

Virsh - Commande

Utilisation de la commande

Virsh

Nous allons voir ici l'Utilisation de la commande **Virsh** suite à l'installation de notre **Xen** ou encore **KVM**. Celle-ci est intégrée à la librairie **libvirt** et permet donc la gestion des machines virtuelles.

Virsh

Bien sûr, il est possible d'utiliser l'outil graphique [Virt-manager](#) mais l'outil en ligne de commande nous permettra surtout l'écriture de script par la suite.

Pour commencer, dans son mode le plus simple, **Virsh** s'exécute en mode interactif :

```
[root@localhost ~]# virsh
Bienvenue dans virsh, le terminal de virtualisation interactif.

Taper : « help » pour l'aide ou « help » avec la commande
        « quit » pour quitter

virsh #
```

Ensuite, pour lister toutes les machines actives, il faudra effectuer la commande suivante :

```
# virsh list
```

Mais aussi, vous pouvez ajouter l'option **-inactive** pour les machines inactives et **-all** pour lister tout sans faire de différence :

```
# virsh list --inactive
# virsh list --all
```

Puis, pour démarrer, arrêter ou encore redémarrer une machine, nous pourrions utiliser les commandes suivantes :

```
# virsh start VmName
# virsh shutdown VmName
# virsh reboot VmName
```

D'autre part, pour forcer l'arrêt d'une VM :

```
# virsh destroy VmName
```

Et si vous voulez mettre en pause une VM :

```
# virsh suspend VmName
# virsh resume VmName (pour la reprise d'une VM précédemment mise en pause)
```

Aussi, comme vu dans les articles précédents sur [Xen](#) et [KVM](#), vous pouvez vous connecter via :

```
# virsh console
```

Mais encore, pour afficher les informations d'une machine virtuelle :

```
# virsh dominfo VmName
ID :      2
Nom :     VmName
UUID :    df7ad48b-f752-4f5d-be4b-84407c6fe5f9
Type de SE : hvm
État :    en cours d'exécution
CPU :     1
Temps CPU : 19,2s
Mémoire Max : 1048576 KiB
Mémoire utilisée : 1048576 KiB
Persistent: yes
Démarrage automatique : disable
Managed save: no
Security model: selinux
Security DOI: 0
Security label: system_u:system_r:svirt_t:s0:c758,c991 (enforcing)
```

Vous pouvez aussi afficher le fichier de configuration **xml** de la machine virtuelle :

```
# virsh dumpxml VmName
```

Puis modifier cette configuration ici :

```
# virsh edit VmName
```

De plus, vous pouvez voir la liste des pools et les noms des volumes utilisés par celui-ci :

```
# virsh pool-list
Nom          État   Démarrage automatique
-----
images       actif  yes
# virsh vol-list images (images est ici le nom du pool pour la commande virsh vol-list NomDuPool)
Nom          Chemin
-----
VmDebianTest.img  /var/lib/libvirt/images/VmDebianTest.img
VmDebianTestKVM.img /var/lib/libvirt/images/VmDebianTestKVM.img
VmFedora29.img    /var/lib/libvirt/images/VmFedora29.img
VmFedora31.img    /var/lib/libvirt/images/VmFedora31.img
```

Petit aparté, pour manipuler les images disques, il faudra vous aider de **qemu-img** :

```
Informations sur une image:
qemu-img info fichier.img

Étendre la taille de l'image:
qemu-img resize fichier.img +tailleG
```

Ensuite, du côté réseau, il est possible de lister la liste des réseaux virtuels :

```
# virsh net-list --all (ici nous listons les réseaux actifs et inactifs)
```

Et il est possible d'avoir un peu plus d'informations sur un réseau en particulier via la commande suivante :

```
# virsh net-info NomDuReseau
```

Par ailleurs, vous pouvez aussi activer ou désactiver un réseau virtuel avec les commandes suivantes :

```
# virsh net-start NomDuReseau
# virsh net-destroy NomDuReseau
```

Virsh permet également de créer et gérer des **snapshot**. Pour la création et suppression d'un **snapshot**, il faudra utiliser les commandes suivantes :

```
# virsh snapshot-create VmName
# virsh snapshot-delete VmName
```

Pour lister les différents **snapshot** d'une VM :

```
# virsh snapshot-list VmName
```

De plus, vous pouvez revenir au **snapshot** créé précédemment via cette commande :

```
# virsh snapshot-revert VmName
```

Pour finir, celle-ci vous permettra d'avoir des informations à propos de la **node** :

```
# virsh nodeinfo
modèle de CPU : x86_64
CPU : 2
Fréquence de la CPU : 4008 MHz
socket(s) CPU : 1
Coeur(s) par emplacements : 2
Thread(s) par coeur : 1
cellule(s) NUMA : 1
Taille mémoire : 3848800 KiB
```

Revision #1

Created 4 February 2021 23:33:44 by garfi

Updated 4 February 2021 23:34:15 by garfi