

Virtual Machines

Erick Gonzalez Parada ID: 178145

February 1, 2025

1 Introduction

This document presents the results of the process of installing a Linux virtual machine with VirtualBox, what was necessary to get web browsing working or create a file, and a comparison between this virtual machine and the native operating system.

2 Methodology

First, we have to choose a Linux operating system. There are several flavors of Linux distributions, most of them aiming to be the best at some task, such as personalization, security, cybersecurity, serving, etc. In this case, I chose openSUSE OS because it is minimalistic, gets the job done, and is very stable and popular in the world of Linux distros.

Next, we need to find and download the ISO file, which can be found on the official webpage of the distro. There are several formats and mirrors available for downloading the image.

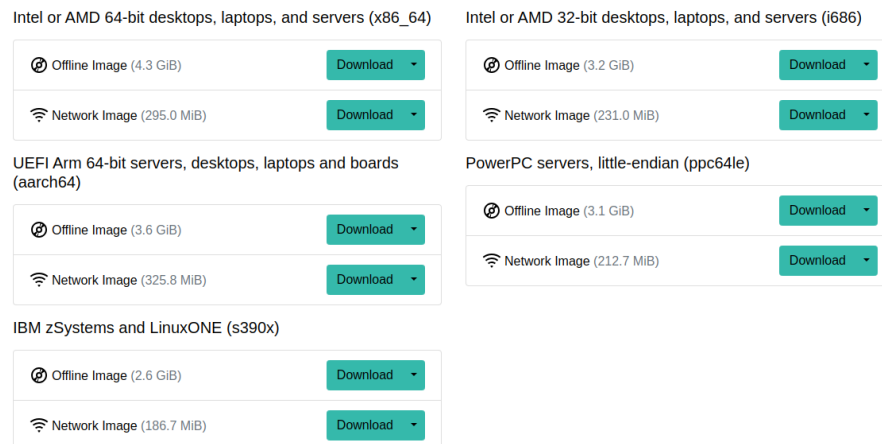


Figure 1: Download Formats

If VirtualBox software is not already installed, it should be downloaded. The app is available for Windows, Mac, and Linux, so one can have Linux on Linux, Windows on Mac, or any combination desired.

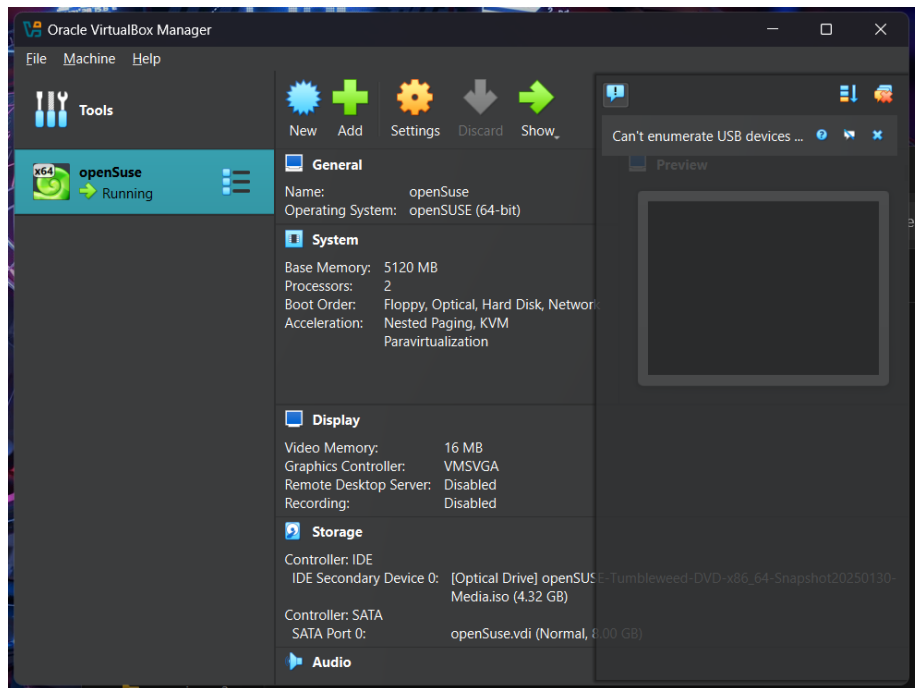


Figure 2: VirtualBox Interface

Pressing the "New" button in the VM (virtual machine) manager allows processing the ISO file and allocating resources to the VM. In this case, I allocated 5 GB of RAM and 2 CPU cores.

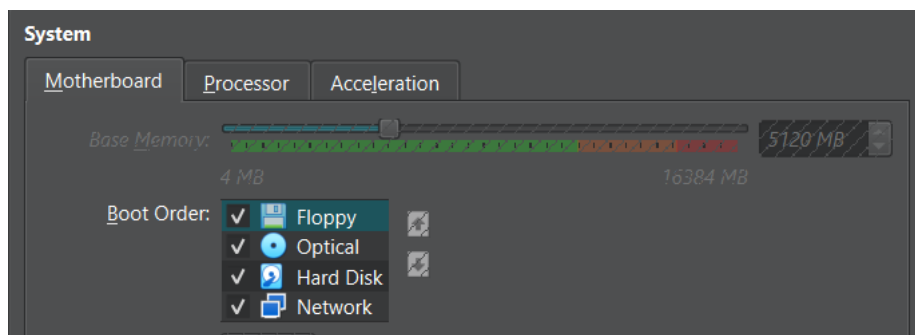


Figure 3: RAM Allocated

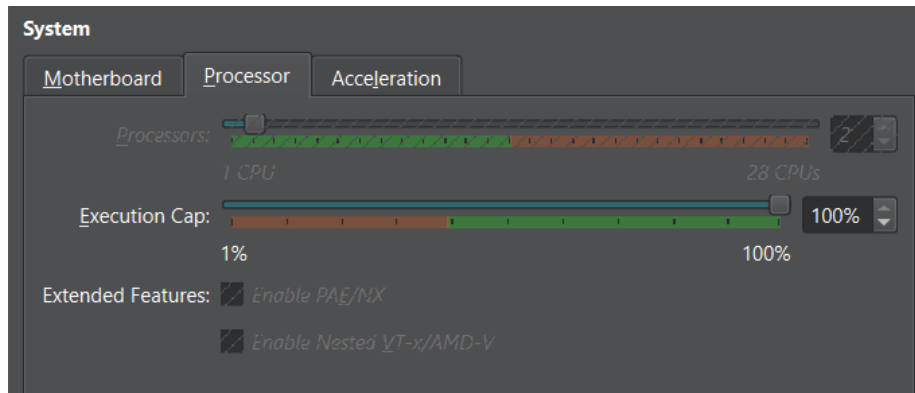


Figure 4: CPUs Allocated

Finally, we can run the VM and start executing basic commands to check Wi-Fi connectivity, update the system, and install software like Chromium.

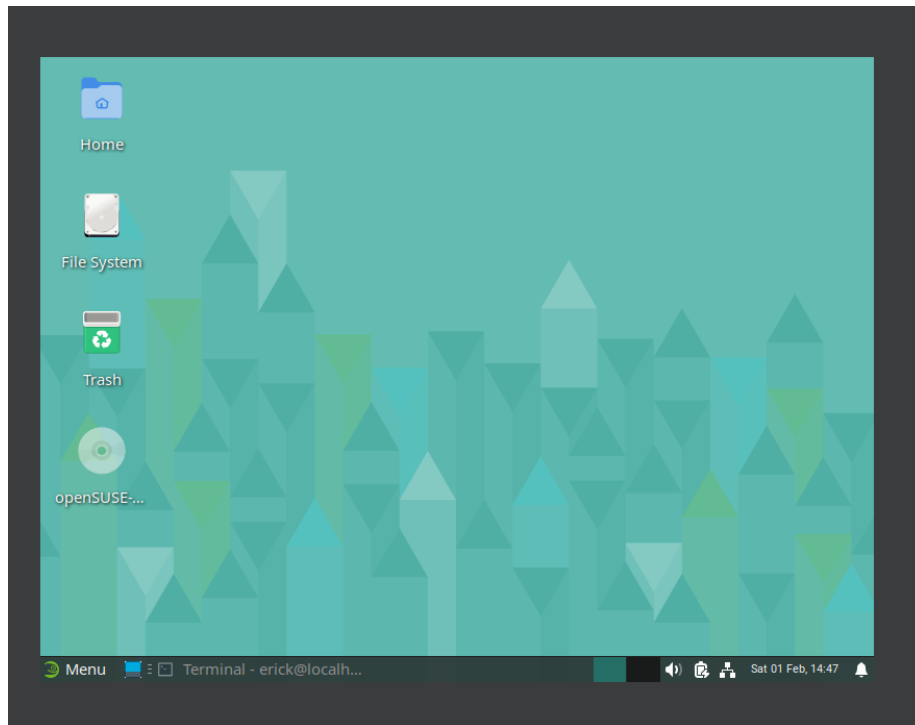
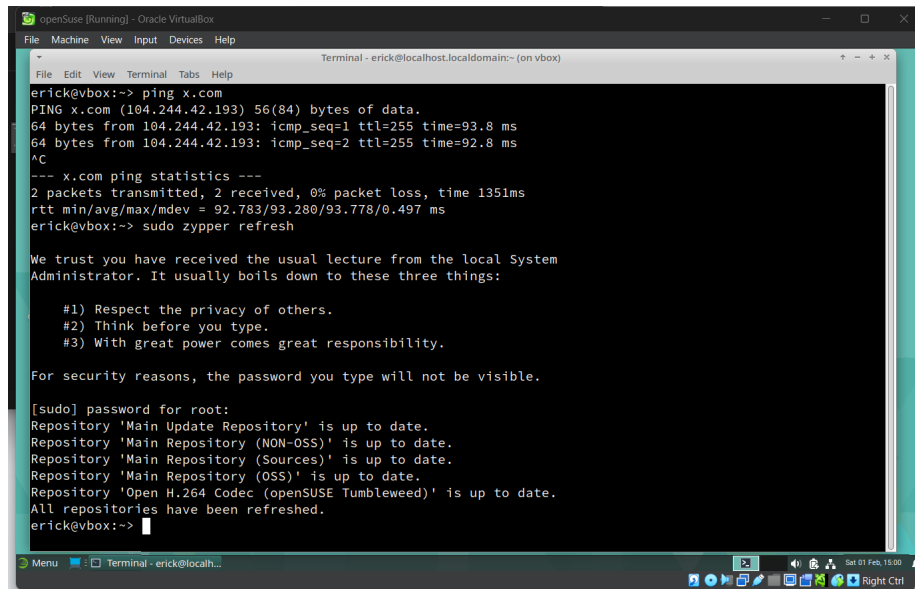


Figure 5: Logged In



```
erick@vbox:~> ping x.com
PING x.com (104.244.42.193) 56(84) bytes of data.
64 bytes from 104.244.42.193: icmp_seq=1 ttl=255 time=93.8 ms
64 bytes from 104.244.42.193: icmp_seq=2 ttl=255 time=92.8 ms
^C
--- x.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1351ms
rtt min/avg/max/mdev = 92.783/93.280/93.778/0.497 ms
erick@vbox:~> sudo zypper refresh

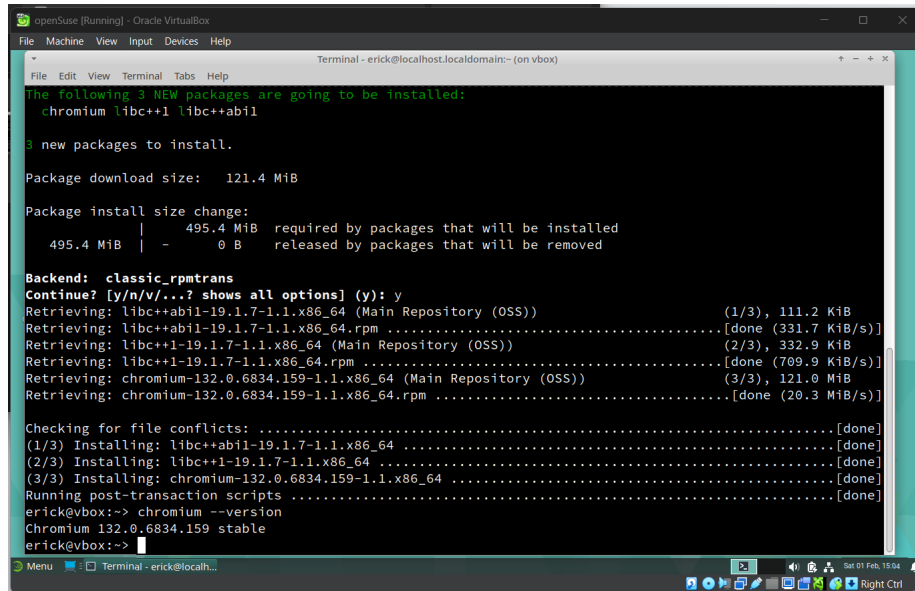
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

For security reasons, the password you type will not be visible.

[sudo] password for root:
Repository 'Main Update Repository' is up to date.
Repository 'Main Repository (NON-OSS)' is up to date.
Repository 'Main Repository (Sources)' is up to date.
Repository 'Main Repository (OSS)' is up to date.
Repository 'Open H.264 Codec (openSUSE Tumbleweed)' is up to date.
All repositories have been refreshed.
erick@vbox:~>
```

Figure 6: Checking Wi-Fi and Updating



```
The following 3 NEW packages are going to be installed:
 chromium libcc++1 libcc++abi1

3 new packages to install.

Package download size: 121.4 MiB

Package install size change:
 495.4 MiB | - 495.4 MiB required by packages that will be installed
           | - 0 B released by packages that will be removed

Backend: classic_rpmtrans
Continue? [y/n/v/...? shows all options] (y): y
Retrieving: libcc++abi1-19.1.7-1.1.x86_64 (Main Repository (OSS)) (1/3), 111.2 KiB
Retrieving: libcc++abi1-19.1.7-1.1.x86_64.rpm .....[done (331.7 KiB/s)]
Retrieving: libcc++1-19.1.7-1.1.x86_64 (Main Repository (OSS)) (2/3), 332.9 KiB
Retrieving: libcc++1-19.1.7-1.1.x86_64.rpm .....[done (789.9 KiB/s)]
Retrieving: chromium-132.0.6834.159-1.1.x86_64 (Main Repository (OSS)) (3/3), 121.0 MiB
Retrieving: chromium-132.0.6834.159-1.1.x86_64.rpm .....[done (20.3 MiB/s)]

Checking for file conflicts: .....[done]
(1/3) Installing: libcc++abi1-19.1.7-1.1.x86_64 .....[done]
(2/3) Installing: libcc++1-19.1.7-1.1.x86_64 .....[done]
(3/3) Installing: chromium-132.0.6834.159-1.1.x86_64 .....[done]
Running post-transaction scripts .....[done]
erick@vbox:~> chromium --version
Chromium 132.0.6834.159 stable
erick@vbox:~>
```

Figure 7: Installing Chromium

3 Comparative Analysis

Table 1: Comparison Between Windows (Native) and openSUSE (VM)

Aspect	Windows (Native)	openSUSE (VM)
Performance	Faster due to direct hardware access	Slightly slower due to virtualization overhead
Resource Usage	Utilizes full system resources	Limited by allocated VM resources (5 GB RAM)
Software Installation	Uses GUI or PowerShell	Relies on terminal commands (e.g., <code>zypper</code>)
Customization	Limited customization options	Highly customizable, tailored to user needs
Stability	Stable for general use	Very stable, designed for reliability
Use Case	Ideal for general-purpose tasks	Great for development, testing, and learning

4 Conclusion

In conclusion, while Windows as a native system offers better performance and ease of use for general tasks, openSUSE in a virtual machine provides a highly customizable and stable environment for development and learning. The choice between the two depends on the user's needs: Windows for everyday use and openSUSE for specialized tasks and experimentation.

5 References

No references were used in this article.