

Multipass

Présentation

- Permet d'avoir une machine virtuelle Ubuntu facilement
- Les commandes et configurations de base sont identiques sous Windows/MacOs/Linux

Pré-requis

- Windows :
 - VirtualBox (possible mais déconseillé)
 - *Recommandé* : Windows version Pro (acheter une licence à quelques euros) pour pouvoir utiliser HyperV
- Linux
 - snap (pré-installé avec les versions récentes d'ubuntu)
- MacOS : aucun

Installation

Suivre les instructions depuis [la page de téléchargement/d'installation de multipass](#)

Raccourci vers l'instance primaire

Ctrl+Alt+U : démarre (créé si nécessaire) une machine virtuelle qui s'appelle **primary**

Voir ce raccourci : `multipass get client.gui.hotkey`

Supprimer ce raccourci : `multipass set client.gui.hotkey=""`

Commandes multipass

- `multipass list` : liste les machines
- `multipass start <nom>` : démarre une machine
- `multipass stop <nom>` : arrête une machine
- `multipass shell <nom>` : lance un shell dans la machine
- `multipass info <nom>` : affiche des informations détaillées
- `multipass mount <chemin_source> <nom>:<chemin_destination>` : effectue un point de montage
- `multipass delete <nom>` : affiche des informations détaillées
- `multipass purge` : supprimer définitivement une VM
- `multipass launch --name <nom>` : créer une VM
- `multipass launch --name <nom> --cloud-init <FICHIER>` : créer une VM avec fichier de configuration cloud-init

Exemple : création / lancement / arrêt et suppression d'une machine virtuelle

1. `multipass launch --name test --disk 20G`
2. `multipass list`

3. `multipass start test`
4. `multipass shell test`
5. `exit`
6. `multipass stop test`
7. `multipass delete test`
8. `multipass purge`

Création de la machine virtuelle et lancement

1. `multipass launch --name dev --disk 20G`
2. `multipass list`
3. `multipass shell dev`

Partage de fichiers

Utiliser `sftp` de préférence Autres solutions possibles **à éviter** : `multipass mount` ; `samba`

Se connecter en ssh

- Sur la machine hôte, avec `bash` ou `Git Bash`, créer les clés publiques et privées :

```
# Générer les clés publiques/privées
ssh-keygen -t ed25519 -m PEM
# afficher la clé publique
cat ~/.ssh/id_ed25519.pub
# copier le fichier
multipass transfer ~/.ssh/id_ed25519.pub dev:/home/ubuntu/host_key.pub
# vérifier
multipass exec dev -- sh -c 'cat ~/host_key.pub'
# ajouter la clé ssh aux clés autorisées
multipass exec dev -- sh -c 'cat ~/host_key.pub >> ~/.ssh/authorized_keys'
# vérifier la liste des clés ssh autorisées
multipass exec dev -- sh -c 'cat ~/.ssh/authorized_keys'
# suppression du fichier intermédiaire sur la VM
multipass exec dev -- sh -c 'rm ~/host_key.pub'

# configurer ssh
multipass shell dev
sudo nano /etc/ssh/sshd_config
# activer PubkeyAuthentication
    PubkeyAuthentication yes
# si nécessaire, activer les logs
# fichier log /var/log/auth.log
    LogLevel VERBOSE
# sauvegarder

sudo service sshd restart

# tester
ssh ubuntu@dev.mshome.net -i ~/.ssh/id_ed25519
```

Se connecter en sftp avec VSCode

- Installer l'extension SFTP pour VSCode (Natizyskunk est plus récente et mise à jour par rapport à liximomo)
- Dans la palette de commandes (Ctrl Shift P), saisir **SFTP: Config**
- Configuration sftp.json. Remplacer \ par / dans le chemin de **privateKeyPath**

```
{
  "name": "dev",
  "host": "dev.mshome.net",
  "protocol": "sftp",
  "port": 22,
  "username": "ubuntu",
  "remotePath": "/var/www/html",
  "uploadOnSave": true,
  "useTempFile": false,
  "privateKeyPath": "C:/Users/user/id_rsa"
}
```

- Dans la palette de commandes (Ctrl Shift P), voir et tester les commandes disponibles : **SFTP:**

Adresse IP fixe

Voir le script **multipass_newvm_ipfix.ps1**

1. Voir **multipass networks**
 2. voir **multipass launch --network**
 3. Voir **multipass get local.bridged-network** et **multipass set local.bridged-network**
- Sous Windows :

```
# cf. https://learn.microsoft.com/en-us/windows-server/virtualization/hyper-v/get-started/create-a-virtual-switch-for-hyper-v-virtual-machines
# cf. https://hyper-v.goffinet.org/gestion-du-reseau.html#71-connecter%C3%A9connecter-une-vm-%C3%A0-un-commutateur
#Requires -RunAsAdministrator

# LA MACHINE VIRTUELLE DOIT ÊTRE STOPPÉE

# Lister les cartes réseau
Get-NetAdapter

# Créer un switch virtuel
```

```
# paramètres
$vmName = "dev"
$switchName = "multipass-static"
$switchGw = "192.168.10.254"
$prefixLength = 24
$macaddress = "00:50:77:e6:f4:62"

# Création du switch virtuel
New-VMSwitch -SwitchType Internal -Name $switchName
# Récupération de l'interface réseau
$index=Get-NetIPInterface | Where-Object { $_.InterfaceAlias-Like
"*$switchName*" -and $_.AddressFamily -eq "IPv4" } | Select-Object -
ExpandProperty ifIndex
# Création de l'adresse IP
New-NetIPAddress -IPAddress $switchGw -AddressFamily IPv4 -PrefixLength
$prefixLength -InterfaceIndex $index
# Connexion de la machine virtuelle au switch virtuel
Add-VMNetworkAdapter -VMname $vmName -SwitchName $switchName -
StaticMacAddress $macaddress

# Générer une adresse MAC Aléatoire
# param(
#     [int] $len = 12,
#     [string] $chars = "0123456789abcdef"
# )
# $bytes = new-object "System.Byte[]" $len
# $rnd = new-object System.Security.Cryptography.RNGCryptoServiceProvider
# $rnd.GetBytes($bytes)
# $macraw = "0050"
# $len = $len - 4
# for( $i=0; $i -lt $len; $i++ ) { $macraw += $chars[ $bytes[$i] %
$chars.Length ] }
# $macaddress =
$macraw[0]+$macraw[1]+":"++$macraw[2]+$macraw[3]+":"++$macraw[4]+$macraw[5]+
":"++$macraw[6]+$macraw[7]+":"++$macraw[8]+$macraw[9]+":"++$macraw[10]+$macra
w[11]
```

Fichier yaml - netplan

```
# /etc/netplan/60-staticip.yaml
# bien vérifier le paramètre macaddress
network:
  ethernets:
    eth1:
      dhcp4: false
      match:
        macaddress: 00:50:77:e6:f4:62
      set-name: eth1
      addresses: [192.168.10.1/24]
  version: 2
...

```

Annexe : Windows/HyperV : réinitialiser le réseau multipass en cas de blocage

- Supprimer le fichier C:\Windows\System32\drivers\etc\hosts.ics
- Get-HNSNetwork | ? Name -Like "Default Switch" | Remove-HNSNetwork
- Restart-Computer
- cf. [<https://techsparx.com/linux/multipass/windows/troubleshoot.html>]

Voici les commandes `windows powershell en mode administrateur` :

```
```powershell
#Requires -RunAsAdministrator
if (Test-Path -Path 'C:\Windows\System32\drivers\etc\hosts.ics' -PathType Leaf)
{
 Remove-Item 'C:\Windows\System32\drivers\etc\hosts.ics'
}
Get-HNSNetwork | Where-Object Name -Like "Default Switch" | Remove-HNSNetwork
Restart-Computer
```

- Dans la palette de commandes (Ctrl Shift P), voir et tester les commandes disponibles : **SFTP**:

## Annexe : Windows Hyper-V : Augmenter la taille de la machine Virtuelle

Resize-VHD is the way to do it on Windows. You can also use Hyper-V Manager, right click on the instance you are interested in, select Settings..., then select the Hard Drive under IDE Controller 0. From there, you can see the path to the virtual disk. You can also click Edit under the Virtual hard disk and that will bring up the Edit Virtual Hard Disk Wizard. Click Next from here and you will be presented with several different options. Choose Expand -> Next and then fill out whatever size you want to make the disk. Click Next, review the change, and then click Finish.

[<https://docs.microsoft.com/en-us/powershell/module/hyper-v/resize-vhd?view=windowsserver2022-ps&viewFallbackFrom=win10-ps>]

[<https://github.com/canonical/multipass/issues/29>]

[Enlarge a virtual machine disk in VirtualBox or VMWare](#)

[<https://github.com/canonical/multipass/issues/62>]