Chassis: vm
    Machine ID: 73ab164e278e48be9bf80e80714a8cd5
      Boot ID: 6b1cbc4177164ef58c0e9ed4adb2904f
  Virtualization: kvm
Operating System: Red Hat Enterprise Linux 8.0 beta (ootpa)
  CPE OS Name: cpe:/o:redhat:enterprise_linux:8.0:beta
        Kernel: Linux 4.18.0-60.el8.x86_64
      Architecture: x86-64
[root@host ~]# cat /etc/hostname
host@example.com
```



IMPORTANT

Dans Red Hat Enterprise Linux 7 et versions ultérieures, le nom d'hôte statique est stocké dans **/etc/hostname**. Red Hat Enterprise Linux 6 and versions antérieures stockent le nom d'hôte sous forme de variable dans le fichier **/etc/sysconfig/network**.

CONFIGURATION DE LA RÉSOLUTION DE NOMS

Le *résolveur stub* sert à convertir les noms d'hôte en adresses IP et vice versa. Il détermine où chercher en fonction de la configuration du fichier **/etc/nsswitch.conf**. Par défaut, le contenu du fichier **/etc/hosts** est vérifié en premier.

```
[root@host ~]# cat /etc/hosts
127.0.0.1      localhost localhost.localdomain localhost4 localhost4.localdomain4
::1            localhost localhost.localdomain localhost6 localhost6.localdomain6

172.25.254.254 classroom.example.com
172.25.254.254 content.example.com
```

La commande **getent hosts hostname** peut être utilisée pour tester la résolution du nom d'hôte au moyen du fichier **/etc/hosts**.

Si aucune entrée n'est trouvée dans le fichier **/etc/hosts**, par défaut, le résolveur stub essaye de trouver le nom d'hôte en utilisant le serveur de noms DNS. Le fichier **/etc/resolv.conf** détermine de quelle manière le serveur est interrogé :

- **search** : liste de noms de domaine à tester avec un nom d'hôte court. Cette liste et **domain** ne doivent pas être définis dans le même fichier ; si tel est le cas, la dernière instance prévaut. Reportez-vous à **resolv.conf(5)** pour plus de détails.
- **nameserver** : adresse IP d'un serveur de noms à interroger. On peut indiquer jusqu'à trois directives de nom de serveur pour fournir des alternatives si l'un d'entre eux est hors service.

```
[root@host ~]# cat /etc/resolv.conf
# Generated by NetworkManager
domain example.com
search example.com
nameserver 172.25.254.254
```

NetworkManager met à jour le fichier **/etc/resolv.conf** à l'aide des paramètres DNS des fichiers de configuration de connexion. Utilisez la commande **nmcli** pour modifier les connexions.

```
[root@host ~]# nmcli con mod ID ipv4.dns IP
[root@host ~]# nmcli con down ID
[root@host ~]# nmcli con up ID
[root@host ~]# cat /etc/sysconfig/network-scripts/ifcfg-ID
...output omitted...
DNS1=8.8.8.8
...output omitted...
```

Le comportement par défaut de la commande **nmcli con mod ID ipv4.dns IP** est de remplacer tout paramètre DNS précédent par la nouvelle liste d'adresses IP fournie. Le symbole + ou - devant l'argument **ipv4.dns** ajoute ou supprime une entrée individuelle.

```
[root@host ~]# nmcli con mod ID +ipv4.dns IP
```

Pour ajouter le serveur DNS avec l'adresse IP IPv6 2001:4860:4860::8888 à la liste des serveurs de noms à utiliser avec la connexion static-ens3 :

```
[root@host ~]# nmcli con mod static-ens3 +ipv6.dns 2001:4860:4860::8888
```

**NOTE**

Les paramètres DNS IPv4 et IPv6 statiques se terminent tous selon les directives `nameserver` dans **/etc/resolv.conf**. Vous devez vous assurer qu'au minimum un serveur de noms accessible via IPv4 est listé (en supposant l'existence d'un système à double pile). Il est préférable d'avoir au moins un serveur de noms utilisant IPv4 et un second utilisant IPv6 au cas où vous auriez des problèmes avec votre réseau IPv4 ou IPv6.

Test de la résolution de noms DNS

La commande **host HOSTNAME** peut être utilisée pour tester la connectivité du serveur DNS.

```
[root@host ~]# host classroom.example.com
classroom.example.com has address 172.25.254.254
[root@host ~]# host 172.25.254.254
254.25.25.172.in-addr.arpa domain name pointer classroom.example.com.
```

**IMPORTANT**

DHCP réécrit automatiquement le fichier **/etc/resolv.conf** lorsque les interfaces sont lancées, sauf si vous spécifiez `PEERDNS=no` dans les fichiers appropriés de configuration des interfaces. Définissez ceci en utilisant la commande **nmcli**.

```
[root@host ~]# nmcli con mod "static-ens3" ipv4.ignore-auto-dns yes
```

**RÉFÉRENCES**

Pages de manuel `nmcli(1)`, `hostnamectl(1)`, `hosts(5)`, `getent(1)`, `host(1)` et `resolv.conf(5)`

Pour plus d'informations, reportez-vous à la section *Configuring and Managing Networking* dans *Red Hat Enterprise Linux 8.0* à l'adresse
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_and_managing_networking/

► EXERCICE GUIDÉ

CONFIGURATION DE NOMS D'HÔTE ET RÉSOLUTION DE NOMS

Au cours de cet exercice, vous allez configurer manuellement le nom d'hôte statique du système, le fichier **/etc/hosts** et le résolveur de noms DNS.

RÉSULTATS

Vous devez pouvoir définir un nom d'hôte personnalisé et configurer les paramètres de résolution de noms.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab net-hostnames start**. La commande exécute un script de démarrage qui détermine si l'hôte, servera, est accessible sur le réseau.

```
[student@workstation ~]$ lab net-hostnames start
```

- ▶ 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis pour se connecter à servera.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. Affichez les paramètres actuels du nom d'hôte.
 - 2.1. Affichez le nom d'hôte actuel.

```
[student@servera ~]$ hostname
servera.lab.example.com
```

- 2.2. Affichez l'état du nom d'hôte.

```
[student@servera ~]$ hostnamectl status
Static hostname: n/a
Transient hostname: servera.lab.example.com
Icon name: computer-vm
Chassis: vm
Machine ID: 73ab164e278e48be9bf80e80714a8cd5
Boot ID: 76b13a300c944ab49445af778cb8f749
```

CHAPITRE 12 | Gestion de réseaux

```
Virtualization: kvm
Operating System: Red Hat Enterprise Linux 8.0 (ootpa)
CPE OS Name: cpe:/o:redhat:enterprise_linux:8.0:GA
Kernel: Linux 4.18.0-80.el8.x86_64
Architecture: x86-64
```

▶ **3.** Définissez un nom d'hôte statique qui correspond au nom d'hôte temporaire actuel.

3.1. Modifiez le nom d'hôte et le fichier de configuration du nom d'hôte.

```
[student@servera ~]$ sudo hostnamectl set-hostname servera.lab.example.com
[sudo] password for student: student
[student@servera ~]$
```

3.2. Affichez le fichier de configuration qui fournit le nom d'hôte au démarrage du réseau.

```
[student@servera ~]$ cat /etc/hostname
servera.lab.example.com
```

3.3. Affichez l'état du nom d'hôte.

```
[student@servera ~]$ hostnamectl status
Static hostname: servera.lab.example.com
Icon name: computer-vm
Chassis: vm
Machine ID: 73ab164e278e48be9bf80e80714a8cd5
Boot ID: 76b13a300c944ab49445af778cb8f749
Virtualization: kvm
Operating System: Red Hat Enterprise Linux 8.0 (ootpa)
CPE OS Name: cpe:/o:redhat:enterprise_linux:8.0:GA
Kernel: Linux 4.18.0-80.el8.x86_64
Architecture: x86-64
```

▶ **4.** Modifiez temporairement le nom d'hôte.

4.1. Modifiez le nom d'hôte.

```
[student@servera ~]$ sudo hostname testname
```

4.2. Affichez le nom d'hôte actuel.

```
[student@servera ~]$ hostname
testname
```

4.3. Affichez le fichier de configuration qui fournit le nom d'hôte au démarrage du réseau.

```
[student@servera ~]$ cat /etc/hostname
servera.lab.example.com
```

4.4. Redémarrez le système.

CHAPITRE 12 | Gestion de réseaux

```
[student@servera ~]$ sudo systemctl reboot  
Connection to servera closed by remote host.  
Connection to servera closed.  
[student@workstation ~]$
```

4.5. À partir de workstation, connectez-vous à servera en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

4.6. Affichez le nom d'hôte actuel.

```
[student@servera ~]$ hostname  
servera.lab.example.com
```

► 5. Ajoutez un surnom local au serveur de la salle de classe.

5.1. Recherchez l'adresse IP de classroom.example.com.

```
[student@servera ~]$ host classroom.example.com  
classroom.example.com has address 172.25.254.254
```

5.2. Modifiez **/etc/hosts** de sorte que le nom supplémentaire de **class** puisse être utilisé pour accéder à l'adresse IP 172.25.254.254.

```
[student@servera ~]$ sudo vim /etc/hosts  
[student@servera ~]$ cat /etc/hosts  
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4  
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6  
  
172.25.254.254 classroom.example.com classroom class  
172.25.254.254 content.example.com content  
...content omitted...
```

5.3. Recherchez l'adresse IP de class.

```
[student@servera ~]$ host class  
Host class not found: 2(SERVFAIL)  
[student@servera ~]$ getent hosts class  
172.25.254.254 classroom.example.com class
```

5.4. Ping class.

```
[student@servera ~]$ ping -c3 class  
PING classroom.example.com (172.25.254.254) 56(84) bytes of data.  
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=1 ttl=64 time=0.397  
ms  
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=2 ttl=64 time=0.447  
ms
```

```
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=3 ttl=64 time=0.470
ms

--- classroom.example.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.397/0.438/0.470/0.030 ms
```

5.5. Quittez servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Fin

Sur workstation, exécutez le script **lab net-hostnames finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab net-hostnames finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

GESTION DE RÉSEAUX

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez configurer les paramètres réseau sur un serveur Red Hat Enterprise Linux.

RÉSULTATS

Vous serez en mesure de configurer deux adresses IPv4 statiques pour l'interface réseau principale.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur `student` à `workstation` avec le mot de passe `student`.

À partir de `workstation`, exécutez la commande `lab net-review start`. La commande exécute un script de démarrage qui détermine si l'hôte, `serverb`, est accessible sur le réseau.

```
[student@workstation ~]$ lab net-review start
```

1. Utilisez la commande `ssh` pour vous connecter à `serverb` en tant qu'utilisateur `student`. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis pour se connecter à `serverb`.
2. Utilisez la commande `sudo -i` pour basculer vers l'utilisateur `root`. Si vous y êtes invité, utilisez le mot de passe `student`.
3. Créez une connexion avec un réseau statique à l'aide des paramètres du tableau.

PARAMÈTRE	RÉGLAGE
Nom de la connexion	lab
Nom de l'interface	enX(peut varier, utilisez l'interface ayant 52:54:00:00:fa:0b comme adresse MAC)
Adresse IP	172.25.250.11/24
Adresse de la passerelle	172.25.250.254
Adresse DNS	172.25.250.254

4. Configurez la nouvelle connexion pour qu'elle se lance automatiquement. Les autres connexions ne doivent pas démarrer automatiquement.
5. Modifiez la nouvelle connexion pour qu'elle utilise aussi l'adresse 10.0.1.24.

6. Configurez le fichier **hosts** pour que 10.0.1.1 puisse être référencé comme **private**.
7. Redémarrez le système.
8. À partir de **workstation**, utilisez la commande **ping** pour vérifier que **serverb** est initialisé.

Évaluation

Sur **workstation**, exécutez le script lab net-review grade pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab net-review grade
```

Finish (Terminer)

Sur **workstation**, exécutez le script **lab net-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab net-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

GESTION DE RÉSEAUX

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez configurer les paramètres réseau sur un serveur Red Hat Enterprise Linux.

RÉSULTATS

Vous serez en mesure de configurer deux adresses IPv4 statiques pour l'interface réseau principale.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur **student** à **workstation** avec le mot de passe **student**.

À partir de **workstation**, exécutez la commande **lab net-review start**. La commande exécute un script de démarrage qui détermine si l'hôte, **serverb**, est accessible sur le réseau.

```
[student@workstation ~]$ lab net-review start
```

- Utilisez la commande **ssh** pour vous connecter à **serverb** en tant qu'utilisateur **student**. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis pour se connecter à **serverb**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- Utilisez la commande **sudo -i** pour basculer vers l'utilisateur **root**. Si vous y êtes invité, utilisez le mot de passe **student**.

```
[student@serverb ~]$ sudo -i
[sudo] password for student: student
[root@serverb ~]#
```

- Créez une connexion avec un réseau statique à l'aide des paramètres du tableau.

PARAMÈTRE	RÉGLAGE
Nom de la connexion	lab
Nom de l'interface	enX(peut varier, utilisez l'interface ayant 52:54:00:00:fa:0b comme adresse MAC)

PARAMÈTRE	RÉGLAGE
Adresse IP	172.25.250.11/24
Adresse de la passerelle	172.25.250.254
Adresse DNS	172.25.250.254

Déterminez le nom de l'interface et le nom de la connexion active actuelle. La solution suppose que le nom de l'interface est **enX** et le nom de la connexion **Connexion filaire 1**.

```
[root@serverb ~]# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0b brd ff:ff:ff:ff:ff:ff
[root@serverb ~]# nmcli con show --active
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
```

Créer le profil de connexion **lab** en fonction des informations du tableau décrit dans les instructions. Associez le profil au nom de votre interface réseau indiqué dans la sortie de la commande **ip link** précédente.

```
[root@serverb ~]# nmcli con add con-name lab iface enX type ethernet \
    ipv4.method manual \
    ipv4.address 172.25.250.11/24 ipv4.gateway 172.25.250.254
[root@serverb ~]# nmcli con mod "lab" ipv4.dns 172.25.250.254
```

- Configurez la nouvelle connexion pour qu'elle se lance automatiquement. Les autres connexions ne doivent pas démarrer automatiquement.

```
[root@serverb ~]# nmcli con mod "lab" connection.autoconnect yes
[root@serverb ~]# nmcli con mod "Wired connection 1" connection.autoconnect no
```

- Modifiez la nouvelle connexion pour qu'elle utilise aussi l'adresse 10.0.1.1/24.

```
[root@serverb ~]# nmcli con mod "lab" +ipv4.addresses 10.0.1.1/24
```

Ou encore :

```
[root@serverb ~]# echo "IPADDR1=10.0.1.1" \
>> /etc/sysconfig/network-scripts/ifcfg-lab
[root@serverb ~]# echo "PREFIX1=24" >> /etc/sysconfig/network-scripts/ifcfg-lab
```

- Configurez le fichier **hosts** pour que 10.0.1.1 puisse être référencé comme **private**.

```
[root@serverb ~]# echo "10.0.1.1 private" >> /etc/hosts
```

7. Redémarrez le système.

```
[root@serverb ~]# systemctl reboot
Connection to serverb closed by remote host.
Connection to serverb closed.
[student@workstation ~]$
```

8. À partir de **workstation**, utilisez la commande **ping** pour vérifier que **serverb** est initialisé.

```
[student@workstation ~]$ ping -c3 serverb
PING serverb.lab.example.com (172.25.250.11) 56(84) bytes of data.
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=1 ttl=64
time=0.478 ms
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=2 ttl=64
time=0.504 ms
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=3 ttl=64
time=0.513 ms

--- serverb.lab.example.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 78ms
rtt min/avg/max/mdev = 0.478/0.498/0.513/0.023 ms
[student@workstation ~]$
```

Évaluation

Sur **workstation**, exécutez le script lab net-review grade pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab net-review grade
```

Finish (Terminer)

Sur **workstation**, exécutez le script **lab net-review finish** pour mettre fin à l'atelier.

```
[student@workstation ~]$ lab net-review finish
```

L'atelier est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Le modèle de réseau TCP/IP est un ensemble, simplifié et comprenant quatre couches, d'abstractions décrivant l'interopérabilité des différents protocoles afin que les ordinateurs puissent envoyer du trafic d'un ordinateur à un autre via Internet.
- IPv4 est le principal protocole réseau utilisé sur Internet aujourd'hui. IPv6 a été conçu pour éventuellement remplacer le protocole réseau IPv4. Par défaut, Red Hat Enterprise Linux fonctionne en mode à double pile, en utilisant les deux protocoles en parallèle.
- NetworkManager est un démon qui surveille et gère la configuration du réseau.
- La commande **nmcli** est un outil de ligne de commande permettant de configurer les paramètres du réseau avec NetworkManager.
- Le nom d'hôte statique du système est enregistré dans le fichier **/etc/hostname**. La commande **hostnamectl** permet de modifier ou d'afficher l'état du nom d'hôte du système et des paramètres associés. La commande **hostname** affiche ou modifie temporairement le nom d'hôte du système.

CHAPITRE 13

ARCHIVAGE ET TRANSFERT DE FICHIERS

PROJET

Archiver et copier des fichiers d'un système à l'autre.

OBJECTIFS

- Archiver des fichiers et des répertoires dans un fichier compressé en utilisant tar, et extraire le contenu d'une archive tar existante.
- Transférer des fichiers depuis ou vers un système distant en toute sécurité à l'aide de SSH.
- Synchronisez le contenu d'un fichier ou d'un répertoire local avec une copie sur un serveur distant.

SECTIONS

- Gestion des archives compressées tar (avec exercice guidé)
- Transfert sécurisé de fichiers entre systèmes (avec exercice guidé)
- Synchronisation sécurisée de fichiers entre systèmes (avec exercice guidé)

ATELIER

Archivage et transfert de fichiers

GESTION DES ARCHIVES TAR COMPRESSÉES

OBJECTIFS

Au terme de cette section, vous serez en mesure d'archiver des fichiers et des répertoires dans un fichier compressé en utilisant **tar**, et d'extraire le contenu d'une archive **tar** existante.

LA COMMANDE tar

L'archivage et la compression des fichiers sont utiles pour la création de sauvegardes et pour le transfert de données sur un réseau. L'une des commandes les plus anciennes et les plus courantes pour créer et exploiter des archives de sauvegarde est la commande **tar**.

Avec **tar**, les utilisateurs peuvent rassembler un grand nombre de fichiers en un seul (une archive). Une archive **tar** est une séquence structurée de données de fichier combinée à des métadonnées sur chaque fichier et à un index permettant d'extraire des fichiers individuels. L'archive peut être compressée avec une compression gzip, bzip2 ou xz.

La commande **tar** peut lister le contenu d'une archive ou extraire ses fichiers sur le système en cours.

OPTIONS DE LA COMMANDE tar SÉLECTIONNÉE

Les options de la commande **tar** sont divisées en opérations (l'action que vous voulez entreprendre) : les options générales et les options de compression. Le tableau ci-dessous présente les options courantes, leur version longue et leur description :

Vue d'ensemble du fonctionnement de tar

OPTION	DESCRIPTION
-c, --create	Créer une archive.
-x, --extract	Extraire depuis une archive existante.
-t, --list	Lister la table des matières d'une archive.

Options générales de la commande tar sélectionnée

OPTION	DESCRIPTION
-v, --verbose	Détaillé. Afficher les fichiers archivés ou extraits.
-f, --file=	Nom du fichier. Cette option doit être suivie par le nom du fichier de l'archive à utiliser ou créer.
-p, --preserve-permissions	Conserver les permissions des fichiers et dossiers lors de l'extraction d'une archive, sans soustraire le umask.

Vue d'ensemble des options de compression de tar

OPTION	DESCRIPTION
-z, --gzip	Utiliser la compression gzip (.tar.gz).
-j, --bzip2	Utiliser la compression bzip2 (.tar.bz2). bzip2 atteint généralement un meilleur taux de compression que gzip.
-J, --xz	Utiliser la compression xz (.tar.xz). La compression xz atteint généralement un meilleur taux de compression que bzip2.

LISTE DES OPTIONS DE LA COMMANDE tar

La commande **tar** attend l'une des trois options suivantes :

- Utilisez l'option **-c** ou **--create** pour créer une archive.
- Utilisez l'option **-t** ou **--list** pour lister le contenu d'une archive.
- Utilisez l'option **-x** ou **--extract** pour extraire une archive.

Autres options couramment utilisées :

- Utilisez l'option **-f** ou **--file=** avec un nom de fichier comme argument de l'archive à utiliser.
- Utilisez l'option **-v** ou **--verbose** pour le mode « détaillé » ; ce qui est utile pour voir quels fichiers sont ajoutés à l'archive ou en sont extraits.



NOTE

La commande **tar** prend en réalité en charge un troisième style ancien d'options qui utilise les options à une seule lettre standard, sans **-**. Cela se produit encore fréquemment et vous pouvez rencontrer cette syntaxe lorsque vous utilisez les instructions ou les commandes d'autres personnes. La commande **info tar 'old options'** explique en quoi elle diffère des options courtes habituelles.

Vous pouvez ignorer les anciennes options pour l'instant et vous concentrer sur la syntaxe standard des options courtes et longues.

ARCHIVAGE DE FICHIERS ET DE RÉPERTOIRES

La première option à utiliser lors de la création d'une archive est **c**, suivie de l'option **f**, d'un espace, puis du nom du fichier d'archive à créer, et enfin de la liste des fichiers et répertoires à intégrer dans l'archive. L'archive est créée dans le répertoire actuel, sauf indication contraire.



MISE EN GARDE

Avant de créer une archive tar, vérifiez que le répertoire ne contient aucune autre archive du même nom que celle que vous vous apprêtez à créer. La commande **tar** écrase une archive existante sans avertissement.

La commande suivante crée une archive nommée **archive.tar** avec le contenu de **file1**, **file2** et **file3** dans le répertoire personnel de l'utilisateur.

```
[user@host ~]$ tar -cf archive.tar file1 file2 file3
[user@host ~]$ ls archive.tar
archive.tar
```

La commande **tar** ci-dessus peut également être exécutée à l'aide des options de la version longue.

```
[user@host ~]$ tar --file=archive.tar --create file1 file2 file3
```



NOTE

Lors de l'archivage de fichiers avec des noms de chemins absolus, le / en tête du chemin est supprimée par défaut du nom du fichier. La suppression du / en tête du chemin aide les utilisateurs à ne pas écraser les fichiers importants lors de l'extraction de l'archive. La commande **tar** extrait les fichiers relatifs au répertoire de travail en cours.

Pour que tar puisse archiver les fichiers sélectionnés, l'utilisateur qui exécute la commande **tar** doit obligatoirement avoir l'autorisation de lire ces fichiers. Par exemple, la création d'une archive du dossier **/etc** et de l'intégralité de son contenu nécessite des priviléges **root**, car seul l'utilisateur **root** est autorisé à lire tous les fichiers du répertoire **/etc**. Un utilisateur sans priviléges peut créer une archive du répertoire **/etc**, mais cette archive omet les fichiers qui n'incluent pas de permission de lecture pour l'utilisateur et omet les répertoires qui n'incluent pas de permission de lecture et d'exécution pour l'utilisateur.

Pour créer l'archive tar nommée, **/root/etc.tar**, avec comme contenu le répertoire **/etc** en tant qu'utilisateur **root** :

```
[root@host ~]# tar -cf /root/etc.tar /etc
tar: Removing leading `/' from member names
[root@host ~]#
```



IMPORTANT

Certaines autorisations avancées non décrites dans ce cours, telles que les ACL et les contextes SELinux, ne sont pas automatiquement stockées dans une archive **tar**. Utilisez l'option **--xattrs** lors de la création d'une archive pour stocker ces attributs étendus dans l'archive tar.

LISTER LE CONTENU D'UNE ARCHIVE

L'option **t** indique à **tar** de lister le contenu (table des matières, d'où la lettre **t**) de l'archive. Utilisez l'option **f** avec le nom de l'archive à interroger. Par exemple :

```
[root@host ~]# tar -tf /root/etc.tar
etc/
etc/fstab
etc/crypttab
etc/mtab
...output omitted...
```

EXTRACTION DES FICHIERS D'UNE ARCHIVE

Une archive tar doit normalement être extraite dans un répertoire vide, afin d'éviter l'écrasement de fichiers existants. Quand root extrait une archive, la commande **tar** conserve la propriété de l'utilisateur et du groupe originaux des fichiers. Si un utilisateur normal extrait des fichiers en utilisant **tar**, la propriété du fichier appartient à l'utilisateur qui extrait les fichiers de l'archive.

Pour restaurer des fichiers à partir de l'archive **/root/etc.tar** dans le répertoire **/root/etcbackup**, exécutez :

```
[root@host ~]# mkdir /root/etcbackup
[root@host ~]# cd /root/etcbackup
[root@host etcbackup]# tar -tf /root/etc.tar
etc/
etc/fstab
etc/crypttab
etc/mtab
...output omitted...
[root@host etcbackup]# tar -xf /root/etc.tar
```

Par défaut, lorsque les fichiers sont extraits d'une archive, le **umask** est soustrait des permissions du contenu de l'archive. Pour préserver les permissions d'un fichier archivé, utilisez l'option **p** lors de l'extraction de l'archive.

Dans cet exemple, une archive nommée **/root/myscripts.tar** est extraite dans le répertoire **/root/scripts**, tout en préservant les permissions des fichiers extraits :

```
[root@host ~]# mkdir /root/scripts
[root@host ~]# cd /root/scripts
[root@host scripts]# tar -xpf /root/myscripts.tar
```

CRÉATION D'UNE ARCHIVE COMPRESSÉE

La commande **tar** prend en charge trois méthodes de compression. Il existe trois méthodes de compression différentes pour la commande **tar**. La compression **gzip** est la méthode la plus rapide, la plus ancienne et la plus répandue sur toutes les distributions, et même sur toutes les plates-formes. La compression **bzip2** crée des fichiers d'archive plus petits que **gzip** mais est moins répandue que **gzip**, alors que la méthode de compression **xz** est relativement nouvelle, mais offre généralement le meilleur taux de compression des méthodes disponibles.



NOTE

L'efficacité d'un algorithme de compression dépend du type de données à compresser. Les fichiers de données déjà compressés, comme les formats d'images compressés ou les fichiers RPM, entraînent généralement un faible taux de compression.

Il est généralement conseillé d'utiliser un répertoire unique au premier niveau de l'arborescence, qui peut contenir d'autres dossiers et fichiers, pour simplifier l'extraction des fichiers de manière organisée.

Utilisez l'une des options suivantes pour créer une archive tar compressée :

- **-z** ou **--gzip** pour la compression **gzip** (**filename.tar.gz** ou **filename.tgz**)

CHAPITRE 13 | Archivage et transfert de fichiers

- **-j** ou **--bzip2** pour la compression bzip2 (**filename.tar.bz2**)
- **-J** ou **-xz** pour la compression xz (**filename.tar.xz**)

Pour créer une archive gzip compressée nommée **/root/etcbackup.tar.gz**, avec le contenu du répertoire **/etc** sur host :

```
[root@host ~]# tar -czf /root/etcbackup.tar.gz /etc
tar: Removing leading `/' from member names
```

Pour créer une archive bzip2 compressée nommée **/root/logbackup.tar.bz2**, avec le contenu du répertoire **/var/log** sur host :

```
[root@host ~]$ tar -cJf /root/logbackup.tar.bz2 /var/log
tar: Removing leading `/' from member names
```

Pour créer une archive xz compressée nommée **/root/sshconfig.tar.xz**, avec le contenu du répertoire **/etc/ssh** sur host :

```
[root@host ~]$ tar -cJf /root/sshconfig.tar.xz /etc/ssh
tar: Removing leading `/' from member names
```

Après avoir créé une archive, vérifiez le contenu d'une archive à l'aide des options **tf**. Il n'est pas obligatoire d'utiliser l'option pour l'agent de compression lors de la création de la liste du contenu d'un fichier d'archive compressé. Par exemple, pour lister le contenu archivé dans le fichier **/root/etcbackup.tar.gz**, qui utilise la compression gzip, utilisez la commande suivante :

```
[root@host ~]# tar -tf /root/etcbackup.tar.gz /etc
etc/
etc/fstab
etc/crypttab
etc/mtab
...output omitted...
```

EXTRACTION D'UNE ARCHIVE COMPRESSÉE

La première étape de l'extraction d'une archive **tar** compressée consiste à déterminer où extraire les fichiers archivés, puis à créer le répertoire cible et à l'ouvrir. La commande **tar** détermine la méthode de compression qui a été utilisée, et il n'est généralement pas nécessaire d'utiliser la même option de compression que lors de la création de l'archive. L'ajout de la méthode de décompression à la commande **tar** est valide. Si vous choisissez de le faire, vous devez utiliser l'option de type de décompression appropriée ; sinon, tar renvoie une erreur concernant le type de décompression spécifié dans les options ne correspondant pas au type de décompression du fichier.

Pour extraire le contenu d'une archive comprimée gzip nommée **/root/etcbackup.tar.gz** dans le répertoire **/tmp/etcbackup** :

```
[root@host ~]# mkdir /tmp/etcbackup
[root@host ~]# cd /tmp/etcbackup
[root@host etcbackup]# tar -tf /root/etcbackup.tar.gz
etc/
etc/fstab
```

CHAPITRE 13 | Archivage et transfert de fichiers

```
etc/crypttab  
etc/mtab  
...output omitted...  
[root@host etcbackup]# tar -xzf /root/etcbackup.tar.gz
```

Pour extraire le contenu d'une archive comprimée bzip2 nommée **/root/logbackup.tar.bz2** dans le répertoire **/tmp/logbackup** :

```
[root@host ~]# mkdir /tmp/logbackup  
[root@host ~]# cd /tmp/logbackup  
[root@host logbackup]# tar -tf /root/logbackup.tar.bz2  
var/log/  
var/log/lastlog  
var/log/README  
var/log/private/  
var/log/wtmp  
var/log/btmp  
...output omitted...  
[root@host logbackup]# tar -xjf /root/logbackup.tar.bz2
```

Pour extraire le contenu d'une archive comprimée xz nommée **/root/sshbackup.tar.xz** dans le répertoire **/tmp/sshbackup** :

```
[root@host ~]$ mkdir /tmp/sshbackup  
[root@host ~]# cd /tmp/sshbackup  
[root@host logbackup]# tar -tf /root/sshbackup.tar.xz  
etc/ssh/  
etc/ssh/moduli  
etc/ssh/ssh_config  
etc/ssh/ssh_config.d/  
etc/ssh/ssh_config.d/05-redhat.conf  
etc/ssh/sshd_config  
...output omitted...  
[root@host sshbackup]# tar -xJf /root/sshbackup.tar.xz
```

Lister le contenu d'une archive tar compressée fonctionne de la même manière que pour une archive **tar** non compressée.

**NOTE**

De plus, **gzip**, **bzip2** et **xz** peuvent être utilisés indépendamment pour compresser des fichiers isolés. Par exemple, la commande **gzip etc.tar** donne le fichier compressé **etc.tar.gz**, tandis que la commande **bzip2 abc.tar** donne le fichier compressé **abc.tar.bz2**, et que la commande **xz myarchive.tar** donne le fichier compressé **myarchive.tar.xz**.

Les commandes correspondantes pour décompresser sont **gunzip**, **bunzip2** et **unxz**. Par exemple, la commande **gunzip /tmp/etc.tar.gz** donne le fichier tar non compressé **etc.tar**, tandis que la commande **bunzip2 abc.tar.bz2** donne le fichier tar non compressé **abc.tar**, et que la commande **unxz myarchive.tar.xz** donne le fichier tar non compressé **myarchive.tar**.



RÉFÉRENCES

Pages de manuel `tar(1)`, `gzip(1)`, `gunzip(1)`, `bzip2(1)`, `bunzip2(1)`, `xz(1)`, `unxz(1)`

► EXERCICE GUIDÉ

GESTION DES ARCHIVES TAR COMPRESSÉES

Au cours de cet exercice, vous allez créer des fichiers d'archive et extraire leur contenu à l'aide de la commande tar.

RÉSULTATS

Vous serez en mesure d'archiver une arborescence de dossiers et d'extraire le contenu de l'archive dans un autre emplacement.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab archive-manage start**. La commande exécute un script de démarrage qui détermine si l'hôte, servera, est accessible sur le réseau. Le script s'assure également que le fichier et le répertoire à créer dans l'exercice n'existent pas sur servera.

```
[student@workstation ~]$ lab archive-manage start
```

- 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Basculez vers l'utilisateur root, étant donné que seul l'utilisateur root peut accéder à l'intégralité du contenu du répertoire **/etc**.

```
[student@servera ~]$ su -
Password: redhat
[root@servera ~]#
```

- 3. Utilisez la commande **tar** avec les options **-czf** pour créer une archive du répertoire **/etc** en utilisant la compression gzip. Enregistrez le fichier archive sous **/tmp/etc.tar.gz**.

```
[root@servera ~]# tar -czf /tmp/etc.tar.gz /etc
tar: Removing leading `/' from member names
[root@servera ~]#
```

- 4. Utilisez la commande **tar** avec les options **-tzf** pour vérifier que l'archive **etc.tar.gz** contient les fichiers du répertoire **/etc**.

```
[root@servera ~]# tar -tzf /tmp/etc.tar.gz
etc/
etc/mtab
etc/fstab
etc/crypttab
etc/resolv.conf
...output omitted...
```

- 5. Sur servera, créez un répertoire nommé **/backuptest**. Vérifiez que le fichier de sauvegarde **etc.tar.gz** est une archive valide en décompressant le fichier dans le répertoire **/backuptest**.

5.1. Créez le répertoire **/backuptest**.

```
[root@servera ~]# mkdir /backuptest
```

5.2. Choisissez le répertoire **/backuptest**.

```
[root@servera ~]# cd /backuptest
[root@servera backuptest]#
```

5.3. Listez le contenu de l'archive **etc.tar.gz** avant de l'extraire.

```
[root@servera backuptest]# tar -tzf /tmp/etc.tar.gz
etc/
etc/mtab
etc/fstab
etc/crypttab
etc/resolv.conf
...output omitted...
```

5.4. Extrayez l'archive **/tmp/etc.tar.gz** dans le répertoire **/backuptest**.

```
[root@servera backuptest]# tar -xzf /tmp/etc.tar.gz
[root@servera backuptest]#
```

5.5. Listez le contenu du répertoire **/backuptest**. Vérifiez que le répertoire contient les fichiers du répertoire **/etc**.

```
[root@servera backuptest]# ls -l
total 12
drwxr-xr-x. 95 root root 8192 Feb  8 10:16 etc
[root@servera backuptest]# cd etc
[root@servera etc]# ls -l
total 1204
-rw-r--r--.  1 root root      16 Jan 16 23:41 adjtime
-rw-r--r--.  1 root root    1518 Sep 10 17:21 aliases
drwxr-xr-x.  2 root root     169 Feb  4 21:58 alternatives
-rw-r--r--.  1 root root     541 Oct  2 21:01 anacrontab
...output omitted...
```

► 6. Quittez servera.

```
[root@servera backuptest]# exit  
logout  
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation]$
```

Fin

Sur workstation, exécutez le script **lab archive-manage finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab archive-manage finish
```

L'exercice guidé est maintenant terminé.

TRANSFERT SÉCURISÉ DE FICHIERS ENTRE SYSTÈMES

OBJECTIFS

Au terme de cette section, vous serez en mesure de transférer des fichiers vers ou depuis un système distant en toute sécurité à l'aide de SSH.

TRANSFERT DE FICHIERS À L'AIDE D'UNE COPIE SÉCURISÉE

OpenSSH est utile pour exécuter les commandes shell en toute sécurité sur les systèmes distants. La commande de copie sécurisée, **scp**, qui fait partie de la suite OpenSSH, copie des fichiers d'un système distant vers le système local ou d'un système local vers un système distant. La commande utilise le serveur SSH pour l'authentification et chiffre les données lors du transfert.

Vous pouvez spécifier un emplacement distant pour la source ou la destination des fichiers que vous copiez. Le format de l'emplacement distant doit être sous la forme **[user@]host:/path**. La partie **user@** de l'argument est facultative. Si elle n'y figure pas, votre nom d'utilisateur local actuel sera utilisé. Lorsque vous exécutez la commande, votre client **scp** s'authentifiera sur le serveur SSH distant comme **ssh**, en utilisant une authentification basée sur une clé ou en vous demandant votre mot de passe.

L'exemple suivant montre comment copier les fichiers locaux **/etc/yum.conf** et **/etc/hosts** sur host, dans le répertoire personnel de **remoteuser** sur le système distant **remotehost** :

```
[user@host ~]$ scp /etc/yum.conf /etc/hosts remoteuser@remotehost:/home/remoteuser
remoteuser@remotehost's password: password
yum.conf                                100%   813      0.8KB/s   00:00
hosts                                    100%   227      0.2KB/s   00:00
```

Vous pouvez également copier un fichier dans l'autre sens, d'un système distant vers le système de fichiers local. Dans cet exemple, le fichier **/etc/hostname** sur **remotehost** est copié dans le répertoire local **/home/user**. La commande **scp** s'authentifie auprès de **remotehost** en tant qu'utilisateur **remoteuser**.

```
[user@host ~]$ scp remoteuser@remotehost:/etc/hostname /home/user
remoteuser@remotehost's password: password
hostname                               100%    22      0.0KB/s   00:00
```

Pour copier une arborescence de répertoires complète de manière récursive, utilisez l'option **-r**. Dans l'exemple suivant, le répertoire distant **/var/log** sur **remotehost** est copié de manière récursive vers le répertoire local **/tmp** sur host. Vous devez vous connecter au système distant en tant que **root** pour vous assurer de pouvoir lire tous les fichiers du répertoire **/var/log** distant.

```
[user@host ~]$ scp -r root@remoteuser:/var/log /tmp
root@remotehost's password: password
...output omitted...
```

TRANSFERT DE FICHIERS À L'AIDE DU SFTP

Pour charger ou télécharger des fichiers de manière interactive depuis un serveur SSH, utilisez la commande **sftp** (Secure File Transfer Program). Une session avec la commande **sftp** utilise le mécanisme d'authentification sécurisée et le transfert de données chiffrées vers et depuis le serveur SSH.

Comme avec la commande **scp**, la commande **sftp** utilise **[user@]host** pour identifier le système cible et le nom d'utilisateur. Si vous ne spécifiez pas d'utilisateur, la commande tentera de se connecter en utilisant votre nom d'utilisateur local en tant que nom d'utilisateur distant. Une invite **sftp>** s'affichera alors.

```
[user@host ~]$ sftp remoteuser@remotehost
remoteuser@remotehost's password: password
Connected to remotehost.
sftp>
```

La session **sftp** interactive accepte diverses commandes qui fonctionnent de la même façon sur le système de fichiers distant que sur le système de fichiers local, telles que **ls**, **cd**, **mkdir**, **rmdir** et **pwd**. La commande **put** charge un fichier sur le système distant. La commande **get** télécharge un fichier depuis le système distant. La commande **exit** met fin à la session **sftp**.

Pour télécharger le fichier **/etc/hosts** sur le système local dans le répertoire **/home/remoteuser/hostbackup** qui vient d'être créé sur **remotehost**. La session **sftp** suppose toujours que la commande **put** est suivie d'un fichier du système de fichiers local et démarre dans le répertoire personnel de l'utilisateur qui se connecte, en l'occurrence **/home/remoteuser** :

```
sftp> mkdir hostbackup
sftp> cd hostbackup
sftp> put /etc/hosts
Uploading /etc/hosts to /home/remoteuser/hostbackup/hosts
/etc/hosts                                         100%   227      0.2KB/s  00:00
sftp>
```

Pour télécharger **/etc/yum.conf** à partir de l'hôte distant vers le répertoire courant sur le système local, exécutez la commande **get /etc/yum.conf** et quittez la session **sftp** avec la commande **exit**.

```
sftp> get /etc/yum.conf
Fetching /etc/yum.conf to yum.conf
/etc/yum.conf                                         100%   813      0.8KB/s  00:00
sftp> exit
[user@host ~]$
```



RÉFÉRENCES

Pages de manuel **scp(1)** et **sftp(1)**

► EXERCICE GUIDÉ

TRANSFERT SÉCURISÉ DE FICHIERS ENTRE SYSTÈMES

Au cours de cet exercice, vous allez copier des fichiers depuis un système distant vers un répertoire local en utilisant **scp**.

RÉSULTATS

Vous serez en mesure de copier des fichiers depuis un hôte distant vers un répertoire de la machine locale.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur **student** à **workstation** avec le mot de passe **student**.

À partir de **workstation**, exécutez la commande **lab archive-transfer start**. La commande exécute un script de démarrage qui détermine si les hôtes **servera** et **serverb** sont accessibles sur le réseau. Le script s'assure également que le fichier et le répertoire à créer dans l'exercice n'existent pas sur **servera**.

```
[student@workstation ~]$ lab archive-transfer start
```

- 1. Utilisez la commande **ssh** pour vous connecter à **servera** en tant qu'utilisateur **student**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Utilisez la commande **scp** pour copier le répertoire **/etc/ssh** à partir de **serverb** vers le répertoire **/home/student/serverbackup** sur **servera**.

- 2.1. Sur **servera**, créez un répertoire nommé **/home/student/serverbackup**.

```
[student@servera ~]$ mkdir ~/serverbackup
```

- 2.2. Utilisez la commande **scp** pour copier de façon récursive le répertoire **/etc/ssh** à partir de **serverb** vers le répertoire **/home/student/serverbackup** sur **servera**. Lorsqu'un message vous y invite, saisissez **redhat** comme mot de passe. Notez que seul l'utilisateur **root** peut lire tout le contenu du répertoire **/etc/ssh**.

```
[student@servera ~]$ scp -r root@serverb:/etc/ssh ~/serverbackup
The authenticity of host 'serverb (172.25.250.11)' can't be established.
ECDSA key fingerprint is SHA256:qaS0PToLrqLC02XGk1A0iY7CaP7aPKimerDoaUkv720.
Are you sure you want to continue connecting (yes/no)? yes
```

```
Warning: Permanently added 'serverb,172.25.250.11' (ECDSA) to the list of known hosts.
root@serverb's password: redhat
moduli                                100%  550KB  57.9MB/s  00:00
ssh_config                             100% 1727    1.4MB/s  00:00
05-redhat.conf                         100%  690    1.6MB/s  00:00
sshd_config                            100% 4469    9.5MB/s  00:00
ssh_host_ed25519_key                  100%  387    1.2MB/s  00:00
ssh_host_ed25519_key.pub              100%   82   268.1KB/s  00:00
ssh_host_ecdsa_key                    100%  492    1.5MB/s  00:00
ssh_host_ecdsa_key.pub                100%  162   538.7KB/s  00:00
ssh_host_rsa_key                      100% 1799    4.9MB/s  00:00
ssh_host_rsa_key.pub                  100%  382    1.2MB/s  00:00
```

- 2.3. Vérifiez que le répertoire **/etc/ssh** de serverb est copié dans le répertoire **/home/student/serverbackup** sur servera.

```
[student@servera ~]$ ls -lR ~/serverbackup
/home/student/serverbackup:
total 0
drwxr-xr-x. 3 student student 245 Feb 11 18:35 ssh

/home/student/serverbackup/ssh:
total 588
-rw-r--r--. 1 student student 563386 Feb 11 18:35 moduli
-rw-r--r--. 1 student student 1727 Feb 11 18:35 ssh_config
drwxr-xr-x. 2 student student 28 Feb 11 18:35 ssh_config.d
-rw-----. 1 student student 4469 Feb 11 18:35 sshd_config
-rw-----. 1 student student 492 Feb 11 18:35 ssh_host_ecdsa_key
-rw-r--r--. 1 student student 162 Feb 11 18:35 ssh_host_ecdsa_key.pub
-rw-r-----. 1 student student 387 Feb 11 18:35 ssh_host_ed25519_key
-rw-r-----. 1 student student 82 Feb 11 18:35 ssh_host_ed25519_key.pub
-rw-r-----. 1 student student 1799 Feb 11 18:35 ssh_host_rsa_key
-rw-r--r--. 1 student student 382 Feb 11 18:35 ssh_host_rsa_key.pub

/home/student/serverbackup/ssh/ssh_config.d:
total 4
-rw-r--r--. 1 student student 690 Feb 11 18:35 05-redhat.conf
```

► 3. Quittez servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

Fin

Sur workstation, exéutez le script **lab archive-transfer finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab archive-transfer finish
```

L'exercice guidé est maintenant terminé.

SYNCHRONISATION DE FICHIERS SÉCURISÉE ENTRE DES SYSTÈMES

OBJECTIFS

Au terme de cette section, vous serez en mesure de synchroniser efficacement et de manière sécurisée le contenu d'un fichier ou d'un répertoire local avec une copie sur un serveur distant.

SYNCHRONISER DES FICHIERS ET DES RÉPERTOIRES AVEC **rsync**

La commande **rsync** constitue un autre moyen de copier des fichiers d'un système à un autre de manière sécurisée. L'outil utilise un algorithme qui minimise la quantité de données copiées en synchronisant uniquement les parties des fichiers qui ont été modifiées. Il diffère de **scp** en ce sens que, si deux fichiers ou répertoires sont similaires sur deux systèmes, **rsync** copie uniquement les différences entre les systèmes de fichiers, là où **scp** effectuerait une copie complète.

L'un des avantages de la commande **rsync** est qu'elle peut copier des fichiers entre un système local et un système distant en assurant sécurité et efficacité. Bien que la synchronisation initiale d'un répertoire prenne environ le même temps que sa copie, toute synchronisation ultérieure se contente de copier les différences via le réseau, ce qui accélère les mises à jour de façon considérable.

L'une des options importantes de la commande **rsync** est **-n**, qui permet d'effectuer un essai à blanc. Un essai à blanc est une simulation de ce qui se produit lorsque la commande est exécutée. L'essai à blanc montre les changements que **rsync** effectuerait lorsque la commande est exécutée normalement. Effectuez un essai à blanc avant l'opération **rsync**, afin de vous assurer qu'aucun fichier important ne sera écrasé ou supprimé.

Les deux options courantes lors de la synchronisation à l'aide de **rsync** sont **-v** et **-a**.

L'option **-v** ou **--verbose** fournit une sortie plus détaillée, ce qui est utile pour la résolution de problèmes et pour voir les progrès en direct.

L'option **-a** ou **--archive** active le « mode Archive », ce qui permet de copier de façon récursive et d'activer un grand nombre d'options utiles qui conservent la plupart des caractéristiques des fichiers. Le mode Archive revient à spécifier les options suivantes :

Options activées avec **rsync -a (mode Archive)**

OPTION	DESCRIPTION
-r, --recursive	synchronise l'arborescence de répertoires complète de manière récursive
-l, --links	synchronise les liens symboliques
-p, --perms	permet de conserver les permissions
-t, --times	préserve les horodatages

OPTION	DESCRIPTION
-g, --group	préserve la propriété du groupe
-o, --owner	préserve le propriétaire des fichiers
-D, --devices	synchronise les fichiers de périphérique

Le mode Archive ne conserve pas les liens physiques, car cela peut prolonger considérablement la synchronisation. Si vous souhaitez conserver également les liens physiques, ajoutez l'option **-H**.



NOTE

Si vous utilisez des autorisations avancées, vous pourriez avoir besoin de deux options supplémentaires :

- **-A** pour conserver les ACL
- **-X** pour conserver les contextes SELinux

Vous pouvez utiliser **rsync** pour synchroniser le contenu d'un fichier ou d'un répertoire local avec un fichier ou un répertoire sur un ordinateur distant, en utilisant l'un des ordinateurs comme source. Vous pouvez également synchroniser le contenu de deux fichiers ou répertoires locaux.

Par exemple, pour synchroniser le contenu du répertoire **/var/log** avec le répertoire **/tmp** :

```
[user@host ~]$ su -
Password: password
[root@host ~]# rsync -av /var/log /tmp
receiving incremental file list
log/
log/README
log/boot.log
...output omitted...
log/tuned/tuned.log

sent 11,592,423 bytes received 779 bytes 23,186,404.00 bytes/sec
total size is 11,586,755 speedup is 1.00
[root@host ~]$ ls /tmp
log  ssh-RLjDdarkKiW1
[root@host ~]$
```

Une barre oblique sur le répertoire source permet de synchroniser le contenu de ce répertoire sans créer de nouveau le sous-répertoire dans le répertoire cible. Dans cet exemple, le répertoire **log** n'est pas créé dans le répertoire **/tmp**, seul le contenu de **/var/log/** est synchronisé dans **/tmp**.

```
[root@host ~]# rsync -av /var/log/ /tmp
sending incremental file list
./
README
boot.log
...output omitted...
tuned/tuned.log
```

```
sent 11,592,389 bytes received 778 bytes 23,186,334.00 bytes/sec
total size is 11,586,755 speedup is 1.00
[root@host ~]# ls /tmp
anaconda           dnf.rpm.log-20190318  private
audit              dnf.rpm.log-20190324  qemu-ga
boot.log           dnf.rpm.log-20190331  README
...output omitted...
```



IMPORTANT

Lors de la saisie du nom du répertoire source dans la commande **rsync**, il est important de savoir si le nom du répertoire contient ou non une barre oblique de fin. Cela détermine si c'est le *répertoire* qui est synchronisé avec la cible, ou seulement *le contenu du répertoire*.

La saisie semi-automatique bash via la touche de **tabulation** ajoute automatiquement une barre oblique à la fin des noms de répertoire.

Comme avec les commandes **scp** et **sftp**, **rsync** spécifie les emplacements distants à l'aide du format **[user@]host :/path**. L'emplacement distant peut être le système source ou le système de destination, mais l'un des deux ordinateurs doit être local.

Pour conserver la propriété du fichier, vous devez être un utilisateur **root** sur le système de destination. Si la destination est distante, authentifiez-vous en tant que **root**. Si la destination est locale, vous devez exécuter **rsync** comme **root**.

Dans cet exemple, synchronisez le répertoire local **/var/log** avec le répertoire **/tmp** sur le système **remotehost**:

```
[root@host ~]# rsync -av /var/log remotehost:/tmp
root@remotehost's password: password
receiving incremental file list
log/
log/README
log/boot.log
...output omitted...
sent 9,783 bytes received 290,576 bytes 85,816.86 bytes/sec
total size is 11,585,690 speedup is 38.57
```

De la même manière, le répertoire distant **/var/log** sur **remotehost** peut être synchronisé avec le répertoire local **/tmp** sur **host**:

```
[root@host ~]# rsync -av remotehost:/var/log /tmp
root@remotehost's password: password
receiving incremental file list
log/boot.log
log/dnf.librepo.log
log/dnf.log
...output omitted...
sent 9,783 bytes received 290,576 bytes 85,816.86 bytes/sec
total size is 11,585,690 speedup is 38.57
```



RÉFÉRENCES

Page de manuel `rsync(1)`

► EXERCICE GUIDÉ

SYNCHRONISATION DE FICHIERS SÉCURISÉE ENTRE DES SYSTÈMES

Au cours de cet exercice, vous allez synchroniser le contenu d'un répertoire local avec une copie sur un serveur distant à l'aide de **rsync** sur SSH.

RÉSULTATS

Vous serez capable d'utiliser la commande **rsync** pour synchroniser le contenu d'un répertoire local avec une copie sur un serveur distant.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de **workstation**, exécutez la commande **lab archive-sync start**. La commande exécute un script de démarrage qui détermine si les hôtes, **servera** et **serverb**, sont accessibles sur le réseau. Le script s'assure également que le fichier et le répertoire à créer dans l'exercice n'existent pas sur **servera**.

```
[student@workstation ~]$ lab archive-sync start
```

- 1. Utilisez la commande **ssh** pour vous connecter à **servera** en tant qu'utilisateur **student**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Créez un répertoire nommé **/home/student/serverlogs** sur **servera**. Utilisez la commande **rsync** pour créer en toute sécurité une copie initiale de l'arborescence de répertoires **/var/log** sur **serverb** dans le répertoire **/home/student/serverlogs** sur **servera**.
- 2.1. Sur **servera**, créez le répertoire cible nommé **/home/student/serverlogs** pour stocker les fichiers journaux synchronisés depuis **serverb**.

```
[student@servera ~]$ mkdir ~/serverlogs
```

- 2.2. Utilisez la commande **rsync** pour synchroniser l'arborescence de répertoires **/var/log** sur **serverb** avec le répertoire **/home/student/serverlogs** sur **servera**. Notez que seul l'utilisateur **root** peut lire tout le contenu du répertoire **/var/log** sur **serverb**. Tous les fichiers sont transférés lors de la synchronisation initiale.

```
[student@servera ~]$ rsync -av root@serverb:/var/log ~/serverlogs
root@serverb's password: redhat
receiving incremental file list
log/
log/README
log/boot.log
...output omitted...
log/tuned/tuned.log

sent 992 bytes received 13,775,064 bytes 2,119,393.23 bytes/sec
total size is 13,768,109 speedup is 1.00
```

- ▶ 3. En tant qu'utilisateur **root** sur **serverb**, exécutez la commande **logger "Log files synchronized"** pour créer un nouvel enregistrement dans le fichier journal **/var/log/messages** pour montrer quand la dernière synchronisation a eu lieu.

```
[student@servera ~]$ ssh root@serverb 'logger "Log files synchronized"'
Password: redhat
[student@servera ~]$
```

- ▶ 4. Utilisez la commande **rsync** pour synchroniser en toute sécurité l'arborescence de répertoires **/var/log** sur **serverb** avec le répertoire **/home/student/serverlogs** sur **servera**. Notez qu'à ce stade, seuls les fichiers journaux modifiés sont transférés.

```
[student@servera ~]$ rsync -av root@serverb:/var/log ~/serverlogs
root@serverb's password: redhat
receiving incremental file list
log/messages
log/secure
log/audit/audit.log

sent 3,496 bytes received 27,243 bytes 8,782.57 bytes/sec
total size is 11,502,695 speedup is 374.21
```

- ▶ 5. Quittez **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

Fin

Sur **workstation**, exécutez le script **lab archive-sync finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab archive-sync finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

ARCHIVAGE ET TRANSFERT DE FICHIERS

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez utiliser les commandes **tar**, **rsync** et **scp** pour archiver et sauvegarder le contenu de répertoires.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Synchroniser un répertoire distant sur un répertoire local.
- Créer une archive du contenu d'un répertoire synchronisé.
- Copier en toute sécurité une archive sur un hôte distant.
- Extraire une archive.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab archive-review start**. La commande exécute un script de démarrage qui détermine si les hôtes servera et serverb sont accessibles sur le réseau. Le script s'assure également que les fichiers et les répertoires à créer dans l'atelier n'existent pas sur serverb et workstation.

```
[student@workstation ~]$ lab archive-review start
```

1. Sur serverb, synchronisez l'arborescence de répertoire **/etc** de servera vers le répertoire **/configsync**.
2. Utilisez la compression gzip pour créer une archive nommée **configfile-backup-servera.tar.gz** avec le contenu du répertoire **/configsync**.
3. Copiez en toute sécurité le fichier d'archive **/root/configfile-backup-servera.tar.gz** à partir de serverb vers le répertoire **/home/student** sur workstation en tant qu'utilisateur student avec le mot de passe student.
4. Sur workstation, extrayez le contenu de l'archive **/home/student/configfile-backup-servera.tar.gz** dans le répertoire **/tmp/savedconfig/**.
5. Sur workstation, retournez dans le répertoire personnel student.

```
[student@workstation savedconfig]$ cd
```

Évaluation

Sur workstation, exécutez le script **lab archive-review grade** pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab archive-review grade
```

Finish (Terminer)

Sur workstation, exécutez le script **lab archive-review finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab archive-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

ARCHIVAGE ET TRANSFERT DE FICHIERS

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez utiliser les commandes **tar**, **rsync** et **scp** pour archiver et sauvegarder le contenu de répertoires.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Synchroniser un répertoire distant sur un répertoire local.
- Créer une archive du contenu d'un répertoire synchronisé.
- Copier en toute sécurité une archive sur un hôte distant.
- Extraire une archive.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur **student** à **workstation** avec le mot de passe **student**.

À partir de **workstation**, exécutez la commande **lab archive-review start**. La commande exécute un script de démarrage qui détermine si les hôtes **servera** et **serverb** sont accessibles sur le réseau. Le script s'assure également que les fichiers et les répertoires à créer dans l'atelier n'existent pas sur **serverb** et **workstation**.

```
[student@workstation ~]$ lab archive-review start
```

1. Sur **serverb**, synchronisez l'arborescence de répertoire **/etc** de **servera** vers le répertoire **/configsync**.
 - 1.1. Utilisez la commande **ssh** pour vous connecter à **serverb** en tant qu'utilisateur **student**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Utilisez la commande **su** pour basculer vers l'utilisateur **root** parce que la création du répertoire **/configsync** nécessite des priviléges de superutilisateur. Au cours des étapes suivantes, vous allez archiver les fichiers présents dans l'arborescence de répertoires **/etc**, qui appartiennent à l'utilisateur **root**. Cela nécessite également des priviléges de superutilisateur.

```
[student@serverb ~]$ su -
Password: redhat
[root@serverb ~]#
```

- 1.3. Créez le répertoire **/configsync** pour stocker les fichiers de configuration synchronisés à partir de servera.

```
[root@serverb ~]# mkdir /configsync
```

- 1.4. Utilisez la commande **rsync** pour synchroniser l'arborescence de répertoires **/etc** sur servera avec le répertoire **/configsync** sur serverb.

Rappelez-vous que seul l'utilisateur root peut lire tout le contenu du répertoire **/etc** sur server.

```
[root@serverb ~]# rsync -av root@servera:/etc /configsync
root@servera's password: redhat
receiving incremental file list
etc/
etc/.pwd.lock
...output omitted...
etc/yum/protected.d -> ../dnf/protected.d
etc/yum/vars -> ../dnf/vars

sent 10,958 bytes received 21,665,987 bytes 3,334,914.62 bytes/sec
total size is 21,615,767 speedup is 1.00
```

2. Utilisez la compression gzip pour créer une archive nommée **configfile-backup-servera.tar.gz** avec le contenu du répertoire **/configsync**.

- 2.1. Utilisez la commande **tar** avec les options **-czf** pour créer une archive gzip compressée.

```
[root@serverb ~]# tar -czf configfile-backup-servera.tar.gz /configsync
tar: Removing leading `/' from member names
[root@serverb ~]#
```

- 2.2. Utilisez la commande **tar** avec les options **-tzf** pour lister le contenu de l'archive **configfile-backup-servera.tar.gz**.

```
[root@serverb ~]# tar -tzf configfile-backup-servera.tar.gz
...output omitted...
configsync/etc/vimrc
configsync/etc/wgetrc
configsync/etc/xattr.conf
```

3. Copiez en toute sécurité le fichier d'archive **/root/configfile-backup-servera.tar.gz** à partir de serverb vers le répertoire **/home/student** sur workstation en tant qu'utilisateur student avec le mot de passe student.

```
[root@serverb ~]# scp ~/configfile-backup-servera.tar.gz \
student@workstation:/home/student
...output omitted...
student@workstation's password: student
configfile-backup-servera.tar.gz          100% 5110KB 64.5MB/s  00:00
```

4. Sur workstation, extrayez le contenu de l'archive **/home/student/configfile-backup-servera.tar.gz** dans le répertoire **/tmp/savedconfig**.

4.1. Quittez serverb.

```
[root@serverb ~]# exit
logout
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation]$
```

4.2. Créez le répertoire **/tmp/savedconfig** pour y extraire le contenu de l'archive **/home/student/configfile-backup-servera.tar.gz**.

```
[student@workstation ~]$ mkdir /tmp/savedconfig
```

4.3. Choisissez le répertoire **/tmp/savedconfig**.

```
[student@workstation ~]$ cd /tmp/savedconfig
[student@workstation savedconfig]$
```

4.4. Utilisez la commande **tar** avec les options **-tzf** pour lister le contenu de l'archive **configfile-backup-servera.tar.gz**

```
[student@workstation savedconfig]$ tar -tzf ~/configfile-backup-servera.tar.gz
...output omitted...
configsync/etc/vimrc
configsync/etc/wgetrc
configsync/etc/xattr.conf
```

4.5. Utilisez la commande **tar** avec les options **-xzf** pour extraire le contenu de l'archive **/home/student/configfile-backup-servera.tar.gz** dans le répertoire **/tmp/savedconfig**.

```
[student@workstation savedconfig]$ tar -xzf ~/configfile-backup-servera.tar.gz
[student@workstation savedconfig]$
```

4.6. Listez l'arborescence de répertoires pour vérifie que le répertoire contient les fichiers du répertoire **/etc**.

```
[student@workstation savedconfig]$ ls -lR
.:
total 0
```

```
drwxr-xr-x. 3 student student 17 Feb 13 10:13 configsync  
  
./configs sync:  
total 12  
drwxr-xr-x. 95 student student 8192 Feb 13 09:41 etc  
  
. / configs sync /etc:  
total 1212  
-rw-r--r--. 1 student student 16 Jan 16 23:41 adjtime  
-rw-r--r--. 1 student student 1518 Sep 10 17:21 aliases  
drwxr-xr-x. 2 student student 169 Feb 4 21:58 alternatives  
...output omitted...
```

5. Sur **workstation**, retournez dans le répertoire personnel **student**.

```
[student@workstation savedconfig]$ cd
```

Évaluation

Sur **workstation**, exéutez le script **lab archive-review grade** pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab archive-review grade
```

Finish (Terminer)

Sur **workstation**, exéutez le script **lab archive-review finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab archive-review finish
```

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- La commande **tar** crée un fichier d'archive à partir d'un ensemble de fichiers et de répertoires, extrait des fichiers de l'archive et liste le contenu d'une archive.
- La commande **tar** offre un ensemble de méthodes de compression permettant de réduire la taille des archives.
- En plus de fournir un shell distant sécurisé, le service SSH fournit également les commandes **scp** et **sftp** en tant que moyens sécurisés pour transférer des fichiers depuis et vers un système distant qui exécute le serveur SSH.
- La commande **rsync** synchronise les fichiers de manière sécurisée et efficace entre deux répertoires, l'un ou l'autre pouvant se trouver sur un système distant.

CHAPITRE 14

INSTALLATION ET MISE À JOUR DE PAQUETAGES LOGICIELS

PROJET

Télécharger, installer, mettre à jour et gérer les paquetages logiciels depuis les dépôts de paquetages Red Hat et Yum.

OBJECTIFS

- Enregistrer un système sur votre compte Red Hat et lui attribuer les droits pour les mises à jour logicielles et les services de support en utilisant Red Hat Subscription Management.
- Expliquer comment les logiciels sont fournis sous forme de paquetages RPM et inspecter les paquetages installés sur le système avec Yum et RPM.
- Trouver, installer et mettre à jour des paquetages logiciels à l'aide de la commande **yum**.
- Activer et désactiver l'utilisation de référentiels Yum tiers ou Red Hat par un serveur.
- Expliquer comment les modules permettent l'installation de versions spécifiques de logiciels, lister, activer et permuter les flux de modules, et installer et mettre à jour les paquetages à partir d'un module.

SECTIONS

- Enregistrement de systèmes pour le support Red Hat (avec quiz)
- Explication et analyse des paquetages logiciels RPM (avec quiz)
- Installation et mise à jour de paquetages logiciels avec Ym (avec exercice guidé)
- Activation des référentiels logiciels Yum (avec exercice guidé)
- Gestion des flux de modules de paquetages (avec exercice guidé)

ATELIER

Installation et mise à jour de paquetages logiciels

ENREGISTREMENT DE SYSTÈMES POUR LE SUPPORT RED HAT

OBJECTIFS

Au terme de cette section, vous serez en mesure d'enregistrer un système sur votre compte Red Hat et lui attribuer les droits pour les mises à jour logicielles et les services de support en utilisant Red Hat Subscription Management.

RED HAT SUBSCRIPTION MANAGEMENT

Red Hat Subscription Management fournit des outils qui peuvent être utilisés pour accorder aux machines des abonnements à des produits. Cela permet aux administrateurs de recevoir les mises à jour de paquetages logiciels, et de suivre les informations relatives aux contrats d'assistance et aux abonnements utilisés par les systèmes. Des outils standard comme PackageKit et **yum** peuvent obtenir des paquetages logiciels et des mises à jour par l'intermédiaire d'un réseau de distribution de contenu mis en place par Red Hat.

Les outils de Red Hat Subscription Management accomplissent quatre tâches de base :

- **Enregistrer** un système pour l'associer à un compte Red Hat. Cela permet au gestionnaire des abonnements (Subscription Manager) d'inventorier ce système de façon unique. L'enregistrement d'un système peut être annulé lorsque celui-ci n'est plus utilisé.
- **Abonner** un système pour lui accorder des mises à jour de produits Red Hat spécifiques. Les abonnements sont liés à des niveaux d'assistance, dates d'expiration et référentiels par défaut spécifiques. Les outils peuvent être utilisés pour joindre automatiquement un droit spécifique ou pour le sélectionner. Les abonnements peuvent être supprimés en fonction de l'évolution des besoins.
- **Activer les référentiels** pour qu'ils fournissent des paquetages logiciels. Par défaut, plusieurs référentiels sont activés avec chaque abonnement, mais d'autres référentiels, tels que des mises à jour ou du code source, peuvent être activés ou désactivés suivant les besoins.
- **Examiner et effectuer le suivi** des droits disponibles ou utilisés. On peut examiner les informations d'abonnement localement sur un système spécifique ou, pour un compte, soit sur la page Subscriptions du Portail des clients Red Hat, soit dans le *gestionnaire des ressources d'abonnement (SAM, Subscription Asset Manager)*.

Enregistrement d'un système

Vous pouvez enregistrer un système avec le portail client de Red Hat de différentes manières, c'est-à-dire avec une interface graphique à laquelle vous pouvez accéder avec une application GNOME ou via le service de console Web et avec un outil de ligne de commande.

Pour enregistrer un système avec l'application GNOME, lancez Red Hat Subscription Manager en sélectionnant Activities. Tapez *subscription* dans le champ Type to search... et cliquez sur Red Hat Subscription Manager. Saisissez le mot de passe approprié lorsque vous êtes invité à vous authentifier. Cela affiche la fenêtre Subscriptions suivante :

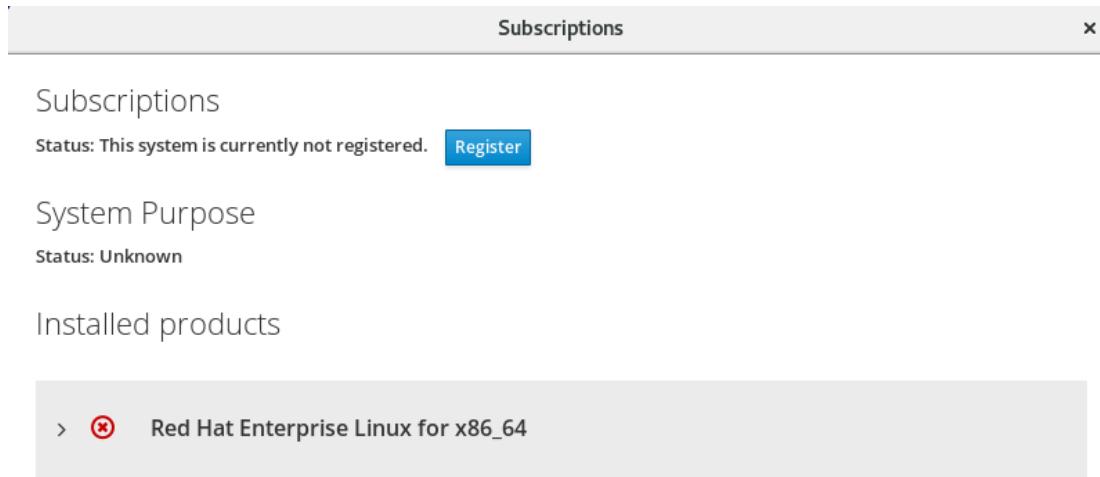


Figure 14.1: Fenêtre principale de Red Hat Subscription Manager

Pour enregistrer le système, cliquez sur le bouton Register dans la fenêtre Subscriptions. Cela affiche la boîte de dialogue suivante :

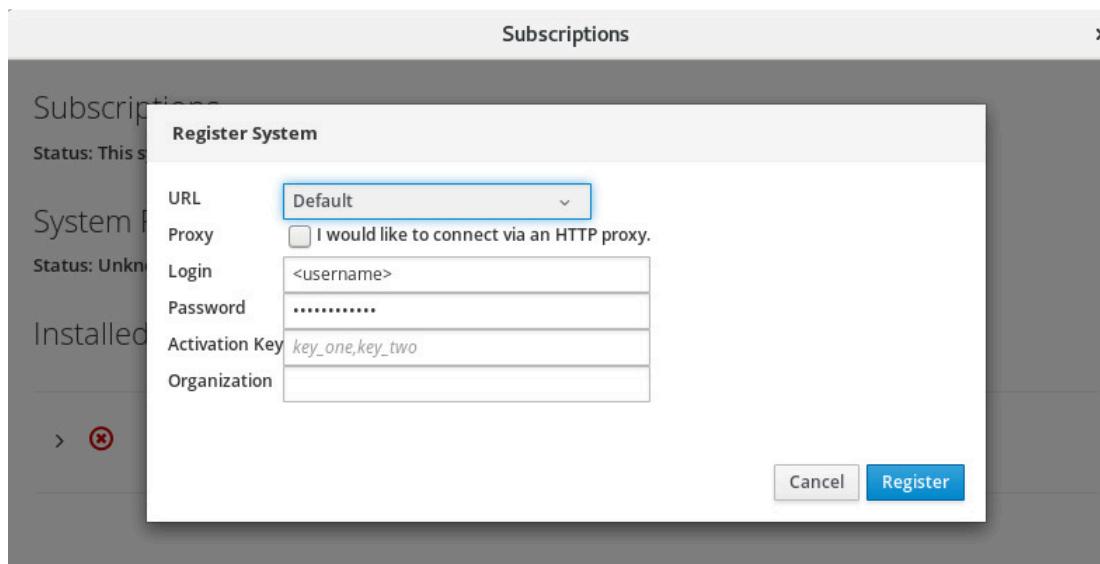


Figure 14.2: Boîte de dialogue d'enregistrement du système de Red Hat Subscription Manager

Cette boîte de dialogue enregistre un système auprès d'un serveur d'abonnement. Par défaut, il enregistre le serveur sur le portail client de Red Hat. Complétez les champs Login et Password pour le compte du portail client de Red Hat sur lequel le système doit être enregistré, puis cliquez sur le bouton Register.

Une fois enregistré, le système est automatiquement lié à un abonnement, le cas échéant.

Après l'enregistrement du système et une fois qu'un abonnement a été affecté, fermez la fenêtre Subscriptions. Le système est désormais correctement abonné et prêt à recevoir des mises à jour ou à installer de nouveaux logiciels de Red Hat.

ENREGISTREMENT À PARTIR DE LA LIGNE DE COMMANDE

Utilisez **subscription-manager**(8) pour enregistrer un système sans passer par un environnement graphique. La commande **subscription-manager** peut lier automatiquement un système aux abonnements les plus appropriés.

- Enregistrer un système sur un compte Red Hat :

```
[user@host ~]$ subscription-manager register --username=yourusername \
--password=yourpassword
```

- Afficher les abonnements disponibles :

```
[user@host ~]$ subscription-manager list --available | less
```

- Lier automatiquement un abonnement :

```
[user@host ~]$ subscription-manager attach --auto
```

- Vous pouvez également lier un abonnement à partir d'un pool spécifique de la liste des abonnements disponibles :

```
[user@host ~]$ subscription-manager attach --pool=poolID
```

- Afficher les abonnements utilisés :

```
[user@host ~]$ subscription-manager list --consumed
```

- Annuler l'enregistrement d'un système :

```
[user@host ~]$ subscription-manager unregister
```



NOTE

On peut aussi utiliser **subscription-manager** conjointement avec des clés d'activation, ce qui permet l'enregistrement et l'attribution d'abonnements prédefinis, sans utiliser ni nom d'utilisateur, ni mot de passe. Cette méthode d'enregistrement peut s'avérer très utile pour les installations et déploiements automatisés. Les clés d'activation sont souvent émises par un service de gestion des abonnements sur site, comme Subscription Asset Manager ou Red Hat Satellite. Elles ne font pas l'objet d'un traitement détaillé dans ce cours.

CERTIFICATS DE DROITS

Un droit est un abonnement qui a été lié à un système. On utilise des certificats numériques pour stocker sur le système local les informations actuelles relatives aux droits. Une fois enregistrés, les certificats de droits sont stockés dans **/etc/pki** et ses sous-répertoires.

- **/etc/pki/product** contient des certificats indiquant les produits Red Hat installés sur le système.

- **/etc/pki/consumer** contient des certificats identifiant le compte Red Hat sur lequel le système est enregistré.
- **/etc/pki/entitlement** contient les certificats indiquant quels abonnements sont joints au système.

On peut inspecter les certificats directement avec l'utilitaire **rct**, mais les outils **subscription-manager** offrent des façons plus faciles d'examiner les abonnements joints au système.



RÉFÉRENCES

Pages de manuel **subscription-manager(8)** et **rct(8)**

Premier pas avec Red Hat Subscription Management

<https://access.redhat.com/site/articles/433903>

► QUIZ

ENREGISTREMENT DE SYSTÈMES POUR LE SUPPORT RED HAT

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. **Quelle commande permet d'enregistrer un système sans passer par un environnement graphique ?**
 - a. **rct**
 - b. **subscription-manager**
 - c. **rpm**
 - d. **yum**
- ▶ 2. **Quel outil graphique est utilisé pour enregistrer et abonner un système ?**
 - a. PackageKit
 - b. **gpk-application**
 - c. Red Hat Subscription Manager
 - d. **gnome-software**
- ▶ 3. **Quelles tâches peuvent être accomplies avec les outils de Red Hat Subscription Management ?**
 - a. Enregistrer un système.
 - b. Abonner un système.
 - c. Activer des référentiels.
 - d. Examiner et suivre des droits.
 - e. Toutes les affirmations précédentes.

► SOLUTION

ENREGISTREMENT DE SYSTÈMES POUR LE SUPPORT RED HAT

Répondez aux questions suivantes en sélectionnant une réponse :

- ▶ 1. **Quelle commande permet d'enregistrer un système sans passer par un environnement graphique ?**
 - a. `rct`
 - b. `subscription-manager`
 - c. `rpm`
 - d. `yum`
- ▶ 2. **Quel outil graphique est utilisé pour enregistrer et abonner un système ?**
 - a. PackageKit
 - b. `gpk-application`
 - c. Red Hat Subscription Manager
 - d. `gnome-software`
- ▶ 3. **Quelles tâches peuvent être accomplies avec les outils de Red Hat Subscription Management ?**
 - a. Enregistrer un système.
 - b. Abonner un système.
 - c. Activer des référentiels.
 - d. Examiner et suivre des droits.
 - e. Toutes les affirmations précédentes.

EXPLICATION ET ANALYSE DES PAQUETAGES LOGICIELS RPM

OBJECTIFS

Au terme de cette section, vous serez en mesure d'expliquer comment les logiciels sont fournis sous forme de paquetages RPM et d'inspecter les paquetages installés sur le système avec Yum et RPM.

PAQUETAGES LOGICIELS ET RPM

RPM Package Manager, initialement développé par Red Hat, fournit une méthode standard pour empaqueter le logiciel en vue de sa distribution. Il est bien plus simple de gérer des logiciels sous la forme de *paquetages RPM* que d'utiliser des logiciels qui ont été simplement extraits d'une archive et intégrés à un système de fichiers. Cela permet aux administrateurs de savoir quels fichiers ont été installés par le paquetage logiciel et quels fichiers doivent être supprimés si le logiciel est désinstallé, et de vérifier que les paquetages complémentaires sont présents lorsque le logiciel est installé. Les informations relatives aux paquetages installés sont stockées dans une base de données RPM locale sur chaque système. Tous les logiciels fournis par Red Hat pour Red Hat Enterprise Linux prennent la forme de paquetages RPM.

Les noms de fichiers des paquetages RPM se composent de quatre éléments (en plus du suffixe **.rpm**) : **name-version-release.architecture** :

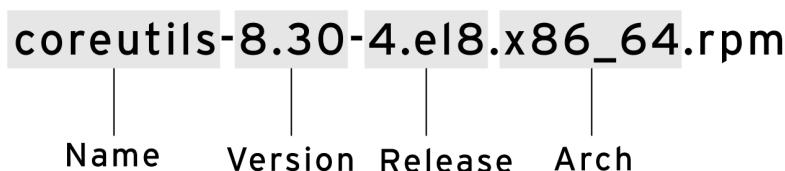


Figure 14.3: Éléments de nom de fichier RPM

- NAME correspond à un ou plusieurs mots qui décrivent le contenu (coreutils).
- VERSION correspond au numéro de version du logiciel d'origine (8.30).
- RELEASE correspond au numéro d'édition du paquetage basé sur cette version. L'édition est définie par le responsable du paquetage, qui peut ne pas être le développeur d'origine du logiciel (4.el8).
- ARCH correspond à l'architecture de processeur pour laquelle le paquetage a été compilé. **noarch** indique que le contenu du paquetage n'est pas spécifique à une architecture (contrairement à **x86_64** pour 64 bits, **aarch64** pour ARM 64 bits, et ainsi de suite).

Seul le nom du paquetage est requis pour l'installation de paquetages à partir de référentiels. S'il existe plusieurs versions, le paquetage portant le numéro de version le plus élevé est installé. S'il existe plusieurs éditions d'une seule version, le paquetage portant le numéro d'édition le plus élevé est installé.

Chaque paquetage RPM est une archive spéciale qui comporte trois composants :

- Les fichiers installés par le paquetage.

CHAPITRE 14 | Installation et mise à jour de paquetages logiciels

- Les informations relatives au paquetage (métadonnées), comme le nom, la version, l'édition et l'architecture ; un résumé et une description du paquetage ; s'il requiert ou non l'installation d'autres paquetages ; les licences ; le journal des modifications du paquetage et d'autres détails.
- Les scripts qui peuvent s'exécuter lorsque le paquetage est installé, mis à jour ou supprimé, ou qui sont déclenchés lors de l'installation, la mise à jour ou la suppression d'autres paquetages.

Généralement, les fournisseurs de logiciels signent numériquement les paquetages RPM à l'aide de clés GPG (Red Hat signe numériquement tous les paquetages qu'il publie). Le système RPM vérifie l'intégrité du paquetage en vérifiant que celui-ci a été signé par la clé GPG appropriée. Le système RPM refuse d'installer un paquetage si la signature GPG ne correspond pas.

Mise à jour des logiciels avec paquetages RPM

Red Hat génère un paquetage RPM complet pour mettre à jour les logiciels. Un administrateur qui installe ce paquetage ne récupère que la version la plus récente du paquetage. Red Hat n'exige pas que les anciens paquets soient installés, puis corrigés. Pour mettre à jour un logiciel, RPM supprime l'ancienne version du paquetage et installe la nouvelle version. Les mises à jour conservent généralement les fichiers de configuration, mais l'outil de création de paquetages de la nouvelle version définit le comportement exact.

La plupart du temps, on ne peut installer qu'une seule version ou édition d'un paquetage à la fois. Toutefois, si un paquetage est créé de façon à éviter les conflits de noms, plusieurs versions peuvent être installées. L'exemple le plus important de ceci est le paquetage **kernel**. Puisque l'on ne peut tester un nouveau noyau qu'en démarrant avec, le paquetage est spécialement conçu de façon à pouvoir installer plusieurs versions à la fois. Si le nouveau noyau échoue au démarrage, l'ancien sera toujours disponible et amorçable.

EXAMEN DES PAQUETAGES RPM

L'utilitaire **rpm** est un outil de bas niveau permettant d'obtenir des informations sur le contenu des fichiers de paquetage et sur les paquetages installés. Par défaut, il obtient des informations de la base de données locale des paquetages installés. Cependant, vous pouvez utiliser l'option **-p** pour indiquer que vous souhaitez obtenir des informations sur un fichier de paquetage téléchargé. Vous pourrez ainsi, si vous le souhaitez, inspecter le contenu du fichier de paquetage avant de l'installer.

La forme générale d'une requête est la suivante :

- **rpm -q [select-options] [query-options]**

Requêtes RPM : informations générales sur les paquetages installés

- **rpm -qa** : affiche la liste de tous les paquetages installés
- **rpm -qf FILENAME** : trouve les paquetages qui fournit NOMFICHIER

```
[user@host ~]$ rpm -qf /etc/yum.repos.d  
redhat-release-8.0-0.39.el8.x86_64
```

Requêtes RPM : informations sur des paquetages spécifiques

- **rpm -q** : indique la version du paquetage actuellement installée

```
[user@host ~]$ rpm -q yum  
yum-4.0.9.2-4.el8.noarch
```

CHAPITRE 14 | Installation et mise à jour de paquetages logiciels

- **rpm -qi**: donne des informations détaillées sur le paquetage
- **rpm -ql**: affiche la liste des fichiers installés par le paquetage

```
[user@host ~]$ rpm -ql yum
/etc/yum.conf
/etc/yum/pluginconf.d
/etc/yum/protected.d
/etc/yum/vars
/usr/bin/yum
/usr/share/man/man1/yum-aliases.1.gz
/usr/share/man/man5/yum.conf.5.gz
/usr/share/man/man8/yum-shell.8.gz
/usr/share/man/man8/yum.8.gz
```

- **rpm -qc**: affiche simplement la liste des fichiers de configuration installés par le paquetage

```
[user@host ~]$ rpm -qc openssh-clients
/etc/ssh/ssh_config
/etc/ssh/ssh_config.d/05-redhat.conf
```

- **rpm -qd**: affiche simplement la liste des fichiers de documentation installés par le paquetage

```
[user@host ~]$ rpm -qd openssh-clients
/usr/share/man/man1/scp.1.gz
/usr/share/man/man1/sftp.1.gz
/usr/share/man/man1/ssh-add.1.gz
/usr/share/man/man1/ssh-agent.1.gz
/usr/share/man/man1/ssh-copy-id.1.gz
/usr/share/man/man1/ssh-keyscan.1.gz
/usr/share/man/man1/ssh.1.gz
/usr/share/man/man5/ssh_config.5.gz
/usr/share/man/man8/ssh-pkcs11-helper.8.gz
```

- **rpm -q --scripts**: affiche la liste des scripts shell exécutés avant ou après l'installation ou la suppression du paquetage

```
[user@host ~]$ rpm -q --scripts openssh-server
preinstall scriptlet (using /bin/sh):
getent group sshd >/dev/null || groupadd -g 74 -r sshd || :
getent passwd sshd >/dev/null || \
    useradd -c "Privilege-separated SSH" -u 74 -g sshd \
    -s /sbin/nologin -r -d /var/empty/sshd sshd 2>/dev/null || :
postinstall scriptlet (using /bin/sh):

if [ $1 -eq 1 ] ; then
    # Initial installation
    /usr/bin/systemctl preset sshd.service sshd.socket >/dev/null 2>&1 || :
fi
preuninstall scriptlet (using /bin/sh):

if [ $1 -eq 0 ] ; then
    # Package removal, not upgrade
```

CHAPITRE 14 | Installation et mise à jour de paquetages logiciels

```
/usr/bin/systemctl --no-reload disable sshd.service sshd.socket > /dev/null 2>&1 || :  
    /usr/bin/systemctl stop sshd.service sshd.socket > /dev/null 2>&1 || :  
fi  
postuninstall scriptlet (using /bin/sh):  
  
/usr/bin/systemctl daemon-reload >/dev/null 2>&1 || :  
if [ $1 -ge 1 ] ; then  
    # Package upgrade, not uninstall  
    /usr/bin/systemctl try-restart sshd.service >/dev/null 2>&1 || :  
fi
```

- **rpm -q --changelog**: répertorie les informations de modification pour le paquetage

```
[user@host ~]$ rpm -q --changelog audit  
* Wed Jan 09 2019 Steve Grubb <sgrubb@redhat.com> 3.0-0.10.20180831git0047a6c  
resolves: rhbz#1655270] Message "audit: backlog limit exceeded" reported  
- Fix annobin failure  
  
* Fri Dec 07 2018 Steve Grubb <sgrubb@redhat.com> 3.0-0.8.20180831git0047a6c  
resolves: rhbz#1639745 - build requires go-toolset-7 which is not available  
resolves: rhbz#1643567 - service audidd stop exits prematurely  
resolves: rhbz#1616428 - Update git snapshot of audit package  
- Remove static libs subpackage  
...output omitted...
```

Interrogation de fichiers de paquetage locaux :

```
[user@host ~]$ ls -l wonderwidgets-1.0-4.x86_64.rpm  
-rw-rw-r-- 1 user user 257 Mar 13 20:06 wonderwidgets-1.0-4.x86_64.rpm  
[user@host ~]$ rpm -qlp wonderwidgets-1.0-4.x86_64.rpm  
/etc/wonderwidgets.conf  
/usr/bin/wonderwidgets  
/usr/share/doc/wonderwidgets-1.0  
/usr/share/doc/wonderwidgets-1.0/README.txt
```

INSTALLATION DES PAQUETAGES RPM

La commande **rpm** peut également être utilisée pour installer un paquetage RPM que vous avez téléchargé dans votre répertoire local.

```
[root@host ~]# rpm -ivh wonderwidgets-1.0-4.x86_64.rpm  
Verifying... ##### [100%]  
Preparing... ##### [100%]  
Updating / installing...  
 1:wonderwidgets-1.0-4 ##### [100%]  
[root@host ~]#
```

Cependant, la prochaine section de ce chapitre abordera un outil plus puissant pour gérer l'installation de RPM et les mises à jour à partir de la ligne de commande **yum**.

**MISE EN GARDE**

Soyez prudent en installant des paquetages tiers, non seulement en raison des logiciels qu'ils peuvent installer, mais aussi car le paquetage RPM peut inclure des scripts arbitraires s'exécutant en tant qu'utilisateur `root` dans le cadre du processus d'installation.

**NOTE**

Vous pouvez extraire des fichiers d'un fichier de paquetage RPM sans installer le paquetage. L'utilitaire `rpm2cpio` peut transmettre le contenu du RPM à un outil d'archivage spécial appelé `cpio`, qui peut extraire tous les fichiers ou des fichiers individuels.

Acheminez la sortie de `rpm2cpio PACKAGEFILE.rpm` vers `cpio -id` pour extraire tous les fichiers stockés dans le paquetage RPM. Des arborescences de sous-répertoires sont créées si nécessaire à partir du répertoire de travail actuel.

```
[user@host tmp-extract]$ rpm2cpio wonderwidgets-1.0-4.x86_64.rpm | cpio -id
```

Il est possible d'extraire des fichiers individuels en indiquant leur chemin d'accès :

```
[user@host ~]$ rpm2cpio wonderwidgets-1.0-4.x86_64.rpm | cpio -id "*txt"
11 blocks
[user@host ~]$ ls -l usr/share/doc/wonderwidgets-1.0/
total 4
-rw-r--r--. 1 user user 76 Feb 13 19:27 README.txt
```

RÉSUMÉ DES COMMANDES DE REQUÊTE RPM

Les paquetages installés peuvent être demandés directement avec la commande `rpm`. Ajoutez l'option `-p` pour demander un fichier de paquetage avant son installation.

COMMANDÉ	TÂCHE
<code>rpm -qa</code>	Afficher la liste de tous les paquetages RPM actuellement installés
<code>rpm -q NAME</code>	Afficher la version de NAME installée sur le système
<code>rpm -qi NAME</code>	Afficher des informations détaillées sur un paquetage
<code>rpm -ql NAME</code>	Afficher la liste de tous les fichiers d'un paquetage
<code>rpm -qc NAME</code>	Afficher la liste des fichiers de configuration d'un paquetage
<code>rpm -qd NAME</code>	Afficher la liste des fichiers de documentation d'un paquetage

COMMANDÉ	TÂCHE
rpm -q --changelog NAME	Afficher un bref résumé des raisons de la création d'une nouvelle version de paquetage
rpm -q --scripts NAME	Afficher les scripts shell exécutés lors de l'installation, de la mise à niveau ou de la suppression d'un paquetage



RÉFÉRENCES

Pages de manuel **rpm(8)**, **rpm2cpio(8)**, **cpio(1)** et **rpmkeys(8)**

► EXERCICE GUIDÉ

EXPLICATION ET ANALYSE DES PAQUETAGES LOGICIELS RPM

Au cours de cet exercice, vous allez collecter des informations sur un paquetage provenant d'une tierce partie, en extraire des fichiers à des fins d'inspection, puis l'installer sur un serveur.

RÉSULTATS

Vous serez en mesure d'installer un paquetage non fourni par les référentiels de logiciels sur un serveur.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab software-rpm start**. Le script exécute un script de démarrage qui détermine si l'hôte, servera, est accessible sur le réseau. Le script télécharge également le paquetage *rhcsa-script-1.0.0-1.noarch.rpm* dans le répertoire **/home/student** sur servera.

```
[student@workstation ~]$ lab software-rpm start
```

- 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Affichez les informations sur les paquetages et listez les fichiers du paquetage *rhcsa-script-1.0.0-1.noarch.rpm*. Affichez également le script qui s'exécute lorsque le paquetage est installé ou désinstallé.

- 2.1. Affichez les informations du paquetage *rhcsa-script-1.0.0-1.noarch.rpm*.

```
[student@servera ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm -i
Name      : rhcsa-script
Version   : 1.0.0
Release   : 1
Architecture: noarch
Install Date: (not installed)
Group     : System
Size      : 1056
License   : GPL
Signature : (none)
Source RPM : rhcsa-script-1.0.0-1.src.rpm
```

CHAPITRE 14 | Installation et mise à jour de paquetages logiciels

```
Build Date : Wed 06 Mar 2019 03:59:46 PM IST
Build Host : foundation0.ilt.example.com
Relocations : (not relocatable)
Packager : Snehangshu Karmakar
URL : http://example.com
Summary : RHCSA Practice Script
Description :
A RHCSA practice script.
The package changes the motd.
```

2.2. Listez les fichiers du paquetage *rhcsa-script-1.0.0-1.noarch.rpm*.

```
[student@servera ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm -l
/opt/rhcsa-script/mymotd
```

2.3. Affichez le script qui s'exécute lorsque le paquetage *rhcsa-script-1.0.0-1.noarch.rpm* est installé ou désinstallé.

```
[student@servera ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm --scripts
preinstall scriptlet (using /bin/sh):
if [ "$1" == "2" ]; then
    if [ -e /etc/motd.orig ]; then
        mv -f /etc/motd.orig /etc/motd
    fi
fi
postinstall scriptlet (using /bin/sh):
...output omitted...
```

► 3. Extrayez le contenu du paquetage *rhcsa-script-1.0.0-1.noarch.rpm* dans le répertoire **/home/student**.

3.1. Utilisez les commandes **rpm2cpio** et **cpio -tv** pour lister les fichiers du paquetage *rhcsa-script-1.0.0-1.noarch.rpm*.

```
[student@servera ~]$ rpm2cpio rhcsa-script-1.0.0-1.noarch.rpm | cpio -tv
-rw-r--r-- 1 root      root     1056 Mar  6 15:59 ./opt/rhcsa-script/mymotd
3 blocks
```

3.2. Extrayez tous les fichiers du paquetage *rhcsa-script-1.0.0-1.noarch.rpm* dans le répertoire **/home/student**. Utilisez les commandes **rpm2cpio** et **cpio -idv** permettant d'extraire les fichiers et de créer les principaux répertoires nécessaires en mode détaillé.

```
[student@servera ~]$ rpm2cpio rhcsa-script-1.0.0-1.noarch.rpm | cpio -idv
./opt/rhcsa-script/mymotd
3 blocks
```

3.3. Créez une liste pour vérifier les fichiers extraits dans le répertoire **/home/student/opt**.

```
[student@servera ~]$ ls -lR opt
opt:
total 0
drwxrwxr-x. 2 student student 20 Mar  7 14:44 rhcsa-script

opt/rhcsa-script:
total 4
-rw-r--r--. 1 student student 1056 Mar  7 14:44 mymotd
```

- 4. Installez le paquetage *rhcsa-script-1.0.0-1.noarch.rpm*. Utilisez la commande **sudo** pour obtenir des priviléges de superutilisateur permettant d'installer le paquetage.

- 4.1. Utilisez la commande **sudo rpm -ivh** pour installer le paquetage RPM *rhcsa-script-1.0.0-1.noarch.rpm*.

```
[student@servera ~]$ sudo rpm -ivh rhcsa-script-1.0.0-1.noarch.rpm
[sudo] password for student: student
Verifying...                                         #####[100%]
Preparing...                                         #####[100%]
Updating / installing...
 1:rhcsa-script-1.0.0-1                           #####[100%]
[student@servera ~]$
```

- 4.2. Utilisez la commande **rpm** pour vérifiez que le paquetage est installé.

```
[student@servera ~]$ rpm -q rhcsa-script
rhcsa-script-1.0.0-1.noarch
```

- 5. Quittez servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Fin

Sur workstation, exécutez le script **lab software-rpm finish** pour mettre fin à l'exercice. Ce script supprime tous les paquetages installés sur servera pendant l'exercice.

```
[student@workstation ~]$ lab software-rpm finish
```

L'exercice guidé est maintenant terminé.

INSTALLATION ET MISE À JOUR DE PAQUETAGES LOGICIELS AVEC YUM

OBJECTIFS

Au terme de cette section, vous serez en mesure de rechercher, d'installer et de mettre à jour des paquetages logiciels à l'aide de la commande **yum**.

GESTION DE PAQUETAGES LOGICIELS AVEC YUM

La commande **rpm** de bas niveau peut être utilisée pour installer des paquetages, mais elle n'est pas conçue pour fonctionner avec des référentiels de paquetages ou pour résoudre automatiquement les dépendances de plusieurs sources.

Yum est conçu pour mieux gérer l'installation et les mises à jour des logiciels basés sur RPM. La commande **yum** vous permet d'installer, de mettre à jour, de supprimer et d'obtenir des informations sur les paquetages de logiciels et leurs dépendances. Vous pouvez obtenir un historique des transactions effectuées et travailler avec plusieurs référentiels de logiciels Red Hat et tiers.

Recherche de logiciel avec Yum

- **yum help** affiche des informations d'utilisation.
- **yum list** affiche les paquetages installés et disponibles.

```
[user@host ~]$ yum list 'http*'
Available Packages
http-parser.i686          2.8.0-2.el8           rhel8-appstream
http-parser.x86_64          2.8.0-2.el8           rhel8-appstream
httpcomponents-client.noarch 4.5.5-4.module+e18+2452+b359bfcd rhel8-appstream
httpcomponents-core.noarch   4.4.10-3.module+e18+2452+b359bfcd rhel8-appstream
httpd.x86_64                2.4.37-7.module+e18+2443+605475b7 rhel8-appstream
httpd-devel.x86_64          2.4.37-7.module+e18+2443+605475b7 rhel8-appstream
httpd-filesystem.noarch     2.4.37-7.module+e18+2443+605475b7 rhel8-appstream
httpd-manual.noarch         2.4.37-7.module+e18+2443+605475b7 rhel8-appstream
httpd-tools.x86_64          2.4.37-7.module+e18+2443+605475b7 rhel8-appstream
```

- **yum search KEYWORD** répertorie les paquetages en fonction des mots-clés trouvés dans les champs de nom et de résumé uniquement.

Pour rechercher les paquetages dont les champs de nom, de résumé et de description contiennent « serveur Web », utilisez **search all**:

```
[user@host ~]$ yum search all 'web server'
=====
Summary & Description Matched: web server =====
pcp-pmda-weblog.x86_64 : Performance Co-Pilot (PCP) metrics from web server logs
nginx.x86_64 : A high performance web server and reverse proxy server
=====
Summary Matched: web server =====
libcurl.x86_64 : A library for getting files from web servers
libcurl.i686 : A library for getting files from web servers
```

CHAPITRE 14 | Installation et mise à jour de paquetages logiciels

```
libcurl.x86_64 : A library for getting files from web servers
=====
===== Description Matched: web server =====
httpd.x86_64 : Apache HTTP Server
git-instaweb.x86_64 : Repository browser in gitweb
...output omitted...
```

- **yum info PACKAGE NAME** renvoie des informations détaillées sur un paquetage, y compris l'espace disque requis pour l'installation.

Pour obtenir des informations sur le serveur HTTP Apache :

```
[user@host ~]$ yum info httpd
Available Packages
Name        : httpd
Version     : 2.4.37
Release     : 7.module+el8+2443+605475b7
Arch        : x86_64
Size        : 1.4 M
Source      : httpd-2.4.37-7.module+el8+2443+605475b7.src.rpm
Repo        : rhel8-appstream
Summary     : Apache HTTP Server
URL         : https://httpd.apache.org/
License     : ASL 2.0
Description : The Apache HTTP Server is a powerful, efficient, and extensible
              : web server.
```

- **yum provides PATHNAME** affiche les paquetages qui correspondent au nom de chemin spécifié (qui comprend souvent des caractères génériques).

Pour rechercher les paquetages qui fournissent le répertoire **/var/www/html**, utilisez :

```
[user@host ~]$ yum provides /var/www/html
httpd-filesystem-2.4.37-7.module+el8+2443+605475b7.noarch : The basic directory
  layout for the Apache HTTP server
Repo        : rhel8-appstream
Matched from:
Filename   : /var/www/html
```

Installation et suppression de logiciels avec yum

- **yum install PACKAGE NAME** obtient et installe un paquetage logiciel, y compris toutes ses dépendances.

```
[user@host ~]$ yum install httpd
Dependencies resolved.
=====
=====
Package          Arch    Version       Repository      Size
=====
Installing:
  httpd           x86_64  2.4.37-7.module...  rhel8-appstream  1.4 M
Installing dependencies:
  apr             x86_64  1.6.3-8.el8       rhel8-appstream  125 k
  apr-util        x86_64  1.6.1-6.el8       rhel8-appstream  105 k
...output omitted...
```

```
Transaction Summary
=====
Install 9 Packages

Total download size: 2.0 M
Installed size: 5.4 M
Is this ok [y/N]: y
Downloading Packages:
(1/9): apr-util-bdb-1.6.1-6.el8.x86_64.rpm           464 kB/s | 25 kB   00:00
(2/9): apr-1.6.3-8.el8.x86_64.rpm                     1.9 MB/s | 125 kB  00:00
(3/9): apr-util-1.6.1-6.el8.x86_64.rpm               1.3 MB/s | 105 kB  00:00
...output omitted...
Total                                         8.6 MB/s | 2.0 MB  00:00

Running transaction check
Transaction check succeeded.

Running transaction test
Transaction test succeeded.

Running transaction
  Preparing          : 1/1
  Installing        : apr-1.6.3-8.el8.x86_64             1/9
  Running scriptlet: apr-1.6.3-8.el8.x86_64             1/9
  Installing        : apr-util-bdb-1.6.1-6.el8.x86_64  2/9
...output omitted...
Installed:
  httpd-2.4.37-7.module+el8+2443+605475b7.x86_64 apr-util-bdb-1.6.1-6.el8.x86_64
  apr-util-openssl-1.6.1-6.el8.x86_64                apr-1.6.3-8.el8.x86_64
...output omitted...
Complete!
```

- **yum update PACKAGE_NAME** obtient et installe une version plus récente du paquetage spécifié, y compris toutes les dépendances. En règle générale, le processus essaie de conserver les fichiers de configuration en place mais, dans certains cas, ils doivent être renommés si le responsable du paquetage pense que l'ancienne version ne fonctionnera pas après la mise à jour. Si NOMPAAUETAGE n'est pas spécifié, toutes les mises à jour pertinentes sont installées.

```
[user@host ~]$ sudo yum update
```

Puisque l'on ne peut tester un nouveau noyau qu'en démarrant avec, le paquetage est spécialement conçu de façon à pouvoir installer plusieurs versions à la fois. Si le nouveau noyau échoue au démarrage, l'ancien sera toujours disponible. L'utilisation de **yum update kernel** permettra de véritablement *installer* le nouveau noyau. Les fichiers de configuration contiennent une liste de paquetages à *toujours installer*, même si l'administrateur demande une mise à jour.

**NOTE**

Utilisez **yum list kernel** pour répertorier tous les noyaux installés et disponibles. Pour afficher le noyau en cours d'exécution, utilisez la commande **uname**. L'option **-r** n'indique que la version et l'édition du noyau, et l'option **-a** affiche la version du noyau et des informations supplémentaires.

```
[user@host ~]$ yum list kernel
Installed Packages
kernel.x86_64          4.18.0-60.el8      @anaconda
kernel.x86_64          4.18.0-67.el8      @rhel-8-for-x86_64-baseos-htb-rpms
[user@host ~]$ uname -r
4.18.0-60.el8.x86_64
[user@host ~]$ uname -a
Linux host.lab.example.com 4.18.0-60.el8.x86_64 #1 SMP Fri Jan 11 19:08:11 UTC
2019 x86_64 x86_64 x86_64 GNU/Linux
```

- **yum remove PACKAGE_NAME** supprime un paquetage logiciel installé, y compris tous les paquetages pris en charge.

```
[user@host ~]$ sudo yum remove httpd
```

**MISE EN GARDE**

La commande **yum remove** supprime les paquetages listés, *ainsi que tous les paquetages qui nécessitent la suppression des paquetages* (et les paquetages qui nécessitent ces paquetages, et ainsi de suite). Cela peut entraîner la suppression inopinée de paquetages. Vous devez donc examiner attentivement la liste des paquetages à supprimer.

Installation et suppression de groupes de logiciels avec yum

- **yum** reconnaît également le concept de *groupes*, qui sont des ensembles de logiciels connexes installés ensemble dans un but spécifique. Dans Red Hat Enterprise Linux 8, il existe deux types de groupes. Les groupes réguliers sont des ensembles de paquetages. Les *groupes d'environnements* sont des ensembles de groupes réguliers. Les paquetages ou les groupes fournis par un groupe peuvent être **obligatoires** (ils doivent être installés avec le groupe), **par défaut** (ils sont normalement installés avec le groupe) ou **facultatifs** (ils ne sont pas installés avec le groupe, sauf demande spécifique).

Comme **yum list**, la commande **yum group list** affiche le nom des groupes installés et disponibles.

```
[user@host ~]$ yum group list
Available Environment Groups:
  Server with GUI
  Minimal Install
  Server
...output omitted...
Available Groups:
```

```
Container Management
.NET Core Development
RPM Development Tools
...output omitted...
```

Certains groupes sont normalement installés par l'intermédiaire de groupes d'environnements, et sont masqués par défaut. Listez ces groupes masqués avec la commande **yum group list hidden**.

- **yum group info** affiche des informations relatives à un groupe. Ces informations incluent une liste des noms de paquetages, par défaut et facultatifs.

```
[user@host ~]$ yum group info "RPM Development Tools"
Group: RPM Development Tools
Description: These tools include core development tools such rpmbuild.
Mandatory Packages:
    redhat-rpm-config
    rpm-build
Default Packages:
    rpmdevtools
Optional Packages:
    rpmlint
```

- **yum group install** installe un groupe qui installe ses paquetages obligatoires et par défaut, ainsi que les paquetages dont ils dépendent.

```
[user@host ~]$ sudo yum group install "RPM Development Tools"
...output omitted...
Installing Groups:
RPM Development Tools

Transaction Summary
=====
Install 64 Packages

Total download size: 21 M
Installed size: 62 M
Is this ok [y/N]: y
...output omitted...
```

**IMPORTANT**

Le comportement des groupes Yum a changé à partir de Red Hat Enterprise Linux 7. Dans RHEL 7 et les versions ultérieures, les groupes sont considérés comme des *objets* et sont suivis par le système. Si un groupe installé est mis à jour et que de nouveaux paquetages obligatoires ou par défaut ont été ajoutés au groupe par le référentiel Yum, ces nouveaux paquetages seront installés lors de la mise à jour.

RHEL 6 (ou version antérieure) considère qu'un groupe est installé si tous ses paquetages obligatoires ont été installés ou, s'il n'avait aucun paquetage obligatoire, si tous les paquetages par défaut ou facultatifs du groupe sont installés. À partir de RHEL 7, un groupe est considéré comme installé *uniquement* si **yum group install** a été utilisé pour le faire. La commande **yum group mark install GROUPNAME** peut être utilisée pour marquer un groupe comme installé, et tout paquetage manquant est installé lors de la prochaine mise à jour, ainsi que ses dépendances.

Enfin, RHEL 6 (ou version antérieure) ne disposait pas de la forme en deux mots des commandes **yum group**. En d'autres termes, dans RHEL 6, la commande **yum grouplist** existait, mais pas la commande **yum group list** RHEL 7 et RHEL 8 équivalente.

Affichage de l'historique des transactions

- Toutes les transactions d'installation et de suppression sont consignées dans **/var/log/dnf.rpm.log**.

```
[user@host ~]$ tail -5 /var/log/dnf.rpm.log
2019-02-26T18:27:00Z SUBDEBUG Installed: rpm-build-4.14.2-9.el8.x86_64
2019-02-26T18:27:01Z SUBDEBUG Installed: rpm-build-4.14.2-9.el8.x86_64
2019-02-26T18:27:01Z SUBDEBUG Installed: rpmdevtools-8.10-7.el8.noarch
2019-02-26T18:27:01Z SUBDEBUG Installed: rpmdevtools-8.10-7.el8.noarch
2019-02-26T18:38:40Z INFO --- logging initialized ---
```

- yum history** affiche un récapitulatif des transactions d'installation et de suppression.

ID	Command line	Date and time	Action(s)	Altered
7	group install RPM Develo	2019-02-26 13:26	Install	65
6	update kernel	2019-02-26 11:41	Install	4
5	install httpd	2019-02-25 14:31	Install	9
4	-y install @base firewal	2019-02-04 11:27	Install	127 EE
3	-C -y remove firewalld -	2019-01-16 13:12	Removed	11 EE
2	-C -y remove linux-firmw	2019-01-16 13:12	Removed	1
1		2019-01-16 13:05	Install	447 EE

- L'option **history undo** inverse une transaction.

```
[user@host ~]$ sudo yum history undo 5
Undoing transaction 7, from Tue 26 Feb 2019 10:40:32 AM EST
Install apr-1.6.3-8.el8.x86_64 @rhel8-appstream
Install apr-util-1.6.1-6.el8.x86_64 @rhel8-appstream
Install apr-util-bdb-1.6.1-6.el8.x86_64 @rhel8-appstream
Install apr-util-openssl-1.6.1-6.el8.x86_64 @rhel8-appstream
Install httpd-2.4.37-7.module+el8+2443+605475b7.x86_64 @rhel8-appstream
...output omitted...
```

RÉSUMÉ DES COMMANDES YUM

Vous pouvez localiser, installer, mettre à jour et supprimer des paquetages à partir de leur nom ou de groupes de paquetages.

TÂCHE :	COMMANDÉ :
Afficher la liste des paquetages installés et disponibles par leur nom	yum list [NAME-PATTERN]
Afficher la liste des groupes installés et disponibles	yum group list
Rechercher un paquetage par mot-clé	yum search KEYWORD
Afficher les détails d'un paquetage	yum info PACKAGE_NAME
Installer un paquetage	yum install PACKAGE_NAME
Installer un groupe de paquetages	yum group install GROUPNAME
Mettre à jour tous les paquetages	yum update
Supprimer un paquetage	yum remove PACKAGE_NAME
Afficher l'historique des transactions	yum history



RÉFÉRENCES

Pages de manuel **yum(1)** and **yum.conf(5)**

Pour plus d'informations, reportez-vous au chapitre *Installing software with yum* de *Red Hat Enterprise Linux 8.0 Configuring basic system settings* sur https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/installing-software-with-yum_configuring-basic-system-settings

► EXERCICE GUIDÉ

INSTALLATION ET MISE À JOUR DE PAQUETAGES LOGICIELS AVEC YUM

Au cours de cet exercice, vous allez installer et supprimer des paquetages et des groupes de paquetages.

RÉSULTATS

Vous serez en mesure d'installer et de supprimer des paquetages avec des dépendances.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur **student** à **workstation** avec le mot de passe **student**.

À partir de **workstation**, exécutez la commande **lab software-yum start**. La commande exécute un script de démarrage qui détermine si l'hôte, **servera**, est accessible sur le réseau.

```
[student@workstation ~]$ lab software-yum start
```

- ▶ 1. Utilisez la commande **ssh** pour vous connecter à **servera** en tant qu'utilisateur **student**. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis pour se connecter à **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. Utilisez la commande **su -** pour basculer vers l'utilisateur **root** à l'invite du shell.

```
[student@servera ~]$ su -
Password: redhat
[root@servera ~]#
```

- ▶ 3. Recherchez un paquetage spécifique.

- 3.1. Essayez d'exécuter la commande **guile**. Vous devriez constater qu'elle n'est pas installée.

```
[root@servera ~]# guile
-bash: guile: command not found
```

- 3.2. Utilisez la commande **yum search** pour rechercher des paquetages incluant **guile** dans leur nom ou résumé.

```
[root@servera ~]# yum search guile
=====
Name Exactly Matched: guile =====
guile.i686 : A GNU implementation of Scheme for application extensibility
guile.x86_64 : A GNU implementation of Scheme for application extensibility
```

3. Utilisez la commande **yum info** pour trouver plus d'informations sur le paquetage **guile**.

```
[root@servera ~]# yum info guile
Available Packages
Name        : guile
Epoch      : 5
Version    : 2.0.14
Release    : 7.el8
...output omitted...
```

- ▶ 4. Utilisez la commande **yum install** pour installer le paquetage **guile**.

```
[root@servera ~]# yum install guile
...output omitted...
Dependencies resolved.

=====
Package      Arch   Version       Repository      Size
=====
Installing:
guile        x86_64 5:2.0.14-7.el8   rhel-8.0-for-x86_64-appstream-rpms 3.5 M
Installing dependencies:
gc           x86_64 7.6.4-3.el8      rhel-8.0-for-x86_64-appstream-rpms 109 k
libatomic_ops x86_64 7.6.2-3.el8      rhel-8.0-for-x86_64-appstream-rpms 38 k
libtool-ltdl x86_64 2.4.6-25.el8     rhel-8.0-for-x86_64-baseos-rpms   58 k

Transaction Summary
=====
Install 4 Packages

Total download size: 3.7 M
Installed size: 12 M
Is this ok [y/N]: y
...output omitted...
Complete!
```

- ▶ 5. Supprimez des paquetages.

- 5.1. Utilisez la commande **yum remove** pour supprimer le paquetage **guile**, mais répondre **no** à l'invite. Combien de paquetages seraient supprimés ?

```
[root@servera ~]# yum remove guile
...output omitted...
Dependencies resolved.

=====
Package      Arch   Version       Repository      Size
=====
```

```
=====
Removing:
guile           x86_64 5:2.0.14-7.el8  @rhel-8.0-for-x86_64-appstream-rpms 12 M
Removing unused dependencies:
gc               x86_64 7.6.4-3.el8      @rhel-8.0-for-x86_64-appstream-rpms 221 k
libatomic_ops    x86_64 7.6.2-3.el8      @rhel-8.0-for-x86_64-appstream-rpms 75 k
libtool-ltdl    x86_64 2.4.6-25.el8     @rhel-8.0-for-x86_64-baseos-rpms   69 k

Transaction Summary
=====
Remove 4 Packages

Freed space: 12 M
Is this ok [y/N]: n
Operation aborted.
```

- 5.2. Utilisez la commande **yum remove** pour supprimer le paquetage **gc**, mais répondre **no** à l'invite. Combien de paquetages seraient supprimés ?

```
[root@servera ~]# yum remove gc
...output omitted...
Dependencies resolved.
=====
Package      Arch Version       Repository      Size
=====
Removing:
gc           x86_64 7.6.4-3.el8  @rhel-8.0-for-x86_64-appstream-rpms 221 k
Removing dependent packages:
guile        x86_64 5:2.0.14-7.el8  @rhel-8.0-for-x86_64-appstream-rpms 12 M
Removing unused dependencies:
libatomic_ops x86_64 7.6.2-3.el8      @rhel-8.0-for-x86_64-appstream-rpms 75 k
libtool-ltdl x86_64 2.4.6-25.el8     @rhel-8.0-for-x86_64-baseos-rpms   69 k

Transaction Summary
=====
Remove 4 Packages

Freed space: 12 M
Is this ok [y/N]: n
Operation aborted.
```

- 6. Recueillez des informations sur le groupe de composants « RPM Development Tools » et installez-le sur **servera**.

- 6.1. Utilisez la commande **yum group list** pour lister tous les groupes de composants disponibles.

```
[root@servera ~]# yum group list
```

- 6.2. Utilisez la commande **yum group info** pour rechercher plus d'informations sur le groupe de composants **RPM Development Tools**, y compris la liste des paquetages inclus.

```
[root@servera ~]# yum group info "RPM Development Tools"
Group: RPM Development Tools
Description: These tools include core development tools such rpmbuild.
Mandatory Packages:
  redhat-rpm-config
  rpm-build
Default Packages:
  rpmdevtools
Optional Packages:
  rpmlint
```

6.3. Utilisez la commande **yum group install** pour installer le groupe de composants RPM Development Tools.

```
[root@servera ~]# yum group install "RPM Development Tools"
Dependencies resolved.
=====
Package           Arch    Version        Repository      Size
=====
Installing group/module packages:
  redhat-rpm-config
                noarch  115-1.el8   rhel-8.0-for-x86_64-appstream-rpms 82 k
  rpm-build       x86_64  4.14.2-9.el8  rhel-8.0-for-x86_64-appstream-rpms 166 k
Installing dependencies:
  dwz             x86_64  0.12-9.el8   rhel-8.0-for-x86_64-appstream-rpm 109 k
  efi-srpm-macros noarch  3-2.el8     rhel-8.0-for-x86_64-appstream-rpm 22 k
...output omitted...

Transaction Summary
=====
Install  60 Packages

Total download size: 17 M
Installed size: 50 M
Is this ok [y/N]: y
...output omitted...
Installing      : perl-Exporter-5.72-396.el8.noarch          1/60
Installing      : perl-libs-4:5.26.3-416.el8.x86_64          2/60
Installing      : perl-Carp-1.42-396.el8.noarch          3/60
...output omitted...
Verifying       : dwz-0.12-9.el8.x86_64                  1/60
Verifying       : efi-srpm-macros-3-2.el8.noarch          2/60
Verifying       : gdb-headless-8.2-5.el8.x86_64          3/60
...output omitted...
Installed:
  redhat-rpm-config-115-1.el8.noarch
  rpm-build-4.14.2-9.el8.x86_64
  rpmdevtools-8.10-7.el8.noarch
...output omitted...

Complete!
```

► 7. Explorez l'historique et les options d'annulation de **yum**.

7.1. Utilisez la commande **yum history** pour afficher l'historique récent de **yum**.

ID	Command line	Date and time	Action(s)	Altered
6	group install RPM Develo	2019-02-26 17:11	Install	61
5	install guile	2019-05-26 17:05	Install	4
4	-y install @base firewal	2019-02-04 11:27	Install	127 EE
3	-C -y remove firewalld -	2019-01-16 13:12	Removed	11 EE
2	-C -y remove linux-firmw	2019-01-16 13:12	Removed	1
1		2019-01-16 13:05	Install	447 EE

7.2. Utilisez la commande **yum history info** pour vérifier que la dernière transaction correspond à l'installation du groupe.

```
[root@servera ~]# yum history info 6
Transaction ID : 6
Begin time      : Tue 26 Feb 2019 05:11:25 PM EST
Begin rpmdb     : 563:bf48c46156982a78e290795400482694072f5ebb
End time        : Tue 26 Feb 2019 05:11:33 PM EST (8 seconds)
End rpmdb       : 623:bf25b424ccf451dd0a6e674fb48e497e66636203
User            : Student User <student>
Return-Code     : Success
Releasever     : 8
Command Line    : group install RPM Development Tools
Packages Altered:
  Install dwz-0.12-9.el8.x86_64          @rhel-8.0-for-x86_64-appstream-rpms
  Install efi-srpm-macros-3-2.el8.noarch @rhel-8.0-for-x86_64-appstream-rpms
...output omitted...
```

7.3. Utilisez la commande **yum history undo** pour supprimer le jeu de paquetages qui ont été installés lors de l'installation du paquetage **guile**.

```
[root@servera ~]# yum history undo 5
```

► 8. Déconnectez-vous du système servera.

```
[root@servera ~]# exit
logout
[student@servera ~]$ exit
Connection to servera closed.
[student@workstation ~]$
```

Finish (Terminer)

Sur workstation, exéutez le script **lab software-yum finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab software-yum finish
```

L'exercice guidé est maintenant terminé.

ACTIVATION DE RÉFÉRENTIELS LOGICIELS YUM

OBJECTIFS

Au terme de cette section, vous devez être en mesure d'activer et de désactiver l'utilisation des référentiels Yum de Red Hat ou de tierces parties par un serveur.

ACTIVATION DES RÉFÉRENTIELS LOGICIELS RED HAT

L'inscription d'un système auprès du service de gestion des abonnements configure automatiquement l'accès aux référentiels logiciels en fonction des abonnements liés. Pour afficher tous les référentiels disponibles :

```
[user@host ~]$ yum repolist all
...output omitted...
rhel-8-for-x86_64-appstream-debug-rpms    ... AppStream (Debug RPMs)  disabled
rhel-8-for-x86_64-appstream-rpms           ... AppStream (RPMs)        enabled:
5,045
rhel-8-for-x86_64-appstream-source-rpms   ... AppStream (Source RPMs) disabled
rhel-8-for-x86_64-baseos-debug-rpms       ... BaseOS (Debug RPMs)   enabled:
2,270
rhel-8-for-x86_64-baseos-rpms            ... BaseOS (RPMs)         enabled:
1,963
...output omitted...
```

La commande **yum-config-manager** peut être utilisée pour activer ou désactiver les référentiels. Pour activer un référentiel, la commande définit le paramètre **enabled** sur **1**. Par exemple, la commande suivante active le référentiel **rhel-8-server-debug-rpms** :

```
[user@host ~]$ yum-config-manager --enable rhel-8-server-debug-rpms
Updating Subscription Management repositories.
=====
repo: rhel-8-for-x86_64-baseos-debug-rpms =====
[rhel-8-for-x86_64-baseos-debug-rpms]
bandwidth = 0
baseurl = [https://cdn.redhat.com/content/dist/rhel8/8/x86_64/baseos/debug]
cachedir = /var/cache/dnf
cost = 1000
deltarpm = 1
deltarpm_percentage = 75
enabled = 1
...output omitted...
```

Les sources non Red Hat fournissent des logiciels via des référentiels tiers, accessibles à l'aide de la commande **yum** à partir d'un site Web, d'un serveur FTP ou du système de fichiers local. Par exemple, Adobe fournit certains de ses logiciels gratuits pour Linux par le biais d'un référentiel Yum. Dans une salle de classe Red Hat, le serveur de classe **content.example.com** héberge les référentiels Yum.

CHAPITRE 14 | Installation et mise à jour de paquetages logiciels

Pour permettre la prise en charge d'un nouveau référentiel tiers, créez un fichier dans le répertoire **/etc/yum.repos.d/**. Les fichiers de configuration de référentiel doivent se terminer par une extension **.repo**. La définition du référentiel contient l'URL et le nom de ce dernier. Elle indique, en outre, s'il faut utiliser GPG pour vérifier les signatures des paquetages et, si tel est le cas, pour vérifier l'URL pointant vers la clé GPG approuvée.

Création de référentiels Yum

Créez des référentiels Yum avec la commande **yum-config-manager**.

La commande suivante crée un fichier nommé **/etc/yum.repos.d/dl.fedoraproject.org_pub_epel_8_x86_64.repo** avec la sortie affichée.



MISE EN GARDE

EPEL 8 n'avait pas encore été publié par le projet communautaire au moment de la rédaction de cette section. Une version d'EPEL peut ne pas correspondre à la version de Red Hat Enterprise Linux, étant donné qu'elle n'est pas fournie par Red Hat mais par une communauté de bénévoles. L'URL ci-dessous peut ne pas être valide. Nous avons utilisé l'exemple ci-dessous pour illustrer comment activer un référentiel de paquetage tiers.

```
[user@host ~]$ yum-config-manager --add-repo="http://dl.fedoraproject.org/pub/epel/8/x86_64/"
Loaded plugins: langpacks
adding repo from: http://dl.fedoraproject.org/pub/epel/8/x86_64/
[dl.fedoraproject.org_pub_epel_8_x86_64_]
name=added from: http://dl.fedoraproject.org/pub/epel/8/x86_64/
baseurl=http://dl.fedoraproject.org/pub/epel/8/x86_64/
enabled=1
```

Modifiez ce fichier pour indiquer des valeurs personnalisées et l'emplacement d'une clé GPG. Les clés sont stockées à divers emplacements sur le site de référentiel distant, tels que `http://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-8`. Les administrateurs devraient télécharger la clé dans un fichier local, plutôt qu'autoriser **yum** à la récupérer depuis une source externe. Par exemple :

```
[EPEL]
name=EPEL 8
baseurl=http://dl.fedoraproject.org/pub/epel/8/x86_64/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-8
```

Paquetages de configuration RPM pour les référentiels locaux

Certains référentiels fournissent un fichier de configuration et la clé publique GPG dans le cadre d'un paquetage RPM qui peut être téléchargé et installé à l'aide de la commande **yum localinstall**. Par exemple, le projet EPEL (Extra Packages for Enterprise Linux), géré par une communauté de bénévoles fournit des logiciels non pris en charge par Red Hat, mais compatibles avec Red Hat Enterprise Linux.

**MISE EN GARDE**

EPEL 8 n'avait pas encore été publié par le projet communautaire au moment de la rédaction de cette section. Nous avons rédigé cette section pour illustrer le processus de configuration prévu pour EPEL 8, basé sur le processus d'EPEL 7, comme exemple pour activer un référentiel de paquetage tiers qui fournit sa configuration de référentiel aux clients à l'aide d'un paquetage RPM.

La commande suivante installe le paquetage de référentiel EPEL de Red Hat Enterprise Linux 8 :

```
[user@host ~]$ rpm --import http://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-8
[user@host ~]$ yum install http://dl.fedoraproject.org/pub/epel/8/x86_64/e/epel-release-8-2.noarch.rpm
```

Les fichiers de configuration répertorient souvent plusieurs références de référentiel dans un seul et même fichier. Chaque référence commence par un nom composé d'un seul mot et placé entre crochets.

```
[user@host ~]$ cat /etc/yum.repos.d/epel.repo
[epel]
name=Extra Packages for Enterprise Linux 8 - $basearch
#baseurl=http://download.fedoraproject.org/pub/epel/8/$basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-8&arch=$basearch
failovermethod=priority
enabled=1
gpgcheck=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-8

[epel-debuginfo]
name=Extra Packages for Enterprise Linux 8 - $basearch - Debug
#baseurl=http://download.fedoraproject.org/pub/epel/8/$basearch/debug
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-debug-8&arch=
$basearch
failovermethod=priority
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-8
gpgcheck=1

[epel-source]
name=Extra Packages for Enterprise Linux 8 - $basearch - Source
#baseurl=http://download.fedoraproject.org/pub/epel/8/SRPMS
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-source-8&arch=
$basearch
failovermethod=priority
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-8
gpgcheck=1
```

Pour définir un référentiel, mais ne pas le rechercher par défaut, insérez le paramètre `enabled=0`. Les référentiels peuvent être activés et désactivés de façon persistante avec la commande

yum-config-manager ou de façon temporaire avec les options de la commande **yum**, **--enablerepo= PATTERN** et **--disablerepo= PATTERN**.



MISE EN GARDE

Installez la clé GPG RPM avant d'installer des paquetages signés. Cela permet de vérifier que les paquetages appartiennent à une clé qui a été importée. Sinon, la commande **yum** échoue à cause d'une clé manquante. Il est possible d'utiliser l'option **--nogpgcheck** pour ignorer les clés GPG manquantes, mais cela peut autoriser l'installation de paquetages falsifiés ou dangereux sur le système, compromettant potentiellement sa sécurité.



RÉFÉRENCES

Pages de manuel **yum(1)**, **yum.conf(5)** et **yum-config-manager(1)**

Pour plus d'informations, reportez-vous au chapitre *Installing software with yum* de *Red Hat Enterprise Linux 8.0 Configuring basic system settings* sur
https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/installing-software-with-yum_configuring-basic-system-settings

► EXERCICE GUIDÉ

ACTIVATION DE RÉFÉRENTIELS LOGICIELS YUM

Au cours de cet exercice, vous allez configurer votre serveur pour obtenir des paquetages à partir d'un référentiel Yum distant, puis mettre à jour ou installer un paquetage à partir de ce référentiel.

RÉSULTATS

Vous serez en mesure de configurer un système pour obtenir les mises à jour logicielles d'un serveur de la classe et mettre à jour le système pour utiliser les paquetages plus récents.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab software-repo start**. La commande exécute un script de démarrage qui détermine si l'hôte servera est accessible sur le réseau. Le script s'assure également que le paquetage yum est installé.

```
[student@workstation ~]$ lab software-repo start
```

- 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Utilisez la commande **su -** pour basculer vers root à l'invite du shell.

```
[student@servera ~]$ su -
Password: redhat
[root@servera ~]#
```

- 3. Configurez les référentiels de logiciels sur servera pour obtenir des paquetages et des mises à jour personnalisés à partir de l'URL suivante :

- Paquetages personnalisés fournis sur http://content.example.com/rhel8.0/x86_64/rhcsa-practice/rht
 - Mises à jour des paquetages personnalisés fournis sur http://content.example.com/rhel8.0/x86_64/rhcsa-practice/errata
- 3.1. Utilisez **yum-config-manager** pour ajouter le référentiel de paquetages personnalisés.

```
[root@servera ~]# yum-config-manager \
--add-repo "http://content.example.com/rhel8.0/x86_64/rhcsa-practice/rht"
Adding repo from: http://content.example.com/rhel8.0/x86_64/rhcsa-practice/rht
```

- 3.2. Examinez le fichier de référentiels logiciels créé par la commande précédente dans le répertoire **/etc/yum.repos.d**. Utilisez la commande **vim** pour éditer le fichier et ajouter le paramètre **gpgcheck=0** permettant de désactiver la vérification de la clé GPG pour le référentiel.

```
[root@servera ~]# vim \
/etc/yum.repos.d/content.example.com_rhel8.0_x86_64_rhcsa-practice_rht.repo
[content.example.com_rhel8.0_x86_64_rhcsa-practice_rht]
name=created by dnf config-manager from http://content.example.com/rhel8.0/x86_64/
rhcsa-practice/rht
baseurl=http://content.example.com/rhel8.0/x86_64/rhcsa-practice/rht
enabled=1
gpgcheck=0
```

- 3.3. Créez le fichier **/etc/yum.repos.d/errata.repo** pour activer le référentiel des mises à jours avec le contenu suivant :

```
[rht-updates]
name=rht updates
baseurl=http://content.example.com/rhel8.0/x86_64/rhcsa-practice/errata
enabled=1
gpgcheck=0
```

- 3.4. Utilisez la commande **yum repolist all** pour lister tous les référentiels du système :

```
[root@servera ~]# yum repolist all
repo id                                repo name      status
content.example.com_rhel8.0_x86_64_rhcsa-practice_rht created by .... enabled: 2
rht-updates                               rht updates    enabled: 2
...output omitted...
```

► 4. Désactivez le référentiel logiciel **rht-updates** et installez le paquetage **rht-system**.

- 4.1. Utilisez **yum-config-manager --disable** pour désactiver le référentiel **rht-updates**.

```
[root@servera ~]# yum-config-manager --disable rht-updates
```

- 4.2. Listez, puis installez le paquetage **rht-system**:

```
[root@servera ~]# yum list rht-system
Available Packages
rht-system.noarch 1.0.0-1 content.example.com_rhel8.0_x86_64_rhcsa-practice_rht
[root@servera ~]# yum install rht-system
Dependencies resolved.
```

```
=====
 Package           Arch      Version       Repository      Size
=====
Installing:
 rht-system        noarch    1.0.0-1      content..._rht   3.7 k
 ...output omitted...
 Is this ok [y/N]: y
 ...output omitted...
Installed:
 rht-system-1.0.0-1.noarch
Complete!
```

- 4.3. Vérifiez que le paquetage *rht-system* est installé et notez le numéro de version du paquetage.

```
[root@servera ~]# yum list rht-system
Installed Packages
rht-system.noarch 1.0.0-1 @content.example.com_rhel8.0_x86_64_rhcsa-practice_rht
```

- 5. Activez le référentiel logiciel *rht-updates* et mettez à jour tous les paquetages logiciels pertinents.

- 5.1. Utilisez **yum-config-manager --enable** pour activer le référentiel *rht-updates*.

```
[root@servera ~]# yum-config-manager --enable rht-updates
```

- 5.2. Utilisez la commande **yum update** pour mettre à jour tous les paquetages de logiciels sur *servera*.

```
[root@servera ~]# yum update
Dependencies resolved.
=====
 Package           Arch      Version       Repository      Size
=====
Upgrading:
 rht-system        x86_64    1.0.0-2.el7    rht-updates   3.9 k
 ...output omitted...
 Is this ok [y/N]: y
 ...output omitted...
Complete!
```

- 5.3. Vérifiez que le paquetage *rht-system* est mis à niveau et notez le numéro de version du paquetage.

```
[root@servera ~]# yum list rht-system
Installed Packages
rht-system.noarch 1.0.0-2.el7          @rht-updates
```

- 6. Quittez *servera*.

```
[root@servera ~]# exit  
logout  
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

Fin

Sur workstation, exéutez le script **lab software-repo finish** pour mettre fin à l'exercice. Ce script supprime tous les référentiels logiciels et paquetages installés sur servera pendant l'exercice.

```
[student@workstation ~]$ lab software-repo finish
```

L'exercice guidé est maintenant terminé.

GESTION DES FLUX DE MODULES DE PAQUETAGES

OBJECTIFS

Après avoir terminé cette section, vous devez pouvoir :

- Expliquer comment les modules permettent l'installation de versions spécifiques des logiciels.
- Comment lister, activer et changer les flux de modules.
- Installer et mettre à jour des paquetages à partir d'un module.

INTRODUCTION AU FLUX D'APPLICATIONS

Red Hat Enterprise Linux 8.0 introduit le concept de flux d'applications. Plusieurs versions des composants de l'espace utilisateur fournis avec la distribution sont désormais livrées en même temps. Ils peuvent être mis à jour plus fréquemment que les paquetages du système d'exploitation principal. Cela vous offre une plus grande flexibilité pour personnaliser Red Hat Enterprise Linux sans impacter la stabilité sous-jacente de la plateforme ni des déploiements spécifiques.

Traditionnellement, la gestion de versions alternatives du paquetage logiciel d'une application et des paquetages associés impliquait de conserver des référentiels différents pour chaque version différente. Cela a créé une situation fastidieuse à gérer pour les développeurs qui souhaitaient disposer de la dernière version d'une application et les administrateurs qui voulaient obtenir la version la plus stable de l'application. Ce processus est simplifié dans Red Hat Enterprise Linux 8 au moyen d'une nouvelle technologie appelée *Modularité*. La modularité permet à un référentiel unique d'héberger plusieurs versions du paquetage d'une application et de ses dépendances.

Le contenu Red Hat Enterprise Linux 8 est distribué via deux référentiels logiciels principaux : *BaseOS* et *AppStream* (*flux d'applications*).

BaseOS

Le référentiel BaseOS fournit le contenu du système d'exploitation principal pour Red Hat Enterprise Linux en tant que paquetages RPM. Le cycle de vie des composants de BaseOS est identique à celui du contenu des versions précédentes de Red Hat Enterprise Linux.

Application Stream

Le référentiel Application Stream fournit du contenu avec différents cycles de vie à la fois sous forme de modules et de paquetages traditionnels. Application Stream contient les composants nécessaires du système, ainsi qu'un large éventail d'applications auparavant disponibles dans Red Hat Software Collections, ainsi que dans d'autres produits et programmes.



IMPORTANT

BaseOS et AppStream sont des composantes essentielles d'un système Red Hat Enterprise Linux 8.

Le référentiel Application Stream présente deux types de contenus : *Modules* et les paquetages RPM traditionnels. Un module décrit un ensemble de paquetages RPM qui vont ensemble. Les

modules peuvent contenir plusieurs flux pour rendre plusieurs versions d'applications disponibles pour l'installation. L'activation d'un flux de module donne au système un accès aux paquetages RPM contenus dans ce flux de module.

LES MODULES

Un module est un ensemble de paquetages RPM qui forment un ensemble cohérent allant ensemble. En règle générale, cela s'organise autour d'une version spécifique d'une application logicielle ou d'un langage de programmation. Un module typique peut contenir des paquetages avec une application, des paquetages avec les bibliothèques de dépendances spécifiques à l'application, des paquetages avec une documentation pour l'application et des paquetages avec des utilitaires auxiliaires.

Flux de modules

Chaque module peut avoir un ou plusieurs flux de modules, qui contiennent différentes versions du contenu. Chacun des flux reçoit des mises à jour indépendamment. Vous pouvez vous représenter le flux de modules comme un référentiel virtuel dans le référentiel physique du flux d'applications.

Pour chaque module, un seul de ses flux peut être activé et fournir ses paquetages.

Profils de modules

Chaque module peut avoir un ou plusieurs profils. Un profil est une liste de certains paquetages à installer conjointement pour une application donnée, comme pour un serveur, un client, un développement, une installation minimale ou autre.

L'installation d'un profil de module particulier consiste simplement à installer un ensemble particulier de paquetages à partir du flux de modules. Vous pouvez ensuite installer ou désinstaller les paquetages normalement. Si vous n'indiquez pas de profil, le module installera son *profil par défaut*.

GESTION DE MODULES À L'AIDE DE YUM

La version 4 de Yum, nouveau dans Red Hat Enterprise Linux 8, prend en charge les nouvelles fonctionnalités modulaires du flux d'applications.

La commande **yum module** a été ajoutée pour gérer le contenu modulaire. Autrement, **yum** fonctionne avec les modules comme avec les paquetages ordinaires.

Liste des modules

Pour afficher une liste des modules disponibles, utilisez **yum module list**:

```
[user@host ~]$ yum module list
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMs)
Name           Stream      Profiles   Summary
389-ds          1.4        default    389 Directory Server (base)
ant             1.10 [d]    common    [d] Java build tool
container-tools 1.0 [d]    common    [d] Common tools and dependencies for
                           container runtimes
...output omitted...
Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

**NOTE**

Utilisez le *Hint* à la fin de la sortie pour aider à déterminer quels flux et profils sont activés, désactivés, installés, ainsi que ceux qui sont les valeurs par défaut.

Pour afficher la liste des flux de modules d'un module spécifique et récupérer leur statut :

```
[user@host ~]$ yum module list perl
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMS)
Name Stream Profiles Summary
perl 5.24 common [d], minimal Practical Extraction and Report Language
perl 5.26 [d] common [d], minimal Practical Extraction and Report Language
```

Pour afficher les détails d'un module :

```
[user@host ~]$ yum module info perl
Name : perl
Stream : 5.24
Version : 820190207164249
Context : ee766497
Profiles : common [d], minimal
Default profiles : common
Repo : rhel-8-for-x86_64-appstream-rpms
Summary : Practical Extraction and Report Language
...output omitted...
Artifacts : perl-4:5.24.4-403.module+el8+2770+c759b41a.x86_64
            : perl-Algorithm-Diff-0:1.1903-9.module+el8+2464+d274aed1.noarch
            : perl-Archive-Tar-0:2.30-1.module+el8+2464+d274aed1.noarch
...output omitted...
```

**NOTE**

Sans spécifier de flux de modules, **yum module info** affiche la liste des paquetages installés par le profil par défaut d'un module à l'aide du flux. Utilisez le format *module-name:stream* pour afficher un flux de modules spécifique. Ajoutez l'option **--profile** pour afficher des informations sur les paquetages installés par chacun des profils du module. Par exemple :

```
[user@host ~]$ yum module info --profile perl:5.24
```

Activation des flux de modules et installation de modules

Les flux de modules doivent être activés pour pouvoir installer leur module. Pour simplifier ce processus, lorsqu'un module est installé, il active son flux de module si nécessaire. Les flux de modules peuvent être activés manuellement à l'aide de **yum module enable** et en fournissant le nom du flux de module.

**IMPORTANT**

Un seul flux de module peut être activé pour un module donné. L'activation d'un flux de module supplémentaire désactivera le flux de module d'origine.

Installez un module en utilisant le flux et les profils par défaut :

```
[user@host ~]$ sudo yum module install perl
Dependencies resolved.
=====
Package      Arch    Version       Repository      Size
=====
Installing group/module packages:
perl          x86_64  4:5.26.3-416.el8
                           rhel-8-for-x86_64-appstream-htb-rpms 72 k
Installing dependencies:
...output omitted...
Running transaction
Preparing : 1/1
Installing : perl-Exporter-5.72-396.el8.noarch 1/155
Installing : perl-Carp-1.42-396.el8.noarch 2/155
...output omitted...
Installed:
perl-4:5.26.3-416.el8.x86_64
perl-Encode-Locale-1.05-9.el8.noarch
...output omitted...
Complete!
```

**NOTE**

Les mêmes résultats auraient pu être obtenus en exécutant **yum install @perl**.

La notation @ informe **yum** que l'argument est un nom de module au lieu d'un nom de paquetage.

Pour vérifier l'état du flux de module et du profil installé :

```
[user@host ~]$ yum module list perl
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMS)
Name Stream Profiles Summary
perl 5.24 common, minimal Practical Extraction and Report Language
perl 5.26 [d][e] common [i], minimal Practical Extraction and Report Language

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

Suppression de modules et désactivation des flux de modules

La suppression d'un module supprime tous les paquetages installés par les profils du flux de modules actuellement activé, ainsi que tous les paquetages et modules supplémentaires qui en dépendent. Les paquetages installés à partir de ce flux de modules qui ne figurent dans aucun de ses profils restent installés sur le système et peuvent être supprimés manuellement.



MISE EN GARDE

La suppression de modules et le changement de flux de modules peuvent s'avérer un peu compliqués. Changer le flux activé d'un module revient à réinitialiser le flux actuel et à activer le nouveau flux. Cela ne modifie pas automatiquement les paquetages installés. Vous devez le faire manuellement.

L'installation directe d'un flux de modules différent de celui qui est actuellement installé n'est pas recommandée, car des scripts de mise à niveau risquent de s'exécuter pendant l'installation, ce qui aurait un impact négatif sur le flux de modules d'origine. Cela pourrait entraîner une perte de données ou d'autres problèmes de configuration.

Procédez avec prudence.

Pour supprimer un module installé :

```
[user@host ~]$ sudo yum module remove perl
Dependencies resolved.
=====
Package           ArchVersion      Repository
Size
=====
Removing:
perl              x86_64:5.26.3-416.el8    @rhel-8.0-for-x86_64-
appstream-rpms 0
Removing unused dependencies:
...output omitted...
Running transaction
Preparing       :                                         1/1
Erasing        : perl-4:5.26.3-416.el8.x86_64          1/155
Erasing        : perl-CPAN-2.18-397.el8.noarch         2/155
...output omitted...
Removed:
perl-4:5.26.3-416.el8.x86_64
dwz-0.12-9.el8.x86_64
...output omitted...
Complete!
```

Une fois le module supprimé, le flux de module est toujours activé. Pour vérifier que le flux de module est toujours activé :

```
[user@host ~]$ yum module list perl
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMs)
Name   Stream      Profiles          Summary
perl  5.24        common [d], minimal Practical Extraction and Report Language
perl  5.26 [d][e]  common [d], minimal Practical Extraction and Report Language

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

Pour désactiver le flux de module :

```
[user@host ~]$ sudo yum module disable perl
...output omitted...
Dependencies resolved.
=====
Package           Arch      Version       Repository      Size
=====
Disabling module streams:
perl              5.26
Is this ok [y/N]: y
Complete!
```

Changement de flux de modules

Le changement de flux de modules nécessite généralement la mise à niveau ou la rétrogradation du contenu vers une version différente.

Pour garantir un changement sans heurt, vous devez d'abord supprimer les modules fournis par le flux de modules. Tous les paquetages installés par les profils du module, ainsi que tous les modules et paquetages sur lesquels ces derniers ont des dépendances seront supprimés.

Pour répertorier les paquetages installés à partir du module, dans l'exemple ci-dessous, le module `postgresql:9.6` est installé :

```
[user@host ~]$ sudo yum module info postgresql | grep module+el8 | \
sed 's/.*/ //g;s/\n/ /g' | xargs yum list installed
Installed Packages
postgresql.x86_64          9.6.10-1.module+el8+2470+d1bafa0e   @rhel-8.0-for-
x86_64-appstream-rpms
postgresql-server.x86_64     9.6.10-1.module+el8+2470+d1bafa0e   @rhel-8.0-for-
x86_64-appstream-rpms
```

Supprimez les paquetages répertoriés dans la commande précédente. Marquez les profils de modules à désinstaller.

```
[user@host ~]$ sudo yum module remove postgresql
...output omitted...
Is this ok [y/N]: y
...output omitted...
Removed:
  postgresql-server-9.6.10-1.module+el8+2470+d1bafa0e.x86_64
  libpq-10.5-1.el8.x86_64  postgresql-9.6.10-1.module+el8+2470+d1bafa0e.x86_64
Complete
```

Après avoir supprimé les profils de modules, réinitialisez le flux de modules. Utilisez la commande `yum module reset` pour réinitialiser le flux de modules.

```
[user@host ~]$ sudo yum module reset postgresql
=====
Package           Arch      Version       Repository      Size
=====
Resetting module streams:
postgresql        9.6
```

Transaction Summary

```
=====
Is this ok [y/N]: y
Complete!
```

Pour activer un autre flux de modules et installer le module :

```
[user@host ~]$ sudo yum module install postgresql:10
```

Le nouveau flux de module sera activé et le flux actuel désactivé. Il peut être nécessaire de mettre à jour ou de rétrograder des paquetages du flux de module précédent qui ne figurent pas dans le nouveau profil. Utilisez **yum distro-sync** pour effectuer cette tâche si nécessaire. Il se peut également que des paquetages restent installés à partir du flux de module précédent. Supprimez-les à l'aide de **yum remove**.



RÉFÉRENCES

Pour plus d'informations, reportez-vous au chapitre *Using AppStream de Red Hat Enterprise Linux 8.0 Installing, managing and removing user space components* à l'adresse

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/installing_managing_and_removing_user_space_components/

Modularité

<https://docs.fedoraproject.org/en-US/modularity/>

► EXERCICE GUIDÉ

GESTION DES FLUX DE MODULES DE PAQUETAGES

Dans cet exercice, vous allez lister les modules disponibles, activer un flux de module spécifique et installer les paquetages à partir de ce flux.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Lister les modules installés et examiner les informations d'un module.
- Activer et installer un module à partir d'un flux.
- Basculer vers un flux de module spécifique.
- Supprimer et désactiver un module.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab software-module start**. La commande exécute un script de démarrage qui détermine si l'hôte, servera, est accessible sur le réseau. Le script s'assure également que les référentiels de logiciels requis sont disponibles et installe le module *postgresql:9.6*.

```
[student@workstation ~]$ lab software-module start
```

- 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Basculez vers root à l'invite du shell.

```
[student@servera ~]$ su -
Password: redhat
[root@servera ~]#
```

- 3. Listez les modules, les flux et les modules installés disponibles. Examinez les informations pour le module *python36*.

- 3.1. Utilisez la commande **yum module list** pour lister les modules et les flux disponibles.

```
[root@servera ~]# yum module list
Red Hat Enterprise Linux 8.0 AppStream (dvd)
Name      Stream     Profiles          Summary
...
python27   2.7 [d]    common [d]       Python programming ..., version 2.7
python36   3.6 [d]    common [d], build Python programming ..., version 3.6
...
Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

- 3.2. Utilisez la commande **yum module list --installed** pour lister les modules et les flux installés.

```
[root@servera ~]# yum module list --installed
Red Hat Enterprise Linux 8.0 AppStream (dvd)
Name      Stream     Profiles          Summary
...
postgresql 9.6 [e]  client, server [d] [i] PostgreSQL server and client ...
...
Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

- 3.3. Utilisez la commande **yum module info** pour examiner les détails du module *python36*.

```
[root@servera ~]# yum module info python36
Name           : python36
Stream         : 3.6 [d]
Version        : 820190123171828
Context        : 17efdbc7
Profiles       : common [d], build
Default profiles: common
Repo           : rhel-8.0-for-x86_64-appstream-rpms
Summary        : Python programming language, version 3.6
...output omitted...
Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled, [a]ctive]
```

- 4. Installez le module *python36* à partir du flux 3.6 et le profil **common**. Vérifiez l'état actuel du module.

- 4.1. Utilisez la commande **yum module install** pour installer le module *python36*. Utilisez la syntaxe **name:stream/profile** pour installer le module *python36* à partir du flux 3.6 et du profil **common**.



NOTE

Vous pouvez omettre **/profile** pour utiliser le profil par défaut et **:stream** pour utiliser le flux par défaut.

```
[root@servera ~]# yum module install python36:3.6/common
...output omitted...
Is this ok [y/N]: y
...output omitted...
Complete!
```

4.2. Examinez le statut actuel du module *python36*.

```
[root@servera ~]# yum module list python36
Red Hat Enterprise Linux 8.0 AppStream (dvd)
Name      Stream      Profiles          Summary
python36   3.6 [d][e]  common [d] [i], build Python programming ..., version 3.6

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

▶ 5. Changez le module *postgresql* du profil *server* pour utiliser le flux **10**.

5.1. Utilisez la commande **yum module list** pour lister le module *postgresql* et le flux. Notez que le flux de modules *postgresql:9.6* est actuellement installé.

```
[root@servera ~]# yum module list postgresql
Red Hat Enterprise Linux 8.0 AppStream (dvd)
Name      Stream      Profiles          Summary
postgresql  10 [d]    client, server [d]  PostgreSQL server and client ...
postgresql  9.6 [e]    client, server [d] [i]  PostgreSQL server and client ...

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

5.2. Supprimez et désactivez le flux de modules *postgresql* en même temps que tous les paquetages installés par le profil.

```
[root@servera ~]# yum module remove postgresql
...output omitted...
Is this ok [y/N]: y
...output omitted...
Removed:
  postgresql-server-9.6.10-1.module+el8+2470+d1bafa0e.x86_64
  libpq-10.5-1.el8.x86_64  postgresql-9.6.10-1.module+el8+2470+d1bafa0e.x86_64
Complete
```

5.3. Réinitialisez le module *postgresql* et ses flux.

```
[root@servera ~]# yum module reset postgresql
=====
 Package      Arch      Version      Repository      Size
 =====
 Resetting module streams:
 postgresql          9.6

 Transaction Summary
```

```
=====
Is this ok [y/N]: y
Complete!
```

- 5.4. Utilisez la commande **yum module install** pour basculer vers le flux de modules *postgresql:10*.

```
[root@servera ~]# yum module install postgresql:10
...output omitted...
Is this ok [y/N]: y
...output omitted...
Complete!
```

- 5.5. Vérifiez que le module *postgresql* est basculé vers le flux 10.

```
[root@servera ~]# yum module list postgresql
Red Hat Enterprise Linux 8.0 AppStream (dvd)
Name           Stream      Profiles          Summary
postgresql     10 [d] [e] client, server [d] [i] PostgreSQL server and client ...
postgresql     9.6         client, server [d] PostgreSQL server and client ...
```

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled

- 6. Supprimez et désactivez le flux de module *postgresql* en même temps que tous les paquetages installés par le profil.

- 6.1. Utilisez la commande **yum remove module** pour supprimer le module *postgresql*. La commande supprime également tous les paquetages installés à partir de ce module.

```
[root@servera ~]# yum module remove postgresql
...output omitted...
Is this ok [y/N]: y
...output omitted...
Complete!
```

- 6.2. Désactivez le flux de module *postgresql*.

```
[root@servera ~]# yum module disable postgresql
...output omitted...
Is this ok [y/N]: y
...output omitted...
Complete!
```

- 6.3. Vérifiez que le flux de module *postgresql* a été supprimé et désactivé.

```
[root@servera ~]# yum module list postgresql
Red Hat Enterprise Linux 8.0 AppStream (dvd)
Name      Stream     Profiles          Summary
postgresql 10 [d][x]  client, server [d]  PostgreSQL server and client ...
postgresql 9.6 [x]   client, server [d]  PostgreSQL server and client ...

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

► 7. Quittez servera.

```
[root@servera ~]# exit
logout
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

Fin

Sur workstation, exédez le script **lab software-module finish** pour mettre fin à l'exercice. Ce script supprime tous les modules installés sur servera pendant l'exercice.

```
[student@workstation ~]$ lab software-module finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

INSTALLATION ET MISE À JOUR DE PAQUETAGES LOGICIELS

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez gérer les référentiels logiciels et les flux de modules, puis installer et mettre à niveau les paquetages à partir de ces référentiels et de ces flux.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Gérer les référentiels logiciels et les flux de modules.
- Installer et mettre à niveau des paquetages à partir de référentiels et de flux.
- Installer un paquetage RPM.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez la commande `lab software-review start`. Ce script s'assure que `serverb` est disponible. Il télécharge également tous les paquetages requis pour l'exercice d'atelier.

```
[student@workstation ~]$ lab software-review start
```

1. Sur `serverb`, configurez un référentiel de logiciels pour obtenir des mises à jour. Nommez le référentiel `errata` et configuez-le dans le fichier `/etc/yum.repos.d/errata.repo`. Il doit pouvoir accéder à `http://content.example.com/rhel8.0/x86_64/rhcsa-practice/errata`. Ne vérifiez pas les signatures GPG.
2. Sur `serverb`, installez le nouveau paquetage `xsane-gimp` et le module `Apache HTTP Server` à partir du flux `2.4` et du profil `common`.
3. Pour des raisons de sécurité, `serverb` ne devrait pas pouvoir envoyer quoi que ce soit à imprimer. Pour ce faire, supprimez le paquetage `cups`. Quittez le compte `root`.
4. Le script de démarrage télécharge le paquetage `rhcsa-script-1.0.0-1.noarch.rpm` dans le répertoire `/home/student` sur `serverb`. Vérifiez que le paquetage `rhcsa-script-1.0.0-1.noarch.rpm` est disponible sur `serverb`. Installez le paquetage. Vous aurez besoin de priviléges de superutilisateur pour installer le paquetage. Vérifiez que le paquetage est installé. Quittez `serverb`.

Évaluation

Sur `workstation`, exécutez le script `lab software-review grade` pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab software-review grade
```

Finish (Terminer)

Sur workstation, exéutez le script **lab software-review finish** pour mettre fin à l'exercice. Ce script supprime le référentiel et les paquetages créés lors de cet exercice.

```
[student@workstation ~]$ lab software-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

INSTALLATION ET MISE À JOUR DE PAQUETAGES LOGICIELS

LISTE DE CONTRÔLE DES PERFORMANCES

Dans cet atelier, vous allez gérer les référentiels logiciels et les flux de modules, puis installer et mettre à niveau les paquetages à partir de ces référentiels et de ces flux.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Gérer les référentiels logiciels et les flux de modules.
- Installer et mettre à niveau des paquetages à partir de référentiels et de flux.
- Installer un paquetage RPM.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez la commande `lab software-review start`. Ce script s'assure que `serverb` est disponible. Il télécharge également tous les paquetages requis pour l'exercice d'atelier.

```
[student@workstation ~]$ lab software-review start
```

1. Sur `serverb`, configurez un référentiel de logiciels pour obtenir des mises à jour. Nommez le référentiel `errata` et configurez-le dans le fichier `/etc/yum.repos.d/errata.repo`. Il doit pouvoir accéder à `http://content.example.com/rhel8.0/x86_64/rhcsa-practice/errata`. Ne vérifiez pas les signatures GPG.
 - 1.1. À partir de `workstation`, utilisez la commande `ssh` pour vous connecter à `serverb` en tant qu'utilisateur `student`.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Utilisez la commande `su -` pour basculer vers l'utilisateur `root`. Le mot de passe est `redhat`.

```
[student@serverb ~]$ su -
Password: redhat
[root@serverb ~]#
```

CHAPITRE 14 | Installation et mise à jour de paquetages logiciels

1.3. Créez le fichier **/etc/yum.repos.d/errata.repo** avec le contenu suivant :

```
[errata]
name=Red Hat Updates
baseurl=http://content.example.com/rhel8.0/x86_64/rhcsa-practice/errata
enabled=1
gpgcheck=0
```

2. Sur **serverb**, installez le nouveau paquetage *xsane-gimp* et le module *Apache HTTP Server* à partir du flux 2.4 et du profil common.

2.1. Utilisez la commande **yum list** pour lister les paquetages disponibles pour *xsane-gimp*.

```
[root@serverb ~]# yum list xsane-gimp
Last metadata expiration check: 0:24:30 ago on Thu 07 Mar 2019 03:50:55 PM CET.
Available Packages
xsane-gimp.x86_64      0.999-30.el8    rhel-8.0-for-x86_64-appstream-rpms
```

2.2. Installez la dernière version du paquetage *xsane-gimp* en utilisant la commande **yum install**.

```
[root@serverb ~]# yum install xsane-gimp
...output omitted...
Install 57 Packages

Total download size: 51 M
Installed size: 205 M
Is this ok [y/N]: y
...output omitted...
Complete!
[root@serverb ~]#
```

2.3. Listez les modules et les flux disponibles. Cherchez le module *httpd*. Utilisez la commande **yum install** pour installer le module *httpd* avec le flux 2.4 et le profil common.

```
[student@serverb ~]$ yum module list
Name      Stream      Profiles          Summary
...output omitted...
httpd     2.4 [d]    common [d], devel, minimal   Apache HTTP Server
...output omitted...
[root@serverb ~]# yum module install httpd:2.4/common
Install 10 Packages

Total download size: 2.1 M
Installed size: 5.7 M
Is this ok [y/N]: y
...output omitted...
Complete!
[root@serverb ~]#
```

CHAPITRE 14 | Installation et mise à jour de paquetages logiciels

3. Pour des raisons de sécurité, `serverb` ne devrait pas pouvoir envoyer quoi que ce soit à imprimer. Pour ce faire, supprimez le paquetage `cups`. Quittez le compte `root`.

3.1. Utilisez la commande `yum list` pour lister le paquetage `cups` installé.

```
[root@serverb ~]# yum list cups
Installed Packages
cups.x86_64          1:2.2.6-25.el8           @rhel-8.0-for-x86_64-appstream-rpms
[root@serverb ~]#
```

3.2. Utilisez la commande `yum remove` pour supprimer le paquetage `cups`.

```
[root@serverb ~]# yum remove cups.x86_64
...output omitted...
Remove 9 Packages

Freed space: 11 M
Is this ok [y/N]: y
...output omitted...
Complete!
```

3.3. Quittez le compte `root`.

```
[root@serverb ~]# exit
[student@serverb ~]$
```

4. Le script de démarrage télécharge le paquetage `rhcsa-script-1.0.0-1.noarch.rpm` dans le répertoire `/home/student` sur `serverb`.

Vérifiez que le paquetage `rhcsa-script-1.0.0-1.noarch.rpm` est disponible sur `serverb`. Installez le paquetage. Vous aurez besoin de priviléges de superutilisateur pour installer le paquetage. Vérifiez que le paquetage est installé. Quittez `serverb`.

4.1. Utilisez la commande `rpm` pour vérifier que le paquetage `rhcsa-script-1.0.0-1.noarch.rpm` est disponible sur `serverb` en consultant les informations du paquetage.

```
[student@serverb ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm -i
Name        : rhcsa-script
Version     : 1.0.0
Release     : 1
Architecture: noarch
Install Date: (not installed)
Group       : System
Size        : 1056
License     : GPL
Signature   : (none)
Source RPM  : rhcsa-script-1.0.0-1.src.rpm
Build Date  : Wed 06 Mar 2019 11:29:46 AM CET
Build Host  : foundation0.ilt.example.com
Relocations : (not relocatable)
Packager    : Snehangshu Karmakar
URL         : http://example.com
Summary     : RHCSA Practice Script
```

```
Description :  
A RHCSA practice script.  
The package changes the motd.
```

4.2. Utilisez la commande **sudo yum localinstall** pour installer le paquetage *rhcsa-script-1.0.0-1.noarch.rpm*. Le mot de passe est **student**.

```
[student@serverb ~]$ sudo yum localinstall rhcsa-script-1.0.0-1.noarch.rpm  
[sudo] password for student: student  
Last metadata expiration check: 1:31:22 ago on Thu 07 Mar 2019 03:50:55 PM CET.  
Dependencies resolved.  
=====  
Package           Arch    Version     Repository      Size  
=====  
Installing:  
  rhcsa-script   noarch  1.0.0-1       @commandline      7.6 k  
  
Transaction Summary  
=====  
Install 1 Package  
  
Total size: 7.6 k  
Installed size: 1.0 k  
Is this ok [y/N]: y  
Downloading Packages:  
Running transaction check  
Transaction check succeeded.  
Running transaction test  
Transaction test succeeded.  
Running transaction  
  Preparing          :                           1/1  
  Running scriptlet: rhcsa-script-1.0.0-1.noarch 1/1  
  Installing        : rhcsa-script-1.0.0-1.noarch 1/1  
  Running scriptlet: rhcsa-script-1.0.0-1.noarch 1/1  
  Verifying          : rhcsa-script-1.0.0-1.noarch 1/1  
  
Installed:  
  rhcsa-script-1.0.0-1.noarch  
  
Complete!
```

4.3. Utilisez la commande **rpm** pour vérifiez que le paquetage est installé.

```
[student@serverb ~]$ rpm -q rhcsa-script  
rhcsa-script-1.0.0-1.noarch  
[student@serverb ~]$
```

4.4. Quittez serverb.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

Évaluation

Sur workstation, exécutez le script **lab software-review grade** pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab software-review grade
```

Finish (Terminer)

Sur workstation, exécutez le script **lab software-review finish** pour mettre fin à l'exercice. Ce script supprime le référentiel et les paquetages créés lors de cet exercice.

```
[student@workstation ~]$ lab software-review finish
```

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Red Hat Subscription Management fournit des outils permettant aux ordinateurs de s'abonner à des produits, de recevoir des mises à jour de paquetages logiciels et de suivre les informations relatives aux contrats d'assistance et aux abonnements utilisés par les systèmes.
- Les logiciels sont fournis sous forme de paquetages RPM, ce qui facilite l'installation, la mise à niveau et la désinstallation des logiciels du système.
- La commande **rpm** peut être utilisée pour interroger une base de données locale afin de fournir des informations sur le contenu des paquetages installés et d'installer des fichiers de paquetages téléchargés.
- **yum** est un puissant outil de ligne de commande qui peut être utilisé pour installer, mettre à jour, supprimer et interroger les paquetages logiciels.
- Red Hat Enterprise Linux 8 utilise les flux d'application pour fournir un référentiel unique permettant d'héberger plusieurs versions du paquetage d'une application et de ses dépendances.

CHAPITRE 15

ACCÈS AUX SYSTÈMES DE FICHIERS LINUX

PROJET

Accéder aux systèmes de fichiers existants, les inspecter et les utiliser sur un stockage connecté à un serveur Linux.

OBJECTIFS

- Expliquer ce qu'est un périphérique de traitement par blocs, interpréter les noms de fichiers des périphériques de stockage et identifier le périphérique de stockage utilisé par le système de fichiers pour un répertoire ou un fichier particulier.
- Accéder aux systèmes de fichiers en les attachant à un répertoire dans la hiérarchie du système de fichiers.
- Rechercher des fichiers sur les systèmes de fichiers montés au moyen des commandes **find** et **locate**.

SECTIONS

- Identification des systèmes de fichiers et des périphériques (avec quiz)
- Montage et démontage des systèmes de fichiers (avec exercice guidé)
- Localisation des fichiers sur le système (avec exercice guidé)

ATELIER

Accès aux systèmes de fichiers Linux

IDENTIFICATION DES SYSTÈMES DE FICHIERS ET DES PÉRIPHÉRIQUES

OBJECTIFS

Au terme de cette section, vous serez en mesure d'identifier un répertoire dans la hiérarchie du système de fichiers et le périphérique sur lequel il est stocké.

CONCEPTS DE GESTION DU STOCKAGE

Les fichiers sur un serveur Linux sont accessibles via la hiérarchie du système de fichiers, une arborescence inversée unique de répertoires. Cette hiérarchie du système de fichiers est assemblée à partir des *systèmes de fichiers* fournis par les périphériques de stockage disponibles sur votre système. Chaque système de fichiers est un périphérique de stockage formaté pour stocker des fichiers.

En un sens, la hiérarchie du système de fichiers présente un ensemble de systèmes de fichiers sur des périphériques de stockage distincts comme s'il s'agissait d'un ensemble de fichiers sur un énorme périphérique de stockage sur lequel vous pouvez naviguer. La plupart du temps, vous n'avez pas besoin de savoir sur quel périphérique de stockage un fichier particulier se trouve, mais juste le répertoire où il se trouve.

Mais, parfois, cela peut être important. Vous devrez peut-être déterminer la capacité disponible d'un périphérique de stockage et les répertoires affectés dans la hiérarchie du système de fichiers. Les journaux d'un périphérique de stockage peuvent contenir des erreurs et vous devez savoir quels systèmes de fichiers sont exposés. Vous pouvez simplement vouloir créer un lien physique entre deux fichiers. Vous devez donc savoir s'ils se trouvent sur le même système de fichiers pour déterminer si c'est possible.

Systèmes de fichiers et points de montage

Pour que le contenu d'un système de fichiers soit disponible dans la hiérarchie du système de fichiers, il doit être *monté* sur un répertoire vide. Ce répertoire est appelé *point de montage*. Une fois monté, si vous utilisez **ls** pour répertorier ce répertoire, vous verrez le contenu du système de fichiers monté et vous pourrez accéder à ces fichiers et les utiliser normalement. De nombreux systèmes de fichiers sont automatiquement montés lors du processus de démarrage.

Si vous avez uniquement travaillé avec les lettres de lecteur de Microsoft Windows, il s'agit d'un concept fondamentalement différent. Cela ressemble un peu à la fonctionnalité de dossiers montés sur NTFS.

Systèmes de fichiers, stockage et périphériques en mode bloc

L'accès de bas niveau aux périphériques de stockage sous Linux est fourni par un type spécial de fichier appelé *périphérique en mode bloc*. Ces périphériques en mode bloc doivent être formatés avec un système de fichiers avant de pouvoir être montés.

Les fichiers des périphériques en mode bloc sont stockés dans le répertoire **/dev** avec d'autres fichiers de périphérique. Les fichiers des périphériques sont créés automatiquement par le système d'exploitation. Dans Red Hat Enterprise Linux, le premier disque dur SATA/PATA, SAS, SCSI ou USB détecté est appelé **/dev/sda**, le deuxième **/dev/sdb**, et ainsi de suite. Ces noms représentent l'intégralité du disque dur.

D'autres types de stockage auront d'autres formes de nommage.

Dénomination des périphériques en mode bloc

TYPE DE PÉRIPHÉRIQUE	MODÈLE DE DÉNOMINATION DES PÉRIPHÉRIQUES
Stockage connecté à un SATA/SAS/USB	<code>/dev/sda</code> , <code>/dev/sdb</code> ...
Stockage paravirtualisé <code>virtio-blk</code> (certaines machines virtuelles)	<code>/dev/vda</code> , <code>/dev/vdb</code> ...
Stockage connecté à un NVMe (plusieurs SSD)	<code>/dev/nvme0</code> , <code>/dev/nvme1</code> ...
Stockage SD/MMC/eMMC (cartes SD)	<code>/dev/mmcblk0</code> , <code>/dev/mmcblk1</code> ...



NOTE

De nombreuses machines virtuelles utilisent le stockage paravirtualisé `virtio-scsi` le plus récent dont le nommage ressemble à `/dev/sd*`.

Partitions de disque

Normalement, vous ne créez pas l'intégralité du périphérique de stockage dans un système de fichiers. Les périphériques de stockage sont généralement divisés en plus petites portions appelées *partitions*.

Les partitions vous permettent de compartimenter un disque : les différentes partitions peuvent être formatées avec des systèmes de fichiers différents, ou être utilisées à des fins différentes. Par exemple, une partition peut contenir le répertoire personnel de l'utilisateur, tandis qu'un autre peut contenir les données et journaux du système. Si un utilisateur remplit la partition de son répertoire personnel avec des données, la partition système peut encore disposer d'espace disque.

Les partitions sont des périphériques en mode bloc à part entière. Sur un stockage connecté à un SATA, la première partition sur le premier disque est `/dev/sda1`. La troisième partition sur le deuxième disque est `/dev/sdb3`, etc. Les périphériques de stockage paravirtualisés ont un système de nommage similaire.

Un périphérique SSD connecté à un NVMe nomme ses partitions différemment. Dans ce cas, la première partition sur le premier disque est `/dev/nvme0p1`. La troisième partition sur le deuxième disque est `/dev/nvme1p3`, etc. Les cartes SD ou MMC ont un système de nommage similaire.

Une longue liste du fichier de périphérique `/dev/sda1` sur host révèle son type de fichier spécial `b`, qui représente le périphérique en mode bloc :

```
[user@host ~]$ ls -l /dev/sda1
brw-rw----. 1 root disk 8, 1 Feb 22 08:00 /dev/sda1
```

Volumes logiques

Une autre manière d'organiser les disques et les partitions implique la *gestion de volume logique* (LVM – Logical Volume Management). Avec la LVM, il est possible de regrouper un ou plusieurs

périphériques de traitement par blocs dans un pool de stockage nommé *groupe de volumes*. L'espace disque du groupe de volumes est ensuite réparti en un ou plusieurs *volumes logiques*, qui sont l'équivalent fonctionnel d'une partition résidant sur un disque physique.

Le système LVM attribue des noms aux groupes de volumes et aux volumes logiques lors de leur création. La LVM crée un répertoire dans **/dev** qui correspond au nom du groupe, puis crée un lien symbolique dans ce nouveau répertoire avec le même nom que le volume logique. Ce fichier de volume logique est alors disponible pour être monté. Par exemple, si un groupe de volumes est appelé **myvg** et que le volume logique à l'intérieur est appelé **mylv**, alors le nom de chemin complet du fichier de périphérique du volume logique est **/dev/myvg/mylv**.



NOTE

La forme du nom du périphérique de volume logique mentionnée ci-dessus est en réalité implémentée sous la forme d'un lien symbolique vers le fichier de périphérique utilisé pour y accéder, qui peut varier d'un démarrage à l'autre. Il existe une autre forme de nom de périphérique de volume logique lié à partir de fichiers dans **/dev/mapper** qui sont souvent utilisés et sont également des liens symboliques vers le fichier de périphérique réel.

EXAMEN DES SYSTÈMES DE FICHIERS

Pour avoir un aperçu des périphériques du système de fichiers local et distant et de la quantité d'espace disponible libre, exécutez la commande **df**. Lorsque la commande **df** est exécutée sans argument, elle renvoie l'espace disque total, l'espace disque utilisé, l'espace disque libre ainsi que le pourcentage de l'espace disque total utilisé sur tous les systèmes de fichiers normaux montés. Elle génère un rapport à la fois sur les systèmes de fichiers locaux et distants.

L'exemple suivant affiche les systèmes de fichiers et les points de montage sur host.

```
[user@host ~]$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
/devtmpfs        912584      0   912584  0% /dev
tmpfs           936516      0   936516  0% /dev/shm
tmpfs           936516  16812   919704  2% /run
tmpfs           936516      0   936516  0% /sys/fs/cgroup
/dev/vda3       8377344 1411332  6966012 17% /
/dev/vda1       1038336 169896   868440 17% /boot
tmpfs          187300      0   187300  0% /run/user/1000
```

Le partitionnement sur le système host affiche deux systèmes de fichiers physiques, qui sont montés sur **/** et **/boot**. Cela est courant pour les machines virtuelles. Les périphériques **tmpfs** et **devtmpfs** sont des systèmes de fichiers situés dans la mémoire du système. Tous les fichiers enregistrés dans **tmpfs** ou **devtmpfs** disparaissent après le redémarrage du système.

Pour améliorer la lisibilité des formats de sortie, il existe deux options différentes « lisibles par l'humain » : **-h** ou **-H**. La différence entre ces deux options est la suivante : **-h** indique la valeur en Kio (2^{10}), Mio (2^{20}) ou Gio (2^{30}), tandis que l'option **-H** la renvoie en unités SI : Ko (10^3), Mo (10^6), Go (10^9). Les fabricants de disques durs utilisent généralement des unités SI lorsqu'ils font la promotion de leurs produits.

Afficher un rapport sur les systèmes de fichiers situés sur le système host, avec toutes les unités converties en format lisible par l'utilisateur :

```
[user@host ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        892M    0  892M   0% /dev
tmpfs          915M    0  915M   0% /dev/shm
tmpfs          915M   17M  899M   2% /run
tmpfs          915M    0  915M   0% /sys/fs/cgroup
/dev/vda3       8.0G  1.4G  6.7G  17% /
/dev/vda1     1014M 166M  849M  17% /boot
tmpfs         183M    0  183M   0% /run/user/1000
```

Pour obtenir des informations plus détaillées sur l'espace utilisé par une arborescence de répertoires spécifique, utilisez la commande **du**. La commande **du** accepte les options **-h** et **-H** pour convertir la sortie dans un format lisible par l'utilisateur. La commande **du** affiche la taille de tous les fichiers dans l'arborescence des répertoires actuels, de manière récursive.

Afficher un rapport d'utilisation du disque pour le répertoire **/usr/share** sur host :

```
[root@host ~]# du /usr/share
...output omitted...
176 /usr/share/sm智ntools
184 /usr/share/nano
8 /usr/share/cmake/bash-completion
8 /usr/share/cmake
356676 /usr/share
```

Afficher un rapport d'utilisation du disque dans un format lisible par l'utilisateur pour le répertoire **/usr/share** sur host :

```
[root@host ~]# du -h /var/log
...output omitted...
176K /usr/share/sm智ntools
184K /usr/share/nano
8.0K /usr/share/cmake/bash-completion
8.0K /usr/share/cmake
369M /usr/share
```



RÉFÉRENCES

Pages de manuel **df(1)** et **du(1)**

► QUIZ

IDENTIFICATION DES SYSTÈMES DE FICHIERS ET DES PÉRIPHÉRIQUES

Choisissez les réponses aux questions suivantes :

- ▶ 1. Quel est le nom du fichier de périphérique de l'intégralité d'un disque dur SATA dans le répertoire /dev ?
 - a. /dev/vda
 - b. /dev/sda1
 - c. /dev/sda
 - d. /dev/vg_install/lv_home

- ▶ 2. Choisissez le nom du fichier de périphérique de la troisième partition sur le second disque dur SATA.
 - a. /dev/vda2
 - b. /dev/sda3
 - c. /dev/sdb2
 - d. /dev/sdb3

- ▶ 3. Quel est le nom du fichier de périphérique de l'intégralité du deuxième disque virtio-blk connecté à une machine virtuelle ?
 - a. /dev/vda2
 - b. /dev/sda2
 - c. /dev/vdb2
 - d. /dev/vdb

- ▶ 4. Choisissez le nom correct du fichier de périphérique de la troisième partition sur le deuxième disque virtio-blk connecté à une machine virtuelle ?
 - a. /dev/vda3
 - b. /dev/sda3
 - c. /dev/vdb3
 - d. /dev/vda3

- ▶ 5. Quelle commande fournit une vue d'ensemble des points de montage du système de fichiers et de la quantité d'espace libre disponible dans les unités SI ?
 - a. df
 - b. df -H
 - c. df -h
 - d. du -h

► SOLUTION

IDENTIFICATION DES SYSTÈMES DE FICHIERS ET DES PÉRIPHÉRIQUES

Choisissez les réponses aux questions suivantes :

- 1. Quel est le nom du fichier de périphérique de l'intégralité d'un disque dur SATA dans le répertoire /dev ?
- a. /dev/vda
 - b. /dev/sda1
 - c. /**dev/sda**
 - d. /dev/vg_install/lv_home
- 2. Choisissez le nom du fichier de périphérique de la troisième partition sur le second disque dur SATA.
- a. /dev/vda2
 - b. /dev/sda3
 - c. /dev/sdb2
 - d. /**dev/sdb3**
- 3. Quel est le nom du fichier de périphérique de l'intégralité du deuxième disque virtio-blk connecté à une machine virtuelle ?
- a. /dev/vda2
 - b. /dev/sda2
 - c. /dev/vdb2
 - d. /**dev/vdb**
- 4. Choisissez le nom correct du fichier de périphérique de la troisième partition sur le deuxième disque virtio-blk connecté à une machine virtuelle ?
- a. /dev/vda3
 - b. /dev/sda3
 - c. /**dev/vdb3**
 - d. /dev/vda3
- 5. Quelle commande fournit une vue d'ensemble des points de montage du système de fichiers et de la quantité d'espace libre disponible dans les unités SI ?
- a. df
 - b. **df -H**
 - c. df -h
 - d. du -h

MONTAGE ET DÉMONTAGE DE SYSTÈMES DE FICHIERS

OBJECTIFS

Au terme de cette section, vous serez en mesure d'accéder au contenu des systèmes de fichiers en ajoutant et en supprimant des systèmes de fichiers à la hiérarchie des systèmes de fichiers.

MONTAGE MANUEL DES SYSTÈMES DE FICHIERS

Pour accéder à un système de fichiers qui réside sur un périphérique de stockage amovible, il faut le monter. La commande **mount** permet à l'utilisateur **root** de monter manuellement un système de fichiers. Le premier argument de la commande **mount** spécifie le système de fichiers à monter. Le deuxième argument spécifie le répertoire à utiliser comme point de montage dans la hiérarchie du système de fichiers.

Deux méthodes courantes permettent de spécifier le système de fichiers sur une partition de disque pour la commande **mount** :

- avec le nom du fichier de périphérique dans **/dev** contenant le système de fichiers
- avec l'**UUID** (identificateur unique universel) écrit dans le système de fichiers.

Monter un périphérique est relativement simple. Vous devez identifier le périphérique à monter, vous assurer que le point de montage existe et monter le périphérique sur le point de montage.

Identification du périphérique en mode bloc

Qu'il s'agisse d'un lecteur de disque dur, d'un périphérique SSD dans un boîtier de serveur ou d'un périphérique de stockage USB, un périphérique de stockage hot-plug peut être connecté à un port différent chaque fois qu'il est connecté à un système.

Utilisez la commande **lsblk** pour lister les détails d'un périphérique en mode bloc spécifié ou de tous les périphériques disponibles.

```
[root@host ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
vda       253:0   0 12G  0 disk
└─vda1    253:1   0   1G  0 part /boot
└─vda2    253:2   0   1G  0 part [SWAP]
└─vda3    253:3   0 11G  0 part /
vdb       253:16  0 64G  0 disk
└─vdb1    253:17  0 64G  0 part
```

Si vous savez que vous venez d'ajouter un périphérique de stockage de 64 Go avec une partition, alors vous pouvez deviner d'après la sortie précédente que **/dev/vdb1** est la partition à monter.

Montage par nom du périphérique en mode bloc

L'exemple suivant monte le système de fichiers dans la partition **/dev/vdb1** sur le répertoire **/mnt/data**.

```
[root@host ~]# mount /dev/vdb1 /mnt/data
```

Pour monter un système de fichiers, le répertoire de destination doit déjà exister. Le répertoire **/mnt** existe par défaut et doit être utilisé comme point de montage temporaire.

Vous pouvez utiliser le répertoire **/mnt**, ou mieux encore créer un sous-répertoire de **/mnt** à utiliser comme point de montage temporaire, sauf si vous avez une bonne raison de le monter à un emplacement spécifique dans la hiérarchie du système de fichiers.



IMPORTANT

Si le répertoire qui joue le rôle de point de montage n'est pas vide, les fichiers copiés dans ce répertoire avant le montage du système de fichiers restent inaccessibles tant que le système de fichiers n'est pas à nouveau démonté.

Cette approche fonctionne bien à court terme. Toutefois, l'ordre dans lequel le système d'exploitation détecte les disques peut changer si des périphériques sont ajoutés ou supprimés du système. Cela modifiera le nom du périphérique associé à ce périphérique de stockage. Une meilleure approche consisterait à utiliser une caractéristique intégrée au système de fichiers.

Montage par UUID du système de fichiers

Un identifiant stable associé à un système de fichiers constitue son UUID, un nombre hexadécimal très long servant d'identificateur unique universel. Cet UUID fait partie du système de fichiers et reste inchangé tant qu'un système de fichiers n'est pas recréé.

La commande **lsblk -fp** liste le chemin d'accès complet du périphérique, ainsi que les UUID, les points de montage et le type de système de fichiers dans la partition. Si le système de fichiers n'est pas monté, le point de montage sera vide.

```
[root@host ~]# lsblk -fp
NAME      FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vda
└─/dev/vda1 xfs   swap   23ea8803-a396-494a-8e95-1538a53b821c /boot
└─/dev/vda2 swap   swap   cdf61ded-534c-4bd6-b458-cab18b1a72ea [SWAP]
└─/dev/vda3 xfs   swap   44330f15-2f9d-4745-ae2e-20844f22762d /
/dev/vdb
└─/dev/vdb1 xfs   swap   46f543fd-78c9-4526-a857-244811be2d88
```

Montez le système de fichiers avec son UUID.

```
[root@host ~]# mount UUID="46f543fd-78c9-4526-a857-244811be2d88" /mnt/data
```

MONTAGE AUTOMATIQUE DE PÉRIPHÉRIQUES DE STOCKAGE AMOVIBLES

Si vous êtes connecté et que vous utilisez l'environnement de travail graphique, tous les supports de stockage amovibles seront automatiquement montés lors de leur insertion.

Le périphérique de stockage amovible est monté sur **/run/media/USERNAME/LABEL** où **NOM D'UTILISATEUR** correspond au nom de l'utilisateur connecté à l'environnement graphique et

ÉTIQUETTE est un identifiant, correspondant souvent au nom donné au système de fichiers lors de sa création s'il est disponible.

Avant de retirer le périphérique, vous devez le démonter manuellement.

DÉMONTAGE DES SYSTÈMES DE FICHIERS

Les procédures d'arrêt et de redémarrage démontent automatiquement tous les systèmes de fichiers. Dans le cadre de ce processus, toutes les données du système de fichiers mises en cache dans la mémoire sont vidées sur le périphérique de stockage, ce qui garantit que le système de fichiers ne subit aucune corruption de données.



MISE EN GARDE

Les données du système de fichiers sont souvent mises en cache dans la mémoire. Par conséquent, afin d'éviter d'endommager les données sur le disque, il est essentiel de démonter les lecteurs amovibles avant de les débrancher. La procédure de démontage synchronise les données avant de libérer le lecteur, garantissant ainsi l'intégrité des données.

Pour démonter un système de fichiers, il faut indiquer le point de montage comme argument de la commande **umount**.

```
[root@host ~]# umount /mnt/data
```

Le démontage n'est pas possible si le système de fichiers monté est en cours d'utilisation. Pour que la commande **umount** réussisse, tous les processus doivent cesser d'accéder aux données situées sous le point de montage.

Dans l'exemple ci-dessous, la commande **umount** échoue parce que le système de fichiers est en cours d'utilisation (la coquille utilise **/mnt/data** en tant que répertoire de travail actuel), générant ainsi un message d'erreur.

```
[root@host ~]# cd /mnt/data
[root@host data]# umount /mnt/data
umount: /mnt/data: target is busy.
```

La commande **lsof** énumère tous les fichiers ouverts et le processus qui a accès dans le répertoire indiqué. Il est utile d'identifier les processus qui empêchent le démontage correct du système de fichiers.

```
[root@host data]# lsof /mnt/data
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
bash    1593 root cwd DIR 253,17      6  128 /mnt/data
lsof    2532 root cwd DIR 253,17     19  128 /mnt/data
lsof    2533 root cwd DIR 253,17     19  128 /mnt/data
```

Une fois les processus identifiés, vous pouvez accomplir une action : soit attendre que le processus se termine, soit lui envoyer un signal **SIGTERM** ou **SIGKILL**. Dans ce cas, il suffit d'ouvrir un dossier situé en dehors du point de montage à la place du dossier de travail courant.

```
[root@host data]# cd  
[root@host ~]# umount /mnt/data
```



NOTE

Une raison fréquente de l'échec du démontage des systèmes de fichiers est qu'un shell Bash utilise le point de montage ou un sous-répertoire comme répertoire de travail en cours. Utilisez la commande **cd** pour changer le système de fichiers afin de résoudre ce problème.



RÉFÉRENCES

Pages de manuel **lsblk(8)**, **mount(8)**, **umount(8)** et **lsof(8)**

► EXERCICE GUIDÉ

MONTAGE ET DÉMONTAGE DE SYSTÈMES DE FICHIERS

Au cours de cet exercice, vous allez vous entraîner à monter et démonter des systèmes de fichiers.

RÉSULTATS

Vous serez en mesure d'identifier et de monter un nouveau système de fichiers à un point de montage spécifié, puis de le démonter.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab fs-mount start**. La commande exécute un script de démarrage qui détermine si l'hôte, servera, est accessible sur le réseau. Le script crée également une partition sur le deuxième disque connecté à servera .

```
[student@workstation ~]$ lab fs-mount start
```

- 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Une nouvelle partition dotée d'un système de fichiers a été ajoutée au second disque (**/dev/vdb**) sur servera. Montez par UUID la nouvelle partition disponible sur le point de montage qui vient d'être créé, **/mnt/newspace**.

2.1. Utilisez la commande **su -** pour basculer vers **root**, étant donné que l'utilisateur **root** peut uniquement monter manuellement un périphérique.

```
[student@servera ~]$ su -
Password: redhat
[root@servera ~]#
```

2.2. Créez le répertoire **/mnt/newspace**.

```
[root@servera ~]# mkdir /mnt/newspace
```

2.3. Utilisez la commande **lsblk** avec l'option **-fp** pour détecter l'UUID du périphérique, **/dev/vdb1**.

```
[root@servera ~]# lsblk -fp /dev/vdb
NAME      FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vdb
└─/dev/vdb1 xfs   a04c511a-b805-4ec2-981f-42d190fc9a65
```

2.4. Montez le système de fichiers en utilisant l'UUID sur le répertoire **/mnt/newspace**.

Remplacez l'UUID par celui du disque **/dev/vdb1** provenant de la sortie de la commande précédente.

```
[root@servera ~]# mount UUID="a04c511a-b805-4ec2-981f-42d190fc9a65" /mnt/newspace
```

2.5. Vérifiez que le périphérique **/dev/vdb1** est monté sur le répertoire **/mnt/newspace**.

```
[root@servera ~]# lsblk -fp /dev/vdb
NAME      FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vdb
└─/dev/vdb1 xfs   a04c511a-b805-4ec2-981f-42d190fc9a65 /mnt/newspace
```

► 3. Accédez au répertoire **/mnt/newspace** et créez un répertoire, **/mnt/newspace/newdir**, avec un fichier vide, **/mnt/newspace/newdir/newfile**.

3.1. Choisissez le répertoire **/mnt/newspace**.

```
[root@servera ~]# cd /mnt/newspace
```

3.2. Créez le répertoire **/mnt/newspace/newdir**.

```
[root@servera newspace]# mkdir newdir
```

3.3. Créez un fichier vide, **/mnt/newspace/newdir/newfile**.

```
[root@servera newspace]# touch newdir/newfile
```

► 4. Démontez le système de fichiers monté dans le répertoire **/mnt/newspace**.

4.1. Utilisez la commande **umount** pour démonter **/mnt/newspace** pendant que le répertoire actif dans le shell est encore **/mnt/newspace**. La commande **umount** ne parvient pas à démonter le périphérique.

```
[root@servera newspace]# umount /mnt/newspace
umount: /mnt/newspace: target is busy.
```

4.2. Remplacez le répertoire actif dans le shell par **/root**.

```
[root@servera newspace]# cd
[root@servera ~]#
```

4.3. Maintenant, démontez avec succès **/mnt/newspace**.

```
[root@servera ~]# umount /mnt/newspace
```

► 5. Quittez servera.

```
[root@servera ~]# exit  
logout  
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation]$
```

Fin

Sur workstation, exéutez le script **lab fs-mount finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab fs-mount finish
```

L'exercice guidé est maintenant terminé.

LOCALISATION DE FICHIERS DANS LE SYSTÈME

OBJECTIFS

Au terme de cette section, vous serez en mesure de rechercher des fichiers dans des systèmes de fichiers montés à l'aide de **find** et de **locate**.

RECHERCHE DE FICHIERS

Un administrateur système a besoin d'outils pour rechercher dans un système de fichiers des fichiers qui correspondent à certains critères. Cette section aborde deux commandes qui servent à rechercher des fichiers dans la hiérarchie d'un système de fichiers.

- La commande **locate** recherche dans un index généré à l'avance des noms de fichiers ou des chemins d'accès, et renvoie les résultats instantanément.
- La commande **find** effectue des recherches de fichiers en temps réel dans la hiérarchie du système de fichiers en le parcourant intégralement.

LOCALISATION DES FICHIERS PAR LEUR NOM

La commande **locate** permet de trouver des fichiers en fonction du nom ou du chemin d'accès au fichier. Cette opération ne prend pas beaucoup de temps, car elle recherche ces informations dans la base de données **mlocate**. Cependant, cette base de données n'est pas mise à jour en temps réel et doit être fréquemment mise à jour pour que les résultats soient précis. Cela signifie aussi que **locate** ne trouvera pas les fichiers créés depuis la dernière mise à jour de la base de données.

La base de données de **locate** est automatiquement mise à jour tous les jours. Cependant, l'utilisateur **root** peut à tout moment lancer la commande **updatedb** pour forcer une mise à jour immédiate.

```
[root@host ~]# updatedb
```

La commande **locate** restreint les résultats pour les utilisateurs sans autorisation. Pour voir le nom du fichier résultant, l'utilisateur doit disposer du droit de recherche dans le répertoire dans lequel se trouve le fichier.

Recherchez les fichiers dont le nom ou le chemin d'accès contient **passwd** dans les arborescences de répertoires que **user** peut lire sur **host**.

```
[user@host ~]$ locate passwd
/etc/passwd
/etc/passwd-
/etc/pam.d/passwd
/etc/security/opasswd
/usr/bin/gpasswd
/usr/bin/grub2-mkpasswd-pbkdf2
```

```
/usr/bin/lppasswd
/usr/bin/passwd
...output omitted...
```

Les résultats sont renvoyés, même si le nom de fichier ou son chemin d'accès ne correspond que partiellement à la requête de recherche.

```
[root@host ~]# locate image
/etc/selinux/targeted-contexts/virtual_image_context
/usr/bin/grub2-mkimage
/usr/lib/sysimage
/usr/lib/dracut/dracut.conf.d/02-generic-image.conf
/usr/lib/firewalld/services/ovirt-imageio.xml
/usr/lib/grub/i386-pc/lnxboot.image
...output omitted...
```

L'option **-i** sert à effectuer une recherche sans tenir compte de la casse. Avec cette option, toutes les combinaisons possibles de lettres minuscules et majuscules peuvent correspondre à la recherche.

```
[user@host ~]$ locate -i messages
...output omitted...
/usr/share/vim/vim80/lang/zh_TW/LC_MESSAGES
/usr/share/vim/vim80/lang/zh_TW/LC_MESSAGES/vim.mo
/usr/share/vim/vim80/lang/zh_TW.UTF-8/LC_MESSAGES
/usr/share/vim/vim80/lang/zh_TW.UTF-8/LC_MESSAGES/vim.mo
/usr/share/vim/vim80/syntax/messages.vim
/usr/share/vim/vim80/syntax/msmessages.vim
/var/log/messages
```

L'option **-n** limite le nombre de résultats de recherche renvoyés par la commande **locate**. L'exemple suivant limite les résultats de recherche renvoyés par **locate** aux cinq premières correspondances :

```
[user@host ~]$ locate -n 5 snow.png
/usr/share/icons/HighContrast/16x16/status/weather-snow.png
/usr/share/icons/HighContrast/22x22/status/weather-snow.png
/usr/share/icons/HighContrast/24x24/status/weather-snow.png
/usr/share/icons/HighContrast/256x256/status/weather-snow.png
/usr/share/icons/HighContrast/32x32/status/weather-snow.png
```

RECHERCHE DE FICHIERS EN TEMPS RÉEL

La commande **find** localise les fichiers en effectuant une recherche en temps réel dans la hiérarchie du système de fichiers. C'est plus lent que **locate**, mais plus précis. Elle peut également rechercher des fichiers en fonction de critères autres que le nom du fichier, tels que les autorisations du fichier, son type, sa taille ou son heure de modification.

La commande **find** examine les fichiers du système de fichiers à l'aide du compte utilisateur qui a exécuté la recherche. L'utilisateur qui invoque la commande **find** doit disposer des permissions de lecture et d'écriture sur un répertoire pour pouvoir en examiner le contenu.

Le premier argument de la commande **find** est le répertoire dans lequel doit se faire la recherche. Si l'argument du répertoire est omis, **find** commence la recherche dans le répertoire actuel et cherche des correspondances dans tous les sous-répertoires.

Pour rechercher des fichiers par nom, utilisez l'option **-name FILENAME**. Avec cette option, **find** renvoie le chemin des fichiers correspondant exactement à *NOM DE FICHIER*. Par exemple, pour rechercher des fichiers nommés **sshd_config** à partir du répertoire **/**, exécutez la commande suivante :

```
[root@host ~]# find / -name sshd_config  
/etc/ssh/sshd_config
```



NOTE

Avec la commande **find**, les options de mot complet utilisent un tiret unique et les options suivent l'argument de nom de chemin, contrairement à la plupart des autres commandes Linux.

Des caractères génériques peuvent servir à rechercher un nom de fichier et à renvoyer tous les résultats qui correspondent partiellement. Quand on utilise des caractères génériques, il est important de mettre entre guillemets le nom de fichier à rechercher, afin d'éviter que le terminal interprète ces caractères génériques.

Dans l'exemple suivant, recherchez des fichiers en commençant dans le répertoire **/** et se terminant par **.txt** :

```
[root@host ~]# find / -name '*.txt'  
/etc/pki/nssdb/pkcs11.txt  
/etc/brltty/brl-lt-all.txt  
/etc/brltty/brl-mb-all.txt  
/etc/brltty/brl-md-all.txt  
/etc/brltty/brl-mn-all.txt  
...output omitted...
```

Pour rechercher des fichiers dans le répertoire **/etc/** qui contiennent le mot, **pass**, n'importe où dans leurs noms sur host, exécutez la commande suivante :

```
[root@host ~]# find /etc -name '*pass*'  
/etc/security/opasswd  
/etc/pam.d/passwd  
/etc/pam.d/password-auth  
/etc/passwd-  
/etc/passwd  
/etc/authselect/password-auth
```

Pour rechercher un nom de fichier donné sans tenir compte de la casse, utilisez l'option **-iname** suivie du nom de fichier à rechercher. Pour trouver des noms de fichiers contenant le texte **messages** sans tenir compte de la casse, dans le répertoire **/** sur host, exécutez la commande suivante :

```
[root@host ~]# find / -iname '*messages*'
...output omitted...
/usr/share/vim/vim80/lang/zh_CN.UTF-8/LC_MESSAGES
/usr/share/vim/vim80/lang/zh_CN.cp936/LC_MESSAGES
/usr/share/vim/vim80/lang/zh_TW/LC_MESSAGES
/usr/share/vim/vim80/lang/zh_TW.UTF-8/LC_MESSAGES
/usr/share/vim/vim80/syntax/messages.vim
/usr/share/vim/vim80/syntax/msmessages.vim
```

Recherche de fichiers en fonction de leur propriétaire ou de leurs permissions

La commande **find** permet de rechercher des fichiers d'après leur propriétaire ou leurs permissions. **-user** et **-group** sont des options utiles lors de la recherche par propriétaire, qui permettent de rechercher par nom, et **-uid** et **-gid**, qui recherchent par ID.

Recherchez des fichiers appartenant à **user** dans le répertoire **/home/user** sur host.

```
[user@host ~]$ find -user user
.
./.bash_logout
./.bash_profile
./.bashrc
./.bash_history
```

Recherchez des fichiers appartenant au groupe **user** dans le répertoire **/home/user** sur host.

```
[user@host ~]$ find -group user
.
./.bash_logout
./.bash_profile
./.bashrc
./.bash_history
```

Recherchez des fichiers appartenant à l'ID d'utilisateur **1000** dans le répertoire **/home/user** sur host.

```
[user@host ~]$ find -uid 1000
.
./.bash_logout
./.bash_profile
./.bashrc
./.bash_history
```

Recherchez des fichiers qui appartiennent à l'identifiant de groupe **1000** dans le répertoire **/home/user** sur host.

```
[user@host ~]$ find -gid 1000
.
./.bash_logout
./.bash_profile
./.bashrc
./.bash_history
```

Les options **-user** et **-group** peuvent être utilisées ensemble pour rechercher des fichiers lorsque le propriétaire du fichier et le propriétaire du groupe sont différents. L'exemple ci-dessous liste les fichiers appartenant à la fois à l'utilisateur **root** et affiliés au groupe **mail**.

```
[root@host ~]# find / -user root -group mail
/var/spool/mail
...output omitted...
```

L'option **-perm** sert à rechercher des fichiers avec un ensemble d'autorisations particulier. Les permissions peuvent être décrites par des valeurs octales, avec des combinaisons de **4, 2 et 1** pour la lecture, l'écriture et l'exécution. Les permissions peuvent être précédées d'un signe **/** ou **-**.

Une permission numérique précédée d'un **/** correspond aux fichiers dont au moins un bit de cette permission est activé pour l'utilisateur, le groupe ou les autres. Un fichier avec les permissions **r--r--r--** ne correspond pas à **/222**, contrairement à un fichier avec **rw-r--r--**. Le signe **-** devant une permission signifie que les trois instances de ce bit doivent toutes être activées. Aucun des exemples précédents ne correspondrait donc, contrairement à un fichier avec **rw-rw-rw-**.

Pour prendre un exemple plus complexe, la commande suivante correspond à n'importe quel fichier auquel l'utilisateur peut accéder en lecture, écriture et exécution, les membres du groupe en lecture et en écriture, et les autres en lecture seule :

```
[root@host ~]# find /home -perm 764
```

Pour désigner les fichiers auxquels l'utilisateur peut accéder au moins en écriture et en exécution, et le groupe au moins en écriture, et les autres au moins en lecture :

```
[root@host ~]# find /home -perm -324
```

Pour désigner les fichiers auxquels l'utilisateur peut accéder en lecture, ou le groupe au moins en lecture, ou les autres au moins en écriture :

```
[root@host ~]# find /home -perm /442
```

Utilisée avec les signes **/** ou **-**, une valeur de **0** agit comme un caractère générique, puisqu'elle signifie *une permission au moins égale à rien*.

Pour désigner les fichiers enregistrés dans le répertoire **/home/user** auxquels les autres ont au moins accès en lecture sur host, exécutez :

```
[user@host ~]$ find -perm -004
```

Recherchez tous les fichiers enregistrés dans le répertoire **/home/user** auxquels les autres peuvent accéder en écriture sur host.

```
[user@host ~]$ find -perm -002
```

Recherche de fichiers en fonction de la taille

Grâce à l'option **-size** suivie d'une valeur numérique et d'une unité, la commande **find** peut rechercher des fichiers qui correspondent à une taille spécifique. Utilisez la liste suivante comme unités avec l'option **-size** :

- **k** pour kilo-octet
- **M** pour mégaoctet
- **G** pour gigaoctet

L'exemple ci-dessous montre comment rechercher des fichiers d'une taille de 10 mégaoctets, arrondis.

```
[user@host ~]$ find -size 10M
```

Pour rechercher les fichiers d'une taille *supérieure* à 10 gigaoctets.

```
[user@host ~]$ find -size +10G
```

Pour lister tous les fichiers d'une taille *inférieure* à 10 kilo-octets.

```
[user@host ~]$ find -size -10k
```



IMPORTANT

Les modificateurs d'unité de l'option **-size** arrondissent tout à l'unité. Par exemple, la commande **find -size 1M** renvoie les fichiers de moins de 1 Mo, car elle arrondit tous les fichiers qui font jusqu'à 1 Mo.

Recherche de fichiers en fonction du temps de modification

L'option **-mmin**, suivie d'une durée en minutes, recherche tous les fichiers dont le contenu a été modifié il y a **n** minutes. L'horodatage du fichier est toujours arrondi. Il prend également en charge les valeurs fractionnaires lorsqu'il est utilisé avec des plages (**+n** et **-n**).

Pour rechercher tous les fichiers dont le contenu a été modifié il y a 120 minutes sur **host**, exécutez :

```
[root@host ~]# find / -mmin 120
```

Le modificateur **+** devant le nombre de minutes sert à rechercher tous les fichiers de **/** modifiés il y a plus de **n** minutes. Dans cet exemple, les fichiers modifiés il y a plus de 200 minutes sont listés.

```
[root@host ~]# find / -mmin +200
```

Le modificateur **-** modifie la requête de manière à rechercher tous les fichiers du répertoire **/** qui ont été modifiés il y a moins de **n** minutes. Dans cet exemple, les fichiers modifiés il y a moins de 150 minutes sont listés.

```
[root@host ~]# find / -mmin -150
```

Recherche de fichiers en fonction du type

L'option **-type** dans la commande **find** limite l'étendue de la recherche à un type de fichier donné. Utilisez la liste suivante pour passer les indicateurs requis afin de limiter la portée de la recherche :

- **f** pour un fichier standard
- **d** pour un répertoire (directory),
- **l** pour un lien symbolique,
- **b** pour un périphérique en mode bloc

Recherchez tous les répertoires dans le répertoire **/etc** sur host.

```
[root@host ~]# find /etc -type d
/etc
/etc/tmpfiles.d
/etc/systemd
/etc/systemd/system
/etc/systemd/system/getty.target.wants
...output omitted...
```

Recherchez tous les liens symboliques sur host.

```
[root@host ~]# find / -type l
```

Générez la liste de tous les périphériques de traitement par blocs dans le répertoire **/dev** sur host :

```
[root@host ~]# find /dev -type b
/dev/vda1
/dev/vda
```

L'option **-links** suivie d'un nombre recherche tous les fichiers vers lesquels pointe un certain nombre de liens physiques. Ce nombre peut être précédé du modificateur **++** pour rechercher les fichiers dont le nombre de liens en dur est supérieur au nombre indiqué. Si ce nombre est précédé du modificateur **--**, la recherche se limite à tous les fichiers dont le nombre de liens physiques est inférieur au nombre indiqué.

Recherchez tous les fichiers standard qui comptent plusieurs liens physiques sur host :

```
[root@host ~]# find / -type f -links +1
```



RÉFÉRENCES

Pages de manuel **locate(1)**, **updatedb(8)** et **find(1)**

► EXERCICE GUIDÉ

LOCALISATION DE FICHIERS DANS LE SYSTÈME

Au cours de cet exercice, vous allez trouver des fichiers spécifiques sur des systèmes de fichiers montés à l'aide des commandes **find** et **locate**.

RÉSULTATS

Vous serez en mesure de rechercher des fichiers en utilisant les commandes **find** et **locate**.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab fs-locate start**. La commande exécute un script de démarrage qui détermine si l'hôte, servera, est accessible sur le réseau.

```
[student@workstation ~]$ lab fs-locate start
```

- ▶ 1. Utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. Utilisez la commande **locate** pour rechercher des fichiers sur servera.

- 2.1. Bien que la base de données **locate** soit actualisée automatiquement et quotidiennement, assurez-vous qu'elle est à jour en lançant manuellement une mise à jour sur servera. Utilisez la commande **sudo updatedb** pour mettre à jour la base de données utilisée par la commande **locate**.

```
[student@servera ~]$ sudo updatedb
[sudo] password for student: student
[student@servera ~]$
```

- 2.2. Localisez le fichier de configuration **logrotate.conf**.

```
[student@servera ~]$ locate logrotate.conf
/etc/logrotate.conf
/usr/share/man/man5/logrotate.conf.5.gz
```

- 2.3. Localisez le fichier de configuration **networkmanager.conf**, en ignorant la casse.

```
[student@servera ~]$ locate -i networkmanager.conf
/etc/NetworkManager/NetworkManager.conf
/etc/dbus-1/system.d/org.freedesktop.NetworkManager.conf
/usr/share/man/man5/NetworkManager.conf.5.gz
```

- 3. Utilisez la commande **find** pour effectuer des recherches en temps réel sur **servera**, en respectant les exigences suivantes :
- Recherchez tous les fichiers dans le répertoire **/var/lib** appartenant à l'utilisateur **chrony**.
 - Listez tous les fichiers dans le répertoire **/var** appartenant à **root** et dont le propriétaire de groupe est **mail**.
 - Listez tous les fichiers du répertoire **/usr/bin** dont la taille est supérieure à 50 ko.
 - Recherchez tous les fichiers du répertoire **/home/student** qui n'ont pas été modifiés au cours des 120 dernières minutes.
 - Lisez tous les fichiers de périphériques de traitement par blocs dans le répertoire **/dev**.
- 3.1. Utilisez la commande **find** pour rechercher tous les fichiers dans le répertoire **/var/lib** appartenant à l'utilisateur **chrony**. Utilisez la commande **sudo** si les fichiers à l'intérieur du répertoire **/var/lib** appartiennent à **root**.

```
[student@servera ~]$ sudo find /var/lib -user chrony
[sudo] password for student: student
/var/lib/chrony
/var/lib/chrony/drift
```

- 3.2. Listez tous les fichiers dans le répertoire **/var** appartenant à **root** et affilés au groupe **mail**.

```
[student@servera ~]$ sudo find /var -user root -group mail
/var/spool/mail
```

- 3.3. Listez tous les fichiers du répertoire **/usr/bin** dont la taille est supérieure à 50 ko.

```
[student@servera ~]$ find /usr/bin -size +50k
/usr/bin/iconv
/usr/bin/locale
/usr/bin/localedef
/usr/bin/cmp
...output omitted...
```

- 3.4. Recherchez tous les fichiers du répertoire **/home/student** qui n'ont pas été modifiés au cours des 120 dernières minutes.

```
[student@servera ~]$ find /home/student -mmin +120
/home/student/.bash_logout
/home/student/.bash_profile
/home/student/.bashrc
...output omitted...
```

- 3.5. Lisez tous les fichiers de périphériques de traitement par blocs dans le répertoire **/dev**.

```
[student@servera ~]$ find /dev -type b
/dev/vdb
/dev/vda3
/dev/vda2
/dev/vda1
/dev/vda
```

► 4. Quittez servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

Fin

Sur workstation, exécutez le script **lab fs-locate finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab fs-locate finish
```

L'exercice guidé est maintenant terminé.

► OPEN LAB

ACCÈS AUX SYSTÈMES DE FICHIERS LINUX

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez monter un système de fichiers local et localiser des fichiers spécifiques sur ce système de fichiers.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Monter un système de fichiers.
- Générer un rapport d'utilisation d'un disque.
- Rechercher des fichiers dans le système de fichiers local.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab fs-review start**. La commande exécute un script de démarrage qui détermine si l'hôte, serverb, est accessible sur le réseau. Le script crée également une partition sur le deuxième disque connecté à serverb .

```
[student@workstation ~]$ lab fs-review start
```

1. Sur serverb en tant que root, identifiez l'UUID de **/dev/vdb1** et montez **/dev/vdb1** au moyen de son UUID sur le répertoire **/mnt/freespace**.
2. Générez un rapport d'utilisation de disque du répertoire **/usr/share** et enregistrez le résultat dans le fichier **/mnt/freespace/results.txt**.
3. Utilisez la commande **locate** pour trouver tous les fichiers de configuration **rsyslog.conf** et stocker le résultat dans le fichier **/mnt/freespace/search1.txt**.
4. Stockez le résultat de la recherche de tous les fichiers dans le répertoire **/usr/share** qui est supérieur à 50 Mo et inférieur à 100 Mo dans le fichier **/mnt/freespace/search2.txt**.
5. Quittez serverb.

Évaluation

Sur workstation, exécutez le script **lab fs-review grade** pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab fs-review grade
```

Finish (Terminer)

Sur workstation, exéutez le script **lab fs-review finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab fs-review finish
```

L'atelier est maintenant terminé.

► SOLUTION

ACCÈS AUX SYSTÈMES DE FICHIERS LINUX

LISTE DE CONTRÔLE DES PERFORMANCES

Au cours de cet atelier, vous allez monter un système de fichiers local et localiser des fichiers spécifiques sur ce système de fichiers.

RÉSULTATS

Vous devez pouvoir réaliser les tâches suivantes :

- Monter un système de fichiers.
- Générer un rapport d'utilisation d'un disque.
- Rechercher des fichiers dans le système de fichiers local.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab fs-review start**. La commande exécute un script de démarrage qui détermine si l'hôte, serverb, est accessible sur le réseau. Le script crée également une partition sur le deuxième disque connecté à serverb .

```
[student@workstation ~]$ lab fs-review start
```

1. Sur serverb en tant que root, identifiez l'UUID de **/dev/vdb1** et montez **/dev/vdb1** au moyen de son UUID sur le répertoire **/mnt/freespace**.
 - 1.1. Utilisez la commande **ssh** pour vous connecter à serverb en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Utilisez la commande **su -** pour basculer vers root.

```
[student@serverb ~]$ su -
Password: redhat
[root@serverb ~]#
```

- 1.3. Utilisez la commande **lsblk** pour déterminer l'UUID du périphérique **/dev/vdb1**.

```
[root@serverb ~]# lsblk -fp /dev/vdb
NAME      FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vdb
└─/dev/vdb1 xfs    a04c511a-b805-4ec2-981f-42d190fc9a65
```

1.4. Créez le répertoire **/mnt/freespace**.

```
[root@serverb ~]# mkdir /mnt/freespace
```

1.5. Montez le périphérique **/dev/vdb1** en utilisant l'UUID sur le répertoire **/mnt/freespace**.

```
[root@serverb ~]# mount UUID="a04c511a-b805-4ec2-981f-42d190fc9a65" /mnt/freespace
```

1.6. Vérifiez que le périphérique **/dev/vdb1** est monté sur le répertoire **/mnt/freespace**.

```
[root@serverb ~]# lsblk -fp /dev/vdb1
NAME      FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vdb
└─/dev/vdb1 xfs    a04c511a-b805-4ec2-981f-42d190fc9a65 /mnt/freespace
```

- Générez un rapport d'utilisation de disque du répertoire **/usr/share** et enregistrez le résultat dans le fichier **/mnt/freespace/results.txt**.

```
[root@serverb ~]# du /usr/share > /mnt/freespace/results.txt
```

- Utilisez la commande **locate** pour trouver tous les fichiers de configuration **rsyslog.conf** et stocker le résultat dans le fichier **/mnt/freespace/search1.txt**.

3.1. Utilisez la commande **updatedb** pour mettre à jour la base de données utilisée par **locate**.

```
[root@serverb ~]# updatedb
```

3.2. Localisez les fichiers de configuration **rsyslog.conf** et enregistrez le résultat dans le fichier **/mnt/freespace/search1.txt**.

```
[root@serverb ~]# locate rsyslog.conf > /mnt/freespace/search1.txt
```

- Stockez le résultat de la recherche de tous les fichiers dans le répertoire **/usr/share** qui est supérieur à 50 Mo et inférieur à 100 Mo dans le fichier **/mnt/freespace/search2.txt**.

```
[root@serverb ~]# find /usr/share -size +50M -size 100M > \
/mnt/freespace/search2.txt
```

- Quittez serverb.

```
[root@serverb ~]$ exit  
logout  
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation]$
```

Évaluation

Sur workstation, exécutez le script **lab fs-review grade** pour confirmer que l'atelier est réussi.

```
[student@workstation ~]$ lab fs-review grade
```

Finish (Terminer)

Sur workstation, exécutez le script **lab fs-review finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab fs-review finish
```

L'atelier est maintenant terminé.

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- Les périphériques de stockage sont représentés par un type de fichier spécial nommé périphérique bloc.
- La commande **df** renvoie l'espace disque total, l'espace disque utilisé ainsi que l'espace disque libre sur tous les systèmes de fichiers normaux montés.
- La commande **mount** permet à l'utilisateur **root** de monter manuellement un système de fichiers.
- Tous les processus doivent cesser d'accéder au point de montage pour pouvoir démonter correctement le périphérique.
- Les périphériques de stockage amovibles sont montés dans le répertoire **/run/media** lors de l'utilisation de l'environnement graphique.
- La commande **find** effectue une recherche en temps réel dans les systèmes de fichiers locaux pour trouver les fichiers qui correspondent aux critères de recherche.

CHAPITRE 16

ANALYSER LES SERVEURS ET OBTENIR UNE ASSISTANCE

PROJET

Examiner les problèmes et les résoudre dans l'interface de gestion Web, et obtenir une assistance auprès de Red Hat dans le cadre de leur résolution.

OBJECTIFS

- Activer l'interface de gestion de la console Web pour gérer et surveiller à distance les performances d'un serveur Red Hat Enterprise Linux.
- Décrire les principales ressources disponibles via le portail client Red Hat, et rechercher des informations dans la documentation Red Hat et dans la base de connaissances.
- Analyser les serveurs à la recherche de problèmes, résoudre ces derniers et vérifier que la solution fonctionne avec Red Hat Insights.

SECTIONS

- Analyse et gestion de serveurs distants (et exercice guidé)
- Obtenir de l'aide auprès du portail client Red Hat (et exercice guidé)
- Détection et résolution des problèmes avec Red Hat Insights (et quiz)

ANALYSE ET GESTION DE SERVEURS DISTANTS

OBJECTIFS

Après avoir terminé cette section, vous serez en mesure d'activer l'interface de gestion de la console Web pour gérer et surveiller à distance les performances d'un serveur Red Hat Enterprise Linux.

DESCRIPTION DE LA CONSOLE WEB

La console Web est une interface Web de gestion pour Red Hat Enterprise Linux 8, conçue pour gérer et surveiller vos serveurs. Elle est basée sur le service Open Source Cockpit.

Vous pouvez utiliser la console Web pour surveiller les journaux du système et afficher des graphiques des performances du système. En outre, vous pouvez utiliser votre navigateur Web pour modifier les paramètres à l'aide des outils graphiques de l'interface de la console Web, dont une session de terminal interactive entièrement opérationnelle.

ACTIVATION DE LA CONSOLE WEB

Red Hat Enterprise Linux 8 installe la console Web par défaut dans toutes les versions d'installation, à l'exception de l'installation minimale. Utilisez la commande suivante pour installer la console Web :

```
[user@host ~]$ sudo yum install cockpit
```

Activez et démarrez le service `cockpit.socket` qui exécute un serveur Web. Cette étape est nécessaire si vous devez vous connecter au système par le biais de l'interface Web.

```
[user@host ~]$ sudo systemctl enable --now cockpit.socket
Created symlink /etc/systemd/system/sockets.target.wants/cockpit.socket → /usr/
lib/systemd/system/cockpit.socket.
```

Si vous utilisez un profil de pare-feu personnalisé, vous devez ajouter le service `cockpit` à `firewalld` pour ouvrir le port 9090 dans le pare-feu :

```
[user@host ~]$ sudo firewall-cmd --add-service=cockpit --permanent
success
[user@host ~]$ sudo firewall-cmd --reload
success
```

CONNEXION À LA CONSOLE WEB

La console Web fournit son propre serveur Web. Lancez Firefox pour vous connecter à la console Web. Vous pouvez vous connecter avec le nom d'utilisateur et le mot de passe de n'importe quel compte local du système, y compris l'utilisateur `root`.

Ouvrez `https://servername:9090` dans votre navigateur Web, où `nom_serveur` est le nom d'hôte ou l'adresse IP de votre serveur. La connexion sera protégée par une session TLS. Le

CHAPITRE 16 | Analyser les serveurs et obtenir une assistance

système est installé avec un certificat TLS autosigné par défaut. Lors de la première connexion, votre navigateur Web affichera probablement un avertissement de sécurité. La page du manuel `cockpit-ws(8)` explique comment remplacer le certificat TLS par un certificat qui est correctement signé.

Saisissez votre nom d'utilisateur et votre mot de passe dans l'écran de connexion.

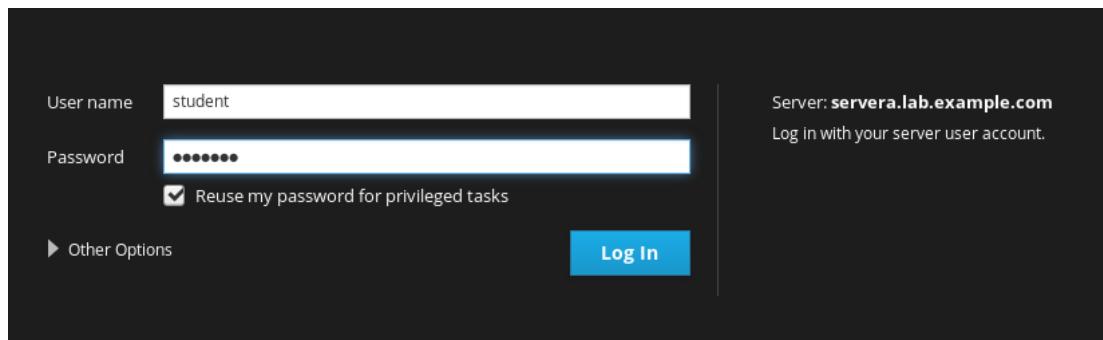


Figure 16.1: Écran de connexion à la console Web

Vous pouvez éventuellement cliquer sur l'option Reuse my password for privileged tasks. Cela vous permet d'exécuter des commandes avec des priviléges sudo et donc d'effectuer des tâches telles que la modification des informations système ou la configuration de nouveaux comptes.

Cliquez sur Log In.

La console Web affiche le nom d'utilisateur dans la partie droite de la barre de titre. Si vous sélectionnez l'option Reuse my password for privileged tasks, l'icône Privileged s'affiche à gauche du nom d'utilisateur.



Figure 16.2: Barre de titre de l'utilisateur privilégié

Si vous êtes connecté en tant qu'utilisateur non privilégié, l'icône Privileged n'est pas affichée.



Figure 16.3: Barre de titre de l'utilisateur non privilégié

MODIFICATION DES MOTS DE PASSE

Les utilisateurs privilégiés et non privilégiés peuvent modifier leurs propres mots de passe lorsqu'ils sont connectés à la console Web. Cliquez sur Accounts dans la barre de navigation de gauche. Cliquez sur l'étiquette de votre compte pour ouvrir la page des détails du compte.

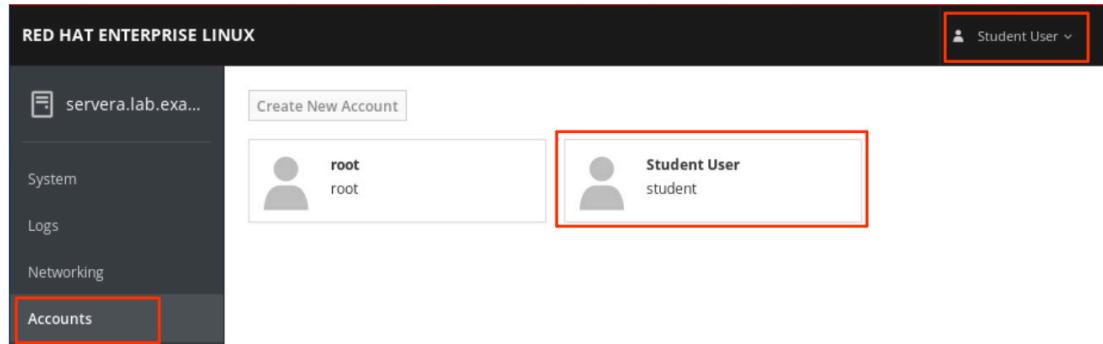


Figure 16.4: Affichage des comptes d'utilisateurs

En tant qu'utilisateur non privilégié, vous pouvez simplement définir ou réinitialiser votre mot de passe et gérer des clés SSH publiques. Pour définir ou réinitialiser votre mot de passe, cliquez sur le bouton Set Password.

The screenshot shows the Red Hat Enterprise Linux web interface. On the left, a sidebar lists 'System', 'Logs', 'Networking', 'Accounts' (which is selected and highlighted with a red box), 'Services', 'Applications', 'Diagnostic Reports', and 'Kernel Dump'. The main content area shows a 'Student User' account with the following details:

- Full Name:** Student User
- User Name:** student
- Roles:** Server Administrator
- Last Login:** 3/6/2019, 8:03:52 PM
- Access:** Lock Account (unchecked)
- Password:** **Set Password** (highlighted with a red box) | Force Change

Buttons at the top right include 'Terminate Session' and 'Delete'. A user icon in the top right corner is also highlighted with a red box.

Figure 16.5: Détails du compte d'utilisateur

Saisissez vos informations dans les champs Old Password, New Password et Confirm New Password. Cliquez sur Set pour activer le nouveau mot de passe.

The screenshot shows a 'Set Password' dialog box. It contains three input fields: 'Old Password' (containing '*****'), 'New Password' (containing '*****'), and 'Confirm New Password' (containing '*****'). Above the fields is a 'Set Password' button, which is highlighted with a red box. At the bottom right are 'Cancel' and 'Set' buttons, with the 'Set' button highlighted with a blue box.

Figure 16.6: Définition et réinitialisation des mots de passe

RÉSOLUTION DES PROBLÈMES À L'AIDE DE LA CONSOLE WEB

La console Web est un puissant outil de résolution des problèmes. Vous pouvez surveiller les statistiques de base du système en temps réel, examiner les journaux du système et basculer rapidement vers une session de terminal dans la console Web afin de collecter des informations supplémentaires à partir de l'interface de ligne de commande.

Contrôle des statistiques du système en temps réel

Cliquez sur System dans la barre de navigation de gauche pour afficher des informations sur le système, telles que son type de matériel, son système d'exploitation, son nom d'hôte, etc. Notez que si vous êtes connecté en tant qu'utilisateur non privilégié, toutes les informations sont visibles, mais vous n'êtes pas autorisé à modifier les valeurs. L'image suivante illustre la partie supérieure de la page d'options de menu System.

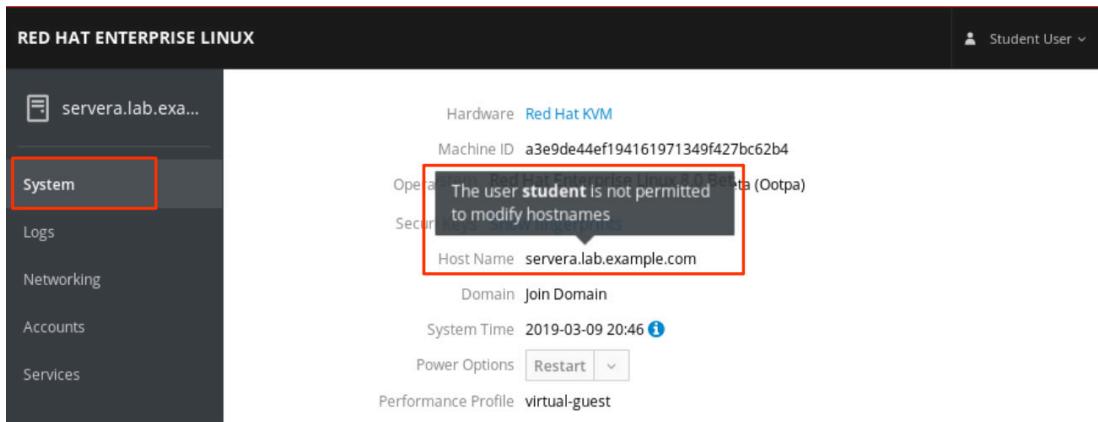


Figure 16.7: Page d'informations système de l'utilisateur non privilégié

Faites défiler la page d'informations System vers le bas pour afficher les graphiques suivants sur les performances actuelles du système : activité du processeur, utilisation de la mémoire, E/S de disque et utilisation du réseau.

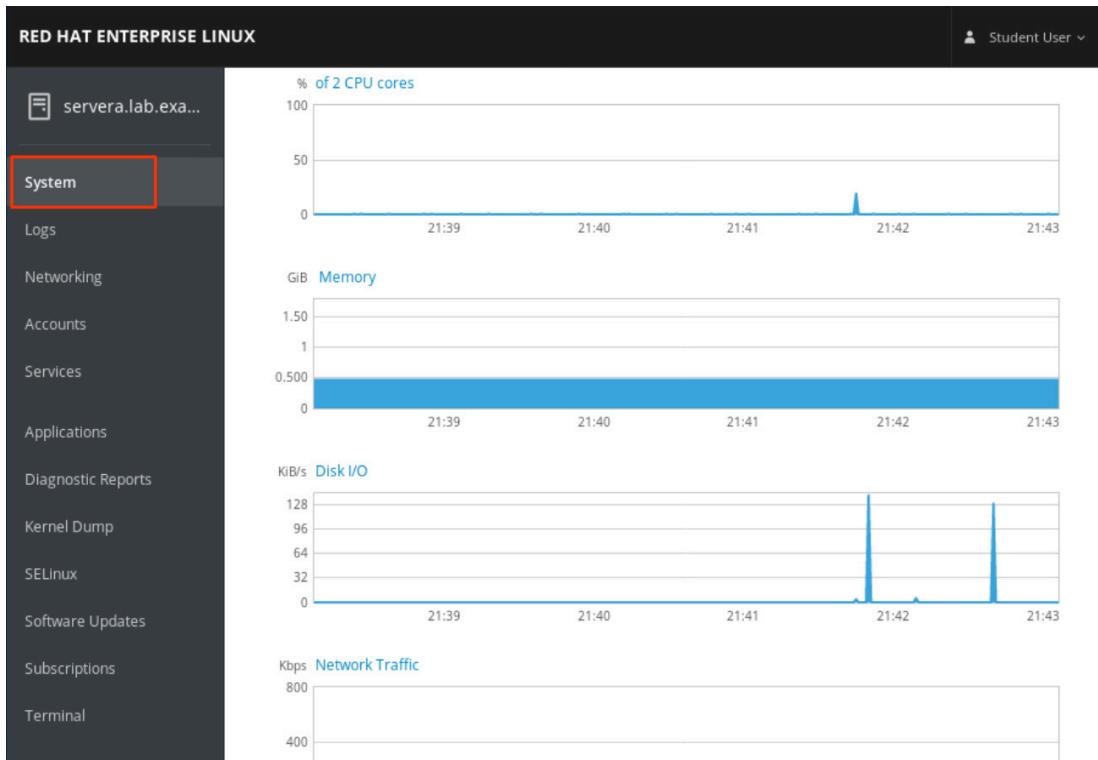


Figure 16.8: Mesures de performances du système de l'utilisateur non privilégié

Inspection et filtrage des événements Syslog

L'option Logs de la barre de navigation de gauche permet d'accéder aux outils d'analyse des journaux système. Vous pouvez utiliser les menus de la page pour filtrer les messages du journal en fonction d'une plage de dates de journalisation, d'un niveau de gravité ou de ces deux critères. La console Web utilise la date du jour comme valeur par défaut, mais vous pouvez cliquer sur le menu Date et spécifier une plage. De même, le menu Severity fournit des options allant de Everything à des conditions de gravité plus précises, telles que Alert and above, Debug and above, etc.

The screenshot shows the Red Hat Enterprise Linux interface with the 'Logs' section selected. The date dropdown is set to 'March 9, 2019'. The severity dropdown is set to 'Error and above'. A red box highlights the dropdown menu, which lists various severity levels: Everything, Only Emergency, Alert and above, Critical and above, Error and above, Warning and above, Notice and above, Info and above, and Debug and above.

Figure 16.9: Sélections du niveau de gravité des journaux

Cliquez sur une ligne pour afficher les détails du rapport de journal. Dans l'exemple ci-dessous, notez la première ligne qui indique un message de journal sudo.

The screenshot shows the Red Hat Enterprise Linux interface with the 'Logs' section selected. The date dropdown is set to 'March 9, 2019'. The severity dropdown is set to 'Error and above'. A red box highlights the second line of the log entry for March 9, 2019, which shows a 'student' user attempting to log in via sudo.

Figure 16.10: Sélection d'une entrée de journal

L'exemple ci-dessous montre les détails affichés lorsque vous cliquez sur la ligne sudo. Les détails du rapport comprennent notamment l'entrée de journal sélectionnée (sudo), la date, l'heure, la priorité et la fonction syslog de l'entrée de journal, ainsi que le nom d'hôte du système ayant signalé le message de journal, etc.

The screenshot shows the Red Hat Enterprise Linux cockpit interface. On the left, there is a sidebar with various navigation options: System, Logs (which is selected and highlighted with a red box), Networking, Accounts, Services, Applications, Diagnostic Reports, Kernel Dump, SELinux, Software Updates, Subscriptions, and Terminal. The main content area is titled 'Logs > Entry'. A search bar at the top of this area contains the word 'sudo', which is also highlighted with a red box. Below the search bar, the log entry details are displayed. The log entry is timestamped 'Sat Mar 09 2019 18:00:19 GMT-0600 (CST)'. The log message itself is: 'student : 3 incorrect password attempts ; TTY=unknown ; PWD=/run/user/1000 ; USER=root ; COMMAND=/bin/cockpit-bridge --privileged'. Below the log message, several metadata fields are listed: 'PRIORITY 1', 'SYSLOG_FACILITY 10', 'SYSLOG_IDENTIFIER sudo', '_BOOT_ID 8696e5d0639b4798862e107d045d71d7', '_CAP_EFFECTIVE 3fffffff', '_COMM sudo', '_GID 1000', '_HOSTNAME servera.lab.example.com', '_MACHINE_ID a3e9de44ef194161971349f427bc62b4', '_PID 19507', '_SELINUX_CONTEXT unconfined_u:unconfined_r:unconfined_t:ts0', '_SOURCE_REALTIME_TIMESTAMP 1552176019971260', and '_TRANSPORT syslog'.

Figure 16.11: Détails de l'entrée de journal

Exécution de commandes à partir d'une session de terminal

L'entrée Terminal dans la barre de navigation de gauche donne accès à une session de terminal entièrement opérationnelle dans l'interface de la console Web. Cela vous permet d'exécuter des commandes arbitraires pour gérer et utiliser le système, ainsi que pour effectuer des tâches qui ne sont pas prises en charge par les autres outils fournis par la console Web.

L'image ci-dessous présente des exemples de commandes courantes utilisées pour collecter des informations supplémentaires. Le fait de lister le contenu du répertoire **/var/log** fournit des rappels des fichiers journaux susceptibles de contenir de précieuses informations. La commande **id** fournit des informations rapides, telles que l'appartenance à un groupe, qui peuvent faciliter la résolution des problèmes de restriction d'accès aux fichiers. La commande **ps au** fournit un aperçu rapide des processus en cours d'exécution dans le terminal et de l'utilisateur associé au processus.

```

RED HAT ENTERPRISE LINUX
student@servera:~ [student@servera ~]$ ls /var/log
anaconda      cron-20190310      hawkey.log-20190310  rhsm
audit         dnf.librepo.log    lastlog           samba
boot.log      dnf.librepo.log-20190310  maillog          secure
boot.log-20190306  dnf.log        maillog-20190310  secure-20190310
btmp          dnf.rpm.log      messages          spooler
chrony        dnf.rpm.log-20190310  messages-20190310  spooler-20190310
cloud-init.log   dnf.rpm.log-20190310  private          sssd
cloud-init-output.log  firewalld     qemu-ga          tuned
cron          hawkey.log      README            wtmp
[student@servera ~]$
[student@servera ~]$ id
uid=1000(student) gid=1000(student) groups=1000(student),10(wheel) context=unconfined_u:unconfined_r:unconfined_t:s0
[student@servera ~]$
[student@servera ~]$ ps au
USER      PID %CPU %MEM   VSZ   RSS TTY      STAT START   TIME COMMAND
root     1255  0.0  0.1 225392  1996  ttyS0   Ss+ Mar05  0:00 /sbin/agetty -o -p -- \u --keep-baud 1
root     2048  0.0  0.2 236164  5068  tty1   Ss+ Mar05  0:00 -bash
student  19484  0.0  0.2 233932  4932  pts/0   Ss  Mar09  0:00 /bin/bash -i
student  21243  0.0  0.2 266920  3748  pts/0   R+  08:31  0:00 ps au
[student@servera ~]$ ■

```

Figure 16.12: Résolution des problèmes d'une session de terminal sans priviléges

Création de rapports de diagnostic

Un rapport de diagnostic est un ensemble d'informations de configuration, système et de diagnostic provenant d'un système Red Hat Enterprise Linux. Les données collectées dans le rapport généré sont notamment les journaux système et des informations de débogage pouvant être utilisées dans le cadre de la résolution des problèmes.

Connectez-vous à la console Web en tant qu'utilisateur privilégié. Cliquez sur Diagnostic Reports dans la barre de navigation de gauche pour ouvrir la page qui crée ces rapports. Cliquez sur Create Report pour générer un nouveau rapport de diagnostic.

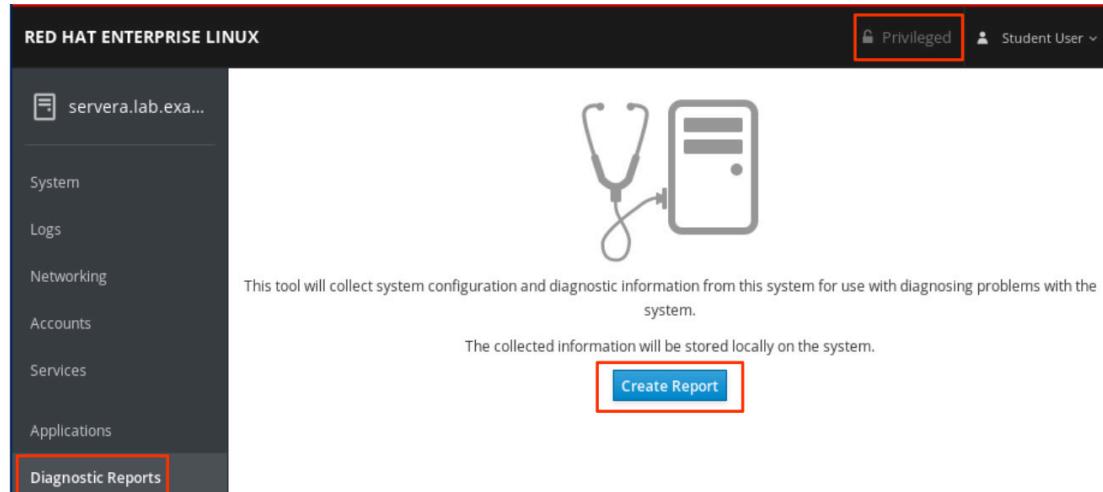


Figure 16.13: Crédit d'un rapport de diagnostic

L'interface affiche Done! une fois le rapport terminé. Cliquez sur Download report pour enregistrer le rapport.

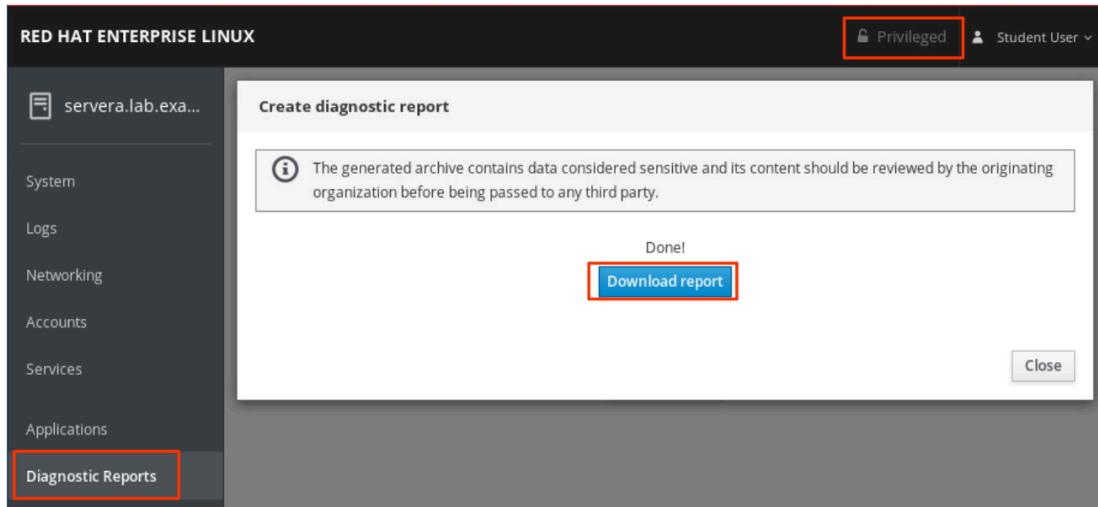


Figure 16.14: Téléchargement d'un rapport généré

Cliquez sur Save File pour terminer le processus.

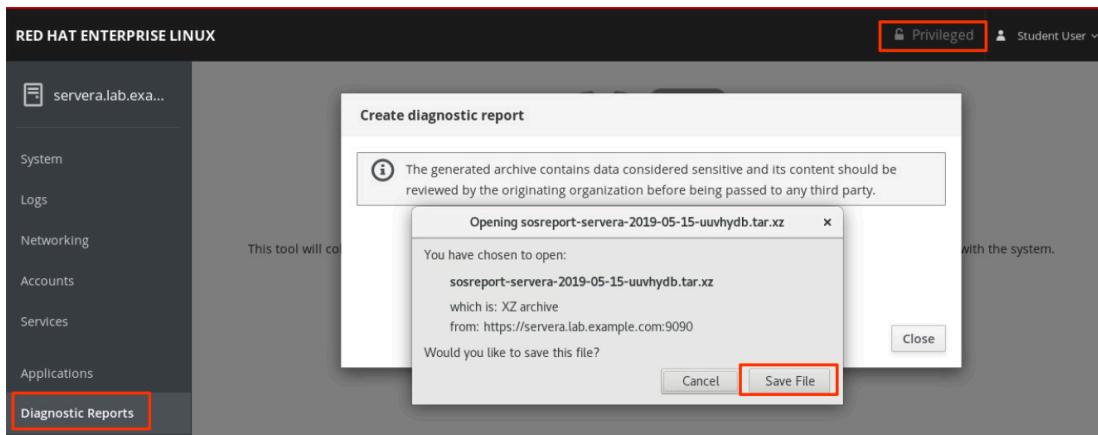


Figure 16.15: Enregistrement d'un rapport de diagnostic

Le rapport généré est enregistré dans le répertoire **Downloads** sur le système qui héberge le navigateur Web utilisé pour accéder à la console Web. Dans cet exemple, l'hôte est **workstation**.

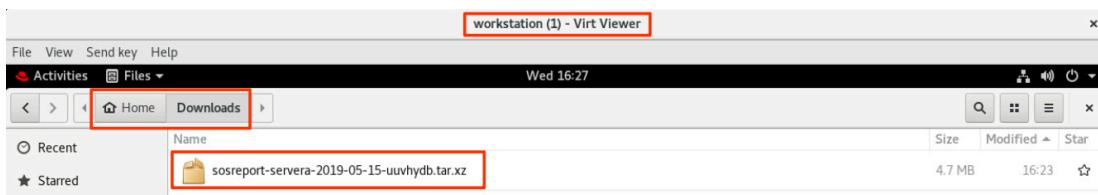


Figure 16.16: Accès à un rapport généré

GESTION DES SERVICES SYSTÈME AVEC LA CONSOLE WEB

En tant qu'utilisateur privilégié de la console Web, vous pouvez arrêter, démarrer, activer et redémarrer les services système. De plus, vous pouvez configurer des interfaces réseau et des services de pare-feu, administrer des comptes d'utilisateurs, etc. Les images suivantes illustrent des exemples courants d'utilisation des outils de gestion de la console Web.

Options d'alimentation du système

La console Web vous permet de redémarrer ou d'arrêter le système. Connectez-vous à la console Web en tant qu'utilisateur privilégié. Cliquez sur System dans la barre de navigation de gauche pour accéder aux options d'alimentation du système.

Sélectionnez l'option souhaitée dans le menu Power Options pour redémarrer ou arrêter un système.

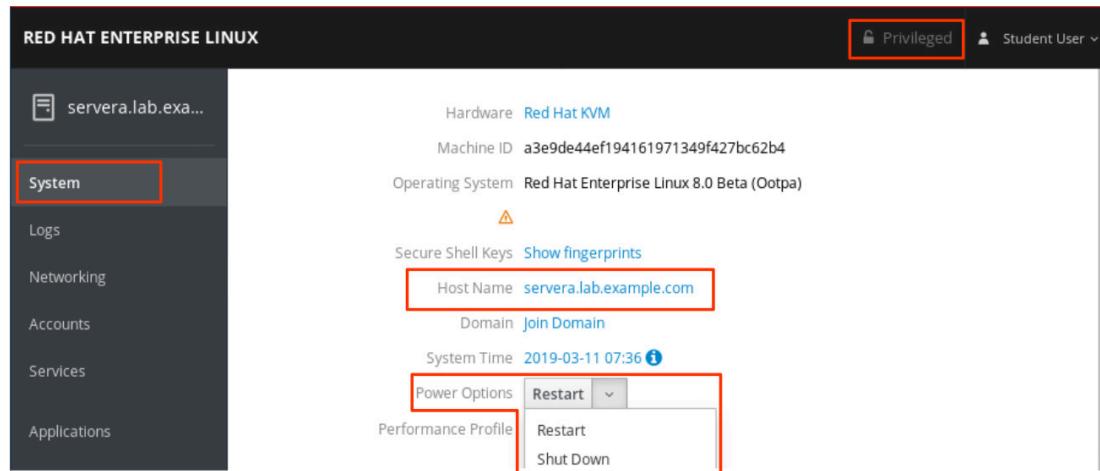


Figure 16.17: Options d'alimentation du système

Contrôle des services système en cours d'exécution

Vous pouvez démarrer, activer, désactiver et arrêter des services à l'aide d'outils graphiques dans la console Web. Cliquez sur Services dans la barre de navigation de gauche pour accéder à la page initiale des services de la console Web. Pour gérer les services, cliquez sur System Services dans le coin supérieur droit de la page initiale des services. Les services sont affichés dans des sections intitulées Enabled, Disabled et Static. Faites défiler la page pour sélectionner le service que vous souhaitez gérer.

Dans l'exemple ci-dessous, sélectionnez la ligne chronyd.service pour ouvrir la page de gestion des services.

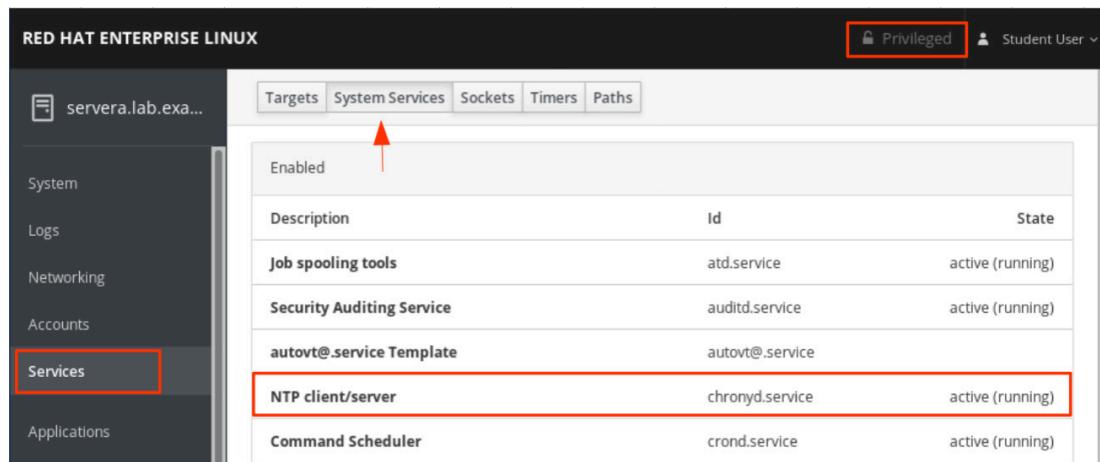


Figure 16.18: Services : vue initiale

Cliquez sur Stop, Restart ou Disable selon le cas pour gérer le service. Dans cette vue, le service est déjà en cours d'exécution. Le bouton Start n'est donc pas affiché. Des informations

supplémentaires relatives au service sont disponibles en cliquant sur l'un des liens en surbrillance ou en faisant défiler les journaux de service affichés sous la section de gestion des services.

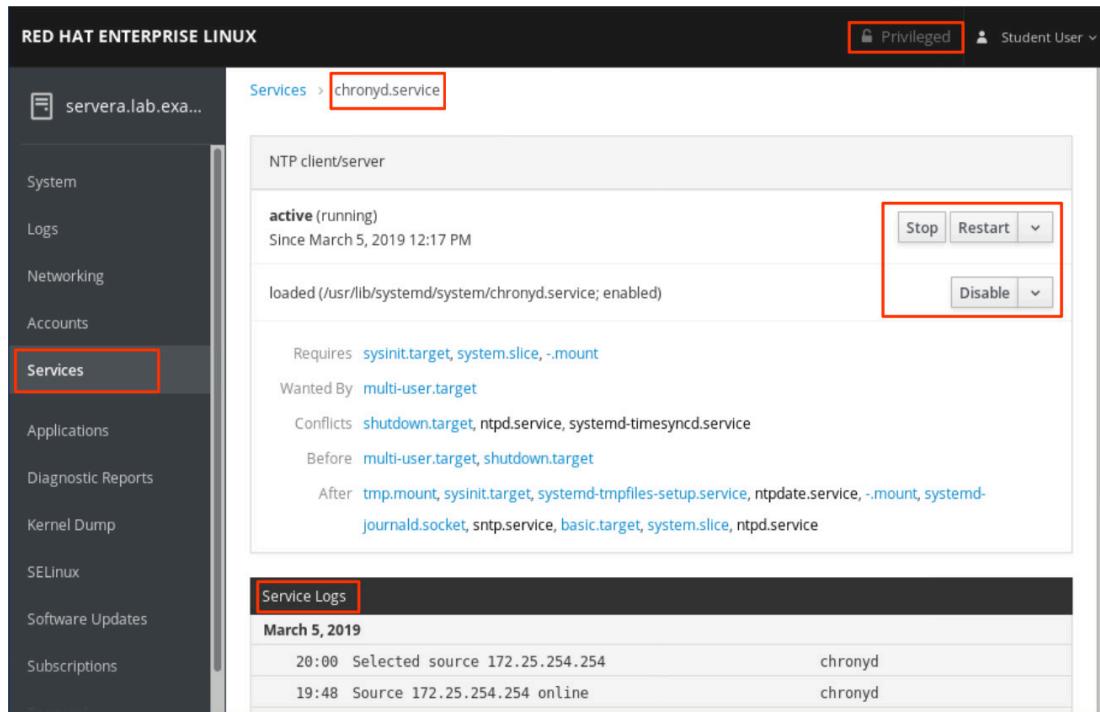


Figure 16.19: Services : détails du service et interface de gestion

Configuration des interfaces réseau et du pare-feu

Pour gérer les règles de pare-feu et les interfaces réseau, cliquez sur Networking dans la barre de navigation de gauche. L'exemple suivant montre comment collecter des informations sur les interfaces réseau et comment les gérer.

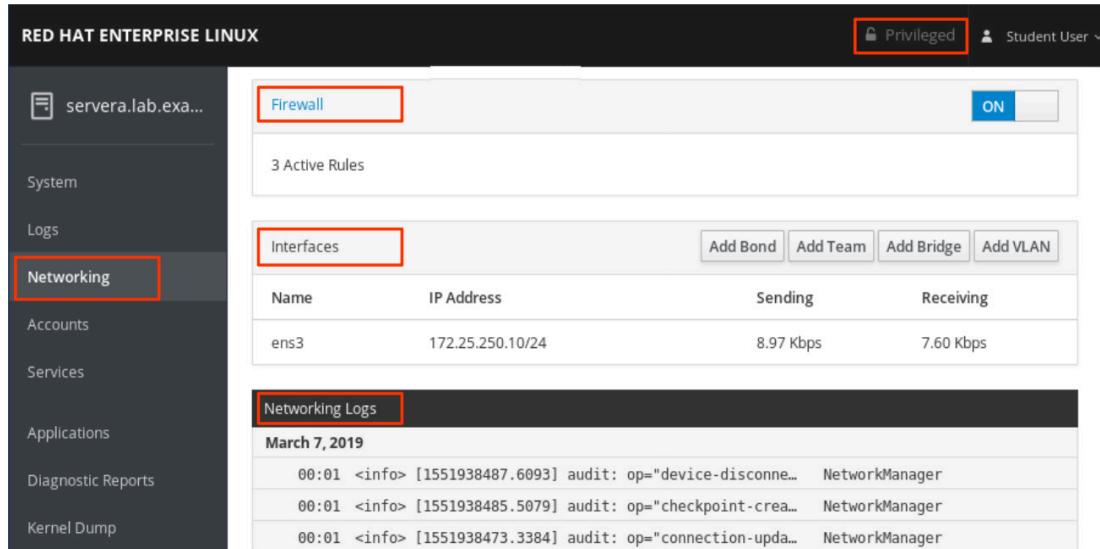


Figure 16.20: Networking : vue initiale

Cliquez sur le nom d'interface de votre choix dans la section Interfaces pour accéder à la page de gestion. Dans cet exemple, l'interface ens3 est sélectionnée.

Interfaces		Add Bond	Add Team	Add Bridge	Add VLAN
Name	IP Address	Sending	Receiving		
ens3	172.25.250.10/24	8.48 Kbps	4.75 Kbps		

Figure 16.21: Networking : Interfaces

La partie supérieure de la page de gestion affiche l'activité du trafic réseau pour le périphérique sélectionné. Faites défiler la page vers le bas pour afficher les paramètres de configuration et les options de gestion.

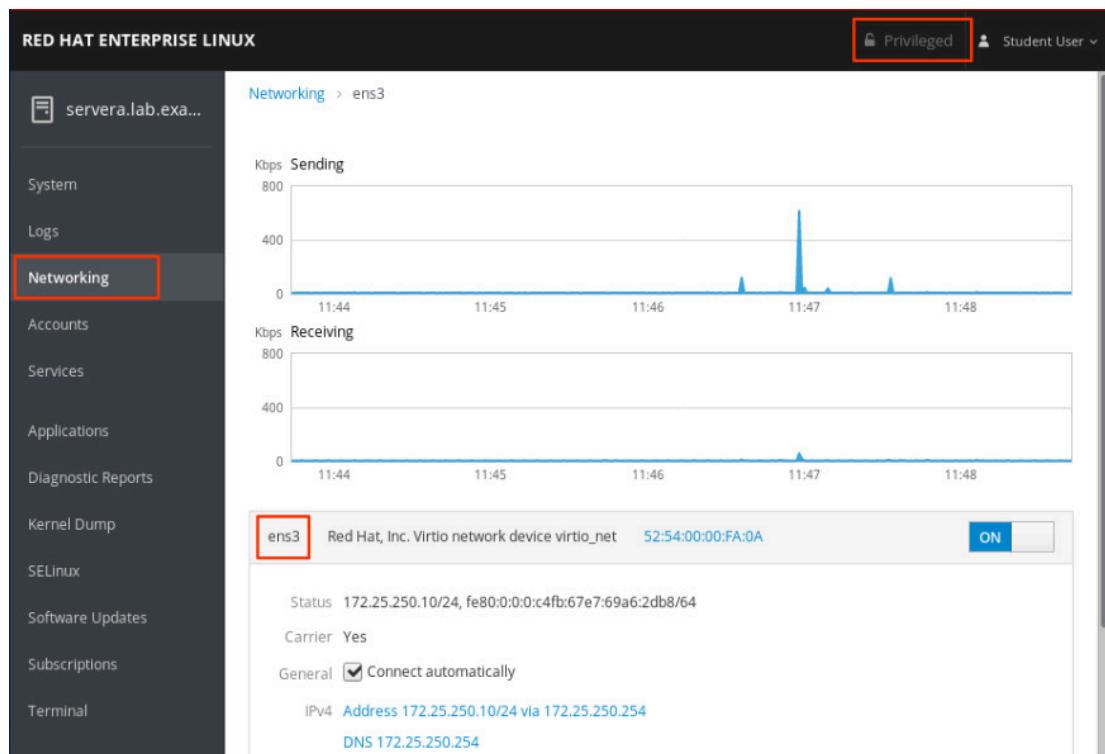


Figure 16.22: Networking : détails de l'interface

Pour modifier ou ajouter des options de configuration à une interface, cliquez sur les liens en surbrillance correspondant à la configuration souhaitée. Dans cet exemple, le lien IPv4 présente une seule adresse IP et un seul masque de réseau, **172.25.250.10/24**, pour l'interface réseau ens3. Pour ajouter une adresse IP à l'interface réseau ens3, cliquez sur le lien en surbrillance.

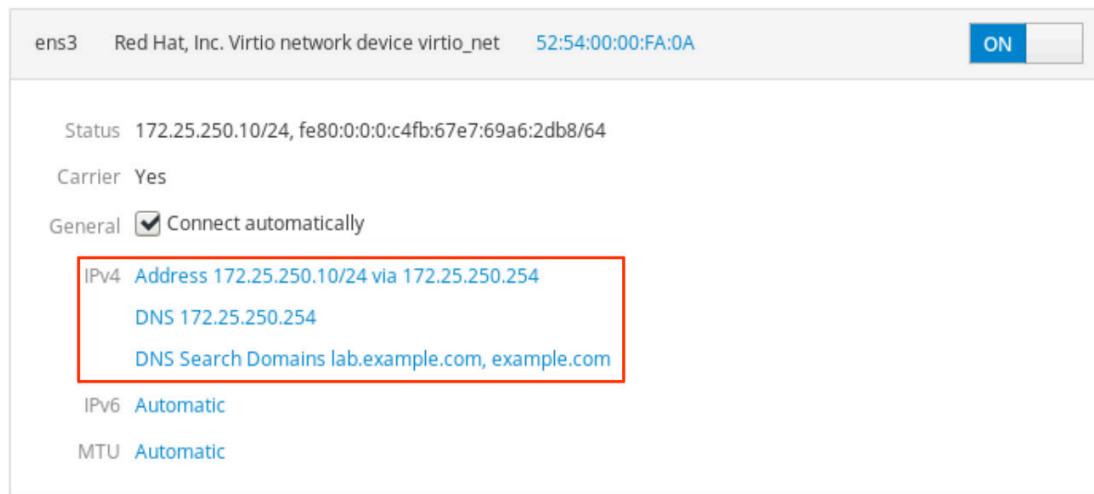


Figure 16.23: Networking : section de configuration de l'interface ens3

Cliquez sur + à droite de la liste Manual pour ajouter une adresse IP. Saisissez une adresse IP et un masque de réseau dans les champs appropriés. Cliquez sur Apply pour activer les nouveaux paramètres.

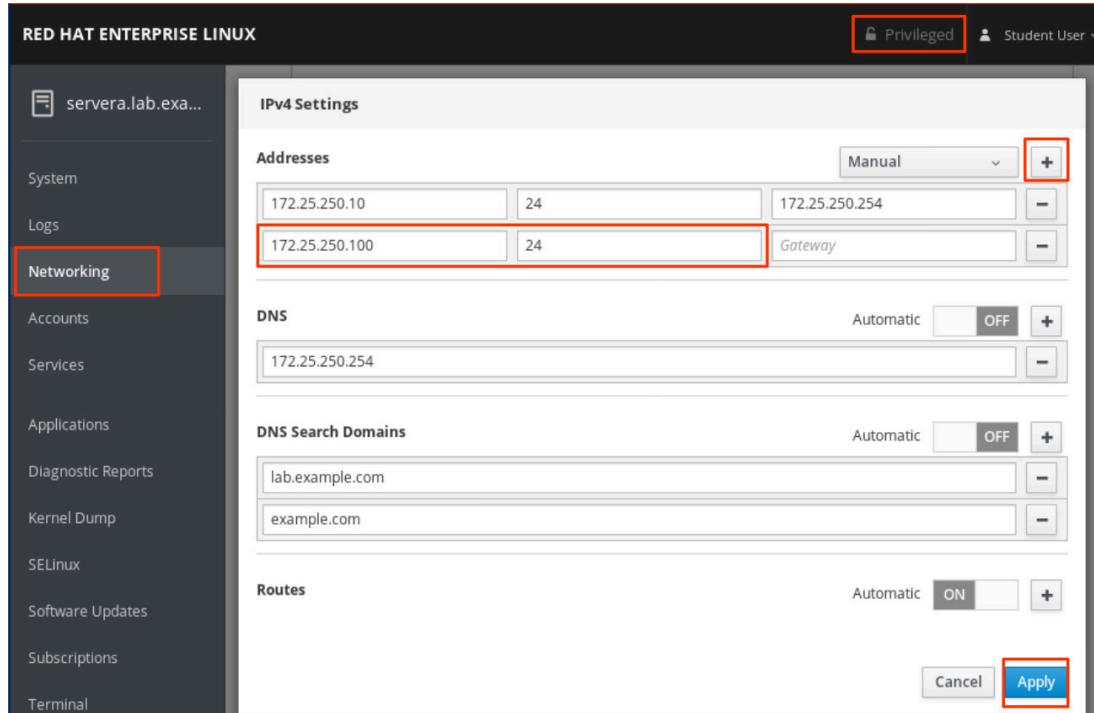


Figure 16.24: Ajout d'une adresse IP à une interface existante

L'affichage revient automatiquement à la page de gestion de l'interface où vous pouvez confirmer la nouvelle adresse IP.

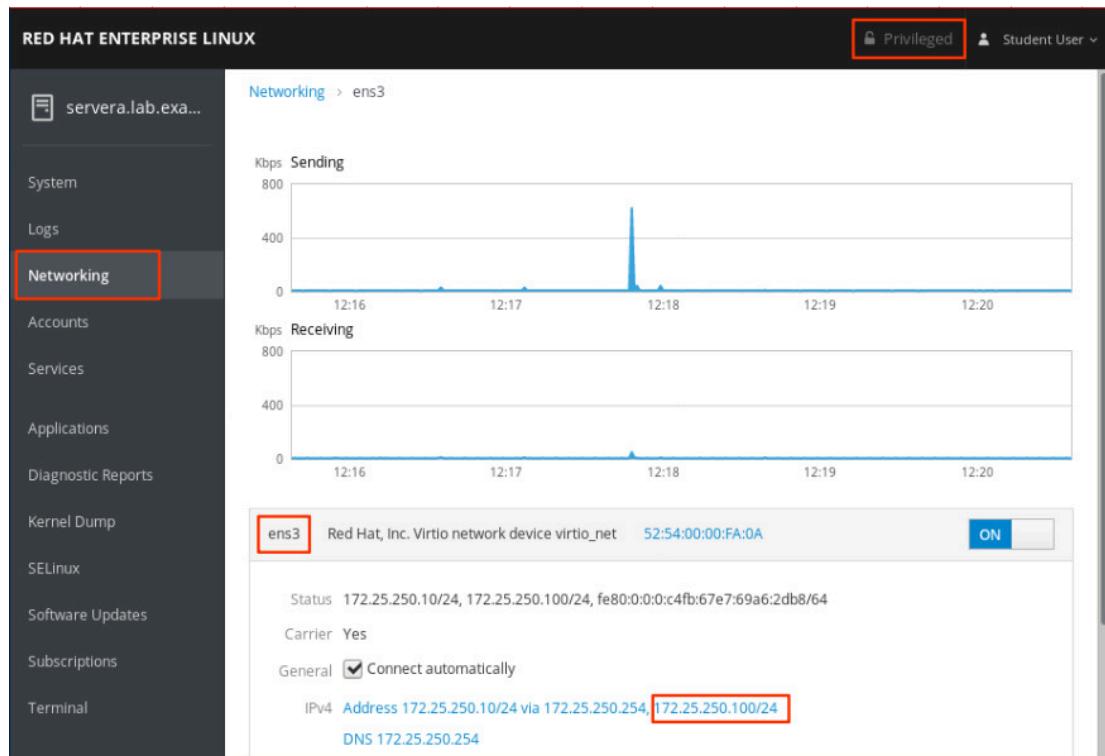


Figure 16.25: Confirmation de la nouvelle adresse IP

Administration de comptes d'utilisateurs

En tant qu'utilisateur privilégié, vous pouvez créer des comptes d'utilisateurs dans la console Web. Cliquez sur Accounts dans la barre de navigation de gauche pour afficher les comptes existants. Cliquez sur Create New Account pour ouvrir la page de gestion de compte.

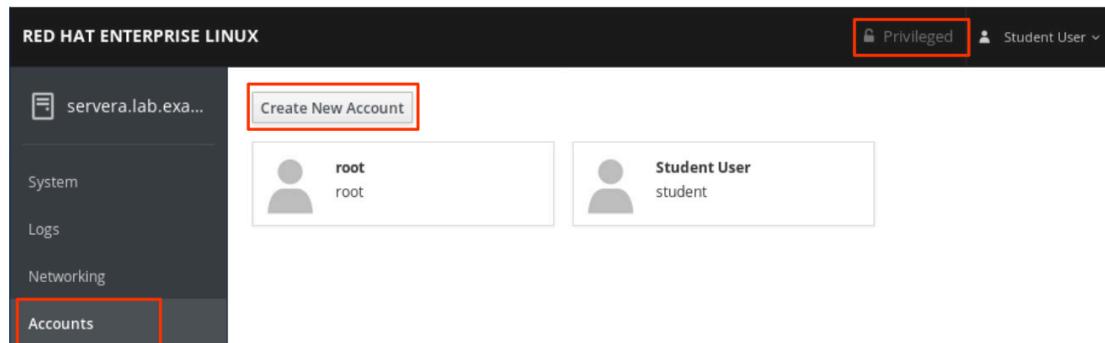


Figure 16.26: Comptes d'utilisateurs existants

Saisissez les informations du nouveau compte, puis cliquez sur Create.

Create New Account

Full Name: New User

User Name: nuser

Password: *****

Confirm: *****

Access: Lock Account

Create

Figure 16.27: Crédit d'un nouveau compte

L'affichage revient automatiquement à la page de gestion du compte où vous pouvez confirmer le nouveau compte d'utilisateur.

RED HAT ENTERPRISE LINUX

servera.lab.exa... **Privileged** Student User

System Logs Networking Accounts

Create New Account

New User
nuser

root
root

Student User
student

Figure 16.28: Page de gestion du compte



RÉFÉRENCES

Pages de manuel `cockpit(1)`, `cockpit-ws(8)` et `cockpit.conf(5)`

Pour plus d'informations, reportez-vous à la section *Managing systems using the Web Console* dans le guide d'utilisation de Cockpit pour gérer des systèmes dans *Red Hat Enterprise Linux 8.0* à l'adresse

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/managing_systems_using_the_web_console/

► EXERCICE GUIDÉ

ANALYSE ET GESTION DE SERVEURS DISTANTS

Au cours de cet exercice, vous allez activer la console Web sur un serveur et y accéder pour la gérer, ainsi que pour diagnostiquer et résoudre les problèmes.

RÉSULTATS

Vous serez en mesure d'utiliser la console Web pour surveiller les fonctions de base du système, inspecter les fichiers journaux, créer des comptes d'utilisateurs et accéder au terminal.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur `student` à `workstation` avec le mot de passe `student`.

Sur `workstation`, exécutez la commande **`lab support-cockpit start`**. Cette commande exécute un script de démarrage pour déterminer et modifier si les hôtes `servera` et `serverb` sont accessibles sur le réseau.

```
[student@workstation ~]$ lab support-cockpit start
```

- ▶ 1. Utilisez la commande `ssh` pour vous connecter à `servera` en tant qu'utilisateur `student`. Les systèmes sont configurés pour utiliser des clés SSH pour l'authentification. Par conséquent, aucun mot de passe n'est requis pour se connecter à `servera`.

```
[student@workstation ~]$ ssh student@servera
Activate the web console with: systemctl enable --now cockpit.socket

[student@servera ~]$
```

- ▶ 2. La console Web est déjà installée sur le système, mais elle n'est pas activée. Activez et démarrez le service `cockpit`.

- 2.1. Utilisez la commande **`systemctl enable --now cockpit.socket`** pour activer le service de console Web. Utilisez la commande `sudo` pour obtenir les priviléges du superutilisateur, et à l'invite, utilisez `student` comme mot de passe.

```
[student@servera ~]$ sudo systemctl enable --now cockpit.socket
[sudo] password for student: student
Created symlink /etc/systemd/system/sockets.target.wants/cockpit.socket -> /usr/
lib/systemd/system/cockpit.socket.
```

- 3. Sur workstation, ouvrez Firefox et connectez-vous à l'interface de la console Web sur `servera.lab.example.com`. Connectez-vous en tant qu'utilisateur `student` avec le mot de passe `student`.

3.1. Ouvrez Firefox et accédez à `https://servera.lab.example.com:9090`.

3.2. Acceptez le certificat autosigné en l'ajoutant comme exception.

3.3. Connectez-vous en tant qu'utilisateur `student` avec le mot de passe `student`.

Vous êtes maintenant connecté en tant qu'utilisateur normal, avec les priviléges minimaux.

- 4. Vérifiez votre autorisation actuelle dans l'interface de la console Web.

4.1. Cliquez sur Terminal dans la barre de navigation de gauche pour accéder au terminal.

Une session de terminal s'ouvre avec l'utilisateur `student` déjà connecté. Utilisez la commande `id` pour confirmer que l'exécution de la commande fonctionne dans le terminal intégré.

```
[student@servera ~]$ id  
uid=1000(student) gid=1000(student) groups=1000(student),10(wheel)  
context=unconfined_u:unconfined_r:unconfined_t:s0
```

4.2. Cliquez sur Accounts dans la barre de navigation de gauche pour gérer les utilisateurs.

Déplacez le pointeur de la souris sur le bouton Create New Account situé dans le coin supérieur gauche. Notez que l'utilisateur `student` n'est pas autorisé à créer des comptes.

4.3. Cliquez sur le lien Student User.

Dans la page des détails du compte de l'utilisateur `student`, notez que l'utilisateur est uniquement autorisé à définir un nouveau mot de passe ou à ajouter des clés publiques SSH autorisées.

4.4. Dans le coin supérieur droit, cliquez sur Student User → Log Out .

- 5. Accédez à la console Web avec des priviléges d'administrateur.

5.1. Reconnectez-vous à l'interface de la console Web en tant qu'utilisateur `student` avec le mot de passe `student` mais, cette fois, cochez la case `Reuse my password for privileged tasks`.

5.2. Pour vérifier l'accès administratif, assurez-vous que le libellé `Privileged` est affiché en regard du nom de compte `Student User` dans le coin supérieur droit de l'interface de la console Web.

- 6. Cliquez sur System dans la barre de navigation de gauche pour examiner les statistiques du système.

Cette page présente diverses statistiques de base du système d'exploitation, telles que la charge actuelle, l'utilisation du disque, les E/S de disque et le trafic réseau.

- 7. Pour inspecter les journaux système, cliquez sur Logs dans la barre de navigation de gauche.

CHAPITRE 16 | Analyser les serveurs et obtenir une assistance

Cette page affiche les journaux système `systemd`. Utilisez les boutons situés dans le coin supérieur gauche pour modifier le mode d'affichage des entrées de journal en fonction de la date et de la gravité des journaux.

- 7.1. Cliquez sur la liste Severity, puis sélectionnez Everything.
 - 7.2. En fonction du jour du mois, cliquez sur une entrée de journal de la liste. Une page de détails de l'entrée de journal s'ouvre avec des informations supplémentaires sur l'événement, telles que le nom d'hôte, le contexte SELinux ou le numéro PID du processus correspondant à l'entrée.
- **8.** Ajoutez une deuxième adresse IP à un périphérique d'interface réseau existant.

- 8.1. Cliquez sur Networking dans la barre de navigation de gauche.

Cette page affiche les détails de la configuration actuelle du réseau pour `servera`, ainsi que les statistiques en temps réel du réseau, la configuration du pare-feu et les entrées de journal relatives à la mise en réseau.

- 8.2. Faites défiler la page jusqu'à la section Interfaces, puis cliquez sur la ligne correspondant au nom de l'interface réseau.

Une page de détails affiche les statistiques en temps réel du réseau, ainsi que la configuration actuelle de cette interface réseau.

- 8.3. Cliquez sur le lien Address 172.25.250.10/24 via 172.25.250.254.

Une fenêtre IPv4 Settings s'ouvre pour vous permettre de modifier la configuration de l'interface réseau.

- 8.4. Dans la fenêtre IPv4 Settings, cliquez sur + en regard de Manual.

- 8.5. Dans la zone de texte Address, saisissez **172.25.250.99** comme deuxième adresse IP.

- 8.6. Dans la zone de texte Prefix length or Netmask, saisissez **24** en tant que valeur de masque de réseau.

- 8.7. Cliquez sur Apply pour enregistrer la nouvelle configuration réseau.

Comme vous pouvez le constater, la nouvelle configuration est immédiatement appliquée. La nouvelle adresse IP est visible dans la ligne IPv4.

- **9.** Créez un compte d'utilisateur.

- 9.1. Cliquez sur Accounts dans la barre de navigation de gauche.

- 9.2. Cliquez sur Create New Account.

- 9.3. Dans la fenêtre Create New Account, ajoutez les informations suivantes :

CHAMP	VALEUR
Full Name	manager1
User Name	manager1
Password	redh@t123
Confirm (Confirmer)	redh@t!23

9.4. Cliquez sur Create.

- 10. Accédez à une session Terminal dans la console Web pour ajouter l'utilisateur manager1 au groupe wheel.

10.1. Cliquez sur Terminal dans la barre de navigation de gauche.

10.2. Utilisez la commande **id manager1** pour afficher l'appartenance au groupe de l'utilisateur manager1.

```
[student@servera ~]$ id manager1
uid=1001(manager1) gid=1001(manager1) groups=1001(manager1)
[student@servera ~]$
```

10.3. Utilisez la commande **sudo usermod -aG wheel manager1** pour ajouter manager1 au groupe wheel.

```
[student@servera ~]$ sudo usermod -aG wheel manager1
[sudo] password for student: student
[student@servera ~]$
```

10.4. Utilisez la commande **id manager1** pour vérifier que manager1 fait partie du groupe wheel.

```
[student@servera ~]$ id manager1
uid=1001(manager1) gid=1001(manager1) groups=1001(manager1),10(wheel)
[student@servera ~]$
```

- 11. Activez et démarrez le service Kernel process accounting (psacct).

11.1. Cliquez sur Services dans la barre de navigation de gauche.

11.2. Faites défiler la liste System Services vers le bas jusqu'à la section Disabled. Recherchez le lien Kernel process accounting et cliquez dessus.

11.3. Cliquez sur Enable.

11.4. Cliquez sur Start.

Le service est maintenant activé et démarré.

- 12. Déconnectez-vous de l'interface de la console Web.

- 13. Déconnectez-vous de servera.

```
[student@servera ~]$ exit
[student@workstation ~]$
```

Fin

Sur workstation, exécutez le script **lab support-cockpit finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab support-cockpit finish
```

L'exercice guidé est maintenant terminé.

OBTENIR DE L'AIDE AUPRÈS DU PORTAIL CLIENT RED HAT

OBJECTIFS

Après avoir terminé cette section, vous serez en mesure de décrire les principales ressources disponibles via le portail client Red Hat et de les utiliser pour rechercher des informations dans la documentation Red Hat et dans la base de connaissances.

ACCÈS AUX RESSOURCES D'ASSISTANCE SUR LE PORTAIL CLIENT RED HAT

Le portail client Red Hat (<https://access.redhat.com>) permet aux clients d'accéder à la documentation, à des téléchargements, à des outils et à une expertise technique. Dans la base de connaissances, les clients peuvent chercher des solutions, des FAQ et des articles. Plusieurs options s'offrent à vous à partir du portail client :

- Accéder à la documentation officielle des produits.
- Envoyer et gérer des tickets d'assistance.
- Gérer les droits d'accès et abonnements aux logiciels.
- Obtenir des téléchargements, mises à jour et évaluations de logiciels.
- Consulter les outils qui peuvent vous aider à optimiser la configuration de vos systèmes.

Certaines parties du site sont accessibles par tous, d'autres sont uniquement disponibles pour les clients qui disposent d'un abonnement en cours de validité. Pour obtenir de l'aide sur l'accès au portail client, rendez-vous à l'adresse suivante : <https://access.redhat.com/help/>.

ACCÈS AU PORTAIL CLIENT

Vous pouvez vous rendre sur le portail client Red Hat à l'aide d'un navigateur Web. Cette section présente la visite guidée Tour the Customer Portal, accessible à l'adresse suivante : <https://access.redhat.com/start>.

Cette visite guidée est un outil très pratique pour découvrir tout ce que le portail peut vous offrir et apprendre à tirer pleinement parti de votre abonnement à Red Hat. Une fois connecté au portail client Red Hat, cliquez sur Tour the Customer Portal.



[Tour the Customer Portal](#)

Figure 16.29: Visite guidée du portail client

La fenêtre WELCOME TO THE RED HAT CUSTOMER PORTAL! s'ouvre. Elle vous propose deux options : CLOSE et NEXT. Cliquez sur NEXT pour commencer la visite guidée. Cette fenêtre est la première d'une série mettant en évidence différentes parties de l'interface.

Barre de navigation supérieure

Les trois premiers arrêts de la visite guidée du portail client se trouvent dans la barre de navigation supérieure du site Web du portail client Red Hat :

SUBSCRIPTIONS DOWNLOADS CONTAINERS SUPPORT CASES

Figure 16.30: Barre de navigation supérieure

Subscriptions ouvre une nouvelle page sur laquelle vous pouvez gérer vos systèmes enregistrés, ainsi que l'utilisation de vos abonnements et droits d'accès. Cette section rassemble des informations sur les errata applicables et vous permet de créer des *clés d'activation* que vous pouvez utiliser lors de l'enregistrement des systèmes pour vous assurer qu'ils obtiennent les droits d'accès des abonnements appropriés. Notez que si vous faites partie d'une organisation, votre administrateur peut limiter votre accès à cette page.

Downloads ouvre une nouvelle page qui vous permet d'accéder aux téléchargements de vos produits et de demander des droits d'évaluation pour les produits pour lesquels vous n'en possédez pas.

Support Cases ouvre une nouvelle page qui vous permet de créer, de suivre et de gérer vos dossiers d'assistance via le système de gestion des dossiers, en supposant que votre organisation ait autorisé ce niveau d'accès.

Votre nom est le titre de User Menu, qui vous permet de gérer votre compte, les comptes pour lesquels vous occupez le rôle d'administrateur d'entreprise, votre profil personnel et les options de notification (par e-mail) disponibles pour le nouveau contenu.

L'icône représentant le globe terrestre vous permet d'ouvrir le menu Select Your Language afin de spécifier vos préférences linguistiques pour le portail client.

Menus des rubriques

Sous la barre de navigation supérieure de la page principale du portail client, vous trouverez des menus permettant de parcourir les quatre catégories de ressources principales disponibles sur le site.

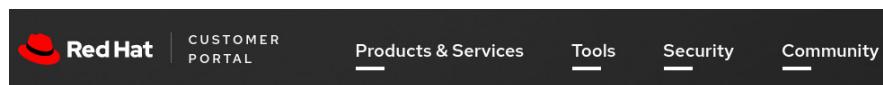


Figure 16.31: Menus des ressources

Products & Services donne accès aux *hubs de produits*. Ces pages permettent d'accéder à des évaluations, présentations et guides de mise en route spécifiques à certains produits, ainsi qu'à d'autres informations d'assistance sur les produits. Vous pouvez également accéder à la documentation des produits Red Hat, à des liens directs vers la base de connaissances d'articles d'assistance, ainsi qu'à des informations sur les politiques d'assistance et les moyens de contacter le support technique Red Hat.

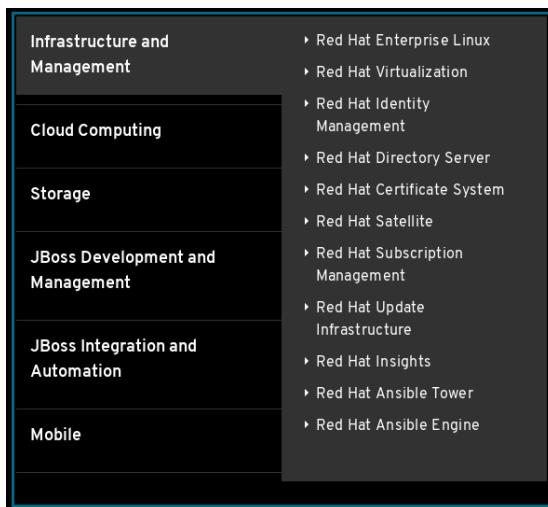


Figure 16.32: Produits et services

Le menu Tools fournit des liens vers les outils qui vous aident à utiliser efficacement les produits Red Hat. Avec la section Solution Engine, vous disposez d'une méthode efficace pour rechercher rapidement des solutions à vos problèmes, en fonction du produit, et d'ouvrir un ticket d'assistance si vous ne trouvez pas de solution satisfaisante. La section Customer Portal Labs fournit un ensemble d'applications Web et d'outils pour vous aider à améliorer les performances, à diagnostiquer les problèmes, à identifier les failles de sécurité et à optimiser vos configurations. Product Life Cycle Checker, par exemple, vous permet de sélectionner un produit spécifique et d'afficher son calendrier de cycle de vie d'assistance. Un autre outil, Rescue Mode Assistant, vous aide à réinitialiser le mot de passe racine d'un système, à générer des rapports de diagnostic ou à corriger des problèmes affectant les systèmes de fichiers lors de la phase de démarrage. Il existe encore bien d'autres outils sur ce site.



Figure 16.33: Menu Tools du portail client

La section Security vous donne accès au *Centre de sécurité des produits Red Hat* à l'adresse <https://access.redhat.com/security/>. Cette section fournit également des informations sur les problèmes de sécurité importants, un accès à la base de données CVE de Red Hat, au canal de sécurité du blog Red Hat et aux processus de réponses de sécurité de Red Hat, ainsi que sur la manière dont nous évaluons les problèmes et nous les résolvons.

Enfin, le menu Community est un endroit où les experts, les partenaires et les clients Red Hat peuvent communiquer et collaborer. Vous y trouverez également des forums de discussion, des blogs et des informations sur les événements à venir dans votre région.



NOTE

Pour tout savoir sur le portail client, vous devez terminer toutes les étapes de la visite guidée à l'adresse Prise en main de Red Hat [<https://access.redhat.com/start>], y compris les sections sur la personnalisation de l'interface du portail client et l'énumération des avantages liés à l'abonnement Red Hat. Pour accéder à cette page, au moins un abonnement doit être actif sur votre compte du portail client.

RECHERCHE DANS LA BASE DE CONNAISSANCES AVEC L'UTILITAIRE RED HAT SUPPORT TOOL

L'utilitaire Red Hat Support Tool, **redhat-support-tool**, fournit une interface textuelle qui vous permet d'effectuer des recherches dans les articles de la base de connaissances et déposer des dossiers d'assistance sur le portail client à partir de la ligne de commande de votre système. Cet outil est dépourvu d'interface graphique. Étant donné qu'il interagit avec le portail client Red Hat, un accès Internet est requis. Exécutez la commande **redhat-support-tool** à l'aide d'une connexion par SSH ou de terminal.

Il est possible d'utiliser la commande **redhat-support-tool** dans un mode interactif ou en l'appelant sous la forme d'une commande avec des options et des arguments. La syntaxe de l'outil est identique pour les deux méthodes. Par défaut, le programme est lancé en mode interactif. Utilisez la sous-commande **help** pour afficher toutes les commandes disponibles. Le mode interactif prend en charge la saisie semi-automatique via la touche de tabulation et la possibilité d'appeler des programmes dans le shell parent.

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

Lors du premier appel, **redhat-support-tool** invite l'abonné au portail client Red Hat à saisir ses identifiants de connexion. Pour éviter à l'utilisateur de devoir fournir ces informations de manière répétitive, l'outil demande à conserver les identifiants du compte dans le répertoire personnel de l'utilisateur (`~/.redhat-support-tool/redhat-support-tool.conf`). Si tous les problèmes sont signalés via un compte spécifique du portail client Red Hat, l'option **--global** peut enregistrer les informations de compte dans `/etc/redhat-support-tool.conf`, avec une autre configuration à l'échelle du système. La commande **config** de l'outil modifie ses paramètres de configuration.

La commande **redhat-support-tool** permet aux abonnés de rechercher et d'afficher le contenu de la base de connaissances à partir du portail client Red Hat. La base de connaissances permet les recherches par mot-clé, à l'instar de la commande **man**. Vous pouvez saisir des codes d'erreur, la syntaxe provenant de fichiers journaux ou n'importe quelle combinaison de mots-clés pour afficher une liste de documents avec une solution appropriée.

L'exemple ci-dessous fait la démonstration d'une configuration initiale et d'une recherche de base :

```
[user@host ~]$ redhat-support-tool
Welcome to the Red Hat Support Tool.
Command (? for help): search How to manage system entitlements with subscription-
manager
Please enter your RHN user ID: subscriber
Save the user ID in /home/student/.redhat-support-tool/redhat-support-tool.conf
(y/n): y
Please enter the password for subscriber: password
Save the password for subscriber in /home/student/.redhat-support-tool/redhat-
support-tool.conf (y/n): y
```

Après avoir demandé à l'utilisateur la configuration utilisateur requise, l'outil poursuit avec la requête d'origine :

```
Type the number of the solution to view or 'e' to return to the previous menu.
1 [ 253273:VER] How to register and subscribe a system to the Red Hat Customer
    Portal using Red Hat Subscription-Manager
2 [ 265523:VER] Enabling or disabling a repository using Red Hat Subscription
    Management
3 [ 100423:VER] Why does subscription-manager list return: "No Installed
    Products found" ?
...output omitted...
Select a Solution: 1
```

Sélectionnez le numéro d'article 1 comme ci-dessus. Vous êtes alors invité à sélectionner la section du document à lire. Pour terminer, utilisez la touche **Q** pour quitter la section dans laquelle vous vous trouvez, ou utilisez-la à plusieurs reprises pour quitter la commande **redhat-support-tool**.

```
Select a Solution: 1

Type the number of the section to view or 'e' to return to the previous menu.
1 Title
2 Issue
3 Environment
4 Resolution
5 Display all sections
End of options.
Section: 1

Title
=====
How to register and subscribe a system to the Red Hat Customer Portal using Red
    Hat Subscription-Manager
URL:      https://access.redhat.com/solutions/253273
Created On:  None
Modified On:  2017-11-29T15:33:51Z

(END) q
Section:
Section: q

Select a Solution: q

Command (? for help): q
[user@hosts ~]#
```

Accès aux articles de la base de connaissances par identifiant de document

Localisez directement les articles en ligne en utilisant la commande **kb** de l'outil avec l'identifiant de document de la Base de connaissances. Un document renvoyé défile à l'écran sans pagination, mais vous pouvez le rediriger vers un fichier pour l'enregistrer et utiliser **less** pour le faire défiler d'un écran à la fois.

```
[user@host ~]$ redhat-support-tool kb 253273

Title
=====
How to register and subscribe a system to the Red Hat Customer Portal using Red
Hat Subscription-Manager
URL:      https://access.redhat.com/solutions/253273
Created On: None
Modified On: 2017-11-29T15:33:51Z

Issue
=====
* How to register a new `Red Hat Enterprise Linux` system to the Customer Portal
using `Red Hat Subscription-Manager`
...output omitted...
```

GESTION DES DOSSIERS D'ASSISTANCE AVEC L'UTILITAIRE RED HAT SUPPORT TOOL

L'un des avantages de l'abonnement à un produit est l'accès à l'assistance technique via le portail client Red Hat. Selon le niveau d'assistance offert par l'abonnement, Red Hat peut être contacté via des outils en ligne ou par téléphone. Pour plus d'informations, voir https://access.redhat.com/site/support/policy/support_process.

Préparation d'un rapport de bogue

Avant de contacter l'assistance Red Hat, rassemblez les informations pertinentes pour effectuer un rapport de bogue.

Définissez le problème. Soyez en mesure d'exposer clairement le problème et ses symptômes. Soyez aussi précis que possible. Détaillez les étapes qui permettront de reproduire le problème.

Rassemblez des informations complémentaires. Quel est le produit affecté et quelle est sa version ? Soyez prêt à fournir des informations de diagnostic pertinentes. Cela peut comprendre le résultat de **sosreport**, traité plus tard dans cette section. Pour les problèmes de noyau, cela peut comprendre un vidage de la mémoire en cas de panne **kdump** du système ou une photo numérique des messages de débogage du noyau affichés sur l'écran du système en panne.

Déterminez le niveau de gravité. Red Hat utilise quatre niveaux de gravité pour classer les problèmes. Les rapports de problèmes de gravité *Urgente* et *Élevée* doivent être suivis d'un appel téléphonique au centre d'assistance local compétent (voir <https://access.redhat.com/site/support/contact/technicalSupport>).

GRAVITÉ	DESCRIPTION
Urgente (Gravité de niveau 1)	Problème entravant sérieusement l'utilisation du logiciel dans un environnement de production. Il peut s'agir, par exemple, d'une perte des données de production ou d'un dysfonctionnement des systèmes de production. La situation empêche le déroulement normal des opérations de votre activité, et il n'existe aucune procédure alternative.

GRAVITÉ	DESCRIPTION
Élevée (Gravité de niveau 2)	Problème au cours duquel le logiciel fonctionne toujours, moyennant une réduction sévère de son utilisation dans un environnement de production. La situation a de graves conséquences sur le déroulement de vos activités, et il n'existe aucune procédure de contournement.
Moyenne (Gravité de niveau 3)	Problème impliquant une perte d'utilisation partielle, mais non critique, du logiciel dans un environnement de production ou de développement. Dans les environnements de production, l'incidence sur l'activité est de moyenne à faible. Les activités de l'entreprise peuvent se poursuivre via la mise en place d'une procédure de contournement. Dans les environnements de développement, le problème empêche le projet de passer en phase de production.
Faible (Gravité de niveau 4)	Question d'ordre général, signalement d'une erreur dans la documentation ou recommandation pour une amélioration ou modification future du produit. Dans les environnements de production, l'incidence sur l'activité, les performances ou le fonctionnement du système est faible à nulle. Dans les environnements de développement, l'incidence sur l'activité est moyenne ou faible, mais l'activité peut se poursuivre par la mise en place d'une procédure de contournement.

Gestion d'un rapport de bogues avec redhat-support-tool

Vous pouvez créer, afficher, modifier et fermer les dossiers d'assistance Red Hat à l'aide de la commande **redhat-support-tool**. Lorsqu'un dossier d'assistance se trouve dans l'état **opened** ou **maintained**, l'utilisateur peut y associer des fichiers ou de la documentation, comme des rapports de diagnostic (sosreport). L'outil télécharge et joint les fichiers aux dossiers.

Les détails d'un dossier comme le nom du produit, sa version, le récapitulatif, la description, la gravité et le groupe de dossiers peuvent être attribués à l'aide d'options de commande, ou en laissant l'outil demander les informations nécessaires. Dans l'exemple suivant, un nouveau dossier est ouvert. Les options **--product** et **--version** sont spécifiées.

```
[user@host ~]$ redhat-support-tool
Welcome to the Red Hat Support Tool.
Command (? for help): opencase --product="Red Hat Enterprise Linux" --
version="7.0"
Please enter a summary (or 'q' to exit): System fails to run without power
Please enter a description (Ctrl-D on an empty line when complete):
When the server is unplugged, the operating system fails to continue.
1 Urgent
2 High
3 Normal
4 Low
Please select a severity (or 'q' to exit): 4
Would you like to assign a case group to this case (y/N)? N
Would see if there is a solution to this problem before opening a support case?
(y/N) N
-----
Support case 01034421 has successfully been opened.
```

Si les options **--product** et **--version** ne sont pas spécifiées, la commande **redhat-support-tool** propose une liste de choix.

```
[user@host ~]$ redhat-support-tool
Welcome to the Red Hat Support Tool.
Command (? for help): opencase
Do you want to use the default product - "Red Hat Enterprise Linux" (y/N)?: y
...output omitted...
29 7.4
30 7.5
31 7.6
32 8.0 Beta
Please select a version (or 'q' to exit): 32
Please enter a summary (or 'q' to exit): yum fails to install apache
Please enter a description (Ctrl-D on an empty line when complete):
yum cannot find correct repo
1 Urgent
2 High
3 Normal
4 Low
Please select a severity (or 'q' to exit): 4
Would you like to use the default (Ungrouped Case) Case Group (y/N)? : y
Would you like to see if there's a solution to this problem before opening a
support case? (y/N) N
-----
Support case 010355678 has successfully been opened.
```

Association d'informations de diagnostic à un dossier d'assistance

Inclure des informations de diagnostic peut se traduire par une résolution plus rapide. Joignez le rapport **sosreport** lorsque le dossier est ouvert. La commande **sosreport** génère une archive tar compressée avec des informations de diagnostic collectées sur le système actif. La commande **redhat-support-tool** vous invite à l'inclure si cette archive a déjà été créée :

```
Please attach a SoS report to support case 01034421. Create a SoS report as
the root user and execute the following command to attach the SoS report
directly to the case:
redhat-support-tool addattachment -c 01034421 path to sosreport

Would you like to attach a file to 01034421 at this time? (y/N) N
Command (? for help):
```

S'il n'existe aucun rapport SoS actif, un administrateur peut en générer un et le joindre ultérieurement. Utilisez la commande **redhat-support-tool addattachment** pour joindre le rapport.

Les dossiers d'assistance peuvent également être affichés, modifiés et fermés par l'abonné :

```
Command (? for help): listcases

Type the number of the case to view or 'e' to return to the previous menu.
1 [Waiting on Red Hat] System fails to run without power
```

```
No more cases to display
Select a Case: 1

Type the number of the section to view or 'e' to return to the previous menu.
1 Case Details
2 Modify Case
3 Description
4 Recommendations
5 Get Attachment
6 Add Attachment
7 Add Comment
End of options.
Option: q

Select a Case: q

Command (? for help):q

[user@host ~]$ redhat-support-tool modifycase --status=Closed 01034421
Successfully updated case 01034421
[user@host ~]$
```

L'utilitaire Red Hat Support Tool dispose de capacités avancées de diagnostic et d'analyse d'applications. À l'aide de fichiers bruts de vidage de la mémoire suite à un problème de noyau, **redhat-support-tool** peut créer et extraire une *trace*. Ce type de fichier est créé à l'aide de la commande **kdump**. Une trace est un rapport des trames de pile actives au moment du vidage sur incident. Elle fournit des diagnostics sur site. L'une des options de la commande **redhat-support-tool** permet d'ouvrir un dossier d'assistance.

L'outil fournit également une analyse des fichiers journaux. À l'aide de la commande **analyze** de l'outil, il est possible d'analyser des fichiers journaux de nombreux types (système d'exploitation, JBoss, Python, Tomcat, oVirt, etc.) afin d'identifier les symptômes de problèmes. Les fichiers journaux peuvent ensuite être affichés et diagnostiqués un par un. En fournissant une analyse prétraitée, par opposition à des données brutes comme un vidage de la mémoire en cas de panne, on accélère l'ouverture des dossiers d'assistance et leur mise à la disposition des ingénieurs.

PARTICIPATION AU PROGRAMME RED HAT DEVELOPER

Red Hat Developer est une autre ressource bien utile mise à votre disposition par Red Hat. Ce programme est hébergé à l'adresse <https://developer.redhat.com>. Il fournit des abonnements aux logiciels Red Hat, de la documentation et des ouvrages de qualité rédigés par nos experts en microservices, en informatique sans serveur, Kubernetes et Linux. Vous y trouverez également un blog, des liens vers des informations sur les formations et les événements à venir, d'autres ressources d'aide, ainsi que des liens vers le portail client Red Hat.

L'inscription est gratuite et peut être réalisée à l'adresse suivante : <https://developer.redhat.com/register>.



RÉFÉRENCES

Page de manuel sosreport(1)

Red Hat Access: Red Hat Support Tool

<https://access.redhat.com/site/articles/445443>

Red Hat Support Tool First Use

<https://access.redhat.com/site/videos/534293>

Contacting Red Hat Technical Support

https://access.redhat.com/site/support/policy/support_process/

Help - Red Hat Customer Portal

<https://access.redhat.com/site/help/>

► EXERCICE GUIDÉ

OBTENIR DE L'AIDE AUPRÈS DU PORTAIL CLIENT RED HAT

Au cours de cet exercice, vous allez générer un rapport de diagnostic à l'aide de la console Web.

RÉSULTATS

Vous serez en mesure de générer un rapport de diagnostic à l'aide de la console Web, lequel pourra être envoyé au portail client Red Hat dans le cadre d'un dossier d'assistance.

AVANT DE COMMENCER

Connectez-vous en tant qu'utilisateur student à workstation avec le mot de passe student.

À partir de workstation, exécutez la commande **lab support-portal start**. La commande exécute un script de démarrage qui détermine si servera est accessible sur le réseau. Elle démarre et active également la console Web sur servera.

```
[student@workstation ~]$ lab support-portal start
```

- 1. À partir de workstation, utilisez la commande **ssh** pour vous connecter à servera en tant qu'utilisateur student.

```
[student@workstation ~]$ ssh student@servera
Web console: https://servera.lab.example.com:9090/ or https://172.25.250.10:9090/
[student@servera ~]$
```

- 2. Utilisez la commande **systemctl** pour vérifier que le service **cockpit** est en cours d'exécution. Saisissez student comme mot de passe lorsque vous y êtes invité.

```
[student@servera ~]$ sudo systemctl status cockpit.socket
[sudo] password for student: student
● cockpit.socket - Cockpit Web Service Socket
  Loaded: loaded (/usr/lib/systemd/system/cockpit.socket; enabled; vendor preset: disabled)
  Active: active (listening) since Thu 2019-05-16 10:32:33 IST; 4min 37s ago
    Docs: man:cockpit-ws(8)
   Listen: [::]:9090 (Stream)
  Process: 676 ExecStartPost=/bin/ln -snf active.motd /run/cockpit/motd (code=exited, status=0/SUCCESS)
  Process: 668 ExecStartPost=/usr/share/cockpit/motd/update-motd localhost (code=exited, status=0/SUCCESS)
  Tasks: 0 (limit: 11405)
```

```
Memory: 1.5M  
CGroup: /system.slice/cockpit.socket  
...output omitted...
```

► 3. Déconnectez-vous de servera.

```
[student@servera ~]$ exit  
[student@workstation ~]$
```

► 4. Sur workstation, ouvrez Firefox et connectez-vous à l'interface de la console Web exécutée sur servera.lab.example.com en tant qu'utilisateur root avec le mot de passe redhat.

- 4.1. Ouvrez Firefox et accédez à l'adresse <https://servera.lab.example.com:9090>.
- 4.2. Si vous y êtes invité, acceptez le certificat autosigné en l'ajoutant comme exception.
- 4.3. Connectez-vous en tant qu'utilisateur root avec le mot de passe redhat. Vous êtes maintenant connecté en tant qu'utilisateur privilégié, ce qui est nécessaire pour créer un rapport de diagnostic.
- 4.4. Cliquez sur Diagnostic Reports dans la barre de navigation de gauche. Cliquez sur Create Report. La création du rapport prend quelques minutes.

► 5. Lorsque le rapport est prêt, cliquez sur Download report. Enregistrez le fichier.

- 5.1. Cliquez sur le bouton Download report, puis sur Save File.
- 5.2. Cliquez sur le bouton Close.
- 5.3. Déconnectez-vous de l'interface de la console Web.

Fin

Sur workstation, exécutez le script **lab support-portal finish** pour mettre fin à l'exercice.

```
[student@workstation ~]$ lab support-portal finish
```

L'exercice guidé est maintenant terminé.

DÉTECTION ET RÉSOLUTION DES PROBLÈMES AVEC RED HAT INSIGHTS

OBJECTIFS

Après avoir terminé cette section, vous serez en mesure d'utiliser Red Hat Insights pour analyser les serveurs à la recherche de problèmes, résoudre ces derniers et vérifier que la solution fonctionne.

PRÉSENTATION DE RED HAT INSIGHTS

Red Hat Insights est un outil d'analyse prédictive qui vous aide à identifier et à corriger les menaces pesant sur la sécurité, les performances, la disponibilité et la stabilité des systèmes exécutant des produits Red Hat au sein de votre infrastructure. Red Hat Insights est fourni en tant que produit SaaS (Software-as-a-Service), de sorte que vous puissiez le déployer et le dimensionner rapidement, sans exigences supplémentaires au niveau de l'infrastructure. De plus, cela signifie que vous pouvez profiter immédiatement des dernières recommandations et mises à jour de Red Hat concernant vos systèmes déployés.

Red Hat met régulièrement à jour la base de connaissances utilisée par Red Hat Insights, en fonction des risques courants, des failles de sécurité, des configurations incorrectes connues et d'autres problèmes identifiés par Red Hat. Les actions visant à corriger ces problèmes sont validées et vérifiées par Red Hat. Cela vous permet d'identifier, de hiérarchiser et de résoudre les problèmes de manière proactive, avant qu'ils ne deviennent un problème majeur.

Pour chaque problème détecté, Red Hat Insights fournit des estimations du risque que cela représente, ainsi que des recommandations sur la manière d'atténuer ou de résoudre ledit problème. Ces recommandations peuvent se présenter sous la forme de playbooks Ansible ou d'instructions détaillées pour vous aider à résoudre le problème.

Les recommandations Red Hat Insights sont adaptées à chaque système enregistré auprès du service. Vous installez chaque système client avec un agent qui collecte des métadonnées sur la configuration d'exécution du système. Ces données constituent un sous-ensemble de ce que vous pourriez fournir au support Red Hat à l'aide de la commande **sosreport** afin de résoudre un ticket d'assistance. Vous pouvez limiter ou masquer les données envoyées par vos clients. Cela empêchera l'exécution de certaines règles analytiques, en fonction des limites que vous avez définies.

Presque immédiatement après l'enregistrement d'un serveur et la fin de la synchronisation initiale des métadonnées du système, vous devez être en mesure de voir votre serveur et toute recommandation applicable dans la console Red Hat Insights dans le portail cloud Red Hat.

Red Hat Insights fournit actuellement des analyses prédictives et des recommandations pour les produits Red Hat suivants :

- Red Hat Enterprise Linux 6.4 et versions ultérieures
- Red Hat Virtualization 4 et versions ultérieures
- Red Hat OpenShift Container Platform
- Red Hat OpenStack Platform 7 et versions ultérieures

Description de l'architecture de Red Hat Insights

Vous pouvez enregistrer un système sur Red Hat Insights via le portail cloud Red Hat. Lorsque vous enregistrez le système, il fournit à Red Hat Insights des métadonnées sur sa configuration actuelle. Ces données sont envoyées à Red Hat Insights à l'aide du chiffrement TLS pour les protéger en transit. Elles sont également anonymisées avant l'envoi.

Sur la base des recommandations fournies par le moteur de règles Red Hat Insights, les résultats de l'analyse sont affichés dans la console de Red Hat Insights dans le portail cloud Red Hat à l'adresse <https://cloud.redhat.com/insights>.

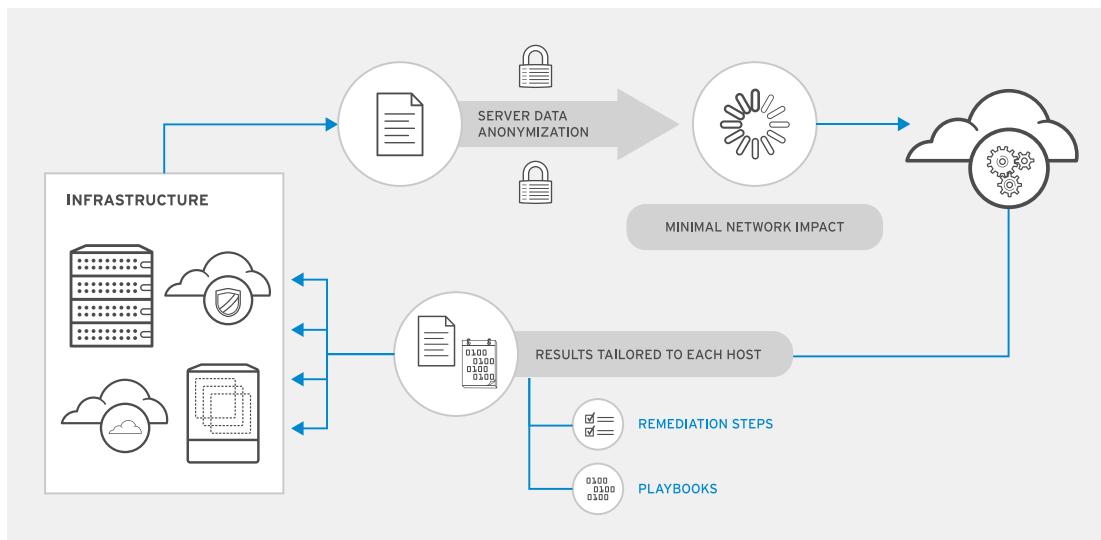


Figure 16.34: Architecture de haut niveau de Red Hat Insights

INSTALLATION DE CLIENTS RED HAT INSIGHTS

Red Hat Insights est inclus avec Red Hat Enterprise Linux 8 dans le cadre de l'abonnement. Les anciennes versions des serveurs Red Hat Enterprise Linux nécessitent l'installation du paquetage *insights-client* sur le système.



IMPORTANT

Le paquetage *insights-client* remplace l'ancien paquetage *redhat-access-insights* depuis la version Red Hat Enterprise Linux 7.5.

Si votre système est enregistré pour les droits d'accès logiciels via le service de gestion des abonnements du portail client, vous pouvez activer Red Hat Insights avec une seule commande. Utilisez la commande **`insights-client --register`** pour enregistrer le système.

```
[root@demo ~]# insights-client --register
```

Le client Insights met régulièrement à jour les métadonnées fournies à Red Hat Insights. Utilisez la commande **`insights-client`** pour actualiser les métadonnées du client à tout moment.

```
[root@demo ~]# insights-client
Starting to collect Insights data for demo.lab.example.com
Uploading Insights data.
Successfully uploaded report from 773b351b-dfb1-4393-afa8-915cc2875e06 to
account XXXXX.
```

Enregistrement d'un système RHEL sur Red Hat Insights

Pour enregistrer un serveur RHEL sur Red Hat Insights, le processus général est le suivant :

1. Enregistrez le système de manière interactive auprès du service de gestion des abonnements Red Hat.

```
[root@demo ~]# subscription-manager register --auto-attach
```

Un droit valide pour Red Hat Insights doit être associé au système. Vous pouvez le recevoir dans le cadre d'un abonnement Red Hat Enterprise Linux.

2. Assurez-vous que le paquetage *insights-client* est installé sur le système. Dans RHEL 7, ce paquetage se trouve dans le canal *rhel-7-server-rpms*.



NOTE

Cette étape n'est pas obligatoire sur les systèmes Red Hat Enterprise Linux 8.

```
[root@demo ~]# yum install insights-client
```

3. Utilisez la commande **insights-client --register** pour enregistrer le système auprès du service Red Hat Insights et télécharger les métadonnées initiales du système.

```
[root@demo ~]# insights-client --register
```

4. Vérifiez que le système est visible à l'adresse <https://cloud.redhat.com/insights>.

The screenshot shows the Red Hat Insights cloud portal interface. The top navigation bar includes the Red Hat logo, user information (Snehangshu Karmakar), and a search bar. The left sidebar has links for Overview, Rules, Inventory, Remediations, and Documentation. The main content area is titled 'Overview'. It displays 'Rule hits by severity' (1 Medium affecting 2 systems) and 'Rule hits by category' (1 Total hits, Security 1). Below this is a section titled 'Get started with Red Hat Insights' with three cards: 'Connect your first systems' (with a server icon), 'Remediate Insights findings with Ansible' (with an 'A' icon), and 'Deploy Insights at scale' (with a globe icon). A button at the bottom right says 'Download Ansible Playbook'.

Figure 16.35: Aperçu de Red Hat Insights sur le portail cloud

AFFICHAGE DES RAPPORTS FOURNIS PAR RED HAT INSIGHTS

Un rapport Red Hat Insights indique l'état d'un système au fil du temps. Avec ces rapports, vous pouvez facilement visualiser les évaluations actuelles des risques et identifier les tendances historiques afin d'améliorer votre prise de décisions.

L'interface de Red Hat Insights vous fournit les informations suivantes :

- Score de risque global actuel basé sur vos systèmes enregistrés.
- Mesures qu'il est recommandé de prendre sur vos systèmes, subdivisées en catégories et en degrés de gravité.
- Informations sur la dernière connexion des systèmes sur Red Hat Insights.
- Problèmes à classer en fonction de leur impact.

Navigation dans la console Red Hat Insights

La console de Red Hat Insights sur le portail cloud contient les pages suivantes :

Présentation

La page Overview fournit une vue des risques actuels de l'infrastructure enregistrée. Overview sert à examiner comment une règle spécifique affecte les systèmes enregistrés ou à voir toutes les règles qui présentent un risque pour un système sélectionné.

Cette page vous permet d'afficher les règles en fonction de la gravité et de classer le risque lié à l'infrastructure en fonction de la catégorie. Chaque règle est classée en fonction de l'impact potentiel sur l'un des domaines d'opérations suivants : **Availability, Stability, Performance et Security**

Rules

La page Rules fournit la liste des règles d'Insights et des hôtes affectés.

Dans la page Rules, vous remarquerez que certains de ces problèmes ont une coche sous la colonne de logo Ansible. Cela indique qu'un playbook de correction Ansible est disponible pour ce problème. Dans le cas des problèmes sans coche, aucun playbook de correction Ansible n'est disponible. Cependant, des instructions de correction ou de limitation manuelle des risques sont fournies dans les détails du problème.

Vous pouvez cliquer sur le nom de la règle pour afficher tous les systèmes affectés. Chaque problème contient une description de la manière dont le problème peut se manifester sur le système, et Remediate with Ansible pour créer un playbook de correction.

Rule	Last Updated	Total Risk	Systems	Ansible
Kernel vulnerable to side-channel attacks in modern microprocessors (CVE-2017-5715/Spectre)	a year ago	1	✓	
Kernel vulnerable to side-channel attacks in modern microprocessors (CVE-2017-5753/Spectre, CVE-2017-5715/Spectre, CVE-2017-5754/Meltdown)	a year ago	1	✓	
Kernel vulnerable to side-channel attacks in modern microprocessors using Speculative Store Bypass when CPU microcode is outdated (CVE-2018-3639)	a year ago	3		

Figure 16.36: Page Rules dans la console de Red Hat Insights

CHAPITRE 16 | Analyser les serveurs et obtenir une assistance**Inventory (Inventaire)**

La page Inventory fournit la liste des systèmes que vous avez enregistrés sur Red Hat Insights.

Vous pouvez facilement filtrer l'inventaire en fonction de systèmes spécifiques. La colonne Last Sync affiche l'heure à laquelle s'est produite la dernière mise à jour des métadonnées pour chaque système.

Name	Last Sync
rhel8.test.atl.redhat.com	2/28/2019, 6:10:37 AM
rhel8.test.atl.redhat.com	2/28/2019, 6:13:31 AM
rhel8.test.atl.redhat.com	3/8/2019, 11:47:36 AM
rhel8s6test.atl.redhat.com	4/2/2019, 12:02:08 PM

Figure 16.37: Page Inventory dans la console de Red Hat Insights

Remediations

La page Remediations fournit une liste des playbooks Ansible créés et permet de les télécharger.

Playbook	Systems	Actions	Last modified
testplay1	1	2	17 hours ago
testplay	1	1	a day ago

Figure 16.38: Page Remediations dans la console de Red Hat Insights

Affichage des problèmes signalés par Red Hat Insights

Pour afficher les problèmes signalés par Red Hat Insights, le processus général est le suivant :

1. Connectez-vous au portail cloud Red Hat et accédez à la page Red Hat Insights à l'adresse <https://cloud.redhat.com/insights>.
2. Dans le portail, accédez à la page Overview.
3. Sélectionnez Rule hits by severity pour afficher les règles en fonction du Total Risk qu'elles représentent pour l'infrastructure enregistrée. Vous pouvez également sélectionner Rule hits by category pour afficher le type de risque en fonction de la catégorie.
4. Faites défiler la liste des règles pour afficher les informations de haut niveau sur les risques, les systèmes exposés et la disponibilité du playbook Ansible pour automatiser la correction.
5. Cliquez sur une règle pour afficher une description plus détaillée de la règle, cliquez sur le lien pour lire les articles pertinents de la base de connaissances et affichez la liste de tous les hôtes affectés.

- Cliquez sur un hôte pour afficher des informations spécifiques sur les problèmes détectés et les étapes à suivre pour résoudre le problème.

INTERPRÉTATION DES RAPPORTS RED HAT INSIGHTS

Dans Red Hat Insights, les règles déterminent les problèmes recherchés sur vos systèmes. Red Hat ajoute fréquemment de nouvelles règles à Red Hat Insights afin de rechercher des problèmes récemment identifiés. Les règles peuvent rechercher les incidents qui se sont produits sur votre système et qui indiquent un problème ou anticiper les problèmes en fonction de la configuration actuelle de votre système.

Lorsqu'une règle correspond à votre système, indiquant l'existence d'un problème, des informations supplémentaires sont fournies avec la règle pour vous aider à comprendre le problème, à hiérarchiser les tâches à effectuer pour le résoudre, à déterminer les mesures d'atténuation ou de correction disponibles et à automatiser sa résolution.

Chaque règle est classée par type et comprend un nom récapitulatif et une description plus détaillée qui explique le problème. En règle générale, les règles sont liées aux articles de la base de connaissances sur le portail client, avec des informations supplémentaires. L'article de la base de connaissances peut fournir des informations sur les différentes méthodes à mettre en œuvre pour atténuer ou résoudre un problème. La règle peut fournir des playbooks Ansible ou d'autres documents permettant d'automatiser l'atténuation et la correction des problèmes.

Certains problèmes sont complexes à résoudre et un correctif complet peut nécessiter un redémarrage ou un temps d'arrêt. Dans ce cas, une mesure temporaire peut éventuellement être proposée afin d'atténuer les problèmes en limitant les risques. La règle fournit les scores du risque présenté par le problème, dans plusieurs catégories.

Supposons, par exemple, que la résolution d'un problème de sécurité nécessite une mise à jour des paquetages du noyau et un redémarrage. Cependant, quelques modifications de configuration temporaires rendraient l'exploitation de ce problème très difficile. Vous pouvez donc choisir d'appliquer les modifications temporaires immédiatement et de différer le redémarrage jusqu'à ce que vous puissiez planifier une fenêtre de maintenance d'urgence.

Red Hat Insights classe le risque qu'un problème présente pour votre système dans quatre catégories. Le niveau de risque est évalué à l'aide des niveaux suivants : **Low**, **Moderate**, **Important** et **Critical**.

Les catégories **Likelihood**, **Impact**, **Total Risk** et **Risk of Change** prévoient les facteurs de risque d'un problème détecté sur les systèmes abonnés.

Impact

Indique le niveau d'impact prévu de ce problème sur le système.

Likelihood

Indique la probabilité qu'un problème donné ait une incidence sur le système.

Total Risk

Indique l'impact des problèmes de sécurité rencontrés dans les produits Red Hat sur une échelle à quatre niveaux (Low, Moderate, Important et Critical), et elle exploite également les notes de base du système CVSS (Common Vulnerability Scoring System). Elles fournissent une évaluation des risques par ordre de priorité pour vous aider à prendre des décisions éclairées sur les risques que chaque problème fait courir à votre infrastructure.

Risk of Change

Indique le risque que la mesure corrective recommandée puisse perturber le système.

CHAPITRE 16 | Analyser les serveurs et obtenir une assistance

Pour afficher les facteurs de risque prévus selon diverses règles dans Red Hat Insights, accédez à la page Overview ou Rules. Chaque règle affiche l'icône relative à Total Risk et Risk of Change.

The screenshot shows the Red Hat Insights interface. At the top, it displays a rule titled "Kernel vulnerable to side-channel attacks in modern microprocessors (CVE-2017-5715/Spectre)". It includes details like "Publish Date: 1/17/2018" and a link to a "Knowledgebase Article". Below this, sections for "Total Risk" (Low) and "Risk of Change" (Moderate) are shown, along with a note about requiring a reboot. A large section titled "Affected systems" lists one system: "servera.lab.example.com", which was last sync'd on "5/14/2019, 12:01:41 PM".

Figure 16.39: Règles de Red Hat Insights qui s'appliquent à un hôte

Une fois que vous avez identifié les problèmes à résoudre, vous pouvez les traiter manuellement ou automatiquement. Une fois que le problème a été résolu, et que Red Hat Insights a téléchargé de nouvelles métadonnées, la règle ne devrait plus correspondre à ce système et le problème devrait disparaître de la liste des actions recommandées.

Correction manuelle d'un problème signalé par Red Hat Insights

1. Connectez-vous au portail cloud Red Hat et accédez à la page Red Hat Insights à l'adresse <https://cloud.redhat.com/insights>. Accédez à la page Rules. Cliquez sur le nom de la règle à résoudre.
2. Faites défiler jusqu'à Affected systems pour voir tous les systèmes affectés à cause de la règle.
3. Cliquez sur l'un des liens des systèmes affectés sous la colonne Name. La page décrit comment le problème peut affecter le système et les étapes à suivre pour résoudre le problème sur le système. Suivez les instructions dans Steps to resolve pour résoudre le problème sur le système.

CHAPITRE 16 | Analyser les serveurs et obtenir une assistance

The screenshot shows the Red Hat Insights interface. At the top, it displays the URL: Rules > Kernel Vulnerable To Side-Channel Attacks In Modern Microprocessors (CVE-2017-5715/Spectre) > Servera.lab.example.com. Below this, the host information is listed: Hostname: servera.lab.example.com, Ansible host: servera.lab.example.com, UUID: bb07c89-83f0-4f51-a458-8482a478cd55, Last seen: 5/14/2019, 12:01:41 PM. On the right, there is an 'Actions' dropdown menu. The main content area shows a rule titled 'Kernel vulnerable to side-channel attacks in modern microprocessors (CVE-2017-5715/Spectre)'. It includes tabs for Impact, Likelihood, Total Risk, and Risk Of Change. A section titled 'Detected issues' states: 'This machine is vulnerable, because it runs a vulnerable kernel. An unprivileged attacker could use the vulnerability to read privileged memory by conducting targeted cache side-channel attacks, including memory locations that cross the syscall boundary or the guest/host boundary, or potentially arbitrary host memory addresses.' Another section, 'Steps to resolve', recommends updating the kernel with the commands '# yum update kernel' and '# reboot'. It also notes that additional steps like Meltdown and CPU microcode/firmware updates are required. A note from Red Hat says: 'Subscribers are advised to contact their hardware OEM to receive the appropriate microcode/firmware for their processor. Red Hat may be providing microcode_ctl and linux_firmware packages that will cover the limited subset of chipsets we were able to test, but this will **not** address many CPUs that you may have in use in your server fleet. Again, contacting your hardware vendor will ensure you have the appropriate software to enable the protections for Variant 2 of this issue.'

Figure 16.40: Correction manuelle des règles sur les systèmes

4. Suivez la procédure sur le système concerné pour résoudre le problème.
5. Après avoir effectué les étapes de correction, exécutez la commande suivante en tant que root sur le système pour signaler les modifications à Red Hat Insights :

```
[root@demo ~]# insights-client
Starting to collect Insights data for demo.lab.example.com
Uploading Insights data.
Successfully uploaded report from 773b351b-dfb1-4393-afa8-915cc2875e06 to
account xxxxxx.
```

6. Dans la console Red Hat Insights, accédez à la page Rules. Cliquez sur la règle et faites défiler jusqu'à Affected systems. Vérifiez ensuite que le problème n'apparaît plus dans la liste des systèmes affectés.



RÉFÉRENCES

Pages de manuel **insights-client(8)** et **insights-client.conf(5)**

Pour plus d'informations, reportez-vous au chapitre **GET STARTED (PRISE EN MAIN)** du manuel *Red Hat Insights 1.0 Getting Started Guide* à l'adresse <https://access.redhat.com/products/red-hat-insights/#getstarted>

Pour plus d'informations sur les mises à jour de fonctionnalités pour Red Hat Insights, rendez-vous sur
https://access.redhat.com/documentation/en-us/red_hat_insights/1.0/html-single/release_notes/#release_information

Des informations sur les données collectées par Red Hat Insights sont disponibles sur la page

System Information Collected by Red Hat Insights
<https://access.redhat.com/articles/1598863>

Des informations sur l'exclusion des données collectées par Red Hat Insights sont disponibles sur la page

Opting Out of Sending Metadata from Red Hat Insights Client
<https://access.redhat.com/articles/2025273>

► QUIZ

DÉTECTION ET RÉSOLUTION DES PROBLÈMES AVEC RED HAT INSIGHTS

Répondez aux questions suivantes en sélectionnant un ou plusieurs éléments :

► 1. Dans quel ordre les événements suivants se produisent-ils lors de la gestion d'un système Red Hat Enterprise Linux qui utilise Red Hat Insights ?

1. Red Hat Insights analyse les métadonnées du système pour déterminer les problèmes et recommandations applicables.
 2. Le client Insights télécharge les métadonnées système vers le service Red Hat Insights.
 3. L'administrateur affiche les actions recommandées dans le portail client Red Hat.
 4. Red Hat Insights collecte des métadonnées système sur le système Red Hat Enterprise Linux.
- a. 1, 2, 3, 4
b. 4, 2, 1, 3
c. 4, 2, 3, 1
d. 4, 1, 2, 3

► 2. Quelle commande est utilisée pour enregistrer un client sur Red Hat Insights ?

- a. `insights-client --register`
- b. `insights-client --no-upload`
- c. `subscription-manager register`
- d. `insights-client --unregister`

► 3. Quelles sont les deux pages de la console de Red Hat Insights qui vous permettent d'afficher une liste de règles, en utilisant des filtres basés sur la catégorie de risque ? (Choisissez-en deux.)

- a. Présentation
- b. Inventory (Inventaire)
- c. Rules
- d. Remédiation

► SOLUTION

DÉTECTION ET RÉSOLUTION DES PROBLÈMES AVEC RED HAT INSIGHTS

Répondez aux questions suivantes en sélectionnant un ou plusieurs éléments :

- 1. Dans quel ordre les événements suivants se produisent-ils lors de la gestion d'un système Red Hat Enterprise Linux qui utilise Red Hat Insights ?
- Red Hat Insights analyse les métadonnées du système pour déterminer les problèmes et recommandations applicables.
 - Le client Insights télécharge les métadonnées système vers le service Red Hat Insights.
 - L'administrateur affiche les actions recommandées dans le portail client Red Hat.
 - Red Hat Insights collecte des métadonnées système sur le système Red Hat Enterprise Linux.
- 2. Quelle commande est utilisée pour enregistrer un client sur Red Hat Insights ?
- `insights-client --register`
 - `insights-client --no-upload`
 - `subscription-manager register`
 - `insights-client --unregister`
- 3. Quelles sont les deux pages de la console de Red Hat Insights qui vous permettent d'afficher une liste de règles, en utilisant des filtres basés sur la catégorie de risque ? (Choisissez-en deux.)
- Présentation
 - Inventory (Inventaire)
 - Rules
 - Remédiation

RÉSUMÉ

Dans ce chapitre, vous avez appris les principes suivants :

- La console Web est une interface de gestion Web vers votre serveur, basée sur le service Open Source Cockpit.
- La console Web fournit des graphiques de performances du système, des outils graphiques pour gérer la configuration du système et examiner les journaux, ainsi que des interfaces de terminal interactives.
- Le portail client Red Hat vous permet d'accéder à la documentation, à des téléchargements, à des outils d'optimisation, à la gestion des dossiers d'assistance, ainsi qu'à la gestion des abonnements et des droits pour vos produits Red Hat.
- **redhat-support-tool** est un outil de ligne de commande permettant d'interroger la base de connaissances et d'utiliser des dossiers d'assistance à partir de la ligne de commande du serveur.
- Red Hat Insights est un outil d'analyse prédictive SaaS qui vous aide à identifier et à corriger les menaces pesant sur la sécurité, les performances, la disponibilité et la stabilité de vos systèmes.

CHAPITRE 17

RÉVISION COMPLÈTE

PROJET

Tâches de révision depuis *Red Hat System Administration I*

OBJECTIFS

- Tâches de révision depuis *Red Hat System Administration I*

SECTIONS

- Révision complète

ATELIER

- Atelier : Gestion de fichiers à partir de la ligne de commande
- Atelier : Gestion des utilisateurs et des groupes, des autorisations et des processus
- Atelier : Configuration et gestion d'un serveur
- Atelier : Gestion de réseaux
- Atelier : Montage de systèmes de fichiers et recherche de fichiers

RÉVISION COMPLÈTE

OBJECTIFS

Après avoir terminé cette section, les stagiaires doivent avoir révisé et actualisé les connaissances et les compétences acquises dans *Red Hat System Administration I*.

RÉVISION DE RED HAT SYSTEM ADMINISTRATION I

Avant de commencer la révision exhaustive de ce cours, les stagiaires doivent être familiarisés avec les rubriques abordées dans chaque chapitre.

Les stagiaires peuvent se référer aux précédentes sections du manuel pour en savoir plus.

Chapitre 1, *Prise en main de Red Hat Enterprise Linux*

Décrire et définir Open Source, Linux, les distributions Linux et Red Hat Enterprise Linux.

- Décrire et expliquer l'objet de Linux, d'Open Source, des distributions Linux et de Red Hat Enterprise Linux.

Chapitre 2, *Accès à la ligne de commande*

Se connecter à un système Linux et exécuter des commandes simples à l'aide du shell.

- Se connecter à un système Linux sur une console locale et exécuter des commandes simples à l'aide du shell.
- Se connecter à un système Linux en utilisant l'environnement de bureau GNOME 3 et exécuter des commandes depuis l'invite du shell dans un programme de terminal.
- Gagner du temps en utilisant la saisie semi-automatique par tabulation, l'historique des commandes et les raccourcis de modification de commande pour exécuter des commandes dans le shell bash.

Chapitre 3, *Gestion de fichiers à partir de la ligne de commande*

Copier, déplacer, créer, supprimer et organiser les fichiers depuis le shell bash.

- Décrire comment Linux organise les fichiers, et l'objet des divers répertoires dans la hiérarchie du système de fichiers.
- Spécifier l'emplacement des fichiers par rapport au répertoire de travail actuel et par emplacement absolu, déterminer et modifier votre répertoire de travail et lister le contenu des répertoires.
- Créer, copier, déplacer et supprimer des fichiers et des répertoires.
- Faire en sorte que plusieurs noms de fichiers référencent le même fichier en utilisant des liens fixes et symboliques.
- Exécuter efficacement les commandes qui affectent de nombreux fichiers en utilisant les fonctionnalités de filtrage par motif du shell bash.

Chapitre 4, *Aide dans Red Hat Enterprise Linux*

Résoudre les problèmes en utilisant les systèmes d'aide en local.

- Rechercher des informations dans les pages de manuel du système Linux local.
- Rechercher des informations dans la documentation locale de GNU Info.

Chapitre 5, *Création, affichage et modification de fichiers texte*

Créer, afficher et modifier des fichiers texte à partir de la sortie d'une commande ou dans un éditeur de texte.

- Enregistrer la sortie de la commande ou les erreurs dans un fichier avec la redirection du shell et traiter la sortie de la commande via plusieurs programmes de ligne de commande avec des pipes.
- Créer et modifier des fichiers texte en utilisant l'éditeur **vim**.
- Utiliser des variables shell pour vous aider à exécuter des commandes et modifier les scripts de démarrage bash pour définir des variables shell et d'environnement afin de modifier le comportement du shell et des programmes exécutés à partir de celui-ci.

Chapitre 6, *Gestion des utilisateurs et des groupes locaux*

Créer, gérer et supprimer les utilisateurs et groupes locaux, et administrer les politiques locales relatives aux mots de passe.

- Décrire l'objet des utilisateurs et des groupes sur un système Linux.
- Se connecter en tant que super utilisateur pour gérer un système Linux et accorder à d'autres utilisateurs un accès super utilisateur à l'aide de la commande **sudo**.
- Créer, modifier et supprimer des comptes d'utilisateur définis localement.
- Créer, modifier et supprimer des comptes de groupes définis localement.
- Définir une politique de gestion des mots de passe pour les utilisateurs, ainsi que verrouiller et déverrouiller manuellement les comptes d'utilisateur.

Chapitre 7, *Contrôle de l'accès aux fichiers*

Définir les permissions de système de fichiers de Linux sur des fichiers et évaluez les effets sur la sécurité de différents paramètres de permissions.

- Lister les permissions du système de fichiers sur les fichiers et les répertoires, et évaluez l'effet de ces permissions sur l'accès des utilisateurs et des groupes.
- Changer les permissions et la propriété des fichiers en utilisant des outils de ligne de commande.
- Contrôler les permissions par défaut des fichiers créés par les utilisateurs, expliquer l'effet des permissions spéciales, et utiliser des permissions spéciales et par défaut pour définir le groupe propriétaire des fichiers créés dans un répertoire particulier.

Chapitre 8, *Contrôle et gestion des processus Linux*

Évaluer et contrôler les processus exécutés sur un système Red Hat Enterprise Linux.

CHAPITRE 17 | Révision complète

- Obtenir des informations sur les programmes en cours d'exécution sur le système afin de pouvoir déterminer leur statut, l'utilisation des ressources et leur propriété pour les contrôler.
- Utiliser le contrôle de tâche Bash pour gérer plusieurs processus démarrés à partir de la même session de terminal.
- Contrôler et terminer les processus qui ne sont pas associés à votre shell et forcer la fin des processus et sessions utilisateur.
- Décrire la charge moyenne et déterminer les processus responsables d'une utilisation intensive des ressources sur un serveur.

Chapitre 9, Contrôle des services et des démons

Contrôler et surveiller les services réseau et les démons système à l'aide de Systemd.

- Répertorier les démons système et les services réseau démarrés par le service systemd et les unités de socket.
- Contrôler les démons système et les services réseau, en utilisant **systemctl**.

Chapitre 10, Configuration et sécurisation de SSH

Configurer un service de ligne de commande sécurisé sur les systèmes distants à l'aide d'OpenSSH.

- Se connecter à un système distant et exécuter des commandes à l'aide de **ssh**.
- Configurer une authentification par clé permettant à un compte d'utilisateur de se connecter à des systèmes distants de manière sécurisée, sans mot de passe.
- Restreindre les connexions directes en tant que root et désactiver l'authentification par mot de passe pour le service OpenSSH.

Chapitre 11, Analyse et stockage des journaux

Localiser et analyser avec précision les journaux d'événements système à des fins de diagnostic.

- Décrire l'architecture de journalisation de base utilisée par Red Hat Enterprise Linux pour enregistrer des événements.
- Interpréter les événements dans les fichiers syslog pertinents pour résoudre des problèmes ou vérifier l'état du système.
- Trouver et interpréter des entrées dans le journal système pour résoudre des problèmes ou vérifier l'état du système.
- Configurer le journal système pour conserver l'enregistrement des événements lorsqu'un serveur est redémarré.
- Maintenir une synchronisation précise de l'horloge à l'aide de NTP et configurer le fuseau horaire pour garantir des horodatages corrects pour les événements enregistrés par le journal système et les journaux.

Chapitre 12, Gestion de réseaux

Configurer les interfaces réseau et les paramètres sur des serveurs Red Hat Enterprise Linux.

- Décrire les concepts fondamentaux de l'adressage réseau et du routage pour un serveur.

CHAPITRE 17 | Révision complète

- Tester et inspecter la configuration réseau actuelle avec les utilitaires de ligne de commande.
- Gérer les paramètres réseau et les périphériques à l'aide de **nmcli**.
- Modifier les paramètres réseau en modifiant les fichiers de configuration.
- Configurer le nom d'hôte statique d'un serveur et sa résolution, puis tester les résultats.

Chapitre 13, Archivage et transfert de fichiers

Archiver et copier des fichiers d'un système à l'autre.

- Archiver des fichiers et des répertoires dans un fichier compressé en utilisant tar, et extraire le contenu d'une archive tar existante.
- Transférer des fichiers depuis ou vers un système distant en toute sécurité à l'aide de SSH.
- Synchronisez le contenu d'un fichier ou d'un répertoire local avec une copie sur un serveur distant.

Chapitre 14, Installation et mise à jour de paquetages logiciels

Télécharger, installer, mettre à jour et gérer les paquetages logiciels depuis les dépôts de paquetages Red Hat et Yum.

- Enregistrer un système sur votre compte Red Hat et lui attribuer les droits pour les mises à jour logicielles et les services de support en utilisant Red Hat Subscription Management.
- Expliquer comment les logiciels sont fournis sous forme de paquetages RPM et inspecter les paquetages installés sur le système avec Yum et RPM.
- Trouver, installer et mettre à jour des paquetages logiciels à l'aide de la commande **yum**.
- Activer et désactiver l'utilisation de référentiels Yum tiers ou Red Hat par un serveur.
- Expliquer comment les modules permettent l'installation de versions spécifiques de logiciels, lister, activer et permuter les flux de modules, et installer et mettre à jour les paquetages à partir d'un module.

Chapitre 15, Accès aux systèmes de fichiers Linux

Accéder aux systèmes de fichiers existants, les inspecter et les utiliser sur un stockage connecté à un serveur Linux.

- Expliquer ce qu'est un périphérique de traitement par blocs, interpréter les noms de fichiers des périphériques de stockage et identifier le périphérique de stockage utilisé par le système de fichiers pour un répertoire ou un fichier particulier.
- Accéder aux systèmes de fichiers en les attachant à un répertoire dans la hiérarchie du système de fichiers.
- Rechercher des fichiers sur les systèmes de fichiers montés au moyen des commandes **find** et **locate**.

Chapitre 16, Analyser les serveurs et obtenir une assistance

Examiner les problèmes et les résoudre dans l'interface de gestion Web, et obtenir une assistance auprès de Red Hat dans le cadre de leur résolution.

- Activer l'interface de gestion de la console Web pour gérer et surveiller à distance les performances d'un serveur Red Hat Enterprise Linux.
- Décrire les principales ressources disponibles via le portail client Red Hat, et rechercher des informations dans la documentation Red Hat et dans la base de connaissances.
- Analyser les serveurs à la recherche de problèmes, résoudre ces derniers et vérifier que la solution fonctionne avec Red Hat Insights.

► OPEN LAB

GESTION DE FICHIERS À PARTIR DE LA LIGNE DE COMMANDE

Au cours de cette révision, vous allez gérer des fichiers, rediriger un ensemble spécifique de lignes d'un fichier texte vers un autre fichier et modifier les fichiers texte.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Gérer des fichiers à partir de la ligne de commande.
- Afficher un certain nombre de lignes à partir de fichiers texte et rediriger la sortie vers un autre fichier.
- Modifier des fichiers texte.

AVANT DE COMMENCER

Copiez tous les fichiers ou travaux que vous souhaitez conserver sur d'autres systèmes avant de procéder à la réinitialisation. Réinitialisez les systèmes `workstation`, `servera` et `serverb` maintenant. Patientez jusqu'au démarrage des systèmes `workstation`, `servera` et `serverb`.

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review1 start` pour commencer la révision complète. Ce script crée les fichiers nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.

- Créez un répertoire nommé `/home/student/grading`.
- Créez trois fichiers vides dans le répertoire `/home/student/grading` : `grade1`, `grade2` et `grade3`.
- Capturez les cinq premières lignes du fichier `/home/student/bin/manage-files` dans `/home/student/grading/manage-files.txt`.
- Ajoutez les trois dernières lignes de `/home/student/bin/manage-files` au fichier `/home/student/grading/manage-files.txt`. Vous ne devez pas écraser le texte qui figure déjà dans le fichier `/home/student/grading/manage-files.txt`.
- Copiez `/home/student/grading/manage-files.txt` dans `/home/student/grading/manage-files-copy.txt`.
- Modifiez le fichier `/home/student/grading/manage-files-copy.txt` afin qu'il y ait deux lignes de texte successives indiquant `Test JJ`.

- Modifiez le fichier **/home/student/grading/manage-files-copy.txt** de sorte que la ligne de texte **Test HH** n'existe pas dans le fichier.
- Modifiez le fichier **/home/student/grading/manage-files-copy.txt** de sorte que la ligne **A new line** figure entre les lignes **Test BB** et **Test CC**.
- Créez un lien fixe nommé **/home/student/hardlink** vers le fichier **/home/student/grading/grade1**. Vous devrez procéder de la sorte après avoir créé le fichier vide **/home/student/grading/grade1**, comme indiqué ci-dessus.
- Créez un lien symbolique nommé **/home/student/softlink** vers le fichier **/home/student/grading/grade2**.
- Enregistrez le résultat d'une commande qui répertorie le contenu du répertoire **/boot** dans le fichier **/home/student/grading/longlisting.txt**. La sortie doit être une « longue liste » qui inclut les autorisations de fichier, le propriétaire et le propriétaire du groupe, la taille et la date de modification de chaque fichier.

Évaluation

Sur workstation, exécutez la commande **lab rhcsa-rh124-review1 grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 grade
```

Fin

Sur workstation, exécutez **lab rhcsa-rh124-review1 finish** pour terminer la révision complète. Ce script supprime les fichiers et les répertoires créés au début de la révision complète et garantit que l'environnement sur **serverb** est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 finish
```

Vous avez maintenant terminé la révision complète.

► SOLUTION

GESTION DE FICHIERS À PARTIR DE LA LIGNE DE COMMANDE

Au cours de cette révision, vous allez gérer des fichiers, rediriger un ensemble spécifique de lignes d'un fichier texte vers un autre fichier et modifier les fichiers texte.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Gérer des fichiers à partir de la ligne de commande.
- Afficher un certain nombre de lignes à partir de fichiers texte et rediriger la sortie vers un autre fichier.
- Modifier des fichiers texte.

AVANT DE COMMENCER

Copiez tous les fichiers ou travaux que vous souhaitez conserver sur d'autres systèmes avant de procéder à la réinitialisation. Réinitialisez les systèmes `workstation`, `servera` et `serverb` maintenant. Patientez jusqu'au démarrage des systèmes `workstation`, `servera` et `serverb`.

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review1 start` pour commencer la révision complète. Ce script crée les fichiers nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.

- Créez un répertoire nommé `/home/student/grading`.
- Créez trois fichiers vides dans le répertoire `/home/student/grading` : `grade1`, `grade2` et `grade3`.
- Capturez les cinq premières lignes du fichier `/home/student/bin/manage-files` dans `/home/student/grading/manage-files.txt`.
- Ajoutez les trois dernières lignes de `/home/student/bin/manage-files` au fichier `/home/student/grading/manage-files.txt`. Vous ne devez pas écraser le texte qui figure déjà dans le fichier `/home/student/grading/manage-files.txt`.
- Copiez `/home/student/grading/manage-files.txt` dans `/home/student/grading/manage-files-copy.txt`.
- Modifiez le fichier `/home/student/grading/manage-files-copy.txt` afin qu'il y ait deux lignes de texte successives indiquant `Test JJ`.

CHAPITRE 17 | Révision complète

- Modifiez le fichier **/home/student/grading/manage-files-copy.txt** de sorte que la ligne de texte **Test HH** n'existe pas dans le fichier.
- Modifiez le fichier **/home/student/grading/manage-files-copy.txt** de sorte que la ligne **A new line** figure entre les lignes **Test BB** et **Test CC**.
- Créez un lien fixe nommé **/home/student/hardlink** vers le fichier **/home/student/grading/grade1**. Vous devrez procéder de la sorte après avoir créé le fichier vide **/home/student/grading/grade1**, comme indiqué ci-dessus.
- Créez un lien symbolique nommé **/home/student/softlink** vers le fichier **/home/student/grading/grade2**.
- Enregistrez le résultat d'une commande qui répertorie le contenu du répertoire **/boot** dans le fichier **/home/student/grading/longlisting.txt**. La sortie doit être une « longue liste » qui inclut les autorisations de fichier, le propriétaire et le propriétaire du groupe, la taille et la date de modification de chaque fichier.

1. Créez un répertoire nommé /home/student/grading.

1.1. À partir de **workstation**, ouvrez une session SSH sur **serverb** en tant que **student**.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

1.2. Utilisez la commande **mkdir** pour créer le répertoire **/home/student/grading**.

```
[student@serverb ~]$ mkdir grading
```

Comme vous avez exécuté la commande précédente dans le répertoire personnel de l'utilisateur **student**, vous n'avez pas spécifié le chemin absolu vers le répertoire **grading** lors de sa création.

2. Créez trois fichiers vides dans le répertoire /home/student/grading : grade1, grade2 et grade3.

2.1. Utilisez la commande **touch** pour créer les fichiers vides nommés **grade1**, **grade2**, et **grade3** dans le répertoire **/home/student/grading**. Appliquez la fonctionnalité de shell d'extension d'accolade pour créer les trois fichiers avec une seule commande **touch**.

```
[student@serverb ~]$ touch grading/grade{1,2,3}
```

2.2. Utilisez la commande **ls** pour vérifier que les fichiers **grade1**, **grade2** et **grade3** existent sous le répertoire **/home/student/grading**.

```
[student@serverb ~]$ ls grading/  
grade1 grade2 grade3
```

3. Capturez les cinq premières lignes du fichier /home/student/bin/manage-files dans /home/student/grading/manage-files.txt.

CHAPITRE 17 | Révision complète

- 3.1. Utilisez la commande **head** pour afficher les cinq premières lignes du fichier **/home/student/bin/manage-files** et rediriger la sortie vers le fichier **/home/student/grading/manage-files.txt**.

```
[student@serverb ~]$ head -5 bin/manage-files > grading/manage-files.txt
```

La commande précédente utilise le symbole de redirection unique (**>**) pour enregistrer la sortie de la commande dans **/home/student/grading/manage-files.txt**, de sorte que tout contenu existant dans le fichier soit écrasé.

- 3.2. Vérifiez que le fichier **/home/student/grading/manage-files.txt** contient le texte suivant.

```
Test AA  
Test BB  
Test CC  
Test DD  
Test EE
```

4. Ajoutez les trois dernières lignes de **/home/student/bin/manage-files** au fichier **/home/student/grading/manage-files.txt**. Vous ne devez pas écraser le texte qui figure déjà dans le fichier **/home/student/grading/manage-files.txt**.

- 4.1. Utilisez la commande **tail** pour afficher les trois dernières lignes du fichier **/home/student/bin/manage-files** et ajouter la sortie à **/home/student/grading/manage-files.txt**.

```
[student@serverb ~]$ tail -3 bin/manage-files >> grading/manage-files.txt
```

La commande précédente utilise le symbole de redirection double (**>>**) pour ajouter la sortie à **/home/student/grading/manage-files.txt**, de sorte que le contenu existant dans le fichier soit conservé.

- 4.2. Vérifiez que le fichier **/home/student/grading/manage-files.txt** contient le texte suivant.

```
Test AA  
Test BB  
Test CC  
Test DD  
Test EE  
Test HH  
Test II  
Test JJ
```

5. Copiez le fichier **/home/student/grading/manage-files.txt** dans **/home/student/grading/manage-files-copy.txt**.

- 5.1. Utilisez la commande **cd** pour accéder au répertoire **/home/student/grading**.

```
[student@serverb ~]$ cd grading/  
[student@serverb grading]$
```

CHAPITRE 17 | Révision complète

- 5.2. Utilisez la commande **cp** pour copier le fichier **/home/student/grading/manage-files.txt** dans **/home/student/grading/manage-files-copy.txt**.

```
[student@serverb grading]$ cp manage-files.txt manage-files-copy.txt
```

- 5.3. Revenez au répertoire personnel de l'utilisateur student.

```
[student@serverb grading]$ cd  
[student@serverb ~]$
```

6. Modifiez le fichier **/home/student/grading/manage-files-copy.txt** afin qu'il y ait deux lignes de texte successives indiquant **Test JJ**.

- 6.1. Utilisez l'éditeur de texte **vim** pour ouvrir le fichier **/home/student/grading/manage-files-copy.txt**.

```
[student@serverb ~]$ vim grading/manage-files-copy.txt
```

- 6.2. À partir du mode de commande dans **vim**, faites défiler la page jusqu'à la ligne qui contient la ligne de texte **Test JJ**. Appuyez deux fois sur la touche **y** de votre clavier pour copier la ligne de texte et appuyez ensuite sur la touche **p** pour la coller sous le curseur. Entrez **:wq** pour enregistrer les modifications et quitter **vim**. Vérifiez que le fichier **/home/student/grading/manage-files-copy.txt** contient le texte suivant.

```
Test AA  
Test BB  
Test CC  
Test DD  
Test EE  
Test HH  
Test II  
Test JJ  
Test JJ
```

Notez que le contenu précédent comprend deux copies de la ligne de texte **Test JJ**.

7. Modifiez le fichier **/home/student/grading/manage-files-copy.txt** de sorte que la ligne de texte **Test HH** n'existe pas dans le fichier.

- 7.1. Utilisez l'éditeur de texte **vim** pour ouvrir le fichier **/home/student/grading/manage-files-copy.txt**.

```
[student@serverb ~]$ vim grading/manage-files-copy.txt
```

- 7.2. À partir du mode de commande dans **vim**, faites défiler la page jusqu'à la ligne qui contient la ligne de texte **Test HH**. Appuyez deux fois sur la touche **d** de votre clavier pour supprimer la ligne de texte. Entrez **:wq** pour enregistrer les modifications et quitter **vim**. Vérifiez que le fichier **/home/student/grading/manage-files-copy.txt** contient le texte suivant.

```
Test AA  
Test BB  
Test CC  
Test DD  
Test EE  
Test II  
Test JJ  
Test JJ
```

Notez que la ligne de texte **Test JJ** ne figure pas dans le contenu précédent.

8. Modifiez le fichier **/home/student/grading/manage-files-copy.txt** de sorte que la ligne **A new line** figure entre les lignes **Test BB** et **Test CC**.

- 8.1. Utilisez l'éditeur de texte **vim** pour ouvrir le fichier **/home/student/grading/manage-files-copy.txt**.

```
[student@serverb ~]$ vim grading/manage-files-copy.txt
```

- 8.2. À partir du mode de commande dans **vim**, faites défiler la page jusqu'à la ligne qui contient la ligne de texte **Test CC**. Appuyez sur la touche **i** du clavier pour passer en mode insertion, tout en maintenant le curseur au début de la ligne de texte **Test CC**. En mode insertion, appuyez sur la touche **Entrée** du clavier pour créer une ligne vide au-dessus du curseur. Utilisez la flèche vers le haut pour accéder à la ligne vide et créer la ligne de texte **A new line**. Appuyez sur la touche **Échap** du clavier pour repasser en mode de commande. Entrez **:wq** pour enregistrer les modifications et quitter **vim**. Vérifiez que le fichier **/home/student/grading/manage-files-copy.txt** contient le texte suivant.

```
Test AA  
Test BB  
A new line  
Test CC  
Test DD  
Test EE  
Test II  
Test JJ  
Test JJ
```

Notez que la ligne de texte **A new line** figure dans le contenu précédent.

9. Créez un lien fixe nommé **/home/student/hardlink** vers le fichier **/home/student/grading/grade1**.

- 9.1. Utilisez la commande **ln** pour créer le lien fixe nommé **/home/student/hardlink** vers le fichier **/home/student/grading/grade1**. Vous devrez procéder de la sorte après avoir créé le fichier vide **/home/student/grading/grade1**, comme indiqué ci-dessus.

```
[student@serverb ~]$ ln grading/grade1 hardlink
```

- 9.2. Utilisez la commande **ls -l** pour afficher le nombre de liens du fichier **/home/student/grading/grade1**.

CHAPITRE 17 | Révision complète

```
[student@serverb ~]$ ls -l grading/grade1  
-rw-rw-r--. 2 student student 0 Mar 6 16:45 grading/grade1
```

10. Créez un lien symbolique nommé **/home/student/softlink** vers le fichier **/home/student/grading/grade2**.

10.1. Utilisez la commande **ln -s** pour créer le lien symbolique nommé **/home/student/softlink** vers le fichier **/home/student/grading/grade2**.

```
[student@serverb ~]$ ln -s grading/grade2 softlink
```

10.2. Utilisez la commande **ls -l** pour afficher les propriétés du lien symbolique **/home/student/softlink**.

```
[student@serverb ~]$ ls -l softlink  
lrwxrwxrwx. 1 student student 14 Mar 6 17:58 softlink -> grading/grade2
```

11. Enregistrez le résultat d'une commande qui répertorie le contenu du répertoire **/boot** dans le fichier **/home/student/grading/longlisting.txt**. La sortie doit être une « longue liste » qui inclut les autorisations de fichier, le propriétaire et le propriétaire du groupe, la taille et la date de modification de chaque fichier.

11.1. Utilisez la commande **ls -l** pour afficher le contenu du répertoire **/boot** au format « longue liste » et rediriger la sortie vers le fichier **/home/student/grading/longlisting.txt**.

```
[student@serverb ~]$ ls -l /boot > grading/longlisting.txt
```

11.2. Déconnectez-vous de serverb.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.
```

Évaluation

Sur workstation, exécutez la commande **lab rhcsa-rh124-review1 grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 grade
```

Fin

Sur workstation, exécutez **lab rhcsa-rh124-review1 finish** pour terminer la révision complète. Ce script supprime les fichiers et les répertoires créés au début de la révision complète et garantit que l'environnement sur serverb est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 finish
```

Vous avez maintenant terminé la révision complète.

► OPEN LAB

GESTION DES UTILISATEURS ET DES GROUPES, DES AUTORISATIONS ET DES PROCESSUS

Au cours de cette révision, vous allez gérer des comptes d'utilisateurs et de groupes, définir des autorisations sur des fichiers et des répertoires, et gérer des processus.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Gérer des groupes et des utilisateurs.
- Définir des autorisations sur des fichiers et des répertoires.
- Supprimer les processus qui consomment trop de ressources processeur.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review2 start` pour lancer la révision complète. Ce script exécute un processus qui utilise le maximum de ressources processeur et crée les fichiers nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.

- Terminez le processus qui utilise actuellement le plus de temps processeur.
- Créez un groupe appelé `database` qui porte le GID **50000**.
- Créez un utilisateur nommé `dbuser1` qui utilise le groupe `database` comme l'un de ses groupes secondaires. Le mot de passe initial de `dbuser1` doit être défini sur `redhat`. Configurez l'utilisateur `dbuser1` pour forcer un changement de mot de passe lors de sa première connexion. L'utilisateur `dbuser1` doit être en mesure de changer son mot de passe après **10** jours à compter de la date de modification. Le mot de passe de `dbuser1` doit expirer **30** jours après la date de la dernière modification.
- Configurez l'utilisateur `dbuser1` pour qu'il utilise `sudo` pour exécuter toute commande en tant que superutilisateur.
- Configurez l'utilisateur `dbuser1` pour que son umask par défaut soit **007**.
- Les autorisations sur `/home/student/grading/review2` doivent permettre aux membres du groupe `database` et à l'utilisateur `student` d'accéder au répertoire et d'y créer du contenu. Tous les autres utilisateurs doivent posséder des autorisations de lecture et d'exécution sur le répertoire. Assurez-vous également que les utilisateurs ne

sont autorisés à supprimer de **/home/student/grading/review2** que les fichiers dont ils sont propriétaires, et non les fichiers qui ne leur appartiennent pas.

Évaluation

Sur workstation, exécutez la commande **lab rhcsa-rh124-review2 grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 grade
```

Fin

Sur workstation, exécutez **lab rhcsa-rh124-review2 finish** pour terminer la révision complète. Ce script met fin au processus, et supprime les fichiers et les répertoires créés au début de la révision complète et garantit que l'environnement sur serverb est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 finish
```

Vous avez maintenant terminé la révision complète.

► SOLUTION

GESTION DES UTILISATEURS ET DES GROUPES, DES AUTORISATIONS ET DES PROCESSUS

Au cours de cette révision, vous allez gérer des comptes d'utilisateurs et de groupes, définir des autorisations sur des fichiers et des répertoires, et gérer des processus.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Gérer des groupes et des utilisateurs.
- Définir des autorisations sur des fichiers et des répertoires.
- Supprimer les processus qui consomment trop de ressources processeur.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review2 start` pour lancer la révision complète. Ce script exécute un processus qui utilise le maximum de ressources processeur et crée les fichiers nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.

- Terminez le processus qui utilise actuellement le plus de temps processeur.
- Créez un groupe appelé `database` qui porte le GID **50000**.
- Créez un utilisateur nommé `dbuser1` qui utilise le groupe `database` comme l'un de ses groupes secondaires. Le mot de passe initial de `dbuser1` doit être défini sur `redhat`. Configurez l'utilisateur `dbuser1` pour forcer un changement de mot de passe lors de sa première connexion. L'utilisateur `dbuser1` doit être en mesure de changer son mot de passe après **10** jours à compter de la date de modification. Le mot de passe de `dbuser1` doit expirer **30** jours après la date de la dernière modification.
- Configurez l'utilisateur `dbuser1` pour qu'il utilise `sudo` pour exécuter toute commande en tant que superutilisateur.
- Configurez l'utilisateur `dbuser1` pour que son umask par défaut soit **007**.
- Les autorisations sur `/home/student/grading/review2` doivent permettre aux membres du groupe `database` et à l'utilisateur `student` d'accéder au répertoire et d'y créer du contenu. Tous les autres utilisateurs doivent posséder des autorisations de

CHAPITRE 17 | Révision complète

lecture et d'exécution sur le répertoire. Assurez-vous également que les utilisateurs ne sont autorisés à supprimer de **/home/student/grading/review2** que les fichiers dont ils sont propriétaires, et non les fichiers qui ne leur appartiennent pas.

1. Terminez le processus qui utilise actuellement le plus de temps processeur.
 - 1.1. À partir de **workstation**, ouvrez une session SSH sur **serverb** en tant que **student**.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

- 1.2. Utilisez la commande **top** pour afficher l'état du système en temps réel.

```
[student@serverb ~]$ top
```

- 1.3. À partir de l'interface interactive de **top**, observez la colonne **%CPU** et vérifiez qu'il existe bien un processus appelé **dd** qui consomme le plus de ressources processeur.

```
...output omitted...  
 PID USER      PR  NI    VIRT      RES      SHR S %CPU %MEM     TIME+ COMMAND  
2303 student    20   0 217048    944    876 R 99.7    0.1 100:11.64 dd  
...output omitted...
```

Notez le processus **dd** avec le PID **2303** dans la sortie précédente, qui consomme la majorité des ressources processeur (**99,7 %**). Le PID et le pourcentage d'utilisation des ressources processeur peuvent varier sur votre système.

- 1.4. À partir de l'interface interactive de **top**, saisissez **k** pour arrêter le processus **dd** avec le PID **2303**, comme vous l'avez déterminé à l'étape précédente. Si le PID par défaut indiqué dans l'invite correspond à celui du processus qui utilise la majorité des ressources processeur, appuyez sur la touche **Entrée** du clavier. S'il ne correspond pas, indiquez le PID de manière interactive.

```
...output omitted...  
PID to signal/kill [default pid = 2303] Enter  
...output omitted...
```

- 1.5. Utilisez le signal par défaut **SIGTERM** pour mettre fin au processus.

```
...output omitted...  
Send pid 2833 signal [15/sigterm] Enter  
...output omitted...
```

- 1.6. À partir de l'interface interactive, appuyez sur la touche **q** du clavier pour quitter **top**.

2. Créez un groupe appelé **database** avec le GID **50000**.

- 2.1. Basculez vers l'utilisateur **root**.

```
[student@serverb ~]$ sudo su -  
[sudo] password for student: student  
[root@serverb ~]#
```

- 2.2. Utilisez la commande **groupadd** pour créer un groupe appelé **database** avec le GID **50000**.

```
[root@serverb ~]# groupadd -g 50000 database
```

3. Créez un utilisateur nommé **dbuser1** avec le groupe **database** comme l'un de ses groupes secondaires. Définissez le mot de passe initial de **dbuser1** sur **redhat**. Configurez l'utilisateur **dbuser1** pour forcer un changement de mot de passe lors de la première connexion. L'utilisateur **dbuser1** doit être en mesure de changer son mot de passe après **10** jours à compter de la date de la dernière modification. Le mot de passe de **dbuser1** doit expirer **30** jours après la date de la dernière modification.
- 3.1. Utilisez la commande **useradd** pour créer un utilisateur nommé **dbuser1** qui utilise le groupe **database** comme l'un de ses groupes secondaires.

```
[root@serverb ~]# useradd -G database dbuser1
```

- 3.2. Utilisez la commande **passwd** pour définir le mot de passe de **dbuser1** sur **redhat**.

```
[root@serverb ~]# passwd dbuser1  
Changing password for user dbuser1.  
New password: redhat  
BAD PASSWORD: The password is shorter than 8 characters  
Retype new password: redhat  
passwd: all authentication tokens updated successfully.
```

- 3.3. Utilisez la commande **chage** pour forcer **dbuser1** à changer son mot de passe lors de la première connexion.

```
[root@serverb ~]# chage -d 0 dbuser1
```

- 3.4. Utilisez la commande **chage** pour définir l'âge minimum du mot de passe de **dbuser1** sur **10** jours.

```
[root@serverb ~]# chage -m 10 dbuser1
```

- 3.5. Utilisez la commande **chage** pour définir l'âge maximum du mot de passe de **dbuser1** sur **30** jours.

```
[root@serverb ~]# chage -M 30 dbuser1
```

4. Créez le fichier **/etc/sudoers.d/dbuser1** pour configurer l'utilisateur **dbuser1** de sorte qu'il puisse utiliser **sudo** pour exécuter toute commande en tant que superutilisateur. Vous pouvez utiliser la commande **vim /etc/sudoers.d/dbuser1** pour créer le fichier. **/etc/sudoers.d/dbuser1** doit comprendre le contenu suivant.

```
dbuser1 ALL=(ALL) ALL
```

5. Configurez l'utilisateur dbuser1 pour que son umask par défaut soit **007**.

- 5.1. Basculez vers l'utilisateur dbuser1.

```
[root@serverb ~]# su - dbuser1  
[dbuser1@serverb ~]$
```

- 5.2. Ajoutez la ligne **umask 007** aux fichiers **/home/dbuser1/.bash_profile** et **/home/dbuser1/.bashrc**.

```
[dbuser1@serverb ~]$ echo "umask 007" >> .bash_profile  
[dbuser1@serverb ~]$ echo "umask 007" >> .bashrc
```

- 5.3. Quittez le shell de l'utilisateur dbuser1.

```
[dbuser1@serverb ~]$ exit  
logout  
[root@serverb ~]#
```

6. Créez un répertoire appelé **/home/student/grading/review2** avec student et database en tant qu'utilisateur et que groupe propriétaires, respectivement. Configurez les autorisations sur ce répertoire afin que tout nouveau fichier qu'il contient hérite de database en tant que groupe propriétaire, indépendamment de l'utilisateur qui l'a créé. Les autorisations sur **/home/student/grading/review2** doivent permettre aux membres du groupe database et à l'utilisateur student d'accéder au répertoire et d'y créer du contenu. Tous les autres utilisateurs doivent posséder des autorisations de lecture et d'exécution sur le répertoire. Assurez-vous également que les utilisateurs ne sont autorisés à supprimer de **/home/student/grading/review2** que les fichiers dont ils sont propriétaires, et non les fichiers qui ne leur appartiennent pas.

- 6.1. Utilisez la commande **mkdir** pour créer **/home/student/grading/review2**.

```
[root@serverb ~]# mkdir /home/student/grading/review2
```

- 6.2. Sur **/home/student/grading/review2**, utilisez la commande **chown** pour définir student et database en tant qu'utilisateur et que groupe propriétaires, respectivement.

```
[root@serverb ~]# chown student:database /home/student/grading/review2
```

- 6.3. Utilisez la commande **chmod** pour appliquer l'autorisation spéciale SetGID sur **/home/student/grading/review2**.

```
[root@serverb ~]# chmod g+s /home/student/grading/review2
```

- 6.4. Utilisez la commande **chmod** pour appliquer le mode d'autorisation **775** sur **/home/student/grading/review2**.

```
[root@serverb ~]# chmod 775 /home/student/grading/review2
```

- 6.5. Utilisez la commande **chmod** pour appliquer l'autorisation spéciale stickybit sur **/home/student/grading/review2**.

```
[root@serverb ~]# chmod o+t /home/student/grading/review2
```

- 6.6. Quittez le shell de l'utilisateur root.

```
[root@serverb ~]# exit  
logout  
[student@serverb ~]$
```

- 6.7. Déconnectez-vous de serverb.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

Évaluation

Sur workstation, exécutez la commande **lab rhcsa-rh124-review2 grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 grade
```

Fin

Sur workstation, exécutez **lab rhcsa-rh124-review2 finish** pour terminer la révision complète. Ce script met fin au processus, et supprime les fichiers et les répertoires créés au début de la révision complète et garantit que l'environnement sur serverb est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 finish
```

Vous avez maintenant terminé la révision complète.

► OPEN LAB

CONFIGURATION ET GESTION D'UN SERVEUR

Au cours de cette révision, vous allez configurer, sécuriser et utiliser le service SSH pour accéder à la machine distante, configurer le service `rsyslog`, archiver des fichiers locaux, transférer des fichiers locaux vers une machine distante et gérer des paquetages à l'aide de `yum`.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Créer une paire de clés SSH.
- Désactiver les connexions SSH en tant qu'utilisateur `root`.
- Désactiver les connexions SSH à l'aide d'un mot de passe.
- Mettre à jour le fuseau horaire d'un serveur.
- Installer des paquets et des modules de paquetages à l'aide de `yum`.
- Archiver des fichiers locaux en vue de la sauvegarde.
- Transférer des fichiers locaux sur une machine distante.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review3 start` pour commencer la révision complète. Ce script crée les fichiers nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.

- Générez des clés SSH pour l'utilisateur `student` sur `serverb`. Ne protégez pas la clé privée avec une phrase de passe.
- Sur `servera`, configurez l'utilisateur `student` afin qu'il accepte les connexions authentifiées par la paire de clés SSH que vous avez créée pour l'utilisateur `student` sur `serverb`. L'utilisateur `student` sur `serverb` doit être en mesure de se connecter à `servera` à l'aide de SSH sans entrer de mot de passe.
- Sur `serverb`, configurez le service `sshd` pour empêcher les utilisateurs de se connecter en tant que `root` via SSH.

CHAPITRE 17 | Révision complète

- Sur **serverb**, configurez le service `sshd` pour empêcher les utilisateurs d'utiliser leurs mots de passe pour se connecter. Les utilisateurs doivent toujours pouvoir authentifier les connexions à l'aide d'une paire de clés SSH.
- Créez une archive tar nommée **/tmp/log.tar** qui inclut le contenu de **/var/log** sur **serverb**. Transférez à distance l'archive tar vers le répertoire **/tmp** sur **servera**, en vous authentifiant comme **student** en utilisant la clé privée de la paire de clés SSH de l'utilisateur **student**.
- Configurez le service `rsyslog` sur **serverb** pour enregistrer tous les messages reçus avec le niveau de priorité **debug** ou supérieur dans le fichier **/var/log/grading-debug**. Cette configuration doit être définie dans un fichier **/etc/rsyslog.d/grading-debug.conf**, que vous devez créer.
- Installez le paquetage `zsh`, disponible dans le référentiel `BaseOS`, sur **serverb**.
- Activer le flux de modules par défaut pour le module `python36` et installez tous les paquetages fournis à partir de ce flux sur **serverb**.
- Définissez le fuseau horaire de **serverb** sur **Asia/Kolkata**.

Évaluation

Sur **workstation**, exécutez la commande **lab rhcsa-rh124-review3 grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 grade
```

Fin

Sur **workstation**, exécutez **lab rhcsa-rh124-review3 finish** pour terminer la révision complète. Ce script supprime les fichiers et les répertoires créés au début de la révision complète et garantit que l'environnement sur **serverb** est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 finish
```

Vous avez maintenant terminé la révision complète.

► SOLUTION

CONFIGURATION ET GESTION D'UN SERVEUR

Au cours de cette révision, vous allez configurer, sécuriser et utiliser le service SSH pour accéder à la machine distante, configurer le service `rsyslog`, archiver des fichiers locaux, transférer des fichiers locaux vers une machine distante et gérer des paquetages à l'aide de `yum`.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Créer une paire de clés SSH.
- Désactiver les connexions SSH en tant qu'utilisateur `root`.
- Désactiver les connexions SSH à l'aide d'un mot de passe.
- Mettre à jour le fuseau horaire d'un serveur.
- Installer des paquets et des modules de paquetages à l'aide de `yum`.
- Archiver des fichiers locaux en vue de la sauvegarde.
- Transférer des fichiers locaux sur une machine distante.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review3 start` pour commencer la révision complète. Ce script crée les fichiers nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.

- Générez des clés SSH pour l'utilisateur `student` sur `serverb`. Ne protégez pas la clé privée avec une phrase de passe.
- Sur `servera`, configurez l'utilisateur `student` afin qu'il accepte les connexions authentifiées par la paire de clés SSH que vous avez créée pour l'utilisateur `student` sur `serverb`. L'utilisateur `student` sur `serverb` doit être en mesure de se connecter à `servera` à l'aide de SSH sans entrer de mot de passe.
- Sur `serverb`, configurez le service `sshd` pour empêcher les utilisateurs de se connecter en tant que `root` via SSH.

CHAPITRE 17 | Révision complète

- Sur **serverb**, configurez le service **sshd** pour empêcher les utilisateurs d'utiliser leurs mots de passe pour se connecter. Les utilisateurs doivent toujours pouvoir authentifier les connexions à l'aide d'une paire de clés SSH.
- Créez une archive tar nommée **/tmp/log.tar** qui inclut le contenu de **/var/log** sur **serverb**. Transférez à distance l'archive tar vers le répertoire **/tmp** sur **servera**, en vous authentifiant comme **student** en utilisant la clé privée de la paire de clés SSH de l'utilisateur **student**.
- Configurez le service **rsyslog** sur **serverb** pour enregistrer tous les messages reçus avec le niveau de priorité **debug** ou supérieur dans le fichier **/var/log/grading-debug**. Cette configuration doit être définie dans un fichier **/etc/rsyslog.d/grading-debug.conf**, que vous devez créer.
- Installez le paquetage **zsh**, disponible dans le référentiel **BaseOS**, sur **serverb**.
- Activer le flux de modules par défaut pour le module **python36** et installez tous les paquetages fournis à partir de ce flux sur **serverb**.
- Définissez le fuseau horaire de **serverb** sur **Asia/Kolkata**.

1. Générez des clés SSH pour l'utilisateur **student** sur **serverb**. Ne protégez pas la clé privée avec une phrase de passe.

- 1.1. À partir de **workstation**, ouvrez une session SSH sur **serverb** en tant que **student**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Utilisez la commande **ssh-keygen** pour générer des clés SSH pour l'utilisateur **student**. Les fichiers de clés privées et publiques doivent être nommés **/home/student/.ssh/review3_key** et **/home/student/.ssh/review3_key.pub**, respectivement.

```
[student@serverb ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/student/.ssh/id_rsa): /home/
student/.ssh/review3_key
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /home/student/.ssh/review3_key.
Your public key has been saved in /home/student/.ssh/review3_key.pub.
The key fingerprint is:
SHA256:Uqefehw+vRfm94fQZDoz/6IfNYSLK/OpiQ4n6lrKIBY student@serverb.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
|                   |
|                   . |
|       . . . .   |
|       . o . =   |
|       . S . * ..|
```

CHAPITRE 17 | Révision complète

```
|      . . .B +..|
|.o . o . =o+ 0.o |
|+= . + ..X o * .o|
| Eoo .o.+.+o=.+=|
+---[SHA256]---
```

2. Sur servera, configurez l'utilisateur student afin qu'il accepte les connexions authentifiées par la paire de clés SSH que vous avez créée pour l'utilisateur student sur serverb. L'utilisateur student sur serverb doit être en mesure de se connecter à servera à l'aide de SSH sans entrer de mot de passe.

- 2.1. Utilisez la commande **ssh-copy-id** pour exporter la clé publique **/home/student/.ssh/review3_key.pub** de servera vers serverb.

```
[student@serverb ~]$ ssh-copy-id -i .ssh/review3_key.pub student@servera
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ".ssh/review3.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys
student@servera's password: student

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'student@servera'"
and check to make sure that only the key(s) you wanted were added.
```

- 2.2. Utilisez la commande **ssh** pour vérifier que vous pouvez vous connecter à servera à partir de serverb en tant que student en utilisant la clé privée SSH **/home/student/.ssh/review3_key** sans être invité à saisir le mot de passe.

```
[student@serverb ~]$ ssh -i .ssh/review3_key student@servera
...output omitted...
[student@servera ~]$
```

- 2.3. Déconnectez-vous de servera.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@serverb ~]$
```

3. Sur serverb, configurez le service sshd pour empêcher les utilisateurs de se connecter en tant que root avec SSH.

- 3.1. Définissez le paramètre **PermitRootLogin** sur **no** dans le fichier **/etc/ssh/sshd_config**. Vous pouvez utiliser la commande **sudo vim /etc/ssh/sshd_config** pour modifier le fichier de configuration.

- 3.2. Relancez le service sshd.

```
[student@serverb ~]$ sudo systemctl reload sshd.service
```

CHAPITRE 17 | Révision complète

4. Sur **serverb**, configurez le service **sshd** pour empêcher les utilisateurs d'utiliser leurs mots de passe pour se connecter. Les utilisateurs doivent toujours pouvoir authentifier les connexions à l'aide de leur clé privée de la paire de clés SSH.
- 4.1. Définissez le paramètre **PasswordAuthentication** sur **no** dans le fichier **/etc/ssh/sshd_config**. Vous pouvez utiliser la commande **sudo vim /etc/ssh/sshd_config** pour modifier le fichier de configuration.
 - 4.2. Utilisez la commande **sudo systemctl** pour relancer le service **sshd**.

```
[student@serverb ~]$ sudo systemctl reload sshd.service
```

5. Créez une archive tar nommée **/tmp/log.tar** qui inclut le contenu de **/var/log** sur **serverb**. Transférez à distance l'archive tar vers le répertoire **/tmp** sur **servera**, en vous authentifiant comme **student** en utilisant **/home/student/.ssh/review3_key** comme clé privée de la paire de clés SSH de l'utilisateur **student** pour l'authentification.
- 5.1. Utilisez la commande **sudo tar** pour créer une archive nommée **/tmp/log.tar** en tant que superutilisateur et insérez-y le contenu de **/var/log**.

```
[student@serverb ~]$ sudo tar -cvf /tmp/log.tar /var/log  
[sudo] password for student: student  
...output omitted...
```

- 5.2. Utilisez la commande **scp** pour transférer à distance le fichier d'archive **/tmp/log.tar** vers le répertoire **/tmp** sur **servera**. Spécifiez **/home/student/.ssh/review3_key** en tant que clé privée de la paire de clés SSH.

```
[student@serverb ~]$ scp -i .ssh/review3_key /tmp/log.tar student@servera:/tmp  
log.tar                                         100%    14MB   57.4MB/s   00:00
```

6. Configurez le service **rsyslog** sur **serverb** pour enregistrer tous les messages reçus avec le niveau de priorité **debug** ou supérieur dans le fichier **/var/log/grading-debug**. Cette configuration doit être définie dans un fichier **/etc/rsyslog.d/grading-debug.conf**, que vous devez créer.
- 6.1. Créez le fichier **/etc/rsyslog.d/grading-debug.conf** avec le contenu suivant. Vous pouvez utiliser la commande **sudo vim /etc/rsyslog.d/grading-debug.conf** pour créer le fichier.

```
* .debug /var/log/grading-debug
```

- 6.2. Utilisez la commande **sudo systemctl** pour redémarrer le service **rsyslog**.

```
[student@serverb ~]$ sudo systemctl restart rsyslog.service
```

- 6.3. Utilisez la commande **logger** pour générer le message de journal **Debug Testing** ayant la priorité **debug**.

```
[student@serverb ~]$ logger -p debug Debug Testing
```

- 6.4. Vérifiez que le message de journal **Debug Testing** est bien enregistré dans le fichier **/var/log/grading-debug**.

```
[student@serverb ~]$ sudo tail /var/log/grading-debug
...output omitted...
Mar 12 09:55:23 serverb student[32383]: Debug Testing
```

7. Utilisez la commande **sudo yum** pour installer le paquetage *zsh*, disponible dans le référentiel BaseOS, sur *serverb*.

```
[student@serverb ~]$ sudo yum install zsh
...output omitted...
Is this ok [y/N]: y
...output omitted...
Installed:
  zsh-5.5.1-6.el8.x86_64

Complete!
```

8. Utilisez la commande **yum** pour activer le flux de modules par défaut pour le module *python36* et installez tous les paquetages fournis à partir de ce flux sur *serverb*.

```
[student@serverb ~]$ sudo yum module install python36
...output omitted...
Is this ok [y/N]: y
...output omitted...
Installed:
  python36-3.6.6-18.module+el8+2339+1a6691f8.x86_64           python3-
  pip-9.0.3-13.el8.noarch

Complete!
```

9. Définissez le fuseau horaire de *serverb* sur **Asia/Kolkata**.

- 9.1. Utilisez la commande **sudo timedatectl** pour définir le fuseau horaire de *serverb* sur **Asia/Kolkata**.

```
[student@serverb ~]$ sudo timedatectl set-timezone Asia/Kolkata
```

- 9.2. Déconnectez-vous de *serverb*.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

Évaluation

Sur *workstation*, exécutez la commande **lab rhcsa-rh124-review3 grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 grade
```

Fin

Sur `workstation`, exéutez `lab rhcsa-rh124-review3 finish` pour terminer la révision complète. Ce script supprime les fichiers et les répertoires créés au début de la révision complète et garantit que l'environnement sur `serverb` est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 finish
```

Vous avez maintenant terminé la révision complète.

► OPEN LAB

GESTION DE RÉSEAUX

Au cours de cette révision, vous allez configurer et tester la connectivité réseau.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Configurer les paramètres réseau.
- Tester la connectivité réseau.
- Définir un nom d'hôte statique pour le système.
- Utiliser des noms d'hôte canoniques pouvant être résolus localement pour vous connecter aux systèmes.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review4 start` pour commencer la révision complète. Ce script crée les fichiers nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.



MISE EN GARDE

Il est utile d'apporter des modifications au réseau à partir de la console du serveur, que ce soit localement ou via le matériel d'accès à la console distante. Lors de l'utilisation de `ssh` pour ajuster les paramètres réseau, une commande erronée peut bloquer ou verrouiller votre session. Il faut donc effectuer les corrections de la configuration réseau via la console.

Dans la page Web qui contrôle votre environnement d'atelier, cliquez sur le bouton **OPEN CONSOLE** pour `serverb`. Un onglet s'ouvre dans votre navigateur avec la session de console pour `serverb`. Connectez-vous en tant qu'utilisateur `student` à l'invite.

- Déterminez le nom de l'interface Ethernet et son profil de connexion active sur `serverb`.
- Sur `serverb`, créez un profil de connexion appelé `static` pour l'interface Ethernet disponible qui définit les paramètres réseau de manière statique et n'utilise pas DHCP. Utilisez les paramètres du tableau suivant :

Adresse IPv4	172.25.250.111
Masque de réseau	255.255.255.0
Passerelle	172.25.250.254
Serveur DNS	172.25.250.254

Définissez l'interface Ethernet du serveur de manière à utiliser les paramètres réseau mis à jour affichés dans le tableau ci-dessus.

- Assurez-vous que le nom d'hôte de **serverb** est défini de manière statique sur **server-review4.lab4.example.com**.
- Sur **serverb**, définissez **client-review4** comme nom d'hôte canonique pour l'adresse IPv4 **172.25.250.10** de l'hôte **servera.lab.example.com**.
- Configurez l'adresse IPv4 supplémentaire **172.25.250.211** avec le masque de réseau **255.255.255.0** sur la même interface de **serverb** que celle qui possède les paramètres réseau statiques existants. Ne supprimez pas l'adresse IPv4 existante. Assurez-vous que **serverb** répond à toutes les adresses lorsque la connexion que vous avez configurée de manière statique sur son interface est active.
- Sur **serverb**, rétablissez les paramètres réseau d'origine en activant la connexion réseau d'origine et en désactivant la connexion réseau **static** que vous avez créée manuellement.

Évaluation

Sur **workstation**, exécutez la commande **lab rhcsa-rh124-review4 grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 grade
```

Fin

Sur **workstation**, exécutez **lab rhcsa-rh124-review4 finish** pour terminer la révision complète. Ce script supprime les fichiers et les répertoires créés au début de la révision complète et garantit que l'environnement sur **serverb** est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 finish
```

Vous avez maintenant terminé la révision complète.

► SOLUTION

GESTION DE RÉSEAUX

Au cours de cette révision, vous allez configurer et tester la connectivité réseau.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Configurer les paramètres réseau.
- Tester la connectivité réseau.
- Définir un nom d'hôte statique pour le système.
- Utiliser des noms d'hôte canoniques pouvant être résolus localement pour vous connecter aux systèmes.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review4 start` pour commencer la révision complète. Ce script crée les fichiers nécessaires pour configurer correctement l'environnement.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.



MISE EN GARDE

Il est utile d'apporter des modifications au réseau à partir de la console du serveur, que ce soit localement ou via le matériel d'accès à la console distante. Lors de l'utilisation de `ssh` pour ajuster les paramètres réseau, une commande erronée peut bloquer ou verrouiller votre session. Il faut donc effectuer les corrections de la configuration réseau via la console.

Dans la page Web qui contrôle votre environnement d'atelier, cliquez sur le bouton OPEN CONSOLE pour `serverb`. Un onglet s'ouvre dans votre navigateur avec la session de console pour `serverb`. Connectez-vous en tant qu'utilisateur `student` à l'invite.

- Déterminez le nom de l'interface Ethernet et son profil de connexion active sur `serverb`.
- Sur `serverb`, créez un profil de connexion appelé `static` pour l'interface Ethernet disponible qui définit les paramètres réseau de manière statique et n'utilise pas DHCP. Utilisez les paramètres du tableau suivant :

Adresse IPV4	172.25.250.111
Masque de réseau	255.255.255.0
Passerelle	172.25.250.254
Serveur DNS	172.25.250.254

Définissez l'interface Ethernet du serveur de manière à utiliser les paramètres réseau mis à jour affichés dans le tableau ci-dessus.

- Assurez-vous que le nom d'hôte de **serverb** est défini de manière statique sur **server-review4.lab4.example.com**.
- Sur **serverb**, définissez **client-review4** comme nom d'hôte canonique pour l'adresse IPv4 **172.25.250.10** de l'hôte **servera.lab.example.com**.
- Configurez l'adresse IPv4 supplémentaire **172.25.250.211** avec le masque de réseau **255.255.255.0** sur la même interface de **serverb** que celle qui possède les paramètres réseau statiques existants. Ne supprimez pas l'adresse IPv4 existante. Assurez-vous que **serverb** répond à toutes les adresses lorsque la connexion que vous avez configurée de manière statique sur son interface est active.
- Sur **serverb**, rétablissez les paramètres réseau d'origine en activant la connexion réseau d'origine et en désactivant la connexion réseau **static** que vous avez créée manuellement.

- Utilisez la console pour vous connecter en tant que **student** à **serverb** localement.

Dans la page Web qui contrôle votre environnement d'atelier, cliquez sur le bouton OPEN CONSOLE pour **serverb**. Un onglet s'ouvre dans votre navigateur avec la session de console pour **serverb**. Connectez-vous en tant qu'utilisateur **student** à l'invite.

Il est utile d'apporter des modifications au réseau à partir de la console du serveur, que ce soit localement ou via le matériel d'accès à la console distante. Lors de l'utilisation de **ssh** pour ajuster les paramètres réseau, une commande erronée peut bloquer ou verrouiller votre session. Il faut donc effectuer les corrections de la configuration réseau via la console.

- Déterminez le nom de l'interface Ethernet sur **serverb** et le profil de connexion active utilisé.

Dans cet exemple, **enX** est le nom de l'interface Ethernet. Le nom du profil de connexion est **Connexion filaire 1**. Créez le profil de connexion **static** pour cette interface.

Les noms de l'interface réseau et du profil de connexion initial peuvent être différents sur votre ordinateur **serverb**. Utilisez le nom indiqué par votre système pour remplacer le marqueur **enX** dans les étapes de cette solution.

- Sur **serverb**, créez un profil de connexion appelé **static** pour l'interface Ethernet disponible. Définissez les paramètres réseau de manière statique, de sorte que DHCP ne soit pas utilisé. Définissez les paramètres sur la base du tableau suivant :

Adresse IPV4	172.25.250.111
Masque de réseau	255.255.255.0

Passerelle	172.25.250.254
Serveur DNS	172.25.250.254

L'interface Ethernet du serveur **serverb** doit utiliser les paramètres réseau mis à jour, comme indiqué dans le tableau précédent.

- Utilisez **nmcli** pour créer la connexion **static** avec les paramètres réseau indiqués.

```
[student@serverb ~]$ sudo nmcli connection add con-name static type ethernet \
  ifname enX ipv4.addresses '172.25.250.111/24' ipv4.gateway '172.25.250.254' \
  ipv4.dns '172.25.250.254' ipv4.method manual
[sudo] password for student: student
Connection 'static' (ac8620e6-b77e-499f-9931-118b8b015807) successfully added.
```

- Utilisez la commande **nmcli** pour activer les nouveaux paramètres de connexion.

```
[student@serverb ~]$ sudo nmcli connection up static
```

- Utilisez la commande **hostnamectl** pour définir le nom d'hôte du **serverb** sur **server-review4.lab4.example.com**. Vérifiez le nouveau nom d'hôte.

```
[student@serverb ~]$ sudo hostnamectl set-hostname server-review4.lab4.example.com
[sudo] password for student: student
[student@serverb ~]$ hostname
server-review4.lab4.example.com
```

- Sur **serverb**, modifiez le fichier **/etc/hosts** pour définir **client-review4** comme nom d'hôte canonique pour l'adresse IPv4 **172.25.250.10** de l'hôte **servera.lab.example.com**.

- Modifiez le fichier **/etc/hosts** pour ajouter **client-review4** en tant que nom pour l'adresse IPv4 **172.25.250.10**.

```
172.25.250.10 servera.lab.example.com servera client-review4
```

- Utilisez la commande **ping** pour vérifier que vous pouvez accéder à **172.25.250.10** en utilisant le nom d'hôte canonique **client-review4**.

```
[student@serverb ~]$ ping -c2 client-review4
PING servera.lab.example.com (172.25.250.10) 56(84) bytes of data.
64 bytes from servera.lab.example.com (172.25.250.10): icmp_seq=1 ttl=64
  time=0.259 ms
64 bytes from servera.lab.example.com (172.25.250.10): icmp_seq=2 ttl=64
  time=0.391 ms

--- servera.lab.example.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 33ms
rtt min/avg/max/mdev = 0.259/0.325/0.391/0.066 ms
```

- Modifiez le profil de connexion **static** pour configurer l'adresse IPv4 **172.25.250.211** avec le masque de réseau **255.255.255.0** sur la même interface **serverb** ayant des

paramètres statiques existants. Ne supprimez pas l'adresse IPv4 existante. Vérifiez que **serverb** répond à toutes les adresses lorsque le profil de connexion modifié est actif.

- Utilisez la commande **nmcli** pour ajouter la nouvelle adresse IP.

```
[student@serverb ~]$ sudo nmcli connection modify static \
+ipv4.addresses '172.25.250.211/24'
```

- Utilisez la commande **nmcli** pour activer la nouvelle adresse IP.

```
[student@serverb ~]$ sudo nmcli connection up static
...output omitted...
```

- À partir de **workstation**, utilisez la commande **ping** pour vérifier que l'adresse IPv4 172.25.250.211 est accessible.

```
[student@workstation ~]$ ping -c2 172.25.250.211
PING 172.25.250.211 (172.25.250.211) 56(84) bytes of data.
64 bytes from 172.25.250.211: icmp_seq=1 ttl=64 time=0.246 ms
64 bytes from 172.25.250.211: icmp_seq=2 ttl=64 time=0.296 ms

--- 172.25.250.211 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 50ms
rtt min/avg/max/mdev = 0.246/0.271/0.296/0.025 ms
```

- Sur **serverb**, rétablissez les paramètres d'origine en activant la connexion réseau d'origine.

- Revenez à la console et utilisez la commande **nmcli** pour activer le profil réseau d'origine.

```
[student@serverb ~]$ sudo nmcli connection up "Wired connection 1"
...output omitted...
```

Le nom du profil de connexion d'origine peut être différent sur votre ordinateur **serverb**. Remplacez le nom affiché dans cette solution par celui de votre système. Trouvez le nom avec **nmcli connection show**.

- À partir de **workstation**, ouvrez une session SSH sur **serverb** en tant que **student** pour vérifier que les paramètres réseau d'origine ont bien été activés.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@server-review4 ~]$
```

- Déconnectez-vous de **serverb** et quittez tous les terminaux sauf un sur **workstation**.

```
[student@server-review4 ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

Évaluation

Sur workstation, exécutez la commande **lab rhcsa-rh124-review4 grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 grade
```

Fin

Sur workstation, exécutez **lab rhcsa-rh124-review4 finish** pour terminer la révision complète. Ce script supprime les fichiers et les répertoires créés au début de la révision complète et garantit que l'environnement sur serverb est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 finish
```

Vous avez maintenant terminé la révision complète.

► OPEN LAB

MONTAGE DE SYSTÈMES DE FICHIERS ET RECHERCHE DE FICHIERS

Au cours de cette révision, vous allez monter un système de fichiers et rechercher des fichiers en fonction de différents critères.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Monter un système de fichiers existant.
- Rechercher des fichiers en fonction du nom de fichier, des autorisations et de la taille.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review5 start` pour lancer la révision complète. Ce script crée le système de fichiers, les comptes d'utilisateurs et les comptes de groupes nécessaires.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.

- Sur `serverb`, un périphérique bloc contenant le système de fichiers XFS existe, mais n'est pas encore monté. Déterminez le périphérique bloc et montez-le sur le répertoire `/review5-disk`. Si nécessaire, créez le répertoire `/review5-disk`.
- Sur `serverb`, recherchez le fichier nommé `review5-path`. Créez un fichier nommé `/review5-disk/review5.txt` qui contient une seule ligne constituée du chemin d'accès absolu au fichier `review5`.
- Sur `serverb`, recherchez tous les fichiers dont l'utilisateur et le groupe propriétaires sont, respectivement, `contractor1` et `contractor`. Les fichiers doivent également posséder les autorisations octales `640`. Enregistrez la liste de ces fichiers dans `/review5-disk/review5-perms.txt`.
- Sur `serverb`, recherchez tous les fichiers dont la taille est de 100 octets. Enregistrez les chemins absolus de ces fichiers dans `/review5-disk/review5-size.txt`.

Évaluation

Sur `workstation`, exécutez la commande `lab rhcsa-rh124-review5 grade` pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 grade
```

Fin

Sur workstation, exécutez **lab rhcsa-rh124-review5 finish** pour terminer la révision complète. Ce script supprime le système de fichiers, les comptes d'utilisateurs et les comptes de groupes créés au début de la révision complète, et garantit que l'environnement sur serverb est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 finish
```

Vous avez maintenant terminé la révision complète.

► SOLUTION

MONTAGE DE SYSTÈMES DE FICHIERS ET RECHERCHE DE FICHIERS

Au cours de cette révision, vous allez monter un système de fichiers et rechercher des fichiers en fonction de différents critères.

RÉSULTATS

Vous serez en mesure de réaliser les tâches suivantes :

- Monter un système de fichiers existant.
- Rechercher des fichiers en fonction du nom de fichier, des autorisations et de la taille.

AVANT DE COMMENCER

Connectez-vous à `workstation` en tant qu'utilisateur `student` avec le mot de passe `student`.

Sur `workstation`, exécutez `lab rhcsa-rh124-review5 start` pour lancer la révision complète. Ce script crée le système de fichiers, les comptes d'utilisateurs et les comptes de groupes nécessaires.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 start
```

INSTRUCTIONS

Exécutez les tâches suivantes sur `serverb` pour réaliser l'exercice.

- Sur `serverb`, un périphérique bloc contenant le système de fichiers XFS existe, mais n'est pas encore monté. Déterminez le périphérique bloc et montez-le sur le répertoire `/review5-disk`. Si nécessaire, créez le répertoire `/review5-disk`.
- Sur `serverb`, recherchez le fichier nommé `review5-path`. Créez un fichier nommé `/review5-disk/review5.txt` qui contient une seule ligne constituée du chemin d'accès absolu au fichier `review5`.
- Sur `serverb`, recherchez tous les fichiers dont l'utilisateur et le groupe propriétaires sont, respectivement, `contractor1` et `contractor`. Les fichiers doivent également posséder les autorisations octales `640`. Enregistrez la liste de ces fichiers dans `/review5-disk/review5-perms.txt`.
- Sur `serverb`, recherchez tous les fichiers dont la taille est de 100 octets. Enregistrez les chemins absolus de ces fichiers dans `/review5-disk/review5-size.txt`.

1. Sur `serverb`, montez le périphérique bloc inactif contenant le système de fichiers XFS sur le répertoire `/review5-disk`.
 - 1.1. À partir de `workstation`, ouvrez une session SSH sur `serverb` en tant que `student`.

CHAPITRE 17 | Révision complète

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

- 1.2. Utilisez la commande **lsblk -fs** pour déterminer le périphérique bloc inactif contenant le système de fichiers XFS.

```
[student@serverb ~]$ lsblk -fs  
NAME FSTYPE LABEL UUID MOUNTPOINT  
...output omitted...  
vdb1 xfs 3d97c5ef-23e7-4c1c-a9be-d5c475b3d0d5  
└vdb  
...output omitted...
```

Sur base de la sortie précédente, notez que le périphérique bloc vdb1 contient le système de fichiers XFS, qui n'est monté sur aucun répertoire.

- 1.3. Utilisez la commande **sudo mkdir** pour créer le répertoire **/review5-disk** en tant que superutilisateur. Lorsque la commande **sudo** vous invite à saisir un mot de passe, indiquez **student**.

```
[student@serverb ~]$ sudo mkdir /review5-disk  
[sudo] password for student: student
```

- 1.4. Utilisez la commande **sudo mount** pour monter le périphérique bloc vdb1 sur le répertoire **/review5-disk** en tant que superutilisateur.

```
[student@serverb ~]$ sudo mount /dev/vdb1 /review5-disk
```

- 1.5. Vérifiez que le périphérique bloc vdb1 est bien monté sur le répertoire **/review5-disk**.

```
[student@serverb ~]$ df -Th  
Filesystem Type Size Used Avail Use% Mounted on  
...output omitted...  
/dev/vdb1 xfs 2.0G 47M 2.0G 3% /review5-disk  
...output omitted...
```

2. Sur serverb, recherchez le fichier nommé **review5-path**. Enregistrez son chemin absolu dans le fichier texte **/review5-disk/review5-path.txt**.

- 2.1. Utilisez la commande **find** pour rechercher le fichier nommé **review5-path**. Redirigez toutes les erreurs de la commande **find** vers **/dev/null**. Cette redirection vous permet d'éliminer toute erreur de la sortie de la commande **find**.

```
[student@serverb ~]$ find / -iname review5-path 2>/dev/null  
/var/tmp/review5-path
```

Notez le chemin d'accès absolu au fichier **review5-path** dans la sortie précédente.

2.2. Créez le fichier texte **/review5-disk/review5-path.txt**. Enregistrez le chemin d'accès absolu au fichier **review5-path**, comme déterminé à l'étape précédente, dans le fichier texte **/review5-disk/review5-path.txt**. Vous pouvez utiliser la commande **sudo vim /review5-disk/review5-path.txt** pour créer le fichier texte. Entrez **:wq!** à partir du mode de commande dans **vim** pour enregistrer les modifications et quitter le fichier. La sortie suivante montre le contenu du fichier texte **/review5-disk/review5-path.txt**.

```
/var/tmp/review5-path
```

3. Sur **serverb**, recherchez tous les fichiers dont l'utilisateur et le groupe propriétaires sont, respectivement, **contractor1** et **contractor**. Les fichiers doivent également posséder les autorisations octales 640. Enregistrez les chemins d'accès absolus à tous ces fichiers dans le fichier texte **/review5-disk/review5-perms.txt**.
 - 3.1. Utilisez les options **-user**, **-group** et **-perm** avec la commande **find** pour rechercher tous les fichiers dont l'utilisateur propriétaire, le groupe propriétaire et les autorisations octales sont respectivement **contractor1**, **contractor** et **640**. Redirigez toutes les erreurs de la commande **find** vers **/dev/null**.

```
[student@serverb ~]$ find / -user contractor1 \
-group contractor \
-perm 640 2>/dev/null
/usr/share/review5-perms
```

Notez le chemin d'accès absolu au fichier **review5-perms** dans la sortie précédente. Le fichier **/usr/share/review5-perms** est le seul qui répond aux critères de la commande **find** précédente.

3.2. Créez le fichier texte **/review5-disk/review5-perms.txt**. Enregistrez le chemin d'accès absolu au seul fichier (**review5-perms**) dont l'utilisateur propriétaire, le groupe propriétaire et les autorisations octales sont respectivement **contractor1**, **contractor** et **640**, comme déterminé à l'étape précédente, dans le **/review5-disk/review5-perms.txt** fichier texte. Vous pouvez utiliser la commande **sudo vim /review5-disk/review5-perms.txt** pour créer le fichier texte. Entrez **:wq!** à partir du mode de commande dans **vim** pour enregistrer les modifications et quitter le fichier. La sortie suivante montre le contenu du fichier texte **/review5-disk/review5-perms.txt**.

```
/usr/share/review5-perms
```

4. Sur **serverb**, recherchez tous les fichiers dont la taille est de 100 octets. Enregistrez les chemins d'accès absolus à tous ces fichiers dans le fichier **/review5-disk/review5-size.txt**.
 - 4.1. Utilisez l'option **-size** avec la commande **find** pour rechercher tous les fichiers dont la taille est de 100 octets. Redirigez toutes les erreurs de la commande **find** vers **/dev/null**.

```
[student@serverb ~]$ find / -size 100c 2>/dev/null
/dev/disk
/run/initramfs
/etc/lvm
```

CHAPITRE 17 | Révision complète

```
/etc/audit  
/etc/sos.conf  
/usr/lib/python3.6/site-packages/dnf/conf  
/usr/lib/python3.6/site-packages/ptyprocess  
/usr/share/licenses/ethtool/LICENSE  
/usr/share/doc/libuser  
/usr/share/doc/python3-cryptography/docs/x509  
/usr/share/doc/python3-jinja2/ext  
/usr/share/doc/plymouth/AUTHORS  
/usr/share/vim/vim80/macros/maze/main.aap  
/usr/libexec/plymouth  
/opt/review5-size
```

La sortie précédente peut varier dans votre système en fonction du nombre de fichiers dont la taille est de 100 octets. Notez les chemins d'accès absous à tous les fichiers dans la sortie précédente.

- 4.2. Créez le fichier texte **/review5-disk/review5-size.txt**. Enregistrez les chemins d'accès absous à tous les fichiers ayant une taille de 100 octets, comme déterminé à l'étape précédente, dans le fichier texte **/review5-disk/review5-size.txt**. Vous pouvez utiliser la commande **sudo vim /review5-disk/review5-size.txt** pour créer le fichier texte. Entrez :**wq!** à partir du mode de commande dans **vim** pour enregistrer les modifications et quitter le fichier. Le fichier texte **/review5-disk/review5-size.txt** doit notamment contenir le chemin d'accès absolu au fichier **review5-size**.

```
...output omitted...  
/opt/review5-size  
...output omitted...
```

- 4.3. Déconnectez-vous de serverb.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

Évaluation

Sur workstation, exécutez la commande **lab rhcsa-rh124-review5 grade** pour confirmer que l'exercice est réussi.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 grade
```

Fin

Sur workstation, exécutez **lab rhcsa-rh124-review5 finish** pour terminer la révision complète. Ce script supprime le système de fichiers, les comptes d'utilisateurs et les comptes de groupes créés au début de la révision complète, et garantit que l'environnement sur serverb est propre.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 finish
```

Vous avez maintenant terminé la révision complète.

