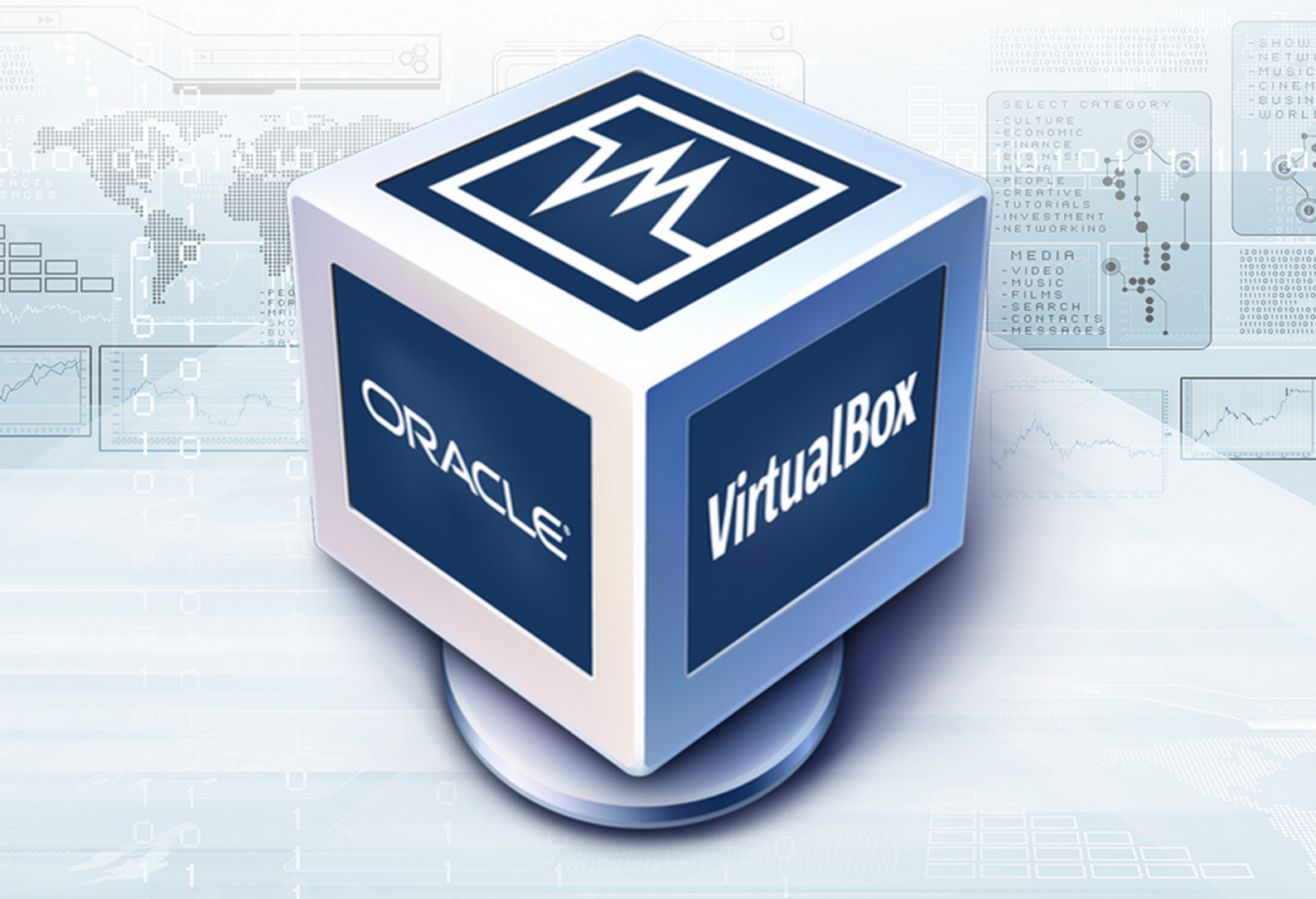
VIRTUALBOX ESSENTIALS

Your startup guide to virtualization



GABRIEL CANEPA

SYSTEM CODE GEEKS
SYSADMINS RESOURCE CENTER

VirtualBox Essentials

Contents

1 Virtualization with VirtualBox: Installation and Configuration 1

1.1 Introducing VirtualBox . . . 1
1.2 Installing VirtualBox . . . 1
1.3 Installing the guest additions and the extension pack . . . 6
1.4 Summary . . . 9

2 Cloning, exporting, importing, and removing virtual machines in VirtualBox 10

2.1 Cloning virtual machines . . . 11
2.2 Exporting virtual machines . . . 13
2.3 Importing virtual machines . . . 15
2.4 Removing virtual machines . . . 15
2.5 Inspecting logs . . . 16
2.6 Summary . . . 16

3 How to create, restore, and delete virtual machine snapshots in VirtualBox 17

3.1 Taking a snapshot . . . 17
3.2 Restoring snapshots . . . 19
3.3 Things to consider when restoring a snapshot . . . 21
3.4 Summary . . . 21

4 How to configure the settings of a Virtual Machine in VirtualBox 22

4.1 Configuring a virtual machine . . . 22
4.2 General . . . 23
4.3 System . . . 24
4.4 Storage . . . 25
4.5 Network . . . 25
4.6 Summary . . . 27

5 Managing VirtualBox Virtual Machines through Command Line (with VBoxManage) 28

5.1 Introducing VBoxManage 28

5.2 Listing virtual machines 28

5.3 Showing and modifying information and settings of virtual machines 30

5.4 Exporting and importing appliances 32

5.5 Starting and stopping virtual machines 33

5.6 Summary 33

Copyright (c) Exelixis Media P.C., 2016

All rights reserved. Without limiting the rights under
copyright reserved above, no part of this publication
may be reproduced, stored or introduced into a retrieval system, or
transmitted, in any form or by any means (electronic, mechanical,
photocopying, recording or otherwise), without the prior written
permission of the copyright owner.

Preface

Oracle VM VirtualBox is a free and open-source hypervisor for x86 computers from Oracle Corporation. VirtualBox may be installed on a number of host operating systems, including: Linux, OS X, Windows, Solaris, and OpenSolaris. There are also ports to FreeBSD and Genode.

It supports the creation and management of guest virtual machines running versions and derivations of Windows, Linux, BSD, OS/2, Solaris, Haiku, OSx86 and others, and limited virtualization of OS X guests on Apple hardware.

For some guest operating systems, a "Guest Additions" package of device drivers and system applications is available which typically improves performance, especially of graphics. (Source: https://en.wikipedia.org/wiki/VirtualBox)

In this ebook, we provide a compilation of VirtualBox tutorials that will help you get started with this virtualization platform. We cover a wide range of topics, from installing the software and performing a basic configuration, to cloning, exporting, importing, and removing virtual machines. With our straightforward tutorials, you will be able to get your own Virtual Machines up and running in minimum time.

About the Author

Gabriel Canepa is a Linux Foundation Certified System Administrator (LFCS-1500-0576-0100) and web developer from Villa Mercedes, San Luis, Argentina.

He works for a worldwide leading consumer product company and takes great pleasure in using FOSS tools to increase productivity in all areas of his daily work.

When he's not typing commands or writing code or articles, he enjoys telling bedtime stories with his wife to his two little daughters and playing with them, the great pleasure of his life.

Chapter 1

Virtualization with VirtualBox: Installation and Configuration

Although the concept of virtualization has been around since the 1960's, it wasn't until recent (10-15) years that it became a well-known term in the IT industry. In simple terms, virtualization allows to create several virtual resources (known as virtual machines or guests) on top on physical hardware (which is known as the host).

Through the use of virtualization software (also known as Virtual Machine Monitor, VMM, or hypervisor), the host allows each guest to utilize its computing resources (a portion of storage space, RAM, CPU time, and one or more network interfaces, to name a few examples). Thus, hardware maintenance costs and required physical space are reduced significantly, while making it possible for related applications (i.e. a web server and a database server) to run on separate "boxes" to improve security and stability.

1.1 Introducing VirtualBox

VirtualBox is a popular virtualization solution for both enterprise and personal use. In addition, it is distributed at no cost under the terms of version 2 of the GNU General Public License (GPL v2), and is available for 32 and 64-bit platforms. VirtualBox currently can run on Linux, Windows, Macintosh and Solaris hosts and supports a wide variety of operating systems as guests. In other words, by installing and using VirtualBox on your computer you can run several virtual machines on top of it - each one with its own operating system, while sharing the physical resources provided by the underlying hardware.

To enhance one's experience with VirtualBox, two types of additional software are provided:

- *Guest additions* are drivers and applications that optimize the communication and integration between host and guest and improve the usability of the guest operating system (shared clipboard and folders between host and guests, mouse pointer integration, support for multiple screen resolutions, to name a few examples). Without the guest additions, these features are not present.
- *Extension packs* provide additional functionality to support USB 2.0 and 3.0 integration between the host and guests (allowing a guest to use an USB drive or device connected to the host, such as a webcam), virtual remote desktop (meaning a virtual machine can run on one host but be displayed and controlled from a separate computer), and encryption for disk images.

We will now explain how to install VirtualBox along with the guest additions and the extension pack.

1.2 Installing VirtualBox

Both the guest additions and the extension packs are distributed as one software resource each, and can be easily installed along with VirtualBox. However, let us keep in mind that as the word suggests, the guest additions should be installed on a per guest basis, as opposed to the extension pack which is supposed to be integrated with the main VirtualBox installation.

To download the latest VirtualBox release for your host, go to https://www.virtualbox.org/wiki/Downloads and choose your operating system version and architecture.

- If you are running Microsoft Windows Vista SP1 or newer on the host, the installation file will consist of a .exe file. Double click on that file to install VirtualBox with the default settings by pressing the Next button a couple of times.
- If you are running Linux on your host, the installation command will vary depending on the distribution family. For Fedora-based distributions you will need to do `rpm -Uvh file.rpm`, whereas for Debian and derivatives you will use `dpkg -i file.deb` (with `sudo` if you're on Ubuntu or a similar distribution). Replace **file** with the name of the actual installation file in your case.

Regardless of your operating system, you will then be able to launch the administration interface by double-clicking an icon on your desktop or by browsing the list of installed software on your system. It should look as shown in Fig. 1.1 (with the exception that the virtual machine list at the left will be empty in your case):

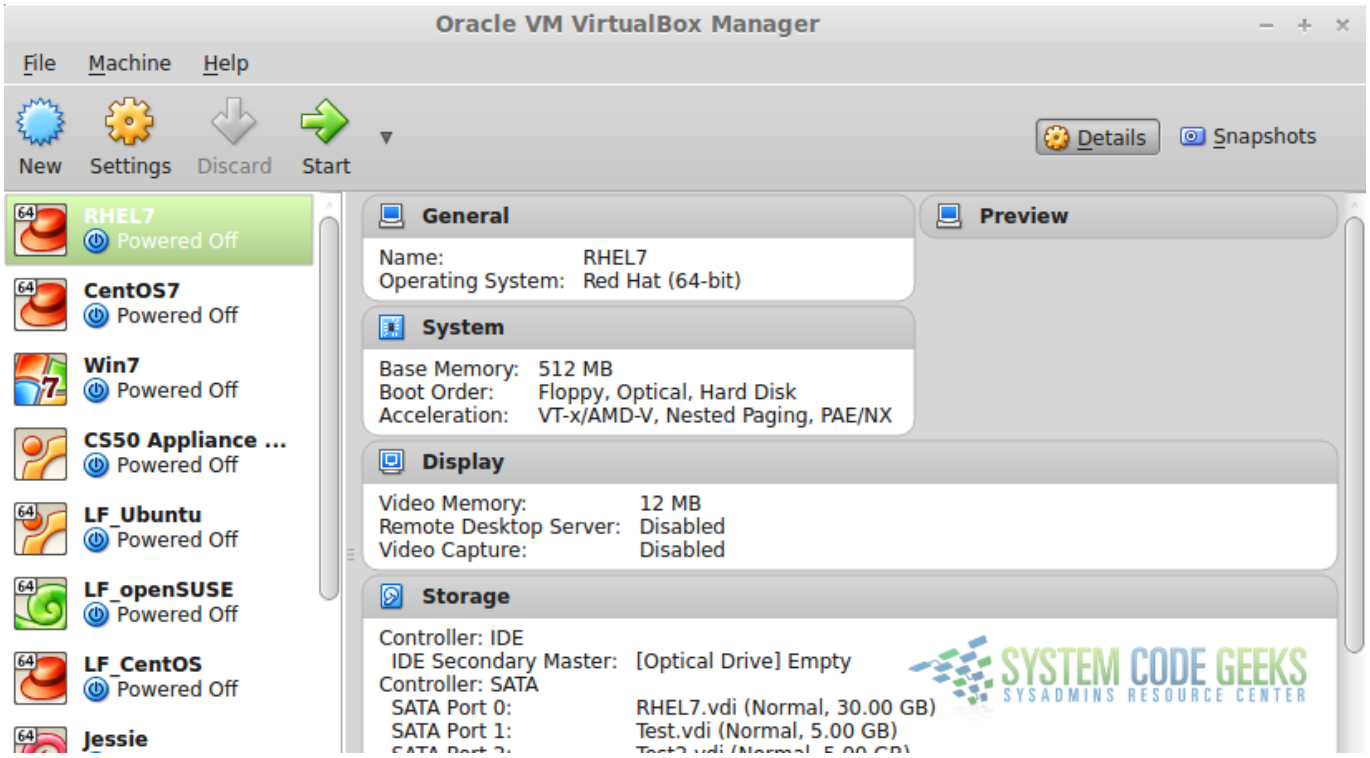


Figure 1.1: VirtualBox administration screen

We will now walk you through the initial setup of an Ubuntu 14.04 desktop as your first virtual machine. Note that this will require that you download an .iso file from https://www.ubuntu.com/desktop first.

1) Open VirtualBox and click **New**, as shown in Fig. 1.2:

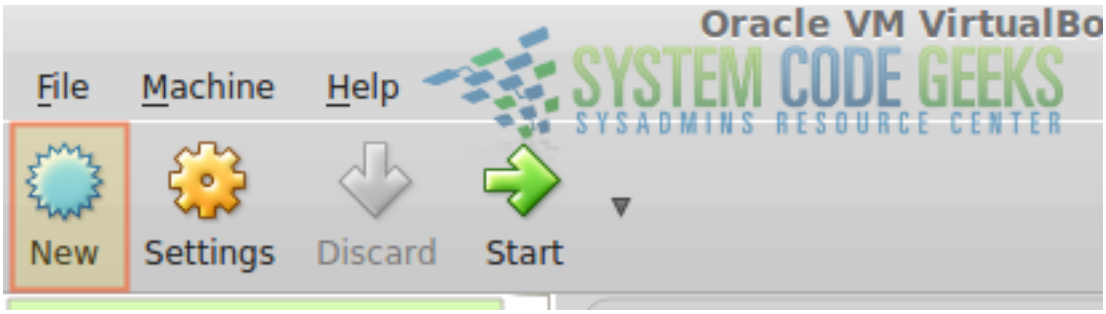


Figure 1.2: Starting the process of creating a new virtual machine

2) Choose a descriptive name for your virtual machine and make sure the right operating system and version have been detected. If not, pick the right settings from the **Type** and **Version** drop-down boxes, then click **Next** as indicated in Fig. 1.3:

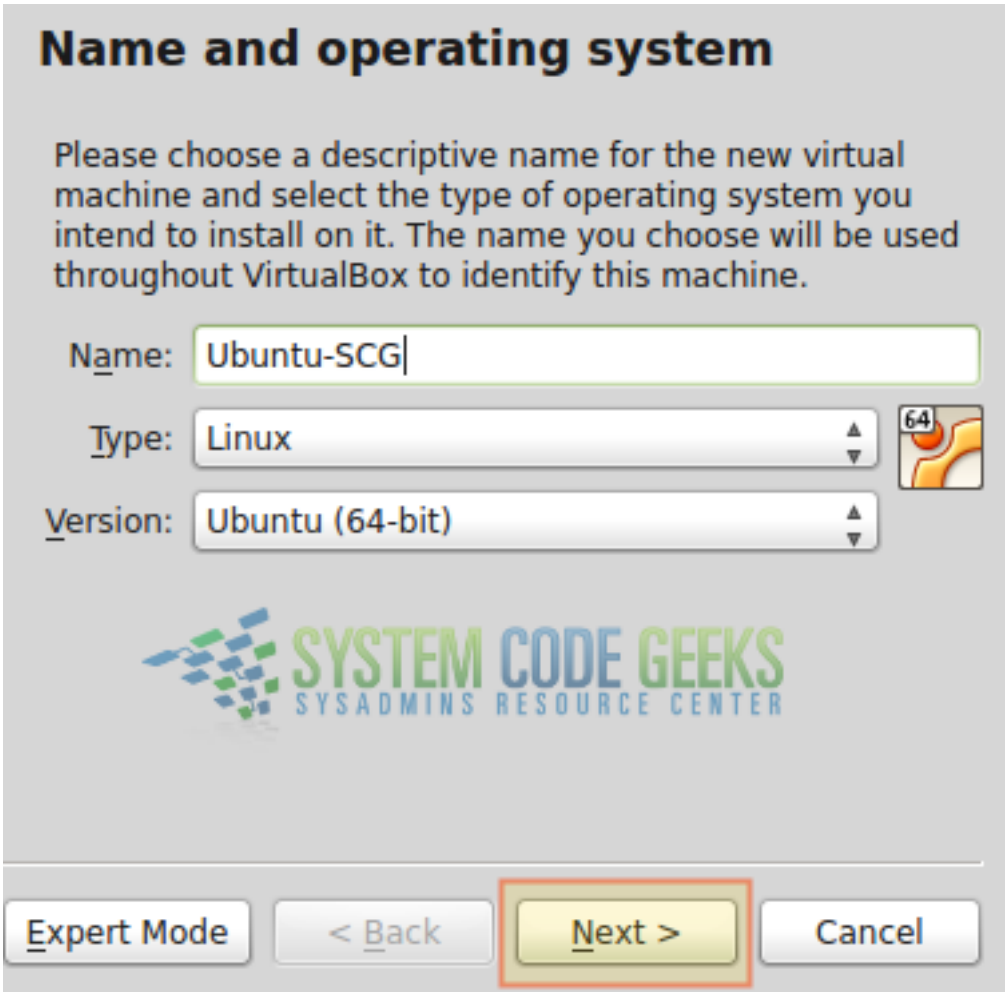


Figure 1.3: Choosing the type of our virtual machine

3) As you can see in Fig. 1.4, it's now time to allocate a safe amount of memory for your virtual machine (feel free to allocate more than the suggested amount if you can afford to). Note that you will need to have that same amount of RAM available as it will be taken by the virtual machine when it boots; otherwise it will refuse to start.

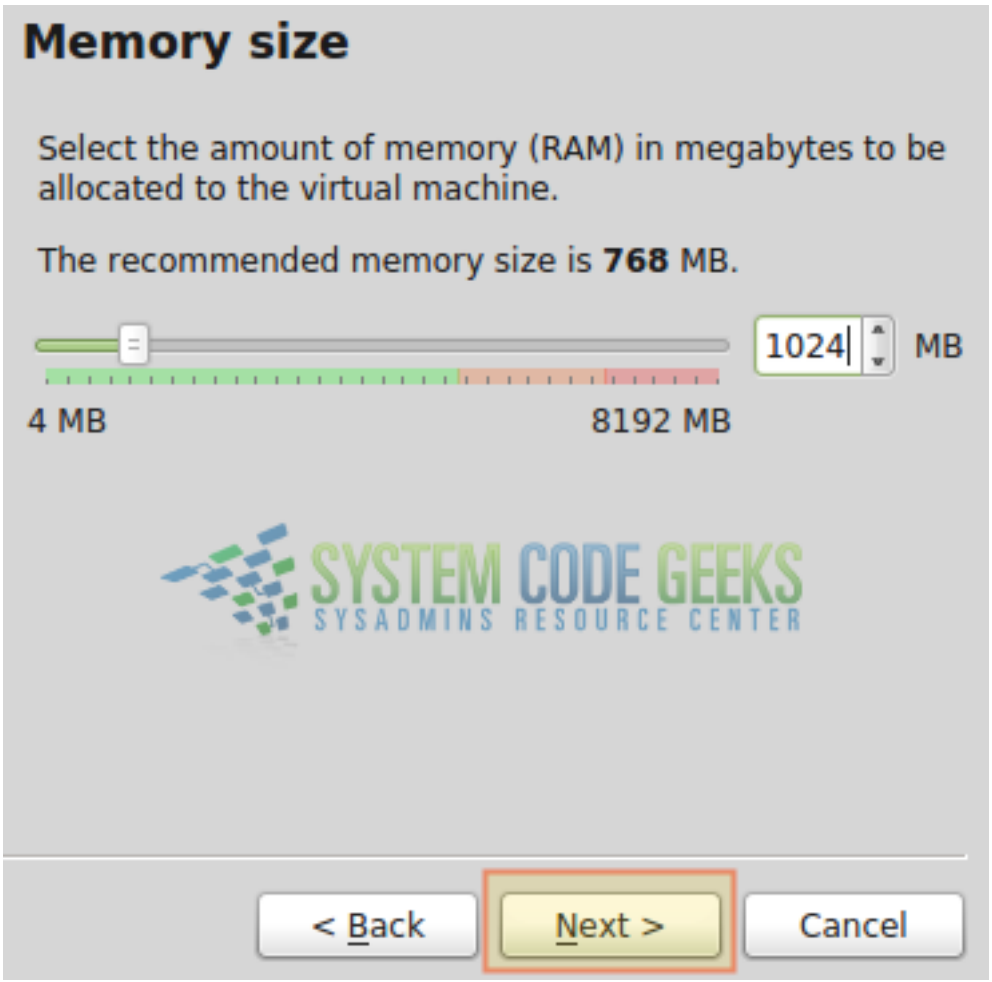


Figure 1.4: Allocating RAM for the virtual machine

If you notice that the virtual machine could use some more memory after you start it for the first time -and you can afford to add it-, you will be able to do so later on using the Settings menu. To modify the settings of a specific virtual machine, it needs to be turned off. Although we will explain this in more detail in the next tutorial, it is important to know that this is another advantage of virtualization: you can provision your virtual machines when you first create them but also can modify its resources without much hassle later. 4) The creation of the hard drive follows the same pattern as the allocation of RAM memory. Choose a space that will satisfy your needs (at least initially, because as we will see in a later tutorial, you can modify this value as well). The default choice of a Dynamically Allocated Virtual Disk Image of 8 GB in the next two steps (see Fig. 1.5) is safe and enough for our purpose:

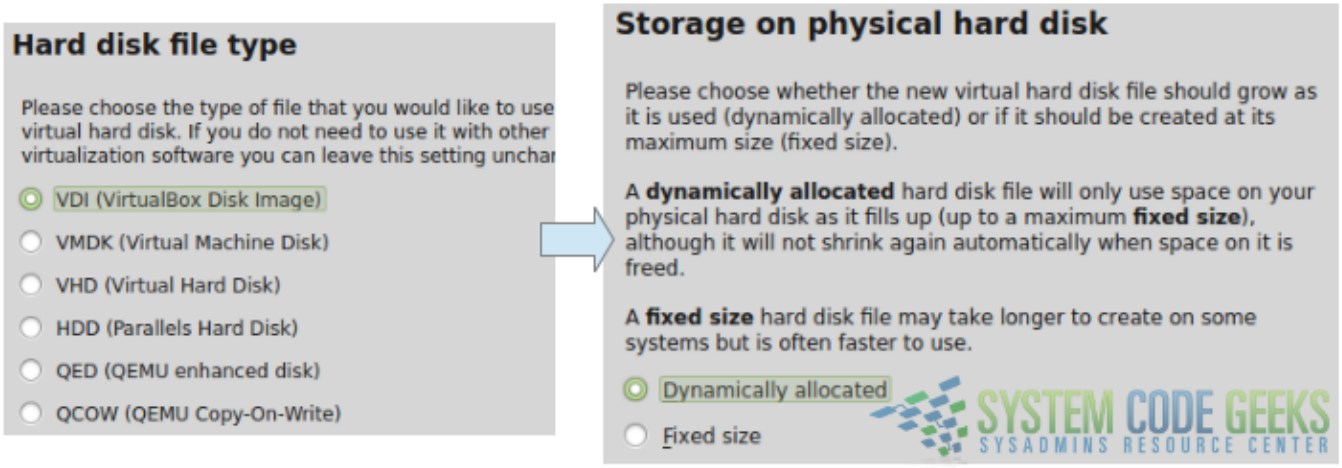


Figure 1.5: Creating a virtual hard drive

Once you have confirmed the settings mentioned above (perhaps choosing a different disk size, which is OK if you can afford it), click the Create button to finish the configuration part of the process.

It is important to note that the virtual hard disk will NOT initially take up the amount of space we indicated previously. Instead, that value is the limit of storage space that is enforced for this virtual machine. Once that limit is reached you will not be able to save other files.

5) The virtual machine now shows at the top of the list in VirtualBox. To load the .iso file for installation, select it and click Start as shown in Fig. 1.6. You will be prompted to select the installation media. As you can see, this can either be an .iso file as in our case, or an optical drive such as a DVD drive where a disk containing the operating system installation files has been placed.

Figure 1.6: Selecting installation media

From this point forward, the creation of the virtual machine follows the exact same procedure as if you were performing the installation on bare metal. Once the installation completes we can proceed with the guest additions and the extension pack.

1.3 Installing the guest additions and the extension pack

To install the guest additions on the newly created virtual machine, you will need to start it and login first. Next, go to **Devices** → **Insert Guest Additions CD image** (see Fig. 1.7):

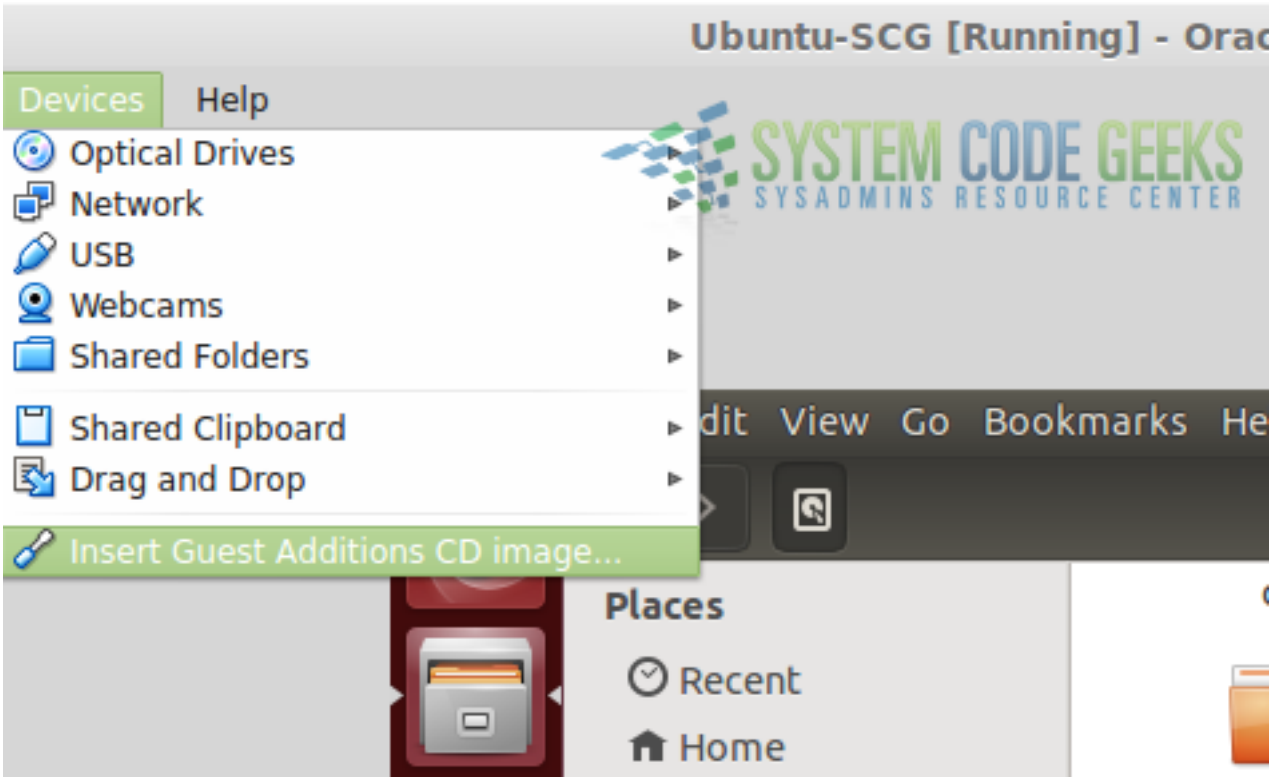


Figure 1.7: Using the virtual CD to install the guest additions

Then, use the File manager or File explorer to view the storage devices connected to the virtual machine and you will notice that a virtual CD has been added, which contains the guest additions files. As you open it, you will find inside an .exe file to install it on Windows, or a .sh script to do it on Linux and Unix-like operating systems. Double click on the right file depending on your case and follow the installation instructions as will be displayed on the screen. Fig. 1.8 illustrates this in our Ubuntu virtual machine (click **Run software** to proceed with the installation):

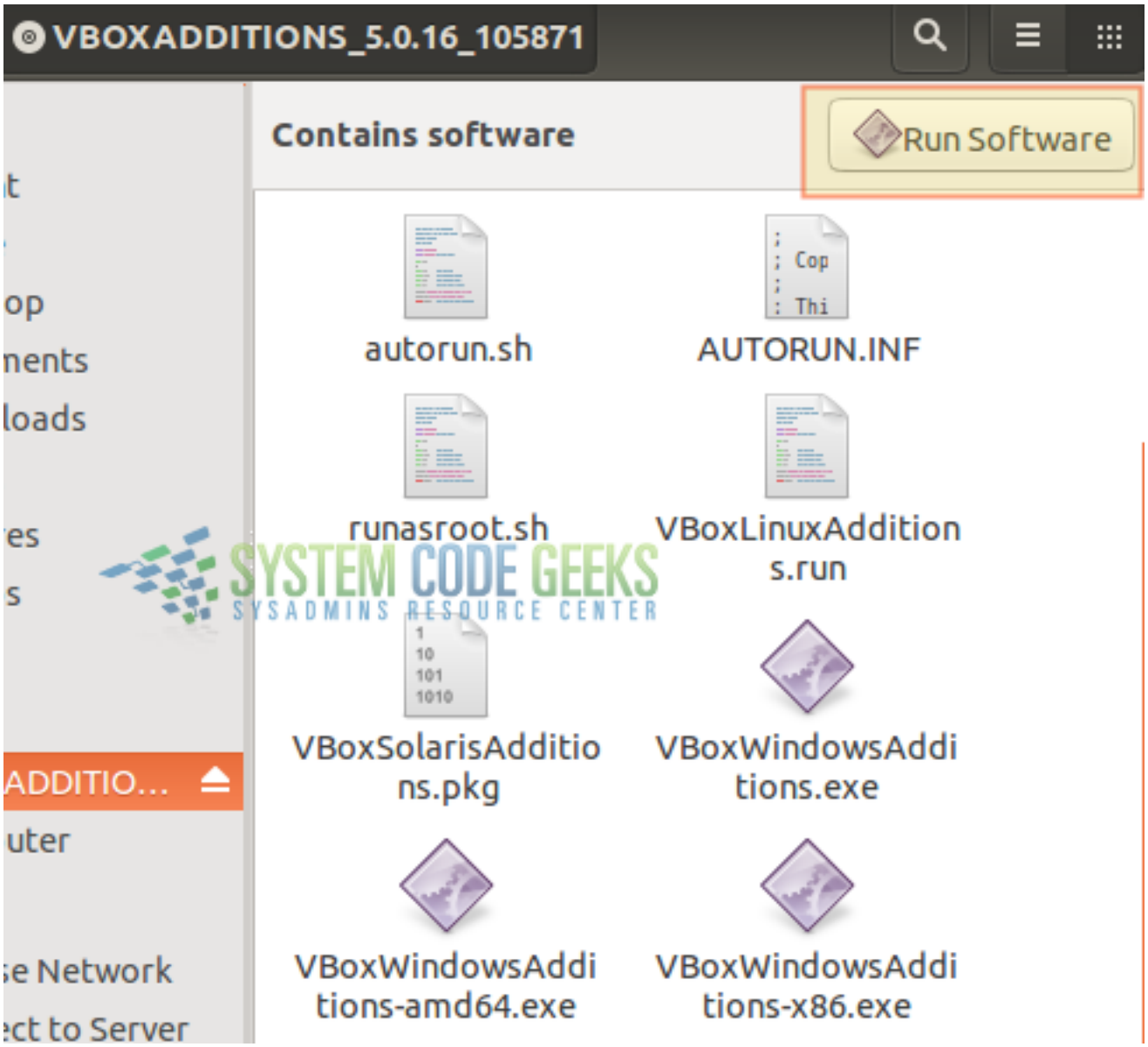


Figure 1.8: Running the guest additions installation program

The guest additions will then be applied to your new virtual machine the next time it boots (for example, you will notice that the screen of the virtual machine can now be resized to occupy the entire screen of the host).

Finally, to install the extension pack you will need to download the file from the VirtualBox site (click on the All supported platforms link). Once downloaded, open VirtualBox and go to **File** → **Preferences** and then choose **Extensions**, as seen on Fig. 1.9. Click on the highlighted button to choose the location of the extension pack file and then click on OK to install it:

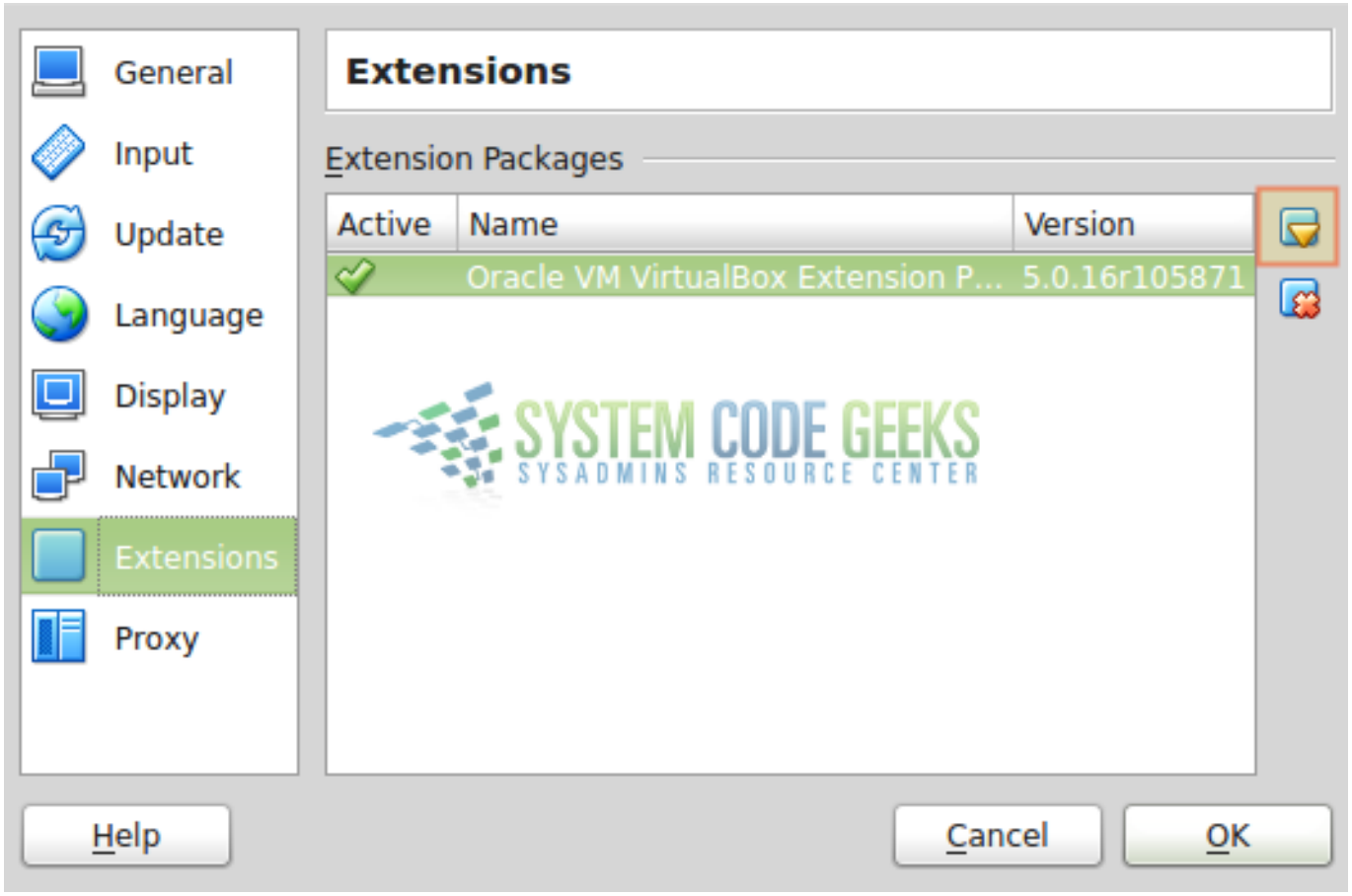


Figure 1.9: Installing the extension pack

After reviewing the license terms, you may be prompted to enter superuser credentials or confirm that you want to perform the installation of the extension pack using administrator privileges. In any event, choose Yes and wait until the process completes.

At this point you will have the features available through the extension pack, which we mentioned earlier, in your VirtualBox installation.

1.4 Summary

In this article we have introduced you to the concept of virtualization and the features of VirtualBox to create virtual machines. As a plus, everyone with the necessary development skills is invited and welcomed to participate in the development and improvement of VirtualBox. At the same time, Oracle ensures the product meets strict quality standards. Thus, you can rest assured that this product has the capability and the potential to meet your virtualization needs, regardless of what those may be.

Chapter 2

Cloning, exporting, importing, and removing virtual machines in VirtualBox

In the first part of this series, you learned how to create and start a virtual machine after installing VirtualBox on your system.

As you will recall from that article, starting and stopping a virtual machine is analogous to pressing the Power button in a computer and shutting it down using the corresponding operating system menu, respectively.

VirtualBox, as a virtualization manager interface, allows us to perform other operations as well: cloning, exporting, importing, and removing virtual machines, as shown in Fig. 2.1.

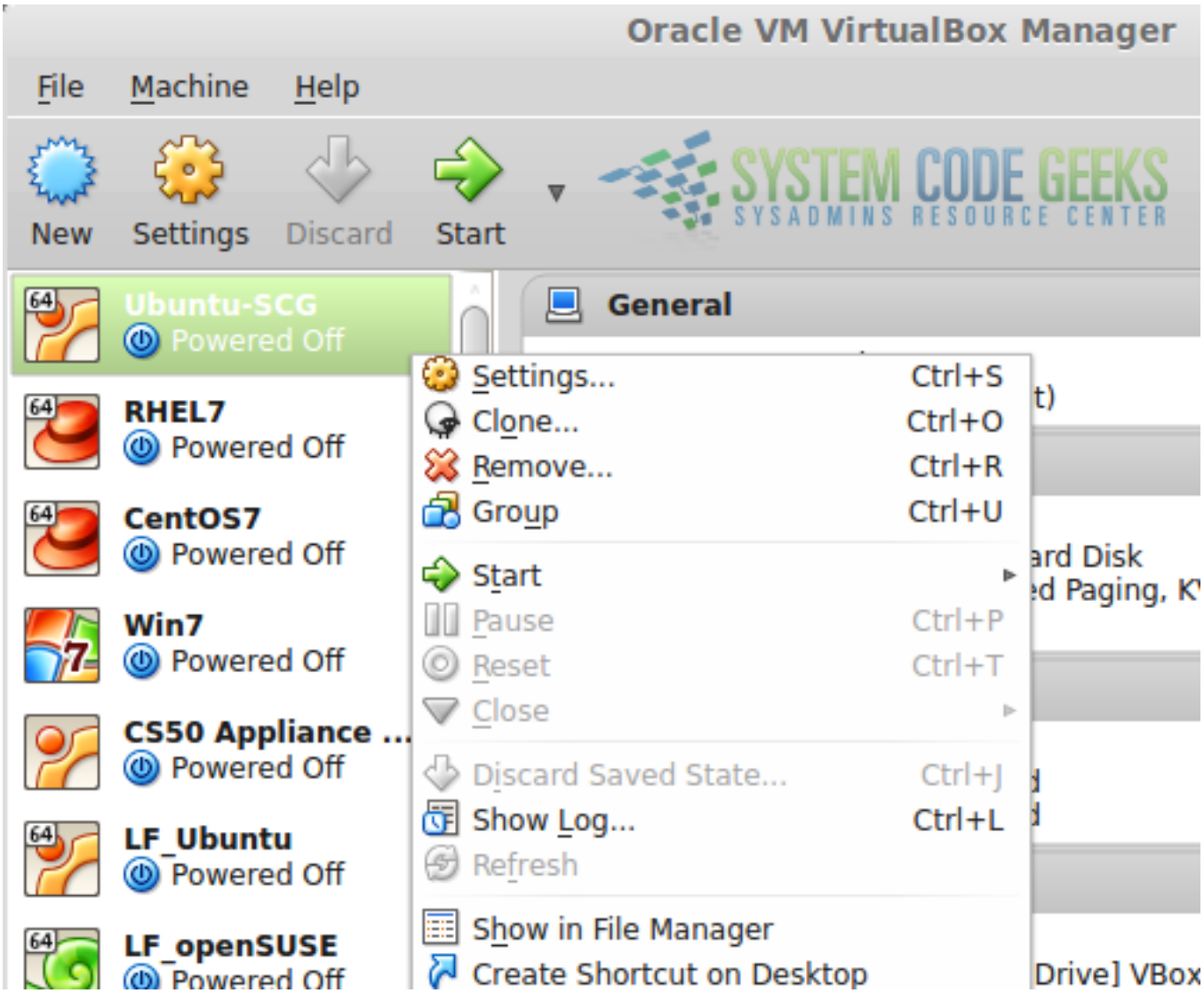


Figure 2.1: Available options for managing virtual machines

In addition, we can also create shortcuts to virtual machines on our desktop for our convenience (click on Create Shortcut on Desktop). Let us explore each of these operations in the following sections.

2.1 Cloning virtual machines

Having been introduced in VirtualBox 4.1, the process of cloning a virtual machine, as its name suggests, will result in an exact copy of such virtual machine. This feature is highly useful when you need to setup / configure a system and install further programs in it and then make exact copies that should work just like the original (for example, while installing nodes for a cluster setup). Cloning a virtual machine saves the time that would otherwise be spent in doing all of this from scratch.

To clone a virtual machine, right click on it and choose **Clone** (see Fig. 2.2):

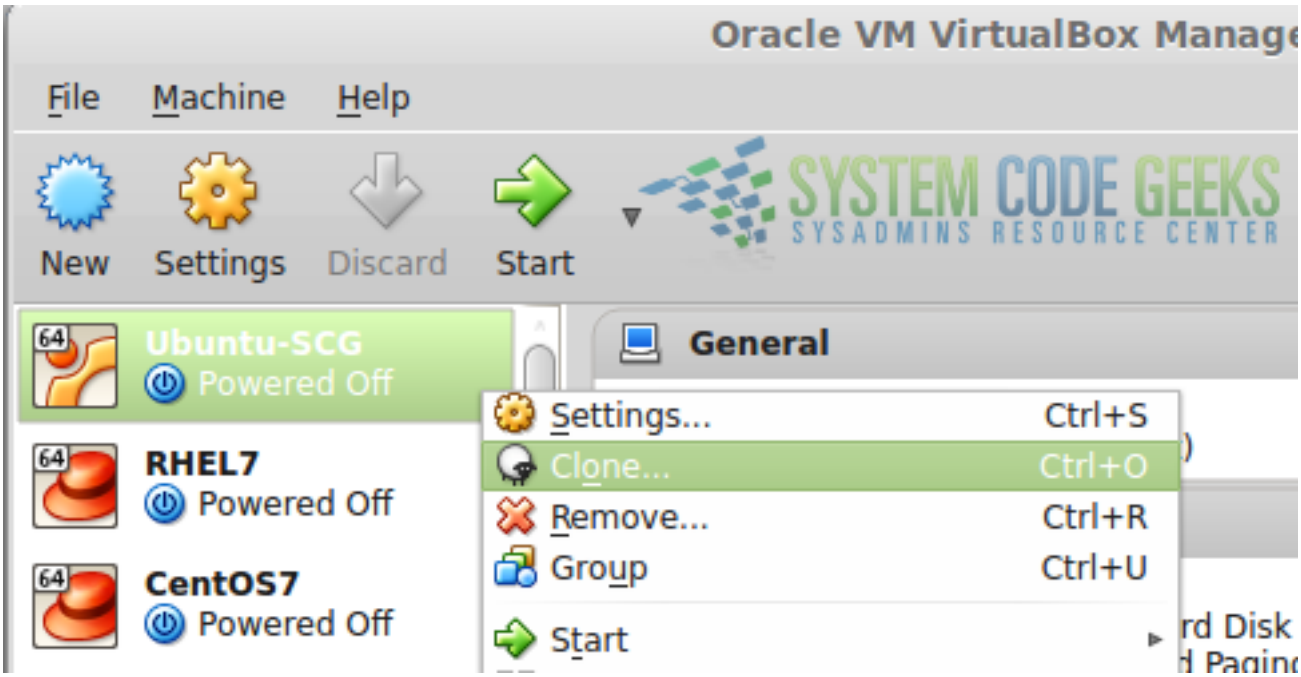


Figure 2.2: Starting the process of cloning a virtual machine

You will then be prompted to choose a name for the clone and asked whether you want to reinitialize the MAC of the network card(s) included with the machine. If the clone will be part of the same network as the original virtual machine, make sure the checkbox next to *Reinitialize MAC address of all network cards* is checked, as seen in Fig. 2.3:

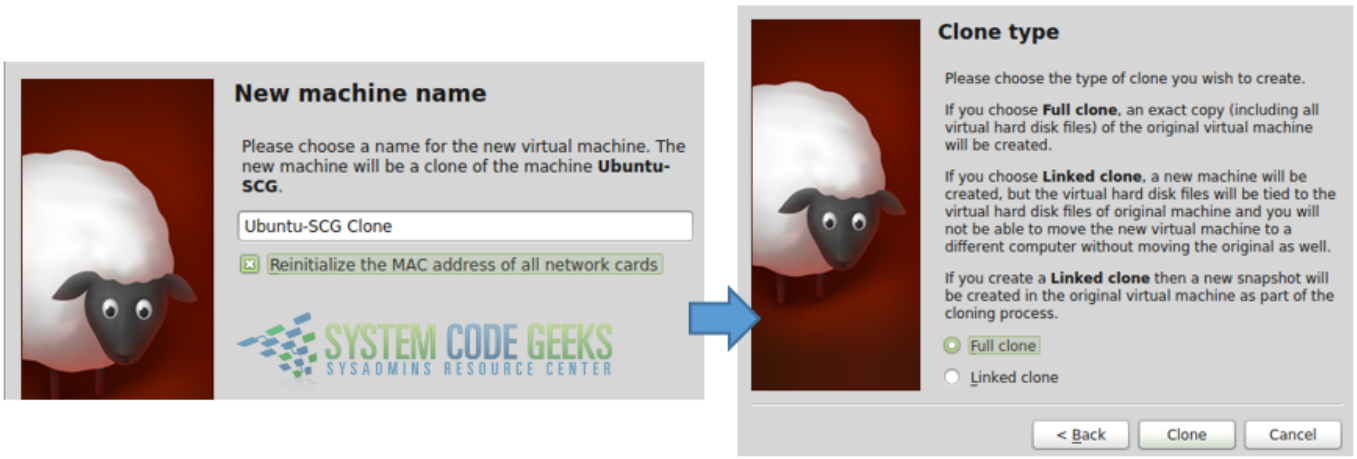


Figure 2.3: Choosing a clone name and type

You can choose either a full clone or a linked one. As indicated in Fig. 2.3, a full clone guarantees that each virtual will have its own disk(s) and will make moving resources around much easier. When the process completes, the clone will show up as a separate virtual machine on the list.

2.2 Exporting virtual machines

If you are backing up your virtual machines or want to pack up one for distribution (provided that you have the appropriate license to do so), you can export it to a single, bundled .ova file that can be imported into a separate VirtualBox installation.

To export a virtual machine, click on the **File** menu and go to **Export Appliance** (by the way, you can use the words *appliance* and *virtual machine* interchangeably in this context) as shown in Fig. 2.4:

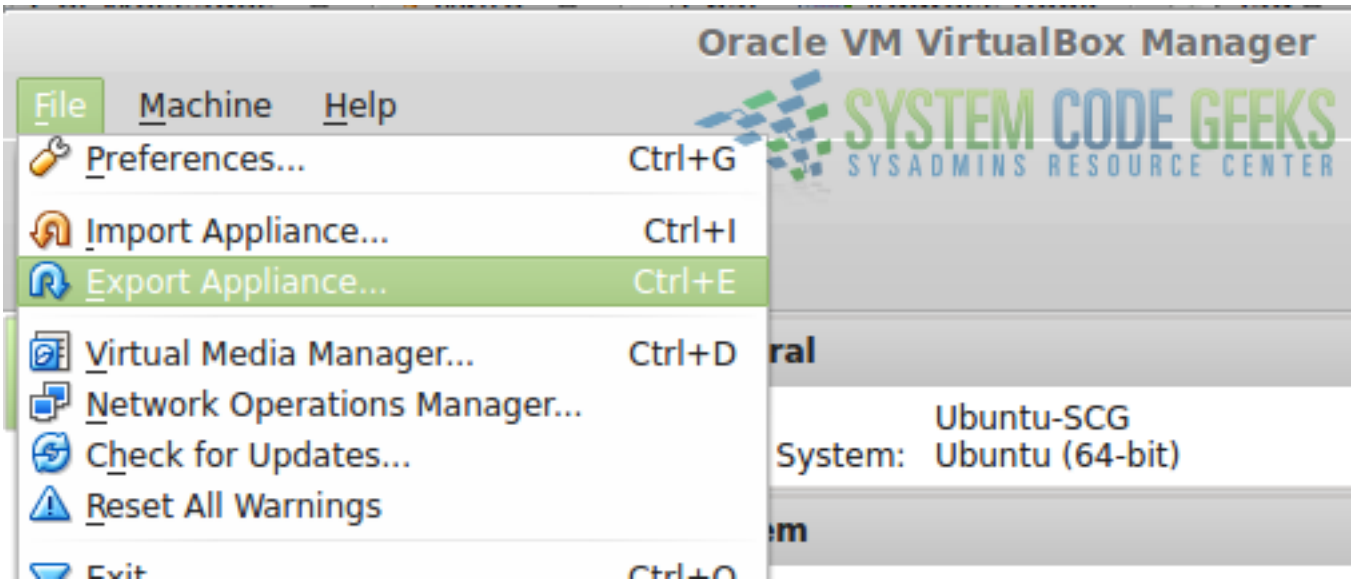


Figure 2.4: Starting the export process

Next, you will be presented with the list of virtual machines where you will be able to select one (or more, by pressing Ctrl + click), as shown in Fig. 2.5.

Figure 2.5: Selecting virtual machines to export

You can use the Expert Mode button to view the next steps (appliance and storage settings) in one single screen. However, it is often more useful for beginners to proceed one step at a time, as explained below. When you press *Next* in the above screen, you can skip over the appliance and storage settings, and the export process will begin (refer to Fig. 2.6).

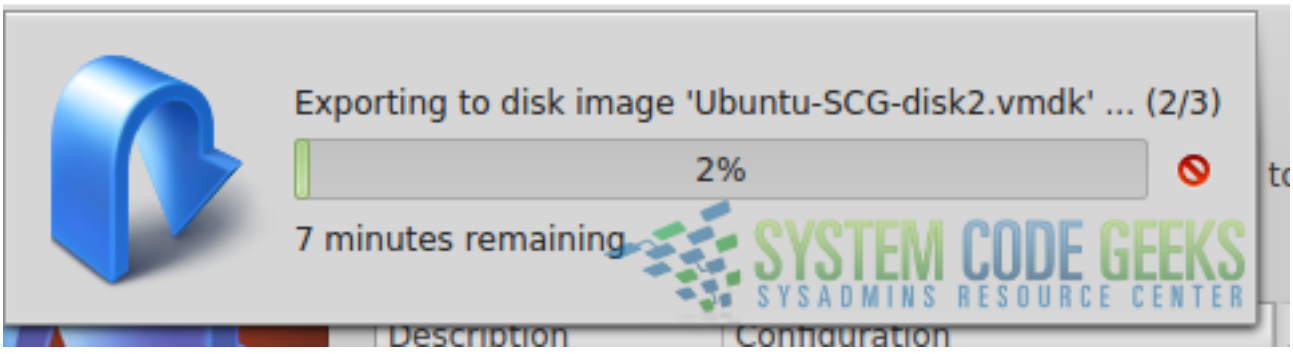


Figure 2.6: Beginning the Export Process

The export process should not take more than a few minutes to complete. Once done, you can find the associated .ova file in your Documents folder. This is the file that you want to back up or use for distribution. VirtualBox uses the Open Virtualization

Format (OVF) to export and import virtual machines. This is a cross-vendor standard and cross-platform solution that allows appliances created using other virtualization software to be imported into VirtualBox, and vice versa.

2.3 Importing virtual machines

In the previous section you learned how to export a virtual machine. That process bundled all of the virtual machine resources into a single .ova file. To import the virtual machine, go to **File** and choose **Import Appliance**. You will then be prompted to browse your system to look for the .ova file, as shown in Fig. 2.7:

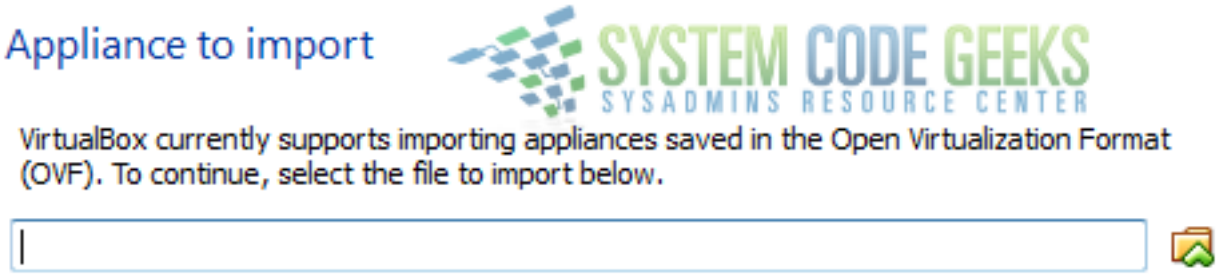


Figure 2.7: Selecting a virtual machine file to import

Once you selected the .ova file, follow the indications provided by the Next and Import buttons. As in the export process, importing a virtual machine may take a few minutes to complete, and will result in a separate virtual machine with the same name as the original one, plus a numbered suffix that identifies this as a different appliance. During the import process, you will see a screen that shows the appliance and storage settings for the virtual machine, as they were set originally when it was first created. You can edit those settings by double clicking on the corresponding text box and entering the desired value. We will now discuss how to remove virtual machines once they have served their purpose.

2.4 Removing virtual machines

If you set up one or more virtual machines just for testing, or for another temporary purpose, you may want to remove it once you are done. To do this, right click on the virtual machine in the list and choose Remove, as shown in Fig. 2.8. You will asked if you want to remove all the virtual storage files (which were created along with the appliance in the previous tutorial) associated with the virtual machine or just delete it from the list. For most cases, deleting all files is the way to go (unless a virtual disk is part of a RAID array or a logical volume that you want to keep for future use).

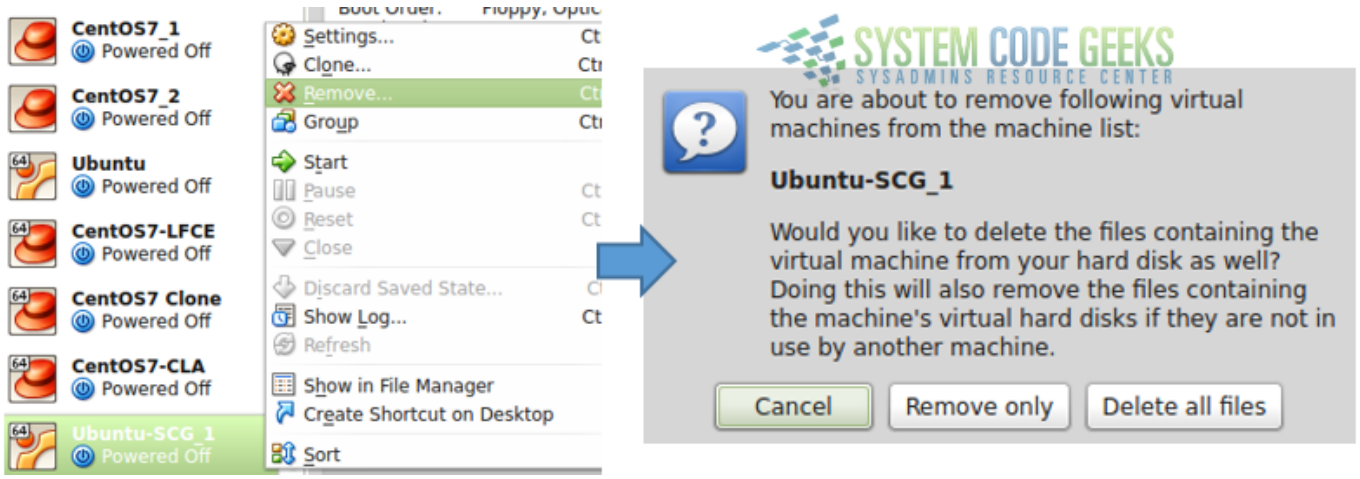


Figure 2.8: Removing a virtual machine

Please note that the *Remove* button remains disabled while a virtual machine is running, so you will need to shut it down before being able to delete it.

2.5 Inspecting logs

When you right click on a virtual machine, there are other options you may want to look at in the menu that pops up. For example, Show Log will open a separate VirtualBox window where you will be able to inspect the log of that machine. This will serve as a debugging tool should something go wrong while starting up or operating the appliance.

For example, the following lines describe a Windows 7 virtual machine running on a host which only has 939 MB of RAM available (output has been truncated for the sake of brevity):

```
VirtualBox VM 5.0.14 r105127 win.x86 (Jan 19 2016 17:01:13) release log
00:00:01.593500 Log opened 2016-04-11T17:59:13.526233700Z
00:00:01.593501 Build Type: release
00:00:01.593504 OS Product: Windows 7
00:00:01.593506 OS Release: 6.1.7601
00:00:01.593507 OS Service Pack: 1
00:00:01.701203 DMI Product Name: HP ProBook 6450b
00:00:01.706539 DMI Product Version:
00:00:01.706563 Host RAM: 2991MB total, 939MB available
00:00:01.706566 Executable: C:\Program Files\Oracle\VirtualBox\VirtualBox.exe
...
```

That amount of available RAM is not enough to launch a virtual machine that has been configured with 1 GB of memory. This leads to “HostMemoryLow” error preventing it boot, also present in the log:

```
00:00:19.696789 VM: Raising runtime error 'HostMemoryLow' (fFlags=0x2)
00:00:19.758101 Changing the VM state from 'RUNNING' to 'SUSPENDING'
```

Make inspecting the logs a habit. That will help you ensure that your virtual machines operate without issues.

2.6 Summary

In this article we have explained how to clone, export, import, and remove virtual machines in VirtualBox. In addition, we have shown how the operation of a virtual machine is kept in logs from the time it boots until the time it is shutdown. The logs will be a great troubleshooting resource when you experience issues.

Chapter 3

How to create, restore, and delete virtual machine snapshots in VirtualBox

In our previous article we discussed how to perform several operations with virtual machines including cloning, exporting, importing, and deleting appliances. Among other things, this allows us to back up our virtual machines and to distribute them in a single, bundled file. So far so good.

But what happens if you need to have restore points of your virtual machines you can roll back to in case something goes wrong or does not work as expected while you are making changes (or deploying code, to name another example)? The same principle applies when you want to go back to a given state of the appliance, and that is where snapshots come in handy.

In this tutorial we will explain what snapshots are, how taking a snapshot and restoring it is different than exporting and importing the virtual machine, and give a few suggestions to consider in order to effectively use them in your daily tasks.

3.1 Taking a snapshot

The very first difference between exporting a virtual machine and taking a snapshot is the fact that you can perform the latter operation regardless of the current status of the machine. You don't necessarily have to shut it down in order to take a snapshot.

When you take a snapshot of a running appliance, the exact memory state is saved as well, along with the active processes. This may result in a security risk if you are logged on to that machine since restoring this snapshot will bypass the need to enter login credentials in order to use it. However, this may be what you want, so use it at your own discretion.

To begin, open VirtualBox and select the appliance which you want to take a snapshot of. Then click on the Snapshots button as illustrated in Fig. 3.1. The right pane will then display the list of snapshots that have been taken of that virtual machine (if any).

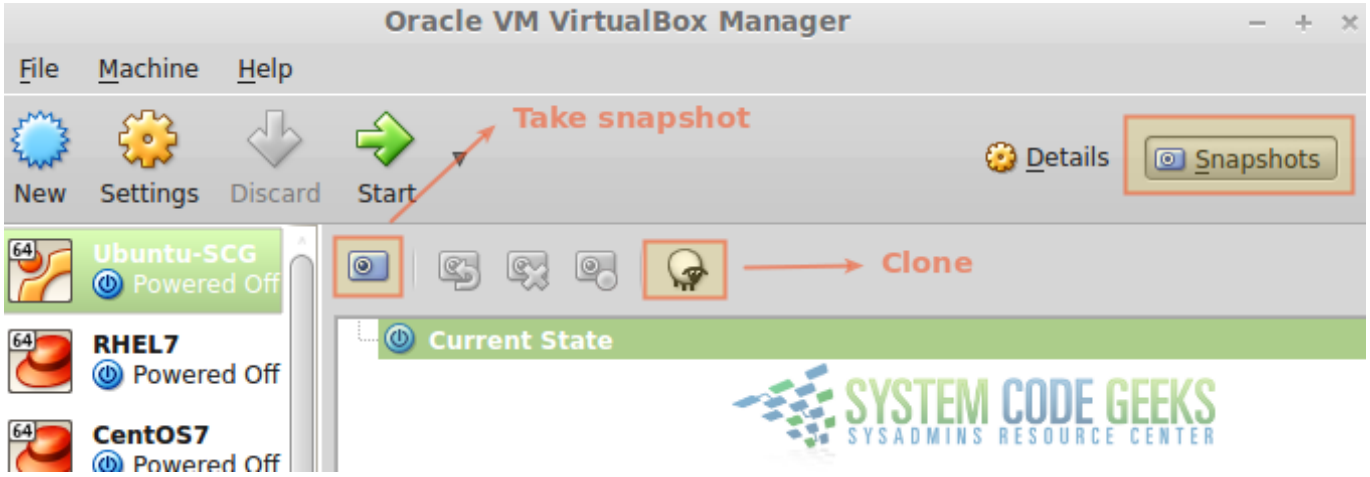


Figure 3.1: Switching to the Snapshots pane

To create an initial snapshot, click on the camera icon shown in Fig. 3.1. You will then be prompted to choose a name for the snapshot and enter a brief description for it, as shown in Fig. 3.2. Then click OK.

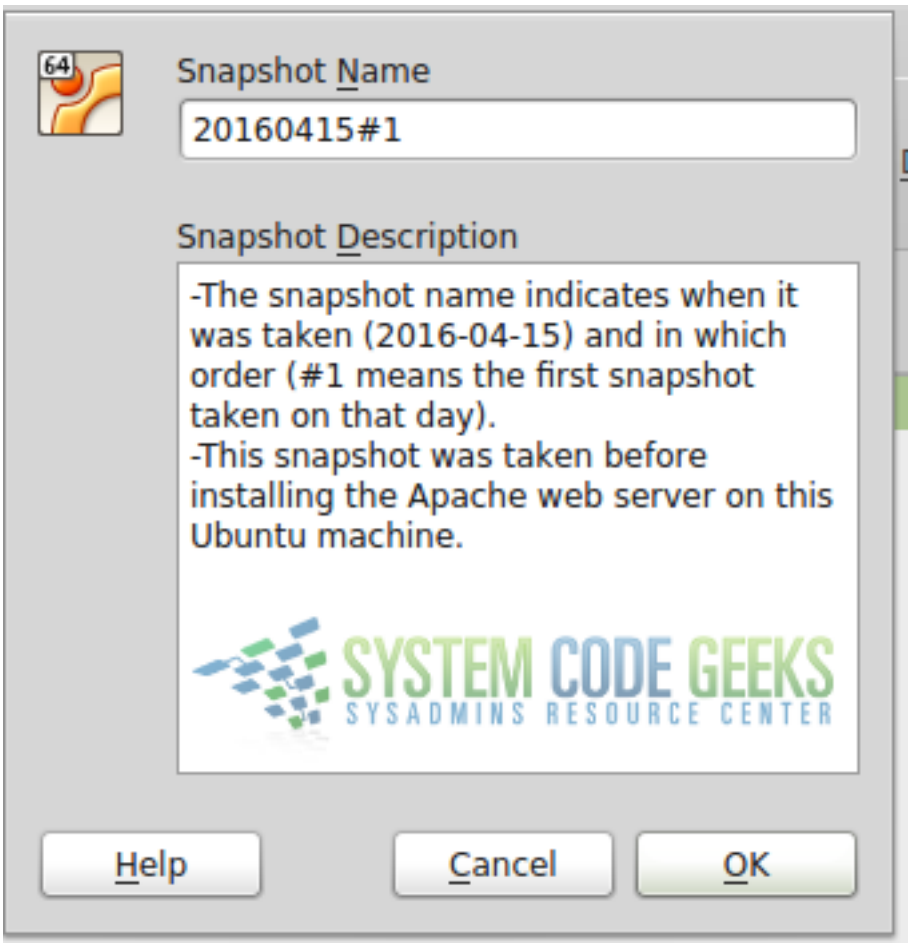


Figure 3.2: Choosing a name and entering a description for the current snapshot

It is a good idea to use the date when the snapshot was taken and another detail you choose. Again, write a description that is as

detailed as possible in order to help you know which snapshot you should restore to go back to a specific point in time.

You should now see the snapshot in the list as shown in Fig. 3.3:

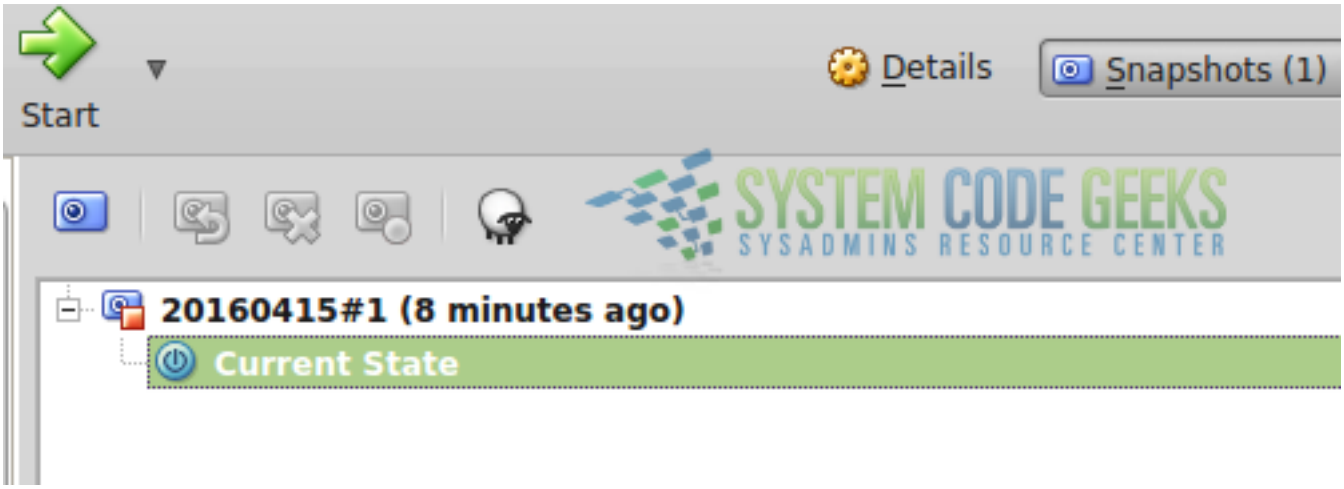


Figure 3.3: Listing virtual machine snapshots for the first time

That said, let's start the machine, make some changes (we will install the Apache web server as explained in "How to install the Apache web server (CentOS / Ubuntu Linux Installation)" using the terminal, which will create the corresponding configuration files as well). When we are done, Apache should be running and its configuration files present as shown in Figure 3.4.

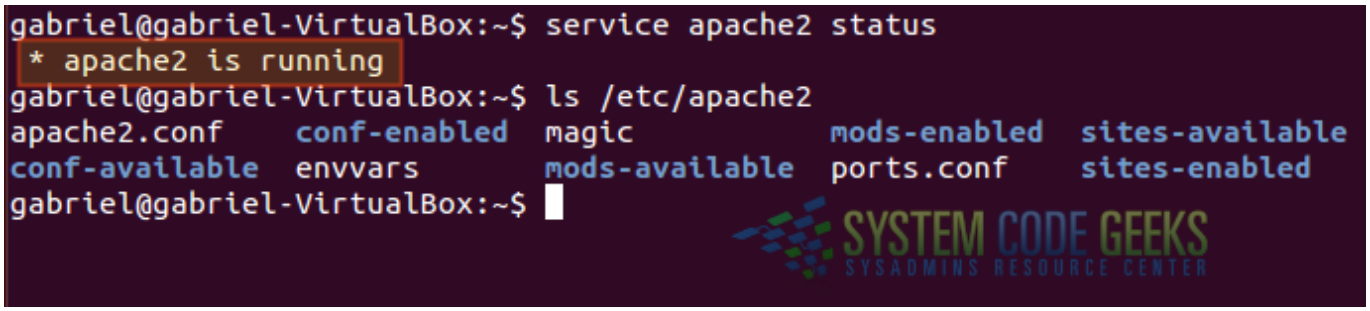


Figure 3.4: Checking the status of Apache and configuration files after installation

Now let's shut the machine down and take another snapshot as explained earlier.

3.2 Restoring snapshots

You should now have two snapshots of the appliance as seen on Fig. 3.5. Note how there are three buttons that are now enabled as compared to Figure 1.

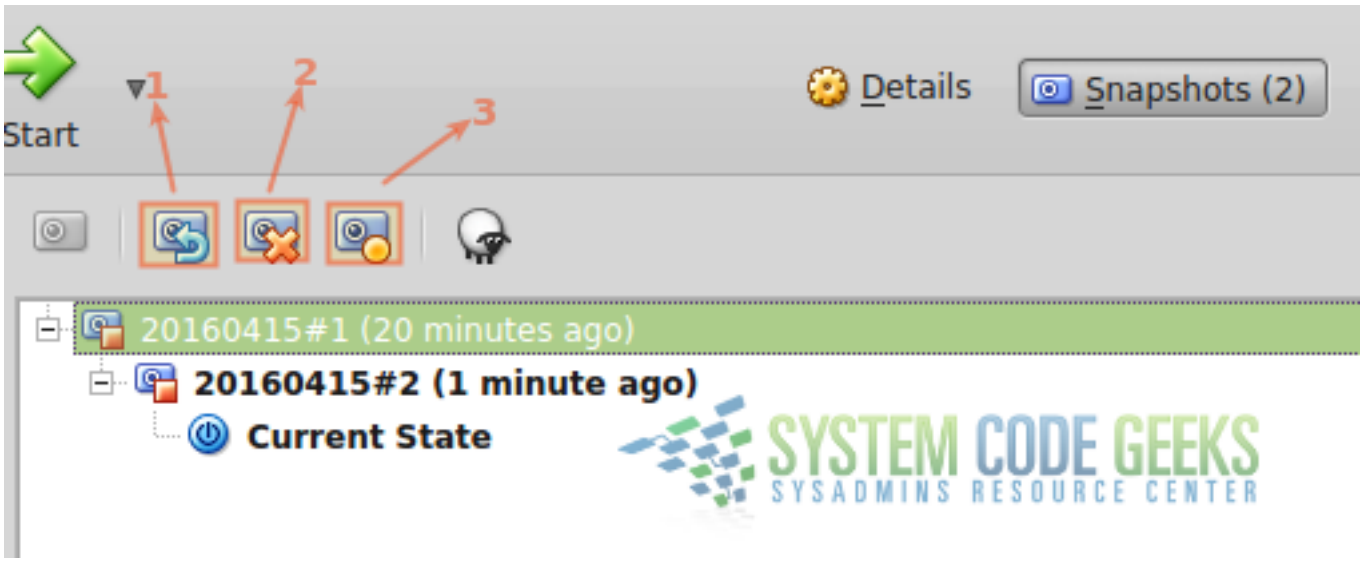


Figure 3.5: Listing snapshots after creating a second one

Pay special attention to the highlighted buttons in Fig. 3.5:

- Restore the selected snapshot.
- Delete the selected snapshot.
- Display a window with information about the selected snapshot.

Once we have made sure the first snapshot is selected, let's click on the button indicated as #1 above. You will be prompted to confirm the restoration. Choose **Restore** in the popup window that will appear (see Fig. 3.6 for more details).

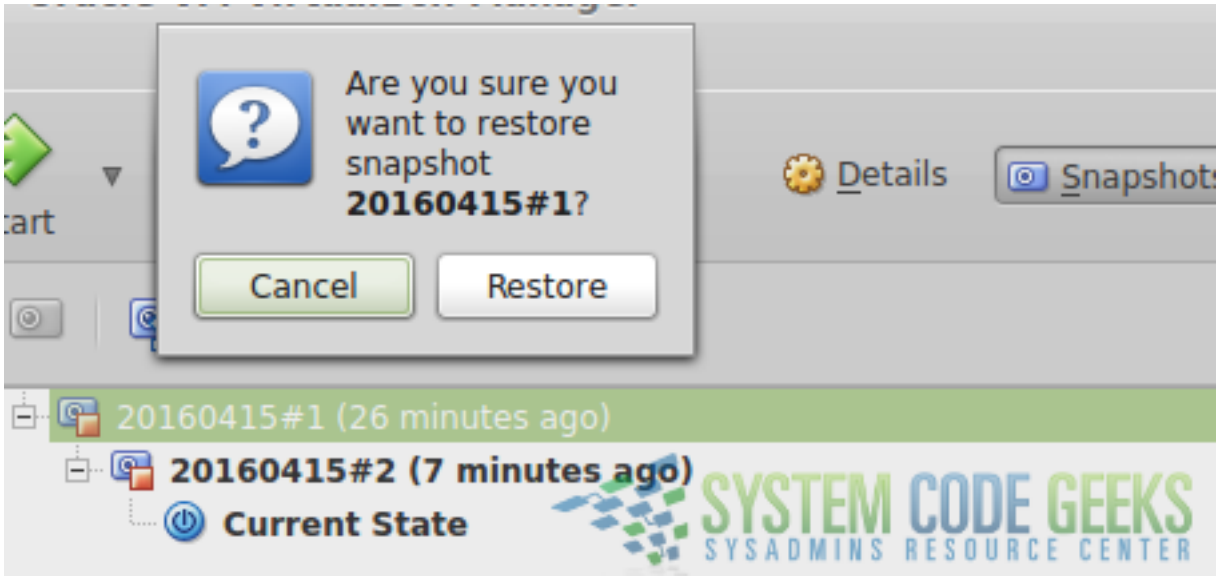


Figure 3.6: Confirm before restoring the snapshot

Then let's start it and check that Apache and its configuration files are not present. The output should be as shown in Fig. 3.7:

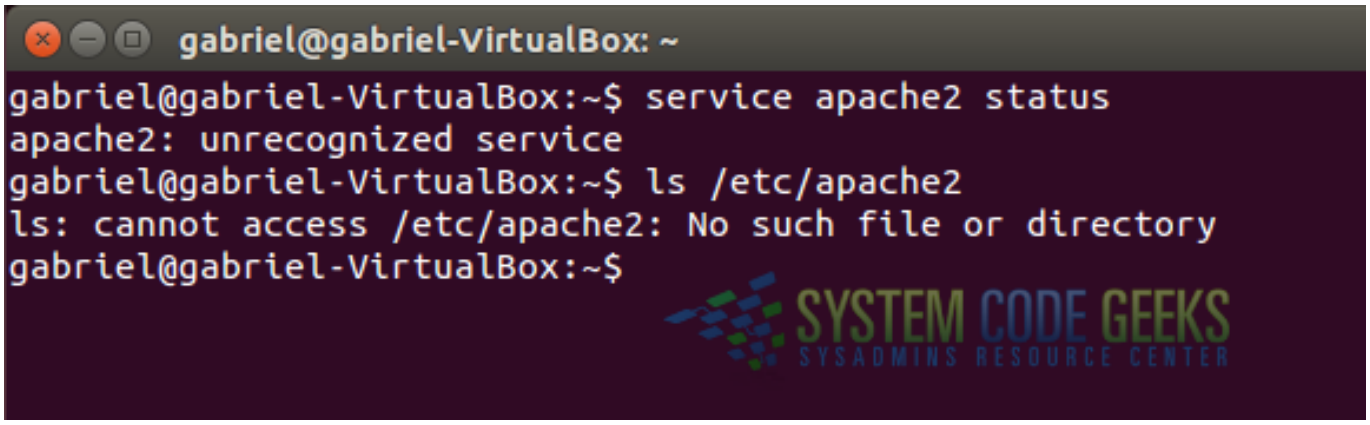


Figure 3.7: Checking the status after restoring the virtual machine to a previous snapshot

As a personal exercise, you can turn off the appliance and restore the second snapshot. You should then be able to use Apache again. As you create and restore snapshots, pay attention that this process is generally a faster solution than exporting and importing an entire virtual machine.

3.3 Things to consider when restoring a snapshot

You have now learned how to create snapshots and how to restore them. It is a good time to point a few important facts you need to consider:

- The hardware settings of the virtual machine are restored as well. If you had assigned more memory, for example, restoring a previous snapshot will undo this change.
- As for modifications made to files, keep in mind that files previously created will be gone, files that had been deleted will be restored, and any changes to existing files will be reverted.
- The snapshots are stored as .vdi (virtual drive) files in a folder named Snapshots under the same directory where the virtual machine files are kept (you can browse directly to this directory by right-clicking on the appliance and choosing **Show in File Manager** from the popup menu).

You may need to pay special attention to the last item in the list above, as carelessly creating snapshots will result in filling up storage space that may be needed for something else. Make sure to check the snapshots regularly and delete the ones you don't need anymore.

3.4 Summary

In this tutorial we have discussed what virtual machine snapshots are, and explained how to create, restore, and delete them. You are in a better position to operate your appliance now, as you can create restore points to roll it back to a specific state or point in time.

Chapter 4

How to configure the settings of a Virtual Machine in VirtualBox

In previous tutorials you learned how to create virtual machines and perform several operations (clone, import, export, and delete) with them.

We also advised you to become familiar with the logs of your appliances as they are a very useful troubleshooting tool.

Let's start by copying a few lines from the logs of the Windows 7 virtual machine we used as an example in Cloning, exporting, importing, and removing virtual machines in VirtualBox:

```
...
00:00:19.696789 VM: Raising runtime error 'HostMemoryLow' (fFlags=0x2)
00:00:19.758101 Changing the VM state from 'RUNNING' to 'SUSPENDING'
...
```

As the error message suggests, the amount of available RAM was not enough to launch this virtual machine. This is just one example of how the logs will point out the need to update the appliance's settings in order for it to work as expected. To fix this, in this article we will discuss how to configure (or update the settings of) a virtual machine so that its general, system, network, and storage settings meet the requirements of the guest operating system and the applications running therein.

4.1 Configuring a virtual machine

The VirtualBox user interface provides a friendly alternative to the command line utilities that are utilized to configure an appliance. After opening VirtualBox, right click on a virtual machine of your choice and choose *Settings...* from the popup menu. In the next screen (left hand box) you will see the configuration categories. As explained earlier, we will only focus on the following ones because of their importance:

- **General**
  - Basic: Appliance name / type / version.
  - Advanced: Snapshot folder / shared clipboard / Drag and drop feature.
  - Description: virtual machine description.
  - Encryption: self-explanatory. Requires the use of a password to start the virtual machine.
- **System**
  - Motherboard: RAM / Boot order / Chipset.
  - Processor
  - Acceleration

- **Storage**
  - Controllers: add / remove disks.
- **Network**
  - Adapters: add / remove / change interfaces.

Let's begin with the **General** category.

4.2 General

Under General, you will find the mostly-informative categories listed above. The virtual machine type and version are almost always detected by VirtualBox when the appliance is first created, based on its assigned name. If you are planning on distributing the resulting appliance, it is a good idea that you take the time to write a detailed description with as much information about it as possible.

If, in addition, you want to prevent unauthorized people from starting the machine, you can encrypt the appliance by choosing an encryption algorithm and a strong password in the **Encryption** tab. Depending on that combination, and on the size of the virtual disk(s), this operation (illustrated in Fig. 4.1) may take considerable time to complete.

Figure 4.1: Encrypting a virtual machine

After you enter the right password, the boot process will resume as usual. Otherwise, the virtual machine will refuse to start.

4.3 System

This category allows you to provision your virtual machine with processing resources. While still limited by the hardware features present on the host, you will be able to change the amount of RAM allocated to the guest through the use of a slider, indicate the desired boot order and chipset (**Motherboard** tab), the number of CPU cores assigned to the virtual machine in the **Processor** tab (although possible, it is not recommended to assign more virtual CPUs to the guest than the host has available, as it may end up degrading the performance of the virtual machine).

In the Acceleration tab you can indicate whether VirtualBox should use (and how) the virtualization extensions supported by (and enabled in) the host CPU. In order to use a virtualization extension in a virtual machine, you will need to enable that feature in the host BIOS or UEFI first. Otherwise it will not be available for your appliance. Refer to the documentation of your motherboard for specific instructions on how to do this.

4.4 Storage

In a physical (actual) computer, when you need more storage space you buy new disks and install them. If you use Logical Volume Management (LVM), you then create physical volumes, volume groups, and then extend the existing logical volumes.

This, of course, requires that you have physical access to the machine, and know how to install the disks and make sure they are compatible with your motherboard. On the other hand, in a virtual setup -with only a few clicks- you can get all of this done in no more than a couple of minutes.

To add a new virtual disk, click on *Controller: SATA* and then on the plus sign that is highlighted in Fig. 2, where the beginning of the process is illustrated.

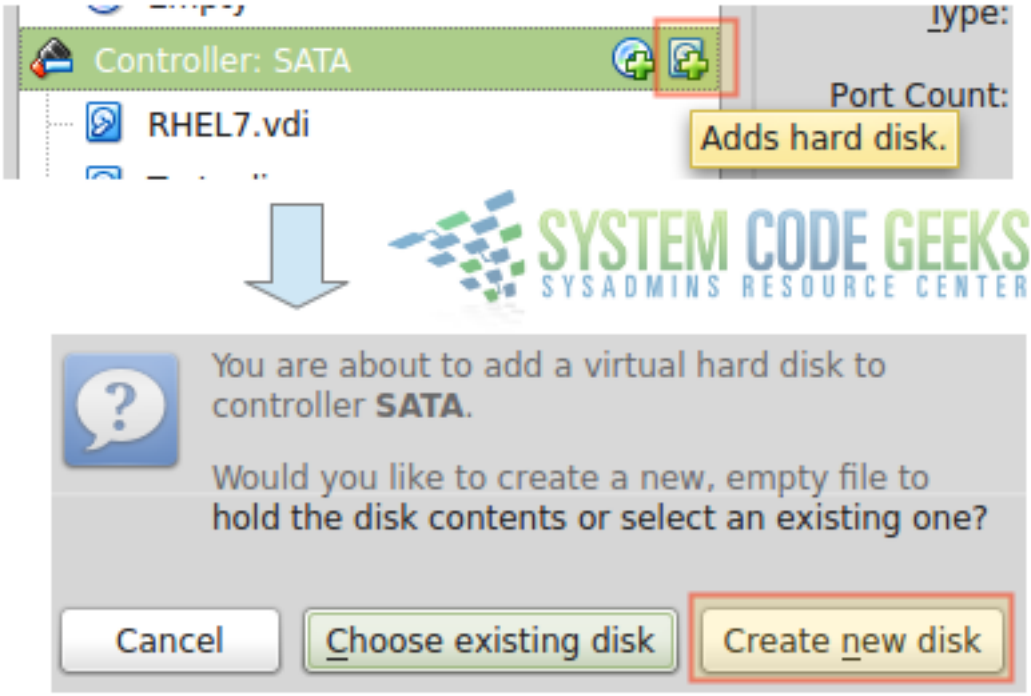


Figure 4.2: Adding new virtual disks

You will notice that the remaining steps (not shown in Fig. 4.2) do not differ significantly from the instructions given in a previous article to add a first hard disk to the virtual machine when you first created it. The only difference lies in the fact that here you can choose whether to use an existing (virtual) disk (perhaps a drive that was left from another machine), or create a new one.

4.5 Network

When you click on the Network category, you will see a list of available adapters for your virtual machine, which can be enabled (this is equivalent to plugging a network interface card (NIC) to an available PCI slot in your motherboard, and making sure it is properly recognized by it) or not.

For each adapter, you can choose a type (available in the advanced settings, as indicated in Fig. 4.3) that will be exposed to the virtual machine, and optionally refresh the MAC address (this is often needed when you have cloned an appliance and did not reset the MAC address previously):

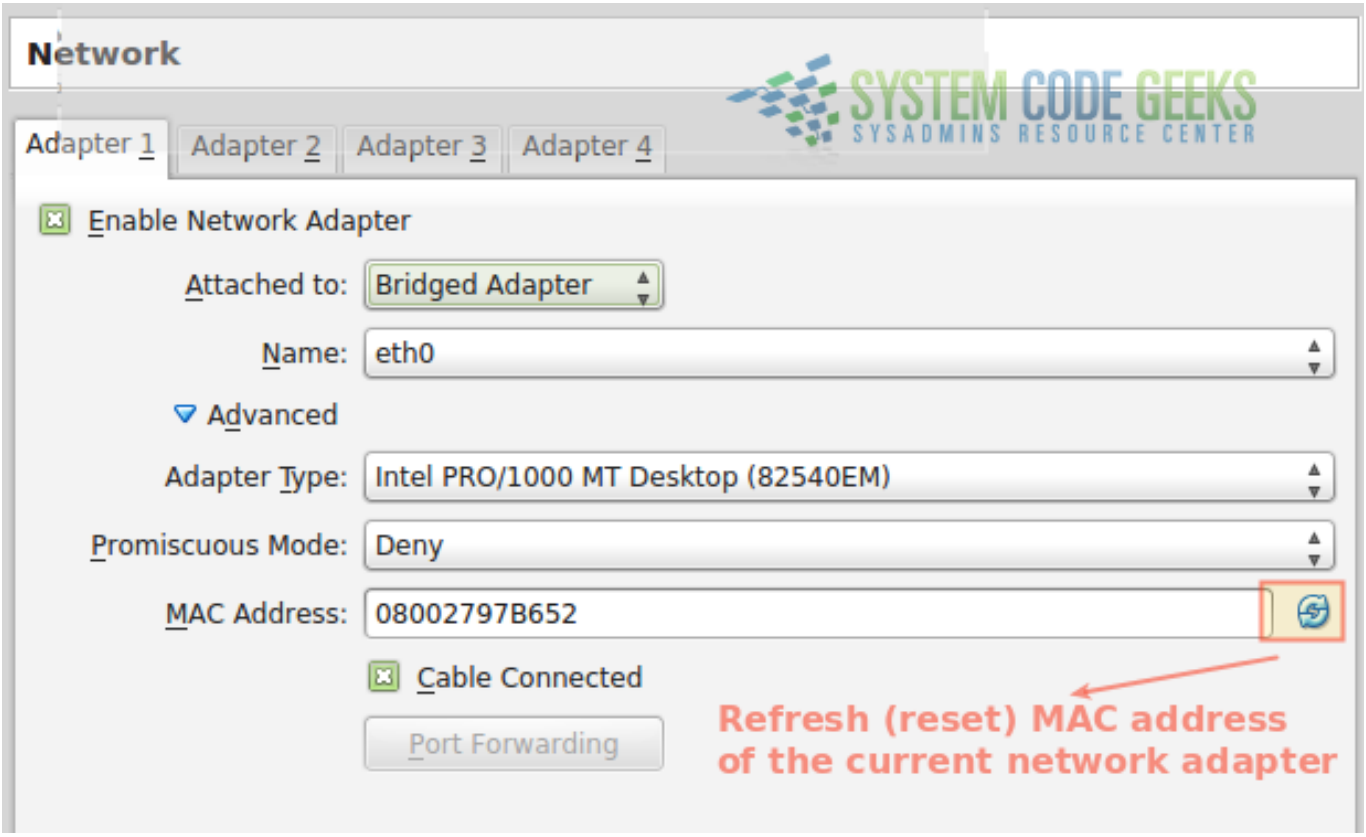


Figure 4.3: Configuring an existing (or a adding a new) network adapter

Perhaps the most important field of the network configuration is **Attached to:** value. The most used settings are:

- **NAT (Network Address Translation)** is the default networking mode, and is sufficient for the common usage of network services such as browsing the Internet.
- When an interface is configured as **Bridged Adapter**, VirtualBox connects directly to a physical network card and bypasses the host operating system's network stack, as opposed to NAT.
- **Internal Network** is useful when you need to avoid external applications or even the host itself to access your virtual machine. When combined with a separate Bridged Adapter, an internal network adapter can connect to an internal network while the bridged adapter connects to another (perhaps the Internet). This type of setup (shown in Fig. 4.4) is typical of a firewall or a HTTP proxy server.

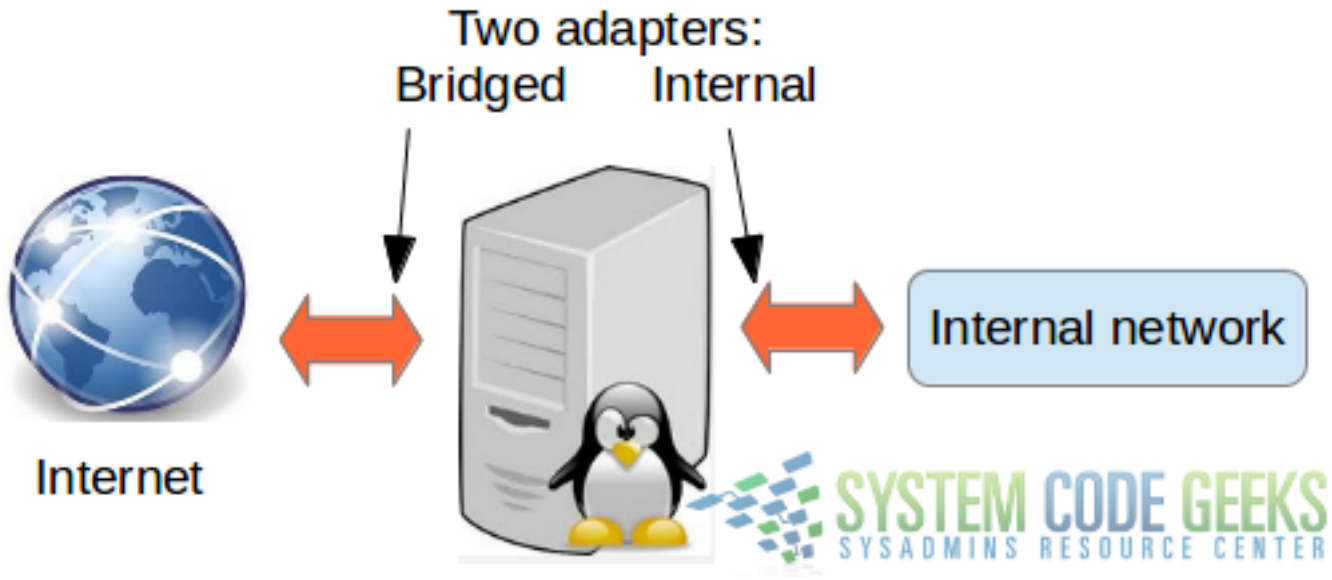


Figure 4.4: Configuring two separate network adapters to connect both to the Internet and to an internal network

4.6 Summary

In this article we have explained what are the most common settings you would like to consider changing while operating one or more virtual machines. Other configuration categories (such as **Audio** or **USB**) are better left off with the default values unless they do not work as expected in your setup. If that is the case, it is highly recommended to check the VirtualBox official documentation and / or support forums.

Chapter 5

Managing VirtualBox Virtual Machines through Command Line (with VBoxManage)

Throughout this series we have learned how to create, manage, take / restore snapshots and configure virtual machines via the VirtualBox user interface. By now you should feel comfortable performing these tasks through the UI.

In this article we will explain how to use `VBoxManage`, the command line alternative, to perform the same operations directly from the shell in your host machine. Among other things, this can prove useful when 1) you need to include your virtual machines in your backup routines (regardless of the operating system you are using), or if you want to perform a management operation through cron in Linux or the Task Scheduler in Windows.

Besides providing the same features of its GUI counterpart, `VBoxManage` provides others that cannot be accessed (yet) from it. On the other hand, there are tasks that are considerably easier to perform using the GUI, such as creating new virtual machines, and for that reason we will not cover how to do it using the command line.

5.1 Introducing VBoxManage

A distinguishing characteristic of `VBoxManage` is that it needs to be followed by a generic operation or an action to be taken on a virtual machine using its id. We will illustrate this through examples in the next section.

If you are using Linux, `VBoxManage` was added to your PATH environment variable when you installed it. You can verify by typing

```
which VBoxManage
```

If the above command returns the absolute path to the binary file (`/usr/bin/VBoxManage` in my case), it means the directory where `VBoxManage` resides (most likely `/usr/bin`) is in your PATH (no surprise here). I would be surprised if that is not the case.

If you are on Windows, `VBoxManage` should be located inside the Program Files → Oracle → VirtualBox folder. In order to be able to invoke the commands listed in this guide directly, you will need to use the command prompt (cmd) and either browse to that directory or add it to the current PATH (note that this method applies only to your current Command Prompt session):

```
set PATH=%PATH%;"C:\Program Files\Oracle\VirtualBox"
```

For the rest of this guide we will use Linux. If you are using Windows you should not see any major differences, if at all.

5.2 Listing virtual machines

To show lists, we will use `VBoxManage list`. Particularly, we are interested in listing

- Appliances currently registered with VirtualBox (see Fig. 5.1), along with their corresponding UUID: `VBoxManage list vms`

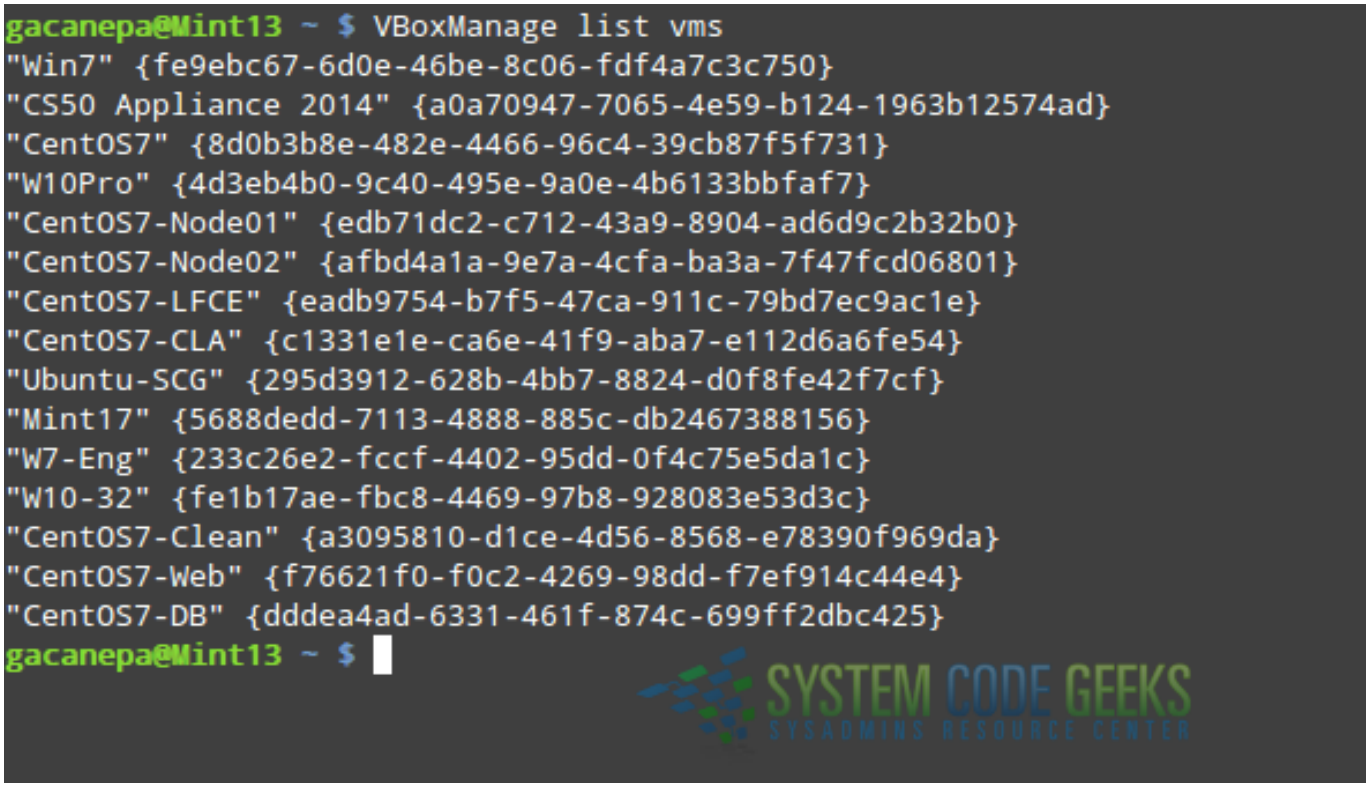


Figure 5.1: Listing appliances registered with VirtualBox

- Virtual machines that are currently running: `VBoxManage list runningvms`
- Extension packs that are installed: `VBoxManage list extpacks`

Fig. 5.2 shows the expected result of the last two commands, provided that you have at least one virtual machine running and have installed the extension pack as discussed at the beginning of this series:

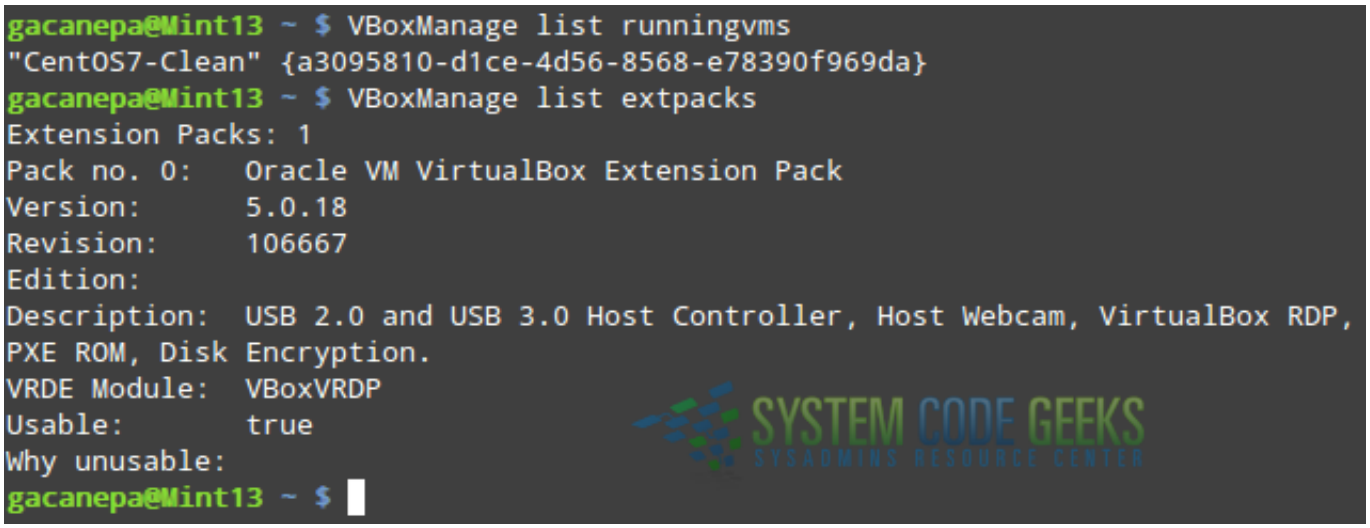


Figure 5.2: Listing running virtual machines and the installed extension pack

Keep in mind that you WILL NEED the extension pack in order to use USB drives and devices, among other things. If you arrived at this tutorial without reading the previous guides, you are strongly encouraged to take a look at Virtualization with VirtualBox: Installation and Configuration as soon as possible.

5.3 Showing and modifying information and settings of virtual machines

To show all the information available for a particular virtual machine, we will use `VBoxManage showvminfo` followed by the name of the appliance (as returned by `VBoxManage list vms`). You will not only be able to see settings as the name, the UUID, and the amount of RAM allocated for this appliance, but also the list (and description) of snapshots taken, the path to the snapshots directory, the location of the log and configuration files, and much more. For example,

```
VBoxManage showvminfo Ubuntu-SCG
```

will display all the information about the virtual machine named Ubuntu-SCG. Part of the expected output is shown in Fig. 5.3:

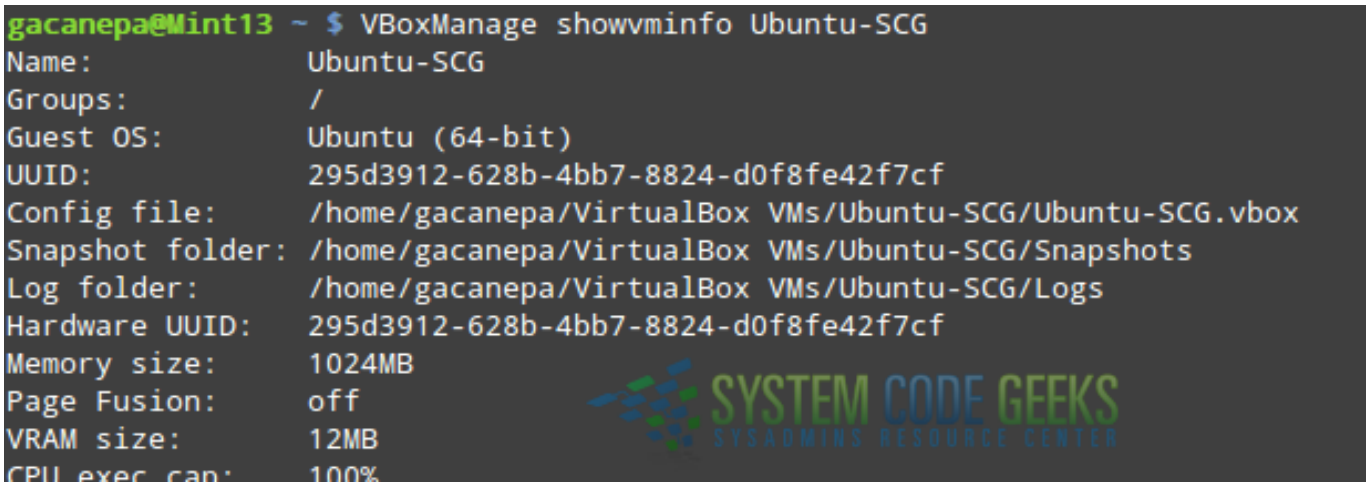


Figure 5.3: Displaying information about a virtual machine

Perhaps you will want to group virtual machines following some sort of criteria. For example, you may want to create a group for a specific project and add some virtual machines to it for easier administration.

Using the following command we will add **CentOS7-Node01** and **CentOS-Node02** to a group named **Cluster**. If the group does not exist, it will be created when the first appliance is added to the group. The forward bar (/) before the group name is necessary. Please note that operations that are targeted at modifying settings of virtual machines use the `VBoxManage modifyvm`:

```
VBoxManage modifyvm "CentOS7-Node01" --groups "/Cluster"
VBoxManage modifyvm "CentOS7-Node02" --groups "/Cluster"
```

After you run the above commands, the Cluster group will appear in the user interface with the list of machines that have been assigned to it, as can be seen in Fig. 5.4:

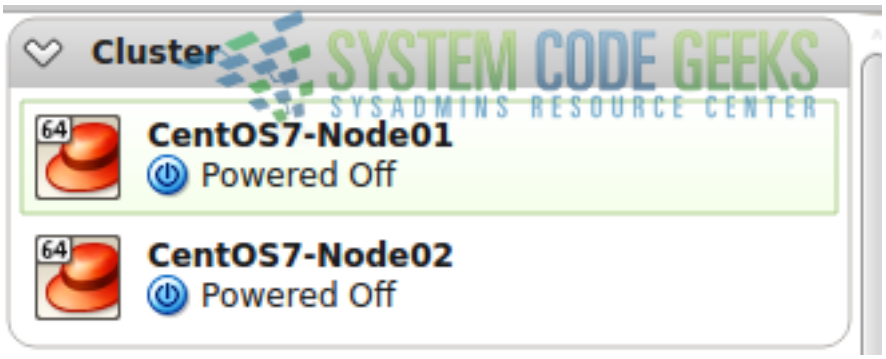


Figure 5.4: Grouping virtual machines

To remove a virtual machine from a group, do:

```
VBoxManage modifyvm "CentOS7-Node01" --groups ""
```

Now picture the following scenario. Perhaps you bought some more RAM for your host and can now afford to allocate 2048 MB (instead of the original 1 GB that was allocated during creation). In addition, you just realized that you need to resize the existing virtual disk of Ubuntu-SCG (currently 8 GB) to -let's say- 15 GB. We can perform both operations from the command line in a snap (note that the virtual machine must be stopped prior to running the following commands):

```
VBoxManage modifyvm Ubuntu-SCG --memory 2048
```

To resize a disk, we must first identify it and reference it using either its specific UUID (NOT the appliance's UUID) or its name. That means we will have to use VBoxManage showvminfo Ubuntu-SCG first to find out this information:

```
VBoxManage showvminfo Ubuntu-SCG | grep -Ei '(ide|sata)' | grep -i uuid
VBoxManage showhdinfo 6eed1d19-874b-43e2-9278-ef485660731e | grep -i capacity
```

Fig. 5.5 shows the values before and after the changes:

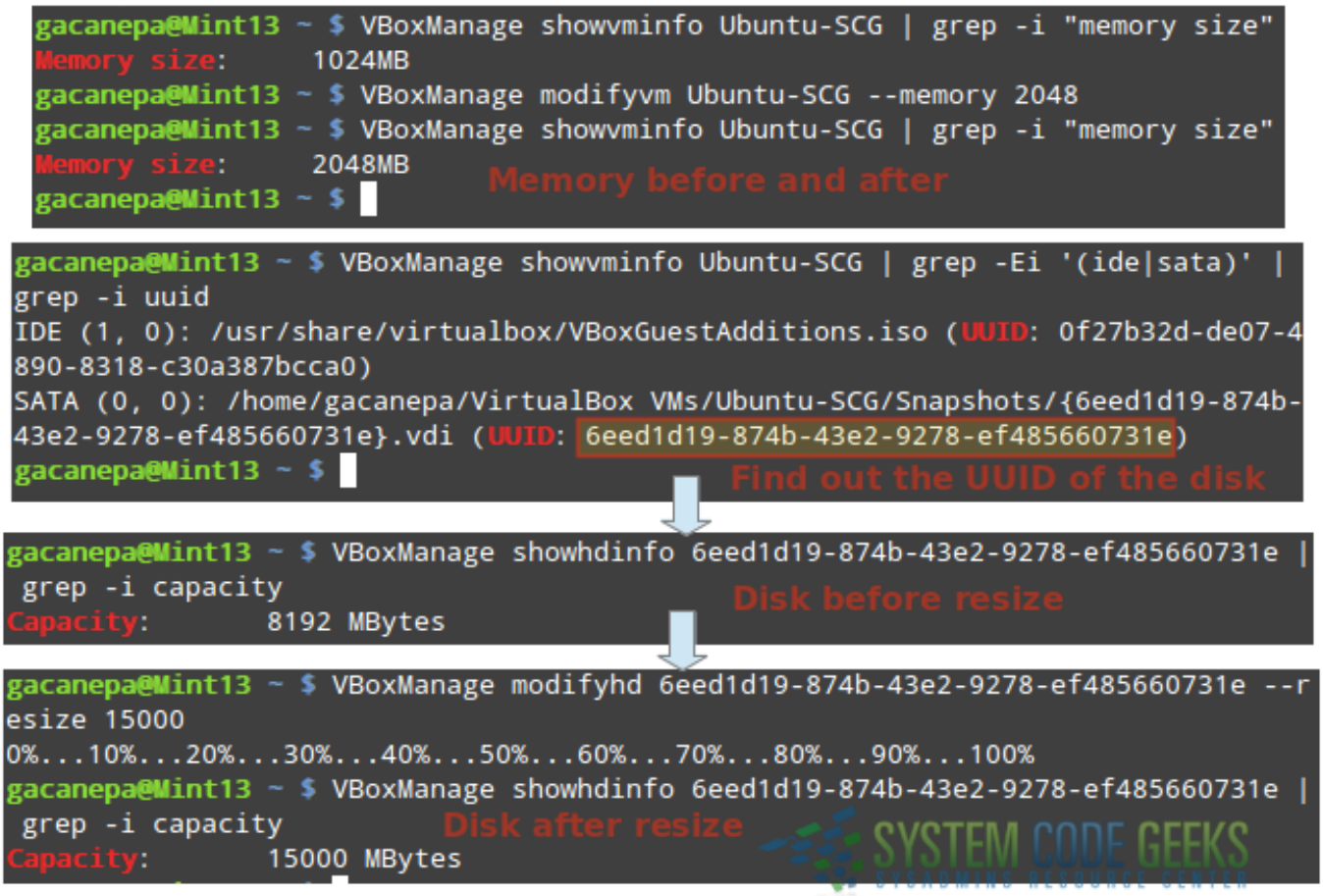


Figure 5.5: Resizing a virtual disk and allocating more memory for an appliance

Similarly, to change the name or the device the first network interface is attached to (bridged to eth0 in this case), you can do

```
VBoxManage modifyvm Ubuntu-SCG --name "Ubuntu-SCG-New"
```

and (use the new name here if you ran the above command previously):

```
VBoxManage modifyvm "Ubuntu-SCG-New" --nic1 bridged --bridgeadapter1 eth0
```

Let's now see how we can export virtual machines using the command line.

5.4 Exporting and importing appliances

In the case you do your backups using some sort of shell script, VBoxManage export (followed by the virtual machine name, can help you get the job done. The following command will export the CentOS7 appliance to **~/VMexports/CentOS7.ova**:

```
VBoxManage export CentOS7 --output ~/VMexports/CentOS7.ova
```

To import it (not surprisingly), we will use VBoxManage import:

```
VBoxManage import ~/VMexports/CentOS7.ova
```

You can use the `-n` (`--dry-run`) option with `VBoxManage` to simulate the import process. The machine will not be imported, just a simulation will be performed.

5.5 Starting and stopping virtual machines

Suppose you need to start a virtual machine at a give time. To accomplish this, you can create a shell script that runs -through cron- the following command:

```
VBoxManage startvm CentOS7 --type headless
```

The `--type headless` option ensures no GUI is brought up.

To force stop the machine, simply do:

```
VBoxManage controlvm CentOS7 poweroff
```

Note that you should resort to the above alternative in case where you cannot access the appliance by any other method (SSH, RDP, etc). Otherwise, you run the risk of damaging it.

5.6 Summary

In this article we have explained how to use `VBoxManage`, VirtualBox's command line alternative for managing appliances. Since this topic would deserve a series of its own, we only covered the most frequently used options. You can refer to Chapter 8 of the VirtualBox documentation guide for a complete list of options and available command line operations.