



Red Hat Enterprise Linux 5

Virtualization

A reference guide for virsch, xm, vmm and xend.

Édition 2.0

Red Hat Enterprise Linux 5 Virtualization

A reference guide for virsch, xm, vmm and xend.

Édition 2.0

Landmann

rlandmann@redhat.com

Notice légale

Copyright © 2007 Red Hat Inc..

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](https://creativecommons.org/licenses/by-sa/3.0/). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Résumé

Ce guide contient des informations sur la configuration, la création et le contrôle de systèmes d'exploitation invités sur Red Hat Enterprise Linux 5, en utilisant virsh, xm, vmm et xend. Nous vous invitons vivement à nous écrire si vous trouvez des fautes de frappe dans le Guide de Virtualisation ou si vous souhaitez nous faire part de vos suggestions pour améliorer ce manuel. Pour ce faire, veuillez nous soumettre un rapport par l'entremise de Bugzilla (à l'adresse suivante : <http://bugzilla.redhat.com/bugzilla/>) après le produit Red Hat Enterprise Linux et le composant Virtualization_Guide.

Table des matières

CHAPITRE 1. ARCHITECTURE SYSTÈME DE LA VIRTUALISATION RED HAT	4
CHAPITRE 2. SUPPORT DU SYSTÈME D'EXPLOITATION	6
CHAPITRE 3. SUPPORT MATÉRIEL	8
CHAPITRE 4. PRÉREQUIS DU SYSTÈME DE VIRTUALISATION RED HAT	10
CHAPITRE 5. DÉMARRAGE DU SYSTÈME	11
CHAPITRE 6. CONFIGURATION DE GRUB	12
CHAPITRE 7. DÉMARRAGE D'UN DOMAINE INVITÉ	14
CHAPITRE 8. DÉMARRAGE/ARRÊT D'UN DOMAINE LORS DE L'AMORÇAGE DU SYSTÈME	15
CHAPITRE 9. FICHIERS DE CONFIGURATION	16
CHAPITRE 10. GESTION DES CPU	17
CHAPITRE 11. MIGRATION D'UN DOMAINE	18
CHAPITRE 12. CONFIGURATION POUR UNE UTILISATION RÉSEAU	19
CHAPITRE 13. SÉCURISATION DE DOMAIN0	20
CHAPITRE 14. STOCKAGE	21
CHAPITRE 15. GESTION DE MACHINES VIRTUELLES AU MOYEN DE VIRSH	22
15.1. CONNEXION À UN HYPERVISEUR	22
15.2. CRÉATION D'UNE MACHINE VIRTUELLE	22
15.3. CONFIGURATION D'UN DUMP XML	22
15.4. SUSPENSION D'UNE MACHINE VIRTUELLE	22
15.5. REPRISE D'UNE MACHINE VIRTUELLE	22
15.6. ENREGISTREMENT D'UNE MACHINE VIRTUELLE	23
15.7. RESTAURATION D'UNE MACHINE VIRTUELLE	23
15.8. ARRÊT D'UNE MACHINE VIRTUELLE	23
15.9. REDÉMARRAGE D'UNE MACHINE VIRTUELLE	23
15.10. ARRÊT BRUTAL D'UN DOMAINE	23
15.11. CONVERSION D'UN NOM DE DOMAINE EN UN ID DE DOMAINE	24
15.12. CONVERSION D'UN ID DE DOMAINE EN UN NOM DE DOMAINE	24
15.13. CONVERSION D'UN NOM DE DOMAINE EN UN UUID	24
15.14. AFFICHAGE DES INFORMATIONS D'UNE MACHINE VIRTUELLE	24
15.15. AFFICHAGE DES INFORMATIONS D'UN NOEUD	24
15.16. AFFICHAGE DES MACHINES VIRTUELLES	25
15.17. AFFICHAGE DES INFORMATIONS D'UN CPU VIRTUEL	25
15.18. CONFIGURATION DE L'AFFINITÉ DES CPU VIRTUELS	25
15.19. CONFIGURATION DU NOMBRE DE CPU VIRTUELS	25
15.20. CONFIGURATION DE L'ALLOCATION DE LA MÉMOIRE	26
15.21. CONFIGURATION DE LA MÉMOIRE MAXIMUM	26
15.22. MANAGING VIRTUAL NETWORKS	26
CHAPITRE 16. GESTION DE MACHINES VIRTUELLES AU MOYEN DE XEND	28
CHAPITRE 17. GESTION DES MACHINES VIRTUELLES AVEC LE GESTIONNAIRE DE MACHINES VIRTUELLES	30

17.1. ARCHITECTURE DU GESTIONNAIRE DE MACHINES VIRTUELLES	30
17.2. LA FENÊTRE OUVRIR UNE CONNEXION	30
17.3. LA FENÊTRE GESTIONNAIRE DE MACHINES VIRTUELLES	30
17.4. LA FENÊTRE DÉTAILS D'UNE MACHINE VIRTUELLE	31
17.5. LA CONSOLE GRAPHIQUE D'UNE MACHINE VIRTUELLE	32
17.6. STARTING THE VIRTUAL MACHINE MANAGER	33
17.7. CRÉATION D'UNE NOUVELLE MACHINE VIRTUELLE	33
17.8. RESTAURATION D'UNE MACHINE ENREGISTRÉE	43
17.9. DISPLAYING VIRTUAL MACHINE DETAILS	45
17.10. CONFIGURING STATUS MONITORING	48
17.11. AFFICHAGE DE L'ID DU DOMAINE	49
17.12. AFFICHAGE DE L'ÉTAT D'UNE MACHINE VIRTUELLE	50
17.13. AFFICHAGE DES CPU VIRTUELS	51
17.14. AFFICHE DE L'UTILISATION CPU	52
17.15. AFFICHAGE DE L'UTILISATION MÉMOIRE	53
17.16. MANAGING A VIRTUAL NETWORK	54
17.17. CREATING A VIRTUAL NETWORK	55
CHAPITRE 18. RÉOLUTION DES PANNES DE LA VIRTUALISATION RED HAT	64
18.1. VUE D'ENSEMBLE SUR LES FICHIERS JOURNAUX ET EMBLEMES	64
18.2. DESCRIPTIONS DES FICHIERS JOURNAUX	64
18.3. EMBLEMES IMPORTANTS DANS LES RÉPERTOIRES	65
18.4. OUTILS DE RÉOLUTION DES PANNES	65
18.5. RÉOLUTION DES PANNES AVEC LES FICHIERS JOURNAUX	67
18.6. RÉOLUTION DES PANNES AVEC LA CONSOLE SÉRIE	67
18.7. ACCÈS À LA CONSOLE DE L'INVITÉ PARAVIRTUALISÉ	68
18.8. ACCÈS À LA CONSOLE DE L'INVITÉ PLEINEMENT VIRTUALISÉ	68
18.9. IMPLEMENTATION DE LA PERSISTANCE LUN	68
18.10. CONSIDÉRATIONS AVEC SELINUX	70
18.11. ACCÈS AUX DONNÉES SUR L'IMAGE DISQUE DE L'INVITÉ	70
18.12. SITUATIONS FRÉQUENTES DANS LA RÉOLUTION DES PANNES	71
18.13. ERREURS DE PÉRIPHÉRIQUES LOOP	72
18.14. ERREURS DE CRÉATION D'INVITÉS	72
18.15. ERREURS DE LA CONSOLE SÉRIE	72
18.16. ERREURS DE PONT RÉSEAU	73
18.17. CONFIGURATIONS DES ORDINATEURS PORTABLES	74
18.18. STARTING DOMAINS AUTOMATICALLY DURING SYSTEM BOOT	76
18.19. MODIFICATION DE DOMAIN0	76
18.20. FICHIERS DE CONFIGURATION DES INVITÉS	77
18.21. CLONAGE DES FICHIERS DE CONFIGURATION DES INVITÉS	78
18.22. CREATION D'UN SCRIPT POUR GÉNÉRER DES ADRESSES MAC	78
18.23. CONFIGURATION DE LA MIGRATION EN DIRECT DE MACHINES VIRTUELLES	78
18.24. INTERPRÉTATION DES MESSAGES D'ERREUR	79
18.25. RESSOURCES DE RÉOLUTION DES PANNES EN LIGNE	82
CHAPITRE 19. RESSOURCES SUPPLÉMENTAIRES	83
19.1. SITES WEB UTILES	83
19.2. DOCUMENTATION INSTALLÉE	83
ANNEXE A. EXERCICE 1	84
ANNEXE B. EXERCICE 2	89
ANNEXE C. HISTORIQUE DES VERSIONS	91

CHAPITRE 1. ARCHITECTURE SYSTÈME DE LA VIRTUALISATION RED HAT

Un système de Virtualisation Red Hat fonctionnel est composé de plusieurs couches et dirigé par le composant privilégié de Virtualisation Red Hat. La virtualisation peut héberger plusieurs systèmes d'exploitation invités. Chaque système d'exploitation invité exécute son propre domaine, la Virtualisation Red Hat planifie les CPU virtuels au sein des machines virtuelles afin d'utiliser de façon optimale les CPU physiques disponibles. Chaque système d'exploitation invité s'occupe de ses propres applications. Les systèmes d'exploitation invités planifient chaque application en conséquence.

Vous pouvez déployer la Virtualisation Red Hat de deux manières différentes : **virtualisation complète** ou **paravirtualisation** (virtualisation partielle). La virtualisation complète fournit une abstraction totale du système physique sous-jacent et crée un nouveau système virtuel dans lequel les systèmes d'exploitation invités peuvent démarrer. Aucune modification du SE ou de l'application invité n'est nécessaire (le SE ou l'application invité n'est pas informé de l'environnement paravirtualisé et démarre normalement). La paravirtualisation requiert une modification utilisateur des systèmes d'exploitation invités qui sont exécutés sur les machines virtuelles (ces systèmes d'exploitation invités savent qu'ils sont exécutés sur une machine virtuelle) et qui fournissent des performances presque natives. Vous pouvez déployer la virtualisation partielle et complète à travers l'infrastructure de votre virtualisation.

Le premier domaine, appelé **domain0** (dom0), est créé automatiquement lorsque vous démarrez le système. Domain0 est l'invité privilégié et possède des capacités de gestion permettant de créer de nouveaux domaines et de gérer leurs périphériques virtuels. Domain0 s'occupe du matériel physique, tel que les contrôleurs de cartes réseau et de disques durs. Domain0 s'occupe également des tâches administratives telles que la suspension, la reprise ou la migration de domaines invités vers d'autres machines virtuelles.

The **hypervisor** (Red Hat's Virtual Machine Monitor) is a virtualization platform that allows multiple operating systems to run on a single host simultaneously within a full virtualization environment. A guest is an operating system (OS) that runs on a virtual machine in addition to the host or main OS.

With Red Hat Virtualization, each guests **memory** comes from a slice of the host's physical memory. For paravirtual guests, you can set both the initial memory and the maximum size of the virtual machine. You can add (or remove) physical memory to the virtual machine at runtime without exceeding the maximum size you specify. This process is called ballooning.

Vous pouvez configurer chaque invité avec un nombre de **cpu** virtuels (appelés vcpus). Le gestionnaire de machines virtuelles planifie les vcpus en fonction de la charge de travail sur les CPU physiques.

Vous pouvez allouer à un invité n'importe quel nombre de **disques virtuels**. L'invité les voit comme des disques durs ou (pour les invités pleinement virtuels) des lecteurs de CD-ROM. Chaque disque virtuel est servi à l'invité à partir d'un périphérique bloc ou d'un fichier régulier sur l'hôte. Le périphérique sur l'hôte contient l'image disque complète pour l'invité et inclut généralement des tables de partition, de multiples partitions et potentiellement des volumes physiques LVM.

Les **interfaces de mise en réseau virtuelle** démarrent sur l'invité. Les autres interfaces peuvent démarrer sur l'invité comme des cartes internet ethernet virtuelles (VNIC, de l'anglais Virtual Ethernet Internet Cards). Ces interfaces réseau sont configurées avec une adresse de contrôle d'accès au média (MAC, de l'anglais Media Access Control) virtuelle persistante. L'installation par défaut d'un nouvel invité installe la VNIC avec une adresse MAC sélectionnée au hasard à partir d'un groupement réservé de plus de 16 millions d'adresses, ainsi il y a vraiment peu de chances que deux invités reçoivent la même adresse MAC. Les sites complexes avec beaucoup d'invités peuvent allouer des adresses MAC manuellement afin de s'assurer qu'elles soient uniques sur le réseau.

Chaque invité a une **console texte** virtuelle qui se connecte à l'hôte. Vous pouvez rediriger les logins invités et les sorties de console vers la console texte.

Vous pouvez configurer les invités pour qu'ils utilisent une **console graphique** virtuelle qui correspond à la console vidéo normale sur l'hôte physique. Vous pouvez faire cela pour les invités pleinement virtuels et partiellement virtuels. Elle emploie les fonctionnalités de l'adaptateur graphique standard telles que la messagerie de démarrage, le démarrage graphique, de multiples terminaux virtuels et peut démarrer le système X Window. Vous pouvez également utiliser le clavier graphique pour configurer le clavier et la souris virtuels.

Guests can be identified in any of three **identities**: domain name (domain-name), identity (domain-id), or UUID. The domain-name is a text string that corresponds to a guest configuration file. The domain-name is used to launch the guests, and when the guest runs the same name is used to identify and control it. The domain-id is a unique, non-persistent number that gets assigned to an active domain and is used to identify and control it. The UUID is a persistent, unique identifier that is controlled from the guest's configuration file and ensures that the guest is identified over time by system management tools. It is visible to the guest when it runs. A new UUID is automatically assigned to each guest by the system tools when the guest first installs.

CHAPITRE 2. SUPPORT DU SYSTÈME D'EXPLOITATION

Red Hat Virtualization's paravirtualization mode allows you to utilize high performance virtualization on architectures that are potentially difficult to virtualize such as x86 based systems. To deploy paravirtualization across your operating system(s), you need access to the paravirtual guest kernels that are available from a respective Red Hat distro (for example, RHEL 4.0, RHEL 5.0, etc.). Whilst your operating system kernels must support Red Hat Virtualization, it is not necessary to modify user applications or libraries.

La Virtualisation Red Hat vous permet de démarrer un noyau invité non modifié si vous avez les processeurs Intel VT et AMD SVM. Vous n'avez pas à transférer votre système d'exploitation pour déployer cette architecture sur votre système Intel VT ou AMD SVM. La Virtualisation Red Hat supporte :

- La technologie Intel VT-x ou AMD-V Pacifica et Vanderpool pour la virtualisation partielle et complète.
- Intel VT-i pour ia64
- Les systèmes d'exploitation Linux et UNIX, y compris NetBSD, FreeBSD et Solaris.
- Microsoft Windows as an unmodified guest operating system with Intel Vanderpool or AMD's Pacifica technology.

Pour démarrer les invités de virtualisation complète sur des systèmes avec du matériel assisté par des machines virtuelles (HVM, de l'anglais Hardware-assisted Virtual Machine), Intel ou des plateformes AMD, vous devez vous assurer que votre CPU dispose des capacités requises.

Pour vérifier si vous avez les indicateurs CPU pour le support Intel, saisissez ce qui suit :

```
grep vmx /proc/cpuinfo
```

La sortie affiche :

```
flags      : fpu tsc msr pae mce cx8 apic mtrr mca cmov pat pse36 clflush dts
acpi mmx fxsr sse sse2 ss ht tm syscall nx lm constant_tsc pni monitor
ds_cpl vmx est tm2 cx16 xtpr lahf_lm
```

Si un indicateur vmx apparaît, vous disposez alors du support Intel.

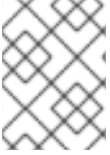
Pour vérifier si vous avez les indicateurs CPU pour le support AMD, saisissez ce qui suit :

```
grep svm /proc/cpuinfo
cat /proc/cpuinfo | grep svm
```

La sortie affiche :

```
flags      : fpu tsc msr pae mce cx8 apic mtrr mca cmov pat pse36 clflush dt
acpi mmx fxsr sse sse2 ss ht tm syscall nx mmtxt fxsr_opt rdtscp lm
3dnowext pni cx16 lahf_lm cmp_legacy svm cr8_legacy
```

Si un indicateur svm apparaît, vous disposez alors du support AMD.

**NOTE**

En plus de vérifier les indicateurs CPU, vous devez activer la virtualisation complète au sein de votre système BIOS.

CHAPITRE 3. SUPPORT MATÉRIEL

La Virtualisation Red Hat supporte des systèmes à processeurs multiples. Vous pouvez démarrer la Virtualisation Red Hat sur des systèmes architecturés x86 avec des processeurs de la classe P6 (ou plus récents) tels que :

- Celeron
- Pentium II
- Pentium III
- Pentium IV
- Xeon
- AMD Athlon
- AMD Duron

Avec la Virtualisation Red Hat, les hôtes 32 bit démarrent uniquement des invités paravirtuels 32 bit. Les hôtes 64 bit démarrent seulement des invités paravirtuels 64 bit. Un hôte 64 bit de virtualisation complète, démarre les invités 32 bit, 32 bit PAE, 64 bit. Un hôte 32 bit de virtualisation complète, démarre les invités PAE et non PAE pleinement virtualisés.

Le noyau de Virtualisation Red Hat Enterprise Linux ne supporte pas plus de 32 Go de mémoire pour les systèmes x86_64. Si vous devez démarrer le noyau de virtualisation sur des systèmes avec plus de 32 Go de mémoire physique, vous devez ajouter `mem=32G` à la ligne de commande du noyau. Cet exemple illustre comment activer le paramètre approprié dans le fichier `grub.conf` :

```
title Red Hat Enterprise Linux Server (2.6.18-4.elxen)
root (hd0, 0)
kernel /xen.gz-2.6.18-4-el5 mem=32G
module /vmlinuz -2.6.18-4.el5xen ro root=LABEL=/
module /initrd-2.6.18-4.el5xen.img
```

PAE (de l'anglais Physical Address Extension) est une technologie qui augmente la quantité de mémoire virtuelle ou physique disponible aux utilisateurs d'applications. La Virtualisation Red Hat requiert que PAE soit actif sur vos systèmes. L'architecture 32 bit de la Virtualisation Red Hat avec PAE supporte jusqu'à 16 Go de mémoire physique. Nous vous recommandons d'avoir au moins 256 méga-octets de mémoire RAM pour chaque invité démarré sur le système. La Virtualisation Red Hat active les machines x86/64 pour adresser jusqu'à 64 Go de mémoire physique. Les noyaux de Virtualisation Red Hat ne démarreront pas sur un système non PAE. Pour déterminer si votre système supporte PAE, saisissez les commandes suivantes :

```
grep pae /proc/cpuinfo
```

La sortie suivante affiche :

```
flags : fpu tsc msr pae mce cx8 apic mtrr mca cmov pat pse36 mmx fxsr sse
syscall mmtxt 3dnowext 3dnow up ts
```

Si votre sortie correspond (ou ressemble) à celle ci-dessus, cela signifie que votre CPU supporte PAE.
Si l'invite de commande n'affiche rien, cela signifie que votre CPU ne supporte pas PAE.

CHAPITRE 4. PRÉREQUIS DU SYSTÈME DE VIRTUALISATION RED HAT

Les éléments listés ci-dessous sont requis par le système de Virtualisation Red Hat :

- Une distribution Red Hat Linux RHEL 5 fonctionnelle
- Un chargeur de démarrage GRUB fonctionnel
- Un accès super-utilisateur
- Un processeur de classe P6 (ou plus récent)
- L'élément bridge-utils de Linux
- Les systèmes hotplug de Linux
- L'installation des bibliothèques de développement zlib
- L'environnement d'exécution Python 2.2
- initscripts

Les dépendances sont configurées automatiquement durant le processus d'installation.



NOTE

If your system CPU architecture is ia64, you need to manually install the **xen-ia64-guest-firmware** package to run a fully virtualized guest. This package is provided in the Supplementary CD and is not installed by default.

CHAPITRE 5. DÉMARRAGE DU SYSTÈME

Après l'installation des composants de Virtualisation Red Hat, vous devez redémarrer le système. Lorsque le démarrage est terminé, vous devez vous connecter à votre système de la façon habituelle. Ensuite, avant de démarrer la Virtualisation Red Hat, vous devez vous connecter en tant que super-utilisateur. Le démon de contrôle xend devrait déjà être initialisé avec initscripts, mais pour démarrer xend manuellement, saisissez :

```
service xend start
```

Vous pouvez également utiliser **chkconfig xend** lors de l'installation afin d'activer **xend** au démarrage.

Le démon de contrôle xend effectue des fonctions de gestion système en relation aux machines virtuelles. Ce démon contrôle les ressources virtualisées, et xend doit être en cours d'exécution pour interagir avec les machines virtuelles. Avant de démarrer xend, vous devez spécifier les paramètres d'opération en modifiant le fichier de configuration xend xend-config.sxp qui se situe dans le répertoire etc/xen.

CHAPITRE 6. CONFIGURATION DE GRUB

GNU Grand Unified Boot Loader (or GRUB) is a program which enables the user to select which installed operating system or kernel to load at system boot time. It also allows the user to pass arguments to the kernel. The GRUB configuration file (located in `/boot/grub/grub.conf`) is used to create a list of operating systems to boot in GRUB's menu interface. When you install the kernel-xen RPM, a post script adds kernel-xen entries to the GRUB configuration file. You can edit the `grub.conf` file and enable the following GRUB parameter:

```
title Red Hat Enterprise Linux Server (2.6.18-3.el5xen)
root    (hd0; 0)
kernel  /xen.gz.-2.6.18-3.el5
module  /vmlinuz-2.6..18-3.el5xen ro root=/dev/VolGroup00/LogVol100 rhgb
quiet
module  /initrd-2.6.18-3. el5xenxen.img
```

Si vous paramétrez vos entrées grub Linux pour qu'elles reflètent cet exemple, le chargeur de démarrage charge l'hyperviseur, l'image `initrd` et le noyau Linux. Étant donné que l'entrée du noyau est la première des entrées, le noyau se charge en mémoire en premier. Le chargeur de démarrage envoie (et reçoit) des arguments en ligne de commande vers et en provenance de l'hyperviseur et du noyau Linux. Cet exemple d'entrée illustre comment restreindre la mémoire du noyau Linux de Domain0 à 800 Mo :

```
title Red Hat Enterprise Linux Server (2.6.18-3.el5xen)
root    (hd0; 0)
kernel  /xen.gz.-2.6.18-3.el5 dom0_mem=800M
module  /vmlinuz-2.6..18-3.el5xen ro root=/dev/VolGroup00/LogVol100 rhgb
quiet
module  /initrd-2.6.18-3. el5xenxen.img
```

Vous pouvez utiliser les paramètres GRUB pour configurer l'hyperviseur de Virtualisation :

```
mem
```

Limite la quantité de mémoire qui est disponible au domain0.

```
com1=115200, 8n1
```

Active le premier port série dans le système de façon à agir comme une console série (com2 est assigné au prochain port, et ainsi de suite...).

```
dom0_mem
```

Limite la quantité de mémoire qui est disponible au domain0.

```
dom0_max_vcpus
```

Limite la quantité de CPU visible par domain0.

acpi

Alterne l'hyperviseur ACPI avec l'hyperviseur et domain0. Les options du paramètre ACPI incluent :

```
/* **** Linux config options: propagated to domain0 ****/  
/* "acpi=off":      Disables both ACPI table parsing and interpreter.  
*/  
/* "acpi=force":    Overrides the disable blacklist.  
*/  
/* "acpi=strict":   Disables out-of-spec workarounds.  
*/  
/* "acpi=ht":       Limits ACPI from boot-time to enable HT.  
*/  
/* "acpi=noirq":    Disables ACPI interrupt routing.  
*/
```

noacpi

Désactive ACPI pour la distribution d'interruptions.

CHAPITRE 7. DÉMARRAGE D'UN DOMAINE INVITÉ

Vous pouvez démarrer des domaines invités en utilisant l'application **xm**. Vous pouvez également utiliser **virsh** et le gestionnaire de machines virtuelles pour démarrer les invités. L'installation d'un hôte invité en premier est une condition préalable pour le démarrage d'un domaine invité. Cet exemple utilise la sous-commande de création **xm** :

```
# xm create -c guestdomain1
```

Le **guestdomain1** est le fichier de configuration pour le domaine que vous démarrez. L'option **-c** permet de se connecter à la console après le démarrage.

CHAPITRE 8. DÉMARRAGE/ARRÊT D'UN DOMAINE LORS DE L'AMORÇAGE DU SYSTÈME

Vous pouvez démarrer ou arrêter des domaines en cours d'exécution à n'importe quel moment. Domain0 attend l'arrêt de tous les domaines en cours d'exécution avant de redémarrer. Vous devez placer les fichiers de configuration des domaines que vous désirez éteindre dans le répertoire `/etc/xen/`. Tous les domaines que vous voulez lancer au démarrage doivent être référencés dans `/etc/xen/auto`.

```
chkconfig xendomains on
```

La commande **chkconfig xendomains on** ne démarre pas automatiquement les domaines ; à la place, elle démarre les domaines au prochain démarrage.

```
chkconfig xendomains off
```

Arrête tous les domaines de Virtualisation Red Hat en cours d'exécution. La commande **chkconfig xendomains off** arrête les domaines au prochain démarrage.

CHAPITRE 9. FICHIERS DE CONFIGURATION

Red Hat Virtualization configuration files contain the following standard variables. Configuration items within these files must be enclosed in quotes ("). These configuration files reside in the `/etc/xen` directory.

Tableau 9.1. Fichiers de configuration de la Virtualisation Red Hat

Élément	Description
pae	Spécifie les données de configuration de l'extension de l'adresse physique.
apic	Spécifie les données de configuration avancées du contrôleur d'interruptions programmable.
mémoire	Spécifie la taille de la mémoire en méga-octets.
vcpus	Spécifie le nombre de CPU virtuels.
console	Spécifie les numéros de ports où les consoles de domaine seront exportées.
nic	Spécifie le nombre d'interfaces réseau virtuelles.
vif	Lists the randomly-assigned MAC addresses and bridges assigned to use for the domain's network addresses.
disque	Liste les périphériques blocs à exporter vers le domaine et exporte les périphériques physiques vers le domaine avec un accès en lecture-seule.
dhcp	Active la mise en réseau en utilisant DHCP.
netmask	Spécifie les masques réseau des adresses IP configurées.
gateway	Spécifie les passerelles des adresses IP configurées.
acpi	Spécifie les données de configuration de l'interface avancée de configuration et de gestion de l'énergie.

CHAPITRE 10. GESTION DES CPU

Red Hat Virtualization allows a domain's virtual CPUs to associate with one or more host CPUs. This can be used to allocate real resources among one or more guests. This approach allows Red Hat Virtualization to make optimal use of processor resources when employing dual-core, hyperthreading, or other advanced CPU technologies. If you are running I/O intensive tasks, it's typically better to dedicate either a hyperthread or entire core to run domain0. The Red Hat Virtualization credit scheduler automatically rebalances virtual cpus between physical ones, to maximize system use. The Red Hat Virtualization system allows the credit scheduler to move CPUs around as necessary, as long as the virtual CPU is pinned to a physical CPU.

CHAPITRE 11. MIGRATION D'UN DOMAINE

Une migration est le transfert d'un domaine virtuel en cours d'exécution d'un hôte physique vers un autre. La Virtualisation Red Hat supporte deux variétés de migration — déconnectée et en direct. Les migrations déconnectées déplacent une machine virtuelle d'un hôte à un autre en la mettant en pause, en transférant sa mémoire et ensuite, en la reprenant sur la destination hôte. Les migrations en direct font la même chose mais n'affectent pas directement le domaine. Lors d'une migration en direct, le domaine continue son activité habituelle et l'utilisateur ne s'aperçoit de rien. Pour initialiser une migration en direct, les hôtes doivent démarrer la virtualisation Red Hat et le démon xend. Les hôtes de destination doivent avoir suffisamment de ressources (par exemple de capacité mémoire) pour satisfaire la bande passante après la migration. Les machines source et de destination doivent avoir les mêmes extensions d'architecture et de virtualisation (par exemple i386-VT, x86-64-VT, x86-64-SVM, etc.) et doivent être sur le même sous-réseau L2.

When a domain migrates its MAC and IP addresses move with it. Only virtual machines with the same layer-2 network and subnets will successfully migrate. If the destination node is on a different subnet, the administrator must manually configure a suitable EtherIP or IP tunnel in the remote node of domain0. The xend daemon stops the domain and copies the job over to the new node and restarts it. The Red Hat Virtualization RPM does not enable migration from any other host except the localhost (see the `/etc/xend-config.sxp` file for information). To allow the migration target to accept incoming migration requests from remote hosts, you must modify the target's `xen-relocation-hosts-allow` parameter. Be sure to carefully restrict which hosts are allowed to migrate, since there is no authentication.

Étant donné que ces domaines ont de grandes allocations de fichiers, ce processus peut prendre du temps. Si vous migrez un domaine avec des connexions réseau ouvertes, elle seront préservées sur l'hôte de destination et les connexions SSH devraient toujours fonctionner. Les règles iptables de la Virtualisation Red Hat par défaut n'autoriseront pas les connexions de migration entrantes. Pour les autoriser, vous devez créer des règles iptables explicites.

Vous pouvez utiliser la commande de migration `xm` pour effectuer une migration déconnectée :

```
xm migrate domain-id [destination domain]
```

Vous pouvez utiliser la commande de migration `xm` pour effectuer une migration en direct :

```
xm migrate domain-id -l [destination domain]
```

You may need to reconnect to the domain's console on the new machine. You can use the `xm console` command to reconnect.

CHAPITRE 12. CONFIGURATION POUR UNE UTILISATION RÉSEAU

L'intégration de la virtualisation Red Hat dans l'architecture de votre réseau est un processus compliqué et selon votre infrastructure, vous pourriez avoir besoin d'une configuration personnalisée pour déployer de multiples interfaces ethernet et installer le pontage.

Each domain network interface is connected to a virtual network interface in **dom0** by a point to point link. These devices are **vif <domid>** and **<vifid>**. **vif1.0** for the first interface in domain 1; **vif3.1** for the second interface in domain 3.

Domain0 traite le trafic sur ces interfaces virtuelles en utilisant des conventions Linux standards pour le pontage, le routage, le contrôle de flux (rate limiting), etc. Le démon **xend** emploie deux scripts shell pour effectuer la configuration initiale de votre réseau et des interfaces virtuelles. Ces scripts configurent un pont unique pour toutes les interfaces virtuelles. Vous pouvez configurer des options de pontage et routage supplémentaires en personnalisant ces scripts.

Red Hat Virtualization's virtual networking is controlled by the two shell scripts, **network-bridge** and **vif-bridge**. **xend** calls these scripts when certain events occur. Arguments can be passed to the scripts to provide additional contextual information. These scripts are located in the **/etc/xen/scripts** directory. You can change script properties by modifying the **xend-config.sxp** configuration file located in the **/etc/xen** directory.

network-bridge — Lorsque **xend** est démarré ou stoppé, le script initialise ou arrête l'interface virtuelle. Ensuite, l'initialisation de la configuration crée le pont **xen-br0** et déplace **eth0** sur ce pont en modifiant le routage en conséquence. Finalement, lorsque **xend** quitte, il supprime le pont, **eth0** et par conséquent restaure la configuration d'origine de la l'adresse IP et du routage.

vif-bridge est un script, invoqué pour toutes les interfaces virtuelles sur le domaine. Il configure les règles du pare-feu et peut ajouter **vif** au pont approprié.

Afin que la Virtualisation Red Hat fonctionne sur votre réseau, vous pouvez utiliser d'autres scripts, tels que **network-route**, **network-nat**, **vif-route**, et **vif-nat** qui vous aideront à la configuration. Ces scripts peuvent aussi être remplacés par des variantes personnalisées.

CHAPITRE 13. SÉCURISATION DE DOMAIN0

When deploying Red Hat Virtualization on your corporate infrastructure, you must ensure that domain0 cannot be compromised. Domain0 is the privileged domain that handles system management. If domain0 is insecure, all other domains in the system are vulnerable. There are several ways to implement security you should know about when integrating Red Hat Virtualization into your systems. Together with other people in your organization, you should create a 'deployment plan' that contains the operating specifications and services that will run on Red Hat Virtualization, and what is needed to support these services. Here are some security issues to consider when putting together a deployment plan:

- Démarrez le moins de services possible. Vous ne devez pas inclure trop de tâches et services dans domain0. Moins il y a d'éléments démarrés sur domain0, plus le niveau de sécurité est élevé.
- Activez SeLINUX pour vous aider à sécuriser domain0.
- Utilisez un pare-feu pour contrôler le trafic de domain0. Vous pouvez installer un pare-feu avec des règles d'exclusion par défaut qui vont aideront à sécuriser les attaques sur domain0. Il est également important de limiter les services exposés au réseau.
- N'autorisez pas l'accès à domain0 aux utilisateurs normaux. Si vous autorisez l'accès à domain0 aux utilisateurs normaux, vous prenez le risque de rendre domain0 vulnérable. Rappelez-vous que domain0 est privilégié et que l'autorisation de comptes sans privilège peut compromettre le niveau de sécurité.

CHAPITRE 14. STOCKAGE

Il y a différentes façons de gérer le stockage des machines virtuelles. Vous pouvez exporter un périphérique bloc physique domain0 (un disque dur ou une partition) vers un domaine invité en tant que périphérique bloc virtuel (VBD de l'anglais Virtual Block Device). Vous pouvez aussi exporter directement à partir d'une image partitionnée telle qu'un fichier basé sur VBD. Par défaut, la Virtualisation Red Hat active LVM et blktap durant l'installation. Vous pouvez également employer des protocoles réseau standards tels que NFS, CLVM ou iSCSI pour fournir de l'espace aux machines virtuelles.

CHAPITRE 15. GESTION DE MACHINES VIRTUELLES AU MOYEN DE VIRSH

Vous pouvez utiliser l'application **virsh** pour gérer les machines virtuelles. Cet utilitaire est construit autour de l'API de gestion libvirt et opère comme une alternative à l'outil **xm** ou au gestionnaire graphique de machines virtuelles. Les utilisateurs sans privilège peuvent employer cet utilitaire pour des opérations en lecture-seule. Si vous avez l'intention de démarrer **xend/qemu**, vous devriez l'activer pour qu'il démarre en tant que service. Après avoir modifié le fichier de configuration respectif, redémarrez le système et **xend/qemu** démarrera comme un service. Vous pouvez utiliser **virsh** pour créer des scripts de travail vm. De la même façon qu'avec l'outil **xm**, vous pouvez démarrer **virsh** en ligne de commande.

15.1. CONNEXION À UN HYPERVISEUR

Vous pouvez utiliser **virsh** pour initier une session hyperviseur :

```
virsh connect <name>
```

Where **<name>** is the machine name of the hypervisor. If you want to initiate a read—only connection, append the above command with **—readonly**.

15.2. CRÉATION D'UNE MACHINE VIRTUELLE

Vous pouvez créer une nouvelle session de machine virtuelle à partir de la définition XML d'une machine. Si vous avez un invité pré-existant que vous avez créé auparavant avec l'outil **xm**, vous pouvez également créer une machine virtuelle pour ce dernier :

```
virsh create <path to XML configuration file>
```

15.3. CONFIGURATION D'UN DUMP XML

Vous pouvez utiliser **virsh** pour effectuer une sauvegarde (dump) de données pour une machine virtuelle existante.

```
virsh dumpxml [domain-id | domain-name | domain-uuid]
```

This command outputs the domain information (in XML) to **stdout** . If you save the data to a file, you can use the **create** option to recreate the virtual machine.

15.4. SUSPENSION D'UNE MACHINE VIRTUELLE

Vous pouvez utiliser **virsh** pour suspendre un domaine :

```
virsh suspend [domain-id | domain-name | domain-uuid]
```

When a domain is in a suspended state, it still consumes system RAM. There will also be no disk or network I/O when suspended. This operation is immediate and the virtual machine must be restarted with the **resume** option.

15.5. REPRISE D'UNE MACHINE VIRTUELLE

Vous pouvez utiliser **virsh** pour restaurer une machine virtuelle suspendue :

```
virsh resume [domain-id | domain-name | domain-uuid]
```

Cette opération est immédiate et les paramètres de la machine virtuelle sont préservés dans un cycle **suspend** et **resume**.

15.6. ENREGISTREMENT D'UNE MACHINE VIRTUELLE

Vous pouvez utiliser **virsh** pour enregistrer l'état courant d'une machine virtuelle dans un fichier :

```
virsh save [domain-name][domain-id | domain-uuid][filename]
```

This stops the virtual machine you specify and saves the data to a file, which may take some time given the amount of memory in use by your virtual machine. You can restore the state of the virtual machine with the **restore** option.

15.7. RESTAURATION D'UNE MACHINE VIRTUELLE

You can use **virsh** to restore a virtual machine that you previously saved with the **virsh save** option:

```
virsh restore [filename]
```

This restarts the saved virtual machine, which may take some time. The virtual machine's name and UUID are preserved but are allocated for a new id.

15.8. ARRÊT D'UNE MACHINE VIRTUELLE

Vous pouvez utiliser **virsh** pour arrêter une machine virtuelle :

```
virsh shutdown [domain-id | domain-name | domain-uuid]
```

Vous pouvez contrôler le comportement au démarrage de la machine virtuelle en modifiant le paramètre **on_shutdown** du fichier **xmdomain.cfg**.

15.9. REDÉMARRAGE D'UNE MACHINE VIRTUELLE

Vous pouvez utiliser **virsh** pour redémarrer une machine virtuelle :

```
virsh reboot [domain-id | domain-name | domain-uuid]
```

Vous pouvez contrôler le comportement au démarrage de la machine virtuelle en modifiant le paramètre **on_reboot** du fichier **xmdomain.cfg**.

15.10. ARRÊT BRUTAL D'UN DOMAINE

Vous pouvez utiliser **virsh** pour arrêter brutalement une machine virtuelle :

```
virsh destroy [domain-name | domain-id | domain-uuid]
```

This command does an immediate ungraceful shutdown and stops any guest domain sessions (which could potentially lead to file corrupted filesystems still in use by the virtual machine). You should use the **destroy** option only when the virtual machine's operating system is non-responsive. For a paravirtualized virtual machine, you should use the **shutdown** option.

15.11. CONVERSION D'UN NOM DE DOMAINE EN UN ID DE DOMAINE

Vous pouvez utiliser **virsh** pour convertir un nom de domaine ou un UUID en un ID de domaine :

```
virsh domid [domain-name | domain-uuid]
```

15.12. CONVERSION D'UN ID DE DOMAINE EN UN NOM DE DOMAINE

Vous pouvez utiliser **virsh** pour convertir un ID de domaine ou un UUID en un nom de domaine :

```
virsh domname [domain-name | domain-uuid]
```

15.13. CONVERSION D'UN NOM DE DOMAINE EN UN UUID

Vous pouvez utiliser **virsh** pour convertir un nom de domaine en un UUID :

```
virsh domuuid [domain-id | domain-uuid]
```

15.14. AFFICHAGE DES INFORMATIONS D'UNE MACHINE VIRTUELLE

Vous pouvez utiliser **virsh** pour afficher les informations d'une machine virtuelle identifiée par son ID de domaine, nom de domaine ou UUID :

```
virsh dominfo [domain-id | domain-name | domain-uuid]
```

15.15. AFFICHAGE DES INFORMATIONS D'UN NOEUD

Vous pouvez utiliser **virsh** pour afficher les informations d'un noeud :

```
virsh nodeinfo
```

Les sorties affichent quelque chose de similaire à ceci :

```
CPU model           x86_64
CPU (s)             8
CPU frequency       2895 Mhz
CPU socket(s)       2
Core(s) per socket  2
Threads per core:   2
Numa cell(s)        1
Memory size:        1046528 kb
```

Voici les informations du noeud et les machines qui supportent le processus de virtualisation.

15.16. AFFICHAGE DES MACHINES VIRTUELLES

Vous pouvez utiliser **virsh** pour afficher la liste des machines virtuelles et l'état courant :

```
virsh list domain-name [ --inactive | -- -all]
```

L'option **--inactive** liste les domaines inactifs (les domaines qui ont été définis mais qui actuellement ne sont pas actifs). Le domaine **-- -all** liste tous les domaines, qu'ils soient actifs ou pas. Votre sortie devrait ressembler à celle-ci :

ID	Name	State
0	Domain0	running
1	Domain202	paused
2	Domain010	inactive
3	Domain9600	crashed

Voici les six états d'un domaine :

```
running lists domains currently active on the CPU
blocked lists domains that are blocked
paused lists domains that are suspended
shutdown lists domains that are in process of shutting down
shutoff lists domains that are completely down.
crashed lists domains that are crashed
```

15.17. AFFICHAGE DES INFORMATIONS D'UN CPU VIRTUEL

Vous pouvez utiliser **virsh** pour afficher les informations d'un CPU virtuel à partir d'une machine virtuelle :

```
virsh vcpuinfo [domain-id | domain-name | domain-uuid]
```

15.18. CONFIGURATION DE L'AFFINITÉ DES CPU VIRTUELS

Vous pouvez utiliser **virsh** pour configurer l'affinité des CPU virtuels avec les CPU physiques :

```
virsh vcpupin [domain-id | domain-name | domain-uuid] [vcpu] , [cpulist]
```

Où **[vcpu]** correspond au nombre de VCPU virtuels et **[cpulist]** liste le nombre de CPU physiques.

15.19. CONFIGURATION DU NOMBRE DE CPU VIRTUELS

You can use **virsh** to modify a Virtual Machine's number of CPUs:

```
virsh setvcpus [domain-name | domain-id | domain-uuid] [count]
```

Notez que le nouveau nombre ne peut pas excéder la quantité de CPU que vous avez spécifiée lorsque vous avez créé la machine virtuelle.

15.20. CONFIGURATION DE L'ALLOCATION DE LA MÉMOIRE

You can use **virsh** to modify a domain's memory allocation:

```
virsh setmem [domain-id | domain-name] [count]
```

You must specify the **[count]** in kilobytes. Note that the new count cannot exceed the amount you specified when you created the Virtual Machine. Values lower than 64 MB probably won't work. You can adjust the Virtual Machine memory as necessary.

15.21. CONFIGURATION DE LA MÉMOIRE MAXIMUM

You can use **virsh** to modify a Virtual Machine's maximum memory:

```
virsh setmaxmem [domain-name | domain-id | domain-uuid] [count]
```

You must specify the **[count]** in kilobytes. Note that the new count cannot exceed the amount you specified when you created the Virtual Machine. Values lower than 64 MB probably won't work. The maximum memory doesn't affect the current use of the Virtual Machine (unless the new value is lower which should shrink memory usage).

15.22. MANAGING VIRTUAL NETWORKS

You can use **virsh** to manage virtual networks. To list virtual networks:

```
virsh net-list
```

This command generates output similar to:

```
[root@domain ~]# virsh net-list
Name                State      Autostart
-----
default             active    yes
vnet1                active    yes
vnet2                active    yes
```

To view network information for a specific virtual network:

```
virsh net-dumpxml [vnet name]
```

This displays information about a specified virtual network in XML format:

```
[root@domain ~]# virsh net-dumpxml vnet1
<network>
  <name>vnet1</name>
  <uuid>98361b46-1581-acb7-1643-85a412626e70</uuid>
```

```
<forward dev='eth0' />
<bridge name='vnet0' stp='on' forwardDelay='0' />
<ip address='192.168.100.1' netmask='255.255.255.0'>
  <dhcp>
    <range start='192.168.100.128' end='192.168.100.254' />
  </dhcp>
</ip>
</network>
```

Other **virsh** commands used in managing virtual networks are:

- **virsh net-autostart [network name]** — Autostart a network specified as [network name]
- **virsh net-create [XML file]** — Generates and starts a new network using a preexisting XML file
- **virsh net-define [XML file]** — Generates a new network from a preexisting XML file without starting it
- **virsh net-destroy [network name]** — Destroy a network specified as [network name]
- **virsh net-name [network UUID]** — Convert a specified [network UUID] to a network name
- **virsh net-uuid [network name]** — Convert a specified [network name] to a network UUID
- **virsh net-start [name of an inactive network]** — Starts a previously undefined inactive network
- **virsh net-undefine [name of an inactive network]** — Undefine an inactive network

CHAPITRE 16. GESTION DE MACHINES VIRTUELLES AU MOYEN DE XEND

Le démon de contrôle **xend** effectue des fonctions de gestion système associées aux machines virtuelles. Ce démon contrôle les ressources virtualisées, et **xend** doit être en cours d'exécution pour interagir avec les machines virtuelles. Avant de démarrer **xend**, vous devez spécifier les paramètres d'opération en modifiant le fichier de configuration **xend-config.sxp** qui se situe dans le répertoire **etc/xen**. Voici les paramètres que vous pouvez activer ou désactiver dans le fichier de configuration **xend-config.sxp** :

Tableau 16.1. Paramètres de configuration xend pour la virtualisation Red Hat

Élément	Description
console-limit	Determines the console server's memory buffer limit and assigns values on a per-domain basis
min-mem	Détermine le nombre minimum de méga-octets réservés au domain0 (si vous entrez 0, la valeur ne change pas).
dom0 cpus	Détermine le nombre de CPU utilisés par domain0 (par défaut, au moins 1 CPU est assigné).
enable-dump	Active une sauvegarde lorsqu'un échec se produit (par défaut, ce paramètre est désactivé).
external-migration-tool	Détermine le script ou l'application qui s'occupe de la migration de périphériques externes (les scripts doivent se trouver dans etc/xen/scripts/external-device-migrate).
logfile	Détermine l'emplacement du fichier journal (la valeur par défaut est /var/log/xend.log).
loglevel	Filtre les valeurs du mode de journalisation : DEBUG, INFO, WARNING, ERROR ou CRITICAL (la valeur par défaut est DEBUG).
network-script	Détermine le script qui active l'environnement de mise en réseau (les scripts doivent se trouver dans le répertoire etc/xen/scripts).
xend-http-server	Active le serveur de gestion des paquets de flux http (par défaut ce paramètre est désactivé).
xend-unix-server	Active le serveur de sockets de domaine unix (un serveur de sockets est un point d'accès de communication qui traite les connexions réseau de bas niveau et accepte ou rejette les connexions entrantes).

Élément	Description
xend-relocation-server	Active le serveur de délocalisation pour les migrations entre-machines (par défaut ce paramètre est désactivé).
xend-unix-path	Détermine l'emplacement où la commande xend-unix-server envoie les données (la valeur par défaut est var/lib/xend/xend-socket).
xend-port	Détermine le port utilisé par le serveur de gestion http (la valeur par défaut est 8000).
xend-relocation-port	Détermine le port utilisé par le serveur de délocalisation (la valeur par défaut est 8002).
xend-relocation-address	Détermine les adresses de la machine virtuelle qui sont autorisées pour la migration système.
xend-address	Détermine l'adresse à laquelle le serveur de sockets de domaine est lié.

Après avoir configuré ces paramètres d'opération, vérifiez que xend est en cours d'exécution et, si ce n'est pas le cas, initialisez le démon. À l'invite de commande, vous pouvez démarrer le démon **xend** en saisissant ce qui suit :

```
service xend start
```

Vous pouvez utiliser **xend** pour arrêter le démon :

```
service xend stop
```

Cela arrête le démon en cours d'exécution.

Vous pouvez utiliser **xend** pour redémarrer le démon :

```
service xend restart
```

Le démon démarre une nouvelle fois.

Vous pouvez vérifier le statut du démon **xend**.

```
service xend status
```

The output displays the daemon's status.

CHAPITRE 17. GESTION DES MACHINES VIRTUELLES AVEC LE GESTIONNAIRE DE MACHINES VIRTUELLES

Cette section décrit les fenêtres, boîtes de dialogue et divers contrôles GUI du gestionnaire de machines virtuelles de la Virtualisation Red Hat (VMM, de l'anglais Virtual Machine Manager).

17.1. ARCHITECTURE DU GESTIONNAIRE DE MACHINES VIRTUELLES

La Virtualisation Red Hat est une collection de composants logiciels qui travaillent ensemble pour héberger et gérer les machines virtuelles. Le gestionnaire de machines virtuelles (VMM) vous fournit une vue graphique des machines virtuelles de votre système. Vous pouvez utiliser VMM pour définir les machines pleinement et partiellement virtuelles. En utilisant le gestionnaire de machines virtuelles, vous pouvez effectuer autant de tâches de gestion de virtualisation que vous le désirez, y compris l'attribution de mémoire, l'attribution de CPU virtuels, le contrôle de performance opérationnelle, l'enregistrement, la restauration, la mise en pause, la reprise et l'arrêt de systèmes virtuels. Il vous permet également d'accéder à la console textuelle et graphique. La Virtualisation Red Hat extrait les ressources de CPU et de mémoire à partir des configurations matérielles et réseau sous-jacentes. Cela permet aux ressources de traitement d'être groupées et dynamiquement assignées aux applications et requêtes de service. La virtualisation au niveau des puces (Chip-Level) permet aux systèmes d'exploitation équipés de matériels Intel VT et AMD Pacifica de s'exécuter sur les hyperviseurs.

17.2. LA FENÊTRE OUVRIR UNE CONNEXION

This window appears first and prompts the user to choose a hypervisor session. Non-privileged users can initiate a read-only session. Root users can start a session with full blown read-write status. For normal use, select the **Local Xen host** option. You start the Virtual Machine Manager test mode by selecting the **Other hypervisor** and then type `test:///default` in the URL field beneath. Once in test mode, you can connect to a libvirt dummy hypervisor. Note that although the **Remote Xen host screen** is visible, the functionality to connect to such a host is not implemented into Red Hat Enterprise Linux 5.1.



Figure 17.1. Virtual Machine Manager Connection window

17.3. LA FENÊTRE GESTIONNAIRE DE MACHINES VIRTUELLES

La fenêtre principale affiche toutes les machines virtuelles et ressources en cours d'exécution qui lui sont allouées (y compris domain0). Vous pouvez décider quels champs afficher. En cliquant deux fois sur la machine virtuelle désirée, la console respectivo de cette machine particulière apparaîtra. Si vous sélectionnez une machine virtuelle et cliquez deux fois sur le bouton **Affichage**, la fenêtre de détails pour cette machine apparaît. Vous pouvez également accéder au menu **Fichier** pour créer une nouvelle machine virtuelle.

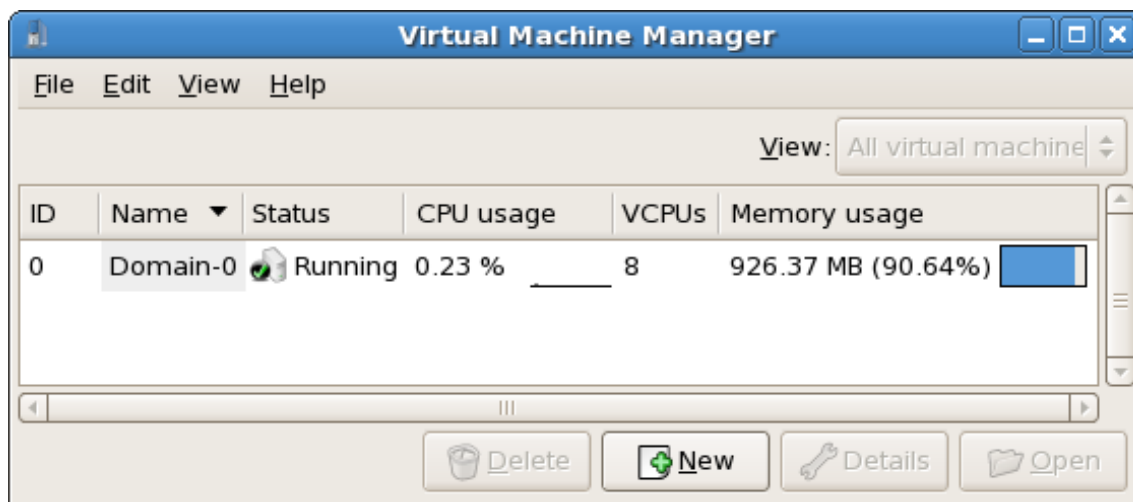


Figure 17.2. Virtual Machine Manager main window

17.4. LA FENÊTRE DÉTAILS D'UNE MACHINE VIRTUELLE

This window displays graphs and statistics of a guest's live resource utilization data available from the Red Hat Virtualization Virtual Machine Manager. The UUID field displays the globally unique identifier for the virtual machines(s).

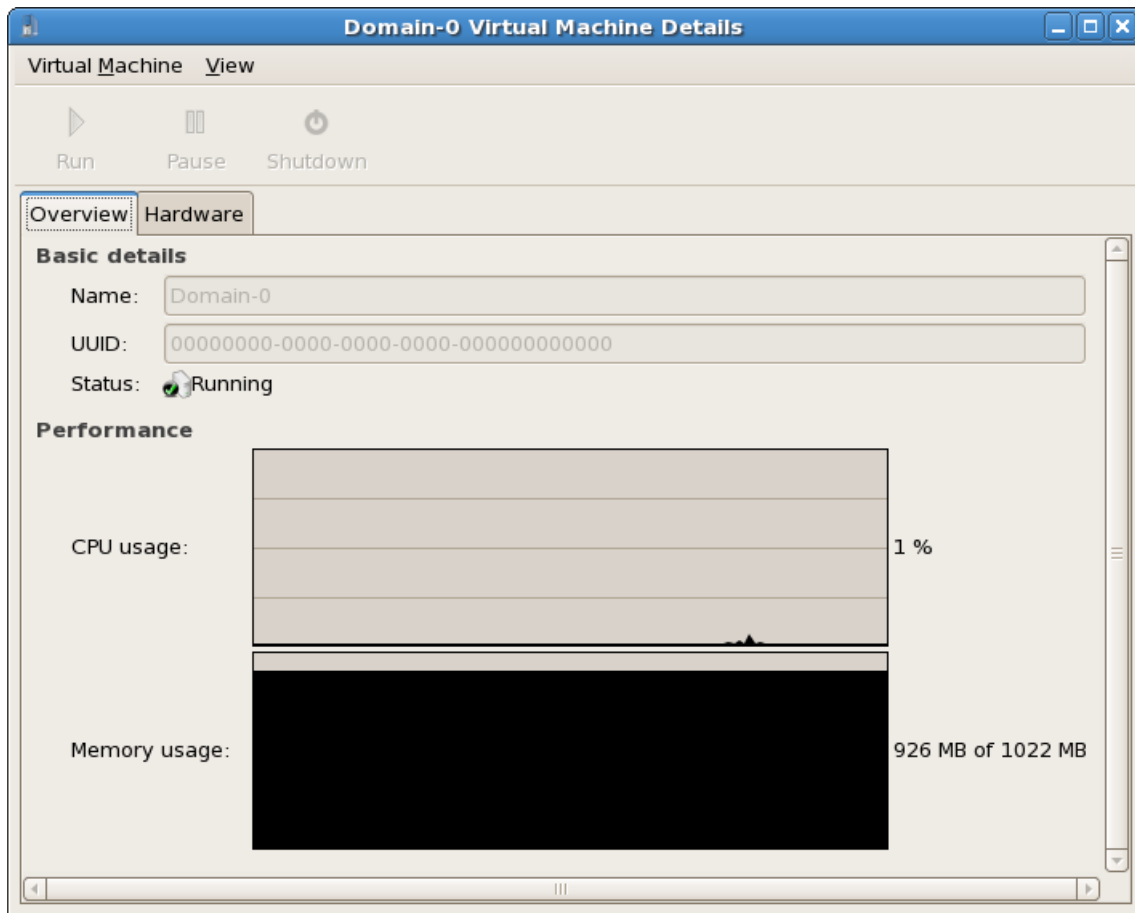


Figure 17.3. Virtual Machine Manager Details window

17.5. LA CONSOLE GRAPHIQUE D'UNE MACHINE VIRTUELLE

This window displays a virtual machine's graphical console. Paravirtual and full virtual machines use different techniques to export their local virtual framebuffers, but both technologies use VNC to make them available to the Virtual Machine Manager's console window. If your virtual machine is set to require authentication, the Virtual Machine Graphical console prompts you for a password before the display appears.

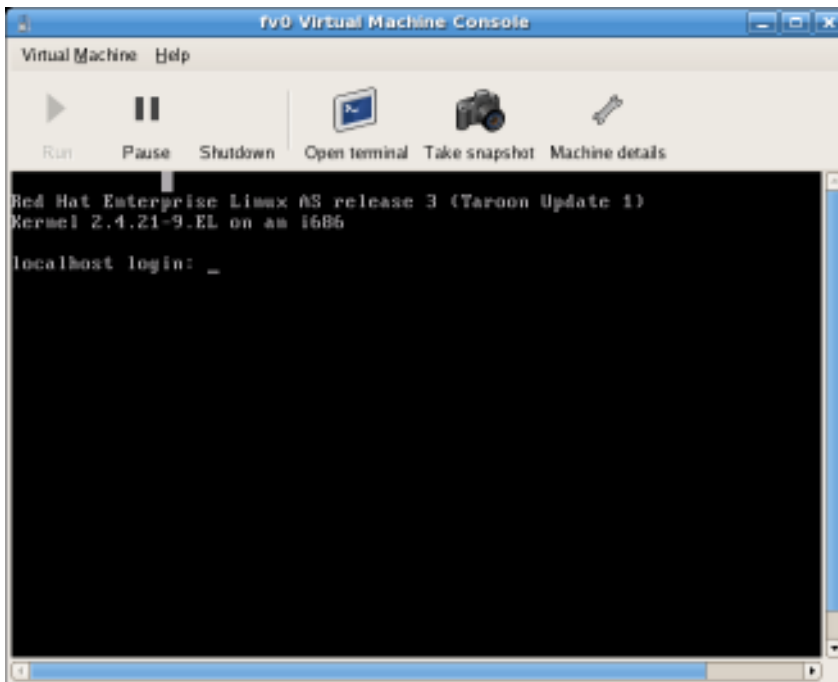


Figure 17.4. La fenêtre de console graphique

Your local desktop can intercept key combinations (for example, Ctrl+Alt+F11) to prevent them from being sent to the guest machine. You can use the Virtual Machine Manager's 'sticky key' capability to send these sequences. You must press any modifier key (like Ctrl or Alt) 3 times and the key you specify gets treated as active until the next non-modifier key is pressed. Then you can send Ctrl-Alt-F11 to the guest by entering the key sequence 'Ctrl Ctrl Ctrl Alt+F1'.

17.6. STARTING THE VIRTUAL MACHINE MANAGER

Pour démarrer le gestionnaire de machines virtuelles à partir du menu **Applications**, cliquez sur **Outils de système** et sélectionnez **Gestionnaire de machines virtuelles**.

La fenêtre principale du gestionnaire de machines virtuelles apparaît.

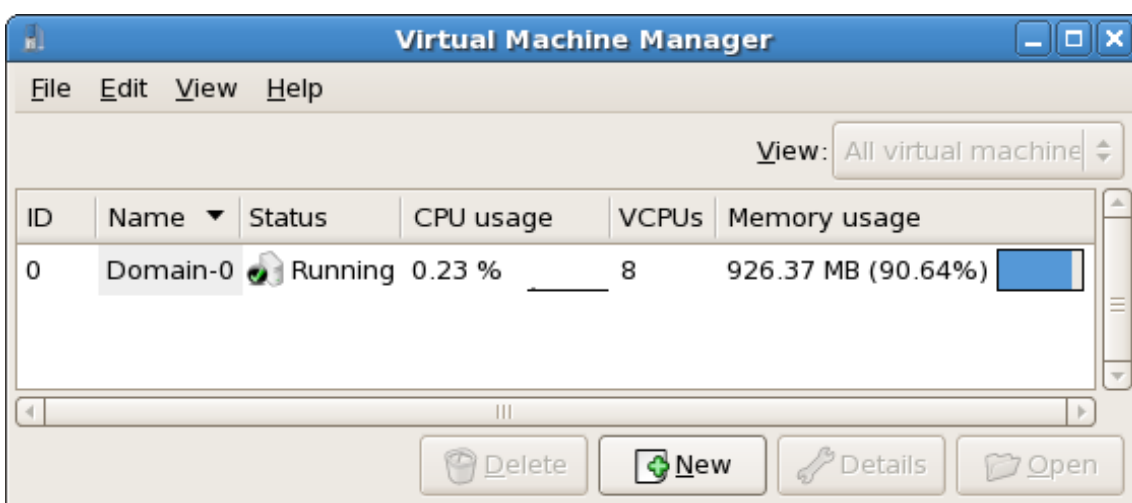


Figure 17.5. Starting the Virtual Machine Manager

17.7. CRÉATION D'UNE NOUVELLE MACHINE VIRTUELLE

Le **gestionnaire de machines virtuelles** (virt-manager) est l'application bureautique qui gère les machines virtuelles.

You can use Red Hat's Virtual Machine Manager to:

- Créer de nouveaux domaines
- Configurer ou ajuster la répartition des ressources et le matériel virtuel d'un domaine.
- Analyser les domaines en cours d'exécution avec les statistiques de performance et d'utilisation des ressources
- Afficher les graphiques qui illustrent les performances et l'utilisation des ressources dans le temps
- Utiliser le client VNC embarqué (visionneuse) qui présente une console graphique complète au domaine invité



NOTE

You must install Red Hat Enterprise Linux 5.1, virt-manager, and the kernel packages on all systems that require virtualization. All systems then must be booted and running the Red Hat Virtualization kernel.

Voici les étapes requises pour installer un système d'exploitation invité sur Red Hat Enterprise Linux 5 en utilisant le contrôleur de machines virtuelles :

Procédure 17.1. Création d'un système d'exploitation invité

1. À partir du menu **Applications**, sélectionnez **Outils de système** et **Gestionnaire de machines virtuelles**.

La fenêtre principale du gestionnaire de machines virtuelles apparaît.

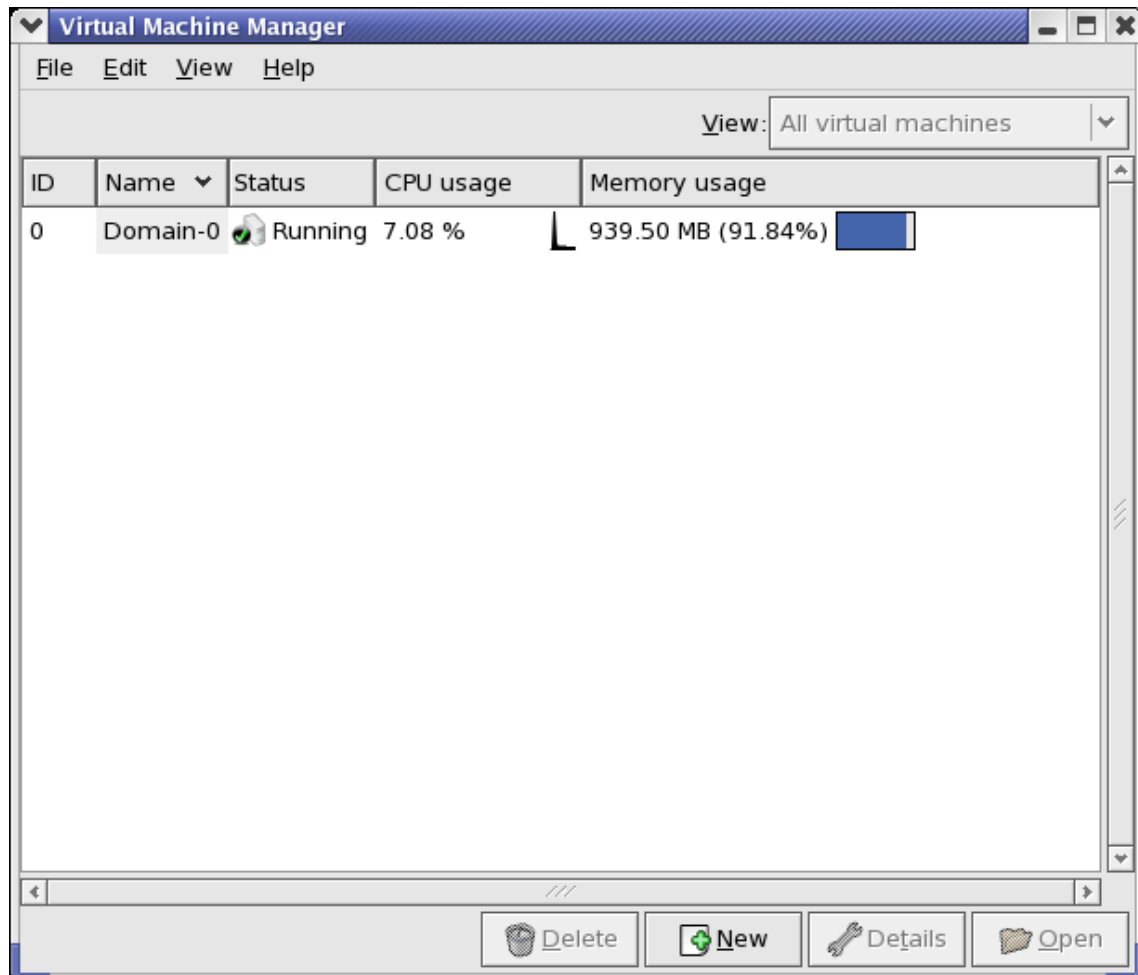


Figure 17.6. Virtual Machine Manager window

2. À partir du menu **Fichier**, sélectionnez **Nouvelle machine**.

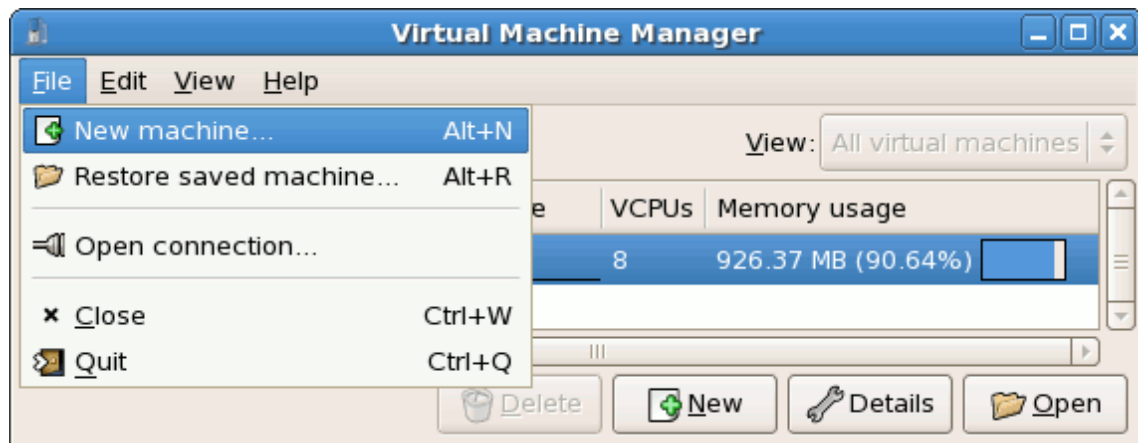


Figure 17.7. Sélection d'une nouvelle machine

L'assistant de création d'un nouveau système virtuel apparaît.

3. Click **Forward**.

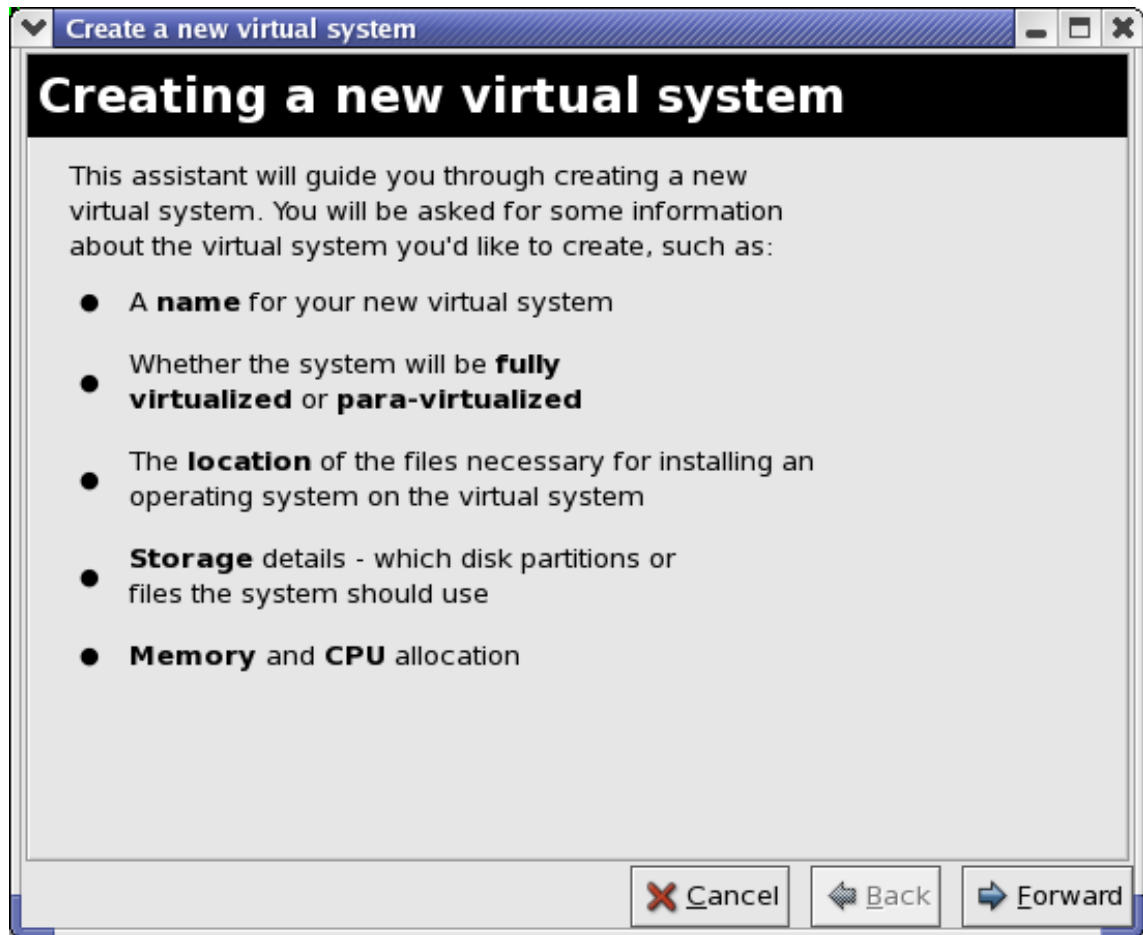


Figure 17.8. Assistant de création d'un nouveau système virtuel

4. Enter the name of the new virtual system and then click **Forward**.

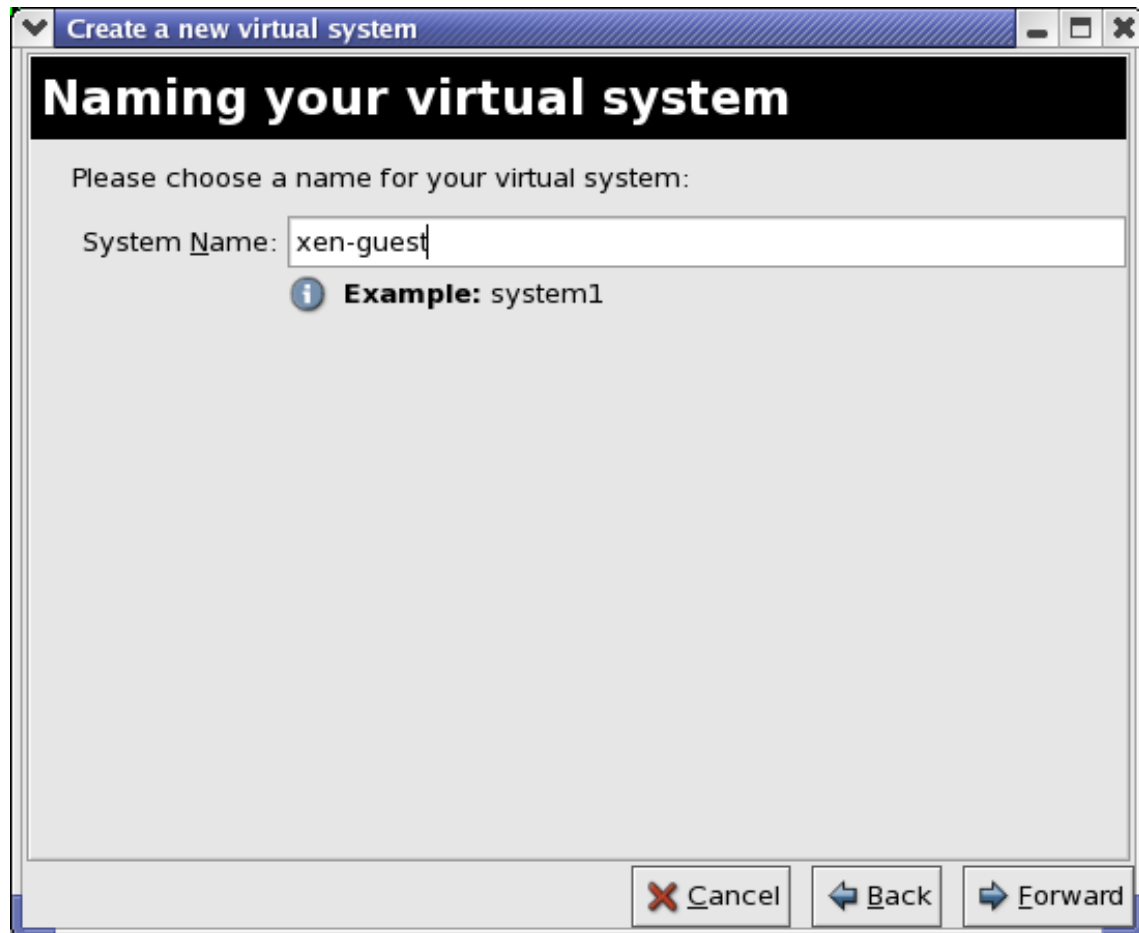


Figure 17.9. Nommage du système virtuel

5. Saisissez l'emplacement de votre support d'installation. L'emplacement du fichier kickstart est optionnel. Cliquez sur **Suivant**.

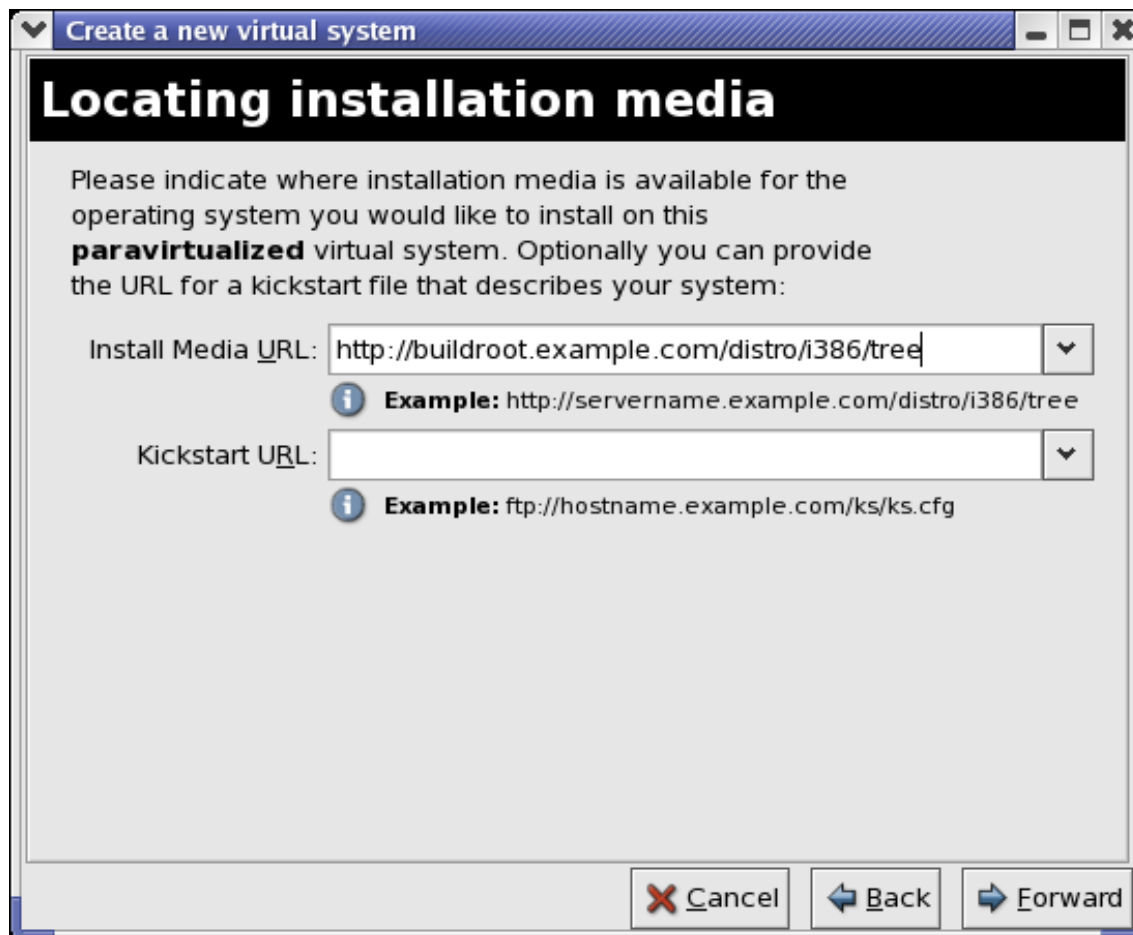


Figure 17.10. Emplacement du support d'installation

6. Installez-le sur une partition de disque physique ou sur un système de fichiers virtuels dans un fichier.



NOTE

Cet exemple installe un système virtuel dans un fichier.

SELinux policy only allows xen disk images to reside in `/var/lib/xen/images`.

Open a terminal and create the `/xen` directory and set the SELinux policy with the command `restorecon -v /xen`. Specify your location and the size of the virtual disk, then click **Forward**.



Figure 17.11. Assigner l'espace de stockage

7. Select memory to allocate the guest and the number of virtual CPUs then click **Forward**.

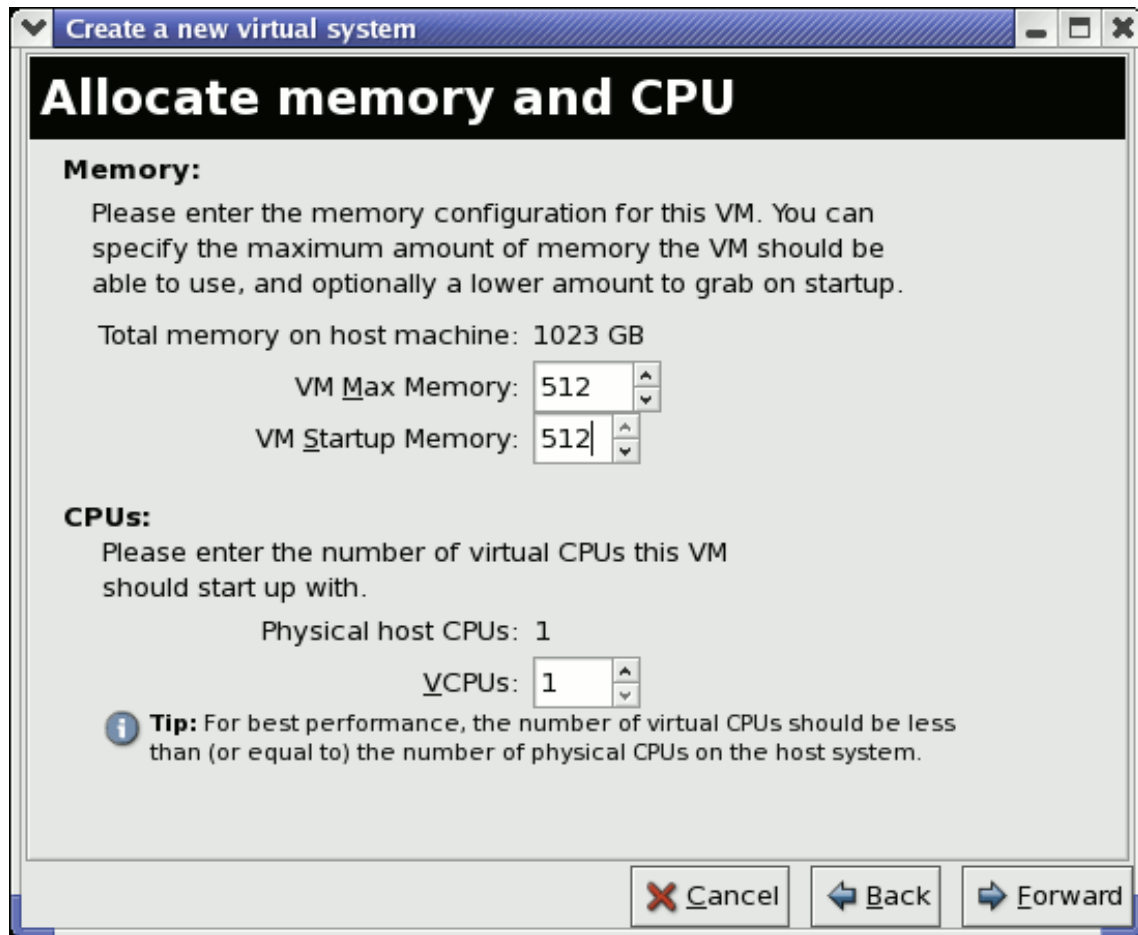
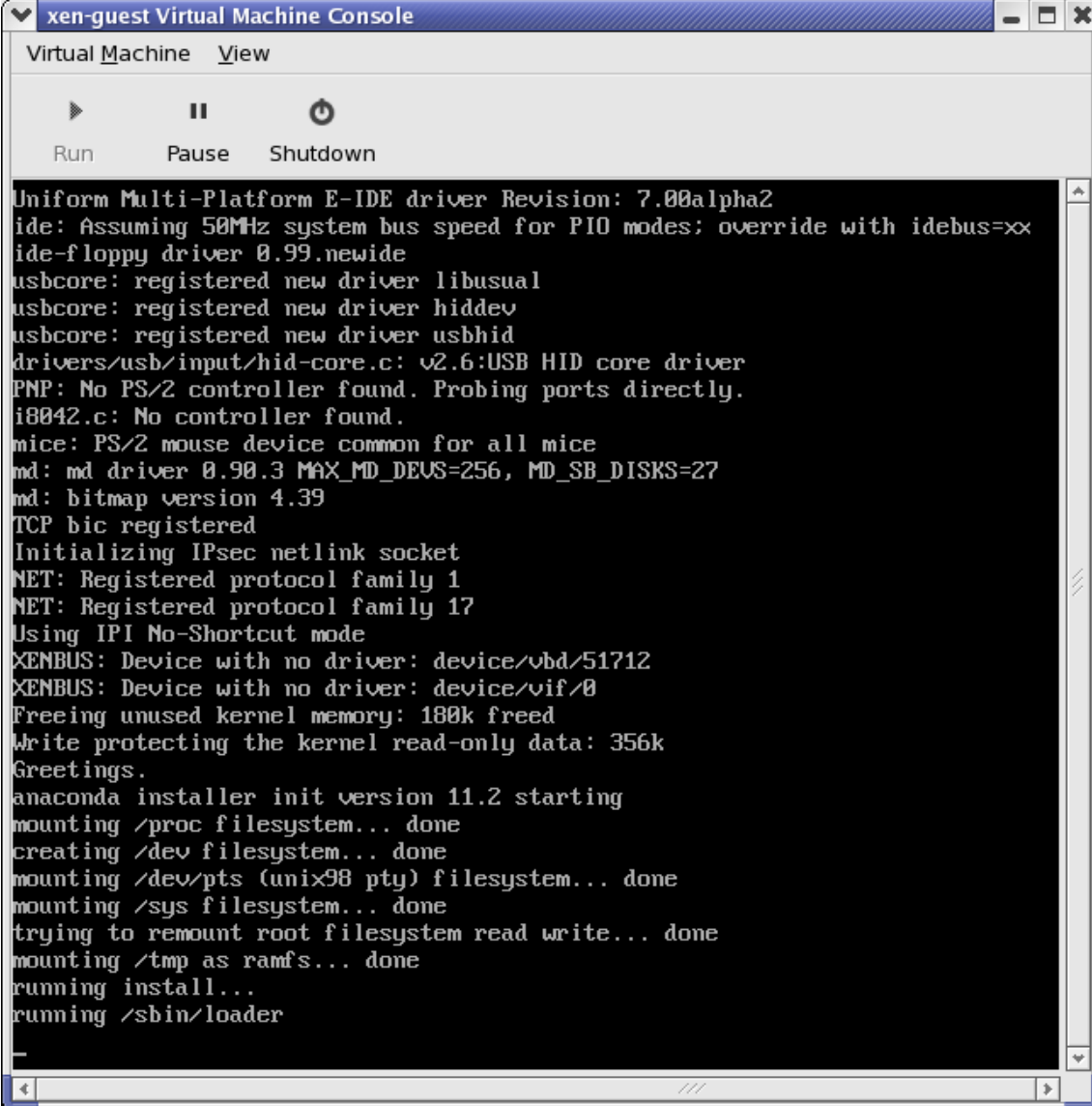


Figure 17.12. Allocation mémoire et CPU

8. Select **Forward** to open a console and the files start to install.

The image shows a window titled "xen-guest Virtual Machine Console". The window has a menu bar with "Virtual Machine" and "View". Below the menu bar are three buttons: "Run", "Pause", and "Shutdown". The main area of the window is a black terminal with white text. The text shows the boot process of a virtual machine, including kernel messages and the start of the anaconda installer. The terminal output is as follows:

```
Uniform Multi-Platform E-IDE driver Revision: 7.00alpha2
ide: Assuming 50MHz system bus speed for PIO modes; override with idebus=xx
ide-floppy driver 0.99.newide
usbcore: registered new driver libusual
usbcore: registered new driver hiddev
usbcore: registered new driver usbhid
drivers/usb/input/hid-core.c: v2.6:USB HID core driver
PNP: No PS/2 controller found. Probing ports directly.
i8042.c: No controller found.
mice: PS/2 mouse device common for all mice
md: md driver 0.90.3 MAX_MD_DEVS=256, MD_SB_DISKS=27
md: bitmap version 4.39
TCP bic registered
Initializing IPsec netlink socket
NET: Registered protocol family 1
NET: Registered protocol family 17
Using IPI No-Shortcut mode
XENBUS: Device with no driver: device/vbd/51712
XENBUS: Device with no driver: device/vif/0
Freeing unused kernel memory: 180k freed
Write protecting the kernel read-only data: 356k
Greetings.
anaconda installer init version 11.2 starting
mounting /proc filesystem... done
creating /dev filesystem... done
mounting /dev/pts (unix98 pty) filesystem... done
mounting /sys filesystem... done
trying to remount root filesystem read write... done
mounting /tmp as ramfs... done
running install...
running /sbin/loader
-
```

Figure 17.13. Allocation mémoire et CPU

9. Poursuivez votre installation avec la fenêtre qui apparaît.

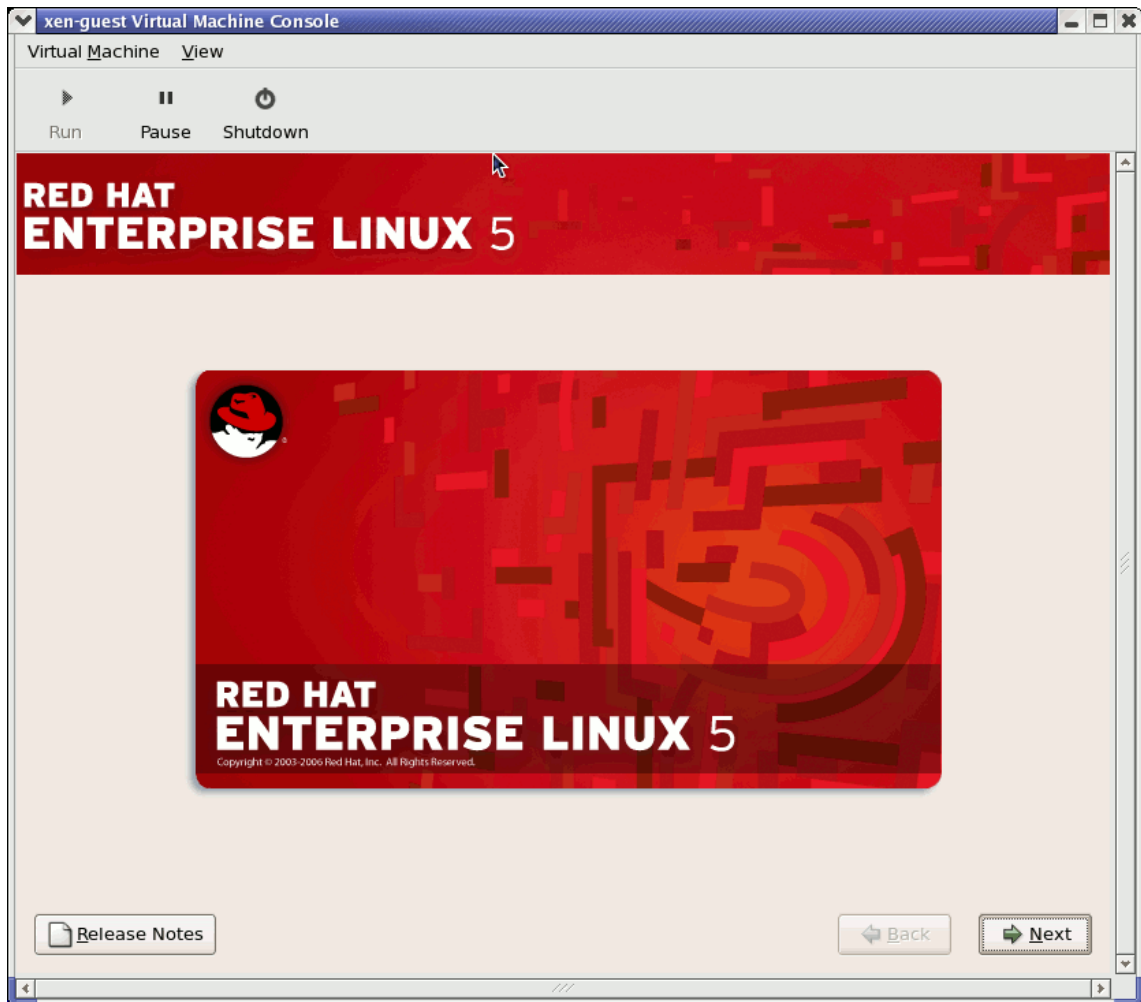


Figure 17.14. L'installation commence...



AVERTISSEMENT

When installing Red Hat Enterprise Linux 5.1 on a fully virtualized guest, do *not* use the **kernel-xen** kernel. Using this kernel on fully virtualized guests can cause your system to hang.

If you are using an *Installation Number* when installing Red Hat Enterprise Linux 5.1 on a fully virtualized guest, be sure to deselect the **Virtualization** package group during the installation. The **Virtualization** package group option installs the **kernel-xen** kernel.

Note that paravirtualized guests are not affected by this issue. Paravirtualized guests always use the **kernel-xen** kernel.

10. Type `xm create -c xen-guest` to start the Red Hat Enterprise Linux 5.1 guest. Right click on the guest in the Virtual Machine Manager and choose **Open** to open a virtual console.

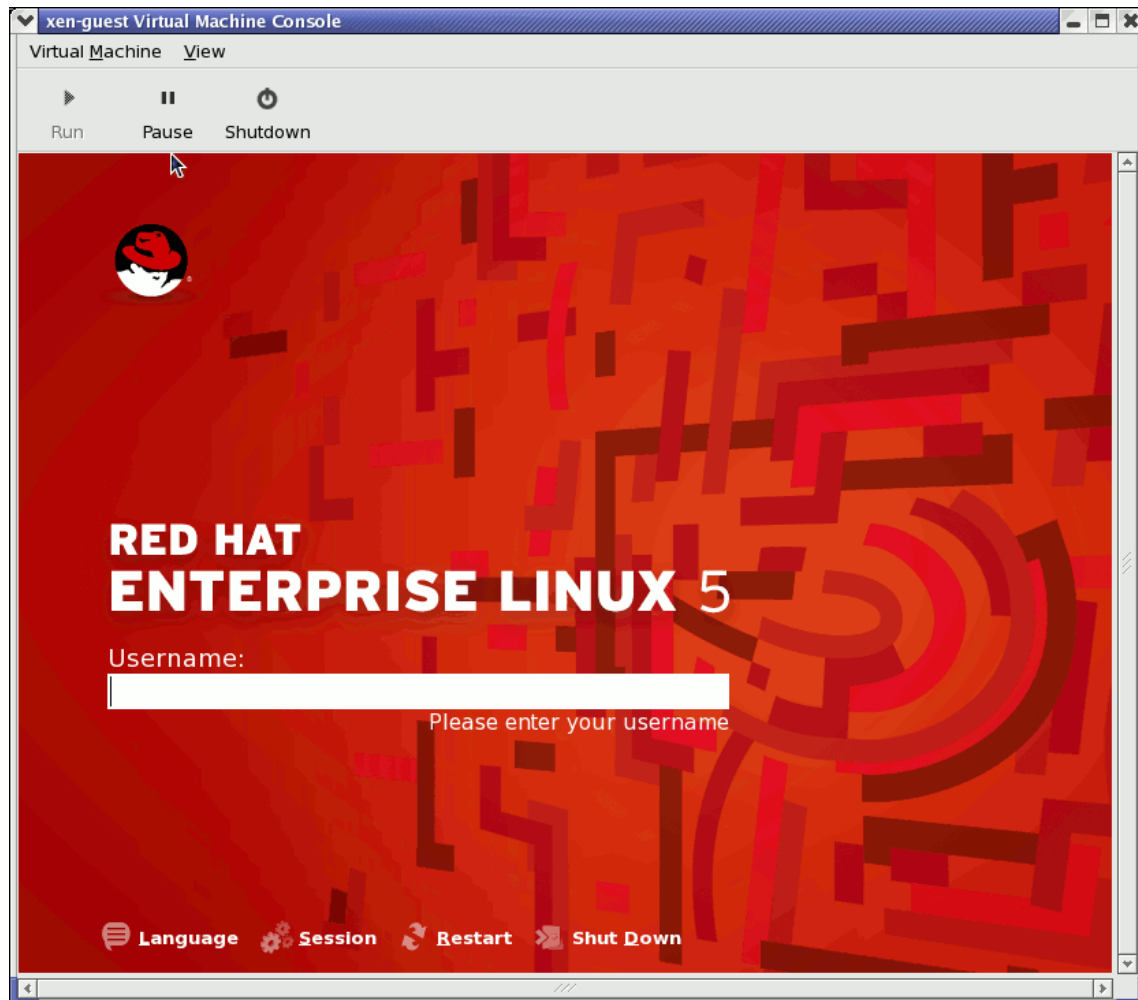


Figure 17.15. Red Hat Enterprise Linux 5.1 (guest)

11. Saisissez le nom d'utilisateur et le mot de passe pour continuer à utiliser le gestionnaire de machines virtuelles.

17.8. RESTAURATION D'UNE MACHINE ENREGISTRÉE

Après avoir démarré le gestionnaire de machines virtuelles, toutes les machines virtuelles de votre système sont affichées dans la fenêtre principale. Domain0 correspond à votre système hôte. Si il n'y a pas de machines affichées, cela signifie qu'il n'y a pas de machines en cours d'exécution sur le système.

Pour restaurer une session précédemment enregistrée :

1. À partir du menu **Fichier**, sélectionnez **Restaurer une machine enregistrée**.

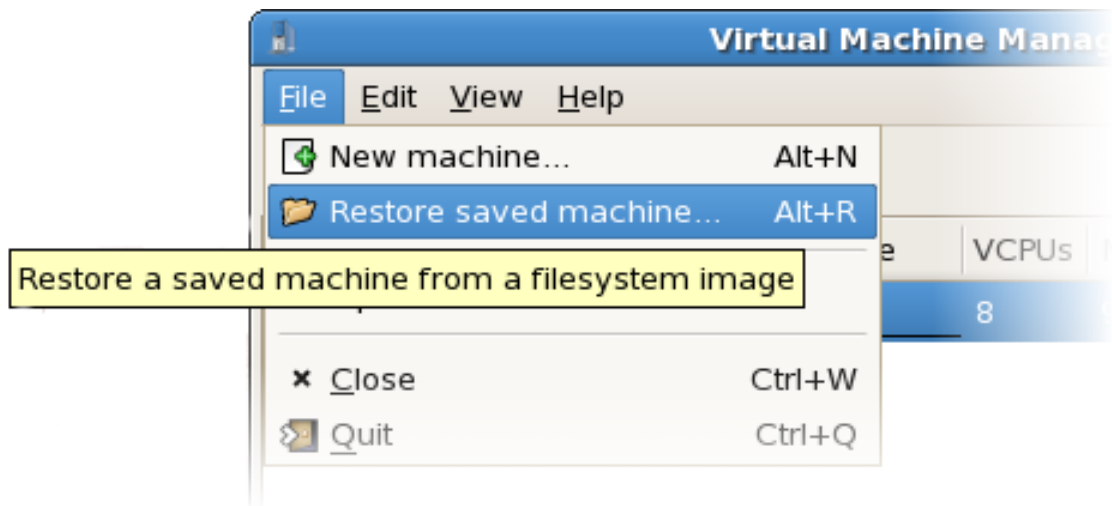


Figure 17.16. Restoring a Virtual Machine

2. La fenêtre principale de restauration d'une machine virtuelle apparaît.

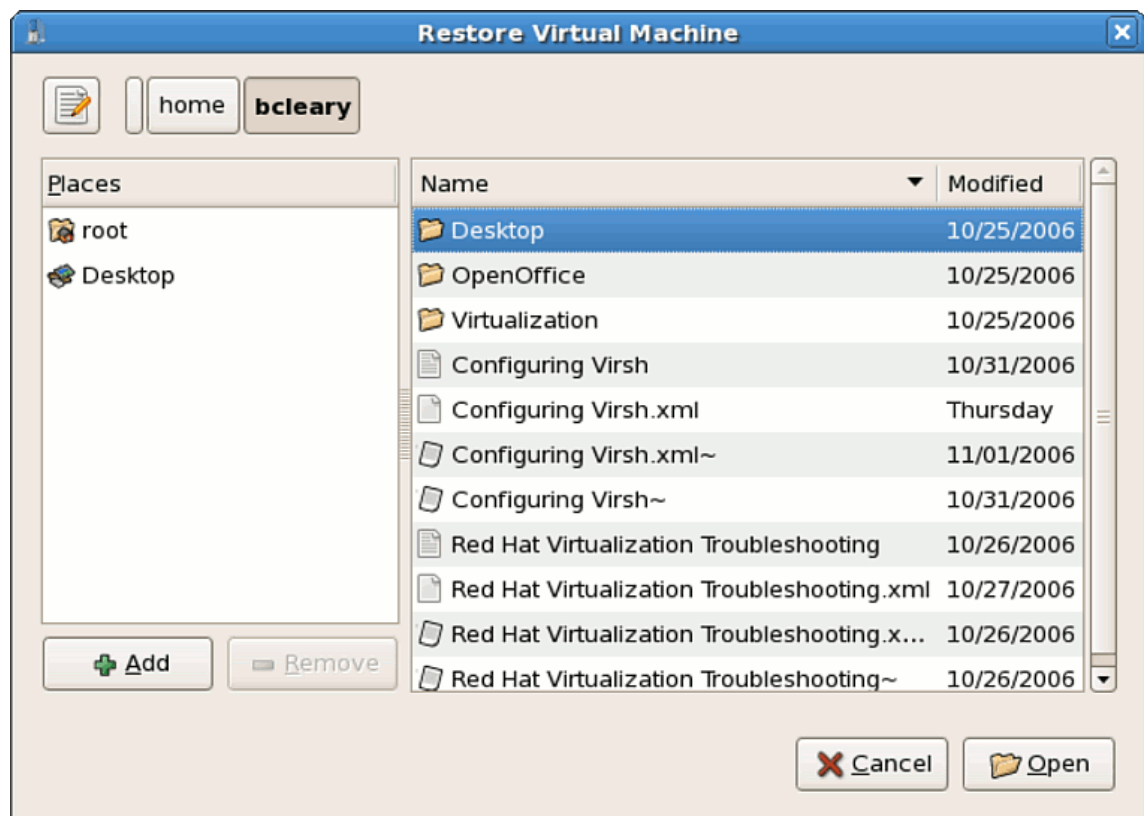


Figure 17.17. Sélection de la session d'une machine virtuelle enregistrée

3. Naviguez dans le répertoire approprié et sélectionnez le fichier enregistré.
4. Cliquez sur **Ouvrir**.

Les systèmes virtuels enregistrés apparaissent dans la fenêtre principale du gestionnaire de machines virtuelles.

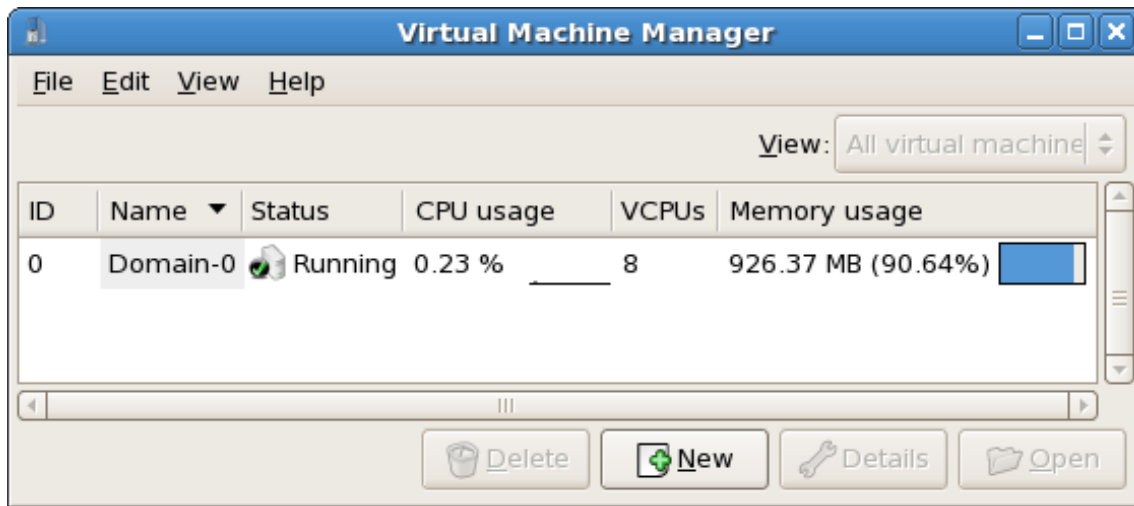


Figure 17.18. La session du gestionnaire de machines virtuelles restaurée

17.9. DISPLAYING VIRTUAL MACHINE DETAILS

Vous pouvez utiliser le contrôleur de machines virtuelles pour voir des informations de données pour toute machine virtuelle de votre système.

To view a virtual system's details:

1. Dans la fenêtre principale du gestionnaire de machines virtuelles, sélectionnez la machine virtuelle que vous voulez afficher.

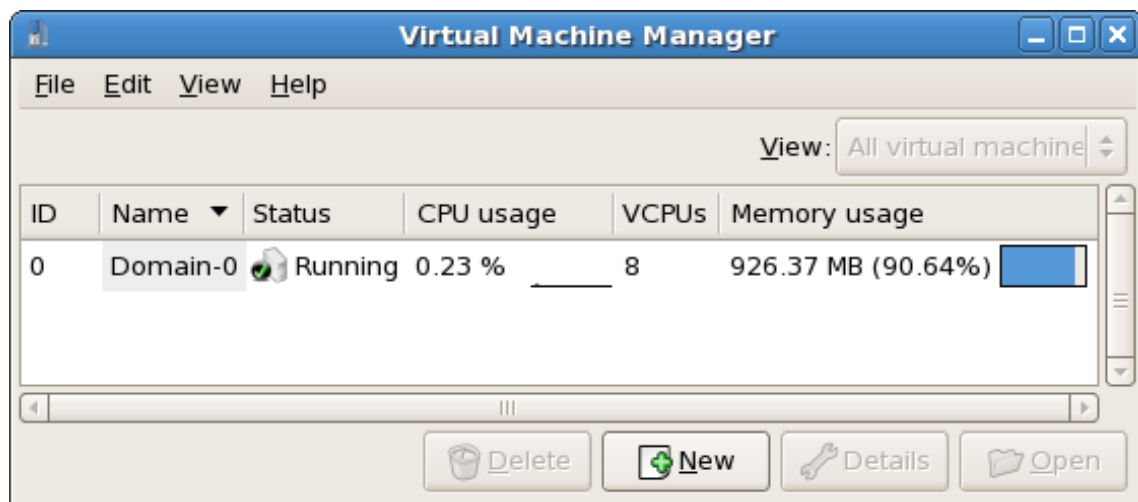


Figure 17.19. Sélection d'une machine virtuelle à afficher

2. À partir du menu **Modifier** du gestionnaire de machines virtuelles, sélectionnez **Détails de la machine** (ou cliquez sur le bouton **Détails** au bas de la fenêtre principale du gestionnaire de machines virtuelles).

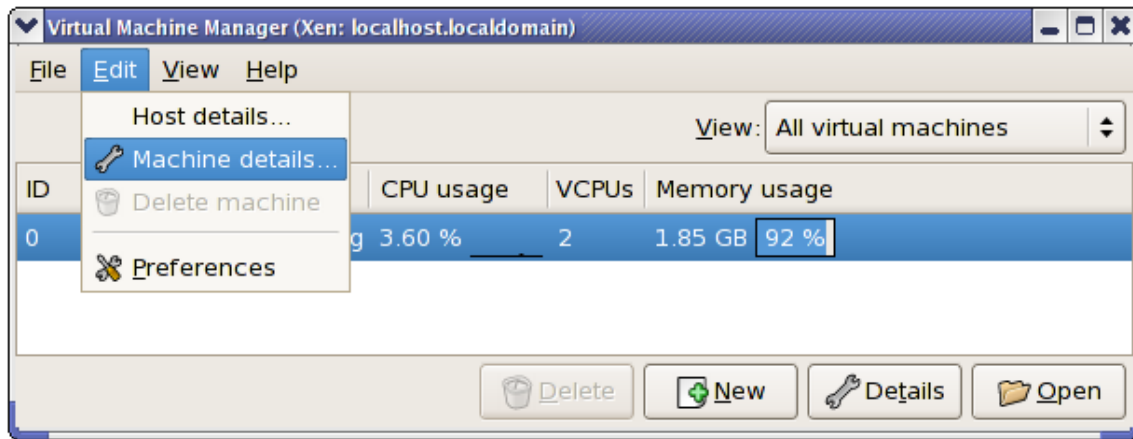


Figure 17.20. Menu Affichage des détails d'une machine virtuelle

La fenêtre d'aperçu des détails d'une machine virtuelle apparaît. Cette fenêtre résume l'utilisation du CPU et de la mémoire pour le/les domaine(s) spécifié(s).

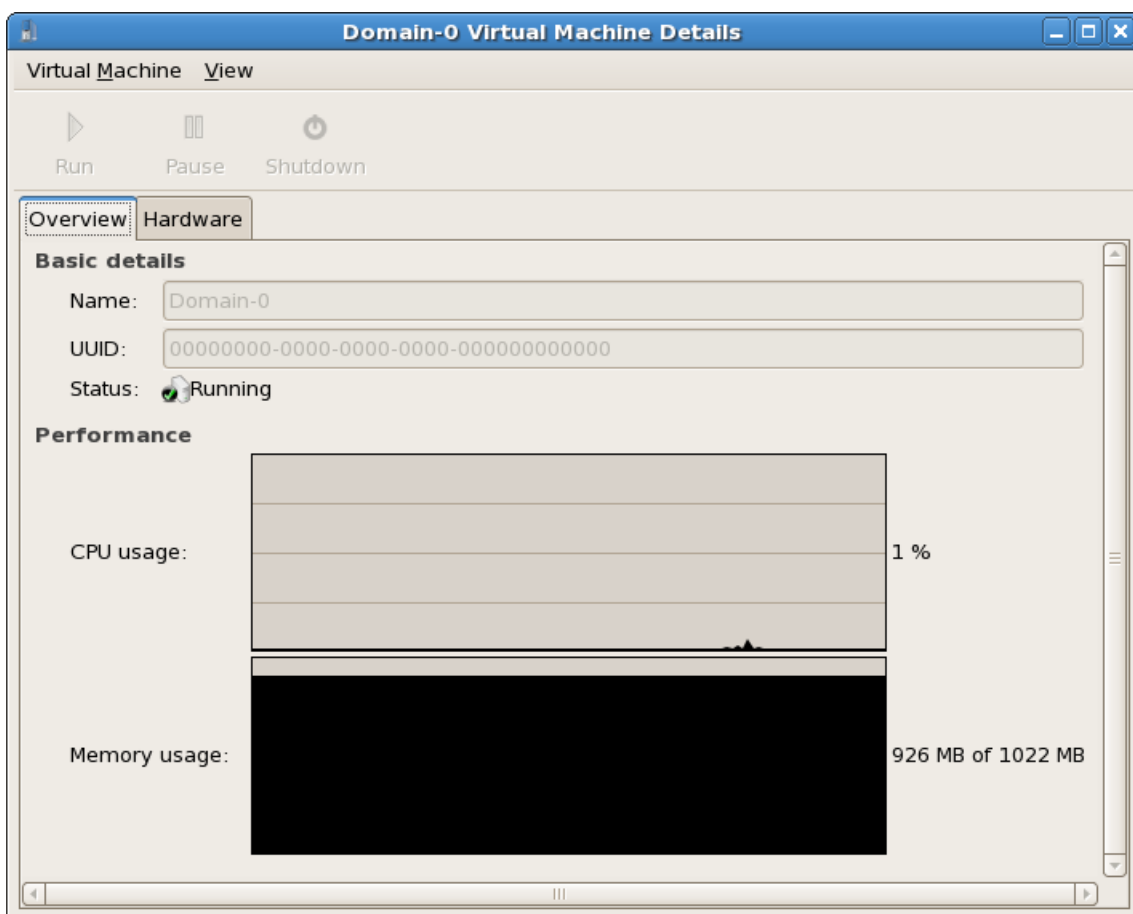


Figure 17.21. Affichage de l'aperçu des détails d'une machine virtuelle

3. Sur la fenêtre des détails d'une machine virtuelle, cliquez sur l'onglet **Matériel**.

La fenêtre des détails du matériel d'une machine virtuelle apparaît.

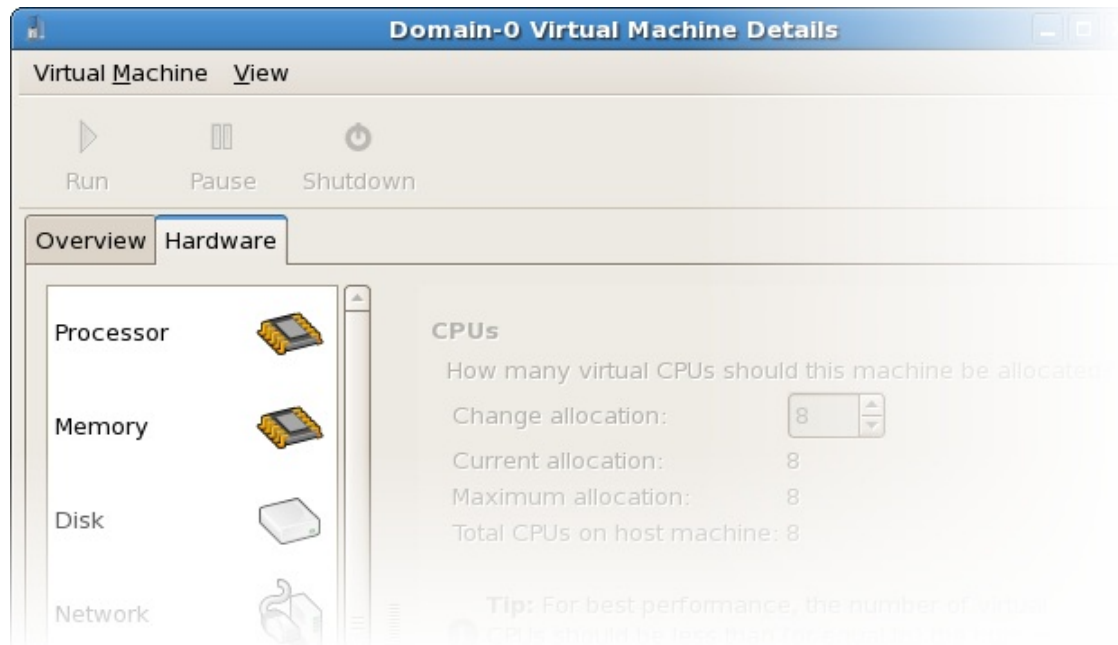


Figure 17.22. Affichage des détails du matériel d'une machine virtuelle

4. Sur l'onglet **Matériel**, cliquez sur **Processeur** pour voir ou changer l'allocation mémoire du processeur.



Figure 17.23. Affichage de l'allocation processeur

5. Sur l'onglet **Matériel**, cliquez sur **Mémoire** pour voir ou changer l'allocation de la mémoire RAM.

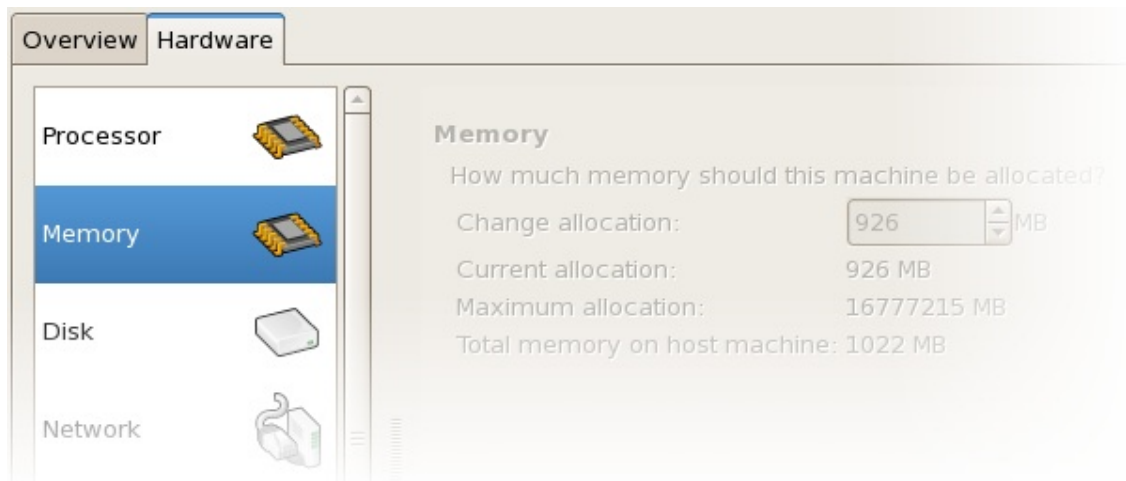


Figure 17.24. Affichage de l'allocation mémoire

6. Sur l'onglet **Hardware**, cliquez sur **Disque** pour voir ou changer la configuration du disque dur.

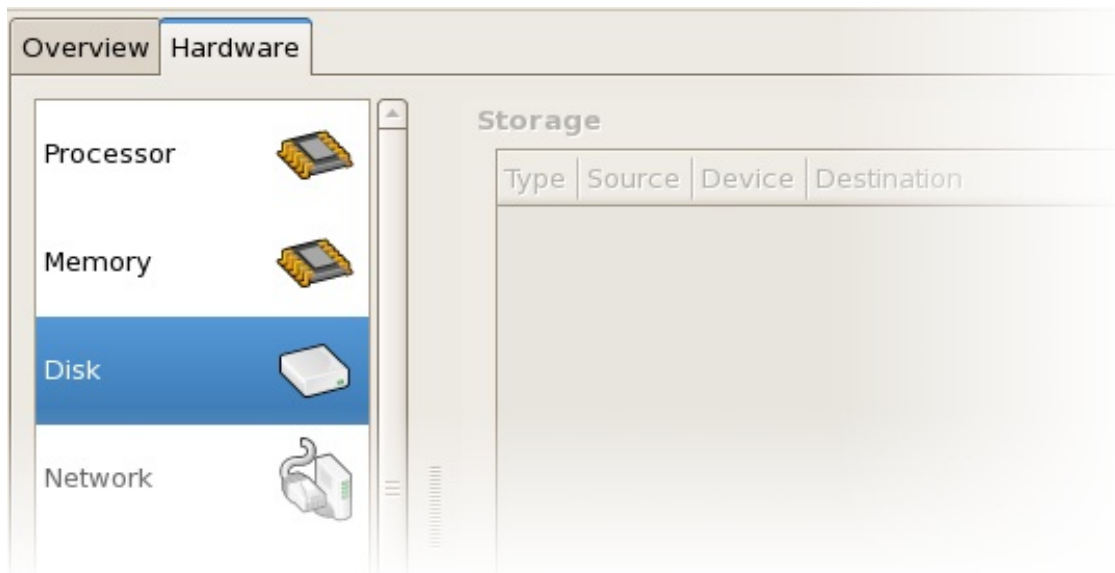


Figure 17.25. Affichage de la configuration du disque

7. Sur l'onglet **Matériel**, cliquez sur **Réseau** pour voir ou changer la configuration du réseau.

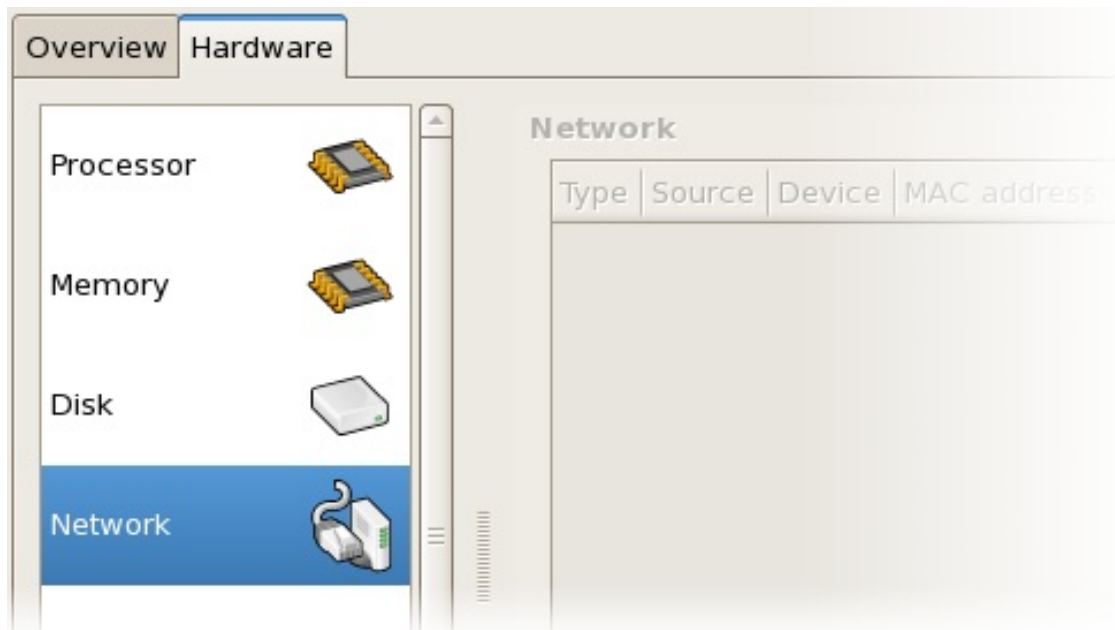


Figure 17.26. Affichage de la configuration du réseau

17.10. CONFIGURING STATUS MONITORING

Vous pouvez utiliser le gestionnaire de machines virtuelles pour modifier la surveillance des états du système virtuel.

Pour configurer la surveillance des états et activer les consoles :

1. À partir du menu **Modifier**, sélectionner **Préférences**.

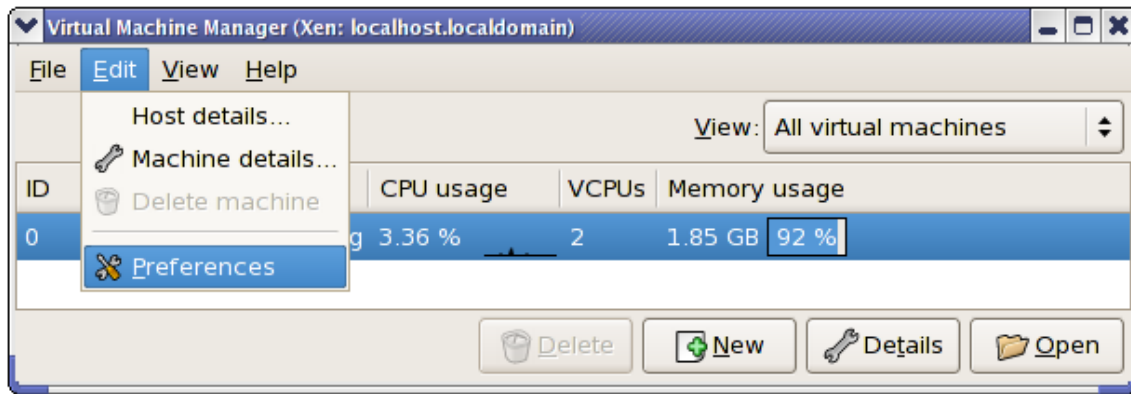


Figure 17.27. Modification des préférences d'une machine virtuelle

La fenêtre des préférences du gestionnaire de machines virtuelles apparaît.

2. À partir de la boîte de dialogue de surveillance des états, spécifiez le délai d'attente (en secondes) avant que le système soit mis à jour.

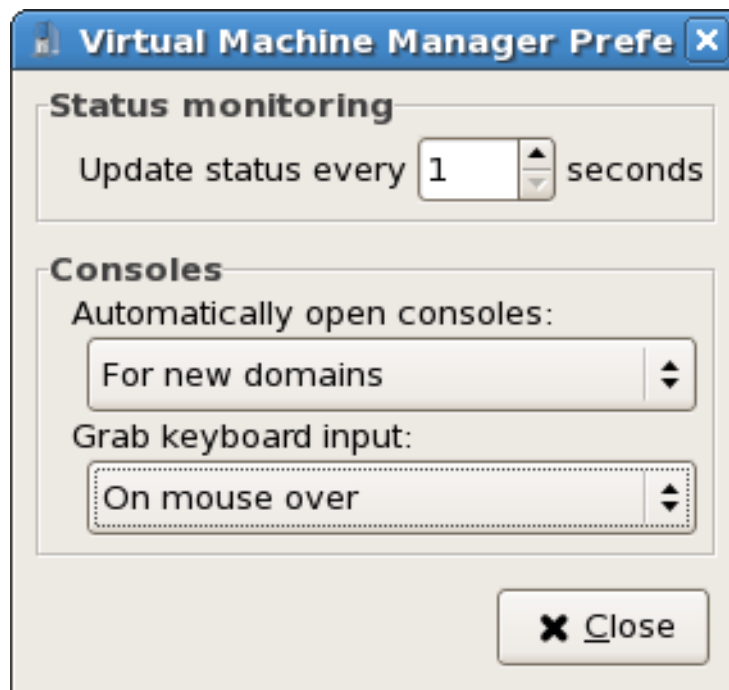


Figure 17.28. Configuring Status Monitoring

3. À partir de la zone des consoles, spécifiez la façon d'ouvrir une console ainsi qu'un périphérique d'entrée.

17.11. AFFICHAGE DE L'ID DU DOMAINE

Pour voir les ID de domaine de toutes les machines virtuelles de votre système :

1. À partir du menu **Affichage**, sélectionnez la case à cocher **ID du domaine**.

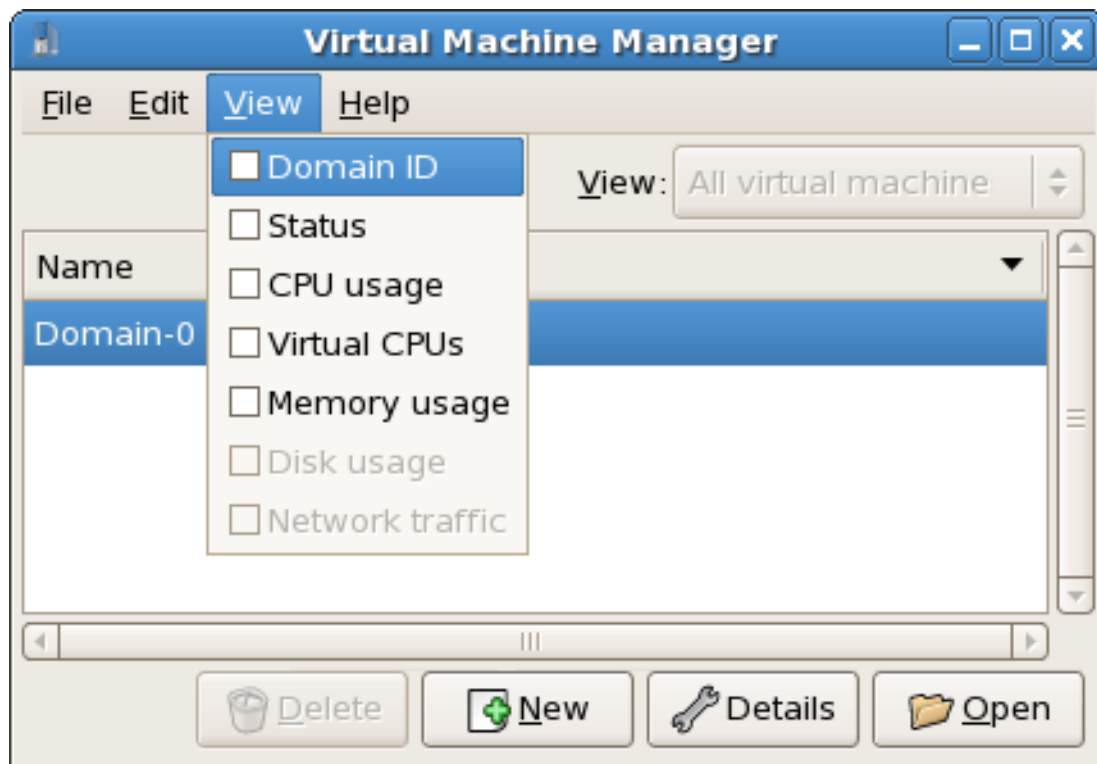


Figure 17.29. Affichage des ID de domaine

2. The Virtual Machine Manager lists the Domain ID's for all domains on your system.

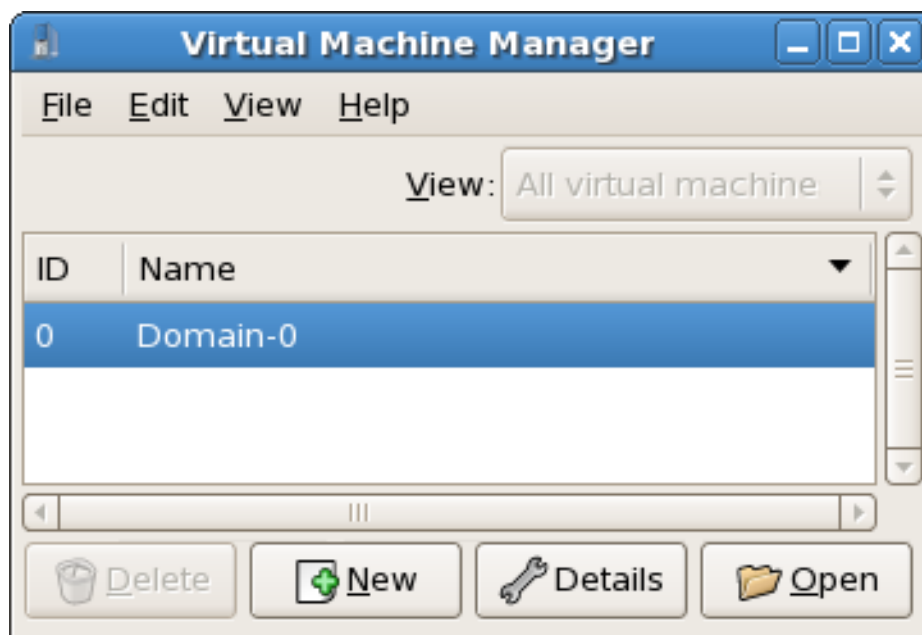


Figure 17.30. Affichage des ID de domaine

17.12. AFFICHAGE DE L'ÉTAT D'UNE MACHINE VIRTUELLE

Pour voir les états de toutes les machines virtuelles de votre système :

1. À partir du menu **Affichage**, sélectionnez la case à cocher **États**.

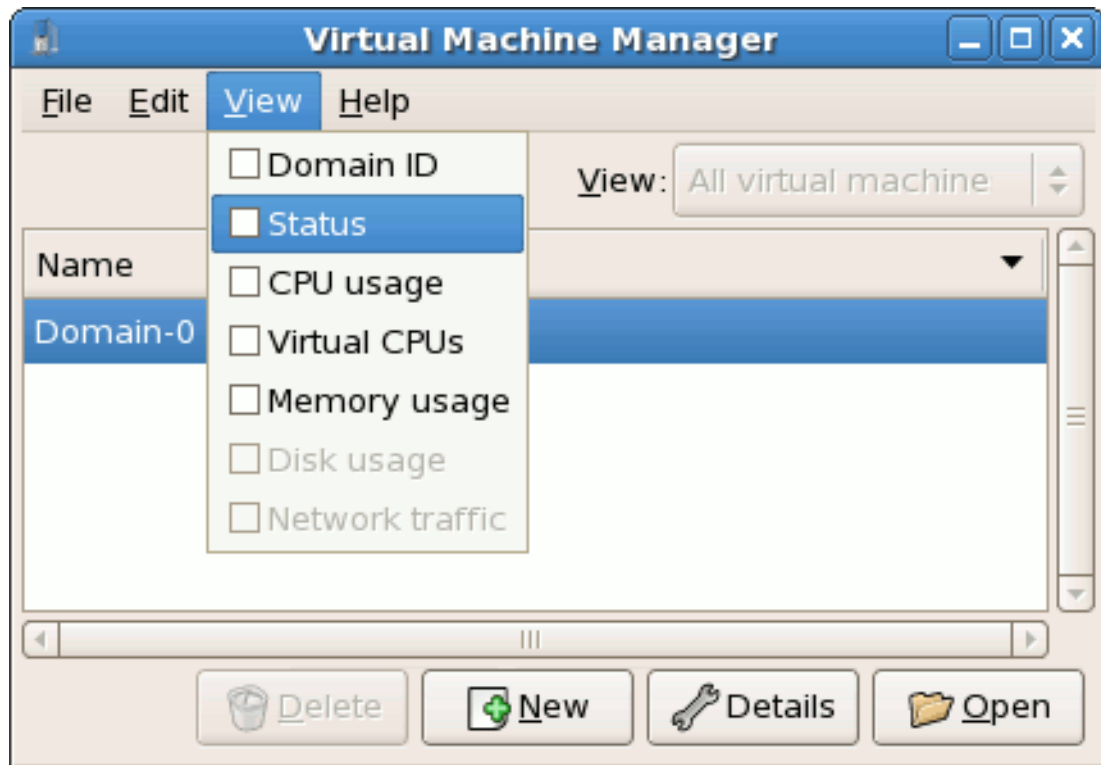


Figure 17.31. Affichage de l'état d'une machine virtuelle

2. Le gestionnaire de machines virtuelles liste l'état de toutes les machines virtuelles de votre système :



Figure 17.32. Affichage de l'état d'une machine virtuelle

17.13. AFFICHAGE DES CPU VIRTUELS

Pour voir la quantité de CPU virtuels de toutes les machines virtuelles de votre système :

1. À partir du menu **Affichage**, sélectionnez la case à cocher **CPU virtuels**.

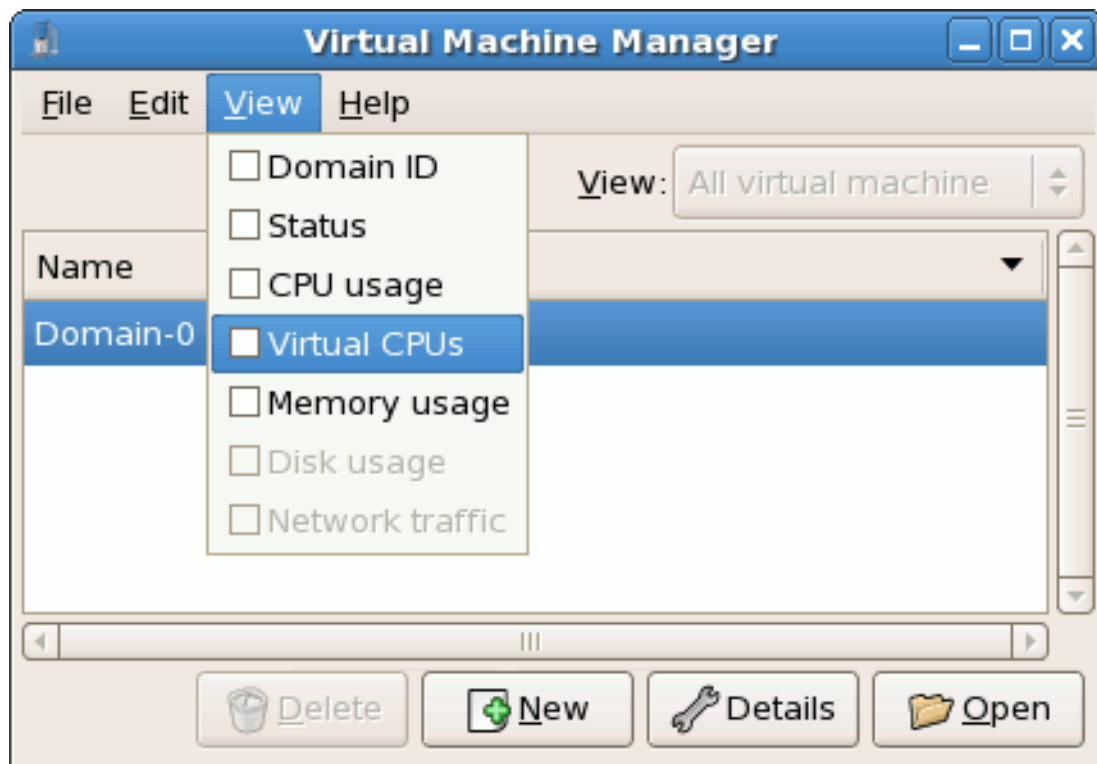


Figure 17.33. Affichage des CPU virtuels

2. Le gestionnaire de machines virtuelles liste les CPU virtuels pour toutes les machines virtuelles sur votre système.

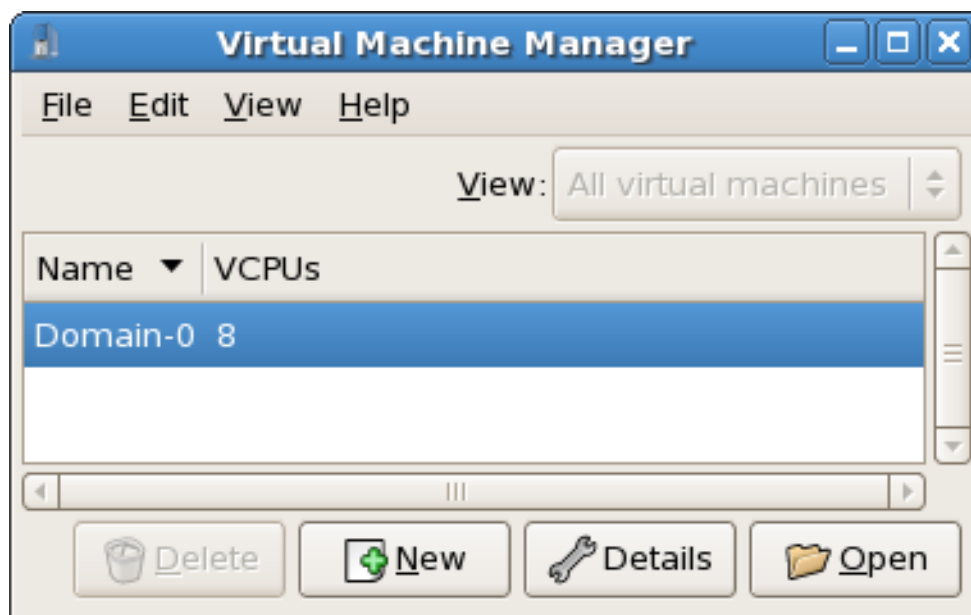


Figure 17.34. Affichage des CPU virtuels

17.14. AFFICHE DE L'UTILISATION CPU

Pour voir l'utilisation CPU de toutes les machines virtuelles de votre système :

1. À partir du menu **Affichage**, sélectionnez la case à cocher **Utilisation CPU**.

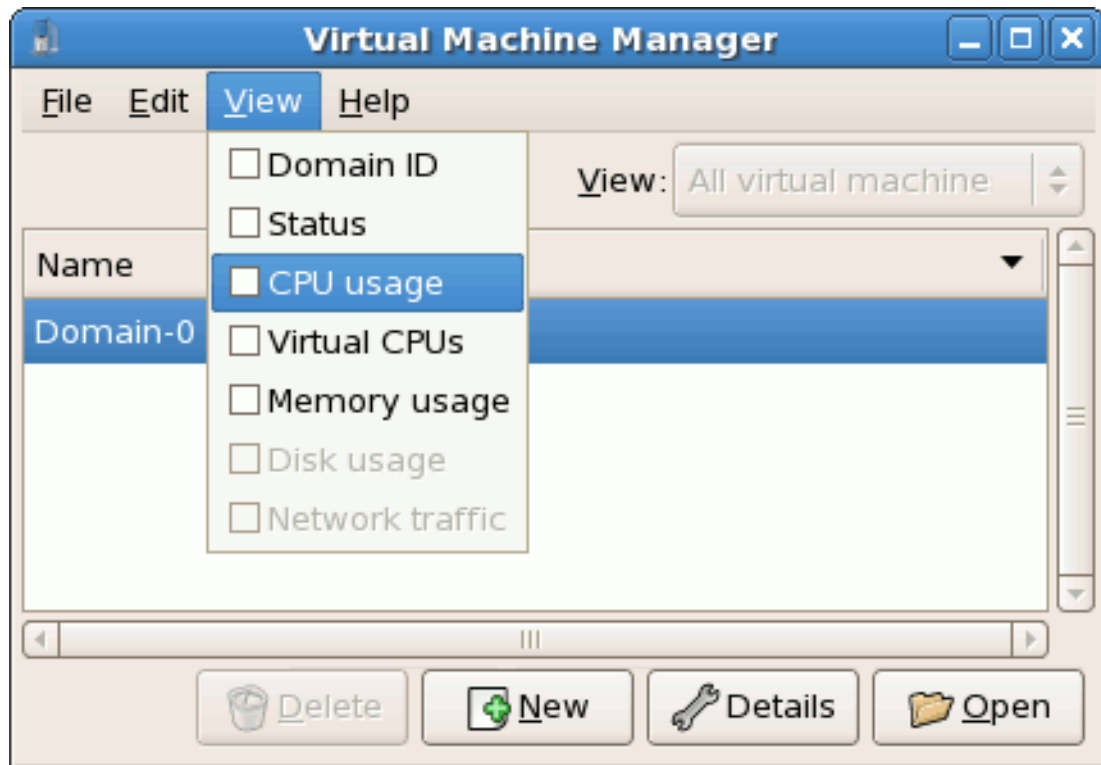


Figure 17.35. Affiche de l'utilisation CPU

2. Le gestionnaire de machines virtuelles liste le pourcentage de CPU en cours d'utilisation de toutes les machines virtuelles de votre système.

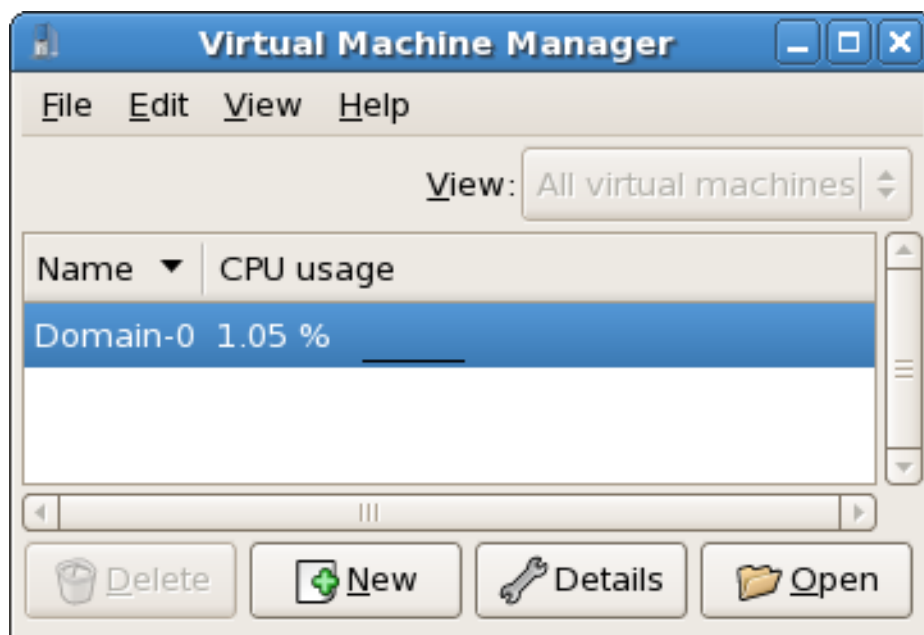


Figure 17.36. Affiche de l'utilisation CPU

17.15. AFFICHAGE DE L'UTILISATION MÉMOIRE

Pour voir l'utilisation mémoire de toutes les machines virtuelles de votre système :

1. À partir du menu **Affichage**, sélectionnez la case à cocher **Utilisation mémoire**.

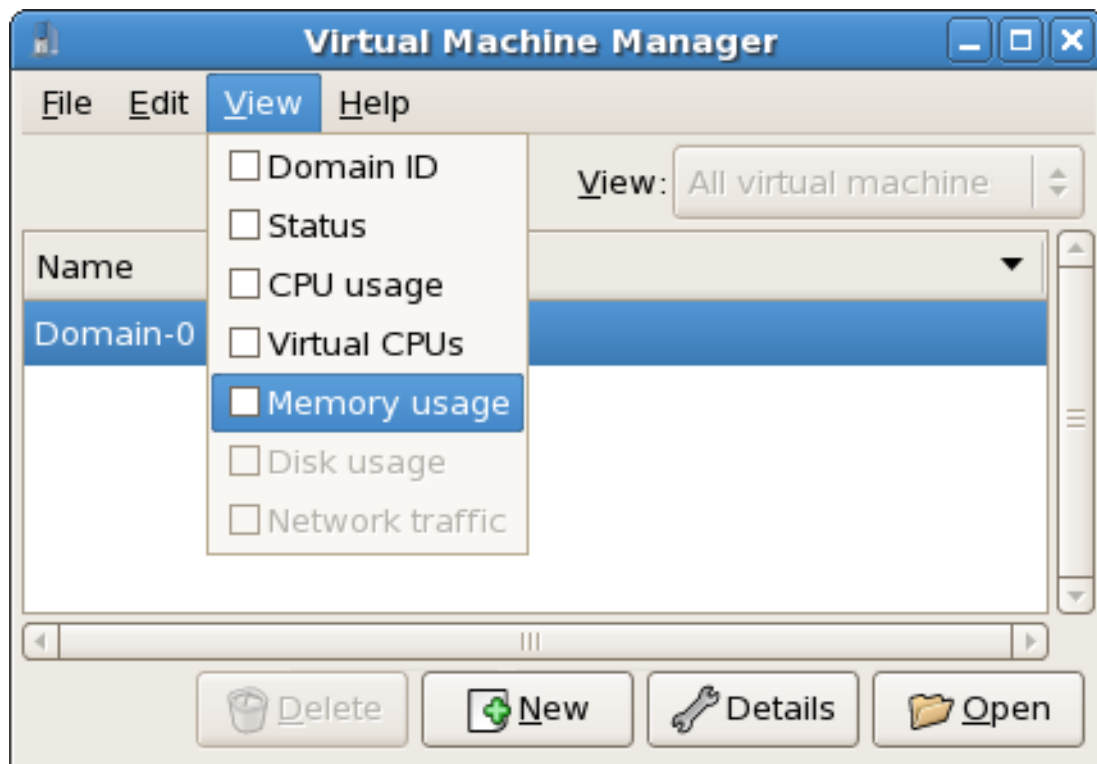


Figure 17.37. Affichage de l'utilisation mémoire

2. Le gestionnaire de machines virtuelles liste le pourcentage de mémoire en cours d'utilisation (en méga-octets) de toutes les machines virtuelles de votre système.

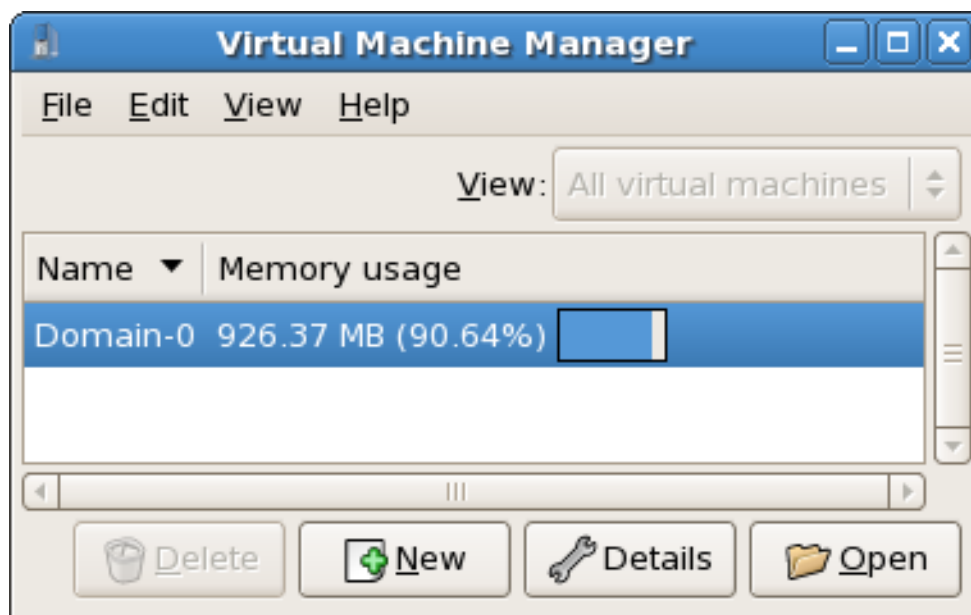


Figure 17.38. Affichage de l'utilisation mémoire

17.16. MANAGING A VIRTUAL NETWORK

To configure a virtual network on your system:

1. From the **Edit** menu, select **Host Details**.

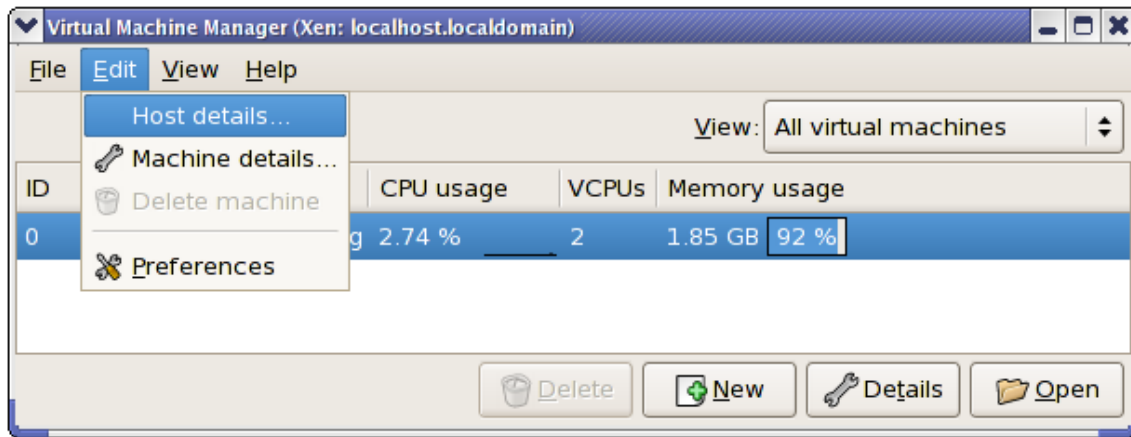


Figure 17.39. Selecting Host Details

2. This will open the **Host Details** menu. Click the **Virtual Networks** tab.

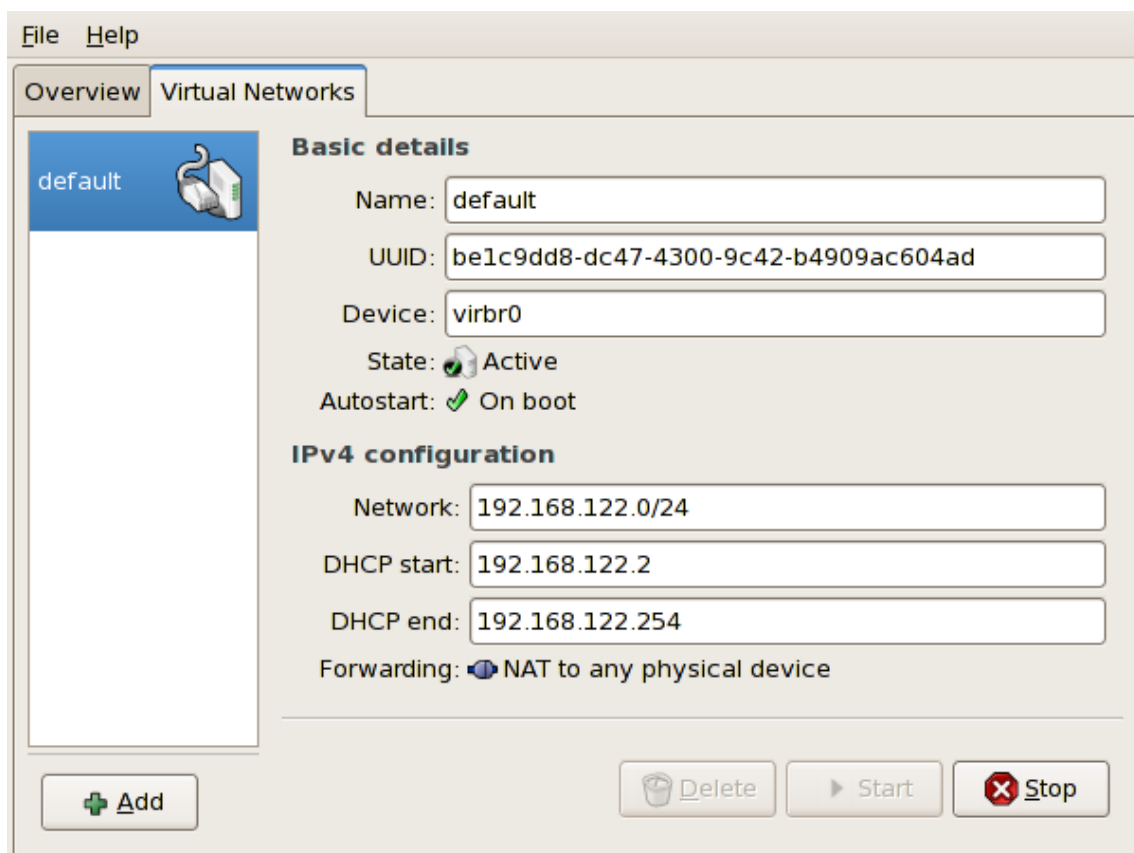


Figure 17.40. Virtual Network Configuration

3. All available virtual networks are listed on the left-hand box of the menu. You can edit the configuration of a virtual network by selecting it from this box and editing as you see fit.

17.17. CREATING A VIRTUAL NETWORK

To create a virtual network on your system:

1. Open the **Host Details** menu (refer to [Section 17.16, « Managing a Virtual Network »](#)) and click the **Add** button.

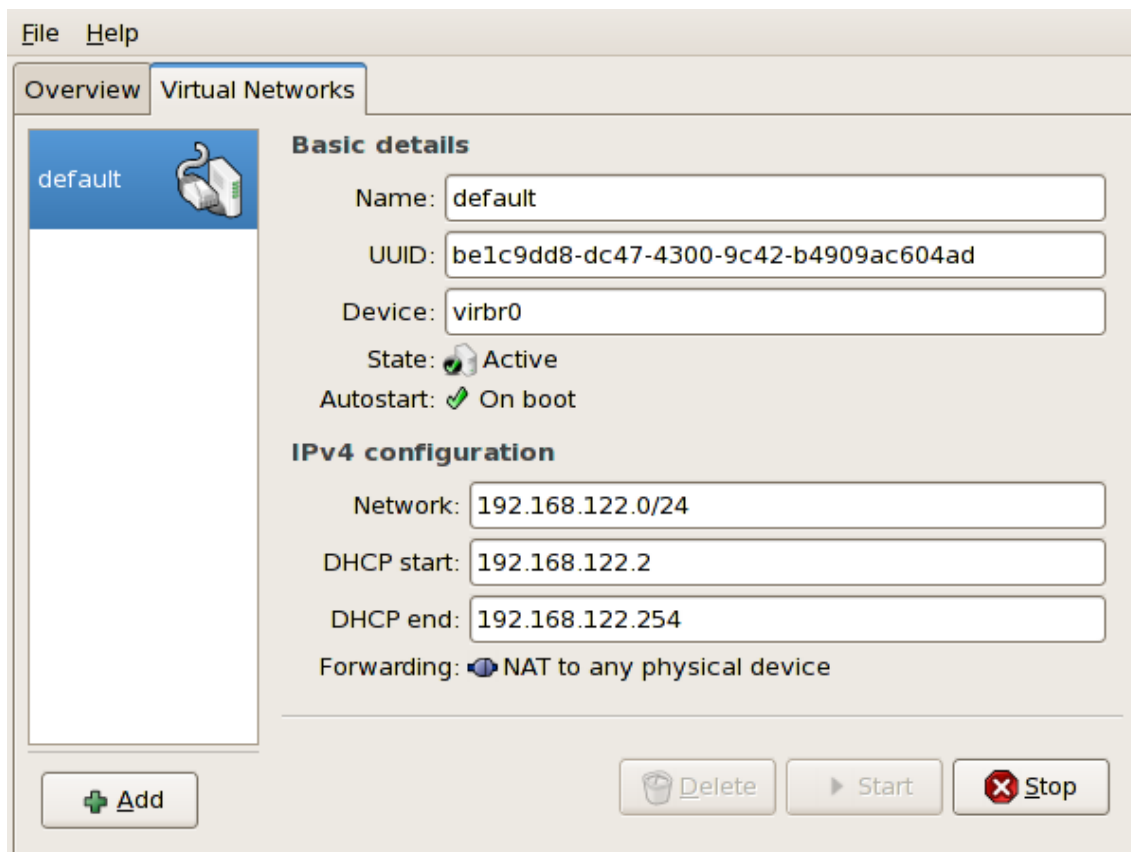


Figure 17.41. Virtual Network Configuration

This will open the **Create a new virtual network** menu. Click **Forward** to continue.

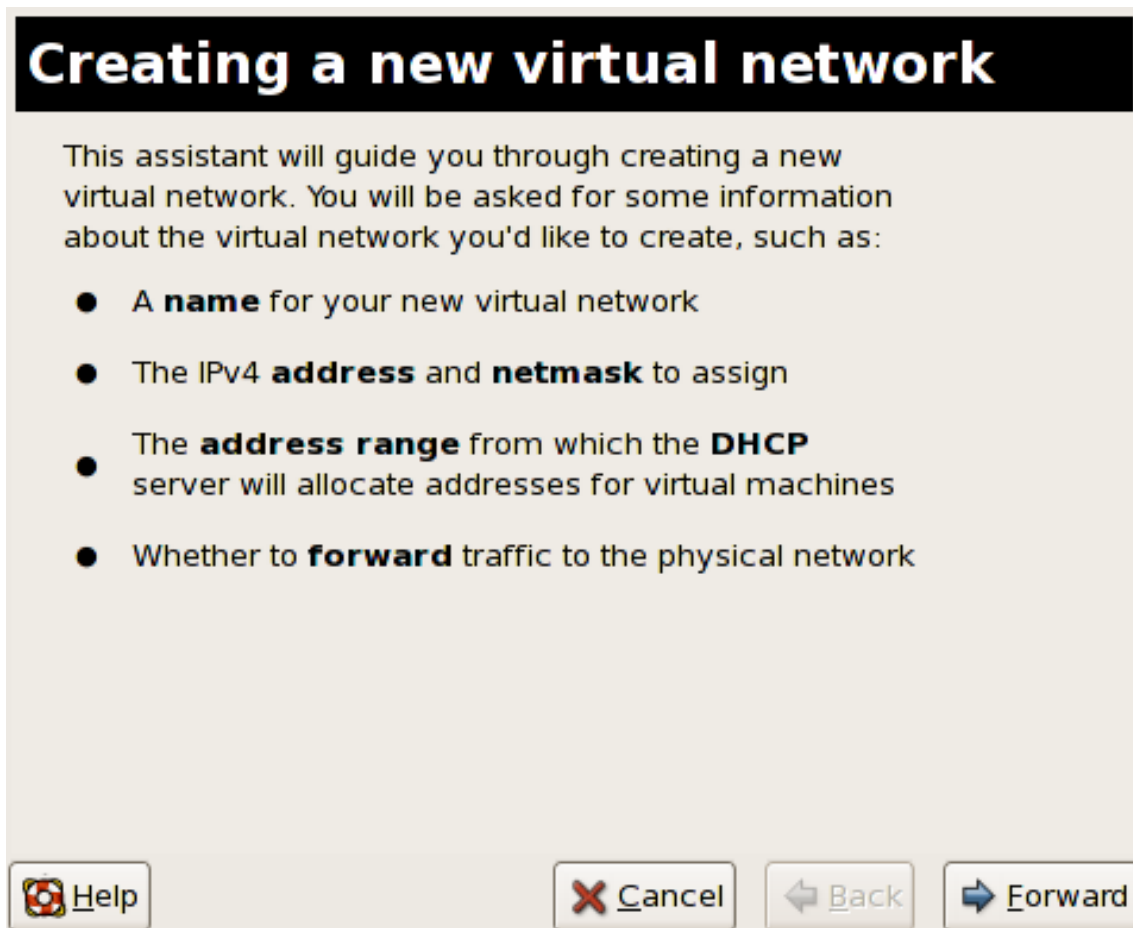


Figure 17.42. Creating a new virtual network

2. Enter an appropriate name for your virtual network and click **Forward**.

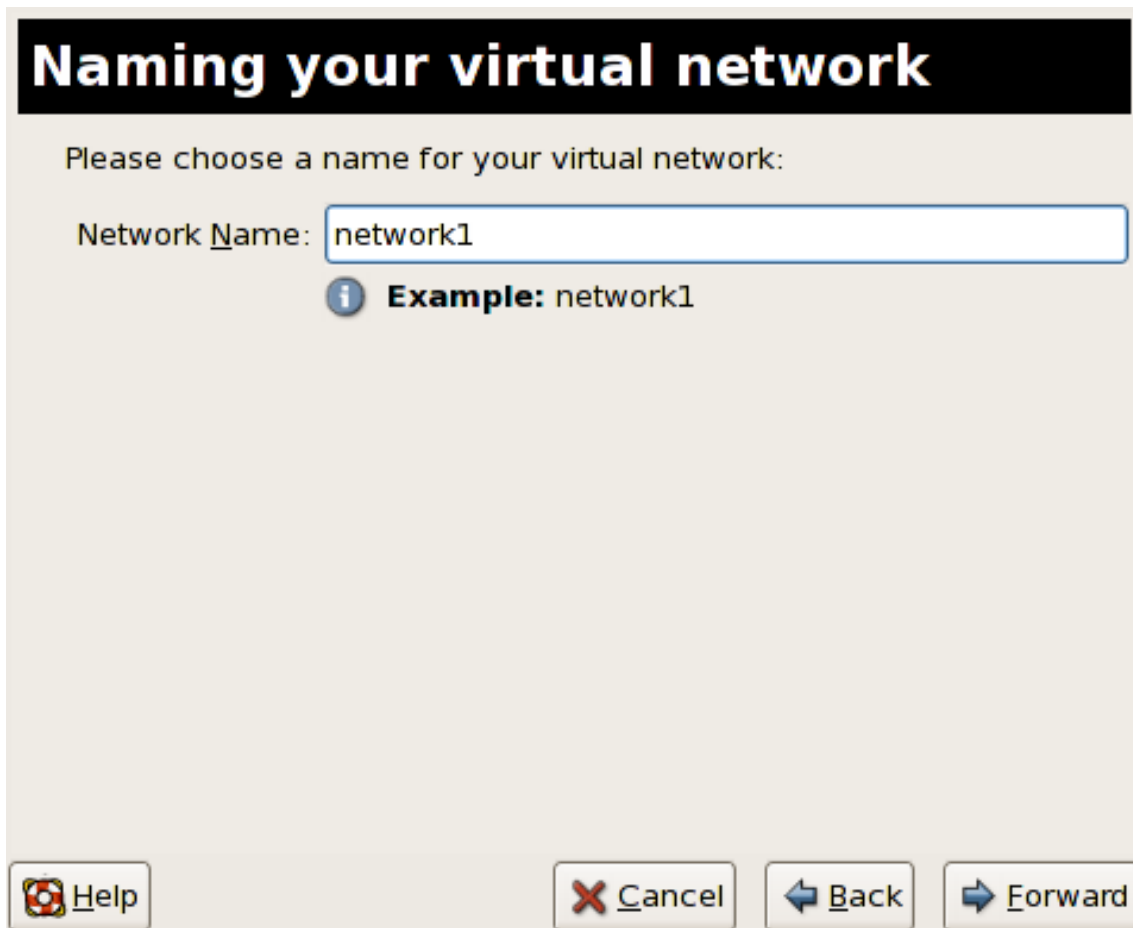


Figure 17.43. Naming your virtual network

3. Enter an IPv4 address space for your virtual network and click **Forward**.

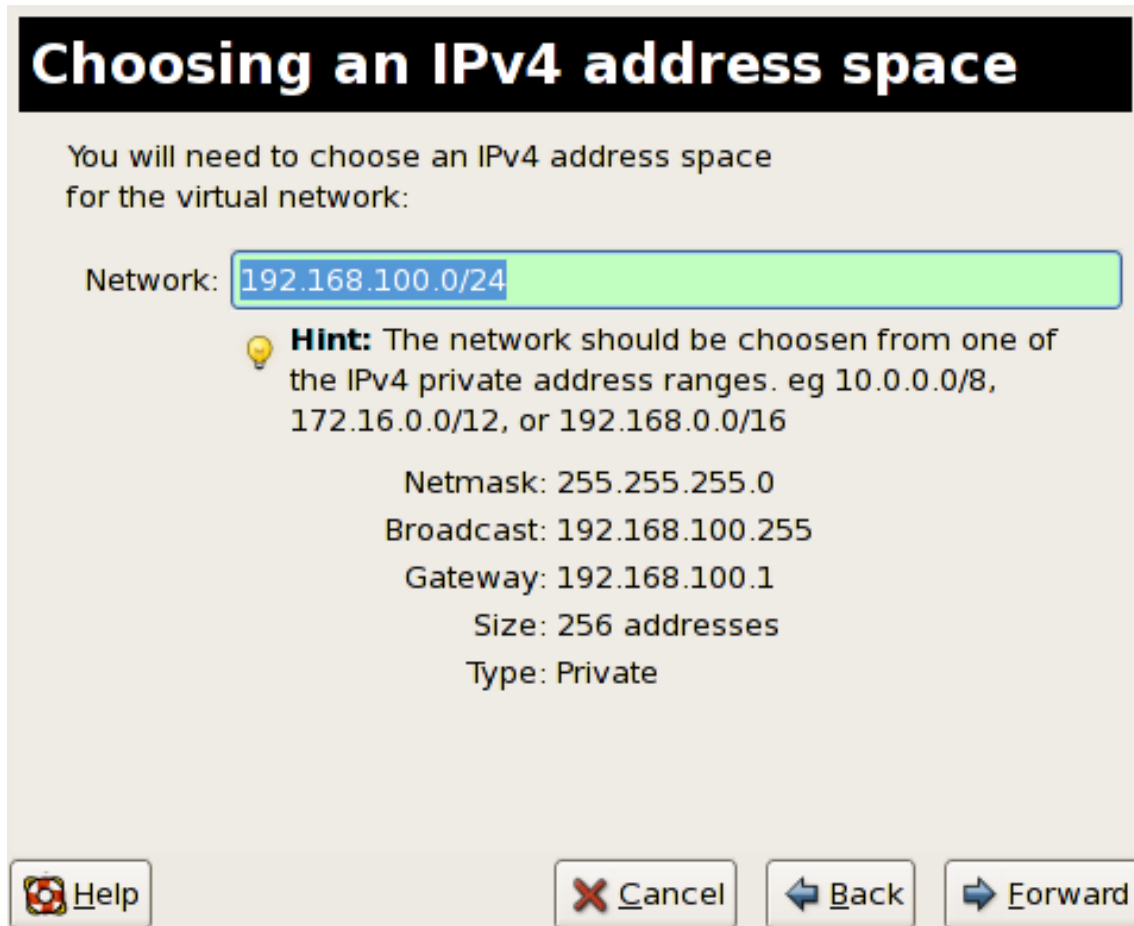


Figure 17.44. Choosing an IPv4 address space


4. Define the DHCP range for your virtual network by specifying a **Start** and **End** range of IP addresses. Click **Forward** to continue.

Selecting the DHCP range

Please choose the range of addresses the DHCP server can use to allocate to guests attached to the virtual network

Start:

End:

 **Tip:** Unless you wish to reserve some addresses to allow static network configuration in virtual machines, these parameters can be left with their default values.





 Help  Cancel  Back  Forward

Figure 17.45. Selecting the DHCP range

5. Select how the virtual network should connect to the physical network.

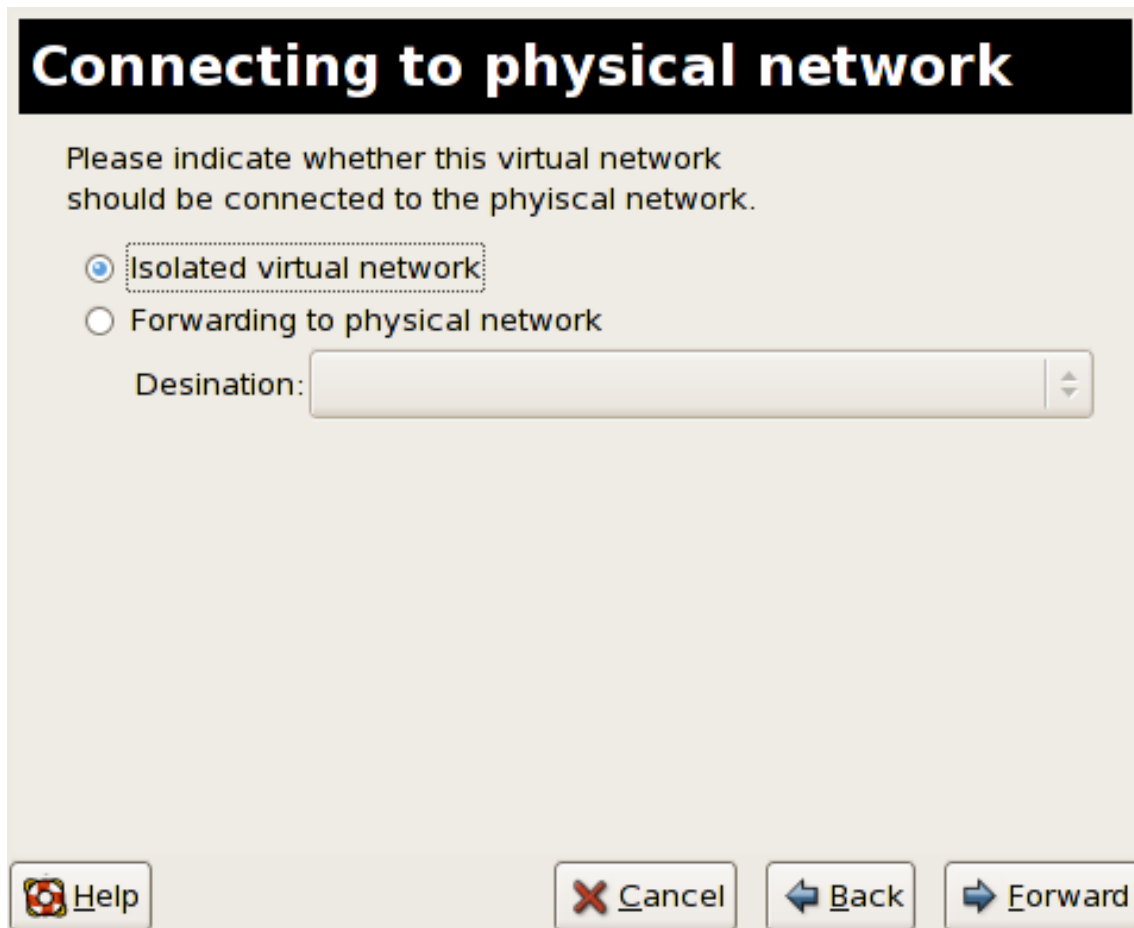


Figure 17.46. Connecting to physical network

If you select **Forwarding to physical network**, choose whether the **Destination** should be **NAT to any physical device** or **NAT to physical device eth0**.

Click **Forward** to continue.

6. You are now ready to create the network. Check the configuration of your network and click **Finish**.

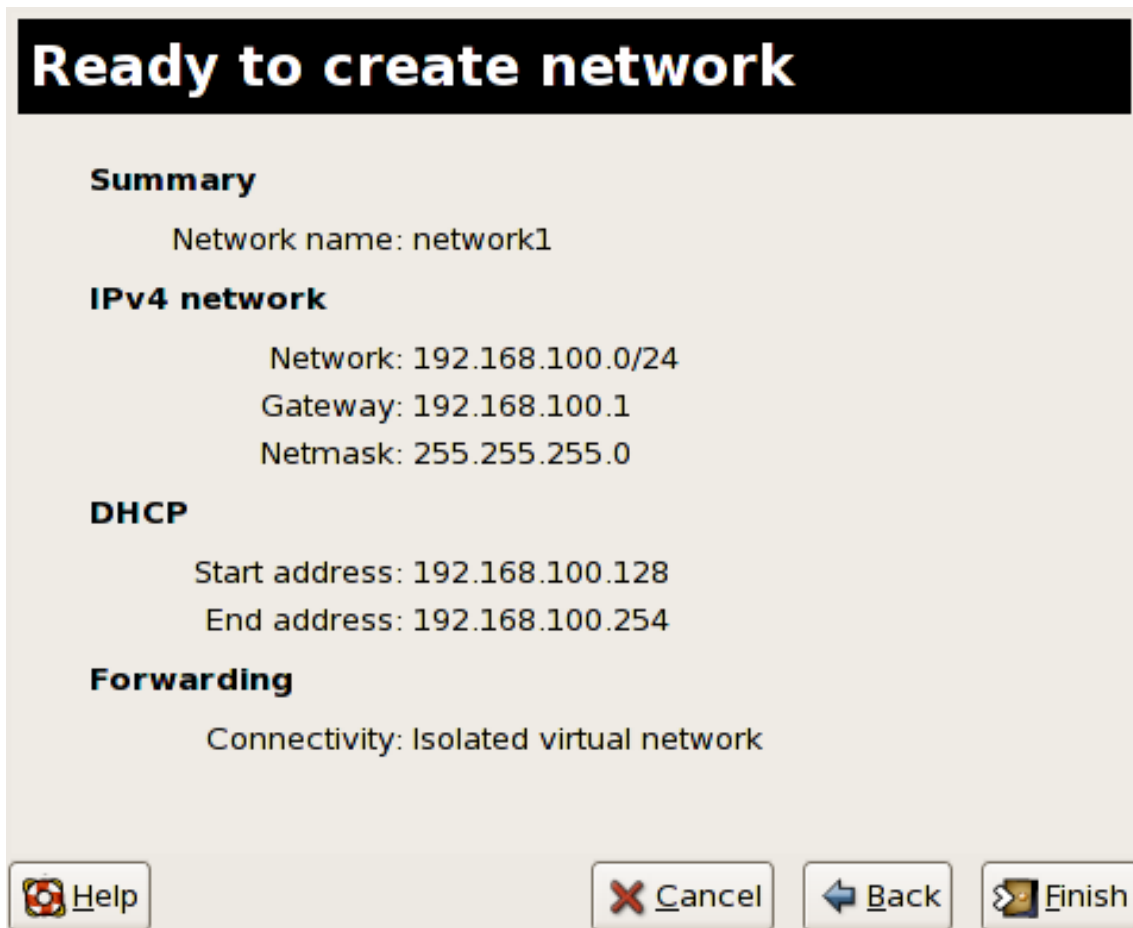


Figure 17.47. Ready to create network

7. The new virtual network is now available in the **Virtual Network** tab of the **Host Details** menu.

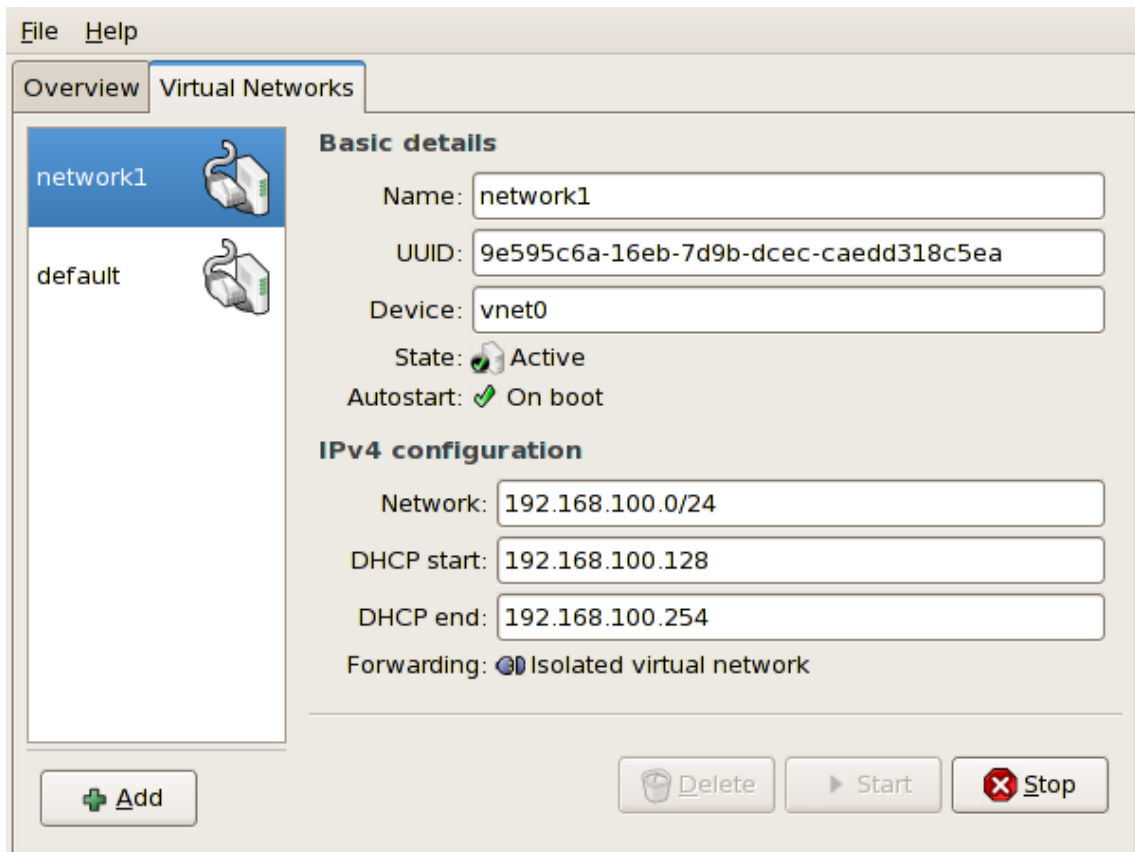


Figure 17.48. New virtual network is now available

CHAPITRE 18. RÉOLUTION DES PANNES DE LA VIRTUALISATION RED HAT

Cette section aborde des problèmes potentiels qui peuvent survenir lors de votre installation, de la gestion, et dans les opérations journalières de vos systèmes de Virtualisation Red Hat. Cette section de résolution des pannes couvre les messages d'erreur, les emplacements de fichier journaux, les outils de système, et les approches générales pour rechercher des données et analyser des problèmes.

18.1. VUE D'ENSEMBLE SUR LES FICHIERS JOURNAUX ET EMBLEMEMENTS

When deploying Red Hat Enterprise Linux 5.1 with Virtualization into your network infrastructure, the host's Virtualization software uses many specific directories for important configuration, log files, and other utilities. All the Red Hat Virtualization logs files are standard ASCII files, and easily accessible with any ASCII based editor:

- Le répertoire de configuration principale de la Virtualisation Red Hat est `/etc/xen/`. Ce répertoire contient le démon `xend` et autres fichiers de configuration de machines virtuelles. Les fichiers de script de mise en réseau résident également ici (dans le sous-répertoire `/scripts`).
- Tous les fichiers journaux que vous consulterez dans le but de résoudre des pannes résident dans le répertoire `/var/log/xen`.
- Il est utile de savoir que le répertoire par défaut pour toutes les images disque des fichiers basés sur les machines virtuelles résident dans le répertoire `/var/lib/xen`.
- Les informations sur la Virtualisation Red Hat pour le système de fichiers `/proc` résident dans le répertoire `/proc/xen/`.

18.2. DESCRIPTIONS DES FICHIERS JOURNAUX

La Virtualisation Red Hat a pour atouts le démon `xend` et le processus `qemu-dm`, deux utilitaires qui écrivent de multiples fichiers journaux dans le répertoire `/var/log/xen/` :

- `xend.log` est le fichier journal qui contient les données collectées par le démon `xend`, que ce soit un évènement système ordinaire, ou une action initiée par l'opérateur. Toutes les opérations des machines virtuelles comme créer, fermer, supprimer etc. apparaissent à cet endroit. Le `xend.log` est généralement le premier endroit où vous pouvez tracer les problèmes d'évènement ou de performance. Il contient des entrées détaillées et les conditions créant des messages d'erreur.
- `xend-debug.log` est le fichier qui contient des données d'erreurs dans les évènements à partir de `xend` et les sous-systèmes de virtualisation (comme la mémoire tampon, les scripts Python, etc.).
- `xen-hotplug-log` est le fichier journal contenant les données des évènements "hotplug". Si un périphérique ou un script réseau ne vient pas en ligne, l'évènement apparaît ici.
- `qemu-dm.[PID].log` est le fichier journal créé par le processus `qemu-dm` pour chaque invité pleinement virtualisé. Quand vous utilisez ce fichier journal, vous devez récupérer le processus PID `qemu-dm` en utilisant la commande `ps` pour examiner les arguments du processus afin

d'isoler le processus **qemu-dm** sur la machine virtuelle. Notez que vous devez remplacer le symbole [PID] par le processus PID **qemu-dm**.

Si des erreurs surviennent avec le gestionnaire de machines virtuelles (Virtual Machine Manager), vous pouvez revoir les données générées dans le fichier **virt-manager.log** qui réside dans le répertoire **/var/lib/virt-manager**. Notez que chaque fois que vous démarrez le gestionnaire de machines virtuelles, il surcharge le contenu des fichiers journaux existants. Assurez-vous de sauvegarder le fichier **virt-manager.log** avant de redémarrer le gestionnaire de machines virtuelles après une erreur système.

18.3. EMBLEMES IMPORTANTS DANS LES RÉPERTOIRES

Il existe des utilitaires et des fichiers journaux supplémentaires à connaître quand vous tracez les erreurs et problèmes à résoudre dans les environnements de Virtualisation Red Hat :

- Les images de machines virtuelles résident dans le répertoire **/var/lib/xen/images**.
- Quand vous redémarrez le démon **xend**, il met à jour la **xend-database** qui réside dans le répertoire **/var/lib/xen/xend-db**.
- Les vidages de la machine virtuelle (que vous exécutez avec la commande **xm dump-core**) résident dans le répertoire **/var/lib/xen/dumps**.
- Le répertoire **/etc/xen** contient les fichiers de configuration que vous utilisez pour gérer les ressources système. Le fichier de configuration du démon **xend** est appelé **xend-config.sxp** et vous pouvez utiliser ce fichier pour implémenter les modifications sur tout le système et configurer les callouts de mise en réseau.
- Les commandes **proc** sont une autre ressource qui vous permet de rassembler des informations système. Ces entrées **proc** résident dans le répertoire **/proc/xen** :

/proc/xen/capabilities

/proc/xen/balloon

/proc/xen/xenbus/

18.4. OUTILS DE RÉOLUTION DES PANNES

Cette section résume les applications de l'administrateur système, les utilitaires de mise en réseau, et les outils de débogage avancés, (pour plus d'informations sur l'utilisation de ces outils pour configurer les services de Virtualisation Red Hat, veuillez consulter la documentation de configuration qui s'y rapporte). Vous pouvez employer ces mêmes outils standards de l'administrateur de système et les fichiers journaux pour vous assister dans la résolution des pannes.

- **xentop**
- **xm dmesg**
- **xm log**
- **vmstat**
- **iostat**

- **lsof**

Utilisez les outils de débogage avancés et les fichiers journaux pour vous assister dans la résolution des pannes :

- **Xen0profile**
- **systemTap**
- **crash**
- **sysrq**
- **sysrq t**
- **sysrq w**

Vous pouvez employer ces outils de mise en réseau pour vous assister dans la résolution des pannes :

- **ifconfig**
- **tcpdump**
- **brctl**

brctl est un outil de mise en réseau qui inspecte et configure la configuration de pont ethernet dans le noyau de Virtualisation Linux. Vous devez posséder un accès super-utilisateur avant d'exécuter ces exemples de commandes :

```
# brctl show

bridge-name      bridge-id          STP  enabled  interfaces
-----
-----
xenbr0           8000.fefffffff    no   no       vif13.0
xenbr1           8000.ffffeffff    yes  no       pddummy0
xenbr2           8000.fffffefef    no   no       vif0.0

# brctl showmacs xenbr0

port-no          mac-addr           local?  ageing timer
1                fe:ff:ff:ff:ff:   yes     0.00
2                fe:ff:ff:fe:ff:   yes     0.00

# brctl showstp xenbr0

xenbr0

bridge-id        8000.fefffffff
designated-root   8000.fefffffff
root-port        0                  path-cost        0
```

max-age	20.00	bridge-max-age	20.00
hello-time	2.00	bridge-hello-time	2.00
forward-delay	0.00	bridge-forward-delay	0.00
ageing-time	300.01		
hello-timer	1.43	tcn-timer	0.00
topology-change-timer	0.00	gc-timer	0.02

18.5. RÉOLUTION DES PANNES AVEC LES FICHIERS JOURNAUX

When encountering issues with installing Red Hat Virtualization, you can refer to the host system's two logs to assist with troubleshooting. The **xend.log** file contains the same basic information as when you run the **xm log** command. It resides in the **/var/log/** directory. Here is an example log entry for when you create a domain running a kernel:

```
[2006-12-27 02:23:02 xend] ERROR (SrvBase: 163) op=create: Error creating
domain: (0, 'Error')
Traceback (most recent call list)
File "/usr/lib/python2.4/site-packages/xen/xend/server/SrvBase.py" line 107
in perform val = op_method (op, req)
File
"/usr/lib/python2.4/site-packages/xen/xend/server/SrvDomainDir.py line 71
in op_create
raise XendError ("Error creating domain: " + str(ex))
XendError: Error creating domain: (0, 'Error')
```

L'autre fichier journal **xend-debug.log**, est très utile aux administrateurs système puisqu'il contient des informations encore plus détaillées que le **xend.log**. Voici les mêmes données d'erreurs que pour le même problème de création de domaine de noyau :

```
ERROR: Will only load images built for Xen v3.0
ERROR: Actually saw: GUEST_OS=netbsd, GUEST_VER=2.0, XEN_VER=2.0;
LOADER=generic, BSD_SYMTAB'
ERROR: Error constructing guest OS
```

Quand vous appelez l'assistance clientèle, incluez toujours une copie de ces deux fichiers journaux.

18.6. RÉOLUTION DES PANNES AVEC LA CONSOLE SÉRIE

La console série est utile dans la résolution des pannes difficiles. Si le noyau de virtualisation plante et l'hyperviseur génère une erreur, il n'y a aucun moyen de tracer l'erreur sur votre hôte local. Cependant la console série vous permet de la capturer avec un hôte distant. Vous devez configurer l'hôte Xen pour envoyer des données à la console série. Ensuite vous devez configurer l'hôte distant pour capturer les données. Pour ce faire, vous devez modifier ces options dans le fichier **grub.conf** pour activer une console série 38400-bps sur com1 **/dev/ttyS0** :

```
title Red Hat Enterprise Linux (2.6.18-8.2080_RHEL5xen0)
  root (hd0,2)
  kernel /xen.gz-2.6.18-8.el5 com1=38400,8n1
```

```

module /vmlinuz-2.618-8.el5xen ro root=LABEL=/rhgb quiet console=xvc
console=tty xencons=xvc
module /initrd-2.6.18-8.el5xen.img

```

The **sync_console** can help determine a problem that causes hangs with asynchronous hypervisor console output, and the "**npacpi=off**" works around a problem that breaks input on the serial console. The parameters "**console=ttyS0**" and "**console=tty**" means that kernel errors get logged with on both the normal VGA console and on the serial console. Then you can install and set up **ttwatch** to capture the data on a remote host connected by a standard null-modem cable. For example, on the remote host you could type:

```
ttwatch --name myhost --port /dev/ttyS0
```

Cela canalise la sortie depuis **/dev/ttyS0** vers le fichier **/var/log/ttwatch/myhost.log**.

18.7. ACCÈS À LA CONSOLE DE L'INVITÉ PARAVIRTUALISÉ

Les systèmes d'exploitation invités paravirtualisés possèdent automatiquement une console texte virtuelle, configurée pour envoyer des données dans le système d'exploitation Domain0. Vous pouvez effectuer cela à partir de la ligne de commande en saisissant :

```
xm console [domain name or number]
```

Où **domain100** représente un nom ou un nombre en exécution. Vous pouvez également utiliser le gestionnaire de machines virtuelles pour afficher la console texte virtuelle. Dans la fenêtre des informations sur la machine virtuelle, sélectionnez **Console série** à partir du menu **Affichage**.

18.8. ACCÈS À LA CONSOLE DE L'INVITÉ PLEINEMENT VIRTUALISÉ

Full Virtualized guest operating systems automatically has a text console configured for use, but the difference is the kernel guest is not configured. To enable the guest virtual serial console to work with the Full Virtualized guest, you must modify the guest's **grub.conf** file, and include the '**console =ttyS0 console=tty0**' parameter. This ensures that the kernel messages are sent to the virtual serial console (and the normal graphical console). If you plan to use the virtual serial console in a full virtualized guest, you must edit the configuration file in the **/etc/xen/** directory. On the host domain, you can then access the text console by typing:

```
xm console
```

Vous pouvez également utiliser le gestionnaire de machines virtuelles pour afficher la console série. Dans la fenêtre des informations sur la machine virtuelle, sélectionnez la console série du menu Affichage.

18.9. IMPLEMENTATION DE LA PERSISTANCE LUN

Si votre système n'utilise pas "multipath", vous pouvez utiliser **udev** pour implémenter la persistance lun. Avant d'implémenter la persistance lun sur votre système, assurez-vous d'obtenir les UUID corrects. Une fois que vous les avez obtenus, vous pouvez configurer la persistance lun en éditant le fichier **scsi_id** qui réside dans le répertoire **/etc**. Une fois que ce fichier est ouvert dans l'éditeur de texte, vous devez décommenter cette ligne.

```
# options=-b
```

Puis remplacez-la par ce paramètre :

```
# options=-g
```

Cela indique à udev de contrôler tous les périphériques système SCSI pour retourner les UUID. Pour déterminer les UUID système, saisissez :

```
# scsi_id -g -s /block/sdc
```

La sortie devrait ressembler à ce qui suit :

```
[root@devices] # scsi_id -g -s /block/sdc
*3600a0b80001327510000015427b625e*
```

La longue chaîne de caractères est l'UUID. Pour faire en sorte que les noms de périphériques suppriment l'UUID, vérifiez chaque chemin de périphérique pour assurer que le numéro du UUID est le même pour chaque périphérique. Les UUID ne changent pas quand vous ajoutez un nouveau périphérique à votre système. Une fois les chemins de périphériques vérifiés, vous devez créer des règles pour le nommage de périphériques. Pour créer ces règles, modifiez le fichier **20-names.rules** qui réside dans le répertoire **/etc/udev/rules.d**. Les règles de nommage de périphérique que vous créez ici devraient suivre ce format :

```
# KERNEL="sd*", BUS="scsi", PROGRAM="sbin/scsi_id", RESULT="UUID",
NAME="devicename"
```

Remplacez votre UUID existant et le nom du périphérique par l'entrée récupérée du UUID ci-dessus. Ainsi la règle devrait ressembler à ce qui suit :

```
KERNEL="sd*", BUS="scsi", PROGRAM="sbin/scsi_id",
RESULT="3600a0b80001327510000015427b625e
", NAME="mydevicename"
```

Cela fait en sorte que le système active tous les périphériques qui correspondent à **/dev/sd*** pour inspecter un UUID donné. Quand il trouve un périphérique correspondant, il crée un noeud de périphérique appelé **/dev/devicename**. Pour cet exemple, le noeud de périphérique est **/dev/mydevice**. Enfin, vous devez ajouter le fichier **rc.local** qui réside dans le répertoire **/etc** avec ce chemin :

```
/sbin/start_udev
```

IMPLEMENTATION DE LA PERSISTANCE LUN AVEC MULTIPATH

Pour implémenter la persistance lun dans un environnement multipath, vous devez définir les noms d'alias pour les périphériques multipath. Pour cet exemple, vous devez définir quatre alias de périphérique en éditant le fichier **multipath.conf** qui réside dans le répertoire **/etc/** :

```
multipath {
    wwid          3600a0b80001327510000015427b625e
    alias         oramp1
}
multipath {
    wwid          3600a0b80001327510000015427b6
```

```

    alias      oramp2
}
multipath {
    wwid      3600a0b80001327510000015427b625e
    alias     oramp3
}
multipath {
    wwid      3600a0b80001327510000015427b625e
    alias     oramp4
}

```

Cela définit 4 lun : **/dev/mpath/oramp1**, **/dev/mpath/oramp2**, **/dev/mpath/oramp3**, et **/dev/mpath/oramp4**. Les périphériques résideront dans le répertoire **/dev/mpath**. Ces noms lun sont persistants sur les redémarrages car cela crée les noms d'alias sur le wwid des lun.

18.10. CONSIDÉRATIONS AVEC SELINUX

Cette section contient des informations que vous devez considérer quand vous implémentez SELinux dans votre environnement de Virtualisation Red Hat. Quand vous déployez des modifications de système ou ajouter des périphériques, vous devez mettre à jour votre politique SELinux en fonction de ces modifications. Pour configurer un volume LVM pour un invité, vous devez modifier le contexte SELinux pour le périphérique bloc sous-jacent et le groupe de volume respectivement.

```

# semanage fcontext -a -t xen_image_t -f -b /dev/sda2
# restorecon /dev/sda2

```

Le paramètre booléen **xend_disable_trans** place **xend** dans un mode unconfined (sans limitation) après le redémarrage du démon. Il est recommandé de désactiver la protection pour un démon unique plutôt que pour tout le système. De même il vaut mieux ne pas ré-étiqueter les répertoires comme **xen_image_t** que vous utiliserez ailleurs.

18.11. ACCÈS AUX DONNÉES SUR L'IMAGE DISQUE DE L'INVITÉ

Vous pouvez utiliser deux applications différentes qui vous assistent dans l'accès aux données à partir de l'intérieur de l'image disque d'un invité. Avant d'utiliser ces outils, vous devez arrêter les invités. Accéder au système de fichiers à partir de l'invité et de dom0 pourrait potentiellement endommager le système.

Vous pouvez utiliser l'application **kpartx** pour gérer les disques partitionnés ou les groupes de volume LVM :

```

yum install kpartx
kpartx -av /dev/xen/guest1
add map guest1p1 : 0 208782 linear /dev/xen/guest1 63
add map guest1p2: 0 16563015 linear /dev/xen/guest1 208845

```

Pour accéder aux volumes sur une seconde partition, vous devez rescanner LVM avec **vgscan** et activer le groupe de volume sur la partition (appelée **VolGroup00** par défaut) en utilisant la commande **vgchange -ay** :

```

# kpartx -a /dev/xen/guest1
#vgscan
Reading all physical volumes . This may take a while...
Found volume group "VolGroup00" using metadata type lvm2

```

```
# vgchange -ay VolGroup00
2 logical volume(s) in volume group VolGroup00 now active.
# lvs
LV VG Attr Lsize Origin Snap% Move Log Copy%
LogVol00 VolGroup00 -wi-a- 5.06G
LogVol01 VolGroup00 -wi-a- 800.00M
# mount /dev/VolGroup00/LogVol00 /mnt/
....
#umount /mnt/
#vgchange -an VolGroup00
#kpartx -d /dev/xen/guest1
```

Rappelez-vous de désactiver les volumes logiques avec **vgchange -an**, supprimer les partitions avec **kpartx -d** et supprimer le périphérique loop avec **losetup -d** quand vous terminez.

18.12. SITUATIONS FRÉQUENTES DANS LA RÉOLUTION DES PANNES

Quand vous tentez de démarrer le service **xend**, il ne se passe rien. Vous saisissez **xm list1** et vous recevez ce qui suit :

```
Error: Error connecting to xend: Connection refused. Is xend running?
```

Vous essayez d'exécuter **xend** manuellement et vous obtenez des erreurs supplémentaires :

```
Error: Could not obtain handle on privileged command interfaces (2 = No
such file or directory)
Traceback (most recent call last:)

File "/usr/sbin/xend/", line 33 in ?

from xen.xend.server import SrvDaemon

File "/usr/lib/python2.4/site-packages/xen/xend/server/SrvDaemon.py" ,
line 26 in ?

from xen.xend import XendDomain

File "/usr//lib/python2.4/site-packages/xen/xend/XendDomain.py" , line 33,
in ?

from xen.xend import XendDomainInfo

File "/usr/lib/python2.4/site-packages/xen/xend/image.py" , line37, in ?

import images

File "/usr/lib/python2.4/site-packages/xen/xend/image.py" , line30, in ?

xc = xen.lowlevel.xc.xc ()

RuntimeError: (2, 'No such file or directory' )
```

Il est probable que vous ayez relancé votre hôte dans un noyau qui n'est pas un **xen-hypervisor kernel**. Pour corriger cela, vous devez sélectionner le noyau **xen-hypervisor** au moment du lancement (ou configurer par défaut, le noyau **xen-hypervisor** dans votre fichier **grub.conf**).

18.13. ERREURS DE PÉRIPHÉRIQUES LOOP

Si vous utilisez les images d'invité basées sur les fichiers, il est possible que vous ayez augmenté le nombre de périphériques loop (par défaut, jusqu'à 8 périphériques loop activés). Si vous avez besoin de plus de 8 périphériques loop/invités basés sur des fichiers, vous devez modifier le fichier **/etc/modprobe.conf**. Quand vous modifiez ce fichier **modprobe.conf**, vous devez inclure cette ligne :

```
options loop max_loop=64
```

Cet exemple utilise la valeur 64 mais vous pouvez spécifier un autre nombre pour configurer la valeur loop maximum. Vous aurez peut-être aussi à implémenter des invités supportés par un périphérique loop sur votre système. Pour employer un invité supporté par un périphérique loop pour un système paravirtuel, utilisez les commandes **phy: block device** ou **tap:aio**. Pour employer des invités supportés par un périphérique loop pour un système totalement virtualisé, utilisez le fichier **phy: device** ou les commandes **file: file**.

18.14. ERREURS DE CRÉATION D'INVITÉS

When you attempt to create a guest, you receive an **"Invalid argument"** error message. This usually means that the kernel image you are trying to boot is incompatible with the hypervisor. An example of this would be if you were attempting to run a non-PAE FC5 kernel on a PAE only FC6 hypervisor.

Vous faites une mise à jour yum et recevez un nouveau noyau, le noyau par défaut **grub.conf** redevient directement un noyau bare-metal au lieu d'un noyau de virtualisation.

Pour remédier à ce problème vous devez modifier le noyau RPM par défaut qui réside dans le répertoire **/etc/sysconfig/kernel/**. Vous devez vous assurer que le paramètre **kernel-xen** est configuré comme l'option par défaut dans votre fichier **gb.conf**.

18.15. ERREURS DE LA CONSOLE SÉRIE

Vous ne recevez pas de sortie sur la console série. Pour résoudre ce problème, vous devez éditer le fichier **grub.conf** et modifier les paramètres de port com à :

```
serial --unit=1 --speed=115200

title RHEL5 i386 Xen (2.6.18-1.2910.el5xen)
root (hd0, 8)
kernel /boot/xen.gz-2.6.18-1.2910.el5 com2=115200,8n1
module /boot/vmlinuz-2.6.18-1.2910.el5xen to root=LABEL=RHEL5_i386
console=tty console=ttyS1115200
module /boot/initrd-2.8.6.18-12910.el5xen.img

title RHEL5 i386 xen (2.6.18.-1.2910.el5xen)
root (hd0, 8)
kernel /boot/xen.gz-2.6.18-1.2910.el5 com2=115200 console=com21
```

```
module /boot/vmlinuz2.6.18-1.2910.el5xen to root=LABEL=RHEL5_i386
console=xvc xencons=xvc
module /boot/ititrd-2.6.18-1.2910.el5xen.img
```

Ces modifications sur le fichier **grub.conf** devraient activer votre console série pour qu'elle fonctionne correctement. Vous devriez être en mesure d'utiliser tout nombre pour le **ttyS** et il devrait fonctionner comme **ttyS0**.

18.16. ERREURS DE PONT RÉSEAU

Red Hat Virtualization can configure multiple Virtualization network bridges to use with multiple ethernet cards. To successfully configure multiple network bridges for ethernet cards, you must configure the second network interface by either using the system-config-network TUI/GUI, or by creating a new configuration file in **/etc/sysconfig/network-scripts** . You should use a process to setup multiple Xen bridges. This is an example config file for a second NIC called '**eth1**' :

```
#/etc/sysconfig/network-scripts/fcfg-eth1
DEVICE=eth1
BOOTPROTO=static
ONBOOT=yes
USERCTL=no
IPV6INIT=no
PEERDNS=yes
TYPE=Ethernet
NETMASK=255.255.255.0
IPADDR=10.1.1.1
GATEWAY=10.1.1.254
ARP=yes
```

Copiez le fichier **/etc/xen/scripts/network-bridge** sur **/etc/xen/scripts/network-bridge.xen**.

Edit **/etc/xen/xend-config.sxp** and add a line to your new network bridge script (this example uses "**network-virtualization-multi-bridge**").

Dans le fichier **xend-config.sxp**, la nouvelle ligne devrait refléter votre nouveau script :

```
network-script network-xen-multi-bridge
```

Assurez-vous d'enlever le commentaire sur cette ligne :

```
network-script network-bridge
```

Si vous désirez créer de multiples ponts Xen, vous devez créer un script personnalisé. L'exemple ci-dessous crée deux ponts Xen (appelés **xenbr0** et **xenbr1**) et les attache à **eth1** et **eth0**, respectivement :

```
# !/bin/sh
# network-xen-multi-bridge
# Exit if anything goes wrong
set -e
# First arg is operation.
OP=$1
```

```

shift
script=/etc/xen/scripts/network-bridge.xen
case ${OP} in
start)
$script start vifnum=1 bridge=xenbr1 netdev=eth1
$script start vifnum=0 bridge=xenbr0 netdev=eth0
;;
stop)
$script stop vifnum=1 bridge=xenbr1 netdev=eth1
$script stop vifnum=0 bridge=xenbr0 netdev=eth0
;;
status)
$script status vifnum=1 bridge=xenbr1 netdev=eth1
$script status vifnum=0 bridge=xenbr0 netdev=eth0
;;
*)
echo 'Unknown command: ' ${OP}
echo 'Valid commands are: start, stop, status'
exit 1
esac

```

Si vous désirez créer des ponts supplémentaires, utilisez simplement le script d'exemple et copier/coller le fichier de façon appropriée.

18.17. CONFIGURATIONS DES ORDINATEURS PORTABLES

The task of configuring your RHEL 5.1 loaded laptop for use on a network environment, presents a number of potential challenges. Most WiFi and wired connections switch constantly during any given day, and Red Hat Virtualization assumes it has access to the same interface consistently. This results in the system performing ifup/ifdown calls to the network interface in use by Red Hat Virtualization. WiFi cards are not the ideal network connection method since Red Hat Virtualization uses the default network interface.

The idea here is to create a 'dummy' network interface for Red Hat Virtualization to use.

This technique allows you to use a hidden IP address space for your guests and Virtual Machines. To do this operation successfully, you must use static IP addresses as DHCP does not listen for IP addresses on the dummy network. You also must configure NAT/IP masquerading to enable network access for your guests and Virtual Machines. You should attach a static IP when you create the 'dummy' network interface.

Pour cet exemple l'interface est appelée dummy0 et l'IP utilisé est 10.1.1. Le script est appelé **ifcfg-dummy0** et réside dans le répertoire **/etc/sysconfig/network-scripts/** :

```

DEVICE=dummy0
BOOTPROTO=none
ONBOOT=yes
USERCTL=no
IPV6INIT=no
PEERDNS=yes
TYPE=Ethernet
NETMASK=255.255.255.0
IPADDR=10.1.1.1
ARP=yes

```

Vous devez lier **xenbr0** à **dummy0** pour permettre la connexion réseau même lorsque la connexion au réseau physique n'est pas établie.

You will need to make additional modifications to the **xend-config.sxp** file. You must locate the (**network-script 'network-bridge' bridge=xenbr0**) section and add include this in the end of the line:

```
netdev=dummy0
```

You must also make some modifications to your guest's domU networking configuration to enable the default gateway to point to dummy0. You must edit the DomU 'network' file that resides in the **/etc/sysconfig/** directory to reflect the example below:

```
NETWORKING=yes
HOSTNAME=localhost.localdomain
GATEWAY=10.1.1.1

IPADDR=10.1.1.10
NETMASK=255.255.255.0
```

Il est recommandé d'activer NAT dans domain0 afin que domU puisse accéder au réseau public. Ainsi, même les utilisateurs sans fil peuvent contourner les limitations de la Virtualisation Red Hat. Pour ce faire, vous devez modifier le fichier **S99XenLaptopNAT** qui réside dans le répertoire **/etc/rc3.d** pour refléter l'exemple ci-dessous :

```
#!/bin/bash
#
# XenLaptopNAT Startup script for Xen on Laptops
#
# chkconfig: - 99 01
# description: Start NAT for Xen Laptops
#
# PATH=/usr/bin:/sbin:/bin:/usr/sbin
# export PATH
GATEWAYDEV=`ip route | grep default | awk {'print $5'}`
iptables -F
case "$1" in
start)
if test -z "$GATEWAYDEV"; then
echo "No gateway device found"
else
echo "Masquerading using $GATEWAYDEV"
/sbin/iptables -t nat -A POSTROUTING -o $GATEWAYDEV -j MASQUERADE
fi
echo "Enabling IP forwarding"
echo 1 > /proc/sys/net/ipv4/ip_forward
echo "IP forwarding set to `cat /proc/sys/net/ipv4/ip_forward`"
echo "done."
;;
*)
echo "Usage: $0 {start|restart|status}"
;;
esac
```

Si vous désirez que le réseau soit automatiquement configuré au lancement, vous devez créer un lien symbolique vers `/etc/rc3.d/S99XenLaptopNAT`.

Quand vous modifiez le fichier `modprobe.conf`, vous devez inclure ces lignes :

```
alias dummy0 dummy
options dummy numdummies=1
```

18.18. STARTING DOMAINS AUTOMATICALLY DURING SYSTEM BOOT

Starting Domains Automatically During System Boot

Vous pouvez configurer vos invités pour qu'ils démarrent automatiquement quand vous lancer le système. Pour ce faire, vous devez modifier les liens symboliques qui résident dans `/etc/xen/auto`. Ce fichier pointe vers les fichiers de configuration des invités que vous devez démarrer automatiquement. Le processus de démarrage est sérialisé, c'est-à-dire que plus le nombre d'invités est élevé, plus le processus de lancement prendra de temps. Cet exemple vous montre comment utiliser les liens symboliques pour l'invité `rhe15vm01` :

```
[root@python xen]# cd /etc/xen
[root@python xen]# cd auto
[root@python auto]# ls
[root@python auto]# ln -s ../rhe15vm01 .
[root@python auto]# ls -l

lrwxrwxrwx 1 root root 14 Dec 14 10:02 rhe15vm01 -> ../rhe15vm01

[root@python auto]#
```

18.19. MODIFICATION DE DOMAIN0

To use Red Hat Virtualization to manage domain0, you will constantly making changes to the `grub.conf` configuration file, that resides in the `/etc` directory. Because of the large number of domains to manage, many system administrators prefer to use the 'cut and paste' method when editing `grub.conf`. If you do this, make sure that you include all five lines in the Virtualization entry (or this will create system errors). If you require Xen hypervisor specific values, you must add them to the 'xen' line. This example represents a correct `grub.conf` Virtualization entry:

```
# boot=/dev/sda/
default=0
timeout=15
#splashimage=(hd0, 0)/grub/splash.xpm.gz
hiddenmenu
serial --unit=0 --speed=115200 --word=8 --parity=no --stop=1
terminal --timeout=10 serial console
title Red Hat Enterprise Linux Server (2.6.17-1.2519.4.21. el5xen)
    root (hd0, 0)
    kernel /xen.gz-2.6.17-1.2519.4.21.el5 com1=115200,8n1
    module /vmlinuz-2.6.17-1.2519.4.21el5xen ro root=/dev/VolGroup00/LogVol100
    module /initrd-2.6.17-1.2519.4.21.el5xen.img
```

For example, if you need to change your dom0 hypervisor's memory to 256MB at boot time, you must edit the 'xen' line and append it with the correct entry, '**dom0_mem=256M**'. This example represents the respective **grub.conf** xen entry:

```
# boot=/dev/sda
default=0
timeout=15
#splashimage=(hd0,0)/grubs/splash.xpm.gz
hiddenmenu
serial --unit=0 --speed =115200 --word=8 --parity=no --stop=1
terminal --timeout=10 serial console
title Red Hat Enterprise Linux Server (2.6.17-1.2519.4.21. el5xen)
    root (hd0,0)
    kernel /xen.gz-2.6.17-1.2519.4.21.el5 com1=115200,8n1 dom0_mem=256MB
    module /vmlinuz-2.6.17-1.2519.4.21.el5xen ro
    root=/dev/VolGroup00/LogVol100
    module /initrd-2.6.17-1.2519.4.21.el5xen.img
```

18.20. FICHIERS DE CONFIGURATION DES INVITÉS

When you install new guests using virt-manager (or virt-install) tool(s) from Red Hat Enterprise Linux 5.1 with Virtualization, the guests configuration files (located in the /etc/xen directory) get modified and setup automatically. This configuration file example is for a para-virtualized guest:

```
name = "rhel5vm01"
memory = "2048"
disk = ['tap:aio:/xen/images/rhel5vm01.dsk,xvda,w',]
vif = ["type=ieomu, mac=00:16:3e:09:f0:12 bridge=xenbr0",
      "type=ieomu, mac=00:16:3e:09:f0:13 ]
vnc = 1
vncunused = 1
uuid = "302bd9ce-4f60-fc67-9e40-7a77d9b4e1ed"
bootloader = "/usr/bin/pygrub"
vcpus=2
on_reboot = "restart"
on_crash = "restart"
```

Note that the **serial="pty"** is the default for the configuration file. This configuration file example is for a fully-virtualized guest:

```
name = "rhel5u5-86_64"
builder = "hvm"
memory = 500
disk = ['file:/xen/images/rhel5u5-x86_64.dsk.hda,w']
vif = [ 'type=ioemu, mac=00:16:3e:09:f0:12, bridge=xenbr0', 'type=ieomu,
mac=00:16:3e:09:f0:13, bridge=xenbr1']
uuid = "b10372f9-91d7-ao5f-12ff-372100c99af5"
device_model = "/usr/lib64/xen/bin/qemu-dm"
kernel = "/usr/lib/xen/boot/hvmloder/"
vnc = 1
vncunused = 1
apic = 1
acpi = 1
pae = 1
```

```

vcpus =1
serial ="pty" # enable serial console
on_boot = 'restart'

```

18.21. CLONAGE DES FICHIERS DE CONFIGURATION DES INVITÉS

You can copy (or clone) an existing configuration file to create an all new guest. You must modify the name parameter of the guests' configuration file. The new, unique name then appears in the hypervisor and is viewable by the management utilities. You must generate an all new UUID as well (using the **uuidgen(1)** command). Then for the **vif** entries you must define a unique MAC address for each guest (if you are copying a guest configuration from an existing guest, you can create a script to handle it). For the xen bridge information, if you move an existing guest configuration file to a new host, you must update the **xenbr** entry to match your local networking configuration. For the Device entries, you must modify the entries in the '**disk=**' section to point to the correct guest image.

You must also modify these system configuration settings on your guest. You must modify the HOSTNAME entry of the **/etc/sysconfig/network** file to match the new guest's hostname.

Vous devez modifier l'adresse HWADDR du fichier **/etc/sysconfig/network-scripts/ifcfg-eth0** pour faire correspondre la sortie du fichier **ifconfig eth0** et si vous utilisez des adresses IP statiques, vous devez modifier l'entrée IPADDR.

18.22. CREATION D'UN SCRIPT POUR GÉNÉRER DES ADRESSES MAC

La Virtualisation de Red Hat peut générer une adresse MAC pour chaque machine virtuelle au moment de sa création. Étant donné qu'il y a une quantité presque illimitée de numéros sur le même sous-réseau, il est peu probable que vous obteniez la même adresse MAC. Pour contourner cela, vous pouvez aussi écrire un script pour générer une adresse MAC. Cet exemple de script contient les paramètres pour générer une adresse MAC :

```

#!/usr/bin/python
# macgen.py script generates a MAC address for Xen guests
#
import random
mac = [ 0x00, 0x16, 0x3e,
        random.randint(0x00, 0x7f),
        random.randint(0x00, 0xff),
        random.randint(0x00, 0xff) ]
print ':'.join(map(lambda x: "%02x" % x, mac))

```

Generates e.g.:

```
00:16:3e:66:f5:77
```

to stdout

18.23. CONFIGURATION DE LA MIGRATION EN DIRECT DE MACHINES VIRTUELLES

Red Hat Virtualization can migrate virtual machines between other servers running Red Hat Enterprise Linux 5.1 with Virtualization. Further, migration is performed in an offline method (using the **xm migrate** command). Live migration can be done from the same command. However there are some additional modifications that you must do to the **xend-config** configuration file. This example identifies the entries that you must modify to ensure a successful migration:

(xend-relocation-server yes)

The default for this parameter is 'no', which keeps the relocation/migration server deactivated (unless on a trusted network) and the domain virtual memory is exchanged in raw form without encryption.

(xend-relocation-port 8002)

Ce paramètre détermine le port utilisé par **xend** pour la migration. La valeur est correcte, veuillez simplement vous assurer que vous enlevez le commentaire qui est placé devant.

(xend-relocation-address)

Ce paramètre représente l'adresse qui écoute les connexions socket de délocalisation. À l'écoute, elle restreint la migration vers une interface particulière.

(xend-relocation-hosts-allow)

This parameter controls the host that communicates with the relocation port. If the value is empty, then all incoming connections are allowed. You must change this to a space-separated sequences of regular expressions (such as **xend-relocation-hosts-allow-'^localhost\\.localdomain\$'**). A host with a fully qualified domain name or IP address that matches these expressions are accepted.

Après la configuration de ces paramètres, vous devez redémarrer l'hôte pour que la virtualisation Red Hat accepte vos nouveaux paramètres.

18.24. INTERPRÉTATION DES MESSAGES D'ERREUR

Vous recevez le message d'erreur suivant :

```
failed domain creation due to memory shortage, unable to balloon domain0
```

Un domaine peut échouer quand il n'y a pas assez de RAM disponible. Domain0 ne fournit pas suffisamment d'espace pour l'invité nouvellement créé. Vous pouvez vérifier le **xend.log** suite à cette erreur :

```
[2006-12-21] 20:33:31 xend 3198] DEBUG (balloon:133) Balloon: 558432 Kib
free; 0 to scrub; need 1048576; retries: 20
[2006-12-21] 20:33:31 xend.XendDomainInfo 3198] ERROR (XendDomainInfo:
202
Domain construction failed
```

You can check the amount of memory in use by domain0 by using the **xm list Domain0** command. If domain0 is not ballooned down, you can use the command "**xm mem-set Domain-0 NewMemSize**" to check memory.

Vous recevez le message d'erreur suivant :

```
wrong kernel image: non-PAE kernel on a PAE
```

This message indicates that you are trying to run an unsupported guest kernel image on your Hypervisor. This happens when you try to boot a non-PAE paravirtual guest kernel on a RHEL 5.1 hypervisor. Red Hat Virtualization only supports guest kernels with PAE and 64bit architectures.

Saisissez cette commande :

```
[root@smith]# xm create -c va base

Using config file "va-base"
Error: (22, 'invalid argument')
[2006-12-14 14:55:46 xend.XendDomainInfo 3874] ERRORs
(XendDomainInfo:202) Domain construction failed

Traceback (most recent call last)
File "/usr/lib/python2.4/site-packages/xen/xend/XendDomainInfo.py", line
195 in create vm.initDomain()
File " /usr/lib/python2.4/site-packages/xen/xend/XendDomainInfo.py", line
1363 in initDomain raise VmError(str(exn))
VmError: (22, 'Invalid argument')
[2006-12-14 14:55:46 xend.XendDomainInfo 3874] DEBUG (XenDomainInfo: 1449]
XendDomainInfo.destroy: domin=1
[2006-12-14 14:55:46 xend.XendDomainInfo 3874] DEBUG (XenDomainInfo: 1457]
XendDomainInfo.destroy:Domain(1)
```

Si vous désirez exécuter un noyau 32bit/non-PAE, il faudra exécuter votre invité en tant que machine virtuelle pleinement virtualisée. Pour les invités paravirtualisés, si vous devez exécuter un invité PAE 32bit, il vous faudra un hyperviseur PAE 32bit. Pour les invités paravirtualisés, si vous devez exécuter un invité PAE 64bit, il vous faudra un hyperviseur PAE 64bit. Pour les invités pleinement virtualisés, vous devez exécuter un invité 64bit avec un hyperviseur 64bit. L'hyperviseur PAE 32bit qui accompagne RHEL 5 i686 ne prend en charge que l'exécution du PAE 32 bit paravirtualisé et l'invité OSes 32bit pleinement virtualisé. L'hyperviseur 64bit ne prend en charge que les invités paravirtualisés 64bit.

This happens when you move the full virtualized HVM guest onto a RHEL 5.1 system. Your guest may fail to boot and you will see an error in the console screen. Check the PAE entry in your configuration file and ensure that pae=1. You should use a 32bit distribution.

Vous recevez le message d'erreur suivant :

```
Unable to open a connection to the Xen hypervisor or daemon
```

Cela arrive quand le démarrage de l'application virt-manager échoue. Cette erreur a lieu quand il n'y a pas d'entrée d'hôte local (localhost) dans le fichier de configuration **/etc/hosts**. Vérifiez le fichier et vérifiez que l'entrée de l'hôte local est activée. Voici un exemple d'entrée d'hôte local incorrecte :

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
localhost.localdomain localhost
```

Voici un exemple d'entrée d'hôte local correcte :

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1 localhost.localdomain localhost
localhost.localdomain. localhost
```

Vous recevez l'erreur suivante (dans le **xen-xend.log** file) :

```
Bridge xenbr1 does not exist!
```

This happens when the guest's bridge is incorrectly configured and this forces the Xen hotplug scripts to

timeout. If you move configuration files between hosts, you must ensure that you update the guest configuration files to reflect network topology and configuration modifications. When you attempt to start a guest that has an incorrect or non-existent Xen bridge configuration, you will receive the following errors:

```
[root@trumble virt]# xm create r5b2-mysql01

Using config file " r5b2-mysql01"
Going to boot Red Hat Enterprise Linux Server (2.6.18.-1.2747 .el5xen)
kernel: /vmlinuz-2.6.18-12747.el5xen
initrd: /initrd-2.6.18-1.2747.el5xen.img
Error: Device 0 (vif) could not be connected. Hotplug scripts not working.
```

Par ailleurs, le fichier **xend.log** affiche les erreurs suivantes :

```
[2006-11-14 15:07:08 xend 3875] DEBUG (DevController:143) Waiting for
devices vif
[2006-11-14 15:07:08 xend 3875] DEBUG (DevController:149) Waiting for 0
[2006-11-14 15:07:08 xend 3875] DEBUG (DevController:464)
hotplugStatusCallback

/local/domain/0/backend/vif/2/0/hotplug-status

[2006-11-14 15:08:09 xend.XendDomainInfo 3875] DEBUG (XendDomainInfo:1449)
XendDomainInfo.destroy: domid=2
[2006-11-14 15:08:09 xend.XendDomainInfo 3875] DEBUG (XendDomainInfo:1457)
XendDomainInfo.destroyDomain(2)
[2006-11-14 15:07:08 xend 3875] DEBUG (DevController:464)
hotplugStatusCallback

/local/domain/0/backend/vif/2/0/hotplug-status
```

Pour résoudre ce problème, vous devez éditer votre fichier de configuration invité et modifier l'entrée **vif**. Quand vous localisez l'entrée **vif** du fichier de configuration, en supposant que vous utilisiez **xenbr0** comme pont par défaut, assurez-vous que l'entrée appropriée ressemble à ce qui suit :

```
# vif = ['mac=00:16:3e:49:1d:11, bridge=xenbr0',]
```

Vous recevez ces erreurs de dépréciation python :

```
[root@python xen]# xm shutdown win2k3xen12
[root@python xen]# xm create win2k3xen12

Using config file "win2k3xen12".

/usr/lib64/python2.4/site-packages/xenxm/opts.py:520: Deprecation Warning:
Non ASCII character '\xc0' in file win2k3xen12 on line 1, but no encoding
declared; see http://www.python.org/peps/pep-0263.html for details

execfile (defconfig, globs, locs,)
Error: invalid syntax 9win2k3xen12, line1)
```

Python génère ces messages quand un fichier de configuration est invalide (ou incorrect). Pour résoudre ce problème, vous devez modifier le fichier de configuration incorrect, ou vous pouvez en créer un nouveau.

18.25. RESSOURCES DE RÉOLUTION DES PANNES EN LIGNE

- Centre de Virtualisation Red Hat

▮ <http://www.openvirtualization.com>

- Documentation Red Hat Enterprise Linux 5 Beta 2

▮ <http://www.redhat.com/docs/manuals/enterprise/RHEL-5-manual/index.html>

- API Libvirt

▮ <http://www.libvirt.org>

- Page accueil du projet virt-manager

▮ <http://virt-manager.et.redhat.com>

- Centre de la communauté Xen

▮ <http://www.xensource.com/xen/xen/>

- Vue d'ensemble des technologies de virtualisation

▮ <http://virt.kernelnewbies.org>

- Projets de technologies émergeants

▮ <http://et.redhat.com>

CHAPITRE 19. RESSOURCES SUPPLÉMENTAIRES

Pour en savoir plus sur la Red Hat Virtualization, reportez-vous aux ressources suivantes.

19.1. SITES WEB UTILES

- <http://www.cl.cam.ac.uk/research/srg/netos/xen/> — Le site Web du projet concernant Xen™, le gestionnaire de machines pour la virtualisation partielle à partir duquel la Red Hat Virtualization est dérivée. Le site maintient en amont le code source et les bibliothèques du projet Xen et contient également des informations, des aperçus de l'architecture, de la documentation et des liens en relation à Xen et ses technologies associées.
- <http://www.libvirt.org/> — Le site Web officiel pour l'API de virtualisation **libvirt** qui interagit avec le framework de virtualisation d'un SE hôte.
- <http://virt-manager.et.redhat.com/> — Le site Web du projet concernant **Gestionnaire de machines virtuelles** (virt-manager), l'application graphique pour la gestion des machines virtuelles.

19.2. DOCUMENTATION INSTALLÉE

- `/usr/share/doc/xen-<version-number>/` — This directory contains a wealth of information about the Xen para-virtualization hypervisor and associated management tools, including a look at various example configurations, hardware-specific information, and the current Xen upstream user documentation.
- `man virsh` and `/usr/share/doc/libvirt-<version-number>` — Contains subcommands and options for the **virsh** virtual machine management utility as well as comprehensive information about the **libvirt** virtualization library API.
- `/usr/share/doc/gnome-applet-vm-<version-number>` — Documentation for the GNOME graphical panel applet that monitors and manages locally-running virtual machines.
- `/usr/share/doc/libvirt-python-<version-number>` — Provides details on the Python bindings for the **libvirt** library. The **libvirt-python** package allows python developers to create programs that interface with the **libvirt** virtualization management library.
- `/usr/share/doc/python-virtinst-<version-number>` — Provides documentation on the **virt-install** command that helps in starting installations of Fedora and Red Hat Enterprise Linux related distributions inside of virtual machines.
- `/usr/share/doc/virt-manager-<version-number>` — Provides documentation on the Virtual Machine Manager, which provides a graphical tool for administering virtual machines.

ANNEXE A. EXERCICE 1

Installation de l'invité Xen

Objectif : installer les invités Xen RHEL 3, 4, ou 5 et Windows XP.

Prérequis : un poste de travail installé avec Red Hat Enterprise Linux 5.0 et le composant de Virtualisation.

Pour cet exercice, vous configurez et installez les invités Xen RHEL 3, 4, ou 5 et Win XP au moyen de divers outils de virtualisation.

Exercice 1, séquence 1 : vérification de la prise en charge PAE

Vous devez déterminer si votre système comporte la prise en charge PAE. La Virtualisation Red Hat prend en charge les architectures CPU basées sur x86_64 ou ia64 pour exécuter des invités paravirtualisés. Pour exécuter les invités i386, le système requiert un CPU avec des extensions PAE. De nombreux ordinateurs portables (surtout ceux basés sur Pentium Mobile ou Centrino) ne prennent pas en charge PAE.

1. Afin de déterminer si votre CPU comporte la prise en charge PAE, saisissez :

```
grep pae /proc/cpuinfo
```

2. La sortie suivante illustre un CPU qui possède la prise en charge PAE. Si la commande ne retourne rien, le CPU ne possède pas de prise en charge PAE. Tous les exercices de ce chapitre requièrent un CPU i386 avec extension PAE ou x86_64 ou ia64 pour fonctionner.

```
flags :  
fpu vme de pse tsc msr pae mce cx8 apic mtrr pge mca cmov pat  
clflush dts acpi  
mmx fxsr sse sse2 ss tm pbe nx up est tm2
```

Exercice 1, séquence 2 : installer l'invité paravirtualisé Xen RHEL5 Beta 2 au moyen de **virt-install**.

Pour cet exercice, vous devez installer un invité Xen Red Hat Enterprise Linux 5 Beta 2 au moyen de **virt-install**.

1. Pour installer votre invité Xen Red Hat Enterprise Linux 5 Beta 2, à l'invite de commande, saisissez : **virt-install**.
2. Lorsqu'on vous demande d'installer un invité pleinement virtualisé, saisissez : **no**.
3. Saisissez **rhe15b2-pv1** pour le nom de votre machine virtuelle.
4. Saisissez 500 pour l'allocation de la mémoire RAM.
5. Saisissez **/xen/rhe15b2-pv1.img** pour votre disque (image invité).
6. Saisissez 6 pour la taille de votre disque (image invité).
7. Saisissez **yes** pour activer la prise en charge graphique.

8. Saisissez **nfs:server:/path/to/rhel5b2** pour votre emplacement d'installation.
9. L'installation commence. Complétez votre installation comme prévu.
10. Après la finalisation de l'installation, saisissez **/etc/xen/rhel5b2-pv1** et effectuez les modifications suivantes : **#vnc=1#vncunused=1sdl=1**
11. Utilisez un éditeur de texte pour modifier **/etc/inittab** et ajoutez ceci au fichier : **init 5.#id:3:initdefault:id:5:initdefault:**

Exercice 1, séquence 3 : installer les invités Xen paravirtualisés RHEL5 Beta 2 au moyen de **virt-manager**.

Pour cet exercice, vous installez un invité paravirtualisé Xen Red Hat Enterprise Linux 5 Beta 2 au moyen de **virt-manager**.

1. Pour installer votre invité paravirtualisé Xen Red Hat Enterprise Linux 5 Beta 2, à l'invite de commande, saisissez : **virt-manager**.
2. Dans la fenêtre Ouvrir la connexion, sélectionnez l'hôte Xen local et cliquez sur **Connecter**.
3. Start Red Hat's Virtual Machine Manager application, and from the **File** menu, click on **New**.
4. Cliquez sur **Continuer**.
5. Saisissez **rhel5b2-pv2** pour le nom de votre système et cliquez sur **Continuer**.
6. Sélectionnez **Paravirtualisé** et cliquez sur **Continuer**.
7. Saisissez **nfs:server:/path/to/rhel5b2** pour l'URL de votre média d'installation et cliquez sur **Continuer**.
8. Sélectionnez **Fichier unique**, saisissez **/xen/rhel5b2-pv2.img** pour l'emplacement de votre fichier. Choisissez 6000 Mo et cliquez sur **Continuer**.
9. Choisissez 500 pour la mémoire de votre MV lors du démarrage et la mémoire maximum, et cliquez sur **Continuer**.
10. Cliquez sur **Arrêter**.

La fenêtre Console de la machine virtuelle s'affiche. Effectuez les opérations comme prévu et terminez votre installation.

Exercice 1, séquence 4 : vérifier la prise en charge Intel-VT ou AMD-V

Pour cet exercice, vous devez déterminer si votre système prend en charge le matériel Intel-VT ou AMD-V. Votre système doit prendre en charge les CPU Intel-VT ou AMD-V pour réussir l'installation de systèmes d'exploitation invités pleinement virtualisés. La Virtualisation Red Hat incorpore une couche générique HVM pour supporter ces vendeurs CPU.

1. To determine if your CPU has Intel-VT or AMD-V support, type the following command: **egrep -e 'vmx|svm' /proc/cpuinfo**
2. La sortie suivante indique un CPU qui prend en charge Intel-VT :

```
■ .flags :
```

```
fpu tsc msr pae mce cx8 apic mtrr mca cmov pat clflush dts acpi
mmx fxsr sse
sse2 ss ht tm pbe constant_tsc pni monitor vmx est tm2 xtpr
```

Si la commande ne retourne rien le CPU ne prend pas en charge Intel-VT ou AMD-V.

3. Pour déterminer si votre CPU prend en charge Intel-VT ou AMD-V, saisissez la commande suivante :

```
cat /sys/hypervisor/properties/capabilities
```

4. The following output shows that Intel-VT support has been enabled in the BIOS. If the command returns nothing, then go into the BIOS Setup Utility and look for a setting related to 'Virtualization', i.e. 'Intel(R) Virtualization Technology' under 'CPU' section on a IBM T60p. Enable and save the setting and do a power off to take effect.

```
xen-3.0-x86_32p hvm-3.0-x86_32 hvm-3.0-x86_32p
```

Exercice 1, séquence 5 : installer l'invité pleinement virtualisé Xen RHEL5 Beta 2 au moyen de `virt-install`.

Pour cet exercice, vous installez un invité pleinement virtualisé Xen Red Hat Enterprise Linux 5 Beta 2 au moyen de `virt-install` :

1. Pour installer votre invité Xen Red Hat Enterprise Linux 5 Beta 2, à l'invite de commande, saisissez : **virt-install**.
2. Quand on vous demande d'installer un invité pleinement virtualisé, saisissez **yes**.
3. Saisissez **rhe15b2-pv2** pour un nom de machine virtuelle.
4. Saisissez 500 pour l'allocation de la mémoire.
5. Saisissez **/xen/rhe15b2-fv1.img** pour votre disque (image invité).
6. Saisissez 6 pour la taille de votre disque (image invité).
7. Saisissez **yes** pour activer la prise en charge graphique.
8. Saisissez **/dev/cdrom** pour l'image du CD virtuel.
9. The VNC viewer appears within the installation window. If there is an error message that says "main: Unable to connect to host: Connection refused (111)", then type the following command to proceed: **vncviewer localhost:5900**. VNC port 5900 refers to the first Xen guest that is running on VNC. If it doesn't work, you might need to use 5901, 5902, etc.

L'installation commence. Complétez votre installation comme prévu.

Exercice 1, séquence 6 : installer l'invité pleinement virtualisé Xen RHEL5 Beta 2 au moyen de `virt-manager`.

Pour cet exercice, vous installez un invité pleinement virtualisé Xen Red Hat Enterprise Linux 5 Beta 2 au moyen de `virt-manager` :

1. Pour installer votre invité paravirtualisé Xen Red Hat Enterprise Linux 5 Beta 2, à l'invite de commande, saisissez : **virt-manager**.
2. Dans la fenêtre **Ouvrir la connexion**, sélectionnez l'hôte Xen local et cliquez sur **Connecter**.
3. Start Red Hat's Virtual Machine Monitor application, and from the **File** menu, click on **New**.
4. Cliquez sur **Continuer**.
5. Saisissez **rhe15b2-fv2** pour votre nom de système et cliquez sur **Continuer**.
6. Sélectionnez Pleinement virtualisé et cliquez sur **Continuer**.
7. Spécifiez CD-ROM ou DVD et entrez le chemin du média d'installation. Spécifiez l'emplacement de l'image ISO si vous installez à partir d'une image ISO. Cliquez sur **Continuer**.
8. Sélectionnez Fichier unique, saisissez **/xen/rhe15b2-fv2.img** pour l'emplacement de votre fichier. Spécifiez 6000 Mo et cliquez sur **Continuer**.
9. Choisissez 500 pour la mémoire de votre MV lors du démarrage et la mémoire maximum, et cliquez sur **Continuer**.
10. Cliquez sur **Arrêter**.
11. La fenêtre Console de la machine virtuelle s'affiche.

Effectuez les opérations comme prévu et complétez l'installation.

Exercice 1, séquence 7 : installer un invité pleinement virtualisé Xen RHEL3 au moyen de **virt-manager**.

Pour cet exercice, vous installez un invité Xen Red Hat Enterprise Linux 3 au moyen de **virt-manager**:

1. Les mêmes instructions que pour la séquence 6 sont applicables ici.

Exercice 1, séquence 8 : installer un invité pleinement virtualisé Xen RHEL4 au moyen de **virt-manager**

Pour cet exercice, vous installez un invité pleinement virtualisé Xen Red Hat Enterprise Linux RHEL4 au moyen de **virt-manager** :

1. Les mêmes instructions que pour la séquence 6 sont applicables ici.

Exercice 1, séquence 9 : installer un invité pleinement virtualisé Xen Windows XP au moyen de **virt-manager**.

Pour cet exercice, vous installez un invité pleinement virtualisé Xen Windows XP au moyen de **virt-manager** :

1. Pour installer votre Red Hat Enterprise Linux 5 sur votre hôte Windows XP, à l'invite de commande, saisissez : **virt-manager**.
2. Dans la fenêtre **Ouvrir la connexion**, sélectionnez l'hôte Xen local, et cliquez sur **Connecter**.

3. Start Red Hat's Virtual Machine Manager application, and from the **File** menu click on **New**.
4. Cliquez sur **Continuer**.
5. Saisissez **winxp** pour votre nom de système et cliquez sur **Continuer**.
6. Sélectionnez Pleinement virtualisé et cliquez sur **Continuer**.
7. Spécifiez CD-ROM ou DVD et entrez le chemin du média d'installation. Spécifiez l'emplacement de l'image ISO si vous installez à partir d'une image ISO. Cliquez sur **Continuer**.
8. Sélectionnez Fichier unique, saisissez **/xen/winxp.img** pour l'emplacement de votre fichier. Spécifiez 6000 Mo et cliquez sur **Continuer**.
9. Sélectionnez 1024 pour la mémoire de votre MV lors du démarrage et la mémoire maximum, et sélectionnez 2 pour les VCPU. Cliquez sur **Continuer**.
10. Cliquez sur **Arrêter**.
11. La fenêtre Console de la machine virtuelle s'affiche. Effectuez les opérations comme prévu et terminez votre installation.
12. Choisissez de formater la partition **C:** dans un format de système de fichiers FAT. Red Hat Enterprise Linux 5 ne comporte pas de modules de noyau NTFS. Si vous formatez la partition dans un format de système de fichiers NTFS, monter ou écrire des fichiers dans l'image de l'invité Xen n'est pas aussi simple.
13. Après le redémarrage du système pour la première fois, éditez l'image de l'invité **winxp** :
losetup /dev/loop0 /xen/winxp.imgkpartx -av /dev/loop0mount /dev/mapper/loop0p1 /mntcp -prv \$WINDOWS/i386 /mnt/. Cela résout un problème potentiel dans l'installation de Windows.
14. Redémarrez l'invité Xen manuellement en saisissant : **xm create -c winxp/**.
15. Dans la fenêtre du Gestionnaire de machines virtuelles, sélectionnez l'invité Xen **winxp** et cliquez sur **Ouvrir**.
16. La console de la machine virtuelle s'affiche. Effectuez les opérations comme prévu et complétez l'installation.
17. Whenever a 'Files Needed' dialog box appears, change the path **GLOBALROOT\DEVICE\CDROM0\I386** to **C:\I386**. Depending on your installation, you may or may not see this problem. You may be prompted for missing files during the installation. Changing the path to **C:\I386** should compensate for this problem.
18. Si la console de l'invité Xen se bloque, cliquez sur **arrêter** et effectuez les modifications suivantes dans **/etc/xen/winxp** : **#vnc=1#vncunused=1sdl=1#vcpus=2**
19. Répétez l'étape 14 et effectuez votre installation comme prévu.

ANNEXE B. EXERCICE 2

Migration en direct (Live Migration)

Objectif : configurer et effectuer une migration en direct entre deux hôtes.

Prérequis : deux postes de travail installés avec Red Hat Enterprise Linux 5.0 Beta 2 avec la plateforme de Virtualisation et un invité Xen Fedora Core 6 sur un des deux postes de travail.

Pour cet exercice, vous configurez la migration et exécutez une migration en direct entre deux hôtes.

Introduction : avant de commencer

Pour cet exercice, il vous faudra deux hôtes de virtualisation : un invité Xen et un stockage partagé. Vous devez connecter les deux hôtes de virtualisation via un câble UTP. Un des hôtes de virtualisation exporte un stockage partagé via NFS. Vous devez configurer les deux hôtes de virtualisation afin qu'ils réussissent la migration. L'invité Xen réside dans le stockage partagé. Sur l'invité Xen, il vous faudra installer un serveur de flux. Vous devez vous assurer que le serveur de flux s'exécute sans interruption sur l'invité Xen, ainsi la migration en direct a lieu entre un hôte de virtualisation et l'autre. Pour l'exercice 2, vous désignez les deux hôtes de virtualisation, comme **host1** et **host2**.

Séquence 1 : configurer xend (les deux hôtes Xen)

Dans cet exercice, vous configurez xend pour démarrer en tant que serveur HTTP et serveur de délocalisation. Le démon xend n'initie pas le serveur HTTP par défaut. Il démarre le serveur de gestion de sockets de domaine UNIX (pour **xm**) et communique avec xend. Pour activer la migration en direct entre-machines vous devez le configurer de façon à ce qu'il prenne en charge la migration en direct :

1. Pour sauvegarder votre fichier **xend-config.sxp** :

```
cp -pr /etc/xen/xend-config.sxp /etc/xen/xend-config.sxp.default
```

2. Éditez **/etc/xen/xend-config.sxp** et effectuez les modifications suivantes :

```

#(xend-unix-server yes)(xend-relocation-server
  yes)(xend-relocation-port 8002)(xend-relocation-address
  '')(xend-relocation-hosts-allow '')#(xend-relocation-hosts-allow
'^localhost$
  ^localhost\\.localdomain$')

```

3. Redémarrez **xend:service** et **xend restart**.

Séquence 2 : exporter un stockage partagé via NFS

Dans cet exercice, vous configurez NFS et vous l'utilisez pour exporter un stockage partagé.

1. Éditez **/etc/exports** et ajoutez la ligne : **/xen *(rw, sync, no_root_squash)/**
2. Enregistrez **/etc/exports** et redémarrez le serveur NFS. Assurez-vous que le serveur NFS démarre par **default:service nfs startchkconfig nfs on**.
3. Après avoir démarré le serveur NFS sur **host1** vous pouvez le monter sur **host2:mount host1:/xen** .

- Maintenant démarrez l'invité Xen sur **host1** et sélectionnez **fc6-pv1** (ou **fc6-pv2** depuis l'exercice 1) :

```
xm create -c fc6-pv1
```

Séquence 3 : installer le serveur de flux de l'invité Xen

Pour cette étape, vous installez un serveur de flux, gnump3d, pour vos démonstrations. Vous sélectionnez gnump3d parce qu'il prend en charge les fichiers vorbis OGG et il est facile à installer, configurer et modifier.

- Téléchargez le tarball **gnump3d-2.9.9.9.tar.bz2** depuis <http://www.gnump3d.org/>. Désempaquetez le tarball et dans le répertoire **gnump3d-2.9.9.9/**, compilez et installez le **gnump3d application:tar xvjf gnump3d-2.9.9.9.tar.bz2cd gnump3d-2.9.9.9/make install**
- Create a **/home/mp3** directory and copy TruthHappens.ogg from Red Hat's Truth Happens page to **mkdir /home/mp3wget -c http://www.redhat.com/v/ogg/TruthHappens.ogg**
- Démarrez le serveur de flux en saisissant :

```
command:gnump3d
```

- Sur l'un des deux hôtes Xen, exécutez Movie Player. S'il n'est pas installé, installez le totem et les rpms iso-codecs avant d'exécuter Movie Player. Cliquez sur Applications, ensuite Son & Vidéo et enfin Movie Player.
- Cliquez sur Movie, ensuite Ouvrir l'emplacement. Entrez <http://guest:8888/TruthHappens.ogg>.

Séquence 4 : effectuer la migration en direct

- Exécutez le fichier **TruthHappens.ogg** sur l'un des deux hôtes Xen.
- Effectuez la migration en direct de **host1** à **host2** :

```
xm migrate -live fc6-pv1 host2
```

- Ouvrez des terminaux à fenêtres multiples sur les deux hôtes Xen avec la commande suivante :

```
watch -n1 xm list
```

- Observez le commencement de la migration en direct. Notez la durée de la migration.

Séquence challenge : configurer le serveur VNC à partir de l'invité Xen

If time permits, from within the Xen guest, configure the VNC server to initiate when **gdm** starts up. Run VNC viewer and connect to the Xen guest. Play with the Xen guest when the live migration occurs. Attempt to pause/resume, and save/restore the Xen guest and observe what happens to the VNC viewer. If you connect to the VNC viewer via **localhost:590x**, and do a live migration, you won't be able to connect to the VNC viewer again when it dies. This is a known bug.

ANNEXE C. HISTORIQUE DES VERSIONS

Version 2.0-11.400
Rebuild with publican 4.0.0

2013-10-31

Rüdiger Landmann

Version 2.0-11
Rebuild for Publican 3.0

2012-07-18

Anthony Towns

Version 5.1.0-10
Resolves: #245684
Content Updates

Fri Aug 3 2007

Michael Hideo Smith