




# Virt-manager : Un gestionnaire graphique de machine virtuelle

- Objet : Utilisation de Virt-manager
- Niveau requis :  
[débutant](#)
- Commentaires : *Gérer ses machines virtuelles avec Virt(ual Machine) Manager.*
- Suivi :  
[à-tester](#)
  - Création par  [gutsblack](#) le 29/07/2011  
[obsolète](#)
  - Réécriture par  [agp91](#) le 02/05/2024
- Commentaires sur le forum : [ici](#) <sup>1)</sup>



Page réécrite... A valider.

**Nota** : Contributeurs, les  sont là pour vous aider, supprimez-les une fois le problème corrigé ou le champ rempli !

## Introduction

L'usage de **machines virtuelles** <sup>2)</sup> permet (entre autres) de tester.  
... De tout tester :

- Tester l'installation et l'usage d'un système d'exploitation.
- Tester un système live sans redémarrer sa machine.
- Tester un programme.
- Tester une configuration particulière.
- Tester la communication entre plusieurs machines sur une seule.
- Jouer avec un système sans redouter de le casser.
- Appliquer les tutos diffusés sur le net.
- Etc.

**Les machines virtuelles peuvent être clonées** (copiées pour fournir une nouvelle machine virtuelle). C'est très pratique pour réaliser des machines virtuelles avec toutes la même configuration matériel virtuel et le même système.

Une autre fonctionnalité est **la prise d'instantané d'une machine virtuelle** à chaud (en cour de fonctionnement). Cela permet de restaurer (à chaud) la machine dans l'état de la prise de l'instantané.

Bref, la virtualisation est plus qu'un outil pratique...  
C'est un jouet. 😊

... Évidemment, [la virtualisation](#) n'est pas qu'un jeu, elle est aussi couramment utilisée dans le développement, le cloisonnement de services ou d'utilisateurs, le partage des ressources, le portage

matériel et logiciel, les tests en tout genre, etc.

C'est aussi un outil éducatif, comme par exemple : En ajoutant un second (ou plusieurs) disque(s) virtuel(s), pour s'initier au partitionnement, au RAID logiciel et aux différents systèmes de fichier.

**Note** : Nous nommons volontiers **machine virtuelle** par son acronyme anglais **VM** (pour **Virtual Machine**).

### Virt-manager,

Est une application bureau (une interface graphique) qui permet la gestion de machines virtuelles. Il utilise l'**hyperviseur** <sup>3)</sup> **KVM** <sup>4)</sup> et l'**émulateur** <sup>5)</sup> **QEMU** <sup>6)</sup>, à travers la librairie **libvirt** <sup>7)</sup>. En plus du couple QEMU/KVM, virt-manager peut utiliser les hyperviseurs **xen** <sup>8)</sup>, **bhyve** <sup>9)</sup> et les **isolateurs** <sup>10)</sup> **OpenVZ** <sup>11)</sup> et **LXC** <sup>12)</sup>.

Il permet de créer/supprimer, configurer, démarrer/arrêter/suspendre, sauvegarder, dupliquer (cloner) des machines virtuelles.

Localement ou sur une machine distante via ssh <sup>13)</sup>.

Il permet aussi de créer des réseaux virtuels pour que les machines virtuelles puissent (si souhaité) communiquer entre elles et avec l'extérieur (si l'hôte est connecté à un réseau).

Pour obtenir l'affichage graphique d'une machine virtuelle (permettant ainsi d'y installer un système graphique et de l'utiliser), **virt-manager** dispose du client **virt-viewer**. Tous deux utilisent le protocole **VNC** <sup>14)</sup> ou **SPICE** <sup>15)</sup> pour communiquer avec la console graphique. (Le protocole VNC ne véhicule que le flux graphique et ne prend pas en charge le flux audio).

Ce tuto propose :

- D'installer virt-manager.
- De configurer un réseau virtuel et un pool de stockage.
- De créer une VM sans disque pour tester un système live.
- De créer une autre VM (avec un disque) pour tester l'installation d'un nouveau système.
- De cloner cette dernière, pour l'usage.
- D'ouvrir ; démarrer ; suspendre/arrêter/redémarrer ; sauvegarder et supprimer une VM (le clone).
- De prendre un instantané d'une VM avant de jouer.
- De configurer une VM pour émuler le CDRom ; modifier la RAM et ajouter un disque virtuel.

L'hyperviseur **KVM** utilise la "**virtualisation matériel\***" intégrée au microprocesseur (**VT-x** pour les processeurs Intel et **AMD-V** pour les processeurs AMD).

La commande suivante permet de savoir si un microprocesseur dispose une de ces technologies.

Ce qui devrait être le cas avec un microprocesseur récent.



```
grep -E 'vmx|svm' /proc/cpuinfo && /dev/null && echo "OK" || echo  
":/ KVM non compatible"
```



Notes : Les technologies **VT-x** ou **AMD-V** doivent-être activées dans le **BIOS**.

## Installation

Comme avant chaque installation, **mettre à jour le système** :

```
apt-get update && apt-get upgrade
```

Puis, **installer le paquet** virt-manager :

```
apt-get install virt-manager
```

Pour obtenir le privilège d'utiliser virt-manager, un utilisateur doit appartenir au groupe libvirt. Nous utilisons donc la commande **adduser** pour **ajouter notre utilisateur à ce groupe**.

```
adduser <utilisateur> libvirt
```

Bien sûr, *<utilisateur>* est à remplacer par le nom de votre utilisateur.



Se déconnecter et se reconnecter de sa session graphique pour que cette modification soit prise en compte.

## Prérequis

La première machine virtuelle que nous créerons n'aura pas de système installé et ne disposera pas de disque. Elle sera démarrée depuis un système live. La seconde disposera d'un disque et nous y installerons un système.

Nous avons donc **récupéré les images iso**<sup>16)</sup>

- Du système live de Debian Facile (**DFiso**), sur la page : <https://debian-facile.org/dflinux/isos/>.
- Et d'installation du système **GNU/Debian** stable (GNU/Debian 12 Bookworm à l'écriture du tuto) par le réseau (net-install), à l'adresse :  
<https://cdimage.debian.org/debian-cd/current/amd64/iso-cd/>

Nous utiliserons ces images pour émuler le lecteur de CDRom de nos machines virtuelles. Au premier démarrage (après création), une machine virtuelle est configurée temporairement pour être démarrée (bootée) depuis le CDRom.

Une fois téléchargées, les intégrités contrôlées, les images ont été copiées dans le répertoire `~/iso`, que nous avons préalablement créé, dans notre répertoire utilisateur.

```
ls ~/iso
```

```
debian-12.5.0-amd64-netinst.iso  debian-facile-12.5_amd64.iso
```

## Détails de l'hyperviseur

Ouvrons virt-manager depuis le menu général **Menu Applications > Système > Gestionnaire de machines virtuelles**.

Ou dans un terminal fenêtre, avec la commande **virt-manager**.

```
virt-manager
```



Par défaut, Virt-manager utilise l'hyperviseur **QEMU/KVM** local.



**Sélectionnons le.**

Puis, un **double-clic** dessus,

ou **depuis le menu Edition > Détails de la connexion**



Affiche les détails de l'hyperviseur, répartis en 3 onglets :

- Affichage
- Réseaux virtuels
- Stockage



**Ouvrons l'onglet Réseaux virtuels**

## Réseau



Pour l'instant, aucun réseau virtuel n'existe.

Pour que nos machines virtuelles puissent communiquer, nous devons en émuler au moins un.

**Clic sur le bouton +** (en bas à gauche) pour en créer un.



Nous nommons le réseau **net42**.

Le **mode NAT** est retenu.

Puis, **Clic sur le bouton** Terminer

En plus d'un **mode isolé**, où une machine virtuelle ne communique qu'avec l'hôte. Nous utilisons trois principaux types de réseaux :

- Le **mode NAT** : Un réseau local virtuel est créé. Les machines virtuelles qui y sont connectées peuvent communiquer entre elles. Elles peuvent joindre l'extérieur, sous le masque de l'hôte (avec l'adresse IP de l'hôte). Leurs adresses IP étant masquées, elles ne sont pas joignables depuis l'extérieur.
- Le **mode route** : Un réseau local virtuel est créé. Les machines virtuelles qui y sont connectées communiquent entre elles. Elles communiquent avec l'extérieur avec leurs adresses IP, appartenant au réseau virtuel. Elles sont accessibles depuis l'extérieur, à condition de connaître (configurer) la route qui mène au réseau virtuel (l'adresse de l'hôte).
- Le **mode pont** : Aucun réseau n'est créé. Le matériel virtuel créé permet de connecter les machines virtuelles au réseau local de l'hôte. Les machines virtuelles qui y sont connectées communiquent entre elles et avec toutes les machines du réseau de l'hôte. Elles y ont leurs propres adresses IP. Elles sont donc joignables depuis l'extérieur.



Par défaut, **une adresse IPv4 à été attribuée** au réseau virtuel.

Ici, le réseau 192.168.100.0/24.

**DHCP**<sup>17)</sup> y est actif, nous permettant ainsi de ne pas nous soucier de la configuration réseau de nos machines virtuelles.

La plage d'adresses qu'il attribuera est configurée entre l'hôte 128 et l'hôte 254. Soit la possibilité d'attribuer 128 adresses.

Maintenant, **Ouvrons l'onglet** Stockage

---

## Stockage

Virt-manager accède au système de fichiers de l'hôte à travers des **pools** (réservoirs).

**Un pool** est un chemin de répertoire du système de fichiers de l'hôte, mémorisé par un nom. Une fois déclaré, un pool permet à virt-manager d'accéder à son contenu.



**Il existe déjà un pool nommé** Default.

Il a pour chemin /var/lib/libvirt/images.

Nous l'utiliserons pour stocker les fichiers images des disques virtuels de nos machines virtuelles.

**Nous devons en créer un autre,**

Pour accéder aux images situées dans notre répertoire ~/iso.

Alors, **cliquez sur le bouton +** (en bas à gauche) pour créer un nouveau pool.

---

Nous **nommons le nouveau pool iso**



**Son chemin** /home/<utilisateur>/iso **est renseigné**.

(Ici, DF est le nom de notre <utilisateur>, remplacez le par le votre.)

Remarque : le bouton Parcourir peut-être utilisé.

**Cliquez sur le bouton Terminer** pour créer le pool.

---



Maintenant que **le pool iso** existe, son contenu est accessible.

---



Voilà, nous sommes prêt,

Nous avons tout ce qu'il nous faut pour créer et démarrer des machines virtuelles.

Fermons la fenêtre QEMU/KVM - Détails de connexion.

- Avec le **bouton de fermeture d'une fenêtre x** (en haut à gauche)
  - Ou avec **la combinaison des touches Ctrl+W**
  - Ou depuis **le menu Fichier / Close**.
- 

## Nouvelle VM

Nous allons dans ce tuto créer 3 machines virtuelles.

- La première nous permettra d'utiliser le système live **DFiso** diffusé sous la forme d'une image iso.
- Dans la seconde, nous installerons le système **GNU\Debian 12.5**.
- La troisième sera créée par le **clonage** de la seconde.



Depuis la fenêtre Gestionnaire de machines virtuelles.

**La connexion à l'hyperviseur**, où nous souhaitons créer une nouvelle machine virtuelle, **doit être sélectionnée**.

---



Pour créer une nouvelle machine virtuelle :

**Clic sur le bouton** Créer une nouvelle machine virtuelle



**Ou depuis le menu** Fichier > Nouvelle machine virtuelle

---

La création d'une nouvelle machine va se réaliser en 5 étapes :

1. Choix de la méthode d'installation du système d'exploitation.
2. Choix du média d'installation.
3. Paramétrages mémoire et processeur (RAM et CPU attribués à la machine virtuelle).
4. Création du disque virtuel attribué à la machine virtuelle.
5. Nom de la machine virtuelle et choix du réseau virtuel auquel elle est connectée.

**C'est parti...**

## Systeme live

Commençons par une chose simple :

Une machine virtuelle sans disque, sans système installé. Utilisant un système live graphique prêt à l'emploi.



### Étape 1

Nous souhaitons démarrer le système depuis une image iso.

Cette méthode d'installation est celle par défaut.

**Clic sur le bouton** Forward

---



### Étape 2

Nous allons sélectionner l'image iso de démarrage.

**Clic sur le bouton** Parcourir

---



Dans le panneau de gauche,  
Nous **sélectionnons le pool** iso.

Et dans le panneau de droite,  
Nous **choisissons l'image de DFiso**.  
Ici debian-facile-12.5\_amd64.iso.

Et, **clic sur le bouton** Choose Volume.

---



Nous sommes de retour sur la fenêtre de la 2ème étape.

Nous devons sélectionner le type de système d'exploitation que nous allons utiliser.

En bas, **nous décochons l'option** Détecter automatiquement depuis la source/medias d'installation.

Puis dans le champs de saisie juste au dessus, **nous saisissons** ge.

Un menu de choix s'affiche,  
Nous **sélectionnons** Generic Linux 2022.

**Clic sur le bouton** Forward pour valider.



### Étape 3

Les valeurs proposées nous conviennent,  
**Clic sur le bouton** Forward.

---



### Étape 4

Nous utilisons un système live, nous n'avons donc pas besoin de disque.

Nous **décochons l'option** Activer le stockage pour cette machine virtuelle.

Puis, **clic sur le bouton** Forward.

**Remarque** : Certains systèmes live, permettent de s'installer sur un disque. C'est le cas du système DFiso.

... Si c'est ce qui est souhaité, alors il faut évidemment ne pas décocher l'option.  
Pour demander la création du disque virtuel.





## Étape 5

La machine virtuelle est **nommée** DF12live.

Nous contrôlons que le réseau **net42** soit sélectionné.

Et **clic sur le bouton** Terminer pour lancer la création de la machine virtuelle.



La console graphique de la machine virtuelle est affichée.

La machine est démarrée...

Le menu de démarrage de **DFiso** est affiché.

En appuyant sur la touche Entrée, nous choisissons de tester DF12.



Nous patientons...

Puis, pour adapter la taille de la console, à la résolution de la session ouverte,

Nous **utilisons le menu** Afficher > Redimensionner à la taille de la VM



Yes, voilà qui est mieux 😎



Par curiosité, nous utilisons l'outil d'inspection À propos de ma debian que nous offre DF12.



Quand nous avons fini de jouer, nous éteignons le système, ce qui éteint aussi la machine virtuelle.

Maintenant que la machine virtuelle est arrêtée, nous avons perdu la configuration de l'émulation du CDROM.

Nous l'avons vu plus haut : Virt-manager configure, l'émulation du CDROM depuis une iso,

uniquement pour le premier démarrage de la machine virtuelle.

Si nous souhaitons que l'émulation persiste (par exemple pour continuer à utiliser DFiso sans l'installer).

Il nous vaut l'indiquer dans les configurationS (détails) de cette machine (voir [Emuler le CDROM](#)).

## Nouveau système



Comme précédemment,

**Dans la fenêtre** Gestionnaire de machine virtuelle,

L'hyperviseur QEMU/KVM **sélectionné**,

**Clic sur le bouton** Créer une nouvelle machine virtuelle



### Étape 1

L'installation depuis une image iso est sélectionnée par défaut,

Parfait !

**Clic sur le bouton** Forward



### Étape 2

Nous **choisissons l'image d'installation du système GNU/Debian**.

En bas, nous **décochons l'option** Détecter automatiquement depuis la source/medias d'installation

Et Nous **sélectionnons** Generic Linux 2022.

Puis **clic sur le bouton** Forward.

(Pour plus de détail, voir plus haut à la même étape.)



### Étape 3

Nous attribuons 2048 Mo de RAM .

**Clic sur le bouton** Forward.

---



### Étape 4

Nous allons installer un système, il nous faut donc un disque dur.

Le disque sera émulé dans un fichier (une **image qcow2**<sup>18)</sup>) placé dans le pool de stockage Default.

Une taille de 20 Go est choisie.

Puis, **clic sur le bouton** Forward.

---



### Étape 5

La machine virtuelle est **nommée** deb12xfce.

Car ici, nous installerons le système avec le bureau xfce4.

... Mais choisissez le bureau que vous souhaitez.

Nous contrôlons que le réseau **net42** soit sélectionné.

Et **clic sur le bouton** Terminer pour lancer la création de la machine virtuelle.

---

La console graphique de la machine virtuelle est affichée.

La machine est démarrée.

Le menu d'installation GNU/Debian est affiché.



Après l'installation du système et de son redémarrage, la console graphique doit être redimensionnée pour afficher la résolution du bureau de la machine invitée.

---

## Clonage



Le clonage de machine virtuelle est un outil puissant et pratique.  
Il permet d'obtenir rapidement une machine virtuelle en en copiant (clonant) une autre.

Pour cloner une machine virtuelle, elle doit être à l'arrêt.

Dans la fenêtre Gestionnaire de machine virtuelle,

**Clic droit** sur la ligne de la machine virtuelle que nous voulons cloner (ici deb12xfce),  
Pour **lancer la commande Cloner... du menu contextuel**.



La fenêtre suivante nous demande de nommer la nouvelle machine que nous allons créer. Et de sélectionner le(s) disque(s) que nous souhaitons cloner (la machine virtuelle d'origine pourrait en disposer plusieurs).

Nous **nommons la machine virtuelle** tutoz.  
Puis **clic sur le bouton Cloner**



L'image disque de la MV d'origine est copiée avec le nom de la nouvelle.

... Et la machine virtuelle tutoz est créée.

Dans les deux sections précédentes, les machines virtuelles que nous avons créé ont été automatiquement démarrées et leurs consoles affichées.

Par le clonage, ce n'est pas le cas, la machine virtuelle créée, est livrée à l'arrêt.

Pour démarrer une machine virtuelle et afficher sa console graphique, voir la section suivante.

## Management

Nous venons de voir comment créer des machines virtuelles.

Ici, nous montrons les autres fonctions de management, que nous propose virt-manager :  
**Ouvrir ; Démarrer ; Suspendre ; Prendre/Appliquer un instantané ; Arrêter/Redémarrer ; Et supprimer** des machines virtuelles.

La plupart de ces opérations de managements se réalisent depuis :

- La fenêtre Gestionnaire de machine virtuelle.
- Et la fenêtre Console/Détails.

## Ouvrir

Ouvrir **la fenêtre Console/Détails** d'une machine virtuelle,  
Permet d'afficher :



- Sa **console graphique**, le terminal (écran, clavier et souris) qui nous permet de communiquer avec elle.
- Son **interface de configuration** (les détails).
- Et son **interface de gestion des instantanés**.

**Nous ouvrirons la fenêtre** Console/Détails,  
**Depuis la fenêtre** Gestionnaire de machine virtuelle :

- Avec un **double-clic** sur la machine virtuelle.
- Ou avec la **commande Ouvrir du menu contextuel** de la machine virtuelle.
- Ou encore en **sélectionnant la machine virtuelle**, puis **clic sur le bouton Ouvrir**.

La fenêtre Console/Détails peut être ouverte, que le machine virtuelle soit, démarrée ou pas.

---

Elle permet d'afficher l'interface de la **Console graphique**, celle des **Détails** ou celle de la **Gestion des instantanés**,  
A l'aide de trois boutons :



---

L'interface de la Console graphique dispose du **menu Touche d'envoi**.



Il permet **d'envoyer à la machine virtuelle des séquences de touches**,  
Qui si, elles étaient produites au clavier, seraient plutôt envoyées, à l'hôte qu'à la machine virtuelle.

## Démarrer



**Depuis la fenêtre** Gestionnaire de machine virtuelle

- La machine virtuelle sélectionnée, **clic sur le bouton Ouvrir**.
- **Clic droit** pour afficher le menu contextuel, puis **clic sur la commande Démarrer**.



Depuis la fenêtre Console/Détails :

- **Clic sur le bouton** Démarrer
- **Depuis le menu** Machine virtuelle > Démarrer



## Instantané

**Un instantané** d'une machine virtuelle est une sauvegarde de son système, C'est **un cliché** (une image) du disque (virtuel), tel qu'il est à l'instant.

Il peut-être réalisé sur une machine virtuelle éteinte, démarrée et même en court d'usage. La réalisation d'un cliché n'arrête pas le système, comme le fait la commande [Sauvegarder](#) (voir plus bas).

**Un instantané créé**, peut-être utilisé à tout moment, pour restaurer le système tel qu'il était lors de la prise de l'instantané.



Virt-manager permet de gérer graphiquement, plusieurs instantanés (clichés) pour une machine virtuelle. Pour les manipuler, il met à disposition une interface de gestion accessible par son bouton depuis la fenêtre Console/Détails. (Voir plus haut [Ouvrir une VM.](#))



Hihi, ici, nous allons jouer à Sid<sup>19)</sup>.

### Démarrons le clone...

Avant de jouer, **faisons un instantané** (un cliché).

**Ouvrons la fenêtre** Console/détails,  
Puis nous **affichons l'interface de** Gestion des instantanés **avec son bouton** (voir ci-dessus).

Et, en bas, à gauche, **Clic sur le bouton** + pour créer un nouveau cliché.



Si la machine virtuelle est démarrée, une capture d'écran est réalisée.:  
Elle sera utilisée pour présenter le cliché.

Nous pouvons nommer le cliché à notre convenance.

**Clic sur le bouton Terminer**



Le cliché est créé.

Voilà qui est bien fait 😊



Jouons maintenant,

**Affichons la console graphique** avec son bouton.

Puis, nous **utilisons le menu** Touche d'envoi > Ctrl+Alt+F1, pour envoyer cette séquence de touches à la machine virtuelle. Afin d'afficher la (première) console virtuelle de la machine virtuelle.



N'utilisez pas le clavier pour envoyer cette séquence de touche.

Car elle serait envoyée à l'hôte. La console virtuelle affichée serait celle de l'hôte et non celle de la machine virtuelle invitée dans sa console graphique.

La console virtuelle est affichée, il nous est demandé un nom d'utilisateur.

**Connectons-nous en root**

(la machine virtuelle nous appartient. Nous y avons nous même installé le système. Nos connaissons donc le mot de passe root.)



Contrôlez que vous utilisez bien la console de la machine virtuelle ;  
Et non pas, un terminal fenêtre de l'hôte.

Et, **Exécutons la commande qui tue.**

Celle, que tous déconseillent : `rm -rf /`

```
rm -rf /
```

```
rm: il est dangereux d'opérer récursivement sur '/'  
rm: utilisez --no-preserve-root pour inhiber cette mesure de sécurité
```

Haha, même le système nous le déconseille...

**Forçons le !**

```
rm -rf --no-preserve-root /
```

... Rooh, root ne peut pas tout supprimer.

Linux (le noyau) serait-il plus puissant que root ?

... Y en a pour un certain temps.

Pour aller plus vite, **forçons le redémarrage** de notre machine.



 Pas de doute, le système est cassé ! 😊



Pour restaurer le système, nous utilisons l'instantané réalisé plus haut.

Nous **affichons l'interface de gestion des instantanés**.

Avec le bouton approprié.

Puis nous **sélectionnons l'instantané** existant.



Et **clic sur le bouton** Exécuter le cliché sélectionné



Une fois fait,

Nous **Affichons la console graphique**.

Et...

Yes ! Le système a été restauré. 😊

Hyper puissant, non ?

## Suspendre/Éteindre/Redémarrer

Depuis les fenêtres Gestionnaire de machines virtuelles ou la fenêtre Console/Détails,



Il nous est facile de trouver les commandes qui permettent de **suspendre/reprendre, arrêter** ou **redémarrer** une machine virtuelle.

Ces commandes sont disponibles :

- **Par des boutons** dans les deux fenêtres.



- **Depuis le menu** machine virtuelle de la fenêtre Console/Détails d'une machine virtuelle.
- **Depuis le menu contextuel** d'une machine virtuelle (dans la fenêtre Gestionnaire de machines virtuelles).



Si le signal Éteindre ou Redémarrer ne suffit pas,  
Et que la machine virtuelle ne s'ordonne pas.  
Sont disponibles les commandes :

- Forcer l'extinction
- Forcer la réinitialisation

## Sauvegarder

La commande Sauvegarder permet d'**arrêter la machine virtuelle en sauvegardant son système** préalablement.

... **Au prochain démarrage**, elle sera **redémarrée dans le même état** que lorsque qu'elle a été arrêtée.



Cette commande est accessible :

- **Par le bouton déroulant** Arrêt.
- **Depuis le menu contextuel** d'une machine virtuelle (dans la fenêtre Gestionnaire de machine virtuelle).
- **Depuis le menu** Machine virtuelle > Éteindre > Sauvegarder de la fenêtre Console/Détails.

## Supprimer



Pour supprimer une machine virtuelle, elle doit être arrêtée.

Une machine virtuelle est supprimée,

**Depuis la fenêtre** Gestionnaire de machine virtuelle,

Soit en utilisant :

- Le **menu** Édition / Supprimer.
- Ou depuis le **menu contextuel** d'une machine virtuelle.



Virt-manager nous demande si nous souhaitons supprimer le ou les périphérique(s) de stockage associé(s) à la machine virtuelle.

Ici, nous avons aucune raison de le conserver.

Alors nous autorisons la suppression de l'image tutoz.qcow2.

... Mais nous aurions aussi pu la conserver.

Et ainsi recréer un jour une machine virtuelle qui utiliserait cette image comme disque virtuel.

**Clic sur le bouton** Supprimer.



Puis **clic sur le bouton** OUI pour confirmer.



Voilà la machine virtuelle tutoz a été supprimé.

## Détails d'une VM

La fenêtre Console/Détails permet d'afficher **les détails (la configuration) d'une machine virtuelle**.

Ouvrons la fenêtre Console/détails,

Puis nous affichons la page Détails avec son bouton.

(Voir plus haut [Ouvrir une VM.](#))



Nous montons ci-dessous, quelques usages de configurations

## Options de démarrage



Les options de démarrage permettent

- D'indiquer à l'hyperviseur de démarrer la machine virtuelle au démarrage de l'hôte.
- D'activer/désactiver le démarrage de la machine virtuelle depuis un périphérique. Et d'en choisir l'ordre des tentatives (le premier périphérique activé, qui démarre, suffit).

---

## Émuler le CDROM



Pour émuler le CDROM par une image iso,  
Par exemple pour utiliser un système live.  
Ou simplement pour rendre son contenu accessible depuis le cdrom de la machine virtuelle.

Pour l'indiquer à l'hyperviseur, il suffit de **cliquer sur le bouton** Naviguer.  
Et de **sélectionner l'image iso**, dans le pool approprié.  
Puis **clique sur le bouton** Apply.

---

## Modifier la RAM



Si la quantité de mémoire attribuée ne suffit pas,  
Il est possible d'en modifier la quantité.

---

## Ajouter un disque



Pour ajouter un matériel,  
**Clique sur le bouton** Ajouter un matériel

La fenêtre Ajouter un matériel virtuelle est affichée.



La liste des matériels qui peuvent être ajoutés est disponible dans le panneau de gauche.  
Le premier de la liste est le stockage.

... Cela tombe bien, c'est ce qu'ici nous souhaitons : **Ajouter un disque dur virtuel**.

Ici, **nous en un créons un nouveau**,  
Dans le pool Default.

Mais nous aurions pu aussi le créer ailleurs, dans un autre pool.  
Ou utiliser une image disque déjà existante.

Nous pouvons aussi **choisir un autre type de périphérique**.



Ou **choisir le type de bus** de communication entre le disque et les composants de la carte mère virtuelle.

Utiliser le type VirtIO, permet l'usage (tant que possible) de la virtualisation matériel, via KVM.  
Cela à pour effet d'accélérer les performances de communication entre les composants virtuels.

D'autre options sont disponibles comme en faire un disque en lecteur seule.

Une fois nos choix établis, pour terminer l'ajout : **Clic sur le bouton Terminer**.



Puis le nouveau disque est connecté à la machine virtuelle.



Dans l'onglet Stockage, nous retrouvons sont image dans le pool Default.

La machine virtuelle démarrée. Connecté en tant que root, nous pouvons lister les disques (virtuels) que le système dispose avec la commande **fdisk -l**.



Nous pouvons maintenant tester fdisk ou gparted, lvm, mkfs, etc, sans soucis.

⚠ A condition de toujours veiller que les commandes soient exécutées dans la console graphique de la machine virtuelle.

Et, non pas, dans un terminal fenêtre de l'hôte.

## Conclusions

**Virt-manager** est un logiciel, qui par son interface graphique, rend abordable la gestion de machine virtuelle.

Sa simplicité d'usage est son atout,

Il offre les fonctionnalités de base pour manipuler les machines virtuelles à travers quelques clics.

Haha, il est plus ludique que la ligne de commande.

... Que nous serons tout de même obligés d'utiliser pour un usage plus avisé.

😊 Ça à été un plaisir de jouer avec vous.

By.

1)

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

2)

**Une machine virtuelle** est une machine dont les composants sont virtualisés et/ou émulés par un système que l'on nome **hôte**. Les machines virtuelles hébergés par un hôte sont dites **invitées**. Les composants virtualisés s'appuient sur le partage du matériel physique de l'hôte. Les composants émulés, utilisent les ressources de l'hôte, pour exécuter un logiciel qui reproduit le comportement d'un matériel. Le matériel virtuel (qu'il soit virtualisé ou émulé), est compris par celui qui l'utilise comme un matériel physique. En cloisonnant tout le matériel virtuel nécessaire au fonctionnement de nos ordinateurs (Processeur, RAM, disque, interface réseau, carte graphique et audio, clavier, souris, etc), nous obtenons une machine virtuelle sur laquelle peut être exécuté/installé un système d'exploitation. [\(wikipedia\)\(fr\) Machine virtuelle](#) - [\(redhat\)\(fr\) Qu'est qu'une machine virtuelle ?](#) - [\(ibm\)\(fr\) Qu'est-ce qu'une machine virtuelle \(VM\) ?](#)

3)

**Un hyperviseur** est un gestionnaire de machine virtuelle. Il permet à un système hôte de partager ses ressources pour l'exécution des machines virtuelles qu'il a démarré. [\(wikipedia\)\(fr\) Hyperviseur](#) - [\(ibm\)\(fr\) Qu'est-ce qu'un hyperviseur ?](#)

4)

**KVM** est l'hyperviseur intégré au noyau Linux. Il utilise la virtualisation matérielle des microprocesseurs. [\(wikipedia\)\(fr\) Kernel-based Virtual Machine](#) - [\(redhat\)\(fr\) Qu'est que KVM ?](#) - [\(linux-kvm\)\(en\)](#)

5)

**Un émulateur** est un logiciel qui simule un matériel. Il fournit aux logiciels qui s'adressent à lui, les fonctionnalités de l'appareil qu'il émule. [\(wikipedia\)\(fr\) Émulation](#)

6)

**QEMU** est un hyperviseur utilisant l'émulation. Il peut émuler complétement le matériel d'une machine virtuelle (processeur compris). Pouvant tout émuler (aux prix de la performance), il est capable d'émuler différentes architectures matériels. [\(wikipedia\)\(fr\) QEMU](#)

7)

**Libvirt** est une librairie logiciel qui permet de manager les machines virtuelles de plusieurs hyperviseurs, émulateurs et isolateurs. [\(wikipedia\)\(fr\) libvirt](#)

8)

**Xen** est un hyperviseur utilisant la para-virtualisation (le système invité sait qu'il fonctionne sur un hôte). Le noyau du système hôte et celui du système invité sont modifiés pour qu'ils puissent communiquer entre eux. Ces modifications sont intégrées aux noyaux Linux modernes. Ainsi un noyau Linux peut être utilisé, aussi bien par le système hôte, que par les systèmes invités.

[\(wikipedia\)\(fr\) Xen](#)

9)

**Bhyve** est l'hyperviseur intégré au noyau BSD. Il utilise la virtualisation matérielle des microprocesseurs. [\(wikipedia\)\(en\) Bhyve](#)

10)

**Un isolateur** est un système de virtualisation, qui permet d'exécuter un système invité, dans le noyau de l'hôte, de façon cloisonnée (nous parlons de container). Directement exécuté par l'hôte, le système invité doit-être compatible avec le système hôte. Ainsi, un système hôte GNU/Linux ne pourra faire fonctionner que des systèmes invités GNU/Linux. Cette restriction offre des

performances d'exécution des systèmes invités supérieurs aux systèmes de virtualisations émulés. Une autre restriction, est que le système invité ne peut bénéficier, du matériel que l'hôte ne dispose pas ; Contrairement aux systèmes de virtualisation qui émulent le matériel (aux prix des performances). ([wikipedia](#))(fr) [Virtualisation : Isolateur](#)

11)

**OpenVZ** est un isolateur open source (provenant de la solution d'isolation propriétaire Virtuozzo). Le noyau du système hôte doit être modifier, pour réaliser la conteneurisation des systèmes invités.

([wikipedia](#))(en) [OpenVZ](#)

12)

**LXC** (Linux Containers en anglais) est l'isolateur intégré au noyau Linux. Pour conteneuriser les systèmes invités, deux fonctionnalités du noyau sont employées : Les cgroups pour contrôler les ressources attribuées. Et les espaces de nom (namespaces) pour isoler les processus. ([wikipedia](#))(fr)

[LXC](#)

13)

**SSH** (Secure SHell en anglais) est un protocole réseau sécurisé (crypté) et une suite logiciel de type client/serveur (OpenSSH sous GNU/Linux). Permettant de se connecter à une machine distante. Sous un système GNU/Linux, **ssh** est la commande (le client) d'OpenSSH, pour se connecter à une machine distante, qui dispose de l'implémentation serveur d'OpenSSH. ([wikipedia](#))(fr) [Secure Shell](#) -

([wikipedia](#))(fr) [OpenSSH](#)

14)

**VNC** est une suite logiciel historique, de type client/serveur, qui permet l'usage d'une console graphique connectée à système distant. Il a alors été développé un protocole réseau, qui établit, la communication entre la console et la machine distante, par un flux graphique. ([wikipedia](#))(fr)

[Virtual\\_Network\\_Computing](#)

15)

**SPICE** est le projet qui fournit une console graphique permettant d'accéder à une machine virtuelle. La communication entre la console et la machine virtuelle est réalisée par un flux vidéo.

([wikipedia](#))(fr) [SPICE \(protocol\)](#) - ([spice-space](#))(en)

16)

**Un fichier iso** nommé aussi **image iso** est une copie intégrale d'un périphérique de stockage (disque optique, disquette, disque dur, clé USB, partition, volume) dans un seul (ou plusieurs) fichier(s). La notion d'image est adaptée car il s'agit d'une photo (un instantané) du périphérique tel qu'il est à un instant. Permettant ainsi de diffuser ou restaurer un périphérique de stockage.

([wikipedia](#))(fr) [Image disque](#)

17)

**DHCP** est un protocole réseau qui permet d'attribuer dynamiquement des adresse IP aux hôtes d'un réseau. ([wikipedia](#))(fr) [Dynamic Host Configuration Protocol](#)

18)

**Qcow** est le format d'image disque qu'utilise QEMU pour émuler un disque dur. ([wikipedia](#))(fr) [Qcow2](#)

19)

**Sid** est réputé chez Debian pour casser ses jouets.

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/doc:autres:vm:virt-manager>



Last update: **04/05/2024 22:47**