

DOCUMENTATION D'EXPLOITATION

Serveur Ubuntu 24.04 et conteneur Nginx sous VirtualBox

Légende des couleurs : commande sudo option chemin / URL "chaîne" argument

1 Création de la machine virtuelle

Paramètres généraux

Paramètre	Valeur
Nom	BTS-LEFOLL-Ubuntu
Dossier	C:\Users\malo.lefoll\VirtualBox VMs
Image ISO	ubuntu-24.04.4-live-server-amd64.iso
Type / Sous-type / Version	Linux / Ubuntu / Ubuntu (64-bit)
Installation sans surveillance	Désactivée (« Skip Unattended Installation » cochée)

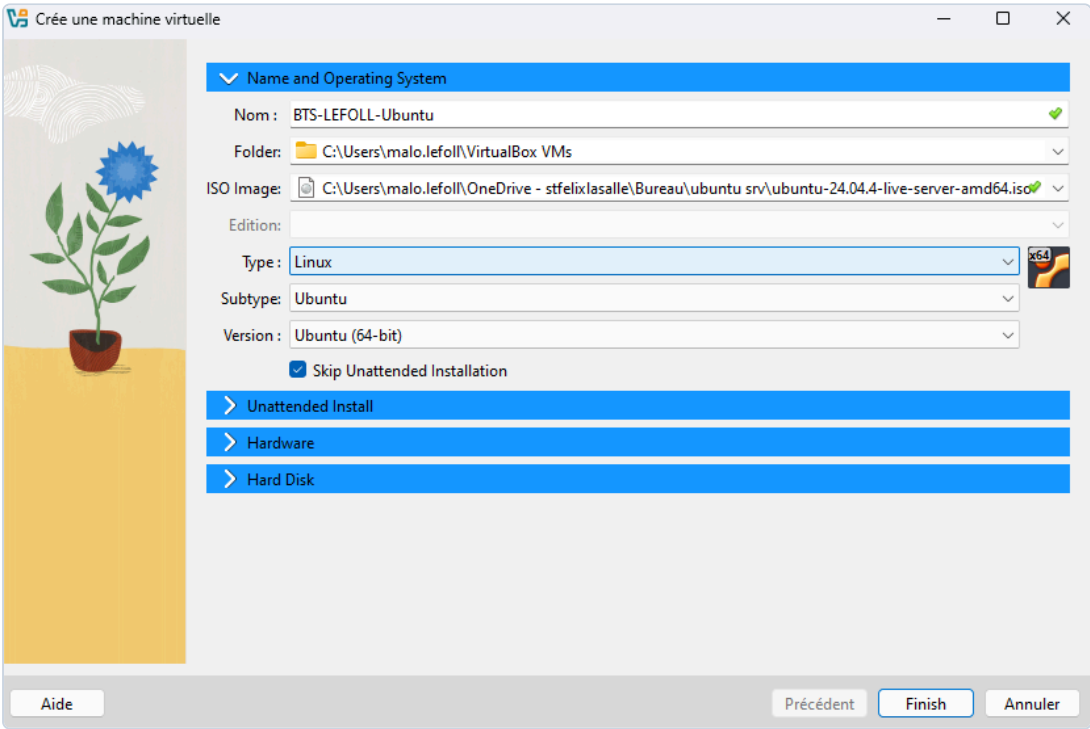


Figure 1 : nom et système d'exploitation de la VM.

Disque dur virtuel

- Création d'un disque virtuel : BTS-LEFOLL-Ubuntu.vdi
- Taille : 25 Gio
- Format : VDI (VirtualBox Disk Image), sans préallocation

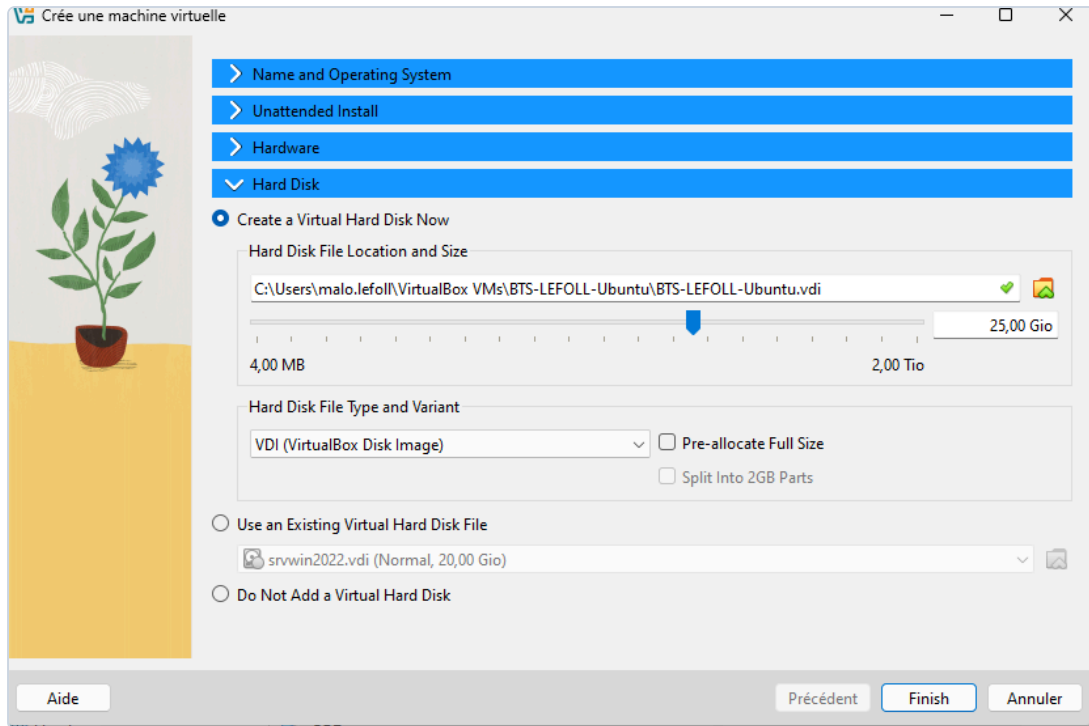


Figure 2 : création du disque dur virtuel de 25 Gio.

2 Mise à jour et contrôle du système

Mise à jour de l'index des paquets, puis installation de Nginx :

BASH

```
$ apt update
$ apt-get install nginx
```

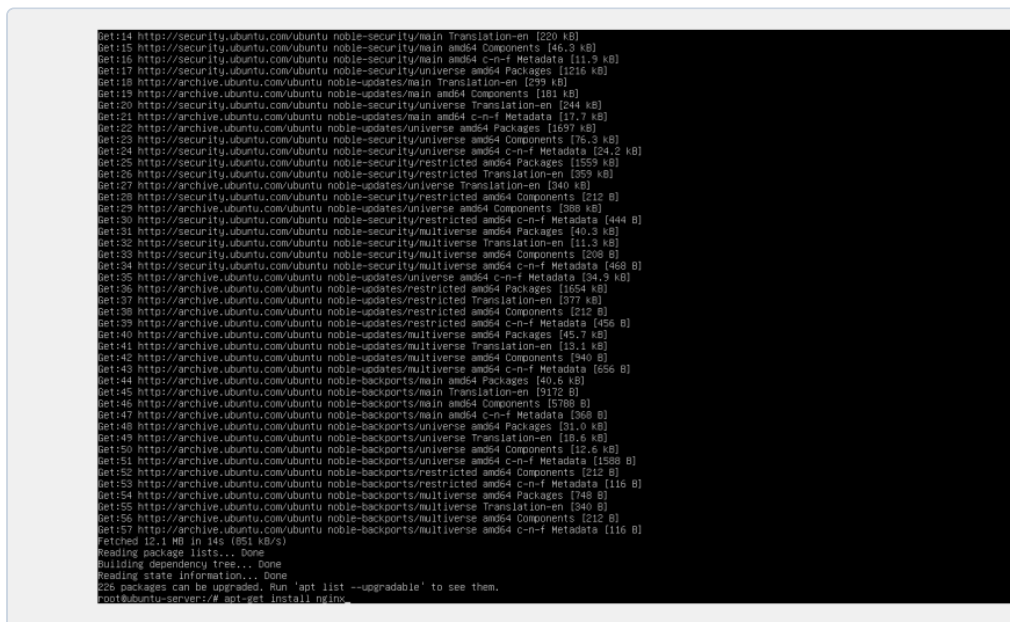


Figure 3 : apt update.

Vérifications de l'identité et du réseau de la machine :

BASH

```
$ hostnamectl
$ ip -br address
$ ip route
```

- Hôte : **ubuntu-server**, Ubuntu 24.04.4 LTS, noyau Linux 6.8.0-100-generic, virtualisation Oracle (VirtualBox)
- Interface **enp0s3** : **10.0.2.15/24**
- Passerelle par défaut : **10.0.2.2** (réseau NAT VirtualBox)

```
[-4739 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
[-4741 "nginx: worker process"
[-4742 "nginx: worker process"

Sep 29 12:43:20 ubuntu-server systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Sep 29 12:43:20 ubuntu-server systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
root@ubuntu-server:~# hostnamectl
Static hostname: ubuntu-server
Icon name: computer-vm
Chassis: vm
Machine ID: 6b95a6cb97c04e97b942dd0b02179239
Boot ID: 2d026319972a4668ab9ca78d301e58ac
Virtualization: oracle
Operating System: Ubuntu 24.04.4 LTS
Kernel: Linux 6.8.0-100-generic
Architecture: x86_64
Hardware Vendor: innotek GmbH
Hardware Model: VirtualBox
Firmware Version: VirtualBox
Firmware Date: Fri 2006-12-01
Firmware Age: 19y 9month 4w
root@ubuntu-server:~#
```

Figure 4 : `hostnamectl`.

```
Firmware Age: 19y 9month 4w
root@ubuntu-server:~# ip -br address
lo        UNKNOWN  127.0.0.1/8 ::1/128
enp0s3    UP        10.0.2.15/24 metric 100 fd17:625c:f037:2:a00:27ff:fe56:931/64 fe80::a00:27ff:fe56:931/64
root@ubuntu-server:~#
```

Figure 5 : `ip -br address`.

```
root@ubuntu-server:~# ip route
default via 10.0.2.2 dev enp0s3 proto dhcp src 10.0.2.15 metric 100
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.15 metric 100
10.0.2.2 dev enp0s3 proto dhcp scope link src 10.0.2.15 metric 100
10.0.2.3 dev enp0s3 proto dhcp scope link src 10.0.2.15 metric 100
root@ubuntu-server:~#
```

Figure 6 : `ip route`.

Instantané : prise d'un instantané de la VM (« Instantané 1 ») une fois le système validé.

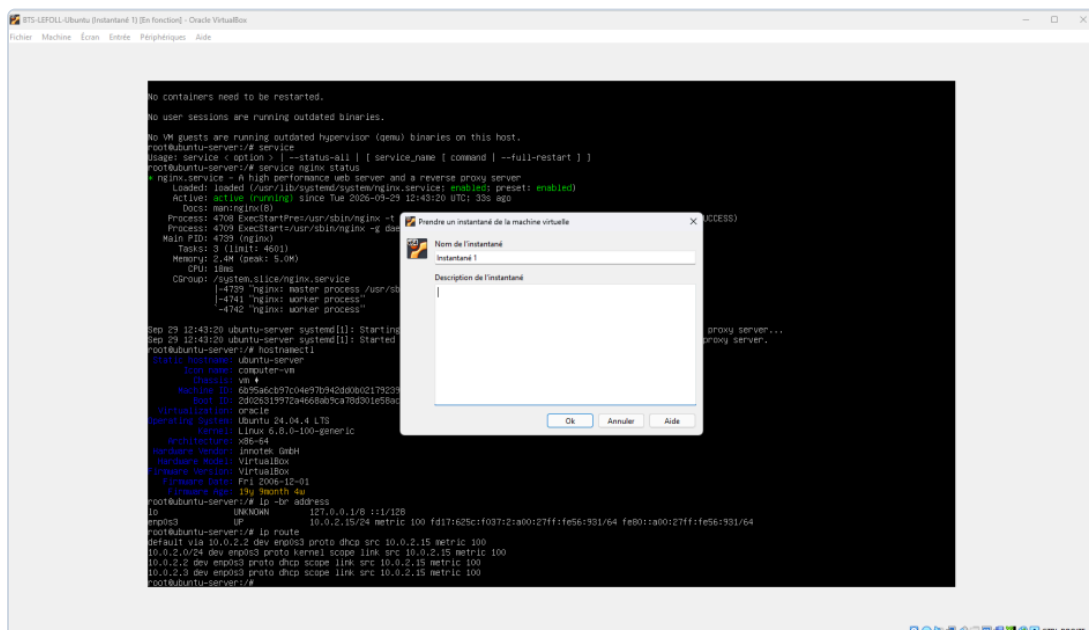


Figure 7 : fenêtre « Prendre un instantané de la machine virtuelle ».

3 Ajout d'un second disque

Ajout d'un disque VDI de **5 Gio** (`BTS-LEFOLL-Ubuntu_1.vdi`) depuis le sélecteur de médias VirtualBox.

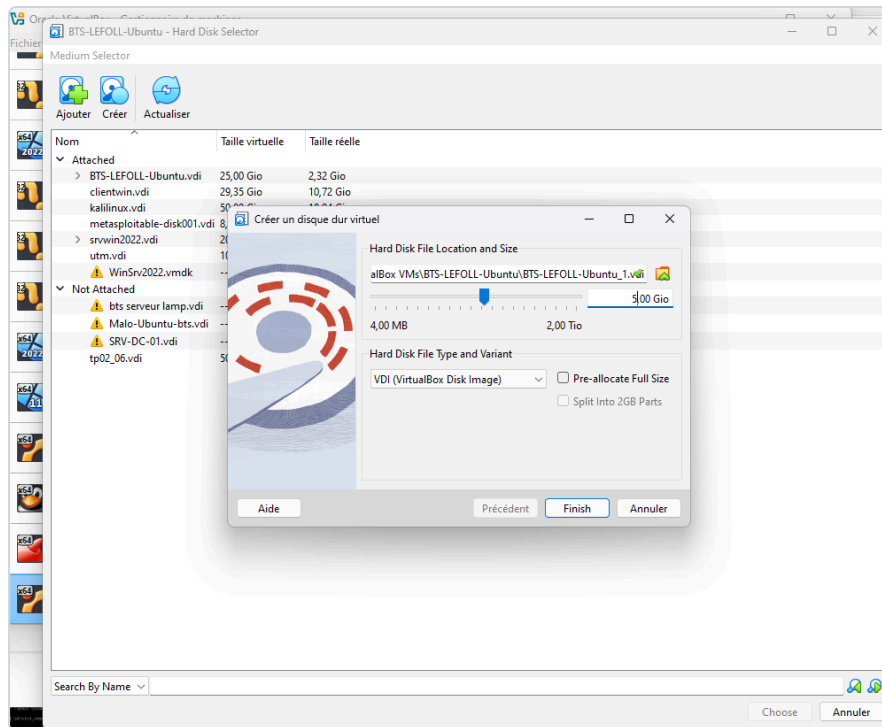


Figure 8 : création du nouveau disque dur virtuel.

Identification du disque dans Ubuntu :

BASH

```
$ lsblk -f
```

Le nouveau disque apparaît sous le nom `sdb`, sans partition ni système de fichiers.

```
malo@malosrv:~$ lsblk -f
NAME        FSTYPE FSVER LABEL UUID                                 FSAVAIL FSUSE% MOUNTPOINTS
loop0       squashf 4.0                                     0      100% /snap/core20/2105
loop1       squashf 4.0                                     0      100% /snap/lxd/27037
loop2       squashf 4.0                                     0      100% /snap/snapd/20671
sda
├─sda1
├─sda2 ext4      1.0                                0efbd951-c9a4-487b-a599-c4f397475d0d    1,7G      7% /boot
├─sda3 LVM2_me LVM2 00                        8iykfo-sUc1-lmdy-euvm-aNX4-Zo8m-H9XNhi
│   └─ubuntu--vg-ubuntu--lv
│       ext4      1.0                                5f6efc59-65c4-4768-96e3-670d0ed28d06    5,4G     47% /
sdb
```

Figure 9 : identification du disque avec `lsblk -f`.

4 Partitionnement et formatage

1. Lancer `sudo fdisk /dev/sdb`.
2. Créer la partition primaire `/dev/sdb1` avec `n` (5 Gio, valeurs par défaut).
3. Écrire la table de partitions avec `w`.

4. Formater en ext4 :

```
BASH
$ sudo mkfs.ext4 /dev/sdb1
```

5. Relever l'UUID :

```
BASH
$ sudo blkid /dev/sdb1
```

RÉSULTAT

```
UUID="a8f35c81-f58b-4d97-b7eb-c8217e60f9b8" BLOCK_SIZE="4096" TYPE="ext4"
PARTUUID="9064555e-01"
```

```
malo@malosrv:~$ sudo fdisk dev/sdb
Welcome to fdisk (util-linux 2.37.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

fdisk: cannot open dev/sdb: No such file or directory
malo@malosrv:~$ sudo fdisk /dev/sdb
Welcome to fdisk (util-linux 2.37.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x9064555e.

Command (m for help): n
Partition type
  p   primary (0 primary, 0 extended, 4 free)
  e   extended (container for logical partitions)
Select (default p): 1
Value out of range.
  p   primary (0 primary, 0 extended, 4 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-10485759, default 2048):
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-10485759, default 10485759):

Created a new partition 1 of type 'Linux' and of size 5 GiB.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

malo@malosrv:~$ sudo
```

Figure 10 : partitionnement avec fdisk.

```
malo@malosrv:~$ sudo mkfs.ext4 /dev/sdb1
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 1310464 4k blocks and 327680 inodes
Filesystem UUID: a8f35c81-f58b-4d97-b7eb-c8217e60f9b8
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

Figure 11 : formatage avec mkfs.ext4.

```
malo@malosrv:~$ sudo blkid /dev/sdb1
/dev/sdb1: UUID="a8f35c81-f58b-4d97-b7eb-c8217e60f9b8" BLOCK_SIZE="4096" TYPE="ext4" PARTUUID="9064555e-01"
malo@malosrv:~$
```

Figure 12 : relevé de l'UUID avec blkid.

5 Montage automatique

1. Créer le point de montage :

```
BASH
$ sudo mkdir -p /data
```

2. Ajouter la ligne suivante à `/etc/fstab` :

```
/ETC/FSTAB
UUID=a8f35c81-f58b-4d97-b7eb-c8217e60f9b8 /data ext4 defaults 0 2
```

3. Tester : `sudo mount -a`

4. Contrôler : `findmnt /data`

5. Vérifier l'écriture dans `/data`.

```
malo@malosrv:~$ sudo mkdir -p /data
[sudo] password for malo:
malo@malosrv:~$
```

Figure 13 : création du dossier /data.

```
GNU nano 6.2 /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).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/ubuntu-vg/ubuntu-lv during curtin installation
/dev/disk/by-id/dm-uuid-LVM-foXI3KfEkKe7zOHDMZSzeee4mxCI93WgdJfj6vfPE9RPbSHiqe6apsoSLbiyPNzd / ext4 defaults 0 1
# /boot was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/0efbd951-c9a4-487b-a599-c4f397475d0d /boot ext4 defaults 0 1
/swap.img none swap sw 0 0
```

Figure 14 : fichier /etc/fstab.

```
malo@malosrv:~$ sudo mount -a
malo@malosrv:~$
malo@malosrv:~$
```

Figure 15 : sudo mount -a.

```
malo@malosrv:~$ findmnt /data
malo@malosrv:~$
```

Figure 16 : findmnt /data.

```
malo@malosrv:~$ cd /data
malo@malosrv:/data$ ls
malo@malosrv:/data$
```

Figure 17 : accès au dossier /data.

6 Installation de Docker

BASH

```
$ sudo apt install docker.io
$ sudo systemctl enable --now docker
$ sudo docker version
```

Le service était initialement inactif (`inactive (dead)`), puis démarré. La commande `docker ps` ne renvoie aucun conteneur. Docker est en version **29.1.3** (client et serveur).

```
malo@malosrv:/$ sudo service docker status
o docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: enabled)
   Active: inactive (dead)
   TriggeredBy: o docker.socket
   Docs: https://docs.docker.com
malo@malosrv:/$ sudo service docker start
malo@malosrv:/$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
malo@malosrv:/$
```

Figure 18 : état du service Docker.

```
malo@malosrv:/$ sudo docker version
Client:
 Version:           29.1.3
 API version:       1.52
 Go version:        go1.24.4
 Git commit:        29.1.3-0ubuntu3~22.04.2
 Built:             Wed Apr 29 22:18:59 2026
 OS/Arch:           linux/amd64
 Context:           default

Server:
 Engine:
  Version:          29.1.3
  API version:      1.52 (minimum version 1.44)
  Go version:       go1.24.4
  Git commit:       29.1.3-0ubuntu3~22.04.2
  Built:            Wed Apr 29 22:18:59 2026
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          2.2.1
  GitCommit:
 runc:
  Version:          1.3.4-0ubuntu1~22.04.1
  GitCommit:
 docker-init:
  Version:          0.19.0
  GitCommit:
malo@malosrv:/$
```

Figure 19 : sudo docker version.

7 Déploiement du conteneur Nginx

BASH

```
$ sudo docker run -d --name web-bts -p 8080:80 nginx:alpine
```

- Image : `nginx:alpine`
- Nom du conteneur : `web-bts`
- Port `8080` de la VM publié vers le port `80` du conteneur

Vérifications :

BASH

```
$ sudo docker ps
$ sudo docker logs web-bts
$ curl http://localhost:8080
```

```
malo@malosrv:/$ sudo docker run -d --name web-bts -p 8080:80 nginx:alpine
Unable to find image 'nginx:alpine' locally
alpine: Pulling from library/nginx
e72112c14215: Pull complete
745dfb2690dd: Pull complete
e2de96513ba9: Pull complete
d9aae54b5831: Pull complete
6c53d0b2a666: Pull complete
64c8194480fe: Pull complete
9a9a644fdd6a: Pull complete
e76228b47809: Pull complete
0deea27e58d9: Download complete
d54d3939625e: Download complete
Digest: sha256:df221db836e1754089190208cee7eeda94f233197056426eda74a43ab1abeac2
Status: Downloaded newer image for nginx:alpine
f8163543ff251954104541c10314b65198f26c1bc6521a2b4c4f45d34f6e8729
malo@malosrv:/$ |
```

Figure 20 : création du conteneur.

```
malo@malosrv:/$ sudo docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS
f8163543ff25   nginx:alpine  "/docker-entrypoint..." About a minute ago Up About a minute 0.0.0.0:8080->80/tcp, [:
```

Figure 21 : `sudo docker ps`.

```

:~:8080->80/tcp web-bts
malo@malosrv:/$ sudo docker logs web-bts
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2026/09/29 14:09:39 [notice] 1#1: using the "epoll" event method
2026/09/29 14:09:39 [notice] 1#1: nginx/1.31.6
2026/09/29 14:09:39 [notice] 1#1: built by gcc 15.2.0 (Alpine 15.2.0)
2026/09/29 14:09:39 [notice] 1#1: OS: Linux 5.15.0-194-generic
2026/09/29 14:09:39 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1024:524288
2026/09/29 14:09:39 [notice] 1#1: start worker processes
2026/09/29 14:09:39 [notice] 1#1: start worker process 30
2026/09/29 14:09:39 [notice] 1#1: start worker process 31
malo@malosrv:/$ |

```

Figure 22 : sudo docker logs web-bts.

```

malo@malosrv:/$ curl http://localhost:8080
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, nginx is successfully installed and working.
Further configuration is required for the web server, reverse proxy,
API gateway, load balancer, content cache, or other features.</p>

<p>For online documentation and support please refer to
<a href="https://nginx.org/">nginx.org</a>.<br/>
To engage with the community please visit
<a href="https://community.nginx.org/">community.nginx.org</a>.<br/>
For enterprise grade support, professional services, additional
security features and capabilities please refer to
<a href="https://f5.com/nginx">f5.com/nginx</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
malo@malosrv:/$ |

```

Figure 23 : test avec curl (page « Welcome to nginx! »).

8 Redirection de port NAT VirtualBox

Règles créées dans « Règles de redirection de ports » :

Nom	Protocole	Port hôte	Port invité
Rule 1	TCP	22	22
Rule 2	TCP	8080	8080

Test depuis le navigateur de l'hôte : `http://127.0.0.1:8080`.

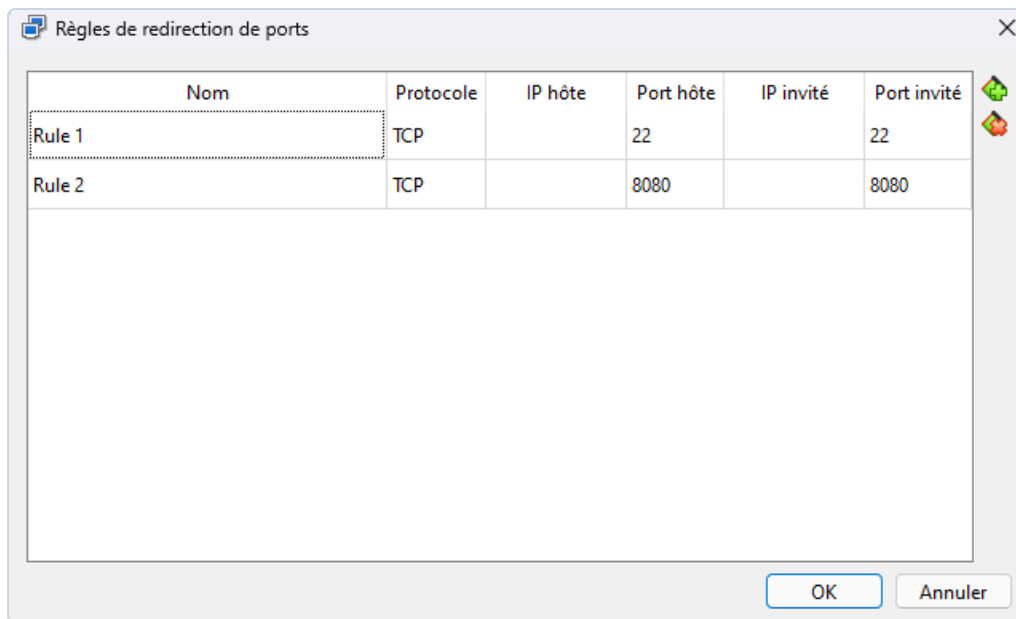


Figure 24 : règles de redirection de ports.

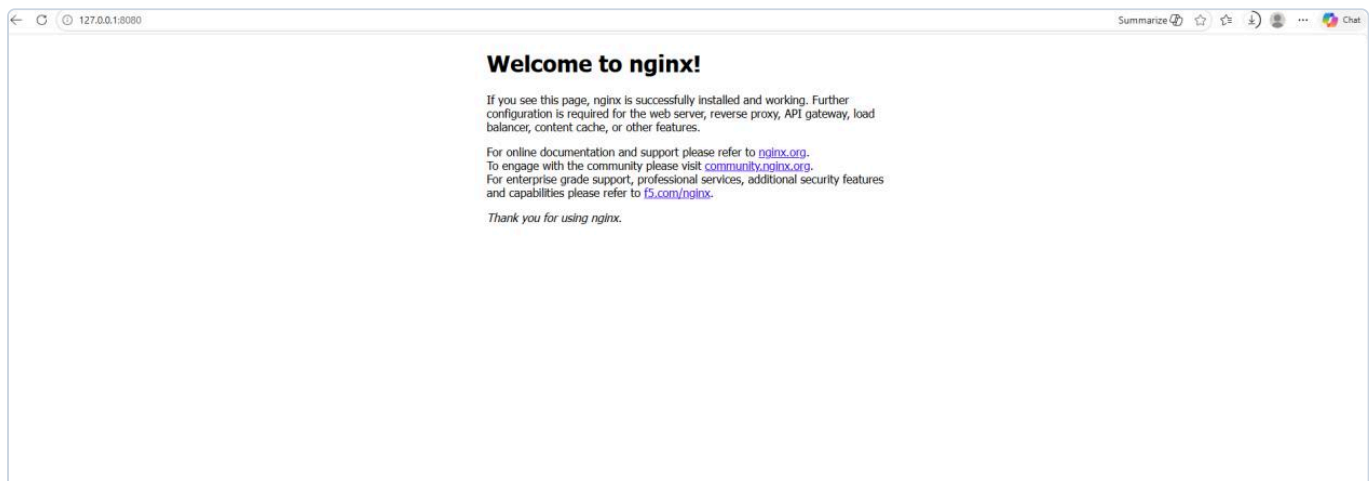


Figure 25 : page Nginx affichée dans le navigateur.

9 Persistance du contenu web

1. Créer le dossier et la page :

```
BASH
$ mkdir -p ~/site
$ printf '<h1>BTS CIEL SI0 Virtualisation</h1>' > ~/site/index.html
```

2. Monter le dossier dans Nginx en lecture seule :

```
BASH
$ sudo docker run -d --name web-bts -p 8080:80 \
> -v "$HOME/site:/usr/share/nginx/html:ro" nginx:alpine
```

3. Supprimer puis recréer le conteneur :

BASH

```
$ sudo docker rm -f web-bts
```

4. Vérifier le fichier source et la page :

BASH

```
$ test -f ~/site/index.html && echo PERSISTANT
$ curl http://localhost:8080
```

Le fichier source est conservé (`PERSISTANT`). Le premier `curl` juste après la recreation échoue (« Connection reset by peer »), car Nginx n'a pas fini de démarrer. Le second renvoie bien `<h1>BTS CIEL SIO Virtualisation</h1>`.

```
malo@malosrv:/$ mkdir -p ~/site
malo@malosrv:/$ printf '<h1>BTS CIEL SIO Virtualisation</h1>' > ~/site/index.html
malo@malosrv:/$
```

```
malo@malosrv:/$ sudo docker run -d --name web-bts -p 8080:80 \
> -v "$HOME/site:/usr/share/nginx/html:ro" nginx:alpine
dc6b6824c77d4cecc156a938e1a419071daf2020e436febd3aab690678dc356f
malo@malosrv:/$
```

```
malo@malosrv:/$ sudo docker rm -f web-bts
web-bts
```

```
malo@malosrv:/$ test -f ~/site/index.html && echo PERSISTANT
PERSISTANT
malo@malosrv:/$
```

```
malo@malosrv:/$ sudo docker rm -f web-bts
sudo docker run -d --name web-bts -p 8080:80 \
-v "$HOME/site:/usr/share/nginx/html:ro" nginx:alpine
curl http://localhost:8080
web-bts
f2fb72ee411e60e8b52625e48c1b82e0e168c625e312789aba9149f2770394aa
curl: (56) Recv failure: Connection reset by peer
malo@malosrv:/$ curl http://localhost:8080
<h1>BTS CIEL SIO Virtualisation</h1>malo@malosrv:/$
```

Figures 26 à 30 : création du dossier et du fichier, montage du volume, suppression du conteneur, vérification de la persistance, recreation du conteneur.

10 Contrôle final

- ✓ Le conteneur apparaît dans `docker ps`.
- ✓ Le port `8080` est publié vers `80/tcp`.
- ✓ La page répond avec `curl` ou dans le navigateur.
- ✓ La page reste disponible après recreation.

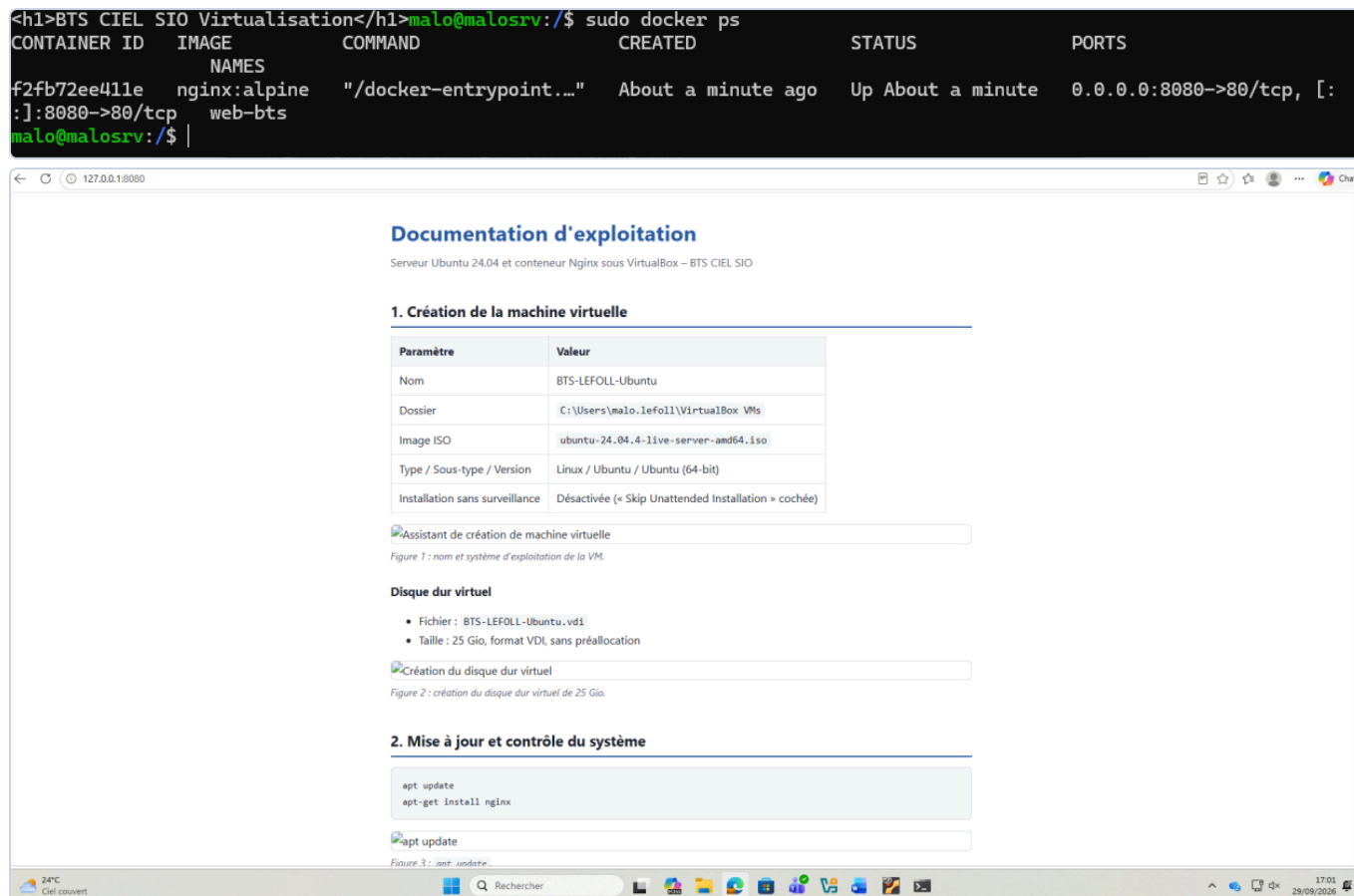


Figure 31 : docker ps et affichage dans le navigateur.