

Kernel Virtualization Machine!

Jean-Marc Pouchoulon

Novembre 2024



1 Objectifs du TP et organisation.

1.1 Les objectifs de cette séance sont les suivants:

- Comprendre l'architecture d'une solution de virtualisation.
- Créer une machine virtuelle Kvm à l'aide de virt-manager et de virt-install.
- Utiliser la libvirt au travers du shell virsh.
- Mettre en réseau une machine virtuelle de différentes façons.
- Migrer une machine virtuelle d'un noeud KVM à un autre en utilisant un espace de stockage NFS.

2 Pré-requis, recommandations et notation du TP.

Les pré-requis sont les suivants:

- Avoir un PC sous Linux.
- La virtualisation consomme de la mémoire. Plus vous en aurez et mieux cela sera.

C'est l'enseignant qui décide du nombre de personnes dans un groupe de travail (de 1 à n). Il peut vous être explicitement demandé de faire valider votre travail au cours du TP par l'enseignant au fur et à mesure de votre avancement pour être noté. Un compte rendu réalisé au fil du TP est obligatoire. Faites impérativement un compte rendu au fur et à mesure avec des copies d'écran et les configurations mises en oeuvre.

Tous les travaux sont à déposer sur l'ENT indiqué par l'enseignant. Un travail doit être enregistré avec les noms des personnes dans le nom du fichier, et l'intitulé du fichier doit être clair (par ex: TP_intitulé_du_tp_Etudiant1_Etudiantn).

Les délais sont parfois et exceptionnellement négociables mais une fois fixés doivent être respectés sous peine d'une note nulle.

3 Environnement du TP

Vous disposez de machines virtuelles prêtes à l'emploi sur <http://moodle.iutbeziers.fr> si vous êtes à l'IUT ou sur un site donné par l'enseignant.

Le TP est dépendant de la bande passante vers l'internet. Au cas ou elle ne serait pas suffisante n'utilisez pas de "net-install" mais téléchargez une iso qui vous servira plusieurs fois.

3.1 Installations nécessaires au TP

Vous pouvez directement sur vos PC sous le user test.

Par convention, dans ce document les machines virtuelles sont désignées par le sigle VM (VMs au pluriel). Une machine créée avec KVM sera désignée par le sigle K-VM.

Vous pouvez utiliser l'OVA fournie par l'enseignant qui contient un serveur NFS (L'installation n'est pas nécessaire au début du TP) ou monter votre propre serveur.

Cette VM contient des images ISO (net -install) et permet de stocker les images des K-VMs afin de réaliser une migration d'une K-VM d'un noeud à un autre.

Pour accéder au partage NFS vous devrez passer *plus tard au cours du TP (Cf 5.2)* ces commandes:

```
mkdir /var/lib/libvirt/images/nfs4
mount -t nfs4 -o sec=sys nfsserveur.example.com:/kvm /var/lib/libvirt/images/nfs4
```

Vous pouvez valider que votre machine supporte KVM par ces deux commandes :

```
# le support des flags de virtualisation sur le CPU.
grep -E '(vmx|svm)' /proc/cpuinfo
virt-host-validate
```

Vous pouvez obtenir la complétion sur les commandes virsh. Pour cela suivez les instructions de <https://github.com/LuyaoHuang/virsh-bash-completion>

La commande suivante¹ permet de lister les versions d'OS reconnues par la libvirt sur votre distribution:

```
osinfo-query os
```

3.2 URLs pour réaliser une net install

Les URLs suivantes permettent de démarrer une "net install" pour les K-VMs. Il s'agit d'une installation avec un boot sur un installeur distant.² Vérifiez les URL avant de créer votre VM.

- Ubuntu
 - <http://fr.archive.ubuntu.com/ubuntu/dists/focal/main/installer-amd64>
 - <http://fr.archive.ubuntu.com/ubuntu/dists/jammy/main/installer-amd64>
- Centos:
 - http://mirror.centos.org/centos/7/os/x86_64
 - http://mirror.in2p3.fr/pub/linux/rocky/8/BaseOS/x86_64/os/
 - http://mirror.in2p3.fr/pub/linux/centos-stream/9-stream/BaseOS/x86_64/os/
- Debian: <http://ftp.fr.debian.org/debian/dists/stable/main/installer-amd64/>

Vous y accéderez via ssh avec l'option -X. Vous pouvez plus simplement installer KVM/LIBVIRT sur votre machine.

1. Installez le package libosinfo-bin pour avoir la commande

2. Pour "centos-9 stream" il faut un disque de 10 G minimum

4 Création de machines virtuelles K-VMs

4.1 Création de VMs avec virt-manager

4.1.1 Accès à virt-manager

virt-manager est une interface graphique qui permet de manager des K-VMs. Vous pouvez accéder à cette interface en la lançant depuis votre VM. Le tunelling X l'affichera sur votre PC (Si vous êtes sous Windows il vous faut un serveur X).

Pour sortir d'une console série sur une K-VM utilisez les touches CTRL SHIFT 5.

Il faut aussi que le port série de la console soit configuré dans la K-VM:

```
systemctl start serial-getty@ttyS0
systemctl enable serial-getty@ttyS0
```

4.1.2 Installation d'une VM Rocky avec virt-manager

Installez une VM Rocky de 2 Go de RAM et de 11 Go de disque à l'aide de virt-manager. Vous essayerez de faire une NetInstall en spécifiant lors du processus d'installation l'utilisation du serveur (ftp|http) et les URL ci-dessus. Vous pouvez aussi récupérer sur ce même serveur une iso complète si la net-install prend trop de temps. Lors de l'installation pensez à connecter le réseau de la K-VM.

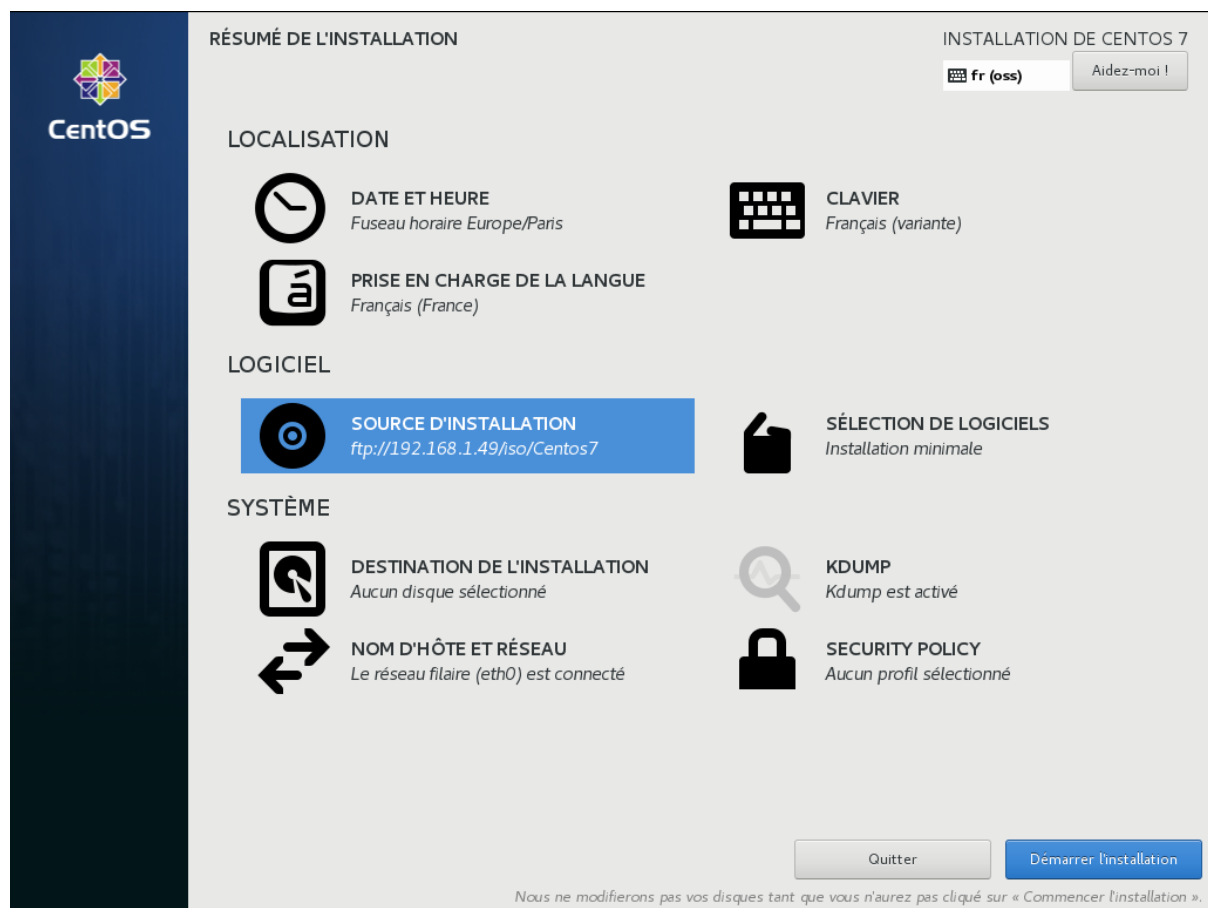


FIGURE 1 – Install K-VM

Attention si vous travaillez avec ssh avec virt-manager en mode "remote" vous devez autoriser le serveur SPICE à recevoir des connexions sur toutes les interfaces de la machine.

Maintenant que cette K-VM est installée nous allons utiliser *le shell virsh* pour la manipuler en ligne de commandes.

Votre utilisateur doit appartenir au groupe "libvirt".

```
sudo usermod -G libvirt -a test
# vérifier avec la commande suivante
getent group
```

Par défaut le "VIRtuel SHell" virsh va essayer de travailler avec les droits de votre utilisateur. Cette approche plus sécurisée néanmoins limite certaine option lors de la création de votre environnement virtuel. Il est conseillé d'exporter cette variable d'environnement en ajoutant la ligne suivante dans le fichier `/.bashrc` de votre utilisateur.

```
export LIBVIRT_DEFAULT_URI=qemu:///system
```

pour la prise en compte de la modification dans votre shell faire:

```
source ~/.bashrc
```

1. Récupérez des informations sur le node KVM depuis le SHELL.
2. Listez les domaines (comprendre les K-VMs) actifs de votre noeud Kvm.
3. Listez les domaines actifs et inactifs de votre noeud Kvm.
4. Démarrez la K-VM.
5. Faites en sorte que cette K-VM démarre automatiquement au démarrage de l'hôte.
6. Récupérez des informations la K-VM (en interrogeant son "domain".)
7. Faites un "gracefull shutdown" de la K-VM. Redémarrez là , puis forcez l'arrêt de la machine.

4.2 Création d'une K-VM Debian avec virt-install

1. Bâissez une K-VM Debian en utilisant la commande `virt-install` à partir d'une iso minimal et une console SPICE.

Elle aura les caractéristiques suivantes:

- 1 VCPU
 - 1024 Go de RAM
 - 5 Gigas de disque.
2. Récupérer des informations sur la vm (`virsh dominfo/schedinfo/domifilist/domblklist/vcpucount ...`)
 3. Attribuez à froid deux V-CPU à cette K-VM.

4.3 Création de VMs avec virt-builder

1. `virt-builder` permet de créer rapidement une VM à partir d'une image téléchargée sans avoir à la construire. Utilisez cette commande afin de personnaliser l'image de rocky (taille 10G `password=root`)³ Utilisez-la ensuite pour créer une VM avec `virt-manager` à partir de cette image modifiée ou utilisez `virt-install`.

4.4 Création de VMs avec virt-customize

L'autre format de disque de VM est `qcow2`.

1. Utilisez `virt-customize` afin de modifier l'image `qcow2`⁴ `debian 11` avec un mot de passe "root". Utilisez-la ensuite pour créer une VM avec `virt-manager` à partir de cette image modifiée.

3. Vous aurez peut être à positionner la variable `LIBGUESTFS_BACKEND=direct`

4. <http://cloud.debian.org/images/cloud/bullseye/latest/>

5 Découverte de l'architecture KVM

5.1 Gestion du réseau

1. Listez les bridges virtuels de la machine avec `brctl` et `virsh` ? Quel est le nom du bridge utilisé par la K-VM Debian créée précédemment ?
2. Faites un petit schéma de l'accès de la K-VM au réseau de la salle. Quel est le principe appliqué et l'outil utilisé par KVM pour mettre en oeuvre cet accès au réseau.
3. Créez un nouveau bridge appelé `monbridge`. Pour cela "dumpez" la configuration du "default network" afin de générer un nouveau fichier et modifiez-le afin de créer le nouveau réseau.
4. Rattachez la K-VM Debian à `monBridge`.
5. Créez deux K-VMs en la rattachant à des interfaces `macvtap`. Explorez les différents modes de ce type de réseau.

6 Utilisation de KCLI

`kcli` est un package Python qui permet d'automatiser la gestion des K-VMs.

1. En vous inspirant de la documentation suivante <https://github.com/karmab/kcli> utilisez-le pour manipuler une VM debian (prenez la version stable à la date du TP).

6.1 Migration d'une K-VM d'un noeud KVM à l'autre

Créez une nouvelle VM dans VMWareWorkstation ainsi qu'une nouvelle K-VM bâtie sur le pool de stockage NFS v4. Avec l'aide de `virt-manager` migrez la K-VM d'un noeud à l'autre. Vos VMs doivent être mises à l'heure avec le protocole NTP indispensable pour NFS.

6.2 Clonage d'une K-VM et gestion du stockage

1. Listez les pools de la machine.
2. Créez un clone (`clone-debian`) d'une K-VM avec `virt-clone`.
3. Créez un nouveau disque `/dev/vdb` de 2 Gigas et rattachez-le à la K-VM `clone-debian`. Vérifiez-le en vous connectant avec `virsh console`. Détachez le disque et supprimez le pool puis le disque.
4. Supprimez le disque et le clone.