Linux

Administration avancée 101

ESGI
école supérieure de génie informatique

Pour des demandes de formations, contactez-moi :
https://pierreau.fr/Contact/index.php

Bonne lecture…

Pierre ROYER Manager | Architecte | Formateur #numérique

INDEX

PREAMBULE

A. Ce document

Informations

Nom du document	Linux.docx	Référence	UNIX-ADM
Version	2020.02.22	Pages	48
Date de création	07/01/1996	Dernière modification :	13/01/2023
Auteur :	Pierre ROYER Tél : (+33) 614 672 909 https://www.linkedin.com/in/pierreau	Contributeur(s) :	
Mode de diffusion	☐ confidentiel ☐ restreint ☐ interne ☒ libre	Liste de diffusion	https://pierreau.fr
Annexes :	LIENS		

B. Conventions

Les syntaxes utilisées dans ce document :

[root@RockyLinux ~]# représente un prompt bash en root sur un serveur RockyLinux
root@Debian:~ # représente un prompt bash en root sur un serveur Debian
[pierreau@RockyLinux ~]$ désigne un compte utilisateur local

Le contenu d'un fichier est encadré, les commandes sont en gras :

```
[RockyLinux@localhost ~]# vi /etc/ssh/sshd_config
PermitRootLogin yes
```

Les caractères en italique sont des exemples de paramètres :

```
192.168.100.100 ServeurA
192.168.100.101 ServeurB
```

Information utile	Attention particulière	Risque important
		

HISTORIQUE

UNIX est un système d'exploitation multitâches et multi-utilisateurs créé en 1969. Il a donné naissance à une famille de systèmes, dont les plus populaires actuellement sont Linux et Mac OS X. Cette famille est définie par la norme POSIX (Portable Operating System Interface, dont le X exprime l'héritage UNIX). POSIX dépend du standard IEEE 1003 (Institute of Electrical and Electronics Engineers).

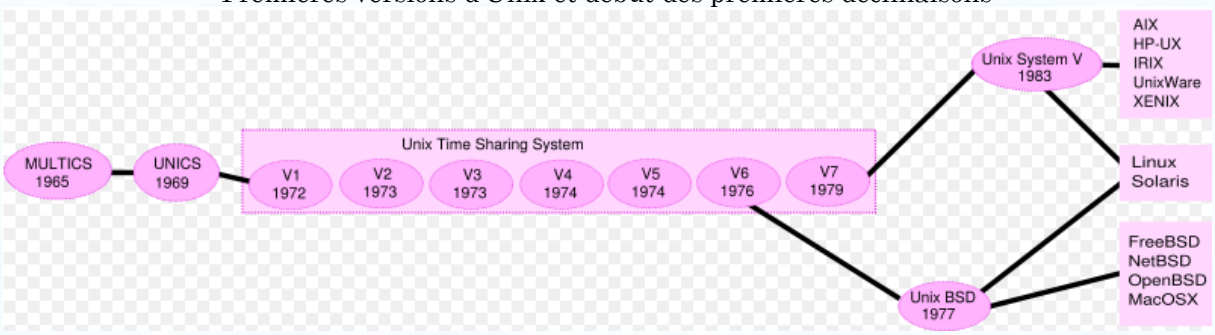
Originellement écrit en langage machine, le système Unix évolua grâce à une compilation en langage C, dont Ken Thompson et Dennis Ritchie furent les fondateurs des bases de l'actuel système.

Dès la fin de l'année 1977, des chercheurs de l'Université de Californie apportèrent de nombreuses améliorations au système UNIX fourni par AT&T et le distribuèrent sous le nom de Berkeley Software Distribution (ou BSD).

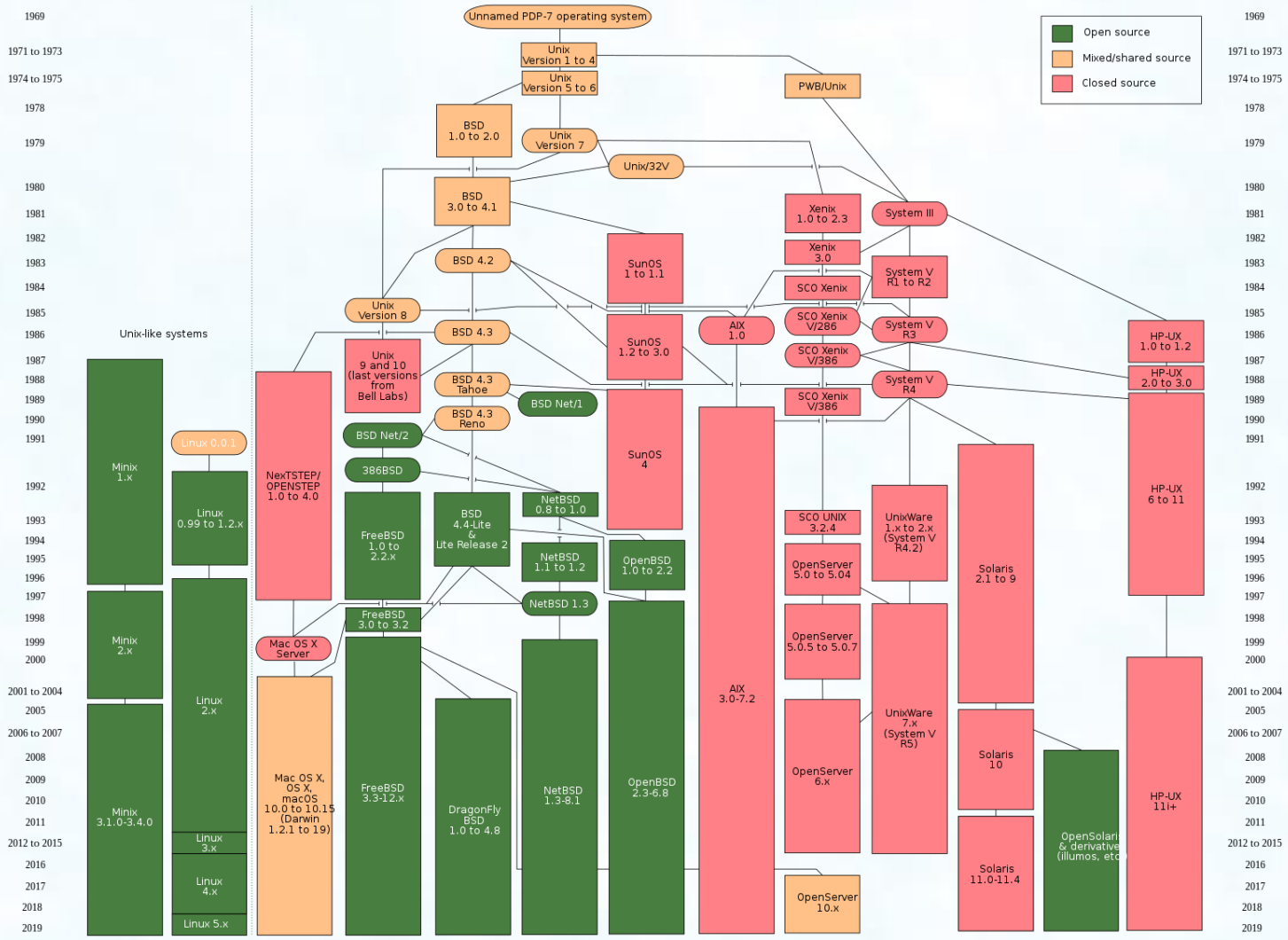
En 1977, AT&T mit les sources d'UNIX à la disposition d'autres entreprises ; des dérivés d'UNIX furent alors développés :

- **AIX**, développé par IBM, dont la première version de 1986 fut basée sur System V release 2.
- **Solaris**, développé par Sun Microsystems, basé au départ sur BSD 4.1c en 1981, puis sur System V release 4 (SVR4).
- **HP-UX**, fondé sur System V, développé à partir de 1986 par Hewlett-Packard

Premières versions d'Unix et début des premières déclinaisons



Les grandes familles d'Unix



En 1985, un professeur américain domicilié aux Pays-Bas, Andrew S. Tanenbaum, développa un système d'exploitation minimal, baptisé **Minix**, afin d'enseigner les concepts des systèmes d'exploitation à ses étudiants. En 1991 un étudiant finlandais, **Linus Torvalds**, décida de concevoir, sur le modèle de Minix, un système d'exploitation (Linux) capable de fonctionner sur les architectures à base de processeur **Intel 80386**.

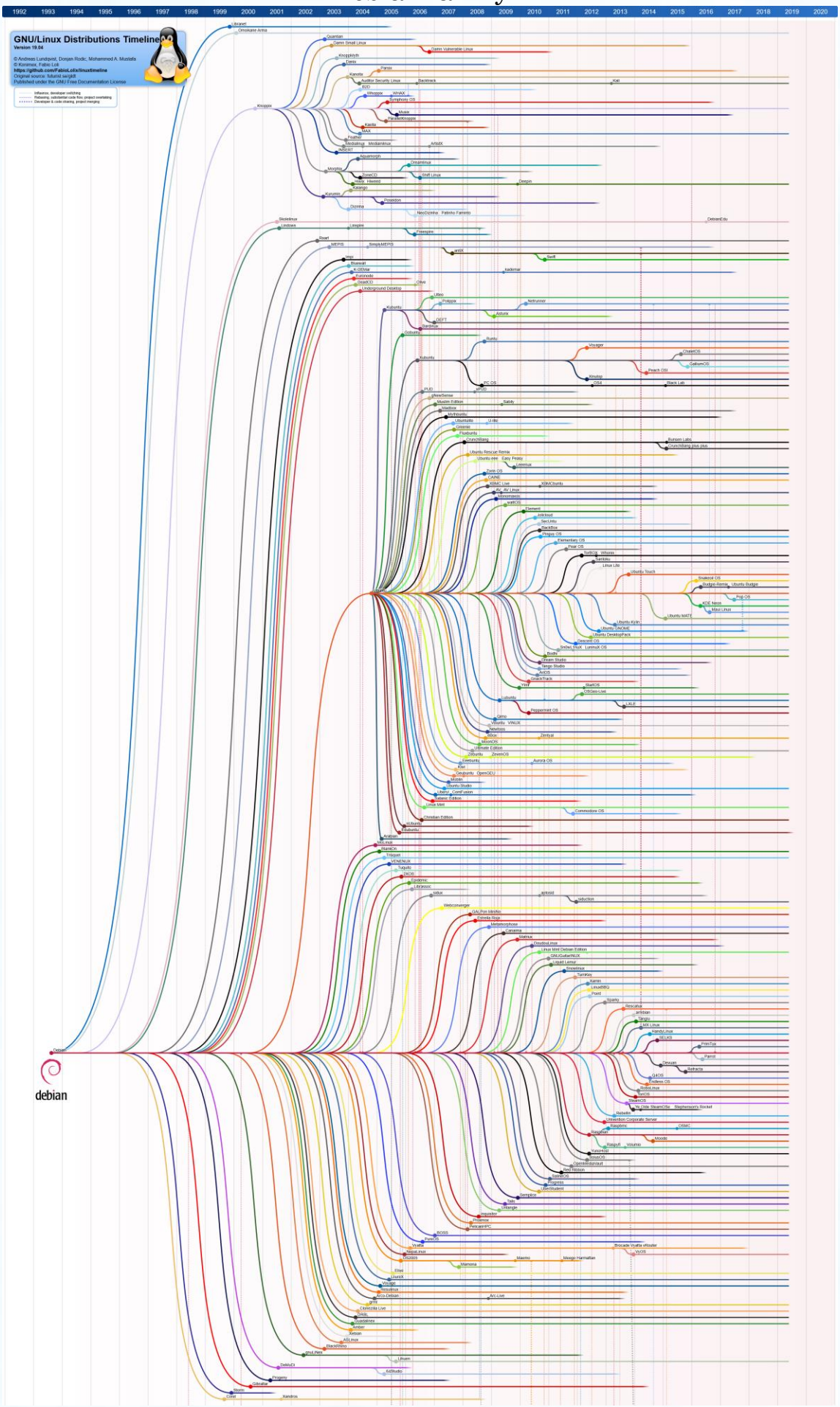
Le nom de « Linux » provient de la personne qui gérait le serveur FTP, hébergeant le projet initial (version 0.0.1, diffusée en 1991). La version 1.0.0 sort en 1994, avec 176 250 lignes de code. **Linux ne contient pas de code provenant d'UNIX**, mais c'est un système inspiré d'UNIX et complètement réécrit. D'autre part, Linux est un logiciel libre. Linux n'étant qu'un noyau, il utilise l'ensemble des logiciels du projet GNU pour faire un système d'exploitation complet.

Quelques distributions libres :

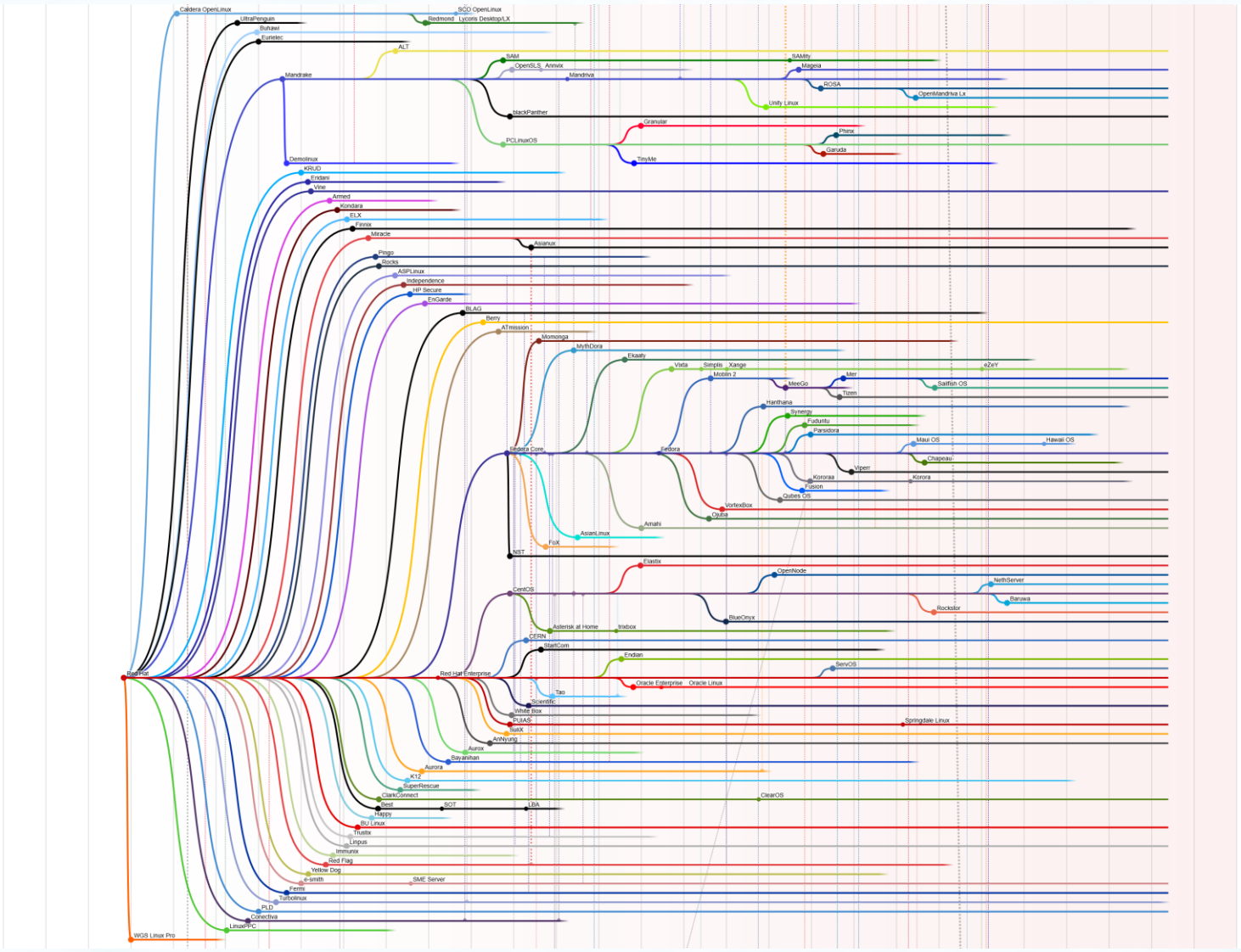
- Slackware : robuste, très « Unix & BSD like », la dépendance des packages est inexistante.
- Debian : installation minimaliste, basées sur la stabilité et l'efficacité, près de 20 000 packages !
- Red-Hat/RockyLinux : installations simples, orienté serveurs ; Fedora : orienté PC).
- Novell SuSE : certifiée pour Oracle, SAP, PeopleSoft ; alliant un bon compromis entre utilisation serveur et poste de travail.
- *BSD : véritables systèmes Unix, axés sur la sécurité.
- LinuxFromScratch : après la lecture de ce document, vous pouvez vous y aventurer…

Le schéma suivant représente l'arbre des très nombreuses distributions de Linux existantes, toutes originaires soit de Slackware, Debian ou Red-Hat…

Debian family

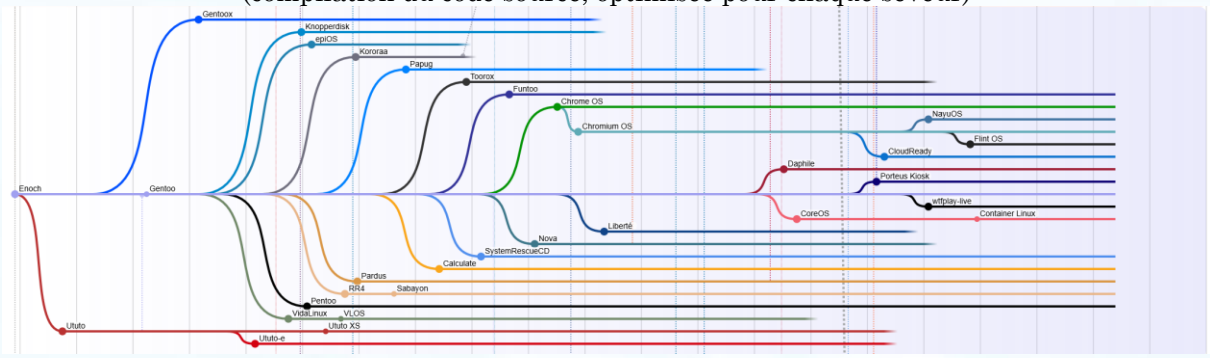


Branche RedHat



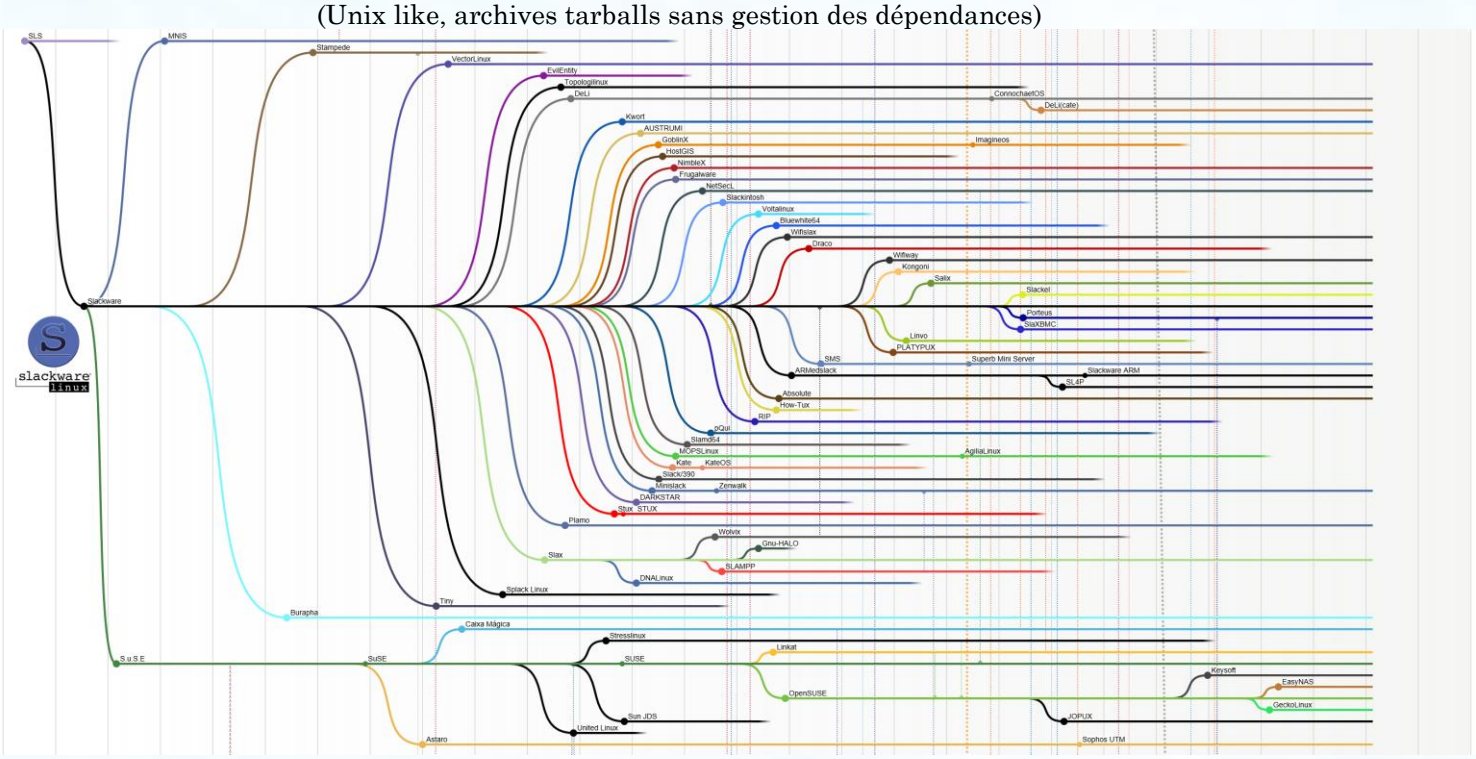
Branche Gentoo

(compilation du code source, optimisée pour chaque seveur)

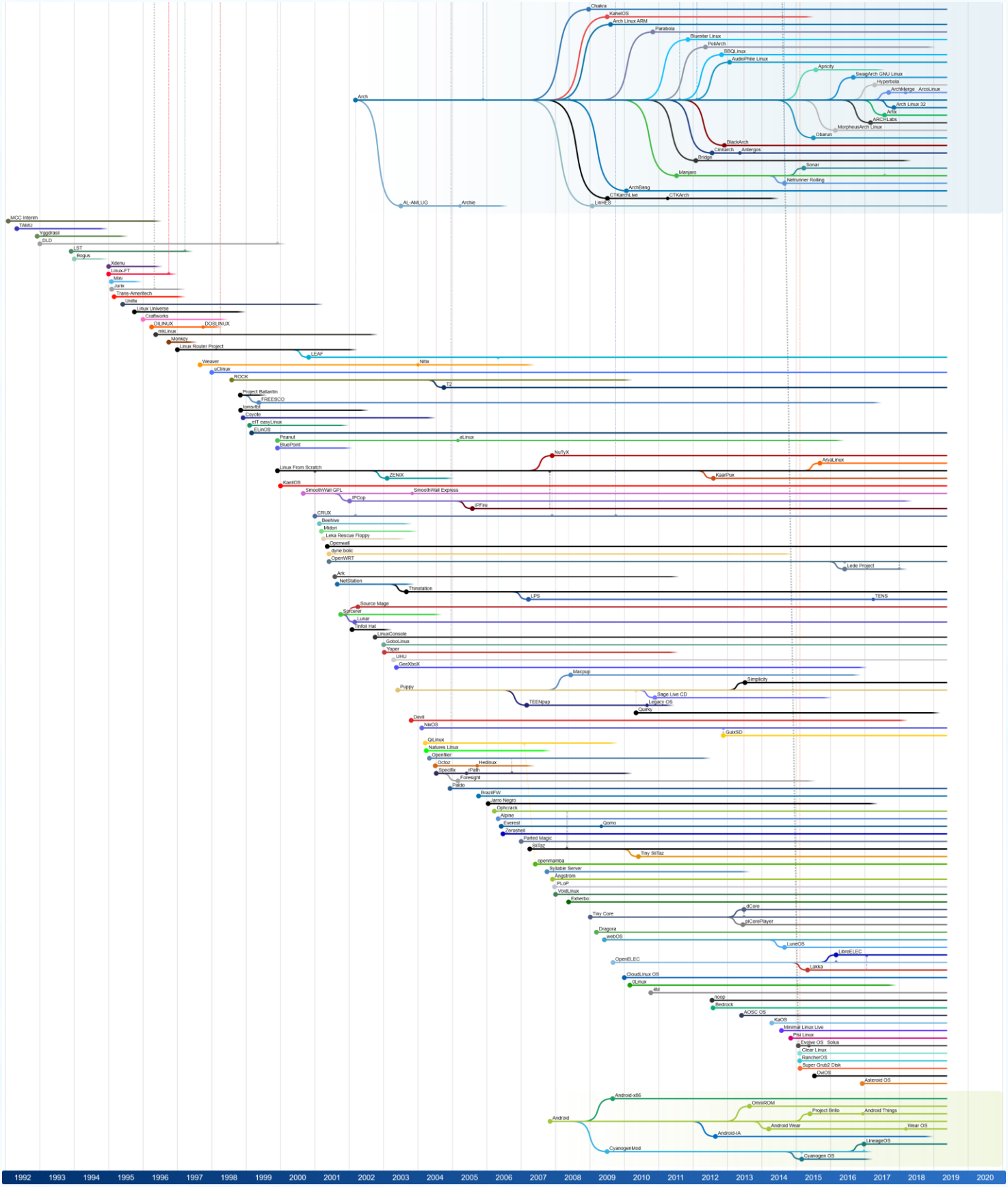


Branche Slackware

(Unix like, archives tarballs sans gestion des dépendances)



Autres ditributions
(arch Linux, Linux from scratch, Android…)



Source : https://commons.wikimedia.org/wiki/File:Linux_Distribution_Timeline.svg

LICENCES

L'originalité de Linux est d'être constitué d'un noyau libre et de logiciels libres.

Pour être qualifié de logiciel libre, un logiciel doit être disponible sous des conditions répondant à des critères stricts. La **FSF** (Free Software Foundation) et le projet Debian étudient les licences de logiciels pour déterminer s'il est libre. C'est en vertu de leurs droits d'auteurs que les contributeurs d'un logiciel libre accordent ces **quatre libertés** :

- Liberté d'**exécuter** le programme, pour tous les usages
- Liberté d'**étudier** le fonctionnement du programme, de l'**adapter** à vos besoins
- Liberté de **redistribuer** des copies afin d'aider votre prochain
- Liberté d'**améliorer** et de **rendre publique** les améliorations du programme

PS : le code source doit être accessible pour jouir des libertés d'études et d'amélioration du programme !

A. BSD (Berkeley software distribution)

Il s'agit des licences qui offrent la plus grande liberté. En général, seule la citation des auteurs originaux est demandée. En particulier, ces licences permettent de **redistribuer un logiciel libre sous une forme non libre**. Ces licences permettent donc à tout acteur de changer la licence sous laquelle le logiciel est distribué. Un cas de changement de licence courant est l'intégration de logiciel sous licence BSD dans un logiciel sous **copyleft** (licence **GPL**). Un autre cas courant est l'intégration de logiciel sous licence **BSD** dans les logiciels propriétaires.

Le code sous licence BSD peut être publié sous licence GPL sans le consentement des auteurs originaux puisque les termes de la GPL respectent tous ceux de la licence BSD. Mais l'inverse n'est pas possible : du code sous licence GPL ne peut pas être mis sous licence BSD sans l'autorisation des auteurs car la licence BSD ne fait pas référence à la notion de copyleft.

B. Copyleft, GNU (GNU's Not Unix) ou GPL (General Public License)

Les licences d'utilisation (copyright) de la plupart des programmes sont définies pour limiter ou supprimer toute liberté à l'utilisateur (interdiction de copier, de distribuer ou de créer tout autre produit dérivé). À l'inverse, la Licence Publique Générale (**G**eneral **P**ublic **L**icense) est destinée à garantir la liberté de partager et de modifier les logiciels libres, et de s'assurer que ces logiciels sont effectivement accessibles à tout utilisateur.

Le projet GNU a été lancé en 1984 pour développer un système d'exploitation complet et libre émanant d'Unix. La **FSF** est la principale organisation qui soutient le Projet **GNU**.

Le copyleft est un terme général, mais il existe beaucoup de variations. Le copyleft est l'outil fondamental de la GPL et est sujet actuellement de nombreuses polémiques, en particulier de nombreux dépôts de plainte du gargantuesque Microsoft.

Le copyleft indique que quiconque les redistribue, avec ou sans modification, **doit aussi transmettre la liberté de les copier et de les modifier**.

Le copyleft permet aussi d'intégrer des logiciels sous licence BSD et de les redistribuer sous licence GPL. L'inverse est toutefois impossible. En publiant du code GPL sous licence BSD, on peut autoriser la redistribution sans fournir le code source alors que c'est interdit par les termes de la licence GPL.

AGPL (Affero General Public License), est basée sur ce type de licence.

C. Licence Apache

Il s'agit d'une licence de logiciel libre et *open source*. Elle autorise la modification et la distribution du code sous toute forme (libre ou propriétaire, gratuit ou commercial) avec l'obligation du maintien du copyright lors de toute modification.

D. Domaine public

C'est une caractéristique juridique qui n'a **pas besoin de licence** du fait que **le logiciel n'a aucun ayant-droit**. **Les droits patrimoniaux concernant ce logiciel disparaissant**, il peut alors être utilisé encore plus librement. Théoriquement, tout logiciel est élevé dans le domaine public une fois les droits d'auteur échus : en général, l'auteur doit être mort pour que son œuvre tombe officiellement dans le domaine public. Toutefois, la durée de protection des droits d'auteur est bien plus longue que le plus ancien des logiciels, par exemple soixante-dix ans après la mort de l'auteur dans tous les pays de l'Union européenne...

Le domaine public permet la redistribution d'un logiciel libre sous forme propriétaire. Au lieu de mettre les logiciels GNU dans le domaine public, la **FSF** les met sous copyleft.

MAIS ALLONS DE CE PAS AU CŒUR DU SUJET…

COMMANDES ESSENTIELLES

Fichiers :

~ : représente le répertoire /home/*user*
ls : liste les fichiers et dossiers
ls -a : affiche tous les fichiers
ls -i : index des fichiers
ls -l : infos supplémentaires
ls -s : taille des fichiers
cp : Copy (le point représente le répertoire courant)
cp -r : **r**ecursive (copie aussi les répertoires)
cp -p : **p**reserve (owner/group id - permissions)

mv : **m**o**v**e -> déplacer ou renommer
rm : **r**e**m**ove files
rm -rf : force remove files & (sub)directories
file : affiche type de fichier
/dev/lp : dirige fichier vers imprimante
find *chemin* **-name** *file* **-print** : chercher un fichier ou un dossier (l'option –xdev ne parcourt pas des unités réseau)
find . -type d : recherche de répertoires

which *fichier* : recherche fichier dans $PATH

wget : télécharger fichier (HTTP.S / FTP)

ln *x y* : affecte le nom y au fichier x
ln -s : créé 1 lien symbolique entre 2 fichiers
chown *owner*[**:***group*] *file* : Change le propriétaire du fichier.
chown -R : recursive

Répertoires :

cd : Change Directory
mkdir : Make Directory
rmdir : Remove Directory
rm -rf : supprime répertoire, sous répertoires & fichiers

Utilisateurs :

users ou **who** : affiche les utilisateurs connectés

w : qui fait quoi ?

su [-] *user* : swap user
adduser ou **useradd, userdel** : gère les utilisateurs
id *user* : affiche le UID, GID
passwd u : change le mot de passe de l'utilisateur u en local (dans /etc/passwd & shadow)
yppasswd : change le mot de passe sur le serveur NIS

Groupes : (fichier info texte : /etc/**group**)

groups : infos groupes
groupadd, **groupdell**, **groupmod** : gère les groupes (voir /etc/**group**)
chgrp *groupe fichier* : change le fichier de groupe d'utilisateurs

usermod -a -G *Groupe1 Utilisateur1* : ajouter un utilisateur dans un groupe
gpasswd -d *Utilisateur1 Groupe1* : retirer un utilisateur d'un groupe

---

Disques :

df : Espace libre
du */etc* : Espace occupé
du -sh */etc* **--apparent-size** : Somme de l'espace occupé
stat -fc %s . : taille d'un bloc
pwd : affiche répertoire courant
mkswap : crée la partition swap
fsck : réparation d'un filesystem (démonté de préférence)
sync : vide les tampons disques

Système :

uname -a : informations système (noyau, version…)
lsmod : affiche les modules chargés dans le kernel
uptime : affiche la charge du système
dmesg : affiche les messages du boot
tty : affiche le nom du terminal
ALT + Fxx : change de terminal texte
halt : arrêt du système
shutdown *time* **'message'** : fermeture du système. *time* peut avoir ***hh:mm***, **+*m*** ou **now**.

dump 0f *archive partition* : sauvegarde d'un disque (en init 1)
restore *archive* : restauration disque
restore tf *archive* : affiche le contenu

Solaris : **ufsdump 0f** *archive partition*
Solaris : **ufsrestore vxf** *archive* (Specify next volume #: **1**)

Processus :

ps (-u : user, **-l** : long format, **-a** : other processes, **-m** : memory info **-H** : hierarchy) : processes en cours – Renvoie :
- PID. : numéro du process
- PPID : numéro du process parent
- C : priorité

service *nom* **[start|stop|restart]** : gestion de l'état d'un service

renice *priority* **PID** : change priorité du process (de -20 haut, à 20 bas)

Les processus zombies sont marqués par un **Z** ou **<defunct>** -> impossible de faire kill.

cmd **&** : met en tâche de fond *cmd*
jobs : affiche les process en tâche de fond
[Ctrl]-z : suspend un process en avant plan
bg : met un process en arrière plan
fg : met un process en avant plan

top : tâches en temps réel (solaris : **prstat**)
time *process* : calcule les ressources d'1 process

kill -9 *PID* : tue 1 processus
kill -HUP `` `cat /etc/syslog.pid` `` tue le processus syslog
kill %*job* : tue le processus n° job vu par la commande jobs

reboot : redémarrage

Solaris : **/usr/sbin/psrinfo -pv** : infos CPU

Mémoire :

free : état de la mémoire vive
free -b : affichage en bits
free -m : affichage en mégas

Réseau :

login : ouverture d'une nouvelle cession
logout : fermeture d'une cession

su - *utilisateur* : swap user

ping adresse
traceroute adresse

last : affiche les dernières connexions (se base sur /var/run/utpm)

hostname : nom de la machine
domainname : NIS /YP domain name
dnsdomainname : DNS domain name

Commandes générales :

man *nom fichier* : manuel d'aide (sortie **:q**)
more ou **less**: éditeur page à âge (sortie **:q**)
set : affiche / définit variable
printenv : variables d'environnement
setterm : modifie les attributs du terminal
mc : gestionnaire fichiers
clear : efface écran
date [MMDDhhmm[[CC]YY][.ss]] : affiche / modifie la date

mail : récupérer courrier
d : delete
q : quit
mail user : envoie 1 mail à user
mailto: envoie 1 mail
mail user -v : affiche tous les mails

Historique des commandes :
history : affiche l'historique des commandes saisies

renommer un ficher avec la date :
fichier=`**/bin/date** '+./*fichier*_%d-%m.%T`
cal : calendrier

write user : envoi d'un message
wall : envoi d'un message à tous les utilisateurs

history -c : efface cet historique

Commandes de diagnostics système :

Linux :
lsof : fichiers actuellement ouverts
strace : déboguage des appels systèmes
iostat : état du CPU & des entrées/sorties périphériques
vmstat : statistiques mémoire virtuelle
netstat -av : connexions actives
fuser -v -m . : process accédant à un fichier

Solaris :
truss : équivalent de strace
dtrace : outil évolué de truss
prtdiag : diagnostics système
prtconf : configuration système
psrinfo : informations sur les processeurs
sysdef : définitions système

Alias :
Ils permettent de définir ses propres commandes

Exemple : **alias Taille='ls -al | sort -n +4 | more'**

Particularité IBM AIX :

L'O.S. dispose d'une console d'administration en mode texte via la commande : **smit**

INSTALLATION

Les types de partitions :

Système de fichiers	Taille maximale d'un fichier	Taille maximale d'une partition	Journalisé	Gestion des droits d'accès	Snapshot	Quotas	Adressage	Notes
FAT (File Allocation Table)	2 GiB	2 GiB	Non	Non	Non	Non	16 bits	Développé par Microsoft sur les disquettes 3½
FAT32	4 GiB	8 TiB	Non	Non	Non	Non	32 bits	FAT32 augmente les limitations de FAT
exFAT (Extensible File Allocation Table)	16 TiB	256 TiB	Non	Non	Non	Non	64 bits	Développé par Microsoft et optimisé pour les mémoires flash (clés USB / cartes SD)
NTFS (New Technology File System)	16 TiB	256 TiB	Oui	Oui	Non	Oui	64 bits	Système de fichiers par défaut pour Windows NT depuis la version 3.1
EXT2 (Extended File System)	2 TiB	4 TiB	Non	Oui	Non	Non	32 bits	Système de fichiers natif de Linux.
EXT3	2 TiB	4 TiB	Oui	Oui	Non	Non	48 bits	ext2 + journalisation
EXT4	16 TiB	1 EiB	Oui	Oui	Non	Non	64 bits	ext4 augmente les limitations de EXT3
ReiserFS	8 TiB	16 TiB	Oui	Oui	Non	Non	64 bits	Idéal pour gérer les fichiers de moins de 4 ko.
BtrFS	16 EiB	16 EiB	Oui	Oui	Oui	Oui	64 bits	Fonction de snapshot et de sauvegarde incrémentale
XFS	8 Eo	8 Eo	Oui	Oui	Oui	Oui	64 bits	Haute performance grace aux entrées-sorties parallèles
ZFS	16 Eo	256 ZiB	Oui	Oui	Oui	Oui	128 bits	Produit par Sun Microsystems pour Solaris 10

Unités de taille :

Décimal			Binaire		
	Métrique	**Valeurs**		**ISO/IEC 80000**	**Valeurs**
Mégaoctet	MB	1000 ^ 2	Mebibyte	MiB	2 ^ 20 octets
Gigaoctet	GB	1000 ^ 3	Gibibyte	GiB	2 ^ 30 octets
Téraoctet	TB	1000 ^ 4	Tebibyte	TiB	2 ^ 40 octets
Pétaoctet	PB	1000 ^ 5	Pebibyte	PiB	2 ^ 50 octets
Exaoctet	EB	1000 ^ 6	Exbibyte	EiB	2 ^ 60 octets
Zettaoctet	ZB	1000 ^ 7	Zebibyte	ZiB	2 ^ 70 octets

A titre indicatif, l'espace occupé après installation en mode graphique
`[root@RockyLinux ~]# du / -h -max-depth=1`

CentOS 8 UEFI : Server + GUI

	Go restant	Volumes	Go provisionnés	Réellement utilisé	
Total	28		0		
	28	/boot	1	0,300	
	27	/	1	0,300	
	26	/root	2	0,030	
	24	swap	2	0,000	
	22	/usr	6	3,700	
	16	/home	2	0,070	
	14	/tmp	3	0,070	
	11	/var	10	0,300	(logs + mails + docker + compilation kernel)
	1	/srv	0	0,000	Pas sur RedHat
	1	/opt	0	0,000	Programmes tiers
	1				
				4,770	

Sans l'interface graphique :

Ubuntu server

	Go restant	Volumes	Go provisionnés	Réellement utilisé
Total	20		0	
	20	/boot	1	0,200
	19	/	6	
	13	/root	0	0,030
	13	/usr	0	1,900
	13	/home	0	0,070
	13	/var	13	0,430
	0			
				2,630

(logs + mails + docker + compilation kernel)

Quelques commandes :

`fdisk`, **`format`** prépare les partitions (maximum 8 par disque) ; avant de quitter l'utilitaire format, sauvegarder les modifications (save).
`mkfs` formate les partitions
`newfs -v` *`/dev/rdsk/c0t1d0s0`* utilise mkfs d'une manière plus conviviale.

`find` *`point_de_montage`* **`-xdev -ls`** récupère les inodes d'une partition. Il peut être utile de conserver ces informations pour d'éventuels problèmes dans le répertoire **lost+found**.

Définition des montages automatiques au démarrage : **/etc/fstab**.

Le fichier **/etc/mtab** ou **/etc/mnttab** contient la liste des montages actifs.

Configuration de X :
Fichier de config. texte pour X : /etc/XF86config

Mandrake : drakconf, linuxconf, XFdrake, sndconfig

Redhat : XF86Setup, Xconfigurator, redhat-config-xfree86

Gestion des couleurs sur les terminaux :
Ajouter : **`alias ls='ls -color=auto'`** dans **`~/.bashrc`**

Changement du PATH :
PATH=$PATH:/nouveau_répertoire
export PATH

Lancements de programmes personnels lors du boot : fichier **/etc/rc.d/rc.local**
Ex : pour la prise en compte du clavier français, rajouter cette ligne :
/usr/bin/loadkeys /usr/lib/kbd/keytables/fr-latin1.map

Changement du login shell : **`chsh`**

Rappels :

/dev/**sda**	: 1er stockage SSD / SCSI	/dev/**hda**	:1er disque dur IDE
/dev/**sda1**	: 1ère partition du 1er stockage	/dev/**hda1**	: 1ère partition du 1er disque

/dev/**sda2**	: 2ème partition du 1er stockage	/dev/**hdb**	: 2ème disque dur
/dev/**sdb**	: 2ème stockage	/dev/**fd0**	: diquette A

DEMARRAGE DU SYSTEME

A. GRUB (GRand Unified Bootloader) :

Pour configurer le multi-boot après l'installation d'un système unix, on peut installer grub à partir d'un shell linux :

Lancer Grub :
```
sudo -s
grub --batch
```

Pour connaître la partition sur laquelle se situent les fichiers de Grub :
```
grub> find /boot/grub/stage1
```

Si la commande renvoie (*hd0,1*), il faut saisir
```
grub> root (hd0,1)
```

Installation de Grub :
```
grub> setup (hd0)
grub> quit
```

Afin d'ajouter Windows dans la liste des systèmes bootable, il faut éditer le fichier :
/boot/grub/menu.lst

```
title       Windows NT/2000/XP
root        (hd0,0)
savedefault
makeactive
chainloader +1
```

B. Initialisation du système (System V):

- Chargement du fichier compressé **initrd** (Initial RAM Disk, souvent situé dans /boot) et de ses modules (voir commande **lsmod**).
- Lancement des deux premiers processus, dont **init** (PID 0 et PID 1).
- Lancement de /etc/**inittab** en fonction du niveau de démarrage.
- Exécution du script /etc/**rc*X*** ou X est le niveau d'exécution.
- /etc/**rc*X*** lance les liens situés dans **/etc/rc*X*.d/** par ordre alphabétique. Ces liens pointent vers **/etc/init.d/** et commencent par S* pour le démarrage des démons, et K* pour leurs arrêts. La commande `chkconfig --level` *x mysqld* `[on | off]` permet de gérer l'activation ou non de ces démons.

Solaris : les fichiers dans /etc/default/ initialisent l'environnement par défaut.

Sur Linux, la commande `runlevel` permet de savoir quel niveau est en cours.
Sur Solaris, `who -r` a la même fonction.

/etc/**rc** est utilisé pour un changement de niveau (avec la commande **init *niveau***).

Le fichier /var/log/**boot.log** contient une trace des arrêts et démarrages de la machine.

C. Montage des systèmes de fichiers

Les partitions statiques sont montées à la lecture du fichier /etc/**fstab** ; `mount -a` utilise ce fichier.
Sur les systèmes Solaris, la commande est `moutall` et le fichier est : /etc/**vfstab**

`mount` : affiche les partitions montées
`mount -t` *`type`* `/dev/`*`peripherique`* *`point_de_montage`*
Pour un montage disque sur Solaris, la commande est :
`mount -F` *`ufs`* `/dev/dsk/disk/`*`c1t0d0s7`* *`point_de_montage`*

Exemple de fichier /etc/fstab :

```
/dev/sda1       /          ext3      acl,user_xattr      1 1
/dev/sda2       /opt       ext3      acl,user_xattr      1 2
/dev/sda3       /srv       ext3      acl,user_xattr      1 2
/dev/sda5       /tmp       ext3      acl,user_xattr      1 2
/dev/sda6       /var       ext3      acl,user_xattr      1 2
/dev/sda7       swap       swap      defaults            0 0
```

Colonne 1 : (fs_spec), périphérique bloc ou le système de fichiers distant à monter
Colonne 2 : (fs_file), point de montage du système de fichier
Colonne 3 : (fs_vfstype), type de système de fichiers
Colonne 4 : options de montage (voir le man de mount ou NFS)
Colonne 5 : (fs_freq), utiliser **dump** (s'il est présent) pour sauvegarder ce filesystem
Colonne 6 : (fs_passno), permet une vérification du système de fichiers par **fsck**

PS : Selon les versions de Linux, la première colonne peut être identifiée par l'Universally Unique Identifier (UUID), que l'on trouve avec ls -lF /dev/disk/by-uuid/

Monter 1 CD-ROM : dans /mnt/CdRom :	Monter une disquette dans /mnt/144 :
`mkdir /mnt/CdRom` `mount -t iso9660 /dev/hda /mnt/CdRom`	`mkdir /mnt/144` `mount -t auto /dev/fd0 /mnt/144`

Création d'une image ISO à partir d'un CD-ROM : `dd if=`*`/dev/cdrom`* `of=`*`image.iso`*
Monter une image ISO : `mount -o loop -t iso9660` *`image.iso /mnt/iso`*

D. Login

Les variables d'environnement par défaut sont initialisées dans /etc/**profile** ou /etc/rc.d/**rc.sysinit** (linux), ou bien encore dans /etc/default/**init** (Solaris).
Ces variables peuvent différer selon le profil grâce à un fichier situé dans le homedirectory de l'utilisateur. Le nom du fichier diffère selon le shell :

`.profile` pour le Bourne (sh) et Korn (ksh)	`.bash_profile` pour le Bourne Again shell
`.login` et `.cshrc` pour le C shell (csh)	`.tcshrc` et `.cshrc` pour le TC shell
`.zlogin` et `.zshrc` pour le Zorn shell	

L'historique des commandes du shell peut être paramétré via les variables système suivantes :
HISTSIZE : nombre maximal de commande dans le fichier d'historique
HISTFILESIZE : nombre maximal de lignes dans le fichier d'historique
HISTFILE : chemin vers le fichier historique

E. Shadow

1) Formatage

Le fichier **/etc/shadow** comprend les mots de passe de utilisateurs, ainsi que les restrictrions d'accès. Comme pour le fichier **/etc/passwd**, les entrées ont délimitées par la caractère " **:** "

Colonne 1 : login
Colonne 2 : mot de passe haché, avec comme premiers caratères :

- "1" : MD5
- "$2a$" ou "$2y$" : Blowfish
- "5" : SHA-256
- "6" : SHA-5124
- " !", "*LK*" ou "*" : le compte local est vérouillé, mais une connexion SSH est possible
- " !!" : le mot de passe n'est pas encore défini

Colonne 3 : dernière modification du mot de passe (exprimé en nombre de jours après le 1er janvier 1970)
Colonne 4 : nombre de jours avant le que le mot de passe puisse être changé
Colonne 5 : nombre de jours après lesquels le mot de passe doit être changé
Colonne 6 : nombre de jours durant lesquels l'utilisateur sera informé de l'expiration de son mot de passe
Colonne 7 : nombre de jours avant la désactivation du compte
Colonne 8 : date de la désactivation d'un compte (nombre de jours après le 1er janvier 1970)

2) Gestion

La commande **getent** affiche les entrées de nsswitch :

```
[root@RockyLinux ~]# getent passwd
[root@RockyLinux ~]# getent shadow
[root@RockyLinux ~]# getent group
```

Affiche le statut d'un compte :

```
[root@RockyLinux ~]# chage -l games
```

Expiration du compte :

```
[root@RockyLinux ~]# chage -E today games
```

Réactivation du compte :

```
[root@RockyLinux ~]# chage -E 99999 games
```

Réactivation complète sans restriction :

```
[root@RockyLinux ~]# chage -m 0 -M 99999 -I -1 -E -1 games
```

Petit script pour connaître l'état d'expiration d'un compte :

```
DAYSSINCE=$(( $(date +%s) / 86400 ))
EXPIREDAY=$(getent shadow | awk -F: '/^games:/{print $8}')
if [[ $DAYSSINCE -ge $EXPIREDAY ]]
then
    EXPIRED=true
fi
```

PRINCIPAUX REPERTOIRES

/boot/ : chargement noyau linux
/root/ : répertoire de l'administrateur
/home/ : répertoires utilisateurs

/bin/ : binaires exécutables système pour tous les utilisateurs
/sbin/ : Static Binaries, binaires d'administration pour le GID root

/etc/ : fichiers de configuration & utilitaires.

/usr/ : Unix System Resources, contient les programmes hors administration système
/usr/lib/ : bibliothèques pour utiliser les programmes situés dans /usr/bin et /usr/sbin
/usr/include/ : entêtes de programmation
/usr/local/lib/ : bibliothèques d'outils personnels
/usr/local/include/ : autres fichiers d'entêtes
/usr/local/etc/ : autres fichiers de config.
/usr/src/ : sources des distributions libres
/usr/X11R6/ : fichiers de X

/lib/ : bibliothèques système nécessaires à /bin

/opt/ : applications annexes

/dev/ : fichiers des périphériques.
/dev/null : périphérique inexistant (utilisé pour les redirections).
/mnt/ : périphériques montés.

/tmp/ : fichiers temporaires (peuvent être effacés au redémarrage, contrairement à **/var**)

/var/ : fichiers à contenu variables
/var/log/ : fichiers de bord du système
/var/spool/mail/ : boites aux lettres
/var/run/ : fichiers des process contenant le PID

/lost+found/ : fichiers orphelins

Certains filesystems sont uniquement présents en mémoire vive. Ils sont appelés « pseudo-filesystems »

Le filesystem virtuel **/proc** , permet de visualiser des éléments système liés à la gestion des processus par le kernel, ainsi que certaines informations système liées au matériel :

```
[root@RockyLinux ~]# cat /proc/cpuinfo
[root@RockyLinux ~]# cat /proc/meminfo
[root@RockyLinux ~]# cat /proc/scsi/scsi
[root@RockyLinux ~]# cat /proc/partitions
```

Le filesystem virtuel **/sys** permet de visualiser des éléments système liés aux périphériques.

```
[root@RockyLinux ~]# cat /sys/class/net/ens33/speed
```

Autres types de montages :

A. SWAP

Elle est montée à partir du fichier /etc/**rc.sysinit**
Montage / démontage du disque swap : **swapon -a** ou **swapoff -a**

Création d'un fichier de swap de 256 Mo :
```
dd if=/dev/zero bs=1M of=/tmp/Fichier count=256
```

Attribution du fichier : **mkswap** /tmp/Fichier

Allocation de l'espace : **swapon** /tmp/Fichier

LVM

A. Logical Volume Manager

Au lieu d'utiliser des disques de volumétrie fixe, pour lesquels il est difficile et risqué d'en modifier leurs tailles, il est devenu courant d'utiliser un gestionnaire de volumes. Les systèmes Linux intègrent la possibilité de gérer, sécuriser et optimiser de manière souple les espaces de stockage en ligne.

De manière globale, nous avons un ou plusieurs disques physiques (PV), sur le(s)queul(s) nous allons définir un ou plusieurs groupes de volumes (VG), dans lesquels nous allons monter un ou plusieurs volumes logiques (LV).

Ces volumes logiques peuvent alors être strippés (agrégation par bandes, ou pseudo RAID 0), mirrorés (pseudo RAID 1), étendus à chaud (depuis le noyau 2.6.22 de Linux), ou « snapshotés ».

Ces opérations s'effectuent sous le profil root.

Imaginons que nous avons un disque qui commence à être saturé (sda). Nous avons branché un nouveau disque (sdb) de 2 Go, sur lequel nous allons allouer une partie de son espace disponible au système, et éventuellement pouvoir l'étendre ultérieurement.

On vérifie que le nouveau disque est bien reconnu par le système :

```
[root@RockyLinux ~]# fdisk -l
Disk /dev/sdb: 2147 MB, 2147483648 bytes, 4194304 sectors
```

L'étape suivante consiste à créer une nouvelle partition sur ce disque. Par défaut, fdsik créé des partitions de type Linux (83) ; il faut modifier le type de partition en LVM (8e)

```
[root@RockyLinux ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).
…
Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x801237b1.

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p):
Using default response p
Partition number (1-4, default 1):
First sector (2048-4194303, default 2048):
Using default value 2048
…
Partition 1 of type Linux and of size 2 GiB is set

Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): 8e
Changed type of partition 'Linux' to 'Linux LVM'

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

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

Pour d'anciennes versions de Linux, il est demandé de redémarrer le système, suite au changement de la table de partitions :

```
AVERTISSEMENT : la re-lecture de la table de partitions a échoué avec l'erreur 22: Argument invalide.
Le kernel va continuer d'utiliser l'ancienne table.
La nouvelle table sera utilisée lors du prochain réamorçage.
```

Nous pouvons maintenant créer le volume physique (PV) sur /dev/sdb1 :

```
[root@RockyLinux ~]# pvcreate /dev/sdb1
  Physical volume "/dev/sdb1" successfully created
```

On vérifie :

```
[root@RockyLinux ~]# pvscan
  PV /dev/sdb1   VG vgExtention     lvm2 [2.00 GiB / 1.02 GiB free]
[root@RockyLinux ~]# pvs
  PV         VG            Fmt  Attr PSize  PFree
  /dev/sdb1                lvm2 ---   2.00g 2.00g
[root@RockyLinux ~]# pvdisplay
  --- Physical volume ---
  PV Name               /dev/sdb1
  VG Name               vgExtention
```

La voie est libre pour créer le groupe de volumes (VG), nommé « vgExtention » pour l'occasion :

```
[root@RockyLinux ~]# vgcreate vgExtention /dev/sdb1
  Volume group "vgExtention" successfully created
```

On vérifie :

```
[root@RockyLinux ~]# vgscan
  Reading all physical volumes.  This may take a while...
  Found volume group "vgExtention" using metadata type lvm2
[root@RockyLinux ~]# vgs
  VG            #PV #LV #SN Attr   VSize  VFree
  vgExtention     1   0   0 wz--n-  2.00g 2.00g
[root@RockyLinux ~]# vgdisplay
  --- Volume group ---
  VG Name               vgExtention
  Format                lvm2
```

Vérifier deux fois, c'est mieux qu'une :

```
[root@RockyLinux ~]# pvs
  PV         VG            Fmt  Attr PSize  PFree
  /dev/sdb1  vgExtention   lvm2 a--   2.00g 2.00g
```

C'est parti pour l'étape suivante, qui consiste à créer un volume logique (LV), nommé « lvPartition1 » pour l'occasion, dans le groupe de volumes vgExtention. Nous allons lui attribuer la moitié de l'espace total (1 000 Mo), afin de poursuivre ce chapitre (extension de cette partition) :

```
[root@RockyLinux ~]# lvcreate -L1000 --name lv_Partition1 vgExtention
  Logical volume "lv_Partition1" created.
```

Si l'on souhaite occuper la totalité de l'espace, nous pourrions faire de la sorte :

```
[root@RockyLinux ~]# lvcreate -l 100%FREE -n lv_Partition1 vgExtention
```

On vérifie :

```
[root@RockyLinux ~]# lvscan
  ACTIVE            '/dev/vgExtention/lv_Partition1' [1000.00 MiB] inherit
[root@RockyLinux ~]# lvs
  LV            VG            Attr       LSize    Pool Origin Data%  Meta%
Move Log Cpy%Sync Convert
  lv_Partition1 vgExtention                        -wi-a-----       1000.00m
[root@RockyLinux ~]# lvdisplay
  --- Logical volume ---
  LV Path                /dev/vgExtention/lv_Partition1
  LV Name                lv_Partition1
  VG Name                vgExtention
```

Vérifier deux fois, c'est mieux qu'une :

```
[root@RockyLinux ~]# pvs
  PV         VG            Fmt  Attr PSize  PFree
  /dev/sdb1  vgExtention   lvm2 a--   2.00g 1.02g
[root@RockyLinux ~]# vgs
  VG            #PV #LV #SN Attr   VSize  VFree
  vgExtention     1   1   0 wz--n-  2.00g 1.02g
[root@RockyLinux ~]# lsblk
NAME                          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sdb                             8:16   0    2G  0 disk
└─sdb1                          8:17   0    2G  0 part
  └─vgExtention-lv_Partition1 253:6    0 1000M  0 lvm
```

C'est tout bon, on passe à l'avant-dernière étape : le formatage de ce volume logique. Là, pour l'exemple, je vais le formater en xfs (attention, pour xfs, il faut une architecture 64 bits), avec comme label « Oracle » :

```
[root@RockyLinux ~]# mkfs.xfs -L Oracle /dev/vgExtention/lv_Partition1
```

Si l'on choisi un formatage en ext4, la commande diffère peu :

```
[root@RockyLinux ~]# mkfs.ext4 -L Oracle /dev/vgExtention/lv_Partition1
```

On vérifie :

```
[root@RockyLinux ~]# blkid |grep Partition1
/dev/mapper/vgExtention-lv_Partition1: LABEL="Oracle" UUID="…" TYPE="xfs"
```

Si l'on à choisi un formatage en ext4 :

```
/dev/mapper/vgExtention-lv_Partition1: LABEL="Oracle" UUID="…" TYPE="ext4"
```

On y est presque : il ne reste plus qu'à monter ce nouveau filesystem dans un répertoire que l'on va créer :

```
[root@RockyLinux ~]# mkdir /mnt/Oracle
[root@RockyLinux ~]# mount /dev/mapper/vgExtention-lv_Partition1 /mnt/Oracle
```

On vérifie :

```
[root@RockyLinux ~]# mount |grep Oracle
/dev/mapper/vgExtention-lv_Partition1 on /mnt/Oracle type xfs …
```

```
[root@RockyLinux ~]# df -h
Filesystem                                Size  Used Avail Use% Mounted on
/dev/mapper/vgExtention-lv_Partition1  997M   33M  965M   4% /mnt/Oracle
...
```

Pour boucler la boucle, se rendre au chapitre « Montage des systèmes de fichiers : », et ajouter une ligne au fichier /etc/fstab ; afin de monter ce filesystem à chaque reboot du serveur…

Afin de clore ce chapitre, je vais imaginer que ma partition LV est presque pleine, et me rappeler qu'il reste 1 Go qui sont encore disponibles. Je vais donc étendre à chaud la partition existante de 1 à son maximum (2 Go).

```
[root@RockyLinux ~]# lvresize -l +100%FREE /dev/vgExtention/lv_Partition1
  Size of logical volume vgExtention/lv_Partition1 changed from 1000.00 MiB
(250 extents) to 2.00 GiB (511 extents).
  Logical volume lv_Partition1 successfully resized

[root@RockyLinux ~]# xfs_info /dev/mapper/vgExtention-lv_Partition1
...

[root@RockyLinux ~]# xfs_growfs /mnt/Oracle
meta-data=/dev/mapper/vgExtention-lv_Partition1  isize=256        agcount=4,
agsize=64000 blks

data blocks changed from 256000 to 523264
```

PS : pour un filesystem **ext4**, remplacer la commande **xfs_growfs** par **resize2fs**

On vérifie :

```
[root@RockyLinux ~]# df -h
Filesystem                                Size  Used Avail Use% Mounted on
/dev/mapper/vgExtention-lv_Partition1  2.0G   33M  2.0G   2% /mnt/Oracle
```

Il est possible d'étendre un volume logique sur un autre PV :
`[root@RockyLinux ~]# vgextend vgExtention /dev/sdc1`

De la même manière que l'on peut augmenter la taille des LVM, il est possible de les diminuer :

```
[root@RockyLinux ~]# lvreduce -L 1G /dev/vgExtention/lv_Partition1
```

Attention, comme indiqué lors de la commande, « THIS MAY DESTROY YOUR DATA » : Il est important de réduire la taille des fichiers résidant au sein du volume, avant de le réduire lui-même...

…et cela peut nécessiter un re-formatage :

```
[root@RockyLinux ~]# umount /mnt/Oracle
[root@RockyLinux ~]# mkfs.xfs -fL ESPI /dev/ vgExtention/lv_Partition1
[root@RockyLinux ~]# mount /dev/mapper/vgExtention-lv_Partition1 /mnt/Oracle
```

B. Suppression des LVM

Les commandes suivantes permettent de détruire les opérations effectuées précédemment :

Dans la pratique, des process en cours peuvent encore accéder à la partition montée, et empêcher de la démonter. Si tel est le cas, se rendre au chapitre « Gestion des démons »

```
[root@RockyLinux ~]# umount /mnt/Oracle
[root@RockyLinux ~]# lvremove /dev/vgExtention/lv_Partition1
Do you really want to remove active logical volume lv_Partition1? [y/n]: y
  Logical volume "lv_Partition1" successfully removed
[root@RockyLinux ~]# vgremove vgExtention
  Volume group "vgExtention" successfully removed
[root@RockyLinux ~]# pvremove /dev/sdb1
  Labels on physical volume "/dev/sdb1" successfully wiped
```

...et effacer l'entrée du fichier /etc/fstab.

PACKAGES

Red-Hat & forks : rpm

Installation : **rpm -ivh** *package*.rpm Mises à jour : **rpm -U** *package*.rpm Suppression : **rpm -e** *package*.rpm	Liste des packages installés : **rpm -q -a**

Liste des fichiers modifiés après l'installation d'un package : **rpm -ql** *<nom du paquet>*

Surcouche RPM : yum

 # yum update yum : mises à jour
 # yum [install | remove] *<nom du paquet>* : gestion des packages
 # yum update : mise à jour de l'ensemble des packages
 # yum list kerne* : affiche les packages
 # yum repo list : affiche la liste des dépôts situés dans lerépertoire /etc/yum.repos.d/

yum (écrit en Python 2) est remplacé par la commande **dnf** (Python 3).

 # dnf [install | remove] *<nom du paquet>*
 # dnf check-update : recherche les mises à jour des programmes installés
 # dnf update : met à jour tous les programmes installés
 # dnf upgrade : met à jour l'ensemble de la distribution
 # dnf repolist -v : affiche les référentiels

Debian : **apt** (Advanced Packaging Tool)

Installation : **apt-get install** Liste des mises à jour : **apt-get update** Mises à jour : **apt-get dist-update**	Liste des packages installés : **apt-cache search**

Slackware : **pkgtool**

FreeBSD :

```
root@FreeBSD:~ # pkg install sudo
root@FreeBSD:~ # pkg delete sudo
root@FreeBSD:~ # pkg update
root@FreeBSD:~ # pkg upgrade
```

GESTION DES DROITS D'ACCES

A. Droits des filesystems

chmod *droits fichier* : change les droits d'accès :
User **G**roup **O**ther (valeurs de 0 à 7) - exemple : chmod 704 fichier
Read - **W**rite - **E**xecute
Exemples : chmod u+rwx,g+rw-x,o-rwx
chmod a+rw-x : a désigne u+g+o

umask droits : modification des droits de création de fichiers par défaut. umask 777 enlève tous les droits.

Droits spéciaux: **setuid**, **setgid**, **sticky bit** (chmod **4**xxx, **2**xxx, ou **1**xxx – bits , **s** sur user, **s** sur group, ou **t**) ->.fichiers / répertoires avec attribut **x**.

Les droits d'endossement :

Les droits d'endossement sont très importants pour la sécurité : au lieu de donner l'accès à un fichier, on donne le droit d'accès à une commande. Le kernel, au moment de l'exécution de la commande endosse l'identité du propriétaire ou du groupe de la commande au lieu de celle de l'utilisateur qui a lancé la commande.
Donc l'accès au fichier se fait par le biais de la commande et non pas directement.

Par exemple, le programme **passwd**, permet à un utilisateur de modifier son mot de passe dans **/etc/passwd**, alors que seul root peut modifier ce fichier.
De la même façon, un fichier **setgid**, s'exécute avec les droits du groupe propriétaire.

Le sticky bit :

Lorsqu'un répertoire à le **sticky** bit (par exemple /tmp), chaque utilisateur ne pourra effacer dans ce répertoire que les fichiers qui lui appartiennent.
Alors qu'un exécutable peut être déclaré setuid et setgid par son propriétaire, seul l'administrateur système peut positionner le sticky bit.

B. SUDO (Super User Do) :

Le programme **sudo** est défini en tant que binaire 'setuid' (voir ci-dessus) :
---**s**--x--x 1 root root 81644 Dec 31 23:59 /usr/bin/sudo

```
root@Debian:~# visudo
```

```
# /etc/sudoers file.
#
# Host alias specification
Host_Alias    LAN = localhost, 192.168.1.137, ma.machine.LocalNet
Host_Alias    WAN = www.pierreau.fr, 193.37.145.41

# User alias specification
User_Alias STAFF=Pierre,Paul

# Cmnd alias specification
Cmnd_Alias NET = /bin/ping, /usr/bin/traceroute, /usr/bin/ftp
Cmnd_Alias Stop = /sbin/shutdown -h now
Cmnd_Alias Attention = /sbin/reboot, !/bin/su, !/usr/sbin/visudo,
!/usr/bin/passwd, !/usr/sbin/userdel
# User privilege specification
root ALL=(ALL) ALL
STAFF ALL=(ALL) NET
Jacques WAN=(ALL) Stop
Lagaffe LAN=(ALL) Attention
Administrateur1 ALL=(ALL) NOPASSWD:ALL
```

User privilege specification
root ALL=(ALL) ALL
Cette ligne signifie que l'utilisateur root peut lancer à partir de n'importe quelle machine (premier ALL) à partir de n'importe quel utilisateur (second ALL) n'importe quelle commande (troisième ALL)

Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL
Ici cela signifie que le groupe admin (issu de /etc/group) peut lancer à n'importe quelle machine (premier ALL) à partir de n'importe quel utilisateur (second ALL) n'importe quelle commande (troisième ALL).

Lagaffe localhost = (ALL) ALL, (root) !ALL
Cette ligne signifie que l'utilisateur Lagaffe à partir de la machine (localhost) peut lancer à partir de n'importe quel utilisateur (ALL entre parenthèse) SAUF le compte root, n'importe quelle commande (second ALL).

REDIRECTIONS, PIPES /DEV/NULL

A. Les canaux d'entrée-sorties

Le canal d'entrée standard **stdin** (clavier) : **0**
L'injection d'un fichier de paramètres à une commande peut s'effectuer via une redirection du canal d'entrée vers ce programme :
commande* 0< *parametres

PS : une écriture plus simple est envisageable : ***commande* < *parametres***

Le canal de sortie standard **stdout** (le terminal) : **1**
Il est utilisé pour l'affichage des résultats des commandes sur l'écran

Le canal des erreurs **stderr** (le terminal) : **2**
Si l'on provoque une erreur, le système utilise le canal 2.
Redirection du canal 2 : `cat /tmp/FichierInexistant 2> /tmp/Resultat`

Redirection d'un canal de sortie vers un autre canal de sortie : **>&**
Redirection du canal stderr vers stdout : ***commande* 2>&1**

Redirection de stdout et stderr vers un fichier Resultat :
Attention : ***commande* 2>&1** >/tmp/Resultat

- **stderr** est redirigé vers la valeur courante de la **stdout**, donc l'**écran**
- **stdout** vers un fichier. Donc **stdout a été redirigé vers ce fichier**.

Alors que Programme > /tmp/Resultat **2>&1**

- **stdout** est redirigée vers un fichier
- **stderr** est redirigée vers la valeur courante sur laquelle pointe **stdout**, donc **le fichier**. En conséquence, **stdout et stderr ont bien été redirigés vers un même fichier**.

Autre syntaxe :

```
cat Fichier.txt 1>>Resultat.txt 2>>Resultat.txt
```

B. LES PIPES

Ils consistent à brancher des canaux entre eux dans le but d'effectuer des opérations à la chaine :

```
ls -ls /etc | sort -n +5f | grep -v root
```

Affiche les fichiers par ordre de taille :
`find . -type f -print0 | xargs -0 du -k | sort -nr` : affiche les fichiers par ordre de taille.

Affiche les dossiers par ordre de taille :

```
du -h --max-depth=3 / 2>/dev/null | sort -h | tail -20
```

Efface les fichiers datés de plus de 30 jours :

`find . -mtime +30 | xargs -r rm` : efface les fichiers datés de plus de 30 jours

C. LE PERIPHERIQUE /dev/null

Il s'agit d'un pseudo périphérique dans lequel les flux ne sont pas traités, perdus à jamais ; l'information est inexistante, à la manière d'un trou noir…

Nous ne voulons pas traiter les messages stdout : `cat FichierInexistant 2> /dev/null`
Nous désirons vider un fichier : `cat /dev/null > FichierPlein`

SYSTEMD

Systemd (System daemon) est une alternative au démon init de System V. Il permet entre autres d'optimiser les services au démarrage du système grâce à un chargement parallélisé.

A. Runlevel

Arrêt du système (runlevel 0) : `systemctl isolate poweroff.target`
Passer en mode single user (runlevel 1) : `systemctl isolate rescue.target`
Passer en mode console (runlevel 3) : `systemctl isolate multi-user.target`
Activer l'interface graphique (runlevel 5) : `systemctl isolate graphical.target`

Connaître le mode de démarrage actuel : `systemctl get-default`

Activer au boot le mode console : `systemctl set-default multi-user.target`
Activer au boot le mode graphique : `systemctl set-default graphical.target`

Ces commandes remplacent le fichier /etc/inittab supporté sans SytemV.

B. Gestion des démons

Pour gérer les services :

```
root@Debian:~# systemctl start httpd.service
root@Debian:~# systemctl stop mariadb
root@Debian:~# systemctl restart zabbix-server
root@Debian:~# systemctl status httpd.service
root@Debian:~# systemctl is-active cron.service
```

Pour activer / désactiver / vérifier les services au boot :

```
root@Debian:~# systemctl enable mariadb
root@Debian:~# systemctl disable postgresql-9.4.service1
root@Debian:~# systemctl is-enabled NetworkManager.service
```

Connaître les démons lancés au démarrage :

```
root@Debian:~# service --status-all |grep +
```

Affichage des services les plus lents à démarrer au boot :
`root@Debian:~# systemd-analyze blame`

Analyser des temps de démarrage :
`root@Debian:~# systemd-analyze time`

Chemin de dépendances qui ont pris le plus de temps à s'achever :
`root@Debian:~# systemd-analyze critical-chain`

Liste des services en échec :
`root@Debian:~# systemctl --type=service --failed`

JOURNALCTL

Gestion des fichiers de logs, remplaçant de syslogd

A. Recherches basiques

Affiche les messages critiques :
```
root@Debian:~# journalctl -p crit
```

Classification des alertes :
0 : emerg : situation d'urgence rendant le système inutilisable
1 : alert : situation critique nécessitant une intervention immédiate
2 : crit : situation critique
3 : err : condition d'erreur
4 : warning : simple avertissement
5 : notice : message d'information
6 : info : message d'information à caractère moins important que « notice »
7 : debug : message de debuggage

Visualiser les logs sudo :

```
root@Debian:~# journalctl _COMM=sudo
```

Visualiser les logs concernant l'utilisateur root :

```
root@Debian:~# journalctl _UID=0
```

B. Recherches chronologiques

Affiche les messages critiques dans une plage horaire :
```
root@Debian:~# journalctl -p crit --since 13:26:20 --until 13:27:00
```

Affiche les logs critiques à partir d'hier :
```
root@Debian:~# journalctl -p alert --since yesterday
```

Affiche les logs Apache du jour :
```
root@Debian:~# journalctl -u httpd.service --since today
```

Affiche les log au fil de l'eau :
```
root@Debian:~# journalctl -f
```

C. Phases de démarrage du système

Visualiser les journaux des 10 derniers boot disponibles :

```
root@Debian:~# journalctl --list-boots | tail -n10
```

Affiche les messages d'avertissement du premier boot disponible, et de l'avant dernier :
```
root@Debian:~# journalctl -p warning -b 1
root@Debian:~# journalctl -p warning -b -1
```

Les journaux peuvent être paramétrés en éditant le fichier **/etc/systemd/journald.conf**

IPTABLES

A. Filtrages

Démarrage du service dans `/etc/rc.d/init.d/iptables start`

```
# Efface les règles mise en place
iptables -F
iptables -X

# Accepte les trafics sur l'interface locale
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

# Accepte les connexions SSH (port 22)
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables -A INPUT -p tcp -s 192.168.0.0/24 --dport 22 -j ACCEPT
iptables -A INPUT -p tcp -i eth0 --dport ssh -j ACCEPT

# Pour les serveurs web HTTPS
iptables -A INPUT -p tcp -i eth0 --dport 443 -j ACCEPT

# Accepte les requêtes DHCP (ports 67 et 68)
iptables -A INPUT -p udp --sport 67:68 --dport 67:68 -j ACCEPT

# HTTP client
iptables -A INPUT -p tcp -m tcp --sport http --dport 1024: -m state --state
ESTABLISHED -j ACCEPT
iptables -A INPUT -p tcp -m tcp --sport https --dport 1024: -m state --state
ESTABLISHED -j ACCEPT

# Restreint les sources de connexion
iptables -A INPUT -s 192.168.0.4 -j ACCEPT
iptables -A INPUT -s 192.168.0.0/24 -j ACCEPT

# Autorise le ping
iptables -A INPUT -p icmp -j ACCEPT
iptables -A OUTPUT -p icmp -m state --state NEW,ESTABLISHED,RELATED -j ACCEPT

# Refuse les autres paquets entrants
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT

# Affiche les règles
iptables -L -v
iptables -L --line-numbers

# Supprime le règle line-number
iptables -D [INPUT | OUTPUT] line-number
```

Visualisation des règles : **iptables-save**
Sauvegarde des règles dans un fichier : `iptables-save -c > ~/iptables-save`
Restauration des règles : `iptables-restore -c < ~/iptables-save`

Ou via la commande `service iptables save` et `service iptables reload`

B. Routages

Effectuer un routage de l'interface LAN vers la passerelle internet

```
# Interface connected to Internet
INTERNET="eth0"
# Interface connected to LAN
LAN_IN="eth1"
# DMZ
ZONE_DMZ=172.16.0.0/255.255.255.0

# IP redirection
echo 1 > /proc/sys/net/ipv4/ip_forward

# IP Masquerading
iptables -t nat -A POSTROUTING -o $INTERNET -j MASQUERADE

iptables -A FORWARD -i $INTERNET -o $LAN_IN -m state --state RELATED,ESTABLISHED -
j ACCEPT
iptables -A FORWARD -i $LAN_IN -o $INTERNET -j ACCEPT

# HTTP route from internet to local web server DMZ
iptables -t nat -A PREROUTING -i eth0 -p tcp --dport 80 -j DNAT --to 172.16.0.1:80
```

Visualisation des translations : **iptables -L -t nat -v**

C. FirewallD

La gestion de netfilter / iptables peut s'évérer compliquée, et des alternatives comme UFW (Uncomplicated Firewall) existent (Debian, Ubuntu…). Cependant, firewalld est aussi une bonne alternative, et est activé par défaut sur RedHat, Rocky Linux, Fedora Il peut aussi être installé sur debian. Firewalld fonctionne sur un principe de zones, comparable aux pare-feux professionnels.

1) Les règles de base

Gestion du service :

```
systemctl start firewalld.service
systemctl stop firewalld.service
firewall-cmd --state
```

Liste des services par défaut :

```
firewall-cmd --get-services
```

Exemple de règles temporaires :

```
firewall-cmd --add-service=http
firewall-cmd --remove-service= http

firewall-cmd --runtime-to-permanent
firewall-cmd --reload
```

Exemple de règles définitives :

```
firewall-cmd --permanent --add-service=smtp
firewall-cmd --permanent --remove-service=smtp
firewall-cmd --permanent --add-port=53/udp
firewall-cmd --permanent --add-port=53/tcp
firewall-cmd --permanent --remove-service=ssh     !!!!
```

Visualiser les règles configurées :

```
firewall-cmd --list-services
```

```
firewall-cmd --list-ports
firewall-cmd --list-all
```

Validation des règles :

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

2) Les zones

```
firewall-cmd --get-zones
firewall-cmd --get-default-zone
firewall-cmd --get-active-zones

firewall-cmd --zone=internal --change-interface=eth0 --permanent
firewall-cmd --zone=external --change-interface=eth1 --permanent
firewall-cmd --set-default-zone=internal
firewall-cmd --zone=internal --list-all
firewall-cmd --zone=external --list-all
firewall-cmd --list-all-zones
firewall-cmd --zone=internal --add-service=https --permanent
firewall-cmd --zone=internal --add-service=http --permanent
firewall-cmd --zone=internal --add-service=dns --permanent

firewall-cmd --zone=internal --add-service=smtps --permanent
firewall-cmd --zone=internal --add-service=imaps --permanent
firewall-cmd --zone=internal --add-port=993/tcp --permanent
firewall-cmd --zone=internal --add-port=465/tcp --permanent

firewall-cmd --zone=internal --add-service=ntp --permanent
firewall-cmd --zone=external --add-service=ftp --permanent
```

3) Translation de ports

```
firewall-cmd                    --zone=external                    --add-forward-
port=port=8080:proto=tcp:toaddr=172.21.21.10:toport=80 -permanent
```

4) La journalisation

```
firewall-cmd --set-log-denied=all
firewall-cmd --get-log-denied
journalctl -x -e
```

5) Interface graphique

```
firewall-config
```

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Security_Guide/sec-Using_Firewalls.html

S.S.H.
(SECURE SHELL)

Création des clés privée & publiques

Sur l'ordinateur local :

```
[root@local]# ssh-keygen -b 2048 -t rsa

[root@local]# ~/.ssh/id_rsa : clef privée
[root@local]# ~/.ssh/id_rsa.pub : clef publique
```

Transférer la clé publique sur le serveur distant :

```
[root@local]# ssh-copy-id -i ~/.ssh/id_rsa.pub root@distant.com:
```

Sécuriser l'accès à la clé :

```
[root@distant]# rm id_rsa.pub
[root@distant]# chmod 700 .ssh
[root@distant]# chmod 600 .ssh/authorized_keys
```

Sur ce même serveur, vérifier les options de /etc/ssh/sshd_config

```
RSA Authentication yes
PubkeyAuthentication yes
```

Copie vers distant : **scp -r *dossier user@hostname*:**
Copie vers local : **scp -r *user@hostname:dossier***

PS : SSH permet aussi d'encapsuler des protocoles (tunneling) et de faire du Xforwarding.

Se référer aux recommendations de l'ANSSI :

Recommendations pour un usage securisé d'openssh

SYSLOGD

Les fichiers de logs systèmes se configurent dans le fichier /etc/syslog.conf
La rotation et l'archivage des logs se paramètre dans /etc/rotatelog.conf

A. Catégories des messages :

« auth » : les messages d'authentification et de sécurité
« authpriv » : les mêmes messages, mais à caractère plus confidentiel
« cron » : les messages des planificateurs de taches (cron et at)
« daemon » : les messages des démons ne disposant pas d'une catégorie dédiée
« ftp » : les messages du serveur ftp
« kern » : les messages du noyau
« lpr » : les messages du serveur d'impression
« mail » : les messages du système de messagerie
« news » : les messages du serveur de newsgroups
« syslog » : les messages internes de syslog

B. Importance des alertes :

0 : **emerg** : situation d'urgence rendant le système inutilisable
1 : **alert** : situation critique nécessitant une intervention immédiate
2 : **crit** : situation critique
3 : **err** : condition d'erreur
4 : **warning** : simple avertissement
5 : **notice** : message d'information
6 : **info** : message d'information à caractère moins important que « notice »
7 : **debug** : message de debuggage

Créer une entrée 'urgente' dans syslog avec la commande **logger** :
logger –p *mail.emerg* –t *PIERRE* « *message* »

Le moment de rotations des logs systèmes sont paramétrables dans :

Sous linux : **/etc/crontab**
Sous Solaris : la ligne qui exécute /usr/sbin/logadm dans la crontab de root.

VI

`:e` file : ouvre ou créé un fichier

`0` (zéro) début de ligne
`$` fin de ligne
`[CTRL] f` : page suivante
`[CTRL] b` : page précédente
`G` dernière ligne du fichier
`gg` début de fichier

`w` mot suivant
`b` mot précédent
`nG` ou `:n` : saut à la ligne n

`h` déplace le curseur vers la gauche
`j` déplace le curseur vers le bas
`k` déplace le curseur vers le haut
`l` déplace le curseur vers la droite

`a` : append
`i` : insert
`A` : ajout fin de ligne
`I` : insert début de ligne
`o` : insertion ligne suivante
`O` : insertion ligne précédente

x : suppression caractère
r : remplacer caractère
cw : change word
dw : delete (or cut) word
c$: changer la fin de ligne
yy : copy current line
10yy : copy 10 lines
dd : delete (or cut) current line
5dd : delete (or cut) 5 lines
3,7dd : efface lignes 3 à 7
2cc : change 2 lignes

p : paste

:g!/linux/d : efface toutes les lignes ne contenant pas linux
:g/^[ ]*$/d : supprime toutes les lignes vides
:g/*microsoft*/s/*windows*/linux/g : change windows par linux uniquement dans les lignes ou "microsoft" se trouve

`:right` ou `:left` : alignement à droite ou à gauche

`:sh` lance un shell
exit ou ^d pour revenir

`:se all` : affiche toutes les variables
fichier .exrc contient ces variables

Rechercher :
`/chaîne` : recherche chaîne vers la fin
`?chaîne` : recherche vers le début
`n` : répète recherche vers fin
`N` : répète vers le début

Remplacer :
`:s/`*`m1`*`/`*`m2`* : remplace la première occurrence de m1 par m2, sur la ligne en cour
`:%s/`*`m1`*`/`*`m2`* : remplace la première occurrence de m1 par m2, sur la ligne en cour et les suivantes
`:10,20s/`*`m1`*`/`*`m2`*`/g` : remplace le texte m1 par m2 des lignes 10 à 20 sur toute la ligne
`:1,$s/`*`m1`*`/`*`m2`*`/gc` : remplace m1 par m2 de toutes les lignes, avec confirmation

`.` répétition de la commande
`u` undo

`:set nu` / `:set nonu` : numéros de ligne

`:5,15w` fichier : crée nouveau fichier avec les lignes 5 à 15

`:r!ls` : insère le résultat de la commande ls
`:r` *`fichier`* : insère le fichier

ESC `:q` quit
ESC `:q!` quit without save
ESC `:w` write buffer to disk

METRIQUES ET DIAGNOSTICS

Voici quelques commandes utiles à des fins d'analyses. Certaines ne font pas partie du package par défaut, et doivent être installées.

A. Le système

Configuration des composants physiques :

```
[root@RockyLinux ~]# lshw
```

Configuration des périphériques en mode block :

```
[root@RockyLinux ~]# lsblk
```

Information sur les bus USB :

```
[root@RockyLinux ~]# lsusb
```

Information sur les bus PCI :

```
[root@RockyLinux ~]# lspci
```

Information des comptes système :

```
[root@RockyLinux ~]# lslogins
```

Etat des modules chargés dans le kernel :

```
[root@RockyLinux ~]# lsmod
```

B. CPU

Le premier chiffre correspond à l'intervalle entre les diagnostiques, le second représente le nombre d'occurrences.

```
[root@RockyLinux ~]# mpstat -P ALL 2 5
```

Le rafraichissement et la mise en valeur des informations

```
[root@RockyLinux ~]# watch -d mpstat -P ALL
```

```
[root@RockyLinux ~]# watch sar -u 1 1
```

C. Réseau

La commande ss (Socket Statistics) :

```
[root@localhost ~]# ss -s
```

```
[root@localhost ~]# netstat -antup

[root@localhost ~]# mtr -b pierreau.fr

[root@RockyLinux ~]# watch -d sar -n DEV 1 1

[root@RockyLinux ~]# watch -n1 --differences cat /proc/net/dev

 [root@RockyLinux ~]# tcpdump -i ens33 'not port 80 and not port 443 and not
udp and (dst 192.168.1.137 and src 192.168.1.254)'

[root@RockyLinux ~]# yum install iptstate

[root@RockyLinux ~]# yum install iftop
[root@RockyLinux ~]# iftop -P -i ens33

[root@RockyLinux ~]# yum install nmap
[root@RockyLinux ~]# nmap -O localhost
[root@RockyLinux ~]# nmap -p 22 pierreau.fr

[root@RockyLinux ~]# dnstop ens33

[root@RockyLinux ~]# jnettop -i ens33

[root@RockyLinux ~]# yum install iptraf-ng
...
```

D. Mémoire

```
[root@RockyLinux ~]# free -h

[root@RockyLinux ~]# watch -u sar -r 1 1

[root@RockyLinux ~]# vmstat -aS M
```

E. Stockage

Disques :

```
[root@RockyLinux ~]# watch -d -p sar -d 1 1

[root@RockyLinux ~]# watch -d iostat -x sda 1 1

[root@RockyLinux ~]# yum install sdparm
[root@RockyLinux ~]# sdparm /dev/sda
```

smartctl : installer le package smartmontools

```
[root@RockyLinux ~]# smartctl -a /dev/sda
```

Swap :

```
[root@RockyLinux ~]# sar -S 1 1

[root@RockyLinux ~]# sar -W 1 1
```

F. Process

pidstat :

```
[root@RockyLinux ~]# dnf install http://mirror.RockyLinux.org/RockyLinux/8/AppStream/x86_64/os/Packages/sysstat-11.7.3-2.el8.x86_64.rpm
[root@RockyLinux ~]# watch -d pidstat

[root@RockyLinux ~]# watch -d pidstat -p process 1 1
```

htop :

```
[root@RockyLinux ~]# yum install epel-release
[root@RockyLinux ~]# dnf install htop
```

Temps d'execution et ressources

```
[root@localhost ~]# time tar cf /tmp/var.tar /var
real    0m3.261s
user    0m0.020s
sys     0m1.132s

[root@localhost ~]# time tar czf /tmp/var.tgz /var
real    0m24.828s
user    0m20.714s
sys     0m1.531s
```

G. Stress tests

Attention à ne pas utiliser ces commandes sur un serveur de production...

1) CPU

```
[root@RockyLinux ~]# dd if=/dev/zero of=/dev/null status=progress
```

2) Stockage

Lecture :

```
[root@RockyLinux ~]# cat /dev/sda > /dev/null
```

Ecriture de 10 Go sur disque :

```
[root@RockyLinux ~]# dd if=/dev/zero of=/tmp/test.io bs=4096 count=10000000 status=progress
```

3) Réseau

Côté serveur :
```
[root@RockyLinux ~]# iperf -s -w 10MB
```

Côté client :
```
[root@RockyLinux ~]# iperf -c IPClient -w 10MB -d -t 20 -P 10
```

4) Autres outils

```
[root@RockyLinux ~]# yum install stress
[root@RockyLinux ~]# stress -c 4 -t 60s
[root@RockyLinux ~]# stress --cpu 8 --io 4 --vm 2 --vm-bytes 128M --timeout
10s
```

SECURITE & RECOMMANDATIONS

La technique à elle seule ne suffit pas à sécuriser un système. Certains process et recommandations contribue à rendre moins vulnérable les systèmes.

- Suivre les recommendations de l'ANSSI (Agence nationale de la sécurité des systèmes d'information) :

Exemples de recommendations :
- ANSSI : recommendations sécurité systèmes GNU / Linux
- ANSSI : guide protection systemes essentiels
- … autres bonnes pratiques

- S'abonner aux alertes du CERT (Computer emergency response team)
- Evaluer les risques selon leurs indices CVSS
- Adopter un SMSI (système de management de la sécurité de l'information), avec l'implémentation de la norme ISO 27001

Consulter et adopter les bonnes pratiques du site Cyber malveillance :

Exemple : évaluer le niveau de cybersécurité de votre site Internet

LIENS

Linux / Unix :

- LinuxFr : http://linuxfr.org
- linuxfr.org : https://linuxfr.org/
- Freecode : http://freecode.com/
- Slashdot : https://slashdot.org
- Planet Libre : http://www.planet-libre.org
- Forum Fedora : http://forums.fedora-fr.org/
- Forum Debian : https://www.debian-fr.org/
- Forum Ubuntu : http://forum.ubuntu-fr.org
- Forum OpenSUSE Alionet : https://www.alionet.org
- OpenBSD : https://www.openbsd.org/
- Debian : http://www.debian.org/index.fr.html
- Gentoo : http://www.gentoo.fr/install/

- Lea Linux : http://lea-linux.org
- The Linux Documentation Project : http://tldp.org
- LinuxDocs : http://linuxdocs.org

- Distro watch : https://distrowatch.com/

Applications tierces :

Webmin (services d'administration simplifiée) : http://www.webmin.com/

PhpMyAdmin (gestion MySQL) : https://www.phpmyadmin.net/

FirewallBuilder : http://fwbuilder.sourceforge.net/

Wireshark (sniffeur de paquets) : https://www.wireshark.org/

Inventory management :
- http://fusioninventory.org/
- https://www.ocsinventory-ng.org/
- https://glpi-project.org/

Apache :

- https://httpd.apache.org/

PHP ~ MySQL :

- http://www.manuelphp.com/

Fog project (déploiement d'images) : https://fogproject.org/