RED HAT ENTREPRISE LINUX

V9.4 - Administration

Elie C Pakabilond

June 23, 2025

A PROPOS DE MOI

Cybersecurity Expert - SOC Manager/ Lead Analyst - Snr. Network/ System Administrator - CEH - CPENT - Digital Forensics Analyst - ESCA, ISO 27002, ISO 27005, ISO 27701 - CISA - Oracle Database Cloud Administrator Certified Professional - KLCP - Red Hat Certified System Administrator - VMware Certified Professional - Azure Cloud Architect - Veeam Certified Engineer V12 – M.Sc. Information Technology

ELIE C PAKABILOND

Global Tech Mastermind IT Trainer – Senior IT Consultant

INTRODUCTION

Introduction à Red Hat Linux

Red Hat Linux est une distribution du système d'exploitation Linux qui a été l'une des premières à se populariser auprès des entreprises. Voici quelques grandes lignes pour comprendre son importance:

- **Origine et évolution**: Créée par Red Hat, Inc., cette distribution est devenue une référence dans le monde des logiciels open source. Red Hat Linux a évolué vers **Red Hat Enterprise Linux (RHEL)** , qui est aujourd'hui son produit phare destiné aux entreprises.
- **Fiabilité et sécurité:** Red Hat Enterprise Linux est reconnu pour sa robustesse, sa stabilité et ses mises à jour régulières, ce qui en fait un choix privilégié pour les infrastructures critiques.
- **Support et services:** Contrairement à d'autres distributions Linux gratuites, RHEL est accompagné d'un support professionnel et d'outils de gestion avancés, ce qui le rend adapté aux grandes entreprises.
- **Communauté et impact:** Red Hat contribue activement au développement de projets open source, et sa distribution RHEL sert de base à d'autres distributions comme **CentOS** et **Rocky Linux**.

Red Hat Linux a marqué l'histoire de Linux en entreprise et reste un acteur clé du monde des systèmes d'exploitation open source.

Abonnements et Licences Red Hat Linux

1. **Abonnements Red Hat (Modèle Principal)**: Red Hat utilise un modèle basé sur les abonnements plutôt que sur les licences traditionnelles. Les abonnements incluent:

   Ce qui est inclus dans un abonnement:

   - Accès aux logiciels (binaires RHEL et mises à jour)
   - Correctifs de sécurité (via Red Hat Network)
   - Rapports de conformité
   - Outils de gestion (Satellite, Insights)

   Principaux types d'abonnements:

   - **Type**: Physique - **Couverture**: 2 sockets (cœurs illimités) - **Idéal pour**: Serveurs bare metal
   - **Type**: Virtuel - **Couverture**: Machines virtuelles illimitées sur 1 hôte - **Idéal pour**: Environnements virtualisés
   - **Type**: Cloud - **Couverture**: Paiement à l'usage - **Idéal pour**: Déploiements AWS/Azure/GCP
   - **Type**: Développeur - **Couverture**: Gratuit pour 16 systèmes - **Idéal pour**: Usage non-production

2. **Fonctionnement des Abonnements**:

   - **Enregistrer le système**:

     Bash: sudo subscription-manager register - -username=< *compte* > - -password=< *motdepasse* >

   - **Attacher un abonnement**:

     Bash: sudo subscription-manager attach - -auto

   - **Vérification**:

     Bash: sudo subscription-manager list - -consumed

Abonnements et Licences Red Hat Linux (suite)

3. **Concepts de Licences**: Bien que Red Hat n'utilise pas de licences traditionnelles, ces termes sont importants:

Licences Effectives via Abonnements:

- **1 Abonnement Physique** = Licence pour 2 sockets CPU
- **1 Abonnement Virtuel** = Licence pour VMs illimitées sur l'hôte souscrit
- Rapports de conformité
- **1 Abonnement Cloud** = Licence pour le temps d'exécution de l'instance

Règles de Mesure:

- **Sockets** (pas les cœurs) déterminent les besoins physiques
- **Hyperthreading** compte comme un seul cœur
- **Cloud**: Facturation horaire

4. **Exigences de Conformité**:

- Les systèmes doivent être **enregistrés dans les 30 jours** après installation
- Les abonnements doivent couvrir **tous les systèmes de production**
- Les abonnements développeur **ne peuvent pas être utilisés en production**

5. **Options de Gestion des Abonnements** :

- **Directement avec Red Hat**
- **Via des partenaires** (revendeurs, MSPs)
- **Red Hat Satellite** pour les grands déploiements
- **Programme Cloud Access** pour le cloud public

6. **Considérations de Coût**:

- **Serveurs physiques**: $349-$1,299/an par abonnement 2-sockets
- **Virtualisation** : Nécessite un abonnement hôte
- **Niveaux de support**: Standard (heures ouvrables) vs Premium (24/7

Aspect	Abonnement Red Hat	Licence Traditionnelle
Paiement	Récurrent annuel	Ponctuel + support optionnel
Mises à jour	Incluses	Nécessitent souvent un contrat de maintenance séparé
Support	Inclus	Généralement un coût supplémentaire
Transfert	Autorisé entre systèmes	Généralement lié à une machine

Table: Différences Clés avec les Licences Traditionnelles

Exigences Matérielles pour Red Hat Enterprise Linux (RHEL) 9.4

Voici les spécifications matérielles officielles et recommandées pour exécuter RHEL 9.4:

Composant	Configuration Minimale
CPU	Architecture 64-bit x86_64, ARM (AArch64), IBM Power (ppc64le), ou IBM Z (s390x)
Cœurs	2 cœurs (1,5 GHz ou plus)
RAM	2 Go (4 Go recommandés)
Stockage	20 Go (minimum pour une installation minimale)
Cœurs	2 cœurs (1,5 GHz ou plus)
Espace Swap	Égal à la RAM (si inferieure a 2Go de RAM) ou 4Go (si superieure ou egale 2Go de RAM)

Table: Exigences Minimales

Composant	Spécification Recommandée
CPU	4+ cœurs (2,0 GHz ou plus)
Cœurs	8 Go+ (16Go pour la virtualisation/containers)
RAM	40Go+ (SSD/NVMe recommandé)
Stockage	20 Go (minimum pour une installation minimale)
Espace Swap	4-8Go (ou 1,5× la RAM si utilisation de l'hibernation)

Table: Configurations Recommandées pour la Production

Exigences pour Charges de Travail Spécialisées

1. **Installation Graphique (GNOME)**
   - **RAM**: 4Go minimum
   - **GPU** : Compatible accélération 3D (Intel/AMD/NVIDIA)
2. **Virtualisation (KVM)**
   - **CPU**: Extensions VT-x/AMD-V requises
   - RAM: 16Go+ (plus besoins des machines virtuelles)
   - **Stockage**: 100Go+ (approvisionnement fin recommandé)
3. **Déploiement de Containers**
   - **RAM**: 4Go de base + 1Go par container actif
   - **CPU**: 2 cœurs + 1 vCPU pour 5-10 containers

Architectures Prises en Charge

- **x86_64** (Intel/AMD standard)
- **ARM 64-bit** (AArch64)
- **IBM Power** (ppc64le)
- **IBM Z** (s390x)

Considérations Supplémentaires

1. **Partitionnement Disque**:
   - **/boot** : 1Go
   - **/boot/efi** : 200Mo (systèmes UEFI)
   - **/**: 20Go+ (système de fichiers racine)
   - **/home**: Partition séparée recommandée
2. **Exigences Réseau**:
   - Carte réseau 1Gbps (10Gbps recommandé pour les serveurs)
   - Support IPv4/IPv6
3. **Noyau Temps-Réel**:
   - Nécessite un CPU avec TSC constant
   - Surcharge mémoire supplémentaire de 1Go

Commandes de Vérification

Vérifiez votre matériel actuel:

- **lscpu**: CPU/RAM
- **df -h**: Espace disque
- **uname -m**: Architecture

Installation et Configuration de Base de Red Hat Enterprise Linux (RHEL) 9.4

1. **Préparation de l'Installation**

Téléchargement

- Obtenez l'ISO depuis le portail Red Hat
- Ou via commande:
  - **sudo subscription-manager release - -set=9.4**
  - **sudo dnf download RHEL-9.4.0-x86_64-dvd.iso**

Support d'Installation:

- **Méthodes**:
  - USB bootable (avec dd ou Rufus)
  - PXE (pour déploiements réseau)
  - Virtualisation (KVM/VMware/Hyper-V)

2. **Processus d'Installation**

- **Démarrage**
  1. Insérez le média d'installation
  2. Sélectionnez **Install Red Hat Enterprise Linux 9.4**

- **Configuration Clé**

Étape	Action
Langue	Français)
Clavier	Français (AZERTY))
Source	Auto-détectée (ISO/USB)
	Recommandé:
Partitionnement	**/boot** (1GiB)
	/ (20GiB min)
	/home (optionnel)
	swap (4GiB)
Réseau	Activer Ethernet/WiFi
Sécurité	Activer le pare-feu (firewalld)
Utilisateur	Créer un compte admin

Table: Configurations Recommandées pour la Production

3 **Configuration Post-Installation**
Enregistrement du Système

- sudo subscription-manager register
  - - -username=votre_compte_redhat
  - --password=votre_mot_de_passe
  - - -auto-attach

Mise à Jour Initiale

- sudo dnf update -y

Activation des Dépôts

- sudo subscription-manager repos --enable=rhel-9-for-x86_64-baseos-rpms .
- sudo subscription-manager repos --enable=rhel-9-for-x86_64-appstream-rpms

4 **Outils Essentiels**
Installation des Paquets de Base

- sudo dnf install -y vim git curl wget net-tools cockpit

Démarrer Cockpit (Interface Web)

- sudo systemctl enable - -now cockpit.socket
- Accès via: **https://** $< IP >$**:9090**

5 **Sécurité de Base**
Configuration du Pare-feu

- sudo firewall-cmd - -permanent --add-service=http
- sudo firewall-cmd - -permanent --add-service=https
- sudo firewall-cmd - -reload

Désactivation de SELinux (Optionnel)

- sudo setenforce 0
- sudo sed -i 's/SELINUX=enforcing/SELINUX=permissive/g' /etc/selinux/config (ecriture dans le fichier de demarrage)

6 **Configuration Réseau**
IP Statique

- sudo nmcli con mod "eth0" ipv4.addresses "192.168.1.100/24" ipv4.gateway "192.168.1.1" ipv4.dns "8.8.8.8" ipv4.method manual
- sudo nmcli con up "eth0"

7 **Vérification**

- **cat /etc/redhat-release**: Vérifier la version .
- **systemctl status firewalld cockpit**: Vérifier les services clés

Prochaines Étapes Recommandées

1. **Sauvegarde Initiale**:
   - sudo tar -czvf /backup/rhel9-initial-config.tar.gz /etc
2. **Configuration des Sauvegardes**:
   - sudo dnf install -y rsync cronie
3. **Monitoring**:
   - sudo dnf install -y telegraf

Note: Pour les environnements de production, envisagez d'utiliser **Ansible** pour automatiser ces configurations.

Prise en Main de l'Interface en ligne de commande (CLI) sous RHEL 9.4

La ligne de commande (CLI) de Red Hat Linux est un outil puissant qui permet d'interagir directement avec le système sans passer par une interface graphique.

3 Connexion au Système Accès Initial

- **ssh** *utilisateur@adresse_ip*
- **su -** :Pour basculer en root
- **cat /etc/redhat-release**: Vérification de la Version

4 Navigation de Base

Commande	Description
pwd	Affiche le répertoire courant
ls -la	Liste les fichiers avec détails
cd /chemin	Change de répertoire
clear	Nettoie le terminal

Table: Navigation de Base

Astuces

- **cd** : Retour au répertoire utilisateur
- **cd -**: Retour au répertoire précédent

3 Gestion des Fichiers

Commandes Essentielles

- **cp** *fichier1 fichier2*: Copie
- **mv** *ancien nouveau*: Renommage/ Déplacement
- **rm** *fichier*: Suppression fichier/ dossier
- **mkdir dossier**: Création dossier

Édition de Fichiers

- **vim** *nom_fichier.conf*: Éditer le fichier avec l'editeur Vim (editeur complexe)
- **nano** *nom_fichier.conf*: Éditer le fichier avec l'editeur nano (plus simple)

4 Gestion des Paquets (DNF) Opérations Clés

- **sudo dnf search** *terme*: Recherche le fichier en parametre
- **sudo dnf install** *paquet* : Installation du paquet en parametre
- **sudo dnf remove** *paquet*: Désinstallation du paquet en parametre
- **sudo dnf update**: Mise à jour système
- **sudo dnf list installed**: Liste paquets installés

5 Gestion des Processus

Monitoring

- **top** : Vue dynamique
- **htop** : Version améliorée (à installer)
- **ps aux**: Liste complète

Contrôle

- **kill -9 PID**: Force l'arrêt
- **pkill** *nom*: Tue par nom en parametre
- **systemctl restart service**: Redémarre un service

6 Gestion des Utilisateurs

- **sudo useradd** *nouvel_utilisateur*: Création de l'utilisateur
- **sudo passwd** *nouvel_utilisateur*: Définition mot de passe
- **sudo usermod -aG wheel** *utilisateur*: Donne les droits sudo

7 Réseau

Commandes Utiles

- **ip addr show**: Afficher parametres IP
- **ping google.com**: Test connectivité
- **ss -tuln**: Ports ouverts
- **nmcli device show** : Affiche configuration réseau

8 Journal Système

- **journalctl -xe**: Derniers logs
- **sudo tail -f /var/log/messages**: Surveillance en temps réel

9 Personnalisation du Shell

Alias Utiles

- Ajouter à **/.bashrc**:
  - **alias ll='ls -alF'**
  - **alias update='sudo dnf update'**

10 Aide et Documentation

- **man** *commande*: Manuel de la commande en parametre
- **commande - -help**: Aide rapide de la commande en parametre
- **dnf docs**: Documentation Red Hat

Gestion des Logiciels avec DNF sous RHEL 9.4

1 Configuration de Base

Vérifier les dépôts activés

- **sudo dnf repolist**

Activer des dépôts supplémentaires (ex: EPEL)

- **sudo dnf install epel-release**
- **sudo dnf config-manager - -set-enabled epel**

2 Commandes Essentielles

Commande	Description
sudo dnf search *terme*	Rechercher un paquet
sudo dnf info *paquet*	Liste les fichiers avec détails
sudo dnf install *paquet*	Installer un paquet
sudo dnf remove *paquet*	Désinstaller
sudo dnf update	Mettre à jour tous les paquets
sudo dnf upgrade	Mise à jour + nettoyage
sudo dnf autoremove	Nettoyer les dépendances inutiles

3 Gestion Avancée

Installation groupée

- **sudo dnf groupinstall "Développement d'outils"**

Liste des groupes disponibles

- **sudo dnf group list**

Historique des transactions

- **sudo dnf history**: Annuler une action

4 Options Utiles

Télécharger sans installer

- **sudo dnf download** *paquet*

Vérifier les mises à jour de sécurité

- **sudo dnf updateinfo list sec**

Exclure un paquet des mises à jour

- **sudo vim /etc/dnf/dnf.conf** Ajouter : exclude=paquet1 paquet2*

5 Gestion des Clés RPM

Liste des clés

- **rpm -qa gpg-pubkey***

Importer une clé

- **sudo rpm –import** */chemin/vers/la/clé*

6 **Dépannage**

Vérifier les dépendances

- **sudo dnf repoquery –requires** paquet

Réparer la base de données DNF

- **sudo rpm –rebuilddb**
- **sudo dnf clean all**

7 **Astuces Productivité**

Alias recommandés

- **Ajouter à ‘ /.bashrc‘**:

```
alias maj='sudo dnf update'
alias net='sudo dnf install'
alias supp='sudo dnf remove'
```

Installation silencieuse

- **sudo dnf install -y –quiet** *paquet*

8 **Exemple Complet**

Rechercher et installer un paquet

- **sudo dnf search nginx**
- **sudo dnf install nginx**

Vérifier la version installée

- **rpm -q nginx**

Mettre à jour uniquement ce paquet

- **sudo dnf update nginx**

Remarque: RHEL 9.4 utilise DNF 4.14+ avec des performances améliorées et un meilleur gestionnaire de dépendances.

ADMINISTRATION

Gestion des Utilisateurs et Groupes sous RHEL 9.4

1 Commandes de Base pour les Utilisateurs

Création d'utilisateur

- **sudo useradd -m -s /bin/bash** *nom_utilisateur*: -m crée le dossier home **sudo passwd** *nom_utilisateur*: Définir le mot de passe

Modification d'utilisateur

- **sudo usermod -aG wheel** *nom_utilisateur*: Ajouter aux administrateurs
- **sudo usermod -L** nom_utilisateur: Verrouiller le compte
- **sudo usermod -U** nom_utilisateur: Déverrouiller le compte

Suppression

- **sudo userdel -r** *nom_utilisateur*: -r supprime le dossier home

2 Gestion des Groupes

Création et gestion

- **sudo groupadd** *nom_groupe*
- **sudo gpasswd -a** *utilisateur groupe*: Ajouter un utilisateur au groupe
- **sudo gpasswd -d** *utilisateur groupe*: Retirer un utilisateur du groupe

Groupes système importants

Groupe	Description
wheel	Accès sudo
docker	Gestion des containers
libvirt	Virtualisation

3 Fichiers de Configuration Clés

Fichier	Contenu
/etc/passwd	Comptes utilisateurs
/etc/shadow	Mots de passe chiffrés
/etc/group	Définition des groupes
/etc/sudoers	Droits d'administration

4 Commandes Avancées

Changer propriétaire home directory

- **sudo chown -R utilisateur:groupe /home/utilisateur**

Vérifier les groupes d'un utilisateur

- **id** *nom_utilisateur*
- **groups** nom_utilisateur

Modifier UID/GID

- **sudo usermod -u 1500** *nom_utilisateur*: Changer UID
- **sudo groupmod -g 1600** *nom_groupe*: Changer GID

5 Sécurité et Politiques

Durée de vie des mots de passe

- **sudo chage -l** *nom_utilisateur*: Voir les paramètres
- **sudo chage -M 90** *nom_utilisateur*: Changer expiration (jours)

Configuration PAM

- **Editer /etc/login.defs pour**:
  - Durée min/max des mots de passe
  - Taille min des mots de passe
  - Politique de vieillissement

6 Bonnes Pratiques

Journalisation

- **sudo grep *nom_utilisateur* /var/log/secure**: Voir les connexions

Template de dossier home

- **sudo cp -r /etc/skel /etc/skel_custom**
- **sudo useradd -m -k /etc/skel_custom** *nouvel_utilisateur*

Accès SSH contrôlé

- **sudo vim /etc/ssh/sshd_config**
- **AllowUsers utilisateur1 utilisateur2**

7 Exemple Complet

Créer un utilisateur avec accès admin

- **sudo useradd -m -G wheel -s /bin/bash admin_user**
- **sudo passwd admin_user**

Créer un groupe et ajouter l'utilisateur

- **sudo groupadd dev_team**
- **sudo usermod -aG dev_team admin_user**

Vérifier

- id admin_user

Astuce: Pour les environnements Enterprise, utilisez Red Hat Identity Management (IdM) pour une gestion centralisée.

Gestion des Permissions et Droits d'Accès sous RHEL 9.4

1 Permissions Basiques (chmod/chown)

Modification des Permissions

- **chmod u+rwx,g+rx,o-rwx fichier**: Syntaxe symbolique
- **chmod 750 fichier**:Syntaxe octale (7:user, 5:group, 0:others)

Valeur	Permission
7	rwx (Lecture+Écriture+Exécution)
6	rw-
5	r-x
4	r–

Changement de Propriétaire

- **chown utilisateur**:groupe fichier
- **chown -R utilisateur:groupe dossier/**: Récurssif

2 Permissions Spéciales

Sticky Bit (pour dossiers partagés)

- **chmod +t** */dossier_partagé*: Empêche la suppression par d'autres utilisateurs
- **ls -ld** */dossier_partagé*: Vérifier (le 't' apparait)

SetUID/SetGID

- **chmod u+s /bin/commande**: Exécution avec les droits du propriétaire
- **chmod g+s /dossier/**: Nouveaux fichiers héritent du groupe parent

3 Liste de Contrôle d'Accès (ACL)

Activation des ACL

- **sudo tune2fs -o acl /dev/sd**:Pour les systèmes de fichiers existants

Gestion des ACL

- **setfacl -m u:utilisateur:rwx fichier** : Ajouter permission
- **setfacl -m g:groupe:r-x fichier**: Pour un groupe
- **setfacl -x u:utilisateur fichier**: Supprimer entrée
- **getfacl** *fichier*: Voir les ACL

4 Sécurité avec SELinux

Commandes de Base

- **sestatus** : Vérifier l'état
- **ls -Z fichier**:Voir le contexte
- **chcon -t httpd_sys_content_t /dossier**: Changer type
- **restorecon -Rv /dossier** : Restaurer contexte par défaut

Types Courants

Contexte	Usage
httpd_sys_content_t	Fichiers web
samba_share_t	Partages Samba
user_home_t	Dossiers utilisateurs

5 Gestion des Capacités (Capabilities)

Exemple : donner accès réseau sans root

- **sudo setcap 'cap_net_bind_service=+ep' /bin/** *mon_programme*
- **getcap /bin/** *mon_programme*: Vérifier

6 Outils de Vérification

Analyse des Permissions

- **find /dossier -type f -perm /4000**:Trouver fichiers SetUID
- **find /dossier -type f -perm /2000**: Trouver fichiers SetGID

Vérification d'intégrité

- **rpm -Va –nomtime –nouser –nogroup**: Vérifier permissions système

7 Bonnes Pratiques

Politique de Base

- **umask 027**: Permissions par défaut : rw-r—–

Hiérarchie Standard

- **/home/utilisateur/ : 700 (drwx——)**
- **/var/www/ : 755 (drwxr-xr-x)**
- **/etc/ : 644 (fichiers), 755 (dossiers)**

Journalisation

- **sudo auditctl -w /dossier_critique -p wa -k acces_dossier**

Exemple Complet

Créer un dossier partagé avec ACL

- **sudo mkdir /opt/***equipe*
- **sudo chown root:equipe /opt/***equipe*
- **sudo chmod 2770 /opt/***equipe*: SetGID + permissions
- **sudo setfacl -m g:dev:rwx /opt/***equipe*: Accès pour groupe supplémentaire

Vérification

- **ls -ld /opt/equipe**
- **getfacl /opt/***equipe*

Planification des Tâches avec ‘Cron‘ et ‘at‘ sous RHEL 9.4

1 Gestion des Tâches Récurrentes avec Cron

Configuration de Base

- **sudo systemctl enable –now crond** : Activer le service
- **crontab -e**: Éditer les crontabs utilisateur.
- **sudo crontab -e**: Éditer les crontabs root
- **crontab -l**: Liste les tâches planifiées.

Format des Entrées Cron

Commencer par **les symboles** ci dessous suivis de la commande en question

Symbole	Description
*	Jour de la semaine (0-7, 0=7=dimanche)
*	Mois (1-12)
*	Jour du mois (1-31)
*	Heure (0-23)
*	Heure (0-23)

2 Exemples Pratiques

- **0 2 * * * /opt/scripts/backup.sh**: Tous les jours à 2h
- ***/15 * * * * ping -c 1 google.com** : Toutes les 15 minutes
- **0 0 1 * * tar -czf /backups/mensuel.tar.gz /data —**: Le 1er de chaque mois

Fichiers Système

- **etc/crontab** : Crontab système
- **/etc/cron.d/** : Configs supplémentaires
- **/etc/cron.hourly/** : Scripts horaires
- **/var/spool/cron/** : Crontabs utilisateurs

3 Planification Ponctuelle avec ‘at‘

Utilisation de Base

- **echo** ”*commande*” **— at 14:30 2025-12-31**: Planifier à une date/heure
- **atq**: Lister les tâches en attente
- **atrm 3**: Supprimer la tâche #3

Options Courantes

- **at now + 1 hour** : Dans 1 heure
- **at 23:59 tomorrow**: Demain à 23:59
- **at noon + 3 days** : Dans 3 jours à midi

4 Bonnes Pratiques

Journalisation

- **sudo grep CRON /var/log/cron**: Voir l'exécution des tâches
- **sudo journalctl -u crond -f** : Surveillance en temps réel

Sécurité

- **/etc/cron.allow**: Liste blanche utilisateurs
- **/etc/cron.deny**: Liste noire (priorité si les deux existent)

5 Outils Avancés

Anacron (pour systèmes non permanents)

- **sudo dnf install anacron**
- **vim /etc/anacrontab**: Format différent (jours entre exécutions)

Exemple de Configuration Anacron

- **1 5 daily.backup /opt/scripts/backup.sh**
- **7 10 weekly.clean /usr/bin/apt-get clean**

6 Dépannage

Commandes Utiles

- **crontab -l** : Lister ses tâches planifiées
- **systemctl status crond**: Vérifier l'état du service

Problèmes Courants

- **Variables d'environnement**: Préciser le PATH dans les scripts
- **Permissions**: Vérifier les droits d'exécution
- **Journalisation** : Rediriger la sortie dans les crontabs :
- *** * * * * /script.sh > > /var/log/script.log 2>&1**

Exemple Complet

Backup quotidien à minuit

- **(crontab -l 2¿/dev/null; echo "0 0 * * * /usr/bin/tar -czf /backups/daily_$(date +%Y%m%d).tar.gz /data") — crontab -**

Fichiers Système

- **etc/crontab** : Crontab système
- **/etc/cron.d/** : Configs supplémentaires
- **/etc/cron.hourly/** : Scripts horaires
- **/var/spool/cron/** : Crontabs utilisateurs

Configuration des Services et Processus avec systemd sous RHEL 9.4

1. **Commandes de Base de systemd**

   Gestion des Services

   - **sudo systemctl start** *nom_service*: Démarrer
   - **sudo systemctl stop** *nom_service* : Arrêter
   - **sudo systemctl restart** *nom_service* : Redémarrer
   - **sudo systemctl reload** *nom_service* : Recharger sans interruption
   - **sudo systemctl status** *nom_service*: Vérifier l'état

   Activation au Démarrage

   - **sudo systemctl enable** *nom_service*: Activer
   - **sudo systemctl disable** *nom_service*: Désactiver
   - **sudo systemctl is-enabled** *nom_service*: Vérifier

2. **Fichiers de Configuration**

   Structure des Unités

   - **/etc/systemd/system/** : Unités personnalisées(priorité)
   - **/usr/lib/systemd/system/** : Unités fournies par les paquets
   - **/run/systemd/system/** : Unités temporaires

   Exemple de Fichier de Service

   Unit] Description=Mon Service Personnalisé
   After=network.target
   [Service] Type=simple
   User=mon_utilisateur
   ExecStart=/usr/bin/python3 /opt/mon_script.py
   Restart=on-failure
   [Install]
   WantedBy=multi-user.target

3 Commandes Avancées

Journalisation

- **sudo journalctl -u** *nom_service*: Logs du service
- **sudo journalctl -f -u** *nom_service*: Suivi en temps réel
- **sudo journalctl –since ”2025-01-01” –until ”2025-01-02”**

Analyse des Dépendances

- **systemctl list-dependencies** *nom_service*
- **systemd-analyze blame**: Temps de démarrage des services

4 Types de Services Courants

simple Exécute immédiatement (défaut) **Scripts simples**

- **forking** Le service se fork en arrière-plan **Démon traditionnel**
- **oneshot** Exécution unique **Scripts d’initialisation**
- **notify** Signale quand prêt **Applications systemd-aware**

5 Sécurité et Contrôle

Confinement

```
[Service]
ProtectSystem=full
PrivateTmp=true

CapabilityBoundingSet=CAP_NET_BIND_SERVICE
```

Politique de Redémarrage

```
[Service] RestartSec=5s
Restart=on-failure
StartLimitInterval=1min
StartLimitBurst=3
```

6 Bonnes Pratiques

Toujours tester après modification

- **sudo systemctl daemon-reload**
- **sudo systemctl restart** *nom_service*

Utiliser des templates pour les instances multiples

- **systemctl start** *myapp@instance1*
- **systemctl start** *myapp@instance2*

Vérifier les ressources

- **systemd-cgtop**
- **systemctl show** *nom_service* **— grep Memory**

7 **Exemple Complet**

Création d'un Service Personnalisé

- **sudo vim /etc/systemd/system/***mon_service.service*
- **Contenu**:
  Unit]
  Description=Service de Backup Quotidien
  [Service]
  Type=oneshot
  ExecStart=/opt/scripts/backup.sh
  User=backup
  Group=backup
  [Install]
  WantedBy=timers.target

Création d'un Timer Associé

- **sudo vim /etc/systemd/system/***mon_service.timer*
- **Contenu**:
  [Unit] Description=Exécution quotidienne du backup
  [Timer]
  OnCalendar=daily
  Persistent=true
  [Install]
  WantedBy=timers.target

Activation:

- **sudo systemctl enable –now** *mon_service.timer*

8 **Important**: RHEL 9.4 utilise systemd v250+ avec des fonctionnalités améliorées comme 'systemd-oomd' pour la gestion de la mémoire.

9 Sur Red Hat Enterprise Linux (RHEL) 9.4, plusieurs systèmes de fichiers sont disponibles, dont **ext4** et **XFS**, qui sont les plus couramment utilisés. Voici une comparaison et des informations sur leur gestion sous RHEL 9.4:

1 **XFS (Système de fichiers par défaut dans RHEL 9.4) Caractéristiques**

- Conçu pour les systèmes haute performance et les gros volumes de données.
- **Évolutivité**: prend en charge des fichiers et des systèmes de fichiers de très grande taille (jusqu'à 8 exaoctets).
- Journalisation (journaling) pour une récupération rapide après un crash.
- Pas de défragmentation en ligne (nécessite un redémarrage ou un outil externe).
- Meilleures performances pour les charges de travail impliquant de gros fichiers (bases de données, virtualisation).

10 ext4 (Alternative classique)

Caractéristiques

- Successeur d'ext3, avec des améliorations de performance et de fiabilité.
- Prend en charge les fichiers jusqu'à 16 téraoctets et les systèmes de fichiers jusqu'à 1 exaoctet.
- Journalisation pour la récupération après crash.
- Défragmentation possible en ligne ('e4defrag').
- Meilleur pour les petits fichiers et les charges de travail généralistes.

Commandes utiles

- **sudo mkfs.ext4 /dev/sdX**: Créer un système de fichiers ext4
- **sudo mount /dev/sdX /mnt/***point_de_montage*
- **sudo fsck.ext4 /dev/sdX**: Vérifier et réparer (unmounted)
- **sudo resize2fs /dev/sdX**:

11 Choix entre XFS et ext4 sous RHEL 9.4

Utilisez XFS si

- Vous avez besoin de performances élevées avec des gros fichiers.
- Vous utilisez des applications comme des bases de données (PostgreSQL, MySQL) ou de la virtualisation.
- Vous avez besoin d'une meilleure scalabilité pour de très gros volumes.

Utilisez ext4 si:

- Vous avez besoin de défragmentation en ligne.
- Vous travaillez avec de nombreux petits fichiers.
- Vous préférez une compatibilité ascendante avec ext3/ext2.

GESTION DU STOCKAGE

Systèmes de fichiers (ext4, xfs, etc.)

Sous **Red Hat Linux**, plusieurs **systèmes de fichiers** sont pris en charge pour la gestion des données sur disque. Voici les principaux:

- **ext4**:
  - Évolution de ext3, offrant une meilleure gestion des fichiers volumineux et des performances accrues.
  - Très utilisé pour les partitions standards sous Linux.
- **XFS**:
  - Optimisé pour les systèmes nécessitant un **haut débit d'écriture et de lecture**, idéal pour les serveurs et le cloud.
  - Prend en charge la journalisation avancée pour une récupération rapide en cas de panne.
- **Btrfs**:
  - Propose des fonctionnalités comme **la compression, les instantanés (snapshots)** et une meilleure gestion des volumes.
  - Moins répandu sur Red Hat Linux, mais utile pour des scénarios nécessitant un suivi précis des modifications.
- **Vfat et NTFS**:
  - **Vfat**: est utilisé pour les disques compatibles avec Windows.
  - **NTFS**:, bien que non natif sous Linux, est pris en charge pour l'échange de fichiers avec des systèmes Windows.

Chaque système de fichiers a ses avantages en fonction des **besoins de performance, de gestion et de fiabilité**.

Création et gestion de partitions

La **création et gestion de partitions** sous Red Hat Linux permet d'organiser efficacement le stockage sur un disque dur. Voici les grandes lignes :

- **Création d'une partition**:
  - Utilisation de **fdisk** ou **parted** pour définir les nouvelles partitions.
  - **Exemple**: **fdisk /dev/sdX** pour ouvrir l'outil de partitionnement interactif.
- **Formatage de la partition**:
  - Une fois créée, elle doit être formatée avec un système de fichiers (**mkfs.ext4 /dev/sdX1**).
- **Montage d'une partition**
  - Utilisez **mount /dev/sdX1 /mnt/*point_de_montage*** pour l'attacher à un dossier accessible.
  - Ajoutez-la au fichier **/etc/fstab** pour un montage automatique au démarrage.
- **Redimensionnement et gestion**:
  - r**esize2fs** pour ajuster la taille d'une partition ext4.
  - **lvextend** et **lvreduce** pour les volumes logiques sous LVM.
- **Suppression d'une partition**:
  - Avec **fdisk** ou **parted**, supprimez une partition existante si nécessaire.

Ces outils permettent une gestion efficace du stockage sous Red Hat Linux.

Utilisation de LVM (Logical Volume Manager)

LVM (**Logical Volume Manager**) sous **Red Hat Linux** permet une gestion flexible du stockage en combinant plusieurs disques ou partitions en volumes logiques. Voici les grandes lignes:

- **Avantages**:
  - Permet d'ajouter ou de redimensionner des partitions sans interrompre le système.
  - Offre une gestion avancée des snapshots et du stockage dynamique.
- **Création d'un groupe de volumes**:
  - **pvcreate /dev/sdX** → Initialise un disque pour LVM.
  - **vgcreate monVG /dev/sdX /dev/sdY** → Crée un groupe de volumes.
- **Création d'un volume logique**
  - **lvcreate -L 10G -n monLV monVG** → Crée un volume logique de 10 Go.
  - Formatage avec **mkfs.ext4 /dev/monVG/monLV**.
- **Montage et gestion**:
  - **mount /dev/monVG/monLV /mnt** → Monte le volume.
  - **lvextend -L +5G /dev/monVG/monLV** → Augmente la taille du volume.
- **Suppression et optimisation**:
  - **lvremove /dev/monVG/monLV** → Supprime un volume logique.
  - **vgremove monVG** → Supprime un groupe de volumes.

LVM est idéal pour les environnements nécessitant une gestion évolutive du stockage.

Montage automatique et gestion du swap

Le **montage automatique** et la **gestion du swap** sous Red Hat Linux sont essentiels pour garantir une gestion efficace du stockage et de la mémoire.

- **Montage automatique des partitions**
  - Les partitions peuvent être montées automatiquement au démarrage en les ajoutant au fichier /etc/fstab.
  - Exemple d'entrée dans fstab: /dev/sdX1 /mnt/data ext4 defaults 0 2
  - Utilisez **mount -a** pour appliquer immédiatement les modifications.

- **Gestion du swap**:
  - Le **swap** permet d'étendre la mémoire virtuelle lorsque la RAM est saturée.
  - Création d'un fichier swap:
    1. **dd if=/dev/zero of=/swapfile bs=1G count=2** -¿ Crée un fichier de 2 Go
    2. chmod 600 /swapfile
    3. mkswap /swapfile
    4. swapon /swapfile
- Ajout du swap au démarrage (**/etc/fstab**):
  - **/swapfile none swap sw 0 0**

Ces configurations améliorent la gestion des ressources système et la stabilité.

RESEAU ET SECURITE

Configuration réseau (IP statique, DHCP, DNS)

configuration d'une adresse IP statique

La **configuration d'une adresse IP statique** sous **Red Hat Linux** permet de définir une adresse IP fixe au lieu d'une attribution dynamique via DHCP. Voici les étapes essentielles :

1. **Modifier le fichier de configuration réseau**
   - Ouvrir le fichier correspondant à l'interface réseau (**eth0, ens160**, etc.):
     - /etc/sysconfig/network-scripts/ifcfg-eth0
   - Exemple de configuration IP statique:

```
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
IPADDR=192.168.1.100
NETMASK=255.255.255.0
GATEWAY=192.168.1.1
DNS1=8.8.8.8
DNS2=1.1.1.1
```

2. **Redémarrer le service réseau**
   - Appliquer les modifications avec: **systemctl restart NetworkManager**

3. **Vérifier la configuration**:
   - Vérifier l'adresse IP attribuée: **ip addr show *name_interface***
   - Tester la connectivité: **ping *ip_address_distant***

Ces configurations améliorent la gestion des ressources système et la stabilité.

configuration d'un serveur DHCP

La configuration d'un serveur DHCP sous Red Hat Linux permet d'attribuer automatiquement des adresses IP aux clients du réseau. Voici les étapes essentielles:

1. **Installation du serveur DHCP**
   - Installez le package DHCP avec: **dnf install dhcp**
   - Le fichier de configuration principal est /etc/dhcp/dhcpd.conf.
2. **Configuration du fichier dhcpd.conf**
   - Exemple de configuration pour un sous-réseau:
     - subnet 192.168.1.0 netmask 255.255.255.0 range 192.168.1.100 192.168.1.200; option routers 192.168.1.1; option domain-name-servers 8.8.8.8, 1.1.1.1; default-lease-time 600; max-lease-time 7200;
   - Ce fichier définit la plage d'adresses IP attribuées aux clients.
3. **Démarrage et activation du service**:
   - Activez et démarrez le service DHCP: **systemctl enable dhcpd** and then **systemctl start dhcpd**
   - Vérifiez son statut avec: **systemctl status dhcpd**
4. **Vérification et gestion des baux DHCP**:
   - Les adresses attribuées sont stockées dans **/var/lib/dhcpd/dhcpd.leases**.
   - Consultez les baux actifs avec: **cat /var/lib/dhcpd/dhcpd.leases**
5. Redémarrage après modification
   - Après toute modification du fichier de configuration, redémarrez le service: **systemctl restart dhcpd**

Consulter la documentation officielle de Red Hat pour plus de détails ici!

configuration d'un serveur DNS

La **configuration d'un serveur DNS** sous **Red Hat Linux** permet de gérer la résolution des noms de domaine et d'associer des adresses IP aux noms d'hôtes. Voici les étapes essentielles:

1. **Installation du serveur DNS (Bind)**
   - Installez le package Bind, qui est le serveur DNS le plus couramment utilisé: dnf install bind bind-utils
   - Le fichier de configuration principal est **/etc/named.conf**.
2. Configuration du fichier **named.conf**
   - Définissez les options de votre serveur DNS:
     - options directory "/var/named"; listen-on port 53 192.168.1.1; ; allow-query any; ; recursion yes; ;
   - Ajoutez une zone pour votre domaine:
     - zone "example.com" IN type master; file "example.com.zone"; ;
3. **Création du fichier de zone DNS**:
   - Créez le fichier /var/named/example.com.zone et ajoutez les enregistrements:
     - $TTL 86400 @ IN SOA ns1.example.com. admin.example.com. (2024051301 ; Numéro de série 3600 ; Rafraîchissement 1800 ; Réessai 604800 ; Expiration 86400) ; TTL minimum @ IN NS ns1.example.com. ns1 IN A 192.168.1.1 www IN A 192.168.1.100

configuration d'un serveur DNS - Suite

La **configuration d'un serveur DNS** sous **Red Hat Linux** permet de gérer la résolution des noms de domaine et d'associer des adresses IP aux noms d'hôtes. Voici les étapes essentielles:

4. **Démarrage et activation du service DNS**
   - Activez et démarrez le service Bind: **systemctl enable named** and then **systemctl start named**
5. Vérifiez son statut: **systemctl status named**
6. **Vérification et tests**
   - Vérifiez la syntaxe du fichier de zone: **named-checkconf** and then **named-checkzone example.com /var/named/example.com.zone**
   - Testez la résolution DNS avec **nslookup** ou **dig**: **dig example.com**

Consultez la documentation officielle de Red Hat pour plus de détails ici!

RESEAU ET SECURITE

Firewall avec Firewalld

Firewall avec firewalld

Le **pare-feu** firewalld sous **Red Hat Linux** permet de gérer les règles de filtrage réseau de manière dynamique et sécurisée. Voici les grandes lignes:

1. **Installation et vérification du service**
   - Firewalld est généralement préinstallé sur Red Hat Linux. Vérifiez son statut avec: **systemctl status firewalld**
   - Activez-le si nécessaire: **systemctl enable –now firewalld**
2. **Zones de sécurité**
   - Firewalld fonctionne avec des zones, qui définissent les règles de sécurité selon l'usage:
     - **public**: Configuré pour une connexion non fiable.
     - **internal**: Pour un réseau interne sécurisé.
     - **trusted**: Autorise tout le trafic.
     - Liste des zones disponibles: **firewall-cmd –get-zones**
     - Définir une zone par défaut: **firewall-cmd –set-default-zone=public**
1. **Gestion des règles**
   - **Ajout d'une règle** pour autoriser un service: **firewall-cmd –zone=public –add-service=http –permanent**
   - Ouverture d'un port spécifique: **firewall-cmd –zone=public –add-port=8080/tcp –permanent**
   - Suppression d'une règle: **firewall-cmd –zone=public –remove-service=http –permanent**
2. **Rechargement et vérification**
   - Appliquer les modifications: **firewall-cmd –reload**
   - Voir les règles actives: **firewall-cmd –list-all**
3. **Journalisation et suivi**
   - Vérifier les logs du pare-feu pour détecter les blocages: **journalctl -u firewalld –since "1 hour ago"**

Firewalld simplifie la gestion du pare-feu tout en offrant une flexibilité accrue pour sécuriser les connexions réseau.

RESEAU ET SECURITE

SELinux: Principe et Configuration

SELinux : principes et configuration

SELinux (Security-Enhanced Linux) est un module de sécurité sous Red Hat Linux qui renforce la protection du système en contrôlant l'accès aux fichiers et processus.

1. **Principes de SELinux**
   - **Contrôle d'accès obligatoire (MAC)**: Contrairement aux permissions classiques, SELinux impose des règles strictes définies par des **politiques de sécurité**.
   - **Confinement des processus**: Chaque processus fonctionne avec des**contextes de sécurité**, limitant leurs interactions avec le système.
   - **Modes d'exécution**:
     - **Enforcing**:: Les règles de SELinux sont appliquées strictement.
     - **Permissive** : Les violations sont enregistrées mais non bloquées.
     - **Disabled** : SELinux est désactivé
2. **Configuration de SELinux**
   - Vérifier le statut: **getenforce** and then **sestatus**
   - **Changer le mode**:
     - **setenforce 0** : Mode permissif
     - **setenforce 1** : Mode enforcing
     - Modifier **/etc/selinux/config** pour un changement permanent: **SELINUX=enforcing**
   - **Gérer les contextes de sécurité**:
     - Voir le contexte d'un fichier: **ls -Z /var/www/html**
     - Modifier le contexte: **chcon -t httpd_sys_content_t /var/www/html/index.html**
   - **Dépannage et journalisation**
     - Consulter les logs de SELinux: **cat /var/log/audit/audit.log — grep AVC**
     - Utiliser **semanage** pour gérer les politiques.

SELinux renforce la sécurité des systèmes Red Hat en limitant les accès non autorisés.

RESEAU ET SECURITE

Gestion des services SSH, FTP, NFS, etc.

RESEAU ET SECURITE

Gestion du service SSH

Gestion du service SSH

La gestion du **service SSH** sous **Red Hat Linux** est essentielle pour l'accès sécurisé à distance à un serveur. Voici les points clés:

1. **Installation et démarrage du service SSH**
   - SSH est généralement installé par défaut, mais si nécessaire, vous pouvez l'installer avec: **dnf install openssh-server**
   - Activez et démarrez le service SSH: **systemctl enable –now sshd**
2. **Configuration du serveur SSH**
   - Le fichier principal de configuration est **/etc/ssh/sshd_config**.
   - Quelques paramètres importants:
     - **Changer le port par défaut (optionnel)**: Port 2222
     - **Interdire l'accès root (recommandé pour la sécurité)**: PermitRootLogin no
     - **Restreindre les utilisateurs autorisés**: AllowUsers user1 user2
3. **Redémarrage et application des modifications**
   - Après toute modification du fichier de configuration, redémarrez SSH: **systemctl restart sshd**
4. **Test et connexion SSH**
   - Depuis un autre ordinateur, testez la connexion: **ssh user@*ip_address***
   - Si un port différent est utilisé, spécifiez-le: **ssh -p *port_number* user@*ip_address***
5. **Sécurisation avec firewall et SELinux**
   - Autorisez le port SSH dans firewalld: **firewall-cmd –add-service=ssh –permanent** and then **firewall-cmd –reload**
   - Vérifiez et ajustez SELinux si nécessaire: **semanage port -a -t ssh_port_t -p tcp 2222**

Une bonne gestion de SSH garantit un accès sécurisé et contrôlé à votre serveur Red Hat.

RESEAU ET SECURITE

Gestion du service SFTP

Gestion du service SFTP

La **gestion du service SFTP** sous **Red Hat Linux** repose sur **OpenSSH**, qui permet un transfert sécurisé des fichiers en utilisant le protocole SSH.

1. **Différence entre SFTP et FTP**
   - Contrairement à **FTP**, **SFTP** (*SSH File Transfer Protocol*) chiffre les connexions et les données échangées, offrant une sécurité renforcée.
   - Il ne nécessite pas de serveur FTP distinct, car il fonctionne via SSH.
2. **Installation et configuration de SFTP**
   - SSH et SFTP sont généralement préinstallés, mais vous pouvez vérifier l'installation avec: **dnf install openssh-server**
   - Le service SSH doit être actif: **systemctl enable –now sshd**
3. **Restriction des accès SFTP**
   - Modifiez **/etc/ssh/sshd_config** pour restreindre l'accès uniquement au transfert de fichiers:
     - **Match User sftpuser**
     - **ForceCommand internal-sftp**
     - **PasswordAuthentication yes**
     - **ChrootDirectory /home/sftpuser**
     - **PermitTunnel no**
     - **AllowAgentForwarding no**
4. **Création et gestion des utilisateurs**
   - Créez un utilisateur dédié à SFTP: **useradd -m -s /sbin/nologin sftpuser**
   - Définissez un mot de passe: **passwd sftpuser**
   - Configurez les permissions du dossier:
     - **chown root:root /home/sftpuser**
     - **chmod 755 /home/sftpuser**
     - **mkdir /home/sftpuser/upload**
     - **chown sftpuser:sftpuser /home/sftpuser/upload**
5. **Redémarrage et tests**
   - Redémarrez le service SSH après modification: **systemctl restart sshd**
   - Testez la connexion SFTP depuis un autre ordinateur: **sftp *sftpuser@ip_address***

SFTP est une solution sécurisée pour le transfert de fichiers, particulièrement adaptée aux environnements professionnels.

RESEAU ET SECURITE

Gestion du service NFS

Gestion du service NFS

La **gestion du service NFS** (Network File System) sous **Red Hat Linux** permet de partager des fichiers entre plusieurs machines sur un réseau. Voici les grandes lignes:

1. **Installation du serveur NFS**
   - Installez le package nfs-utils: **yum install nfs-utils**
   - Activez et démarrez le service: **systemctl enable –now nfs-server**
2. **Configuration du serveur NFS**
   - Définissez les répertoires à partager dans /etc/exports: **/mnt/partage 192.168.1.0/24(rw,sync,no_root_squash)**
   - Appliquez la configuration: **exportfs -a**
3. **Gestion des accès et sécurité**
   - Vérifiez les partages actifs: **exportfs -v**
   - Autorisez NFS dans le pare-feu:
     - **firewall-cmd –add-service=nfs –permanent**
     - **firewall-cmd –reload**
4. **Configuration du client NFS**
   - Installez **nfs-utils** sur le client: **dnf install nfs-utils**
   - Montez le partage distant:
   - Configurez les permissions du dossier: **mount -t nfs *ip_address*:/mnt/partage /mnt/client**
5. **Montage automatique**
   - Ajoutez l'entrée dans **/etc/fsta**b pour un montage au démarrage: ***ip_address*:/mnt/partage /mnt/client nfs defaults 0 0**

NFS est idéal pour le partage de fichiers sur un réseau local, notamment dans les environnements serveurs.

SUPERVISION, DEPANNAGE ET AUTOMATISATION

Outils de surveillance système (top, htop, journalctl)

Les **outils de surveillance système** sous **Red Hat Linux** permettent d'analyser l'activité du système, surveiller les performances et détecter les anomalies. Voici trois outils essentiels:

1. **top**: Surveillance en temps réel des processus
   - Affiche les processus en cours d'exécution et leur consommation de ressources.
   - Informations clés:
     - **Charge CPU et mémoire**.
     - **Processus les plus gourmands**.
     - **Temps d'exécution et priorité**.
   - Commande de base: **top**
   - Pour quitter: **q**

2. **htop**: Alternative avancée à **top**
   - Offre une interface plus lisible avec des couleurs et une meilleure gestion des processus.
   - Fonctionnalités:
     - **Tri interactif des processus**
     - **Graphiques de charge CPU/mémoire**
     - **Possibilité de tuer un processus facilement**
   - Installation: **dnf install htop**
   - Exécution: **htop**
   - Navigation avec les flèches et raccourcis (**F9** pour tuer un processus).

3. **journalctl** : Consultation des logs système
   - Permet de lire les **journaux du système** générés par systemd.
   - Commandes utiles:
     - Voir les logs récents: **journalctl -n 50**
     - Filtrer par service: **journalctl -u sshd**
     - Afficher les logs depuis un certain moment: **journalctl –since "2 hours ago"**
     - Effacer les logs: **journalctl –vacuum-size=500M**

Ces outils sont indispensables pour surveiller la stabilité et les performances d'un système Red Hat Linux.

Dépannage courant

Le **dépannage courant** sous **Red Hat Linux** permet d'identifier et de résoudre les problèmes fréquents liés au système. Voici les principales étapes:

1. **Vérification des logs système**
   - Utilisez journalctl pour consulter les logs:
     - **journalctl -n 50**: Voir les 50 derniers événements
     - **journalctl -u sshd**: Filtrer par service SSH
     - **journalctl –since "1 hour ago"**: Logs de la dernière heure
2. **Dépannage des services et processus**
   - **Vérifiez l'état des services avec systemctl**: **systemctl status *nom_du_service***
   - Redémarrez un service en cas de problème: **systemctl restart *nom_du_service***
3. **Résolution des problèmes réseau**
   - Vérifiez l'adresse IP et les interfaces réseau:**ip a**
   - Testez la connectivité avec ping: **ping *ip_address***
   - Vérifiez les ports ouverts et les règles de pare-feu: **firewall-cmd –list-all**
4. **Dépannage des performances**
   - Surveillez l'utilisation CPU et mémoire avec **top** ou **htop**: **top** and **htop**
   - Vérifiez l'espace disque disponible: **df -h**
5. **Problèmes de fichiers et permissions**
   - Vérifiez les permissions d'un fichier: **ls -l fichier**
   - Modifiez les permissions si nécessaire: **chmod 644 *fichier*** and **chown user:user *fichier***
6. **Résolution des erreurs SELinux**
   - Vérifiez les logs de SELinux en cas de blocage: **cat /var/log/audit/audit.log — grep AVC**
   - Ajustez les contextes SELinux: **chcon -t httpd_sys_content_t /var/www/html**

Ces outils et commandes permettent de diagnostiquer et résoudre efficacement les problèmes sous Red Hat Linux.

Introduction à la rédaction de scripts shell

La **rédaction de scripts Shell** sous **Red Hat Linux** permet d'automatiser des tâches et d'exécuter une série de commandes de manière efficace. Voici les grandes lignes:

1. **Qu'est-ce qu'un script Shell ?**
   Un script Shell est un fichier contenant une suite de commandes Linux qui peuvent être exécutées automatiquement. Les scripts utilisent le **Bash** (**/bin/bash**) comme interpréteur par défaut.

2. **Création d'un script de base**
   - Créez un fichier avec l'extension **.sh**: **nano *monscript*.sh**
   - Ajoutez la première ligne indiquant l'interpréteur: **#!/bin/bash**
     - Insérez quelques commandes: **echo "Bonjour, bienvenue sur Red Hat Linux !"**
     - **ls -l**
   - Sauvegardez et rendez le script exécutable: **chmod +x *monscript*.sh**

3. **Exécution du script**
   - Pour l'exécuter, utilisez: **./*monscript*.sh**

4. **Variables et conditions**
   - Déclarez des variables:
     - **nom="Elie"**
       echo "Bonjour, $nom !"
   - Ajoutez des conditions:
     - **if [-d /home]; then**
       echo "Le répertoire existe."
       else
       echo "Le répertoire n'existe pas."
       fi

5. **Boucles et automatisation**
   - Exemple de boucle **for**:
     - **for fichier in *.txt; do**
       echo "Traitement du fichier : *$fichier*"
       done

Les scripts Shell sont puissants pour automatiser des tâches administratives, gérer les fichiers, surveiller le système et bien plus encore.

Sauvegarde et restauration de données

La **sauvegarde et la restauration de données** sous **Red Hat Linux** sont essentielles pour prévenir la perte de fichiers en cas de panne ou d'erreur. Voici les points clés:

1. **Sauvegarde des données**
   - Sauvegarde avec**tar**: **tar -cvf sauvegarde.tar /home/*user***
     - **-c** : Création d'une archive.
     - **-v** : Affichage du processus.
     - **-f** : Nom du fichier de sauvegarde.
   - Sauvegarde avec **rsync** (synchronisation efficace): **rsync -av /home/*user*/ /backup/*user*/**
     - **-a** : Mode archive (préserve les permissions).
     - **-v** : Mode verbeux.
   - Sauvegarde planifiée avec **cron**:
     - Ajoutez une tâche automatique dans **crontab**: **0 3 * * * rsync -av /home/user/ /backup/user/**
     - Exécution tous les jours à 3h du matin.

2. **Restauration des données**
   - Extraction d'une archive **tar**: **tar -xvf sauvegarde.tar -C /home/*user*/**
     - **-x** : Extraction des fichiers.
   - Restauration avec **rsync**: **rsync -av /backup/user/ /home/*user*/**

3. **Sauvegarde avancée avec Timeshift (snapshots)**
   - Installation: **dnf install timeshift**
   - Création d'un instantané: **timeshift –create –comments "Backup du système" –tags D**

4. **Automatisation et bonnes pratiques**
   - Stockez les sauvegardes sur un disque externe ou un serveur distant (**scp** ou **nfs**).
   - Testez régulièrement la restauration pour éviter les mauvaises surprises.

Une stratégie de sauvegarde bien configurée garantit une récupération rapide en cas de problème.

La Fin

Des questions? Commentaires?

BIBLIOGRAPHIE

Voici une sélection de références bibliographiques et ressources officielles pour l'administration sous **Red Hat Enterprise Linux 9.4** :

1. **Documentation Officielle Red Hat**
   - Red Hat Enterprise Linux 9 Documentation Red Hat, Inc. Disponible en ligne ici
   - Red Hat System Administrator's Guide Red Hat, Inc. Disponible ici. Couvre les tâches d'administration courantes (utilisateurs, stockage, réseaux, etc.)
   - Red Hat Security Hardening Guide. Disponible ici. Best practices pour sécuriser RHEL 9.4 (SELinux, firewalld, chiffrement).
2. **Livres de Référence**
   - Red Hat Enterprise Linux 9 Administration. **Miguel Pérez Colino**, *Pablo Iranzo Gómez, Packt Publishing, 2023, ISBN: 978-1803239827* . Couvre l'administration avancée, les services réseau, et les conteneurs.
   - Linux Bible, 10th Edition. **Christopher Negus** , *Wiley 2023 ISBN: 978-1119578888* . Inclut des chapitres dédiés à RHEL 9 et ses outils.
   - Red Hat RHCSA 9 Cert Guide. **Sander van Vugt**, *Pearson, 2023, ISBN: 978-0137341627*. Orientation pratique pour la certification RHCSA, valable pour RHEL 9.4.