Department of software engineering

OPERATING SYSTEM AND SYSTEM PROGRAMMING

IDIVIDUAL ASSIGNMENT

Parrot OS

Prepared by: Zewetir Antehunegn

Section: B

Introduction

An Operating System is the fundamental software layer that manages computer hardware and software resources, acting as a bridge between the user and the physical machine. It handles essential functions such as memory management, process scheduling, input/output operations, and file systems. Without an OS, users cannot interact with a computer meaningfully or efficiently. Popular examples of operating systems include Windows, macOS, and various distributions of Linux.

For this assignment, I was assigned to work with a popular type of Linux operating system called Parrot OS. It is a Debian-based Linux distribution specifically used for cybersecurity, development, privacy, and digital forensics. It is powerful and has different editions, such as Home Edition (for general use) and Security Edition (packed with penetration testing tools). Parrot OS is known for its clean interface, active community, and built-in tools that support ethical hackers, system administrators, and developers.

To illustrate the installation process and the system call, I used a virtualization tool called Oracle VM VirtualBox. Oracle VirtualBox allows users to run a specific operating system without affecting the host system. It creates a virtual environment where the guest OS (Parrot OS) operates inside the host OS (Windows 10 Pro).

Background History and motivation

Parrot OS, a security-focused Linux distribution that has a history tied to cybersecurity, privacy, and ethical hacking.

Origins and Development (2013)

Parrot OS was created by a group of developers led by Lorenzo "Palinuro" Faletra. The project was first launched in 2013, with a primary focus on providing a secure, privacy-respecting operating system tailored for ethical hackers, penetration testers, and cybersecurity experts. The operating system was built upon Debian, a well-known and stable Linux distribution, recognized for its extensive software sources and security features.
Parrot OS's initial release aimed to offer a more user-friendly environment than existing security-focused Linux distributions like Kali Linux, which at the time was highly useful for the penetration testing community but not designed for general use. Parrot OS sought to strike a balance between offering powerful security tools and providing a practical, everyday usable system.

Security and Privacy Focus

The main objective of Parrot OS was to equip users with a wide range of security tools and utilities useful for penetration testing, digital forensics, cryptography, and privacy. Many of the tools bundled with Parrot OS such as:

- Penetration testing
- Vulnerability assessment
- Wireless network auditing
- Digital forensics and incident response
- Secure communications and encryption

The operating system is designed for professionals, hobbyists, and researchers involved in cybersecurity. It is frequently updated with the latest penetration testing tools and security patches, ensuring users always have access to the best tools in the industry.

Variants of Parrot OS

1. Parrot Security – This is the most well-known edition, designed specifically for security researchers, penetration testers, and ethical hackers. It comes preloaded with a variety of cybersecurity tools for penetration testing, vulnerability analysis, digital forensics, cryptography and so on.

2. Parrot Home – This is a lightweight, more privacy-focused edition of the OS, designed for general users who want a secure and anonymous computing experience. While it doesn't include the additional toolkits provided in security edition.
3. Parrot Architect – is a minimal and flexible edition intended for advanced users who prefer to customize their operating systems. Unlike the other versions, Architect does not come with a desktop environment or pre-installed software. Instead, it provides a basic installation framework, allowing users to choose exactly which packages, tools, and desktop environments they want to include. This makes the user to have a full control over the parrot OS setup.

What, why, and how of virtualization in modern OS

What is Virtualization?

Virtualization in modern operating systems refers to the creation of virtual versions of computing resources, such as hardware platforms, operating systems, storage devices, or network resources. This is can be done through software that enables a physical machine to run multiple virtual machines, each of which acts as an independent computer with its own operating system and applications.

Why is Virtualization Important?

- ✓ Virtualization allows multiple VMs to run on a single physical machine, making better use of hardware resources. This leads to cost savings in terms of hardware, energy consumption, and physical space.
- ✓ Virtual machines (VMs) are isolated from each other, this prevent the other VM to not be harmed when one virtual machine is affected/crushed. This is beneficial for security and stability.
- ✓ Virtualization makes it easy to create, modify, or delete virtual machines, offering flexibility in deploying new applications, testing configurations, and scaling resources.
- ✓ Virtual machines can be used to create test environments without impacting the host system, allowing developers to experiment with different operating systems and configurations safely.
- ✓ Virtual machines have recovery options that allow backed up and restored more easily than physical systems.

How Does Virtualization Work?

1. **Hypervisor (Virtual Machine Monitor)**: The hypervisor is software that manages the virtual machines. There are two types:
   - Type 1 (Bare-metal): Runs directly on the host hardware and manages the VMs without needing a host OS.
   - Type 2 (Hosted): Runs on top of an existing operating system and manages the VMs as applications (e.g., VMware Workstation, Oracle VirtualBox).
2. **VM Creation**: Virtual machines are created with their own virtualized hardware, including CPU, memory, storage, and network adapters. These resources are allocated from the host system.
3. **Guest Operating System**: Each VM runs its own guest operating system, which can be different from the host OS.
4. **Resource Allocation**: The hypervisor allocates and manages system resources like CPU time, RAM, and disk space among the VMs, ensuring the best flow.

Requirements

A. Hardware requirements

- **Operating System**: Windows 10 Pro (host OS)
- **Processor**: Intel Core i5 or equivalent (64-bit capable)
- **Memory (RAM)**: Minimum 4 GB (recommended 8 GB for smoother performance)
- **Storage**: At least 30 GB of free disk space (20 GB allocated to the virtual disk)
- **Virtualization Support**: Hardware virtualization (VT-x/AMD-V) enabled in BIOS
- **Graphics**: Basic support for 2D/3D acceleration (optional)

B. Software requirements

- **Oracle VM VirtualBox**: A virtualization tool used to create and run a virtual environment for Parrot OS. It allows safe experimentation without affecting the host operating system.
- **Parrot OS ISO File**: The official Parrot Home Edition 64-bit ISO, which acts as the bootable installation medium.
- **GCC Compiler (optional)**: Preinstalled or added later in Parrot OS for compiling C/C++ programs used in system call demonstrations.
- **Text Editor (e.g., nano, pluma)**: Used for writing shell scripts and system programming files within Parrot OS.

Filesystem support

Parrot OS supports the ext4 (Fourth Extended Filesystem) as its default because it inherits its base from Debian, one of the most stable and widely used Linux distributions. The ext4 filesystem is deeply integrated into the Linux kernel, making it fully supported and optimized for performance, reliability, and compatibility across Linux environments. One of the most important features of ext4 is journaling, which means it keeps a record of changes before they are made. This helps protect data in case the system crashes or loses power suddenly. Parrot OS is often used for cybersecurity and digital forensics, so having a safe and stable file system like ext4 is very important. It can handle large files and works well on different types of hardware, including virtual machines. Since ext4 is open-source, it also matches Parrot OS's goal of giving users more control and privacy. Overall, ext4 is a good choice for Parrot OS because it is secure, fast, and works well.

INSTALLATION PROCESS

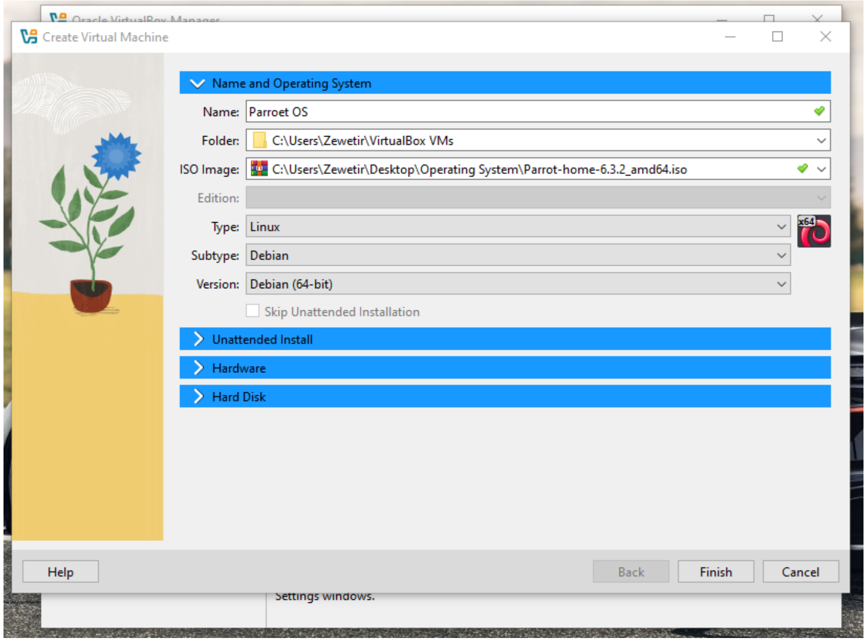
Step 1

prepare required files:

- Download the virtual box from the official website https://www.virtualbox.org
- **Download Parrot OS ISO**: (Home Edition 64-bit) from https://www.parrotsec.org/download/

Step 2

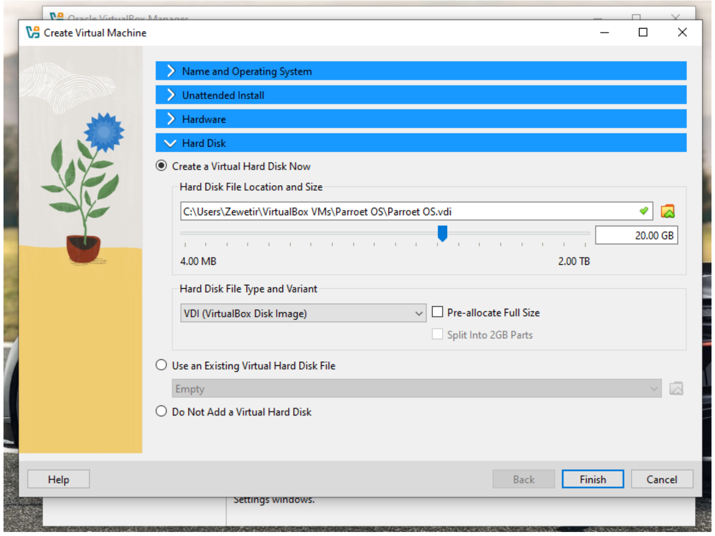
Open oracle VM virtualbox and click new. Then mount/integrate the parrot OS ISO file on the VM and fill out the Name, Type and Version of the OS.



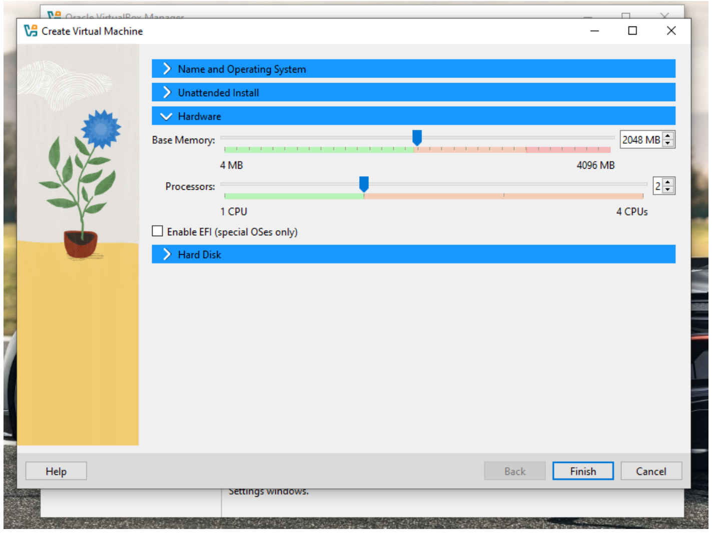
Name: Parroet OS
Folder: C:\Users\Zewetir\VirtualBox VMs
ISO Image: C:\Users\Zewetir\Desktop\Operating System\Parrot-home-6.3.2_amd64.iso
Edition:
Type: Linux
Subtype: Debian
Version: Debian (64-bit)
□ Skip Unattended Installation

Step 3

- Allocate memory (RAM) that the VM going to use.
- Create a virtual hard disk which is hard disk type (virtualbox disc image/VID) and size (min 20GB).



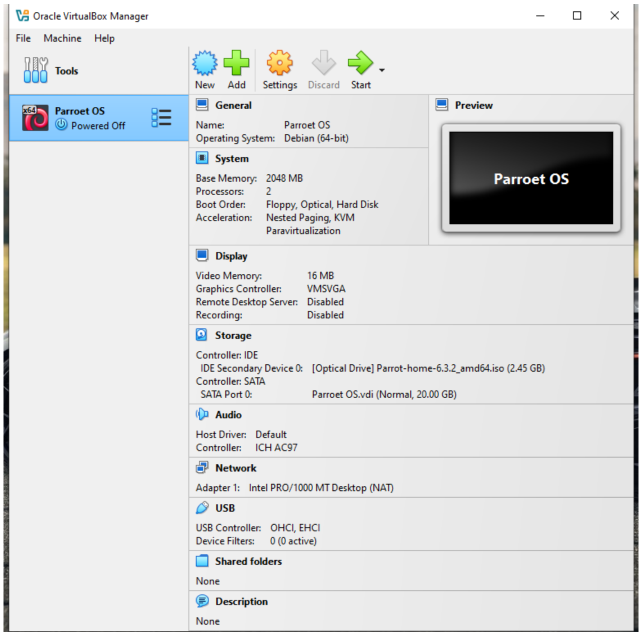
● Create a Virtual Hard Disk Now
Hard Disk File Location and Size
C:\Users\Zewetir\VirtualBox VMs\Parroet OS\Parroet OS.vdi
20.00 GB
Hard Disk File Type and Variant
VDI (VirtualBox Disk Image)
□ Pre-allocate Full Size
□ Split Into 2GB Parts
○ Use an Existing Virtual Hard Disk File
Empty
○ Do Not Add a Virtual Hard Disk



Base Memory: 2048 MB
Processors: 2
□ Enable EFI (special OSes only)

Step 4

- After clicking finish on the previous step, it will show up the summary of setting of what we have allocated.
- This time our operating system is ready to boot.



General

Name: Parroet OS
Operating System: Debian (64-bit)

System

Base Memory: 2048 MB
Processors: 2
Boot Order: Floppy, Optical, Hard Disk
Acceleration: Nested Paging, KVM Paravirtualization

Display

Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage

Controller: IDE
IDE Secondary Device 0: [Optical Drive] Parrot-home-6.3.2_amd64.iso (2.45 GB)
Controller: SATA
SATA Port 0: Parroet OS.vdi (Normal, 20.00 GB)

Audio

Host Driver: Default
Controller: ICH AC97

Network

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

USB

USB Controller: OHCI, EHCI
Device Filters: 0 (0 active)

Shared folders

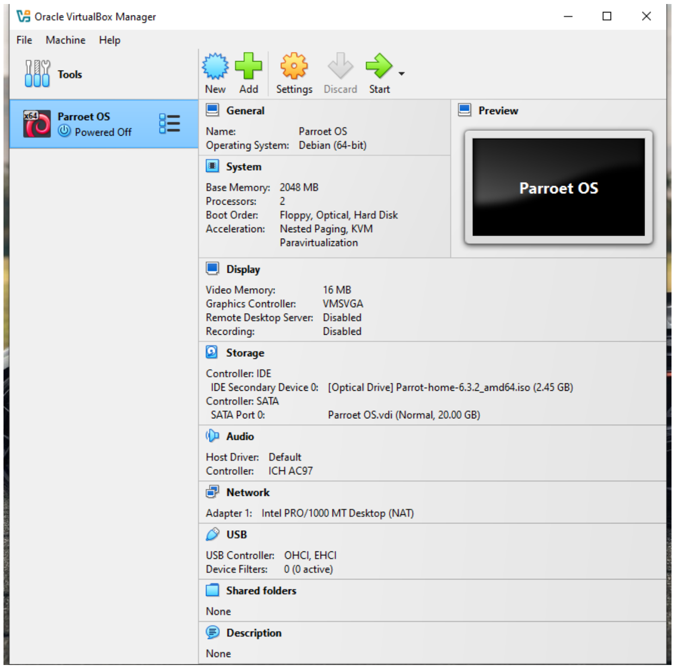
None

Description

None

Step 5

At this stage we click on start (green forward arrow), then the operating system will launch the boot menu. From the boot menu we hit enter on “Try/install” in order to enter the live mode.



General
Name: Parroet OS
Operating System: Debian (64-bit)

System
Base Memory: 2048 MB
Processors: 2
Boot Order: Floppy, Optical, Hard Disk
Acceleration: Nested Paging, KVM Paravirtualization

Display
Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage
Controller: IDE
IDE Secondary Device 0: [Optical Drive] Parrot-home-6.3.2_amd64.iso (2.45 GB)
Controller: SATA
SATA Port 0: Parroet OS.vdi (Normal, 20.00 GB)

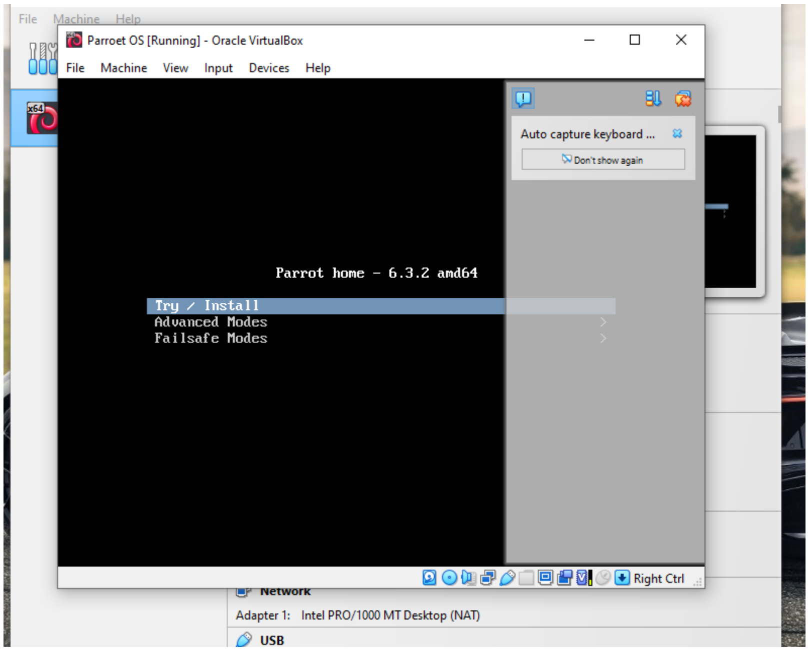
Audio
Host Driver: Default
Controller: ICH AC97

Network
Adapter 1: Intel PRO/1000 MT Desktop (NAT)

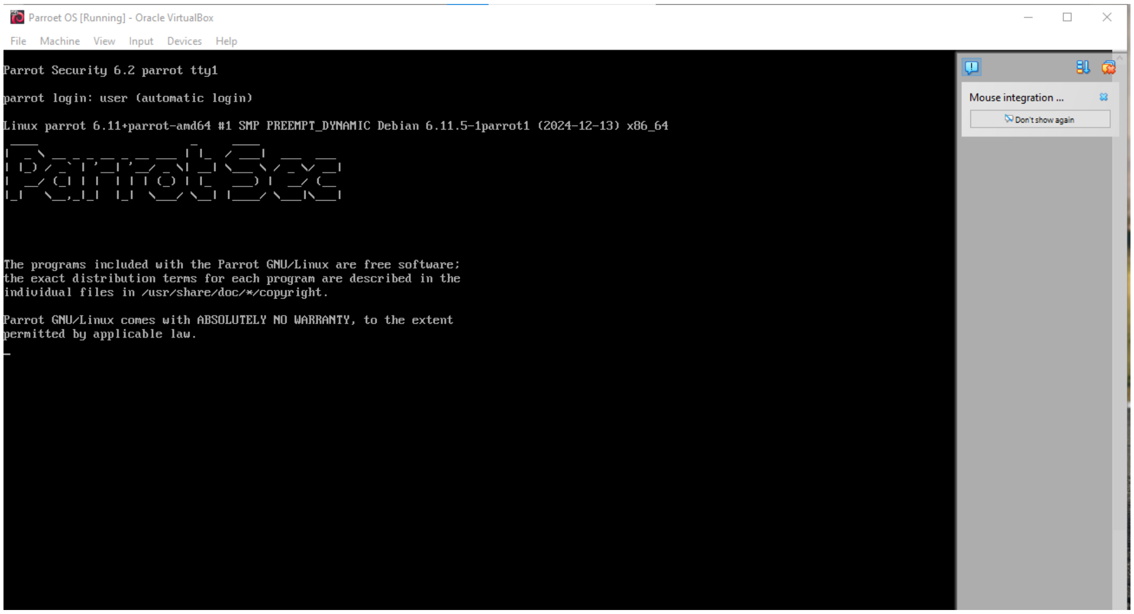
USB
USB Controller: OHCI, EHCI
Device Filters: 0 (0 active)

Shared folders
None

Description
None


File Machine Help
Parroet OS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Auto capture keyboard ...
Don't show again
Right Ctrl
Network
Adapter 1: Intel PRO/1000 MT Desktop (NAT)
USB

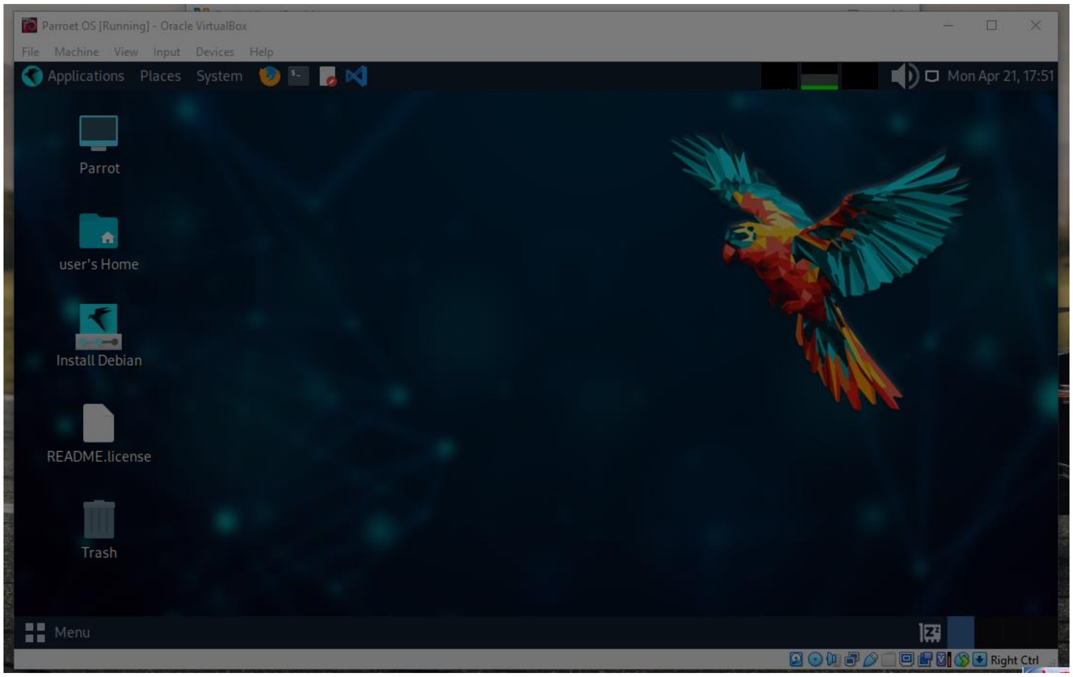
```
Parrot home - 6.3.2 amd64

Try / Install
Advanced Modes
Failsafe Modes
```


Parroet OS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Mouse integration ...
Don't show again

```
Parrot Security 6.2 parrot tty1

parrot login: user (automatic login)

Linux parrot 6.11+parrot-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.11.5-1parrot1 (2024-12-13) x86_64


The programs included with the Parrot GNU/Linux are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Parrot GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
_
```

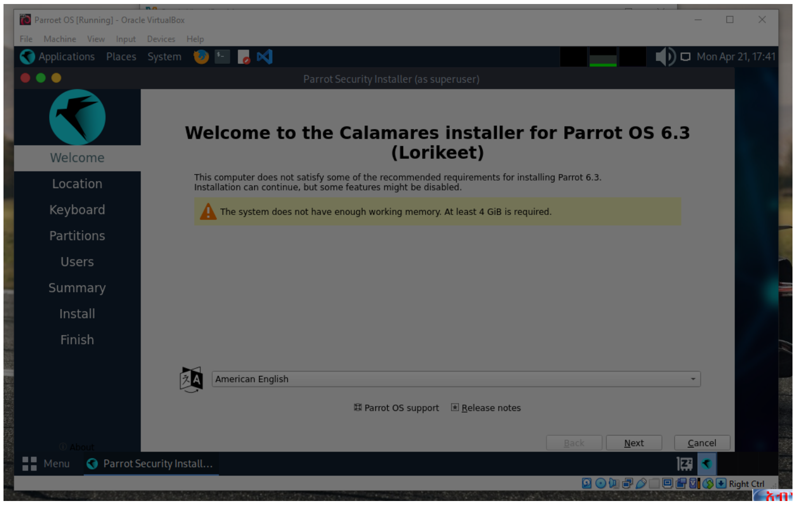
Step 6

After the live mode boots into desktop, double click on “Install Debian”. Then it will start the actual installation process.


Parroet OS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Applications Places System
Mon Apr 21, 17:51
Parrot
user's Home
Install Debian
README.license
Trash
Menu
Right Ctrl

Step 7

- At this stage, it will let us to choose our preferred language, current location and keyboard layout, then hit next
- Disk partitioning: there are two options, Automatic and manual partitioning. If we go through manual partitioning we got two types of table. i.e) MBR & GPT
  - ✓ MBR (master boot record): is the older partitioning standard, compatible with both legacy BIOS and older systems.
  - ✓ GPT (GUID partition table): is the modern standard, and is recommended for most current systems, including Parrot OS running on VirtualBox.
- We will tick on GPT

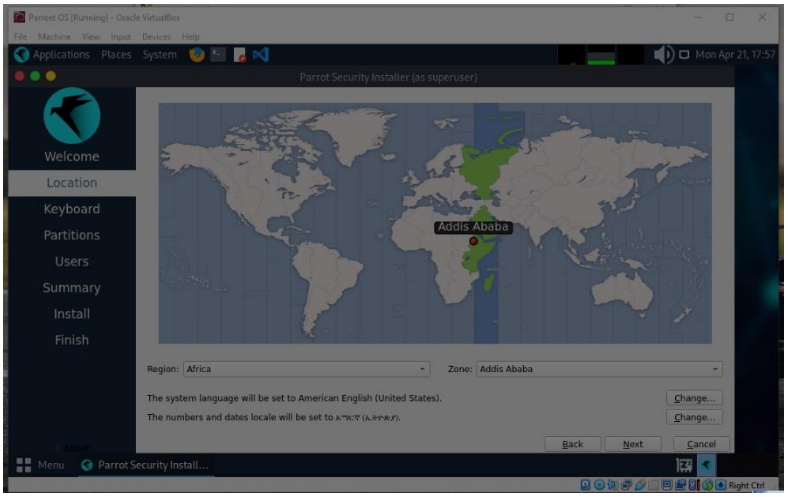

Parrot OS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Applications Places System
Mon Apr 21, 17:41
Parrot Security Installer (as superuser)
Welcome
Location
Keyboard
Partitions
Users
Summary
Install
Finish
Parrot OS support
Release notes
Back
Next
Cancel
Menu
Parrot Security Install...
Right Ctrl

Welcome to the Calamares installer for Parrot OS 6.3 (Lorikeet)

This computer does not satisfy some of the recommended requirements for installing Parrot 6.3.
Installation can continue, but some features might be disabled.

The system does not have enough working memory. At least 4 GiB is required.

American English

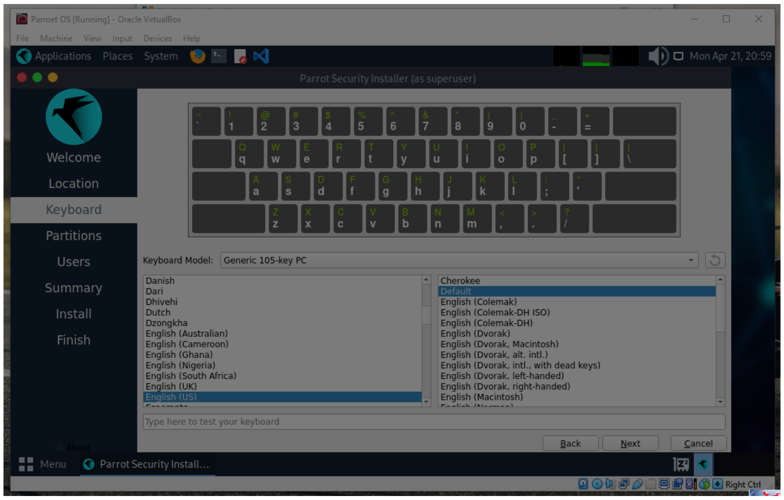

Parrot OS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Applications Places System
Mon Apr 21, 17:57
Parrot Security Installer (as superuser)
Welcome
Location
Keyboard
Partitions
Users
Summary
Install
Finish
Addis Ababa
Change...
Change...
Back
Next
Cancel
Menu
Parrot Security Install...
Right Ctrl

Region: Africa

Zone: Addis Ababa

The system language will be set to American English (United States).

The numbers and dates locale will be set to አማርኛ (ኢትዮጵያ).

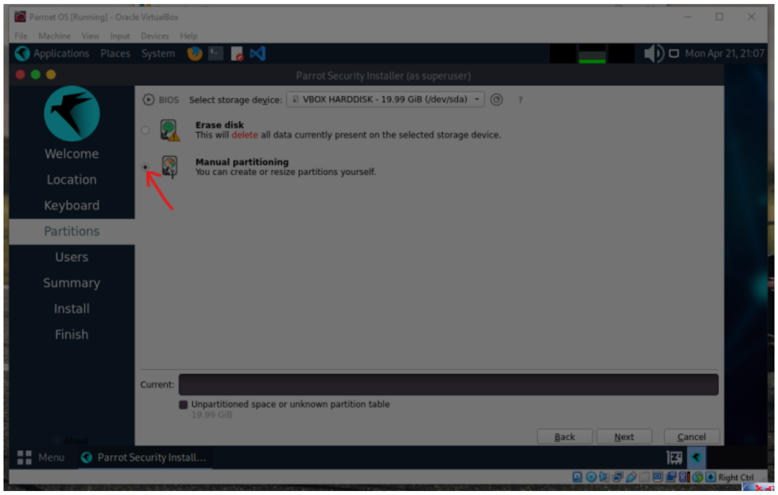

Parrot OS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Applications Places System
Mon Apr 21, 20:59
Parrot Security Installer (as superuser)
Welcome
Location
Keyboard
Partitions
Users
Summary
Install
Finish
Back
Next
Cancel
Menu
Parrot Security Install...
Right Ctrl

Keyboard Model: Generic 105-key PC

Danish
Dari
Dhivehi
Dutch
Dzongkha
English (Australian)
English (Cameroon)
English (Ghana)
English (Nigeria)
English (South Africa)
English (UK)
English (US)

Cherokee
Default
English (Colemak)
English (Colemak-DH ISO)
English (Colemak-DH)
English (Dvorak)
English (Dvorak, Macintosh)
English (Dvorak, alt. intl.)
English (Dvorak, intl., with dead keys)
English (Dvorak, left-handed)
English (Dvorak, right-handed)
English (Macintosh)

Type here to test your keyboard


Parrot OS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Applications Places System
Mon Apr 21, 21:07
Parrot Security Installer (as superuser)
Welcome
Location
Keyboard
Partitions
Users
Summary
Install
Finish
Back
Next
Cancel
Menu
Parrot Security Install...
Right Ctrl

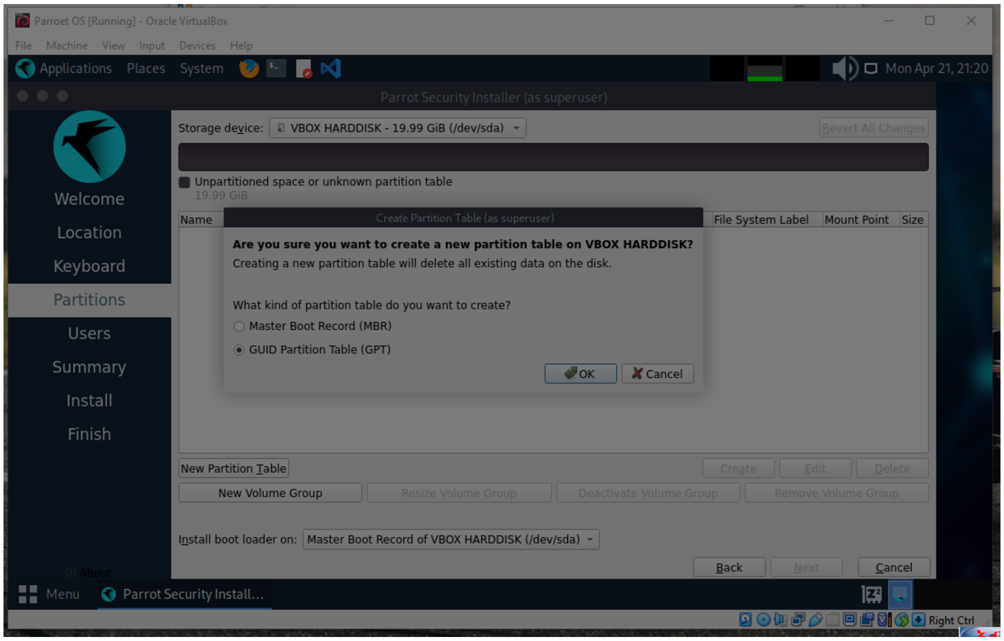
BIOS Select storage device: VBOX HARDDISK - 19.99 GiB (/dev/sda)

○ **Erase disk**
This will delete all data currently present on the selected storage device.

● **Manual partitioning**
You can create or resize partitions yourself.

Current:

Unpartitioned space or unknown partition table
19.99 GiB



Storage device: VBOX HARDDISK - 19.99 GiB (/dev/sda)

Unpartitioned space or unknown partition table
19.99 GiB

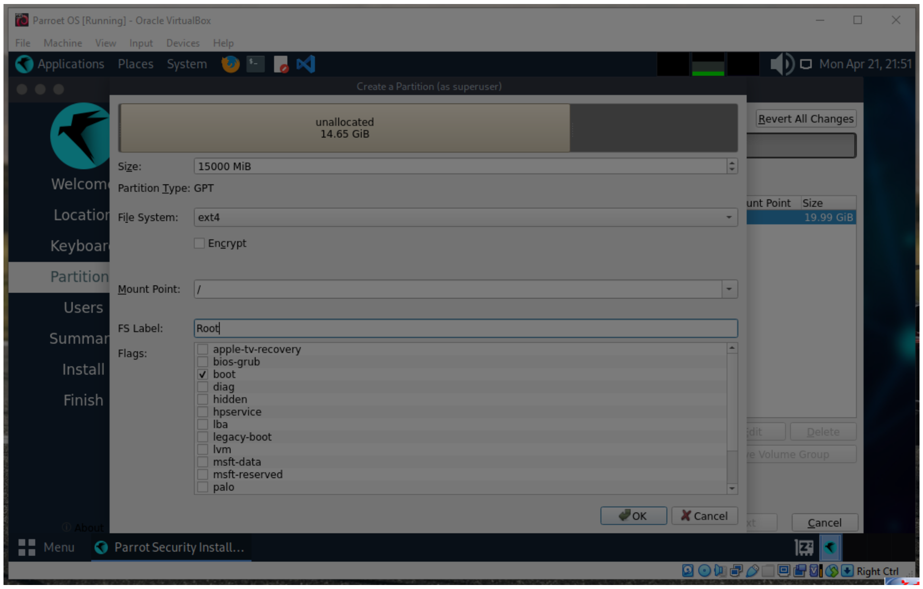
Are you sure you want to create a new partition table on VBOX HARDDISK?
Creating a new partition table will delete all existing data on the disk.

What kind of partition table do you want to create?
○ Master Boot Record (MBR)
● GUID Partition Table (GPT)

Install boot loader on: Master Boot Record of VBOX HARDDISK (/dev/sda)

Step 8

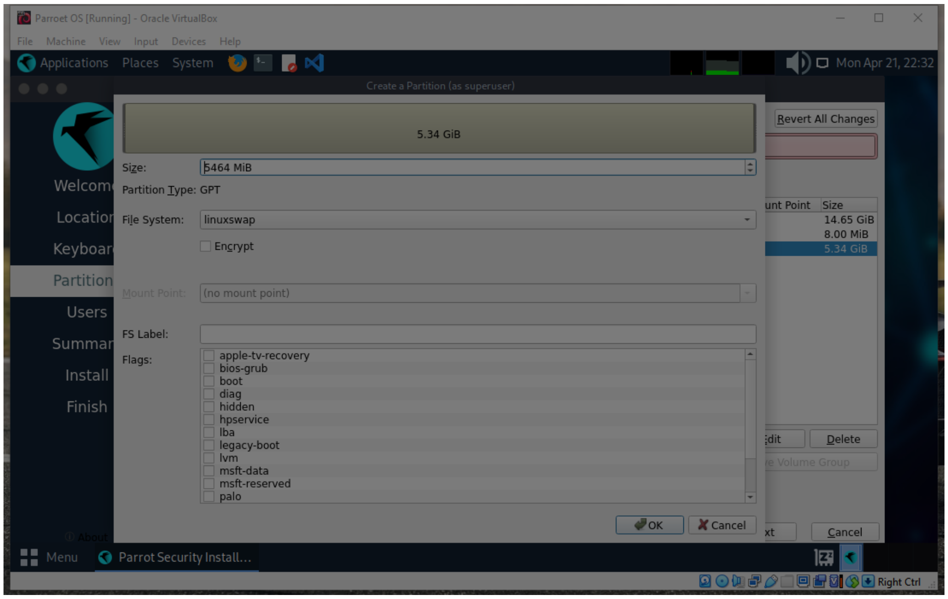
- Create 3 partitions:
  - / → Root (12–15 GB, ext4, boot flag)
  - swap → 2–4 GB (no mount point)
  - /home (optional) → remaining space
- If using GPT, add a small 10 MB bios_grub partition (unformatted with bios_grub flag).



Create a Partition (as superuser)

unallocated
14.65 GiB

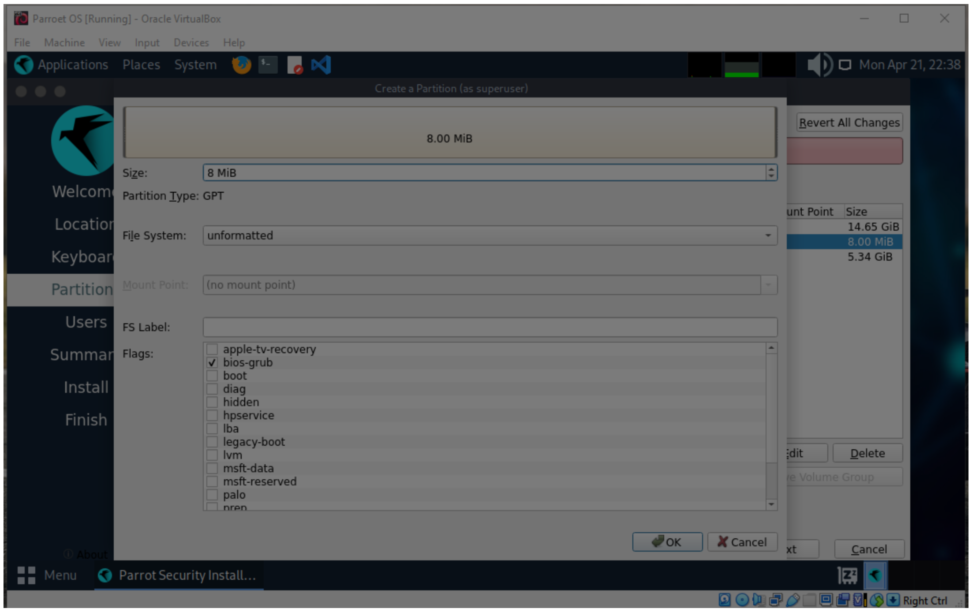
Size: 15000 MiB
Partition Type: GPT
File System: ext4
□ Encrypt
Mount Point: /
FS Label: Root
Flags:
□ apple-tv-recovery
□ bios-grub
☑ boot
□ diag
□ hidden
□ hpservice
□ lba
□ legacy-boot
□ lvm
□ msft-data
□ msft-reserved
□ palo



Create a Partition (as superuser)

5.34 GiB

Size: 5464 MiB
Partition Type: GPT
File System: linuxswap
□ Encrypt
Mount Point: (no mount point)
FS Label:
Flags:
□ apple-tv-recovery
□ bios-grub
□ boot
□ diag
□ hidden
□ hpservice
□ lba
□ legacy-boot
□ lvm
□ msft-data
□ msft-reserved
□ palo



8.00 MiB

Size: 8 MiB

Partition Type: GPT

File System: unformatted

Mount Point: (no mount point)

FS Label:

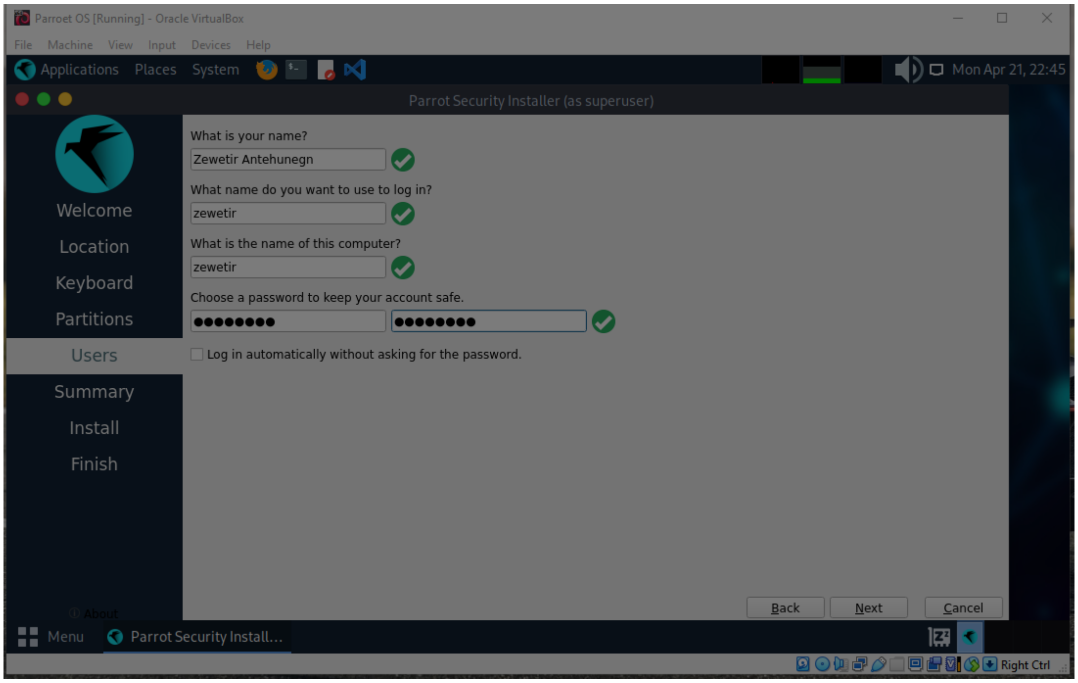
Flags:

- □ apple-tv-recovery
- ☑ bios-grub
- □ boot
- □ diag
- □ hidden
- □ hpservice
- □ lba
- □ legacy-boot
- □ lvm
- □ msft-data
- □ msft-reserved
- □ palo
- □ prep

Step 9

The final stage is:

- setting up user account.
- Reviewing all the information that we have been filling as a summary.
- Then finally install the OS.


Parrot OS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Applications Places System
Mon Apr 21, 22:45
Parrot Security Installer (as superuser)
Welcome
Location
Keyboard
Partitions
Users
Summary
Install
Finish
About

What is your name? Zewetir Antehunegn ✓

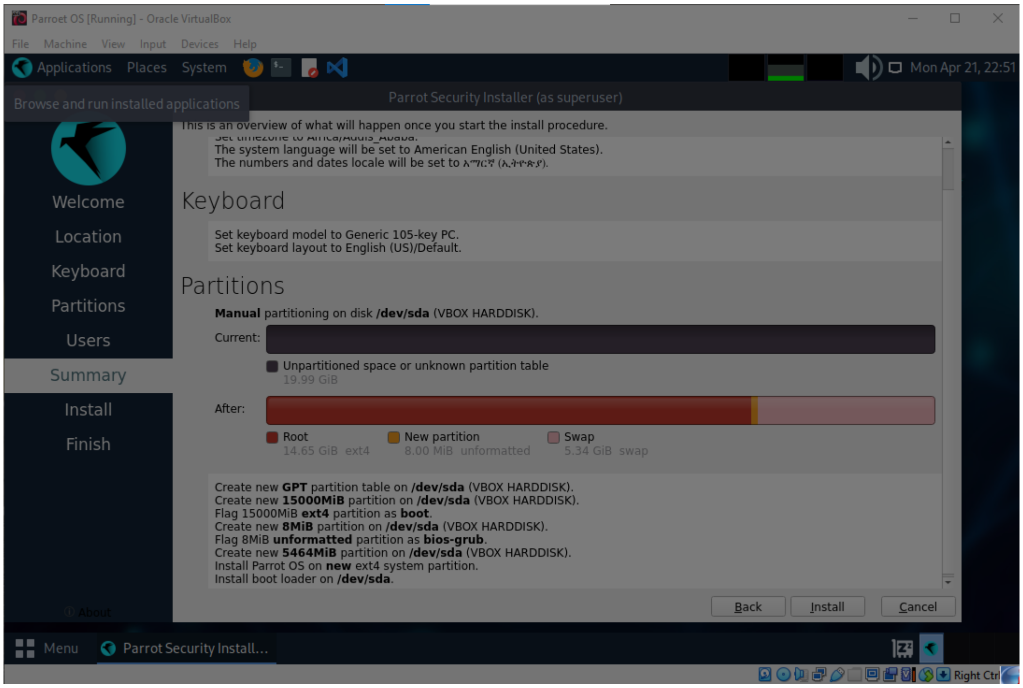
What name do you want to use to log in? zewetir ✓

What is the name of this computer? zewetir ✓

Choose a password to keep your account safe. ●●●●●●●● ●●●●●●●● ✓

□ Log in automatically without asking for the password.

Back
Next
Cancel
Menu
Parrot Security Install...
Right Ctrl


Parroet OS [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Applications Places System
Mon Apr 21, 22:51
Browse and run installed applications
Welcome
Location
Keyboard
Partitions
Users
Summary
Install
Finish
About
Back
Install
Cancel
Menu
Parrot Security Install...
Right Ctrl

Parrot Security Installer (as superuser)

This is an overview of what will happen once you start the install procedure.

Set timezone to Africa/Addis_Ababa.
The system language will be set to American English (United States).
The numbers and dates locale will be set to አማርኛ (ኢትዮጵያ).

Keyboard

Set keyboard model to Generic 105-key PC.
Set keyboard layout to English (US)/Default.

Partitions

Manual partitioning on disk **/dev/sda** (VBOX HARDDISK).

Current:

■ Unpartitioned space or unknown partition table
19.99 GiB

After:

■ Root
14.65 GiB ext4

■ New partition
8.00 MiB unformatted

■ Swap
5.34 GiB swap

Create new **GPT** partition table on **/dev/sda** (VBOX HARDDISK).
Create new **15000MiB** partition on **/dev/sda** (VBOX HARDDISK).
Flag 15000MiB **ext4** partition as **boot**.
Create new **8MiB** partition on **/dev/sda** (VBOX HARDDISK).
Flag 8MiB **unformatted** partition as **bios-grub**.
Create new **5464MiB** partition on **/dev/sda** (VBOX HARDDISK).
Install Parrot OS on **new** ext4 system partition.
Install boot loader on **/dev/sda**.

Challenges I encountered and how I overcame them

While installing and working with Parrot OS on VirtualBox, several challenges were encountered:

1. **Live Mode Only Booting:**
   From the initial the system opens up from live mode and the installation process did not start automatically. So, in that case I thought that the live mode is the actual operating system
2. **rsync failed with code 11 Error:**
   During installation, an error occurred with the rsync code 11 which causes the installation to fail.
3. **Limited System RAM (4 GB):**
   My compute had only 4 GB RAM which make it difficult to allocate enough memory for the virtual machine without affecting system performance.
4. **bios_grub Partition Requirement:**
   When using the GPT partition table during manual partitioning, the installer required an unformatted partition with the bios_grub flag, which may cause an error during the installation process.

To resolve the issues mentioned above, the following actions were taken:

1. After booting into Live Mode, the **"Install Debian"** icon on the desktop was used to begin the full graphical installation manually.
2. The virtual machine's memory was lowered to **2048 MB**, so I balanced the memory usage between host and gust systems to prevent crashes like the rsync error.
3. To fix visual glitches and avoid unnecessary GPU usage, **3D acceleration** was turned off in the VM's display settings.
4. During manual partitioning with GPT, a small (10 MB) unformatted partition with the bios grub flag was added to allow successful installation.