



hacker-man.webp

Préparer son environnement Linux

🔗 À faire avant la séance

Pour le cours Docker rootless, nous aurons besoin de deux choses :

- une petite VM linux (ou même un environnement Linux complet si vous avez)
- Docker Desktop installé et fonctionnel

1. La VM Linux

Choisissez la section qui correspond à votre machine. Une seule.

Windows — WSL 2

Dans un terminal **PowerShell en administrateur** :

```
wsl --install -d Ubuntu-24.04
```

Redémarrez si Windows le demande, puis lancez Ubuntu depuis le menu Démarrer. Au premier lancement, il vous demande de créer un nom d'utilisateur et un mot de passe : retenez-les, le mot de passe servira pour `sudo` .

Vérifiez ensuite, toujours dans PowerShell, que vous êtes bien en version 2 :

```
wsl -l -v
```

La colonne `VERSION` doit afficher `2` .

 Si `Ubuntu-24.04` n'est pas reconnu

Listez ce qui est disponible avec `wsl --list --online` et prenez la ligne Ubuntu LTS proposée. Si la commande `wsl` elle-même est inconnue, lancez d'abord `wsl --update` .

macOS — Lima

Lima lance une VM Linux légère. Il s'installe avec Homebrew ; si vous ne l'avez pas, suivez d'abord les instructions de brew.sh.

```
brew install lima
limactl start --name=tp template://ubuntu-lts
```

La première création est longue : laissez-la terminer. Pour entrer dans la VM :

```
limactl shell tp
cd ~
```

 Le `cd ~` n'est pas optionnel

Lima monte le dossier personnel de votre Mac à l'intérieur de la VM, au même chemin et **en lecture seule**, et vous y dépose en arrivant. Vous ne pourriez rien y écrire. Le `cd ~` vous ramène dans le dossier personnel *de la VM*, qui est le seul endroit utilisable.

Vérifiez que vous y êtes :

```
pwd
touch essai && rm essai && echo "écriture ok"
```

`pwd` doit afficher un chemin commençant par `/home/` , et la seconde ligne doit afficher `écriture ok` . Si `pwd` commence par `/Users/` , vous êtes encore dans les fichiers du Mac.

Le nom exact de ce dossier varie selon la version de Lima — vous verrez un suffixe `.linux` , `.guest` ou autre. Peu importe lequel : le seul critère est d'être sous `/home/` et pas sous `/Users/` .

C'est à l'intérieur de cette VM que tout le TP se fera. Pour en sortir, `exit` . Pour l'éteindre, `limactl stop tp` .

Linux — rien à installer

Vous êtes déjà sur la bonne machine. Vérifiez seulement que vous avez `sudo` .

Secours — Killercodea

Si rien ne fonctionne, allez sur killercodea.com et ouvrez un *playground Ubuntu*. Vous obtenez un terminal Ubuntu dans le navigateur, avec les droits root, sans rien installer. Il existe aussi un *playground Docker*,

qui dépanne si l'étape 4 échoue.

⚠ Deux limites à connaître

La session est limitée dans le temps et **tout votre travail disparaît** à son expiration. Gardez-la comme plan B, pas comme solution par défaut — et dans ce cas, refaites l'étape 2 en arrivant en séance.

2. Les outils du TP

Dans votre VM Linux (ou dans Killercode) :

```
sudo apt update
sudo apt install -y util-linux libcap2-bin iproute2 procps curl tar
```

3. L'archive Alpine

Le TP part d'un système de fichiers Alpine minimal. Téléchargez-le maintenant, pour ne pas dépendre du réseau de la salle :

```
mkdir -p ~/tp-conteneur && cd ~/tp-conteneur
ARCH=$(uname -m)
BASE="https://dl-cdn.alpinelinux.org/alpine/latest-stable/releases/$ARCH"
FILE=$(curl -s "$BASE/" | grep -o "alpine-minirootfs-[0-9.]*-$ARCH\.tar\.gz" | head
-1)
curl -O "$BASE/$FILE"
ls -lh alpine-minirootfs-*.tar.gz
```

La dernière commande doit afficher une archive de quelques mégaoctets.

4. Docker Desktop

Téléchargez l'installateur sur docker.com/products/docker-desktop et prenez la version qui correspond à votre machine.

Windows

À l'installation, laissez cochée l'option « **Use WSL 2 instead of Hyper-V** ».

Une fois Docker Desktop lancé, ouvrez **Settings** → **Resources** → **WSL integration** et activez votre distribution **Ubuntu-24.04**. La commande `docker` sera alors disponible aussi bien dans PowerShell que dans votre Ubuntu de l'étape 1.

macOS

Prenez bien l'installateur correspondant à votre processeur : **Apple Silicon** ou **Intel**. Au premier lancement, macOS demande votre mot de passe pour installer un composant système.

 **Docker Desktop et Lima sont indépendants**

Installez les deux. Avoir Docker Desktop ne remplace pas la VM de l'étape 1, et inversement.

Linux

Sur Linux, Docker s'installe directement, sans Docker Desktop :

```
curl -fsSL https://get.docker.com | sudo sh
sudo usermod -aG docker "$USER"
```

Fermez puis réouvrez votre session pour que l'appartenance au groupe `docker` prenne effet.

Vérifier que tout est prêt

La VM Linux

Collez ce bloc entier dans votre terminal Linux :

```
echo "noyau : $(uname -sr)"
echo "machine : $(uname -m)"
for c in unshare chroot capsh ip ps curl tar; do
    command -v "$c" >/dev/null && echo "ok $c" || echo "MANQUE $c"
done
ls ~/tp-conteneur/alpine-minirootfs-*.tar.gz >/dev/null 2>&1 \
    && echo "ok archive Alpine" || echo "MANQUE archive Alpine"
sudo true && echo "ok sudo"
```

Docker

Dans le terminal où `docker` est disponible — PowerShell ou Ubuntu sous Windows, le Terminal de macOS, votre shell sous Linux :

```
docker --version
docker run --rm hello-world
```

Vous êtes prêt si

- le noyau affiché commence par `Linux` ;
- aucune ligne ne commence par `MANQUE` ;
- `ok sudo` s'affiche en dernière ligne ;
- `docker run --rm hello-world` affiche le message de bienvenue de Docker.

 En cas d'échec

Pour une ligne `MANQUE` , relancez la commande `apt install` de l'étape 2. Si `docker run` échoue, vérifiez d'abord que Docker Desktop est bien **lancé** — ce n'est pas un service qui démarre tout seul. Dans tous les cas, signalez le problème en début de séance.