Système d’exploitation – Lexique – LINUX

sudo => demande le mot de passe (mdp) pour travailler exécuter les commandes en mode Administrateur

sudo -i => permet de rester en mode administrateur

exit => pour quitter le mode administrateur

clear => efface l’host du terminal

man => manuel d’explication sur une commande

man man => explique la commande ‘man’

man -f man => affiche la ‘table des matières’ du manuel expliquant la commande ‘man’

man 7 man => va afficher la « page » 7 du manuel expliquant la commande ‘man’

whatis man ⬄ man -f man

mandb => mise à jour du manuel

--help => permet d’afficher une explication plus brève que le manuel

man --help

pacman => package management utility => gère l’installation, la mise à jour et la désinstallation de programmes pour Linux

pacman -Q => liste tous les ‘packages’ installés

pacman -S 'nom\_du\_package’ => met à jour le package voulu (et si la package voulu n’est pas présent sur la machine, il va l’installer => attention doit se faire ne mode administrateur =>

sudo pacman -S fish

bash et fish => 2 interpreteurs possible pour le terminal

chsh => va forcer le démarrage du terminal avec un interpréteur précis

sudo chsh => va forcer le démarrage du termial avec un interpréteur précis pour le root !

/bin/bash = > chemin pour l’interpréteur bash

/bin/fish => chemin pour l’interpréteur fish

apropos pacman => va afficher une liste des commandes avec explications contenant le terme ‘pacman’ => attention => la recherche du terme ‘pacman’ s’effectue dans les commandes ET dans les explications !

history => affiche l’historique de toutes les dernières commandes réalisées depuis l’ouverture du terminal

history search man => va afficher l’historique de toutes le commandes contenant ‘man’

id => va afficher les droits

id votre\_user => va afficher vos droits d’accès

groups => va afficher tous les groupes

groups votre\_user => va afficher tous les groupes auquel vous appartenez

useradd => crée un nouvel utilisateur

useradd -c 'le petit toto’ toto => crée l’utilisateur toto avec comme commentaire ‘le petit toto’

passwd => change mon mot de passe

passwd toto => va changer le mot de passe de l’utilisateur toto

su => va changer d’utilisateur => si ‘su’ utilisé seul, va changer pour le root

su toto => switch vers l’utilisateur toto

usermod => permet de modifier un utilisateur

groupmod => permet de modifier les groupes

userdel toto => supprime l’utilisateur toto

who => qui est connecté

uptime => affiche depuis combien de temps le système est en route

w => montre qui est connecté et qu’est-il entrain de faire

whoami => qui suis-je ?

date => donne la date du jour

date -d 20140324 => va afficher cette date

date -u => date heure actuelle de Greenwitch

cal => va afficher le calendrier

cal avril 2015 => va afficher le calendrier d’avril 2015

cal -n12 => va afficher les 12 mois calendaire

cat => va lire un fichier

tac => va lire un fichier ligne par ligne à l’envers

yay => même fonctionnement que pacman (ou presque)

chemin absolu => commence par / => par exemple /home/dpi/…

reprend un chemin depuis la racine /

chemin relatif => chemin à partir d’où on est « physiquement » dans le terminal

~ => répertoire utilisateur

.. => remonte d’un niveau dans l’arborescence

ls : liste le contenu de l’emplacement où l’on se trouve

ls /home/dpi/Bureau => liste le contenu de l’emplacement ciblé

ls -l => liste + complète

ls -a => liste également les fichiers cachés (.caché)

ls -A => liste également les fichiers cachés sauf le fichier . et le ..

ls -lah => change le poid en ko ou mo ou…

ls -R => liste les fichiers et les fichiers enfants… etc

cd => permet de changer le chemin d’où l’on se trouve

cd /home/dpi/Bureau => nous envoie dans le bureau

cd - => nous envoie au dernier emplacement cité précédemment

mkdir : crée un dossier

mkdir lolilol => va créer le dossier lolilol à l’emplacement actuel

mkdir -p maison/rezdechaussee/cuisine/frigo => va créer l’arborescence complète des dossiers

mkdir -pv => mode verbeux => va « expliquer » ce qu’il fait

mkdir -pv pays/region/commune/ville/quartier/maison > var/log/création.log => va créer l’arborescence complète et va enregistrer ce qu’il a fait (mode verbeux) dans le fichier création.log

> => crée le fichier ou écrase si existant

>> => crée le fichier ou ajoute dedans si existant

mkdir -pv < fichier.txt => permet de dire quoi créer via le fichier « fichier.txt »

rmdir : va effacer le dossier (uniquement s’il est vide !)

rmdir -p brol/bidule => va effacer brol et bidule

rmdir -p brol/\* => va effacer tout brol et tout ce qu’il contient !

touch => crée un fichier

touch maison/rezdechaussee/cuisine/frigo/bière => va créer le fichier bière (à condition que l’arborescence de dossier existe => ne va donc pas créer ni l’arborescence ni le fichier)

file maison/rezdechaussee/cuisine/frigo/bière => va spécifier le type du fichier bière

pwd => va afficher le chemin absolu d’où l’on est

head maison/rezdechaussee/cuisine/frigo/bière => va afficher les 10 première ligne du fichier

head -n2 maison/rezdechaussee/cuisine/frigo/bière => va afficher les 2 première ligne du fichier

head -c20 maison/rezdechaussee/cuisine/frigo/bière => va afficher les 20 premiers caractères du fichier

tail maison/rezdechaussee/cuisine/frigo/bière => va afficher les 10 dernière ligne

tail -n2 maison/rezdechaussee/cuisine/frigo/bière => va afficher les 2 dernière ligne du fichier

tail -c20 maison/rezdechaussee/cuisine/frigo/bière => va afficher les 20 derniers caractères du fichier

rm maison/rezdechaussee/cuisine/frigo/bière => va supprimer le fichier bière

rm -d => suppression des fichiers et dossiers vide

rm -dr => suppression des fichiers et dossiers vide en mode récursif

df => disk free => donne l’espace disque libre sur le disque dur

df -h => rend les info plus lisible

du : estime l’espace utilisé

🡺en bash et en fish

alias => montre la liste des alias existant

alias miseajour=’sudo pacman -Syyu’ => va créer un alias (uniquement durant le temps que le terminal est ouvert) dont la commande est miseajour et qui va réaliser la commande complète : sudo pacman -Syyu

🡺en fish

fish\_config => lance un configurateur pour fish. Si on sélectionne l’onglet abbréviation, il y a la possibilité d’enregistrer (durablement) nos alias

top => gestionnaire de tâche

htop => gestionnaire de tâche

gnome-system-monitor => gestionnaire de tâche (app) issu de Gnome (nécessite donc d’avoir installé Gnome

type => va montrer si la commande est interne/externe

type sudo => montre /usr/bin/sudo => c’est donc une commande externe (externe au terminal)

type ls => montre ‘ls is a function with definition’ => c’est donc un commande interne (au terminal)

whereis => montre le chemin d’accès à la commande et au manuel de la commande

ls -a > bureau/resultat => enregistre le resultat ls -a dans le fichier ‘resultat’

> va créer le fichier ou l’écrase s’il existait déjà

>> va créer le fichier ou ajoute du contenu s’il existait déjà

sort < ls -a => trie le contenu du ls -a et l’affiche

sort < bureau/resultat > bureau/sorted => on prend le fichier ‘resultat’ on trie son contenu et on l’enregistre dans le fichier ‘sorted’

cat bureau/resultat 2> error.log => va effectuer la commande cat et s’il y a une erreur, va l’enregistrer dans error.log

mkdir /document/brol && touch /document/brol/toto => si la commande mkdir fonctionne (pas d’erreur) alors il effectue la commande suivante (touch)

mkdir /document/brol || echo ‘salut’ => si la commande mkdir ne fonctionne pas alors il va effectuer la commande suivante (echo)

mv => déplacer le fichier (et renommer si nécessaire)

mv maison/rezdechaussee/cuisine/frigo/bière maison/rezdechausse/cuisine/table => va bouger biere dans table

mv maison/rezdechaussee/cuisine/frigo/bière maison/rezdechausse/cuisine/table/alcool => va bouger biere dans table et changer son nom en ‘alcool’

cp => copie complète (et renommer si nécessaire)

cp maison/rezdechaussee/cuisine/frigo/bière maison/rezdechausse/cuisine/table => va copier biere dans table (biere reste dans frigo et il crée une copie dans table)

cp maison/rezdechaussee/cuisine/frigo/bière maison/rezdechausse/cuisine/table/alcool => va copier biere dans table et changer le nom de la copie en ‘alcool’

si l’on a plusieurs dossier ou fichier qui ont plus ou moins la même syntaxe (genre toto, tata, titi) on peut utiliser le ‘joker’ => ?

rm t?t? -i => va supprimer (rm => remove) le fichier/dossier ayant un t en premier et troisième caractère sur quatre et va demander l’autorisation avant chaque suppression ( -i ).

wc compte le nombre de ligne, de mot et puis de caractère d’un fichier

wc -l /var/log/pacman.log => va afficher uniquement le nombre de ligne

contenu du fichier passwd (/etc/passwd)

nom\_utilisateur : password : uid : gid : nom\_complet\_user : répertoire racine : shell

cut => sert à ‘couper’ en tronçon les lignes de caractère

cut -d: => (--delimiter) => va couper à tous les : présent dans la ligne de texte

cut -f1 => (--field) => ne va afficher le que le champ 1

--output-delimiter=’ ’ => lors de l’affichage va définir ce que l’on veut entre les champs choisi (ici => un espace)

cat /etc/passwd | cut -d: -f1,3 –output-delimiter=”+ +” => va afficher le champ 1 et 3 délimité par ‘+ +’ provenant du fichier /etc/passwd

grep => affiche/trie la selection selon une chaîne de caractère spécifique

grep -e ‘’^r’’ /etc/passwd => va afficher uniquement les lignes du fichier /etc/passwd où un ‘r’ se trouve en début de ligne (=> début de ligne référencé par ‘^’)

grep -e ‘’sh\$’’ /etc/passwd => va afficher uniquement les lignes du fichier /etc/passwd où un ‘sh’ se trouve en fin de ligne (=> fin de ligne référencé par ‘$’) => !!! ne pas oublier le ‘\’ devant le ‘$’ !!!

-c => va afficher uniquement le nombre de ligne qu’il a trouvé via la méthode grep

grep -e ‘’sh\$’’ /etc/passwd -c => va afficher ‘3’

🡺 autre possibilité

cat /etc/passwd | grep ‘’sh\$’’

🡺uniquement pour information

locate => permet de localiser une chaine de caractère parmi **tous** les fichiers de l’ordi **instantanément** !!

locate chromium => va afficher uniquement les fichiers ayant ‘chromium’ dans leur adresse

find => passe en revue tous les fichiers à partir d’un dossier donné ayant une chaine de caractère spécifique

-name chromium => défini la chaîne de caractère voulue

find / -name chromium => va rechercher à la racine ( / ) la chaîne de caractère chromium mais va afficher tous les fichiers par lequel il est passé (et donc même ceux qui n’ont pas chromium)!!!!

find / -name chromium | grep ‘’chromium’’ => va afficher uniquement les résultats où chromium se retrouve

/dev/null => genre de ‘trou noir’ où l’on envoie ce que l’on veut détruire

2> /dev/null => s’il y a des erreurs à afficher, au lieu de les afficher on les envoie se faire détruire dans le ‘trou noir’

-size => cherche les fichiers selon une taille donnée

find /usr -size +5M => va chercher un fichier dans /usr ayant 5M ou +

find /usr -size -5M => va chercher un fichier dans /usr ayant 5M ou –

-mtime => cherche les fichiers qui ont été MAJ il y a – de x jours

find / -mtime -3 2> /dev/null => va afficher les fichiers MAJ il y a moins de 3 jours

-newer => va chercher les fichiers qui sont plus récent qu’un autre fichier spécifié

find ~ -newer ~/Documents/toto 2> /dev/null => va rechercher dans ~ les fichiers plus récent que toto

-delete => va effacer tous les fichiers trouvés via la commande find

-exec => va exécuter tous les fichiers trouvés via la commande find

-ok => idem que -exec mais va demander l’autorisation avant d’exécuter

useradd => va ajouter un nouvel utilisateur

-c 'Toto le héro’ => défini le nom complet du nouvel utilisateur

-g wheel => l’utilisateur ne sera que dans le groupe wheel

-G wheel => l’utilisateur sera dans le groupe wheel ET dans le groupe qui porte son nom

-m => crée le répertoire utilisateur

-s /bin/fish => défini le shell par défaut de l’utilisateur

-k => permet de définir un ‘squelette’ supplémentaire dans le dossier ‘home‘ du nouvel utilisateur

🡺 !! ne pas inclure le nouveau mot de passe lors de la création du nouvel user (mot de passe en clair)

sudo useradd -c ‘Toto ce héro’ -m -k -s /bin/fish Toto

passwd => permet la gestion des mot de passe

-e => va rendre le mot de passe directement non valide => va donc obliger l’utilisateur à changer son mot de passe lors de sa prochaine connection

-n => défini le nombre minimum de jour entre 2 changement obligatoire de mot de passe

-x => défini le nombre maximum de jour entre 2 changement obligatoire de mot de passe

-l => verouille le mot de passe de l’utilisateur

-u => déverrouille le mot de passe

-d => rend le mot de passe vide => permet de se connecter sans mot de passe

usermod => permet la modification d’un utilisateur

-a => va permettre d’ajouter l’utilisateur à un autre groupe (ne s’utilise qu’avec -G)

sudo usermod -a -G wheel toto

-g => change le groupe de l’utilisateur (le groupe doit exister)

groupadd => permet d’ajouter de nouveaux groupes

chown => permet de changer le propriétaire d’un groupe/d’un fichier

chown root :staff /u => root devient le propriétaire de /u et son groupe en staff

chmod => permet de changer les droits d’accès à un fichier

🡺🡺2 type d’utilisation

🡺 mode ugoa

(u)=> propriétaire, (g)=> group, (o)=> autre utilisateur qui ne sont pas du groupe, (a) tous les autres

+ ajoute un droit // - retire un droit // = non modifié

(r)=>lecture // (w)=> écriture // (x)=> exécute(ou recherche de fichier)

chmod u-rx, g+w, o= ~/Documents/toto/

🡺 mode octal

Donne des accès via l’addition des bits 4, 2 et 1 (valeur possible 0,1,2,3,4,5,6,7)

le premier => modifie l’accès du propriétaire

le deuxième => modifie l’accès du groupe

le troisième => modifie l’accès de tous les autres

chmod 777 /home/toto => donne l’accès complet (lecture, écriture, exécute) à tout le monde => propriétaire, groupe et tous les autre

chmod 707 /home/toto => donnes l’accès complet (r,w,x) aux propriétaire et à tous les autres qui ne sont pas du groupe