

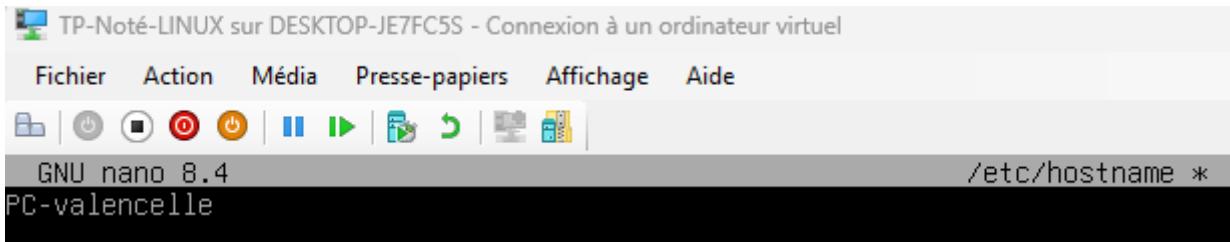
# TP – Mise à niveau Linux

## **Table des matières**

|  |   |
|--|---|
| Configuration Debian.....                          | 3 |
| Arborescence.....                                  | 4 |
| Gestion des utilisateurs et des groupes.....       | 4 |
| Gestion des droits.....                            | 5 |
| Réponse à la question.....                         | 5 |
| Enregistrer des fichiers via des redirecteurs..... | 5 |
| Recherche dans paquets.txt.....                    | 6 |
| Mise en place d'un disque dur supplémentaire.....  | 6 |
| Approfondir une commande.....                      | 7 |
| 3 exemples de commande avec « Tree ».....          | 7 |
| Question.....                                      | 8 |

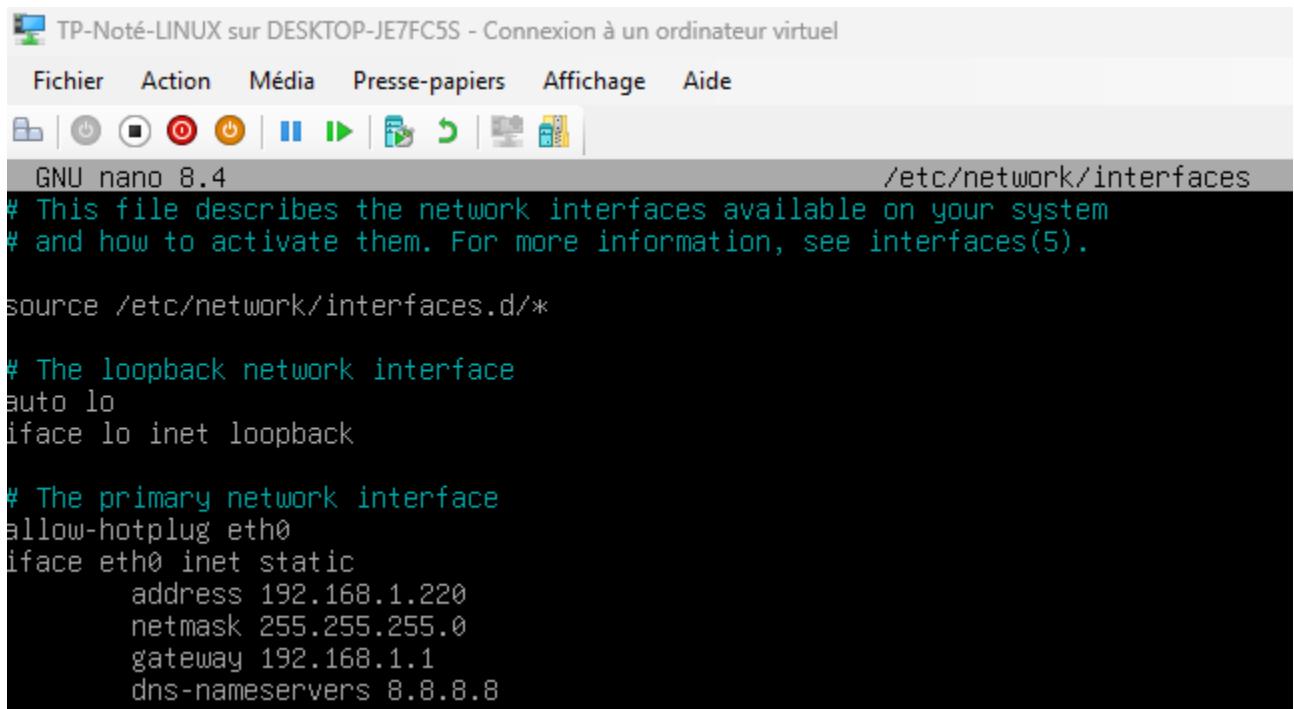
## Configuration Debian

Comme demandé dans le sujet, j'ai configuré le nom de la machine avec le format PC-nom au moment de l'installation de Debian 13 puis je me suis rendu dans le fichier « hostname » avec la commande « nano /etc/hostname » et j'ai pris la capture d'écran suivante :



```
TP-Noté-LINUX sur DESKTOP-JE7FC5S - Connexion à un ordinateur virtuel
Fichier Action Média Presse-papiers Affichage Aide
GNU nano 8.4 /etc/hostname *
PC-valencelle
```

Par la suite j'ai configuré le réseau de la machine avec une IP fixe disponible de mon DHCP, j'ai configuré la 192.168.1.220/24 avec comme passerelle 192.168.1.1 et comme DNS 8.8.8.8 :



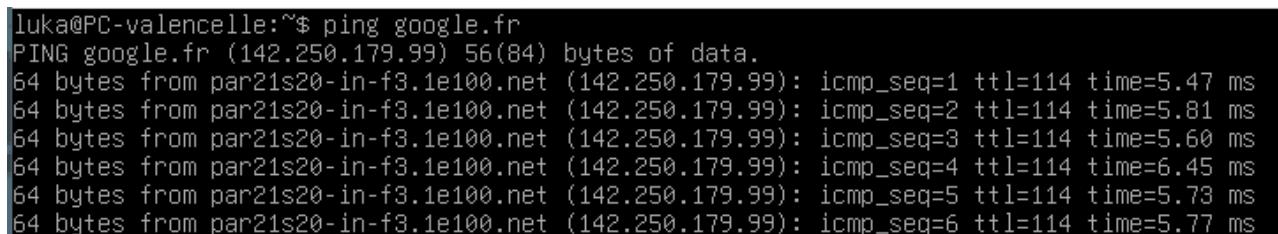
```
TP-Noté-LINUX sur DESKTOP-JE7FC5S - Connexion à un ordinateur virtuel
Fichier Action Média Presse-papiers Affichage Aide
GNU nano 8.4 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug eth0
iface eth0 inet static
    address 192.168.1.220
    netmask 255.255.255.0
    gateway 192.168.1.1
    dns-nameservers 8.8.8.8
```

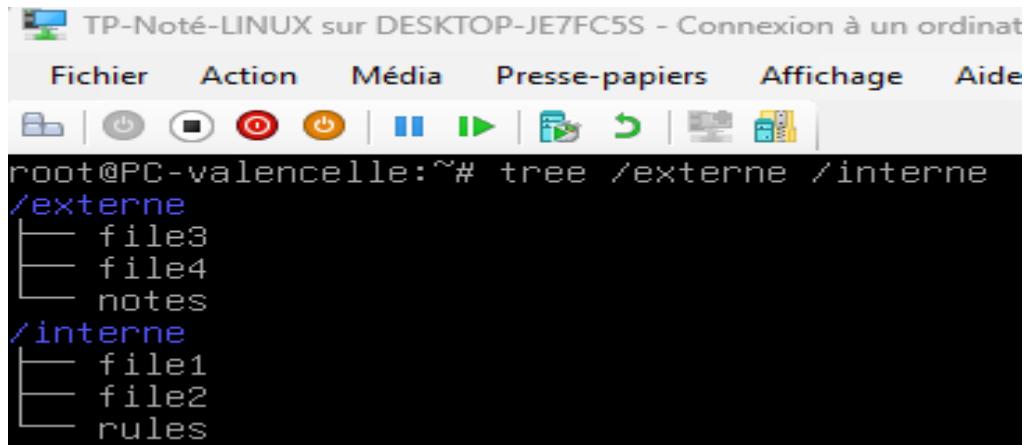
Puis afin de tester la connectivité suite au changement de la configuration réseau, j'ai effectué un ping vers google.fr :



```
luka@PC-valencelle:~$ ping google.fr
PING google.fr (142.250.179.99) 56(84) bytes of data.
64 bytes from par21s20-in-f3.1e100.net (142.250.179.99): icmp_seq=1 ttl=114 time=5.47 ms
64 bytes from par21s20-in-f3.1e100.net (142.250.179.99): icmp_seq=2 ttl=114 time=5.81 ms
64 bytes from par21s20-in-f3.1e100.net (142.250.179.99): icmp_seq=3 ttl=114 time=5.60 ms
64 bytes from par21s20-in-f3.1e100.net (142.250.179.99): icmp_seq=4 ttl=114 time=6.45 ms
64 bytes from par21s20-in-f3.1e100.net (142.250.179.99): icmp_seq=5 ttl=114 time=5.73 ms
64 bytes from par21s20-in-f3.1e100.net (142.250.179.99): icmp_seq=6 ttl=114 time=5.77 ms
```

## Arborescence

J'ai configuré l'arborescence suivante, conforme avec la demande :



```
root@PC-valencelle:~# tree /externe /interne
/externe
├── file3
└── file4
    └── notes
/interne
└── file1
    └── file2
        └── rules
```

## Gestion des utilisateurs et des groupes

Suite à quoi j'ai créé 3 utilisateurs, respectivement nommé Dewey, Reese et Francis :

```
luka:x:1000:1000:luka,,,:/home/luka:/bin/bash
Dewey:x:1001:1001::/home/Dewey:/bin/sh
Reese:x:1002:1002::/home/Reese:/bin/sh
Francis:x:1003:1003::/home/Francis:/bin/sh
```

Puis j'ai créé les groupes Interne et Externe aux quels j'ai ajouté les utilisateurs demandés :

```
tape:x:26:
sudo:x:27:luka,Dewey
audio:x:29:luka
lute:x:31:lute
luka:x:1000:
Dewey:x:1001:
Reese:x:1002:
Francis:x:1003:
Interne:x:1004:Reese,Dewey
Externe:x:1005:Reese,Francis
```

## Gestion des droits

J'ai par la suite configuré les droits indiqué dans la demande :

```
luka@PC-valencelle:~$ sudo ls -l /interne /externe
/externe:
total 0
-rwxr-x--- 1 root Externe 0 12 oct. 09:46 file3
-rwxr-x--- 1 root Externe 0 12 oct. 09:46 file4
-rwxrwx--- 1 root Externe 0 12 oct. 09:46 notes

/interne:
total 0
-rwxr-x--- 1 root Interne 0 12 oct. 09:46 file1
-rwxr-x--- 1 root Interne 0 12 oct. 09:46 file2
-rwxr-x--- 1 root Interne 0 12 oct. 09:46 rules
```

Nous pouvons lire dans la capture que dans le répertoire externe le groupe associé est « Externe » et que pour les fichiers « file3 et file4 » le groupe ne peut les lire sans pouvoir les modifier car il est indiqué

r-x avec r pour read et x pour execute. Concernant le fichier « notes » le groupe « Externe » peut lire et écrire car il est indiqué rwx r et x étant expliqués précédemment et w pour write.

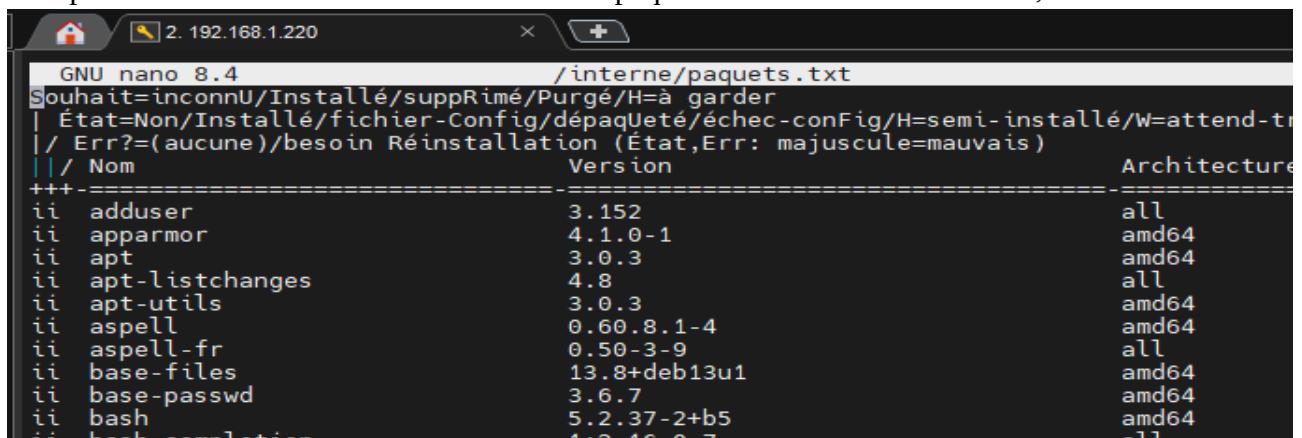
Concernant le répertoire interne nous pouvons voir dans la capture d'écran que le groupe ne peut que lire les 3 fichiers du répertoire car une nouvelle fois nous avons r-x pour read et execute (pour pouvoir lire un fichier sous debian il faut pouvoir l'exécuter d'où la présence du x)

## Réponse à la question

Afin d'empêcher l'utilisateur « root » d'accéder au répertoire externe je pense qu'il faudrait mettre en place un docker afin d'isoler le répertoire complètement.

## Enregistrer des fichiers via des redirecteurs

Avec l'utilisation de la commande sudo « dpkg -l > /interne/paquets.txt » j'ai pu créer dans le répertoire interne un fichier txt avec tout les paquets installés sur ma machine, voici un extrait :



```
GNU nano 8.4          /interne/paquets.txt
Souhait=inconnu/Installé/suppRimé/Purgé/H=à garder
| État=Non/Installé/fichier-Config/dépaqueté/échec-conFig/H=semi-installé/W=attend-tr
| / Err?=(aucune)/besoin Réinstallation (État,Err: majuscule=mauvais)
| / Nom                         Version           Architecture
++-----+
ii  adduser                     3.152             all
ii  apparmor                    4.1.0-1          amd64
ii  apt                          3.0.3             amd64
ii  apt-listchanges              4.8               all
ii  apt-utils                   3.0.3             amd64
ii  aspell                      0.60.8.1-4       amd64
ii  aspell-fr                   0.50-3-9          all
ii  base-files                  13.8+deb13u1      amd64
ii  base-passwd                 3.6.7             amd64
ii  bash                         5.2.37-2+b5      amd64
ii  bsd-mailx                   1.0-16.2-7        all
```

## Recherche dans paquets.txt

Avec la commande « grep openssh /interne/paquets.txt » précédé d'un « sudo -i » j'ai pu vérifier si le paquets « openssh » était indiqué dans le fichier txt « paquets ». Voici le résultat que j'ai obtenu :

```
root@PC-valencelle:~# grep openssh /interne/paquets.txt
ii  openssh-client           1:10.0p1-7          amd64
secure shell (SSH) client, for secure access to remote machines
ii  openssh-server            1:10.0p1-7          amd64
secure shell (SSH) server, for secure access from remote machines
ii  openssh-sftp-server       1:10.0p1-7          amd64
secure shell (SSH) sftp server module, for SFTP access from remote machines
root@PC-valencelle:~#
```

## Mise en place d'un disque dur supplémentaire

J'ai ajouté un disque de 15Go avec une partition de 10Go avec le système de fichier ext4 et j'ai monté la partition sur le répertoire Backup situé à la racine :

```
GNU nano 8.4                                     /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# systemd generates mount units based on this file, see systemd.mount(5).
# Please run 'systemctl daemon-reload' after making changes here.
#
# <file system> <mount point>  <type>    <options>      <dump>  <pass>
# / was on /dev/sda1 during installation
UUID=e19a4992-6c9c-4eb2-9fc8-53f85d32824c  /          ext4    errors=re
# swap was on /dev/sda5 during installation
UUID=016d0c10-3776-44d0-bf03-2318317cab60  swap      sw
/dev/sr0          /media/cdrom0  udf,iso9660 user,noauto  0        0
UUID=36bd97c5-07f4-4d1e-b8c1-535962898c20  /backup  ext4 defaults 0 2 ←
```

```

root@PC-Valencelle:~# fdisk -l

Disque /dev/sdb : 15 GiB, 16106127360 octets, 31457280 secteurs
Modèle de disque : Virtual Disk
Unités : secteur de 1 x 512 = 512 octets
Taille de secteur (logique / physique) : 512 octets / 4096 octets
taille d'E/S (minimale / optimale) : 4096 octets / 4096 octets
Type d'étiquette de disque : dos
Identifiant de disque : 0xdd7c5697

Périphérique Amorçage Début      Fin Secteurs Taille Id Type
/dev/sdb1            2048 20973567 20971520    10G 83 Linux

Disque /dev/sda : 20 GiB, 21474836480 octets, 41943040 secteurs
Modèle de disque : Virtual Disk
Unités : secteur de 1 x 512 = 512 octets
Taille de secteur (logique / physique) : 512 octets / 4096 octets
taille d'E/S (minimale / optimale) : 4096 octets / 4096 octets
Type d'étiquette de disque : dos
Identifiant de disque : 0x4fd0d05e

Périphérique Amorçage     Début      Fin Secteurs Taille Id Type
/dev/sda1      *        2048 39684095 39682048  18,9G 83 Linux
/dev/sda2          39686142 41940991 2254850   1,1G  f Étendue W95 (LBA)
/dev/sda5          39686144 41940991 2254848   1,1G 82 partition d'échange Linux / Solaris

```

## Approfondir une commande

### 3 exemples de commande avec « Tree »

Il y a la commande simple mais efficace « tree » qui nous donne l'arborescence du répertoire dans lequel nous nous trouvons :

```

luka@PC-valencelle:/home$ tree
.
└── luka
    2 directories, 0 files
luka@PC-valencelle:/home$ █

```

Nous pouvons rajouter le paramètre -d à cette commande afin qu'elle nous sorte uniquement les dossiers, afin de la tester j'ai tapé « sudo -i » puis j'ai créé 4 dossiers et 2 fichiers :

```

root@PC-valencelle:~# tree
.
├── test
│   ├── test1
│   ├── test2
│   └── test2.txt
└── test.txt
4 directories, 2 files
root@PC-valencelle:~# █

```

En tapant « tree -d » la commande ne nous sort que les dossiers :

```
root@PC-valencelle:~# tree -d
.
└── test
    ├── test1
    └── test2

4 directories
root@PC-valencelle:~#
```

Et nous pouvons aussi exporter une arborescence avec la commande « tree /nomdurépertoire > arborescence.txt ». Il est également possible d'ajouter un paramètre tel que -d afin d'exporter uniquement l'arborescence des dossiers, avec la commande « sudo tree -d /home > arbo.txt » j'exporte l'arborescence des dossiers du répertoire home dans un fichier txt nommé arbo qui sera dans le répertoire où je me trouve :

```
luka@PC-valencelle:~$ sudo tree -d > arbo.txt
luka@PC-valencelle:~$ ls
arbo.txt
luka@PC-valencelle:~$
```

```
luka@PC-valencelle:~$ cat arbo.txt
/home
└── luka

2 directories
luka@PC-valencelle:~$
```

## Question

1)

Avec la commande tree -L 2 il est possible d'afficher que 2 niveaux de profondeur, le paramètre -L permet d'indiquer le niveau que l'on souhaite (L pour Level).

2)

Il est également possible d'afficher les permissions. En effet le paramètres -p (pour permission) affiche les droits rwx de chaque utilisateurs et groupes pour chaque fichiers et dossiers.

3)

La différence principale que je note entre tree et ls est la clarté des informations. Avec tree tout est facile à comprendre et à lire. Avec un ls le résultat de la commande est plus « brut ». Personnellement j'ai préféré utiliser tree de manière générale pendant ce tp. Le seul moment où j'ai

utilisé ls c'était pour afficher les droits, car je ne connaissais pas le paramètres -p de la commande tree.