FORMATION

LINUX - ADMINISTRATION

19-20-21-22-23/05/2025

m2iformation.fr

Montée en compétences

https://www.m2iformation.fr/parcours/informatique/systemes/linux/35/

AlmaLinux

Linux Administration

Tour de table

- **Parcours et attentes**

Linux Administration

Installation

Architecture système

Utilisateurs

Noyau

Modules

Amorçage du système

Maintenance du système

Stockage

Applications

Optimisations

Installation

- Tour d'horizon des OS Serveurs
- Préparation à l'installation et bonnes pratiques
- Processus d'installation
- Résoudre les problèmes rencontrés lors de l'installation
- Tâches post-installation pour un OS stable
- Configuration des fonctionnalités de base

Installation

Versions CentOS

Centos 8	Centos7	Centos 6
Kernel Version 4.18	3.10.0-123	2.6.32-71
Systemd	Systemd	upstart
Dnf	Yum	Yum
Sortie 24 septembre 2019	7 juillet 2014	10 juillet 2011
Mises à jour 1er mai 2024	4e trimestre 2020	10 mai 2017
Fin de support 31 mai 2029	30 juin 2024	30 novembre 2020

Versions CentOS

CentOS a sorti également une version dite Stream (rachetée par RedHat).

Il s'agit d'une sorte de Rolling Release.

http://isoredirect.centos.org/centos/8-stream/isos/x86_64/CentOS-Stream-8-x86_64-20191219-dvd1.iso

Alma Linux

Alma Linux est une distribution libre pour Linux basée sur Red Hat Enterprise Linux et destinée à prendre la place de CentOS.

Elle succède au très populaire CentOS.

https://repo.almalinux.org/almalinux/8/live/x86_64/AlmaLinux-8-latest-x86_64-Live-GNOME.iso

Debian

Debian

Il existe beaucoup de raisons pour lesquelles Debian est choisie comme système d'exploitation par les utilisateurs, les développeurs et même dans un environnement d'entreprise.

La plupart des utilisateurs apprécient sa stabilité et la douceur des processus de mise à niveau aussi bien des paquets que de la distribution dans son entier.

Debian est aussi très largement utilisée par les développeurs de logiciels et de matériels parce qu'elle fonctionne sur de nombreux périphériques et architectures, et elle fournit un système de suivi de bogues public ainsi que d'autres outils pour les développeurs.

Si vous projetez d'utiliser Debian dans un environnement professionnel, il existe d'autres avantages comme les versions avec suivi à long terme et les images pour l'informatique dématérialisée.

https://cdimage.debian.org/debian-cd/current-live/amd64/iso-hybrid/debian-live-12.10.0-amd64-gnome.iso

Autres

Rocky Linux...
RedHat...
Kali Linux
Parrot Linux
Arch Linux...

Distrowatch

Installation

Méthode d'installation (CD, réseau, clé USB…)

Préparation a l'installation de Linux

Espace disque… / 32 ou 64 bits ? /Ram ?

Processus d'Installation (serveur ou poste)

Quel type de partition ?

/boot

/home

/

Le rôle du compte administrateur (root)

su - pour devenir root

ctrl c pour annuler une action

Installation

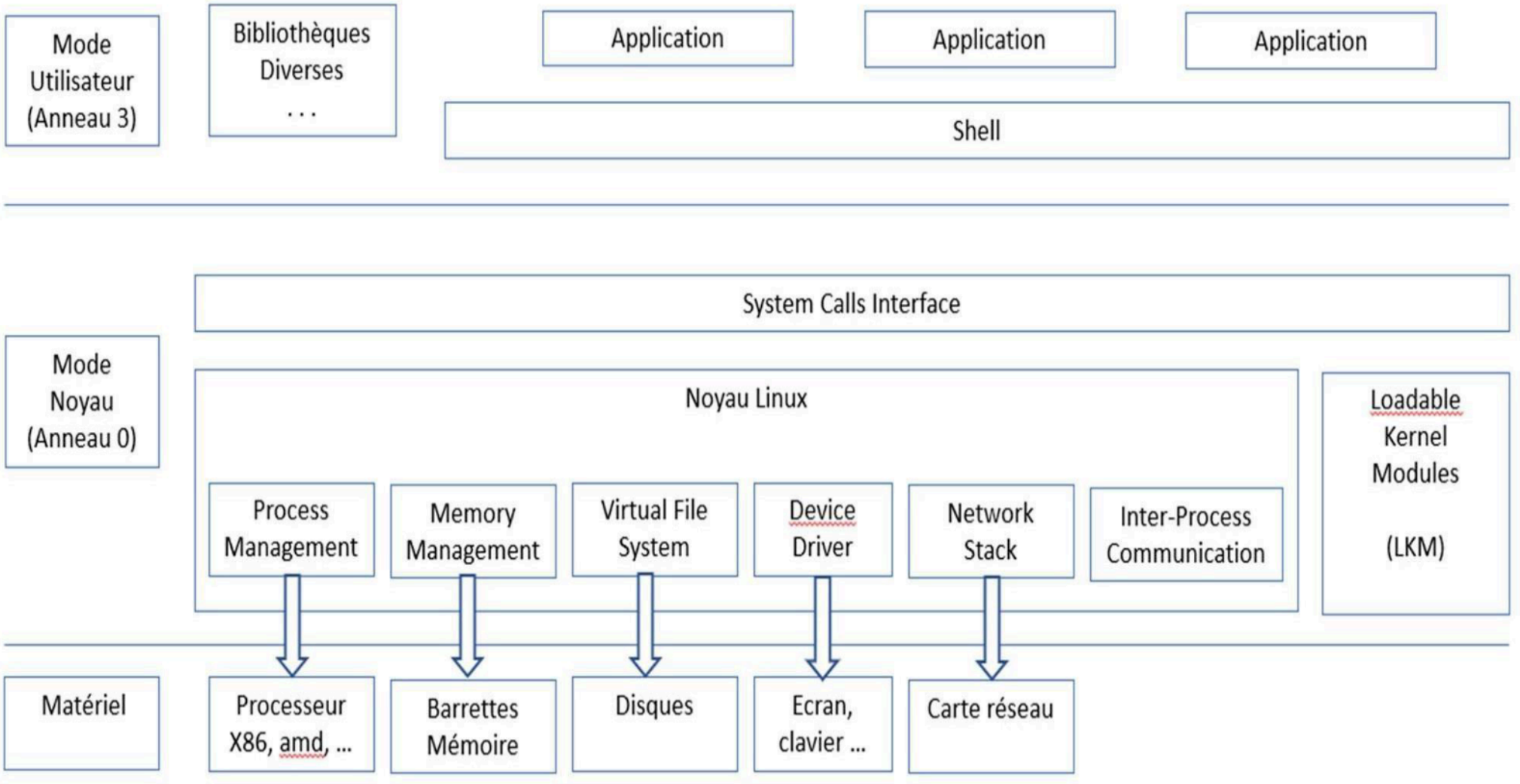
Il faut connaître son système…

Pour rappel une distribution est l'assemblage d'un outil d'installation, d'un outil d'amorçage, d'un noyau, de différents outils du projet GNU (driver, shell, compilateur, logiciels, …)

Pour vérifier le système d'exploitation, utiliser la commande uname avec l'option -o

```
uname -o
```

Installation



Installation

Pour connaître la distribution de son système...

Utiliser la commande lsb_release avec l'option d ou editer le fichier os-release de **/etc**

```
cat /etc/os-release
```

```
NAME="AlmaLinux"   VERSION="8.9 (Midnight Oncilla)"

ID="almalinux"

ID_LIKE="rhel centos fedora"   VERSION_ID="8.9"

PLATFORM_ID="platform:el8"   PRETTY_NAME="AlmaLinux 8.9

(Midnight Oncilla)"   ANSI_COLOR="0;34"

LOGO="fedora-logo-icon"

CPE_NAME="cpe:/o:almalinux:almalinux:8::baseos"

HOME_URL="https://almalinux.org/"

DOCUMENTATION_URL="https://wiki.almalinux.org/"

BUG_REPORT_URL="https://bugs.almalinux.org/"
```

Installation

L’architecture son système…

La commande uname avec différentes options,

-m ou - -machine pour l'architecture de la machine,

-p ou - -processor pour le type de processeur,

-i ou - -hardware-plateform pour la plateforme matérielle.

Mais aussi la commande arch ou la variable MACHTYPE.

uname -m

uname -p

uname -i

arch

echo $MACHTYPE

Installation

Le mécanisme d'amorçage

Le mécanisme d'amorçage (boot loader) est un processus qui permet de charger le système d'exploitation.

Il peut proposer un menu pour choisir le système d'exploitation à charger parmi une liste de plusieurs système (cas du multi-boot).

Après LILO et GRUB Legacy, aujourd'hui c'est généralement GRUB2 qui est utilisé.

Traditionnellement c'était le micrologiciel BIOS qui avait en charge de trouver le disque d'amorçage grâce à son MBR de seulement 512 octets. Aujourd'hui c'est au micrologiciel EFI de trouver le disque système et son GPT plus étendu et donc amène de contenir plus d'informations pour pouvoir gérer de nouvelle structure (LVM) et une famille plus étendue de systèmes de fichiers.

La présence du fichier grub.cfg sous /boot/grub2 signifiera que le système utilise GRUB2. Sinon c'est le fichier grub.conf ou menu.lst qui sera présent sous /boot/grub.

Gestion de l'identité

- Gestion des utilisateurs et groupes
- Profils et environnements
- Durcir l'identité

Utilisateurs

Tout ce trouve dans un fichier

/etc/passwd

/etc/shadow (mots de passe)

nom password numerouser numerogroup commentaire repertoire shell

UID identifiant unique

GID identifiant de groupe

Groupes

/etc/group

nomgroup champ numerogroup membre1,membre2

Commandes

adduser user (commande interactive)

useradd [-u uid [-o] [-i]] | -g group | -G group[[,group]...] | -d dir | -s shell | -c comment | -m [-k skel_dir] | -f inactive | -e expire] login

addgroup (commande interactive)

useradd --home-dir /home/user --gid group1 --groups group1 --uid 1001 --password 123456 user (commande non interactive)

groupadd --gid 1000 group (commande non interactive)

Les droits sur les fichiers ou répertoires

r (4) autorisation de lecture

w (2) autorisation d'écriture

x (1) autorisation d'exécution

Aucun droit	---	0
Exécution	--x	1
Écriture	-w-	2
Écriture et exécution	-wx	3
Lecture seulement	r--	4
Lecture et exécution	r-x	5
Lecture et écriture	rw-	6
Tous les droits	rwx	7

Exemples

chmod u+rw mon_fichier donne au propriétaire les droits en écriture et en lecture au fichier mon_fichier.

chmod -R a+rx mon_dossier donne à tous les utilisateurs les droits en lecture et en exécution à tout ce que contient le dossier mon_dossier. Le "a" est facultatif : chmod -R +rx mon_dossier fonctionne tout aussi bien.

chmod 755 mon_dossier donne au propriétaire tous les droits, aux membres du groupe et aux autres les droits de lecture et d'accès. C'est un droit utilisé traditionnellement sur les répertoires.

chmod 644 ou 640 mon_fichier donne au propriétaire les droits de modification et lecture, aux membres du groupe et aux autres uniquement les droits de lecture. C'est un droit utilisé traditionnellement sur les fichiers.

Commande

chown [-hHLPR] [utilisateur][:groupe] cible1 [cible2 ..]

exemple :

chown -R user:group /documents/travail

TP

1/ Création compte formation et nouveau home (/documents/travail)

2/ Modifier les droits de l'ancien home (plus de droits) (si /home/formation existe)

3/ Changer les droits du nouveau home (chmod et chown)

4/ Test de connexion voir reboot de la machine

UID 10000

GID 10000

Initiation aux packages

- Introduction aux systèmes de gestion des packages
- Déployer des packages, gérer les conflits
- Mettre à jour
- Compiler et installer des sources
- Gestion des paquets RPM, DEB, Snap
- Installation d'une application depuis une archive Tar
- Compilation et installation à partir de sources

Installation d'applications

- L'installation des applications

- Mise a jour d'une application, du système, du noyau

- RPM vs APT

rpm / yum

rpm -ivh (installation du paquet)

rpm -e (suppression du paquet)

rpm -Uvh (mise à jour du paquet)

rpm -q (pour connaitre les détails d'un paquet)

rpm -qa (pour connaitre tous les paquets existants)

rpm -f (pour interroger un paquet contenant un fichier)

rpm -p (pour connaitre les infos d'un paquet)

rpm -Va (pour vérifier tous les paquets)

rpm -Vp (pour vérifier un paquet en particulier)

yum install (installation d'un paquet)

yum update paquet (met à jour le paquet)

yum update (met tout à jour)

yum check-update (vérifie les mises à jour)

yum search (cherche un paquet)

yum list installed (liste les paquets installés)

yum remove (supprime le paquet)

apt

apt-get install (installation depuis internet)

dpkg -i (installation locale)

apt-get install <paquet(s)>

apt-get -f install (ajoute les dépendances manquantes)

apt-cache search (liste le programme)

apt-get remove (supprime le paquet)

apt-get autoremove (supprime le paquet avec ses dépendances)

apt-get purge (supprime le paquet ainsi que sa configuration)

apt-get autoremove --purge (supprime le paquet ainsi que sa configuration et ses dépendances)

apt-get autoclean (supprime la copie locale des paquets installés)

apt-get update (met à jours la liste des dépots ou serveurs)

apt-get dist-upgrade (met à jour en tenant compte de la version de la distribution)

Compilation et installation

Compilation et installation à partir de sources

http://sourceforge.net

http://freecode.com

(pong…) https://d1cxvcw9gjxu2x.cloudfront.net/projects/2076#pong-1.0.tar.gz

tar zxvf archive.gz (apt-get install make gcc)

cd archive

./configure

make

make install (root)

Travaux pratiques

- Installation d'un paquet RPM ou DEB
- Compilation et installation de Apache HTTP Server depuis les sources

Architecture système

Architecture système

Le noyau Linux

Le noyau aussi appelé "kernel" en Anglais, est le composant principal du système. Il gère les ressources de l'ordinateur et sert de moyen de communication entre le matériel et les programmes.

Architecture système

Les sous-systèmes du noyau

Les six principaux sous système du noyau sont :

- Le "process manager" qui gère l'accès aux processeurs.

- Le "Memory manager" qui gère l'affectation des zones mémoire.

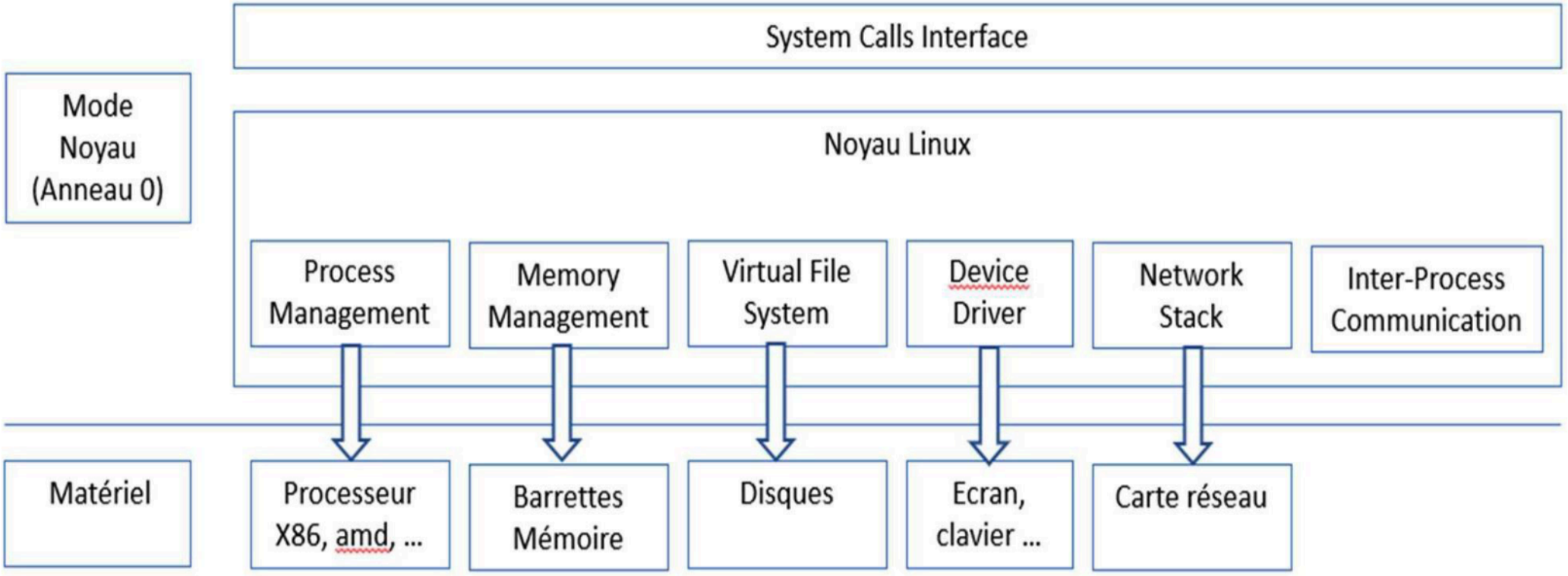
- L'"Inter-Process communication" qui permet la communication entre processus.

- Le 'Virtual File System" qui s'occupe de la gestion et de l'accès aux fichiers. Il dispose d'un jeu d'instruction unique quel que soit le système de fichiers utilisé. C'est le noyau qui fera l'adaptation en fonction du système de fichiers utilisé.

- Le "Device driver" permet l'accès aux ressources matérielles.

- Le "Network stack" assure la connexion des systèmes entre eux via le réseau informatique. Il supporte de nombreux matériels et plusieurs protocoles réseaux.

Architecture système



Architecture système

Les versions du noyau

Depuis la première version 0.01 du noyau Linux, créé par Linus Torvalds en 1991, de nombreuses évolutions sont apparues. En 1994 la version 1.0.0 devient la première version stable du noyau. La version 2.0.0 sortie en 1996 apporte entre-autre le support des plateformes multi-processeurs et la prise en charge de nouveaux matériels. En 2011 c'est la sortie de la version 3.0.0, en 2015 la version 4.0.0 fait son apparition.

La version 5.0.0 sort en mars 2019.

Bien entendu, les différentes distributions n'intègrent pas toutes les versions du noyau. Certaines essaye de suivre plus ou moins ces évolutions, d'autres fige un noyau pour toute la durée de vie d'une version de la distribution.

Dernière version 6.9.1 (17 mai 2024)

https://www.kernel.org/

Architecture système

Le numéro de version du noyau est composé de plusieurs nombres :

Le premier est le numéro majeur de la version.

Le second est le numéro mineur (avant le noyau 2.6, une valeur paire indiquait une version stable).

Le troisième nombre indique le niveau de correction de bugs ou de nouvelles fonctionnalités.

Le quatrième nombre indique le niveau de "patch", correction de bugs, de sécurité sans apport de nouvelles fonctionnalités.

Il existe plusieurs façons pour connaître la version de son noyau.

La traditionnelle commande uname avec les options suivantes :

-s ou - -kernel-name affiche le nom du noyau

-r ou - -kernel-release affiche la version du noyau

-v ou - -kernel-version affiche la date de la version du noyau

Il est également possible d'éditer le fichier /proc/version

Architecture système

Comme son grand frère Unix, le noyau Linux est aujourd'hui modulaire, et ce depuis la version 2.0.

Cela veut dire que le noyau n'intègre plus la totalité du code dont il a besoin, mais seulement une base de référence. Cette base sera complétée par le chargement de modules complémentaire, les LKM (Loadable Kernel Modules). Ces LKM peuvent être un gestionnaire de système de fichiers, des pilotes de

périphériques, un pare-feu, des protocoles réseau, etc.

Les LKM peuvent être chargés et déchargés dynamiquement sans besoin de recompiler le noyau.

La gestion d'une fonctionnalité en tant que module, évite de devoir l'intégrer au noyau et d'alourdir celui-ci.

Architecture système

Un noyau léger sera plus performant qu'un noyau lourd. Il ne gère que l'essentiel. Cela ne sert à rien de disposer de nombreux drivers dans le noyau si l'on ne s'en sert pas.

De plus les développeurs de la fonctionnalité (d'un fournisseur tier) restent indépendants des équipes du noyau.

Pour que cela fonctionne il faut seulement s'assurer que le noyau utilisé accepte l'utilisation des modules.

C'est une option de compilation du noyau.

Pour le vérifier, il faut regarder dans le fichier config-version_du_noyau sous /boot

```
grep -i module /boot/config-$(uname -r)
```

Architecture système

Le noyau acceptera son extension via les modules si l'option "CONFIG_MODULES est à Y

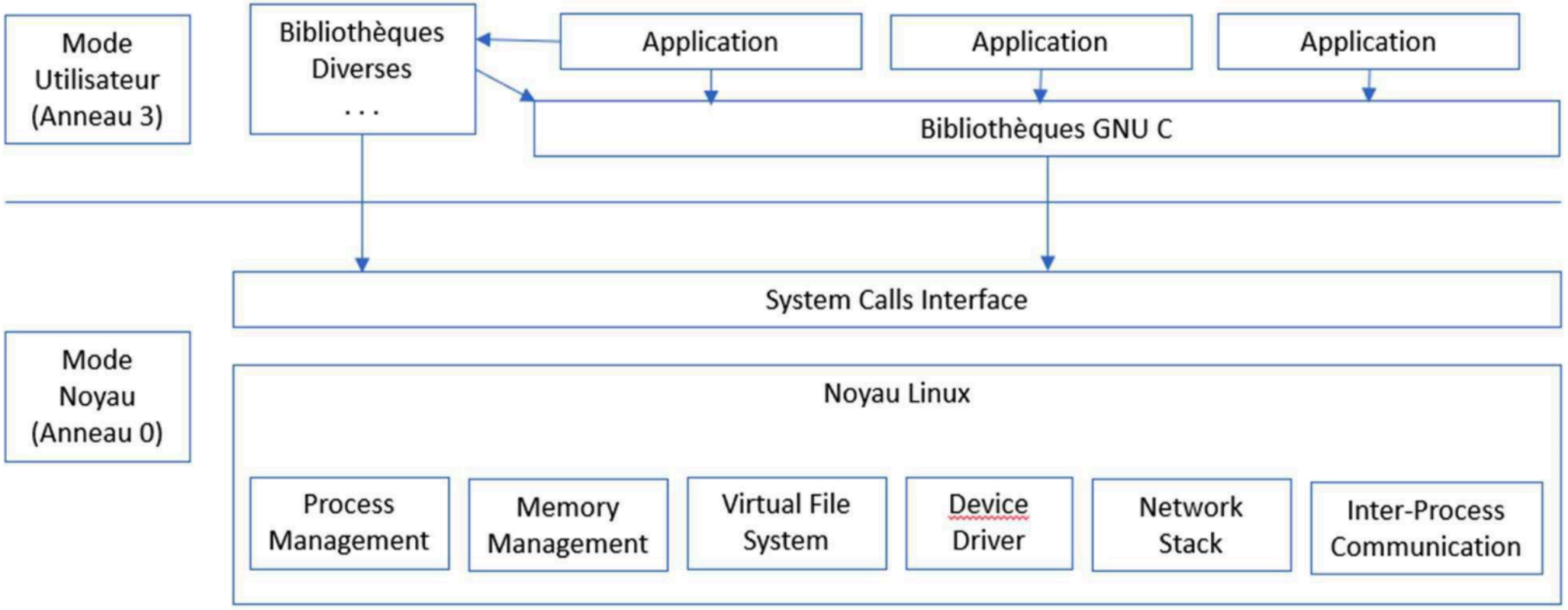
Pour visualiser la liste des modules chargés, on utilisera la commande lsmod.

```
lsmod | more
```

Architecture système

Les appels système

Pour communiquer avec le noyau les applications utilisent les appels système (syscall).



Architecture système

Le système Linux comporte un peu moins de 400 appels système organisés par catégories.

La gestion des processus et threads

La gestion des signaux

La gestion du système de fichiers

La gestion du temps et statistiques

La gestion des protections

Les appels système sont décrits dans la section 2 des pages du manuel (man 2 intro)

Architecture système

Les bibliothèques

Lors de l'écriture d'un programme, le développeur ne va pas tout réécrire, il va s'appuyer sur des fonctions existantes (par exemple printf pour formater un affichage). Le code de ces fonctions est présent dans une bibliothèque (library). Les bibliothèques, conformément au FHS (Filesystem Hierarchy Standard) sont situées dans les répertoires /lib, /lib64, ou /usr/lib, /usr/lib64.

Aussi, lors de la compilation du programme, le développeur va pouvoir choisir entre deux possibilités.

Architecture système

Utilisation de bibliothèques statiques.

Une bibliothèque statique, fichier avec extension ".a", sera copiée dans le code de l'exécutable. Cela donnera la création d'un programme complet avec l'assemblage de tout le code nécessaire à son exécution, ce programme aura l'inconvénient d'être relativement lourd mais le gros avantage d'être autonome.

Pour visualiser les fichiers objets ".o" contenus dans une bibliothèque, il est possible d'utiliser la commande ar. La commande ar est une commande de gestion d'archives qui permet aussi de lister le contenu d'une bibliothèque statique. Cette commande est présente sur CentOS 7 mais doit être installée sur Debian (paquetage binutils).

Architecture système

Quelques options de la commande ar :

-t liste de contenu d'une archive ou bibliothèque

-x extrait un fichier objet

-v mode verbeux

Exemple de liste d'objets pour la bibliothèque librt.a

ar -t librt.a

cd /root

Installer wget

wget https://github.com/darkw0lf/centos/raw/master/librt.a

ar -t librt.a

Attention !!! --- y a plus de librairies statiques ---

Architecture système

Utilisation de bibliothèque partagées

Une bibliothèque partagée, fichier d'extension .so (share object) devra être présente au moment de la compilation pour établir un lien entre le code de l'exécutable et celui de la fonction dans la bibliothèque partagée.

Le programme créé sera plus léger que précédemment, mais si la bibliothèque partagée est absente ou dans un mauvais répertoire, le programme ne pourra pas fonctionner.

Lors de l'installation du système, il existe de nombreuses bibliothèques partagées dédiées au langage GNUC (glibc), aux fonctions mathématiques (libm), aux fonctions temps réel (librt) et a bien d'autres fonctionnalités.

Architecture système

Les pilotes de périphériques

Un "driver" ou pilote de périphérique est un programme qui permet à l'ensemble des applications du système de communiquer avec un périphérique.

Les systèmes GNU/Linux comportent trois types de pilotes.

Les pilotes en mode bloc pour la communication en bloc de données avec les périphériques tels que les disques ou le lecteur cdrom.

Les pilotes en mode caractère qui communique avec le périphérique par un flux d'octets (caractère).

Ce mode de communication est utilisé sur les ports séries, parallèle, les lecteurs de bandes magnétiques et surtout les terminaux.

Les pilotes réseau destinés à contrôler les différentes ressources réseau.

Architecture système

Les fichiers spéciaux

Les pilotes en mode bloc ou en mode caractère sont représentés par des fichiers de taille nulle appelés des fichiers spéciaux.

Ils sont situés dans le répertoire /dev. Leur type est b ou c.

Exemple de liste partielle du répertoire /dev :

```
ls -l /dev
```

Architecture système

Les fichiers spéciaux contiennent deux valeurs numériques visibles par l'affichage long de la commande ls à la place de la taille des fichiers.

Le major qui représente le pilote de périphérique.

Le minor qui représente le périphérique.

Un pilote de périphérique peut bien entendu gérer plusieurs périphériques.

http://pficheux.free.fr/articles/lmf/drivers/

Architecture système

La liste des pilotes de périphériques découverts par le noyau est visible dans le fichier /proc/devices

Exemple partiel du fichier /proc/devices

cat /proc/devices

La première colonne indique le numéro de major et la seconde le nom du pilote de périphérique.

Architecture système

Une autre façon de visualiser les périphériques de type bloc est d'utiliser la commande lsblk qui les montre par défaut sous forme arborescente.

lsblk

La commande lsblk dispose également des options suivantes :

-d - -nodeps N'affiche pas les dépendances

-f - -fs Affiche des informations sur les systèmes de fichiers

-m - -perms Affiche les droits et le propriétaire du périphérique

-l - -list Affiche le résultat au format liste

-n - -noheadings N'affiche pas la ligne d'entête

-r - -raw Affiche le résultat non formaté

Architecture système

Les pilotes réseau

L'interface réseau peut être un physique ou logique. Elle est en charge d'envoyer et recevoir les paquets sur le réseau.

L'interface n'est pas représentée par un fichier dans le répertoire /dev. Il n'y a pas de représentation de l'interface dans le système de fichiers.

La communication entre le noyau et l'interface réseau devra donc s'effectuer de manière spécifique à ce type de périphérique.

Architecture système

Depuis la version 7 de Red Hat (CentOS) et 9 de Debian le nom logique de l'interface a évolué.

Il comporte maintenant l'emplacement physique du périphérique (ex : enp0s3) si possible, sinon l'ancienne appellation ethx est reconduite.

Le nom logique est maintenant construit en tenant compte du type d'interface, éthernet (en), réseau local sans fil (wl, pour WLAN) et, réseau étendu sans fil (ww, pour WWAN).

Ce préfix est suivi par l'indication du bus (p0) et celui du slot (s3) soit par exemple enp0s3.

Architecture système

La gestion des services

Un service est un programme qui est activé lors du chargement du système d'exploitation et qui s'exécute en arrière-plan.

Ces programmes peuvent être par exemple une base de données, un pare-feu, un serveur web, etc.

Architecture système

Un peu d'histoire

Depuis le début de Linux, les services étaient démarrés par le process /sbin/init, héritage de System V init d'Unix.

Son fichier de configuration était /etc/inittab, il était basé sur la notion de niveau (Run Level).

Bien que cette notion n'ait plus vraiment de sens aujourd'hui, elle est toujours présente pour des raisons de compatibilité.

Le niveau de fonctionnement peut être obtenu par les commandes :

"runlevel" et " who -r"

who -r

runlevel

Architecture système

Utilisation de System V Init.

Au démarrage du système le processus /etc/init est lancé. Celui-ci va chercher dans le fichier /etc/inittab le niveau de fonctionnement souhaité pour le système.

Cela va lui éviter de démarrer tous les services du système, mais seulement ceux nécessaires au niveau de fonctionnement sélectionné.

Le principe est basé sur le contenu du répertoire /etc/init.d qui contient tous les scripts qui peuvent être lancés au démarrage, et chaque niveau dispose de son propre répertoire sous la forme /etc/rcx.d dans lequel se trouve des liens symboliques préfixés S ou K suivi du numéro d'ordre d'activation du script pour lancer ou arrêter le service concerné, exemple S10cron.

Architecture système

La gestion manuelle d'un service s'effectue avec la commande "service <nom_du_service> option.

Les options possibles sont start, stop, restart, reload et status.

Ce mécanisme n'est plus utilisé sur les distributions actuelles sauf exception. Pour info il existe un fork de Debian basé sur System V init qui s'appelle Devuan. RHEL5 est la dernière version de Red Hat à l'utiliser.

Canonical Ltd a développé une version plus moderne de démarrage, toujours basé sur le processus /etc/init, connu sous le nom d'upstart.

Cette méthode fut utilisée par Ubuntu à partir de la version 9.10 et par Red Hat sur la version 6.

Elle a sa propre commande de gestion de service : initctl.

Elle s'utilise sous la forme : initctl option <nom_du_service>

Architecture système

Aujourd'hui Systemd

Red Hat 7 ou 8 et Debian 9 ou 10 gèrent maintenant leurs services avec Systemd.

Systemd est un nouveau gestionnaire de services au sens large conçu pour le noyau Linux avec les possibilités d'aujourd'hui tout en gardant une certaine compatibilité avec System V Init et Upstart

Il permet une sélection fine des services et est capable de paralléliser les actions.

Systemd ne gère pas seulement les services, mais il gère aussi les journaux (journald), le swap, les périphériques (udev), les systèmes de fichiers et bien d'autres encore. De ce fait la documentation de systemd est très vaste.

Architecture système

La configuration de systemd s'appuie sur les unités (units). Elles sont rangées dans les répertoires /usr/lib/systemd/system ou /etc/systemd/system. Elles sont visibles sous la forme d'un nom et d'un type, exemple crond.service pour le lancement du service cron.

La principale commande de systemd est "systemctl"

Visualisation des différents types d'unités.

systemctl -t help

La liste paginée des différentes unités chargées actives peut s'obtenir avec la commande :

systemctl list-units

Architecture système

La gestion des services sera également faite par la commande systemctl.

Son usage sera : systemctl option <nom_du_service> avec dans les options start, stop, restart, reload, et status.

Exemple d'usage pour connaître l'état d'un service.

systemctl status bluetooth

bluetooth.service - Bluetooth service

Loaded: loaded (/usr/lib/systemd/system/bluetooth.service; enabled; vendor preset: enabled)

Active: inactive (dead)

Architecture système

La journalisation système

C’est toujours effectué par le service rsyslogd. Toutefois pour être exact, avec l'arrivée de Systemd, c'est maintenant le service systemd-journald qui collecte les évènements et les transmets à rsyslogd pour stockage dans le répertoire /var/log. Attention tous les fichiers de ce répertoire ne sont pas éditables directement, certains sont en binaire et nécessite des commandes spécifiques pour les éditer.

Les règles de rsyslogd sont toujours basées sur 24 "facilités" (type de message) et 8 niveaux de gravité.

Architecture système

Présentation de journald

Aujourd'hui journald et rsyslogd cohabite, ne serait-ce que pour assurer la continuité des exploitations, toutefois c'est bien journald qui est en charge du travail de collecte des différentes informations et qui les stocke dans un espace mémoire.

Ce principe est visible dans le fichier de configuration /etc/rsyslog.conf.

head /etc/rsyslog.conf

Architecture système

Les fichiers journaux issus de journald sont en mémoire dans un buffer sous /run/log/journal. Ils ne sont donc pas permanents. La quantité d'information stockée dépend de la mémoire disponible. Il y a là aussi un mécanisme circulaire.

Pour un stockage permanent, journald compte sur rsyslogd pour ranger les données dans /var/log.

Journald dispose potentiellement de son propre mécanisme de stockage permanent mais il n'est pas activé par défaut. Voir le fichier /etc/systemd/journald.conf. Des limites de taille de stockage et d'utilisation des ressources du système sont paramétrables. Il est de même possible de gérer les choix de rotation des journaux de journald.

Architecture système

La commande journalctl permet d'exploiter directement les informations collectées par journald.

Voici quelques options de journalctl.

-f - -follow Affiche en continu sur l'écran les entrées du journal

-r - -reverse Affiche les messages les plus récents en premier

-S - -since Affiche depuis une date et heure donnée selon le format "2020-05-27 15:00:00" les mots today, tomorrow et yesterday sont aussi utilisable. Il est aussi possible d'exprimer une durée avant ou après l'heure actuelle avec "+" ou "-" suivi d'une valeur

-U - -until Affiche jusqu'à une date (voir format ci-dessus)

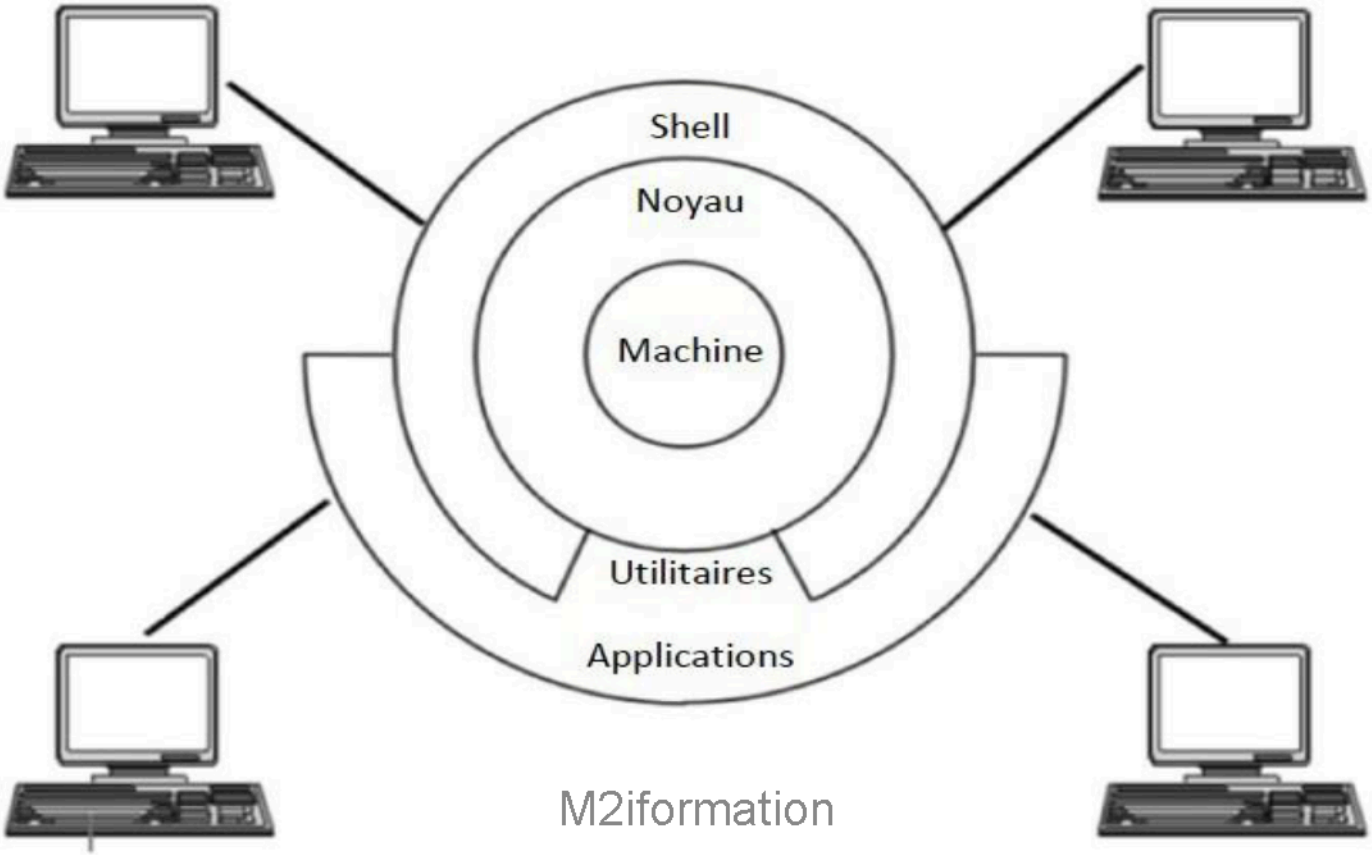
-k --dmesg Affiche seulement les messages du noyau.

- -disk-usage Affiche l'espace disque occupé par les fichiers journaux.

Architecture système

Les Shells

Un shell est un programme qui sert d'interface entre l'utilisateur et le système. Il sera utilisé directement à travers l'interpréteur de lignes de commandes CLI (Command Line Interface) ou habillé sous l'interface graphique GUI (Graphical User Interface).



Architecture système

Sous GNU/Linux nous disposons de plusieurs shell:

Le Bourne Shell (sh), shell par défaut d'Unix depuis 1977 n'est pas open source. Son abréviation historique sh signifie aujourd'hui shell par défaut et est un lien symbolique vers celui-ci

which sh

Architecture système

L'Almquist Shell et Debian Almquist Shell (Ash et Dash), est un shell dérivé du Bourne shell, rapide, peu gourmand et compatible avec la norme Posix.

Il est intégré dans "BusyBox" avec un ensemble de commandes standards d'Unix pour former un seul exécutable.

Il est de ce fait présent dans de nombreux systèmes embarqués.

Une version libre est développée par Debian pour sa distribution, c'est le Debian Almquist Shell (Dash) qui est le shell par défaut.

Architecture système

Le C-shell et le Tenex C-shell (csh et tcsh), évolution du Bourne shell, il s'en éloigne par sa syntaxe proche du langage C. Unix BSD en sera le premier utilisateur.

Le Tenex C-shell apporte de nouvelles fonctionnalités.

Le tcsh est proposé par Debian et Red Hat mais n'est pas installé par défaut.

Architecture système

Le Korn shell (ksh), développé à partir du Bourne shell, il intègre des fonctionnalités du C-shell, compatible Posix, il devient le shell de référence de nombreux système Unix.

Sa version la plus répandue est le ksh.

Architecture système

Le Bourne Again shell (bash), développé dans le cadre du projet GNU en lieu et place du Bourne shell, il est compatible avec ce dernier ainsi que Posix. Il intègre des fonctionnalités du ksh et du csh.

Le projet est toujours ouvert et évolue régulièrement. C'est le shell emblématique de Linux.

Pour connaître sa version il faut éditer la variable BASH_VERSION

```
echo $BASH_VERSION
```

Architecture système

Le shell de connexion, shell qui comme son nom l'indique est le shell d'accueil de l'utilisateur, est associé à celui-ci lors de la création de son compte.

Il est déclaré dans le fichier /etc/passwd

grep stagiaire /etc/passwd

Architecture système

La documentation

La première documentation de Linux est le manuel, connu sous le nom de "man pages". C'est la documentation de référence aussi bien pour l'utilisateur, que le développeur ou l'administrateur.

Si cette documentation n'est pas installée, il faudra installer le paquetage "manpages" chez Debian et "man-pages" chez Red Hat. Pour la version française il suffit d'ajouter -fr au nom du paquetage.

Attention, les contenus ne sont pas identiques dans les différentes langues.

Sur Debian, les pages de manuel pour le développement ne sont pas toujours installées. Il faut alors installer le paquetage "manpages-dev" pour disposer de la section 2.

Une consultation de man <n° section> intro est souvent intéressant à lire.

Le noyau

Le noyau

Faut-il toujours compiler le noyau ?

La compilation du noyau a toujours été un vaste sujet de conversation.

La compilation du noyau, obligatoire au début des systèmes GNU/Linux, n'est plus forcément d'actualité depuis la sortie du noyau 2.0

Quelles sont les raisons qui pourraient amener à compiler un noyau ?

- Vouloir comprendre le fonctionnement du noyau

- Prendre en charge un matériel qui n'est pas supporté par le noyau

- Appliquer des correctifs

- Redimensionner son noyau pour des raisons de performance ou de sécurité

Le noyau

En entreprise, il peut exister un service spécifique d'intégration et de tuning du noyau pour fournir aux autres entités un noyau maison affiné et adapté à des besoins spécifiques.

Mais cela ne concernera qu'un faible nombre de personnes.

Ne perdez pas de vue que toute une équipe de développeurs spécialisés a contribué au développement des versions de noyau mis à disposition sur le site kernel.org, alors que vous êtes seul pour mettre au point vos améliorations.

Le noyau

La liste des noyaux proposés aujourd'hui devrait répondre à la majorité de vos besoins. Dans ces noyaux, pour un système en production, préférez toujours les versions stables plutôt que la dernière évolution.

Le noyau

Les fichiers du noyau.

Le noyau est constitué de quatre fichiers

vmlinuz l'image du noyau

System.map symboles du noyau requis par les modules pour permettre le lancement des fonctions du noyau

initrd chargeur des pilotes nécessaires au démarrage du noyau. Il utilise le système de fichiers "initramfs". L'image est chargée en RAM et offre un système minimal pour charger le système de fichiers principal.

symvers fichier construit lors de la compilation du noyau, il contient la liste des symboles exportés et les modules compilés pour ce noyau.

Ces quatre fichiers sont liés et correspondent à une compilation. Ils sont complétés par un fichier de configuration qui s'appelle config. Le nom de tous ces fichiers comporte en extension la version du noyau.

Le noyau

Installation d'un nouveau noyau

Il n'est pas nécessaire de compiler un noyau pour en changer. Si l'on souhaite utiliser un noyau plus récent qui prenne en compte un nouveau matériel par exemple, on peut facilement télécharger une version plus récente.

Deux possibilités s'offrent à nous un ancien noyau LTS mais plus récent que le nôtre ou le dernier noyau ML (Main Line) stable disponible.

Le noyau

Installation d'un nouveau noyau

CentOS Linux 7.x exécute la version **3.10.xx LTS** du kernel Linux, kernel qui ne supporte pas les nouveaux matériels actuels.

Il peut être donc nécessaire de mettre à jour l'ancien noyau pour un meilleur support matériel.

ELRepo est un référentiel RPM pour les packages Enterprise Linux.

Celui-ci prend en charge Red Hat Enterprise Linux (RHEL) et ses dérivés (Scientific Linux, CentOS et autres).

Le projet ELRepo se concentre sur les packages liés au matériel.

Le référentiel contient un grand nombre des derniers pilotes pour les systèmes de fichiers, les cartes graphiques, les cartes réseau, les cartes son, les webcams et les écrans.

Le noyau

Installation d'un nouveau noyau

- Premièrement s'assurer que votre noyau est à jour.

yum update (facultatif)

- Installer le dépôt additionnel ELRepo

rpm --import https://www.elrepo.org/RPM-GPG-KEY-elrepo.org

yum install https://www.elrepo.org/elrepo-release-8.el8.elrepo.noarch.rpm

yum --disablerepo='*' --enablerepo='elrepo-kernel' list

Le noyau

Installation d'un nouveau noyau

- Installation du Kernel LT (LTS).

```
yum --disablerepo='*' --enablerepo='elrepo-kernel' install kernel-lt
```

- Rebooter le système et sélectionner le nouveau noyau.

Le noyau

Le cas du Kernel panic

Le "Kernel panic" est une erreur noyau qui peut survenir dans plusieurs circonstances.

Il existe deux catégories de kernel panic :

- Matériel : un élément matériel est défaillant, ou géré par le noyau...

- Logiciel : erreurs dans le code du noyau, dans un module noyau, une compilation ou une installation du noyau incorrectes, etc.

Un kernel panic peut arriver à tous moments et provoque l'indisponibilité du serveur. Un message d'erreur peut apparaître à l'écran. C'est un niveau de gravité très élevé.

Le noyau

Analyse de la situation.

- Est-ce la première fois ?
- Avez-vous modifié quelque chose juste avant l'apparition du problème ? Si oui, quoi ?
- Est-ce un plantage récurrent ? À quelle fréquence ?
- Depuis combien de temps avez-vous ce problème ?
- Pouvez-vous reproduire le problème ?
- Le système est-il totalement figé ?
- Y a-t-il eu des interventions sur cet ordinateur ?

Lors de l'amorçage, notez les messages du noyau précédant le kernel panic...

Le noyau

Consultez les différents journaux du système, messages, dmesg, kern… Sur Red Hat il faut penser à tracer le journal du noyau en modifiant la ligne "kern.*" dans le fichier /etc/rsyslog.conf

Par exemple, ajouter une ligne pour renvoyer les messages du noyau vers un fichier.

Comme ceci :

```
vi /etc/rsyslog.conf

. . .

#### RULES ####

# Log all kernel messages to the console.

# Logging much else clutters up the screen.

#kern.* /dev/console

kern.* /var/log/kern.log
```

Le noyau

Après, vous pouvez paramétrer la rotation du journal dans le dossier /etc/logrotate.d/.

Il ne reste plus qu'à redémarrer le service rsyslog :

```
systemctl restart rsyslog
```

Dès que vous aurez appréhendé le problème, et appliqué la solution, n'hésitez pas à documenter le sujet.

Le noyau (TP)

Customisation d'un nouveau kernel...

yum install ncurses-devel (nécessaire pour menuconfig)

yum install -y make gcc bc bison flex elfutils-libelf-devel openssl-devel grub2

Au sein du système, l'emplacement pour construire le noyau doit disposer d'un espace disque suffisant, prévoyez au minimum 10 Go.

Télécharcher le dernier noyau stable

wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.4.44.tar.gz

Décompresser

tar xf linux-5.4.44.tar.gz

ls -l lin*

Le noyau (TP)

Configurer convenablement toutes les options du Noyau Linux implique d'une part une bonne connaissance sur le fonctionnement de celui-ci et du matériel utilisé (périphériques, processeur(s), mémoire, disques physiques et logiques...) sur lequel vous allez installer le nouveau noyau Linux. Il faut aussi bien identifier l'objectif de la conception de ce dernier.

cd linux-5.4.44

make config

make menuconfig (mieux)

Ou https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.15.159.tar.xz

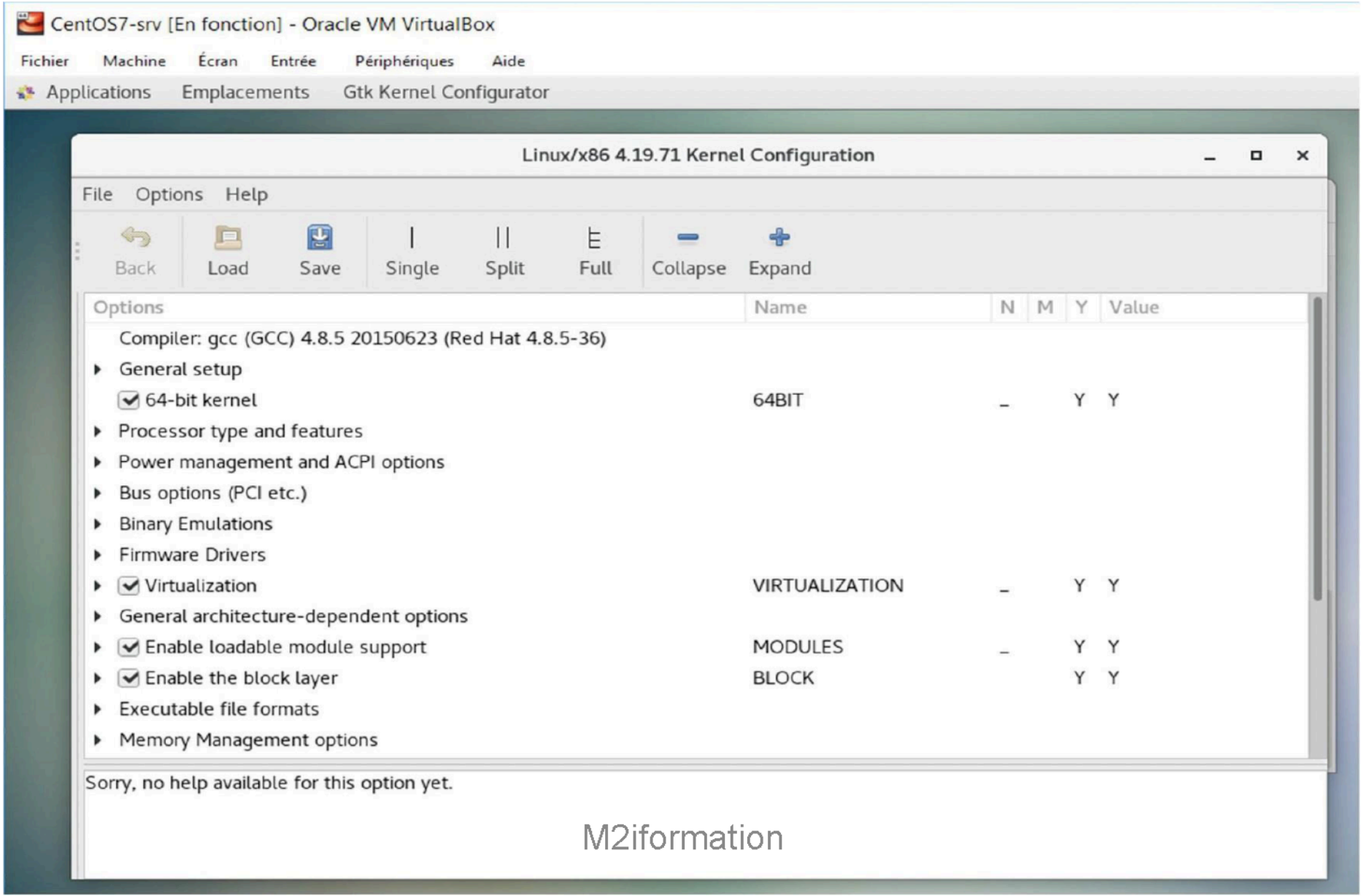
Le noyau (TP)

Autres outils

Bien que peu répandu sur un serveur, si vous disposez d'un environnement graphique, il existe deux outils suivant l'environnement.

gconfig pour le bureau gnome. Il nécessite les bibliothèques GTK.

Le noyau (TP)


CentOS7-srv [En fonction] - Oracle VM VirtualBox
Fichier Machine Écran Entrée Périphériques Aide
Applications Emplacements Gtk Kernel Configurator
Linux/x86 4.19.71 Kernel Configuration
File Options Help
Back Load Save Single Split Full Collapse Expand
Options Name N M Y Value
Compiler: gcc (GCC) 4.8.5 20150623 (Red Hat 4.8.5-36)
General setup
64-bit kernel 64BIT _ Y Y
Processor type and features
Power management and ACPI options
Bus options (PCI etc.)
Binary Emulations
Firmware Drivers
Virtualization VIRTUALIZATION _ Y Y
General architecture-dependent options
Enable loadable module support MODULES _ Y Y
Enable the block layer BLOCK Y Y
Executable file formats
Memory Management options
Sorry, no help available for this option yet.
M2iformation

Le noyau (TP)

Si vous administrez votre serveur en mode texte depuis un terminal distant, make menuconfig ou make nconfig sont probablement les meilleurs choix. Les différentes options du noyau sont activées (**Y**), désactivées (**N**) ou chargeables en tant que modules (**M**) et sont répertoriées ainsi :

64-bit kernel (sélectionné nativement) contient le code 64 bits du noyau Linux. C'est obligatoirement activé.

General setup contient les options générales de configuration du noyau comme le mécanisme de pagination de la mémoire virtuelle, la gestion des IPC compatibles System V, etc.

Enable loadable module support (sélectionné nativement) contient le support des Loadable Kernel Modules (LKM). Ce sujet est du reste abordé dans le chapitre Modules.

Enable the block layer (sélectionné nativement) contient la prise en charge des périphériques de type bloc.

Processor type and features contient le support de plusieurs processeurs.

Power management and ACPI options contient la gestion d'énergie.

Bus options contient la gestion des bus tels que PCI, PCMCIA...

Le noyau (TP)

Executable file formats gère l'utilisation des fichiers binaires au format ELF. Du fait que ce format de fichier est devenu un standard, l'activation de cette option est recommandée.

Networking support (sélectionné nativement) contient la prise en charge des fonctionnalités réseau. C'est une option quasiment obligatoire. Il est rare de voir un serveur GNU/Linux sans services réseau.

Device drivers gère les options relatives aux gestionnaires de périphériques.

Firmware drivers gère les options relatives aux firmwares (BIOS, EFI, DMI...).

File systems active ou non la prise en charge des systèmes de fichiers (ext2, ext3, ext4, JFS, XFS, BRFS...).

Le noyau (TP)

Kernel hacking ajoute une datation aux traces du noyau Linux. Cette option n'est pas recommandée pour un usage courant.

Security options permet le stockage des clés privées dans le noyau au lieu des processus.

Cryptographic API (sélectionné nativement) permet d'intégrer la prise en charge les algorithmes de chiffrement SHA1, AES, MD5, MD4...

Virtualization (sélectionné nativement) permet la prise en charge de Kernel-based Virtual Machine (KVM).

Library routines offre la possibilité d'inclure différentes fonctions.

Le noyau (TP)

Fichiers de configuration

Dans le répertoire /boot de votre distribution, un fichier texte contient la configuration du noyau que votre système d'exploitation utilise.

La nomenclature du nom est : config-<version_du_noyau><architecture>.

Distribution	Nom du fichier
Debian	config-4.19.0-9-amd64
CentOS	config-3.10.0-1160.45.1.el7.x86_64
Alma	config-4.18.0-513.9.1.el8_9.x86_64

more /boot/config-4.18.0-513.9.1.el8_9.x86_64

Le noyau (TP)

Avant de construire le noyau, vous souhaitez probablement reprendre les options du noyau en cours d'exécution, donc depuis l'un des fichiers mentionnés ci-dessus.

Avec la commande cp, il suffit de copier le fichier de configuration du noyau en cours situé dans le dossier /boot vers la racine du dossier des sources du nouveau noyau. Il faut lui donner obligatoirement le nom .config :

```
cp /boot/config-4.18.0-513.9.1.el8_9.x86_64 linux-5.4.44/.config
```

Le noyau (TP)

Supprimer les fichiers d'une précédente compilation

Pour nettoyer les traces d'une précédente compilation qui pourraient interférer avec la nouvelle compilation, la commande make dispose de deux directives :

mrproper qui enlève tous les fichiers générés, le fichier .config mais aussi les fichiers de sauvegarde afin de retrouver la configuration de base telle qu'elle était dans l'archive.

Tapez make mrproper dans un terminal :

make mrproper

clean, qui enlève les fichiers générés. Le fichier .config est conservé.

Tapez make clean dans un terminal :

make clean

Le noyau (TP)

Une autre solution est d'utiliser la commande make defconfig :

Attention le résultat obtenu est un fichier de config par défaut. Il ne contient pas forcément tout ce dont on à besoin par la suite sur son serveur.

make defconfig

make menuconfig puis sauvegarder la conf...

Si erreur de compilation...

You can change your config file **.config**

CONFIG_SYSTEM_TRUSTED_KEYS="...-certs.pem"

to

CONFIG_SYSTEM_TRUSTED_KEYS=""

Le noyau (TP)

Si erreur de compilation…

Try to disable CONFIG_DEBUG_INFO_BTF

You can change your config file **.config**

Or you can try disabling CONFIG_DEBUG_INFO_BTF/DEBUG_INFO_BTF_MODULES in .config

CONFIG_DEBUG_INFO_BTF=n

Le noyau (TP)

Construire le noyau

La compilation s'effectue avec la commande make. L'option -j permet d'indiquer le nombre de processeurs du serveur.

make -j2

Un simple make suffit...

make (plusieurs heures...)

make modules

make modules_install

make install

dans le dossier /boot...

ls -l /boot | grep 5.4.44

Le noyau (TP)

Construire le noyau

Ajouter le fichier config :

```
cp .config /boot/config-5.4.44
```

redémarrer le système...

```
reboot
```

Le noyau (TP Ubuntu)

Construire le noyau

Installer les pré-requis :

apt install build-essential libncurses-dev bison flex libssl-dev libelf-dev fakeroot dwarves make

Télécharger le kernel...

wget https://cdn.kernel.org/pub/linux/kernel/v6.x/linux-6.9.1.tar.xz

Copier le fichier config...

cp /boot/config-6.5.0-35-generic ../linux-6.9.1/.config

Le noyau (TP Ubuntu)

Construire le noyau

Lancer le menu et sauvegarder la configuration :

make menu-config (attention quelques erreurs...)

Compiler...

make

Le noyau (TP Ubuntu)

Construire le noyau

Si erreur :

Ouvrir le fichier .config et chercher la ligne :

CONFIG_SYSTEM_TRUSTED_KEYS="/usr/local/src/debian/canonical-certs.pem"

Remplacez par :

CONFIG_SYSTEM_TRUSTED_KEYS=""

Les modules

Les modules

Les Modules

Un module noyau (Loadable Kernel Module en anglais) est un morceau de code destiné à enrichir le noyau.

Il pourra être chargé ou déchargé en fonction du besoin. La liste des modules chargés peut être consulté par la commande lsmod. Ils s'exécutent en mode noyau.

L'intérêt de développer une fonctionnalité sous forme de module est très grand. Cela n'alourdit pas le noyau, et laisse libre son utilisation ou pas.

De plus, si vous êtes une société tierce vous pouvez développer le ou les modules correspondant à votre besoin sans avoir à composer avec les développeurs du noyau. C'est l'administrateur système qui se contentera de le chargé le moment venu.

Les modules

Compilation d'un module, prérequis.

Un module est toujours compilé pour une version précise du noyau. Si celui-ci évolue, les modules liés à la distribution évolueront automatiquement, en revanche les modules issus de développements tiers devront

être recompilés. Toujours bien réfléchir aux différentes conséquences, avant de faire évoluer un système de production.

Avant toutes choses, il est nécessaire de s'assurer de la présence des outils tel le compilateur gcc et l'utilitaire make.

Si ce n'est pas le cas, il faudra les installer (outils de développement et pour le noyau).

yum update (reboot)

yum group install "Development tools"

yum install kernel-devel

Les modules

Création d'un module

Pour approfondir la compréhension des modules et de leur mise en oeuvre, nous allons créer un module basique pour voir les différentes étapes du processus.

Pour cela nous utiliserons le traditionnel "Hello World". L'idée de ce dernier est d'envoyer un message dans le journal "syslog" lors de son chargement ou déchargement.

Nous allons donc écrire un petit code source en langage C, et le compiler pour en faire un module noyau.

Elément d'extension ".ko" (kernel object) ou ".ko.xz" s'il est compressé.

Les modules

Création d'un module

Ecriture du fichier "monmodule.c" à l'aide d'un éditeur de texte.

vi monmodule.c

Les modules

```
// Nom : monmodule.c

// Sujet: Module de noyau Linux

// Formation
#include <linux/module.h>

#include <linux/kernel.h>

MODULE_LICENSE("GPL");

MODULE_AUTHOR("Stagiaire");

MODULE_DESCRIPTION("Module de noyau Linux");

MODULE_VERSION("Version 1");

int init_module(void)

{
printk(KERN_INFO "Le module est chargé.€n");

return(0);

}

void cleanup_module(void)

{

printk(KERN_INFO "Le module est déchargé.€n");

}
```

Les modules

Les trois premières lignes sont des commentaires.

Les deux lignes suivantes citent des fichiers d'entête (.h) nécessaire pour les modules et la définition de macro tel "KERN_INFO".

Les quatre macros qui suivent donnent des informations sur le module. Les deux fonctions "int init_module (void)" et "void cleanup_module (void) vont s'exécuter au chargement du module pour l'une et au déchargement pour l'autre.

La fonction printk permet l'affichage des messages dans le journal du noyau.

Les modules

Création du Makefile

Ecriture du fichier Makefile pour définir les consignes à utiliser lors de la compilation par make.

vi Makefile

Les modules

```
obj-m += monmodule.o

all:
TAB     make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules
clean:
TAB     make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean
install:
TAB     cp ./monmodule.ko* /lib/modules/$(shell uname -r)/kernel/drivers/misc
uninstall:
TAB     rm -i /lib/modules/$(shell uname -r)/kernel/drivers/misc/monmodule.ko*
TAB     depmod -a
```

Les modules

La ligne obj-m (object module) indique qu'il faut créer un objet monmodule.o à partir du fichier source monmodule.c

Le fichier contient ensuite quatre entêtes que l'on pourra utiliser avec le make.

all: sert à construire le module et est utilisé systématiquement avec make.

clean: utilisé lors de l'appel par make clean pour supprimer les fichiers d'une précédente compilation qui se serait mal déroulée.

install: sert à spécifier le lieu d'installation de l'exécutable.

uninstall: indique les fichiers à désinstaller au cas où.

Les modules

La compilation de monmodule.c

Lancer make pour construire le module

```
make
```

Liste des fichiers créés par la compilation.

```
ls
```

Sous Red Hat il est possible de compresser le module noyau avec la commande xz.

La commande xz dispose d'options :

-d ou - -decompress de décompresser de l'archive

-k ou - -keep de conserver le fichier d'origine

-x d'indiquer un niveau de compression (x de 0 à 9)

```
xz -9 monmodule.ko
```

```
ls *.xz
```

Les modules

Installation et chargement du module

Le chargement du module se fait avec la commande modprobe (ou insmod).

Pour installer le module on fera :

make install

Le fichier monmodule.ko ou monmodule.ko.xz sera alors copié dans le répertoire /lib/modules/$(uname - r)/kernel/drivers/misc

Les modules

Pour charger le module mondule.ko.xz et ses éventuelles dépendances il faut utiliser la commande modprobe

Mais avant il faut mettre à jour le fichier modules.dep, fichier qui contient la liste des dépendances de chargement de tous les modules. La commande depmod met à jour le fichier et vérifie tous les modules disponibles.

depmod -a

modprobe monmodule

On peut vérifier si le module est bien chargé

lsmod | grep monmodule

Dans ce cas on doit retrouver le message de lancement dans le journal "messages"

tail /var/log/messages

Les modules

Pour charger le module mondule.ko.xz et ses éventuelles dépendances il faut utiliser la commande modprobe

Mais avant il faut mettre à jour le fichier modules.dep, fichier qui contient la liste des dépendances de chargement de tous les modules. La commande depmod met à jour le fichier et vérifie tous les modules disponibles.

depmod -a

modprobe monmodule

On peut vérifier si le module est bien chargé

lsmod | grep monmodule

Dans ce cas on doit retrouver le message de lancement dans le journal "messages"

tail /var/log/messages

Les modules

Déclaration d'un module pour chargement au démarrage.

Sur CentOS version 7 ou 8, il faut ajouter un script avec l'extension ".conf" sous le répertoire /etc/modules-load.d

Le script et très simple, il doit comporter le nom du module.

vi /etc/modules-load.d/monmodule.conf

Chargement au boot de "monmodule"

monmodule

Les modules

Décharger un module noyau

Deux commandes permettent de décharger un module, rmmod et modprobe.

Options de rmmod:

-f - -force Force le déchargement du module

Options de modprobe:

-r - -remove Décharge le module

-v - -verbose mode verbeux

modprobe -r monmodule

lsmod | grep monmodule

Les modules

Informations diverses sur les modules

Liste des modules présent sur le système

```
find /lib/modules/$(uname -r) -name '*.ko*' | awk -F '/' '{print $NF}' | more
```

Les modules

Liste des modules chargés sur le système

lsmod | more

Les modules

Information sur un module

modinfo xfs

Les modules

Liste des dépendances d'un module,

La commande modprobe avec l'option –show-depends montre les dépendances du module

modprobe --show-depends xfs

Amorçage du système

Amorçage du système

A la mise sous tension du système, le processus de démarrage s'effectue en plusieurs étapes.

Pour commencer, le BIOS effectue des tests de bon fonctionnement du matériel appelés POST (Power-On Self Test), puis il va chercher le secteur d'amorçage.

Le secteur d'amorçage MBR (Master Boot Record), situé dans le 1er secteur du disque, est bootable, il va exécuter le chargeur d'amorçage (bootloader) GRUB. Quant aux systèmes x86 basés sur UEFI, ils montent une partition système EFI qui contient une version du chargeur de démarrage GRUB.

Le gestionnaire de démarrage EFI charge et exécute GRUB comme une application EFI.

Amorçage du système

Le chargeur d'amorçage GRUB (Grand Unified Bootloader) Legacy (version 1) ou GRUB-PC (version 2) proposent dans un écran d'accueil pour sélectionner le noyau Linux à amorcer si, bien sûr, votre système en contient plusieurs. Un délai de quelques secondes vous permet d'effectuer le choix sinon, le processus de démarrage continue avec les paramètres définis par défaut dans le fichier de configuration de GRUB.

Le noyau Linux se charge, initialise les périphériques par le biais des pilotes, démarre le processus noyau kswapd qui est le gestionnaire de swap. Il monte ensuite la racine du système de fichiers, soit /.

Sur la grande majorité des distributions actuelles, il exécute ensuite le programme /libsystemd/systemd, lequel aura le PID 1. Son rôle est de coordonner la fin du processus de démarrage et de configurer également l'environnement de l'utilisateur.

Il devient le parent ou grand-parent de tous les processus qui sont lancés automatiquement sur le système.

Amorçage du système

GRUB

Les versions actuelles de Debian et Red Hat utilisent GRUB Version2.

Selon l'environnement, les commandes ne portent pas tout à fait le même nom : grub- pour Debian et grub2- pour Red Hat.

La version de GRUB est visible sur debian avec la commande :

grub-install -V

et sur Red Hat :

grub2-install -V

Amorçage du système

Les fichiers de configuration :

Le fichier de configuration de GRUB2 est /boot/grub2/grub.cfg

Attention ce fichier ne doit pas être modifié manuellement. La commande grub2-mkconfig est faite pour cela. Elle s'appuie sur le fichier de paramétrage /etc/default/grub que l'on renseignera et sur les scripts présents dans le répertoire /etc/grub.d.

Vue partielle du fichier /boot/grub2/grub.cfg

```
cat /boot/grub2/grub.cfg
```

Amorçage du système

La modification du paramétrage de grub.cfg s'effectuera à partir du fichier /etc/default/grub

```
cat /etc/default/grub

GRUB_TIMEOUT=5

GRUB_DISTRIBUTOR="$(sed 's, release .*$,,g' /etc/system-release)"

GRUB_DEFAULT=saved

GRUB_DISABLE_SUBMENU=true

GRUB_TERMINAL_OUTPUT="console"

GRUB_CMDLINE_LINUX="resume=/dev/mapper/cl-swap  rd.lvm.lv=cl/root

rd.lvm.lv=cl/swap rhgb quiet"

GRUB_DISABLE_RECOVERY="true"

GRUB_ENABLE_BLSCFG=true
```

Le contenu du fichier peut légèrement varier d'une distribution à l'autre.

Amorçage du système

Le répertoire /etc/grub.d contient les scripts de création du menu. Les noms de ces scripts commencent par un nombre qui indique l'ordre de traitement.

ll

```
-rwxr-xr-x 1 root root    01_users
-rwxr-xr-x 1 root root    10_linux
-rwxr-xr-x 1 root root    20_linux_xen
-rwxr-xr-x 1 root root    20_ppc_terminfo
-rwxr-xr-x 1 root root    30_os-prober
-rwxr-xr-x 1 root root    40_custom
-rwxr-xr-x 1 root root    41_custom
```

Après modification de l'un de ces éléments, on utilisera la commande :

grub2-mkconfig -o /boot/grub2/grub.cfg pour mettre à jour le fichier de configuration grub.cfg.

Amorçage du système

Systemd, la gestion des cibles

Pour continuer le démarrage du système, après le traitement d'initramfs, c'est systemd qui prend la relève.

Systemd va tenter d'activer la cible par défaut qui est : default.target. En fait c'est un lien symbolique situé dans le répertoire /etc/systemd/system vers graphical.target en environnement graphique (GUI) et multiuser.target en environnent texte (CLI).

Amorçage du système

La cible par défaut

La commande systemctl get-default affiche la cible par defaut.

systemctl get-default

multi-user.target

Pour modifier la cible par défaut il faut utiliser la commande suivante :

systemctl set-default graphical.target

Amorçage du système

Ou pour revenir sur le choix de l'environnement initial au démarrage :

systemctl set-default multi-user.target

Removed /etc/systemd/system/default.target.

Created symlink /etc/systemd/system/default.target →

/usr/lib/systemd/system/multi-user.target.

Pour une prise en compte immédiate du changement il faut utiliser la commande "systemctl isolate graphical.target". Attention seules les cibles contenant la ligne "AllowIsolate=yes" peuvent être concernées.

Amorçage du système

Liste des cibles

La liste des cibles actuellement chargées et actives s'obtient par la commande :

```
systemctl list-units --type target
```

Si l'on veut voir toutes les cibles chargées même celles non actives il faut ajouter l'option –all

```
systemctl list-units --type target --all
```

Amorçage du système

Les cibles systemd ne sont pas toutes indépendantes. La plupart du temps les cibles forment des arbres de dépendances. Pour atteindre l'état voulu, il va falloir activer un certain nombre de cibles.

L'arbre des dépendances d'une cible, par exemple "graphical.target", s'obtient de la façon suivante :

```
systemctl list-dependencies multi-user.target | grep 'target$'
```

Amorçage du système

Modification de la cible par défaut au moment du boot.

Lors du boot, il est possible d'interrompre le démarrage et d'aller modifier la ligne de lancement du noyau en ajoutant systemd.unit=rescue.target ou emergency.target Lors du boot, il est possible d'interrompre le chargement du système en frappant n'importe quelle touche pendant le décompte lors de l'affichage du menu. Puis de tapper "e" pour rentrer en mode saisie et modifier la ligne commençant par linux16 et lui ajouter systemd.unit=rescue.target

Il est aussi possible, depuis l'environnement utilisateur, de passer en mode secours (rescue) avec la commande :

```
systemctl rescue
```

Ou sans envoie de message aux utilisateurs en ajoutant l'option - -no-wall.

```
systemctl - -no-wall rescue
```

Amorçage du système

Récupération du mot de passe de root

redémarrez le serveur et appuyez sur la touche e dans le menu Démarrer de GRUB pour modifier l'entrée de démarrage.

Supprimez les paramètres rhgb et quiet de la ligne commençant par linux16.

Ajoutez les paramètres suivants à la fin de la ligne Linux16 :

rd.break enforcing=0

Le paramètre rd.break interrompt le processus de démarrage avant que initramfs ne passe le contrôle à systemd. Ainsi, l'invite initramfs peut être utilisée pour la saisie de commandes.

Le paramètre enforcing=0 met SELinux en mode permissif. Cela permet d'économiser le ré-étiquetage du système de fichiers qui serait nécessaire lorsque SELinux est éteint, ce qui peut prendre beaucoup de temps.

Appuyez sur Ctrl+x pour démarrer le système avec les paramètres modifiés.

Amorçage du système

Lorsque l'invite "switch_root:/#" apparait, passer les commandes suivantes :

```
mount –oremount,rw /sysroot
chroot /sysroot
```

Changer le mot de passe de root:

```
passwd root
```

La modification du fichier /etc/shadow en dehors du contexte SELinux nécessite un réétiquetage complet du système au prochain reboot. Pour cela, saisir la commande suivante :

```
touch /.autorelabel
```

Frappez deux fois la commande "exit", une première fois pour quitter le chroot et une deuxième pour relancer le système.

Au reboot de la machine, le nouveau mot de passe est opérationnel.

Amorçage du système

Idem avec un LiveCD Ubuntu…

https://releases.ubuntu.com/24.04.2/ubuntu-24.04.2-desktop-amd64.iso

Booter dessus

sudo su -

monter la partition primaire...

editer le fichier /etc/shadow

supprimer les caractères entre les deux premiers ::

rebooter

Amorçage du système

Protection par GRUB de l'accès restreint au système par un mot de passe.

Il est possible de sécuriser l'accès restreint au système, pour craquer le mot de passe de root par exemple en installant un mot de passe au niveau de GRUB. Il faut pour cela utiliser la commande grub2-setpassword

grub2-setpassword

Cette commande crée le fichier /boot/grub2/user.cfg qui contiendra le mot de passe.

cat /boot/grub2/user.cfg

GRUB2_PASSWORD=grub.pbkdf2.sha512.10000.457CB64294AB71747D3A2EF391F117
65633BE644741FF3236A02542F089DDB3D785B1992EB11DD0BF8C82E6CB87B063010
0FAB2523BF1BFBEFAA01D2603D4BDE.454B6D76A5626B2B881750DD378FA5ABD3FB1
8EE625F2AABC18C5F646D0F9448ECE4B17E5AAF0E76BAA6852E9CADAF1F8EB5107B1
CA64EB792FB977E4E01BBB2

Pour supprimer le mot de passe, il suffira de supprimer le fichier user.cfg

Amorçage du système

Le processus de démarrage, analyse et journalisation

La commande "systemd-analyze" permet d'évaluer le temps de la séquence d'amorçage, le temps de chaque processus et la liste de la chaine critique d'un ou de tous les services lors du démarrage.

Temps de démarrage total du système.

systemd-analyze

```
Startup finished in 611ms (kernel) + 4min 42.938s (initrd) + 8.922s (userspace) = 4min 52.472s
```

Avec la directive "blame" on aurra le temps de chaque processus

systemd-analyze blame

Amorçage du système

Pour avoir la liste de la chaine critique, il faut utiliser la directive "critical-chain"

```
systemd-analyze critical-chain
```

La liste de la chaine critique d'un service sera obtenue en précisant le nom du service. Cela revient à un affichage partiel de la liste complète.

```
systemd-analyze critical-chain NetworkManager.service
```

Amorçage du système

Les caractéristiques d'un service peuvent être obtenues à l'aide de la commande systemctl et de la directive show.

systemctl show sshd

Si l'on veut une information précise, ne pas hésiter à filtrer le résultat avec grep.

systemctl show sshd | grep -i requires

La visualisation du journal de boot peut être obtenue directement par la commande journalctl et l'option b

journalctl -b

Le fichier /var/log/boot.log affiche également des informations sur la phase de boot.

tail /var/log/boot.log

Maintenance

Maintenance du système

Lors du démarrage du système, le matériel est détecté par le service "udev" partie intégrante de systemd.

Udev est le mécanisme qui a en charge la découverte du matériel (Plug and Play)

Avant d'aller analyser le cas des principaux matériels et leur influence sur le système, envisageons les différents types de problèmes que nous pourrions rencontrer.

Maintenance du système

Matériel détecté par le système

Même si le matériel est indépendant du système Linux, si celui-ci ne fonctionne pas correctement, le système ne pourra pas non plus bien fonctionner.

Un composant ayant une panne franche, ne fonctionnant pas, est le cas le plus simple à étudier.

Lors du démarrage du système nous avons vu que le BIOS effectue des tests dès la mise sous tension.

A partir de là on peut avoir des informations sur de nombreux composants :

- L'initialisation des processeurs

- La stabilité des tensions

- La carte mère

- Les différents contrôleurs (graphiques, réseaux, disques ...)

Maintenance du système

Dans certains cas le système peut afficher des messages d'informations à l'écran, dans les différents journaux, mais parfois, il peut aussi se figer sans rien afficher. A-t-il eu le temps de renseigner les journaux ?

Dans ce cas-là on peut toujours essayer le mode "rescue" pour voir les journaux et savoir s'ils ont pu enregistrer quelque chose ou pas.

Si le matériel est victime d'un comportement aléatoire, il va falloir prendre le maximum de traces pour identifier au plus vite la cause du problème.

Est-ce qu'une mise à jour a été faite sur le système, ou sur un périphérique ? Dans ce cas-là le pilote de périphérique est-il à jour ? Est-il compatible ?

Lors d'une évolution du matériel ou du logiciel, toujours avoir en tête la chaine de liaison et les éventuelles contraintes de compatibilités.

Maintenance du système

Matériel non détecté par le système

Le matériel est présent mais il n'est pas détecté par le système.

La première question, le matériel est-il sous tension ? C'est un classique avec le matériel externe.

- Le matériel est-il compatible avec Linux ?
- Le matériel est-il supporté par la distribution ?
- Le pilote du périphérique est-il installé ?
- Existe-t-il un pilote générique pour ce matériel ?

Maintenance du système

Etat des lieux du matériel présent.

Pour découvrir la topographie du système, il y a de nombreux outils, présents dès l'installation ou à installer en complément sur le système.

Par exemple prenons le cas de "lshw"

Cette commande exploite différentes sources d'informations.

Le fichier pci.ids qui est une liste publique de tous les identifiants de périphériques PCI

Les fichiers /proc/bus/pci, /proc/ide, /proc/scsi, /dev/cpu, /proc/device-tree, /proc/bus-usb ...

De base sur Debian, elle n'est pas toujours installée sur Red Hat, toutefois elle est intégrée aux dépôts de la distribution, aussi un yum install lshw remédiera, le cas échéant, à son absence.

Maintenance du système

La version de lshw est visualisable par :

lshw -version

Cet outil utilise la notion de classe pour filtrer son affichage.

Pour découvrir les différentes classes du matériel présent sur le système, il y a l'option -short.

lshw -short

Sans option, la commande lshw liste toutes les informations sur l'ensemble du matériel détecté.

lshw

L'option businfo, comme son nom le laisse penser, nous montre les bus présents sur le système.

lshw -businfo

Maintenance du système

Le système, les CPU

La classe "system" de lshw indique sur quelle plate-forme est installé le système. Ici sur une machine virtuelle sous VirtualBox.

lshw -c system

Concernant le ou les CPU proprement dit il y a plusieurs possibilités.

On peut consulter le fichier /proc/cpuinfo

cat /proc/cpuinfo

On peut utiliser la commande lscpu

lscpu

Maintenance du système

Ou encore la commande lshw

lshw -c processor

Maintenance du système

La mémoire vive (RAM) et la mémoire physique (les disques)

Les informations sur la mémoire se trouvent dans le fichier /proc/meminfo. Il est également possible d'obtenir d'autres informations avec la commande lshw -c memory

lshw -c memory

cat /proc/meminfo

Si vous soupçonnez des dysfonctionnements mémoire, il est possible de télécharger depuis le site https://www.memtest86.com/ l'utilitaire memtest86+. Il faut l'installer sur un CD ou une clé USB, et booter à partir de là. Le diagnostic s'effectue automatiquement. Si dans l'affichage il y a des lignes rouges, c'est qu'il y a un souci sur la barrette concernée.

https://www.memtest86.com/downloads/memtest86-usb.zip

Maintenance du système

Concernant la mémoire physique, ou les disques, il y a plusieurs options de lshw utilisables.

L'option storage donnera des informations sur les contrôleurs, alors que l'option disk ciblera bien sur les disques physiques, sans oublier volume pour une visibilité des volumes accueillant des systèmes de fichiers.

lshw -c storage

lshw -c disk

lshw -c volume

Maintenance du système

Les cartes réseaux

La commande lshw avec la classe network affiche des informations sur les interfaces réseau.

```
lshw -c network
```

Aujourd'hui vous pouvez aussi utiliser la commande ip pour obtenir des informations sur une carte réseau.

```
ip a
```

Stockage

Stockage

Le partitionnement des disques

Une partition est un espace disque contigu destiné à recevoir un système de fichiers.

La principale raison pour créer plusieurs partitions est la sécurité.

Si vous avez une partition corrompue, les autres restent saines.

Une autre raison de partitionner son système est de pouvoir mettre à jour une partie du serveur sans impacter une autre partie (Par exemple la zone des utilisateurs /home ne sera pas impacté par une mise à jour du système).

Stockage

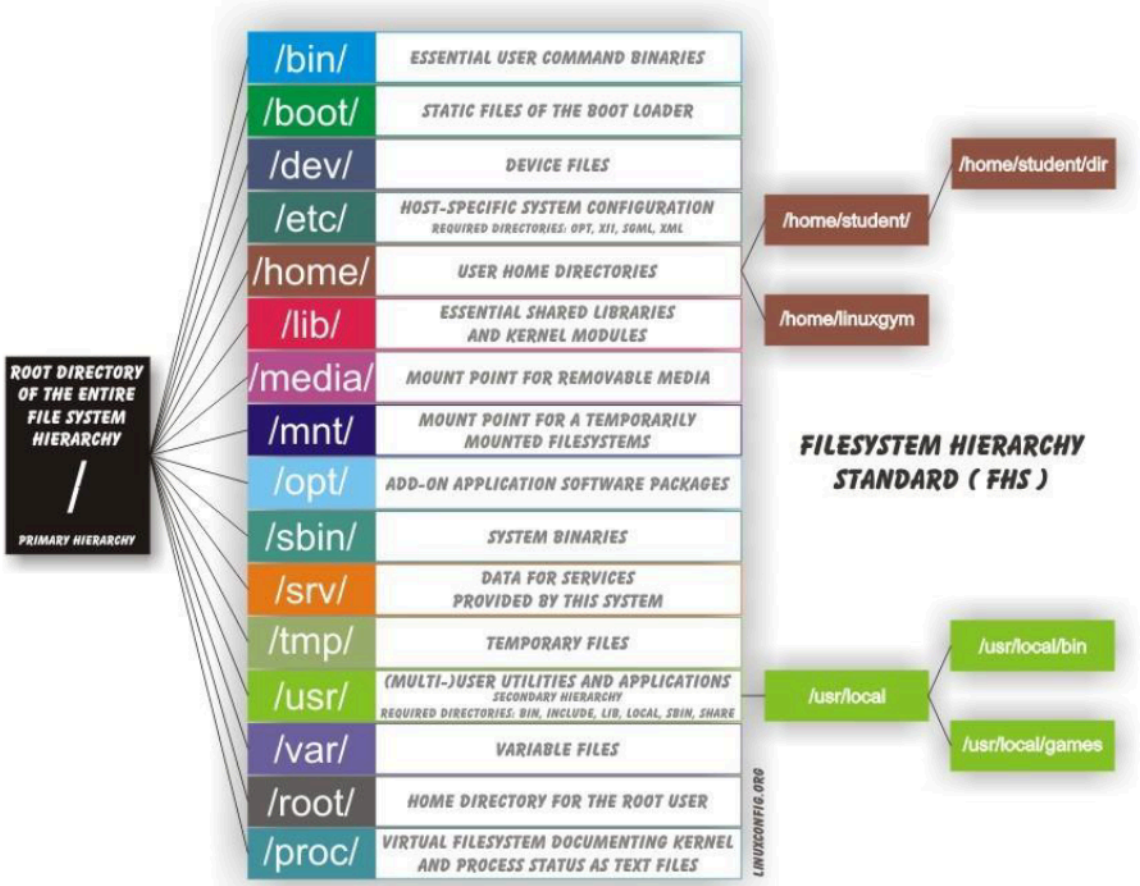
Rien n'empêche de mettre le système, les applications et les données dans une seule et unique partition montée en tant que /, mais ce ne sera pas très souple à maintenir.

De toute façon, il ne faut pas n'oublier la partition swap. En effet, pour installer Linux, deux partitions sont nécessaires au minimum :

- Une pour la racine /. Le système de base n'excède pas plus d'une dizaine de Go.

- Une pour le swap. Sa taille dépend de l'utilisation de l'ordinateur, de la quantité de RAM et de l'espace disque disponible dont vous disposez.

Stockage


ROOT DIRECTORY OF THE ENTIRE FILE SYSTEM HIERARCHY
/
PRIMARY HIERARCHY
/bin/
ESSENTIAL USER COMMAND BINARIES
/boot/
STATIC FILES OF THE BOOT LOADER
/dev/
DEVICE FILES
/etc/
HOST-SPECIFIC SYSTEM CONFIGURATION
REQUIRED DIRECTORIES: OPT, X11, SGML, XML
/home/
USER HOME DIRECTORIES
/lib/
ESSENTIAL SHARED LIBRARIES AND KERNEL MODULES
/media/
MOUNT POINT FOR REMOVABLE MEDIA
/mnt/
MOUNT POINT FOR A TEMPORARILY MOUNTED FILESYSTEMS
/opt/
ADD-ON APPLICATION SOFTWARE PACKAGES
/sbin/
SYSTEM BINARIES
/srv/
DATA FOR SERVICES PROVIDED BY THIS SYSTEM
/tmp/
TEMPORARY FILES
/usr/
(MULTI-)USER UTILITIES AND APPLICATIONS
SECONDARY HIERARCHY
REQUIRED DIRECTORIES: BIN, INCLUDE, LIB, LOCAL, SBIN, SHARE
/var/
VARIABLE FILES
/root/
HOME DIRECTORY FOR THE ROOT USER
/proc/
VIRTUAL FILESYSTEM DOCUMENTING KERNEL AND PROCESS STATUS AS TEXT FILES
LINUXCONFIG.ORG
/home/student/
/home/student/dir
/home/linuxgym
/usr/local
/usr/local/bin
/usr/local/games
FILESYSTEM HIERARCHY STANDARD (FHS)

Stockage

Les tailles de swap couramment recommandées sont :

Pour une mémoire inférieure à 1Go, prévoir un swap du double de la mémoire.

Pour une mémoire entre 1 et 4 Go, prévoir un minimum de 2 Go de swap.

Pour une mémoire entre 4 et 16 Go, prévoir un minimum de 4 Go de swap.

Pour une mémoire entre 16 et 64 Go, prévoir un minimum de 8 Go de swap.

Pour une mémoire entre 64 et 256 Go, prévoir un minimum de 16 Go de swap.

Pour une mémoire entre 256 et 512 Go, prévoir un minimum de 32 Go de swap.

Si vous utilisez une station de travail Linux sur un ordinateur portable, la taille du swap doit être au moins équivalente à celle de la RAM pour répondre au besoin de l'hibernation.

La façon de partitionner varie d'une machine à l'autre en fonction de son utilisation.

Stockage

Partitionnement d'un serveur (500 Go)

Un serveur applicatif a besoin de partitions :

/boot	500 Mo
/	100 Go
SWAP	4 Go
/tmp	2 Go
/var	50 Go
/opt	50 Go
/appli	le reste (ou /home) = 293.50 Go

Stockage

Les tables de partitionnement

Lorsque l'on crée une partition sous Linux, il faut indiquer le type de partition : aix, amiga, bsd, dvh, gpt, mac, msdos, pc98, sun, loop.

Stockage

Partition de type msdos:

La table de partition type msdos est le partitionnement traditionnel du PC. Elle contient un MBR (Master Boot Record), en français un secteur de démarrage, qui est le premier secteur du disque dur, soit le cylindre 0, tête 0 et secteur 1. Sa taille est de 512 octets, dans laquelle figurent la table de partition principale et le chargeur d'amorçage (bootloader).

440 octets sont utilisés par le programme d'amorçage, 6 octets pour la signature du MBR, 64 octets pour la table de partitionnement proprement dite (4x16 octets pour les 4 partitions primaires ou 3 primaires et une étendue) et les deux derniers pour la signature du boot MBR (0xAA55).

L'utilisation d'une partition étendue permettra de créer dix partitions logiques.

La taille des partitions est limitée à 2,2 Tio octets

Stockage

Partition type GPT:

Linux fonctionne également avec une table de partition de type GPT (GUID Partition Table) sur les platesformes : x86-64, IA-64 et x86.

Le noyau doit être compilé avec l'option CONFIG_EFI_PARTITION, ce qui est le cas dans la plupart des distributions.

Stockage

Avec l'UEFI (Unified Extensible Firmware Interface), le modèle de table de partition est donc de type GPT et offre notamment deux avantages :

- Le nombre maximal de partitions est par défaut augmenté à 128. Néanmoins, il suffisait d'augmenter la taille de la table de partition pour obtenir davantage de partitions.

- La taille des partitions peut aller jusqu'à 9,4 Zio.

Une table de partition GUID utilise des GUID (Globally Unique Identifier) ou pour Linux des UUID (Universal Unique Identifier) définis dans la RFC 4122 afin de déclarer de façon unique les partitions et leurs types.

Stockage

La commande blkid affiche les attributs des périphériques de bloc :

blkid

```
/dev/sda1: UUID="8d0fa428-55c4-42bc-a850-0611a2bc5028" TYPE="xfs"
/dev/sda2: UUID="Z30dSS-KRkY-KywU-vWW4-PrZT-eb62-zaLt58" TYPE="LVM2_member"
/dev/mapper/cl-root: UUID="3c906e12-5f24-46fc-a340-e4214f73cf11" TYPE="xfs"
/dev/mapper/cl-swap: UUID="bc89b099-59ea-4e17-ba29-c5aaca6258da" TYPE="swap"
```

Stockage

Conversion d'une partition type MBR vers GPT

La conversion peut se réaliser avec l'outil gdisk ou fdisk.

L'outil gdisk est similaire à fdisk mais prend en charge les deux types de partitions.

Il transforme toutes les partitions initiales en partitions GPT avec chacune un UUID.

gdisk n'est pas installé systématiquement sur toutes les distributions.

Stockage

Suppression une table de partition

Lorsque la table de partition est effacée, les données du disque sont détruites. Vous pouvez la supprimer en mettant à zéro le premier secteur du disque avec la commande dd.

```
dd if=/dev/zero of=/dev/sdb bs=512 count=1
```

Stockage

Quelques outils de partitionnement

fdisk, l'outil traditionnel en ligne de commande fourni dans toutes les distributions.

fdisk -l /dev/sda

cfdisk, outil similaire à fdisk mais en mode TUI...

GNU parted, permet de gérer les deux types de partitions, présent par défaut sur Red Hat, il faut l'installer sur Debian (apt-get install parted)

Gparted et QtParted

Si vous avez un environnement graphique comme GNOME 3, vous pouvez utiliser GParted, un front-end de GNU parted écrit en C++ et GTK+.

Destiné à l'environnement KDE, QtParted est aussi un front-end de GNU parted écrit en C++ et Qt.

Stockage

LiveCD...

Ultimate Boot CD

http://mirror.koddos.net/ubcd/ubcd539.iso

Gparted

https://downloads.sourceforge.net/gparted/gparted-live-1.3.1-1-amd64.iso

Stockage

Différents systèmes de fichiers

Quel système de fichier choisir ? Question ouverte où il n'est pas simple de répondre.

Cela dépend. Cela dépend en partie de l'usage que l'on veut faire de son système de fichiers, stockage de petits ou gros fichiers, quantité d'espace souhaité entre autres.

Voici un comparatif de quelques systèmes de fichiers pour vous donner des idées :

Système de fichiers	Taille max. d'un fichier	Taille max. du FS	Journalisé
ext2	2 Tio	4 Tio	non
ext3	2 Tio	4 Tio	oui
ext4	16 Tio	1 Eio	oui
reiserfs	8 Tio	16 Tio	oui
xfs	8 Eio	16 Eio	oui
zfs	16 Eio	16 Eio	oui
btrfs	16 Eio	16 Eio	oui

Stockage

FS	Fichier	FS	JFS	Droits	
ext2fs	**2 TiB**	**4 TiB**	**Non**	**Oui**	*Extended File System est le système de fichiers natif de Linux. En ses versions 1 et 2, on peut le considérer comme désuet, car il ne dispose pas de la journalisation. Ext2 peut tout de même s'avérer utile sur des disquettes 3½ et sur les autres périphériques dont l'espace de stockage est restreint, car aucun espace ne doit être réservé à un journal.*
ext3fs	**2 TiB**	**4 TiB**	**Oui**	**Oui**	*ext3 est essentiellement ext2 avec la gestion de la journalisation. Il est possible de passer une partition formatée en ext2 vers le système de fichiers ext3 (et vice versa) sans formatage.*
Ext4fs	**16 TiB**	**1 EiB**	**Oui**	**Oui**	*ext4 est le successeur du système de fichiers ext3. Il est cependant considéré par ses propres concepteurs comme une solution intermédiaire en attendant le vrai système de nouvelle génération que sera Btrfs*
ReiserFS	**8 TiB**	**16 TiB**	**Oui**	**Oui**	*Développé par Hans Reiser et la société Namesys, ReiserFS est reconnu particulièrement pour bien gérer les fichiers de moins de 4 ko. Un avantage du ReiserFS, par rapport à ext3, est qu'il ne nécessite pas une hiérarchisation aussi poussée: il s'avère intéressant pour le stockage de plusieurs fichiers temporaires provenant d'Internet. Par contre, ReiserFS n'est pas recommandé pour les ordinateurs portables, car le disque dur tourne en permanence, ce qui consomme beaucoup d'énergie.*
FAT	**2 GiB**	**2 GiB**	**Non**	**Non***	*Développé par Microsoft, ce système de fichiers se rencontre moins fréquemment aujourd'hui. Il reste néanmoins utilisé sur les disquettes 3½ formatées sous Windows et devrait être utilisé sous Linux si une disquette doit aussi être lue sous Windows. Il est aussi utilisé par plusieurs constructeurs comme système de fichiers pour cartes mémoires (memory sticks), car, bien documenté, ce système de fichiers reste le plus universellement utilisé et accessible.*
FAT32	**4 GiB**	**8 TiB**	**Non**	**Non***	*Ce système de fichiers, aussi créé par Microsoft, est une évolution de son prédécesseur. Depuis ses versions 2000 SP4 et XP, Windows ne peut pas formater (ou bloque volontairement le formatage) une partition en FAT32 d'une taille supérieure à 32 Go. Cette limitation ne s'applique pas sous Linux, de même qu'avec des versions antérieures de Windows. Une partition FAT32 d'une taille supérieure à 32 Go déjà formatée pourra être lue par Windows, peu importe sa version.*
NTFS	**Limité**	**2 TiB**	**Oui**	**Oui***	*Ce système de fichiers a aussi été développé par Microsoft, et il reste très peu documenté. L'écriture depuis Linux sur ce système de fichiers est stable à l'aide du pilote ntfs-3g. Ce pilote est inclus de base dans Ubuntu 7.10, et disponible en paquets dans les dépôts pour les versions antérieures.*

Stockage

Ext4 versus xfs

Un des changements majeurs de RHEL7/8 et CentOS7/8 est l'utilisation par défaut des systèmes de fichiers xfs au lieu de ext4 dans les versions précédentes.

Cela donne lieu à l'utilisation de commandes spécifiques à xfs.

A noter, que les systèmes de fichiers ext4 restent totalement supportés et peuvent être choisis lors de l'installation du système.

Migrer ses systèmes de fichiers d'ext4 vers xfs, bien que possible, n'est pas requis.

Stockage

Le tableau suivant montre quelques correspondances entres les commandes ext3/4 et xfs.

Action	ext3/4	xfs
Créer un système de fichiers	mkfs.ext3	mkfs.xfs
	mkfs.ext4	
Vérifier	e2fsck	xfs_repair
Changer la taille	resize2fs	xfs_growfs
Visualiser les caractéristiques	dumpe2fs	xfs_info
Ajuster les paramètres	tune2fs	xfs_admin

Stockage

Création d'un système de fichiers

La création d'un système de fichiers, ou le formatage du volume logique, se réalise à partir d'une commande universelle : mkfs.<type_FS>

```
mkfs.ext4 /dev/sdb1
```

Stockage

Montage des systèmes de fichiers

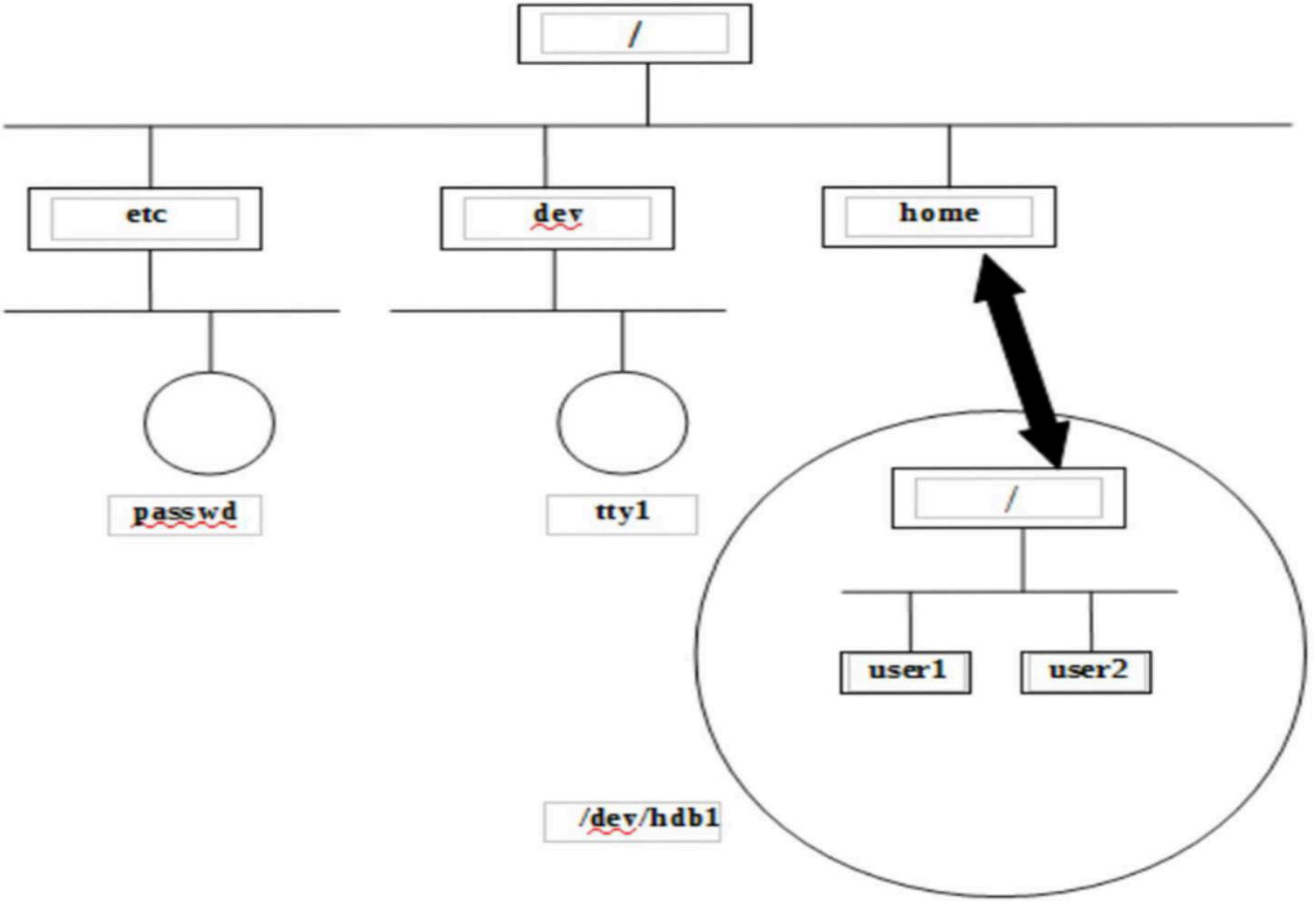
Tout fichier d'un système Linux est inséré dans une arborescence unique commençant par le répertoire racine "/" .

Ces fichiers peuvent résider sur n'importe quelle partition ou volume logique, à condition qu'il soit formaté avec un système de fichiers.

La commande mount permet d'attacher la racine d'un système de fichiers à un répertoire (créé au préalable) de l'arborescence du système.

A l'inverse, la commande umount défera le lien créé par la commande mount et détachera le système de fichier.

Stockage



Stockage

Gestion automatique des systèmes de fichiers, le fichier fstab

Le fichier /etc/fstab contient des informations sur les différents systèmes de fichiers.

Son rôle est de permettre un montage automatique des systèmes de fichiers lors de l'initialisation du système Linux.

Il sert aussi à faciliter la tâche de l'administrateur lors des montages manuels de systèmes de fichiers, ceci en le dispensant de donner à la commande mount toutes les options ou tous les arguments nécessaires.

Chaque système de fichiers est décrit sur une ligne indépendante.

Les champs contenus sur chaque ligne sont séparés par des espaces ou des tabulations.

L'ordre des lignes dans ce fichier a une importance car il conditionne l'exécution des commandes mount, umount, fsck.

exercice avec mount...

Stockage

Contrôle du FS

Les commandes e2fsck pour ext4 ou xfs_repair pour xfs permettent de vérifier la cohérence du système de fichiers.

umount /sdb1

e2fsck /dev/sdb1

Stockage

Gestion du swap

L'espace de SWAP est utilisé quand la mémoire (RAM) est pleine. Si le système a besoin de place mémoire mais qu'il ne reste plus de ressources mémoires, les pages mémoires inactives sont déplacées vers l'espace de SWAP.

Bien que le mécanisme de SWAP soit très utile sur des machines avec peu de RAM, il ne faut pas considérer ce mécanisme comme une alternative à plus de RAM. En effet, le SWAP étant sur disque, média beaucoup plus lent que la RAM, plus il sera sollicité moins le système sera performant.

L'espace de SWAP peut se trouver sur une partition disque (recommandé), un volume logique ou même un fichier, ou une combinaison de tout cela.

Stockage

Etat des lieux

On peut surveiller l'utilisation du SWAP à l'aide de la commande free ou en visualisant le contenu du fichier /proc/swap.

free

```
             total       used       free     shared    buffers     cached
Mem:       2046936    1832664     214272       4896          0    1315196
-/+ buffers/cache:     517468    1529468
Swap:      2113532       4872    2108660
```

cat /proc/swaps

Stockage

Les commandes de base : mkswap, swapon, swapoff

La commande mkswap crée une zone de SWAP dans une partition (type 82), un volume logique ou un fichier

mkswap fichier

"Fichier" pouvant être le fichier spécial d'une partition (ex : /dev/sda8), le fichier spécial d'un volume logique (ex : /dev/VolGroup00/LogVol02) ou un fichier (/swapfile).

Activation d'un SWAP : swapon -options fichier

Désactivation d'un SWAP : swapoff -options fichier

Avec l'option -a activation ou désactivation de tous les SWAPS déclarés dans /etc/fstab.

Stockage (TP)

LVM *(Logical Volume Manager, ou gestionnaire de volumes logiques en français)* permet la création et la gestion de volume logique sous Linux. L'utilisation de volumes logiques remplace en quelque sorte le partitionnement des disques.

C'est un système beaucoup plus souple, qui permet par exemple de diminuer la taille d'un système de fichier pour pouvoir en agrandir un autre, sans se préoccuper de leur emplacement sur le disque.

Avantages de LVM

Il n'y a pas de limitations « étranges » comme avec les partitions (primaire, étendue, etc.).

On ne se préoccupe plus de l'emplacement exact des données.

On peut conserver quelques giga-octets de libres pour pouvoir les ajouter n'importe où et n'importe quand.

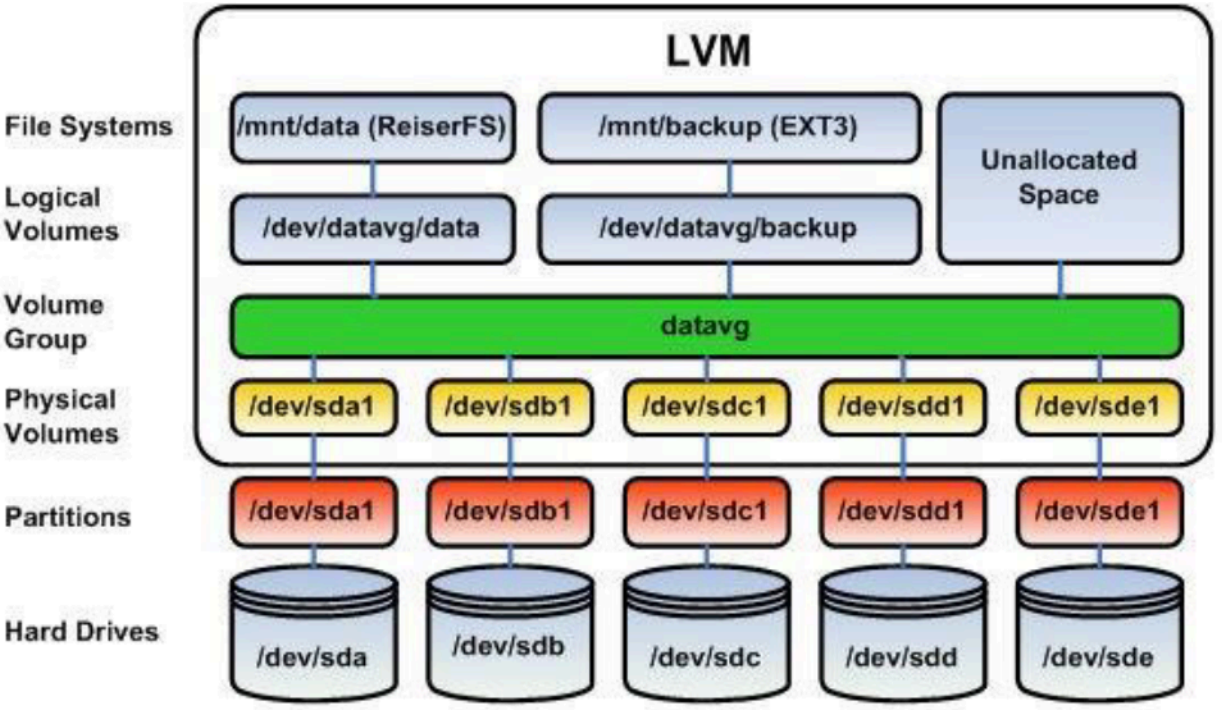
Les opérations de redimensionnement deviennent quasiment sans risques, contrairement au redimensionnement des partitions.

Inconvénients de LVM

Si un des volumes physiques devient HS, alors c'est l'ensemble des volumes logiques qui utilisent ce volume physique qui sont perdus. Pour éviter ce désastre, il faudra utiliser LVM sur des disques raid par exemple.

Il existe le paquet system-config-lvm dans les dépôts Universe, qui propose une interface graphique.

Stockage (TP)


LVM
File Systems
Logical Volumes
Volume Group
Physical Volumes
Partitions
Hard Drives
/mnt/data (ReiserFS)
/mnt/backup (EXT3)
Unallocated Space
/dev/datavg/data
/dev/datavg/backup
datavg
/dev/sda1
/dev/sdb1
/dev/sdc1
/dev/sdd1
/dev/sde1
/dev/sda1
/dev/sdb1
/dev/sdc1
/dev/sdd1
/dev/sde1
/dev/sda
/dev/sdb
/dev/sdc
/dev/sdd
/dev/sde

Stockage (TP)

LVM (apt-get install lvm2)

fdisk /dev/sd*

command (m for help): n

command (m for help): t (pour sélectionner le type de partition, 8e pour linux LVM)

command (m for help): w (pour sauvegarder la configuration)

pvcreate /dev/sdb* (initialiser la partition en tant que LVM)

vgcreate documents /dev/sdb* (création du volume groupe)

vgdisplay -v documents (affiche les informations du volume groupe)

lvcreate -L4000 -n travail documents (création d'un volume logique de 4 Go)

lvdisplay (affiche les informations du volume logique)

mkfs.ext4 /dev/documents/travail (formater le volume logique)

mkdir -p /documents/travail (création du point de montage)

mount /dev/documents/travail /documents/travail

Stockage (TP)

Agrandir un système de fichiers

umount /documents/travail

lvextend -L +1024M /dev/documents/travail

resize2fs /dev/documents/travail (faire un e2fsck puis à nouveau resize2fs)

mount /dev/documents/travail /documents/travail

Réduire un système de fichiers

lvreduce -L-1024M

ATTENTION CELA EST PEUT ETRE UN RISQUE !!!

Ajouter ou enlever une partition au VG

vgextend documents /dev/sdb*

vgreduce documents /dev/sdb*

Stockage (TP)

LVM - Les snapshots

A quoi ça sert ?

Historiquement, les snapshots ont été conçus pour permettre les sauvegardes des serveurs qui ne pouvaient pas être arrêtés pendant une longue période. Imaginons le cas suivant. Vous avez un serveur utilisé par des programmes qui font de nombreuses lectures/écritures à n'importe quelle heure du jour et la nuit. Sauvegarder tous les fichiers de ce serveur prend des heures. Mais comme il s'agit d'un serveur "critique", il n'est pas question de l'arrêter plus de quelques minutes par jours, grand maximum. Alors, comment faire les sauvegardes ?

Tout d'abord, j'espère que tout le monde comprend qu'il n'est pas question de sauvegarder un système de fichier en cours d'utilisation. Même si les lectures sont plus nombreuses que les écritures, le système de fichier est monté, il est en cours d'utilisation. Vous pouvez essayer de le sauvegarder si vous voulez, mais vous pouvez être certain que la restauration va très mal fonctionner.

Stockage (TP)

Alors on fait comment ?

Il faut faire un snapshot du système de fichier.

Un snapshot, c'est une photo.

On "fige" le système de fichier tel qu'il est à un instant T

Etapes...

Arrêt des programmes qui tournent sur le serveur

Démontage du système de fichier à sauvegarder

Snapshot de ce système de fichier (opération instantanée)

Montage du système de fichier

Relance des applications

Stockage (TP)

/etc/init.d/mon_application stop

umount /data

lvcreate -s -n mon_snapshot -L 20G /dev/monvg/monlv

mount /data

/etc/init.d/mon_application start

Toutes les données sont maintenant accessibles dans /data.

Vous devez vous demander a quoi correspond le "-L 20G" de la création du snapshot. En fait, un snaphost peut prendre de la place, car il doit stocker toutes les modifications qui sont faites sur les données depuis la réalisation du snapshot. Cette taille doit être choisie avec soin, il ne faut surtout pas prendre une valeur trop faible. Si vous pensez que pendant la durée de vie du snapshot, quelques centaines de Mo seront écrits dans /data, alors mettez au moins 1 ou 2 Go. Si vous pensez que 4 ou 5 Go seront écrits, mettez-en 10 ou 20... (voir plus si vous pouvez vous le permettre). Bref, soyez large. Il est toujours possible de vérifier le taux de remplissage du snapshot avec la commande : **lvs**

Stockage (TP)

Il ne reste plus maintenant qu'à lancer la sauvegarde du snapshot...

```
mount /dev/monvg/mon_snapshot /mnt
cp -a /mnt/* /backup
```

Une fois la sauvegarde terminée, on peut détruire le snapshot...

```
umount /mnt
lvremove /monvg/mon_snapshot
```

http://www.traduc.org/Guides_pratiques/Suivi/LVM-HOWTO/Document#Snapshots_Backup

Stockage

Les pseudos systèmes de fichiers.

Les pseudos systèmes de fichiers /proc et /sys ou plus exactement les points de montage des pseudos systèmes de fichiers procfs et sysfs sont appelés pseudo car ils n'ont pas de réelle représentation disque.

Leur taille visible est 0. Ils représentent de petites zones de la mémoire où sont stockées des informations dynamiques concernant le système.

Procfs le plus anciens, est plutôt destiné à ce qui concerne le noyau, les applications, le logiciel en général.

Sysfs apparu avec le noyau 2.6 est lui plus orienté vers le matériel et les pilotes de périphériques.

Stockage (/proc)

/proc

Le pseudo-système de fichiers procfs (process file system) est dynamiquement mis à jour par le noyau. Il est monté pour qu'il soit accessible sur le dossier /proc. Bien qu'il soit non vide, il affiche une taille de 0 Kio :

ls -dl /proc

En effet, le concept de taille de fichier n'a pas de sens pour procfs puisqu'il occupe uniquement unequantité limitée de mémoire vive.

Le dossier /proc est utilisé pour accéder aux informations du noyau qui est en cours d'exécution. Vous pouvez ainsi consulter en temps réel des informations sur le matériel, le système, le réseau, les processus, etc. Il est possible de changer la configuration du noyau en écrivant dans certains fichiers stockés dans /proc. Attention toutefois cette possibilité peut vite être dangereuse.

ls /proc

Les fichiers dont le nom est sous forme de nombre, sont des répertoires qui contiennent une multitude d'information liés au processus qui a le même PID que le nom du répertoire.

Stockage (/proc)

/proc

Pour les autres fichiers, ils contiennent des informations sur des éléments détectés par le système depuis le démarrage de ce dernier. Certains de ces fichiers ne sont pas directement visibles, mais sont rangés dans des sous répertoires.

Quelques exemples pour illustrer le propos :

cat partitions

cat swaps

cat version

...

Stockage (sysctl)

Modification de paramètres du noyau depuis /proc

En plus de la consultation de fichiers représentant les processus et des fichiers liés à la configuration du système, il est aussi possible de changer la configuration du noyau Linux en écrivant dans certains fichiers stockés dans /proc/sys.

Modification volatile d'un paramètre du noyau.

En tant qu'utilisateur root, vous pouvez utiliser un éditeur de texte, la commande echo ou bien sysctl pour effectuer la modification. Cela étant, au prochain démarrage, procfs sera régénéré et vous aurez donc perdu toutes vos modifications.

Stockage (sysctl)

Syntaxe

sysctl <option> <variable=valeur>

La variable désigne le fichier à modifier dans /proc et la valeur sera écrite dans ce dernier.

Options utiles :

Option courte	Option longue	Description
-w	--write	Modifie les paramètres sysctl.
-a	--all Af	fiche toutes les valeurs disponibles.
-p	--load	Charge les paramètres sysctl depuis le fichier par défaut /etc/sysctl.conf.

Vous pouvez spécifier un autre fichier :

--load=mon_fichier.conf

Stockage (sysctl)

Empêcher que votre ordinateur reçoive des pings sans créer de règles dans le pare-feu.

Pour rappel, le ping reçu est un message ICMP ECHO REQUEST (type 8) et votre ordinateur répond un message ICMP ECHO REPLY (type 0).

Le fichier /proc/sys/net/icmp_echo_ignore_all permet de configurer le noyau pour ignorer les messages ICMP ECHO entrants.

```
cd /proc/sys/net/ipv4
```

```
cat icmp_echo_ignore_all
```

La valeur 0 active et la valeur 1 désactive l'acceptation des "pings".

Remplacez 0 par 1 avec echo :

```
echo "1" > icmp_echo_ignore_all
```

Ou bien remplacez 0 par 1 avec sysctl :

```
sysctl -w net.ipv4.icmp_echo_ignore_all=1
```

Stockage (sysctl)

Modification permanente d'un paramètre du noyau.

Pour effectuer une modification persistante, il faut entrer les paramètres dans le fichier /etc/sysctl.conf.

Le fichier /etc/sysctl.conf contient les paramètres fournis par la distribution. Pour le personnaliser, il faut créer dans le répertoire /etc/sysctl.d un fichier de nom : "*.conf"

Les différents fichiers "*.conf" se trouvant dans le répertoire /etc/sysctl.d seront lancés au démarrage du système. Par défaut, il n'y a qu'un lien symbolique vers sysctl.conf.

cat /etc/sysctl.conf

Stockage (sysctl)

La création de ce fichier peut s'effectuer avec un éditeur de texte ou bien la commande echo.

Exemple de création de fichier pour que votre ordinateur ne reçoive plus des pings sans créer de règles dans le pare-feu.

```
echo "net.ipv4.icmp_echo_ignore_all = 1" >> /etc/sysctl.d/no-ping.conf
```

Redémarrez l'ordinateur, et affichez le contenu du fichier /proc/sys/net/ipv4/icmp_echo_ignore_all :

```
cat /proc/sys/net/ipv4/icmp_echo_ignore_all
```

Ou affichez le contenu avec sysctl :

```
sysctl net.ipv4.icmp_echo_ignore_all

net.ipv4.icmp_echo_ignore_all = 1
```

Stockage (/sys)

/sys

Introduit par le noyau Linux 2.6, le pseudo-système de fichiers sysfs est, comme procfs, un système de fichiers virtuel monté dans le dossier /sys. Il n'occupe donc pas d'espace disque et sa taille est de 0 Kio :

```
ls -ld /sys
```

Du fait que procfs contenait de nombreuses informations non liées aux processus, sysfs a été conçu pour exporter depuis l'espace noyau vers l'espace utilisateur des informations sur les périphériques et leurs pilotes.

Stockage (/sys)

/sys

/sys/fs, contient des informations sur les systèmes de fichiers montés comme ext4 et aussi sur la fonctionnalité du noyau cgroups (control groups) qui permet de limiter, compter et isoler l'utilisation de ressources telles que le processeur, la mémoire, etc.

Stockage (/sys)

/sys/kernel, est une arborescence qui contient des informations sur le noyau :

/sys/kernel/debug, est utilisé par le système de fichiers debugfs, lequel permet de déboguer le code du noyau Linux. Ne pas confondre ce système de fichiers debugfs avec l'utilitaire debugfs.

L'option debugfs doit être activée dans la configuration du noyau lors de la compilation.

Normalement, les distributions récentes intègrent cette option.

/sys/module, contient tous les modules du noyau (Loadable Kernel Modules) du système d'exploitation

Stockage (/sys)

/sys/power, ce dossier contient tous les fichiers qui fournissent une interface unifiée pour le sous-système gestionnaire de l'alimentation (power management)

/sys/power/state, contrôle l'état du niveau d'économie d'énergie utilisée par l'ordinateur. Les

valeurs possibles sont :

- standby (Power-On Suspend). Il correspond à la mise en veille de l'ordinateur.
- mem (Suspend-to-RAM). Il permet une suspension du système en mémoire.
- disk (Suspend-to-Disk). Il permet d'éteindre complètement l'ordinateur (c'est-à-dire de

faire une mise en veille prolongée) après avoir stocké son état dans la partition d'échange (swap). Il est recommandé pour cela que cette partition soit au moins égale à la RAM.

/sys/power/disk, contient les informations pour le mécanisme de la mise en veille prolongée.

/sys/power/image_size, gère la taille de l'image créée lors de la mise en veille prolongée.

Stockage (/sys)

L'utilitaire systool

La commande systool affiche les informations des périphériques du système par bus, par classe et par topologie. Cet outil doit être installé par le biais du paquet sysfsutils.

Installation de systool :

yum -y install sysfsutils

Stockage (/sys)

systool <option>

Les options sont :

Option courte	Description
-a	Montre les attributs de la ressource demandée.
-b	Affiche les informations pour un bus précis.
-c	Affiche les informations pour une classe de périphériques précise.
-d	Montre uniquement les périphériques.
-h	Affiche l'aide de la commande.
-m	Affiche les informations pour un module précis.
-p	Affiche le chemin absolu d'une ressource dans sysfs.
-v	Montre tous les attributs avec leurs valeurs.
-A	Affiche les attributs pour la ressource demandée.
-D	Montre uniquement les pilotes de périphériques.
-P	Affiche le périphérique parent.

Applications

Applications

Une application peut être sujette à différents types de dysfonctionnement répertoriés comme suit :

- l'application ne s’exécute pas
- l'application ne répond pas
- l'application a un fonctionnement dégradé
- l'application a un comportement inattendu

Applications

Application ne s'exécutant pas

Pour diagnostiquer une application qui échoue au démarrage, examinez les éléments suivants :

- la description du paquet DEB ou RPM qui est à l'origine de l'installation du programme. Dans le cas où le programme a été compilé par vos soins, consultez dans ce cas les paramètres de compilation
- les fichiers de configuration
- les journaux d'événements
- les filtrages réseau
- les dépendances nécessaires
- l'emplacement des bibliothèques
- l'exécution de l'application en mode débogage (debug)

Applications

Description d'un paquet installé et de ses dépendances.

La commande à utiliser varie d'une distribution à l'autre :

Sur Debian et Ubuntu, on affichera la description du paquet concerné avec la commande apt-cache show.

Les dépendances peuvent aussi être affichées avec apt-cache show, mais il peut être plus confortable d'utiliser la commande apt-cache depends.

Avec CentOS, yum info, affiche les informations du paquet concerné.

Exemple

```
yum info httpd
```

Applications

Vous pouvez également utiliser la commande rpm pour connaître les fichiers de configuration ou lire des informations d'un paquet.

Syntaxe de la commande rpm

rpm <options> <nom>

Option		Description
courte	longue	
-q		Option de requête
-c	--configfiles	Liste les fichiers de configuration
-i		Affiche les informations du paquet

Lister les fichiers de configuration de httpd (Apache2) :

rpm -qc httpd

Applications

Vous pouvez également utiliser la commande rpm pour connaître les fichiers de configuration ou lire des informations d'un paquet.

Obtenir des informations sur le paquet httpd :

```
rpm -qi httpd
```

Applications

Fichiers de configuration

Certains programmes proposent un utilitaire pour vérifier la syntaxe de leurs fichiers de configuration.

Par exemple, pour tester la configuration du programme httpd, l'utilitaire est apachectl. Sur Debian et Ubuntu l'option utile est -t et sur CentOS c'est configtest.

Voici un résultat sans erreurs :

```
apachectl configtest
```

Si erreur de syntaxe :

```
aapachectl configtest
```

```
AH00526: Syntax error on line 34 of /etc/httpd/conf/httpd.conf:
Invalid command 'ServeurRoot', perhaps misspelled or defined by a module
not included in the server configuration
```

Applications

Consulter les journaux

Habituellement, lorsqu'un programme échoue parce qu'il ne peut pas accéder à un fichier ou à une ressource, il écrit une entrée dans un journal système ou dans son propre journal.

Comme évoqué précédemment, la gestion des journaux système de Debian 9 et CentOS 7 est géré par le service systemd-journald qui transmet ensuite les évènements au service rsyslogd. Ceci permet une compatibilité avec les versions antérieures.

Seul le compte root et le groupe systemd-journal peuvent accéder le journal avec la commande journalctl.

```
journalctl

journalctl -u httpd

journalctl _PID=1

journalctl /bin/bash

systemctl status httpd
```

Applications

Visualiser les 10 derniers messages et attendre les suivants :

```
journalctl -f
```

Exemples de syntaxe pour visualiser les messages depuis et/ou jusqu'à une date précise :

```
journalctl --since=today
```

```
journalctl --since "2020-05-29 15:30:00"
```

```
journalctl --until="2020-05-29 15:30:00"
```

```
journalctl --since="2020-05-29 15:00:00" --until="2020-05-29 15:30:00"
```

Il est encore possible de filtrer les messages par niveau d'importance de l'évènement.

A savoir : "emerg" (0), "alert" (1), "crit" (2), "err" (3), "warning" (4), "notice" (5), "info" (6) et, "debug" (7).

```
journalctl -p err
```

Applications

Filtrages réseau

Il est important de vérifier si vous n'avez pas des règles de sécurité qui pourraient bloquer ou restreindre le fonctionnement du programme.

La commande netstat permet de visualiser les connexions réseau, notamment les ports TCP ou UDP qui sont en écoute et leurs statuts.

netstat <options>

Option courte	Option longue	Description
-t	--tcp	Affiche le protocole TCP.
-u	--udp	Affiche le protocole UDP.
-l	--listening	Affiche les sockets du serveur à l'écoute.
-p	--programs	Affiche le nom et le PID des processus propriétaires.
-n	--numeric	Affiche les adresses en format numérique.

Applications

Afficher le port d'écoute du programme httpd :

```
netstat -tulpn | grep httpd
```

L'écoute se fait sur le port TCP 80.

Afficher le port d'écoute du programme httpd :

```
ss -tulpn | grep httpd
```

Applications

La commande iptables, permet de regarder la configuration du pare-feu iptables.

Syntaxe : iptables <options>

Option	Description
-v	--verbose Mode verbeux (verbose).
-L	--list Liste les règles définies.

```
iptables -L
```

Applications

La commande firewall-cmd, permet de regarder la configuration du pare-feu firewalld.

Syntaxe : firewall-cmd <options>

Option	Description
- -list-port	Affiche les ports ouverts.
- -list-services	Affiche les services autorisés.
- -list-all	Affiche la configuration

Exemple

firewall-cmd –list-services

firewall-cmd --list-all

Applications

Dépendances du programme

Le programme peut faire appel à des fonctions stockées dans des bibliothèques. La commande ldd (List Dynamic Dependencies) affiche la liste des bibliothèques partagées qui sont nécessaires pour un programme ou une bibliothèque.

Syntaxe de la commande ldd: ldd <fichier>

Afficher les bibliothèques dépendantes pour le programme httpd :

ldd /usr/sbin/httpd

À son tour, une bibliothèque partagée peut avoir une dépendance. Et ainsi de suite... Voici l'exemple de libpcre.so.1 :

lldd /lib64/libpcre.so.1

ldd -v /lib64/libpcre.so.1

Applications

Emplacement des bibliothèques

Attention, si une bibliothèque n'est pas dans un emplacement standard, elle est signalée manquante par le chargeur.

Il faut ajouter son chemin à la variable d'environnement LD_LIBRARY_PATH pour qu'elle soit prise en compte :

```
export LD_LIBRARY_PATH=/home/bob/lib
```

Cette méthode est généralement utilisée pour la mise au point d'un programme.

Si vous souhaitez ajouter définitivement un emplacement de bibliothèques, il est préférable d'utiliser le fichier /etc/ld.so.conf :

```
cat /etc/ld.so.conf

include /etc/ld.so.conf.d/*.conf
```

Applications

Application ne répondant pas

Mode débogage

Si une application ne répond pas, elle peut attendre par exemple une connexion réseau. Vous pouvez rattacher la commande strace à ce processus grâce à l'option -p.

Identification du processus...

pidof httpd

Pidof : Cette commande affiche le ou les PID du programme invoqué.

On peut utiliser aussi top ou htop

ps -ef | grep http | grep -v grep

Applications

Utilisation de strace pour suivre le processus httpd à partir de son PID :

strace -p PID

strace systemctl restart httpd

Faites [CTRL]+C pour interrompre.

Applications

Terminer un processus

Il est possible de demander au système d'exploitation de terminer un processus ayant un comportement anormal ou qui est planté en invoquant son identifiant de processus (PID) ou bien son nom.

Rechercher un PID

Il est possible de rechercher le PID d'une application qui ne répond plus avec les commandes suivantes :

- pidof
- pgrep
- ps

Exemple

Connaître les PID du programme ssh :

pidof sshd

Applications

La commande pgrep affiche les identifiants des processus.

pgrep <options> <nom_du_programme>

Option courte	longue	Description
-l	--list-name	Affiche le processus avec son PID.
-u <utilisateur>	--euid <utilisateur>	Affiche les processus de l'UID mentionné.

Exemple

Connaître le PID du programme sshd :

```
pgrep -l sshd
```

Applications

Connaître le PID des programmes dont le propriétaire est stagiaire :

pgrep -lu stagiaire

2011 systemd

Applications

La commande ps permet d'afficher l'état des processus en cours.

ps <options>

Option courte	Description
-a	Affiche tous les processus.
-u	Affiche le nom de l'utilisateur et l'heure de lancement.
-x	Affiche également les processus qui n'ont pas de terminal de contrôle.

Exemple

Connaître uniquement le PID du programme sshd :

```
ps aux | grep sshd | grep -v grep
```

Applications

Outils pour terminer un processus

Si un programme ne répond plus du tout, il est donc nécessaire d'envoyer un signal au système d'exploitation pour demander de terminer son exécution. Pour effectuer cette opération, vous disposez de

plusieurs commandes :

- kill
- pkill
- killall
- xkill

Applications

kill permet d'envoyer différents types de signaux.

Syntaxe : kill <signal> <pid>

Les signaux utiles :

Signal	Description
-2 (SIGINT)	Interrompt le processus. [Ctrl]+C utilise ce signal.
-15 (SIGTERM)	Termine le processus.
-9 (SIGKILL)	Contrairement à SIGINT et SIGTERM, le signal ne pourra pas être ignoré par le processus.

Pour connaître la liste de tous les signaux disponibles :

```
kill -l
```

Applications

Pour terminer le programme httpd proprement, vous devez utiliser le signal SIGTERM :

```
kill -15 $(pidof httpd)
```

Il est à noter que la valeur par défaut est SIGTERM. Donc vous pouvez également taper ceci :

```
kill $(pidof httpd)
```

Si la demande SIGTERM n'aboutit pas, il est nécessaire d'envoyer le signal SIGKILL :

```
kill -9 $(pidof httpd)
```

Applications

pkill envoie également un signal à des processus en fonction de leur nom.

Syntaxe : pkill <options> <nom_du_programme>

Option courte	longue	Description
-l	--list-name	Affiche le processus avec son PID.
-u <utilisateur>	--euid <utilisateur>	Affiche les processus de l'UID mentionné.

-P --parent Sélectionne les processus du parent PID (PPID).

Exemple :

Listez le processus sshd pour constater s'il est en cours d'exécution :

```
pidof sshd
```

Applications

Terminer les processus associés à httpd :

```
pkill httpd
```

ou bien :

```
pkill --signal 15 vi
```

Le signal par défaut est SIGTERM (-15).

Vous n'avez pas à connaître le PID du processus, contrairement à la commande kill. Cela étant, si on est root, toutes les instances de vi sont terminées.

Vérifiez le résultat :

```
pidof httpd
```

Applications

killall termine un processus en invoquant son nom. Toutes les instances du processus invoqué sont terminées. Par défaut, la commande envoie le signal SIGTERM (-15).

Syntaxe : killall <options> <processus>

Option courte	longue	Description
-l	--list	Liste les signaux disponibles.
-s	--signal	Envoie un signal spécifique
-u	--user	Termine les processus de l'utilisateur mentionné.

Exemple

Listez le processus httpd pour savoir s'il est en cours d'exécution :

```
pidof httpd
```

Applications

Terminez avec le signal SIGKILL (-9) les processus associés à httpd :

killall -s 9 httpd

Vous n'avez pas à connaître le PID du processus, contrairement à la commande kill.

Vérifiez le résultat :

pidof httpd

Applications (TP)

Créer un package RPM

Installer les prérequis

yum install rpm-build rpmdevtools

cd /root

Créer la structure des répertoires...

rpmdev-setuptree

ls -l rpmbuild

Si erreur, le faire manuellement

mkdir -p ~/rpmbuild/{BUILD,RPMS,SOURCES,SPECS,SRPMS,tmp}

Applications (TP)

Créer un fichier rpmmacro

nano ~/.rpmmacros

ou

wget https://raw.githubusercontent.com/darkw0lf/centos/master/.rpmmacros

Applications (TP)

Récupérer les sources du programme

cd ~/rpmbuild/SOURCES

wget https://github.com/darkw0lf/purge_ram-1/archive/refs/heads/main.zip

mv purge_ram-1-main purge_ram-1

ls -l purge_ram1

La commande sync écrit sur le disque toutes les données dans les tampons en mémoire. Ceci peut inclure les superblocs modifiés, les inœuds modifiés, et les écritures différées. Ceci est implémenté dans le noyau…

La deuxième ligne efface le cache de la RAM, le BUFFER et libère le SWAP...

Applications (TP)

Compresser les sources du programme

```
tar czf purge_ram-1.0.tar.gz purge_ram-1.0
```

Applications (TP)

Créer un fichier SPEC

```
cd ~/rpmbuild/SPECS/
wget https://raw.githubusercontent.com/darkw0lf/centos/master/purge_ram.spec
```

Applications (TP)

Créer le build...

```
cd ~/rpmbuild
rpmbuild -ba SPECS/purge_ram.spec
cd ~/rpmbuild/RPMS/noarch
ls
purge_ram-1-0.rpm
```

installer le nouveau package

```
yum -ivh purge_ram-1-0.rpm
cd /opt
ls
```

Applications (mirroir)

Créer un miroir local...

yum install epel-release

yum install nginx

Démarrer les services

systemctl start nginx

systemctl enable nginx

systemctl status nginx

Applications (mirroir)

Ajouter au firewall

```
firewall-cmd --zone=public --permanent --add-service=http
```

```
firewall-cmd --zone=public --permanent --add-service=https
```

```
firewall-cmd --reload
```

Tester l'URL

http://IP

Applications (mirroir)

Créer le miroir local

yum install createrepo yum-utils

Créer les répertoires

mkdir -p /var/www/html/repos/{base,centosplus,extras,updates}

Synchroniser avec ceux de CentOS

reposync -g -l -d -m --repoid=base --newest-only --download-metadata
--download_path=/var/www/html/repos/

reposync -g -l -d -m --repoid=centosplus --newest-only --download-metadata
--download_path=/var/www/html/repos/

reposync -g -l -d -m --repoid=extras --newest-only --download-metadata
--download_path=/var/www/html/repos/

reposync -g -l -d -m --repoid=updates --newest-only --download-metadata
–download_path=/var/www/html/repos/

Faire un CTRL C

Applications (mirroir)

Mettre à jour les entêtes

```
createrepo -g comps.xml /var/www/html/repos/base/

createrepo -g comps.xml /var/www/html/repos/centosplus/

createrepo -g comps.xml /var/www/html/repos/extras/

createrepo -g comps.xml /var/www/html/repos/updates/
```

Applications (mirroir)

Créer un serveur pour Nginx

vim /etc/nginx/conf.d/repos.conf

```
server {
        listen   80;
        server_name  repos.test.lab;        #change  test.lab to your real domain
        root   /var/www/html/repos;
        location / {
                index  index.php index.html index.htm;
                autoindex on;     #enable listing of directory index
        }
}
```

Redémarrer le service nginx

systemctl restart nginx (tester l'URL)

Applications (mirroir)

Configurer la crontab

vim /etc/cron.daily/update-localrepos

```
#!/bin/bash
##specify all local repositories in a single variable
LOCAL_REPOS="base centosplus extras updates"
##a loop to update repos one at a time
for REPO in ${LOCAL_REPOS}; do
reposync -g -l -d -m --repoid=$REPO --newest-only --download-metadata --download_path=/var/www/html/repos/
createrepo -g comps.xml /var/www/html/repos/$REPO/
done
```

Donner les droits

chmod 755 /etc/cron.daily/update-localrepos

Applications (mirroir)

Configurer le client

vim /etc/yum.repos.d/local-repos.repo

Applications (mirroir)

[local-base]

name=CentOS Base

baseurl=http://repos.test.lab/base/

gpgcheck=0

enabled=1

[local-centosplus]

name=CentOS CentOSPlus

baseurl=http://repos.test.lab/centosplus/

gpgcheck=0

enabled=1

Applications (mirroir)

[local-extras]

name=CentOS Extras

baseurl=http://repos.test.lab/extras/

gpgcheck=0

enabled=1

[local-updates]

name=CentOS Updates

baseurl=http://repos.test.lab/updates/

gpgcheck=0

enabled=1

Applications (mirroir)

Tester le client

yum repolist

ou

yum repolist all

Optimisations

Optimisations

Le suivi des performances du système, ainsi que l'usage des ressources, sont des tâches quotidiennes de l'administrateur. Un suivi régulier du système permettra d'être dans les meilleurs conditions pour effectuer l'analyse et comprendre le problème rencontré. La lecture des journaux tels que /var/log/syslog et /var/log/messages est un bon début afin de noter les entrées inhabituelles. Vérifiez également que les logiciels sont bien mis à jour.

Configurer finement le système d'exploitation ne résout pas tous les problèmes. Vous pouvez aussi rencontrer des applications mal codées ou un matériel défectueux. Pour mener à bien votre investigation, il faut :

- identifier le goulot d'étranglement
- identifier la cause
- définir les solutions potentielles
- implémenter une solution
- évaluer la solution

Optimisations

Point de contention ou goulot d'étrangement

Un point de contention ou goulot d’étranglement est un constituant du système qui limite les performances globales. Il peut parfois en cacher un autre.

Les serveurs doivent être efficaces et fiables en exploitation. Pour optimiser leurs performances, vous devez collecter des données qui permettent d’identifier les goulots d’étranglement du système et créer une base de référence.

Pour concevoir une base de référence, vous devez :

- identifier les ressources

- capturer des données

- stocker ces données

Optimisations

Identifier les ressources matérielles

Il s'agit d'élaborer un inventaire matériel et logiciel du système. Les commandes lshw, lspci, lsscsci, lsusb entre autres, pourront vous aider. Ces dernières ont été évoquées dans les chapitres précédents.

Matériel	**Commandes suggérées**
Nombre de CPU	cat /proc/cpuinfo, lshw -C cpu
Caractéristiques du CPU	cat /proc/cpuinfo, lshw -C cpu
Mémoire RAM	cat /proc/meminfo, free -t
Stockage	lshw -C disk, lsblk, df
Interfaces réseau	lshw -C network

Optimisations

Identifier les ressources logicielles

Système	**Commandes suggérées**
Distribution Linux et version	cat /etc/*release*, lsb_release -a
Noyau	uname -r
Liste des modules chargés	lsmod
Partitionnement	cat /proc/partitions, lsblk, fdisk -l
Version du shell	echo $BASH_VERSION
Liste des processus	ps -ef

Configuration système	Fichiers dans /etc
Configuration réseau	Fichiers dans /etc
Configuration des applications	Variable selon l'application

Optimisations

La surveillance d'un serveur peut s'effectuer de deux façons

De façon instantanée

Vous disposez de deux sortes de commandes :

Celles qui capturent dans l'instant les valeurs et qui vous les affichent. Celles-ci n'évoluent pas. Il faut relancer la commande pour les réactualiser.

Celles qui capturent dans l'instant les valeurs et qui les réactualisent selon un intervalle de temps.

Optimisations

La surveillance d'un serveur peut s'effectuer de deux façons

Sur la durée

Relevez des échantillons de valeurs des ressources du système tous les quarts d'heure ou toutes les heures.

Stocker ces informations, elles vous serviront de référence pour pouvoir comparer l'activité du système

quand il fonctionnait normalement, et maintenant qu'il y a un souci. Sans ce référentiel il sera plus difficile

de voir ce qui a évolué sur le système et isoler le composant en cause. La commande sar est prévu à cet

effet avec ses fonctions associées, sa1 et sa2.

Côté commandes, il y a des commandes généralistes, qui remontent des informations sur plusieurs

secteurs, top, vmstat, sar par exemple et des commandes spécialisées qui ne remonteront d'informations

que sur un type de ressources free, iostat ...

Optimisations

Le CPU

La majeure partie des activités en cours d'exécution sur un ordinateur font appel au(x) processeur(s).

Un serveur peut avoir un ou plusieurs rôles :

- serveur d'authentification (NIS, LDAP...),
- serveur de services réseau (DHCP, DNS...),
- serveur d'application (PostgreSQL, MariaDB...),
- serveur de fichiers (Samba, NFS...) ou d'impressions.

Optimisations

Le CPU

L'activité normale d'un processeur dépend du rôle qu'on lui attribue. Un serveur d'application fait appel davantage au processeur et à la mémoire vive pour les traitements de données alors qu'un serveur de fichiers utilise plutôt en premier lieu une interface réseau et des accès aux disques.

Si vous combinez plusieurs rôles, vous risquez de voir les quatre ressources critiques sollicitées de façon importante.

Les principales causes d'engorgement du processeur sont les programmes qui prennent trop de temps de traitement et des ressources disques ou réseau qui génèrent des interruptions excessives.

Optimisations

La commande top

La commande top permet de visualiser, de façon synthétique, l'activité de l'ordinateur et, en ce qui nous concerne pour le moment, celle du processeur.

Syntaxe

top <options>

Option	**Description**
-d <secondes>	Spécifie le délai de rafraîchissement. La valeur par défaut est de 3 secondes.
-u <nom_utilisateur>	Affiche les processus de l'utilisateur mentionné.

Optimisations

Afficher la mémoire du système

Pour visualiser la mémoire disponible utilisez la commande free.

free <options>

Option courte	Option longue	Description
-b	--bytes	Affiche la valeur en octets.
-k	--kilo	Affiche la valeur en Ko
-m	--mega	Affiche la valeur en Mo
-g	--giga	Affiche la valeur en Go
--	--tera	Affiche la valeur en To.
-h		Affiche selon l'unité la plus appropriée.
-t	--total	Affiche le total des colonnes.

Optimisations

La commande vmstat

La commande vmstat peut apporter également des statistiques concernant la mémoire.

Syntaxe : vmstat <option>

Option		**Description**
courte	longue	
-s	--stats	aAffiche des statistiques de la mémoire.

Optimisations

Le stockage (Disques)

La surveillance des disques permet de déterminer la présence de goulots d'étranglement :

- le besoin de disques supplémentaires ou plus rapides,
- un excès de pagination.

Optimisations

La commande iostat, faisant aussi partie du paquet sysstat, rapporte des statistiques d'E/S pour les périphériques et les partitions.

Syntaxe

iostat <options> <périphérique> <intervalle> <quantité>

Option	**Description**
-p	Stipule le périphérique de bloc.

Voici l'analyse du périphérique /dev/sda. L'affichage comportera 3 rapports actualisés toutes les 5 secondes :

```
iostat -p sda 5 3
```

DHCP

Serveur DHCP (TP)

Intallation et configuration d'un serveur DHCP

Le protocole DHCP Dynamic Host Configuration Protocol est un service réseau TCP/IP.

Il est possible de manière statique d'attribuer une adresse IP à chaque machine. Ce qui veut dire qu'il faudrait rentrer manuellement l'adresse IP de 500 machines si votre parc informatique en possède autant. Le DHCP (Dynamic Host Configuration Protocol) permet d'attribuer de manière automatique une adresse IP à chaque machine connectée sur un réseau LAN.

L'avantage est qu'il simplifie l'administration du réseau par une affectation simple et une mise à jour des adresses IP. Il permet aussi de les économiser en attribuant les adresses IP aux machines connectées uniquement.

C'est très pratique si le réseau est constitué d'un nombre important d'ordinateurs portables.

Serveur DHCP (TP)

vi /etc/sysconfig/network-scripts/ifcfg-enp0s3

Modifier la partie suivante :

BOOTPROTO=static

BROADCAST=192.168.1.255

IPADDR=192.168.1.60

NETMASK=255.255.255.0

NETWORK=192.168.1.0

/etc/init.d/network restart

Serveur DHCP (TP)

nmcli -d

nmtui

configurer en manuel...

/etc/init.d/network restart

Serveur DHCP (TP)

Enfin installer le service DHCP sur votre serveur

apt-get install dhcp

cd /etc/dhcp

vi dhcpd.conf

Décommenter la ligne :

authoritative;

Serveur DHCP (TP)

```
# A slightly different configuration for an internal subnet.
# subnet 10.5.5.0 netmask 255.255.255.224 {
# range 10.5.5.26 10.5.5.30;
# option domain-name-servers ns1.internal.example.org;
# option domain-name "internal.example.org";
# option routers 10.5.5.1;
# option broadcast-address 10.5.5.31;
# default-lease-time 600;
# max-lease-time 7200;
#}
```

Serveur DHCP (TP)

Le « subnet » est notre adresse de réseau et le « netmask » notre masque en classe C.

subnet 10.0.2.0 netmask 255.255.255.0 {

C'est la plage IP que nous souhaitons utiliser, c'est-à-dire que la distribution des adresses IP débuteront de « 10.0.2.10 » et termineront en « 10.0.2.50 ». Ce qui veut dire que nous pouvons adresser en dynamique 40 machines différentes et connectées en même temps.

range 10.0.2.10 10.0.2.50;

Serveur DHCP (TP)

Cette partie reste commenter avec le dièse mais nous l'utiliserons plus tard, ne vous inquiétez pas. Elle servira pour le service DNS.

```
# option domain-name-servers ns1.internal.example.org;

# option domain-name "internal.example.org";
```

« option routers » a pour adresse celle de votre passerelle par défaut pour sortir.

```
option routers 10.10.100.254;
```

Serveur DHCP (TP)

L'adresse de Broadcast c'est une adresse de diffusion. Pour faire très court, c'est l'adresse sur laquelle votre machine va émettre sur tout le réseau pour réclamer par exemple une adresse IP au serveur DHCP qui de son côté écoutera sur l'adresse de Broadcast.

option broadcast-address 10.10.100.255;

Ces deux lignes définissent en seconde la durée des baux remis à vos machines. Si vous souhaitez que ceux-ci durent plus longtemps vous n'avez qu'à en modifier les valeurs. Sachez tout de même que lorsqu'une machine est déconnectée l'adresse IP lui appartenant le restera encore pendant les 2/3 du temps du bail. Par exemple si votre bail dure 10 jours, l'adresse sera réservée pendant 7 jours au delà de cette période elle pourra être remise à une autre machine.

default-lease-time 600;

max-lease-time 7200;

Serveur DHCP (TP)

Redémarrer le service

systemctl restart dhcpd

Avec une machine sous Ubuntu avec la commande suivante

sudo dhclient eth0

Et sous Windows 10, taper dans le CMD...

ipconfig /release

ipconfig /renew

ipconfig /all

Annexes

Gestion des disques, volumes et systèmes de fichiers

- Présentation des différents systèmes de fichiers
- Gestion des partitions traditionnelles et des volumes logiques
- Formatage
- Montage et démontage des systèmes de fichiers
- Gestion des quotas
- Gestion du RAID

Travaux pratiques

- Création d'un volume logique, formatage et montage permanent dans fstab ou avec systemd
- Gestion des droits standards, SUID, SGID
- Listes de Contrôle d'Accès (ACL)
- Droits classiques
- Création d'un répertoire collaboratif

Gestion des daemons et services

- Description du processus de démarrage
- Définition des runlevels et des services
- Le processus d'arrêt du système
- Gestionnaires de GRUB2 boot
- Gestion des unités service et cible (target) de systemd
- Gestion des services SysVinit
- Dépannage en mode rescue ou emergency

Gestion des processus

- Définition des processus, des threads et de l'ordonnancement
- Analyse de l'activité système (top, pstree, ps...)
- Gestion des signaux (kill, pkill, killall, nohup...)
- Tâches avant et arrière plans (jobs, bg, fg, SIGTSTP...)
- Planification de tâches (at et cron)

Gestion du kernel

- Architecture et définition
- Les modules
- Commandes pour manipuler le noyau
- Compiler et déployer un nouveau noyau
- Mettre à jour et débuguer un noyau
- Déploiement de drivers spécifiques

Sauvegarde et restauration

- Compression et décompression d'un fichier (gzip, bzip2, lzma, lzw)
- Gestion d'une archive avec ou sans compression
- Mettre en oeuvre des packages de synchronisation

Mise en réseau

- Architectures réseaux typiques
- Configuration des réseaux
- Mettre en oeuvre un serveur DNS, DHCP, SSH, Nginx ou Caddy...
- Analyser les flux réseaux

Mettre en oeuvre la sécurité

- Vue d'ensemble des bonnes pratiques de sécurité
- Introduction

Gestion du service de temps

- **Réglages de la date et de l'heure**
- **Paramétrer le client NTP**

Les outils

- **Vue d'ensemble des outils tiers pour une meilleure administration**

Merci pour votre attention

Des questions ?