***By Falko Timme***
Published: 2009-02-08 18:56

# KVM Guest Management With Virt-Manager  On Ubuntu 8.10

Version 1.0
Author: Falko Timme <ft [at] falkotimme [dot] com>
Last edited 02/05/2009

Virt-Manager (Virtual Machine Manager) is a graphical interface for managing KVM and Xen guests on the local and also on remote systems. You can use it to start, stop, pause, create, and delete guests, and you can connect to the guests using the graphical console. This guide shows how you can use it to manage KVM guests on an Ubuntu 8.10 desktop.

I do not issue any guarantee that this will work for you!

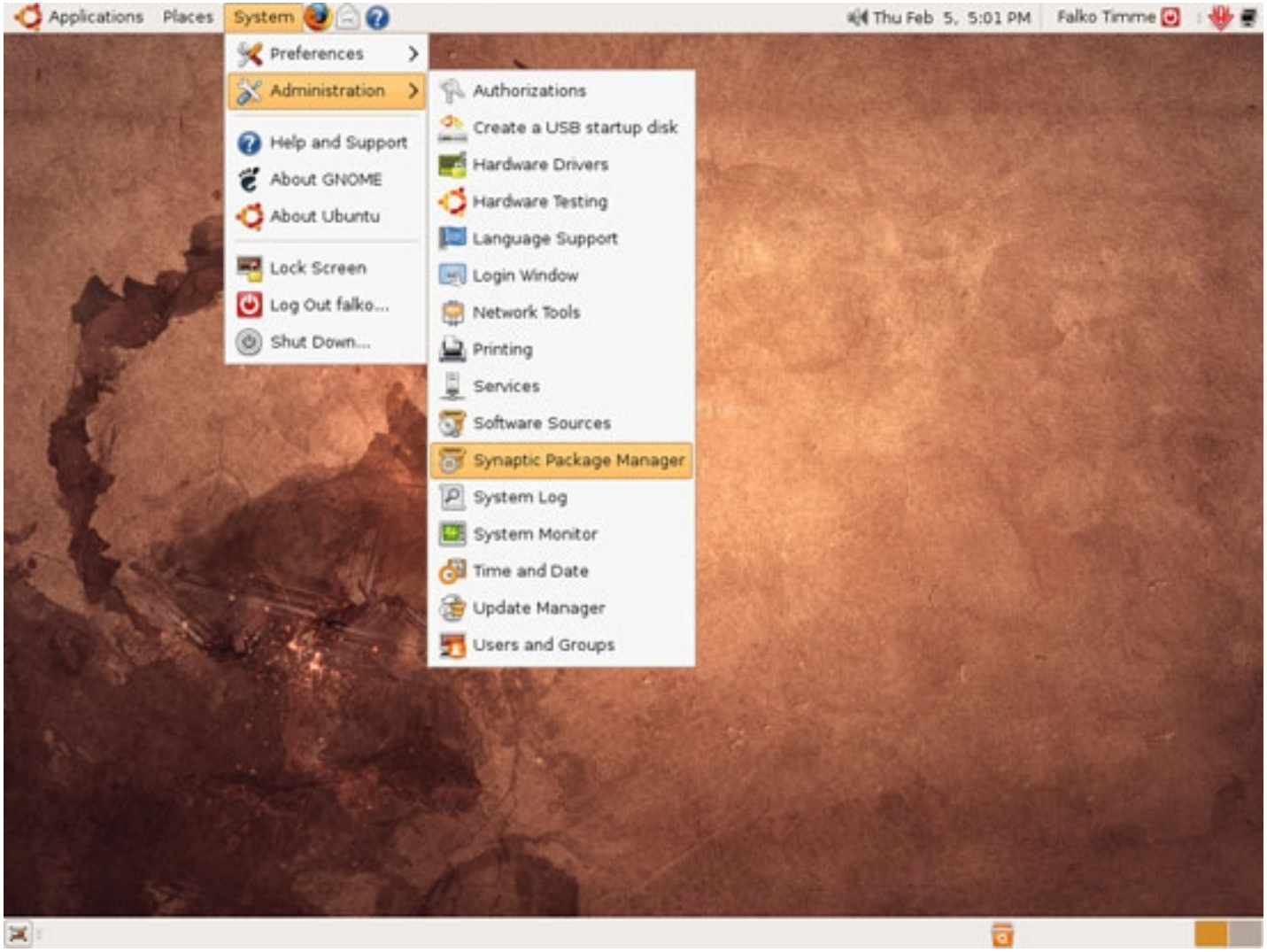
## 1 Preliminary Note

Make sure that you have a KVM host (can be on the same machine or on a remote system) that you can connect to using virt-manager. You can use this guide to set up KVM on an Ubuntu 8.10 host: **Virtualization With KVM On Ubuntu 8.10**.

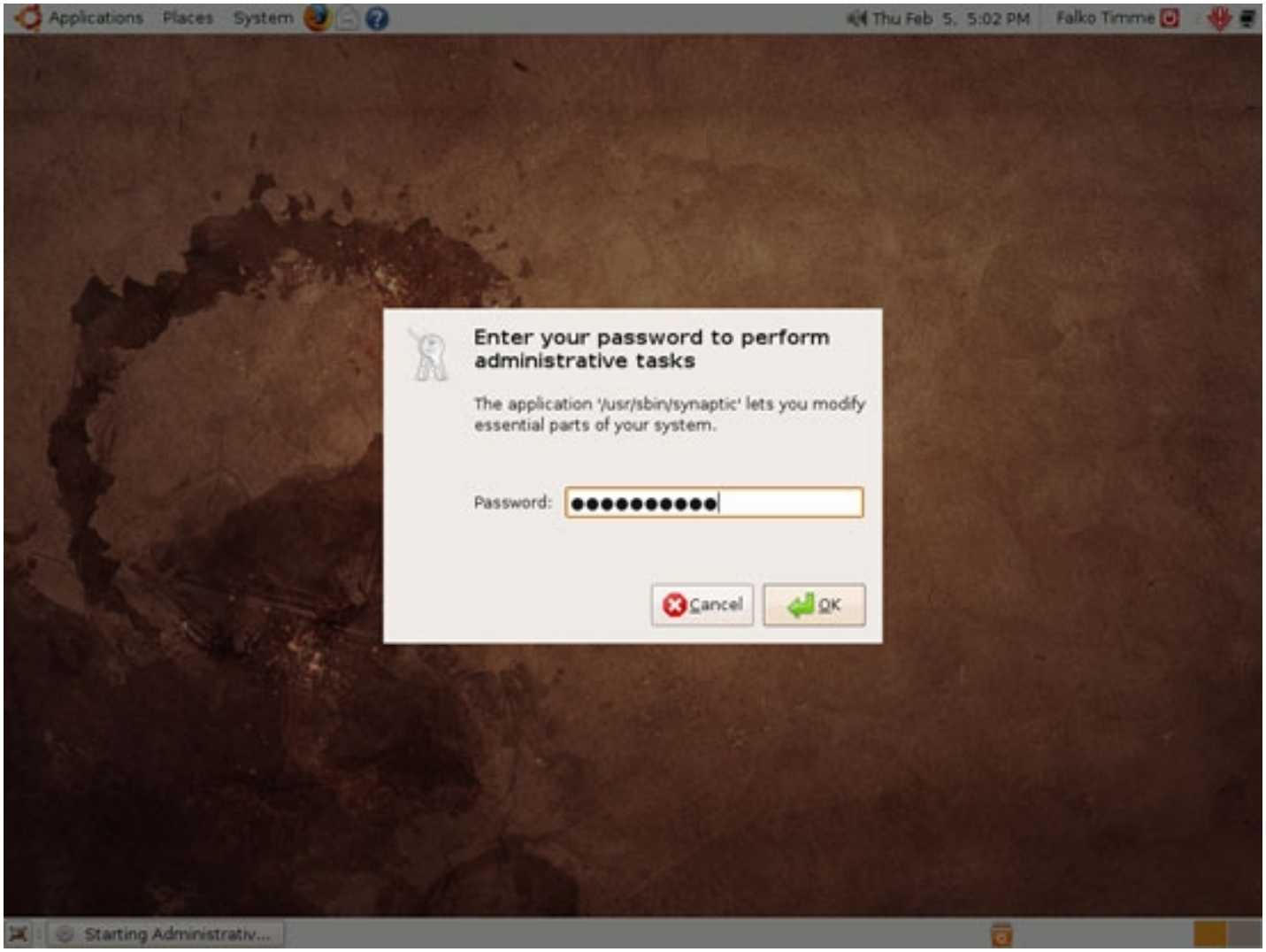
Creating guests is supported only if virt-manager is installed directly on the KVM host.

## 2 Installing Virt-Manager

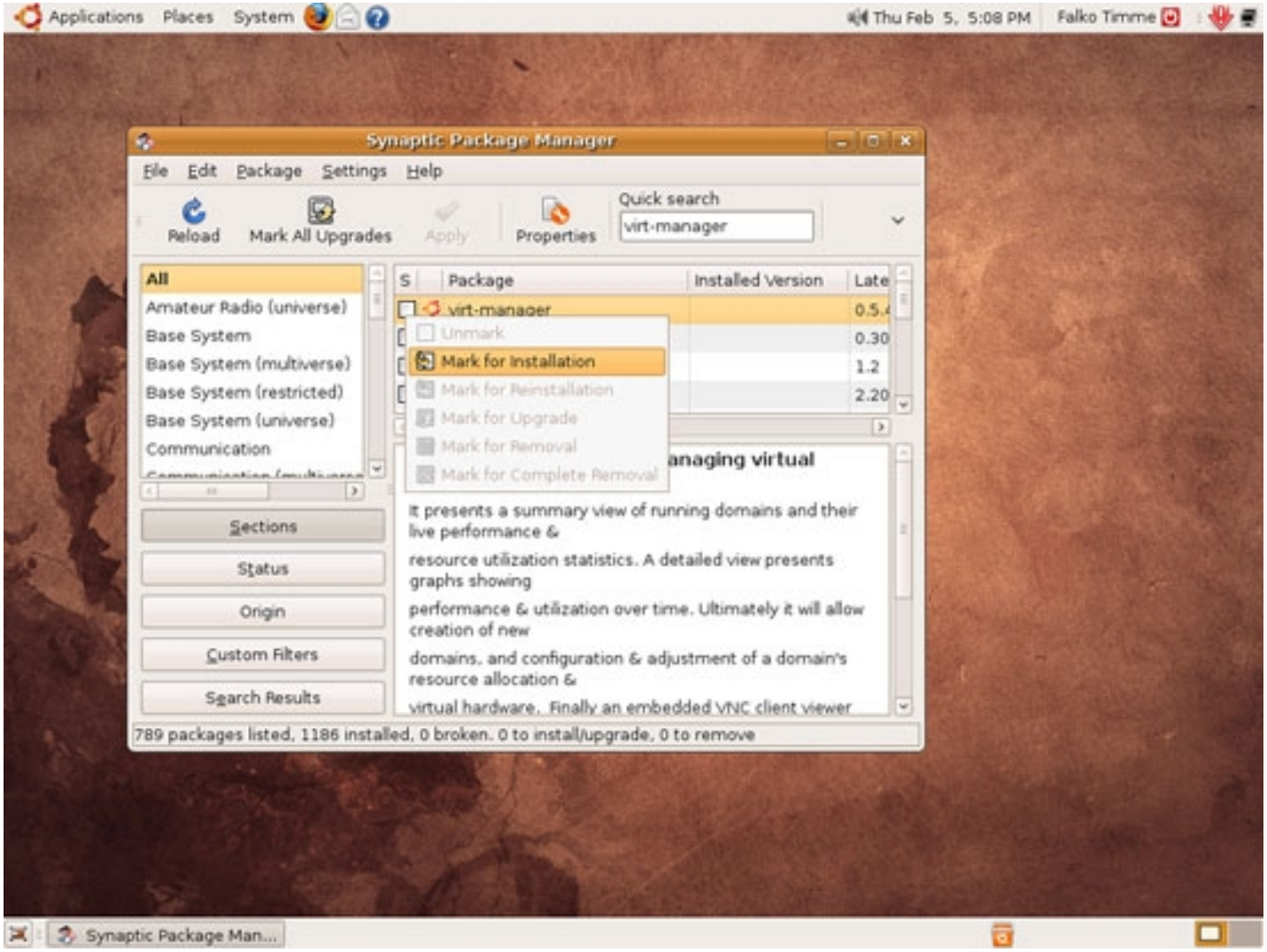
Open the *Synaptic Package Manager* (*System > Administration > Synaptic Package Manager*):



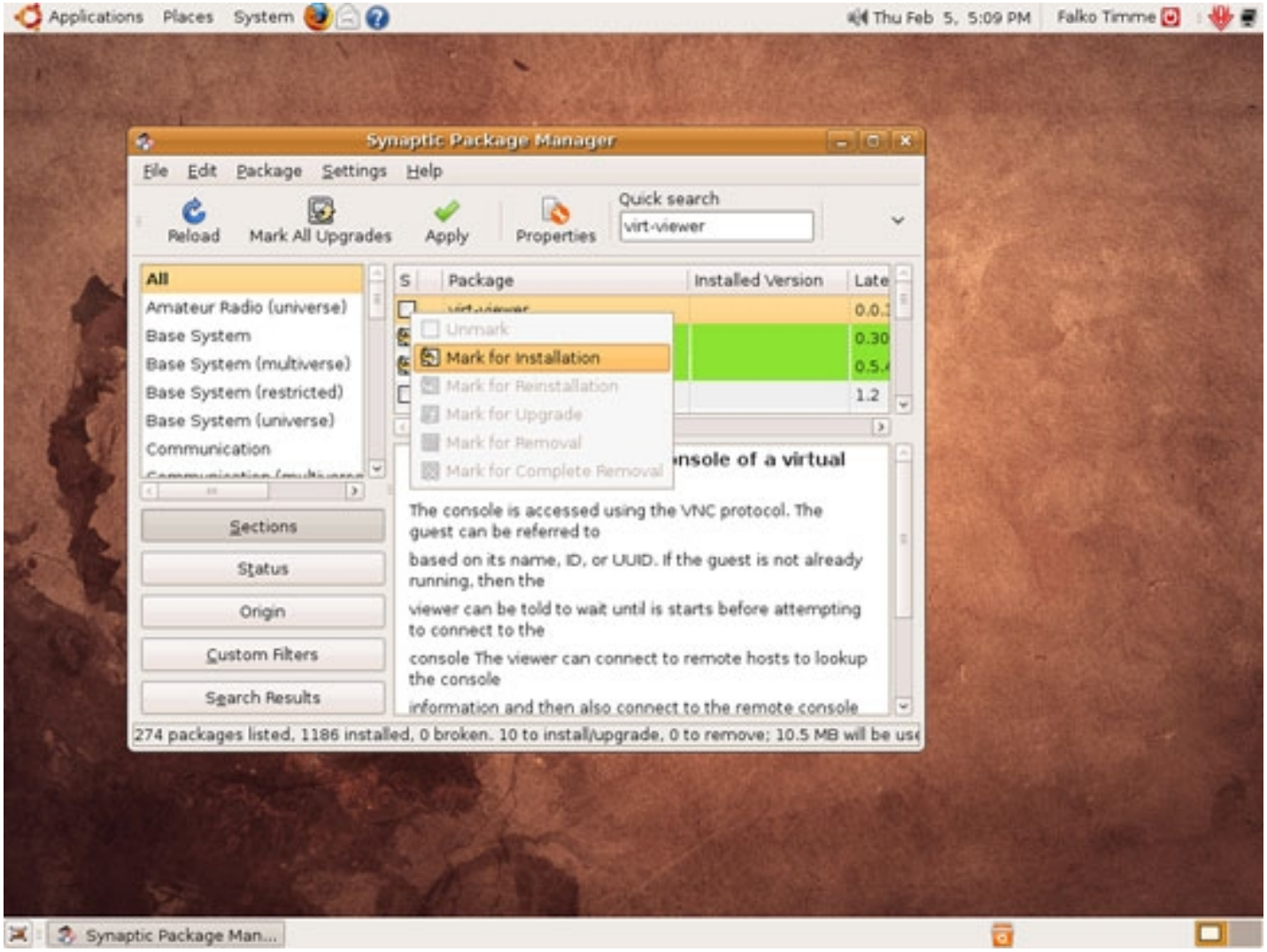

Type in your password:



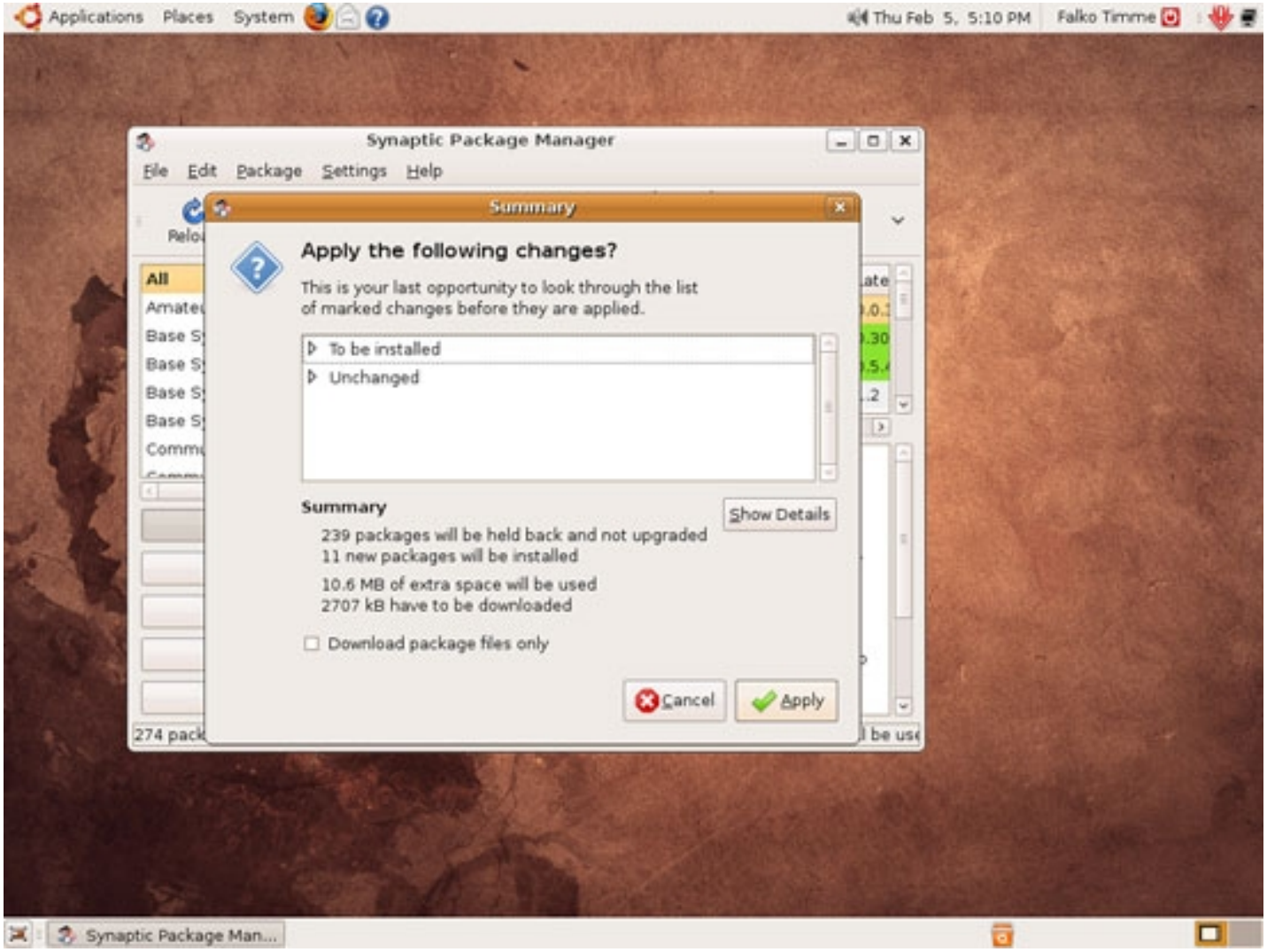

In Synaptic, search for *virt-manager*...



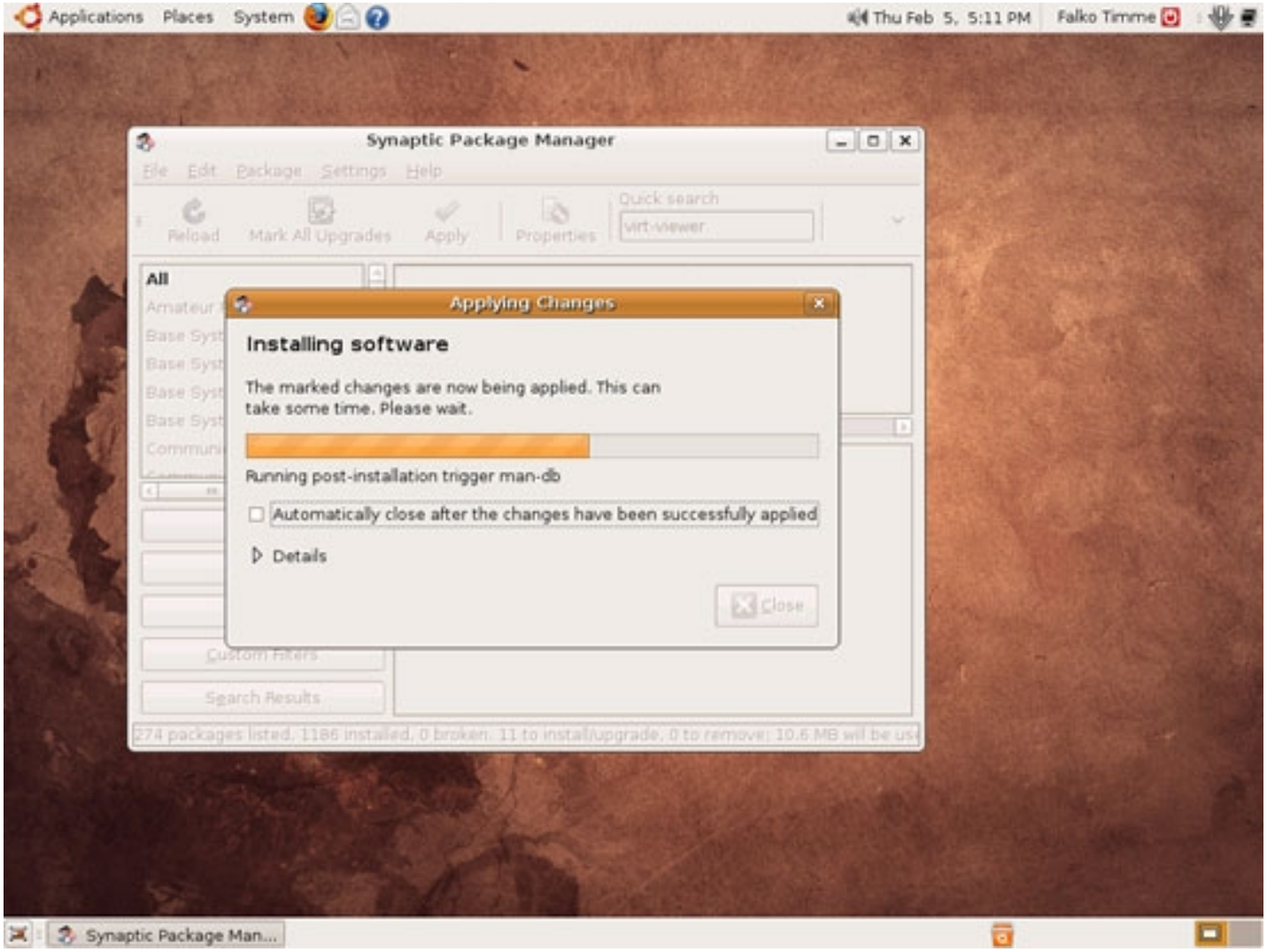

... and *virt-viewer* and mark both packages for installation; click on *Apply* afterwards:



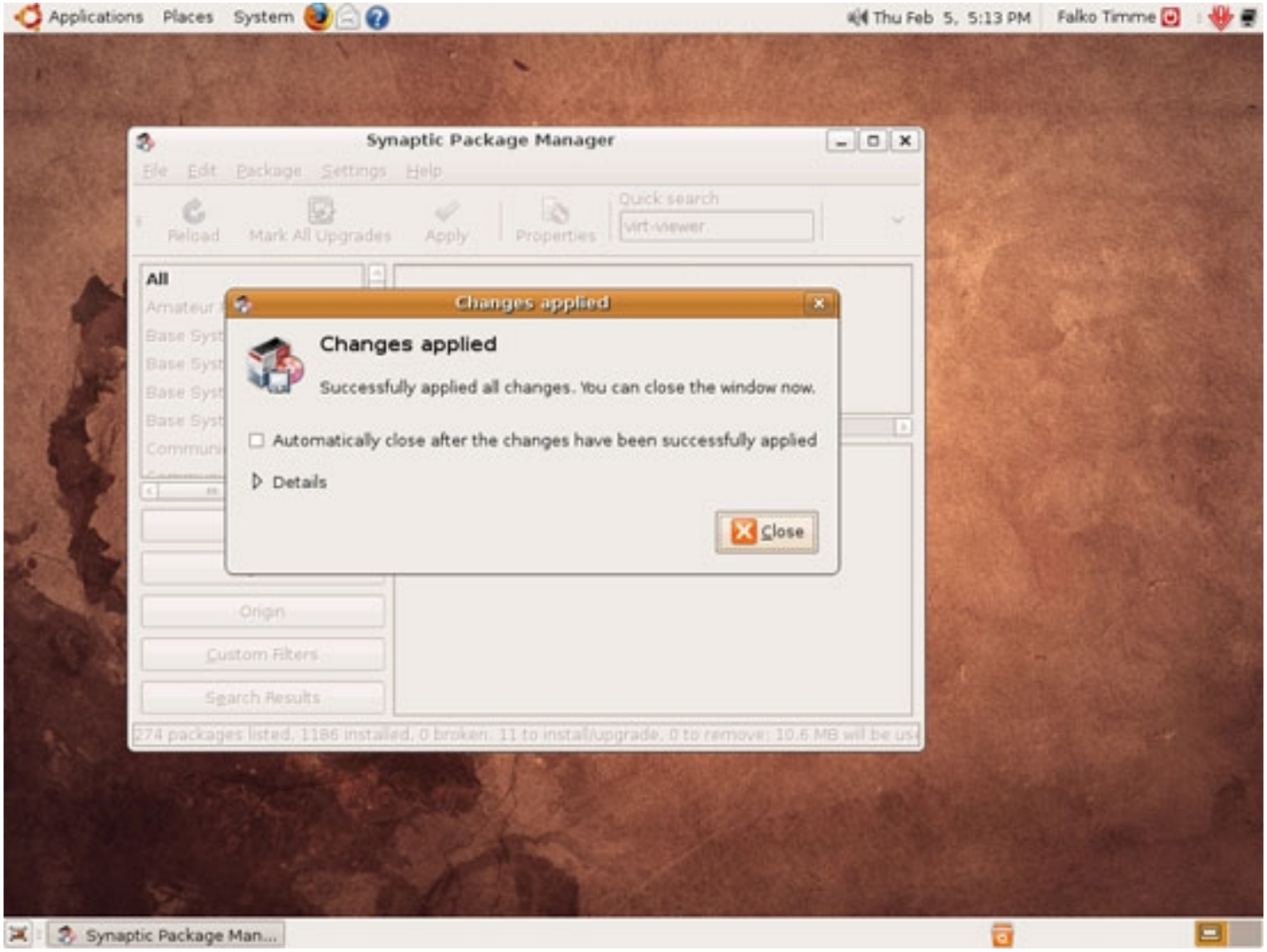

Confirm your selection by clicking on *Apply* again:




Afterwards, both packages and their dependencies are installed:




Click on `Close` afterwards and leave Synaptic:




## 2.1 Make Virt-Manager Run With root Privileges

Normally virt-manager is run as a normal user (i.e., without root privileges). This is fine as long as you only want to start, stop, or pause KVM guests. If

you want to create or delete KVM guests (works only if KVM is installed on the same system!), virt-manager must be run with root privileges.

If KVM is installed on the same system as virt-manager, and you want to create or delete KVM guests, open a terminal (*Applications > Accessories > Terminal*) and open the file */usr/bin/virt-manager*:

```
sudo gedit /usr/bin/virt-manager
```

Insert the word *sudo* between *exec* and *python* so that the file looks as follows:

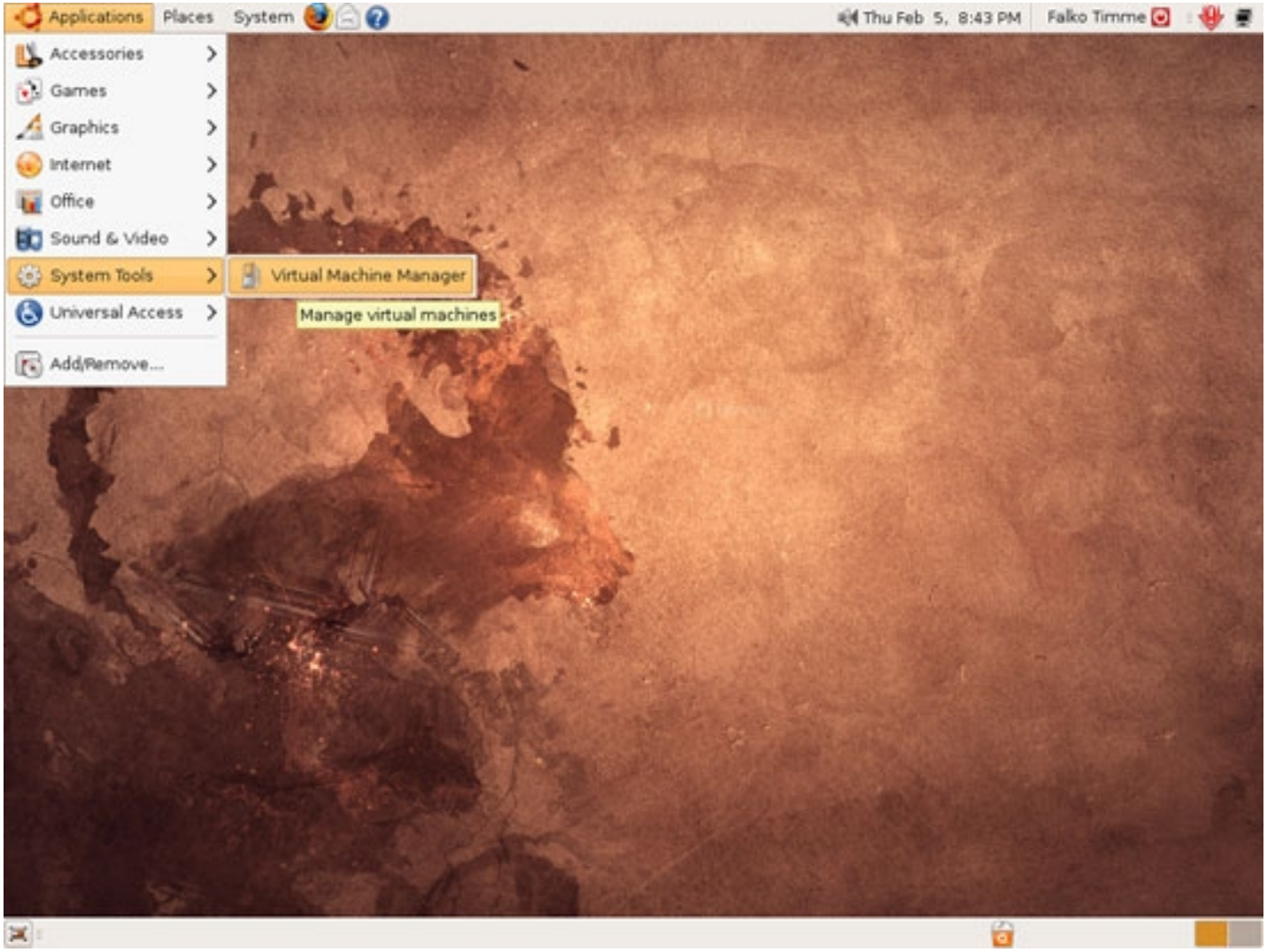
```
#!/bin/sh

exec sudo python "/usr/share/virt-manager/virt-manager.py" "$@"
```

That's it - virt-manager will be executed with root privileges from now on.

## 3 Using Virt-Manager

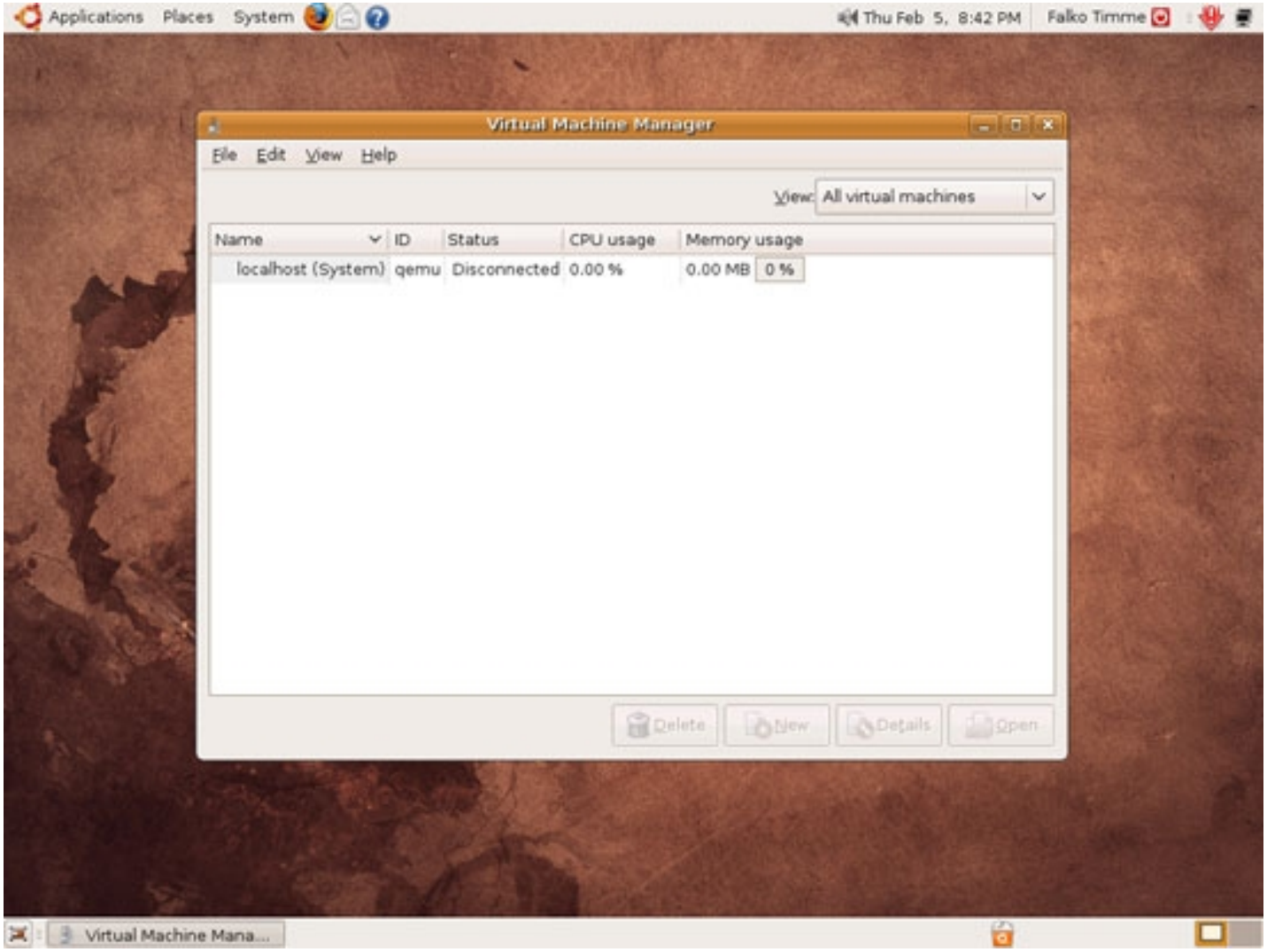
You can now start virt-manager (*Applications > System Tools > Virtual Machine Manager*):



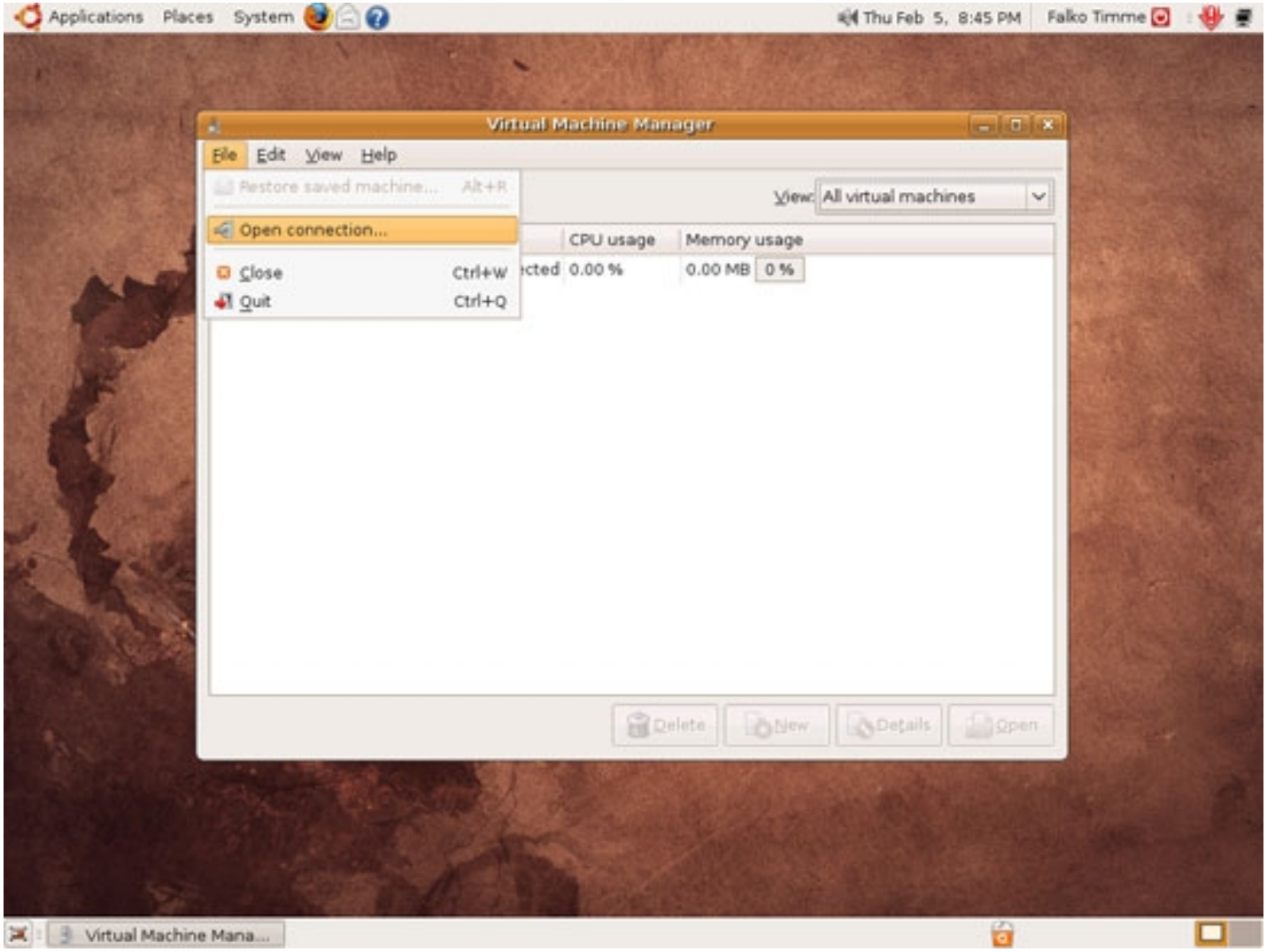

Depending on if you have followed chapter 2.2, you are asked or not asked for your password. If you have followed chapter 2.2, but virt-manager doesn't start, open a terminal and start it as follows:

```
sudo virt-manager
```

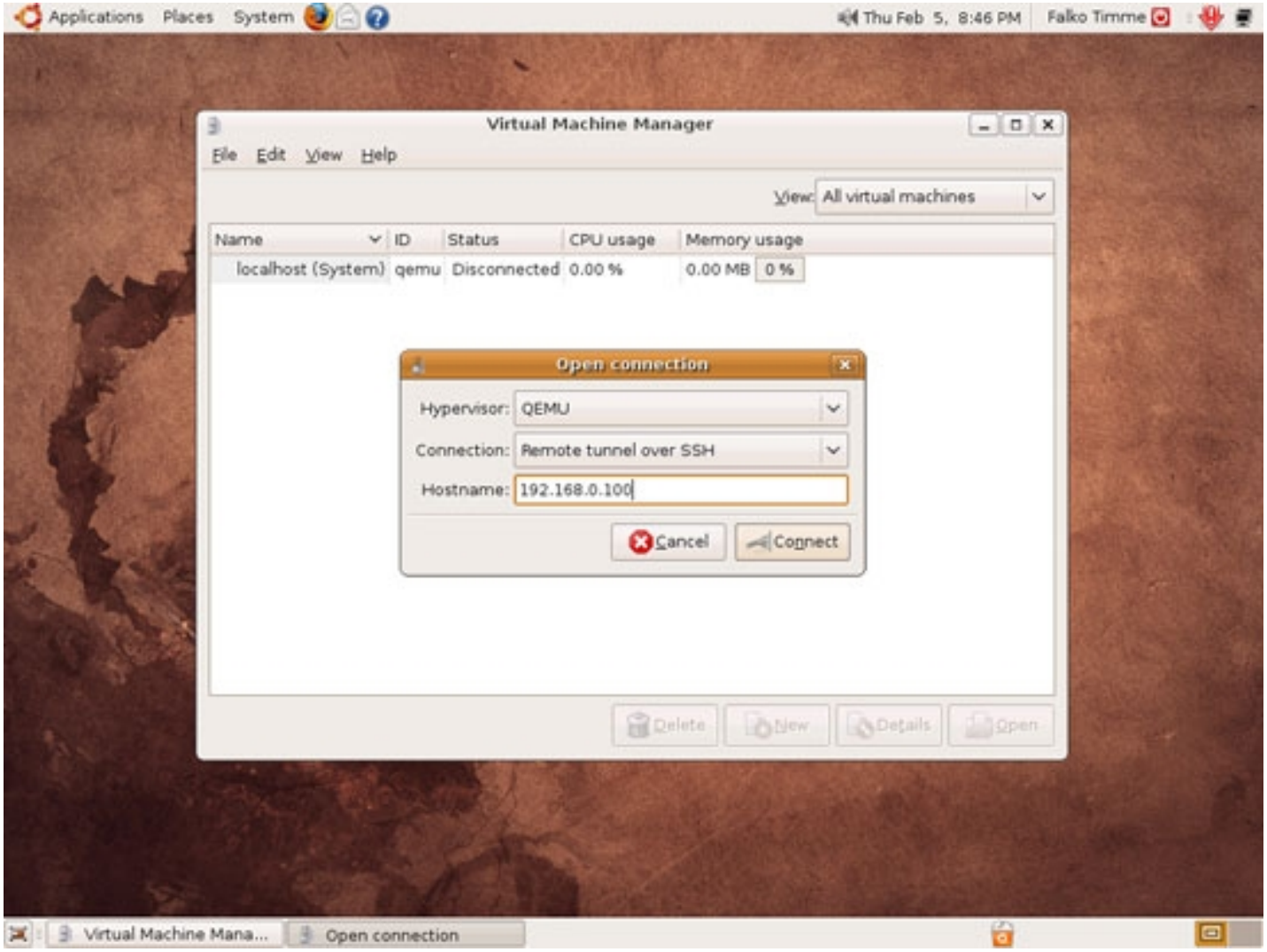
This is how virt-manager looks:



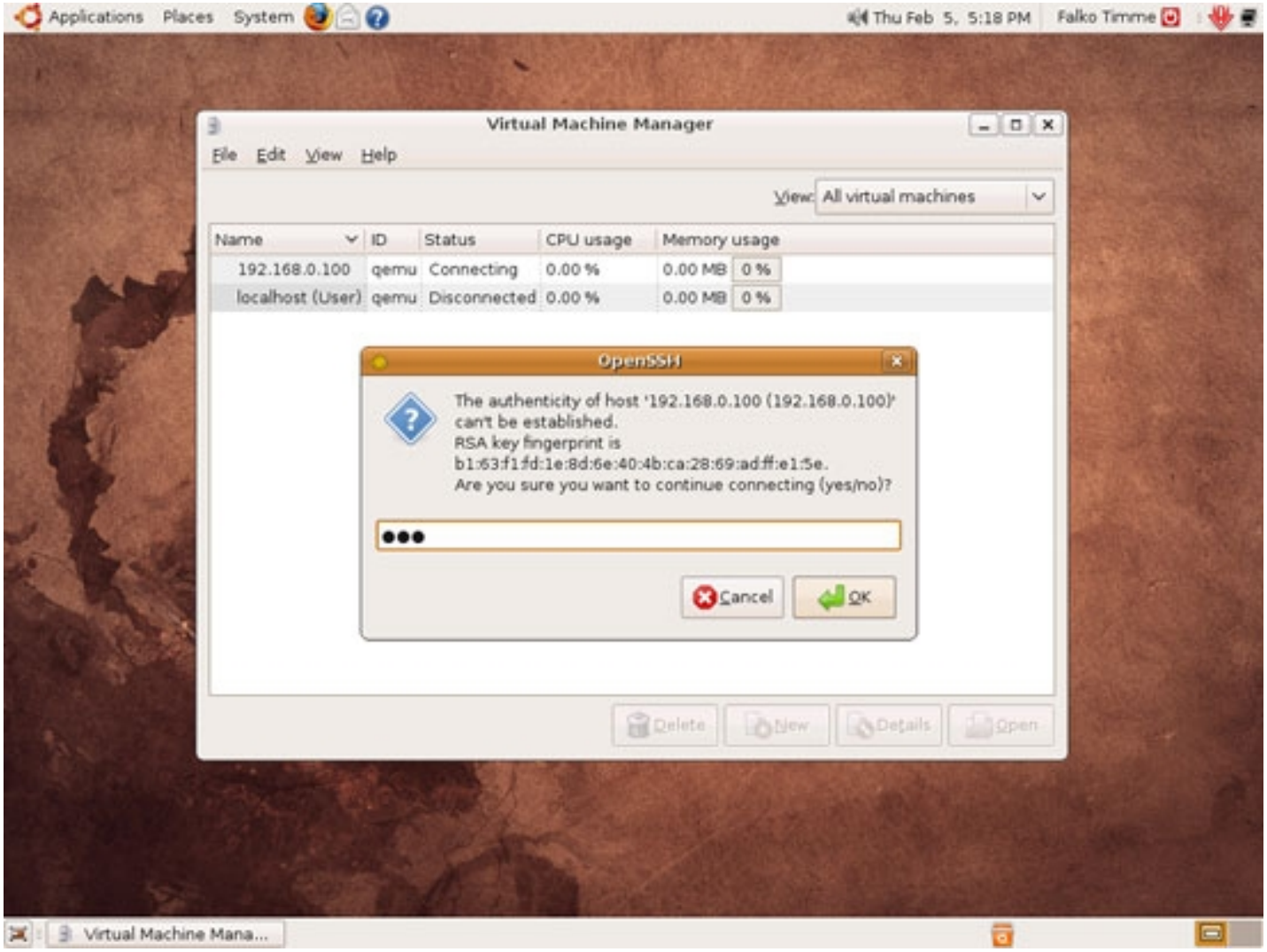

If your KVM host is on a remote system, you must connect to it - go to *`File > Open connection...`*:



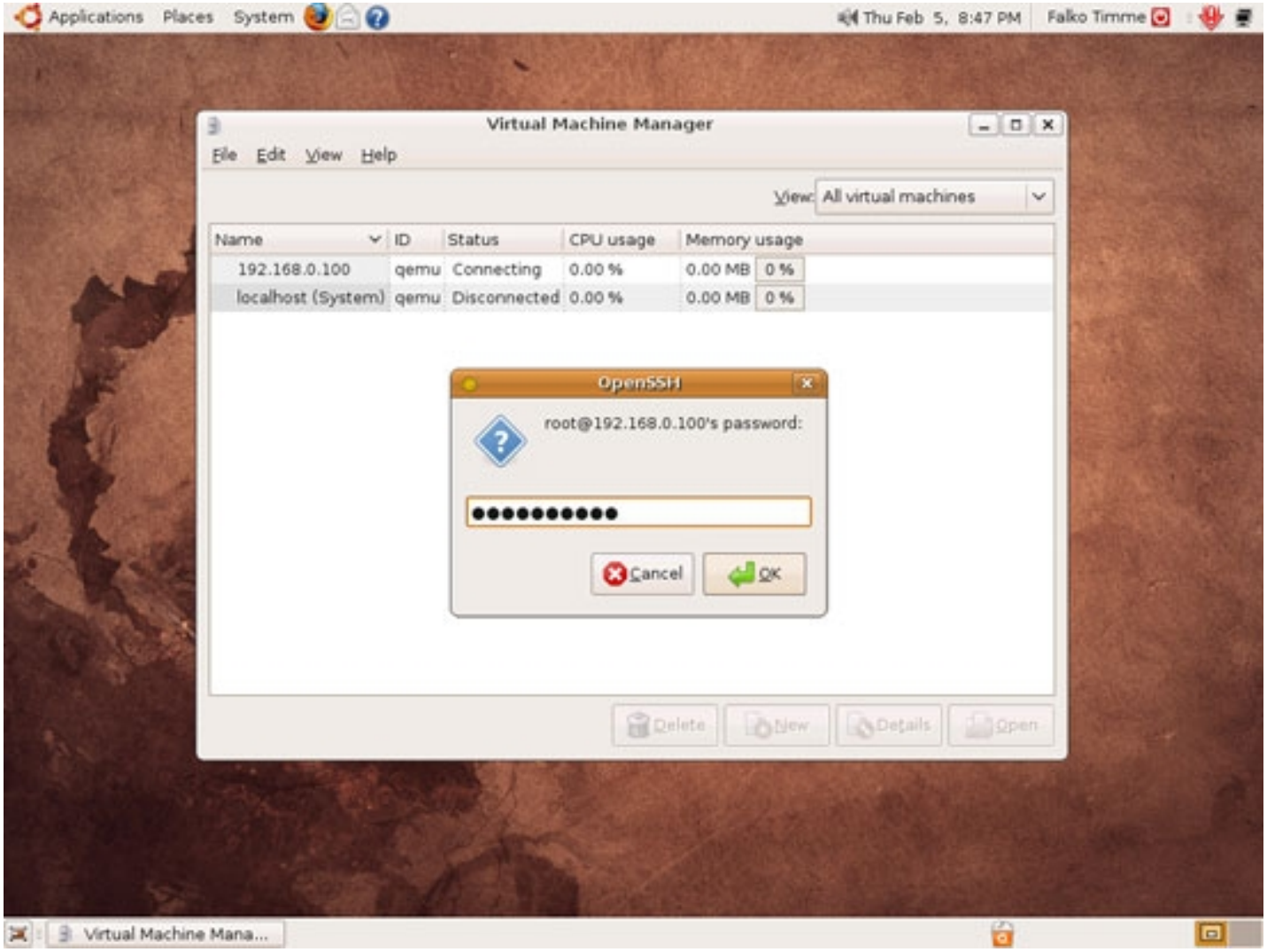

Select *QEMU* and *Remote tunnel over SSH* and type in the hostname or IP address of the KVM host:




If this is the first time you're trying to connect to the remote KVM host, you must type in *yes*:




Next type in the root password of the KVM host.

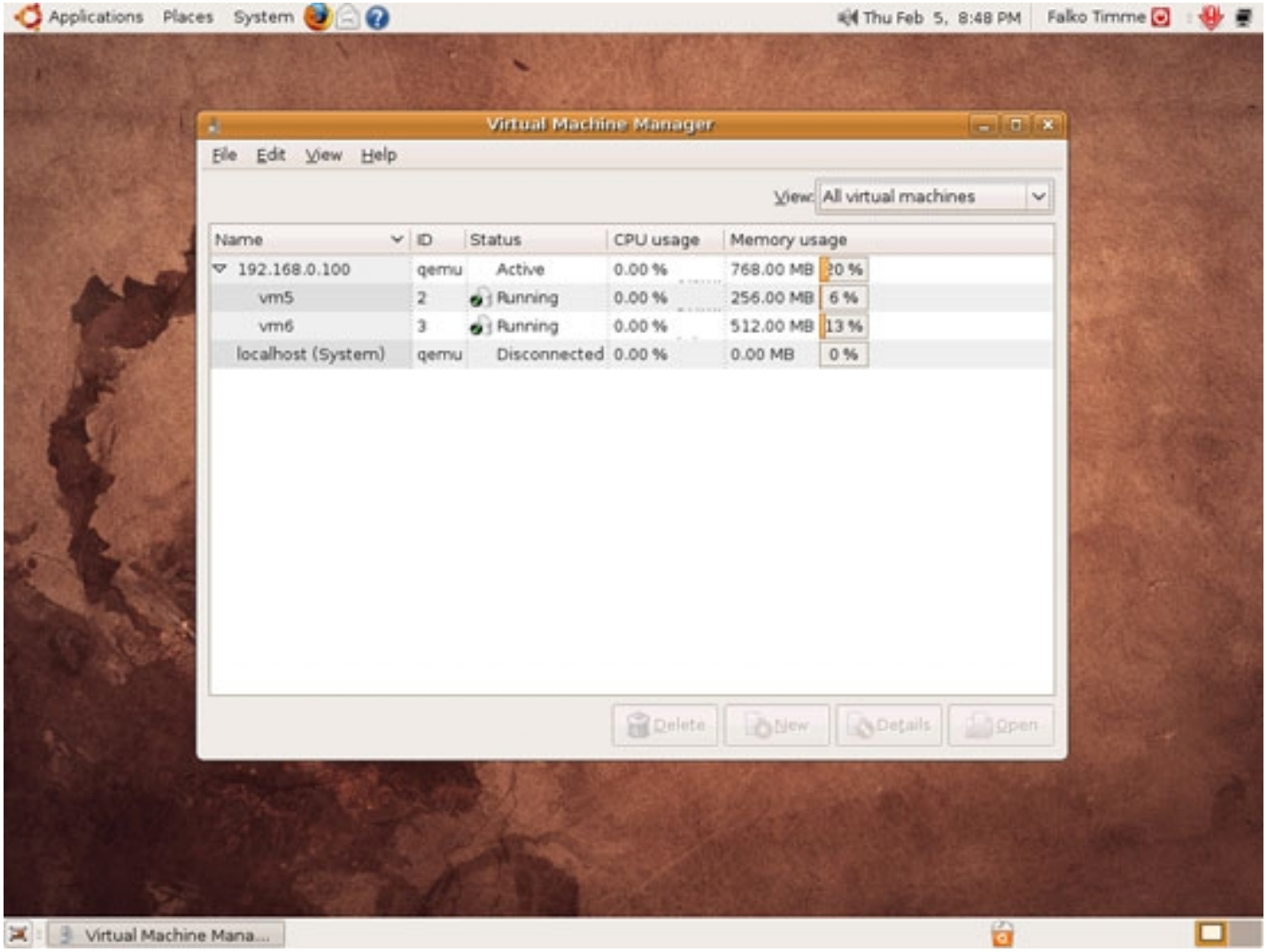



(Please note that the root account must be enabled, and that root logins must be allowed on the remote system. If the remote system is an Ubuntu system, run

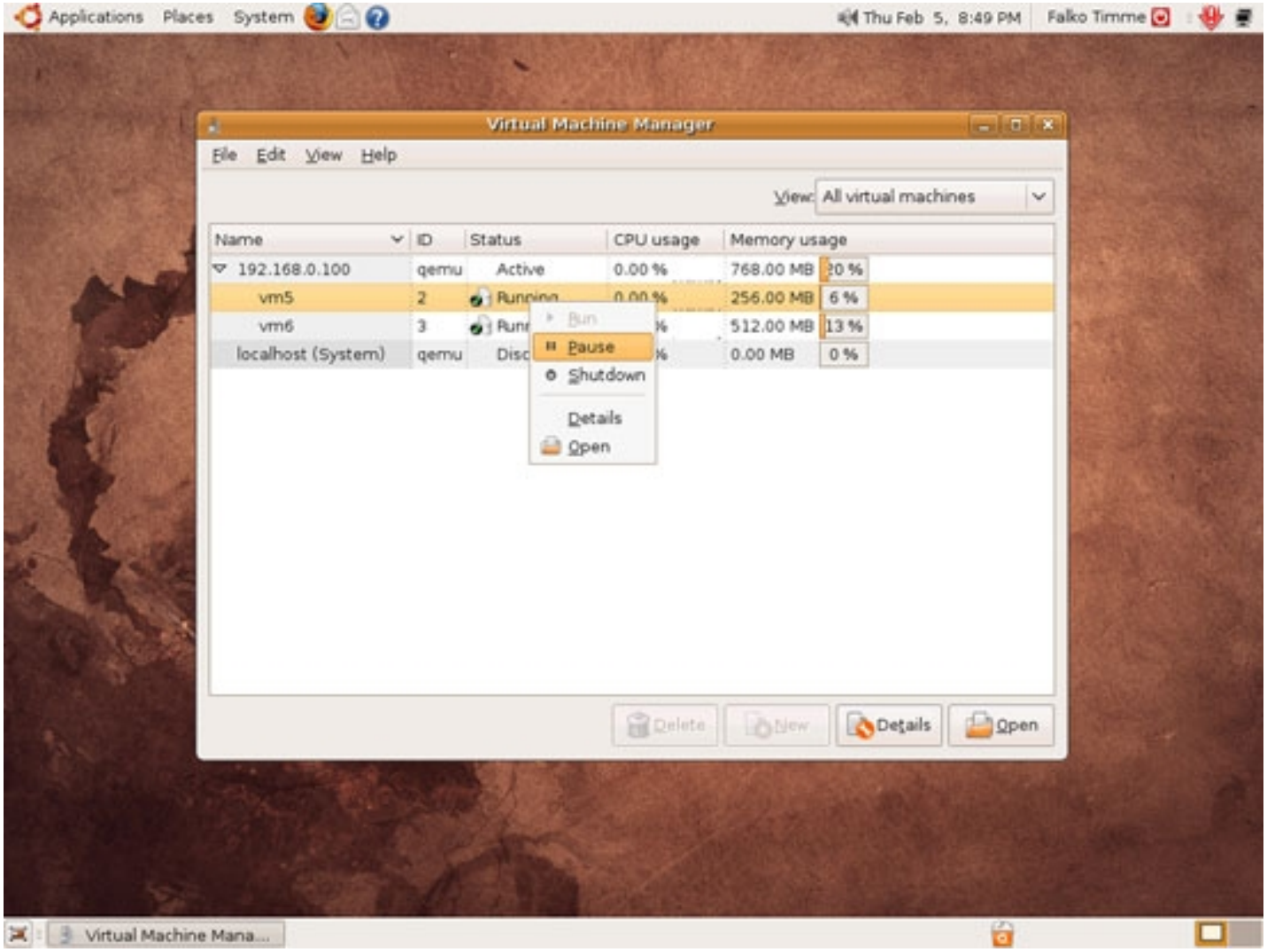
```
sudo passwd root
```

to enable the root account; to check if root logins are allowed check the directive `PermitRootLogin` in `/etc/ssh/sshd_config` - you might have to restart the SSH daemon afterwards. )

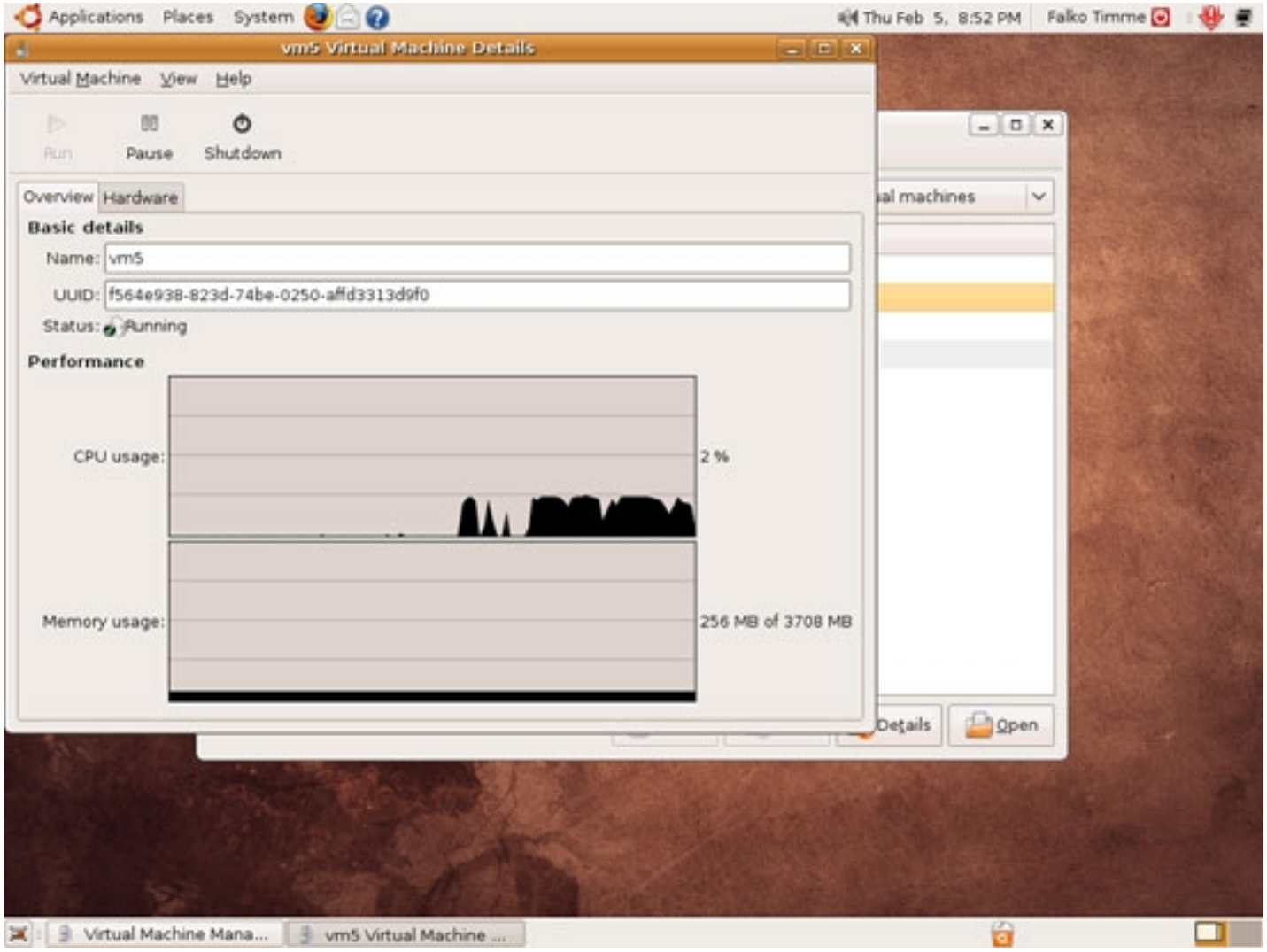
You should now see a list of virtual machines running on the KVM host:



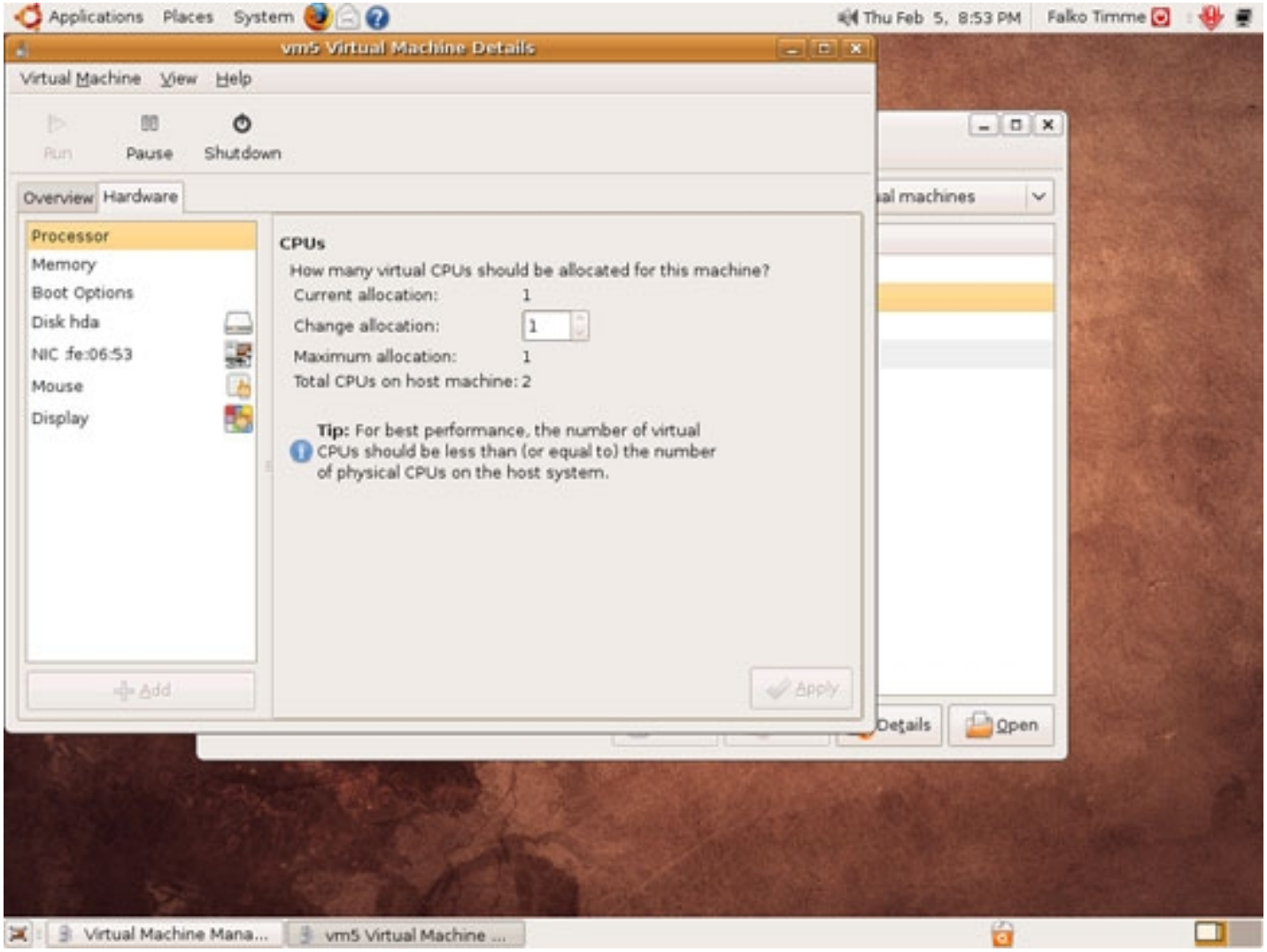

By right-clicking on a virtual machine, you can start/pause/stop it; if you select `Details`,...



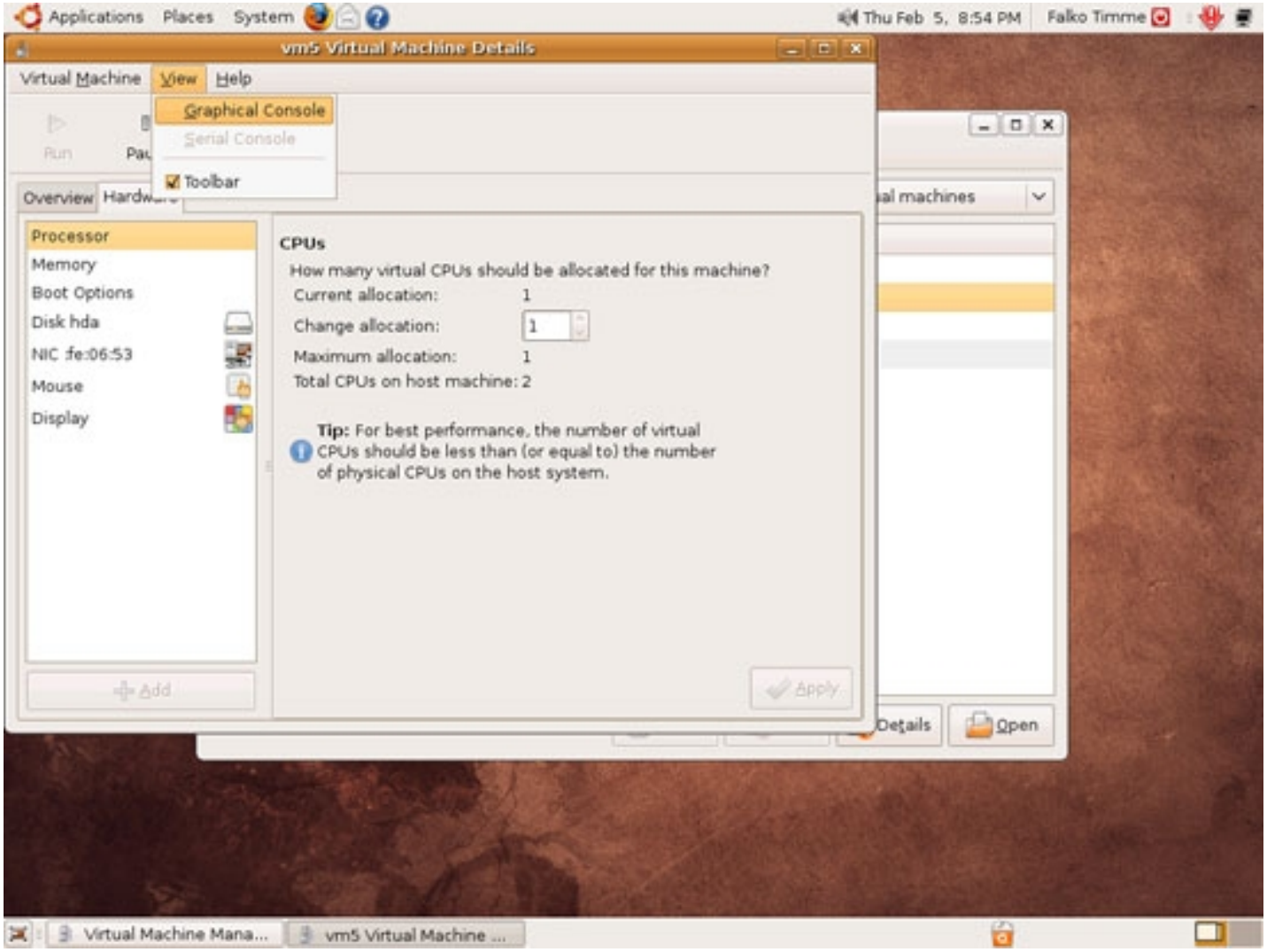

... you will see something like this (performance graphs of the virtual machine). By clicking on the *Hardware* tab,...



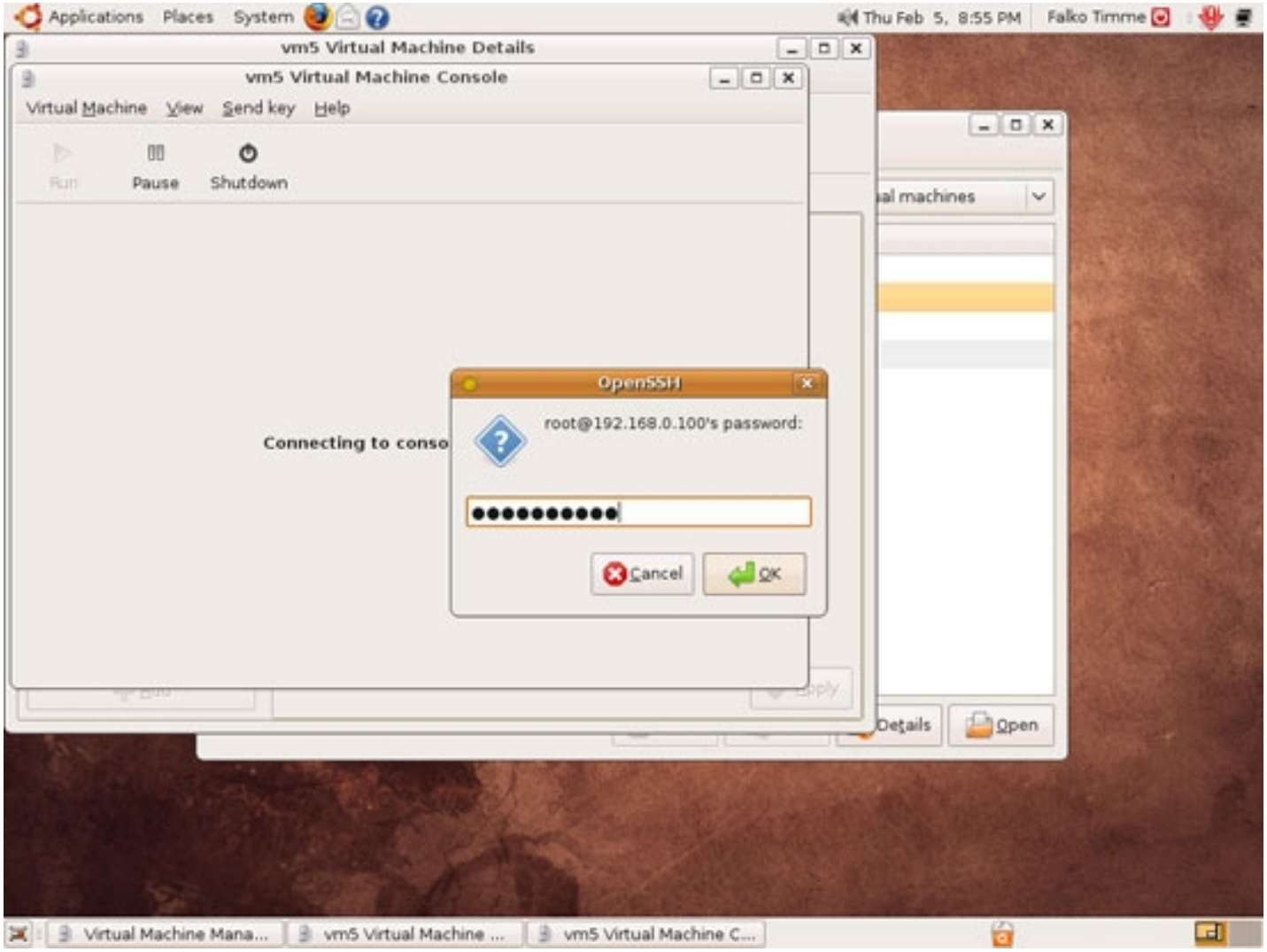

... you can learn more about the "hardware" used by the virtual machine (and tweak some parameters):



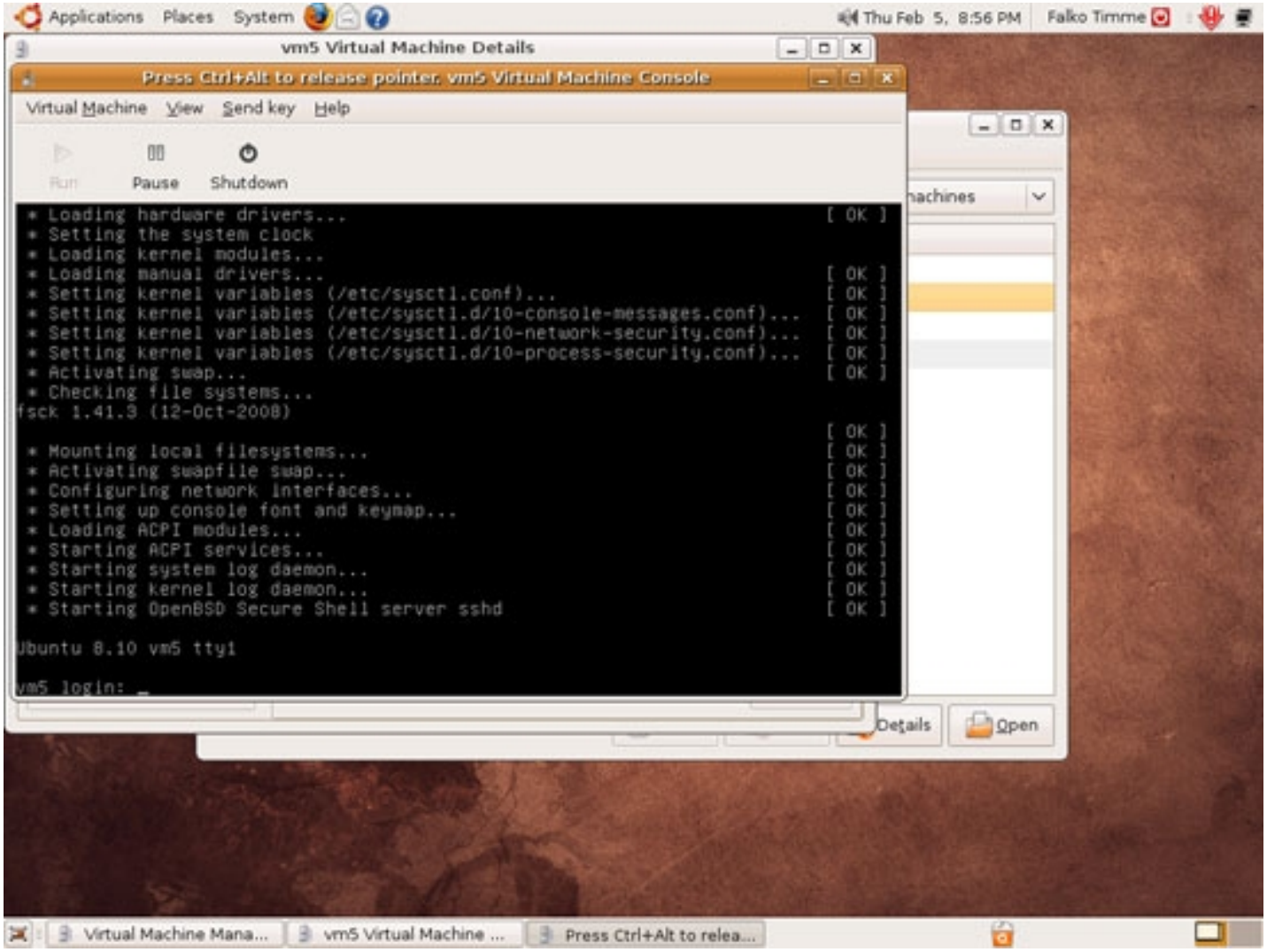

You can connect to the virtual machine by going to *`View > Graphical Console`*:

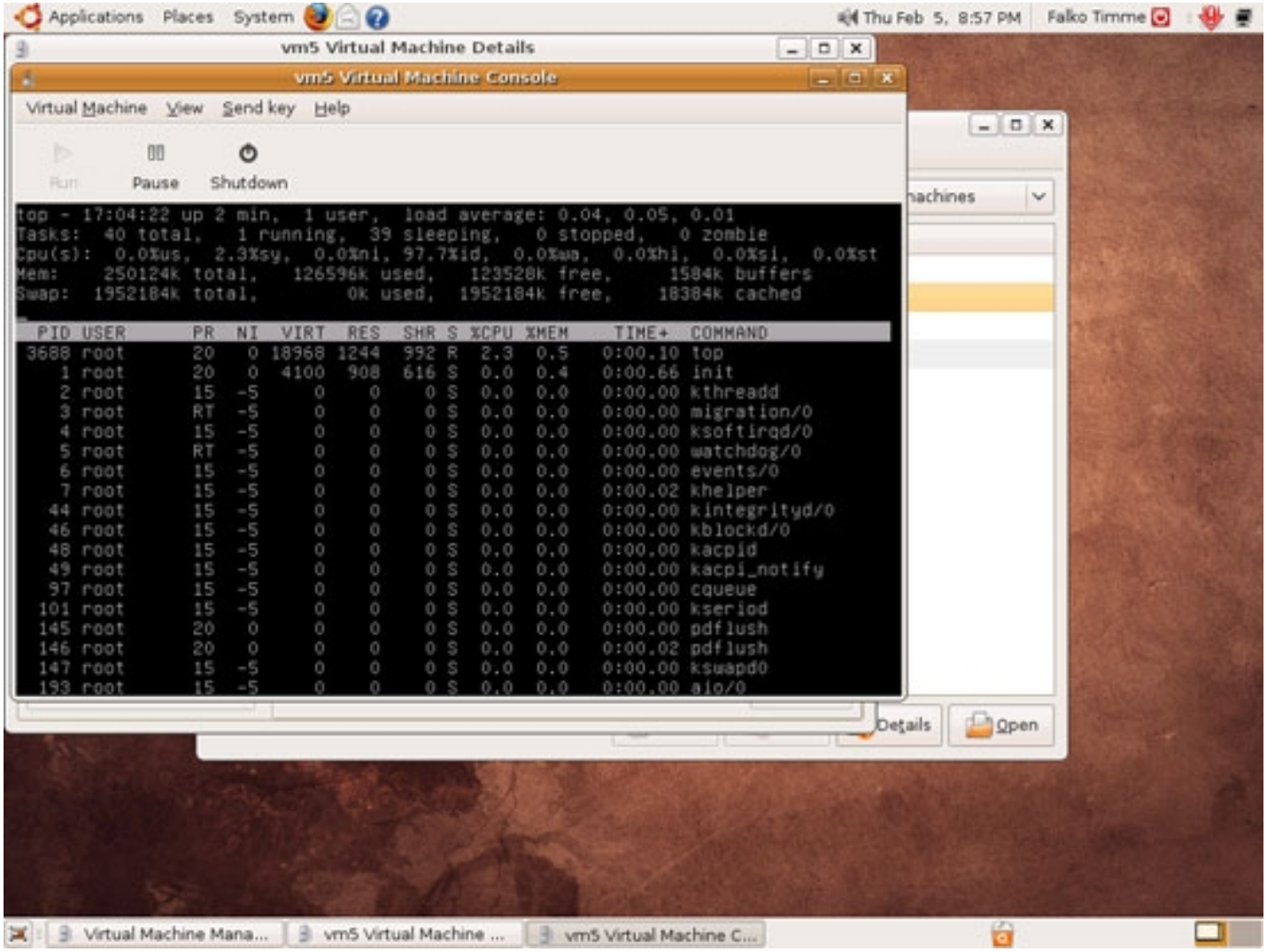



Type in the root password of the KVM host (this will pop up only if it is a remote host):




And voilÃ , here's your virtual machine (in this example, it's a server vm, but of course, you can also install a desktop as a virtual machine). Click on the console to work in the virtual machine, and press *CTRL+ALT* to release the mouse pointer from the console:

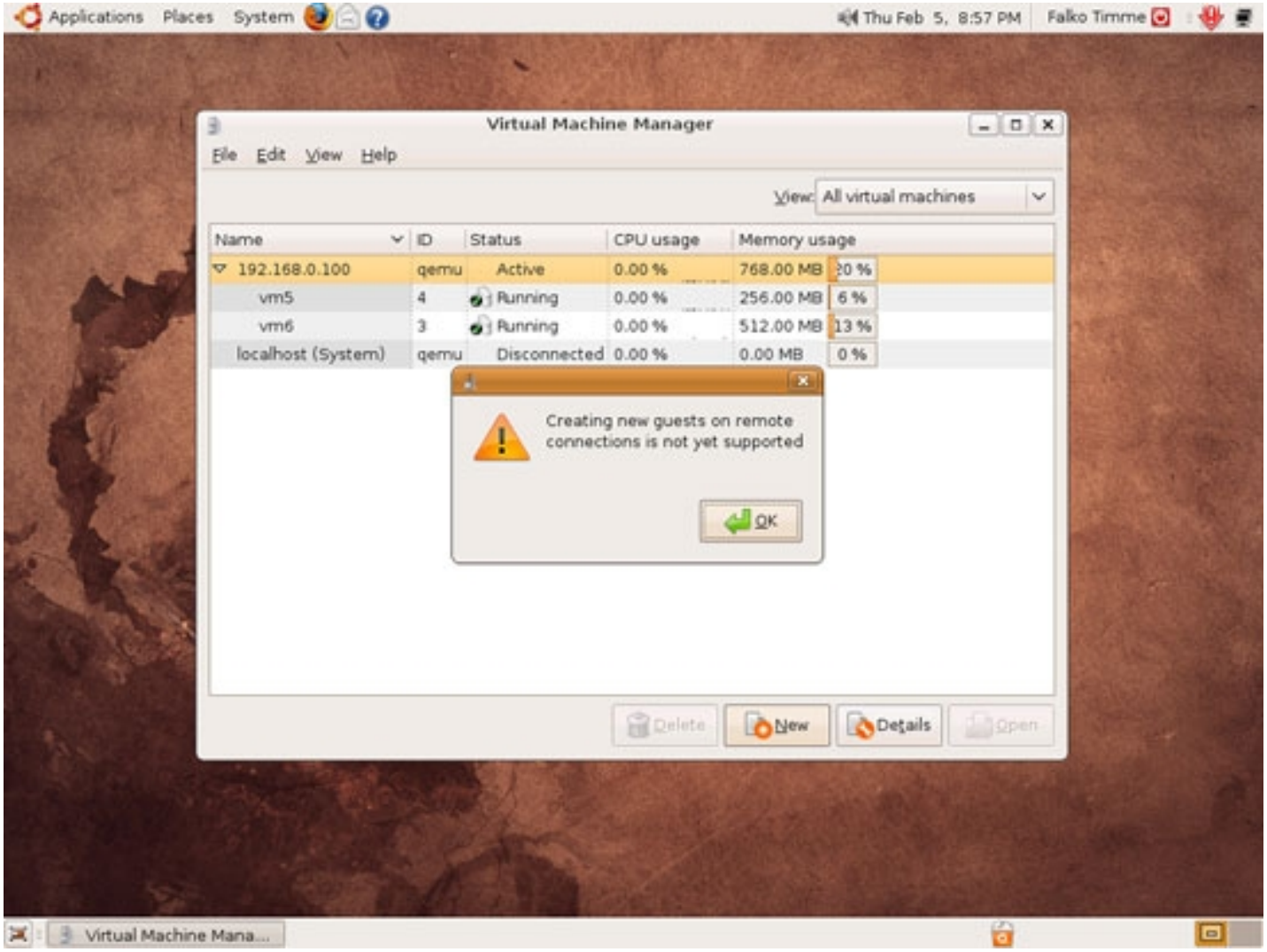





## 3.1 Creating Virtual Machines

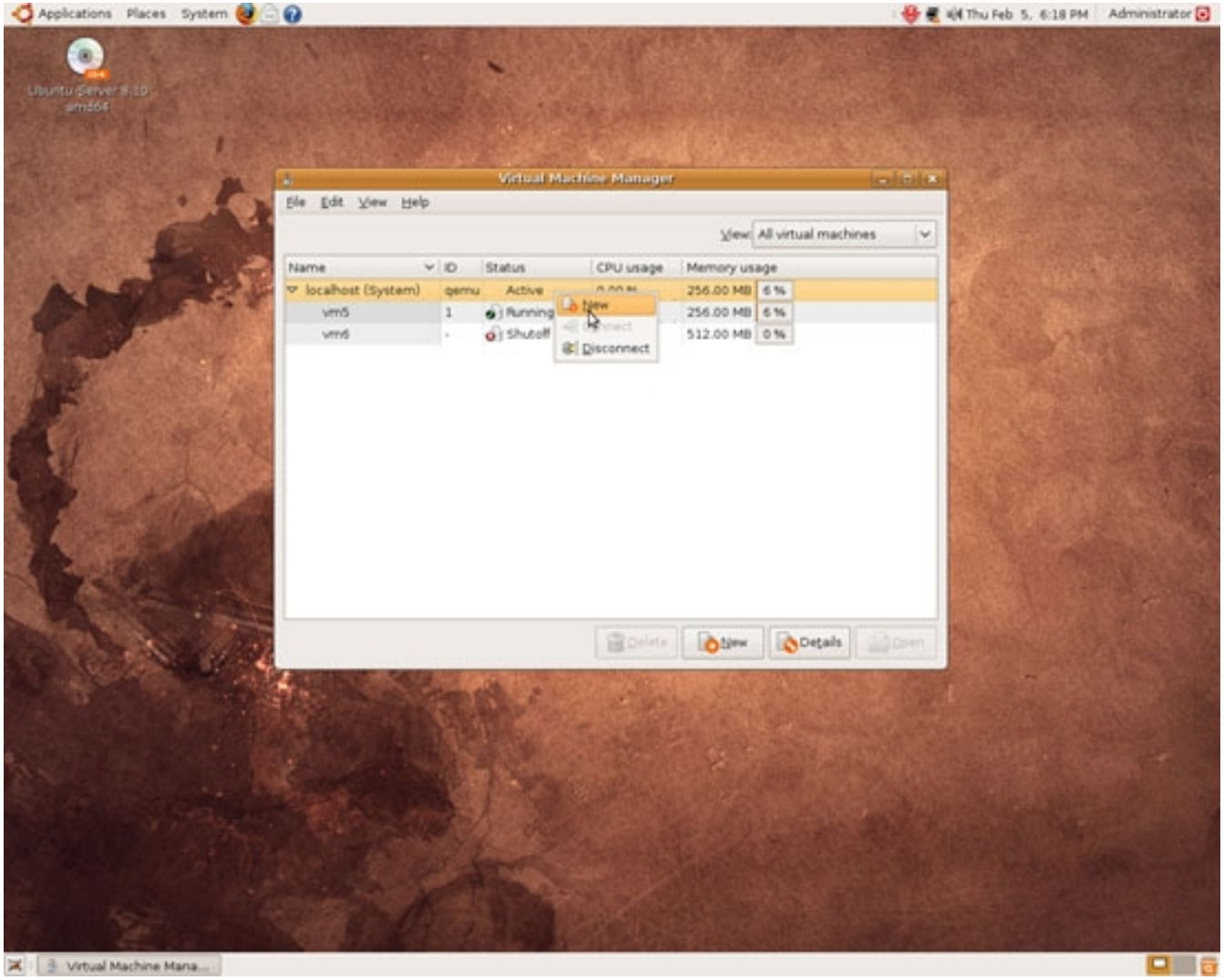
Creating guests on remote KVM hosts is not yet supported - you will see the following message when you click on the *New* button while the remote KVM

host is marked: *Creating new gusts on remote connections is not yet supported*.

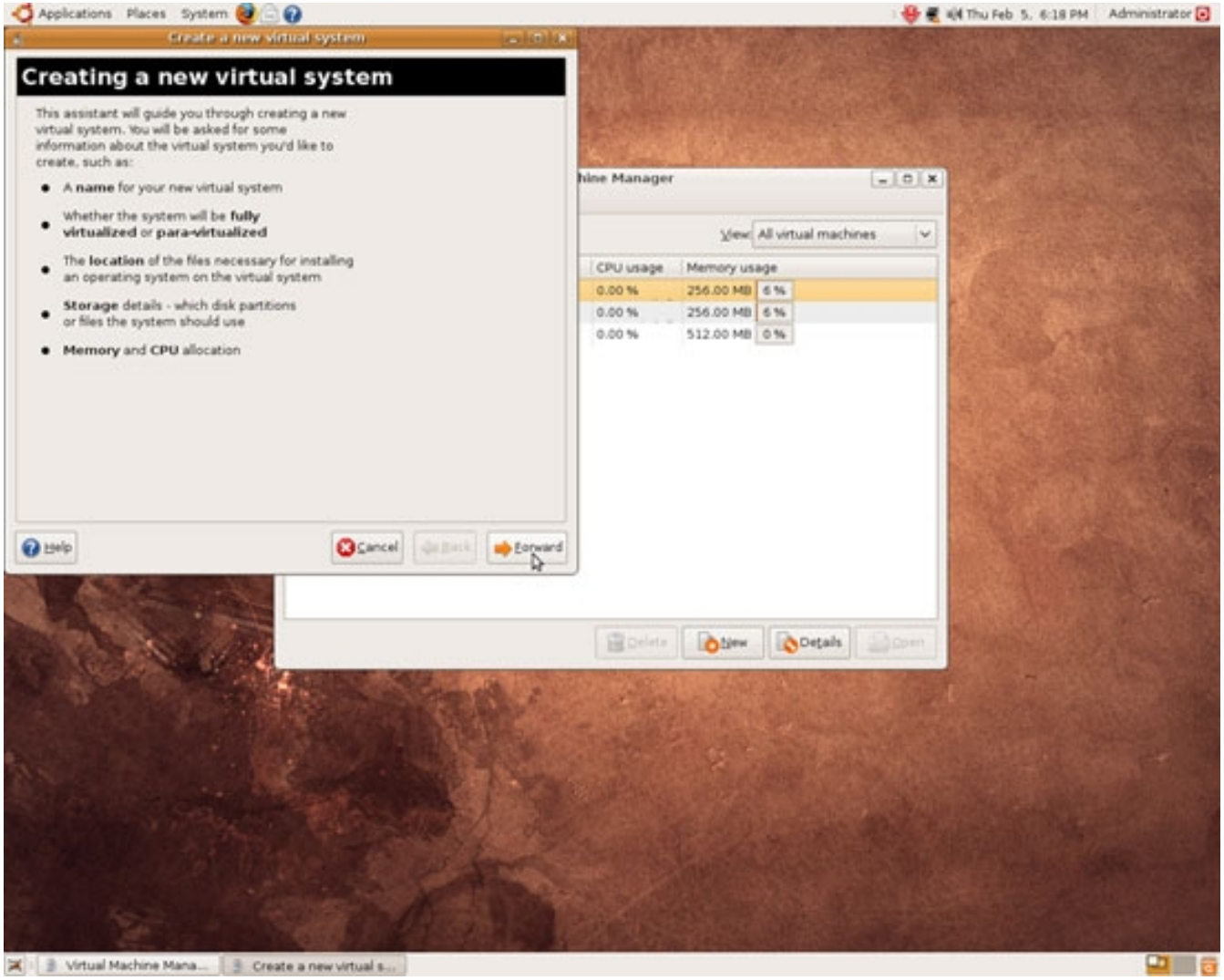



If have therefore installed virt-manager on my KVM host as well to show you how to create a new virtual machine (if you want to do this, please make sure you have followed chapter 2.1!).

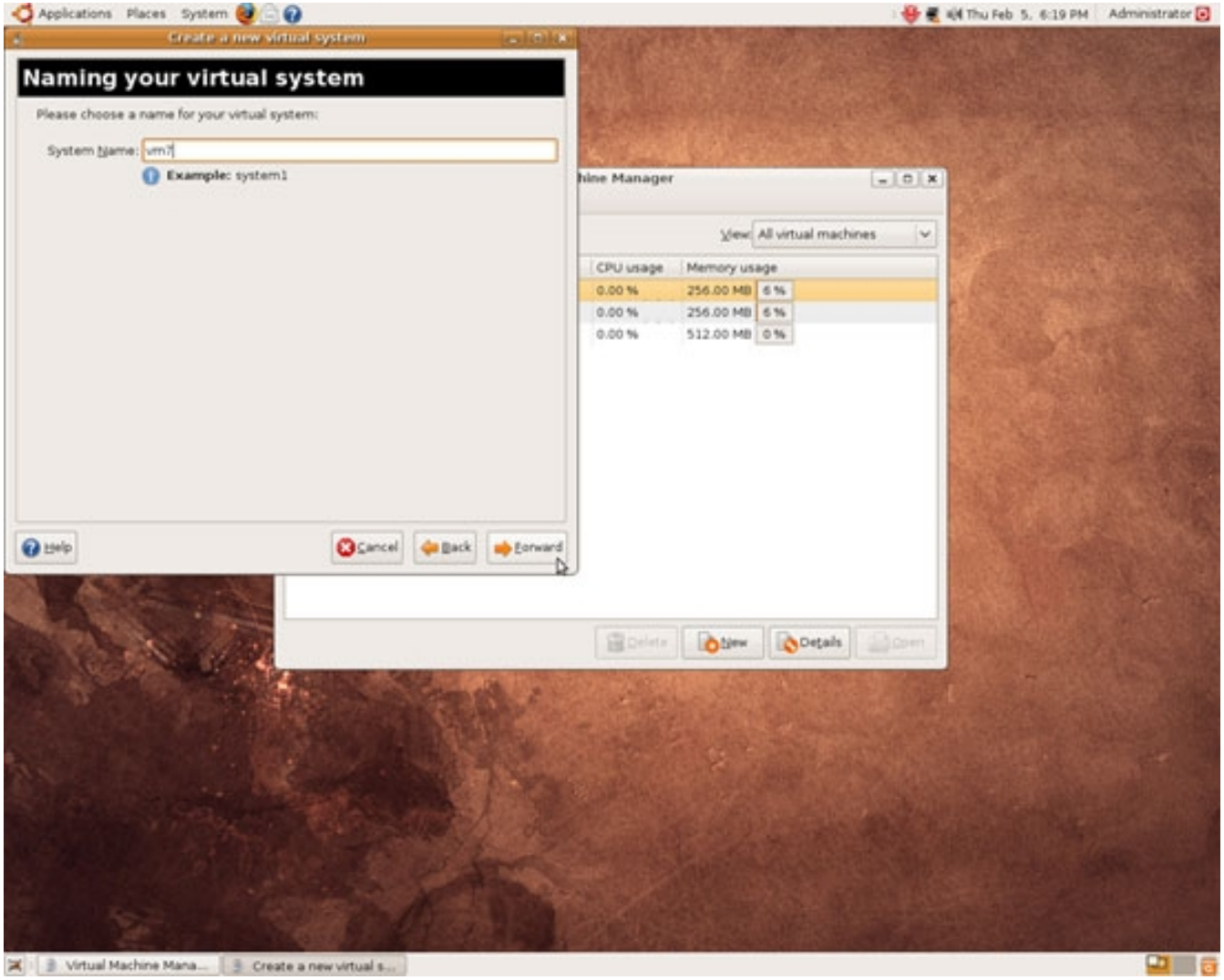
Right-click on *localhost* and select *New*:



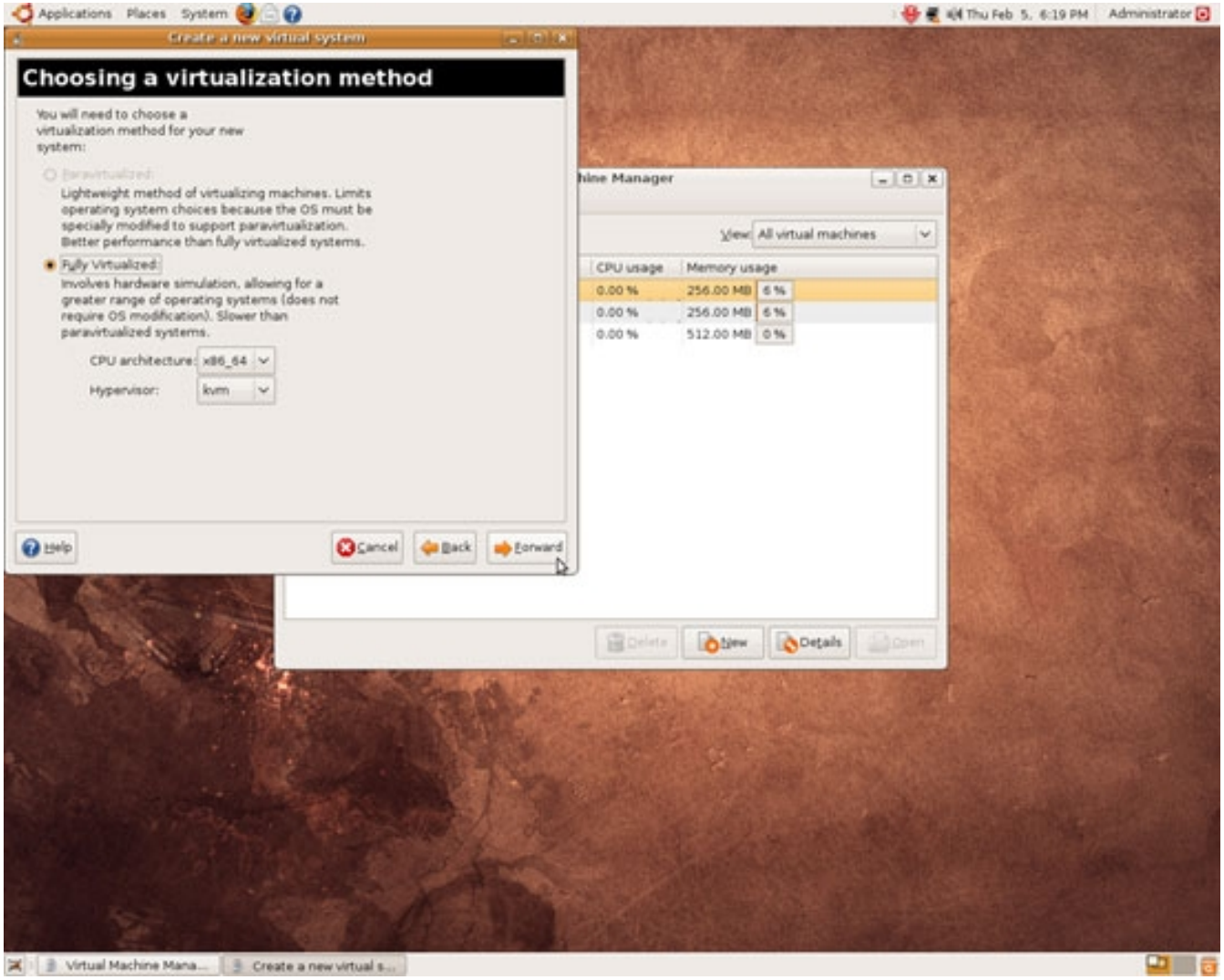

The virtual machine creation wizard comes up - click on *Forward*:



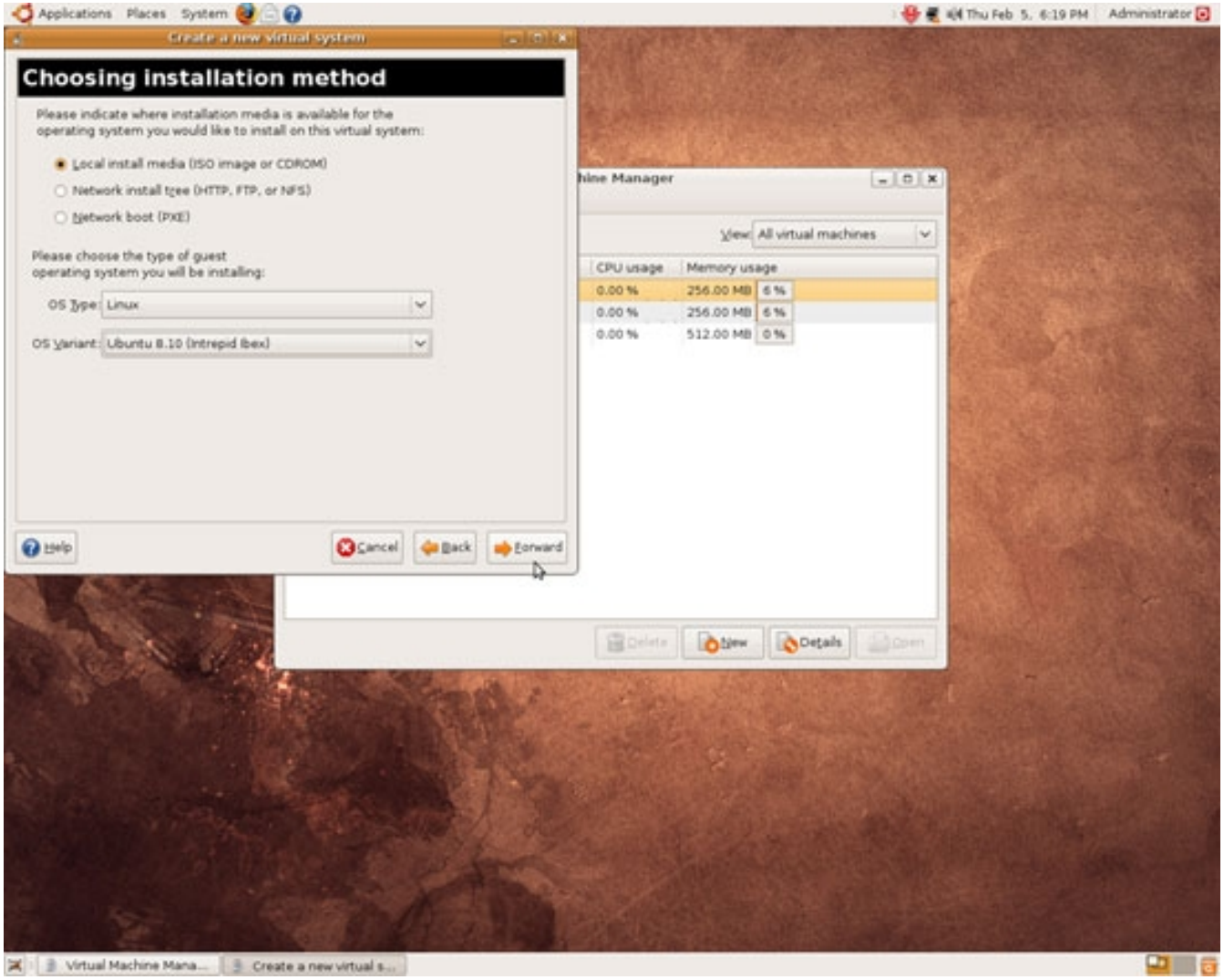

Type in a name for your new virtual machine (e.g. *vm7*):



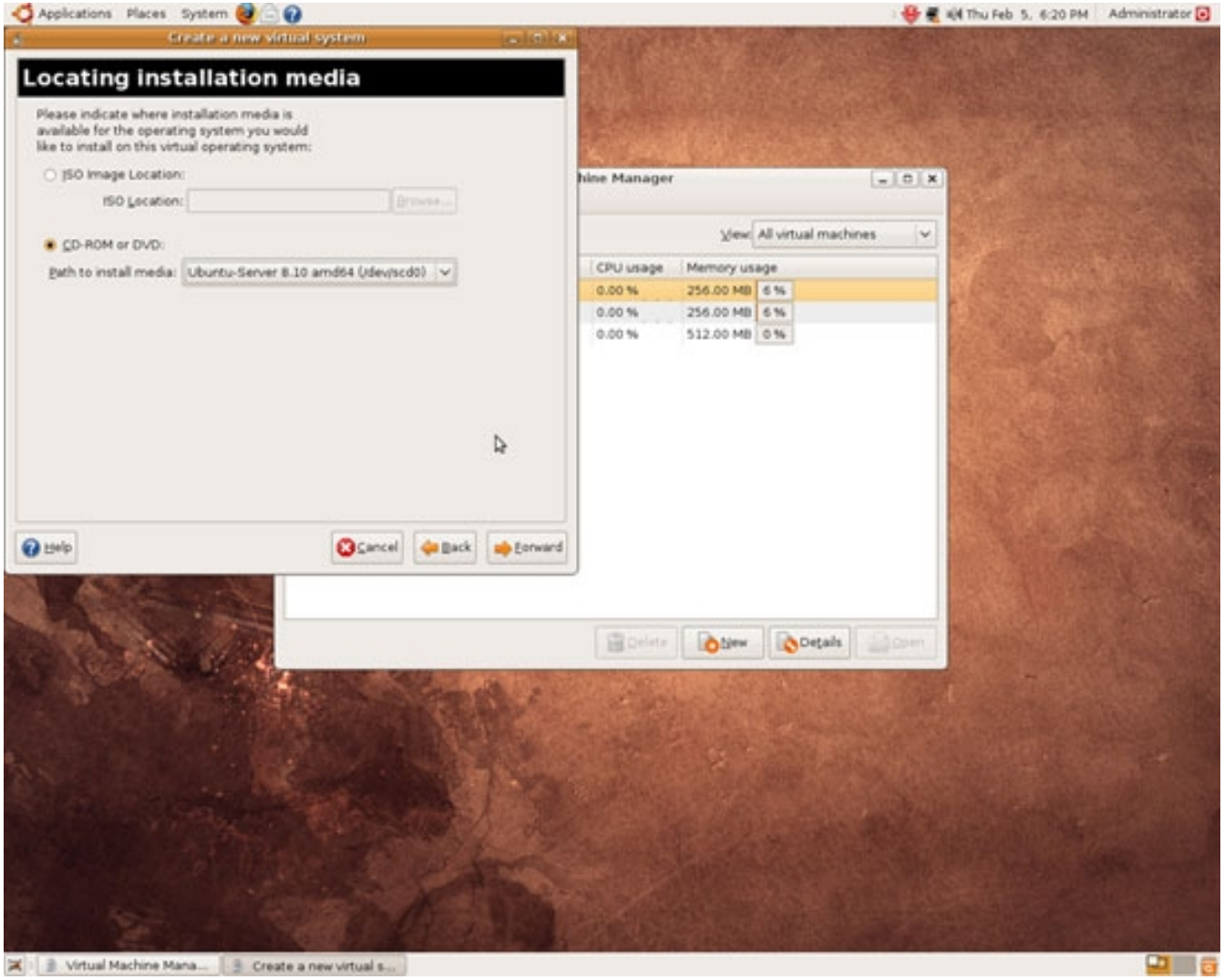

On the next screen, select *Fully Virtualized* (*Paravirtualized* should be greyed out anyway) as well as your CPU architecture. Pick *kvm* from the *Hypervisor* drop-down menu:



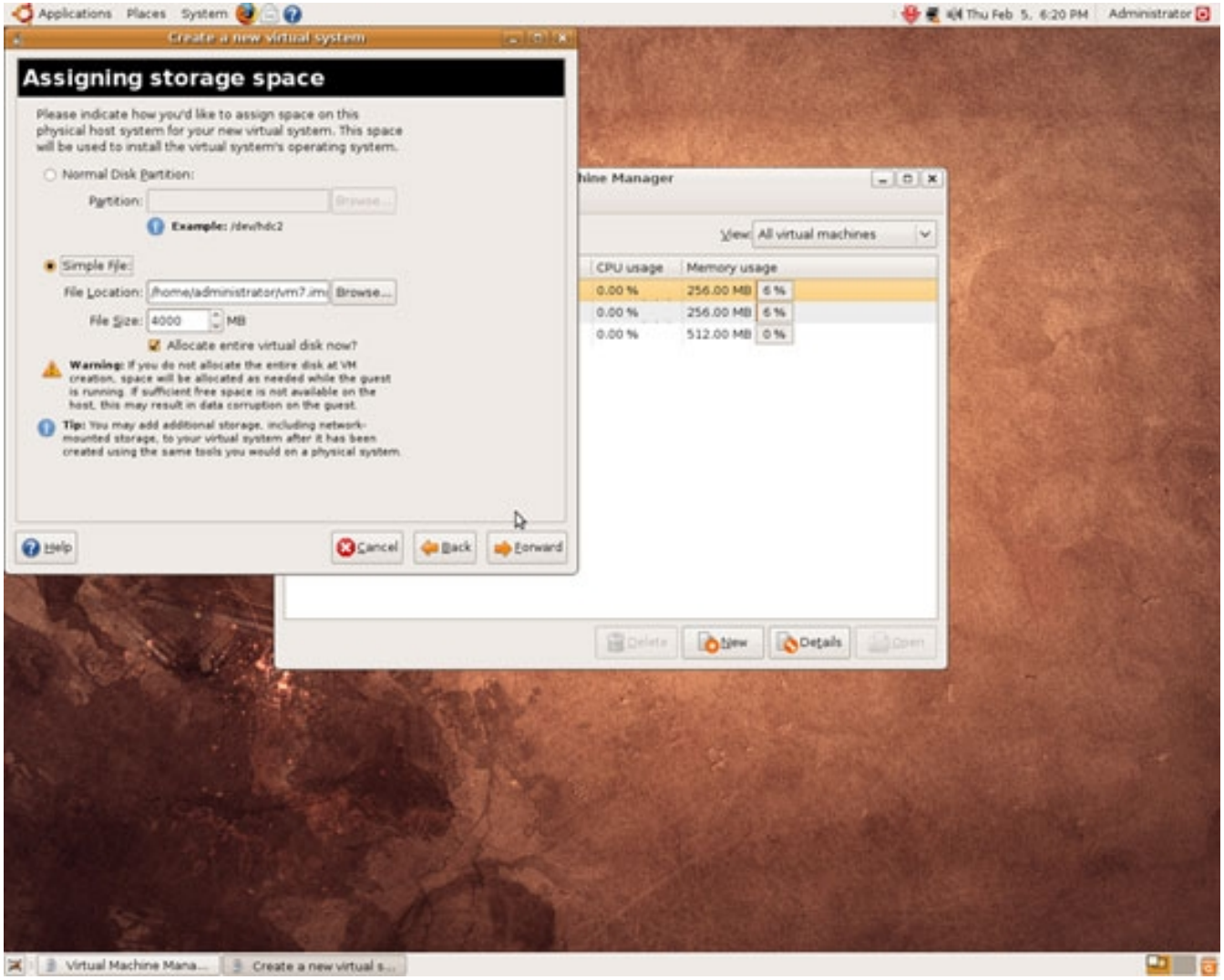

Select the installation method - normally you choose *`Local install media (ISO image or CDROM)`* here. Then select the operating system (e.g. *`Linux`*) and the variant (e.g. *`Ubuntu 8.10`*):




If you want to install from a CD or DVD, insert it, select CD-ROM or DVD and choose the CD from the drop-down menu. If you want to install from an ISO file, please specify the full path to the file:




Now we come to the storage space. You can either install the virtual machine in an image file (select *`Simple File`*)...

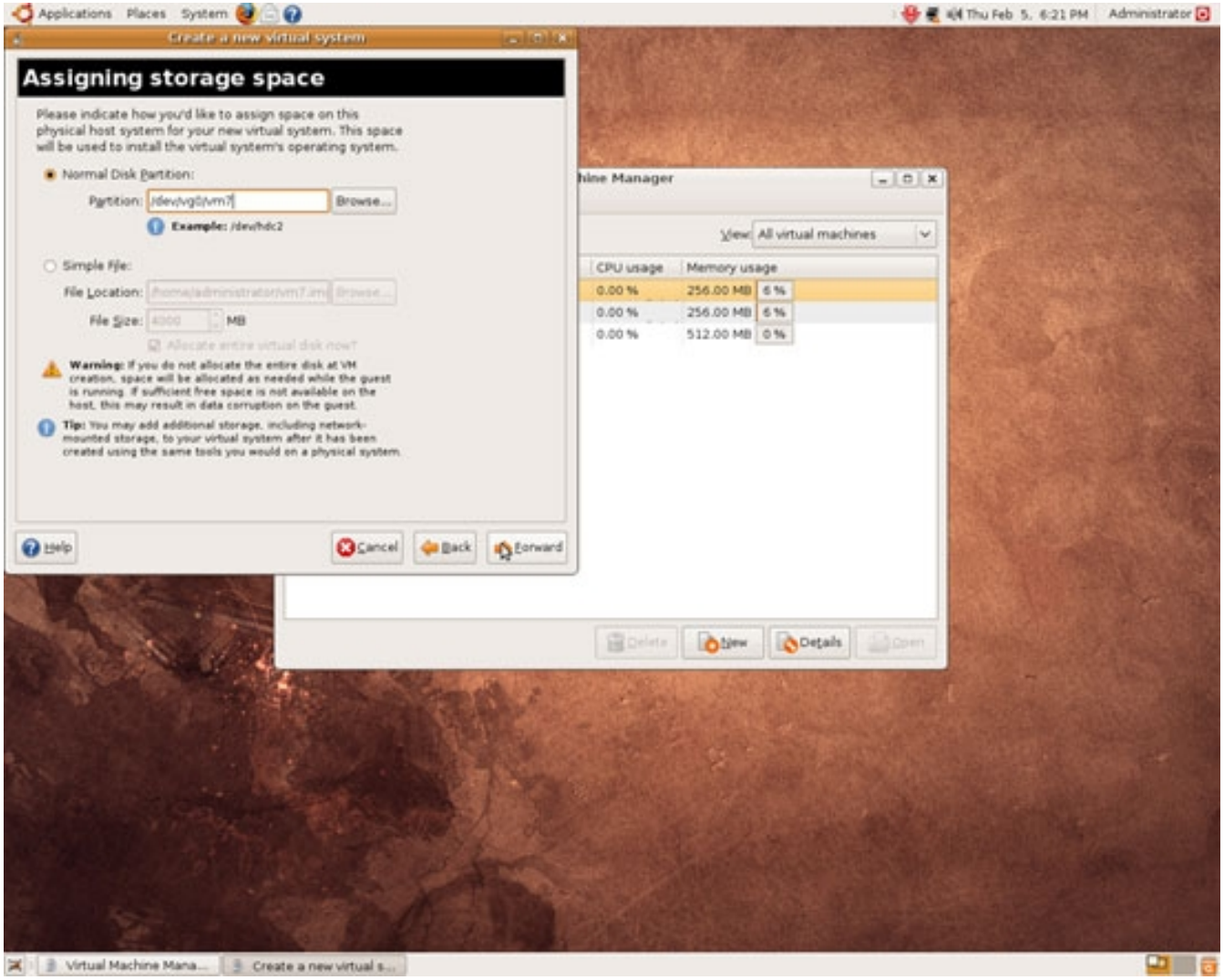



... or, if you have an unused partition or are using LVM and have some unallocated space in your volume group, you can specify a disk partition. Disk partitions are faster and not as IO-heavy as image files.

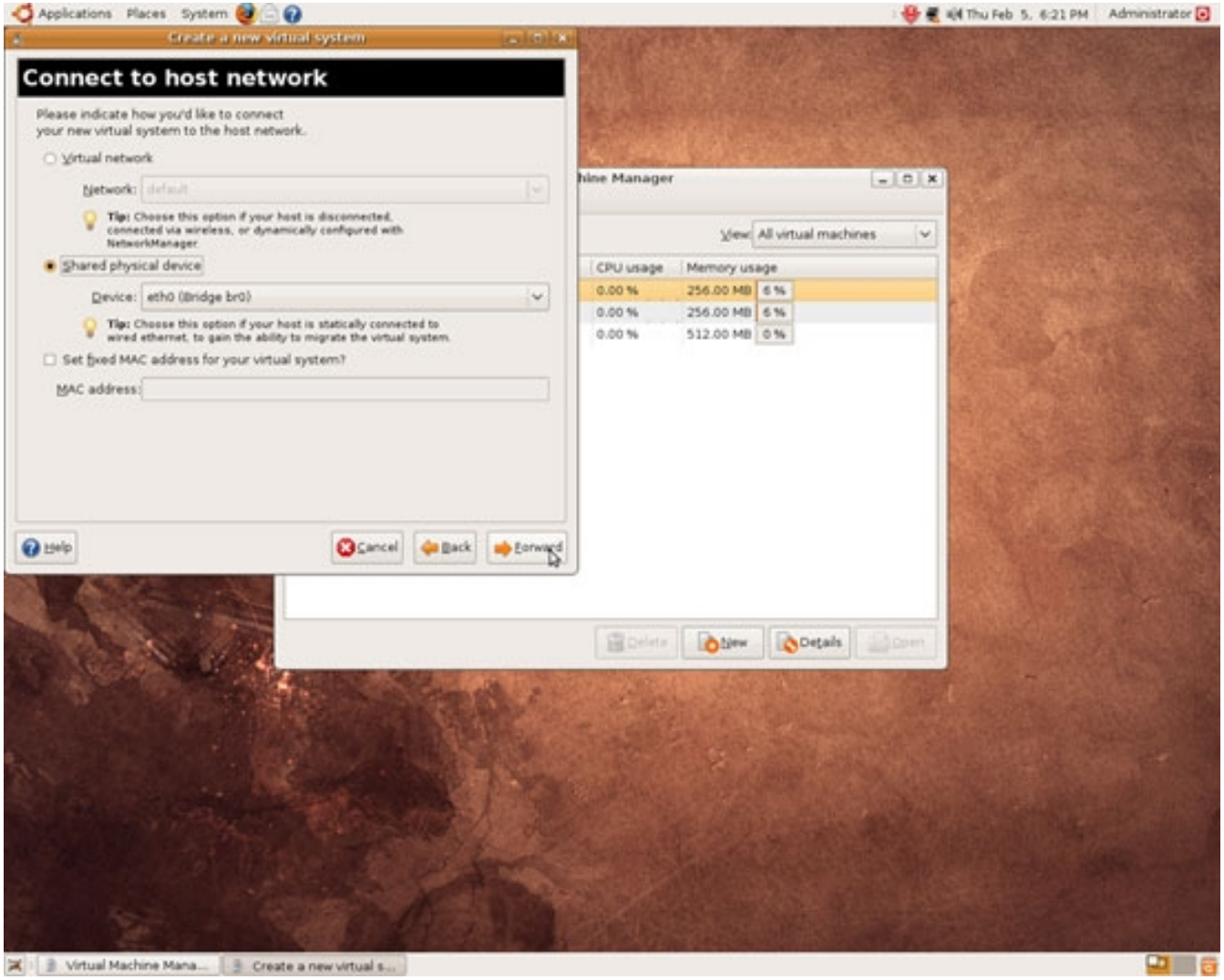
Let's assume you are using LVM and have some unallocated space in your volume group (e.g. *vg0*), then you can create a new logical volume (e.g. *vm7*) with a size of 4GB as follows:

```
lvcreate -L4G -n vm7 vg0
```

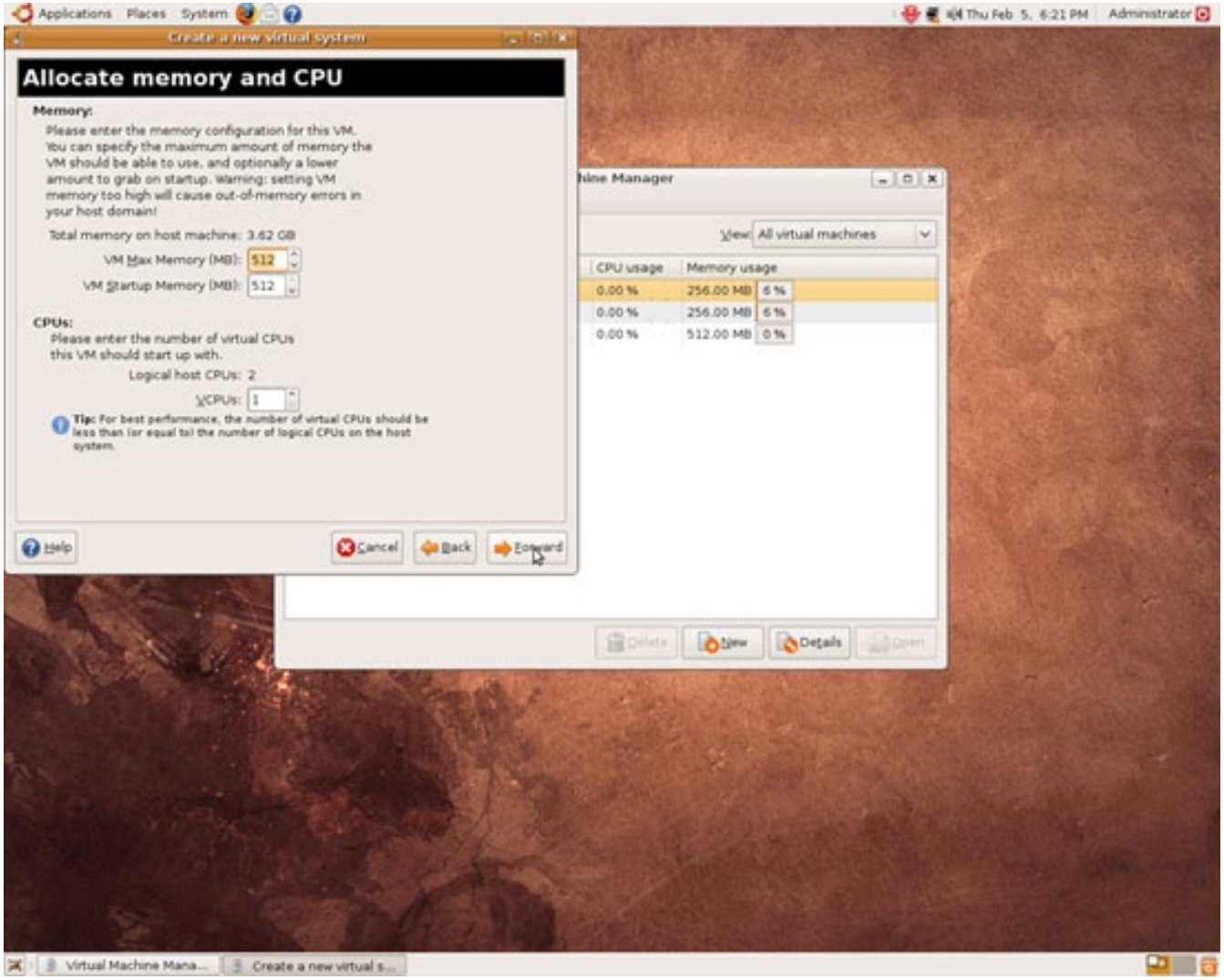
Afterwards, you fill in */dev/vg0/vm7* in the *Partition* field:



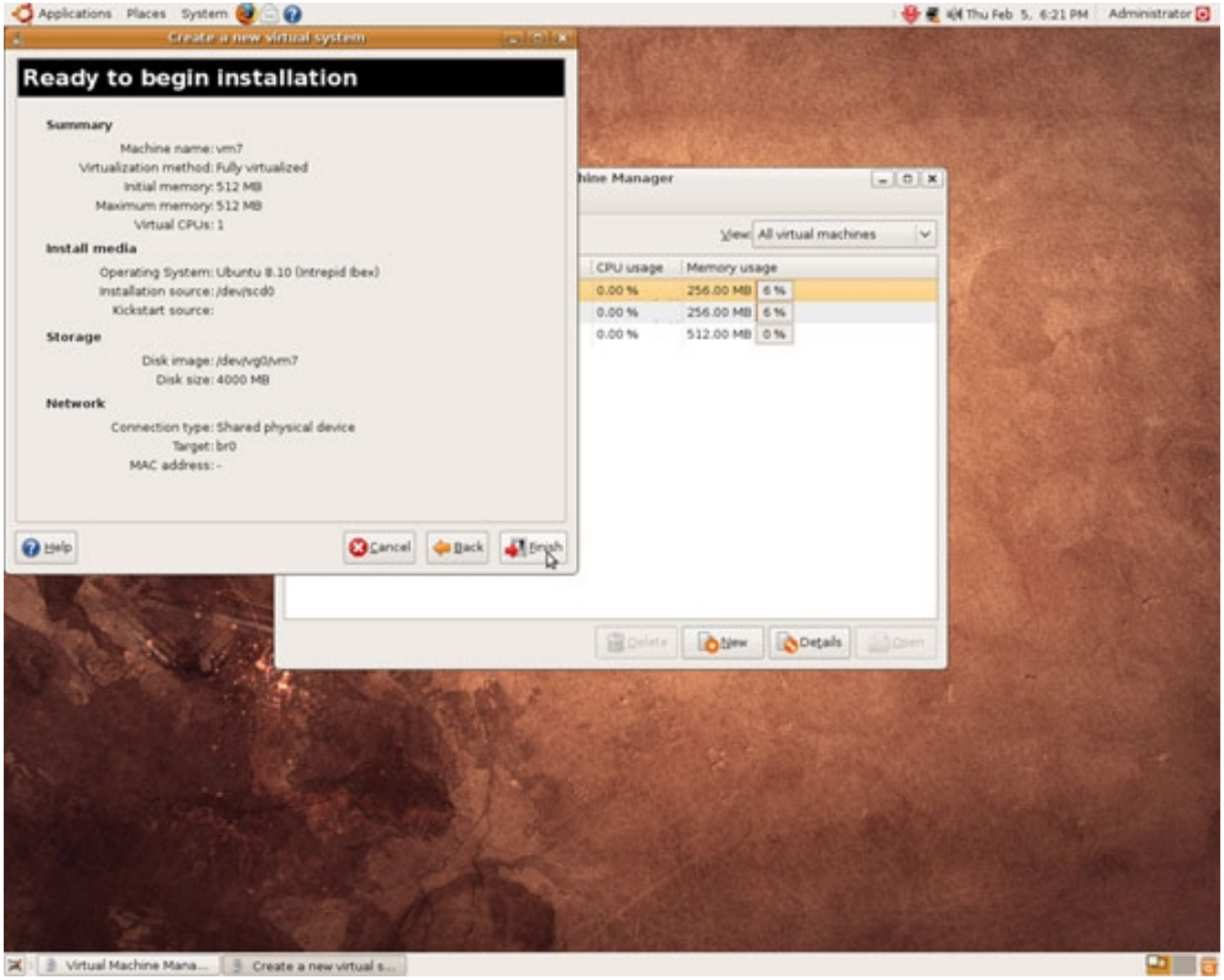

After you've configured your storage type, you must configure the network - select *Shared physical device* (this will create a bridge so that the virtual machine will have a working network connection from the start on):



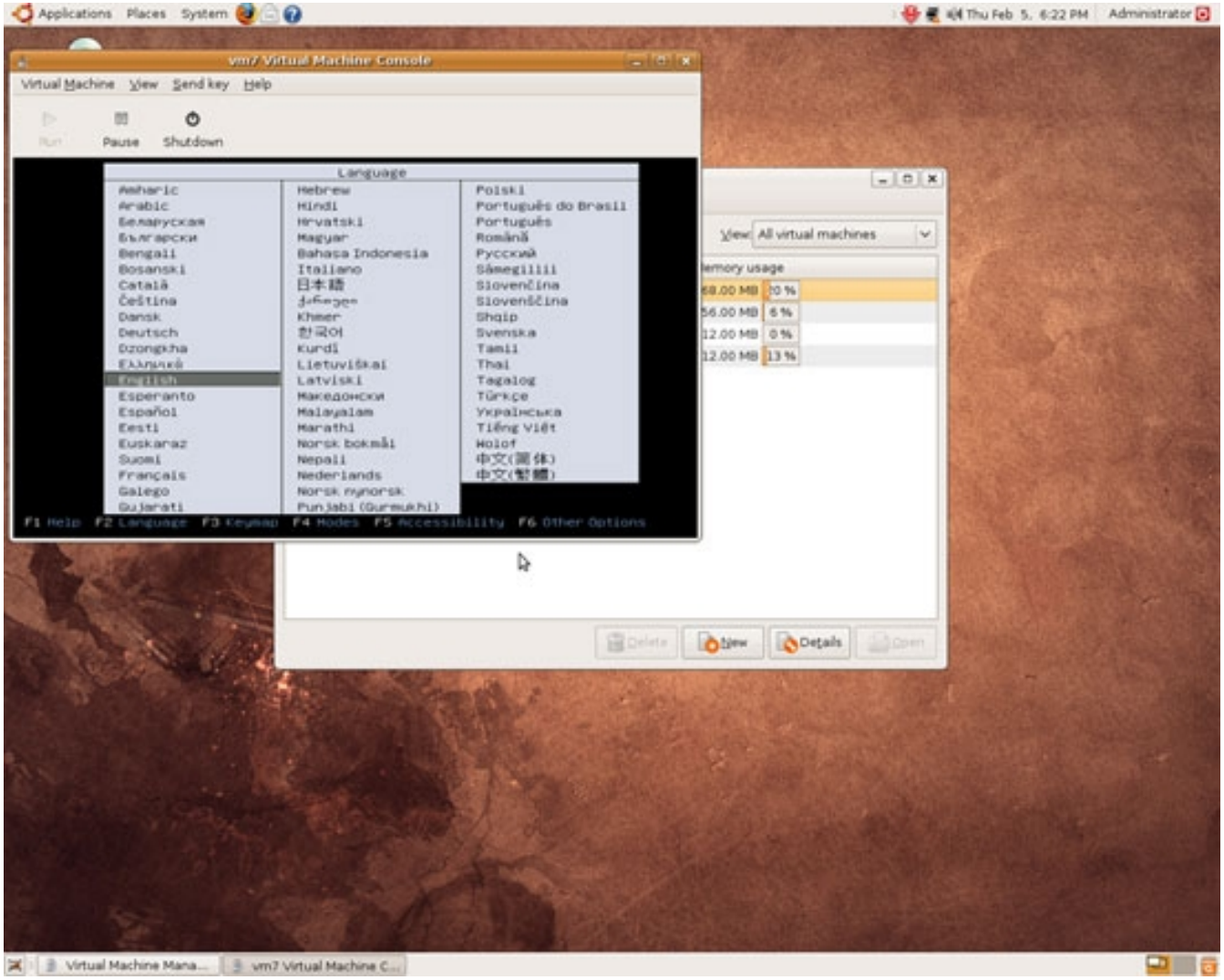

Next allocate some RAM and CPU(s) to the virtual machine:

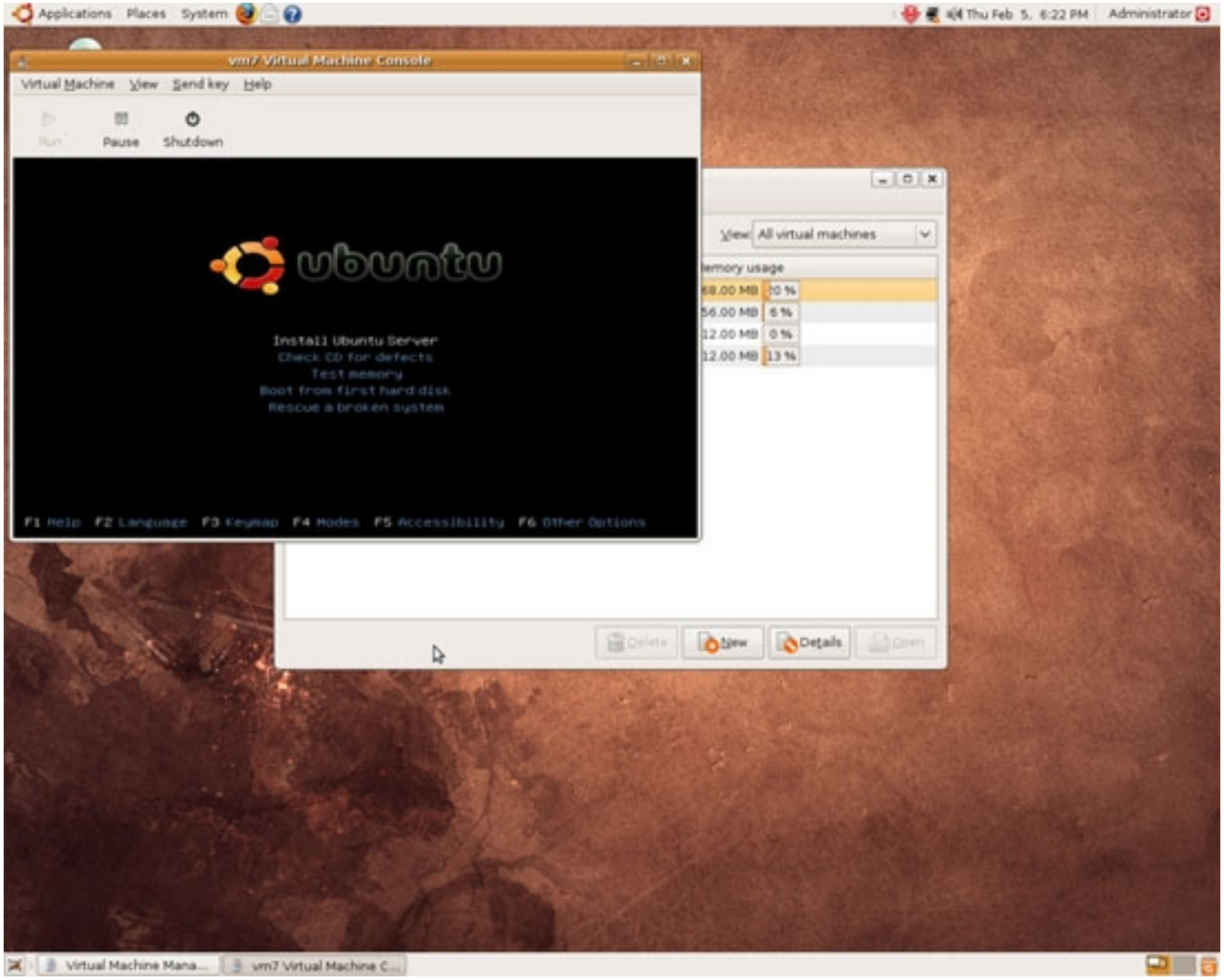



Finally, there's a summary screen which lists all selected options. If they look ok, click on *Finish* to start the installation of the virtual machine:




A graphical console comes up in which you can install your new virtual machine (click on it to work in the virtual machine, and press `CTRL+ALT` to release the mouse pointer from the console):






# 4 Links

- KVM (Ubuntu Community Documentation): **https://help.ubuntu.com/community/KVM**

- Ubuntu: **http://www.ubuntu.com/**