



Red Hat Enterprise Linux 5

Virtualization

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

Edizione 2.0

Red Hat Enterprise Linux 5 Virtualization

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

Edizione 2.0

Landmann

rlandmann@redhat.com

Nota Legale

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.

Sommario

Questa guida contiene le informazioni su come configurare, creare e controllare i sistemi operativi guest su Red Hat Enterprise Linux 5, utilizzando virsh, xm, vmm e xend. Se trovate un errore nella Red Hat Enterprise Linux Virtualization Guide, o se avete pensato ad un modo per migliorare questo manuale, inviateci i vostri suggerimenti! Inviateli tramite Bugzilla (<http://bugzilla.redhat.com/bugzilla/>), specificando il prodotto, Red Hat Enterprise Linux, ed il componente Virtualization_Guide.

CAPITOLO 1. ARCHITETTURA DEL SISTEMA DI RED HAT VIRTUALIZATION	6
CAPITOLO 2. SUPPORTO SISTEMA OPERATIVO	8
CAPITOLO 3. SUPPORTO HARDWARE	10
CAPITOLO 4. REQUISITI DEL SISTEMA PER RED HAT VIRTUALIZATION	11
CAPITOLO 5. AVVIO DEL SISTEMA	12
CAPITOLO 6. CONFIGURAZIONE DI GRUB	13
CAPITOLO 7. AVVIO DI UN DOMINIO GUEST	15
CAPITOLO 8. AVVIO/ARRESTO DI UN DOMINIO AL MOMENTO DELL'AVVIO	16
CAPITOLO 9. FILE DI CONFIGURAZIONE	17
CAPITOLO 10. GESTIONE DELLE CPU	18
CAPITOLO 11. MIGRAZIONE DI UN DOMINIO	19
CAPITOLO 12. CONFIGURAZIONE PER UN UTILIZZO SU DI UNA RETE	20
CAPITOLO 13. COME RENDERE SICURO DOMAIN0	21
CAPITOLO 14. STORAGE	22
CAPITOLO 15. GESTIONE DELLE MACCHINE VIRTUALI CON VIRSH	23
15.1. COLLEGAMENTO AD UN HYPERVISOR	23
15.2. CREAZIONE DI UNA MACCHINA VIRTUALE	23
15.3. CONFIGURAZIONE DI UN XML DUMP	23
15.4. SOSPENSIONE DI UNA MACCHINA VIRTUALE	23
15.5. RIPRISTINO DI UNA MACCHINA VIRTUALE	23
15.6. SALVATAGGIO DI UNA MACCHINA VIRTUALE	24
15.7. RIPRISTINO DI UNA MACCHINA VIRTUALE	24
15.8. ARRESTO DI UNA MACCHINA VIRTUALE	24
15.9. RIAVVIO DI UNA MACCHINA VIRTUALE	24
15.10. COME TERMINARE UN DOMINIO	24
15.11. CONVERSIONE DI UN DOMAIN NAME IN UN DOMAIN ID	25
15.12. CONVERSIONE DI UN DOMAIN ID IN UN DOMAIN NAME	25
15.13. CONVERSIONE DI UN DOMAIN NAME IN UN UUID	25
15.14. VISUALIZZAZIONE INFORMAZIONI DELLA MACCHINA VIRTUALE	25
15.15. VISUALIZZAZIONE INFORMAZIONI DEL NODO	25
15.16. VISUALIZZAZIONE DELLE MACCHINE VIRTUALI	26
15.17. VISUALIZZAZIONE INFORMAZIONI DELLA CPU VIRTUALE	26
15.18. CONFIGURAZIONE AFFINITÀ CPU VIRTUALE	26
15.19. CONFIGURAZIONE CONTEGGIO CPU VIRTUALE	26
15.20. CONFIGURAZIONE ASSEGNAZIONE DELLA MEMORIA	27
15.21. CONFIGURAZIONE MEMORIA MASSIMA	27
15.22. GESTIONE RETI VIRTUALI	27
CAPITOLO 16. GESTIONE MACCHINE VIRTUALI UTILIZZANDO XEND	29
CAPITOLO 17. GESTIONE DELLE MACCHINE VIRTUALI UTILIZZANDO XM	31
17.1. FILE DI CONFIGURAZIONE XM	31

17.1.1. Configurazione di vfb	32
17.2. CREAZIONE E GESTIONE DEI DOMINI CON XM	33
17.2.1. Collegamento ad un dominio	33
17.2.2. Creazione di un dominio	33
17.2.3. Come salvare un dominio	33
17.2.4. Come terminare un Domain ID	34
17.2.5. Come arrestare un dominio	34
17.2.6. Ripristino di un dominio	34
17.2.7. Sospensione di un dominio	34
17.2.8. Ripristinare un dominio	34
17.2.9. Come riavviare un dominio	34
17.2.10. Come rinominare un dominio	34
17.2.11. Pausa di un dominio	35
17.2.12. Come rimuovere la pausa di un dominio	35
17.2.13. Conversione di un Domain Name in un Domain ID	35
17.2.14. Conversione di un Domain ID in un Domain Name	35
17.2.15. Configurazione assegnazione della memoria	35
17.2.16. Configurazione della memoria massima	35
17.2.17. Configurazione del conteggio VCPU	36
17.2.18. Come fissare una VCPU	36
17.2.19. Migrazione di un dominio	36
17.3. MONITORAGGIO E DIAGNOSI	36
17.3.1. Esecuzione di un Core Dump	36
17.3.2. Monitoraggio dei domini in tempo reale	37
17.3.3. Visualizzazione degli stati del dominio	37
17.4. VISUALIZZAZIONE UPTIME	38
17.5. VISUALIZZAZIONE INFORMAZIONI DELLA VCPU	38
17.6. VISUALIZZAZIONE INFORMAZIONI DEL DOMINIO	38
17.7. VISUALIZZAZIONE DISPOSITIVI TPM	39
17.8. VISUALIZZAZIONE DEL LOG DI XEND	39
17.9. VISUALIZZAZIONE DEL MESSAGE BUFFER	39
17.10. VISUALIZZAZIONE DELLE INFORMAZIONI SULLO STATO ACM	39
17.11. VISUALIZZAZIONE DEI DISPOSITIVI A BLOCCHI VIRTUALI	40
17.12. VISUALIZZAZIONE DELLE INTERFACCE DI RETE VIRTUALI	40
17.13. CREAZIONE DI UN NUOVO DISPOSITIVO DI RETE VIRTUALE	40
17.14. CREAZIONE DI UNA ETICHETTA DI SICUREZZA DEL DOMINIO	40
17.15. TEST DELLE RISORSE DEL DOMINIO	41
17.16. VISUALIZZAZIONE RISORSE DEL SISTEMA	41
17.17. CONFIGURAZIONE CREDIT SCHEDULING	41
17.18. CREAZIONE DI UN NUOVO DISPOSITIVO A BLOCCHI VIRTUALE	41
17.19. COME TERMINARE UN DISPOSITIVO A BLOCCHI VIRTUALE	42
17.20. SICUREZZA	42
17.20.1. Rimozione di una etichetta di sicurezza del dominio	42
17.20.2. Creazione di una etichetta di sicurezza delle risorse	42
17.20.3. Rimozione di una etichetta di sicurezza delle risorse	42
17.20.4. Configurazione controllo accesso	42
17.20.5. Creazione di una policy	43
17.20.6. Caricamento della policy	43
17.20.7. Creazione policy per una configurazione del processo d'avvio	43
17.20.8. Creazione di una etichetta	44
17.20.9. Visualizzazione etichette della policy	44
17.20.10. Visualizzazione delle etichette di sicurezza del dominio	44
17.20.11. Visualizzazione delle etichette di sicurezza delle risorse	44

17.20.12. Configurazione di Access Control Security	44
17.20.13. Compilazione di una policy di sicurezza	44
17.20.14. Caricamento di una policy di sicurezza	45
17.20.15. Configurazione di una policy di sicurezza per l'avvio	45
17.20.16. Visualizzazione delle etichette di sicurezza	45
17.20.17. Come collegare una etichetta di sicurezza	45
CAPITOLO 18. GESTIONE MACCHINE VIRTUALI CON VIRTUAL MACHINE MANAGER	47
18.1. ARCHITETTURA DEL VIRTUAL MACHINE MANAGER	47
18.2. FINESTRA APRI COLLEGAMENTO	47
18.3. FINESTRA DEL VIRTUAL MACHINE MANAGER	47
18.4. FINESTRA DATTEGLI DELLA MACCHINA VIRTUALE	48
18.5. CONSOLE GRAFICA DELLA MACCHINA VIRTUALE	49
18.6. STARTING THE VIRTUAL MACHINE MANAGER	50
18.7. CREAZIONE DI UNA NUOVA MACCHINA VIRTUALE	50
18.8. RIPRISTINO DI UNA MACCHINA PRECEDENTEMENTE SALVATA	60
18.9. DISPLAYING VIRTUAL MACHINE DETAILS	62
18.10. CONFIGURING STATUS MONITORING	66
18.11. VISUALIZZAZIONE DEL DOMAIN ID	67
18.12. VISUALIZZAZIONE DELLO STATO DELLA MACCHINA VIRTUALE	68
18.13. VISUALIZZAZIONE DELLE CPU VIRTUALI	69
18.14. VISUALIZZAZIONE UTILIZZO DELLA CPU	70
18.15. VISUALIZZAZIONE UTILIZZO DELLA MEMORIA	71
18.16. GESTIONE DI UNA RETE VIRTUALE	72
18.17. CREAZIONE DI UNA RETE VIRTUALE	73
CAPITOLO 19. TROUBLESHOOTING DI RED HAT VIRTUALIZATION	82
19.1. PANORAMICA DI LOGFILE E POSIZIONI	82
19.2. DESCRIZIONI LOGFILE	82
19.3. POSIZIONI DIRECTORY IMPORTANTI	83
19.4. TOOL PER IL TROUBLESHOOTING	83
19.5. TROUBLESHOOTING CON I LOG	85
19.6. TROUBLESHOOTING CON LA CONSOLE SERIALE	85
19.7. ACCESSO CONSOLE GUEST PARAVIRTUALIZZATO	86
19.8. ACCESSO COMPLETO ALLA CONSOLE DEL GUEST DI VIRTUALIZZAZIONE	86
19.9. IMPLEMENTAZIONE PERISTENZA LUN	86
19.10. CONSIDERAZIONI SU SELINUX	88
19.11. ACCESSO DEI DATI SU DI UNA IMMAGINE DEL DISCO GUEST	88
19.12. SITUAZIONI COMUNI DI TROUBLESHOOTING	89
19.13. ERRORI DEL DISPOSITIVO LOOP	89
19.14. ERRORI DURANTE LA CREAZIONE DI UN GUEST	90
19.15. ERRORI DELLA CONSOLE SERIALE	90
19.16. ERRORI NETWORK BRIDGE	90
19.17. CONFIGURAZIONI LAPTOP	92
19.18. STARTING DOMAINS AUTOMATICALLY DURING SYSTEM BOOT	93
19.19. MODIFICA DI DOMAIN0	94
19.20. FILE DI CONFIGURAZIONE DEL GUEST	95
19.21. CLONAZIONE DEI FILE DI CONFIGURAZIONE DEL GUEST	95
19.22. CREAZIONE DI UNO SCRIPT PER GENERARE GLI INDIRIZZI MAC	96
19.23. CONFIGURAZIONE DELLA VIRTUAL MACHINE LIVE MIGRATION	96
19.24. INTERPRETAZIONE DEI MESSAGGI D'ERRORE	97
19.25. RISORSE ONLINE PER IL TROUBLESHOOTING	99
CAPITOLO 20. RISORSE AGGIUNTIVE	101

20.1. SITI WEB UTILI	101
20.2. DOCUMENTAZIONE INSTALLATA	101
APPENDICE A. LAB 1	102
APPENDICE B. LAB 2	107
APPENDICE C. CRONOLOGIA	109

CAPITOLO 1. ARCHITETTURA DEL SISTEMA DI RED HAT VIRTUALIZATION

Un sistema Red Hat Virtualization presenta diversi livelli, ed è guidato da un componente privilegiato di Red Hat Virtualization. Red Hat Virtualization può ospitare diversi sistemi operativi guest. Ogni sistema operativo guest viene eseguito nel proprio dominio, Red Hat Virtualization programma CPU virtuali all'interno delle macchine virtuali, in modo da utilizzare al meglio le CPU fisiche disponibili. Ogni sistema operativo guest gestisce le proprie applicazioni programmandole in modo conforme.

È possibile impiegare Red Hat Virtualization in uno dei seguenti modi: **full virtualization** o **paravirtualization**. Full virtualization fornisce un'astrazione totale del sistema fisico in questione, e crea un nuovo sistema virtuale nel quale i sistemi operativi guest possono essere eseguiti. Non sono necessarie alcune modifiche nell'OS guest o applicazione (l'OS guest o applicazione non è a conoscenza dell'ambiente virtualizzato e per questo motivo viene eseguito normalmente). Paravirtualization richiede invece una modifica dei sistemi operativi guest da parte dell'utente in esecuzione sulle macchine virtuali (i suddetti sistemi operativi guest sanno di essere in esecuzione su di una macchina virtuale), e fornisce prestazioni simili a quelle native. È possibile impiegare sia il metodo paravirtualization che full virtualization attraverso la vostra infrastruttura di virtualizzazione.

Il primo dominio conosciuto come **domain0** (dom0), viene creato automaticamente durante l'avvio del sistema. Domain0 è il guest privilegiato e possiede capacità di gestione, le quali rendono possibile la creazione di nuovi domini e la gestione dei propri dispositivi virtuali. Domain0 gestisce l'hardware fisico come ad esempio le schede di rete, ed i controllori del disco fisso. Domain0 gestisce anche compiti amministrativi come la sospensione, il ripristino o la migrazione dei domini del guest su altre macchine virtuali.

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.

È possibile configurare ogni guest con un numero di **cpus** virtuali (chiamate vcpus). Il Virtual Machine Manager programma le vcpus in base al carico di lavoro sulle CPU fisiche.

È possibile conferire al guest qualsiasi numero di **virtual disks**. Il guest a sua volta considera i suddetti virtual disks come dischi fissi, o (per guest completamente virtuali) come unità CD-ROM. Ogni disco virtuale viene servito al guest da un dispositivo a blocchi o da un file regolare presente sull'host. Il dispositivo sull'host contiene l'immagine del disco intera per il guest, e generalmente include le tabelle della partizione, le partizioni multiple, ed i volumi fisici LVM.

Virtual networking interfaces viene eseguito sul guest. È possibile eseguire altre interfacce sul guest come virtual ethernet internet cards (VNICs). Queste interfacce di rete sono configurate con un indirizzo persistente media access control (MAC) virtuale. L'installazione predefinita di un nuovo guest installa il VNIC con un indirizzo MAC, selezionato in modo randomico da un gruppo riservato di oltre 16 milioni di indirizzi, in questo modo sarà improbabile che due guest possano ricevere lo stesso indirizzo MAC. Siti complessi con un numero molto elevato di guest possono assegnare indirizzi MAC manualmente, in modo da assicurare così la loro unicità sulla rete.

Ogni guest possiede una **text console** virtuale in grado di collegarsi all'host. È possibile ridirezionare i login del guest e l'output della console sulla text console.

Potete configurare qualsiasi guest in modo da utilizzare una **graphical console** virtuale, che

corrisponde alla console video normale sull'host fisico. È possibile eseguire tale operazione per guest di tipo full virtual e paravirtual. In questo caso vengono utilizzate le caratteristiche dell'adattatore grafico standard come il boot messaging, l'avvio grafico, i terminali virtuali multipli, ed è possibile lanciare il sistema x window. Potete altresì utilizzare la tastiera grafica per configurare la tastiera virtuale ed il mouse.

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.

CAPITOLO 2. SUPPORTO SISTEMA OPERATIVO

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.

Red Hat Virtualization vi permette di eseguire un kernel guest non modificato, se siete in possesso di un hardware della CPU AMD SVM e Intel VT. Non è necessario eseguire un port del vostro sistema operativo, per poter impiegare questa architettura sui vostri sistemi Intel VT o AMD SVM. Red Hat Virtualization supporta:

- Intel VT-x o AMD-V Pacifica e la tecnologia Vanderpool per le modalità full e paravirtualization.
- Intel VT-i per ia64
- I sistemi operativi UNIX e Linux, incluso NetBSD, FreeBSD, e Solaris.
- Microsoft Windows as an unmodified guest operating system with Intel Vanderpool or AMD's Pacifica technology.

Per eseguire i full virtualization guest sui sistemi con Hardware-assisted Virtual Machine (HVM), Intel, o piattaforme AMD, è necessario assicurarsi che le vostre CPU abbiano le capacità necessarie per eseguire tale compito.

Per controllare se siete in possesso dei flag per la CPU per un supporto Intel, inserite quanto segue:

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

L'output visualizza:

```
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
```

Se riuscite a visualizzare un flag vmx allora siete in possesso di un supporto Intel.

Per controllare se siete in possesso dei flag per la CPU per un supporto AMD, inserite quanto segue:

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

L'output visualizza:

```
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
```

Se riuscite a visualizzare un flag svm allora siete in possesso di un supporto AMD.

**NOTA**

In aggiunta al controllo dei flag della CPU, sarà necessario abilitare la modalità full virtualization all'interno del BIOS del vostro sistema.

CAPITOLO 3. SUPPORTO HARDWARE

Red Hat Virtualization supporta i sistemi multiprocessor, e permette di eseguire Red Hat Virtualization su sistemi con architettura x86 con un processore simile alla classe P6 (o versioni precedenti):

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

Con Red Hat Virtualization gli host a 32-bit eseguono solo paravirtual guest a 32-bit. Gli host a 64-bit eseguono solo paravirtual guest a 64-bit, ed un full virtualization host a 64-bit esegue guest a 32-bit, 32-bit PAE, o 64-bit. Un full virtualization host a 32-bit esegue full virtualization guest PAE e non-PAE.

Il kernel di Red Hat Enterprise Linux Virtualization non supporta più di 32GB di memoria per i sistemi x86_64. Se desiderate avviare il kernel di virtualizzazione sui sistemi che presentano più di 32GB di memoria fisica, sarà necessario inserire **mem=32G** sulla linea di comando del kernel. Questo esempio mostra come abilitare i parametri corretti nel file **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 (Physical Address Extension) è in grado di aumentare la quantità di memoria fisica o virtuale disponibile alle user applications. Red Hat Virtualization necessita di una attivazione sui vostri sistemi di PAE. L'architettura a 32-bit di Red Hat Virtualization con PAE supporta fino a 16GB di memoria fisica. È consigliato avere a disposizione almeno 256 megabytes di RAM per ogni guest in esecuzione sul vostro sistema. Red Hat Virtualization abilita macchine x86/64 in modo da supportare fino a 64GB di memoria fisica. I kernel di Red Hat Virtualization non verranno eseguiti su sistemi non-PAE. Per determinare se un sistema supporta PAE digitate i seguenti comandi:

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

Il seguente output visualizza:

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

Se il vostro output corrisponde (oppure è simile) a quello sopra riportato, allora la CPU supporta PAE. Se il prompt del comando non visualizza alcun output, allora la CPU non lo supporta.

CAPITOLO 4. REQUISITI DEL SISTEMA PER RED HAT VIRTUALIZATION

Gli oggetti elencati qui di seguito risultano essere necessari per il sistema Red Hat Virtualization:

- Una distribuzione Red Hat RHEL 5 Linux funzionante
- Un bootloader GRUB funzionante
- Accesso root
- Processore di classe P6 (o versione precedente)
- Il Linux bridge-utils
- I sistemi hotplug di Linux
- installazione di sviluppo zlib
- Python 2.2 runtime
- initscripts

Le dipendenze sono configurate automaticamente durante il processo d'installazione.

CAPITOLO 5. AVVIO DEL SISTEMA

Dopo aver installato i componenti di Red Hat Virtualization, è necessario riavviare il sistema. Una volta completato il processo d'avvio, è necessario eseguire un login sul vostro sistema. Successivamente prima di avviare Red Hat Virtualization, eseguite un login come utente root. Il demone di controllo di xend, dovrebbe essere già inizializzato da initscripts, ma per avviare xend manualmente inserite:

```
service xend start
```

È possibile utilizzare **chkconfig xend** durante il processo di installazione, per poter abilitare **xend** al momento dell'avvio.

Il demone di controllo del nodo di xend, esegue le funzioni di gestione del sistema riguardanti le macchine virtuali. Questo demone controlla le risorse virtualizzate, e per poter interagire con le macchine virtuali sarà necessario eseguire xend. Prima di avviare xend, è necessario specificare i parametri operativi modificando il file di configurazione xend, xend-config.sxp, il quale si trova nella directory etc/xen.

CAPITOLO 6. CONFIGURAZIONE DI 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
```

Se impostate le vostre entry di Linux Grub in modo da riflettere questo esempio, il boot loader carica l'hypervisor, l'immagine `initrd`, ed il kernel di Linux. Poichè la entry del kernel si trova sopra le altre entry, il kernel viene caricato prima nella memoria. Il boot loader invia (e riceve) gli argomenti della linea di comando da e per l'hypervisor, ed il kernel di Linux. Il seguente esempio mostra come limitare la memoria del kernel di linux del Domain0 a 800 MB:

```
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
```

È possibile utilizzare i parametri di GRUB per poter configurare l'hypervisor per la Virtualizzazione:

```
mem
```

Ciò limita la quantità di memoria disponibile per domain0.

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

Abilita la prima porta seriale nel sistema in modo da comportarsi come console seriale (com2 viene assegnata per la porta successiva, e così via...).

```
dom0_mem
```

Ciò limita la quantità di memoria disponibile per domain0.

```
dom0_max_vcpus
```

Limita la quantità di CPU visibile da parte del domain0.

acpi

Cambia l'hypervisor ACPI in hypervisor e domain0. Le opzioni del parametro ACPI includono:

```
/* **** 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

Disabilita ACPI per le interrupt delivery.

CAPITOLO 7. AVVIO DI UN DOMINIO GUEST

È possibile avviare domini guest utilizzando l'applicazione **xm**. È possibile altresì usare **virsh** ed il Virtual Machine Manager, per avviare i guest. Un prerequisito per l'avvio di un dominio guest è quello di installare prima un guest host. Questo esempio utilizza **xm** per creare il sottocomando:

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

guestdomain1 è il file di configurazione per il dominio che state avviando. L'opzione **-c** collega alla console dopo l'avvio.

CAPITOLO 8. AVVIO/ARRESTO DI UN DOMINIO AL MOMENTO DELL'AVVIO

È possibile avviare o arrestare i domini in esecuzione in qualsiasi momento. Domain0 attende l'arresto di tutti i domini in esecuzione prima di eseguire un riavvio. È necessario posizionare i file di configurazione dei domini che desiderate arrestare, all'interno della directory `/etc/xen/`. Tutti i domini che desiderate avviare al momento dell'avvio, devono essere collegati tramite link simbolico 'symlinked' a `/etc/xen/auto`.

```
chkconfig xendomains on
```

Il comando **chkconfig xendomains on** non avvia in modo automatico i domini; esso li riavvia durante il processo d'avvio successivo.

```
chkconfig xendomains off
```

Termina tutti i domini Red Hat Virtualization in esecuzione. Il comando **chkconfig xendomains off** arresta i domini durante il processo d'avvio successivo.

CAPITOLO 9. FILE DI CONFIGURAZIONE

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.

Tabella 9.1. File di configurazione di Red Hat Virtualization

Item	Descrizione
paef	Specifica i dati di configurazione dell'estensione physical address extention.
apic	Specifica i dati di configurazione dell'advanced programmable interrupt controller.
memoria	Specifica la dimensione della memoria in megabytes.
vcpus	Specifica i numeri delle CPU virtuali.
console	Specifica i numeri della porta verso la quale esportare le console del dominio.
nic	Specifica il numero di virtual network interfaces.
vif	Lists the randomly-assigned MAC addresses and bridges assigned to use for the domain's network addresses.
disk	Elenca i dispositivi a blocchi sui quali esportare il dominio, ed esporta i dispositivi fisici sul dominio con un accesso di sola lettura.
dhcp	Aabilita il networking utilizzando DHCP.
netmask	Specifica le maschere di rete IP configurate.
gateway	Specifica i gateway IP configurati.
acpi	Specifica i dati di configurazione per l'advanced configuration power interface.

CAPITOLO 10. GESTIONE DELLE 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, its 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.

CAPITOLO 11. MIGRAZIONE DI UN DOMINIO

Il processo di migrazione rappresenta il trasferimento di un dominio virtuale in esecuzione, da un host fisico ad un altro. Red Hat Virtualization supporta due diversi tipi di migrazione — offline e live. La migrazione di tipo offline sposta una macchina virtuale da un host ad un'altro arrestandolo, trasferendo la sua memoria e successivamente ripristinandolo sull'host desiderato. Una migrazione di tipo live segue un processo simile, senza però interferire con il dominio. Se eseguite una migrazione di tipo live il dominio continuerà le sue normali attività, in questo modo l'utente non verrà interessato dal processo. Per iniziare una migrazione di tipo live, entrambi gli host devono eseguire Red Hat Virtualization ed il demone xend. Le destinazioni host devono essere in possesso di risorse sufficienti (come ad esempio una capacità di memoria idonea), in modo da soddisfare la larghezza di banda dopo il processo di migrazione. Entrambe le macchine sia di destinazione che quella sorgente devono avere le stesse estensioni di virtualizzazione e la stessa architettura. (come ad esempio i386-VT, x86-64-VT, x86-64-SVM, ecc.), e devono essere sulla stessa sottorete 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.

Poichè i domini interessati possiedono un processo di assegnazione dei file molto esteso, il suddetto processo può richiedere una quantità di tempo molto elevata. Se eseguite la migrazione di un dominio con collegamenti di rete aperti, essi verranno conservati sulla destinazione host, ed i collegamenti SSH dovrebbero essere ancora attivi. Le regole iptables predefinite di Red Hat Virtualization non permettono collegamenti di migrazione in entrata. Per abilitarli sarà necessario creare regole iptables specifiche.

È possibile utilizzare il comando `xm migrate` per eseguire un processo di migrazione offline:

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

È possibile utilizzare il comando `xm migrate` per eseguire un processo di migrazione live:

```
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.

CAPITOLO 12. CONFIGURAZIONE PER UN UTILIZZO SU DI UNA RETE

L'integrazione di Red Hat Virtualization sulla vostra architettura di rete è un processo molto complicato che dipende dal tipo di infrastruttura. Esso potrebbe richiedere una configurazione personalizzata, in modo da impiegare interfacce ethernet multiple, ed una impostazione del bridging.

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 gestisce il traffico su queste interfacce virtuali utilizzando le convenzioni standard di Linux per il bridging, il routing, il rate limiting ecc. Il demone **xend** utilizza due script della shell per eseguire una configurazione iniziale della vostra rete, e delle nuove interfacce virtuali. I suddetti script configurano un bridge singolo per tutte le interfacce virtuali. È possibile configurare un routing ed un bridging aggiuntivi personalizzando questi script.

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 — Ogni qualvolta **xend** viene avviato o arrestato, il suddetto script inializza o arresta la rete virtuale. Successivamente, l'inizializzazione della configurazione crea il bridge **xen-br0**, spostando **eth0** sul bridge in questione , modificando conseguentemente il routing. Quando **xend** abbandona il processo esistente, il bridge verrà cancellato e **eth0** rimosso, ripristinando così l'IP originale e la configurazione del routing.

vif-bridge rappresenta uno script invocato per ogni interfaccia virtuale presente sul dominio. Configura le regole firewall ed è in grado di aggiungere **vif** al bridge appropriato.

Sono disponibili altri script da poter utilizzare durante l'impostazione di Red Hat Virtualization, come ad esempio **network-route**, **network-nat**, **vif-route**, e **vif-nat**. Questi script possono essere sostituiti anche con varianti appartenenti alla virtualizzazione.

CAPITOLO 13. COME RENDERE SICURO 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:

- Esecuzione del numero più basso di servizi necessari. Non è consigliabile includere un numero troppo elevato di compiti e servizi nel domain0. Minor numero di compiti e servizi in esecuzione presenti all'interno di domain0, più elevato è il livello di sicurezza.
- Abilitate SeLINUX in modo da rendere sicuro domain0.
- Utilizzate un firewall per limitare il traffico per il domain0. È possibile impostare un firewall con regole default-reject il quale assicura una protezione nei confronti del domain0. È altresì importante limitare l'esposizione della rete ai servizi.
- Non permettete ad utenti normali di accedere al domain0. Se abilitate il loro accesso potrete correre il rischio di rendere domain0 vulnerabile. Ricordate, domain0 risulta essere privilegiato, e garantire account non privilegiati potrebbe compromettere il livello di sicurezza.

CAPITOLO 14. STORAGE

Sono disponibili diversi metodi attraverso i quali è possibile gestire lo storage di macchine virtuali. È possibile esportare un dispositivo a blocchi fisico del domain0 (disco fisso o partizione), su di un dominio guest come un virtual block device (VBD). Potete altresì eseguire un processo di esportazione direttamente da una immagine partizionata come file-backed VBD. Red Hat Virtualization abilita per default durante il processo d'installazione sia LVM che blktap. Sarà possibile impiegare protocolli di rete standard come NFS, CLVM, o iSCSI per fornire storage a macchine virtuali.

CAPITOLO 15. GESTIONE DELLE MACCHINE VIRTUALI CON VIRSH

È possibile utilizzare l'applicazione **virsh** per gestire le macchine virtuali. Questa utility è stata creata considerando l'API di gestione libvirt, e funziona come alternativa al tool **xm** o al Virtual Machine Manager grafico. Gli utenti non privilegiati possono usare la suddetta utility per operazioni di sola lettura. Se desiderate eseguire **xend/qemu**, allora dovrete abilitare **xend/qemu** in modo da eseguirlo come un servizio. Dopo aver modificato il file di configurazione corrispondente riavviate il sistema, in questo modo **xend/qemu** verrà eseguito come un servizio. Utilizzate **virsh** per far funzionare **vm**. In modo simile al tool **xm**, è possibile eseguire **virsh** dalla linea di comando.

15.1. COLLEGAMENTO AD UN HYPERVISOR

È possibile utilizzare **virsh** per inizializzare una sessione hypervisor:

```
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. CREAZIONE DI UNA MACCHINA VIRTUALE

È possibile creare una nuova sessione della macchina virtuale, direttamente da una definizione della macchina XML. Se siete in possesso di un guest già esistente creato precedentemente con il tool **xm**, allora sarà possibile creare una macchina virtuale corrispondente:

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

15.3. CONFIGURAZIONE DI UN XML DUMP

È possibile utilizzare **virsh** per eseguire un dump dei dati per una macchina virtuale esistente.

```
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. SOSPENSIONE DI UNA MACCHINA VIRTUALE

È possibile utilizzare **virsh** per sospendere un dominio:

```
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. RIPRISTINO DI UNA MACCHINA VIRTUALE

Per ripristinare una macchina virtuale sospesa è possibile utilizzare **virsh**:

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

Questa operazione è immediata ed i parametri della macchina virtuale vengono preservati in un ciclo **suspend** e **resume**.

15.6. SALVATAGGIO DI UNA MACCHINA VIRTUALE

Per salvare uno stato corrente della macchina virtuale su di un file utilizzate **virsh** :

```
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. RIPRISTINO DI UNA MACCHINA VIRTUALE

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. ARRESTO DI UNA MACCHINA VIRTUALE

È possibile utilizzare **virsh** per arrestare una macchina virtuale

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

Potete controllare il comportamento della macchina virtuale durante l'avvio, modificando il parametro **on_shutdown** del file **xmdomain.cfg**.

15.9. RIAVVIO DI UNA MACCHINA VIRTUALE

Per poter riavviare una macchina virtuale utilizzate **virsh** :

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

Potete controllare il comportamento della macchina virtuale durante l'avvio, modificando il parametro **on_reboot** del file **xmdomain.cfg**.

15.10. COME TERMINARE UN DOMINIO

Per terminare una macchina virtuale utilizzate **virsh**:

```
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. CONVERSIONE DI UN DOMAIN NAME IN UN DOMAIN ID

Per convertire un domain name o UUID in un domain id utilizzate **virsh**:

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

15.12. CONVERSIONE DI UN DOMAIN ID IN UN DOMAIN NAME

Per convertire un domain id o UUID in un domain name, utilizzate **virsh**:

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

15.13. CONVERSIONE DI UN DOMAIN NAME IN UN UUID

Per convertire un domain name in un UUID, utilizzate **virsh** :

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

15.14. VISUALIZZAZIONE INFORMAZIONI DELLA MACCHINA VIRTUALE

Per visualizzare le informazioni relative ad una macchina virtuale specifica identificata dal proprio domain ID, domain name, o UUID, utilizzate **virsh**:

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

15.15. VISUALIZZAZIONE INFORMAZIONI DEL NODO

Per visualizzare le informazioni del nodo, utilizzate **virsh**:

```
virsh nodeinfo
```

Gli output riportano le informazioni seguenti:

```
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
```

Così facendo vengono riportate le informazioni del nodo, e le macchine che supportano il processo di virtualizzazione.

15.16. VISUALIZZAZIONE DELLE MACCHINE VIRTUALI

Per poter visualizzare l'elenco della macchina virtuale insieme con lo stato corrente, utilizzate **virsh**:

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

L'opzione **--inactive** elenca i domini inattivi (i domini definiti ma non ancora attivi). Il **-- -all** domain elenca tutti i domini attivi o meno. Il vostro output dovrebbe somigliare a quanto segue:

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

Di seguito sono riportati i sei stati relativi al dominio:

```
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. VISUALIZZAZIONE INFORMAZIONI DELLA CPU VIRTUALE

Per visualizzare le informazioni della CPU virtuale da una macchina virtuale, utilizzate **virsh** :

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

15.18. CONFIGURAZIONE AFFINITÀ CPU VIRTUALE

Per configurare l'affinità delle CPU virtuali con CPU fisiche, utilizzate **virsh**:

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

Dove **[vcpu]** è il numero della VCPU virtuale e **[cpulist]** elenca il numero fisico delle CPU.

15.19. CONFIGURAZIONE CONTEGGIO CPU VIRTUALE

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

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

Da notare che il nuovo conteggio non può eccedere la quantità specificata durante la creazione della macchina virtuale.

15.20. CONFIGURAZIONE ASSEGNAZIONE DELLA MEMORIA

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. CONFIGURAZIONE MEMORIA MASSIMA

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. GESTIONE RETI VIRTUALI

Utilizzate **virsh** per gestire le reti virtuali. Per un loro elenco:

```
virsh net-list
```

Questo comando genera un output simile a:

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

Per visualizzare le informazioni di rete per una rete virtuale specifica:

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

Così facendo verranno mostrate le informazioni relative ad una rete virtuale specifica in formato XML:

```
[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>
```

Altri comandi **virsh** utilizzati per la gestione di reti virtuali sono:

- **virsh net-autostart [nome rete]** — Avvia automaticamente una rete specificata come [nome rete]
- **virsh net-create [XML file]** — Genera ed avvia una nuova rete utilizzando un file XML già esistente
- **virsh net-define [file XML]** — Genera una nuova rete da un file XML preesistente senza avviarlo
- **virsh net-destroy [nome della rete]** — Annulla una rete specificata come [nome della rete]
- **virsh net-name [UUID della rete]** — Converte un [UUID della rete] specifico in un nome
- **virsh net-uuid [nome della rete]** — Converte un [nome della rete] specificato in un UUID
- **virsh net-start [nome di una rete inattiva]** — Avvia una rete inattiva precedentemente non definita
- **virsh net-undefine [nome di una rete inattiva]** — Rimuove definizione di una rete inattiva

CAPITOLO 16. GESTIONE MACCHINE VIRTUALI UTILIZZANDO XEND

Il demone di controllo del nodo **xend**, esegue alcune funzioni di gestione del sistema che si riferiscono alle macchine virtuali. Questo demone controlla le risorse virtualizzate, ed è necessario che **xend** sia in esecuzione per interagire con le macchine virtuali. Prima di avviare **xend** specificare i parametri operativi modificando il file di configurazione di **xend**, **xend-config.sxp**, il quale è situato all'interno della directory **etc/xen**. Ecco i parametri da abilitare o disabilitare nel file di configurazione **xend-config.sxp**:

Tabella 16.1. Parametri configurazione di xend per Red Hat Virtualization

Oggetto	Descrizione
console-limit	Determines the console server's memory buffer limit and assigns values on a per-domain basis
min-mem	Determina il numero minimo di megabyte riservati per il domain0 (se inserite 0, il valore non varia)
dom0 cpus	Determina il numero delle CPU utilizzate dal domain0 (almeno 1 CPU è assegnata per default)
enable-dump	Determina la presenza di un arresto inaspettato 'crash', e abilita un dump (il default è 0)
external-migration-tool	Determina lo script o l'applicazione che gestisce la migrazione del dispositivo esterno (gli script devono risiedere in etc/xen/scripts/external-device-migrate)
logfile	Determina la posizione del file di log (il default è /var/log/xend.log)
loglevel	Filtra i valori log mode: DEBUG, INFO, WARNING, ERROR, o CRITICAL (il default è DEBUG)
network-script	Determina lo script che abilita l'ambiente di networking (gli script devono risiedere nella directory etc/xen/scripts)
xend-http-server	Abilita il server per l'http stream packet management (il default è no)
xend-unix-server	Abilita il server unix domain socket (un server socket rappresenta un punto finale delle comunicazioni, che gestisce i collegamenti low level network ed accetta o rifiuta i collegamenti in entrata)
xend-relocation-server	Abilita il server di riposizionamento per migrazioni 'cross-machine' (il default è no)

Oggetto	Descrizione
xend-unix-path	Determina la posizione dove il comando xend-unix-server esegue l'output dei dati (il default è var/lib/xend/xend-socket)
xend-port	Determina la porta utilizzata dal server di gestione http (il default è 8000)
xend-relocation-port	Determina la porta utilizzata dal server di riposizionamento (il default è 8002)
xend-relocation-address	Determina gli indirizzi della macchina virtuale abilitati per la migrazione del sistema
xend-address	Determina l'indirizzo al quale si lega il server del domain socket.

Dopo aver impostato i parametri operativi, verificate la corretta esecuzione di xend, ed in caso contrario, inizializzate il demone. Al prompt del comando avviate il demone **xend** inserendo quanto segue:

```
service xend start
```

Utilizzate **xend** per arrestare il demone:

```
service xend stop
```

Questo comando arresterà l'esecuzione del demone.

Per riavviare il demone utilizzate **xend**:

```
service xend restart
```

Per riavviare il demone.

Controllare lo stato del demone **xend**.

```
service xend status
```

The output displays the daemon's status.

CAPITOLO 17. GESTIONE DELLE MACCHINE VIRTUALI UTILIZZANDO XM

L'applicazione **xm** è un tool di gestione molto robusto che vi permette di configurare il vostro ambiente di virtualizzazione di Red Hat. Come prerequisito per l'utilizzo di **xm**, è necessario assicurarsi che il demone **xend** presente sul vostro sistema sia in esecuzione.

17.1. FILE DI CONFIGURAZIONE XM

I parametri operativi da modificare risiedono all'interno del file **xmdomain.cfg**, il quale è situato all'interno della directory **etc/xen**. È possibile abilitare o disabilitare i parametri all'interno del file di configurazione **xmdomain.cfg**:

Tabella 17.1. Il file di configurazione xmdomain.cfg

Oggetto	Descrizione
kernel	Determina il percorso corretto per l'immagine del kernel
ramdisk	Determina il percorso corretto per initrd per la ramdisk iniziale
memoria	Determina la quantità di RAM (in MB) da assegnare al dominio al momento del suo avvio
nome	Determina il nome unico per un dominio
root	Determina il dispositivo root per un dominio
nic	Determina il numero di schede per l'interfaccia di rete per un dominio (il default è 1)
disco	Determina gli array delle stanza del dispositivo a blocchi — le tre Stanza sono: <ul style="list-style-type: none"> • mode - modalità di accesso al dispositivo • backend-dev - il dominio backend esportabile al dominio guest • frontend-dev - determina il modo attraverso il quale deve apparire un dispositivo in un dominio guest
vif	Determina gli array delle stanza dell'interfaccia virtuale (ogni stanza rappresenta un set di operazioni nome=valore).
builder	Determina il builder per la creazione del dominio (il default è linux)

Oggetto	Descrizione
cpu	Determina il conteggio CPU per mezzo del quale viene avviato il dominio. 0 indica la prima CPU, 1 la seconda, ecc. (il default è -1)
cpus	Determines which CPUs on the domain's VCPUs are executable
extra	Determina le informazioni aggiuntive da aggiungere alla fine della riga di parametro del kernel
nfs_server	Determina l'indirizzo IP del server NFS da utilizzare per il dispositivo root
nfs_root	Determina la directory root come percorso corretto per il server NFS
vcpus	Determina il numero di CPU virtuali da assegnare ad un dominio (il default è 1)
on_shutdown	Determina il parametro d'interruzione il quale aziona un processo di spegnimento corretto 'graceful-shutdown' (o spegnimento xm) all'interno di DomU
on_reboot	Determina il parametro d'interruzione del dominio il quale aziona un processo di riavvio corretto 'graceful-reboot' (o un xm reboot) all'interno di DomU
on_crash	Determina il parametro d'interruzione del dominio che genera interruzioni inaspettate di DomU .

17.1.1. Configurazione di vfb

A **vfb** is a virtual frame buffer that is defined as a 'stanza'. The stanza represents a set of name = value options, which when integrated into the **xmdomain.cfg.5** file, must be separated by commas. The **vfb** entry in your configuration file resembles:

```
vfb = [ "stanza" ] "name1=value1, name2=value2, "
```

È possibile altresì configurare il vostro ambiente **vfb** integrando le opzioni mostrate nella Tabella 16.2:

Tabella 17.2. Opzioni di configurazione di vfb

Oggetto	Descrizione
tipo	Il tipo di opzione vnc inizializza una sessione del server VNC, la quale si collega ad un visualizzatore VNC esterno. L'opzione sdl inizializza il visualizzatore interno.

Oggetto	Descrizione
<code>vncdisplay</code>	Determina il numero del display VNC da utilizzare (il default per il valore ID del dominio). Il server VNC risulterà in ascolto sulla porta 5900 + il numero del display.
<code>vnclisten</code>	The VNC server's listening address (defaults to 127.0.0.1).
<code>vncunused</code>	Determina il valore numerico, e se risulta essere un valore diverso da zero abilita il server VNC, in modo da risultare in ascolto per la prima porta non utilizzata oltre 5900.
<code>vncpasswd</code>	Annula la password predefinita configurata da Xend.
<code>display</code>	Abilita il display per il visualizzatore interno da utilizzare (il default è la variabile di ambiente DISPLAY).
<code>xauthority</code>	Abilita il file authority per il visualizzatore interno da utilizzare (il default è la variabile di ambiente XAUTHORITY).

17.2. CREAZIONE E GESTIONE DEI DOMINI CON XM

È possibile utilizzare l'applicazione **xm** per creare e gestire i domini.

17.2.1. Collegamento ad un dominio

È possibile utilizzare **xm** per collegarsi ad un dominio o macchina virtuale:

```
xm console domain-id
```

This causes the console to attach to the **domain-id**'s text console.

17.2.2. Creazione di un dominio

Per creare un dominio è possibile utilizzare **xm**:

```
xm create domain001 [-c]
```

Ciò crea un dominio chiamato **domain001**, con il file situato all'interno della directory **/etc/xen/** . L'opzione **[-c]** è utile durante il processo di troubleshooting, permettendovi di collegarvi alla console di testo.

17.2.3. Come salvare un dominio

È possibile utilizzare **xm** per salvare un dominio:

```
xm save [domain-id] [statefile]
```

-

17.2.4. Come terminare un Domain ID

Per terminare un domain-id è possibile utilizzare **xm**:

```
xm destroy [domain-id]
```

Ciò termina istantaneamente il **domain-id**. Se preferite un altro metodo per terminare in modo sicuro la vostra sessione, potrete utilizzare il parametro **shutdown**.

17.2.5. Come arrestare un dominio

Per arrestare qualsiasi dominio potete utilizzare **xm**:

```
xm shutdown [domain-id] [ -a | -w ]
```

L'opzione [**-a**] arresta tutti i domini presenti sul vostro sistema. L'opzione [**-w**] attende l'arresto completo di un dominio.

17.2.6. Ripristino di un dominio

È possibile utilizzare **xm** per ripristinare un dominio precedentemente salvato.

```
xm restore [state-file]
```

17.2.7. Sospensione di un dominio

È possibile utilizzare **xm** per sospendere un dominio:

```
xm suspend [domain-id]
```

17.2.8. Ripristinare un dominio

Per ripristinare una sessione precedentemente sospesa è possibile utilizzare **xm**:

```
xm resume [domain-id]
```

17.2.9. Come riavviare un dominio

Utilizzate **xm** per riavviare un dominio:

```
xm reboot [domain-id] [ -a | -w ]
```

L'opzione [**-a**] riavvia tutti i domini sul vostro sistema. L'opzione [**-w**] attende il riavvio completo del dominio. È possibile controllare il comportamento di un dominio durante il suo processo d'avvio, modificando il parametro **on_boot** del file **xmdomain.cfg**.

17.2.10. Come rinominare un dominio

Utilizzate **xm** per assegnare un nuovo nome ad un dominio esistente:

```
xm rename [domain-name] [new domain-name]
```

Il processo di rinomina di un dominio manterrà le stesse impostazioni (stesso disco fisso, stessa memoria ecc.).

17.2.11. Pausa di un dominio

Utilizzate **xm** per la pausa di un dominio:

```
xm pause [domain-id]
```

17.2.12. Come rimuovere la pausa di un dominio

Utilizzate **xm** per rimuovere la pausa di un dominio:

```
xm unpause [domain-id]
```

Tale azione renderà il dominio disponibile per una programmazione da parte di un hypervisor.

17.2.13. Conversione di un Domain Name in un Domain ID

È possibile utilizzare **xm** per convertire un domain name in un domain ID:

```
xm domid [domain-name]
```

17.2.14. Conversione di un Domain ID in un Domain Name

È possibile utilizzare **xm** per convertire un domain ID in un domain name:

```
xm domname [domain-id]
```

17.2.15. Configurazione assegnazione della memoria

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

```
xm mem-set [domain-id] [count]
```



NOTA

You cannot grow a domain's memory beyond the maximum amount you specified when you first created the domain.

17.2.16. Configurazione della memoria massima

You can use **xm** to modify a domain's maximum memory:

```
xm mem-max [domain-id] [count]
```

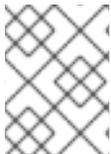
È necessario specificare **[count]** in megabytes.

17.2.17. Configurazione del conteggio VCPU

You can use **xm** to modify a domain's VCPU count:

```
xm vcpu-set [domain-id] [count]
```

È necessario specificare **[count]** in megabytes.



NOTA

You cannot grow a domain's memory beyond the maximum amount you specified when you first created the domain.

17.2.18. Come fissare una VCPU

È possibile utilizzare **xm** per fissare una VCPU:

```
xm vcpu-pin [domain-id] [vcpu] [cpus]
```

Dove **[vcpu]** è la VCPU alla quale desiderate collegarvi, e **[cpus]** è il target. Questo processo assicura l'esecuzione delle VCPU su determinate CPU.

17.2.19. Migrazione di un dominio

Utilizzate **xm** per migrare un dominio:

```
xm migrate [domain-id] [host] [options]
```

Dove **[domain-id]** è il dominio che desiderate migrare, e **[host]** è il target. **[options]** include **—live** (or **-l**) per una migrazione live, o **—resource** (or **-r**) per specificare la velocità massima della migrazione (in Mb).

Per garantire una migrazione corretta, assicuratevi che il demone **xend** sia in esecuzione su tutti i domini host. Altresì tutti gli host devono eseguire Red Hat RHEL 5.0+ e possedere porte TCP di migrazione aperte, per poter accettare i collegamenti da host sicuri.

17.3. MONITORAGGIO E DIAGNOSI

17.3.1. Esecuzione di un Core Dump

Utilizzate **xm** per eseguire un memory dump di una macchina virtuale esistente.

```
xm dump-core [-C] [domain-id]
```

This command dumps the virtual machine's memory to the **xendump** file located in the **/var/lib/xen/dump/** directory. You can terminate the virtual machine by including the **-C** option.

17.3.2. Monitoraggio dei domini in tempo reale

Utilizzate **xm** per monitorare i domini e gli host in tempo reale:

```
xm top [domain-id]
```

17.3.3. Visualizzazione degli stati del dominio

Utilizzate **xm** per visualizzare gli stati delle attività del dominio di uno o più domini:

```
xm list [domain-id] [ --long | --label]
```

È possibile specificare un determinato dominio tramite il suo nome. L'opzione **[--long]** fornisce una descrizione più dettagliata del dominio specificato. L'opzione **[--label]** aggiunge una colonna supplementare contenente lo stato dell'etichetta. L'output mostra quanto segue:

Name	ID	Mem(MiB)	VCPUs	State
Domain0	0	927	8	
r----- 204.9	INACTIVE			
Domain202	1	927	8	s-----
205.0/command	ACTIVE			
DomainQ/A	2	927	8	b-----
INACTIVE				
Domain9600	3	927	8	c
----- 205.1	ACTIVE			

Di seguito vengono riportati i sei stati del dominio per VCPU:

Tabella 17.3. Stati del dominio

Stato	Descrizione
in esecuzione	elenca i domini attualmente attivi su di una CPU
bloccato	elenca i domini bloccati (un dominio risulta bloccato quando la vcpu è in attesa che si verifichi un evento esterno)
sospeso	elenca i domini sospesi
arrestati	elenca i domini in procinto di essere arrestati
interrotti	elenca i domini completamente inattivi.
arrestati inaspettatamente	elenca i domini arrestati inaspettatamente

Stato	Descrizione
inattivi	elenca i domini che sono istanze inattive
—tutti	elenca i domini attivi e le istanze vcpu inattive

17.4. VISUALIZZAZIONE UPTIME

È possibile utilizzare **xm** per visualizzare l'uptime:

```
xm uptime [domain-id]
```

L'output mostra:

Name	ID	Uptime
Domain0	0	4:45:02
Domain202	1	3:32:00
Domain9600	2	0:09:14
DomainR&D	3	2:21:41

17.5. VISUALIZZAZIONE INFORMAZIONI DELLA VCPU

È possibile utilizzare **xm** per visualizzare le informazioni relative alla CPU del dominio:

```
xm vcpu-list [domain-id]
```

È necessario specificare quale vcpu desiderate elencare. In caso contrario, verranno visualizzate le vcpu per tutti i domini.

17.6. VISUALIZZAZIONE INFORMAZIONI DEL DOMINIO

È possibile utilizzare **xm** per visualizzare le informazioni relative al dominio dell'host:

```
xm info
```

L'output mostra:

```
host : redhat83-
157.brisbane.redhat.com
release : 2.6..18-1.2714.e15xen
version : #1 SMP Mon Oct 21
17:57:21 EDT 2006
machine : x86_64
nr_cpus : 8
nr_nodes : 1
sockets_per_node : 2
cores_per_socket : 2
threads_per_core : 2
```

```

cpu_mhz                : 2992
hw_caps                :
bfeebbef:20100000:00000000:00000000
total_mememory        : 1022
free_memory           : 68
xen_major              : 3
xen_minor              : 0
xen_extra              : -unstable
xen_caps               : xen-3.0-x86_84
xen_pagesize          : 4096
platform_params       :
virt_start=0xffff880000000000000000000000
xen_changeset         : unavailable
cc_compiler            : gcc compiler version
4.1.1 200060928
cc_compile_by         : brewbuilder
cc_compile_domain     : build.redhat.com
cc_compile_date       : Mon Oct 2 17:00 EDT 2006
xend_config_format    : 2

```

17.7. VISUALIZZAZIONE DISPOSITIVI TPM

È possibile utilizzare **xm** per visualizzare i dispositivi TPM virtuali:

```
xm vtpm-list [domain-id] [--long]
```

L'opzione **[--long]** fornisce una descrizione più dettagliata del dominio specificato.

17.8. VISUALIZZAZIONE DEL LOG DI XEND

Utilizzate **xm** per visualizzare i contenuti del log di **xend**:

```
xm log
```

L'output visualizza l'attività del log di **xend**.

17.9. VISUALIZZAZIONE DEL MESSAGE BUFFER

Utilizzate **xm** per visualizzare xend message buffer:

```
xm dmesg
```

L'output visualizza i contenuti del message buffer di **xend**.

17.10. VISUALIZZAZIONE DELLE INFORMAZIONI SULLO STATO ACM

È possibile utilizzare **xm** per visualizzare le informazioni sullo stato ACM dell'hypervisor:

```
xm dumppolicy [policy.bin]
```

17.11. VISUALIZZAZIONE DEI DISPOSITIVI A BLOCCHI VIRTUALI

Per visualizzare i dispositivi a blocchi virtuali per un dominio, utilizzate **xm**:

```
xm block-list [domain-id] [ --long]
```

L'output visualizza i dispositivi a blocchi per il dominio specificato.

17.12. VISUALIZZAZIONE DELLE INTERFACCE DI RETE VIRTUALI

Per visualizzare i dispositivi di rete virtuali per un dominio, utilizzate **xm**:

```
xm network-list [domain-id] [ --long]
```

L'output visualizza le interfacce di rete per il dominio da voi specificato.

17.13. CREAZIONE DI UN NUOVO DISPOSITIVO DI RETE VIRTUALE

È possibile utilizzare **xm** per creare un nuovo dispositivo di rete virtuale:

```
xm network-attach [domain-id] [script=scriptname] [ip=ipaddr] [mac-  
macaddr] [bridge=bridge-name] [backend=bedomain-id]
```

Di seguito vengono definite le cinque opzioni del parametro:

Tabella 17.4. Parametri

Parametro	Descrizione
[script=scriptname]	Utilizza il nome dello script specificato per attivare la rete
[ip=ipaddr]	Passa il nome dello script specificato all'adattatore
[mac-macaddr]	L'indirizzo MAC visto dal dominio sul proprio dispositivo ethernet
[bridge-bridgename]	Il nome del dispositivo al quale collegare vif
[backend=bedomain-id]	Il domain id di backend.

17.14. CREAZIONE DI UNA ETICHETTA DI SICUREZZA DEL DOMINIO

Utilizzate **xm** per creare una etichetta di sicurezza del dominio:

```
xm addlabel [labelname] [domain-id] [configfile]
```

17.15. TEST DELLE RISORSE DEL DOMINIO

Per controllare se il dominio è in grado di accedere alle proprio risorse utilizzate **xm**:

```
xm dry-run [configfile]
```

Così facendo controllerete ogni risorsa elencata nel vostro configfile. Elenca lo stato di ogni risorsa e la decisione finale per la sicurezza.

17.16. VISUALIZZAZIONE RISORSE DEL SISTEMA

Utilizzate **xm** per visualizzare le risorse del sistema:

```
xm resources
```

L'output visualizza le risorse per i domini presenti sul vostro sistema.

17.17. CONFIGURAZIONE CREDIT SCHEDULING

È possibile utilizzare **xm** per configurare i parametri del credit scheduler:

```
xm sched-credit -d <domain> [ -w [=WEIGHT] | -c [CAP] ]
```

Potete configurare Weight con l'opzione [**-w**], e Cap con l'opzione [**-c**].

17.18. CREAZIONE DI UN NUOVO DISPOSITIVO A BLOCCHI VIRTUALE

È possibile utilizzare **xm** per creare un nuovo dispositivo a blocchi virtuale:

```
xm block-attach [domain-id] [bedomain-id] [fe-dev] [be-dev] [mode]
```

È possibile collegare (o scollegare) i dispositivi virtuali anche se i guest sono in esecuzione. Vengono di seguito definite le cinque opzioni di parametro:

Tabella 17.5. Nuovi parametri del dispositivo a blocchi

Parametro	Descrizione
[domain-id]	The guest domain's domain-id that attaches to the device
[be-dev]	Il dispositivo nel dominio di backend esportato

Parametro	Descrizione
[fe-dev]	Il dispositivo presentato al dominio guest
[mode]	The guest domain's device access mode
[bedomain-id]	Il dominio di backend che ospita il dispositivo

17.19. COME TERMINARE UN DISPOSITIVO A BLOCCHI VIRTUALE

Per terminare un dispositivo a blocchi virtuale esistente utilizzate **xm**:

```
xm block-detach [domain-id] [DevID]
```

Tale azione terminerà il dispositivo a blocchi virtuale da voi specificato.

17.20. SICUREZZA

17.20.1. Rimozione di una etichetta di sicurezza del dominio

È possibile utilizzare **xm** per rimuovere una etichetta di sicurezza del dominio:

```
xm rmlabel [domain-id] [configfile]
```

Tale operazione rimuove la voce dell'etichetta **acm_policy** dal configfile.

17.20.2. Creazione di una etichetta di sicurezza delle risorse

Per creare una etichetta di sicurezza delle risorse utilizzate **xm**:

```
xm addlabel [labelname] res [resource] [policy]
```

17.20.3. Rimozione di una etichetta di sicurezza delle risorse

È possibile utilizzare **xm** per rimuovere una etichetta di sicurezza delle risorse:

```
mx rmlabel [domain-id] res [resource]
```

Tale operazione rimuoverà il file di risorse globale.

17.20.4. Configurazione controllo accesso

Il controllo dell'accesso di Red Hat Virtualization consiste in due componenti principali. L'Access Control Policy (ACP) definisce le regole di accesso e le etichette di sicurezza. Quando i domini richiedono una comunicazione d'accesso, l'Access Control Module (ACM) interpreta la policy e gestisce le decisioni per

il controllo dell'accesso. L'ACM determina i diritti d'accesso tramite l'etichetta di sicurezza del dominio. Successivamente l'ACP abilita le etichette di sicurezza e le regole d'accesso assegnandole ai domini ed alle risorse. L'ACP utilizza due modi diversi per la gestione dell'etichetta:

Tabella 17.6. Gestione etichetta ACP

Etichetta	Descrizione
Simple Type Enforcement	L'ACP interpreta le etichette assegnando le richieste d'accesso ai domini che richiedono un accesso virtuale (o fisico). La policy di sicurezza controlla l'accesso tra i domini, assegnando le etichette corrette ai rispettivi domini. Per default l'accesso ai domini con un Simple Type Enforcement non è abilitato.
Chinese Wall	La policy di sicurezza di tipo Chinese Wall controlla e risponde alle richieste d'accesso provenienti da un dominio.

Una policy è un elenco separato di nomi tradotto in un percorso locale, il quale indica il file XML della policy stessa (relativo alla directory root della policy globale). Per esempio, il file del dominio `chinese_wall.client_v1` si riferisce al file della policy `/example/chinese_wall.client_v1.xml`.

Red Hat Virtualization include i parametri in grado di abilitarvi alla gestione delle policy di sicurezza e di assegnare le etichette ai domini:

17.20.5. Creazione di una policy

Per creare una policy binaria è possibile utilizzare `xm`:

```
xm makepolicy [policy]
```

Tale operazione crea una policy binaria salvandola come file binario `[policy.bin]`.

17.20.6. Caricamento della policy

Per caricare una policy binaria è possibile utilizzare `xm`:

```
xm loadpolicy [policy.bin]
```

17.20.7. Creazione policy per una configurazione del processo d'avvio

È possibile utilizzare `xm` per creare una policy binaria ed aggiungerla al boot configuration file:

```
xm cfgbootpolicy [kernelversion]
```

Tale operazione copia la policy binaria all'interno della directory `/boot`, modificando la riga corrispondente nel file `/boot/grub/menu.1st`.

17.20.8. Creazione di una etichetta

Per creare una etichetta utilizzate **xm**:

```
xm addlabel [configfile] [policy]
```

Aggiunge una etichetta di sicurezza ad un file di configurazione del dominio. Verifica altresì che la definizione della policy corrisponda al nome dell'etichetta interessato.

17.20.9. Visualizzazione etichette della policy

Per visualizzare le etichette della policy utilizzate **xm** :

```
xm labels [policy] [type=dom | res | any]
```

Tale operazione visualizza le etichette del tipo da voi specificato (il default è **dom**), e da voi definite al momento della creazione della policy.

17.20.10. Visualizzazione delle etichette di sicurezza del dominio

Per visualizzare le etichette di sicurezza per un dominio utilizzate **xm**:

```
xm getlabel domain-id [configfile]
```

17.20.11. Visualizzazione delle etichette di sicurezza delle risorse

È possibile utilizzare **xm** per visualizzare le etichette di sicurezza per una risorsa:

```
xm getlabel res [resource]
```

17.20.12. Configurazione di Access Control Security

Per abilitare il Red Hat Virtualization access security è necessario modificare i seguenti parametri presenti all'interno di **xen_source__dir/Config.mk**

```
ACM_SECURITY ?= y
ACM_DEFAULT_SECURITY_POLICY ? =
ACM_CHINESE_WALL__AND_SIMPLE_TYPE_ENFORCEMENT_POLICY
```

17.20.13. Compilazione di una policy di sicurezza

Questo esempio dimostra come compilare in modo corretto una policy di sicurezza:

```
xm makepolicy chinesewall_ste.client_v1
```

Così facendo verranno creati i file **client_v1.map** and **client_v1.bin** all'interno della directory **/etc/xen/acm-security/policies/example/chinesewall_ste**.

17.20.14. Caricamento di una policy di sicurezza

È possibile utilizzare **xm** per attivare **client_v1.bin** :

```
xm loadpolicy example.chwall_ste.client_v1
```

17.20.15. Configurazione di una policy di sicurezza per l'avvio

È possibile utilizzare **xm** per configurare il boot loader in modo da caricare **client_v1.bin** :

```
xm cfgbootpolicy chineseWall_ste.client_v1
```

Tale operazione causerà l'utilizzo di questa etichetta da parte di ACM, per avviare Red Hat Virtualization.

17.20.16. Visualizzazione delle etichette di sicurezza

Utilizzate **xm** per visualizzare le etichette definite:

```
xm labels chineseWall_ste.client_v1 type=dom
```

L'output visualizza tutte le policy con **dom**:

```
dom_StorageDomain
dom_SystemManagement
dom_NetworkDomain
dom_QandA
dom_R&D
```

17.20.17. Come collegare una etichetta di sicurezza

È possibile utilizzare **xm** per allegare una etichetta di sicurezza ad un file di configurazione del dominio (questo esempio utilizza l'etichetta **SoftwareDev**):

```
xm addlabel myconfig.xml dom_SoftwareDev
```

Attaching the security label ensures that the domain does not share data with other non-**SoftwareDev** user domains. This example includes the **myconfig.xml** configuration file represents a domain that runs workloads related to the **SoftwareDev**'s infrastructure.

Modificate il rispettivo file di configurazione, e verificate che il comando **addlabel** abbia aggiunto correttamente la voce **access_control** (insieme ai parametri associati) alla fine del file:

```
kernel = "/boot/vmlinuz - 2.6.16 -xen"
ramdisk="/boot/U1_SoftwareDev_ramdisk.img"
memory = 164
name = "SoftwareDev"
vif = [ ' ' ]
```

```
dhcp = "dhcp"  
access_control = [policy=example.chwall_ste.client_v1,  
label=dom_SoftwareDev]
```

Se qualcosa sembra non essere corretto, effettuate le modifiche necessarie e salvate il file.

CAPITOLO 18. GESTIONE MACCHINE VIRTUALI CON VIRTUAL MACHINE MANAGER

Questa sezione descrive le finestre, le caselle di dialogo ed i vari controlli GUI di Red Hat Virtualization Virtual Machine Manager (VMM).

18.1. ARCHITETTURA DEL VIRTUAL MACHINE MANAGER

Red Hat Virtualization è una raccolta di componenti software in grado di ospitare e gestire macchine virtuali. Il Virtual Machine Manager (VMM) conferisce una visuale grafica delle macchine virtuali, presenti sul vostro sistema. È possibile utilizzare VMM per definire sia macchine para-virtuali che macchine completamente virtuali. Utilizzando il Virtual Machine Manager è possibile eseguire una serie di compiti di gestione di virtualizzazione, incluso l'assegnazione della memoria, l'assegnazione delle CPU virtuali ed il controllo delle prestazioni operative, insieme con il salvataggio, il ripristino, la sospensione, e l'arresto dei sistemi virtuali. È possibile altresì accedere la console grafica e di testo. Red Hat Virtualization separa le risorse CPU e della memoria, dalle configurazioni di rete e hardware. Ciò permette alle risorse processate, di essere raggruppate e dinamicamente assegnate alle richieste di servizi e di applicazioni. Chip-level virtualization permette ai sistemi operativi con Intel VT e AMD Pacifica hardware, di essere eseguiti sugli hypervisor.

18.2. FINESTRA APRI COLLEGAMENTO

Questa finestra apparirà per prima e richiederà all'utente di selezionare una sessione dell'hypervisor. Gli utenti non privilegiati possono iniziare una sessione di sola lettura. Gli utenti root possono avviare una sessione con un permesso di lettura-scrittura. Per gli utenti normali selezionare l'opzione **host Xen locale**. Avviate la modalità di testo della Virtual Machine Manager selezionando **Altro hypervisor**, e successivamente digitando il comando `test:///default` nel campo dell'URL sotto riportato. Una volta all'interno della modalità di prova, sarà possibile collegarsi ad un hypervisor di prova libvirt. Anche se la **schermata host Xen remoto** risulta visibile, la funzionalità per il collegamento a questo tipo di host non è implementata con RHEL 5.0.



Figura 18.1. Virtual Machine Manager Connection window

18.3. FINESTRA DEL VIRTUAL MACHINE MANAGER

Questa finestra principale visualizza tutte le macchine virtuali in esecuzione, insieme alle risorse ad esse assegnate (incluso il domain0), è altresì possibile visualizzare il campo desiderato. Facendo doppio clic sulla macchina virtuale desiderata, sarete in grado di visualizzare la console relativa alla macchina interessata. Selezionando una macchina virtuale e facendo doppio clic sul pulsante **Dettagli**, visualizzerete le informazioni relative alla macchina interessata. Per creare una nuova macchina virtuale utilizzate il menu **File**.

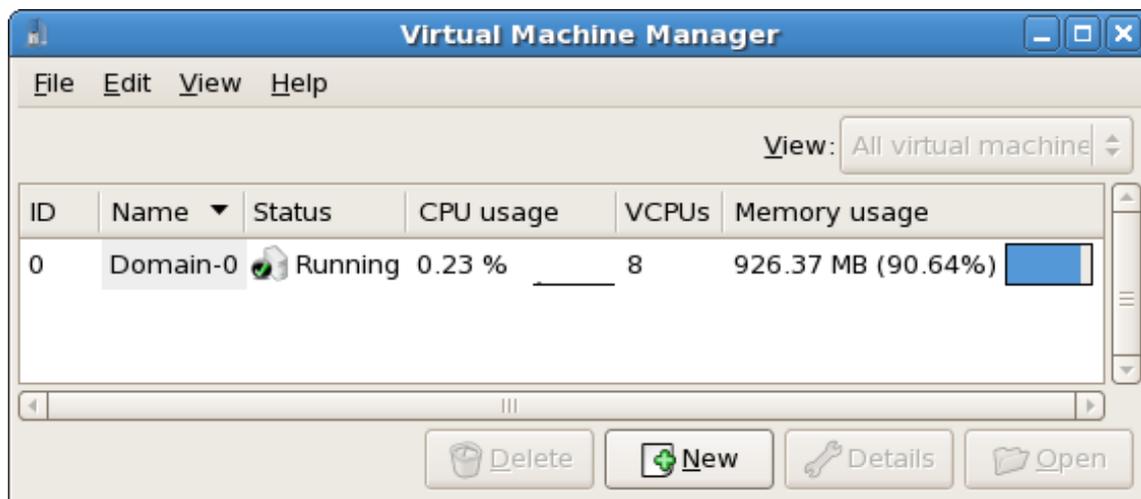


Figura 18.2. Virtual Machine Manager main window

18.4. FINESTRA DATTEGLI DELLA MACCHINA VIRTUALE

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).

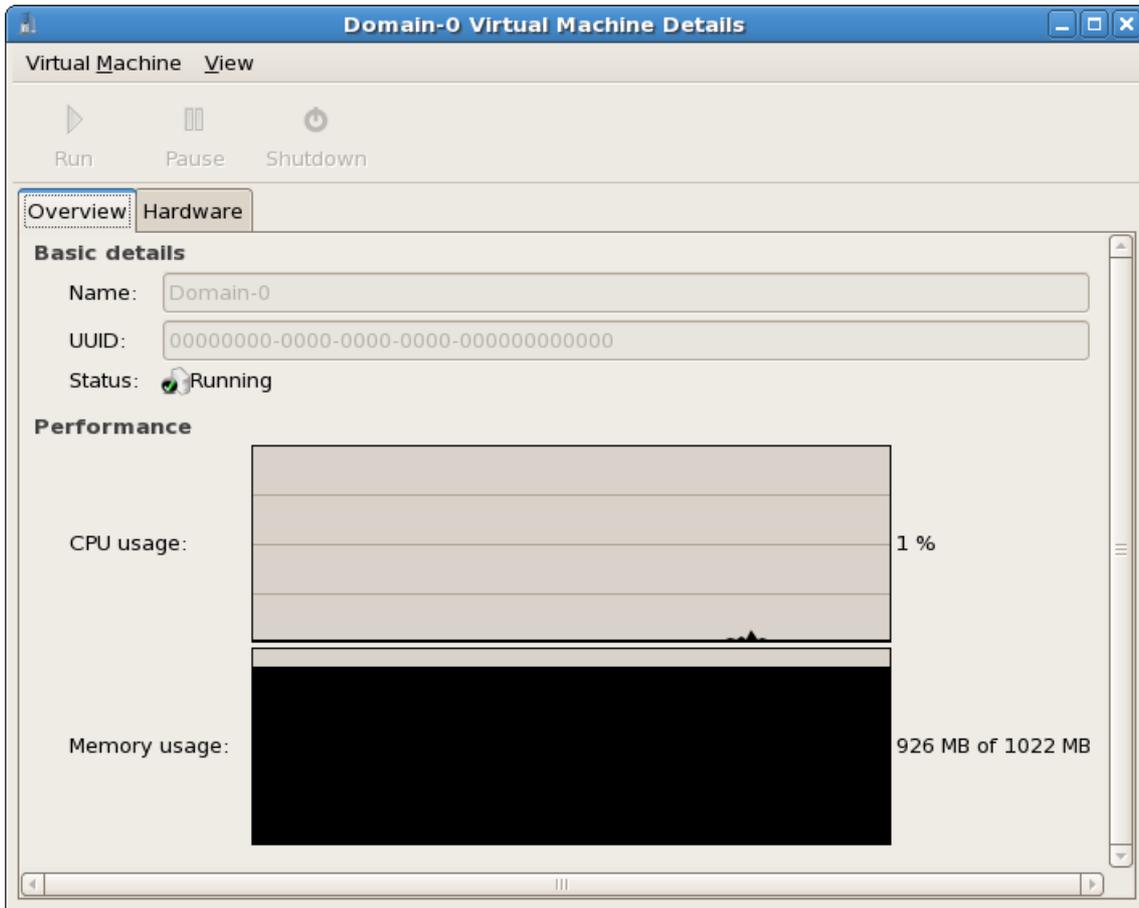


Figura 18.3. Virtual Machine Manager Details window

18.5. CONSOLE GRAFICA DELLA MACCHINA VIRTUALE

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.

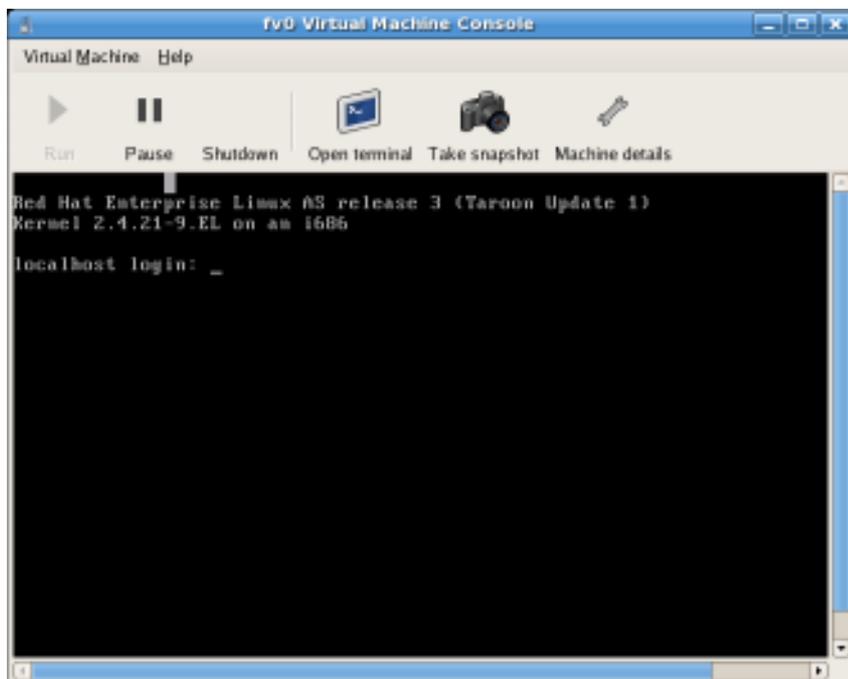


Figura 18.4. Finestra console grafica

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'.

18.6. STARTING THE VIRTUAL MACHINE MANAGER

Per avviare la sessione del Virtual Machine Manager, dal menu **Applications** fate clic su **System Tools** e selezionate **Virtual Machine Manager**.

Apparirà la finestra principale del Virtual Machine Manager.

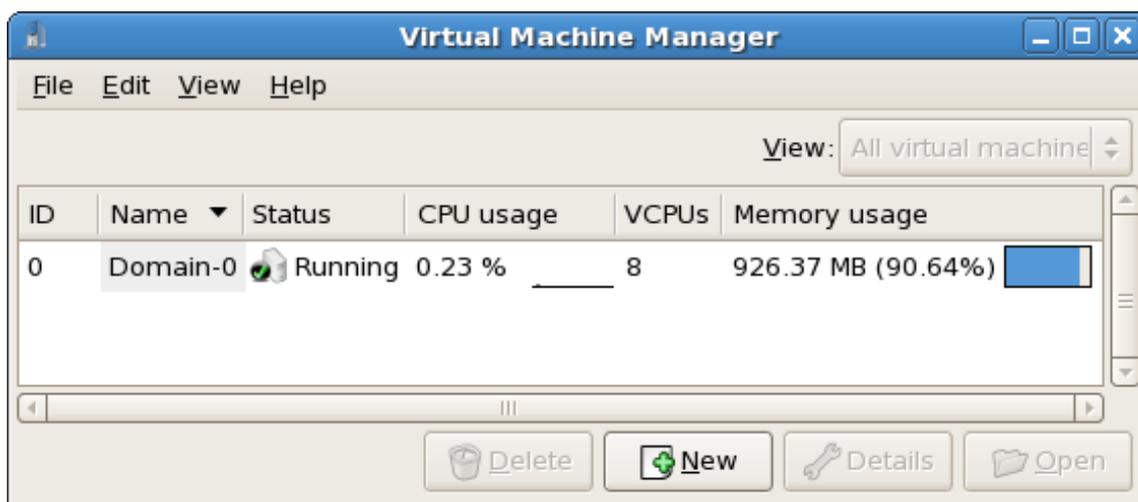


Figura 18.5. Starting the Virtual Machine Manager

18.7. CREAZIONE DI UNA NUOVA MACCHINA VIRTUALE

Virtual Machine Manager (virt-manager) è l'applicazione desktop in grado di gestire le macchine virtuali.

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

- Creare nuovi domini.
- Configurare or adjust a domain's resource allocation and virtual hardware.
- Riassumere i domini in esecuzione tramite statistiche, in tempo reale, relative all'utilizzo delle risorse e sulla prestazione.
- Visualizzare grafici in grado di mostrare le prestazioni e l'utilizzo delle risorse nei periodi desiderati.
- Usare il visualizzatore del client VNC embedded, il quale è in grado di presentare una console grafica al dominio del guest.



NOTA

È necessario installare Red Hat Enterprise Linux 5.0, virt-manager, ed i pacchetti del kernel, su tutti i sistemi che necessitano della virtualizzazione. Tutti i sistemi devono successivamente essere avviati, ed eseguire il kernel di Red Hat Virtualization.

Ecco riportate le fasi necessarie per l'installazione di un sistema operativo guest, su di un Red Hat Enterprise Linux 5 utilizzando il Virtual Machine Monitor:

Procedura 18.1. Creazione di un sistema operativo guest

1. Dal menu **Applicazioni**, selezionate **Tool del sistema** e successivamente **Virtual Machine Manager**.

Apparirà la finestra principale del Virtual Machine Manager.

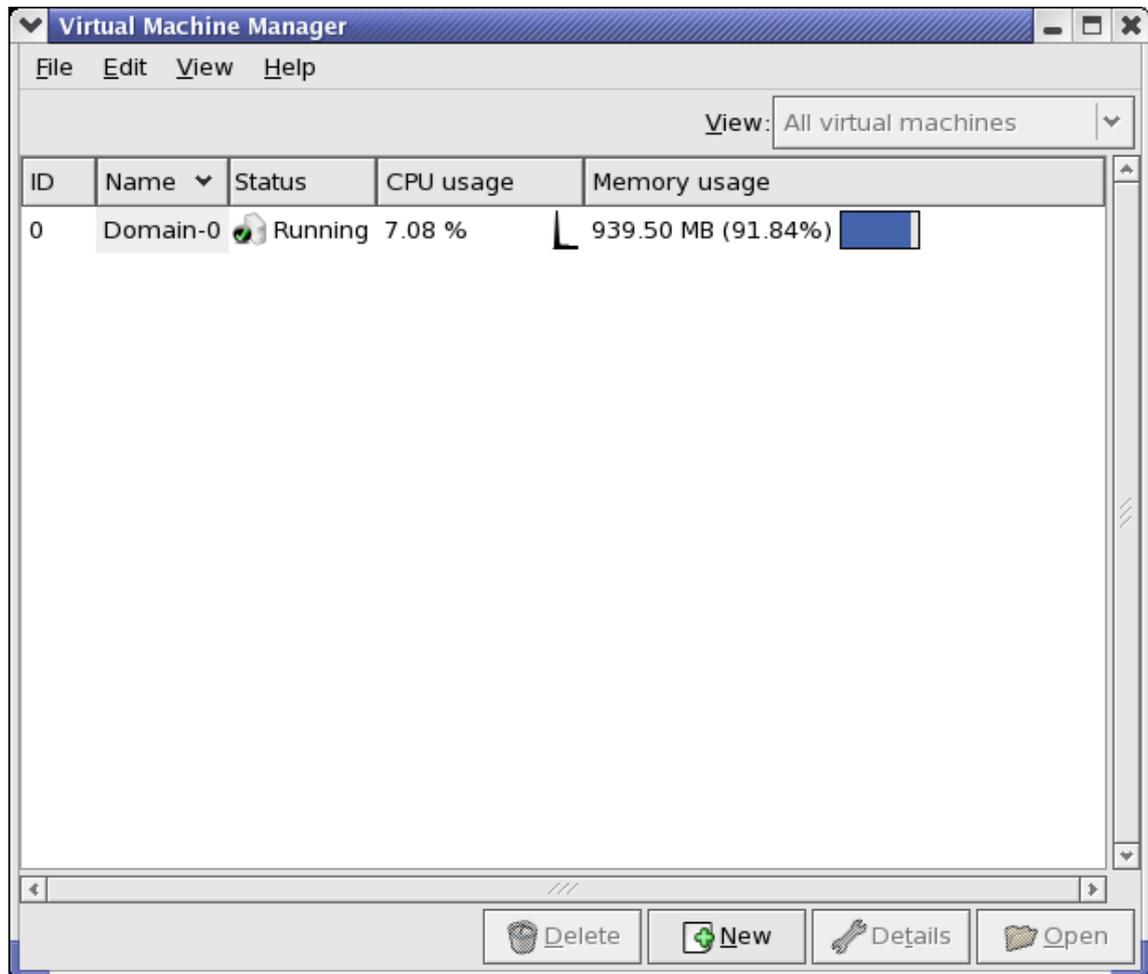


Figura 18.6. Virtual Machine Manager window

2. Dal menu **File**, selezionate **Nuova macchina**.

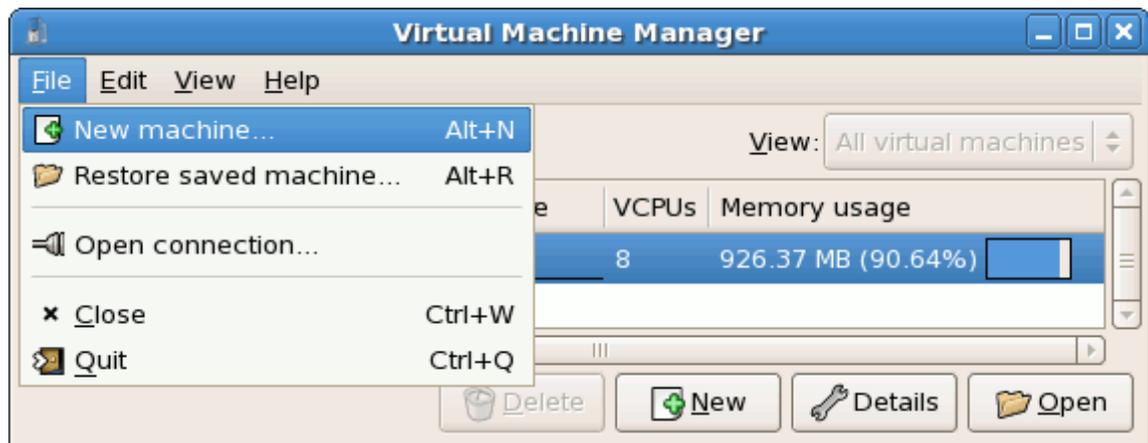


Figura 18.7. Selezione di una nuova macchina

A questo punto apparirà il wizard Creazione di un nuovo sistema virtuale.

3. Fate clic su **Avanti**.

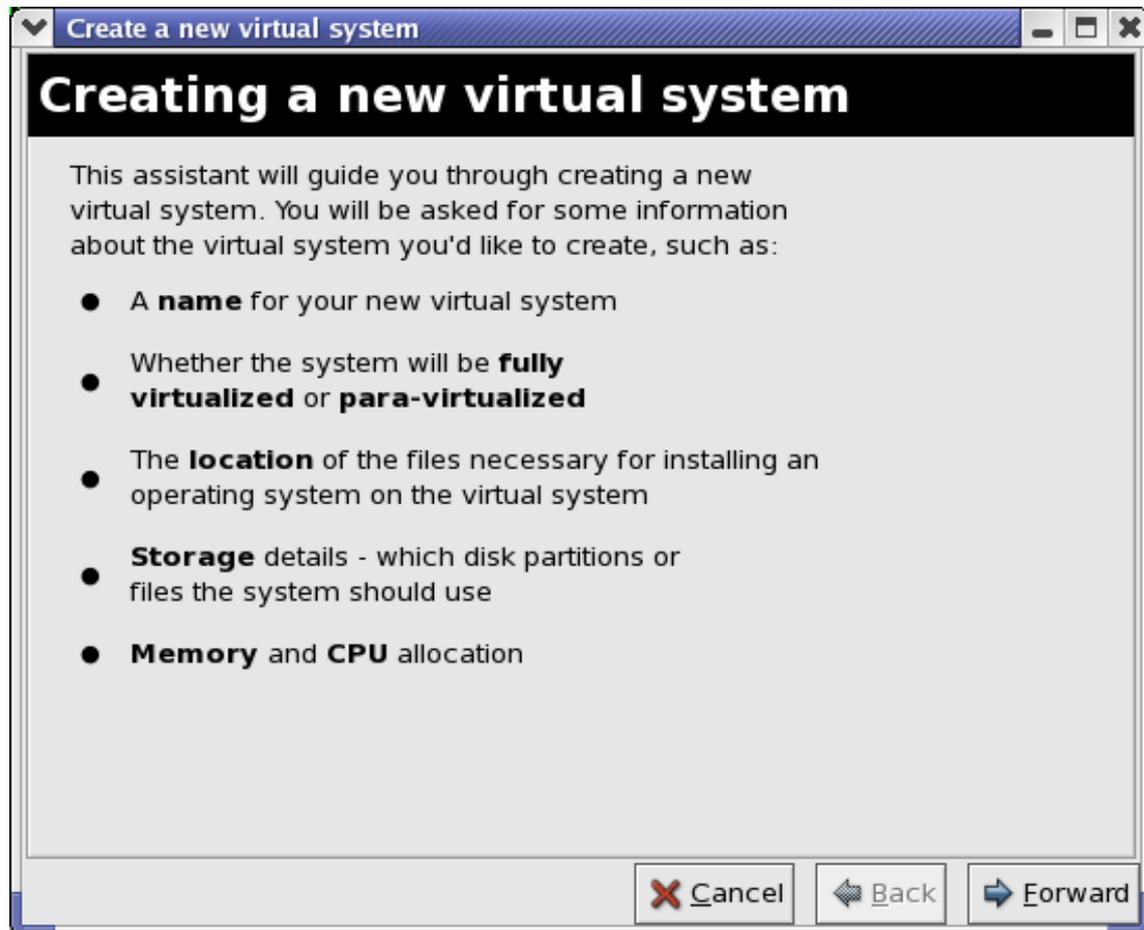


Figura 18.8. Creazione del wizard di un nuovo sistema virtuale

4. Inserite il nome del nuovo sistema virtuale e successivamente fate clic su **Avanti**.



Figura 18.9. Come nominare il sistema virtuale

5. Inserire la posizione del vostro media d'installazione. La posizione del file kickstart è facoltativa. Successivamente fate clic su **Avanti** .

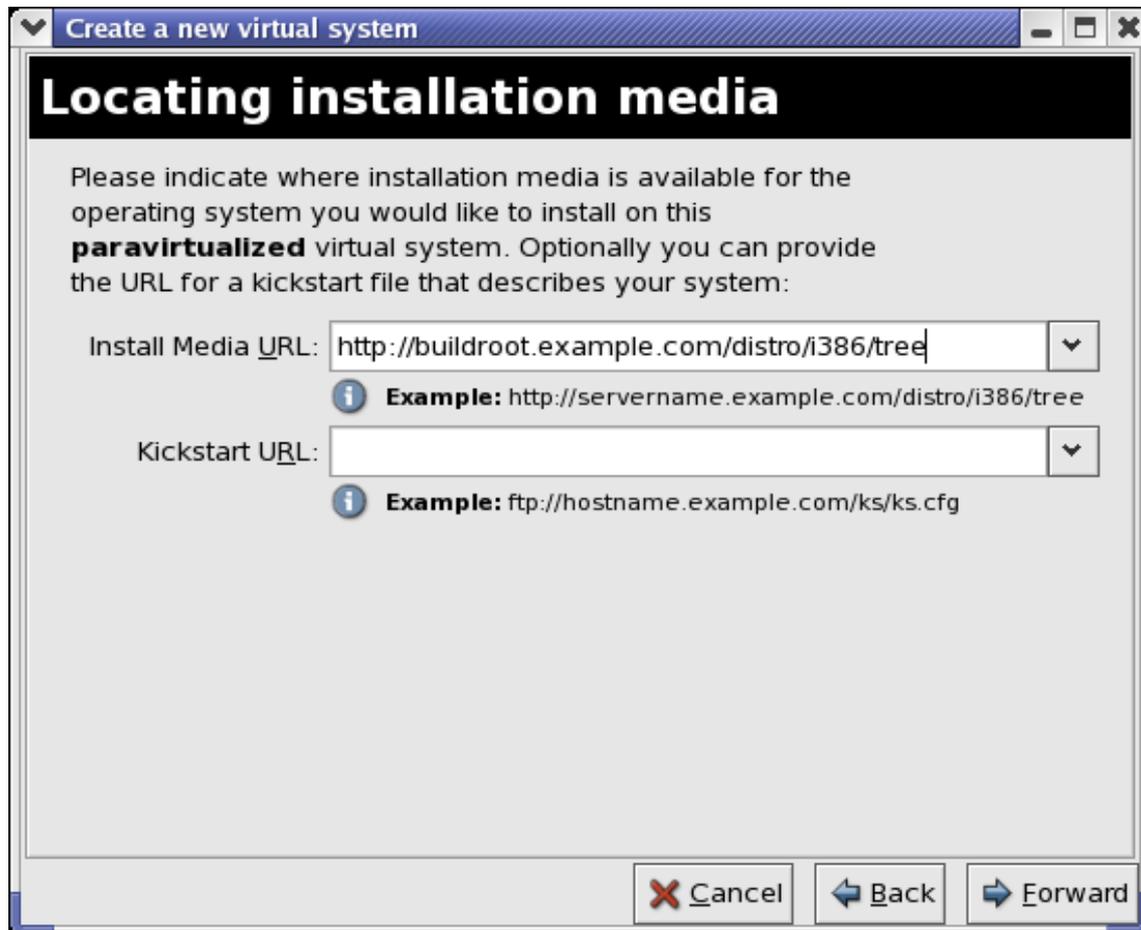


Figura 18.10. Determinare il media d'installazione

6. Eseguite una installazione su di una partizione del disco fisico, oppure su di un file system virtuale all'interno di un file.



NOTA

Questo esempio riporta l'installazione di un sistema virtuale all'interno di un file.

La policy SELinux di default permette alle immagini del disco di xen di risiedere in */var/lib/xen*. Se SELinux risulta essere abilitato, e se desiderate specificare un percorso personalizzato per il disco virtuale, allora sarà necessario modificare conseguentemente la policy di SELinux

Aprire un terminale e create la directory */xen*, impostate la policy di SELinux con il comando **restorecon -v /xen**. Specificate la posizione e la misura del disco virtuale, e successivamente fate clic su **Avanti**.



Figura 18.11. Assegnazione dello spazio di storage

7. Selezionate la memoria per assegnare il guest ed il numero di CPU virtuali, e successivamente fate clic su **Avanti**.

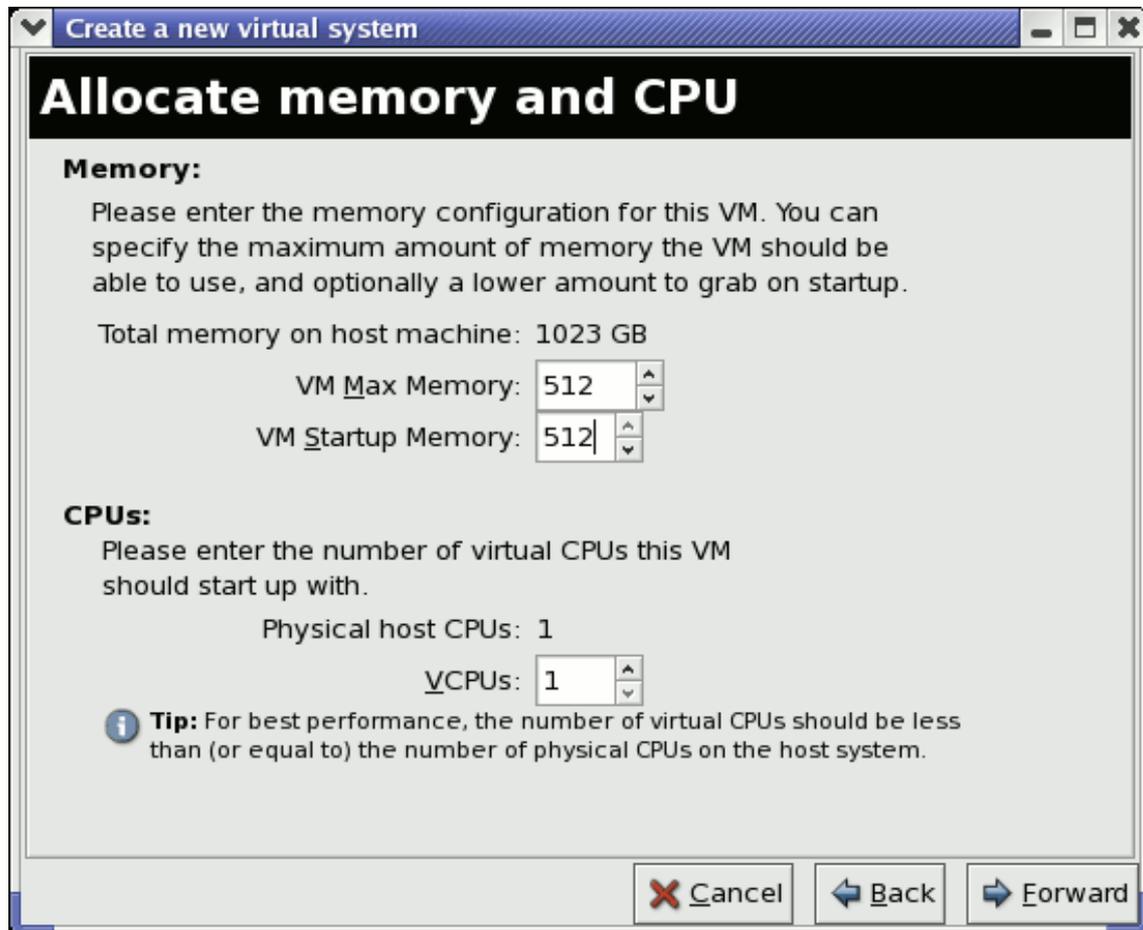


Figura 18.12. Assegnazione della memoria e della CPU

8. Selezionate **Avanti** per aprire la console e per iniziare l'installazione dei file.

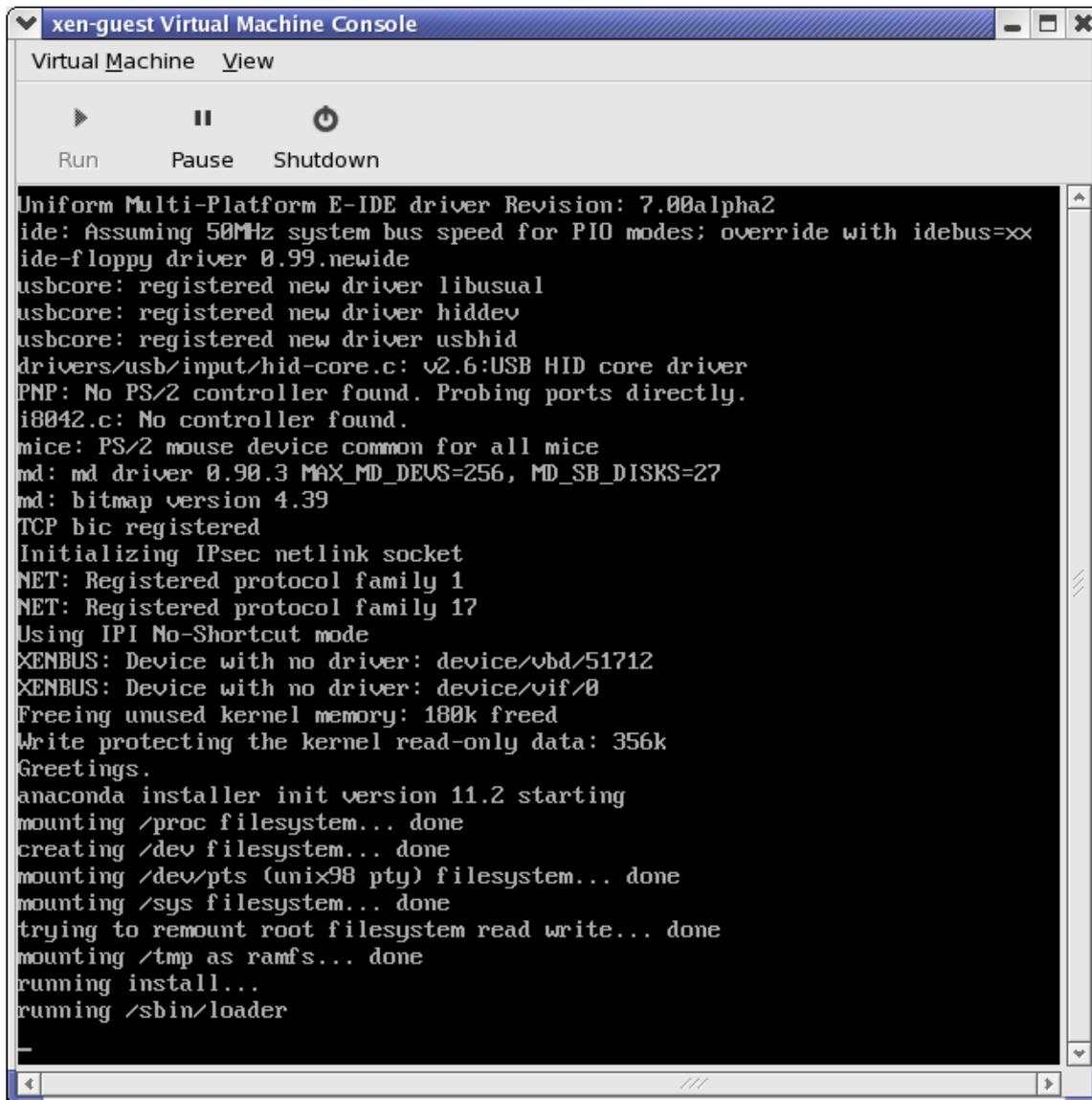


Figura 18.13. Assegnazione della memoria e della CPU

9. Completate la vostra installazione all'interno della finestra.

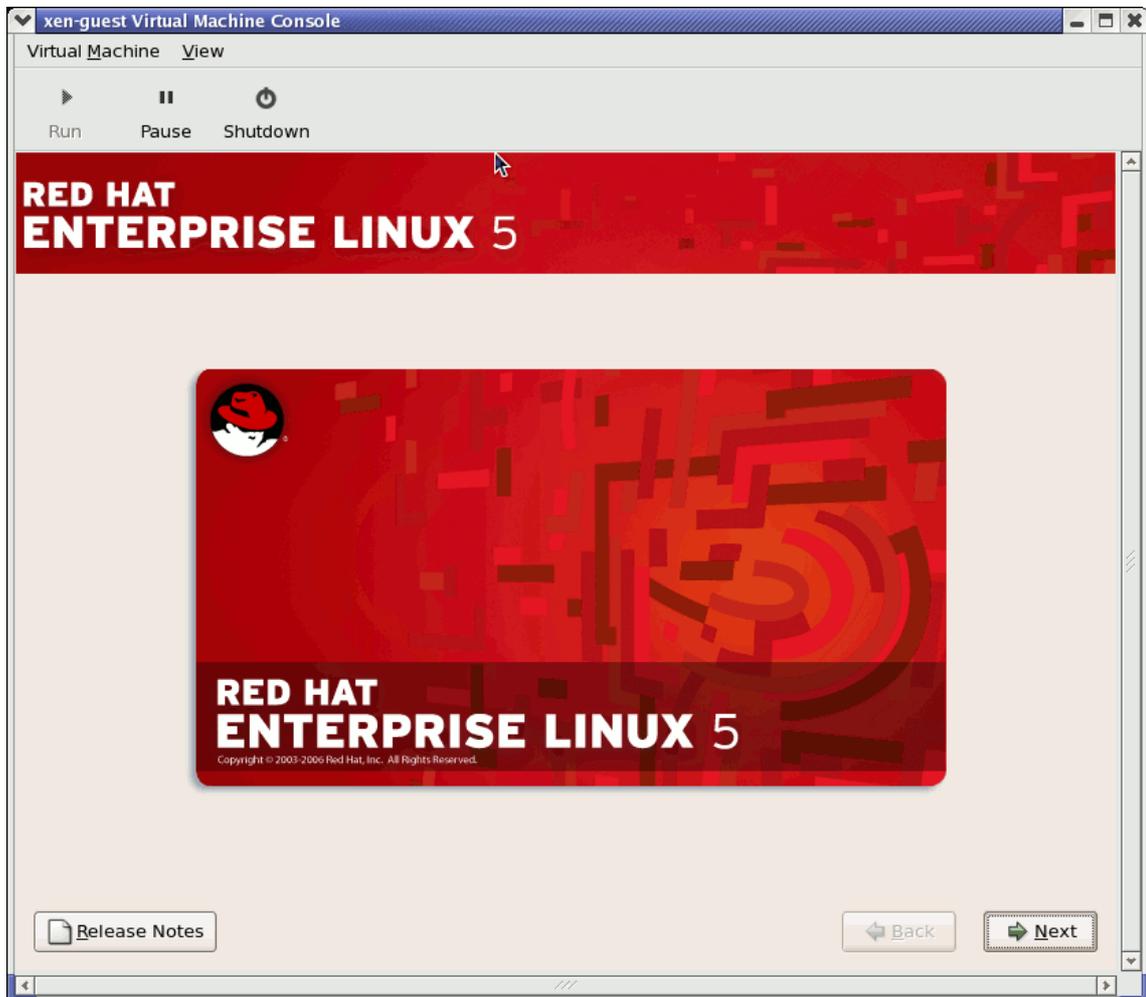


Figura 18.14. L'installazione ha ora inizio...



AVVERTIMENTO

Quando installate Red Hat Enterprise Linux 5.0 su di un guest completamente virtualizzato, *non* utilizzate il kernel **kernel-xen**. Usando il suddetto kernel su guest completamente virtualizzati, potreste causare una sospensione del vostro sistema.

Se state utilizzando un *Numero d'installazione* durante l'installazione di Red Hat Enterprise Linux 5.0 su di un guest completamente virtualizzato, assicuratevi di deselezionare il gruppo del pacchetto **Virtualization** durante l'installazione. L'opzione del gruppo del pacchetto **Virtualization** installa il kernel **kernel-xen**.

Da notare che i guest paravirtualizzati non sono interessati da questo problema. I suddetti guest utilizzano sempre il kernel **kernel-xen**.

10. Digitate `xm create -c xen-guest` per avviare il guest di Red Hat Enterprise Linux 5.0. Fate clic con il pulsante destro del mouse sul guest all'interno del Virtual Machine Manager, e successivamente selezionate **Apri** per aprire la console virtuale.

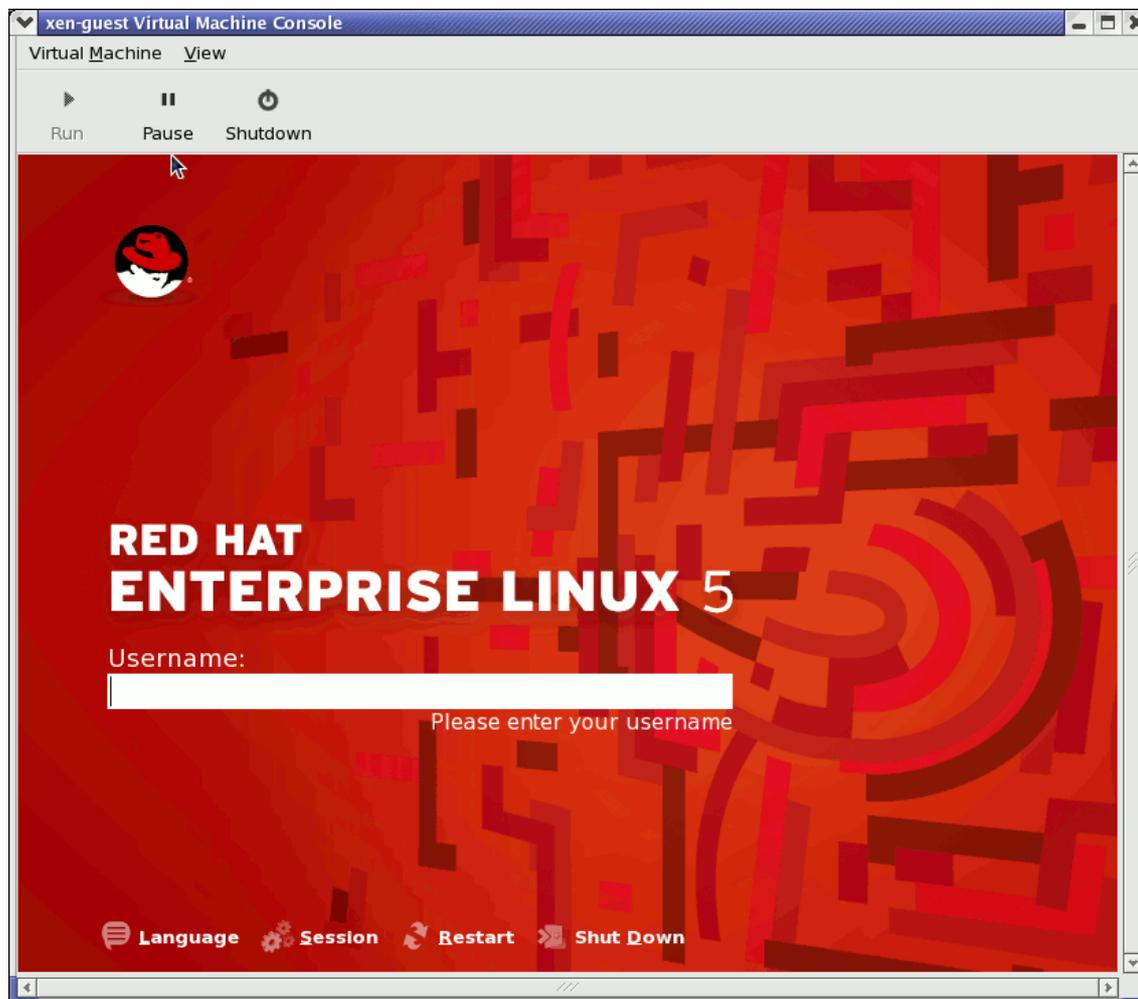


Figura 18.15. Red Hat Enterprise Linux 5.0 (guest)

11. Per poter continuare ad usare il Virtual Machine Manager, inserite il nome utente e la password.

18.8. RIPRISTINO DI UNA MACCHINA PRECEDENTEMENTE SALVATA

Dopo aver avviato il Virtual Machine Manager, tutte le macchine virtuali sul vostro sistema verranno visualizzate nella finestra principale. Domain0 è il vostro sistema host. Se non è presente alcuna macchina, ciò indicherà che nessuna macchina è in esecuzione sul sistema.

Per ripristinare una sessione precedentemente salvata:

1. Dal menu **File**, selezionate **Ripristina una macchina salvata**.

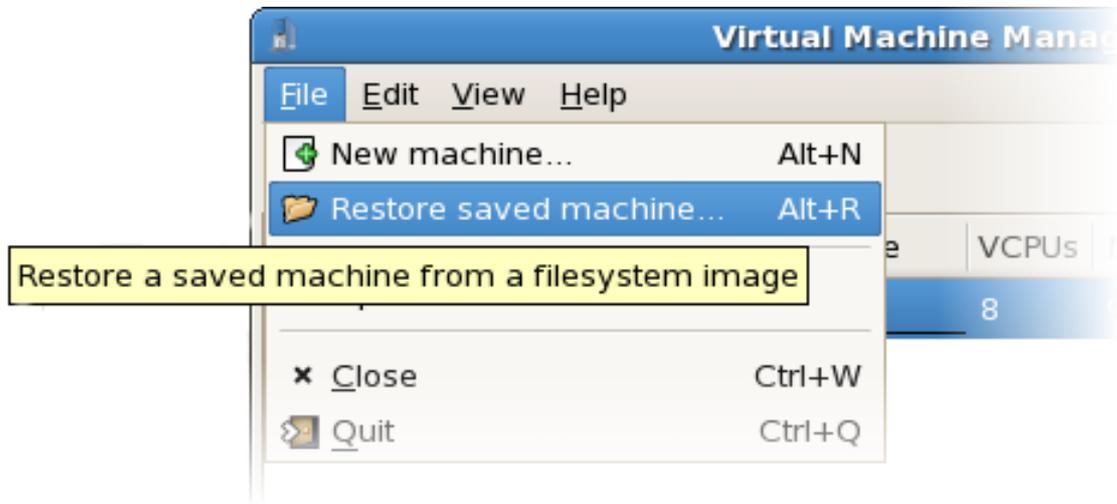


Figura 18.16. Restoring a Virtual Machine

2. Apparirà la finestra principale di ripristino macchina virtuale.

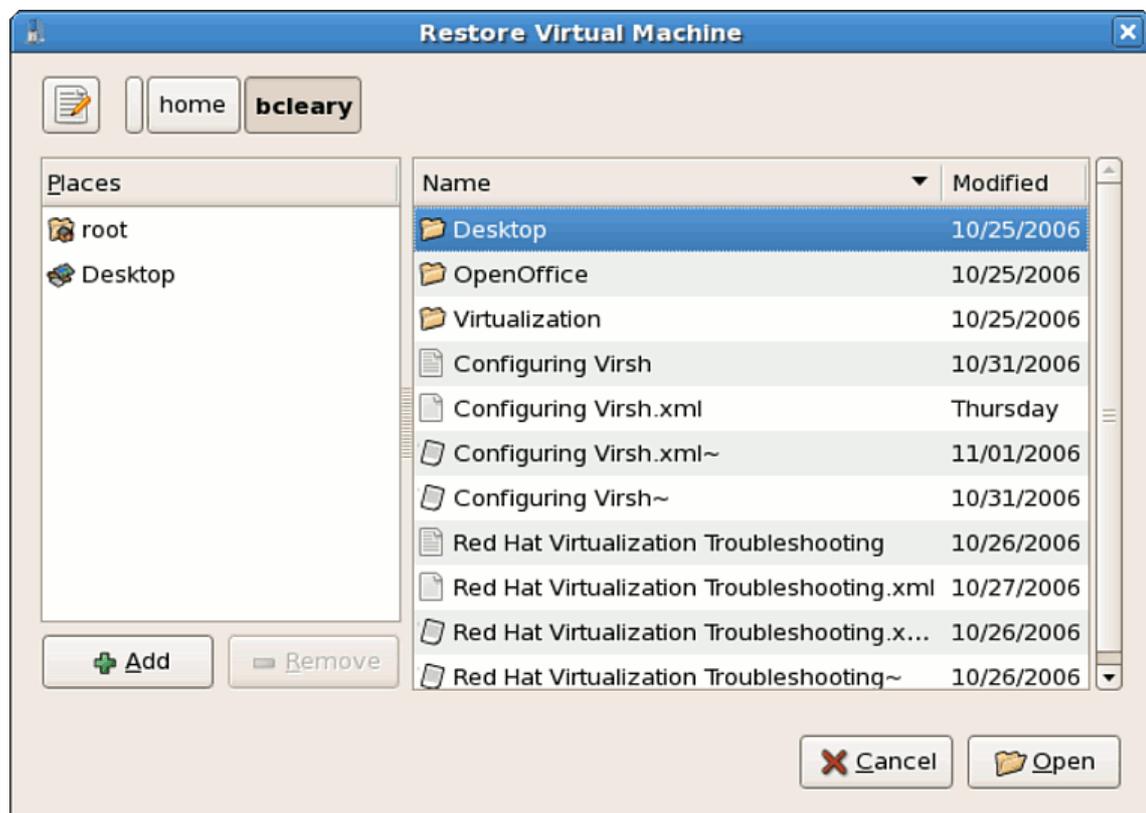


Figura 18.17. Selezione di una sessione virtual machine salvata

3. Andate nella directory corretta e selezionate il file della sessione salvata.
4. Fate clic su **Apri**.

A questo punto apparirà il sistema virtuale salvato all'interno della finestra principale del Virtual Machine Manager.

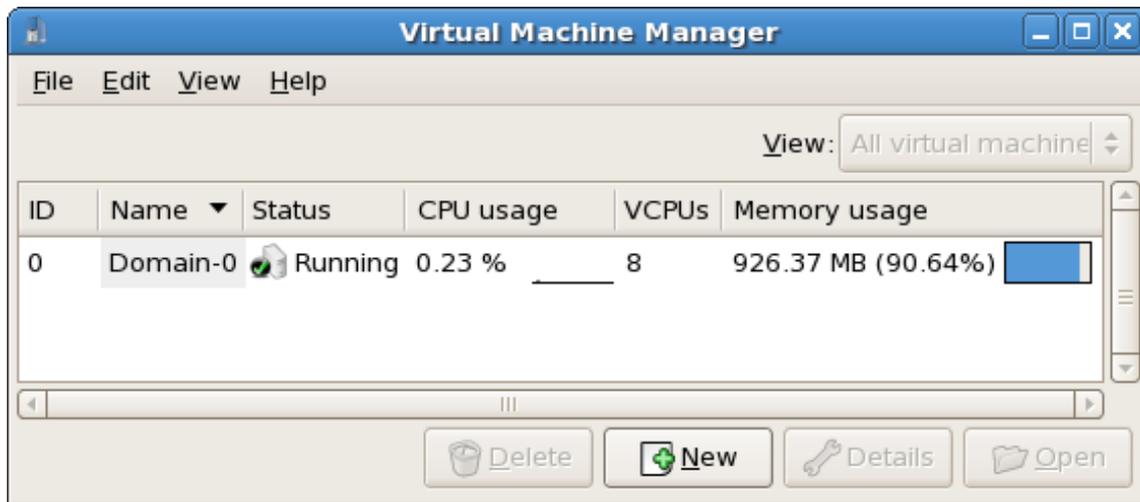


Figura 18.18. Sessione Virtual Machine Manager ripristinata

18.9. DISPLAYING VIRTUAL MACHINE DETAILS

È possibile utilizzare il monitor della macchina virtuale, per visualizzare le informazioni dei dati riguardanti le attività di qualsiasi macchina virtuale presente sul vostro sistema.

To view a virtual system's details:

1. Nella finestra principale del Virtual Machine Manager, selezionate la macchina virtuale che desiderate visualizzare.

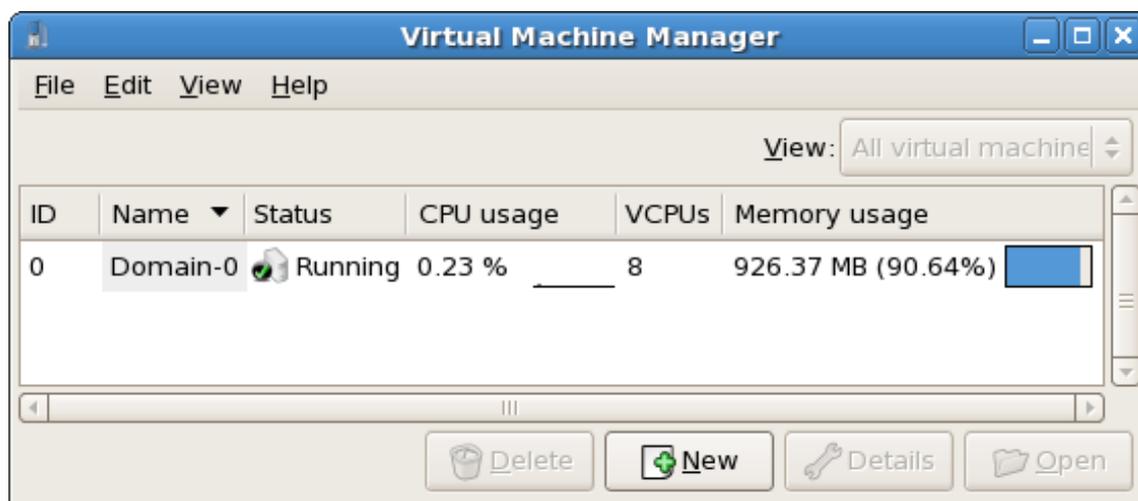


Figura 18.19. Selezione della macchina virtuale da visualizzare

2. Dal menu **Modifica** del Virtual Machine Manager, selezionate **Dettagli macchina** (oppure fate clic sul pulsante **Dettagli**, nella parte bassa della finestra principale del Virtual Machine Manager).

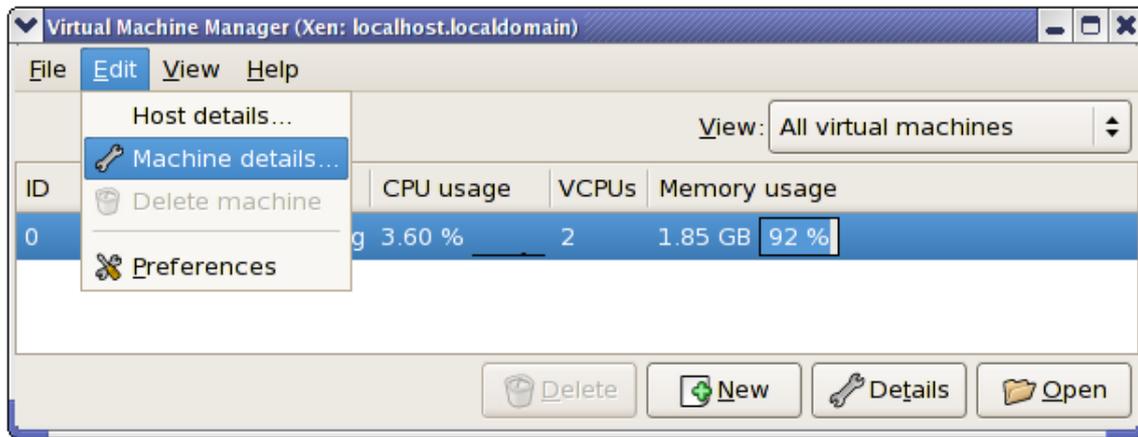


Figura 18.20. Visualizzazione Menu dettagli della macchina virtuale

Visualizzazione finestra panoramica dei dettagli della macchina virtuale. Questa finestra riassume l'utilizzo della memoria e della CPU per i domini specificati.

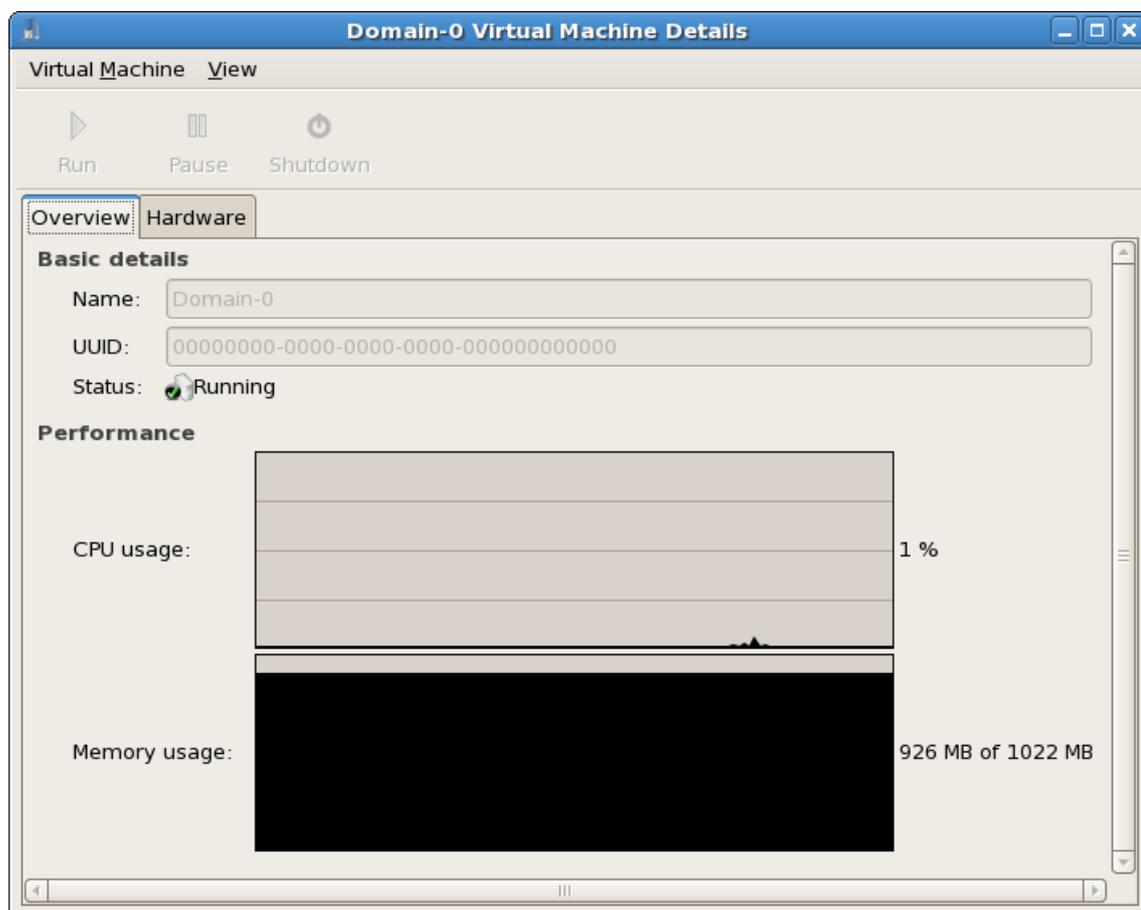


Figura 18.21. Visualizzazione panoramica dettagli della macchina virtuale

3. Nella finestra dei dettagli della macchina virtuale fate clic su **Hardware**.

A questo punto sarete in grado di visualizzare la finestra Hardware dei dettagli della macchina virtuale.

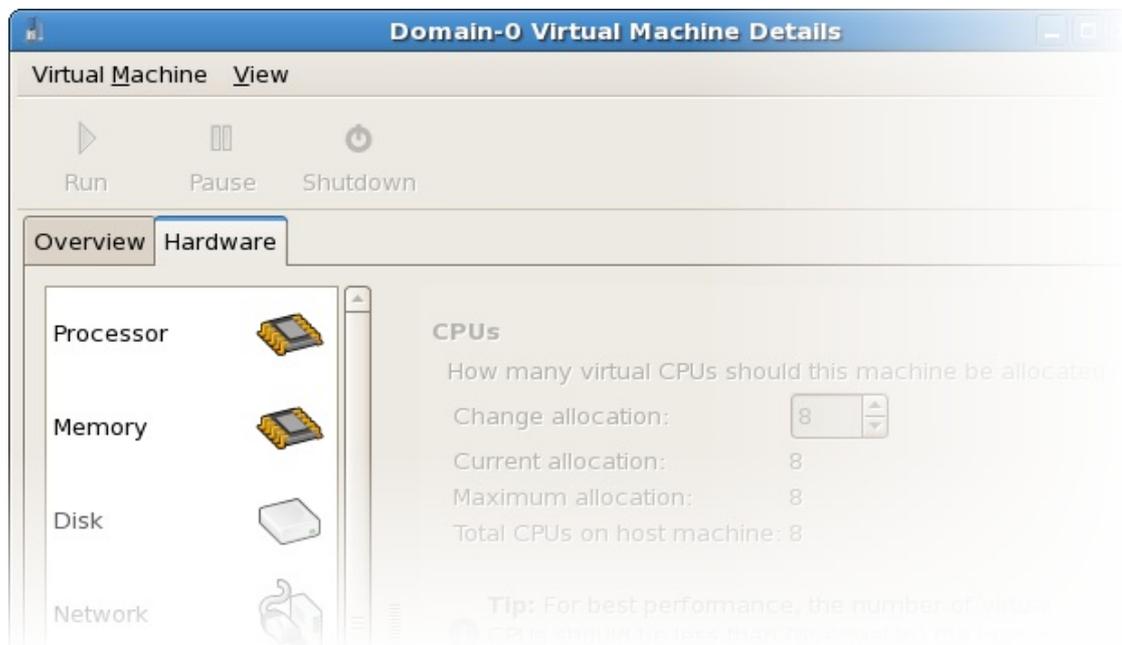


Figura 18.22. Visualizzazione finestra Hardware dei dettagli della macchina virtuale

4. Sulla tabella **Hardware**, fate clic su **Processore** per visualizzare o modificare l'assegnazione corrente della memoria del processore.

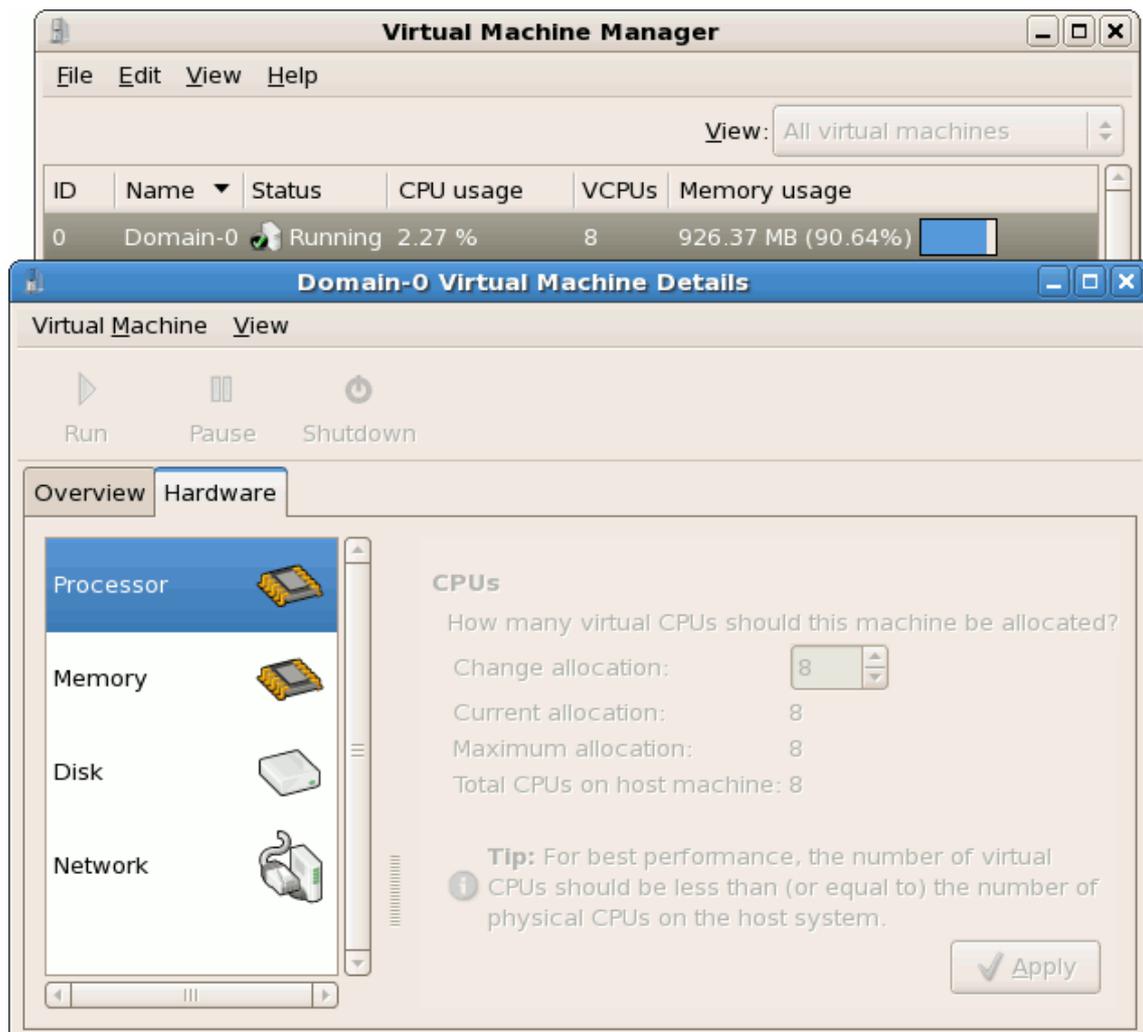


Figura 18.23. Visualizzazione assegnazione del processore

- Sulla tabella **Hardware**, fate clic su **Memoria** per visualizzare o modificare l'assegnazione della memoria RAM corrente.

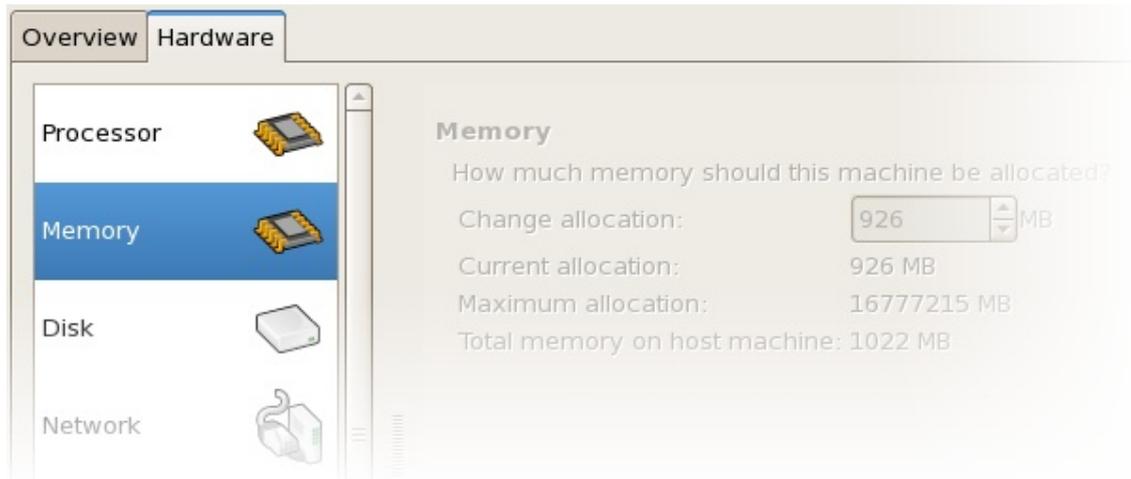


Figura 18.24. Visualizzazione assegnazione della memoria

- Sulla tabella **Hardware**, fate clic su **Disco** per visualizzare o modificare la configurazione corrente del disco fisso.

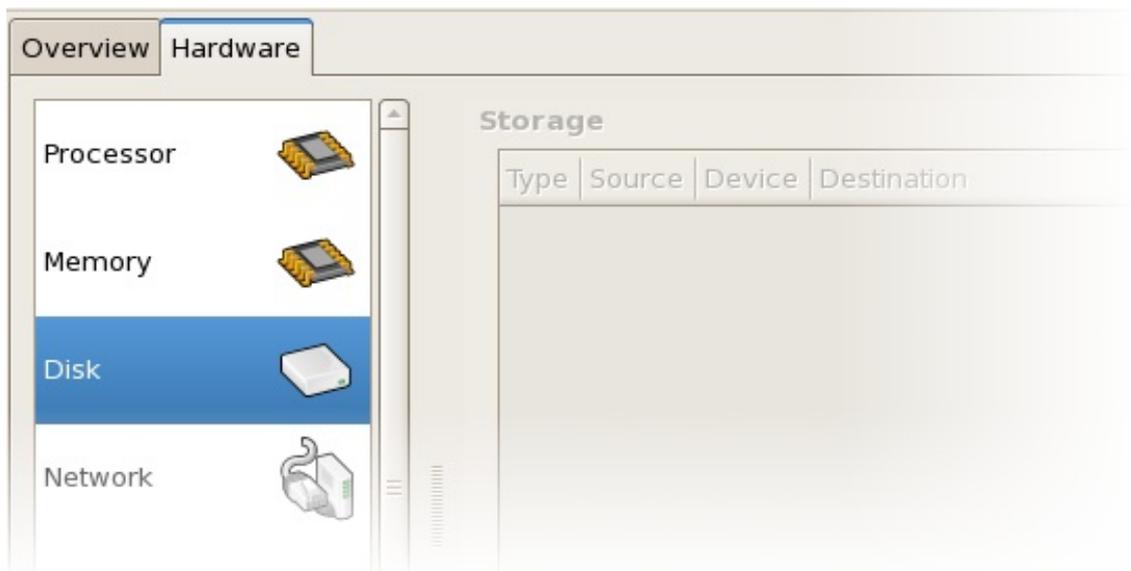


Figura 18.25. Visualizzazione configurazione del disco

- Sulla tabella **Hardware**, fate clic su **Rete** per visualizzare o modificare la configurazione di rete corrente.

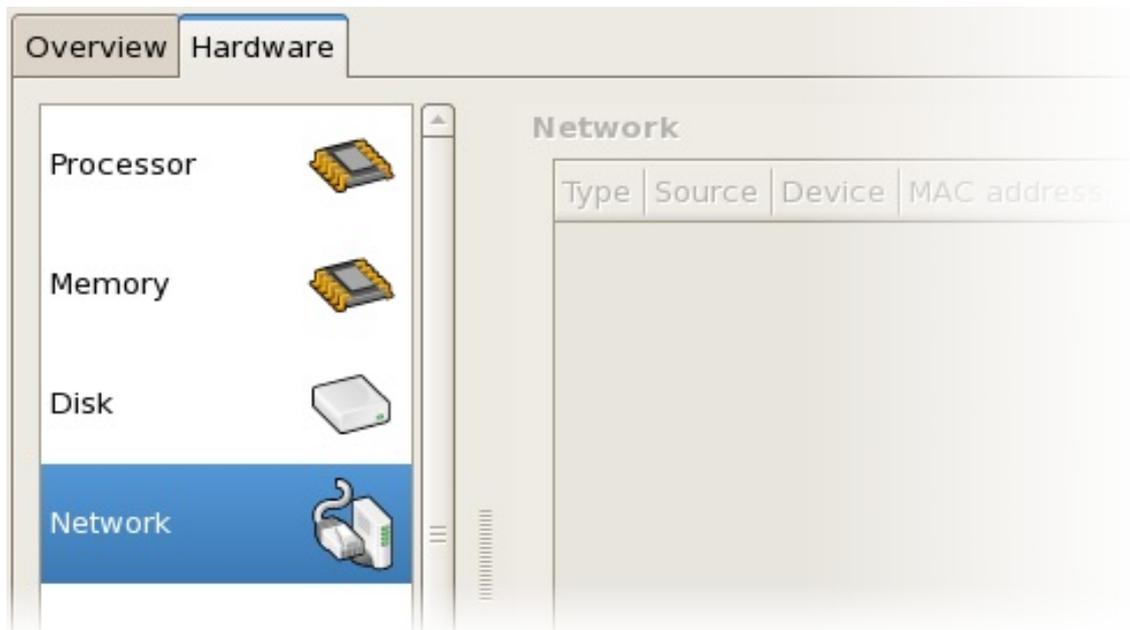


Figura 18.26. Visualizzazione configurazione di rete

18.10. CONFIGURING STATUS MONITORING

È possibile utilizzare il Virtual Machine Manager per modificare lo Status monitoring del sistema virtuale.

Per configurare lo Status monitoring ed abilitare le console:

1. Dal menu **Modifica**, selezionate **Preferenze**.

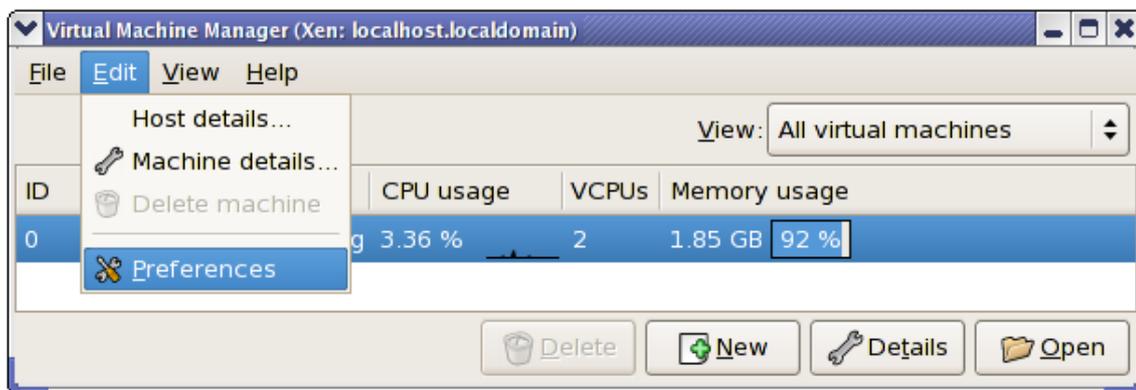


Figura 18.27. Modifica preferenze della macchina virtuale

Verrà visualizzata la finestra preferenze del Virtual Machine Manager.

2. Dalla casella di selezione dello Status monitoring, specificare il periodo (in secondi) che il sistema deve aggiornare.

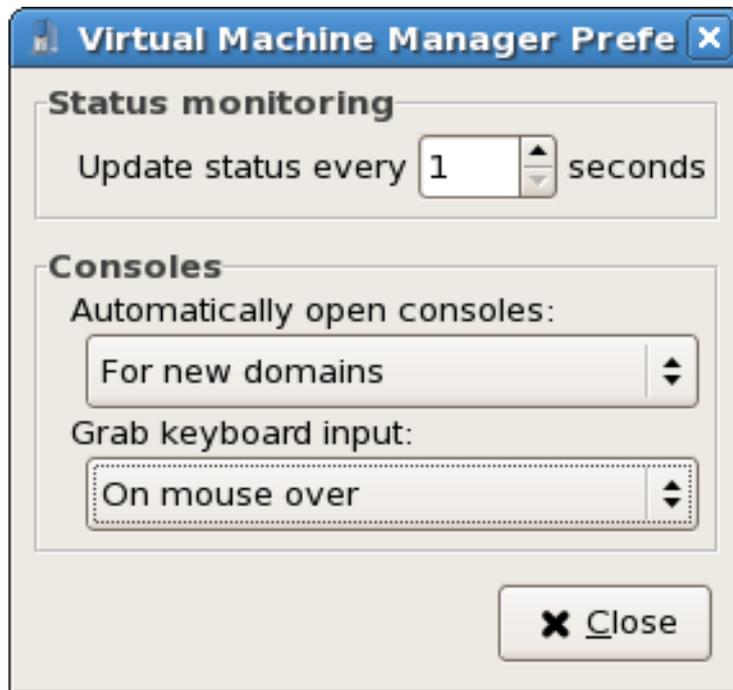


Figura 18.28. Configurando Status Monitoring

3. Dall'area della console, specificare come aprire una console insieme al dispositivo input.

18.11. VISUALIZZAZIONE DEL DOMAIN ID

Per visualizzare i domain ID per tutte le macchine presenti sul vostro sistema:

1. Dal menu **Visualizza**, selezionate la casella **Domain ID**.

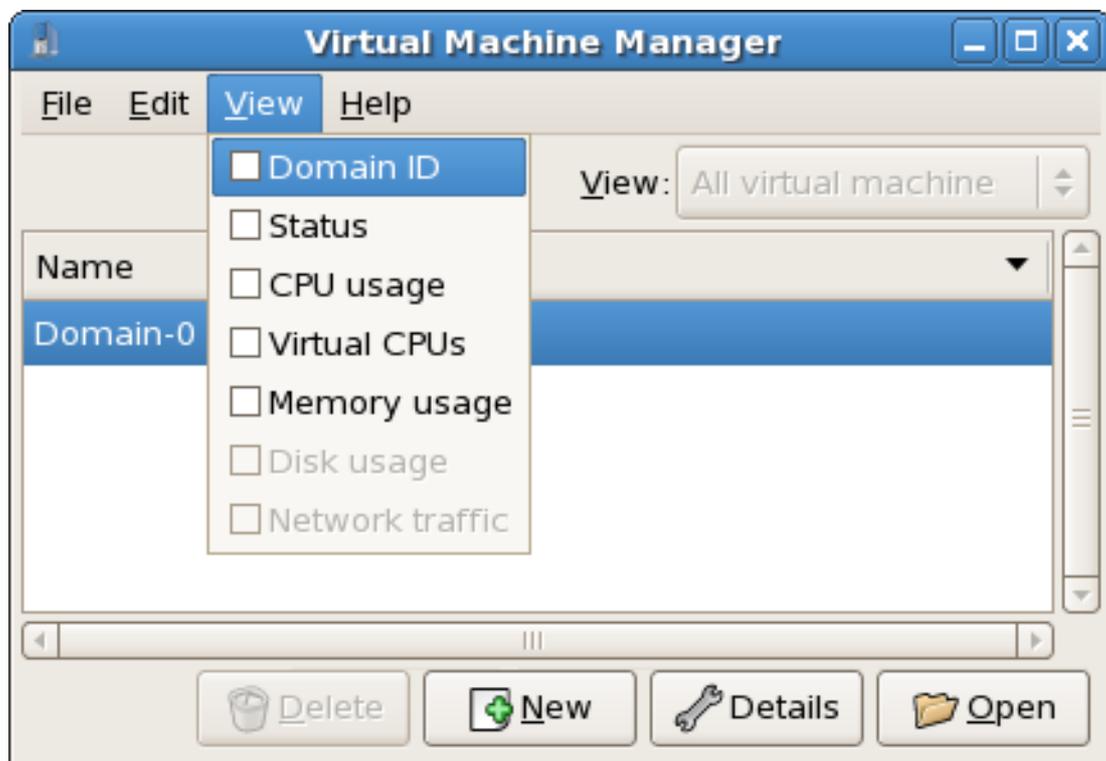


Figura 18.29. Visualizzazione Domain-ID

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

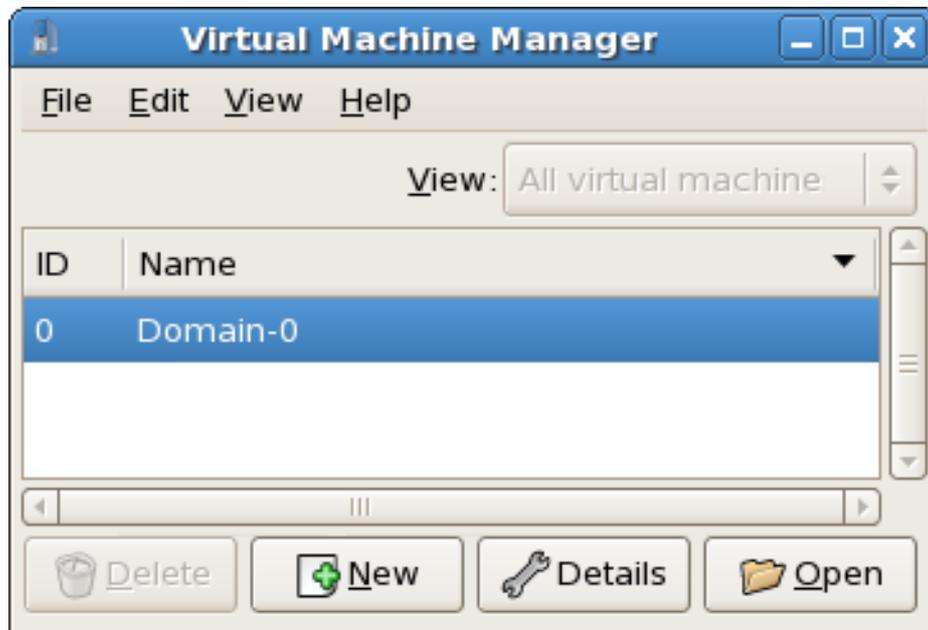


Figura 18.30. Visualizzazione Domain-ID

18.12. VISUALIZZAZIONE DELLO STATO DELLA MACCHINA VIRTUALE

Per visualizzare lo stato di tutte le macchine virtuali sul vostro sistema:

1. Dal menu **Visualizza**, selezionate la casella **Stato**.

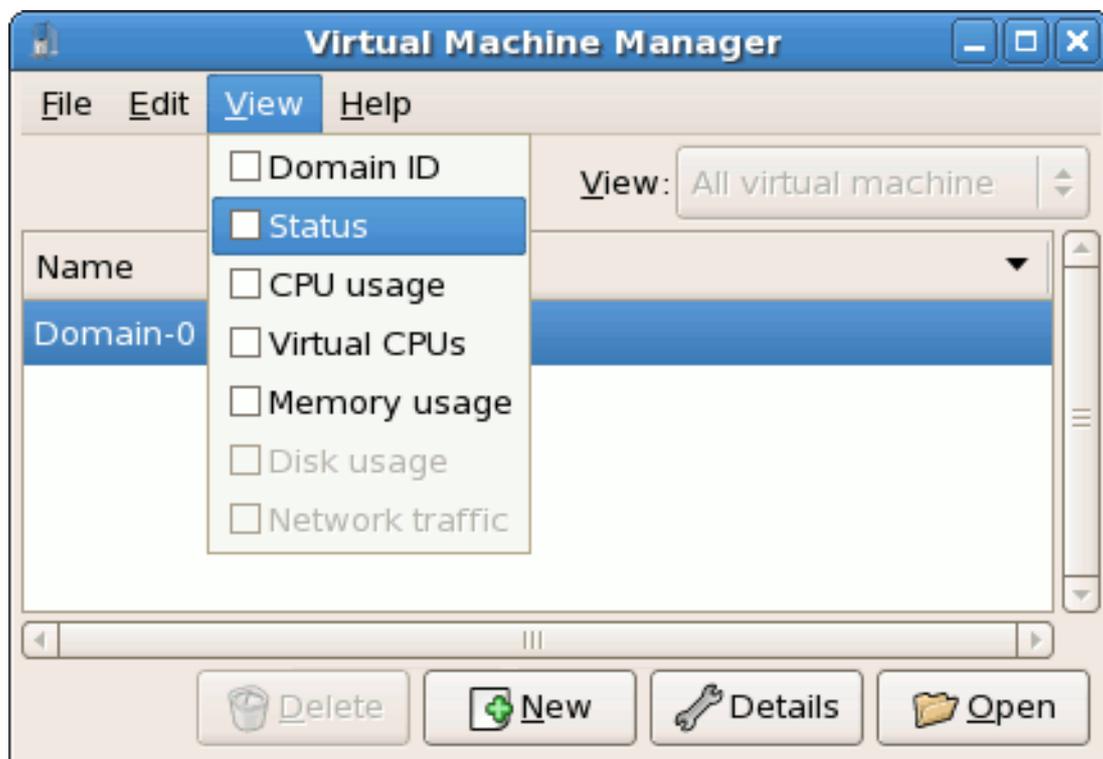


Figura 18.31. Visualizzazione dello stato della macchina virtuale

2. Il Virtual Machine Manager elenca lo stato di tutte le macchine virtuali presenti sul vostro sistema.

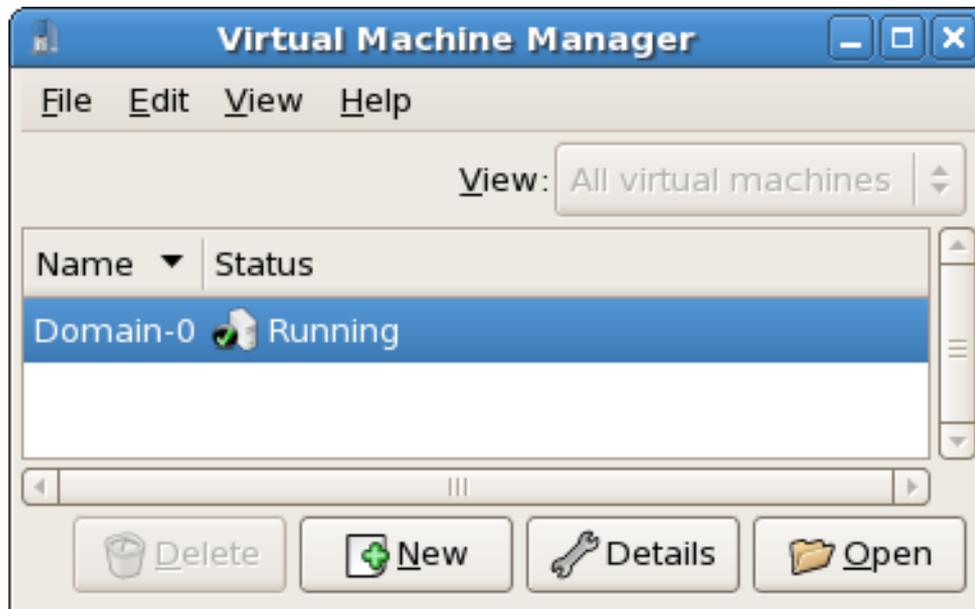


Figura 18.32. Visualizzazione dello stato della macchina virtuale

18.13. VISUALIZZAZIONE DELLE CPU VIRTUALI

Per visualizzare il numero delle CPU virtuali per tutte le macchine virtuali sul vostro sistema:

1. Dal menu **Visualizza**, selezionate la casella **CPU virtuali**.

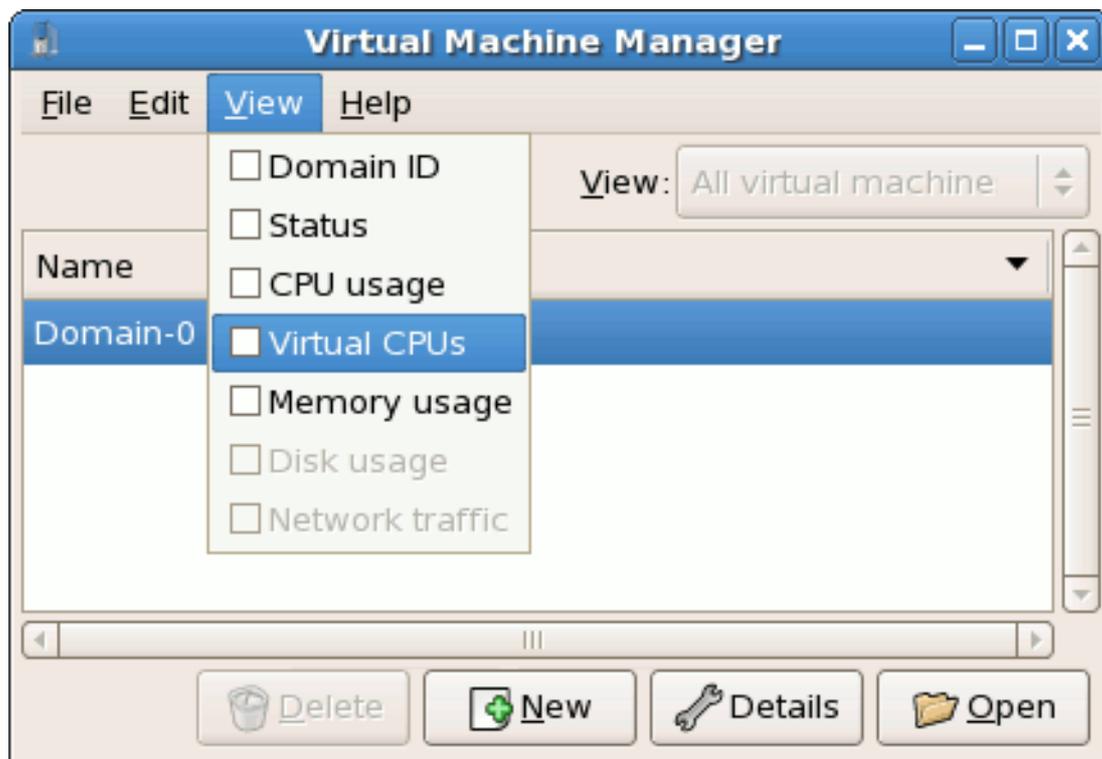


Figura 18.33. Visualizzazione delle CPU virtuali

2. Il Virtual Machine Manager elenca le CPU virtuali per tutte le macchine virtuali presenti sul vostro sistema.

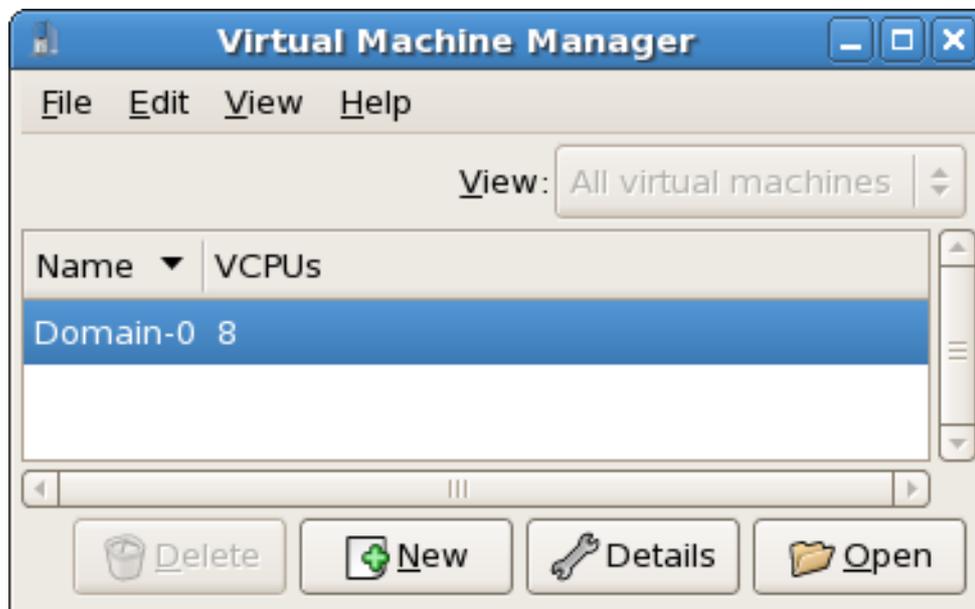


Figura 18.34. Visualizzazione delle CPU virtuali

18.14. VISUALIZZAZIONE UTILIZZO DELLA CPU

Per visualizzare l'utilizzo della CPU per tutte le macchine virtuali presenti sul vostro sistema:

1. Dal menu **Visualizza**, selezionate la casella **Utilizzo CPU**.

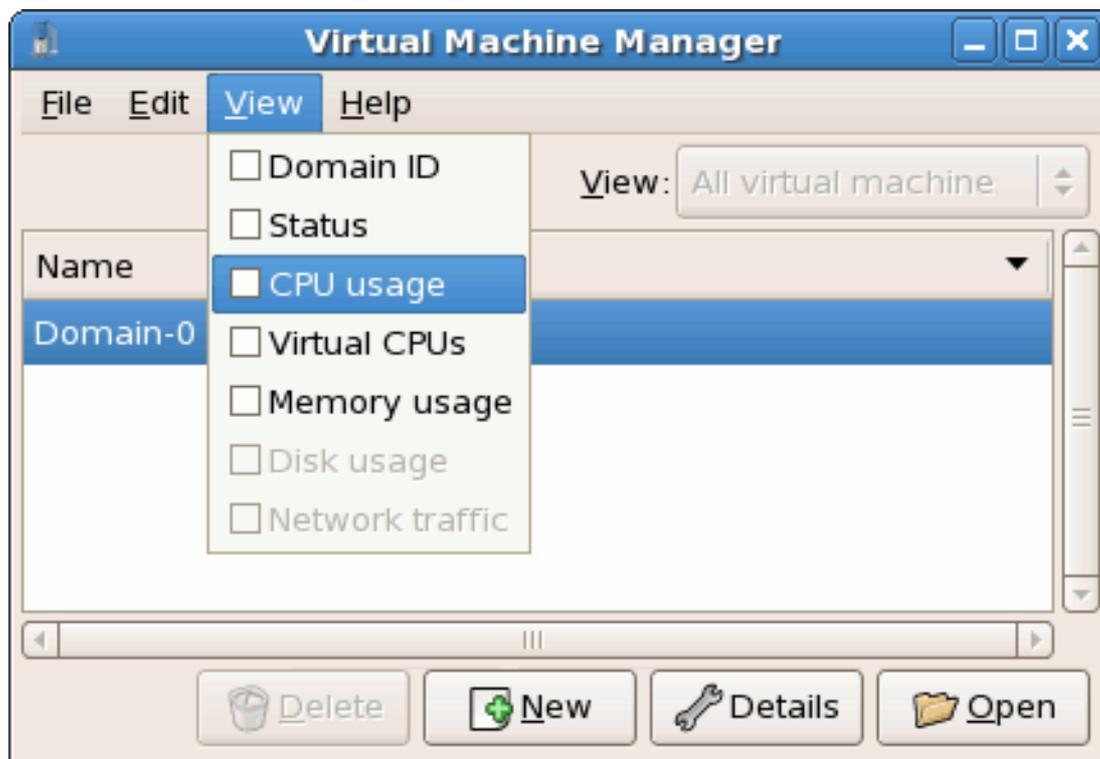


Figura 18.35. Visualizzazione utilizzo della CPU

2. Il Virtual Machine Manager elenca la percentuale di CPU in uso per tutte le macchine virtuali presenti sul vostro sistema.

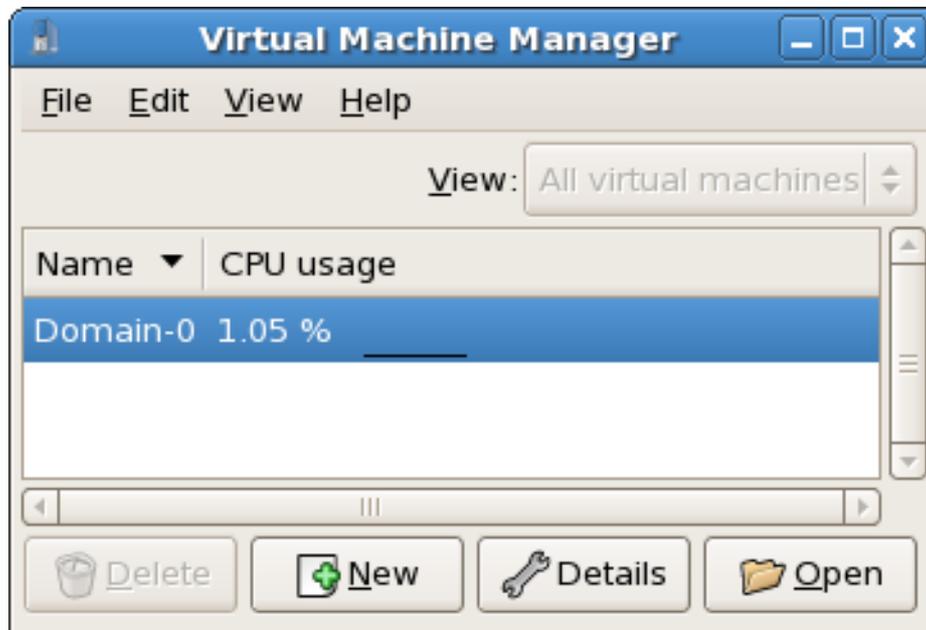


Figura 18.36. Visualizzazione utilizzo della CPU

18.15. VISUALIZZAZIONE UTILIZZO DELLA MEMORIA

Per visualizzare l'uso della memoria per tutte le macchine virtuali sul vostro sistema:

1. Dal menu **Visualizza**, selezionate la casella **Utilizzo memoria**.

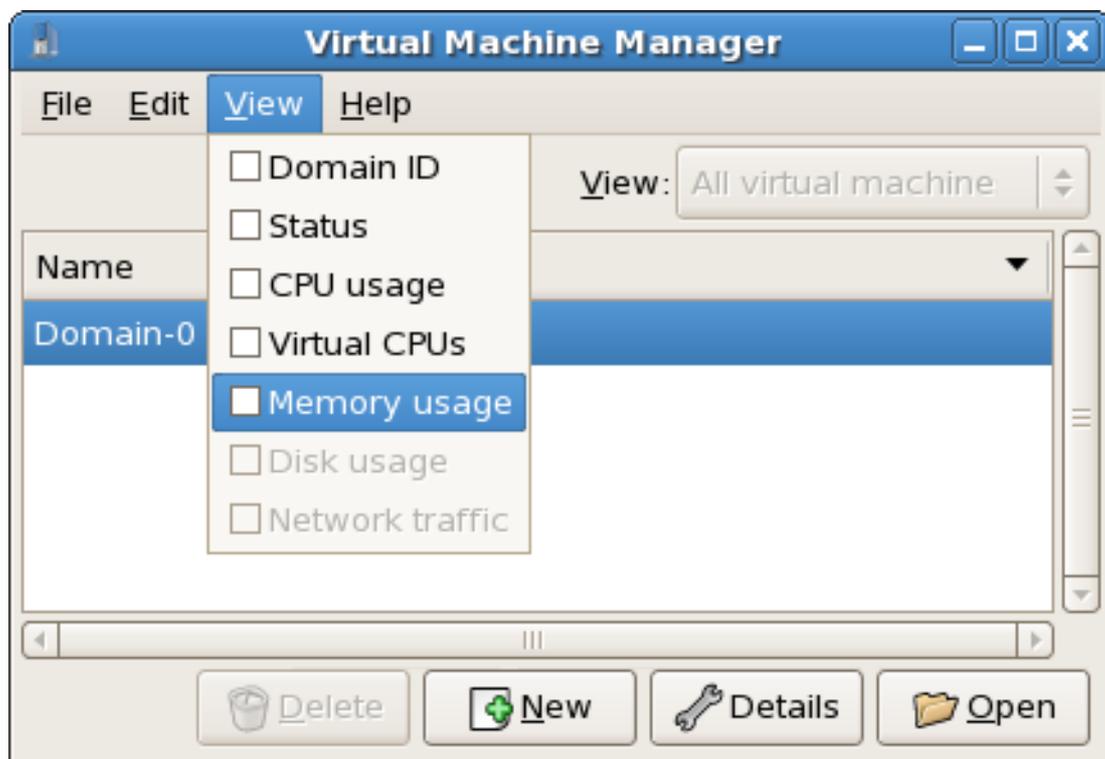


Figura 18.37. Visualizzazione utilizzo della memoria

2. Il Virtual Machine Manager elenca la percentuale di memoria in uso (in megabyte), per tutte le macchine virtuali presenti sul vostro sistema.

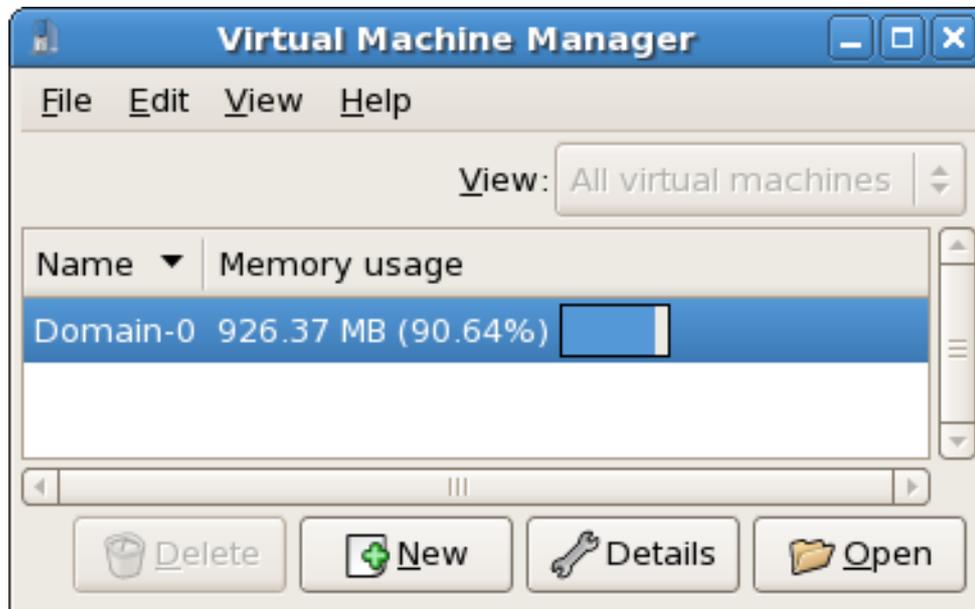


Figura 18.38. Visualizzazione utilizzo della memoria

18.16. GESTIONE DI UNA RETE VIRTUALE

Per configurare una rete virtuale sul vostro sistema:

1. Dal menu **Modifica**, selezionate **Dettagli host**.

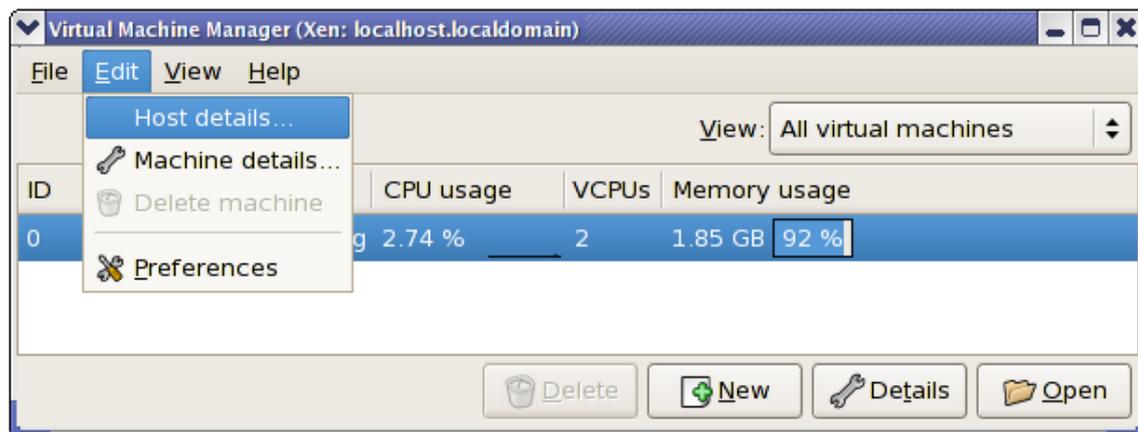


Figura 18.39. Selezione Dettagli host

2. Ciò aprirà il menu **Dettagli host**. A questo punto fate clic sulla scheda **Reti virtuali**.

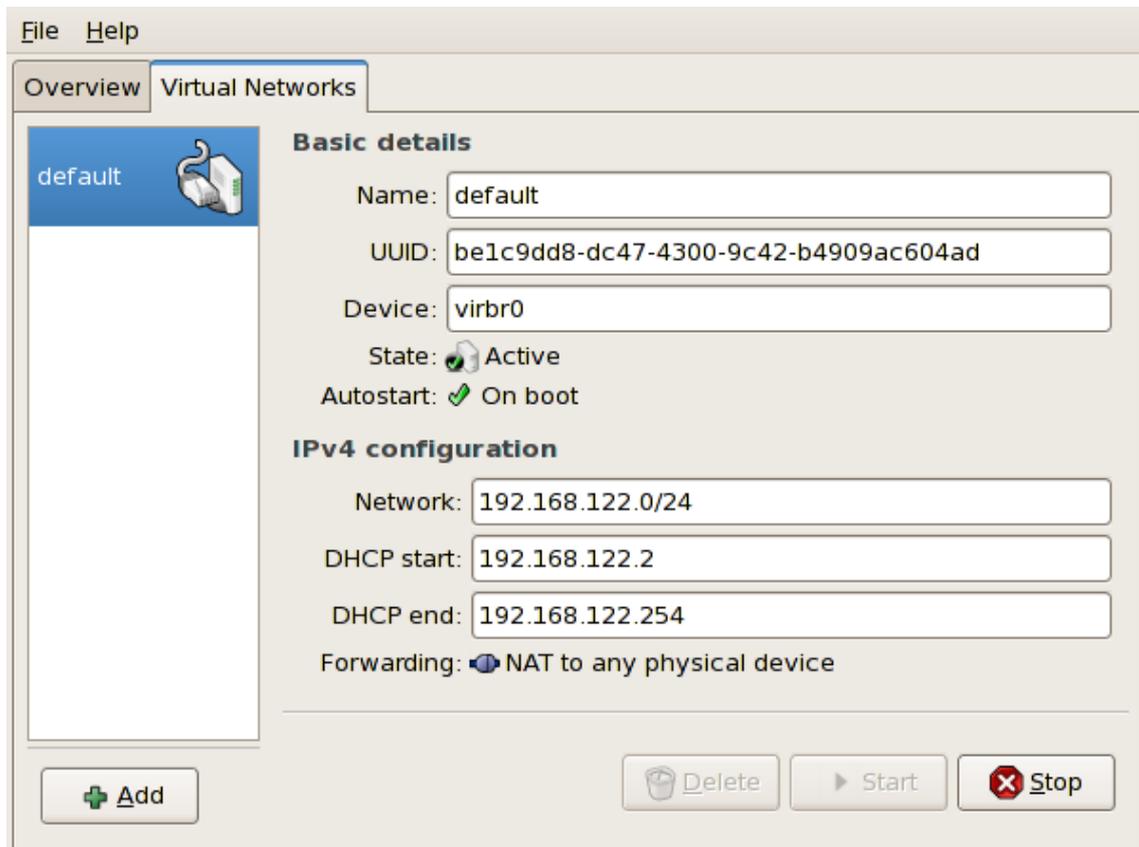


Figura 18.40. Configurazione rete virtuale

3. Tutte le reti virtuali disponibili vengono elencate nella casella sinistra del menu. È possibile modificare la configurazione di una rete virtuale, selezionandola direttamente da questa casella e modificandola in modo da voi desiderato.

18.17. CREAZIONE DI UNA RETE VIRTUALE

Per creare una rete virtuali sul vostro sistema:

1. Open the **Host Details** menu (refer to [Sezione 18.16, «Gestione di una rete virtuale»](#)) and click the **Add** button.

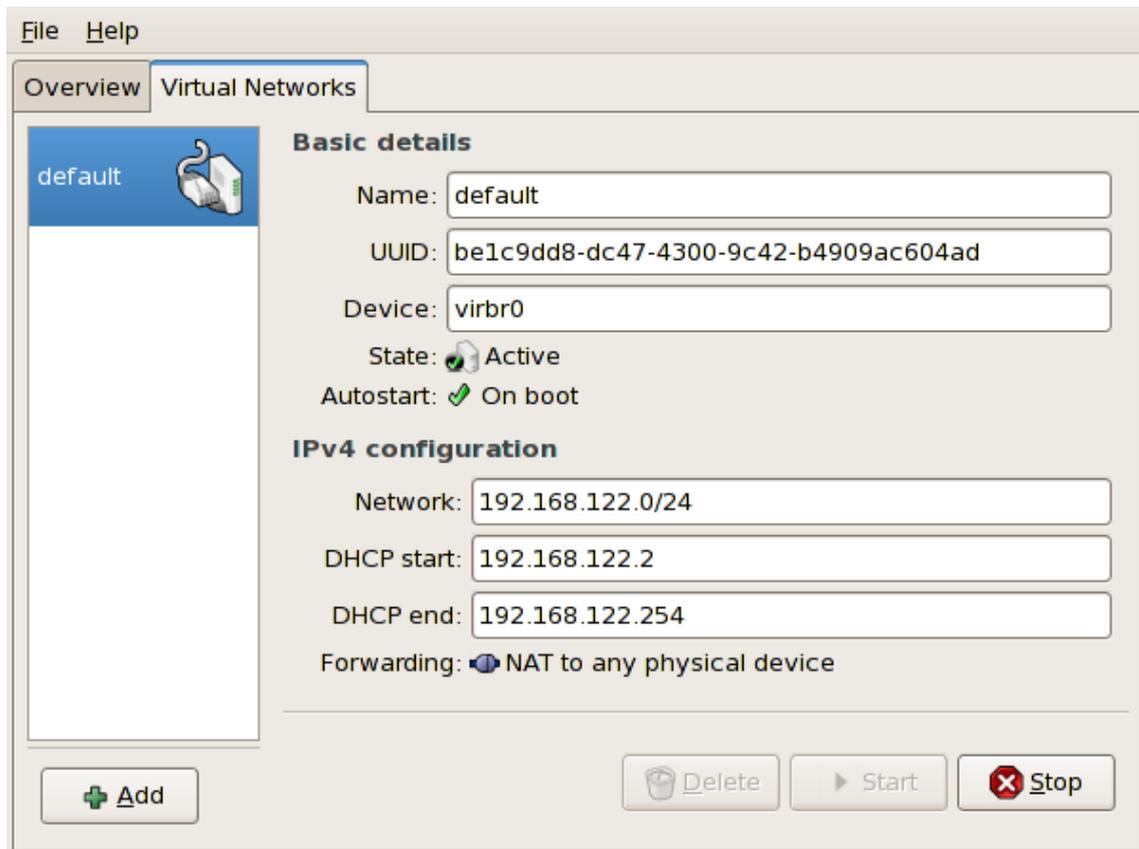


Figura 18.41. Configurazione rete virtuale

In tal modo aprirete il menu **Crea una nuova rete virtuale**. Per continuare fate clic su **Avanti**.

Creating a new virtual network

This assistant will guide you through creating a new virtual network. You will be asked for some information about the virtual network you'd like to create, such as:

- A **name** for your new virtual network
- The IPv4 **address** and **netmask** to assign
- The **address range** from which the **DHCP** server will allocate addresses for virtual machines
- Whether to **forward** traffic to the physical network



Figura 18.42. Creazione di una nuova rete virtuale

2. Inserite il nome appropriato per la vostra rete virtuale e successivamente fate clic su **Avanti**.



Figura 18.43. Come nominare la vostra rete virtuale

3. Inserite uno spazio per l'indirizzo IPv4 per la vostra rete virtuale e fate clic su **Avanti**.

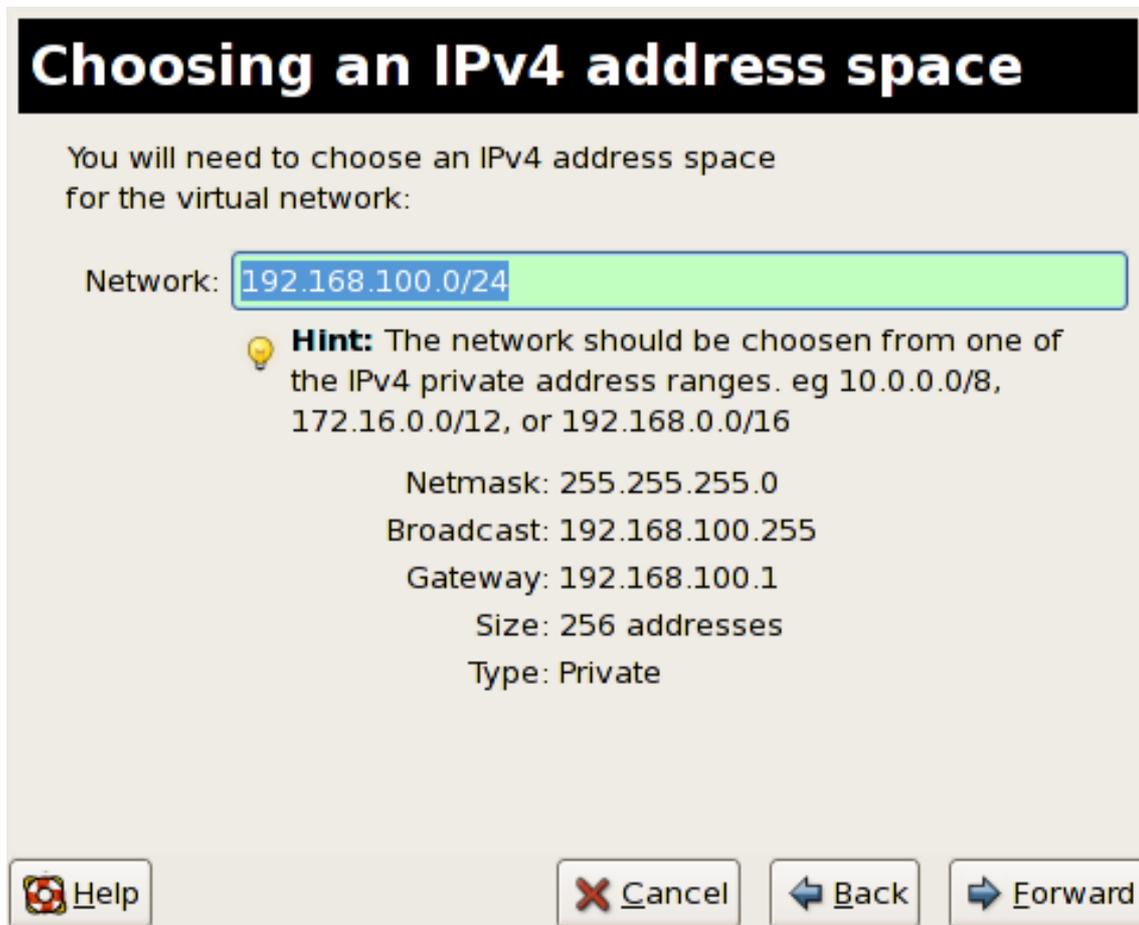


Figura 18.44. Scelta di uno spazio per l'indirizzo IPv4

4. Definire il DHCP range per la vostra rete virtuale specificando un range di **Inizio** e **Fine** di indirizzi IP. Fate clic su **Avanti** per continuare.

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

Figura 18.45. Selezione di una gamma DHCP

5. Selezionate il modo attraverso il quale una rete virtuale si collega alle rete fisica.

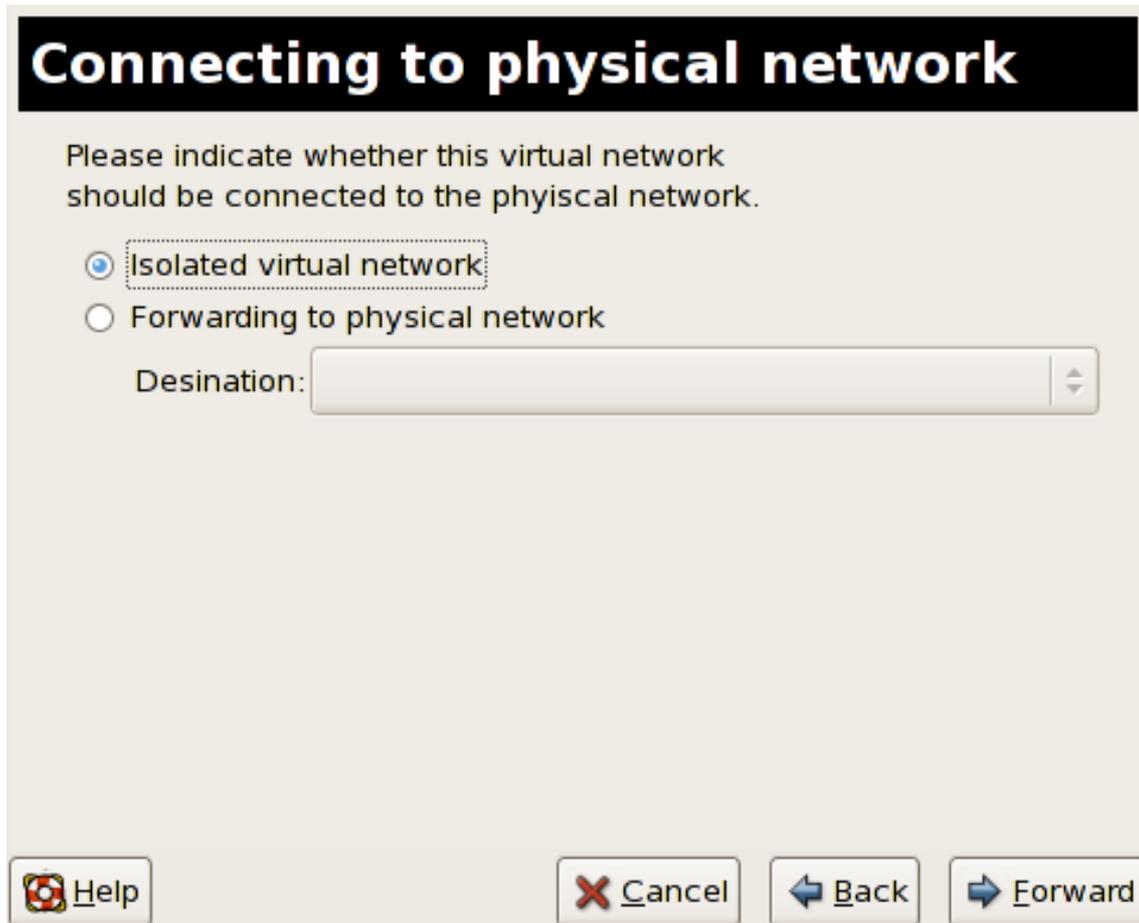


Figura 18.46. Collegamento alla rete fisica

Se selezionate **Inoltro ad una rete fisica**, scegliete se la **Destinazione** debba essere **NAT per qualsiasi dispositivo fisico** o **NAT per il dispositivo fisico eth0**.

Fate clic su **Avanti** per continuare.

6. Ora sarete in grado di creare una rete. Controllate la configurazione della vostra rete e fate clic su **Fine**.

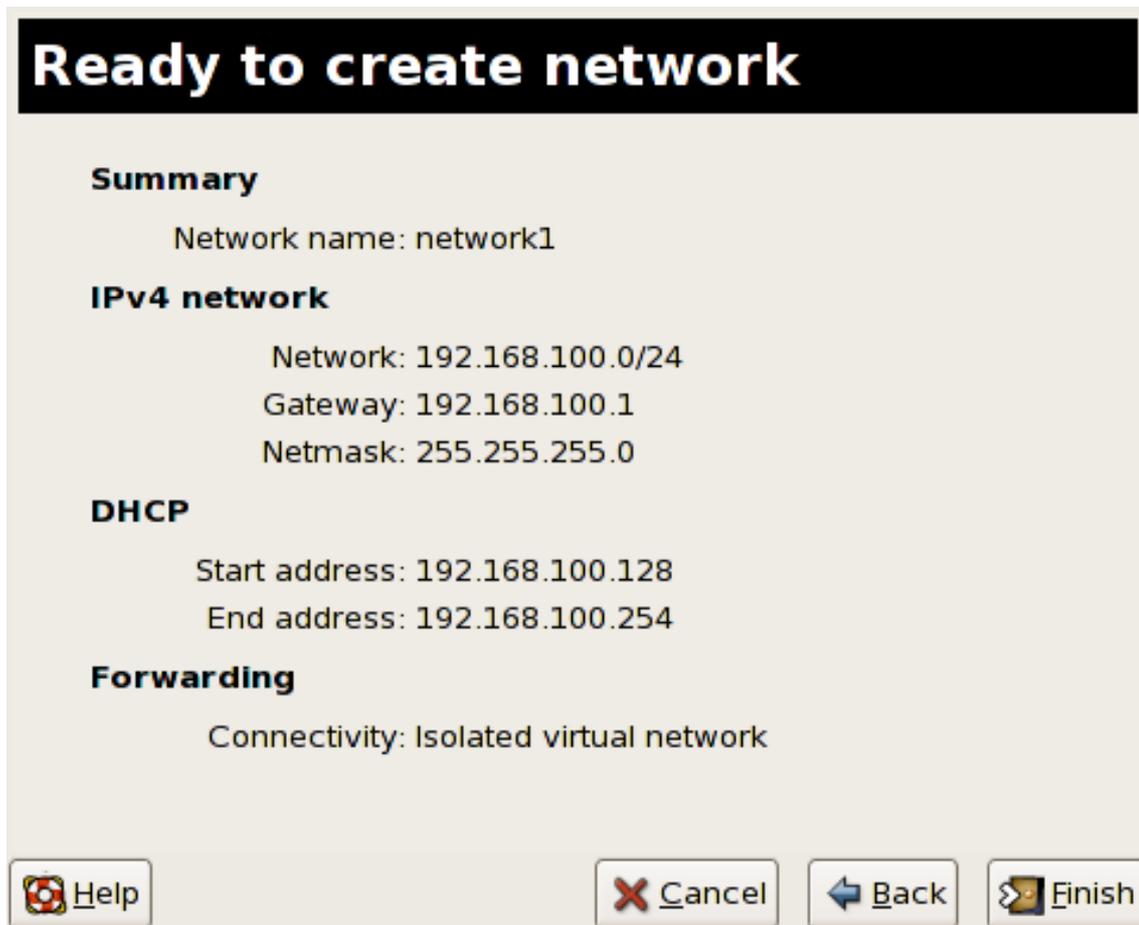


Figura 18.47. Pronti per creare la rete

7. La nuova rete virtuale sarà ora disponibile tramite la scheda **Rete virtuale** del menu **Dettagli host**.

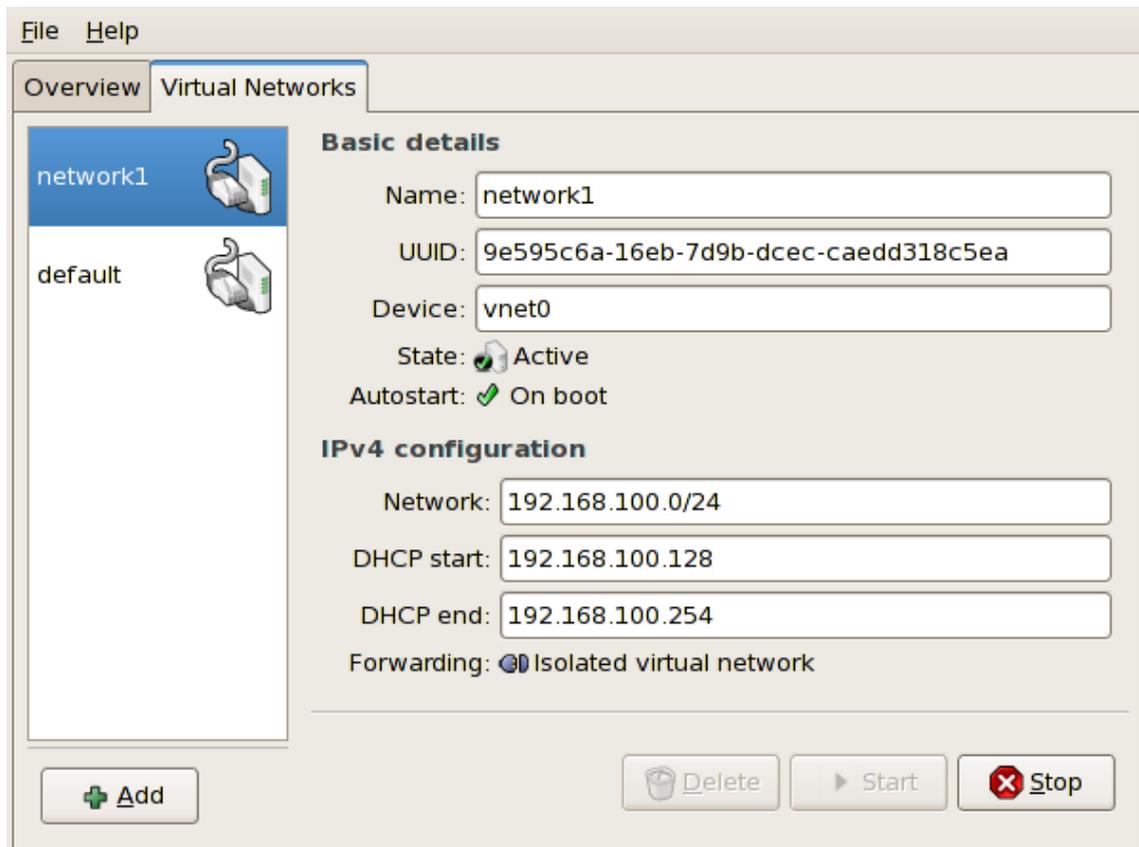


Figura 18.48. È ora disponibile la nuova rete virtuale

CAPITOLO 19. TROUBLESHOOTING DI RED HAT VIRTUALIZATION

Questa sezione riporta possibili problematiche riguardanti l'installazione, la gestione e le operazioni generali dei vostri sistemi di Red Hat Virtualization. Questa sezione di troubleshooting affronta i messaggi d'errore, la posizione dei file di log, i tool del sistema e gli approcci generici per la ricerca dei dati e l'analisi dei problemi.

19.1. PANORAMICA DI LOGFILE E POSIZIONI

When deploying Red Hat Enterprise Linux 5.0 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:

- La directory di configurazione principale di Red Hat Virtualization è `/etc/xen/`. Questa directory contiene il demone `xend` insieme ad altri file di configurazione della macchina virtuale. Anche i file per lo script di networking risiedono qui (nella sottodirectory `/scripts`).
- Tutti i file di log che consulterete per il processo di troubleshooting, risiedono nella directory `/var/log/xen`.
- Ricordate che la directory predefinita per tutte le immagini del disco basate sul file della macchina virtuale, risiede nella directory `/var/lib/xen`.
- Le informazioni di Red Hat Virtualization per il file system `/proc`, risiedono nella directory `/proc/xen/`.

19.2. DESCRIZIONI LOGFILE

Red Hat Virtualization contiene il demone `xend` ed il processo `qemu-dm`, le due utility che scrivono i file di log multipli sulla directory `/var/log/xen/`:

- `xend.log` è il file di log che contiene tutti i dati raccolti dal demone `xend`, sia se si tratta di un evento normale del sistema, che di un'azione inizializzata da un operatore. Tutte le operazioni della macchina virtuale (come la creazione, l'arresto e la distruzione ecc.) appaiono qui. `xend.log` rappresenta generalmente il luogo dove poter controllare eventi o problemi relativi alla prestazione. Contiene voci dettagliate e condizioni relative ai messaggi d'errore.
- `xend-debug.log` è il file di log che contiene i record degli errori relativi ad un evento di `xend`, insieme ai sottosistemi di Virtualizzazione (come ad esempio framebuffer, script Python, ecc.).
- `xen-hotplug-log` è il file di log che contiene i dati degli eventi hotplug. Se un dispositivo o uno script di rete non risultano online, l'evento viene visualizzato qui.
- `qemu-dm.[PID].log` è il file di log creato dal processo `qemu-dm` per ogni guest completamente virtualizzato. Se utilizzate il suddetto file di log, è necessario riprendere il PID del processo di `qemu-dm` in questione, tramite l'utilizzo del comando `ps`, in modo da esaminare gli argomenti del processo per isolare il processo `qemu-dm` sulla macchina virtuale. Da notare che è necessario sostituire il simbolo [PID] con il processo `qemu-dm` del PID.

Se incontrate un errore con il Virtual Machine Manager, ricontrollate i dati generati nel file `virt-manager.log` che risiedono nella directory `/var/log/virt-manager`. Da notare che ogni qualvolta si riavvia

il Virtual Machine Manager, esso sovrascriverà i contenuti del logfile esistente. Assicuratevi di eseguire il backup del file `virt-manager.log`, prima di riavviare il Virtual Machine manager dopo aver incontrato un errore.

19.3. POSIZIONI DIRECTORY IMPORTANTI

Sono disponibili per l'utente una serie di utility e file di log, in grado di assistervi durante la verifica e la soluzione degli errori all'interno di ambienti Red Hat Virtualization:

- Le immagini delle macchine virtuali risiedono nella directory `/var/lib/xen/images`.
- Quando riavviate il demone `xend`, verrà aggiornato `xend-database` il quale risiede nella directory `/var/lib/xen/xend-db`.
- I dump della macchina virtuale (da voi eseguiti con il comando `xm dump-core`), risiedono nella directory `/var/lib/xen/dumps`.
- La directory `/etc/xen` contiene i file di configurazione da voi utilizzati per gestire le risorse del sistema. Il file di configurazione del demone `xend` viene chiamato `xend-config.sxp`, ed è possibile utilizzare il suddetto file per implementare modifiche relative a tutto il sistema, e configurare i networking callouts.
- I comandi `proc` rappresentano un'altra risorsa usata per raccogliere informazioni relative al sistema. Le suddette voci `proc` risiedono nella directory `/proc/xen`:

`/proc/xen/capabilities`

`/proc/xen/balloon`

`/proc/xen/xenbus/`

19.4. TOOL PER IL TROUBLESHOOTING

Questa sezione riporta le applicazioni del System Administrator, le utilità per il networking, ed i tool avanzati per il debugging (per maggiori informazioni su come utilizzare questi tool per configurare i servizi di Red Hat Virtualization, consultate la documentazione per la configurazione relativa). È possibile implementare i suddetti tool standard per il System Administrator insieme ai log, per l'ausilio al processo di troubleshooting di:

- `xentop`
- `xm dmesg`
- `xm log`
- `vmstat`
- `iostat`
- `lsof`

È possibile utilizzare i suddetti log e tool avanzati per il Debugging, per un ausilio al processo di troubleshooting:

- `Xen0profile`

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

È possibile utilizzare i suddetti tool di Networking, come ausilio al processo di troubleshooting:

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

brctl è un tool di networking che controlla e configura la configurazione dell'ethernet bridge nel kernel linux di Virtualizzazione. È necessario possedere un accesso root prima di eseguire i seguenti comandi:

```
# brctl show
```

bridge-name	bridge-id	STP	enabled	interfaces
xenbr0	8000.feffffff	no		vif13.0
xenbr1	8000.ffffeffff	yes		pddummy0
xenbr2	8000.ffffffef	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
```

bridge-id	designated-root	root-port	path-cost	max-age	bridge-max-age	hello-time	bridge-hello-time	forward-delay	bridge-forward-delay	ageing-time
8000.fefffffffffff	8000.fefffffffffff	0	0	20.00	20.00	2.00	2.00	0.00	0.00	300.01

hello-timer	1.43	tcn-timer	0.00
topology-change-timer	0.00	gc-timer	0.02

19.5. TROUBLESHOOTING CON I LOG

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'altro file di log, **xend-debug.log**, è molto utile per gli amministratori di sistema poichè esso contiene informazioni ancora più dettagliate rispetto a **xend.log**. Di seguito viene riportato lo stesso error data relativo allo stesso problema di creazione del dominio del kernel:

```
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
```

Se desiderate contattare l'assistenza clienti, ed in particolare lo staff di supporto tecnico, vi preghiamo di includere una copia di entrambi i file di log.

19.6. TROUBLESHOOTING CON LA CONSOLE SERIALE

La console seriale è molto utile durante il processo di troubleshooting di problemi molto complessi. Se il kernel di Virtualizzazione si arresta inaspettatamente, e l'hypervisor genera un errore, non vi è alcun modo di risalire all'errore su di un host locale. Tuttavia, la console seriale vi permette di riprenderlo su di un host remoto. Per questo motivo è necessario configurare l'host Xen in modo da eseguire l'output dei dati su di una console seriale, e successivamente configurare l'host remoto in modo da catturare i dati in questione. A tale scopo modificate le seguenti opzioni presenti nel file **grub.conf**, in modo da abilitare una console seriale 38400-bps, su 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
```

Questo comando esegue il pipe dell'output da `/dev/ttyS0` nel file `/var/log/ttwatch/myhost.log`.

19.7. ACCESSO CONSOLE GUEST PARAVIRTUALIZZATO

I sistemi operativi guest paravirtualizzati, possiedono automaticamente una console di testo virtuale, in grado di spostare i dati sul sistema operativo Domain0. È possibile eseguire tale operazione dalla linea di comando digitando semplicemente:

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

Dove **domain100** rappresenta un nome in esecuzione oppure un numero. È possibile utilizzare il Virtual Machine Manager per visualizzare la console di testo virtuale. Sulla finestra dei dettagli della macchina virtuale, selezionate **Console seriale** dal menu **Visualizza**.

19.8. ACCESSO COMPLETO ALLA CONSOLE DEL GUEST DI VIRTUALIZZAZIONE

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
```

È possibile utilizzare il Virtual Machine Manager per visualizzare la console seriale. Sulla finestra delle informazioni relative alla macchina virtuale, selezionate **Console seriale** dal menu.

19.9. IMPLEMENTAZIONE PERISTENZA LUN

Se il vostro sistema non utilizza multipath sarà possibile utilizzare **udev** per implementare la persistenza lun. Prima di implementarla sul vostro sistema, assicuratevi di avere gli UUID appropriati. Una volta ottenuti gli UUID corretti, sarà possibile configurare la persistenza lun modificando il file **scsi_id** che risiede nella directory `/etc`. Una volta aperto questo file all'interno di un editor di testo, decommentate la seguente riga:

```
# options=-b
```

Successivamente sostituirla con questo parametro:

```
# options=-g
```

Ciò indicherà a udev di controllare tutti i dispositivi SCSI del sistema, per la presenza di UUID. Per determinare gli UUID del sistema digitate:

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

L'output dovrebbe assomigliare a quanto segue:

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

Questa riga di caratteri rappresenta l'UUID. Per ottenere l'UUID dai nomi del dispositivo, controllate ogni percorso del dispositivo in modo da assicurarvi che il numero UUID risulti essere lo stesso per ogni dispositivo. Gli UUID non cambiano quando vengono aggiunti nuovi dispositivi al sistema. Una volta controllato i percorsi del dispositivo, sarà necessario creare regole per il device naming. Per creare queste regole, è necessario modificare il file **20-names.rules** il quale risiede nella directory **/etc/udev/rules.d**. Le regole per il device naming qui create dovrebbero avere il seguente formato:

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

Sostituite il vostro UUID esistente ed il nome del dispositivo, con la entry relativa all'UUID sopra riportato. La regola in questione dovrebbe essere simile a quanto segue:

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

Ciò causa da parte del sistema, l'abilitazione di tutti i dispositivi che corrispondono a **/dev/sd***, per l'ispezione dell'UUID dato. Una volta trovato un dispositivo corrispondente, verrà creato un nodo chiamato **/dev/devicename**. Per questo esempio, il nodo del dispositivo è **/dev/mydevice**. Per finire, sarà necessario aggiungere il file **rc.local** il quale risiede all'interno della directory **/etc** con il seguente percorso:

```
/sbin/start_udev
```

IMPLEMENTING LUN PERSISTENCE WITH MULTIPATH

Per implementare la persistenza lun in un ambiente multipath, sarà necessario definire i nomi alias per i dispositivi multipath. Per questo esempio definite quattro alias attraverso la modifica del file **multipath.conf**, il quale risiede all'interno della directory **/etc/**:

```
multipath {
    wwid      3600a0b80001327510000015427b625e
    alias     oramp1
}
multipath {
    wwid      3600a0b80001327510000015427b6
    alias     oramp2
}
multipath {
    wwid      3600a0b80001327510000015427b625e
    alias     oramp3
```

```

}
multipath {
    wwid          3600a0b80001327510000015427b625e
    alias         oramp4
}

```

Ciò definisce 4 lun: **/dev/mpath/oramp1**, **/dev/mpath/oramp2**, **/dev/mpath/oramp3**, e **/dev/mpath/oramp4**. I suddetti dispositivi risiedono nella directory **/dev/mpath** . Questi nomi resteranno inalterati dopo l'esecuzione dei processi di riavvio, poiché verranno creati i nomi alias su wwid dei lun.

19.10. CONSIDERAZIONI SU SELINUX

Questa sezione contiene informazioni importanti per l'implementazione di SELinux all'interno dell'ambiente del vostro Red Hat Virtualization. Quando implementate alcune modifiche che riguardano il sistema, oppure desiderate aggiungere alcuni dispositivi, sarà necessario aggiornare di conseguenza la vostra policy SELinux. Per configurare un volume LVM per un guest, è necessario modificare il contesto di SELinux per il dispositivo a blocchi e per il volume group corrispondenti.

```

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

```

Il parametro booleano **xend_disable_trans** imposta **xend** in modalità unconfined dopo il riavvio del demone. Vi suggeriamo disabilitare la protezione per un demone singolo invece di disabilitarla per l'intero sistema. È consigliabile non rietichettare le directory come **xen_image_t** utilizzate in altre parti.

19.11. ACCESSO DEI DATI SU DI UNA IMMAGINE DEL DISCO GUEST

È possibile utilizzare due applicazioni separate, le quali possono assistervi durante l'accesso dei dati dall'interno di una immagine del disco guest. Prima di utilizzare i suddetti tool è necessario arrestare il guest. L'accesso del file system dal guest e da dom0 potrebbe causare danni al vostro sistema.

Potete usare l'applicazione kpart per gestire i dischi partizionati o gli LVM volume groups:

```

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

```

Per accedere ai volumi LVM su di una seconda partizione, eseguite nuovamente la scansione dell'LVM con vgscan, ed attivate il volume group sulla partizione (chiamata VolGroup00 per default), utilizzando il comando 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

```

Ricordate di disattivare i volumi logici con **vgchange -an**, rimuovere le partizioni con **kpartx -d**, e cancellare il dispositivo loop con **losetup -d** una volta terminato.

19.12. SITUAZIONI COMUNI DI TROUBLESHOOTING

Quando cercherete di avviare il servizio **xend** non accadrà nulla. Digitate **xm list1** e ricevete quanto segue:

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

A questo punto provate ad eseguire **xend start** manualmente, ma ricevete più errori:

```

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' )

```

Molto probabilmente avete eseguito il riavvio del vostro host in un kernel, il quale non risulta essere un kernel **xen-hypervisor**. Per correggere questo problema è necessario selezionare il kernel **xen-hypervisor** al momento dell'avvio (oppure impostare il kernel **xen-hypervisor**), come default all'interno del vostro file **grub.conf**.

19.13. ERRORI DEL DISPOSITIVO LOOP

Se utilizzate le immagini guest basate sul file, potrebbe verificarsi l'aumento del numero dei dispositivi loop configurati (il default permette un massimo di 8 dispositivi loop attivi) all'interno di una di essa. Se

avete bisogno di più di 8 dispositivi loop/guest basati sul file, sarà necessario modificare il file `/etc/modprobe.conf`. Durante la modifica del file `modprobe.conf`, è necessario includere questa riga:

```
options loop max_loop=64
```

Questo esempio ne utilizza 64, ma è possibile specificare un altro numero in modo da impostare il valore loop massimo. Potrebbe esser necessario, altresì, implementare dei loop device backed guests sul vostro sistema. Per impiegare i suddetti loop device backed guests per un sistema paravirtualizzato, utilizzate i comandi `phy: block device` o `tap:aio`. Per impiegarli invece su di un sistema completamente virtualizzato, utilizzate i comandi `phy: device` o `file: file`.

19.14. ERRORI DURANTE LA CREAZIONE DI UN GUEST

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.

Se eseguite un yum update e ricevete un nuovo kernel, il kernel di default `grub.conf` si smista sul kernel bare-metal invece di un kernel di Virtualizzazione.

Per correggere questo problema è necessario modificare l'RPM del kernel predefinito il quale risiede nella directory `/etc/sysconfig/kernel/`. Assicuratevi che il parametro `kernel-xen`, sia impostato come opzione di default all'interno del vostro file `gb.conf`.

19.15. ERRORI DELLA CONSOLE SERIALE

Non ricevete alcun output per la console seriale. Per correggere questo problema è necessario modificare `grub.conf` ed i parametri della com port:

```
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=com2l
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
```

Queste modifiche al `grub.conf` dovrebbero permettere alla vostra console seriale di funzionare correttamente. Dovreste essere in grado di utilizzare qualsiasi numero per il `ttyS`, il quale dovrebbe operare come `ttyS0`.

19.16. ERRORI NETWORK BRIDGE

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

```

Copiate `/etc/xen/scripts/network-bridge` su `/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"`).

Nel file `xend-config.sxp` la nuova riga dovrebbe riflettere il vostro nuovo script:

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

Assicuratevi di decommentare la riga che indica:

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

Sarà necessario creare uno script personalizzato se desiderate creare Xen bridge multipli. L'esempio seguente riporta due Xen bridges (chiamati `xenbr0` e `xenbr1`), e li collega rispettivamente a `eth1` e `eth0`.

```

# !/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

```

Se desiderate creare bridge aggiuntivi, utilizzate lo script d'esempio e copiate/incollate il file.

19.17. CONFIGURAZIONI LAPTOP

La configurazione di un laptop con RHEL 5.0 per un suo utilizzo su di un ambiente di rete, può presentare alcune difficoltà. I collegamenti WiFi e wired vengono selezionati costantemente durante il giorno, confondendo così Red Hat Virtualization il quale presume di avere un accesso alla stessa interfaccia. Ne risulta in un sistema il quale esegue chiamate ifup/ifdown per l'interfaccia di rete utilizzata da Red Hat Virtualization. Le schede WiFi non rappresentano il metodo ideale di collegamento di rete, poichè Red Hat Virtualization utilizza l'interfaccia di rete predefinita.

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.

Per questo esempio l'interfaccia viene chiamata `dummy0` e l'IP utilizzato è `10.1.1.1`. Lo script viene chiamato `ifcfg-dummy0` e risiede nella directory `/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

```

A questo punto sarà necessario unire `xenbr0` a `dummy0`, in modo da permettere un collegamento di rete anche quando risulta scollegato dalla rete fisica.

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
```

È buona idea abilitare NAT in domain0, per permettere l'accesso alla rete pubblica di domU. In questo modo anche gli utenti wireless possono risolvere il problema rappresentato dalle limitazioni wireless di Red Hat Virtualization. Per fare questo modificate il file `S99XenLaptopNAT`, il quale risiede nella directory `/etc/rc3.d` in modo da riflettere l'esempio riportato:

```
#!/bin/bash/
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
```

Se desiderate che la vostra rete risulti essere già impostata al momento dell'avvio, create un softlink per `/etc/rc3.d/S99XenLaptopNAT`

Quando modificate il file `modprobe.conf` includere le seguenti righe:

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

19.18. STARTING DOMAINS AUTOMATICALLY DURING SYSTEM BOOT

Starting Domains Automatically During System Boot

È possibile configurare i vostri guest in modo da eseguire un avvio automatico al momento dell'avvio del sistema. Per fare questo, è necessario modificare i link simbolici che risiedono in `/etc/xen/auto`. Il suddetto file indica i file di configurazione del guest da avviare automaticamente. Il processo d'avvio è a fasi, ciò significa che maggiore è il numero dei guest, più lungo sarà il processo d'avvio. Questo esempio mostra come utilizzare i link simbolici per il guest `rhel15vm01`:

```
[root@python xen]# cd /etc/xen
[root@python xen]# cd auto
[root@python auto]# ls
[root@python auto]# ln -s ../rhel15vm01 .
[root@python auto]# ls -l

lrwxrwxrwx 1 root root 14 Dec 14 10:02 rhel15vm01 -> ../rhel15vm01

[root@python auto]#
```

19.19. MODIFICA DI 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/Vol1Group00/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

```

19.20. FILE DI CONFIGURAZIONE DEL GUEST

Quando installate nuovi guest utilizzando il tool `virt-manager` (o `virt-install`) da Red Hat Enterprise Linux 5.0 con Virtualizzazione, i file di configurazione dei guest (posizionati nella directory `/etc/xen`), verranno modificati ed impostati automaticamente. Questo è l'esempio di file di configurazione per un guest paravirtualizzato:

```

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=ioemu,
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/hvmloader/"
vnc = 1
vncunused = 1
apic = 1
acpi = 1
pae = 1
vcpus =1
serial ="pty" # enable serial console
on_boot = 'restart'

```

19.21. CLONAZIONE DEI FILE DI CONFIGURAZIONE DEL GUEST

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.

Modificate l'indirizzo HWADDR del file **/etc/sysconfig/network-scripts/ifcfg-eth0**, in modo da corrispondere all'output del file **ifconfig eth0**, e se utilizzate indirizzi IP statici, modificate la voce IPADDR.

19.22. CREAZIONE DI UNO SCRIPT PER GENERARE GLI INDIRIZZI MAC

Red Hat Virtualization è in grado di generare un indirizzo MAC per ogni macchina virtuale al momento della creazione. Poichè risulta essere una quantità quasi illimitata di numeri presenti sulla stessa sottorete, è improbabile ottenere lo stesso indirizzo MAC. Per risolvere questo eventuale problema, è possibile scrivere uno script per la generazione di un indirizzo MAC. Questo esempio di script contiene i parametri per generare un indirizzo 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
```

19.23. CONFIGURAZIONE DELLA VIRTUAL MACHINE LIVE MIGRATION

Red Hat Virtualization può migrare macchine virtuali tra altri server i quali eseguono Red Hat Enterprise Linux 5.0 con Virtualizzazione. In aggiunta, il processo di migrazione viene eseguito attraverso un metodo offline (utilizzando il comando **xm migrate**), è da notare che è possibile eseguire una migrazione live dallo stesso comando. Ricordiamo che sono presenti altre modifiche aggiuntive riguardanti il file di configurazione **xend-config**. Questo esempio identifica le voci da modificare in modo da eseguire una migrazione corretta:

(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)

Questo parametro imposta la porta utilizzata da **xend** per il processo di migrazione. Questo valore risulta essere corretto, assicuratevi solo di rimuovere il commento che lo precede.

(xend-relocation-address)

Questo parametro risulta essere l'indirizzo in ascolto per i collegamenti socket di riposizionamento, dopo aver abilitato **xend-relocation-server** . Durante l'ascolto esso limita la migrazione ad una particolare interfaccia.

(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.

Dopo averli configurati, sarà necessario riavviare l'host per il Red Hat Virtualization in modo da poter accettare i nuovi parametri.

19.24. INTERPRETAZIONE DEI MESSAGGI D'ERRORE

Se ricevete il seguente errore:

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

Un dominio può fallire se non è disponibile sufficiente RAM. Domain0 non è in grado di diminuire il suo valore in modo tale da fornire spazio sufficiente per il guest appena creato. Per questo errore potete controllare il file **xend.log**:

```
[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.

Se ricevete il seguente errore:

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

Questo messaggio indica il tentativo di eseguire una immagine kernel del guest non supportata sul vostro Hypervisor. Ciò si può verificare quando si prova l'avvio di un kernel del guest paravirtualizzato non-PAE su di un RHEL 5.0 hypervisor. Red Hat Virtualization supporta solo kernel guest con architetture PAE ed a 64bit.

Digitare questo comando:

```
[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 (XendDomainInfo: 1449]
XendDlomainInfo.destroy: domin=1
[2006-12-14 14:55:46 xend.XendDomainInfo 3874] DEBUG (XendDomainInfo: 1457]
XendDlomainInfo.destroy:Domain(1)
```

Se desiderate eseguire un kernel a 32bit/non-PAE, avrete bisogno di eseguire il vostro guest come virtual machine virtualizzata. Per guest paravirtualizzati, se desiderate eseguire un guest a 32bit PAE, allora è necessario essere in possesso di un hypervisor a 32bit PAE. Per eseguire invece un guest a 64bit PAE, allora sarà necessario avere un hypervisor a 64bit. L'hypervisor a 32bit PAE presente con RHEL 5 i686, supporta solo sistemi operativi completamente virtualizzati a 32 bit, e paravirtualizzati a 32bit PAE. L'hypervisor a 64bit supporta solo guest paravirtualizzati.

Ciò accade quando si sposta il guest HVM completamente virtualizzato, su di un sistema RHEL 5.0. Il vostro guest potrà non eseguire l'avvio, a questo punto visualizzerete un errore sulla schermata della console. Controllate la voce PAE sul vostro file di configurazione, ed assicuratevi che pae sia uguale a 1 'pae=1'. È consigliato utilizzare una distribuzione a 32bit.

Se ricevete il seguente errore:

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

Ciò accade quando l'applicazione virt-manager non può essere lanciata. Questo errore si verifica quando non vi è alcuna voce localhost nel file di configurazione **/etc/hosts** . Controllate il file e verificate se la voce localhost è stata abilitata. Di seguito viene riportato un esempio di voce localhost incorretta:

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

Ecco un esempio di una voce localhost corretta:

```
# 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
```

Se ricevete il seguente errore (in **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.
```

In aggiunta **xend.log** visualizza i seguenti errori:

```
[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
```

Per risolvere questo problema modificate il vostro file di configurazione del guest, e successivamente modificate la voce **vif** . Una volta localizzata la voce **vif** del file di configurazione, assumendo che stiate utilizzando **xenbr0** come bridge predefinito, assicuratevi che la voce corretta assomigli alla seguente:

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

Se ricevete i seguenti errori 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)
```

In caso di presenza di un file di configurazione invalido (o errato) Python genererà questi tipi di messaggi. Per risolvere il suddetto problema, è necessario modificare il file di configurazione errato, oppure generarne uno nuovo.

19.25. RISORSE ONLINE PER IL TROUBLESHOOTING

- Red Hat Virtualization Center

█ <http://www.openvirtualization.com>

- Documentazione di Red Hat Enterprise Linux 5 Beta 2

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

- Libvirt API

█ <http://www.libvirt.org>

- virt-manager Project Home Page

█ <http://virt-manager.et.redhat.com>

- Xen Community Center

█ <http://www.xensource.com/xen/xen/>

- Panoramica sulle tecnologie di Virtualizzazione

█ <http://virt.kernelnewbies.org>

- Progetti tecnologie emergenti

█ <http://et.redhat.com>

CAPITOLO 20. RISORSE AGGIUNTIVE

Per saperne di più sulla Red Hat Virtualization, consultate le seguenti risorse.

20.1. SITI WEB UTILI

- <http://www.cl.cam.ac.uk/research/srg/netos/xen/> — Il sito web del para-virtualization machine manager Xen™, dal quale deriva Red Hat Virtualization. Il sito gestisce i binari dell'upstream del progetto Xen ed il codice sorgente insieme alle informazioni, architetture, panoramiche, alla documentazione e link relativi, riguardanti Xen e le tecnologie ad esso associate.
- <http://www.libvirt.org/> — Il sito web ufficiale per il **libvirt** virtualization API che interagisce con la struttura di virtualizzazione di un host OS.
- <http://virt-manager.et.redhat.com/> — Il sito web del progetto per il **Virtual Machine Manager** (virt-manager), l'applicazione grafica per la gestione delle macchine virtuali.

20.2. DOCUMENTAZIONE INSTALLATA

- `/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.

APPENDICE A. LAB 1

Installazione guest di Xen

Obiettivo: Installare guest di Xen per RHEL 3, 4, o 5 e Windows XP.

Prerequisiti: Una workstation installata con Red Hat Enterprise Linux 5.0, con un componente di Virtualizzazione.

In questo Lab configurerete ed installerete guest Xen Win XP e RHEL 3, 4, o 5 utilizzando i diversi tool di virtualizzazione.

Lab, Sequenza 1: Controllare l'esistenza del supporto PAE

È necessario determinare se il vostro sistema presenta un supporto PAE. Red Hat Virtualization supporta le architetture della CPU basate su x86_64 o ia64 per eseguire guest paravirtualizzati. Per eseguire i guest i386, il sistema necessita di una CPU con estensioni PAE. Numerosi laptop più obsoleti (in particolar modo quelli basati su Pentium Mobile o Centrino), non supportano PAE.

1. Per determinare se la vostra CPU possiede un supporto PAE, digitate:

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

2. Il seguente output mostra una CPU con supporto PAE. Se il comando non ritorna alcun output, allora la CPU non presenta alcun supporto PAE. Tutti gli esercizi presenti all'interno del lab, necessitano di una CPU i386 con estensione PAE, x86_64 o ia64 per poter procedere.

```
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
```

Lab, Sequenza 2: Installazione del guest paravirtualizzato Xen di RHEL5 Beta 2 utilizzando **virt-install**.

In questo lab è necessario installare un guest Xen Red Hat Enterprise Linux 5 Beta 2 usando **virt-install**.

1. Per installare il vostro guest Xen di Red Hat Enterprise Linux 5 Beta 2, al prompt digitate quanto segue: **virt-install**.
2. Quando vi verrà richiesto di installare un guest completamente virtualizzato digitate: **no**.
3. Digitate **rhe15b2-pv1** per il nome della vostra macchina virtuale.
4. Digitate 500 per l'allocazione della vostra RAM.
5. Digitate **/xen/rhe15b2-pv1.img** per il vostro disco (immagine guest).
6. Digitate 6 per la misura del vostro disco (immagine guest).
7. Digitate **yes** per abilitare il supporto grafico.

8. Digitate **nfs:server:/path/to/rhel15b2** per la posizione della vostra installazione.
9. L'installazione ha inizio. Procedere normalmente con la vostra installazione.
10. Dopo aver terminato l'installazione, digitate **/etc/xen/rhel15b2-pv1**, ed eseguite le seguenti modifiche: **#vnc=1#vncunused=1sdl=1**
11. Utilizzate un editor di testo per modificare **/etc/inittab**, aggiungendo quanto segue al file:
init 5.#id:3:initdefault:id:5:initdefault:

Lab, Sequenza 3: Installazione del guest paravirtualizzato Xen di RHEL5 Beta 2 utilizzando **virt-manager**.

In questo lab installerete un guest paravirtualizzato Xen di Red Hat Enterprise Linux 5 Beta 2 utilizzando **virt-manager**.

1. Per installare il vostro guest Xen di Red Hat Enterprise Linux 5 Beta 2, al prompt del comando digitate: **virt-manager**.
2. Sulla finestra Apri collegamento, selezionate host Xen locale e cliccate su **Collega**.
3. Start Red Hat's Virtual Machine Manager application, and from the **File** menu, click on **New**.
4. Fate clic su **Avanti**.
5. Digitate **rhel15b2-pv2** per il nome della vostra macchina, e fate clic su **Avanti**.
6. Selezionate **Paravirtualizzato** e fate clic su **Avanti**.
7. Digitate **nfs:server:/path/to/rhel15b2** per l'URL del vostro media d'installazione, e successivamente fate clic su **Avanti**.
8. Selezionate **Simple File**, digitate **/xen/rhel15b2-pv2.img** per la posizione del vostro file. Scegliete 6000 MB, e fate clic su **Avanti**.
9. Selezionate 500 per il vostro VM startup e memoria massima, e successivamente fate clic su **Avanti**.
10. Fate clic su **Fine**.

A questo punto apparirà la finestra per la console della macchina virtuale. Procedete normalmente e terminate l'installazione.

Lab, Sequenza 4: Controllo per il supporto Intel-VT o AMD-V

In questo lab dovrete determinare se il vostro sistema supporti l'hardware Intel-VT o AMD-V. Il vostro sistema deve supportare CPU abilitate Intel-VT o AMD-V, per installare correttamente sistemi operativi per guest completamente virtualizzati. Red Hat Virtualization incorpora un livello HVM generico per supportare le suddette 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. Il seguente output mostra una CPU in grado di supportare 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
```

Se il comando non ritorna alcun output, allora la CPU non presenta alcun supporto Intel-VT o AMD-V.

3. Per determinare se la vostra CPU presenta un supporto Intel-VT o AMD-V, digitate il seguente comando:

```
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
```

Lab, Sequenza 5: Installazione del guest completamente virtualizzato Xen di RHEL5 Beta 2 utilizzando `virt-install`.

In questo lab installerete un guest completamente virtualizzato Xen di Red Hat Enterprise Linux 5 Beta 2, usando **`virt-install`**:

1. Per installare il vostro guest Xen di Red Hat Enterprise Linux 5 Beta 2, al prompt digitate quanto segue: **`virt-install`**.
2. Quando è necessario installare un guest completamente virtualizzato digitate **`yes`**.
3. Digitate **`rhel15b2-pv2`** per il nome della vostra macchina virtuale.
4. Digitate 500 per l'allocazione della vostra memoria.
5. Digitate **`/xen/rhel15b2-fv1.img`** per il vostro disco (immagine guest).
6. Digitate 6 per la misura del vostro disco (immagine guest).
7. Digitate **`yes`** per abilitare il supporto grafico.
8. Digitate **`/dev/cdrom`** per l'immagine CD virtuale.
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'installazione ha inizio. Procedere normalmente con la vostra installazione.

Lab, Sequenza 6: Installazione guest completamente virtualizzato Xen di RHEL5 Beta 2 utilizzando il `virt-manager`.

In questo lab installerete un guest completamente virtualizzato Xen di Red Hat Enterprise Linux 5 Beta 2 utilizzando **`virt-manager`**:

1. Per installare il vostro guest Xen di Red Hat Enterprise Linux 5 Beta 2, al prompt del comando digitate: **virt-manager**.
2. Sulla finestra **Apri collegamento**, selezionate host Xen locale, e successivamente fate clic su **Collega**.
3. Start Red Hat's Virtual Machine Monitor application, and from the **File** menu, click on **New**.
4. Fate clic su **Avanti**.
5. Digitate **rhe15b2-fv2** per il nome del vostro sistema, e successivamente su **Avanti**.
6. Selezionate completamente virtualizzato e successivamente fate clic su **Avanti**.
7. Specificare CD-ROM o DVD, ed inserire il percorso per installare il media. Specificate la posizione dell'immagine ISO se desiderate installare da una immagine ISO. Successivamente fate clic su **Avanti**.
8. Selezionate Simple File, digitate **/xen/rhe15b2-fv2.img** per la posizione del vostro file. Specificate 6000 MB, e successivamente fate clic su **Avanti**.
9. Selezionate 500 per il vostro VM startup e memoria massima, e successivamente fate clic su **Avanti**.
10. Fate clic su **Fine**.
11. Verrà visualizzata la finestra per la console della macchina virtuale.

Procedete normalmente e terminate l'installazione.

Lab, Sequenza 7: Installazione del guest completamente virtualizzato Xen di RHEL3 utilizzando **virt-manager**.

In questo lab installerete un guest Xen di Red Hat Enterprise Linux 3, usando **virt-manager**:

1. Qui sono valide le stesse istruzioni presenti all'interno della Sequenza 6.

Lab, Sequenza 8: Installazione di un guest Xen completamente virtualizzato di RHEL4 utilizzando **virt-manager**

In questo lab installerete un guest Xen di Red Hat Enterprise Linux 4 utilizzando **virt-manager**.

1. Qui sono valide le stesse istruzioni presenti all'interno della Sequenza 6.

Lab, Sequenza 9: Installazione di un guest completamente virtualizzato Xen di Windows XP utilizzando **virt-manager**.

In questo lab installerete un guest completamente virtualizzato Xen di Windows XP utilizzando **virt-manager**.

1. Per installare il vostro Red Hat Enterprise Linux 5 sul vostro host Windows XP, al prompt del comando digitate: **virt-manager**.
2. Nella finestra **Apri collegamento**, selezionate host Xen locale, e fate clic su **Collega**.
3. Start Red Hat's Virtual Machine Manager application, and from the **File** menu click on **New**.

4. Fate clic su **Avanti**.
5. Digitate **winxp** per il nome del vostro sistema, e successivamente fate clic su **Avanti**.
6. Selezionate completamente virtualizzato e successivamente fate clic su **Avanti**.
7. Specificare CD-ROM o DVD, ed inserire il percorso per installare il media. Specificate la posizione dell'immagine ISO se desiderate installare da una immagine ISO. Successivamente fate clic su **Avanti**.
8. Selezionate Simple File, digitate **/xen/winxp.img** per la posizione del vostro file. Specificate 6000 MB, e successivamente fate clic su **Avanti**.
9. Selezionate 1024 per la vostra VM Startup e memoria massima, e selezionate 2 per le vostre VCPU. Successivamente fate clic su **Avanti**.
10. Fate clic su **Fine**.
11. A questo punto apparirà la finestra per la console della macchina virtuale. Procedete normalmente e terminate l'installazione.
12. Scegliete di formattare la partizione **C:** nel formato file system FAT. Red Hat Enterprise Linux 5 non presenta i moduli del kernel NTFS. Il montaggio o la scrittura dei file sull'immagine del guest Xen potrebbe non essere così semplice, se è necessario formattare la partizione in un formato file system NTFS.
13. Dopo aver riavviato per la prima volta il sistema, modificate l'immagine guest **winxp**:
losetup /dev/loop0 /xen/winxp.imgkpartx -av /dev/loop0mount /dev/mapper/loop0p1 /mntcp -prv \$WINDOWS/i386 /mnt/. Ciò correggerà il problema che potrete incontrare in una fase successiva dell'installazione di Windows.
14. Riavviate manualmente il guest Xen digitando: **xm create -c winxp/**.
15. Nella finestra Virtual Machine Manager, selezionate il guest Xen **winxp** e fate clic su **Apri**.
16. A questo punto apparirà la finestra per la console della macchina virtuale. Procedete normalmente e terminate l'installazione.
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. Se la console del guest Xen si blocca, fate clic su **shutdown**, ed apportate le seguenti modifiche in **/etc/xen/winxp:#vnc=1#vncunused=1sd1=1#vcpus=2**
19. Ripetete fase 14 e procedete normalmente con l'installazione.

APPENDICE B. LAB 2

Migrazione Live

Obiettivo: Configurare ed eseguire una migrazione Live tra due host.

Prerequisito: Due workstation con Red Hat Enterprise Linux 5.0 Beta 2 con piattaforma di Virtualizzazione, ed un guest di Xen Fedora Core 6 su di una delle due workstation.

Per questo lab sarà necessario configurare il processo di migrazione e l'esecuzione di una migrazione Live tra due host.

Introduzione: Prima di iniziare

Per questo lab avete bisogno di due host di Virtualizzazione: un guest di Xen ed uno storage condiviso. È necessario collegare i due host di Virtualizzazione tramite un cavo UTP. Uno degli host di Virtualizzazione esporta uno storage condiviso tramite NFS. Sarà necessario configurare entrambi gli host di virtualizzazione in modo da farli migrare correttamente. Il guest di Xen risiede sullo storage condiviso. Sul guest di Xen sarà necessario installare uno streaming server. Assicuratevi che lo streaming server venga eseguito senza interruzioni sul guest di Xen, in questo modo la migrazione Live avviene tra un host di Virtualizzazione e l'altro. Per il lab 2 farete riferimento ai due host di Virtualizzazione **host1** e **host2**.

Sequenza 1: Configurazione xend (entrambi gli host di Xen)

Nella procedura di questo Lab dovrete configurare xend in modo da avviarsi come server HTTP e come un server di relocation. Il demone xend non inizializza il server HTTP per default. Esso avvia il server di gestione del socket del dominio UNIX (per **xm**), iniziando altresì una comunicazione con xend. Per abilitare la migrazione live tra macchine è necessario eseguire una configurazione atta a supportare una migrazione live:

1. Per creare un backup del vostro file **xend-config.sxp**:

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

2. Modificate **/etc/xen/xend-config.sxp** ed apportate i seguenti cambiamenti:

```

#(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. Riavviate **xend:service** e **xend restart**.

Sequenza 2: Esportazione di uno storage condiviso tramite NFS

Per la procedura di questo lab dovrete configurare NFS ed utilizzarlo come uno storage condiviso.

1. Modificate **/etc/exports** ed includere la riga: **/xen *(rw, sync, no_root_squash)/**
2. Salvate **/etc/exports** e riavviate il server NFS. Assicuratevi che il server NFS venga avviato da **default:service nfs startchkconfig nfs on**.

3. Dopo aver avviato il server NFS sull'**host1**, sarà possibile montarlo su **host2:mount host1:/xen** .
4. Ora avviate il guest di Xen su **host1** e selezionate **fc6-pv1** (o **fc6-pv2** dal Lab 1):

```
■ xm create -c fc6-pv1
```

Sequenza 3: Installazione dello streaming server del guest di Xen

In questo lab dovrete installare lo streaming server, gnump3d, per le dimostrazioni relative. Selezionerete gnump3d poichè esso supporta i file OGG vorbis, e risulta essere facile da installare, configurare e modificare.

1. Scaricate **gnump3d-2.9.9.9.tar.bz2** tarball da <http://www.gnump3d.org/> . Estraete il tarball e nella directory **gnump3d-2.9.9.9/** compilate ed installate **gnump3d application:tar xvjf gnump3d-2.9.9.9.tar.bz2cd gnump3d-2.9.9.9/make install**
2. 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**
3. Iniziate lo streaming server digitando


```
■ command:gnump3d
```
4. Su uno dei due host di Xen eseguite Movie Player. Se non è installato, installate gli rpm totem e iso-codec prima di eseguire Movie Player. Fate clic su Applicazioni, successivamente Suono & Video, e per finire Movie Player.
5. Fate clic su Movie e successivamente Apri posizione. Inserite <http://guest:8888/TruthHappens.ogg>.

Sequenza 4: Esecuzione di una migrazione Live

1. Eseguite il file **TruthHappens.ogg** su di uno dei due host di Xen.
2. Eseguite una migrazione live da **host1** a **host2**:

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

3. Aprite la finestra di terminali multipli su entrambi gli host di Xen con il seguente comando:

```
■ watch -n1 xm list
```

4. Osservate l'inizio del processo di migrazione live. Notate quanto tempo è necessario per completare tale processo.

Sequenza di prova: Configurazione del server VNC dall'interno di un guest di 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.

APPENDICE C. CRONOLOGIA

Revisione 2.0-15.400
Rebuild with publican 4.0.0

2013-10-31

Rüdiger Landmann

Revisione 2.0-15
Rebuild for Publican 3.0

2012-07-18

Anthony Towns

Revisione 5.1.0-10
Resolve: #245684
Aggiornamenti del contenuto

Fri Aug 3 2007

Michael Hideo Smith