

Linux General Cheat Sheets

Information système

Commande	Commentaire
uname -a	Afficher les informations systèmes de linux
uname -r	Afficher la version du noyau
uptime	Afficher le temps d'activité du système et affiche la charge
hostname	Affiche le nom d'hôte de la machine
hostname -I	Affiche l'adresse IP de l'hôte
last reboot	Afficher l'historique des redémarrages
date	Afficher la date et l'heure du système
cal	Affiche le calendrier du mois
w	Affiche qui est en ligne
whoami	Affiche où nous sommes loggué en tant que qui

Information matérielle

Commande	Commentaire
dmesg	Affiche les messages du noyau
cat /proc/cpuinfo	Affiche les informations du CPU
cat /proc/meminfo	Affiche les informations de la RAM
free -h	Affiche la mémoire libre et utilisé (-h pour lisible par l'homme, -m for MB et -g pour GB)
lspci -tv	Affiche les périphériques PCI
lsusb -tv	Affiche les périphériques USB
dmidecode	Affiche les information DMI/SMBIOS (informations systèmes) depuis le BIOS

Commande	Commentaire
hdparm -i /dev/sda	Affiche les informations du disque /dev/sda
hdparm -rT /dev/sda	Faire un test de vitesse de lecture sur le disque /dev/sda
badblocks -s /dev/sda	Test le disque /dev/sda pour d'éventuels blocks défectueux

ls - Lister le contenu d'un répertoire

Permet de lister le contenu d'un répertoire

Syntaxe :

```
ls <option> répertoire
```

Option	Commentaire
-a	Lister tout le répertoire (y compris fichier caché)
-l	Afficher le répertoire sous forme de tableau, avec permission, ...
-i	Affiche les inodes
-h	Affiche la taille dans un format lisible par l'homme (Mo par exemple)
-R	Liste également les sous-répertoires
-s	Affiche la taille des répertoires

pwd - Afficher le répertoire courant

Affiche le répertoire dans lequel on se situe.

Syntaxe :

```
pwd
```

mkdir - Créer un répertoire

Créer un répertoire.

```
mkdir -p /chemin/répertoire/à/créer
```

Option	Commentaire
-p	Créer les répertoires parents si ces derniers n'existent pas

Exemple : Je souhaite créer un répertoire **truc** dans `/home/user1/test1`, mais le fichier `test1` n'existe pas.

Syntaxe :

```
mkdir -p /home/user1/test1/truc
```

cd - Changer de répertoire

Permet de se déplacer dans l'arborescence

Syntaxe :

```
cd /répertoire/de/destination
```

cat - Afficher le contenu d'un fichier

Permet d'ouvrir un fichier et d'afficher son contenu.

Syntaxe :

```
cat <option> fichier
```

Option	Commentaire
-b	Numéroter toutes les lignes non vides
-n	Numéroter toutes les lignes

df - Afficher la taille d'un répertoire

Permet d'afficher la taille d'un répertoire.

Syntaxe :

```
df <option> répertoire
```

Option	Commentaire
-h	Permet d'obtenir un résultat plus lisible pour un humain (ex <i>Mo</i> , <i>Ko</i> ,...)
-i	Affiche les inodes
-k	Affiche le résultat en kilobytes
-m	Affiche le résultat en megabytes
-d n .	Affiche la taille des sous-répertoires jusqu'au n ème

mv - Déplacer un fichier ou dossier

Permet de déplacer un fichier ou un répertoire.

Syntaxe :

```
mv <option> /chemin/source /chemin/destination
```

Option	Commentaire
-f	Forcer le déplacement
-i	Demander la confirmation de l'utilisateur

rm - Supprimer un fichier ou dossier

Permet de supprimer un fichier ou un dossier

Syntaxe :

```
rm <option> /chemin/truc/a/supprimer
```

Option	Commentaire
-d	Efface un répertoire
-f	Force la suppression
-i	Demande confirmation à l'utilisateur (<i>Inutile avec <code>-f</code></i>)
-r	Récuratif

groups - Afficher les groupes d'appartenance d'un utilisateur

Permet d'afficher dans quels groupes se trouve l'utilisateur

Syntaxe :

```
groups utilisateur
```

passwd - Changer le mot de passe

****Permet de changer le mot de passe d'un utilisateur**

Syntaxe :

```
passwd <option> utilisateur
```

Option	Commande
-d	Supprimer le mot de passe
-e	Faire expirer le mot de passe
-i	Rendre un compte inactif

Option	Commande
-l	Verouille le mot de passe et empêche sa modification par l'utilisateur
-S	Affiche le status du compte
-u	Déverouille un mot de passe

Faire un rechercher remplacer récursif :

Remplacer "texte1" par "texte2".

```
find . -name "*" -exec sed -i 's/texte1/texte2/g' {} \;
```

Rechercher une chaine présente dans des fichiers d'un dossier de manière récursive :

```
grep -rnw /PATH/TO/THE/FOLDER -e 'MA-CHAÎNE'
```

Répéter une commande en boucle :

```
watch macommande [options]
```

Couper le début d'une réponse

```
cat /mon/fichier | cut -c 19-
```

--> On retire les 18 premiers chars

Envoyer une commande à un autre utilisateur :

```
su - UTILISATEUR -c "MA COMMANDE"
```

Mettre à jour à la fois les paquets mais aussi la version de la distribution :

```
apt-get full-upgrade
```

Lister les IP D'un CIDR :

```
nmap -sL -n 10.10.64.0/27 | awk '/Nmap scan report/{print $NF}'
```

Résultat :

```
10.10.64.0
10.10.64.1
10.10.64.2
10.10.64.3
10.10.64.4
10.10.64.5
10.10.64.6
10.10.64.7
10.10.64.8
10.10.64.9
10.10.64.10
10.10.64.11
10.10.64.12
10.10.64.13
10.10.64.14
10.10.64.15
10.10.64.16
10.10.64.17
10.10.64.18
10.10.64.19
10.10.64.20
10.10.64.21
10.10.64.22
10.10.64.23
10.10.64.24
10.10.64.25
10.10.64.26
10.10.64.27
10.10.64.28
10.10.64.29
10.10.64.30
10.10.64.31
```

Supprimer les dossier datant de plus de quinze jours :

```
DIR=/backup  
find $DIR -type d -ctime +15 -exec rm -rf {};
```

Savoir le load du proc :

```
cat /proc/loadavg
```

Installer un environnement graphique :

```
apt-get install task-lxde-desktop
```

Ajouter un utilisateur à la liste des Sudoers :

Avec l'utilisateur Root exécutez la commande ci-dessous :

```
sudo usermod -a -G sudo <username>
```

L'utilisateur peut désormais faire la commande sudo !

Mettre une adresse IP fixe sur Linux :

On modifie le fichier situé dans /etc/network/interfaces :

```
nano /etc/network/interfaces
```

```
auto eth1  
iface eth1 inet static  
    address 192.168.0.42  
    network 192.168.0.0  
    netmask 255.255.255.0  
    gateway 192.168.0.1
```

Puis on redémarre le service :

```
service networking restart
```

L'IP est désormais fixe !

Grep toutes les IPS d'un fichier

```
cat mon_super_fichier | grep -E '[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}'
```

Savoir son adresse IP d'une manière lisible :

```
ip -br a
```

Savoir la taille totale de RAM non-utilisée :

```
free
```

Détails de la RAM :

```
sudo dmidecode -t 17
```

Savoir le nombre de processeurs :

```
nproc
```

Détails du CPU :

```
lscpu
```

Changer le fuseau d'horaire :

```
dpkg-reconfigure tzdata
```

Changer le fuseau horaire sans interaction :

```
In -fs /usr/share/zoneinfo/Europe/Paris /etc/localtime  
dpkg-reconfigure --frontend noninteractive tzdata
```

Générer une chaine aléatoire :

```
< /dev/urandom tr -dc _A-Z-a-z-0-9 | head -c${1:-32};echo;
```

Filtrer le résultat d'une commande : Récupérer la X ième collone

docker ps

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
b47869c06afa	local_ansible:latest	"tail -F anything"	4 days ago	Up 4 days		
XXXXXXXXXX.1.xkdfozjesx3i9tiiwi7v4mx2						
98d251874810	local_ansible:latest	"tail -F anything"	4 days ago	Up 4 days		
XXXXXXXXXX.1.30q690xvddk600wt35tuii2qr						

```
e59db4bf005e local_ansible:latest "tail -F anything" 4 days ago Up 4 days
XXXXXXXXXX.1.wf6mpbnyl3fmvvl1dm6rvgx7q
d5bb4b393058 local_ansible:latest "tail -F anything" 4 days ago Up 4 days
XXXXXXXXXX.1.ug7xuyrdx61jz2hyir01i85dl
c7e24c57b1c3 local_ansible:latest "tail -F anything" 4 days ago Up 4 days
XXXXXXXXXX.1.xgn2wopz174n4vsq8sfu1hbuz
```

`docker ps | awk '{print $1}'`

```
CONTAINER
b47869c06afa
98d251874810
e59db4bf005e
d5bb4b393058
c7e24c57b1c3
```

\$1 représente la première colonne ; pour la suivante ce sera \$2 etc.

Lister les processus :

```
ps aux
```

KILL un processus :

```
sudo kill -9 ID
```

Faire un SSH via un proxy :

```
# Forward le trafic sur le port 8080 de mon ordinateur vers le port 80 de l'hôte distant
ssh -L 8080:localhost:80 remote_host
```

Afficher les logs du boot précédent et ainsi voir à la toute fin, les logs de la dernière extinction du système.

```
journalctl -b -1
```

Actualiser le /etc/fstab :

```
sudo mount -a
```

Couper un affichage à partir d'un caractère :

```
cat monfichier.log | cut -d "|" -f 2
```

Journalctl purge :

- Per days

```
journalctl --vacuum-time=2d
```

- Per size

```
journalctl --vacuum-size=500M
```

--> Après purge des fichiers de logs, pour libérer l'espace (uniquement si on supprime rsyslog ou daemon.log):

```
systemctl restart rsyslog
```

Taille dossier d'un niveau uniquement :

```
du -h --max-depth=1
```

Supprimer un hostname des know_host ssh :

```
ssh-keygen -R hostname
```

Lister les services actuellement fonctionnels :

```
sudo systemctl --type=service --state=running | awk '{print $1}'  
UNIT  
chrony.service  
containerd.service  
cron.service  
dbus.service  
docker.service  
getty@tty1.service  
google-cloud-ops-agent-diagnostics.service  
google-cloud-ops-agent-fluent-bit.service  
google-cloud-ops-agent-opentelemetry-collector.service
```

```
google-guest-agent.service
google-osconfig-agent.service
haveged.service
mariadb.service
nmbd.service
polkit.service
rsyslog.service
serial-getty@ttyS0.service
smbd.service
ssh.service
systemd-journald.service
systemd-logind.service
```

Certificats :

Etablir une connexion ssl pour tester les certificats :

```
openssl s_client -connect host:port -CAfile chemin_vers_certificat_public_de_la_ca.pem
```

Debug Deep :

Afficher les logs du noyau linux :

```
dmesg
```

Kernel :

--> On va lister les vieux kernels présents sur le system. Puis on va les supprimer pour faire de l'espace dans /boot.

Kernel actuellement en cours d'execution :

```
uname -r
```

```
X.X.X-26-amd64
```

J'utilise donc le kernel : X.X.X-26-amd64

Liste des kernels présents dans le système (exépté le kernel actuellement utilisé) :

```
sudo dpkg --get-architecture | awk '{ if ($1=="i386") print $2 }' | grep -v `uname -r`
```

```
linux-image-X.X.X-18-amd64
linux-image-X.X.X-19-amd64
linux-image-X.X.X-20-amd64
linux-image-X.X.X-21-amd64
linux-image-X.X.X-22-amd64
linux-image-X.X.X-23-amd64
```

Ici, nous allons pouvoir supprimer les versions **X.X-18-amd64** à **X.X-23-amd64** compris. Il faut toujours garder deux versions de kernels précédentes.

Pour supprimer les anciens kernels, je vais réaliser les commandes suivantes :

```
sudo apt purge linux-image-X.X.X-18-amd64
sudo apt purge linux-image-X.X.X-19-amd64
sudo apt purge linux-image-X.X.X-20-amd64
sudo apt purge linux-image-X.X.X-21-amd64
sudo apt purge linux-image-X.X.X-22-amd64
sudo apt purge linux-image-X.X.X-23-amd64
sudo apt purge linux-image-X.X.X-24-amd64
```

Puis, je vais réaliser une purge des packets qui sont désormais obsolètes :

```
sudo apt autoremove
```

Enfin, on met à jour la liste des noyaux grub :

```
sudo update-grub
```

Conclusion :

Mon /boot est passé de 427 Mo utilisés à 168 Mo.

TCPDUMP :

ARP:

```
sudo tcpdump -nni any vrrp
```

--> Sur toutes les interfaces.

Récupérer un fichier perdu, supprimé :

Foremost permet de récupérer les fichiers supprimés. Cette récupération n'est pas parfaite car les données perdues peuvent être écrasés par une réécriture :

```
foremost -t all -i /dev/sda1
```

Gestion du disque linux (LVM) :

Afficher les volumes physiques :

```
pvdisplay
```

Afficher le groupe :

```
vgdisplay
```

Afficher les volumes logiques :

```
lvdisplay
```

Augmenter la taille d'un volume :

```
# Taille définie :  
lvextend -L '+9G' /dev/vg0/lib  
  
# Taille relative :  
lvextend -l '+100%FREE' /dev/vg0/lib
```

Affecter la nouvelle taille :

```
xfs_growfs /var/lib
```

Supprimer un volume récalitrant :

```
# Eteindre le volume.  
lvchange -an -v /dev/vg0/lib
```

```
# Afficher les process qui utilisent le volume
```

```
lsdf | grep /var/lib
```

```
# Si besoin de debug : Afficher les processus cachés qui utilisent le volume
```

```
grep -l /var/lib /proc/*/mountinfo
```

```
# --> Cette commande nous retourne des PID. Faire : ps -aux | grep <PID_ID> ; pour connaître les services qui continuent d'utiliser le volume.
```

```
# Maintenant que nous avons éteints tous les services qui posent problème, nous pouvons supprimer le volume.
```

```
lvremove -f vg0/lib
```

```
lvcreate -n lib -l 100%FREE vg0
```

```
mkfs.xfs /dev/vg0/lib
```

Procédure pour augmenter la SWAP :

```
free -h
```

```
swapoff /dev/vg0/swap
```

```
lvextend -L '+6G' /dev/vg0/swap
```

```
mkswap /dev/vg0/swap
```

```
swapon /dev/vg0/swap
```

```
free -h
```

Procédure pour supprimer un volume "100%FREE" qui nous empêche forcément de créer un volume /home de 20 Go :

```
# Démonter /var/lib ; sauvegarder ; préparer au redémarrage
```

```
cp -pR /home /root/
```

```
cp -pR /var/lib /root/
```

```
umount /var/lib
```

```
cp -pR /root/lib/* /var/lib/
```

```
#####
```

```
#####
```

```
COMMENTER LE /ETC/FSTAB (ligne lib) :
```

```
#/dev/vg0/lib    /var/lib    xfs    rw,noatime,logbufs=8,logbsize=256k,inode64    1    2
```

```
#####
```

```
#####
```

```
Reboot
```

```
#####
```

```
#####
```

```
# Supprimer le volume
```

```
lvremove -f vg0/lib
```

```
#####
```

```
# Créer les volumes qu'il faut
```

```
lvcreate -n home -L "20G" vg0
```

```
mkfs.xfs /dev/vg0/home
```

```
#...
```

```
lvcreate -n lib -l 100%FREE vg0
```

```
mkfs.xfs /dev/vg0/lib
```

```
#####
```

```
# supprimer les données résiduelles avant remontage
```

```
rm -r /var/lib/*
```

```
rm -r /home/*
```

```
#####
```

```
DECOMMENTER LE /ETC/FSTAB !
```

```
+
```

```
AJOUTER LES MONTAGES neccessaires :
```

```
/dev/vg0/lib    /var/lib    xfs    rw,noatime,logbufs=8,logbsize=256k,inode64    1    2
```

```
/dev/vg0/home    /home    xfs    rw,noatime,logbufs=8,logbsize=256k,inode64    1    2
```

```
#####
```

```
mount -a
```

```
# VERIFIER
```

```
mount | grep /home
```

```
mount | grep /var/lib
```

```
# Réimporter les datas dans les volumes
```

```
cp -pR /root/lib/* /var/lib/
```

```
cp -pR /root/home/* /home/
```

```
# Reboot
```

```
reboot
```

```
# vérification
```

```
ls -lisa /var/lib
```

```
ls -lisa /home
```

```
# suppression des données dupliquées
```

```
rm /root/home -r
```

```
rm /root/lib -r
```

SSHD - Only sftp

Prérequis :

- home de l'user toto : /opt/toto

Configuration /etc/ssh/sshd_config :

```
Match User toto
```

```
ChrootDirectory /opt/toto
```

```
X11Forwarding no
```

```
AllowTcpForwarding no
```

```
AllowAgentForwarding no
```

```
PasswordAuthentication no
```

```
ForceCommand internal-sftp -d /%u
```

Commandes pour rendre la chose fonctionnelle :

```
mkdir /opt/toto/data
chown root:root /opt/toto
chown -R toto:toto /opt/toto/*
systemctl restart sshd
```

Test :

```
sftp -i id_rsa toto@SERVEUR <<< 'put fichier.txt /data/'

# Test de perms
ssh -i id_rsa toto@SERVEUR
This service allows sftp connections only.
Connection to SERVEUR closed.
```

Gérer les ACL

Récupérer les informations d'un fichier :

```
stat launch_kee.sh
=====
Fichier : launch_kee.sh
  Taille : 82      Blocs : 8      Blocs d'E/S : 4096  fichier
Périphérique : 254/1 Inœud : 4718858  Liens : 1
Accès : (0755/-rwxr-xr-x) UID : (  0/  root) GID : (  0/  root)
Accès : 2024-06-10 11:24:44.881570498 +0200
Modif. : 2024-06-10 11:24:33.865592246 +0200
Changt : 2024-06-10 11:24:41.201577761 +0200
Créé : 2024-06-10 11:24:29.949599980 +0200
```

Récupérer les ACL d'un fichier :

```
getfacl launch_kee.sh
=====
# file: launch_kee.sh
# owner: root
```

```
# group: root
user::rwx
group::r-x
other::r-x
```

Ajouter les perms à un utilisateur :

```
sudo setfacl -m user:nehemie:rw- launch_kee.sh
```

Constat :

```
getfacl launch_kee.sh
=====
# file: launch_kee.sh
# owner: root
# group: root
user::rwx
user:nehemie:rw-
group::r-x
mask::rwx
other::r-x
```

Supprimer les perms à un utilisateur :

```
sudo setfacl -Rm user:nehemie:--- launch_kee.sh
```

- - R : ACL Récursif

Constat :

```
getfacl launch_kee.sh
=====
# file: launch_kee.sh
# owner: root
# group: root
user::rwx
user:nehemie:---
group::r-x
mask::r-x
other::r-x
```

Réduire la valeur des ports privilégiés:

```
echo 80 | sudo tee /proc/sys/net/ipv4/ip_unprivileged_port_start
```

Lister les démons qui utilisent le dossier :

```
sudo lsof /var/lib/
```

Convertir des timestamps milisencondes en date humaines :

```
# dans mon fichier tmp j'ai des lignes de timestamp en format ms  
for line in $(cat /var/lib/db_backup/tmp); do date -ud @$(($line / 1000)); done;
```

Activer la gestion du temps via vmware :

```
vmware-toolbox-cmd timesync enable
```

Installer un packet dans un conteneur debian bizzare :

```
apk add curl  
apk add jq
```

Revision #4

Created 24 November 2024 17:01:23 by Nicolas

Updated 13 February 2025 21:15:27 by Nicolas