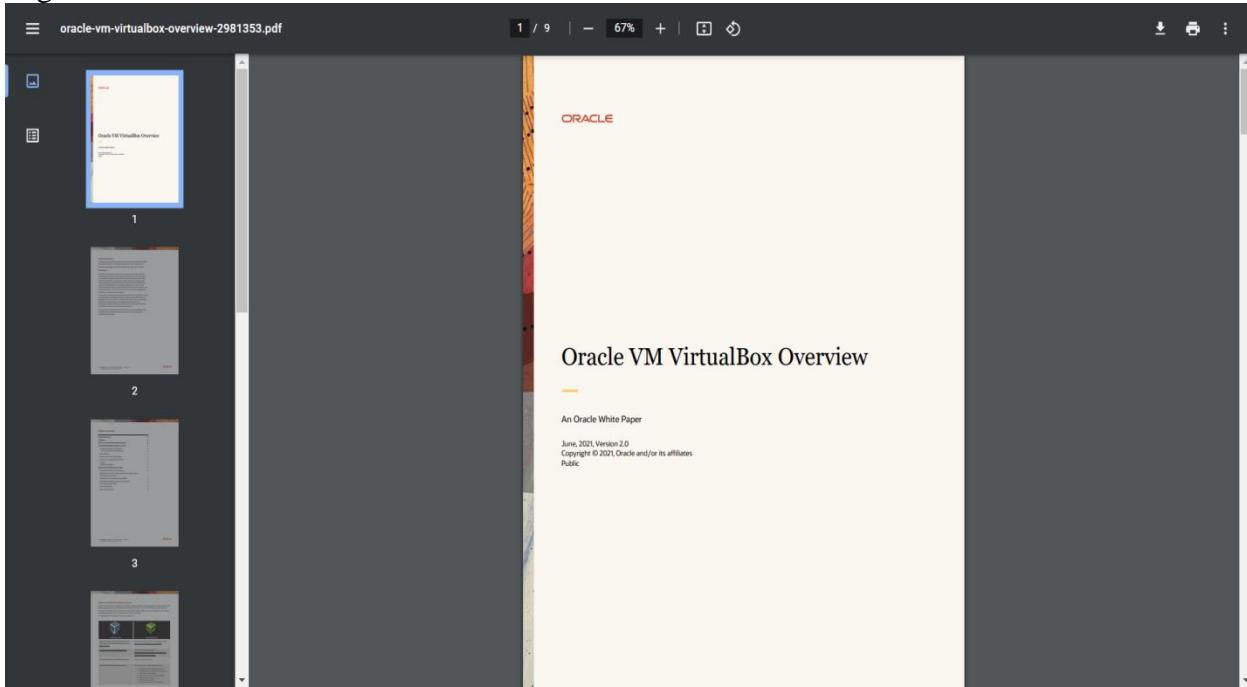
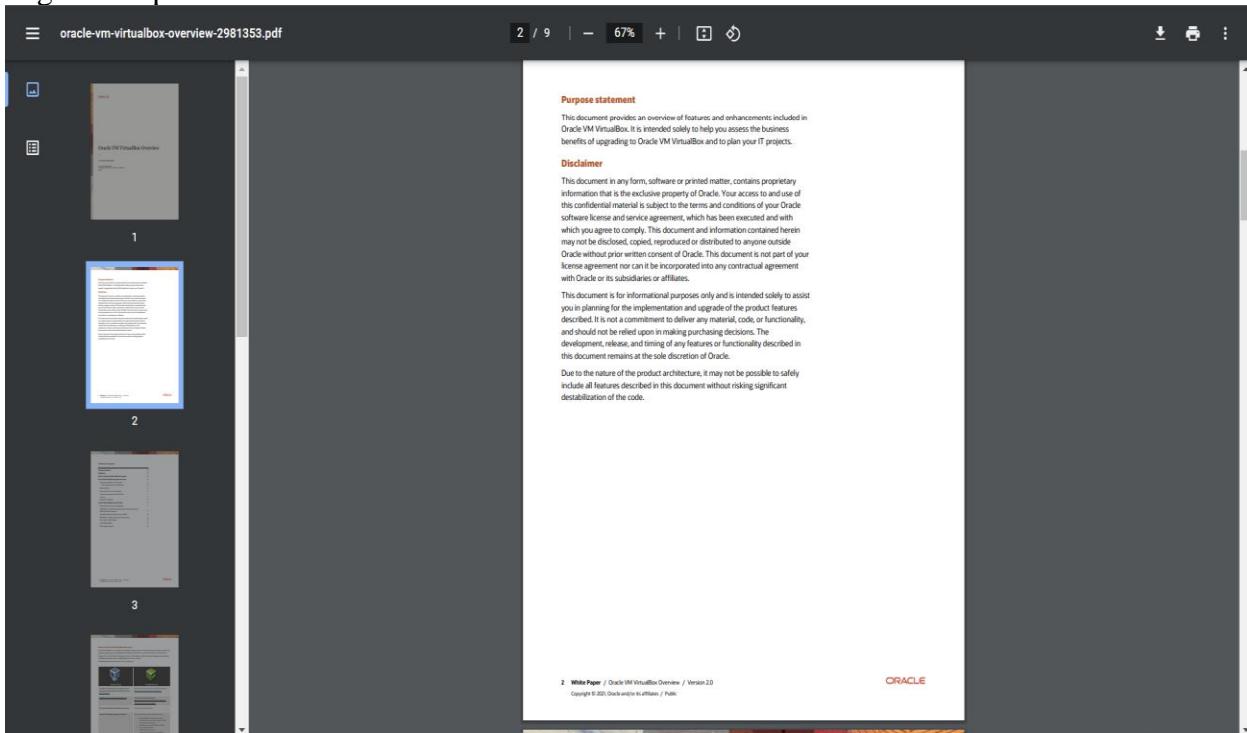


- 1) Setup VM, Linux, and basic testing – must take screen shots at each step to receive points.
  - a. Read Oracle VirtualBox White Paper

## Page 1: Title



## Page 2: Purpose Statement and Disclaimer



## Page 3: Table of contents

The screenshot shows a PDF document titled "oracle-vm-virtualbox-overview-2981353.pdf". The page number at the top is 3 / 9. The main content is the Table of Contents, which includes:

Purpose statement	2
Disclaimer	2
What is in Oracle VM VirtualBox Enterprise	4
Oracle VM VirtualBox Enterprise use cases	5
Development platform for the cloud	5
One unique solution for all platforms	6
QA and testing	6
Demo system for pre-sales support	7
Secure and encrypted virtual machines	7
Training	7
Corporate compliance	7
<b>Oracle VM VirtualBox Extension Pack</b>	<b>7</b>
Oracle Cloud Infrastructure integration	7
USB 2.0/3.0 controller and Enhanced Host Controller Interface (EHCI)/xHCI device support	7
VirtualBox Remote Desktop Protocol (VRDP)	8
NVMe (Non-volatile memory express) emulation	8
Host webcam passthrough	8
Intel PXE boot ROM	8
Disk-image encryption	8

At the bottom of the page, it says "3 White Paper / Oracle VM VirtualBox Overview / Version 1.0 Copyright © 2021 Oracle and/or its affiliates / Public" and "ORACLE".

## Page 4: Definition of Oracle VM VirtualBox Enterprise

The screenshot shows a PDF document titled "oracle-vm-virtualbox-overview-2981353.pdf". The page number at the top is 4 / 9. The main content is the "What is in Oracle VM VirtualBox Enterprise" section, which includes:

**What is in Oracle VM VirtualBox Enterprise**

Oracle VM VirtualBox is cross-platform virtualization software. It allows users to extend their existing computer to run multiple operating systems including Microsoft Windows, Mac OS X, Linux, and Oracle Solaris, at the same time. Designed for IT professionals and developers, Oracle VM VirtualBox is ideal for testing, developing, demonstrating, and deploying solutions across multiple platforms from one machine.

The following table summarizes each of the components:

BASE PACKAGE	EXTENSION PACK
Consists of all open source components and is licensed under the <a href="#">GNU General Public License (GPL) Version 2</a> .	Binaries are released under the Oracle VM VirtualBox Personal Use and Evaluation License (PUEL).
<b>Totally free for personal and business use</b>	<b>A license must be purchased for business/commercial use of the extension pack.</b> The paid for license is perpetual.
Can be distributed and modified by customers	Customers cannot distribute it.
Contains all the basic hypervisor features	The extension pack contains features such as: <ul style="list-style-type: none"><li>Virtual USB 3.0 and 2.0 device support</li><li>VirtualBox Remote Desktop Protocol (VRDP)</li><li>Host webcam passthrough</li><li>Intel Pre boot execution (PXE) boot ROM</li><li>Disk-image encryption</li><li>NVMe Storage emulation</li><li>Oracle Cloud Infrastructure integration</li></ul>

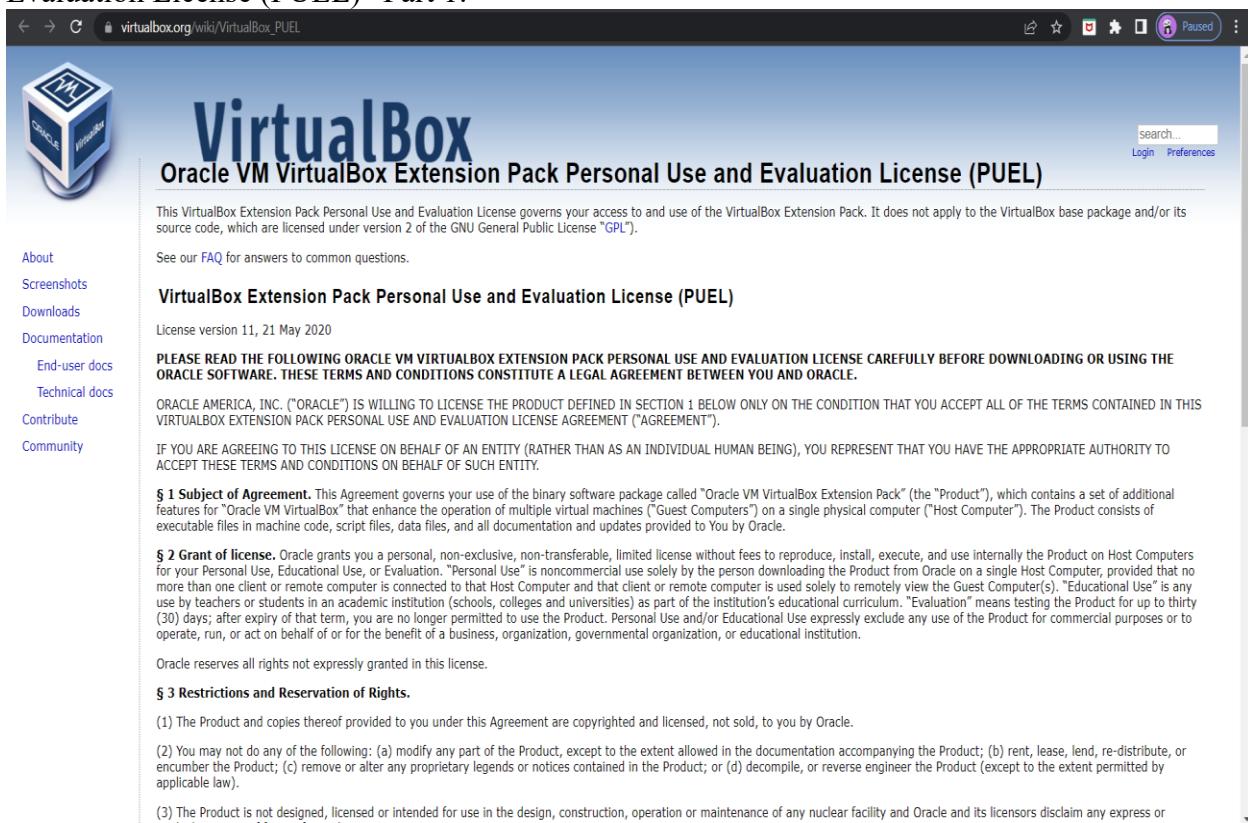
At the bottom of the page, it says "4 White Paper / Oracle VM VirtualBox Overview / Version 1.0 Copyright © 2021 Oracle and/or its affiliates / Public" and "ORACLE".

## Page 4 (external link regarding “The GNU General Public License (GPL) Version 2”:



The screenshot shows a web browser displaying the VirtualBox GPL license page. The URL in the address bar is <https://www.virtualbox.org/wiki/GPL>. The page features the VirtualBox logo and navigation links for About, Screenshots, Downloads, Documentation, End-user docs, Technical docs, Contribute, and Community. The main content discusses the GPL license, with a note that the base package is licensed under GPL V2. It includes links to the full text of the license (<http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>) and the Licensing FAQ. A sidebar lists "See also:" with a link to the Wikipedia page (<http://en.wikipedia.org/wiki/GPL>). The footer contains the ORACLE logo and links to Contact, Privacy policy, and Terms of Use.

## Page 4 (external link regarding “Oracle VM VirtualBox Extension Pack Personal Use and Evaluation License (PUEL)” Part 1:



The screenshot shows a web browser displaying the Oracle VM VirtualBox Extension Pack Personal Use and Evaluation License (PUEL) page. The URL in the address bar is [https://www.virtualbox.org/wiki/VirtualBox\\_PUEL](https://www.virtualbox.org/wiki/VirtualBox_PUEL). The page features the VirtualBox logo and navigation links for About, Screenshots, Downloads, Documentation, End-user docs, Technical docs, Contribute, and Community. The main content is titled "VirtualBox Extension Pack Personal Use and Evaluation License (PUEL)". It states that this license governs access to the Extension Pack and does not apply to the base package. It includes a "FAQ" link for common questions. The "VirtualBox Extension Pack Personal Use and Evaluation License (PUEL)" section details the license terms, noting it applies to version 11, dated 21 May 2020. It requires acceptance of all terms before use. The "§ 1 Subject of Agreement" section describes the Product as a binary software package for multiple virtual machines on one host computer, containing additional features. The "§ 2 Grant of license" section specifies a non-exclusive, limited license for personal, educational, or evaluation use. The "§ 3 Restrictions and Reservation of Rights" section lists prohibited actions, including modification, rental, lending, redistribution, removal of notices, and reverse engineering. The footer contains the ORACLE logo and links to Contact, Privacy policy, and Terms of Use.

## Page 4 (external link regarding “Oracle VM VirtualBox Extension Pack Personal Use and Evaluation License (PUEL)” Part 2:

[virtualbox.org/wiki/VirtualBox\\_PUEL](https://virtualbox.org/wiki/VirtualBox_PUEL)

(3) The Product is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility and Oracle and its licensors disclaim any express or implied warranty of fitness for such uses.

(4) No right, title or interest in or to any trademark, service mark, logo or trade name of Oracle or its licensors is granted under this Agreement.

**§ 4 Termination.** The Agreement is effective on the date you receive the Product and remains effective until terminated. Your rights under this Agreement will terminate immediately without notice from Oracle if you materially breach it or take any action in derogation of Oracle's and/or its licensors' rights to the Product. Oracle may terminate this Agreement immediately should any part of the Product become or in Oracle's reasonable opinion likely to become the subject of a claim of intellectual property infringement or trade secret misappropriation. Upon termination, you will cease use of and destroy all copies of the Product under your control and confirm compliance in writing to Oracle. Neither termination of this Agreement nor any deletion or removal of the Product shall limit any obligations you may have to Oracle, or any rights and/or remedies that Oracle may have with respect to any past or future infringing use of the Product (including but not limited to any use of the Product outside the scope of the license provided in the Agreement). Sections 3-9, inclusive, will survive termination of the Agreement.

**§ 5 Disclaimer of Warranty.** TO THE EXTENT NOT PROHIBITED BY APPLICABLE LAW, ORACLE PROVIDES THE PRODUCT "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED. WITHOUT LIMITING THE FOREGOING, ORACLE SPECIFICALLY DISCLAIMS ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND NON-INFRINGEMENT. The entire risk as to the quality and performance of the Product is with you. Should it prove defective, you assume the cost of all necessary servicing, repair, or correction.

**§ 6 Limitation of Liability.** TO THE EXTENT NOT PROHIBITED BY APPLICABLE LAW, IN NO EVENT WILL ORACLE OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT, DATA, OR DATA USE, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF OR RELATED TO THE USE OF OR INABILITY TO USE THE PRODUCT, EVEN IF ORACLE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. In no event will Oracle's liability to you, whether in contract, tort (including negligence), or otherwise, exceed the amount paid by you for the Product under this Agreement.

**§ 7 Separately Licensed Third Party Technology.** The Product may contain or require the use of third party technology that is provided with the Product. Oracle may provide certain notices to you in the Product's documentation, readmes or notice files in connection with such third party technology. Third party technology will be licensed to you either under the terms of this Agreement or, if specified in the documentation, readmes or notice files, under Separate Terms. Your rights to use Separately Licensed Third Party Technology under Separate Terms are not restricted in any way by this Agreement. However, for clarity, notwithstanding the existence of a notice, third party technology that is not Separately Licensed Third Party Technology shall be deemed part of the Product and is licensed to You under the terms of this Agreement. "Separate Terms" refers to separate license terms that are specified in the Product's documentation, readmes or notice files and that apply to Separately Licensed Third Party Technology. "Separately Licensed Third Party Technology" refers to third party technology that is licensed under Separate Terms and not under the terms of this Agreement.

**§ 8 Export.** Export laws and regulations of the United States and any other relevant local export laws and regulations apply to the Product. You agree that such export laws govern your use of the Product (including technical data) provided under this Agreement, and you agree to comply with all such export laws and regulations (including "deemed export" and "deemed re-export" regulations). You agree that no data, information, and/or Product (or direct product thereof) will be exported, directly or indirectly, in violation of these laws, or will be used for any purpose prohibited by these laws including, without limitation, nuclear, chemical, or biological weapons proliferation, or development of missile technology.

**§ 9 U.S. Government End Users.** Oracle programs, including the Product, any operating system, integrated software, any programs installed on hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

**§ 10 Miscellaneous.** This Agreement is the entire agreement between you and Oracle relating to its subject matter. It supersedes all prior or contemporaneous oral or written communications, proposals, representations and warranties and prevails over any conflicting or additional terms of any quote, order, acknowledgement, or other communication between the parties relating to its subject matter during the term of this Agreement. No modification of this Agreement will be binding, unless in writing and signed by an authorized representative of each party. If any provision of this Agreement is held to be unenforceable, this Agreement will remain in effect with the provision omitted, unless omission would frustrate the intent of the parties, in which case this Agreement will immediately terminate. This Agreement is governed by the laws of the State of California, USA, and you and Oracle agree to submit to the exclusive jurisdiction of, and venue in, the courts of San Francisco or Santa Clara counties in California in any dispute arising out of or relating to this Agreement. Upon 45 days written notice, Oracle may audit your use of the Product to confirm that you are in compliance with the terms of this Agreement. You agree to cooperate with Oracle's audit and provide reasonable assistance and access to information. Any such audit shall not unreasonably interfere with your normal business operations. You agree to pay within 30 days of written notification any fees applicable to your unlicensed use of the Product. You agree that Oracle shall not be responsible for any of your costs incurred in cooperating with the audit. If a legal action or proceeding is commenced by either party in connection with the enforcement of this Agreement, the prevailing party shall be entitled to its costs and attorneys' fees actually incurred in connection with such action or proceeding.

## Page 5: Oracle VM Virtual Box Enterprise use cases

[oracle-vm-virtualbox-overview-2981353.pdf](https://www.oracle.com/virtualization/technologies/virtualbox/enterprise-use-cases.html)

5 / 9 | - 67% + | ☰ 🔍

With thousands of [downloads](#) each day, Oracle VM VirtualBox is the world's most popular free and open source, cross-platform virtualization software, based on vibrant community participation combined with world-class development and support supplied by Oracle.

Oracle VM VirtualBox simplifies cloud deployment by allowing developers to create multiplatform environments and to develop applications for container and virtualization technologies within Oracle VM VirtualBox on a single machine. Operating system and application updates can be done within Oracle VM VirtualBox virtual machines (VMs), and VMs can subsequently be deployed to server virtualization environments such as Oracle Linux KVM or Oracle Private Cloud Appliance.

Oracle VM VirtualBox Enterprise is an ideal choice for a next-generation development solution. The latest release introduces parallelization support for Linux and Windows virtual machines and support for eGDI/USB 3.0 devices and new platforms, and it provides enhanced CPU capabilities and support for bidirectional drag and drop between a host and its guest virtual machines. It also introduces disk-image encryption and many other enhancements.

Oracle VM VirtualBox Enterprise provides world-class support for both the base package and the extension pack and licenses for commercial use of the extension pack.

For further details related to Oracle VM VirtualBox Enterprise please visit:  
<https://www.oracle.com/virtualbox>

**Oracle VM VirtualBox Enterprise use cases**

**Development platform for the cloud**

Software developers rely on Oracle VM VirtualBox Enterprise for the development and debugging of their applications in multiple operating systems and environments on one device. Developers can clone an environment on their personal desktop/laptop without impact to production services.

5 White Paper | Oracle VM VirtualBox Overview | Version 2.0  
Copyright © 2020, Oracle and/or its affiliates. / Public

ORACLE

## Page 6: How Oracle VM VirtualBox acts an unique solution to all platforms along with QA.

oracle-vm-virtualbox-overview-2981353.pdf

6 / 9 | - 67% + | ☰ ⌂

**One unique solution for all platforms**

Oracle VM VirtualBox Enterprise is the only desktop virtualization solution available for x86 operating systems, like Microsoft Windows, Linux, Apple MAC OS X and Solaris x86 that provides [the same solution on all platforms](#).

Oracle VM VirtualBox Enterprise is the desktop virtualization solution that allows software QA teams to control source code, share it within the company and execute software testing on multiple platforms on one unique device.

With Oracle VM VirtualBox Enterprise, VMs can be exported to Oracle Cloud Infrastructure and [all the steps required](#) can be managed through the Graphical User Interface.

By leveraging [Oracle Vagrant Boxes](#), Oracle VM VirtualBox enables the deployment of development environments to be automated.

**QA and testing**

Oracle VM VirtualBox Enterprise allows System Administrators to test patches and system and software upgrades on an isolated sandbox (VM) on a single device, and between other use cases, leverage Oracle VM VirtualBox Enterprise to:

- Recreate customer conditions on a laptop/desktop
  - Need to replicate customer environment easily even on a laptop/desktop
  - Applications could require more than one HW device, due to different platforms/OS
- Test / Experiment sandboxes
  - Preserve customer environments while introducing changes
  - Clone VMs for parallel test runs
  - Revert VMs to a known good state
- Make changes to platform deployments
  - Test different kernel, library, compiler, product installer versions
  - Create demo appliances

6 White Paper / Oracle VM VirtualBox Overview / Version 2.0  
Copyright © 2020, Oracle and/or its affiliates / Public

ORACLE

## Page 7: Features of Oracle VM VirtualBox Extension Pack

oracle-vm-virtualbox-overview-2981353.pdf

7 / 9 | - 67% + | ☰ ⌂

**Demo system for pre-sales support**

Oracle VM VirtualBox Enterprise allows technical sales people to easily show Enterprise solutions in a live demo. With [prebuilt virtual machines](#), sales teams can create, share, present, and demonstrate multi-tier architectures in a complex network topology where the host system interacts with VMs running on top. Engineering teams can also prepare demo environments and share them with sales. It does not matter which platform is used, Oracle VM VirtualBox Enterprise is the same software for all x86 supported platforms.

**Secure and encrypted virtual machines**

In this cloud/social era, where sharing of information is the foundation of IT, VMs created on top of Oracle VM VirtualBox Enterprise could contain confidential information, including software code, or other data that needs the highest security level. Oracle VM VirtualBox Enterprise can encrypt VMs. To copy/clone or move them to external devices, web storage, or cloud backup, built-in encryption can help maintain data security.

**Training**

Oracle VM VirtualBox Enterprise allows the creation of virtual machines for training purposes. In the case of events or training sessions, students can work on enterprise solutions, develop and learn by leveraging VMs running on top of VirtualBox. The same approach can be applied within a company for internal training. Once the training is completed, virtual machines can revert to their original state using Oracle VM VirtualBox Enterprise's snapshot capability.

**Corporate compliance**

Oracle VM VirtualBox Enterprise allows corporate IT to define and maintain a default host platform for different business units, roles, and requirements, with necessary controls and security updates, while each employee can define different virtual machines with different platforms, based on their day-by-day needs.

**Oracle VM VirtualBox Extension Pack**

The following list describes the features provided by Oracle VM VirtualBox Extension Pack:

**Oracle Cloud Infrastructure integration**

Oracle VM VirtualBox provides tight integration with Oracle Cloud Infrastructure (OCI), enabling organizations and developers to more easily and flexibly create applications on premises and deploy to the cloud with a few clicks; further details on these capabilities are available on ["Journey to Oracle Cloud Infrastructure with Oracle VM VirtualBox"](#) paper.

**USB 2.0/3.0 controller and Enhanced Host Controller Interface (EHCI)/xHCI device support**

This option allows users to have USB 2.0/3.0 devices connected to Oracle VM VirtualBox virtual machines. Everything is based on a virtual USB controller that is able to do the following:

- Improve the performance of native USB 2.0 devices on virtual machines
  - By using USB 3.0 virtual USB
  - Obtain similar bare-metal performance for USB 3.0 devices connected to the host

When Oracle VM VirtualBox acts as a virtual Remote Desktop Protocol (RDP) server, it is also possible to use USB devices remotely on RDP clients.

7 White Paper / Oracle VM VirtualBox Overview / Version 2.0  
Copyright © 2020, Oracle and/or its affiliates / Public

ORACLE

## Page 8: Details about Virtual Remote Desktop Protocol, NVMe emulation, Disk-image encryption, etc.

While USB 3.1 support is part of Oracle VM VirtualBox base package, the Extension Pack allows the use of new-generation USB devices that require USB 2.0/3.0.

**VirtualBox Remote Desktop Protocol (VRDP)**  
Oracle VM VirtualBox can display virtual machines remotely, meaning that a virtual machine can execute on one computer even though the virtual machine will be displayed on a second computer. The virtual machine can be controlled from the second computer, as if the virtual machine was running on that computer.

VRDP is a backwards-compatible extension to Microsoft's Remote Desktop Protocol (RDP) and is implemented between the host and its guests. As a result, users can use any standard RDP client to control the remote VM, and any supported guest OS can be used, just like Microsoft Windows.

With this feature, developers can remotely work in their development environment (that is, the same project and the same machine) from nearly anywhere. For example, they can continue to work on their projects from home while connected to a virtual machine that is live on their desktop PC at the office.

**VRDP is a real virtual machine remote console**—able to work on both IPv4 and IPv6—that allows IT administrators to access a virtual machine in cases such as:

- o The virtual machine is starting
- o The virtual machine operating system is not already installed
- o The virtual machine has lost its network connectivity

**NVMe (Non-volatile memory express) emulation**  
This option allows users to have NVMe devices connected to Oracle VM VirtualBox virtual machines. Guest operating systems need to support NVMe devices to make use of them.

**Host webcam passthrough**  
Oracle VM VirtualBox allows a guest to use a host webcam. This complements the general USB passthrough support. If users need to use a webcam for a video conference call, but the software for doing that is not available on the host platform, it's possible to use a webcam on a virtual machine.

**Intel PXE boot ROM**  
Oracle VM VirtualBox allows a guest to use a PXE environment on virtual machines. This means that a user can remotely install a virtual machine (using VRDP) and also supply the operating system packages via network access. Together, the Intel PXE boot ROM feature and the VRDP feature allow users to have installation packages preconfigured and remotely available.

**Disk-image encryption**  
This feature makes it possible to encrypt the data stored in hard disk images transparently to the virtual machine. This provides security similar to encryption software installed on the host system where, usually, a dedicated product license is needed.

8 White Paper / Oracle VM VirtualBox Overview / Version 12.0  
Copyright © 2021 Oracle and/or its affiliates / Public

ORACLE

## Page 9: Contact details

Call +1.800.ORACLE1 or visit oracle.com. Outside North America, find your local office at: oracle.com/contact

[blogs.oracle.com](http://blogs.oracle.com) [facebook.com/oracle](http://facebook.com/oracle) [twitter.com/oracle](http://twitter.com/oracle)

Copyright © 2021, Oracle and/or its affiliates. All rights reserved. This document is provided for informational purposes only, and the contents hereof are subject to change without notice. Oracle and its affiliates are not responsible for typographical or graphical errors that may appear in this document. Oracle and its affiliates disclaim all warranties, express or implied, including warranties of merchantability or fitness for a particular purpose. No part of this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, or otherwise, without prior written permission of Oracle and/or its affiliates. This document is the intellectual property of Oracle and/or its affiliates. Any unauthorized copying, distribution, or storage of this document is illegal. This document is not a contract, although some of the information contained herein may be contractual in nature. This document does not represent, and is not intended to represent, the intent of the parties. The information contained herein is not a commitment to deliver any products or services, and is not an offer to sell or buy any products or services. Further, the development, release, timing, and price of any features or functionality described in this document may change and/or be eliminated without notice. Oracle Corporation.

9 White Paper / Oracle VM VirtualBox Overview / Version 12.0  
Copyright © 2021 Oracle and/or its affiliates / Public

ORACLE

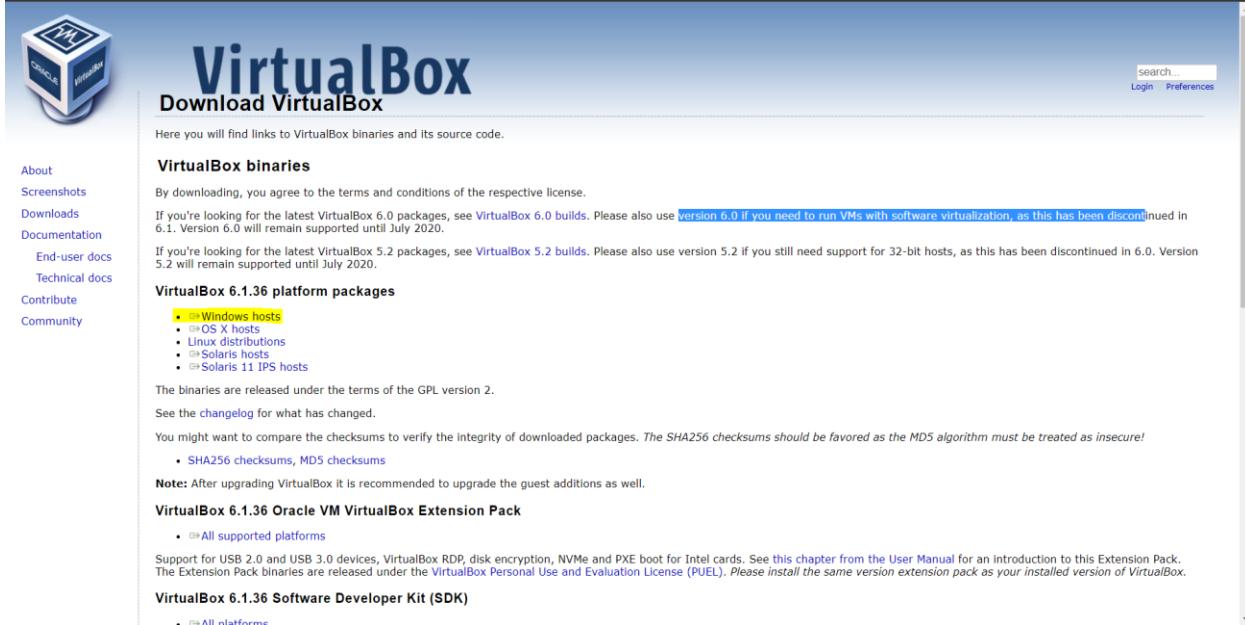
## b. Download Oracle VirtualBox 6.1.36

Step 1: Search VirtualBox on Google or click on the link "<https://www.virtualbox.org/>"



The screenshot shows the official VirtualBox website. At the top right are search, login, and preferences links. On the left, there's a sidebar with links for About, Screenshots, Downloads, Documentation, End-user docs, Technical docs, Contribute, and Community. The main content area features a large blue banner with the text "Download VirtualBox 6.1". Below the banner, there's a section titled "Hot picks:" with three items: "Pre-built virtual machines for developers", "Hyperbox Open-source Virtual Infrastructure Manager", and "phpVirtualBox AJAX web interface". To the right, there's a "News Flash" box containing several news items with dates like July 10th, April 19th, February 24th, January 13th, May 17th, January 18th, and November 22nd, each with a brief description.

Step 2: Click on “Downloads” in the left column to get the below webpage.



The screenshot shows the 'Downloads' page of the VirtualBox website. The left sidebar remains the same as the homepage. The main content area starts with a section titled "VirtualBox binaries" which says "Here you will find links to VirtualBox binaries and its source code." It includes a note about discontinued versions and supported hosts. Below this is a section for "VirtualBox 6.1.36 platform packages" with a list of supported platforms: Windows hosts, OS X hosts, Linux distributions, Solaris hosts, and Solaris 11 IPS hosts. There are also sections for "VirtualBox 6.1.36 Oracle VM VirtualBox Extension Pack" and "VirtualBox 6.1.36 Software Developer Kit (SDK)".

Step 3: As per your operating system, click on the platform package. For windows users (like me), click on Windows hosts and it will automatically down an .exe file as shown in the image below.

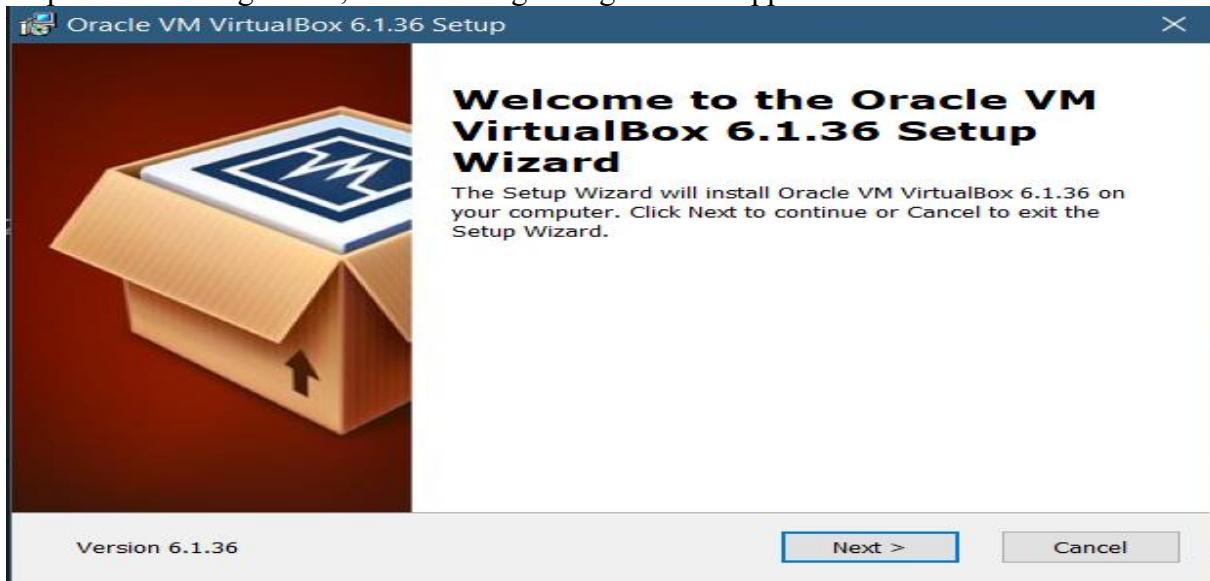


### c. Install VirtualBox

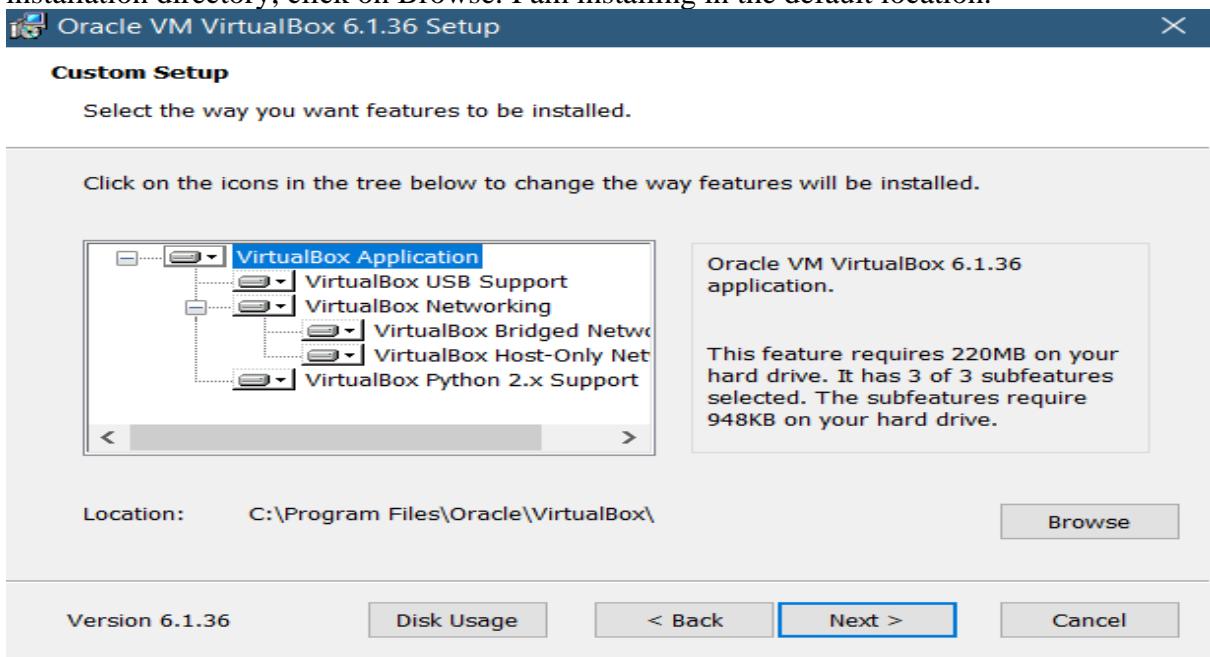
Step 1: Either double click on exe file shown in the above image or go to downloads folder and then double click the exe file.

You will get a dialog box which will prompt to ask if you want to make changes to your computer by installing this exe file. Click “Yes”

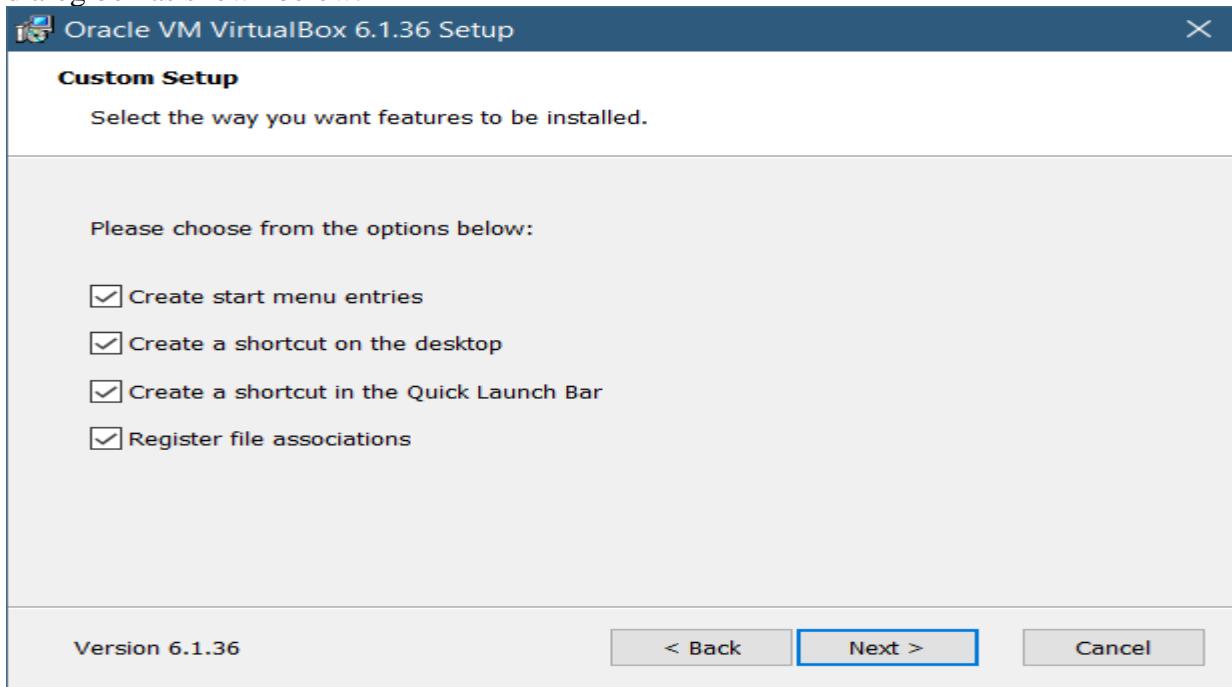
Step 2: On clicking “Yes”, the following dialog box will appear. Click on “Next >”



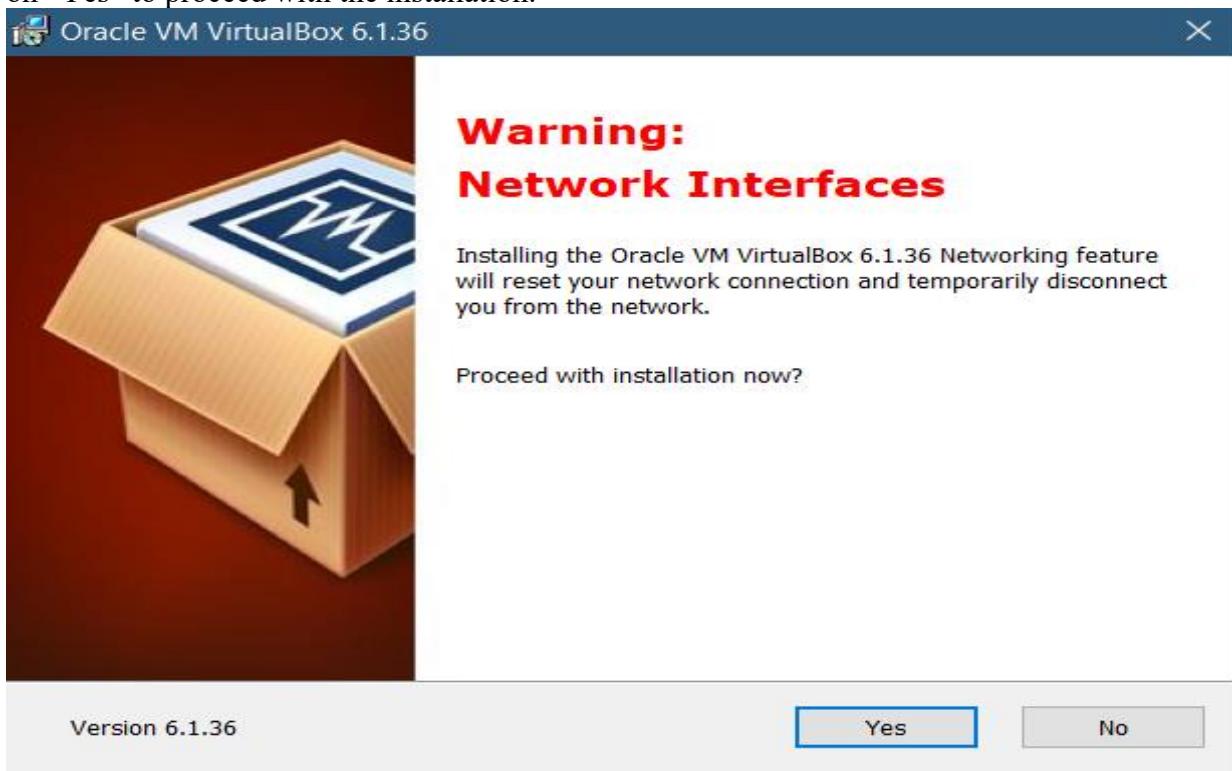
Step 3: Once you click “Next >”, the following dialog box appears which shows the application and default location where the VirtualBox is being installed. In case you want to change the installation directory, click on Browse. I am installing in the default location.



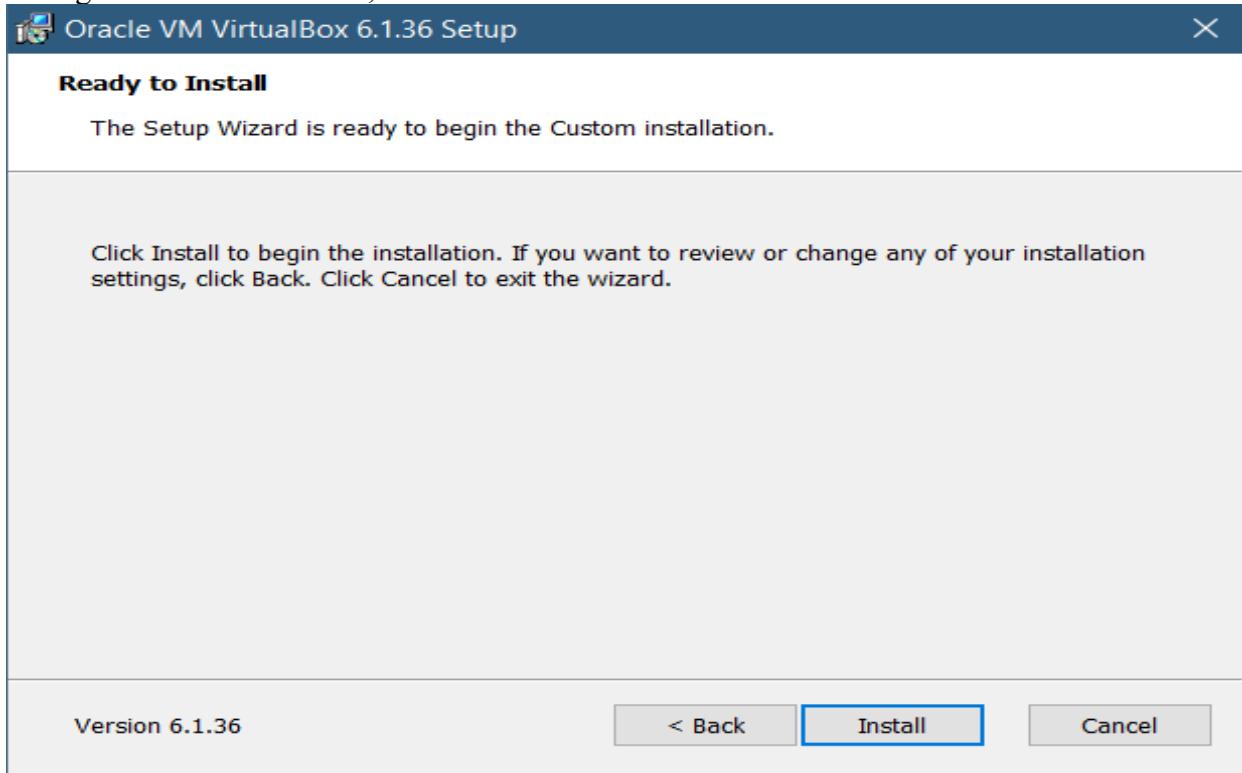
Step 4: Once you decide the path and click “Next >”. You get different set of options in the next dialog box as shown below.



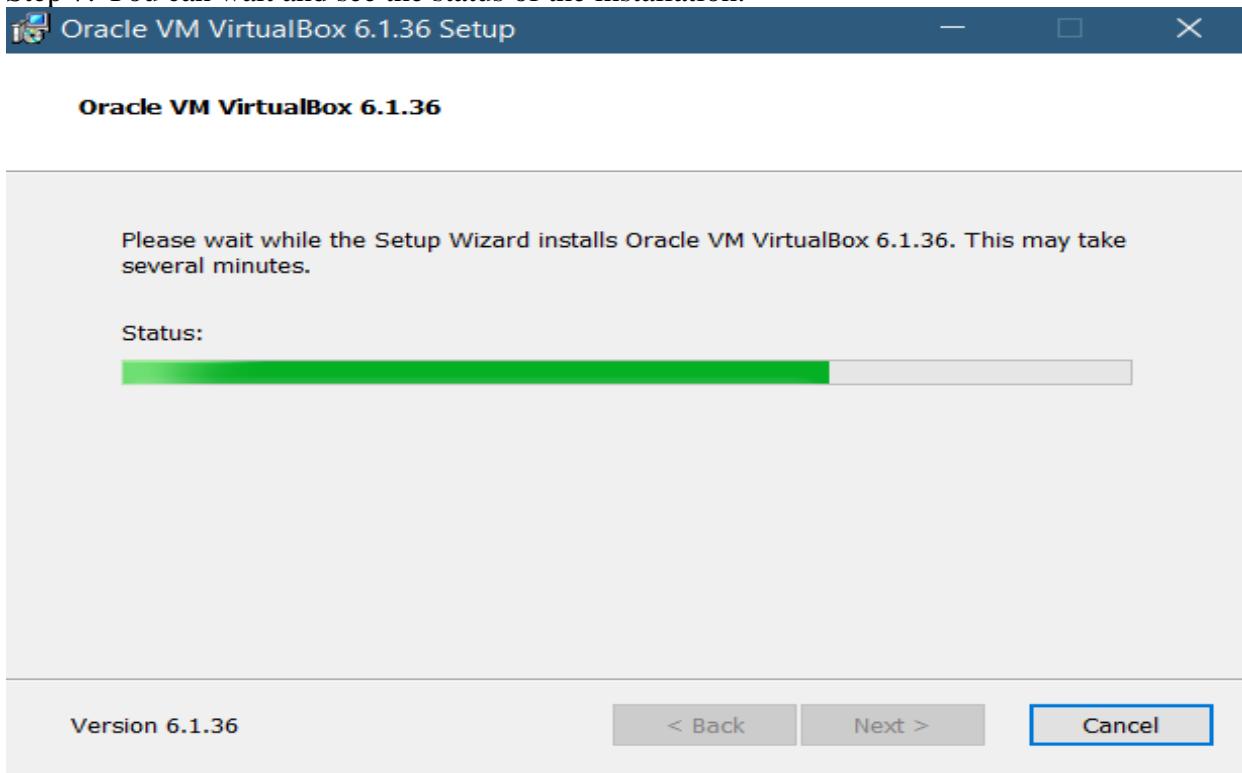
Step 5: After selecting the desired options, click on “Next >”. Post this you will receive a warning dialog box. No need to panic. It just says that in order to install the Virtual Box, your network connection will reset and you will be disconnected temporarily from the network. Click on “Yes” to proceed with the installation.



Step 6: The next dialog box will be a final confirmation. If you don't want to change any of the settings we have done so far, click on "Install".



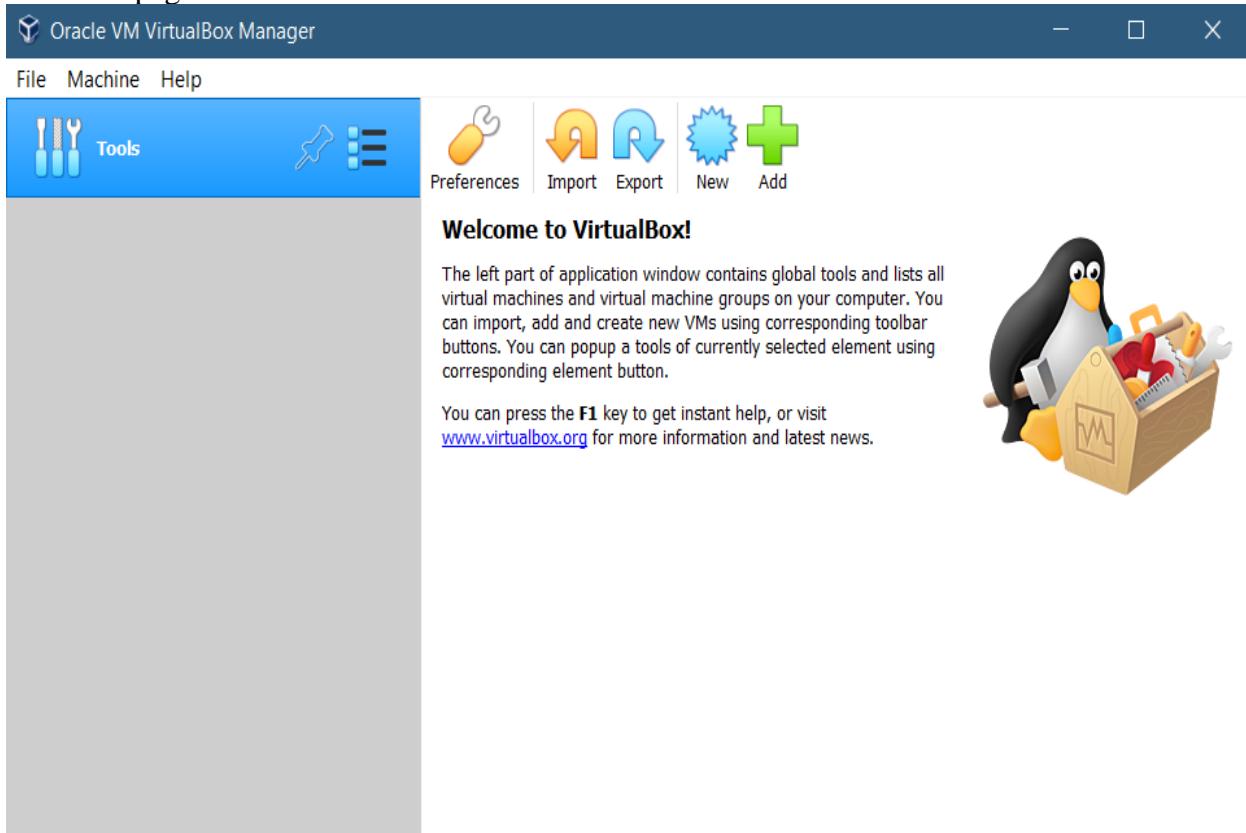
Step 7: You can wait and see the status of the installation.



Step 8: The installation is finished and the final dialog box appears as follows. Click on “Finish”.



The homepage of the Virtual Box looks like this:



d. Download Ubuntu Desktop 22.04.1 LTS Linux (<https://ubuntu.com/download/desktop>) ISO image. (red highlighted in below image). LTS here means Long Term Support.

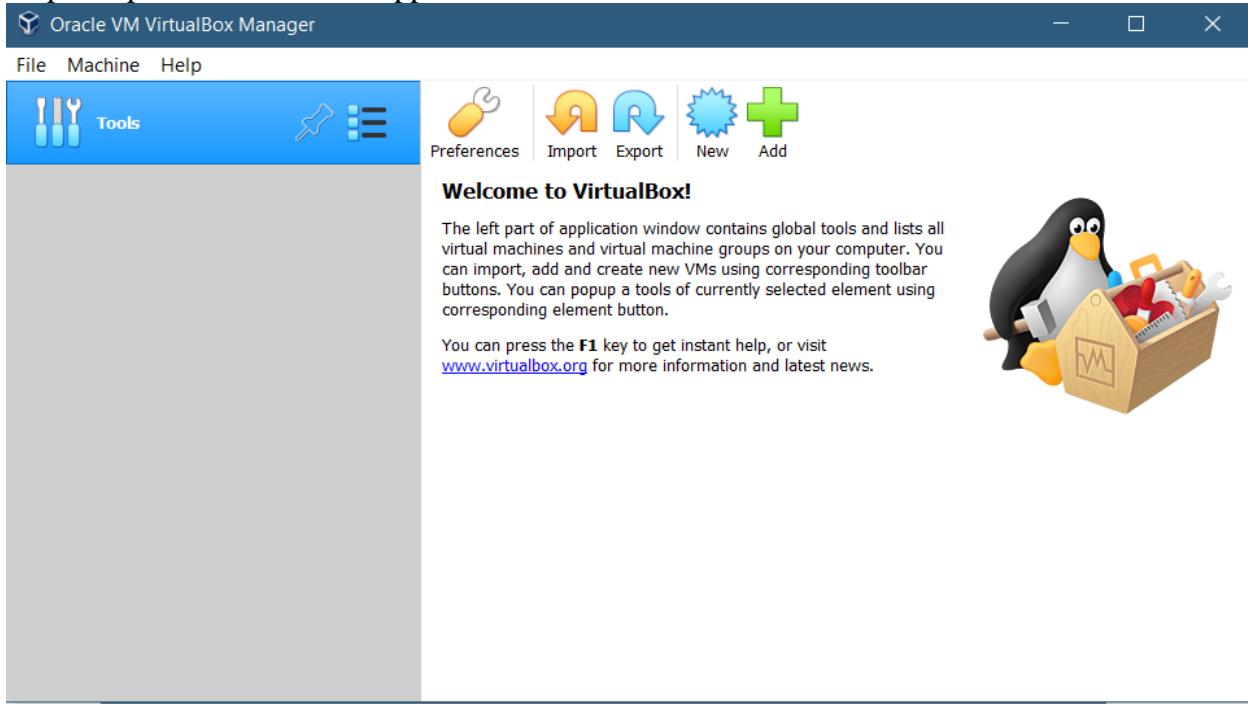
The screenshot shows the Ubuntu download page. At the top, there's a navigation bar with links for Enterprise, Developer, Community, Download, and a search bar. Below the navigation is a secondary menu with options for Desktop, Server, IoT, and Cloud. A red box highlights the text "Ubuntu 22.04.1 LTS". To the right of this, another red box highlights the large green "Download" button. Below the main heading, there's a brief description of what LTS means: "Download the latest LTS version of Ubuntu, for desktop PCs and laptops. LTS stands for long-term support — which means five years, until April 2027, of free security and maintenance updates, guaranteed." At the bottom of the page, there's a link for "For other versions of Ubuntu Desktop including".

As soon as we click the download button, an automatic download will start and the following webpage will be displayed. Since, the iso file is big (around 3.5 GB), it will take time to download.

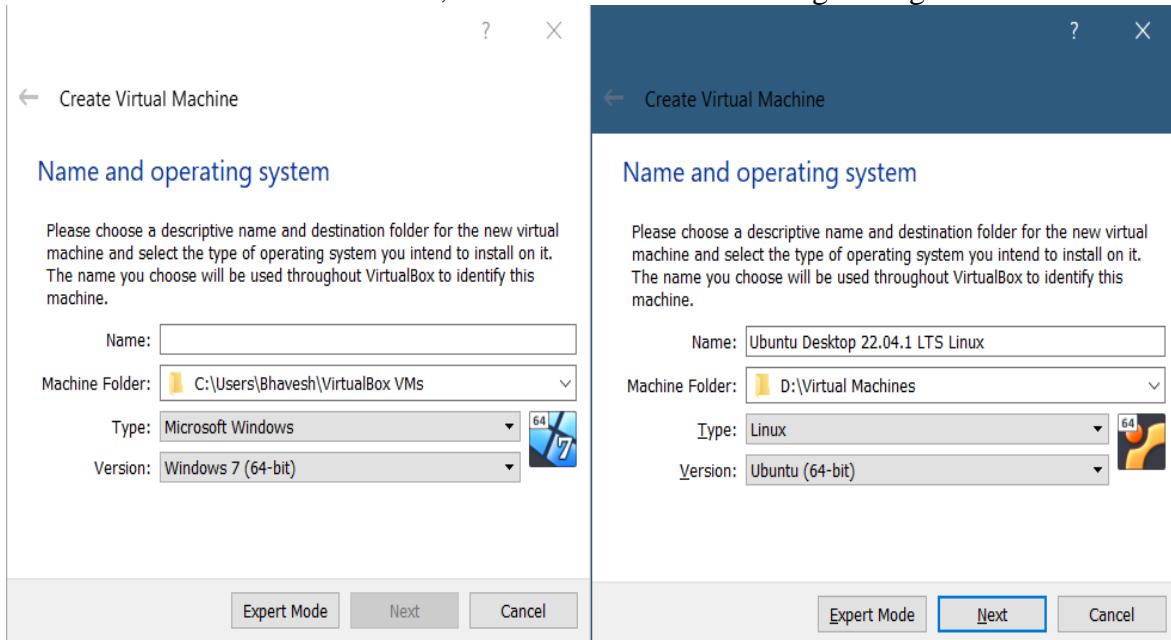
The screenshot shows a confirmation page titled "Thank you for downloading Ubuntu Desktop". It includes instructions: "Your download should start automatically. If it doesn't, [download now](#). You can [verify your download](#), or get [help on installing](#)". Below this, there's a section titled "Getting started with Ubuntu Desktop" with two links: "Install Ubuntu desktop" and "How to run Ubuntu on a virtual machine". A note under "Install Ubuntu desktop" says "How to install Ubuntu Desktop on your laptop or PC." Under "How to run Ubuntu on a virtual machine", it says "Run Ubuntu Desktop in a virtual machine using VirtualBox. A quick start guide that will work across any operating system."

e. Create Virtual Machine (VM), to support Linux, Ubuntu, 64-bit, 4GB RAM, Virtual Disk 25GB, VDI image, dynamically allocated, 2-core, and a network interface (1GbE or WiFi) with NAT support.

Step 1: Open the VirtualBox application which looks as follows:



Step 2: Once we have downloaded the Ubuntu OS iso image, we will use the “New” button to add and start the setup of a virtual machine. By default it looks like the below left image. After we add some details about the OS, it will look like the below right image.



### Step 3: Providing 4GB RAM and creating a new virtual hard drive.

**Memory size**

Select the amount of memory (RAM) in megabytes to be allocated to the virtual machine.

The recommended memory size is **1024 MB**.

4 MB      4096 MB      16384 MB

**Hard disk**

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select one from the list or from another location using the folder icon.

If you need a more complex storage set-up you can skip this step and make the changes to the machine settings once the machine is created.

The recommended size of the hard disk is **10.00 GB**.

Do not add a virtual hard disk  
 Create a virtual hard disk now  
 Use an existing virtual hard disk file

Empty

Next      Cancel      Create      Cancel

### Step 4: VDI Image, dynamically allocated, 25 GB.

**Hard disk file type**

Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.

VDI (VirtualBox Disk Image)  
 VHD (Virtual Hard Disk)  
 VMDK (Virtual Machine Disk)

**Storage on physical hard disk**

Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).

A **dynamically allocated** hard disk file will only use space on your physical hard disk as it fills up (up to a maximum **fixed size**), although it will not shrink again automatically when space on it is freed.

A **fixed size** hard disk file may take longer to create on some systems but is often faster to use.

Dynamically allocated  
 Fixed size

**File location and size**

Please type the name of the new virtual hard disk file into the box below or click on the folder icon to select a different folder to create the file.

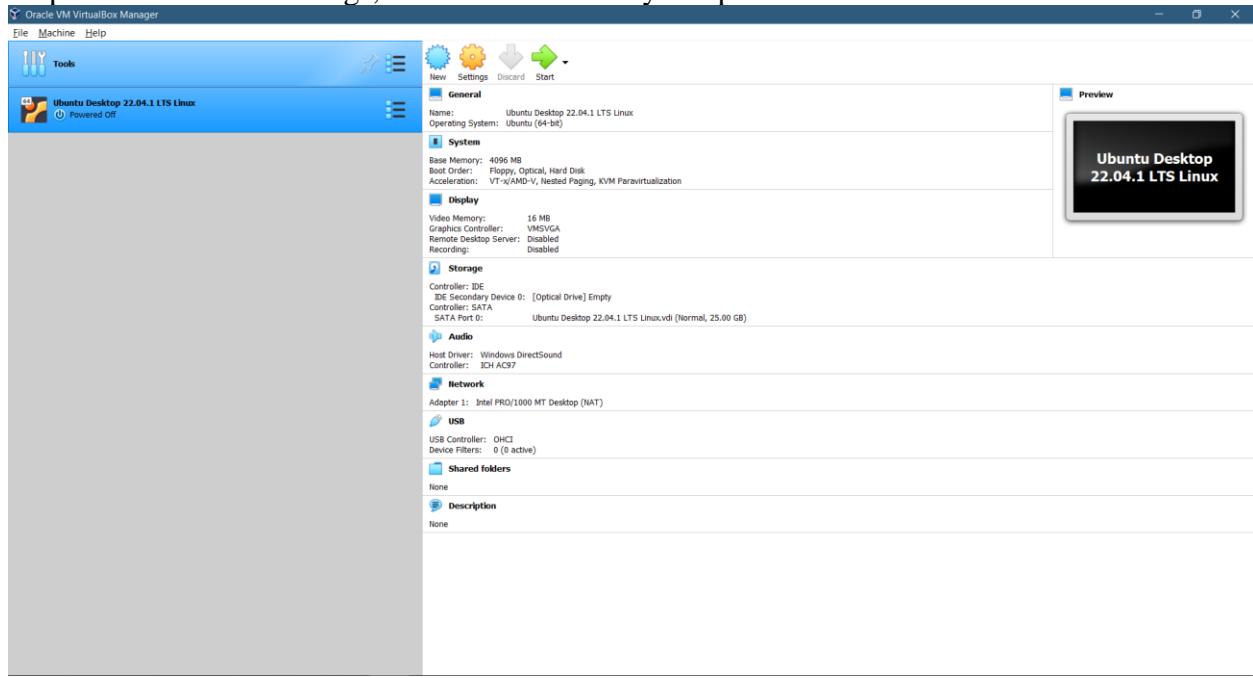
(chines)Ubuntu Desktop 22.04.1 LTS Linux\Ubuntu Desktop 22.04.1 LTS Linux

Select the size of the virtual hard disk in megabytes. This size is the limit on the amount of file data that a virtual machine will be able to store on the hard disk.

4.00 MB      25.00 GB      2.00 TB

Create      Cancel

Step 5: With default settings, we have successfully setup a virtual machine which looks like this.

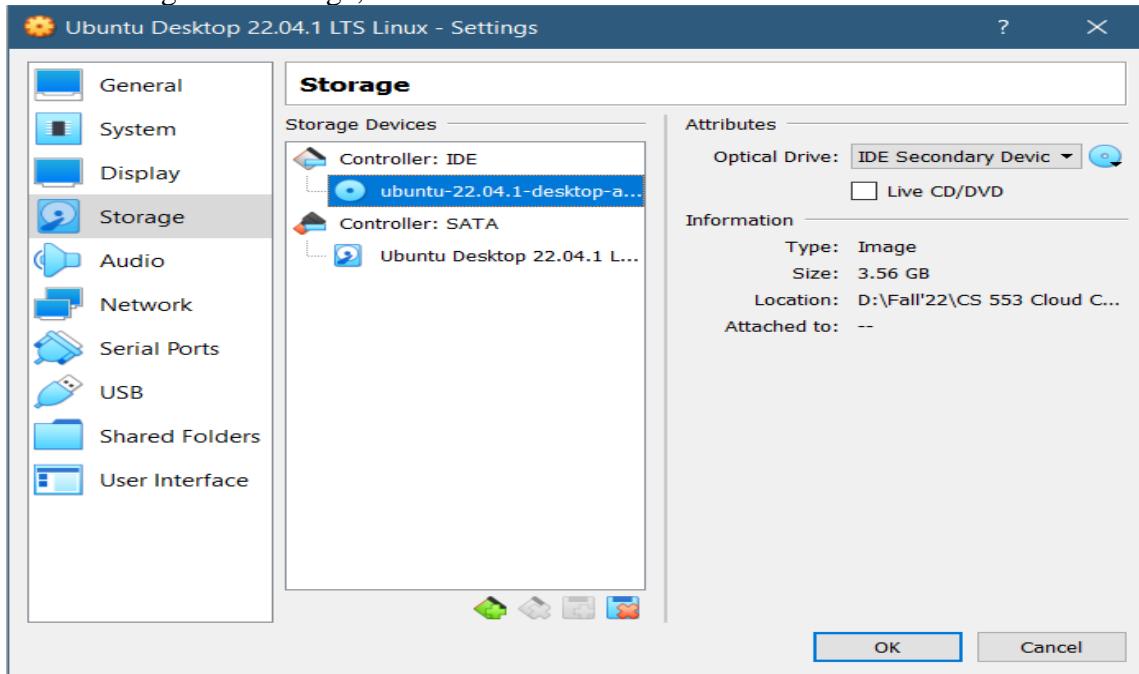


Step 6: We will change the specifications as per our use case/ specification required using the yellow gear icon (top middle) as shown in the above image.

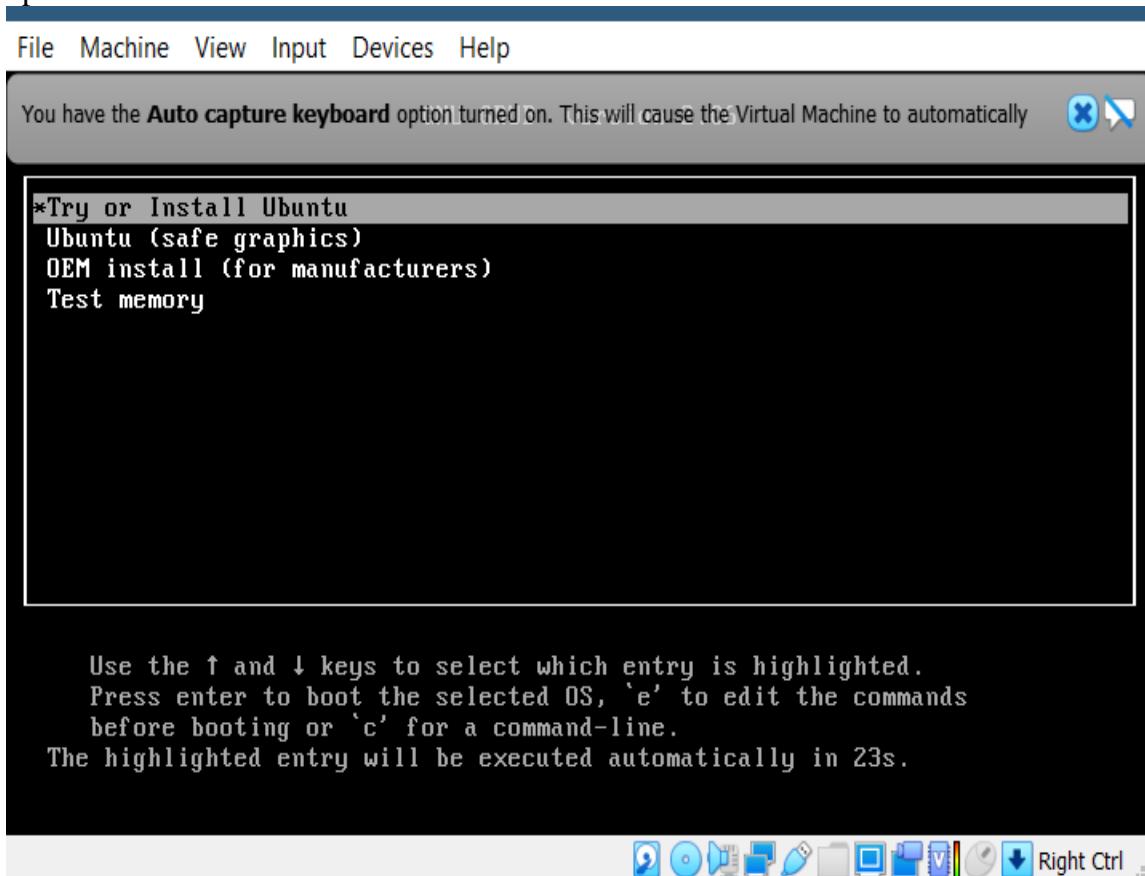
Allocating 2 cores (left), adding the iso image (right) we downloaded using the optical drive button in storage section.

Two side-by-side screenshots of the 'Ubuntu Desktop 22.04.1 LTS Linux - Settings' window. The left window shows the 'System' tab under the 'Processor' tab, where the 'Processor(s)' slider is set to 2. The right window shows the 'Storage' tab, where an 'Empty' optical drive is selected under 'Controller: IDE'. Both windows have a sidebar with icons for General, System, Display, Storage, Audio, Network, Serial Ports, USB, Shared Folders, and User Interface.

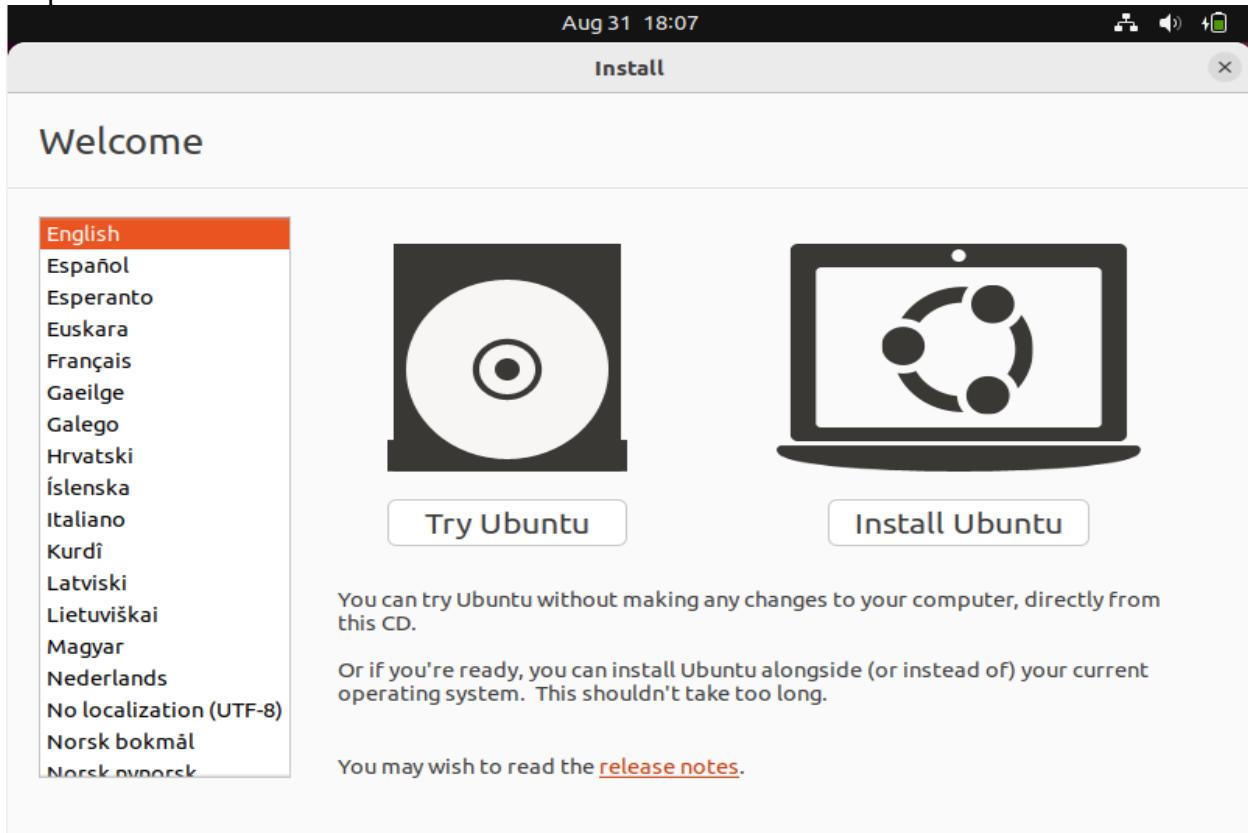
After loading the iso image, it will look like this.



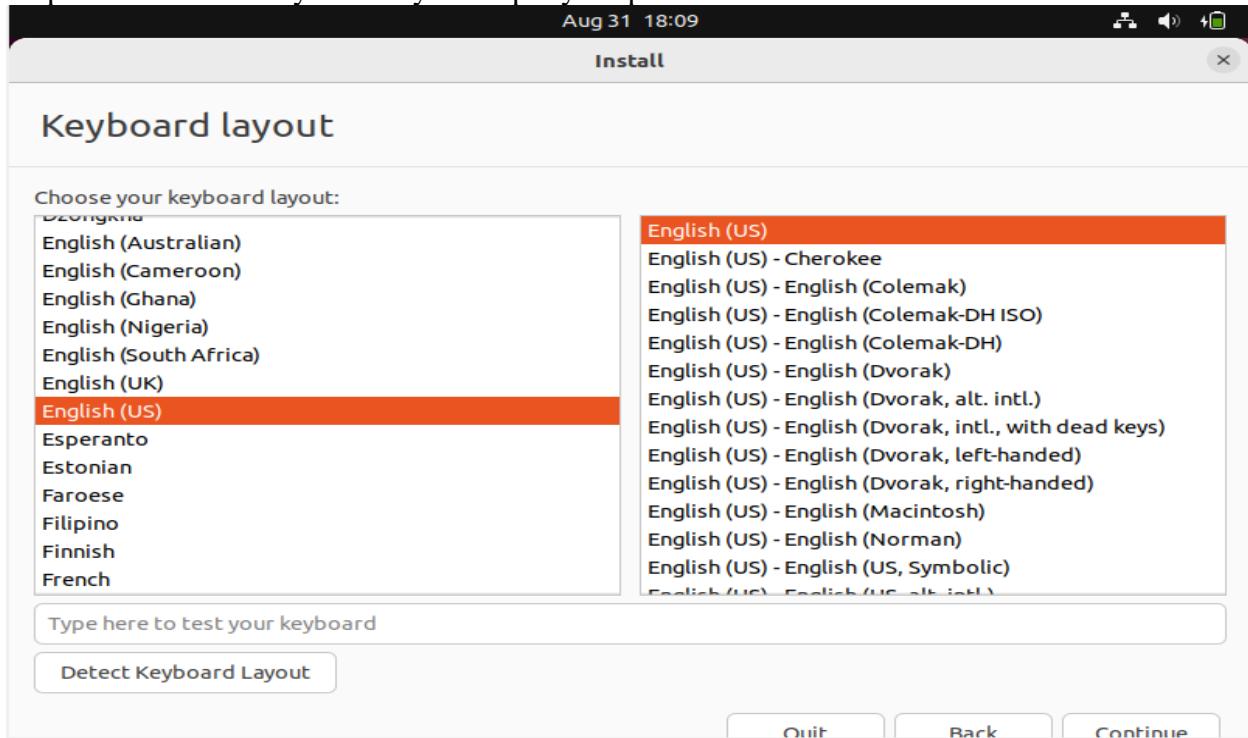
Step 7: After powering up the VM for the first time. It will give a prompt to install it. Use the option to Install Ubuntu.



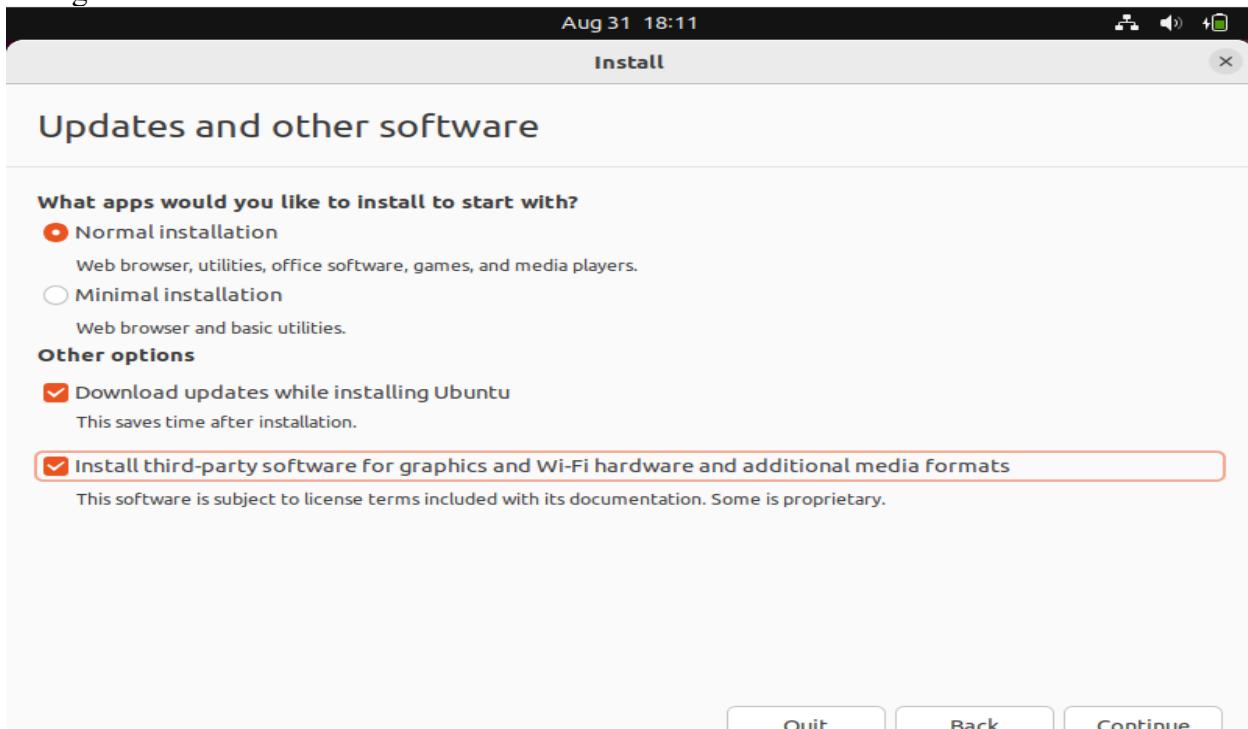
Step 8 : Click on “Install Ubuntu”.



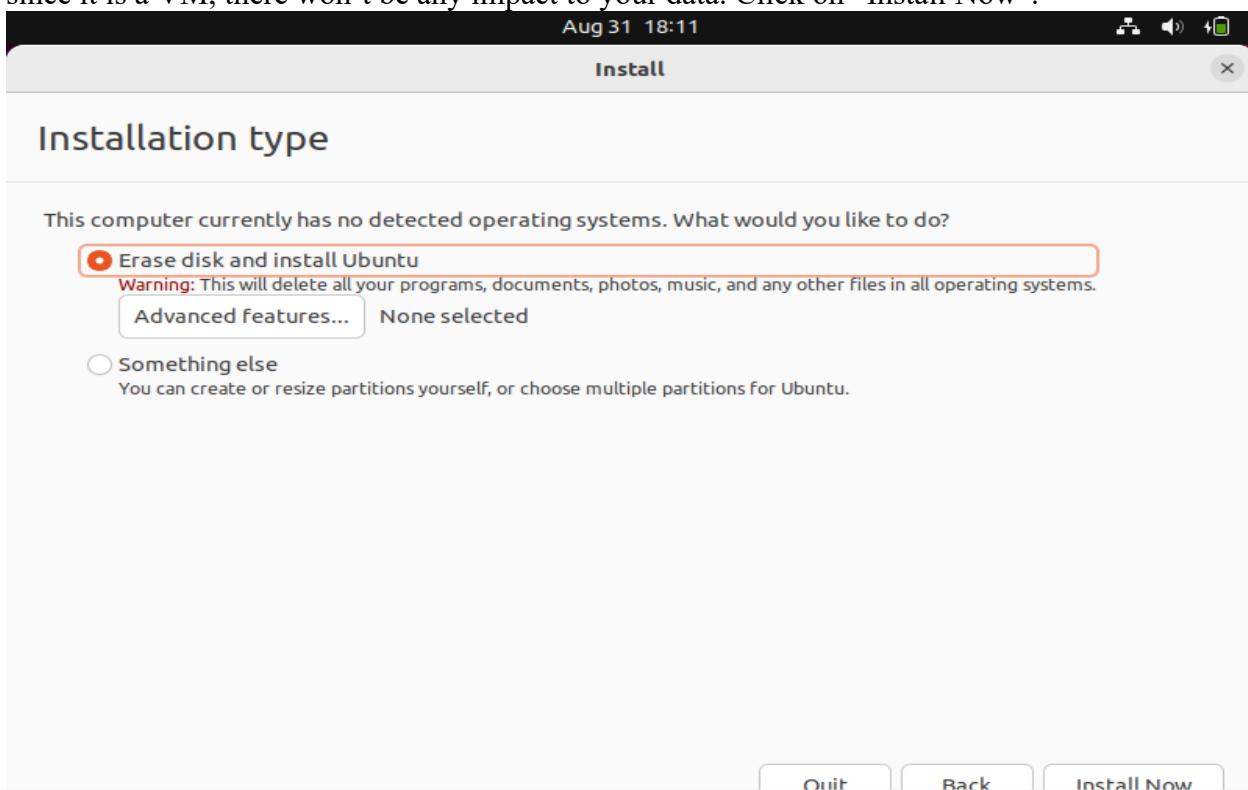
Step 9 : Choose the keyboard layout as per your preference. Click on Continue.



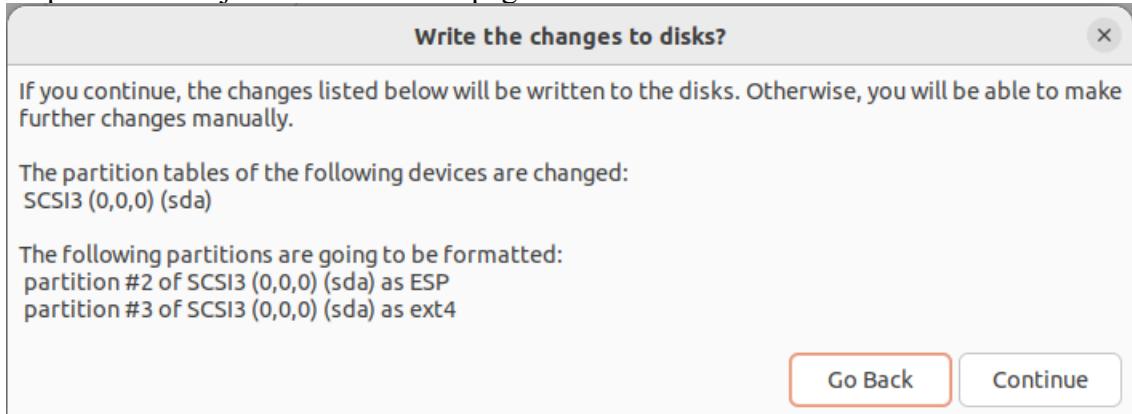
Step 10 : Select the following options in the below screen for update and other software configuration and click on “Continue”.



Step 11 : This final step will ask you if it is okay to erase disk and install Ubuntu. Don't panic, since it is a VM, there won't be any impact to your data. Click on "Install Now".



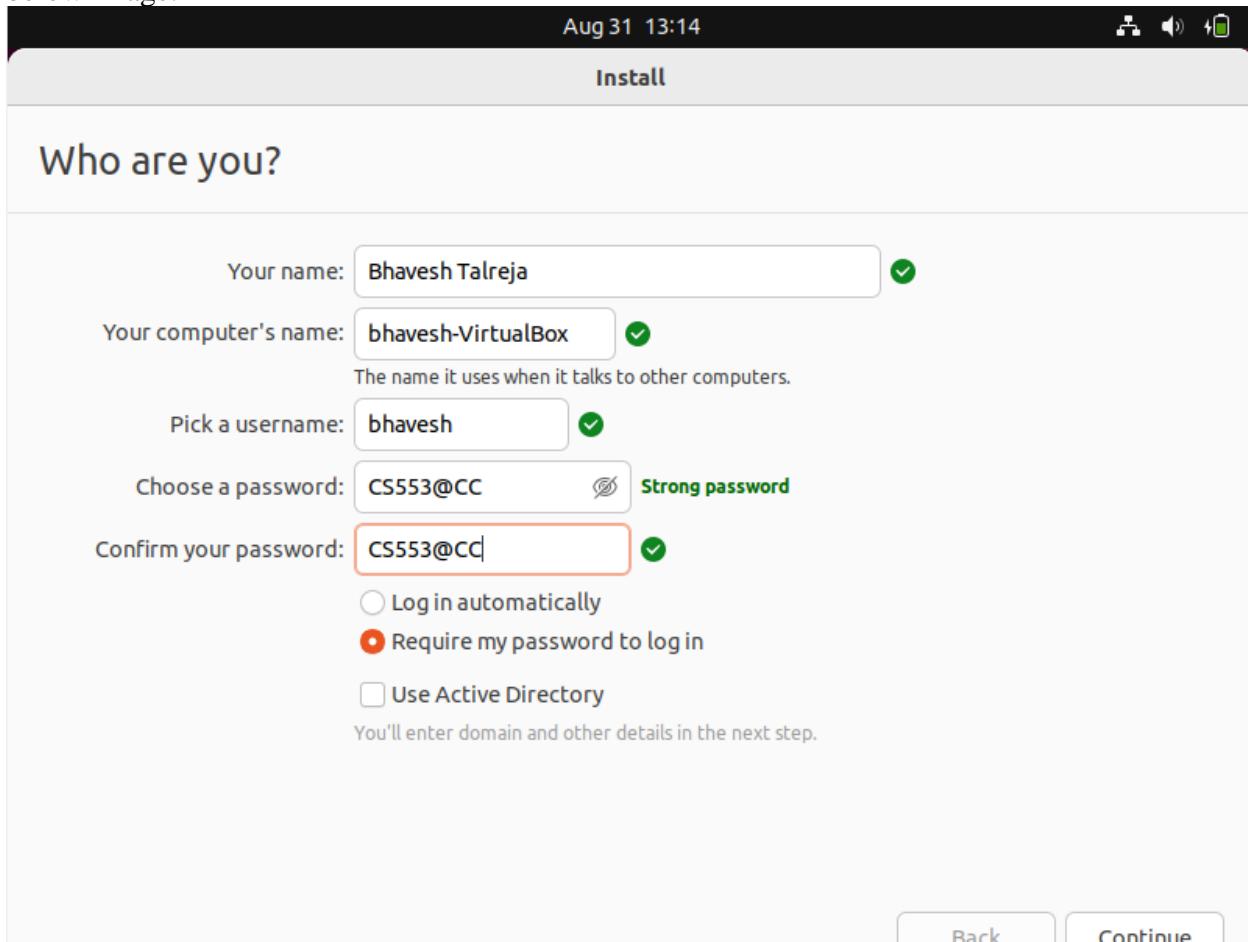
Step 12 : This is just a confirmation page. Click on “Continue”.



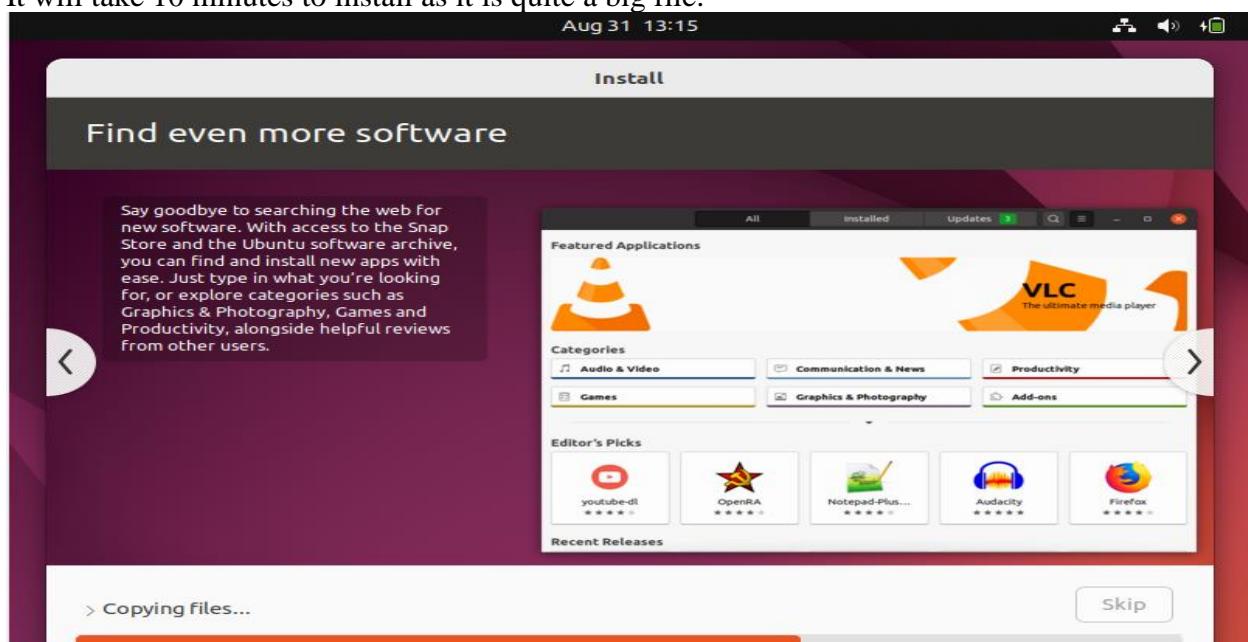
Step 13 : The installation guide asks about the location. Please check and provide accordingly.



Step 14 : Set-up a User ID and password. Click on “Continue” and the installation looks like the below image.



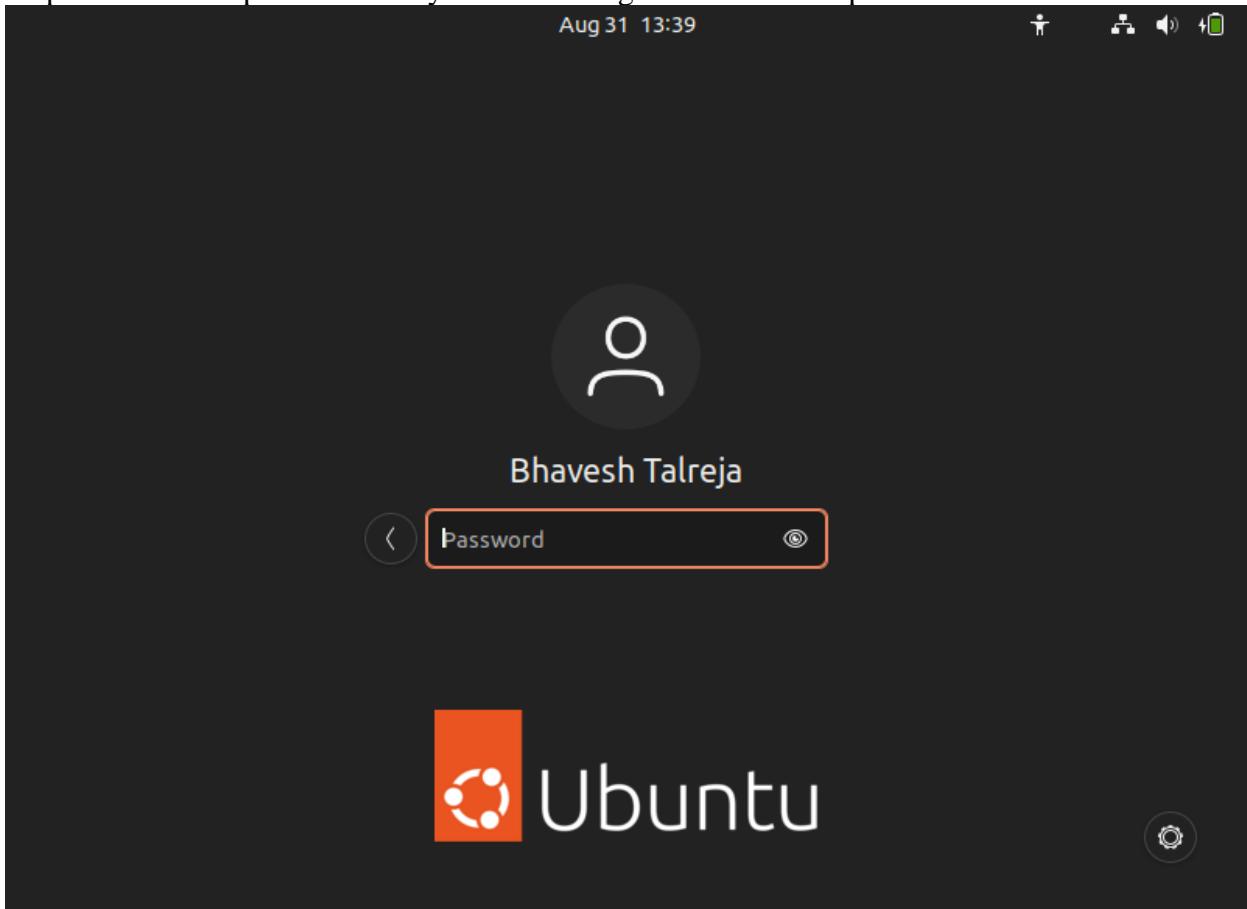
It will take 10 minutes to install as it is quite a big file.



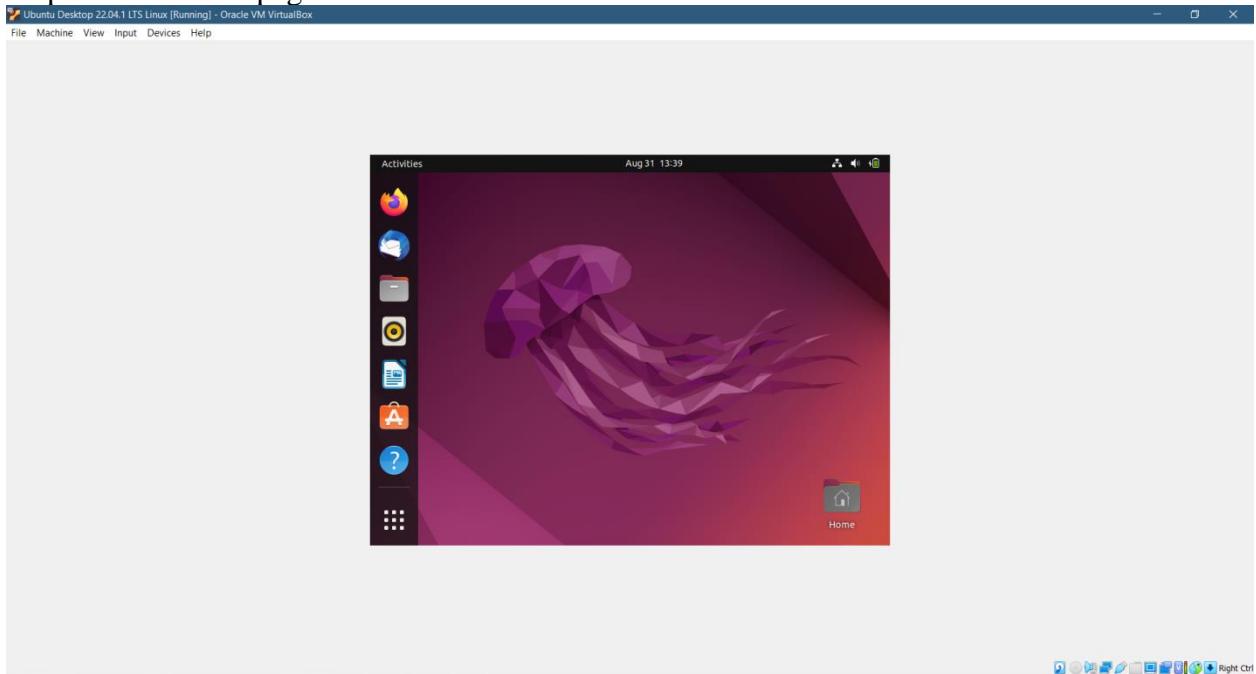
Step 15 : The screen after booting up looks like this and after clicking on the user id, It asks for prompts as below.



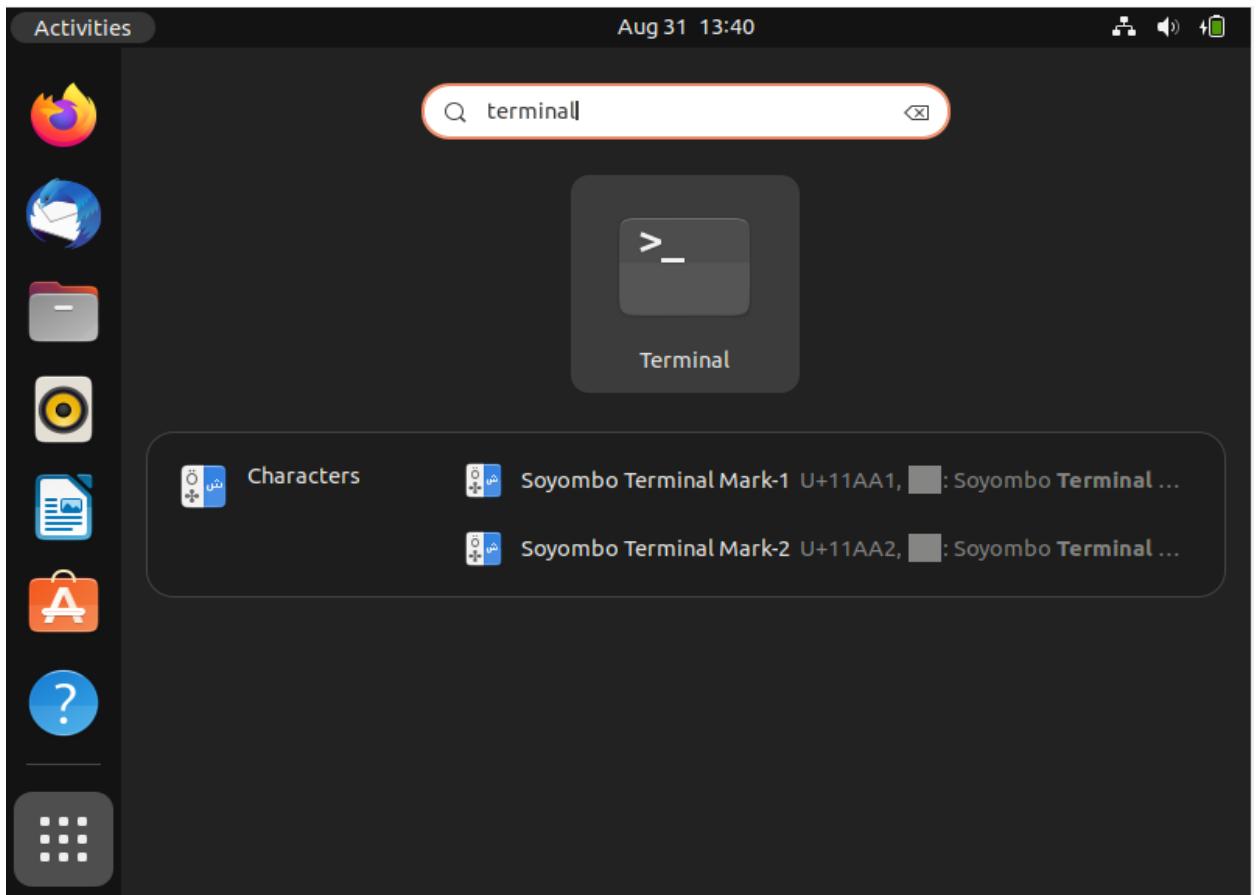
Step 16: Enter the password that you used during the initial backup.



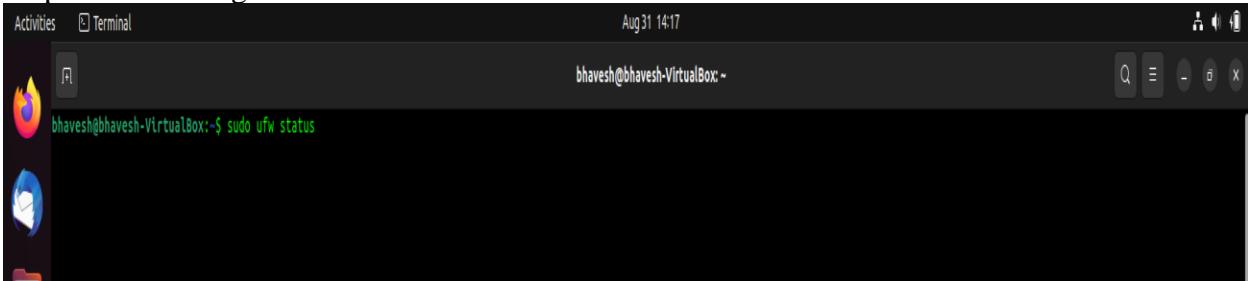
Step 17 : The homepage of this VM looks like below.



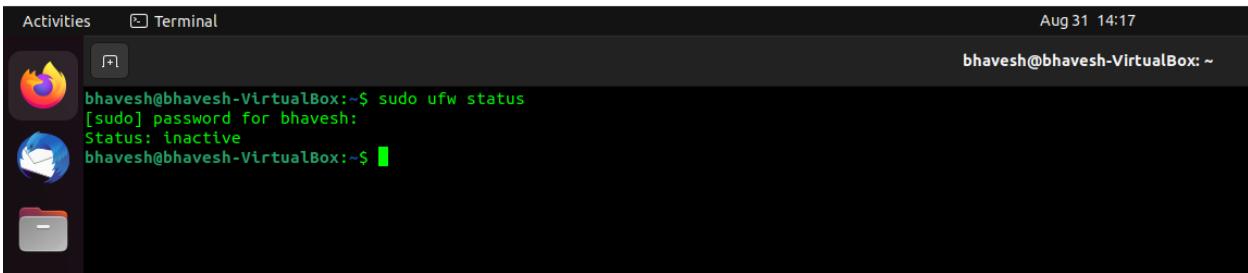
Step 18: We use the terminal utility from the menu button at the bottom left to fire up the terminal.



### Step 19 : Checking the firewall status.

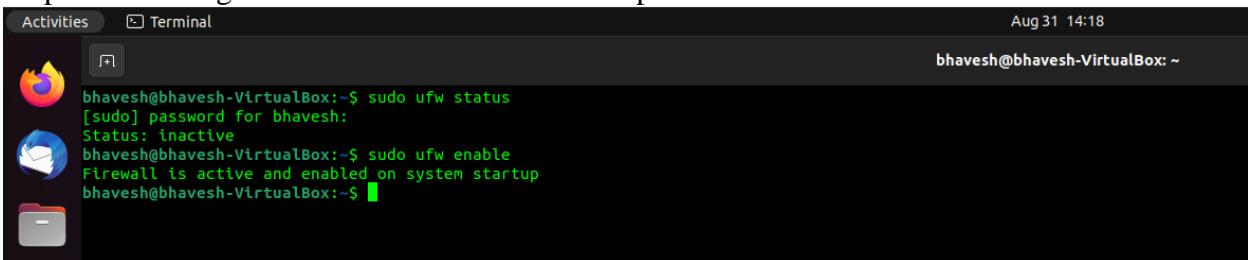


```
Activities Terminal Aug 31 14:17
bhavesh@bhavesh-VirtualBox:~$ sudo ufw status
```

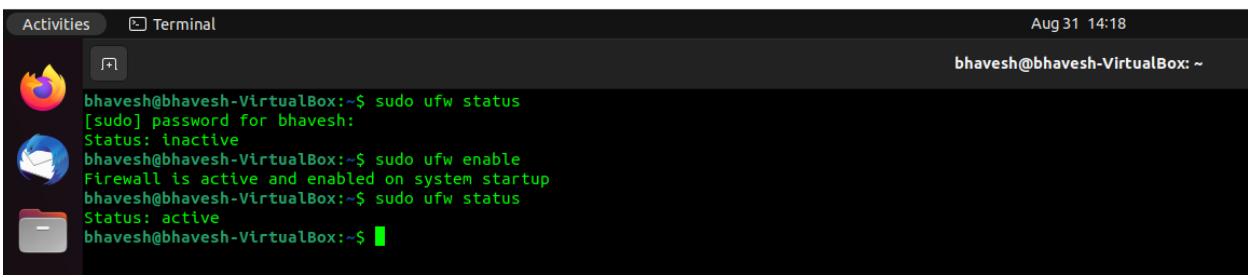


```
Activities Terminal Aug 31 14:17
bhavesh@bhavesh-VirtualBox:~$ sudo ufw status
[sudo] password for bhavesh:
Status: inactive
bhavesh@bhavesh-VirtualBox:~$
```

### Step 20: Enabling the firewall access to block all ports.



```
Activities Terminal Aug 31 14:18
bhavesh@bhavesh-VirtualBox:~$ sudo ufw status
[sudo] password for bhavesh:
Status: inactive
bhavesh@bhavesh-VirtualBox:~$ sudo ufw enable
Firewall is active and enabled on system startup
bhavesh@bhavesh-VirtualBox:~$
```



```
Activities Terminal Aug 31 14:18
bhavesh@bhavesh-VirtualBox:~$ sudo ufw status
[sudo] password for bhavesh:
Status: inactive
bhavesh@bhavesh-VirtualBox:~$ sudo ufw enable
Firewall is active and enabled on system startup
bhavesh@bhavesh-VirtualBox:~$ sudo ufw status
Status: active
bhavesh@bhavesh-VirtualBox:~$
```

## Step 21: Installing and Enabling the SSH server.

```
Reading state information... Done
33 packages can be upgraded. Run 'apt list --upgradable' to see them.
bhavesh@bhavesh-VirtualBox:~$ sudo apt install openssh-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ncurses-term openssh-sftp-server ssh-import-id
Suggested packages:
  molly-guard monkeysphere ssh-askpass
The following NEW packages will be installed:
  ncurses-term openssh-server openssh-sftp-server ssh-import-id
0 upgraded, 4 newly installed, 0 to remove and 33 not upgraded.
Need to get 751 kB of archives.
After this operation, 6,046 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 openssh-sftp-server amd64 1:8.9p1-3 [38.8 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 openssh-server amd64 1:8.9p1-3 [434 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 ncurses-term all 6.3-2 [267 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 ssh-import-id all 5.11-0ubuntu1 [10.1 kB]
Fetched 751 kB in 0s (2,398 kB/s)
Preconfiguring packages ...
Selecting previously unselected package openssh-sftp-server.
(Reading database ... 204137 files and directories currently installed.)
Preparing to unpack .../openssh-sftp-server_1%3a8.9p1-3_amd64.deb ...
Unpacking openssh-sftp-server (1:8.9p1-3) ...
Selecting previously unselected package openssh-server.
Preparing to unpack .../openssh-server_1%3a8.9p1-3_amd64.deb ...
Unpacking openssh-server (1:8.9p1-3) ...
Selecting previously unselected package ncurses-term.
Preparing to unpack .../ncurses-term_6.3-2_all.deb ...
Unpacking ncurses-term (6.3-2) ...

```

## Step 22: Checking the ssh port status on firewall and opening the ssh port.

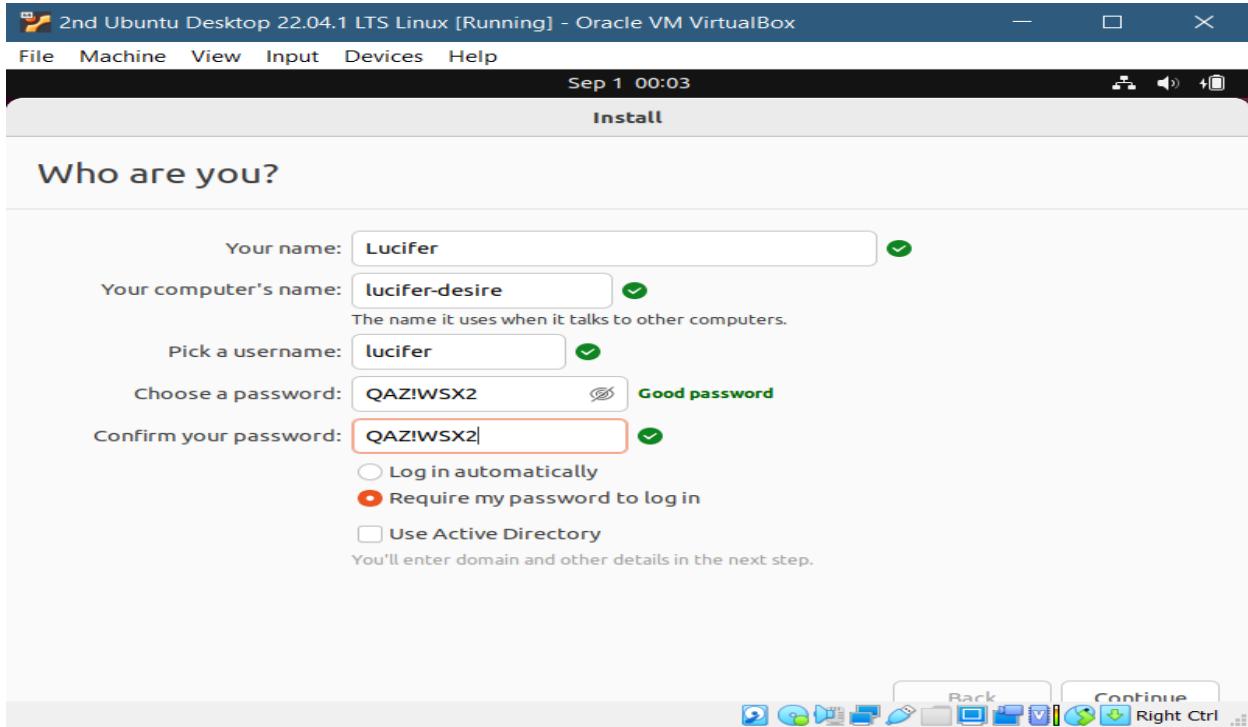
```
bhavesh@bhavesh-VirtualBox:~$ sudo ufw allow ssh
Skipping adding existing rule
Skipping adding existing rule (v6)
bhavesh@bhavesh-VirtualBox:~$
```

```
bhavesh@bhavesh-VirtualBox:~$ sudo ufw status verbose
[sudo] password for bhavesh:
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         ----      --
22/tcp                     ALLOW IN   Anywhere
2222/tcp                   ALLOW IN   Anywhere
22/tcp (v6)                 ALLOW IN   Anywhere (v6)
2222/tcp (v6)               ALLOW IN   Anywhere (v6)

bhavesh@bhavesh-VirtualBox:~$
```

Step 23: Following Step 1 to step 13 to set up another virtual machine and below is the final page after page 14<sup>th</sup>. Then follow all the steps from 15 to step 22 so enable and set up a SSH on the other VM as well.



Step 24 : Creating and installing the public/private key pair in Virtual Machine 1

```
bhavesh@bhavesh-VirtualBox:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/bhavesh/.ssh/id_rsa): elderwand
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in elderwand
Your public key has been saved in elderwand.pub
The key's randomart image is:
+--[RSA 3072]----+
|   . |
|   . |
| o . |
| o . + o |
| .. + S o |
|. .+o o o |
|.+**+ + . |
|=*=*.= . o . .+.E|
|80= . .o .oo.o|
+---[SHA256]-----+
bhavesh@bhavesh-VirtualBox:~$
```

## Step 25 : Creating and installing the public/private key pair in Virtual Machine 2

```
lucifer@lucifer-desire:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/lucifer/.ssh/id_rsa): pikachu
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in pikachu
Your public key has been saved in pikachu.pub
The key fingerprint is:
SHA256:qL4Z1FjH6ht43xuU2c3F9FkLwrjRo/Dtsvm/M+lrIRc lucifer@lucifer-desire
The key's randomart image is:
+---[RSA 3072]----+
|          . . o|
|          o o +. .o=|
|          . = . o+|
|          + + + = E . |
|          o + S = . + |
|          . + o o o |
|          + + = o o |
|          . + + + . = |
|          +... . +o+=+ |
+---[SHA256]----+
lucifer@lucifer-desire:~$
```

## Step 26 : Testing the connection remotely from VM1 to VM2 with the key pair. First checking the ip address using ifconfig.

```
bhavesh@bhavesh-VirtualBox:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
        inet6 fe80::b111:e102:fbaa:a4e1 prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:b5:69:6c txqueuelen 1000 (Ethernet)
                RX packets 196 bytes 35196 (35.1 KB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 326 bytes 34511 (34.5 KB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
            RX packets 289 bytes 25533 (25.5 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 289 bytes 25533 (25.5 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

bhavesh@bhavesh-VirtualBox:~$
```

```
lucifer@lucifer-desire:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.4 netmask 255.255.255.0 broadcast 10.0.2.255
        inet6 fe80::c956:17e1:a1bb:9600 prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:5d:40:2c txqueuelen 1000 (Ethernet)
                RX packets 1251 bytes 937083 (937.0 KB)
                RX errors 0 dropped 0 overruns 0 frame 0
                TX packets 1152 bytes 119925 (119.9 KB)
                TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
        loop txqueuelen 1000 (Local Loopback)
            RX packets 254 bytes 24718 (24.7 KB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 254 bytes 24718 (24.7 KB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lucifer@lucifer-desire:~$
```

Step 26 (continue):

```
bhavesh@bhavesh-VirtualBox:~$ ssh lucifer@10.0.2.4
lucifer@10.0.2.4's password:
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-47-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

24 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Thu Sep  1 17:49:48 2022 from 10.0.2.15
lucifer@lucifer-desire:~$ ls
arc2 arc2.pub Desktop Documents Downloads Music Pictures pikachu pikachu.pub Public snap Templates Videos
lucifer@lucifer-desire:~$ exit
logout
Connection to 10.0.2.4 closed.
bhavesh@bhavesh-VirtualBox:~$
```

Step 28: Installing the public key of VM1 to VM2 to avoid entering the password everytime a ssh connection is requested.

```
bhavesh@bhavesh-VirtualBox:~$ ssh-copy-id -i /home/bhavesh/elderwand.pub lucifer@10.0.2.4
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/bhavesh/elderwand.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
lucifer@10.0.2.4's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'lucifer@10.0.2.4'"
and check to make sure that only the key(s) you wanted were added.

bhavesh@bhavesh-VirtualBox:~$
```

Direct connection to VM2 using the public/private key pair.

```
bhavesh@bhavesh-VirtualBox:~$ ssh lucifer@10.0.2.4 -i elderwand
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-47-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

24 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Last login: Thu Sep  1 17:50:08 2022 from 10.0.2.15
lucifer@lucifer-desire:~$ exit
logout
Connection to 10.0.2.4 closed.
bhavesh@bhavesh-VirtualBox:~$
```

2. Show an example of using the following commands (hint: you can use man to find more information about each one); take screen shots of your commands; make sure to clear the screen between each command; explain in your own words what these commands do:

a. ssh : SSH stands for Secure Shell. It is a protocol that is used for a client-server connection, where the client and server communicate via a secure channel.

\$ man ssh:

```
ssh(1)                                850 General Commands Manual                               ssh(1)

NAME
    ssh - OpenSSH remote login client

SYNOPSIS
    ssh [-46AcGgKkMNngsTtVvxxYy] [-B bind_interface] [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
        [-E log_file] [-e escape_char] [-F configFile] [-I pkcs11]
        [-i identity_file] [-J [user@]host[:port]] [-L address]
        [-l login_name] [-n mac_spec] [-O ctl_cmd] [-o option] [-P port]
        [-Q query_option] [-R address] [-S ctl_path] [-W host:port]
        [-w local_tun[:remote_tun]] destination [command [argument ...]]

DESCRIPTION
    ssh (SSH client) is a program for logging into a remote machine and for executing commands on a remote machine. It is intended to provide secure encrypted communications between two untrusted hosts over an insecure network. X11 connections, arbitrary TCP ports and UNIX-domain sockets can also be forwarded over the secure channel.

    ssh connects and logs into the specified destination, which may be specified as either [user@]hostname or a URI of the form ssh://[user@]hostname[:port]. The user must prove their identity to the remote machine using one of several methods (see below).

    If a command is specified, it will be executed on the remote host instead of a login shell. A complete command line may be specified as command, or it may have additional arguments. If supplied, the arguments will be appended to the command, separated by spaces, before it is sent to the server to be executed.

    The options are as follows:

    -4      Forces ssh to use IPv4 addresses only.
    -6      Forces ssh to use IPv6 addresses only.
    -A      Enables Forwarding of connections from an authentication agent such as ssh-agent(1). This can also be specified on a per-host basis in a configuration file.

            Agent Forwarding should be enabled with caution. Users with the ability to bypass file permissions on the remote host (for the agent's UNIX-domain socket) can access the local agent through the forwarded connection. An attacker cannot obtain key material from the agent, however they can perform operations on the keys that enable them to authenticate using the identities loaded into the agent. A safer alternative may be to use a jump host (see -J).

    -a      Disables Forwarding of the authentication agent connection.

    -B bind_interface
            Bind to the address of bind_interface before attempting to connect to the destination host. This is only useful on systems with more than one address.

    -b bind_address
            Use bind_address on the local machine as the source address of the connection. Only useful on systems with more than one address.

    -c      Requests compression of all data (including stdin, stdout, stderr, and data for forwarded X11, TCP and UNIX-domain connections). The compression algorithm is the same used by gzip(1). Compression is desirable on modem lines and other slow connections, but will only slow down things on fast networks. The default value can be set on a host-by-host basis in the configuration files; see the Compression option.

    -c cipher_spec
            Selects the cipher specification for encrypting the session. cipher_spec is a comma-separated list of ciphers listed in order of preference. See the Ciphers keyword in ssh_config(5) for more information.

    -F configFile
            Use configFile as the configuration file for ssh. By default, /etc/ssh/ssh_config is used.

    -G      Generate host keys. This option is passed directly to ssh-keygen(1).
```

Example ssh:

```
bhavesh@bhavesh-VirtualBox:~$ ssh
usage: ssh [-46AcGgKkMNngsTtVvxxYy] [-B bind_interface]
           [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
           [-E log_file] [-e escape_char] [-F configFile] [-I pkcs11]
           [-i identity_file] [-J [user@]host[:port]] [-L address]
           [-l login_name] [-n mac_spec] [-O ctl_cmd] [-o option] [-P port]
           [-Q query_option] [-R address] [-S ctl_path] [-W host:port]
           [-w local_tun[:remote_tun]] destination [command [argument ...]]
bhavesh@bhavesh-VirtualBox:~$
```

b. ssh-keygen : This command is used for creating the public/private key pair for SSH protocol. Also, there are various other operations that can be used with this command to manage and convert the authentication required for SSH protocol to work.

\$ man ssh-keygen:

```
SSH-KEYGEN(1)                               BSD General Commands Manual                               SSH-KEYGEN(1)

NAME
    ssh-keygen - OpenSSH authentication key utility

SYNOPSIS
    ssh-keygen [-q] [-o rounds] [-b bits] [-C comment] [-f output_keyfile] [-m format] [-N new_passphrase] [-D option] [-t dsa | ecdsa | ecdsa-sk | ed25519 | ed25519-sk | rsa] [-w provider]
    ssh-keygen [-Z cipherids] [-f keyfile] [-m format] [-N new_passphrase] [-D old_passphrase] [-Z cipher]
    ssh-keygen -L [-f input_keyfile] [-m key_format]
    ssh-keygen -E [-f input_keyfile] [-m key_format]
    ssh-keygen -Y [-f input_keyfile]
    ssh-keygen -I [-f input_keyfile] [-C comment] [-f keyfile] [-P passphrase]
    ssh-keygen -L [-V] [-f fingerarint_hash] [-f input_keyfile]
    ssh-keygen -B [-f input_keyfile]
    ssh-keygen -D pkeyall
    ssh-keygen -R known_hosts [-l]
    ssh-keygen -H [-f known_hosts_file]
    ssh-keygen -K [-o rounds] [-w provider]
    ssh-keygen -R hostname [-f known_hosts_file]
    ssh-keygen -M hostname [-f [-f input_keyfile]
    ssh-keygen -N certificate [-D provider] [-f input_keyfile]
    ssh-keygen -M screen [-f input_file] [-o option] output_file
    ssh-keygen -T certificate_identity > ca_key [-HU] [-D pkeyall_provider] [-o principals] [-o option] [-V validity_interval] [-s serial_number] file ...
    ssh-keygen -L [-f input_keyfile]
    ssh-keygen -K [-f kex_file] [-o] [-s ca_public] [-z version_number] file ...
    ssh-keygen -Q [-l] -f krl_file file ...
    ssh-keygen -V add-principals [-o option] > signature_file [-r allowed_signers_file
    ssh-keygen -V check-novate [-o option] > namespace > signature_file
    ssh-keygen -V sign [-o option] -f key_file > namespace ...
    ssh-keygen -V verify [-o option] -f allowed_signers_file > signer_identity > namespace > signature_file (-r revocation_file)

DESCRIPTION
    ssh-keygen generates, manages and converts authentication keys for ssh(1). ssh-keygen can create keys for use by SSH protocol version 2.
    The type of key to be generated is specified with the -t option. If invoked without any arguments, ssh-keygen will generate an RSA key.
    ssh-keygen is also used to generate groups for use in Diffie-Hellman group exchange (DH-GEX). See the MODULI GENERATION section for details.
    Finally, ssh-keygen can be used to generate and update key Revocation Lists, and to test whether given keys have been revoked by one. See the KEY REVOCATION LISTS section for details.
    Normally each user wishing to use SSH with public key authentication runs this once to create the authentication key in ~/.ssh/id_dsa, ~/.ssh/id_ecdsa, ~/.ssh/id_ecdsa_sk, ~/.ssh/id_ed25519, ~/.ssh/id_ed25519_sk or ~/.ssh/id_rsa. Additionally, the system administrator may use this to generate host keys.
    Normally this program generates the key and asks for a file in which to store the private key. The public key is stored in a file with the same name but ".pub" appended. The program also
    normally appends .pub to the file name if it is not present.
```

\$ ssh-keygen -q

```
[+]
bhavesh@bhavesh-VirtualBox:~$ ssh-keygen -q
Enter file in which to save the key (/home/bhavesh/.ssh/id_rsa): demo_key_save
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
bhavesh@bhavesh-VirtualBox:~$
```

(passphrase: qaz)

c. scp : The scp command is known as Secure file Copy protocol. It is used to copy files between servers securely. It can be used to transfer files between local and remote server or between two remote servers. In case there is need of authentication, scp asks for password and passphrase.

\$ man scp :

```
SCP(1)                                BSD General Commands Manual                               SCP(1)

NAME
    scp - OpenSSH secure file copy

SYNOPSIS
    scp [-346ABCpqRstv] [-c cipher] [-D sftp_server_path] [-F ssh_config] [-i identity_file] [-o destination] [-t limit] [-o ssh_option] [-P port] [-S program] source ... target

DESCRIPTION
    scp copies files between hosts on a network.

    It uses ssh(1) for data transfer, and uses the same authentication and provides the same security as a login session.

    scp will ask for passwords or passphrases if they are needed for authentication.

    The source and target may be specified as a local pathname, a remote host with optional path in the form [user@]host[:path], or a URI in the form scp://[user@]host[:port][/:path]. Local file names can be made explicit using absolute or relative pathnames to avoid scp treating file names containing ':' as host specifiers.

    When copying between two remote hosts, if the URI format is used, a port cannot be specified on the target; if the -R option is used.

    The options are as follows:

    -3      Copies between two remote hosts are transferred through the local host. Without this option the data is copied directly between the two remote hosts. Note that, when using the original SCP protocol (the default), this option selects batch mode for the second host as scp cannot ask for passwords or passphrases for both hosts. This mode is the default.

    -4      Forces scp to use IPv4 addresses only.

    -6      Forces scp to use IPv6 addresses only.

    -A      Allows forwarding of ssh-agent(1) to the remote system. The default is not to forward an authentication agent.

    -B      Selects batch mode (prevents asking for passwords or passphrases).

    -C      Compression enable. Passes the -C flag to ssh(1) to enable compression.

    -c cipher
        Selects the cipher to use for encrypting the data transfer. This option is directly passed to ssh(1).

    -D sftp_server_path
        When using the SFTP protocol support via -s, connect directly to a local SFTP server program rather than a remote one via ssh(1). This option may be useful in debugging the client and server.

    -F ssh_config
        Specifies an alternative per-user configuration file for ssh. This option is directly passed to ssh(1).

    -i identity_file
        Selects the file from which the identity (private key) for public key authentication is read. This option is directly passed to ssh(1).

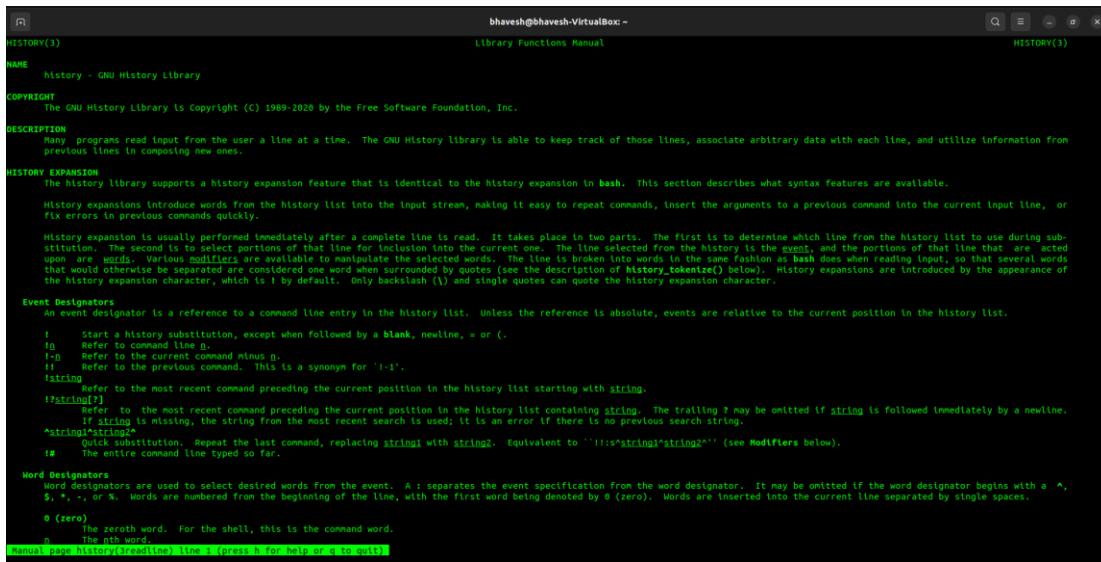
    -o [option]
        Selects the option for the ssh(1) command. See the man page for ssh(1) for help on options to ssh(1).
```

\$ scp (example)

```
bhavesh@bhavesh-VirtualBox:~$ scp Hello.odt Bhavesh@192.168.50.112:0/
ssh: connect to host 192.168.50.112 port 22: Connection refused
lost connection
bhavesh@bhavesh-VirtualBox:~$
```

d. history : This command keeps track of all the commands used in that terminal instance.

\$ man history:



The screenshot shows a terminal window with the title bar "bhavesh@bhavesh-VirtualBox: ~". The window contains the man page for the "history" command. The page is titled "History(3)" and is part of the "Library Functions Manual". It includes sections for HISTORY, NAME, COPYRIGHT, DESCRIPTION, HISTORY EXPANSION, Event Designators, and Word Designators. The text is in a monospaced font, with some words highlighted in green. At the bottom of the window, there is a status bar with the text "Manpage history(3)readline(3) known (press h for help or q to quit)".

```
blavesh@bhavesh-VirtualBox: ~
HISTORY(3)                                         Library Functions Manual                                         HISTORY(3)

NAME
    history - GNU History Library

COPYRIGHT
    The GNU History Library is Copyright (C) 1989-2020 by the Free Software Foundation, Inc.

DESCRIPTION
    Many programs read input from the user a line at a time.  The GNU History library is able to keep track of those lines, associate arbitrary data with each line, and utilize information from previous lines in composing new ones.

HISTORY EXPANSION
    The history library supports a history expansion feature that is identical to the history expansion in bash.  This section describes what syntax features are available.

    History expansions introduce words from the history list into the input stream, making it easy to repeat commands, insert the arguments to a previous command into the current input line, or fix errors in previous commands quickly.

    History expansion is usually performed immediately after a complete line is read.  It takes place in two parts.  The first is to determine which line from the history list to use during substitution.  The second is to perform the substitutions selected from the portion of the line before the event and the portions of that line that are added are words.  Various modifiers are available to manipulate the selected words.  The line is broken into words in the same fashion as bash does when reading input, so that several words that would otherwise be separated are considered one word when surrounded by quotes (see the description of History_tokenize() below).  History expansions are introduced by the appearance of the history expansion character, which is ! by default.  Only backslash (\) and single quotes can quote the history expansion character.

Event Designators
    An event designator is a reference to a command line entry in the history list.  Unless the reference is absolute, events are relative to the current position in the history list.

    !      Start a history substitution, except when followed by a blank, newline, = or ;.
    !n    Refer to the nth line.
    !$D    Refer to the current command minus n.
    !!    Refer to the previous command.  This is a synonym for '-1'.
    !string
        Refer to the most recent command preceding the current position in the history list starting with string.
    !?string!?
        Refer to the most recent command preceding the current position in the history list containing string.  The trailing ? may be omitted if string is followed immediately by a newline.
        If string is missing, the string from the most recent search is used; it is an error if there is no previous search string.
    *string!*string*
        Quick substitution.  Repeat the last command, replacing string1 with string2.  Equivalent to '!:-s:"string1" "string2"' (see Modifiers below).
    #      The entire command line typed so far.

Word Designators
    Word designators are used to select desired words from the event.  A : separates the event specification from the word designator.  It may be omitted if the word designator begins with a -, $, *, ., or %.  Words are numbered from the beginning of the line, with the first word being denoted by 0 (zero).  Words are inserted into the current line separated by single spaces.

    0 (zero)
        The zeroth word.  For the shell, this is the command word.
    n      The nth word.
Manpage history(3)readline(3) known (press h for help or q to quit)
```

\$ history

```
3 pwd
4 cd /media/bhavesh
5 ls
6 cd VBox_GAs_6.1.36/
7 ls -l
8 sudo ./VBoxLinuxAdditions.run
9 sudo ufw status
10 man ssh-keygen
11 clear
12 ssh-keygen -t
13 clear
14 ssh-keygen -q
15 clear
16 ssh-keygen -q
17 clear
18 man scp
19 clear
20 scp
21 clear
22 ifconfig
23 ipconfig
24 clear
25 ifconfig
26 sudo apt install net-tools
27 clear
28 ifconfig
29 clear
30 pwd
31 clear
32 pwd
33 sudo ufw allow ssh
34 ls -l
35 ls
36 ls -a
37 ls -a -l
38 clear
39 sudo ufw status verbose
40 clear
41 ls -l
42 clear
43 scp Hello.odt Bhavesh@192.168.50.112:0:/
44 clear
45 history
46 clear
47 man history
48 clear
49 history
bhavesh@bhavesh-VirtualBox:~$
```

e. sudo : This command grants the user the permission to execute an command as an administrator. There are multiple options that can be used in conjunction with this to execute the full functionality of this command.

\$ man sudo :

```
bhavesh@bhavesh-VirtualBox:~$ man sudo
Man(8)                                BSD System Manager's Manual                               SUDO(8)

NAME
    sudo, sudoedit – execute a command as another user

SYNOPSIS
    sudo [-h | -K | -k | -V]
    sudo [-v] [-AknS] [-g group] [-h host] [-p prompt] [-w user] [command]
    sudo [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-t timeout] [-A directory] [-r role] [-T timeout] [-u user] [VAR=value] [-t t | -k] [command]
    sudoedit [-AknS] [-C cmd] [-D directory] [-g group] [-h host] [-p prompt] [-K directory] [-r role] [-t time] [-T timeout] [-u user] file ...

DESCRIPTION
    sudo allows a permitted user to execute a command as the superuser or another user, as specified by the security policy. The invoking user's real (not effective) user-ID is used to determine the user name with which to query the security policy.

    sudo supports a plugin architecture for security policies, auditing, and input/output logging. Third parties can develop and distribute their own plugins to work seamlessly with the sudo front-end. The default security policy is sudoers, which is configured via the file /etc/sudoers, or via LDAP. See the Plugins section for more information.

    The security policy determines what privileges, if any, a user has to run sudo. The policy may require that users authenticate themselves with a password or another authentication mechanism. If authentication is required, sudo will exit if the user's password is not entered within a configurable time limit. This limit is policy-specific; the default password prompt timeout for the sudoers security policy is 6 minutes.

    Security policies may support credential caching to allow the user to run sudo again for a period of time without requiring authentication. By default, the sudoers policy caches credentials on a per-terminal basis for 15 minutes. See the timestamp_type and timestamp_timeout options in sudoers(5) for more information. By running sudo with the -v option, a user can update the cached credentials without running a command.

    On systems where sudo is the primary method of gaining superuser privileges, it is imperative to avoid syntax errors in the security policy configuration files. For the default security policy, sudoers changes to the configuration files should be made using the visudo() utility which will ensure that no syntax errors are introduced.

    When invoked as sudoedit, the -e option (described below), is implied.

    Security policies and audit plugins may log successful and failed attempts to run sudo. If an I/O plugin is configured, the running command's input and output may be logged as well.

    The options are as follows:

    -A, --askpass
        Normally, if sudo requires a password, it will read it from the user's terminal. If the -A (askpass) option is specified, a (possibly graphical) helper program is executed to read the user's password and output the password to the standard output. If the SUO_ASKPASS environment variable is set, it specifies the path to the helper program. Otherwise, if sudo.conf(5) contains a line specifying the askpass program, that value will be used. For example:
            # Path to askpass helper program
            Path askpass /usr/X11R6/bin/ssh-askpass
        If no askpass program is available, sudo will exit with an error.

    -B, --bell
        Ring the bell as part of the password prompt when a terminal is present. This option has no effect if an askpass program is used.

Manual page sudo(1). See the man(7) for help on qto quit.
```

\$ sudo :

```
bhavesh@bhavesh-VirtualBox:~$ sudo ufw status verbose
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip

To                         Action      From
--                         -----      ---
22/tcp                     ALLOW IN   Anywhere
22/tcp (v6)                ALLOW IN   Anywhere (v6)

bhavesh@bhavesh-VirtualBox:~$
```

f. ip : This command is used to get IP details and statistics in human readable format. Also, it can be used to change the IP details.

```
$ man ip
```

```
ip(8)                                Linux

NAME
    ip - show / manipulate routing, network devices, interfaces and tunnels

SYNOPSIS
    ip [ OPTIONS ] OBJECT [ COMMAND | help ]

    ip [ -force ] -batch filename
        OBJECT := ( link | address | addrlabel | route | rule | neigh | stable | tunnel | tunctl | maddress | mroute | monitor | xfrm | netns | l2tp | tcp_metrics | token | macsec | vrf |
                    nptcp | iom )
        OPTIONS := { -V[ersion] | -h[uman-readable] | -d[etails] | -r(esolve) | -family| { inet | inets | link } | -4 | -6 | -B | -o | -l[oops] | maximum-addr-flush-attempts
                     } | -e(neline) | -r(vbuf) [size] | -t[imestamp] | -s[hort] | -m(etos) name | -N[umeric] | -w[ill] | -c[oloc] | -b[rief] | -3[on] | -p[retty] )

OPTIONS
    -V, -Version
        Print the version of the ip utility and exit.
    -h, -human, -human-readable
        output statistics with human readable values followed by suffix.
    -b, -batch <FILENAME>
        Read commands from provided file or standard input and invoke them. First failure will cause termination of ip.
    -force Don't terminate ip on errors in batch mode. If there were any errors during execution of the commands, the application return code will be non zero.
    -s, -stats, -statistics
        Output more information. If the option appears twice or more, the amount of information increases. As a rule, the information is statistics or some time values.
    -d, -details
        Output more detailed information.
    -l, -loops <COUNT>
        Specify maximum number of loops the 'ip address flush' logic will attempt before giving up. The default is 10. Zero (0) means loop until all addresses are removed.
    -f, -family <FAMILY>
        Specifies the protocol family to use. The protocol family identifier can be one of inet, inets, bridge, nols or link. If this option is not present, the protocol family is guessed from other arguments. If the rest of the command line does not give enough information to guess the family, ip falls back to the default one, usually inet or any. link is a special family identifier meaning that no networking protocol is involved.
    -4      shortcut for -family inet.
    -6      shortcut for -family inets.
    -B      shortcut for -family bridge.

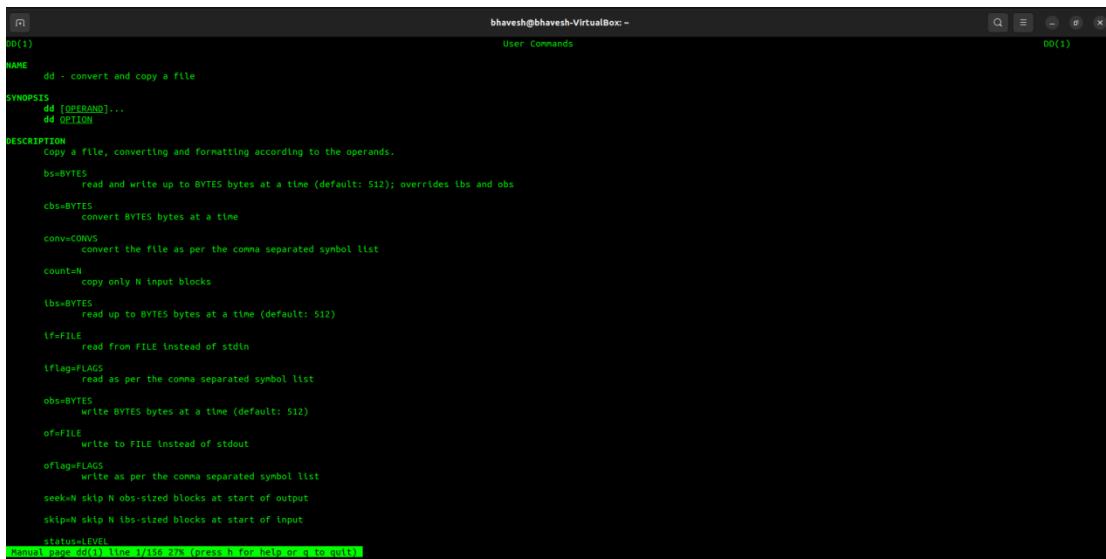
Manual pages for ip(8) Linux (press h for help or q to quit)
```

```
$ ip
```

```
Firefox Web Browser                                bhavesh@bhavesh-VirtualBox: ~
bhavesh@bhavesh-VirtualBox:~$ ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b5:69:6c brd ff:ff:ff:ff:ff:ff
        inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
            valid_lft 82018sec preferred_lft 82018sec
        inet6 fe80::b111:e102:fbba:a4e1/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
bhavesh@bhavesh-VirtualBox:~$
```

g. dd : This command is used to convert and copy files.

```
$ man dd
```



The screenshot shows a terminal window with the title "DD(1)" and the subtitle "User Commands". The window displays the manual page for the dd command. The text is mostly white on a black background. It includes sections for NAME, SYNOPSIS, DESCRIPTION, and various options like bs=BYTES, conv=CONVF, and oflag=FLAGS. At the bottom of the window, there is a status bar with the text "Manual page dd(1) [line 3/256 27% (press h for help or q to quit)].

```
NAME
      dd - convert and copy a file

SYNOPSIS
      dd [OPERAND]...
      dd OPTION

DESCRIPTION
      Copy a file, converting and formatting according to the operands.

      bs=BYTES
          read and write up to BYTES bytes at a time (default: 512); overrides lbs and obs

      cbs=BYTES
          convert BYTES bytes at a time

      conv=CONVF
          convert the file as per the comma separated symbol list

      count=N
          copy only N input blocks

      lbs=BYTES
          read up to BYTES bytes at a time (default: 512)

      if=FILE
          read from FILE instead of stdin

      iflag=FLAGS
          read as per the comma separated symbol list

      obs=BYTES
          write BYTES bytes at a time (default: 512)

      of=FILE
          write to FILE instead of stdout

      oflag=FLAGS
          write as per the comma separated symbol list

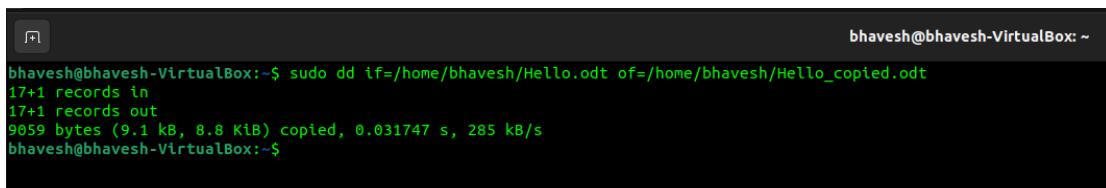
      seek=N
          skip N obs-sized blocks at start of output

      skip=N
          skip N lbs-sized blocks at start of input

      status=LEVEL
          status=LEVEL
```

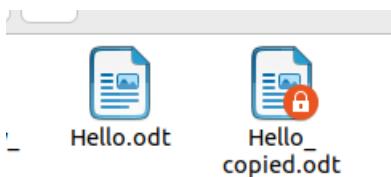
Manual page dd(1) [line 3/256 27% (press h for help or q to quit)]

```
$ dd
```



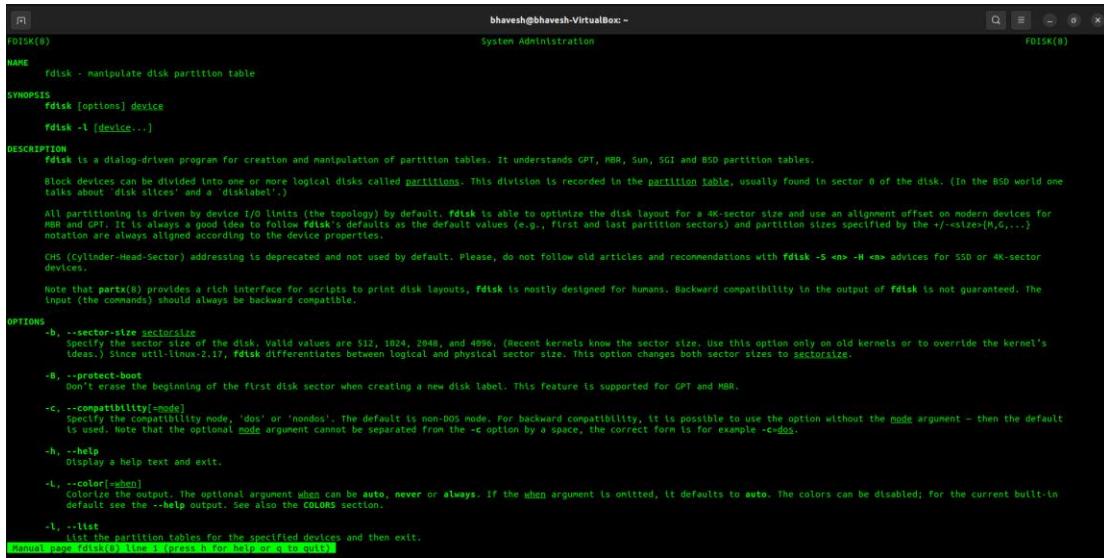
The screenshot shows a terminal window with the title "bhavesh@bhavesh-VirtualBox: ~". The command "sudo dd if=/home/bhavesh/Hello.odt of=/home/bhavesh>Hello\_copied.odt" is being run. The output shows "17+1 records in" and "17+1 records out", followed by "9059 bytes (9.1 kB, 8.8 KiB) copied, 0.031747 s, 285 kB/s". The terminal prompt "bhavesh@bhavesh-VirtualBox: ~" is visible at the bottom.

```
sudo dd if=/home/bhavesh/Hello.odt of=/home/bhavesh>Hello_copied.odt
17+1 records in
17+1 records out
9059 bytes (9.1 kB, 8.8 KiB) copied, 0.031747 s, 285 kB/s
```



h. fdisk : fdisk command in linux is used to create and modify disk partition table. In modify, we have the option to view, delete, change, resize, and other operations that we can perform on the sections of hard disk.

```
$ man fdisk
```



The screenshot shows a terminal window with the man page for the fdisk command. The title bar says "FDISK(8)". The page content includes:

- NAME**: fdisk - manipulate disk partition table
- SYNOPSIS**:  
fdisk [options] device  
fdisk -l [device...]
- DESCRIPTION**:  
fdisk is a dialog-driven program for creation and manipulation of partition tables. It understands GPT, MBR, Sun, SGI and BSD partition tables. Block devices can be divided into one or more logical disks called partitions. This division is recorded in the partition table, usually found in sector 0 of the disk. (In the BSD world one talks about 'disk slices' and a 'disklabel'.)
- All partitioning is driven by device I/O limits (the topology) by default. fdisk is able to optimize the disk layout for a 4K-sector size and use an alignment offset on modern devices for MBR and GPT. It is always a good idea to follow fdisk's defaults as the default values (e.g., first and last partition sectors) and partition sizes specified by the +/-size{M,G,...} notation are always aligned according to the device properties.
- CMS (Cylinder-Head-Sector) addressing is deprecated and not used by default. Please, do not follow old articles and recommendations with fdisk -S <n> -H <m> advices for SSD or 4K-sector devices.
- Note that partx(8) provides a rich interface for scripts to print disk layouts, fdisk is mostly designed for humans. Backward compatibility in the output of fdisk is not guaranteed. The input (the commands) should always be backward compatible.
- OPTIONS**:
  - b, --sector-size *sectorsize*: Specify the sector size of the disk. Valid values are 512, 1024, 2048, and 4096. (Recent kernels know the sector size. Use this option only on old kernels or to override the kernel's ideas.) Since util-linux-2.17, fdisk differentiates between logical and physical sector size. This option changes both sector sizes to *sectorsize*.
  - B, --protect-boot: Don't erase the beginning of the first disk sector when creating a new disk label. This feature is supported for GPT and MBR.
  - c, --compatibility[=*mode*]: Specify the compatibility mode, 'dos' or 'nondos'. The default is non-DOS mode. For backward compatibility, it is possible to use the option without the *mode* argument – then the default is used. Note that the optional *mode* argument cannot be separated from the -c option by a space, the correct form is for example -c=dos.
  - h, --help: Display a help text and exit.
  - L, --color[=*when*]: Colorize the output. The optional argument *when* can be auto, never or always. If the *when* argument is omitted, it defaults to auto. The colors can be disabled; for the current built-in default see the --help output. See also the COLORS section.
  - l, --list: List the partition tables for the specified devices and then exit.

```
$ fdisk
```

```
bhavesh@bhavesh-VirtualBox:~$ fdisk -h

Usage:
  fdisk [options] <disk>          change partition table
  fdisk [options] -l [<disk>...] list partition table(s)

Display or manipulate a disk partition table.

Options:
  -b, --sector-size <size>      physical and logical sector size
  -B, --protect-boot            don't erase bootbits when creating a new label
  -c, --compatibility[=<mode>]  mode is 'dos' or 'nondos' (default)
  -L, --color[=<when>]         colorize output (auto, always or never)
                               colors are enabled by default
  -l, --list                   display partitions and exit
  -x, --list-details           like --list but with more details
  -o, --noauto-pt              don't create default partition table on empty devices
  -O, --output <list>           output columns
  -t, --type <type>            recognize specified partition table type only
  -U, --units[=<unit>]          display units: 'cylinders' or 'sectors' (default)
  -s, --getsz                  display device size in 512-byte sectors [DEPRECATED]
  --bytes                      print SIZE in bytes rather than in human readable format
  --lock[=<mode>]              use exclusive device lock (yes, no or nonblock)
  -w, --wipe <mode>             wipe signatures (auto, always or never)
  -W, --wipe-partitions <mode>  wipe signatures from new partitions (auto, always or never)

  -C, --cylinders <number>     specify the number of cylinders
  -H, --heads <number>          specify the number of heads
  -S, --sectors <number>        specify the number of sectors per track

  -h, --help                   display this help
  -V, --version                display version

Available output columns:
  gpt: Device Start End Sectors Size Type UUID Attrs Name UUID
  dos: Device Start End Sectors Cylinders Size Type Id Attrs Boot End-C/H/S Start-C/H/S
  bsd: Slice Start End Sectors Cylinders Size Type Bsize Cpg Fsize
  sgi: Device Start End Sectors Cylinders Size Type Id Attrs
  sun: Device Start End Sectors Cylinders Size Type Id Flags

For more details see fdisk(8).
bhavesh@bhavesh-VirtualBox:~$
```

i. apt : This command is a high end command line interface. It is designed to be readable by the end user. It is used in conjunction with sudo to execute various commands on high-level. For example to update, upgrade, install, reinstall, etc.

\$ man apt:

```
bhavesh@bhavesh-VirtualBox:~
```

APT(8) APT(8)

**NAME**

apt - command-line interface

**SYNOPSIS**

```
apt [-h] [-econfig_string] [-cconfig_file] [-ttarget_release] [-marchitecture] [list | search | show | update | install pkg [=pkg_version_number] /target_release]... | remove pkg... | upgrade | full-upgrade | edit-sources | (-v | --version) | (-h | --help)
```

**DESCRIPTION**

```
apt provides a high-level commandline interface for the package management system. It is intended as an end user interface and enables some options better suited for interactive usage by default compared to more specialized APT tools like apt-get(8) and apt-cache(8).
```

Much like apt itself, its manpage is intended as an end user interface and as such only mentions the most used commands and options partly to not duplicate information in multiple places and partly to avoid overwhelming readers with a cornucopia of options and details.

```
update (apt-get(8))  
    update is used to download package information from all configured sources. Other commands operate on this data to e.g. perform package upgrades or search in and display details about all packages available for installation.
```

```
upgrade (apt-get(8))  
    upgrade is used to install available upgrades of all packages currently installed on the system from the sources configured via sources.list(). New packages will be installed if required to satisfy dependencies, but existing packages will never be removed. If an upgrade for a package requires the removal of an installed package the upgrade for this package isn't performed.
```

```
full-upgrade (apt-get(8))  
    full-upgrade performs the function of upgrade but will remove currently installed packages if this is needed to upgrade the system as a whole.
```

```
install, reinstall, remove, purge (apt-get(8))  
    Performs the requested action on one or more packages specified via regex(7), glob(7) or exact match. The requested action can be overridden for specific packages by appending a plus (+) to the package name to install this package or a minus (-) to remove it.
```

A specific version of a package can be selected for installation by following the package name with an equals (=) and the version of the package to select. Alternatively the version from a specific release can be selected by following the package name with a forward slash (/) and codename (bullseye, bookworm, sid ...) or suite name (stable, testing, unstable). This will also select versions from this release for dependencies of this package if needed to satisfy the request.

Removing a package removes all packaged data, but leaves usually small (modified) user configuration files behind, in case the remove was an accident. Just issuing an installation request for the accidentally removed package will restore its function as before in that case. On the other hand you can get rid of these leftovers by calling purge even on already removed packages. Note that this does not affect any data or configuration stored in your home directory.

```
autoremove (apt-get(8))  
    autoremove is used to remove packages that were automatically installed to satisfy dependencies for other packages and are now no longer needed as dependencies changed or the package(s) needing them were removed in the meantime.
```

You should check that the list does not include applications you have grown to like even though they were once installed just as a dependency of another package. You can mark such a package as manually installed by using apt-mark(8). Packages which you have installed explicitly via install are also never proposed for automatic removal.

```
Manual page apt(8) (the i (press h for help or q to quit))
```

\$ apt

```
bhavesh@bhavesh-VirtualBox:~$ apt show zypper
```

```
Package: zypper  
Version: 1.14.42-2  
Priority: optional  
Section: universe/admin  
Origin: Ubuntu  
Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>  
Original-Maintainer: Mike Gabriel <umweaver@debian.org>  
Bugs: https://bugs.launchpad.net/ubuntu/+filebug  
Installed-Size: 2,20 KB  
Depends: libgcc1 (>= 0.6.0), libc6 (>= 2.28), libgcc-s1 (>= 3.3.1), libreadline8 (>= 6.0), libstdc++6 (>= 9), libxml2 (>= 2.7.4), libzypp1722, zypper-common (>= 1.14.42-2)  
Suggests: zypper-doc  
Homepage: https://github.com/openSUSE/zypper  
Download-Size: 742 kB  
APT-Sources: http://us.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages  
Description: command line software manager using libzypp  
Zypper is a command line tool for managing software. It can be used to add package repositories, search for packages, install, remove, or update packages, install patches, hardware drivers, verify dependencies, and more.  
zypper can be used interactively or non-interactively by user, from scripts, or front-ends.
```

```
bhavesh@bhavesh-VirtualBox:~$
```

j. vi : vi or vim is a command line text editor application. It is used to open up an existing file by providing the path along with the command. In case the file doesn't exist at the specified path, a new file will be created with that name at the specified location.

```
$ man vi
```

The screenshot shows the man page for vi(1). The title bar reads "bhavesh@bhavesh-VirtualBox: ~" and "General Commands Manual". The page content is as follows:

```
bhavesh@bhavesh-VirtualBox: ~
General Commands Manual
VIM(1)

NAME
    vim - VI IMproved, a programmer's text editor

SYNOPSIS
    vim [options] [file ...]
    vim [options] -t tag
    vim [options] -q [errorfile]

    ex
    vview
    gvim gview evim eview
    rvim rview rgvim rview

DESCRIPTION
    Vim is a text editor that is upwards compatible to Vi. It can be used to edit all kinds of plain text. It is especially useful for editing programs.

    There are a lot of enhancements above Vi: multi level undo, multi windows and buffers, syntax highlighting, command line editing, filename completion, on-line help, visual selection, etc.. See "/help vim_difft.txt" for a summary of the differences between Vim and Vi.

    While running Vim a lot of help can be obtained from the on-line help system, with the "help" command. See the ON-LINE HELP section below.

    Most often Vim is started to edit a single file with the command
        vim file

    More generally Vim is started with:
        vim [options] [filelist]

    If the filelist is missing, the editor will start with an empty buffer. Otherwise exactly one out of the following four may be used to choose one or more files to be edited.

    file ..      A list of filenames. The first one will be the current file and read into the buffer. The cursor will be positioned on the first line of the buffer. You can get to the other files with the "next" command. To edit a file that starts with a dash, precede the filelist with "--".
    .           The file to edit is read from stdin. Commands are read from stderr, which should be a tty.
    -t {tag}     The file to edit and the initial cursor position depends on a "tag", a sort of goto label. {tag} is looked up in the tags file, the associated file becomes the current file and the associated command is executed. Mostly this is used for C programs, in which case {tag} could be a function name. The effect is that the file containing that function becomes the current file and the cursor is positioned on the start of the function. See "help tag-commands".
    -q [errorfile]   Start in quickfix mode. The file [errorfile] is read and the first error is displayed. If [errorfile] is omitted, the filename is obtained from the 'errorfile' option (defaults to "Arrorc.Err" for the Amiga, "errors.err" on other systems). Further errors can be jumped to with the "inc" command. See "help quickfix".

    Manually page vim(1) line 1/2992 37% (press h for help or q to exit)
```

```
$ vi
```

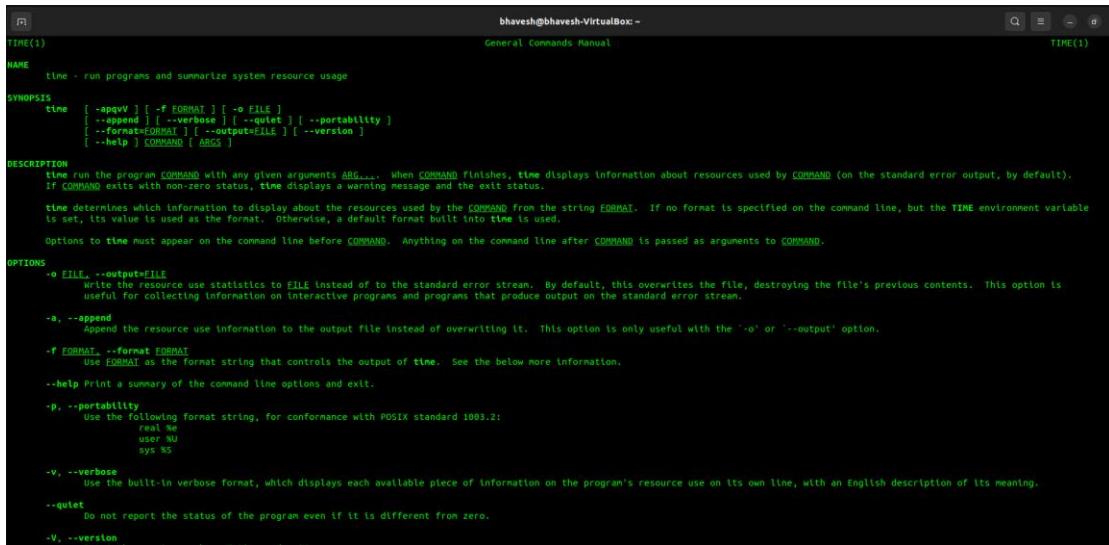
The screenshot shows the vi editor displaying the contents of a zip file. The title bar reads "bhavesh@bhavesh-VirtualBox: ~". The file listing is as follows:

```
bhavesh@bhavesh-VirtualBox: ~
" zip.vim version v32
" Browsing zipfile /home/bhavesh>Hello.odt
" Select a file with cursor and press ENTER

mimetype
Configurations2/accelerator/
Configurations2/images/Bitmaps/
Configurations2/toolpanel/
Configurations2/progressbar/
Configurations2/statusbar/
Configurations2/toolbar/
Configurations2/floater/
Configurations2/popupmenu/
Configurations2/menuubar/
manifest.rdf
Meta.xml
settings.xml
Thumbnails/thumbnail.png
styles.xml
content.xml
META-INF/manifest.xml
~
```

k. time : This command returns the time required for an command/process to run.

```
$ man time
```



The screenshot shows a terminal window with the title "bhavesh@bhavesh-VirtualBox: ~". The window displays the "General Commands Manual" for the "TIME(1)" command. The man page includes sections for NAME, SYNOPSIS, DESCRIPTION, and OPTIONS. The SYNOPSIS section shows the command usage: "time [ -apqv ] [ -F FORMAT ] [ -o FILE ] [ --append ] [ --verbose ] [ --quiet ] [ --portability ] [ --format=FORMAT ] [ --output=FILE ] [ --version ] [ --help ] COMMAND [ ARGS ]". The DESCRIPTION section explains that "time run the program COMMAND with any given arguments ARGS...". It also mentions that if COMMAND exits with non-zero status, time displays a warning message and the exit status. The OPTIONS section details various flags and their meanings, such as "-o FILE" for outputting resource use statistics to a file, "-a" for appending to the output file, and "-v" for verbose output.

```
$ time
```

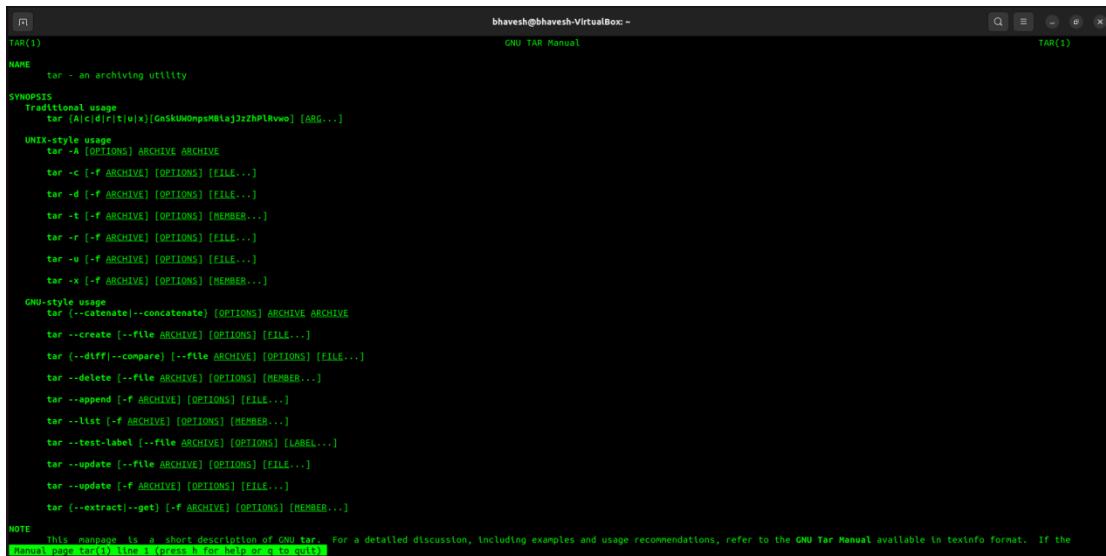


The screenshot shows a terminal window with the title "bhavesh@bhavesh-VirtualBox: ~". The user runs the command "time sleep 1.2". The output shows the execution times: "real 0m1.202s", "user 0m0.001s", and "sys 0m0.000s".

```
real    0m1.202s
user    0m0.001s
sys     0m0.000s
```

1. tar : This command is used as a tool to archive files. There are various options with which we can use tar for applying different compression technique, with or without password archival, etc.

```
$ man tar
```



The screenshot shows a terminal window titled "GNU TAR Manual" with the command "tar(1)" at the top. The manual page content is displayed, starting with the SYNOPSIS section which lists various tar commands like tar -c, tar -x, tar -t, etc., with their options and arguments. Below the synopsis is the NOTE section, which contains a short description of the manual and instructions for viewing it.

```
bhavesh@bhavesh-VirtualBox:~
```

**TAR(1)** TAR(1)

**NAME** tar - an archiving utility

**SYNOPSIS**

Traditional usage

```
tar [A|c|d|r|t|u|x][GnSkWnOnpaM8iajJzZhPjRvwo] [ARG...]
```

UNIX-style usage

```
tar -A [OPTIONS] ARCHIVE ARCHIVE
```

```
tar -c [-f ARCHIVE] [OPTIONS] [FILE...]
```

```
tar -d [-f ARCHIVE] [OPTIONS] [FILE...]
```

```
tar -t [-f ARCHIVE] [OPTIONS] [MEMBER...]
```

```
tar -r [-f ARCHIVE] [OPTIONS] [FILE...]
```

```
tar -u [-f ARCHIVE] [OPTIONS] [FILE...]
```

```
tar -x [-f ARCHIVE] [OPTIONS] [MEMBER...]
```

GNU-style usage

```
tar [-c|catenate|-c|concatenate] [OPTIONS] ARCHIVE ARCHIVE
```

```
tar --create [--file ARCHIVE] [OPTIONS] [FILE...]
```

```
tar (-d|diff)--compare [-f file ARCHIVE] [OPTIONS] [FILE...]
```

```
tar --delete [-f file ARCHIVE] [OPTIONS] [MEMBER...]
```

```
tar --append [-f ARCHIVE] [OPTIONS] [FILE...]
```

```
tar --list [-f ARCHIVE] [OPTIONS] [MEMBER...]
```

```
tar --test-label [-f file ARCHIVE] [OPTIONS] [LABEL...]
```

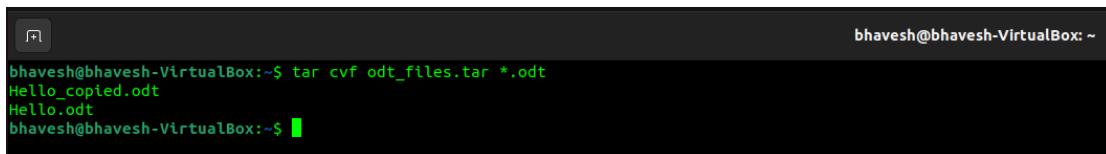
```
tar --update [-f file ARCHIVE] [OPTIONS] [FILE...]
```

```
tar --update [-f ARCHIVE] [OPTIONS] [FILE...]
```

```
tar (-e|extract)--get [-f ARCHIVE] [OPTIONS] [MEMBER...]
```

**NOTE** This manpage is a short description of **GNU tar**. For a detailed discussion, including examples and usage recommendations, refer to the **GNU Tar Manual** available in **texinfo** format. If the **Manual** page (**tar(1)**) line is pressed **h** for help or **q** to quit.

```
$ tar
```



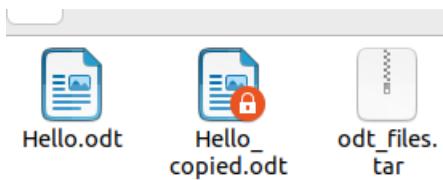
The screenshot shows a terminal window with the command "tar cvf odt\_files.tar \*.odt" being run. The output shows two files being added to the archive: "Hello\_copied.odt" and "Hello.odt". The prompt "bhavesh@bhavesh-VirtualBox:~\$" is visible at the bottom.

```
bhavesh@bhavesh-VirtualBox:~$ tar cvf odt_files.tar *.odt
```

```
Hello_copied.odt
```

```
Hello.odt
```

```
bhavesh@bhavesh-VirtualBox:~$
```



m. cat : This command is used to get the contents of a file on the default output screen and to concatenate files as well.

```
$ man cat
```

```
bhavesh@bhavesh-VirtualBox: ~
```

CAT(1)

**NAME**

cat - concatenate files and print on the standard output

**SYNOPSIS**

```
cat [OPTION]... [FILE]...
```

**DESCRIPTION**

Concatenate FILE(s) to standard output.

With no FILE, or when FILE is -, read standard input.

-A, --show-all  
 equivalent to -vET

-B, --number-nonblank  
 number nonempty output lines, overrides -m

-E  
 equivalent to -vE

-E, --show-ends  
 display \$ at end of each line

-n, --number  
 number all output lines

-s, --squeeze-blank  
 suppress repeated empty output lines

-T  
 equivalent to -vT

-T, --show-tabs  
 display TAB characters as ^I

-u  
 (ignored)

-V, --show-nonprinting  
 use ^ and M- notation, except for LFD and TAB

--help display this help and exit

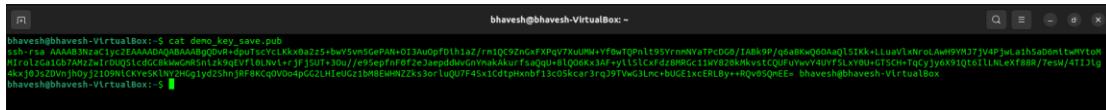
--version  
 output version information and exit

**EXAMPLES**

```
cat F > g
```

[Manuel Kampf cat(1) line 1 (press h for help or q to quit)]

```
$ cat
```



```
bhavesh@bhavesh-VirtualBox:~$ cat demo_key_save.pub
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2.2.10 (Ubuntu)
Comment: https://www.gnupg.org

mQIAYDqCkYcBQJzZdXGKfLwvLkxkdxZz5bweYsymSGPwR+G13u0ef0Lb1az/r7oLc3mGfXpq7ZxGm+xFndOpn1ESV/nmVYTP:G047/AB5-BPjqaABwd0d0aQ121xk+Lluu1uHr0SAwHRYH77jV4PjwLat5aDenLswntor
mTt1Pc1G7AMZ2mfrwpt1cdCcAkmcn5nLzkgp18uNvL+rJ35Uf33ou/yeMSapFnRf2e3hepddeGnVnokAkurFenqyv+4t08kx3Af+y115LcxfdszMBMcG13w833mMvistCQfUvwyv4ufLxymu+ct50n+TQc3y3ox91qt01LneExfAmR/7esu/4t13tg
ekx102szDvn7hOy12109n1CkrexSk1nv2Hg1yd2Shn]BF8KcpqVos4pGCC1HiedGz1mBEHNZZks3or1uQU745x1cdtpgxnb13c0Skcar3rq79TwG3Lec+BuCE1xtEB1By++nQvqSQtEE= bhavesh@bhavesh-VirtualBox:~$
```

n. watch : As the name suggests, the watch command is used to execute of a command at regular intervals. By default, the watch command runs every 2 seconds and shows the output full screen by temporarily clearing the output window.

```
$ man watch
```

```
bhavesh@bhavesh-VirtualBox:~
```

BATCH(1) User Commands WATCH(1)

```
NAME
watch - execute a program periodically, showing output fullscreen

SYNOPSIS
watch [options] command

DESCRIPTION
watch runs command repeatedly, displaying its output and errors (the first screenfull). This allows you to watch the program output change over time. By default, command is run every 2 seconds and watch will run until interrupted.

OPTIONS
-d, --differences[=permanent]
    Highlight the differences between successive updates. If the optional permanent argument is specified then watch will show all changes since the first iteration.

-n, --interval seconds
    Specify update interval. The command will not allow quicker than 0.1 second interval, in which the smaller values are converted. Both '.' and ';' work for any locales. The WATCH_INTERVAL environment can be used to persistently set a non-default interval (following the same rules and formatting).

-p, --precision
    Make watch attempt to run command every --interval seconds. Try it with atptime (if present) and notice how the fractional seconds stays (nearly) the same, as opposed to normal mode where they continuously increase.

-t, --no-title
    Turn off the header showing the interval, command, and current time at the top of the display, as well as the following blank line.

-b, --beep
    Beep if command has a non-zero exit.

-e, --errexit
    Freeze updates on command error, and exit after a key press.

-g, --chgexit
    Exit when the output of command changes.

-c, --color
    Interpret ANSI color and style sequences.

-x, --exec
    Pass command to exec(2) instead of sh -c which reduces the need to use extra quoting to get the desired effect.

-w, --no-linemwrap
    Turn off line wrapping. Long lines will be truncated instead of wrapped to the next line.

-h, --help
    Display help text and exit.

Manual page watch(1) line 1 (press h for help or q to quit.)
```

\$ watch

```
bhavesh@bhavesh-VirtualBox:~$ watch echo $$
```

```
bhavesh@bhavesh-VirtualBox:~$
```

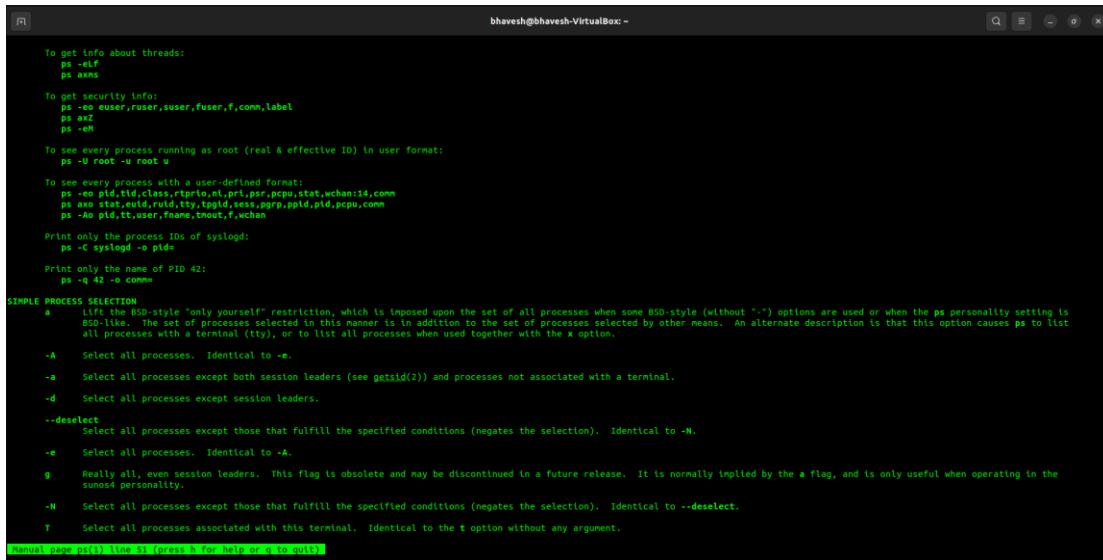
```
bhavesh@bhavesh-VirtualBox:~$ Every 2.0s: echo 2433
```

```
bhavesh@bhavesh-VirtualBox:~$ 2433
```

```
bhavesh@bhavesh-VirtualBox:~$ bhavesh@bhavesh-VirtualBox: Wed Aug 31 18:16:32 2022
```

o. ps : This command is used to get the current status of the program/command running. It also shows other technical information about the processes under execution.

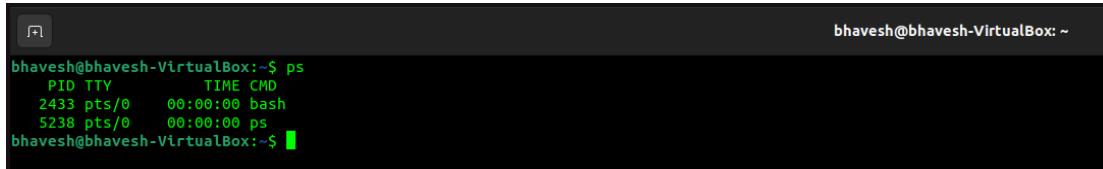
```
$ man ps
```



The screenshot shows a terminal window titled "bhavesh@bhavesh-VirtualBox: ~". The content of the terminal is the man page for the "ps" command. It includes sections for threads, security info, root access, user-defined formats, and process selection options. A detailed description of each option is provided, along with its usage and a note about its compatibility with BSD-style or Linux-style options.

```
To get info about threads:  
ps -elf  
ps axms  
  
To get security info:  
ps -eo euser,ruser,suser,f,comm,label  
ps axz  
ps -m  
  
To see every process running as root (real & effective ID) in user format:  
ps -U root -u root u  
  
To see every process with a user-defined format:  
ps -eo pid,tid,class,rtprio,rl,pri,psr,ppcu,stat,wchan:t4,comm  
ps axo stat,euid,ruid,tty,tgpid,ses,pgrp,pgid,pid,ppcu,comm  
ps -AO pid,tt,user,fnname,tnout,f,wchan  
  
Print only the process IDs of syslogd:  
ps -C syslogd -o pid  
  
Print only the name of PID 42:  
ps -q 42 -o comm  
  
SIMPLE PROCESS SELECTION  
-a      BSD-style "only yourself" restriction, which is imposed upon the set of all processes when some BSD-style (without "-") options are used or when the ps personality setting is  
        BSD-like.  The set of processes selected in this manner is in addition to the set of processes selected by other means.  An alternate description is that this option causes ps to list  
        all processes with a terminal (tty), or to list all processes when used together with the x option.  
-A      Select all processes.  Identical to -e.  
-a      Select all processes except both session leaders (see getsid(2)) and processes not associated with a terminal.  
-d      Select all processes except session leaders.  
--deselect  
       Select all processes except those that fulfill the specified conditions (negates the selection).  Identical to -N.  
-e      Select all processes.  Identical to -A.  
-g      Really all, even session leaders.  This flag is obsolete and may be discontinued in a future release.  It is normally implied by the a flag, and is only useful when operating in the  
        sunos4 personality.  
-N      Select all processes except those that fulfill the specified conditions (negates the selection).  Identical to --deselect.  
-t      Select all processes associated with this terminal.  Identical to the t option without any argument.  
bhavesh@bhavesh-VirtualBox:~$
```

```
$ ps
```



The screenshot shows a terminal window titled "bhavesh@bhavesh-VirtualBox: ~". The command "ps" is run, and the output shows two processes: a bash shell and a ps command itself. The output is formatted with columns for PID, TTY, TIME, and CMD.

```
bhavesh@bhavesh-VirtualBox:~$ ps  
PID TTY      TIME CMD  
2433 pts/0    00:00:00 bash  
5238 pts/0    00:00:00 ps  
bhavesh@bhavesh-VirtualBox:~$
```

p. top : This command clears the output screen temporarily and displays all the linux processes in progress.

```
$ man top
```

```
TOP(1)                               User Commands                             TOP(1)

NAME
    top - display Linux processes

SYNOPSIS
    top [-hv|-bcEMTossi -d secs -n max -wU user -p pids -o field -w [cols]
        The traditional switches '-' and whitespace are optional.

DESCRIPTION
    The top program provides a dynamic real-time view of a running system. It can display system summary information as well as a list of processes or threads currently being managed by the Linux kernel. The types of system summary information shown and the types, order and size of information displayed for processes are all user configurable and that configuration can be made persistent across restarts.

    The program provides a limited interactive interface for process manipulation as well as a much more extensive interface for personal configuration -- encompassing every aspect of its operation. And while top is referred to throughout this document, you are free to name the program anything you wish. That new name, possibly an alias, will then be reflected on top's display and used when reading and writing a configuration file.

OVERVIEW
Documentation
The remaining Table of Contents

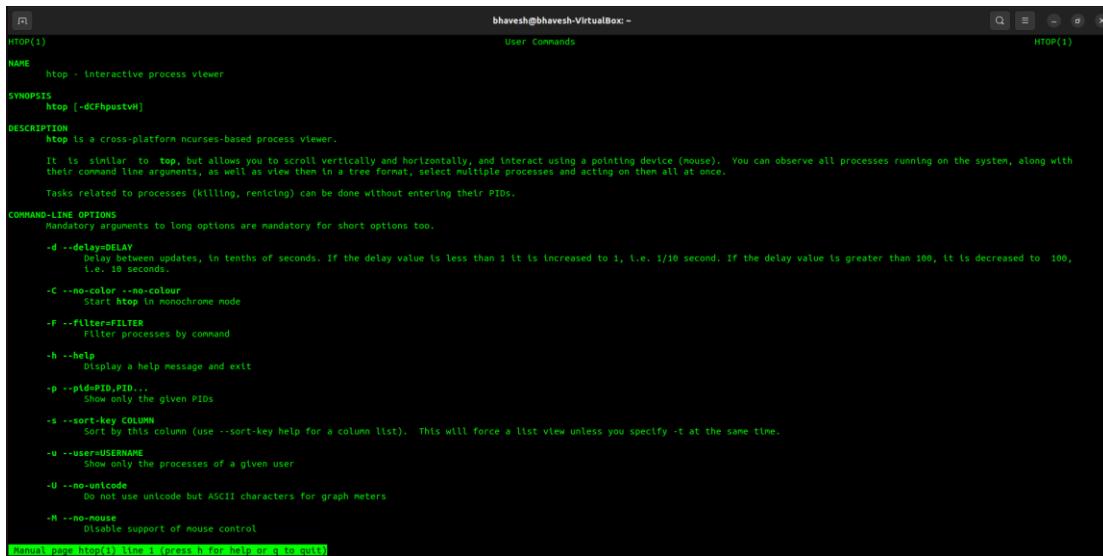
OVERVIEW
    Operation
    Linux Memory Types
    1. COMMAND LINE OPTIONS
    2. SUMMARY DISPLAY
        a. UPTIME and LOAD Averages
        b. TASK and CPU STATES
        c. MEMORY USAGE
    3. FIELDS / Column Display
        a. DESCRIPTIONS of Fields
        b. MANAGING Fields
    4. INTERACTIVE Commands
        a. GLOBAL COMMANDS
        b. SUMMARY AREA Commands
        c. TASK AREA Commands
            1. Appearance
            2. Movement
            3. Size
            4. Sorting
        d. COLOR Mapping
    5. ALTERNATE DISPLAY Provisions
        a. WINDOWS Overview
        b. COMMANDS for Windows
        c. SCROLLING a Window
```

```
$ top
```

PID	USER	PR	NI	VIRT	RSS	SHD	S%CPU	N%MEM	TIME+	COMMAND
1744	bhavesh	20	0	4317320	408436	142752	3.0	10.2	0:00.00	gnome-terminal
2321	bhavesh	20	0	619480	70592	52792	5	0.7	1.8	0:00.38 gnome-terminal
1	root	20	0	166684	11784	8228	5	0.0	0.0	0:01.36 systemd
2	root	20	0	166684	11784	8228	5	0.0	0.0	0:01.36 systemd
3	root	0	0	0	0	0	0.0	0.0	0:00.00 rcu_bh	
4	root	0	0	0	0	0	0.0	0.0	0:00.00 rcu_par_gp	
5	root	0	0	0	0	0	0.0	0.0	0:00.00 netns	
6	root	0	0	0	0	0	0.0	0.0	0:00.00 kworker/0:0-H events_highpri	
7	root	0	0	0	0	0	0.0	0.0	0:00.00 kworker/0:0-H kblockd	
8	root	0	0	0	0	0	0.0	0.0	0:00.00 mn_percpu_wq	
9	root	0	0	0	0	0	0.0	0.0	0:00.00 rcu_tasks_rude	
10	root	0	0	0	0	0	0.0	0.0	0:00.00 rcu_tasks_trace	
11	root	0	0	0	0	0	0.0	0.0	0:00.00 rcu_tasks_trace	
12	root	0	0	0	0	0	0.0	0.0	0:00.00 rcu_tasks_trace	
13	root	0	0	0	0	0	0.0	0.0	0:00.00 rcu_tasks_trace	
14	root	0	0	0	0	0	0.0	0.0	0:02.42 rcu_sched	
15	root	rt	0	0	0	0	0.5	0.0	0:00.07 migration/0	
16	root	-51	0	0	0	0	0.5	0.0	0:00.00 idle_inject/0	
17	root	20	0	0	0	0	0.5	0.0	0:00.00 migration/0	
18	root	20	0	0	0	0	0.5	0.0	0:00.00 cpuhp/1	
19	root	-51	0	0	0	0	0.5	0.0	0:00.00 idle_inject/1	
20	root	-51	0	0	0	0	0.5	0.0	0:00.00 idle_inject/1	
21	root	rt	0	0	0	0	0.5	0.0	0:00.37 migration/1	
22	root	20	0	0	0	0	0.5	0.0	0:00.00 migration/1	
23	root	0	0	0	0	0	0.5	0.0	0:00.00 kworker/1:0-H events_highpri	
24	root	0	0	0	0	0	0.5	0.0	0:00.00 kworker/1:0-H events_highpri	
25	root	20	0	0	0	0	0.5	0.0	0:00.00 kdevtmpfs	
26	root	0	0	0	0	0	0.1	0.0	0:00.00 inet_fq_wq	
27	root	20	0	0	0	0	0.1	0.0	0:00.00 kaudiod	
28	root	20	0	0	0	0	0.1	0.0	0:00.00 kaudiod	
29	root	20	0	0	0	0	0.5	0.0	0:00.00 oom_reaper	
30	root	0	0	0	0	0	0.1	0.0	0:00.00 writeback	
31	root	20	0	0	0	0	0.1	0.0	0:00.00 kcompactd0	
32	root	25	0	0	0	0	0.1	0.0	0:00.00 kcompactd0	
33	root	39	19	0	0	0	0.5	0.0	0:00.00 khugepaged	
60	root	0	0	0	0	0	0.1	0.0	0:00.00 kintegrityd	
81	root	0	0	0	0	0	0.1	0.0	0:00.00 kblockd	
82	root	20	0	0	0	0	0.1	0.0	0:00.00 kmemleak	
83	root	0	0	0	0	0	0.1	0.0	0:00.00 tpm_dev_wq	
84	root	0	0	0	0	0	0.1	0.0	0:00.00 ata_sff	
85	root	0	0	0	0	0	0.1	0.0	0:00.00 md	
86	root	0	0	0	0	0	0.1	0.0	0:00.00 kcompactd0	
87	root	0	0	0	0	0	0.1	0.0	0:00.00 devfreq_wq	
88	root	-51	0	0	0	0	0.5	0.0	0:00.00 watchdogd	
91	root	20	0	0	0	0	0.5	0.0	0:00.00 kswapd0	

q. htop : This command is an advanced version of top which helps to view the processes in an interactive way.

```
$ man htop
```



The screenshot shows a terminal window titled "User Commands" with the title bar "htop(1)". The window displays the man page for htop. The text is as follows:

```
htop(1)
bhavesh@bhavesh-VirtualBox: ~

NAME      htop - interactive process viewer
SYNOPSIS  htop [-dCFhpstvVM]
DESCRIPTION
htop is a cross-platform ncurses-based process viewer.
It is similar to top, but allows you to scroll vertically and horizontally, and interact using a pointing device (mouse). You can observe all processes running on the system, along with their command line arguments, as well as view them in a tree format, select multiple processes and act on them all at once.
Tasks related to processes (killing, renicing) can be done without entering their PIDs.

COMMAND-LINE OPTIONS
Mandatory arguments to long options are mandatory for short options too.
-d --delay=DELAY
    Delay between updates, in tenths of seconds. If the delay value is less than 1 it is increased to 1, i.e. 1/10 second. If the delay value is greater than 100, it is decreased to 100, i.e. 10 seconds.
-C --no-color --no-colour
    Start htop in monochrome mode
-f --filter=FILTER
    Filter processes by command
-h --help
    Display a help message and exit
-p --pid=PID,PID...
    Show only the given PIDs
--sort-key=COLUMN
    Sort by this column (use --sort-key help for a column list). This will force a list view unless you specify -t at the same time.
-u --user=USERNAME
    Show only the processes of a given user
-U --no-unicode
    Do not use unicode but ASCII characters for graph meters
-M --no-mouse
    Disable support of mouse control
[Manual page htop() line 1 (press h for help or q to quit)]
```

```
$ htop -u bhavesh
```



The screenshot shows a terminal window with the title bar "bhavesh@bhavesh-VirtualBox: ~". The text is as follows:

```
bhavesh@bhavesh-VirtualBox:~$ htop -u bhavesh
bhavesh@bhavesh-VirtualBox:~$
```

This screenshot shows a terminal window with a system monitor at the top. The monitor displays CPU usage (2.0%), tasks (122, 265 I/O), load average (0.00, 0.11, 0.07), memory usage (887M/3.83G), and uptime (02:53:47). Below the monitor is a list of processes from the tasklist command. The processes include various system daemons like dhclient, gdm, and libexec, along with user applications like httpd and gnome-shell. The list is sorted by CPU usage.

```

bhavesh@bhavesh-VirtualBox: ~
2.0% Tasks: 122, 265 I/O: 2 running
1.3% Load average: 0.00 0.11 0.07
887M/3.83G Uptime: 02:53:47
0K/2.62G

PID USER PRI NI VIRT RES SHR S CPU%SWCPU% TIME+ Command
5598 bhavesh 20 0 20292 5232 3608 R 1.3 0.1 0:00:11 httpd -u bhavesh
1744 bhavesh 20 0 4216M 398M 1398 S 0.7 10.2 5:46:19 /usr/bin/gnome-shell
1769 bhavesh 20 0 4216M 398M 1398 S 0.7 10.2 2:16:12 /usr/bin/gnome-shell
1771 bhavesh 20 0 4216M 398M 1398 S 0.7 10.2 2:16:12 /usr/bin/gnome-shell
1999 bhavesh 20 0 158M 2588 2196 S 0.7 0.1 0:21:93 /usr/bin/vboxclient --dragandrop
2227 bhavesh 20 0 158M 2588 2196 R 0.7 0.1 0:21:93 /usr/bin/vboxclient --dragandrop
2313 bhavesh 20 0 66M 78280 52732 S 0.7 1.7 0:31:75 /usr/libexec/gnome-terminal-server
1588 bhavesh 20 0 39M 78280 7908 S 0.7 1.7 0:31:75 /usr/libexec/gnome-terminal-systemd - user
1589 bhavesh 20 0 165M 4100 16 S 0.0 0.1 0:00:00 (idle)
1595 bhavesh 9 -11 48228 6492 5368 S 0.0 0.2 0:00:02 /usr/bin/pipewire
1596 bhavesh 20 0 32188 6536 5440 S 0.0 0.2 0:00:01 /usr/bin/pipewire-media-session
1597 bhavesh 9 -11 48228 36596 5440 S 0.0 0.2 0:00:01 /usr/bin/pulseaudio-remap-sessions --log-target=journal
1598 bhavesh 20 0 301M 6536 15960 S 0.0 0.2 0:00:57 /usr/libexec/gdm3-snapd-integration/14/bin/snapd-decktop-integration
1606 bhavesh -21 0 32188 6536 5440 S 0.0 0.2 0:00:00 /usr/bin/pipewire-media-session
1610 bhavesh 20 0 10804 7808 4104 S 0.0 0.2 0:01:00 /usr/bin/dbus-daemon -session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
1612 bhavesh 20 0 243M 8252 7272 S 0.0 0.2 0:00:03 /usr/libexec/gvfsd
1615 bhavesh 20 0 239M 8252 7272 S 0.0 0.2 0:00:03 /usr/libexec/gvfsd
1617 bhavesh 20 0 243M 8252 7272 S 0.0 0.2 0:00:02 /usr/libexec/gvfsd
1618 bhavesh 20 0 532M 7808 6308 S 0.0 0.2 0:00:00 /usr/libexec/dg-document-portal
1619 bhavesh 20 0 371M 6412 5796 S 0.0 0.2 0:00:00 /usr/libexec/gvfs-fuse /run/user/1000/gvfs -f
1620 bhavesh 20 0 371M 6412 5796 S 0.0 0.2 0:00:00 /usr/libexec/gvfs-fuse /run/user/1000/gvfs -f
1625 bhavesh 20 0 371M 6412 5796 S 0.0 0.2 0:00:00 /usr/libexec/gvfs-fuse /run/user/1000/gvfs -f
1626 bhavesh 20 0 371M 6412 5796 S 0.0 0.2 0:00:00 /usr/libexec/gvfs-fuse /run/user/1000/gvfs -f
1627 bhavesh 20 0 371M 6412 5796 S 0.0 0.2 0:00:00 /usr/libexec/gvfs-fuse /run/user/1000/gvfs -f
1628 bhavesh 20 0 371M 6412 5796 S 0.0 0.2 0:00:00 /usr/libexec/gvfs-fuse /run/user/1000/gvfs -f
1629 bhavesh 20 0 532M 7808 6308 S 0.0 0.2 0:00:00 /usr/libexec/dg-document-portal
1630 bhavesh 20 0 532M 7808 6308 S 0.0 0.2 0:00:05 /usr/libexec/dg-document-portal
1631 bhavesh 20 0 239M 5644 5232 S 0.0 0.1 0:00:00 /usr/libexec/dg_permission-store
1632 bhavesh 20 0 239M 5644 5232 S 0.0 0.1 0:00:00 /usr/libexec/dg_permission-store
1634 bhavesh 20 0 239M 5644 5232 S 0.0 0.1 0:00:00 /usr/libexec/dg_permission-store
1635 bhavesh 20 0 532M 7808 6308 S 0.0 0.2 0:00:00 /usr/libexec/dg-document-portal
1636 bhavesh 20 0 532M 7808 6308 S 0.0 0.2 0:00:00 /usr/libexec/dg-document-portal
1647 bhavesh 20 0 125M 7436 6500 S 0.0 0.2 0:00:00 /usr/bin/gnome-keyring-daemon --daemonize --login
1650 bhavesh 20 0 243M 7436 6500 S 0.0 0.2 0:00:00 /usr/bin/gnome-keyring-daemon --daemonize --login
1651 bhavesh 20 0 243M 7436 6500 S 0.0 0.2 0:00:00 /usr/bin/gnome-keyring-daemon --daemonize --login
1652 bhavesh 20 0 243M 7436 6500 S 0.0 0.2 0:00:00 /usr/bin/gnome-keyring-daemon --daemonize --login
1653 bhavesh 20 0 167M 6256 5772 S 0.0 0.2 0:00:00 /usr/libexec/gdm-wayland-session env GNOME_SHELL_SESSION_MODE=ubuntu /usr/bin/gnome-session --sessions=ubuntu
1657 bhavesh 20 0 167M 6256 5772 S 0.0 0.2 0:00:00 /usr/libexec/gdm-wayland-session env GNOME_SHELL_SESSION_MODE=ubuntu /usr/bin/gnome-session --sessions=ubuntu
1658 bhavesh 20 0 167M 6256 5772 S 0.0 0.2 0:00:00 /usr/libexec/gdm-wayland-session env GNOME_SHELL_SESSION_MODE=ubuntu /usr/bin/gnome-session --sessions=ubuntu

```

r. **gcc** : This command is a command line utility for compiling GNU C and C++ scripts. It does all the pre-processing, compiling, assembly, and linking when it is instantiated.

\$ man gcc

This screenshot shows a terminal window displaying the man page for the gcc command. The page is titled "GCC(1)" and "GNU". It starts with a brief description of what GCC is and how it works. The SYNOPSIS section shows the basic command line syntax. The DESCRIPTION section provides detailed information about the options and how they affect the compilation process. It also includes notes on memory units and option summaries. The OPTIONS section lists all available options with their descriptions.

```

bhavesh@bhavesh-VirtualBox: ~
GCC(1)                                     GNU
GCC(1)

NAME
   gcc - GNU project C and C++ compiler

SYNOPSIS
   gcc [-c|-S|-E] [std=standard]
      [-O] [-pg] [-Olevel]
      [-Werror...], [-Wpedantic]
      [-fPIC,-fPIE,-fPIE,-fPIE]
      [-fstack-protector-strong] [-fmacro]
      [-foption...] [-fmachine-option...]
      [-foutline] [file] [file]...
      Only the most useful options are listed here; see below for the remainder.  g++ accepts mostly the same options as gcc.

DESCRIPTION
   When you invoke GCC, it normally does preprocessing, compilation, assembly and linking.  The "overall options" allow you to stop this process at an intermediate stage. For example, the -c option says not to run the linker. Then the output consists of object files output by the assembler.

   Other options are passed on to one or more stages of processing. Some options control the preprocessor and others the compiler itself. Yet other options control the assembler and linker; most of these are not documented here, since you rarely need to use any of them.

   Most of the command-line options that you can use with GCC are useful for C programs; when an option is only useful with another language (usually C++), the explanation says so explicitly. If the description for a particular option does not mention a source language, you can use that option with all supported languages.

   The usual way to run GCC is to run the executable called gcc, or machine-gcc when cross-compiling, or machine-gcc-version to run a specific version of GCC. When you compile C++ programs, you should invoke GCC as g++ instead.

   The gcc program accepts options and file names as operands. Many options have multi-letter names; therefore multiple single-letter options may not be grouped: -dv is very different from -d -v.

   You can mix options and other arguments. For the most part, the order you use doesn't matter. Order does matter when you use several options of the same kind; for example, if you specify -l more than once, the directories are searched in the order specified. Also, the placement of the -l option is significant.

   Many options have long names starting with -f or with -W—for example, -fmove-loop-variants, -Wformat and so on. Most of these have both positive and negative forms; the negative form of -ffoo is -fno-foo. This manual documents only one of these two forms, whichever one is not the default.

   Some options take one or more arguments typically separated either by a space or by the equals sign (=) from the option name. Unless documented otherwise, an argument can be either numeric or string. Numeric arguments must be decimal, binary, octal, or hexadecimal integers. String arguments must be strings of characters, preceded by a double quote. Arguments that do not specify a size must be terminated by a null byte. Following a string argument, you can specify a size suffix designating a multiple of bytes such as "4w" for four words, "4k" for kilobyte, respectively, "1M" and "1M" for megabyte and mebibyte, "1G" and "1G" for gigabyte and gibibyte, and so on. Such arguments are designated by byte-size in the following text. Refer to the NIST, IEC, and other relevant national and international standards for the full listing and explanation of the binary and decimal byte size prefixes.

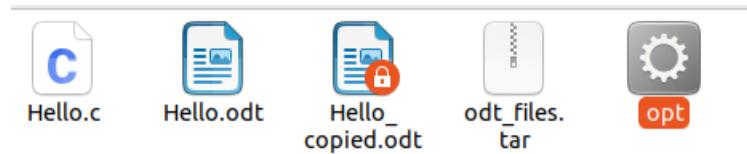
OPTIONS
   Option Summary
      Here is a summary of all the options, grouped by type. Explanations are in the following sections.
   Manual page (gcc(1)) for help or gcc --help.

```

```
$ gcc
```



```
bhavesh@bhavesh-VirtualBox:~$ gcc Hello.c -o opt
bhavesh@bhavesh-VirtualBox:~$
```



s. tail : This command is used to get the last entries/content of a file. We can use multiple options with this command to specify which entries at the end we are looking for.

```
$ man tail
```

```
tail(1)                                User Commands                               TAIL(1)

NAME
    tail - output the last part of files

SYNOPSIS
    tail [OPTION]... [FILE]...

DESCRIPTION
    Print the last 10 lines of each FILE to standard output.  With more than one FILE, precede each with a header giving the file name.

    With no FILE, or when FILE is -, read standard input.

    Mandatory arguments to long options are mandatory for short options too.

-C, --bytes[=NUM]
        output the last NUM bytes; or use -c +NUM to output starting with byte NUM of each file

-F, --follow[=(name|descriptor)]
        output appended data as the file grows;
        an absent option argument means 'descriptor'

-P     same as --follow=name --retry

-N, --lines[=NUM]
        output the last NUM lines, instead of the last 10; or use -n +NUM to output starting with line NUM

--max-unchanged-stats=N
        with --follow=name, reopen a FILE which has not
        changed size after N (default 5) iterations to see if it has been unlinked or renamed (this is the usual case of rotated log files); with inotify, this option is rarely useful

--pid=PID
        with -f, terminate after process ID, PID dies

-q, --quiet, --silent
        never output headers giving file names

--retry
        keep trying to open a file if it is inaccessible

-s, --sleep-interval=N
        with -f, sleep for approximately N seconds (default 1.0) between iterations; with inotify and --pid=0, check process P at least once every N seconds

-v, --verbose
        always output headers giving file names

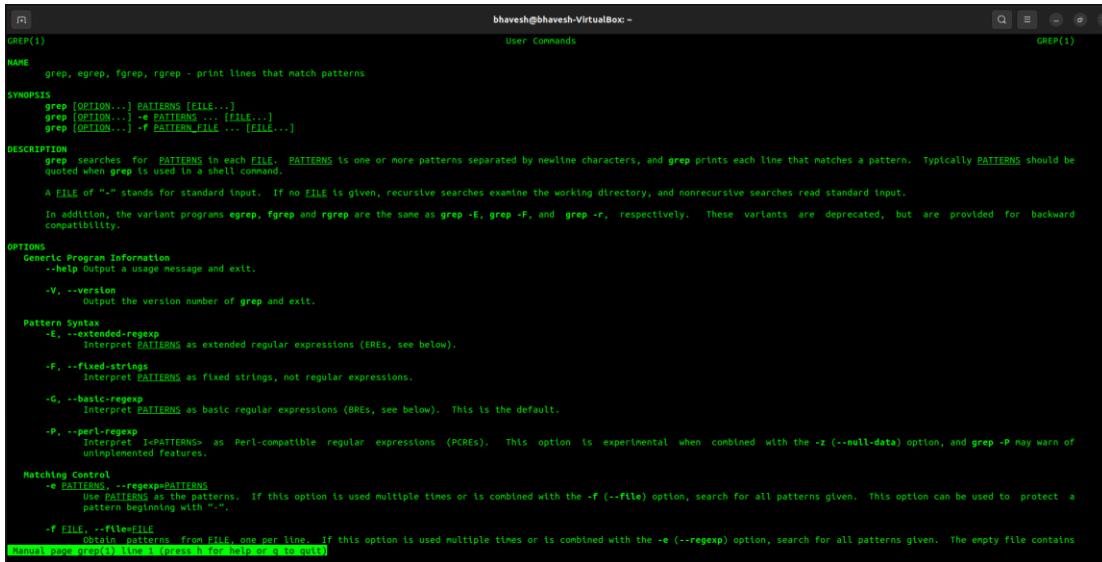
Manual page tail(1) (see 3 (ress.h for help on q to quit))
```

\$ tail

```
bhavesh@bhavesh-VirtualBox:~$ man tail
bhavesh@bhavesh-VirtualBox:~$ tail -n 2 Hello.c
printf("Hello World");
}
bhavesh@bhavesh-VirtualBox:~$
```

t. grep : This command is used to find data in a file by using patterns.

```
$ man grep
```



The screenshot shows a terminal window titled "User Commands" with the title bar "GREP(1)". The content of the window is the man page for the grep command. It includes sections for SYNOPSIS, DESCRIPTION, OPTIONS, and Pattern Syntax. The SYNOPSIS section shows various command-line options like -e, -f, and -r. The DESCRIPTION section explains what grep does and how it handles patterns. The OPTIONS section lists flags for help, version, and different regex engines. The Pattern Syntax section details the ERE, BRE, and PCRE engines. The man page ends with a note about the empty file containing the command itself.

```
grep(1)
User Commands
GREP(1)

NAME
    grep, egrep, fgrep, rgrep - print lines that match patterns

SYNOPSIS
    grep [OPTION...] PATTERN [FILE...]
    grep [OPTION...] -e PATTERN [FILE...]
    grep [OPTION...] -f PATTERNFILE ... [FILE...]

DESCRIPTION
    grep searches for PATTERN in each FILE. PATTERN is one or more patterns separated by newline characters, and grep prints each line that matches a pattern. Typically PATTERN should be quoted when grep is used in a shell command.

    A FILE of "-" stands for standard input. If no FILE is given, recursive searches examine the working directory, and nonrecursive searches read standard input.

    In addition, the variant programs egrep, fgrep and rgrep are the same as grep -E, grep -F, and grep -R, respectively. These variants are deprecated, but are provided for backward compatibility.

OPTIONS
    Generic Program Information
        --help Output a usage message and exit.
        -v, --version          Output the version number of grep and exit.

    Pattern Syntax
        -E, --extended-regexp
            Interpret PATTERN as extended regular expressions (EREs, see below).
        -F, --fixed-strings
            Interpret PATTERN as fixed strings, not regular expressions.
        -G, --basic-regexp
            Interpret PATTERN as basic regular expressions (BREs, see below). This is the default.
        -P, --perl-regexp
            Interpret PATTERN as Perl-compatible regular expressions (PCREs). This option is experimental when combined with the -z (--null-data) option, and grep -P may warn of unimplemented features.

    Matching Control
        -e PATTERN --regexp=PATTERN
            Use PATTERN as the patterns. If this option is used multiple times or is combined with the -f (--file) option, search for all patterns given. This option can be used to protect a pattern beginning with "-".
        -f FILE, --file=FILE
            Obtain patterns from FILE, one per line. If this option is used multiple times or is combined with the -e (--regexp) option, search for all patterns given. The empty file contains

grep(1) man(7) line 1 (or less) for help on grep(1)
```

```
$ grep
```



The screenshot shows a terminal window with the title "bhavesh@bhavesh-VirtualBox: ~". The user has run the command "grep Hi file\_grep.txt". The output shows four lines of text starting with "Hi", indicating that the pattern "Hi" was found in the file "file\_grep.txt".

```
bhavesh@bhavesh-VirtualBox:~$ grep Hi file_grep.txt
Hi
Hi
Hi
Hi*10
bhavesh@bhavesh-VirtualBox:~$
```

u. kill : Kill command transmits a specified signal to a specified process/process groups. This signal can vary from terminate, stop, continue, etc. By default, the signal for kill command is to terminate.

```
$ man kill
```

```
bhavesh@bhavesh-VirtualBox: ~ User Commands KILL(1)
KILL(1)
NAME      kill - send a signal to a process
SYNOPSIS  kill [options] <pid> [...]
DESCRIPTION
The default signal for kill is TERM. Use -l or -L to list available signals. Particularly useful signals include HUP, INT, KILL, STOP, CONT, and QUIT. Alternate signals may be specified in three ways: +n, -SIGnkill or -KILL. Negative PID values may be used to choose whole process groups; see the PGID column in ps command output. A PID of -1 is special; it indicates all processes except the kill process itself and init.

OPTIONS
<pid> [...]
    Send signal to every <pid> listed.

--signals
--signal
--signal extname
    Specify the signal to be sent. The signal can be specified by using name or number. The behavior of signals is explained in signal(7) manual page.

-Q, --queue value
    Use queue(3) rather than kill(2) and the value argument is used to specify an integer to be sent with the signal. If the receiving process has installed a handler for this signal using the SA_SIGINFO flag to sigaction(2), then it can obtain this data via the si_value field of the siginfo_t structure.

-L, --list [signal]
    List signal names. This option has optional argument, which will convert signal number to signal name, or other way round.

--table
    List signal names in a nice table.

NOTES  Your shell (command line interpreter) may have a built-in kill command. You may need to run the command described here as /bin/kill to solve the conflict.

EXAMPLES
kill -9 -1
    Kill all processes you can kill.

kill -1 11
    Translate number 11 into a signal name.

kill -L
    List the available signal choices in a nice table.

kill 123 543 2341 3453
    Send the default signal, SIGTERM, to all those processes.

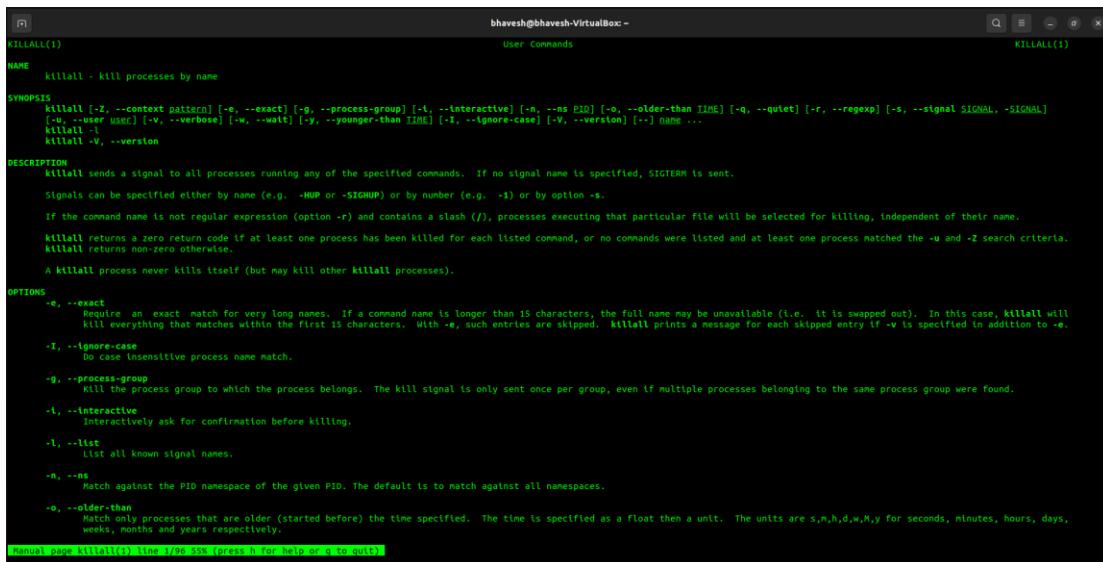
Manual page kill(1) Sim X (press h for help or q to quit)
```

```
$ kill
```

```
bhavesh@bhavesh-VirtualBox: ~ kill -L
1) SIGHUP      2) SIGINT      3) SIGQUIT      4) SIGILL      5) SIGTRAP
6) SIGABRT     7) SIGBUS      8) SIGFPE       9) SIGKILL     10) SIGUSR1
11) SIGSEGV     12) SIGUSR2     13) SIGPIPE     14) SIGALRM     15) SIGTERM
16) SIGSTKFLT   17) SIGCHLD     18) SIGCONT     19) SIGSTOP     20) SIGTSTP
21) SIGTTIN     22) SIGTTOU     23) SIGURG      24) SIGXCPU     25) SIGXFSSZ
26) SIGVTALRM   27) SIGPROF     28) SIGWINCH    29) SIGIO       30) SIGPWR
31) SIGSYS      34) SIGRTMIN   35) SIGRTMIN+1  36) SIGRTMIN+2  37) SIGRTMIN+3
38) SIGRTMIN+4  39) SIGRTMIN+5  40) SIGRTMIN+6  41) SIGRTMIN+7  42) SIGRTMIN+8
43) SIGRTMIN+9  44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7
58) SIGRTMAX-6  59) SIGRTMAX-5  60) SIGRTMAX-4  61) SIGRTMAX-3  62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX
bhavesh@bhavesh-VirtualBox: ~
```

v. killall : This command as the name justifies kills all processes except itself by sending a signal. If the command has killed even one process it will return a zero value. If no process was terminated, it returns nothing.

```
$ man killall
```



```
bhavesh@bhavesh-VirtualBox: ~
User Commands
KILLALL(1)

NAME
    killall - kill processes by name

SYNOPSIS
    killall [-Z, --context pattern] [-e, --exact] [-g, --process-group] [-l, --interactive] [-n, --ns PID] [-o, --older-than TIME] [-q, --quiet] [-r, --regexp] [-s, --signal SIGNAL, -SIGNAL]
    [-u, --user user] [-v, --verbose] [-w, --wait] [-y, --younger-than TIME] [-i, --ignore-case] [-V, --version] [-d] name ...
    killall -l
    killall -V, --version

DESCRIPTION
    Killall sends a signal to all processes running any of the specified commands. If no signal name is specified, SIGTERM is sent.
    Signals can be specified either by name (e.g. -HUP or -SIGNUM) or by number (e.g. -1) or by option -s.
    If the command name is not regular expression (option -r) and contains a slash (/), processes executing that particular file will be selected for killing, independent of their name.
    Killall returns a zero return code if at least one process has been killed for each listed command, or no commands were listed and at least one process matched the -u and -Z search criteria.
    Killall returns non-zero otherwise.
    A killall process never kills itself (but may kill other killall processes).

OPTIONS
    -e, --exact
        Require an exact match for very long names. If a command name is longer than 15 characters, the full name may be unavailable (i.e. it is swapped out). In this case, killall will
        kill everything that matches within the first 15 characters. With -e, such entries are skipped. Killall prints a message for each skipped entry if -v is specified in addition to -e.
    -I, --ignore-case
        Do case insensitive process name match.

    -g, --process-group
        Kill the process group to which the process belongs. The kill signal is only sent once per group, even if multiple processes belonging to the same process group were found.

    -l, --interactive
        Interactively ask for confirmation before killing.

    -l, --list
        List all known signal names.

    -n, --ns
        Match against the PID namespace of the given PID. The default is to match against all namespaces.

    -o, --older-than
        Match only processes that are older (started before) the time specified. The time is specified as a float then a unit. The units are s,m,h,d,w,M,y for seconds, minutes, hours, days,
        weeks, months and years respectively.

    -q, --quiet
        Don't output any messages to standard error.

    -r, --regexp
        Use regular expressions to match command names. This is the default behavior.

    -s, --signal
        Signal to send to the processes. It can be specified by name (SIGTERM, SIGKILL, etc.) or by number (1, 2, etc.).
```

```
$ killall
```

```
[+]
bhavesh@bhavesh-VirtualBox:~$ killall -V
killall (PSmisc) 23.4
Copyright (C) 1993-2021 Werner Almesberger and Craig Small

PSmisc comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it under
the terms of the GNU General Public License.
For more information about these matters, see the files named COPYING.
bhavesh@bhavesh-VirtualBox:~$
```

w. du : This command is used to provide an estimate of size that a file will use.

```
$ man du
```

bhavesh@bhavesh-VirtualBox: ~

User Commands

DU(1)

```
NAME
    du - estimate file space usage

SYNOPSIS
    du [OPTION]... [FILE]...
    du [OPTION]... --files-from=FILE

DESCRIPTION
    Summarize disk usage of the set of FILES, recursively for directories.
    Mandatory arguments to long options are mandatory for short options too.

-B, --null
    end each output line with NUL, not newline

-a, --all
    write counts for all files, not just directories

--apparent-size
    print apparent sizes, rather than disk usage; although the apparent size is usually smaller, it may be larger due to holes in ('sparse') files, internal fragmentation, indirect
    blocks, or the like

-B, --block-size=SIZE
    scale sizes by SIZE before printing them; e.g., '-B1M' prints sizes in units of 1,048,576 bytes; see SIZE format below

-B, --bytes
    equivalent to '--apparent-size --block-size=1'

-c, --total
    produce a grand total

-d, --dereference-args
    dereference only symlinks that are listed on the command line

-d, --max-depth=N
    print the total for a directory (or file, with --all) only if it is N or fewer levels below the command line argument; --max-depth=0 is the same as --summarize

--files-from=FILE
    summarize disk usage of the NUL-terminated file names specified in file FILE; if FILE is -, then read names from standard input

-H     equivalent to --dereference-args (-B)

-H, --human-readable
    print sizes in human readable format (e.g., 1K 234M 2G)

Manual page du(1) line 1 (press h for help or q to quit)
```

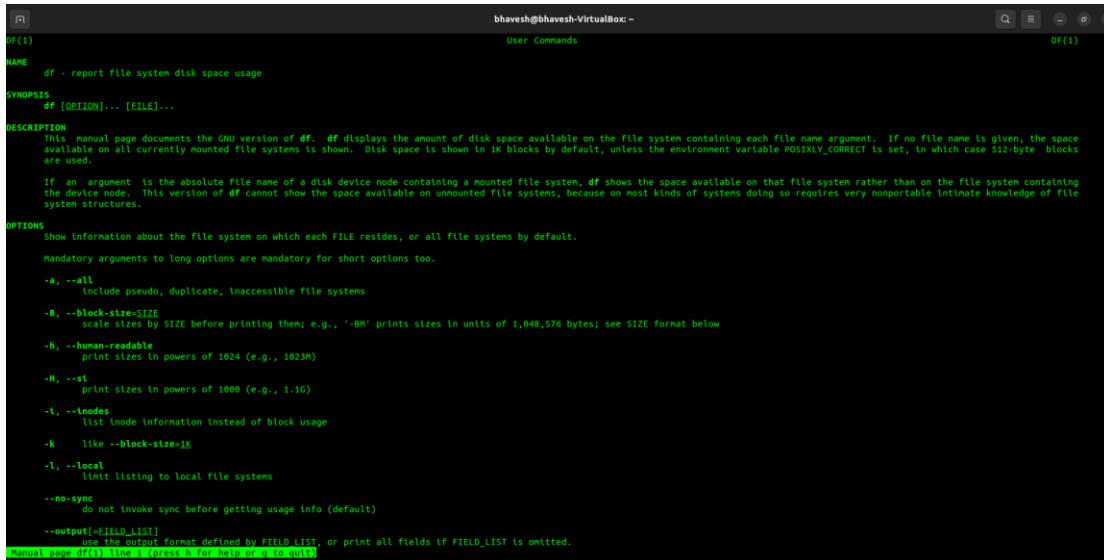
\$ du

bhavesh@bhavesh-VirtualBox: ~

```
bhavesh@bhavesh-VirtualBox:~$ du -b Hello.c
58    Hello.c
bhavesh@bhavesh-VirtualBox:~$
```

x. df : This command is used to display the free disc space of a specific file system and also the disk space usage.

```
$ man df
```



The screenshot shows a terminal window with the title "User Commands" and the command "df(1)". The content of the window is the man page for the "df" command. It includes sections for NAME, SYNOPSIS, DESCRIPTION, and OPTIONS. The DESCRIPTION section provides a detailed explanation of what "df" does, mentioning it displays disk space usage for mounted file systems. The OPTIONS section lists various flags and their meanings, such as -h for human-readable output and -l for listing local file systems.

```
NAME
    df - report file system disk space usage

SYNOPSIS
    df [OPTION]... [FILE]...

DESCRIPTION
    This manual page documents the GNU version of df. df displays the amount of disk space available on the file system containing each file name argument. If no file name is given, the space available on all currently mounted file systems is shown. Disk space is shown in 1K blocks by default, unless the environment variable POSIXLY_CORRECT is set, in which case 512-byte blocks are used.

    If an argument is the absolute file name of a disk device node containing a mounted file system, df shows the space available on that file system rather than on the file system containing the device node. This version of df cannot show the space available on unmounted file systems, because on most kinds of systems doing so requires very nonportable intimate knowledge of file system structures.

OPTIONS
    Show information about the file system on which each FILE resides, or all file systems by default.

    Mandatory arguments to long options are mandatory for short options too.

    -a, --all
        include pseudo, duplicate, inaccessible file systems

    -B, --block-size=SIZE
        scale sizes by SIZE before printing them; e.g., '-BM' prints sizes in units of 1,048,576 bytes; see SIZE format below

    -h, --human-readable
        print sizes in powers of 1024 (e.g., 1023M)

    -k, --kbytes
        print sizes in powers of 1000 (e.g., 1.1G)

    -l, --inodes
        list inode information instead of block usage

    -K      like --block-size=1K

    -L, --local
        limit listing to local file systems

    --no-sync
        do not invoke sync before getting usage info (default)

    --output[=FIELD_LIST]
        use the output format defined by FIELD_LIST, or print all fields if FIELD_LIST is omitted.

Manual pages for df(1) also exist; press h for help or q to quit.
```

```
$ df
```

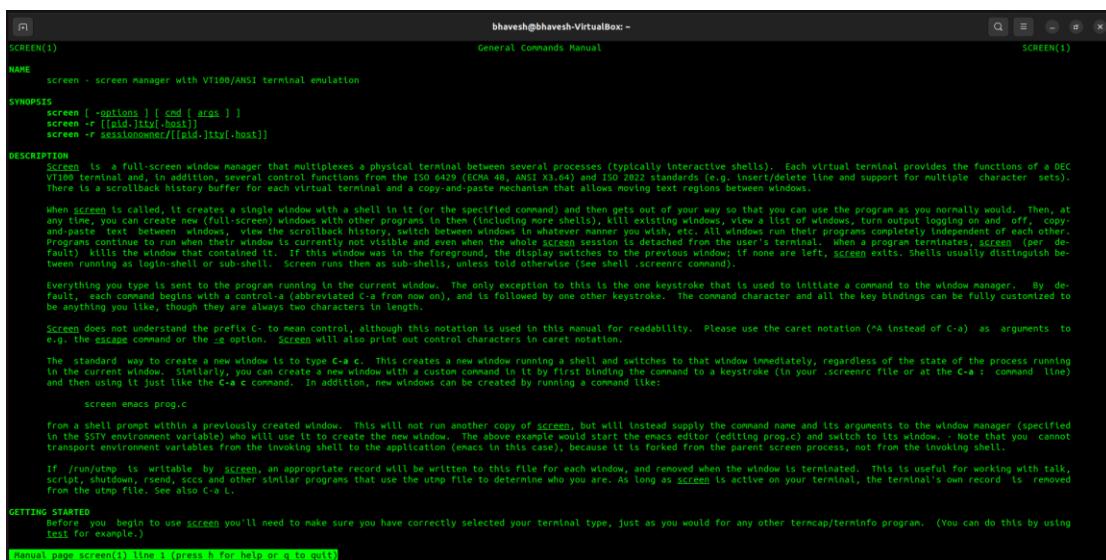


The screenshot shows a terminal window with the title "User Commands" and the command "bhavesh@bhavesh-VirtualBox:~". The content of the window is the output of the "df" command, which displays disk space usage for various file systems. The output is tabular, showing columns for Filesystem, 1K-blocks, Used, Available, Use%, and Mounted on. The file systems listed include /tmpfs, /dev/sda3, /dev/sda2, /tmpfs, /dev/sr0, and /dev/sr0 again. The last entry is a blank line.

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
/tmpfs	401824	1468	400356	1%	/run
/dev/sda3	25106692	12012600	11793408	51%	/
/tmpfs	2009104	0	2009104	0%	/dev/shm
/tmpfs	5120	4	5116	1%	/run/lock
/dev/sda2	524252	5364	518888	2%	/boot/efi
/tmpfs	401820	2424	399396	1%	/run/user/1000
/dev/sr0	62308	62308	0	100%	/media/bhavesh/VBox_GAs_6.1.36

y. screen : This command is used to detach the current screen and run it in the background. Using various options, the screen in the background can be reattached or viewed.

```
$ man screen
```



The screenshot shows a terminal window with the title "bhavesh@bhavesh-VirtualBox:~". The window contains the "General Commands Manual" for the "SCREEN(1)" command. The text is as follows:

```
screen - screen manager with VT100/ANSI terminal emulation

SYNOPSIS
screen [ -options ] [ cmd [ args ] ]
screen -r [[pid.]tty[.host]]
screen -r sessionname/[pid.]tty[.host]

DESCRIPTION
Screen is a full-screen window manager that multiplexes a physical terminal between several processes (typically interactive shells). Each virtual terminal provides the functions of a DEC VT100 terminal and, in addition, several control functions from the ISO 6429 (ECMA 48, ANSI X3.64) and ISO 2022 standards (e.g. insert/delete line and support for multiple character sets). There is a scrollback history buffer for each virtual terminal and a copy-and-paste mechanism that allows moving text regions between windows.

When screen is called, it creates a single window with a shell in it (or the specified command) and then gets out of your way so that you can use the program as you normally would. Then, at any time, you can create new (full-screen) windows with other programs in them (including more shells), kill existing windows, view a list of windows, turn output logging on and off, copy-and-paste text between windows, view the scrollback history, switch between windows in whatever manner you wish, etc. All windows run their programs completely independent of each other. You can continue to interact with any particular window as if it were the only window on the system. When you exit from a window, screen kills the process running in it (by default) kills the window that contained it. If this window was in the foreground, the display switches to the previous window; if none are left, screen exits. Shells usually distinguish between running as login-shell or as sub-shell. Screen runs them as sub-shells, unless told otherwise (See shell .screenrc command).

Everything you type is sent to the program running in the current window. The only exception to this is the one keystroke that is used to initiate a command to the window manager. By default, each command begins with a control-a (abbreviated C-a from now on), and is followed by one other keystroke. The command character and all the key bindings can be fully customized to be anything you like, though they are always two characters in length.

Screen does not understand the prefix C- to mean control, although this notation is used in this manual for readability. Please use the caret notation (^A instead of C-a) as arguments to e.g. the escape command or the -z option. Screen will also print out control characters in caret notation.

The standard way to create a new window is to type C-a c. This creates a new window running a shell and switches to that window immediately, regardless of the state of the process running in the current window. Similarly, you can create a new window with a custom command in it by first binding the command to a keystroke (in your .screenrc file or at the C-a : command line) and then using it just like the C-a c command. In addition, new windows can be created by running a command like:

    screen emacs prog.c

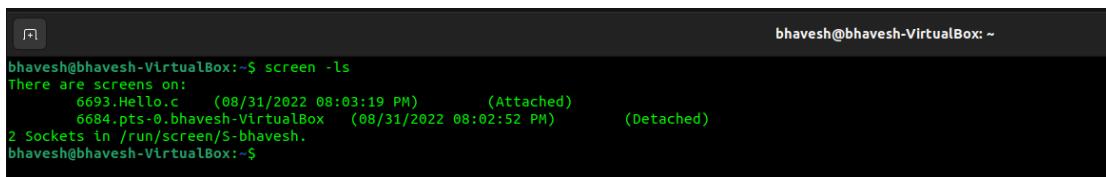
From a shell prompt within a previously created window. This will not run another copy of screen, but will instead supply the command name and its arguments to the window manager (specified in the $STY environment variable) who will use it to create the new window. The above example would start the emacs editor (editing prog.c) and switch to its window. Note that you cannot transport environment variables from the invoking shell to the application (emacs in this case), because it is forked from the parent screen process, not from the invoking shell.

If /tmp/utmp is writable by screen, an appropriate record will be written to this file for each window, and removed when the window is terminated. This is useful for working with talk, script, shutdown, rsh, rrend, rscs and other similar programs that use the utmp file to determine who you are. As long as screen is active on your terminal, the terminal's own record is removed from the utmp file. See also C-a L.

GETTING STARTED
Before you begin to use screen you'll need to make sure you have correctly selected your terminal type, just as you would for any other termcap/terminfo program. (You can do this by using test for example.)
```

[root@bhavesh ~]# screen(1) | less | less(1) for help on less details.

```
$ screen
```

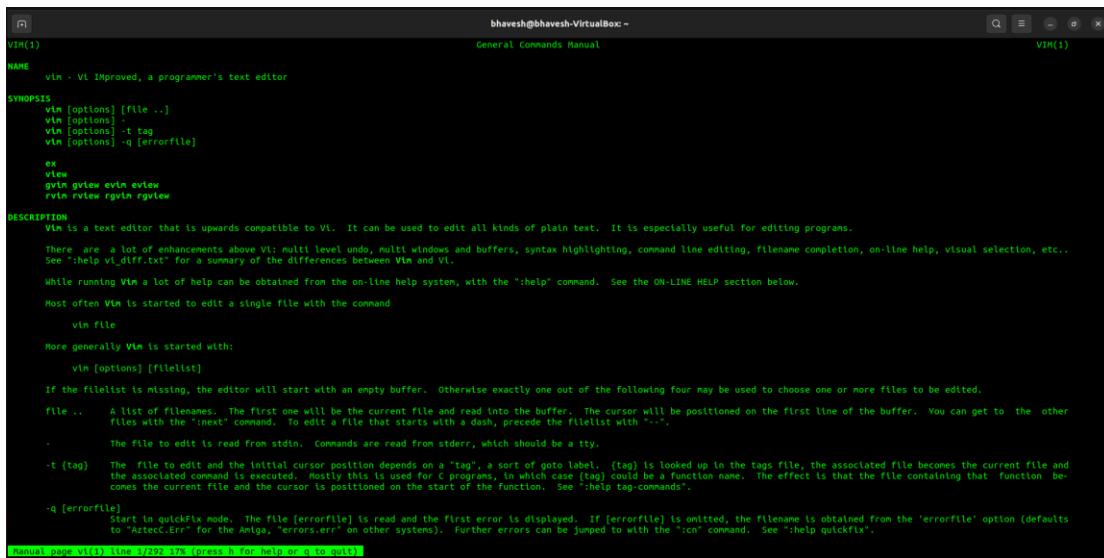


The screenshot shows a terminal window with the title "bhavesh@bhavesh-VirtualBox:~". The text is as follows:

```
bhavesh@bhavesh-VirtualBox:~$ screen -ls
There are screens on:
  6693.Hello.c  (08/31/2022 08:03:19 PM)          (Attached)
  6684 pts-0.bhavesh-VirtualBox  (08/31/2022 08:02:52 PM)          (Detached)
2 Sockets in /run/screen/S-bhavesh.
bhavesh@bhavesh-VirtualBox:~$
```

z. vim : vi or vim is a command line text editor application. It is used to open up an existing file by providing the path along with the command. In case the file doesn't exist at the specified path, a new file will be created with that name at the specified location.

```
$ man vim
```



The screenshot shows a terminal window with the title "VIM(1)" and the subtitle "General Commands Manual". The content of the window is the man page for the "vim" command. The man page starts with a brief description of what vim is and how it compares to vi. It then details various command-line options such as "-t tag", "-q [errorfile]", and "-v". The "SYNOPSIS" section shows examples like "vim [options] [file ..]" and "vim -t tag". The "DESCRIPTION" section provides a more detailed explanation of vim's features, including its compatibility with vi, its support for multiple windows and buffers, syntax highlighting, and command-line editing. It also describes how to start vim with a specific file and how to use the help system. The man page ends with a note about quickfix mode and error files.

```
VIM(1)                                bhavev@bhavev-VirtualBox ~                               VIM(1)

NAME
       vim - VI IMproved, a programmer's text editor

SYNOPSIS
       vim [options] [file ..]
       vim [options] :
       vim [options] -t tag
       vim [options] -q [errorfile]

       ex
       vim
       gvim gview evim evview
       rvim rview rgvim rgview

DESCRIPTION
       Vim is a text editor that is upwards compatible to Vi. It can be used to edit all kinds of plain text. It is especially useful for editing programs.

       There are a lot of enhancements above Vi: multi level undo, multi windows and buffers, syntax highlighting, command line editing, filename completion, on-line help, visual selection, etc.. See :help vim_difff.txt for a summary of the differences between Vim and Vi.

       While running Vim a lot of help can be obtained from the on-line help system, with the ":help" command. See the ON-LINE HELP section below.

       Most often Vim is started to edit a single file with the command

           vim FILE

       More generally Vim is started with:

           vim [options] [filelist]

       If the filelist is missing, the editor will start with an empty buffer. Otherwise exactly one out of the following four may be used to choose one or more files to be edited.

       file ..    A list of filenames. The first one will be the current file and read into the buffer. The cursor will be positioned on the first line of the buffer. You can get to the other files with the "next" command. To edit a file that starts with a dash, precede the filelist with "-".
       .         The file to edit is read from stdin. Commands are read from stderr, which should be a tty.
       -t {tag}   The file to edit and the initial cursor position depends on a "tag", a sort of goto label. {tag} is looked up in the tags file, the associated file becomes the current file and the associated command is executed. Mostly this is used for C programs, in which case {tag} could be a function name. The effect is that the file containing that function becomes the current file and the cursor is positioned on the start of the function. See :help tag-commands".
       -q [errorfile]
           Start in quickFix mode. The file [errorfile] is read and the first error is displayed. If [errorfile] is omitted, the filename is obtained from the 'errorfile' option (defaults to "Artecc.Err" for the Amiga, "errors.err" on other systems). Further errors can be jumped to with the ":cn" command. See :help quickfix.

$ vim
```

```
$ vim
```

```
" zip.vim version v32
" Browsing zipfile /home/bhavesh/Hello.odt
" Select a file with cursor and press ENTER

mimetype
Configurations2/accelerator/
Configurations2/images/Bitmaps/
Configurations2/toolpanel/
Configurations2/progressbar/
Configurations2/statusbar/
Configurations2/toolbar/
Configurations2/floater/
Configurations2/popupmenu/
Configurations2/menubar/
manifest.rdf
meta.xml
settings.xml
Thumbnails/thumbnail.png
styles.xml
content.xml
META-INF/manifest.xml
"
```

aa. chmod : This command is used to modify the read write and edit access of a file.

```
$ man chmod
```

```
bhavesh@bhavesh-VirtualBox:~ CHMOD(1) User Commands CHMOD(1)

NAME      chmod - change file mode bits

SYNOPSIS
        chmod [OPTION]... MODE[.MODE]... FILE...
        chmod [OPTION]... OCTAL-MODE FILE...
        chmod [OPTION]... --reference=FILE FILE...

DESCRIPTION
        This manual page documents the GNU version of chmod. chmod changes the file mode bits of each given file according to mode, which can be either a symbolic representation of changes to make, or an octal number representing the bit pattern for the new mode bits.

        The format of a symbolic mode is [ugoa...][[+|-][perms...]]..., where perms is either zero or more letters from the set rwxst, or a single letter from the set ugo. Multiple symbolic modes can be given, separated by commas.

        A combination of the letters ugoa controls which users' access to the file will be changed: the user who owns it (u), other users in the file's group (g), other users not in the file's group (o), or all users (a). If none of these are given, the effect is as if (u) were given, but bits that are set in the umask are not affected.

        The operator + causes the selected file mode bits to be added to the existing file mode bits of each file; - causes them to be removed; and = causes them to be added and causes unmentioned bits to be removed except that a directory's unmentioned set user and group ID bits are not affected.

        The letters rwxst select file mode bits for the affected users: read (r), write (w), execute (or search for directories) (x), execute/search only if the file is a directory or already has execute permission for some user (x), set user or group ID on execution (s), restricted deletion flag or sticky bit (t). Instead of one or more of these letters, you can specify exactly one of the letters ugo: the permissions granted to the user who owns the file (u), the permissions granted to other users who are members of the file's group (g), and the permissions granted to users that are in neither of the two preceding categories (a).

        A numeric mode is from one to four octal digits (0-7), derived by adding up the bits with values 4, 2, and 1. Omitted digits are assumed to be leading zeros. The first digit selects the set user ID (4) and set group ID (2) and restricted deletion or sticky (1) attributes. The second digit selects permissions for the user who owns the file: read (4), write (2), and execute (1); the third selects permissions for other users in the file's group, with the same values; and the fourth for other users not in the file's group, with the same values.

        chmod never changes the permissions of symbolic links; the chmod system call cannot change their permissions. This is not a problem since the permissions of symbolic links are never used. However, for each symbolic link listed on the command line, chmod changes the permissions of the pointed-to file. In contrast, chmod ignores symbolic links encountered during recursive directory traversals.

SETUID AND SETGID BITS
        chmod clears the set-group-ID bit of a regular file if the file's group ID does not match the user's effective group ID or one of the user's supplementary group IDs, unless the user has appropriate privileges. Additional restrictions may cause the set-user-ID and set-group-ID bits of MODE or REFILE to be ignored. This behavior depends on the policy and functionality of the underlying chmod system call. When in doubt, check the underlying system behavior.

        For directories chmod preserves set-user-ID and set-group-ID bits unless you explicitly specify otherwise. You can set or clear the bits with symbolic modes like u+s and g+s. To clear these bits for directories with a numeric mode requires an additional leading zero, or leading = like 00755, or =755

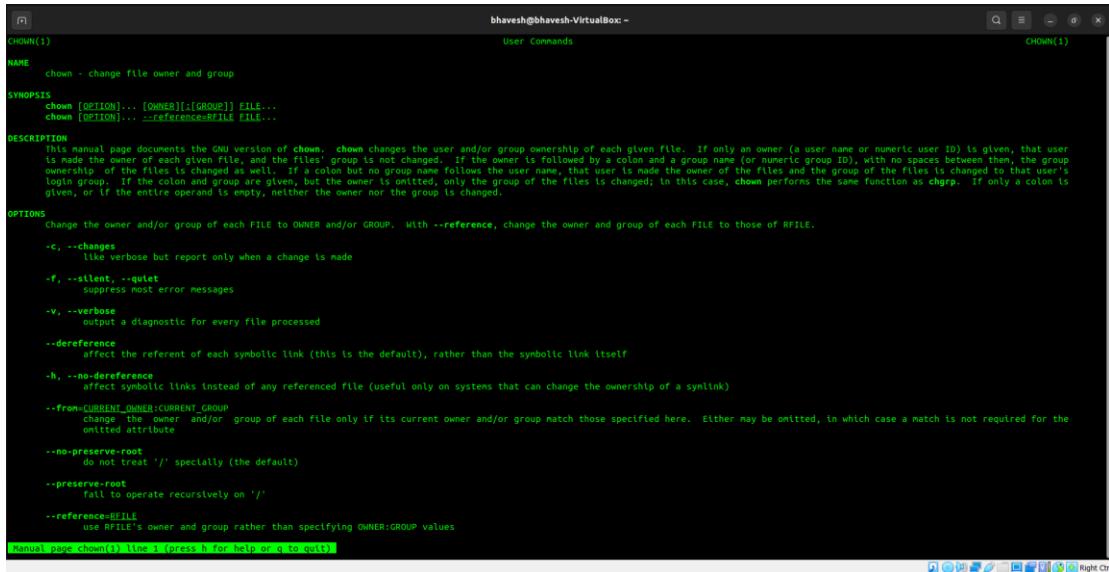
RESTRICTED DELETION FLAG OR STICKY BIT
        The restricted deletion flag or sticky bit is a single bit, whose interpretation depends on the file type. For directories, it prevents unprivileged users from removing or renaming a file in the directory unless they own the file or the directory; this is called the restricted deletion flag for the directory, and is commonly found on world-writable directories like /tmp. For Manual page chmod(1) line 1 (press h for help or q to quit)
        bhavesh@bhavesh-VirtualBox:~ Right Ctrl
```

\$ chmod

```
bhavesh@bhavesh-VirtualBox:~$ chmod +rwx Hello.c
bhavesh@bhavesh-VirtualBox:~$ chmod +rwx Hello.c -v
mode of 'Hello.c' retained as 0775 (rwxrwxr-x)
bhavesh@bhavesh-VirtualBox:~$ chmod -777 Hello.c -v
mode of 'Hello.c' changed from 0775 (rwxrwxr-x) to 0000 (-----)
bhavesh@bhavesh-VirtualBox:~$ chmod +777 Hello.c -v
mode of 'Hello.c' changed from 0000 (-----) to 0777 (rwxrwxrwx)
bhavesh@bhavesh-VirtualBox:~$
```

bb. chown : This command is used to change the ownership of a file/directory. In case of a directory, even though the ownership of directory is changed, the ownership of sub files will still remain with the original user.

```
$ man chown
```



```
bhavesh@bhavesh-VirtualBox:~
```

**NAME**  
chown - change file owner and group

**SYNOPSIS**  
chown [OPTION]... [OWNER]:[GROUP] FILE...  
chown [OPTION]... --reference=RFILE FILE...

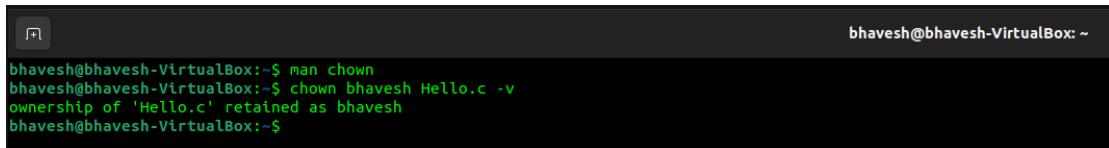
**DESCRIPTION**  
This manual page documents the GNU version of chown. chown changes the user and/or group ownership of each given file. If only an owner (a user name or numeric user ID) is given, that user is made the owner of each given file, and the file's group is not changed. If the owner is followed by a colon and a group name (or numeric group ID), with no spaces between them, the group of each of the files is set to that group. A colon followed by a user name (that user is made the owner of the files and the group of the files is changed to that user's login group). If the colon and group are given, but the owner is omitted, only the group of the files is changed; in this case, chown performs the same function as chgrp. If only a colon is given, or if the entire operand is empty, neither the owner nor the group is changed.

**OPTIONS**  
Change the owner and/or group of each FILE to OWNER and/or GROUP. With --reference, change the owner and group of each FILE to those of RFILE.

- c, --changes  
like verbose but report only when a change is made
- f, --silent, --quiet  
suppress most error messages
- v, --verbose  
output a diagnostic for every file processed
- dereference  
affect the referent of each symbolic link (this is the default), rather than the symbolic link itself
- h, --no-dereference  
affect symbolic links instead of any referenced file (useful only on systems that can change the ownership of a symlink)
- from=CURRENT\_OWNER:CURRENT\_GROUP  
change the owner and/or group of each file only if its current owner and/or group match those specified here. Either may be omitted, in which case a match is not required for the omitted attribute
- no-preserve-root  
do not treat '/' specially (the default)
- preserve-root  
fail to operate recursively on '/'
- reference=RFILE  
use RFILE's owner and group rather than specifying OWNER:GROUP values

```
man(1) chown(1) Linux 4.1.12 - Generated by man-db 2.7.1-1ubuntu1.2
```

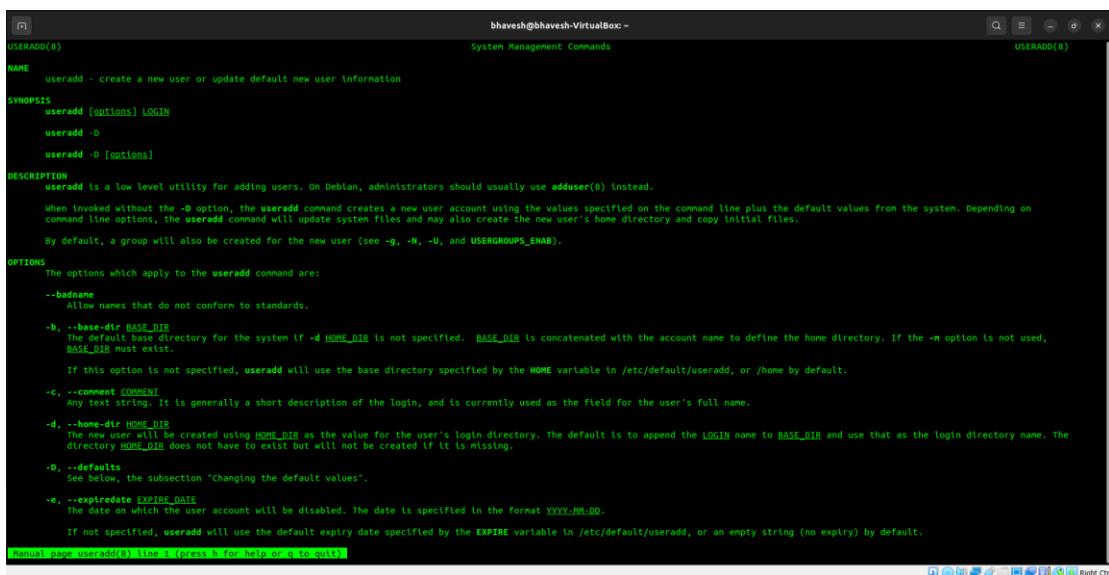
```
$ chown
```



```
bhavesh@bhavesh-VirtualBox:~$ man chown  
bhavesh@bhavesh-VirtualBox:~$ chown bhavesh Hello.c -v  
ownership of 'Hello.c' retained as bhavesh  
bhavesh@bhavesh-VirtualBox:~$
```

cc. useradd : By using useradd, we can add a new user or modify the existing information of an user.

```
$ man useradd
```



The screenshot shows a terminal window titled "bhavesh@bhavesh-VirtualBox: ~" displaying the man page for "useradd". The window has a title bar "System Management Commands" and a status bar "USERADD(8)". The man page content includes sections for NAME, SYNOPSIS, DESCRIPTION, and OPTIONS, detailing the command's purpose and usage options.

```
USERADD(8)                                bhavesh@bhavesh-VirtualBox: ~
                                         System Management Commands
                                         USERADD(8)

NAME
    useradd - create a new user or update default new user information

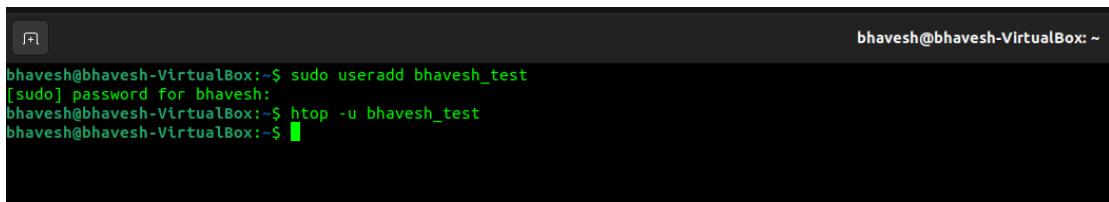
SYNOPSIS
    useradd [options] LOGIN
    useradd -D
    useradd -o [options]

DESCRIPTION
    useradd is a low level utility for adding users. On Debian, administrators should usually use adduser(8) instead.
    When invoked without the -D option, the useradd command creates a new user account using the values specified on the command line plus the default values from the system. Depending on command line options, the useradd command will update system files and may also create the new user's home directory and copy initial files.
    By default, a group will also be created for the new user (see -g, -N, -U, and USERGROUPS_ENAB).

OPTIONS
    The options which apply to the useradd command are:
    --badname
        Allow names that do not conform to standards.
    -B, --base-dir BASE_DIR
        The default base directory for the system if -d HOME_DIR is not specified. BASE_DIR is concatenated with the account name to define the home directory. If the -m option is not used, BASE_DIR must exist.
        If this option is not specified, useradd will use the base directory specified by the HOME variable in /etc/default/useradd, or /home by default.
    -c, --comment COMMENT
        Any text string. It is generally a short description of the login, and is currently used as the field for the user's full name.
    -d, --home-dir HOME_DIR
        The new user will be created using HOME_DIR as the value for the user's login directory. The default is to append the LOGIN name to BASE_DIR and use that as the login directory name. The directory HOME_DIR does not have to exist but will not be created if it is missing.
    -D, --defaults
        See below, the subsection "Changing the default values".
    -e, --expiredate EXPIRE_DATE
        The date on which the user account will be disabled. The date is specified in the format YYYY-MM-DD.
        If not specified, useradd will use the default expiry date specified by the EXPIRE variable in /etc/default/useradd, or an empty string (no expiry) by default.

    (continued) [more text about creating a new user account]
```

```
$ useradd
```

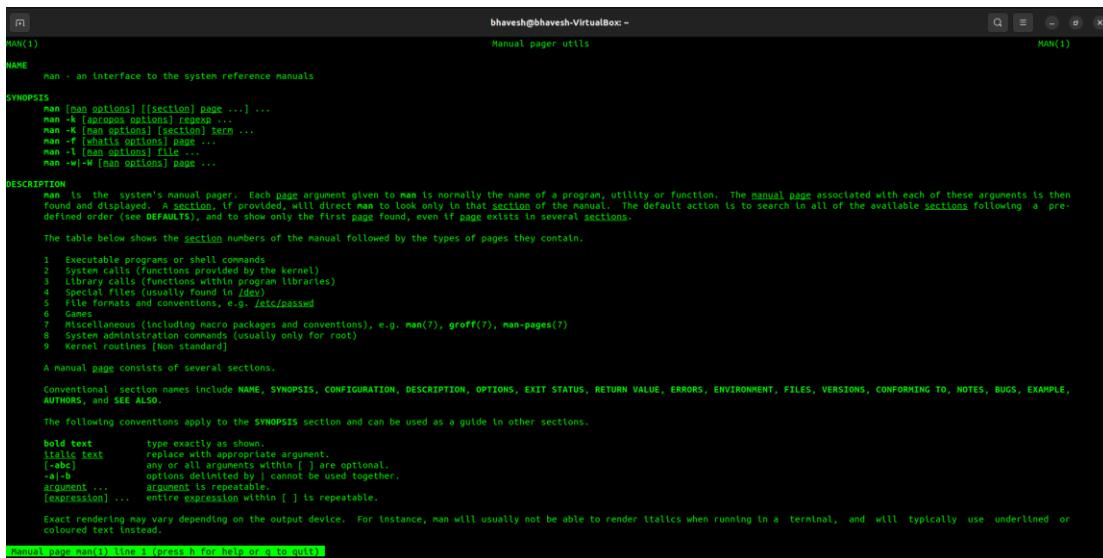


The screenshot shows a terminal window titled "bhavesh@bhavesh-VirtualBox: ~" executing the command "sudo useradd bhavesh\_test". It prompts for a password and then runs "htop -u bhavesh\_test" to show the process. The terminal interface includes a toolbar at the top and a status bar at the bottom.

```
bhavesh@bhavesh-VirtualBox:~$ sudo useradd bhavesh_test
[sudo] password for bhavesh:
bhavesh@bhavesh-VirtualBox:~$ htop -u bhavesh_test
bhavesh@bhavesh-VirtualBox:~$
```

dd. man : This command invokes the manual screen for any command written in conjunction with it.

```
$ man man
```



The screenshot shows a terminal window titled "MAN(1)" with the title bar "Manual pager utils". The content of the terminal is the man page for the "man" command. It includes sections for NAME, SYNOPSIS, DESCRIPTION, and CONVENTIONS. The SYNOPSIS section lists various command-line options. The DESCRIPTION section explains what "man" does and provides a table of section numbers. The CONVENTIONS section details rendering rules for different text types.

```
bhavesh@bhavesh-VirtualBox:~$ man man
                                         Manual pager utils
MAN(1)
NAME    man - an interface to the system reference manuals
SYNOPSIS
man [man options] [[section] page ...] ...
man -k [process options] regexp ...
man -c [category options] [section] term ...
man -r [math options] ...
man -l [man options] file ...
man -w|-W [man options] page ...
DESCRIPTION
man is the system's manual pager. Each page argument given to man is normally the name of a program, utility or function. The manual page associated with each of these arguments is then found and displayed. A section, if provided, will direct man to look only in that section of the manual. The default action is to search in all of the available sections following a pre-defined order (see DEFAULTS), and to show only the first page found, even if page exists in several sections.
The table below shows the section numbers of the manual followed by the types of pages they contain.
1 Executable programs or shell commands
2 System calls (functions provided by the kernel)
3 Library calls (functions within program libraries)
4 Special files (usually found in /dev)
5 File Formats and conventions, e.g. /etc/passwd
6 Games
7 Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7), man-pages(7)
8 System administration commands (usually only for root)
9 Kernel routine [Not standard]
A manual page consists of several sections.
Conventional section names include NAME, SYNOPSIS, CONFIGURATION, DESCRIPTION, OPTIONS, EXIT STATUS, RETURN VALUE, ERRORS, ENVIRONMENT, FILES, VERSIONS, CONFORMING TO, NOTES, BUGS, EXAMPLE, AUTHORS, and SEE ALSO.
The following conventions apply to the SYNOPSIS section and can be used as a guide in other sections.
bold text      type exactly as shown.
italic text    replace with appropriate argument.
[abc]          any of all arguments within [ ] are optional.
{-a|-b}         options delimited by | cannot be used together.
argument ...   argument is repeatable.
(expression) ... entire expression within [ ] is repeatable.
Exact rendering may vary depending on the output device. For instance, man will usually not be able to render italics when running in a terminal, and will typically use underlined or coloured text instead.
bhavesh@bhavesh-VirtualBox:~$
```

```
$ man
```



The screenshot shows a terminal window titled "bhavesh@bhavesh-VirtualBox:~\$". The user typed "man du" and pressed enter. The man page for "du" is displayed, showing its synopsis and a brief description.

```
bhavesh@bhavesh-VirtualBox:~$ man du
bhavesh@bhavesh-VirtualBox:~$
```

ee. locate : This command is used as a quick file finding utility. Syntax wise, it is written as locate/plocate as it searches files by patterns. It is faster than mlocate.

```
$ man locate
```

```

bhavesh@bhavesh-VirtualBox: ~
General Commands Manual
locate(1)
locate(1)

NAME
    plocate - find files by name, quickly

SYNOPSIS
    plocate [OPTION]... PATTERN...

DESCRIPTION
    plocate finds all files on the system matching the given pattern (or all of the patterns if multiple are given). It does this by means of an index made by updatedb() or (less commonly) converted from another index by plocate-build(8).

    plocate is largely argument-compatible with locate(1), but is significantly faster. In particular, it rarely needs to scan through its entire database, unless the pattern is very short (less than three bytes) or you want to search for a regular expression. It does not try to maintain compatibility with BSD locate, or non-UTF-8 filenames and locales. Most I/O is done asynchronously, but the results are synchronized so that output comes in the same order every time.

    When multiple patterns are given, plocate will search for files that match all of them. This is the main incompatibility with locate(1), which searches for files that match one or more patterns, unless the -A option is given.

    By default, patterns are taken to be substrings to search for. If at least one non-excaped globbing metacharacter (*, ? or []) is given, that pattern is instead taken to be a glob pattern (which means it needs to start and end in * for a substring match). If --regexp is given, patterns are instead taken to be (non-anchored) POSIX basic regular expressions, and if --regex is given, patterns are taken to be POSIX extended regular expressions. All of this matches locate(1) behavior.

    Like locate(1), plocate shows all files visible to the calling user (by virtue of having read and execute permissions on all parent directories), and none that are not, by means of running with the setgid bit set to access the index (which is built as root), but by testing visibility as the calling user.

OPTIONS
    -A, --all
        Ignored for compatibility with locate(1).

    -b, --basename
        Match only against the file name portion of the path name, i.e., the directory names will be excluded from the match (but still printed). This does not speed up the search, but can suppress uninteresting matches.

    -c, --count
        Do not print each match. Instead, count them, and print out a total number at the end.

    -d, --database DBPATH
        Find matches in the given database, instead of /var/lib/plocate/plocate.db. This argument can be given multiple times, to search multiple databases. It is also possible to give multiple databases in one argument, separated by :. (Any character, including :, and \, can be escaped by prepending a \:)

    -e, --existing
        Print only entries that refer to files existing at the time locate is run. Note that unlike locate(1), symlinks are not followed by default (and indeed, there is no option to change this).

    -I, --ignore-case
        Do a case-insensitive match as given by the current locale (default is case-sensitive, byte-by-byte match). Note that plocate does not support the full range of Unicode case folding.

Manual page locate(1) line 1 (press h for help or q to quit)

```

\$ plocate

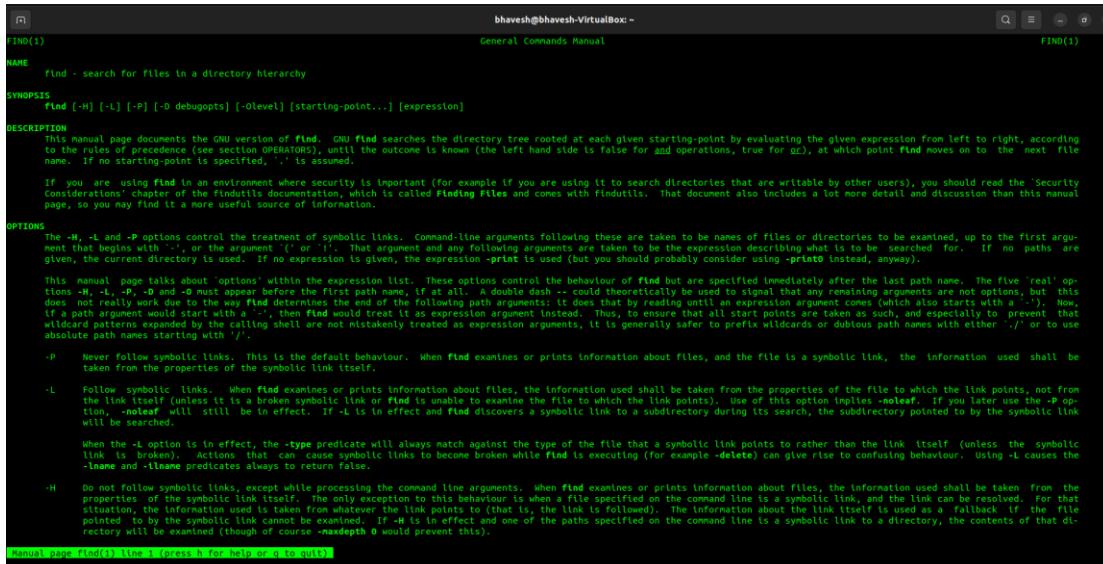
```

bhavesh@bhavesh-VirtualBox: ~
bhavesh@bhavesh-VirtualBox: ~$ locate odt
/home/bhavesh>Hello.odt
/home/bhavesh>Hello_copied.odt
/home/bhavesh/odt_files.tar
/snap/core20/1587/usr/share/zoneinfo/America/Godthab
/snap/core20/1587/usr/share/zoneinfo posix/America/Godthab
/snap/core20/1587/usr/share/zoneinfo/right/America/Godthab
/snap/core20/1611/usr/share/zoneinfo/America/Godthab
/snap/core20/1611/usr/share/zoneinfo posix/America/Godthab
/snap/core20/1611/usr/share/zoneinfo/right/America/Godthab
/usr/lib/libreoffice/CREDITS.fodt
/usr/lib/libreoffice/share/template/common/internal/idxexample.odt
/usr/share/cups/data/form_english_in.odt
/usr/share/cups/data/form_russian_in.odt
/usr/share/doc/fonts-sil-abysinica/documentation/AbyssinicaSILFontFeatures.odt
/usr/share/doc/fonts-sil-padauk/documentation/Padauk-features.odt
/usr/share/doc/fonts-sil-padauk/documentation/Padauk-typesample.odt
/usr/share/help/gnome-help/music-player-ipodtransfer.page
/usr/share/libreoffice/help/media/icon-themes/res/odt_16_8.svg
/usr/share/libreoffice/help/media/icon-themes/res/odt_32_8.svg
/usr/share/libreoffice/help/media/icon-themes/res/odt_48_8.svg
/usr/share/libreoffice/help/media/icon-themes/res/odt_96_8.svg
/usr/share/zoneinfo/America/Godthab
/usr/share/zoneinfo posix/America/Godthab
/usr/share/zoneinfo/right/America/Godthab
bhavesh@bhavesh-VirtualBox: ~$ locate .odt
/home/bhavesh>Hello.odt
/home/bhavesh>Hello_copied.odt
/usr/lib/libreoffice/share/template/common/internal/idxexample.odt
/usr/share/cups/data/form_english_in.odt
/usr/share/cups/data/form_russian_in.odt
/usr/share/doc/fonts-sil-abysinica/documentation/AbyssinicaSILFontFeatures.odt
/usr/share/doc/fonts-sil-padauk/documentation/Padauk-features.odt
/usr/share/doc/fonts-sil-padauk/documentation/Padauk-typesample.odt
bhavesh@bhavesh-VirtualBox: ~

```

ff. find : The command find is another utility for finding files. The difference between locate and find is that locate searches for files by searching for pattern while find uses a more traditional approach and searches for the file directory wise.

```
$ man find
```



The screenshot shows a terminal window titled "General Commands Manual" with the command "FIND(1)" at the top. The content of the man page is displayed, starting with the NAME section:

**NAME**  
find - search for files in a directory hierarchy

**SYNOPSIS**  
`find [-H] [-L] [-P] [-O debugopts] [-Olevel] [starting-point...] [expression]`

**DESCRIPTION**  
This manual page documents the GNU version of `find`. GNU `find` searches the directory tree rooted at each given starting-point by evaluating the given expression from left to right, according to the rules of precedence (see section OPERATORS), until the outcome is known (the left hand side is `False` for `and` operations, `true` for `or`), at which point `find` moves on to the next file name. If no starting-point is specified, `'.'` is assumed.

If you are using `find` in an environment where security is important (for example if you are using it to search directories that are writable by other users), you should read the "Security Considerations" chapter of the `findutils` documentation, which is called `Finding Files` and comes with `findutils`. That document also includes a lot more detail and discussion than this manual page, so you may find it a more useful source of information.

**OPTIONS**  
The `-H`, `-L` and `-P` options control the treatment of symbolic links. Command-line arguments following these are taken to be names of files or directories to be examined, up to the first argument that begins with `'.'`, or the argument `(''` or `')'`. That argument and any following arguments are taken to be the expression describing what is to be searched for. If no paths are given, the current directory is used. If no expression is given, the expression `-print0` is used (but you should probably consider using `-print0` instead, anyway).

This manual page talks about "options" within the expression list. These options control the behaviour of `find` but are specified immediately after the last path name. The five "real" options `-H`, `-L`, `-P`, `-D` and `-O` must appear before the first path name, if at all. A double dash `--` could theoretically be used to signal that any remaining arguments are not options, but this is not actually supported by the `find` implementation. It does not make sense to read options after the `--` separator, as the options themselves define the search space, not the search space itself. If a path argument would start with `'.'`, then `find` would treat it as expression arguments instead. Thus, to ensure that all start points are taken as such, and especially to prevent that wildcard patterns expanded by the calling shell are not mistakenly treated as expression arguments, it is generally safer to prefix wildcards or dubious path names with either `'/'` or to use absolute path names starting with `'/'`.

`-P` Never follow symbolic links. This is the default behaviour. When `find` examines or prints information about files, and the file is a symbolic link, the information used shall be taken from the properties of the symbolic link itself.

`-L` Follow symbolic links. When `find` examines or prints information about files, the information used shall be taken from the properties of the file to which the link points, not from the link itself (unless it is a broken symbolic link or `find` is unable to examine the file to which the link points). Use of this option implies `-notest`. If you later use the `-O` option, `-notest` will still be in effect. If `-L` is in effect and `find` discovers a symbolic link to a subdirectory during its search, the subdirectory pointed to by the symbolic link will be searched.

When the `-L` option is in effect, the `-type` predicate will always match against the type of the file that a symbolic link points to rather than the link itself (unless the symbolic link is broken). Actions that can cause symbolic links to become broken while `find` is executing (for example `-delete`) can give rise to confusing behaviour. Using `-L` causes the `-lname` and `-Lname` predicates always to return false.

`-H` Do not follow symbolic links, except while processing the command line arguments. When `find` examines or prints information about files, the information used shall be taken from the properties of the symbolic link itself. The only exception to this behaviour is when a file specified on the command line is a symbolic link and the link can be resolved. For that situation, the information used is taken from whatever the link points to (that is, the link is followed). The information about the link itself is used as a fallback. If the file pointed to by the symbolic link cannot be examined, if `-H` is in effect and one of the paths specified on the command line is a symbolic link to a directory, the contents of that directory will be examined (though of course `-maxdepth 0` would prevent this).

manpages.org | find(1) | Linux | Oracle | FreeBSD | Mac OS X | OpenBSD | NetBSD |

```
$ find
```

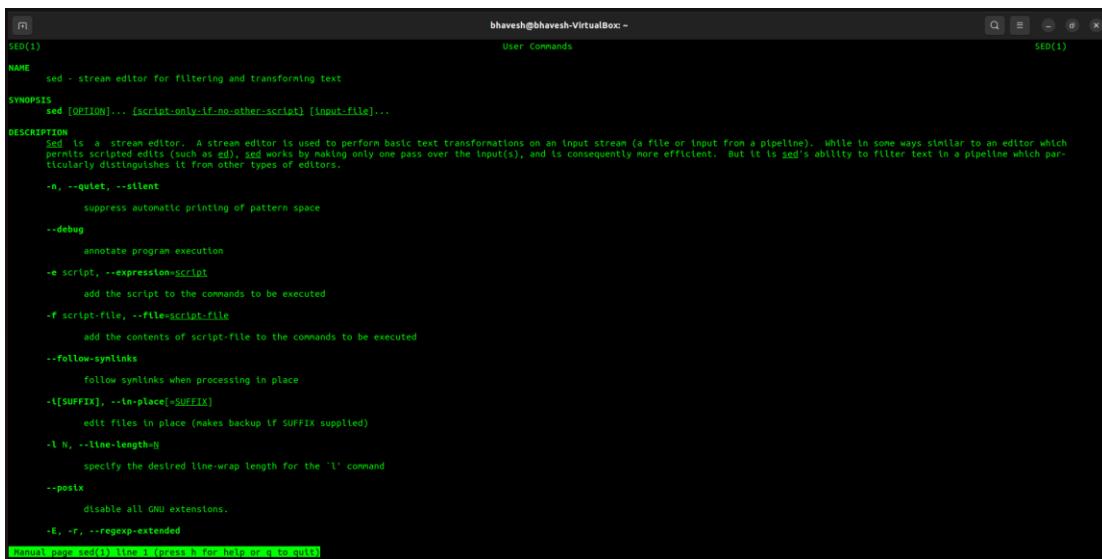


The screenshot shows a terminal window with the command "bhavesh@bhavesh-VirtualBox:~\$ find Hello.c" entered. The output shows the file "Hello.c" was found in the current directory.

```
bhavesh@bhavesh-VirtualBox:~$ find Hello.c
Hello.c
bhavesh@bhavesh-VirtualBox:~$
```

gg. sed : sed is a command line utility which works like ed (i.e. editing text) but the add-on capability of sed to filter text in a pipeline distinguishes it and makes it more efficient.

```
$ man sed
```



The screenshot shows a terminal window with the title "SED(1)" at the top right. The window contains the man page for the "sed" command. The text is white on a black background. The man page includes sections for NAME, SYNOPSIS, DESCRIPTION, and various command-line options. The "DESCRIPTION" section explains that sed is a stream editor used for basic text transformations on an input stream. It highlights its efficiency due to performing one pass over the input(s). Other sections describe options like -n for quiet output, --debug for program annotation, and -e for adding script commands.

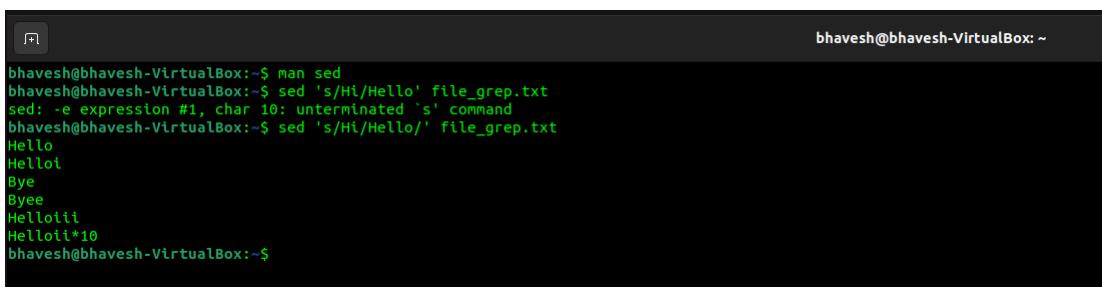
```
bhavesh@bhavesh-VirtualBox:~
```

```
man sed
```

```
NAME      sed - stream editor for filtering and transforming text
SYNOPSIS  sed [OPTION]... [-f no-other-script] [Input-file]...
DESCRIPTION
  Sed is a stream editor. A stream editor is used to perform basic text transformations on an input stream (a file or input from a pipeline). While in some ways similar to an editor which permits scripted edits (such as ed), sed works by making only one pass over the input(s), and is consequently more efficient. But it is sed's ability to filter text in a pipeline which particularly distinguishes it from other types of editors.
  -n, --quiet, --silent
    suppress automatic printing of pattern space
  --debug
    annotate program execution
  -e script, --expression=script
    add the script to the commands to be executed
  -f script-file, --file=script-file
    add the contents of script-file to the commands to be executed
  --follow-symlinks
    follow symlinks when processing in place
  -i[SUFFIX], --in-place[=SUFFIX]
    edit files in place (makes backup if SUFFIX supplied)
  -l N, --line-length=N
    specify the desired line-wrap length for the 'l' command
  --postx
    disable all GNU extensions.
  -E, -F, --regexp-extended

```

```
$ sed
```

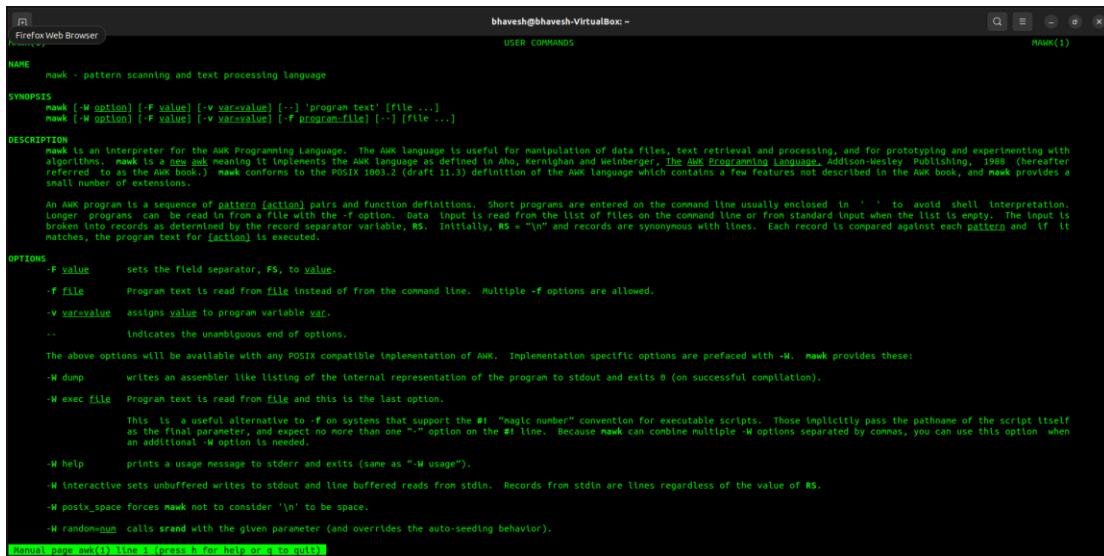


The screenshot shows a terminal window with the title "bhavesh@bhavesh-VirtualBox: ~". The user runs "man sed" to view the manual page. Then, they run "sed 's/Hi>Hello' file\_grep.txt" on a file named "file\_grep.txt". The file contains the text "Hello", "Helloi", "Bye", "Byee", "Helloii", and "Helloii\*10". The "sed" command replaces the first "Hello" with "Hello". The output shows the modified file content: "Hello", "Helloi", "Bye", "Byee", "Helloii", and "Helloii\*10".

```
bhavesh@bhavesh-VirtualBox:~$ man sed
bhavesh@bhavesh-VirtualBox:~$ sed 's/Hi>Hello' file_grep.txt
sed: -e expression #1, char 10: unterminated 's' command
bhavesh@bhavesh-VirtualBox:~$ sed 's/Hi>Hello/' file_grep.txt
Hello
Helloi
Bye
Byee
Helloii
Helloii*10
bhavesh@bhavesh-VirtualBox:~$
```

hh. awk : The awk is a language and the command that is used for interpreting awk language script is mawk. It is used for searching patterns in data, text mining, text retrieval and processing.

```
$ man awk
```



The screenshot shows a Firefox browser window with the title "USER COMMANDS" and a tab labeled "MAWK(1)". The page content is the man page for awk, which includes sections for NAME, SYNOPSIS, DESCRIPTION, and OPTIONS. The SYNOPSIS section shows the command line syntax: awk [options] [pattern] [value] [var=value] [-f] 'program text' [file ...]. The DESCRIPTION section provides a detailed explanation of awk's purpose and how it differs from mawk. The OPTIONS section lists various command-line options with their descriptions.

```
NAME      mawk - pattern scanning and text processing language
SYNOPSIS
        awk [-W option] [-# value] [-v var=value] [-f] 'program text' [file ...]
        awk [-W option] [-# value] [-v var=value] [-f program-file] [-f] [file ...]
DESCRIPTION
        awk is an interpreter for the AWK Programming Language. The AWK language is useful for manipulation of data files, text retrieval and processing, and for prototyping and experimenting with algorithms. mawk is a new awk meaning it implements the AWK language as defined in Aho, Kernighan and Weinberger, The AWK Programming Language, Addison-Wesley Publishing, 1988 (hereafter referred to as the AWK book.) mawk conforms to the POSIX 1003.2 (draft 11.3) definition of the AWK language which contains a few features not described in the AWK book, and mawk provides a small number of extensions.

        An AWK program is a sequence of pattern {action} pairs and program definitions. Short programs are entered on the command line usually enclosed in ' ' to avoid shell interpretation. Longer programs can be read in from a file with the -f option. Data input is read from the list of files on the command line or from standard input when the list is empty. The input is broken into records by the record separator variable, RS. Initially, RS = "\n" and records are synonymous with lines. Each record is compared against each pattern and if it matches, the program text for {action} is executed.

OPTIONS
        -F value      sets the field separator, FS, to value.
        -f file       Program text is read from file instead of from the command line. Multiple -f options are allowed.
        -v var=value   assigns value to program variable var.
        --           indicates the unambiguous end of options.

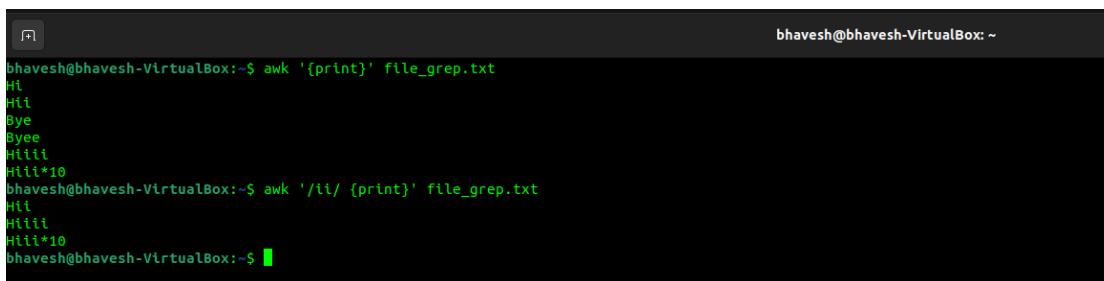
        The above options will be available with any POSIX compatible implementation of AWK. Implementation specific options are prefaced with -W. mawk provides these:
        -W dump       writes an assembler like listing of the internal representation of the program to stdout and exits 0 (on successful compilation).
        -W exec file  Program text is read from file and this is the last option.

        This is a useful alternative to -f on systems that support the #! "magic number" convention for executable scripts. Those implicitly pass the pathname of the script itself as the final parameter, and expect no more than one "-" option on the #! line. Because mawk can combine multiple -W options separated by commas, you can use this option when an additional -W option is needed.

        -W help        prints a usage message to stderr and exits (same as "-W usage").
        -W interactive sets unbuffered writes to stdout and line buffered reads from stdin. Records from stdin are lines regardless of the value of RS.
        -W posix_space forces awk not to consider '\n' to be space.
        -W randomnum  calls srand with the given parameter (and overrides the auto-seeding behavior).

        $ awk --help
        $ awk -W help
        $ awk -W interactive
```

```
$ awk
```

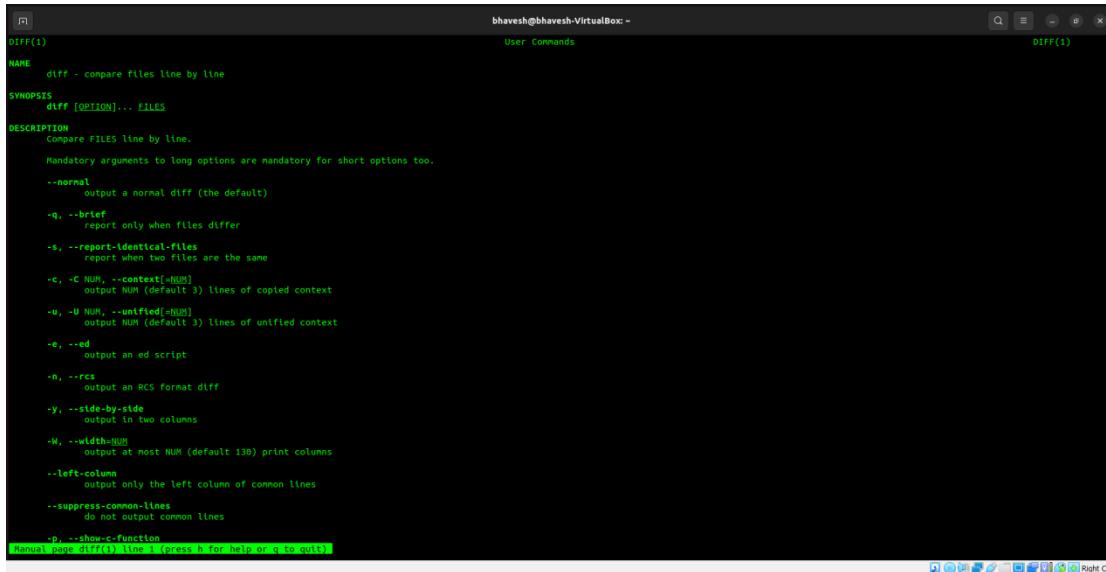


The screenshot shows a terminal window with the prompt "bhavesh@bhavesh-VirtualBox:~\$". The user runs two awk commands. The first command, "awk '{print}' file\_grep.txt", outputs the contents of the file "file\_grep.txt" line by line. The second command, "awk '/li/ {print}' file\_grep.txt", filters the file to only print lines containing the string "li".

```
bhavesh@bhavesh-VirtualBox:~$ awk '{print}' file_grep.txt
Hl
Hil
Bye
Byee
Hlli
Hlli*10
bhavesh@bhavesh-VirtualBox:~$ awk '/li/ {print}' file_grep.txt
Hl
Hlli
Hlli*10
bhavesh@bhavesh-VirtualBox:~$
```

ii. diff : This command is a file comparison utility which can be leveraged to compare two files line by line.

```
$ man diff
```



```
hhavesh@hhavesh-VirtualBox: ~          User Commands          DIFF(1)

diff(1)
NAME      diff - compare files line by line
SYNOPSIS  diff [OPTION]... FILES
DESCRIPTION
Compare FILES line by line.

Mandatory arguments to long options are mandatory for short options too.

--normal          output a normal diff (the default)
-Q, --brief       report only when files differ
-A, --report-identical-files
                  report when two files are the same
-C NUM, --context=NUM
                  output NUM (default 3) lines of copied context
-U NUM, --unified=NUM
                  output NUM (default 3) lines of unified context
-E, --ed          output an ed script
-R, --rcs         output an RCS format difft
-Y, --side-by-side
                  output in two columns
-W, --width=NUM
                  output at most NUM (default 130) print columns
--left-column
                  output only the left column of common lines
--suppress-common-lines
                  do not output common lines
-p, --show-c-function
                  show c-function differences
Manjaro have diff(1).line.1 (press h for help or q to quit)
```

```
$ diff
```



```
bhavesh@bhavesh-VirtualBox:~$ diff Hello.odt Hello_copied.odt
bhavesh@bhavesh-VirtualBox:~$ diff Hello.odt Hello.c
Binary files Hello.odt and Hello.c differ
bhavesh@bhavesh-VirtualBox:~$
```

jj. sort : The use case of this command is to sort the lines in the text files.

```
$ man sort
```

```
sort(1)                                User Commands                               SORT(1)

NAME
    sort - sort lines of text files

SYNOPSIS
    sort [OPTION]... [FILE]...
    sort [OPTION]... -c[files]fromuf

DESCRIPTION
    Write sorted concatenation of all FILE(s) to standard output.
    With no FILE, or when FILE is -, read standard input.

    Mandatory arguments to long options are mandatory for short options too. Ordering options:
    -b, --ignore-leading-blanks
        ignore leading blanks
    -d, --dictionary-order
        consider only blanks and alphanumeric characters
    -f, --ignore-case
        fold lower case to upper case characters
    -g, --general-numeric-sort
        compare according to general numerical value
    -k, --ignore-nonprinting
        consider only printable characters
    -M, --month-sort
        compare (unknown) < 'JAN' < ... < 'DEC'
    -h, --human-numeric-sort
        compare human readable numbers (e.g., 2K 1G)
    -n, --numeric-sort
        compare according to string numerical value
    -R, --random-sort
        shuffle, but group identical keys. See shuf(1)
    --random-source=FILE
        get random bytes from FILE
    -r, --reverse
        reverse
Manual page sort(1), line 1 (press h for help or q to quit)
```

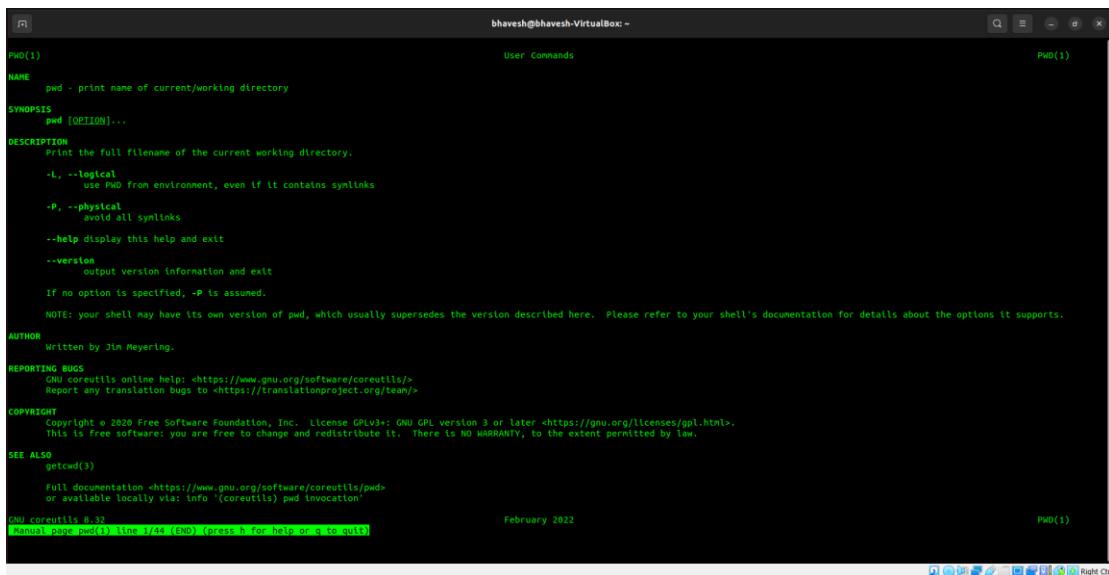
\$ sort

```
bhavesh@bhavesh-VirtualBox:~$ sort file_grep.txt
Bye
Hi
Hit
Hit*
Hit!!
bhavesh@bhavesh-VirtualBox:~$
```



ll. pwd : This command is used very frequently. It displays the current working directory.

```
$man pwd
```



A screenshot of a terminal window titled "User Commands" with the title bar "bhavesh@bhavesh-VirtualBox: ~". The window contains the man page for the "pwd" command. The text is mostly green, with some sections like "SYNOPSIS" and "DESCRIPTION" in blue. The man page details the command's purpose, options (including --logical, --physical, and --help), and version information. It also notes that the shell might have its own version of pwd. The author is Jim Meyering. The reporting bugs section links to the GNU coreutils online help and translation bugs. The copyright section states it's free software under the GPL v3 or later. The "SEE ALSO" section links to "getcwd(3)". At the bottom, it shows "GNU coreutils 8.32" and "Manual page pwd(1) line 1/44 (END) (press h for help or q to quit)".

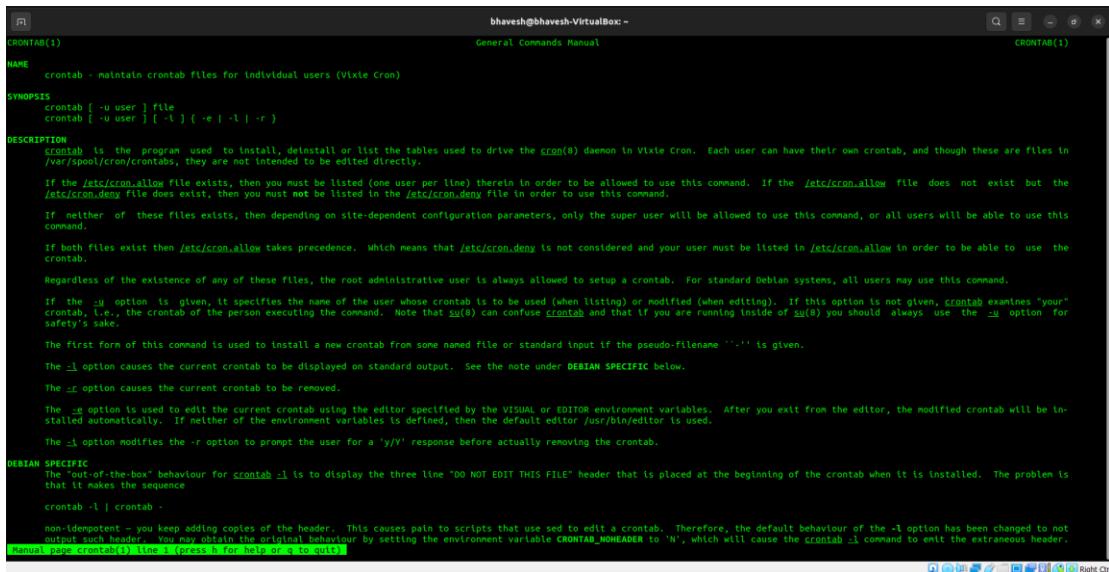
```
$ pwd
```



A screenshot of a terminal window titled "bhavesh@bhavesh-VirtualBox: ~". The user has run the "pwd" command, which outputs the path "/home/bhavesh". The prompt "bhavesh@bhavesh-VirtualBox:~\$ " is visible at the bottom.

mm. crontab : The command crontab is used to display or modify the table of commands that will be executed at regular intervals by cron which is a daemon (i.e. a software program that runs without any user interaction).

```
$ man crontab
```



The screenshot shows a terminal window titled "bhaveh@bhavesh-VirtualBox ~" displaying the man page for "crontab". The title bar also shows "General Commands Manual" and "CRONTAB(1)". The man page content is as follows:

```
NAME      crontab - maintain crontab files for individual users (Vixie Cron)
SYNOPSIS  crontab [-u user] [file]
          crontab [-u user] [-l] [-e | -r | -c]
DESCRIPTION
  crontab is the program used to install, deinstall or list the tables used to drive the cron(8) daemon in Vixie Cron. Each user can have their own crontab, and though these are files in /var/spool/cron/crontabs, they are not intended to be edited directly.

  If the /etc/cron.allow file exists, then you must be listed (one user per line) therein in order to be allowed to use this command. If the /etc/cron.deny file does not exist but the /etc/cron.deny file does exist, then you must not be listed in the /etc/cron.deny file in order to use this command.

  If neither of these files exists, then depending on site-dependent configuration parameters, only the super user will be allowed to use this command, or all users will be able to use this command.

  If both files exist then /etc/cron.allow takes precedence. Which means that /etc/cron.deny is not considered and your user must be listed in /etc/cron.allow in order to be able to use the crontab.

  Regardless of the existence of any of these files, the root administrative user is always allowed to setup a crontab. For standard Debian systems, all users may use this command.

  If the -u option is given, it specifies the name of the user whose crontab is to be used (when listing) or modified (when editing). If this option is not given, crontab examines "your" crontab, i.e., the crontab of the person executing the command. Note that su(8) can confuse crontab and that if you are running inside of su(8) you should always use the -u option for safety's sake.

  The -l option is used to install a new crontab from some named file or standard input if the pseudo-filename ``-'' is given.

  The -e option causes the current crontab to be displayed on standard output. See the note under DEBIAN SPECIFIC below.

  The -r option causes the current crontab to be removed.

  The -u option is used to edit the current crontab using the editor specified by the VISUAL or EDITOR environment variables. After you exit from the editor, the modified crontab will be installed automatically. If neither of the environment variables is defined, then the default editor /usr/bin/editor is used.

  The -c option modifies the -r option to prompt the user for a 'y/Y' response before actually removing the crontab.

DEBIAN SPECIFIC
  The "out-of-the-box" behaviour for crontab -l is to display the three line "DO NOT EDIT THIS FILE" header that is placed at the beginning of the crontab when it is installed. The problem is that it makes the sequence

  crontab -l | crontab -
non-idempotent - you keep adding copies of the header. This causes pain to scripts that use sed to edit a crontab. Therefore, the default behaviour of the -l option has been changed to not output such header. You may obtain the original behaviour by setting the environment variable CRONTAB_NOCHEADER to '1', which will cause the crontab -l command to emit the extraneous header.
  man(1) man(5) (man(1) for help on file(5))
```

```
$ crontab
```

```
bhavesh@bhavesh-VirtualBox:~$ crontab -e
no crontab for bhavesh - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano      <---- easiest
 2. /usr/bin/vim.basic
 3. /usr/bin/vim.tiny
 4. /bin/ed

Choose 1-4 [1]: 1
No modification made
bhavesh@bhavesh-VirtualBox:~$
```

```
bhavesh@bhavesh-VirtualBox:~$ nano .crontab
Edit this file to introduce tasks to be run by cron.

Each task to run has to be defined through a single line
indicating with different fields when the task will be run
and what command to run for the task

To define the time you can provide concrete values for
minute (m), hour (h), day of month (dom), month (mon),
and day of week (dow) or use '*' in these fields (for 'any').

Notice that tasks will be started based on the cron's system
daemon's notion of time and timezones.

Output of the crontab jobs (including errors) is sent through
email to the user the crontab file belongs to (unless redirected).

For example, you can run a backup of all your user accounts
at 5 a.m every week with:
0 5 * * 1 tar -zcf /var/backups/home.tgz /home/

For more information see the manual pages of crontab(5) and cron(8)

m h dom mon dow   command
```

nn. mount : The structure of all files which can be over multiple devices in the system is like a big tree rooted at “/”. The task of mount command is to place the files and sub files in the big tree hierarchy.

```
$ man mount
```

```

bhavesh@bhavesh-VirtualBox: ~
MOUNT(8)                               System Administration                         MOUNT(8)

NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-l] [-t fstype]
    mount [-f|fsvsw] [-t fstype] [-o options]
    mount [-f|fsvsw] [-o options] device[mountpoint]
    mount [-f|fsvsw] [-t fstype] [-o options] device mountpoint
    mount --bind[-r|rbnd]--move olddir newdir
    mount --make[shared|slave|private|unbindable|rshared|rslave|rprivate|runbindable] mountpoint

DESCRIPTION
All files accessible in a Unix system are arranged in one big tree, the file hierarchy, rooted at /. These files can be spread out over several devices. The mount command serves to attach the filesystem found on some device to the big file tree; conversely, the umount(8) command will detach it again. The filesystem is used to control how data is stored on the device or provided in a virtual way by network or other services.

The standard form of the mount command is:
    mount -t type device dir

This tells the kernel to attach the filesystem found on device (which is of type type) at the directory dir. The option -t type is optional. The mount command is usually able to detect a filesystem. The root permissions are necessary to mount a filesystem by default. See section "Non-superuser mounts" below for more details. The previous contents (if any) and owner and mode of dir will become invisible, and as long as this filesystem remains mounted, the pathname dir refers to the root of the filesystem on device.

If only the directory or the device is given, for example:
    mount /dir

then mount looks for a mountpoint (and if not found then for a device) in the /etc/fstab file. It's possible to use the --target or --source options to avoid ambiguous interpretation of the given argument. For example:
    mount --target /mountpoint

The same filesystem may be mounted more than once, and in some cases (e.g., network filesystems) the same filesystem may be mounted on the same mountpoint multiple times. The mount command does not implement any policy to control this behavior. All behavior is controlled by the kernel and it is usually specific to the filesystem driver. The exception is --all, in this case already mounted filesystems are ignored (see --all below for more details).

listing the mounts
Manual page mount(8).line 1 (press h for help or q to quit)

```

\$ mount

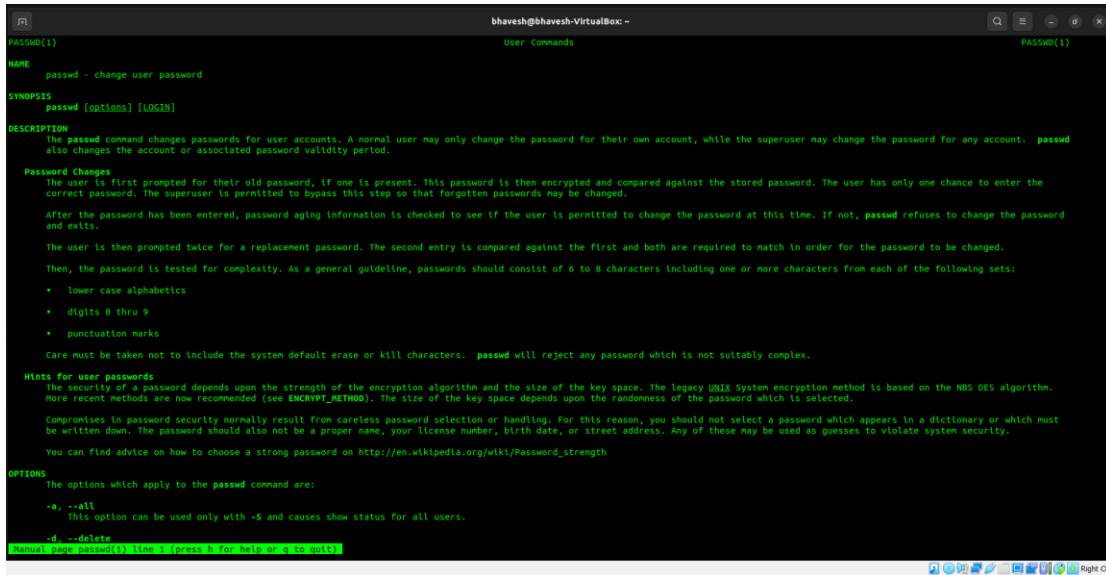
```

bhavesh@bhavesh-VirtualBox: ~
MOUNT(8)                               System Administration                         MOUNT(8)

bhavesh@bhavesh-VirtualBox: ~ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devpts (rw,nosuid,relatime,size=1973804k,nr_inodes=493451,mode=755,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=401824k,mode=755,inode64)
/dev/sda3 on / type ext4 (rw,relatime,errors=remount-ro)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,inode64)
tmpfs on /dev/pts type tmpfs (rw,nosuid,noexec,relatime,size=5120k,inode64)
cgroup on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,nsdelegate)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime)
fstab on /sys/fs/fstab type pstore (rw,nosuid,nodev,noexec,relatime)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=29,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=16590)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime)
tracers on /sys/kernel/tracing type tracers (rw,nosuid,nodev,noexec,relatime)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
autofs on /sys/fs/autofs type autofs (rw,nosuid,nodev,noexec,relatime)
autofs on /sys/fs/autofs type autofs (rw,nosuid,nodev,noexec,relatime)
queues on /dev/queue type queue (rw,nosuid,nodev,noexec,relatime)
none on /run/credentials/systemd-susers.service type ramfs (ro,nosuid,nodev,noexec,relatime,mode=700)
/var/lib/snapd/snaps/bare_5.snap on /snap/bare/5 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/snapd-desktop-integration_14.snap on /snap/snapd-desktop-integration/14 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/core20_1611.snap on /snap/core20/1611 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/core20_1587.snap on /snap/core20/1587 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/firefox_1635.snap on /snap/firefox/1635 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/snapd_16292.snap on /snap/snapd/16292 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/gnome-shell_112.snap on /snap/gnome-shell/112 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/gnome-shell_111.snap on /snap/gnome-shell/111 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/gnome-3-38-2004_112.snap on /snap/gnome-3-38-2004/112 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/gnome-3-38-2004_115.snap on /snap/gnome-3-38-2004/115 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/var/lib/snapd/snaps/firefox_1775.snap on /snap/firefox/1775 type squashfs (ro,nodev,relatime,errors=continue,x-gdu.hide)
/dev/sda2 on /boot/efi type vfat (rw,relatime,fmask=0x077,dmask=0x077,codepage=437,lochardset=iso8859-1,shortname=mixed,errors=remount-ro)
tmpfs on /run/snapd/ns type tmpfs (rw,nosuid,nodev,noexec,relatime,size=401824k,mode=755,inode64)
nsfs on /run/snapd/ns/snapd-desktop-integration.mnt type nsfs (rw,
  mfs on /run/user/1000 type tmpfs (ro,nosuid,nodev,relatime, size=401824k, nr_inodes=100055, mode=700, uid=1000, gid=1000, inode64)
```

oo. passwd : This command gives the user the option to change their password. It asks for an old password if one already exists.

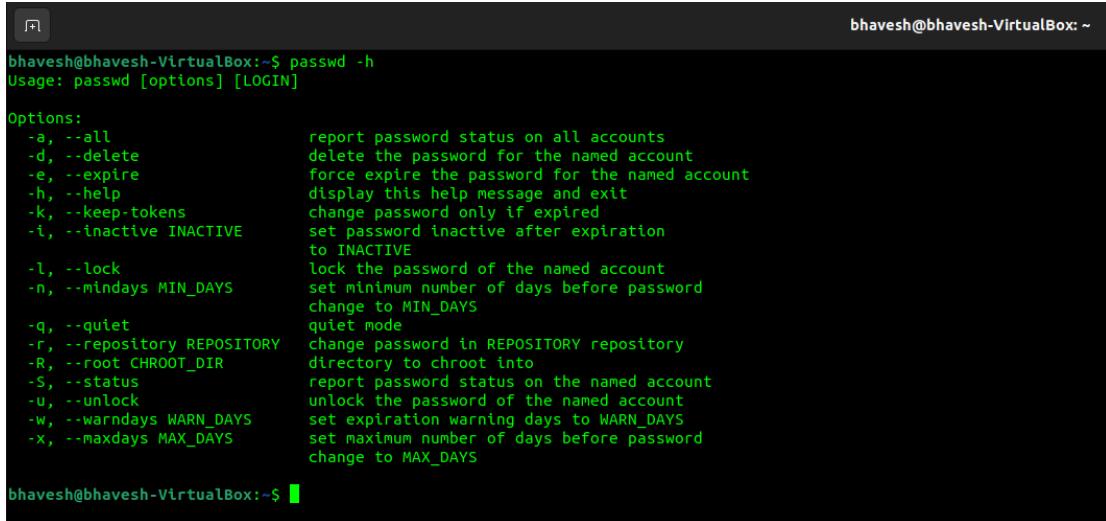
```
$ man passwd
```



The screenshot shows a terminal window with the man page for the passwd command. The title bar says "User Commands". The page content includes:

- NAME**: passwd - change user password
- SYNOPSIS**: passwd [options] [LOGIN]
- DESCRIPTION**: The passwd command changes passwords for user accounts. A normal user may only change the password for their own account, while the superuser may change the password for any account. passwd also changes the account or associated password validity period.
- Password Changes**: The user is first prompted for their old password, if one is present. This password is then encrypted and compared against the stored password. The user has only one chance to enter the correct password. The superuser is permitted to bypass this step so that forgotten passwords may be changed. After the password has been entered, password aging information is checked to see if the user is permitted to change the password at this time. If not, passwd refuses to change the password and exits. The user is then prompted twice for a replacement password. The second entry is compared against the first and both are required to match in order for the password to be changed. Then, the password is tested for complexity. As a general guideline, passwords should consist of 6 to 8 characters including one or more characters from each of the following sets:
  - lower case alphabets
  - digits 0 thru 9
  - punctuation marksCare must be taken not to include the system default erase or kill characters. passwd will reject any password which is not suitably complex.
- Hints for user passwords**: The security of a password depends upon the strength of the encryption algorithm and the size of the key space. The legacy UNIX System encryption method is based on the NBS DES algorithm. More recent methods are now recommended (see ENCRYPT\_METHOD). The size of the key space depends upon the randomness of the password which is selected. Compromises in password security normally result from careless password selection or handling. For this reason, you should not select a password which appears in a dictionary or which must be written down. The password should also not be a proper name, your license number, birth date, or street address. Any of these may be used as guesses to violate system security. You can find advice on how to choose a strong password on [http://en.wikipedia.org/wk1/Password\\_strength](http://en.wikipedia.org/wk1/Password_strength)
- OPTIONS**: The options which apply to the passwd command are:
  - a, --all: This option can be used only with -S and causes show status for all users.
  - d, --delete: Manual page passwd(1) line 1 (press h for help or q to quit)

```
$ passwd
```



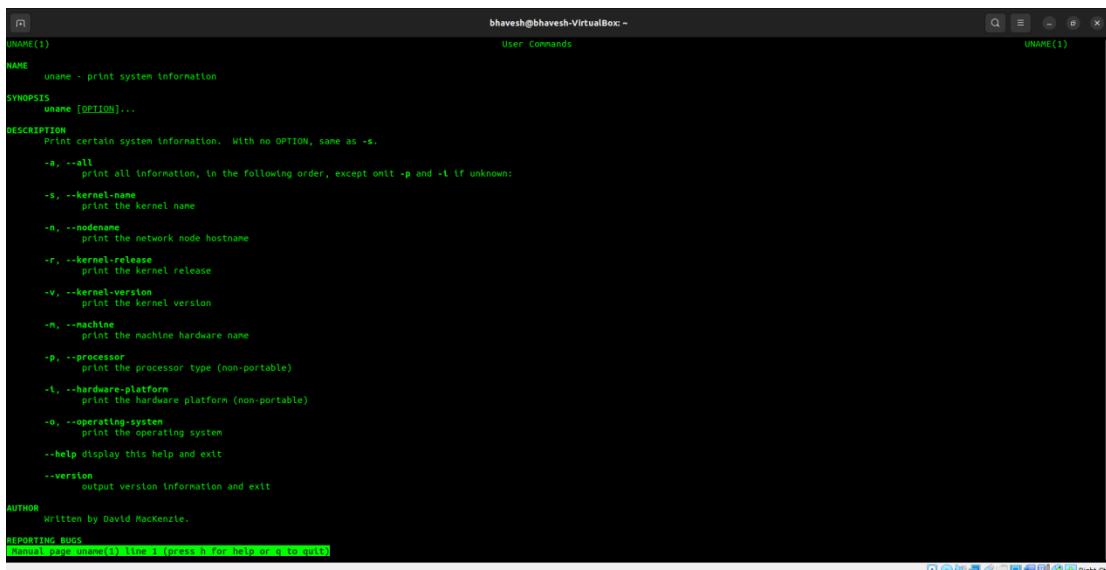
The screenshot shows a terminal window with the usage information for the passwd command. The title bar says "bhavesh@bhavesh-VirtualBox: ~". The output is:

```
Usage: passwd [options] [LOGIN]

Options:
-a, --all          report password status on all accounts
-d, --delete       delete the password for the named account
-e, --expire       force expire the password for the named account
-h, --help         display this help message and exit
-k, --keep-tokens change password only if expired
-i, --inactive INACTIVE set password inactive after expiration
                      to INACTIVE
-l, --lock          lock the password of the named account
-n, --mindays MIN_DAYS set minimum number of days before password
                      change to MIN_DAYS
-q, --quiet        quiet mode
-f, --repository REPOSITORY change password in REPOSITORY repository
-R, --root CHROOT_DIR directory to chroot into
-S, --status        report password status on the named account
-u, --unlock       unlock the password of the named account
-w, --warndays WARN_DAYS set expiration warning days to WARN_DAYS
-x, --maxdays MAX_DAYS set maximum number of days before password
                      change to MAX_DAYS
```

pp. uname : This command displays the basic information about the system in use.

```
$ man uname
```



A screenshot of a terminal window titled "bhavesh@bhavesh-VirtualBox: ~". The window shows the man page for the "uname" command. The text is as follows:

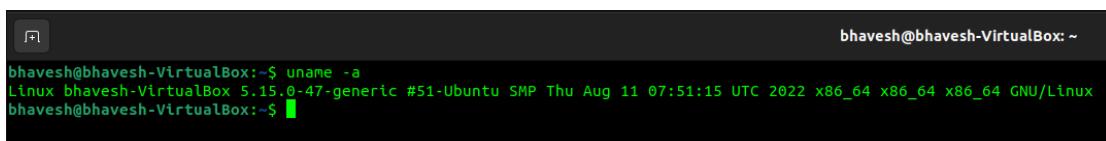
```
UNAME(1)
NAME      uname - print system information
SYNOPSIS  uname [OPTION]...
DESCRIPTION
Print certain system information. With no OPTION, same as -s.

-a, --all
        print all information, in the following order, except omit -p and -t if unknown:
-k, --kernel-name
        print the kernel name
-n, --nodename
        print the network node hostname
-r, --kernel-release
        print the kernel release
-v, --kernel-version
        print the kernel version
-m, --machine
        print the machine hardware name
-p, --processor
        print the processor type (non-portable)
-l, --hardware-platform
        print the hardware platform (non-portable)
-o, --operating-system
        print the operating system
--help display this help and exit
--version
        output version information and exit

AUTHOR   Written by David MacKenzie.

REPORTING BUGS
Report bugs to <mailto:uname@FreeBSD.org> (press h for help or q to quit).
```

```
$ uname
```

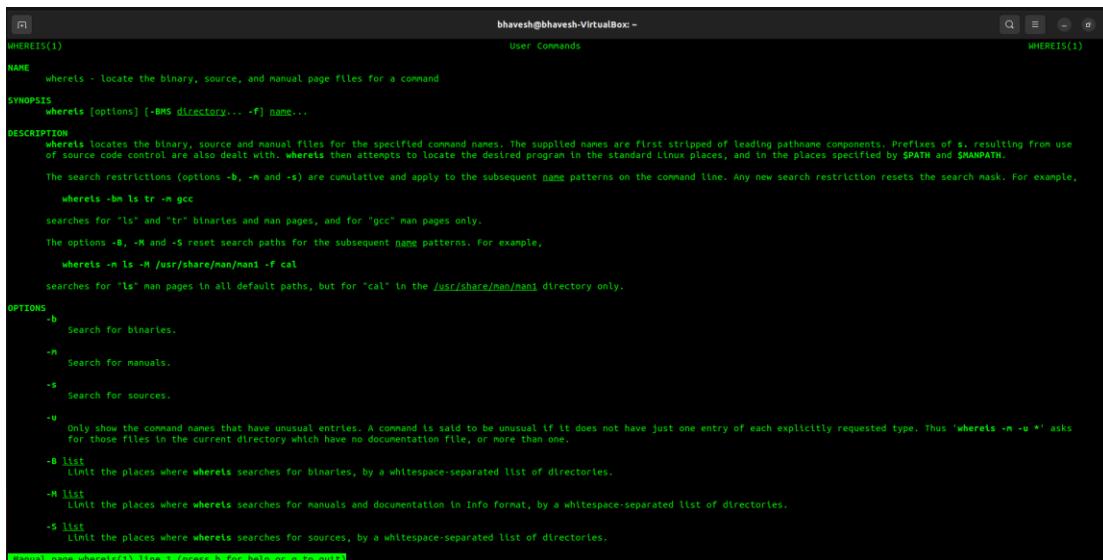


A screenshot of a terminal window titled "bhavesh@bhavesh-VirtualBox: ~". The window shows the output of the "uname -a" command. The text is as follows:

```
bhavesh@bhavesh-VirtualBox:~$ uname -a
Linux bhavesh-VirtualBox 5.15.0-47-generic #51-Ubuntu SMP Thu Aug 11 07:51:15 UTC 2022 x86_64 x86_64 x86_64 GNU/Linux
bhavesh@bhavesh-VirtualBox:~$
```

qq. whereis : whereis command literally tells you where can you find the file you are searching. It works like a select query, just enter the filename and it will tell you the path where the file is located.

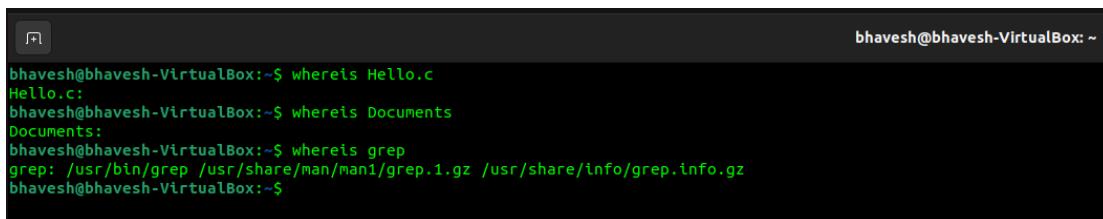
```
$ man whereis
```



The screenshot shows a terminal window with the man page for the 'whereis' command. The title bar says 'WHEREIS(1)'. The page content includes:

- NAME**: whereis - locate the binary, source, and manual page files for a command
- SYNOPSIS**: whereis [options] [-BMS directory... -F] name...
- DESCRIPTION**: whereis locates the binary, source and manual files for the specified command names. The supplied names are first stripped of leading pathname components. Prefixes of s, resulting from use of source code control are also dealt with. whereis then attempts to locate the desired program in the standard Linux places, and in the places specified by \$PATH and \$MANPATH.
- The search restrictions (options -B, -M and -S) are cumulative and apply to the subsequent name patterns on the command line. Any new search restriction resets the search mask. For example,  
`whereis -Bm ls tr -m gcc`  
searches for "ls" and "tr" binaries and man pages, and for "gcc" man pages only.
- The options -B, -M and -S reset search paths for the subsequent name patterns. For example,  
`whereis -m ls -M /usr/share/man/man1 -f cal`  
searches for "ls" man pages in all default paths, but for "cal" in the /usr/share/man/man1 directory only.
- OPTIONS**:
  - B Search for binaries.
  - M Search for manuals.
  - S Search for sources.
  - U Only show the command names that have unusual entries. A command is said to be unusual if it does not have just one entry of each explicitly requested type. Thus 'whereis -m -s \*' asks for those files in the current directory which have no documentation file, or more than one.
  - B list Limit the places where whereis searches for binaries, by a whitespace-separated list of directories.
  - M list Limit the places where whereis searches for manuals and documentation in Info format, by a whitespace-separated list of directories.
  - S list Limit the places where whereis searches for sources, by a whitespace-separated list of directories.

```
$ whereis
```



The screenshot shows a terminal window with the output of the 'whereis' command for several files. The commands entered were:

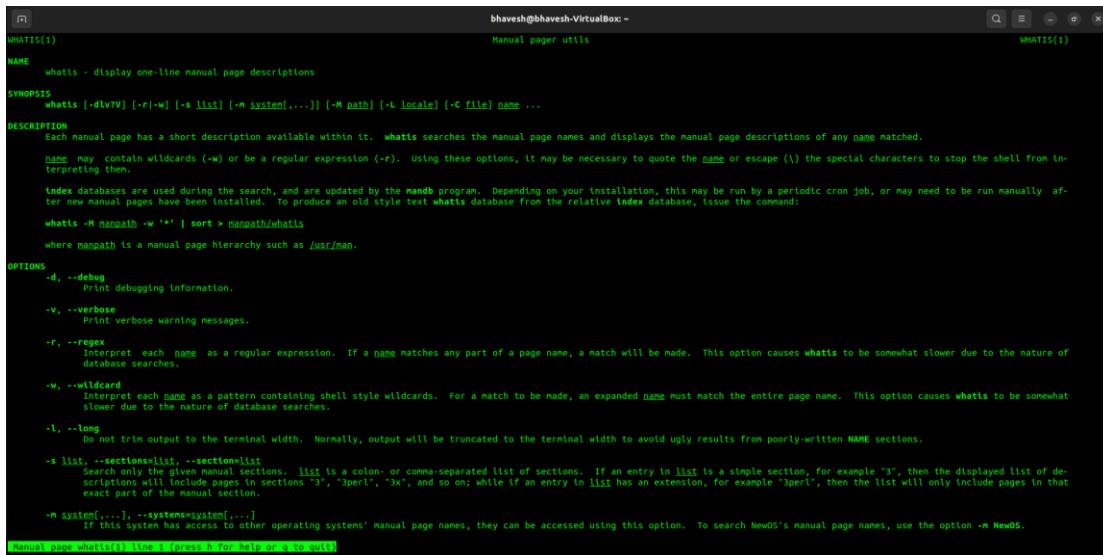
- whereis Hello.c
- whereis Documents
- whereis grep

The output shows the locations of the files:

- Hello.c: Found in /usr/include/hello.c
- Documents: Found in /usr/share/doc/
- grep: Found in /usr/bin/grep, /usr/share/man/man1/grep.1.gz, and /usr/share/info/grep.info.gz

rr. whatis : This command is used to get a brief introduction of any command. It displays the one line description of the command from the manual page.

```
$ man whatis
```



The screenshot shows a terminal window titled "WHATIS(1)" with the command "man whatis" entered. The output displays the man page for the "whatis" command, which is a manual pager utility. The page includes sections for NAME, SYNOPSIS, DESCRIPTION, and OPTIONS, along with detailed explanations of each option and its behavior. The terminal window has a standard OS X interface with a title bar, menu bar, and scroll bars.

```
WHATIS(1)                               Manual pager utils                               WHATIS(1)

NAME
    whatis - display one-line manual page descriptions

SYNOPSIS
    whatis [-dIv?W] [-r|-W] [-s list] [-M system[,...]] [-L locale] [-C file] name ...

DESCRIPTION
    Each manual page has a short description available within it.  whatis searches the manual page names and displays the manual page descriptions of any name matched.  name may contain wildcards (-w) or be a regular expression (-r).  Using these options, it may be necessary to quote the name or escape () the special characters to stop the shell from interpreting them.

    Index databases are used during the search, and are updated by the man-db program.  Depending on your installation, this may be run by a periodic cron job, or may need to be run manually after new manual pages have been installed.  To produce an old style text whatis database from the relative index database, issue the command:

        whatis -M manpath -w '*' | sort > manpath/whatis

    where manpath is a manual page hierarchy such as /usr/man.

OPTIONS
-d, --debug
    Print debugging information.

-v, --verbose
    Print verbose warning messages.

-r, --regex
    Interpret each name as a regular expression.  If a name matches any part of a page name, a match will be made.  This option causes whatis to be somewhat slower due to the nature of database searches.

-W, --wildcard
    Interpret each name as a pattern containing shell style wildcards.  For a match to be made, an expanded name must match the entire page name.  This option causes whatis to be somewhat slower due to the nature of database searches.

-l, --long
    Do not trim output to the terminal width.  Normally, output will be truncated to the terminal width to avoid ugly results from poorly-written NAME sections.

-s list, --sections=list, --section=list
    list only contains manual sections.  list is a colon- or comma-separated list of sections.  If an entry in list is a single section, for example "3", then the displayed list of descriptions will include pages in sections "3", "3perl", "3x", and so on; while if an entry in list has an extension, for example "3perl", then the list will only include pages in that exact part of the manual section.

-M system[,...], --systems=sytem[,...]
    If this system has access to other operating systems' manual page names, they can be accessed using this option.  To search NewOS's manual page names, use the option -M NewOS.

Manual page whatis(1) line 1 (press h for help or q to quit)
```

```
$ whatis
```



```
bhavesh@bhavesh-VirtualBox:~$ whatis ls
ls (1)           - list directory contents
bhavesh@bhavesh-VirtualBox:~$ █
```

A screenshot of a terminal window titled 'Terminal'. The window shows a single line of text: 'bhavesh@bhavesh-VirtualBox:~\$ whatis ls'. Below this, the output of the 'whatis ls' command is displayed: 'ls (1) - list directory contents'. The terminal window has a dark background with light-colored text.

ss. su : su or most commonly known as Switch-User command is used to change the user or substitute a user/group to run a command. When no user is specified, by default su - is fired.

```
$ man su
```

```
NAME      su - run a command with substitute user and group ID
SYNOPSIS  su [options] [-] [user [argument...]]
DESCRIPTION
    su allows commands to be run with a substitute user and group ID.

    When called with no user specified, su defaults to running an interactive shell as root. When user is specified, additional arguments can be supplied, in which case they are passed to the shell.

    For backward compatibility, su defaults to not change the current directory and to only set the environment variables HOME and SHELL (plus USER and LOGNAME if the target user is not root). It is recommended to always use the --login option (instead of its shortcut -) to avoid side effects caused by mixing environments.

    This version of su uses PAM for authentication, account and session management. Some configuration options found in other su implementations, such as support for a wheel group, have to be configured via PAM.

    su is mostly designed for unprivileged users, the recommended solution for privileged users (e.g., scripts executed by root) is to use non-set-user-ID command runuser() that does not require authentication and provides separate PAM configuration. If the PAM session is not required at all then the recommended solution is to use command setpriv().

    Note that su in all cases uses PAM (pam_getenvlist()) to do the final environment modification. Command-line options such as --login and --preserve-environment affect the environment before it is modified by PAM.

OPTIONS
    -c, --command=command
        Pass command to the shell with the -c option.

    -f, --fast
        Pass -f to the shell, which may or may not be useful, depending on the shell.

    -g, --group=group
        Specify the primary group. This option is available to the root user only.

    -G, --supp-group=group
        Specify a supplementary group. This option is available to the root user only. The first specified supplementary group is also used as a primary group if the option --group is not specified.

    -, -l, --login
        Start the shell as a login shell with an environment similar to a real login:
        +   clears all the environment variables except TERM and variables specified by --whitelist-environment
        +   initializes the environment variables HOME, SHELL, USER, LOGNAME, and PATH

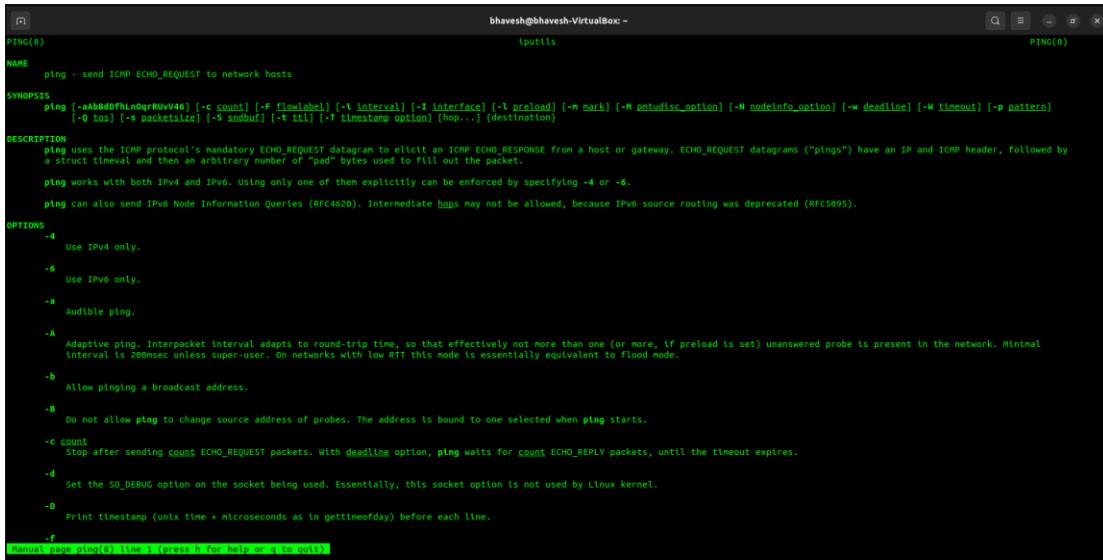
Manual page su(1), line 1 (press h for help or q to quit)
```

\$ su

```
bhavesh@bhavesh-VirtualBox:~$ sudo -i
[sudo] password for bhavesh:
root@bhavesh-VirtualBox:~# su bhavesh
bhavesh@bhavesh-VirtualBox:~/root$ cd /home/bhavesh/
bhavesh@bhavesh-VirtualBox:~$ pw
pw: command not found
bhavesh@bhavesh-VirtualBox:~$ pwd
/home/bhavesh
bhavesh@bhavesh-VirtualBox:~$
```

**tt. ping :** Ping command is used to check the connectivity between host and server/host. It does so by sending a packet to the input IP/URL and gets a response from the server/host.

```
$ man ping
```



The screenshot shows a terminal window with the title "ping(8)" and the subtitle "lputils". The content of the window is the man page for the ping command. It includes sections for NAME, SYNOPSIS, DESCRIPTION, OPTIONS, and EXAMPLES. The SYNOPSIS section shows the command line syntax with various options like -c, -f, -I, -l, -M, -N, -W, -w, -A, -B, -c, -d, -t, and -F. The DESCRIPTION section explains the ICMP protocol and the purpose of the command. The OPTIONS section lists specific flags such as -4, -6, -B, -A, -B, -c, -d, -t, and -F. The EXAMPLES section shows how to use the command to ping a website.

```
PING(8)                                         lputils                                         PING(8)

NAME
      ping - send ICMP ECHO_REQUEST to network hosts

SYNOPSIS
      ping [-mb8d0ff] [-mgrnvv4] [-c count] [-f [loopback]] [-I interface] [-l preload] [-M mark] [-M pmtdisc_option] [-N nodeinfo_option] [-w deadline] [-W timeout] [-p pattern]
              [-q ttl] [-F socketlist] [-S source] [-T ttl] [-F [multicast option]] [hop...]
              [destination]

DESCRIPTION
      ping uses the ICMP protocol's mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway. ECHO_REQUEST datagrams ("pings") have an IP and ICMP header, followed by a struct timeval and then an arbitrary number of "pad" bytes used to fill out the packet.

      ping works with both IPv4 and IPv6. Using only one of them explicitly can be enforced by specifying -4 or -6.

      ping can also send IPv6 Node Information Queries (RFC4620). Intermediate hops may not be allowed, because IPv6 source routing was deprecated (RFC5095).

OPTIONS
      -4
          Use IPv4 only.

      -6
          Use IPv6 only.

      -B
          Audible ping.

      -A
          Adaptive ping. Interpacket interval adapts to round-trip time, so that effectively not more than one (or more, if preload is set) unanswered probe is present in the network. Minimal interval is 200ms unless super-user. On networks with low RTT this mode is essentially equivalent to flood mode.

      -B
          Allow pinging a broadcast address.

      -B
          Do not allow ping to change source address of probes. The address is bound to one selected when ping starts.

      -c count
          Stop after sending count ECHO_REQUEST packets. With deadline option, ping waits for count ECHO_REPLY packets, until the timeout expires.

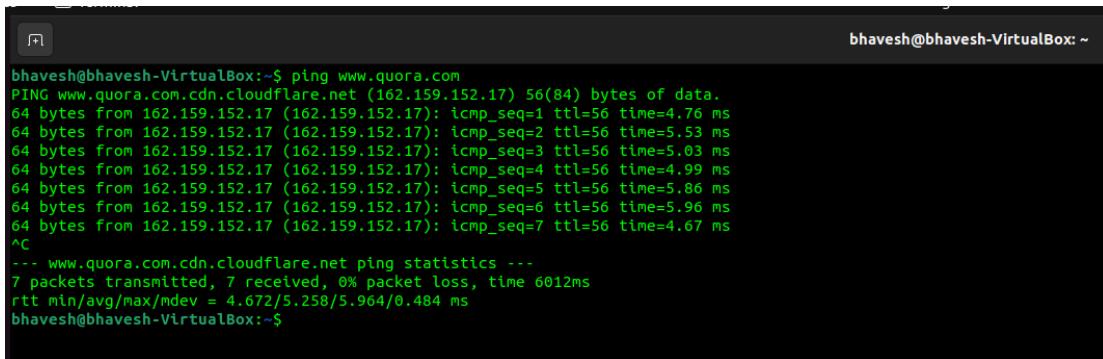
      -d
          Set the SO_DEBUG option on the socket being used. Essentially, this socket option is not used by Linux kernel.

      -d
          Print timestamp (unix time + microseconds as in gettimeofday) before each line.

      -F
          Manual page ping(8). Press h for help or q to quit.

      Manual page ping(8). Press h for help or q to quit.
```

```
$ ping
```

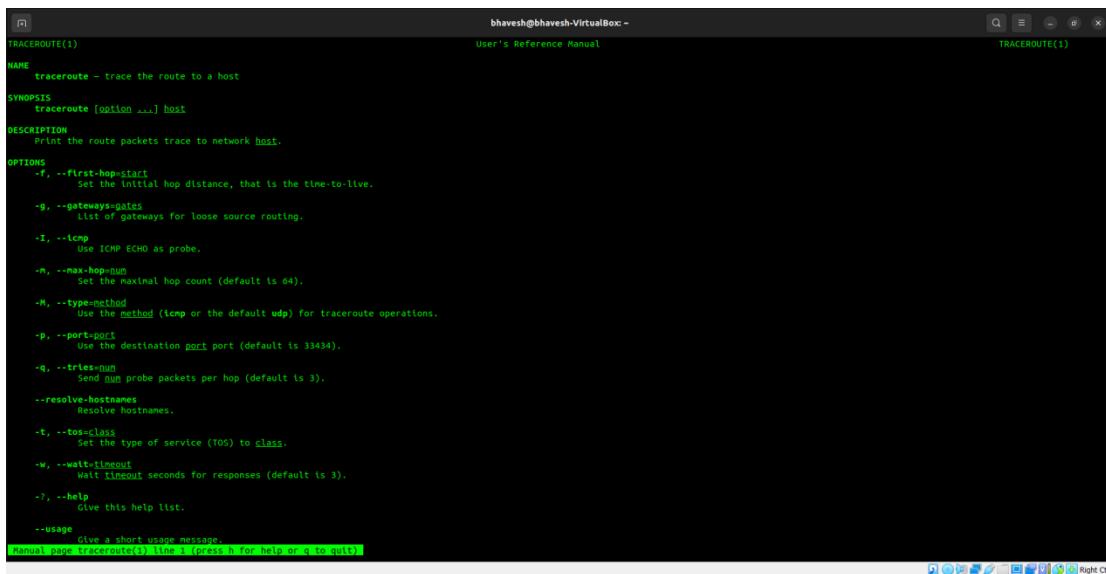


The screenshot shows a terminal window with the title "bhavesh@bhavesh-VirtualBox:~" and the subtitle "lputils". The content of the window is the output of the ping command to the website www.quora.com. It shows the transmission of 7 packets, all received successfully with 0% loss, over a time of 6012ms. The statistics show an average round trip time (rtt) of 4.672ms.

```
ping www.quora.com.cdn.cloudflare.net (162.159.152.17) 56(84) bytes of data.
64 bytes from 162.159.152.17 (162.159.152.17): icmp_seq=1 ttl=56 time=4.76 ms
64 bytes from 162.159.152.17 (162.159.152.17): icmp_seq=2 ttl=56 time=5.53 ms
64 bytes from 162.159.152.17 (162.159.152.17): icmp_seq=3 ttl=56 time=5.03 ms
64 bytes from 162.159.152.17 (162.159.152.17): icmp_seq=4 ttl=56 time=4.99 ms
64 bytes from 162.159.152.17 (162.159.152.17): icmp_seq=5 ttl=56 time=5.86 ms
64 bytes from 162.159.152.17 (162.159.152.17): icmp_seq=6 ttl=56 time=5.96 ms
64 bytes from 162.159.152.17 (162.159.152.17): icmp_seq=7 ttl=56 time=4.67 ms
^C
--- www.quora.com.cdn.cloudflare.net ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6012ms
rtt min/avg/max/mdev = 4.672/5.258/5.964/0.484 ms
bhavesh@bhavesh-VirtualBox:~$
```

uu. traceroute : This command helps us to get information about the path and the hops that a packet goes through to get to the mentioned URL.

```
$ man traceroute
```



A screenshot of a terminal window titled "bhavesh@bhavesh-VirtualBox ~". The window displays the man page for the "traceroute" command. The title bar also shows "User's Reference Manual" and "TRACEROUTE(1)". The man page content includes sections for NAME, SYNOPSIS, DESCRIPTION, and OPTIONS, detailing various command-line flags and their meanings. At the bottom of the window, there is a status bar with icons and the text "Terminal Page: traceroute(1) Line: 1 (Press h for help or q to quit)".

```
bhavesh@bhavesh-VirtualBox ~
TRACEROUTE(1)                               User's Reference Manual
TRACEROUTE(1)

NAME
       traceroute - trace the route to a host

SYNOPSIS
       traceroute [option ...] host

DESCRIPTION
       Print the route packets trace to network host.

OPTIONS
       -F, --first-hop=ttl
           set the initial hop distance, that is the time-to-live.

       -g, --gateways=gates
           List of gateways for loose source routing.

       -I, --icmp
           Use ICMP ECHO as probe.

       -m, --max-hop=nm
           Set the maximal hop count (default is 64).

       -M, --type=method
           Use the method (icmp or the default udp) for traceroute operations.

       -P, --port=port
           Use the destination port (default is 33434).

       -q, --tries=n
           Send num probe packets per hop (default is 3).

       --resolve-hostnames
           Resolve hostnames.

       -t, --tos=class
           Set the type of service (TOS) to class.

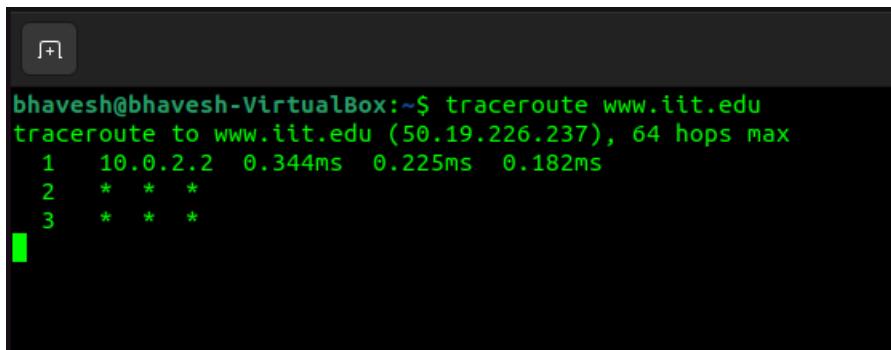
       -w, --wait-timeout
           Wait timeout seconds for responses (default is 3).

       -?, --help
           Give this help list.

       --usage
           Give a short usage message.

Terminal Page: traceroute(1) Line: 1 (Press h for help or q to quit)
```

```
$ traceroute
```

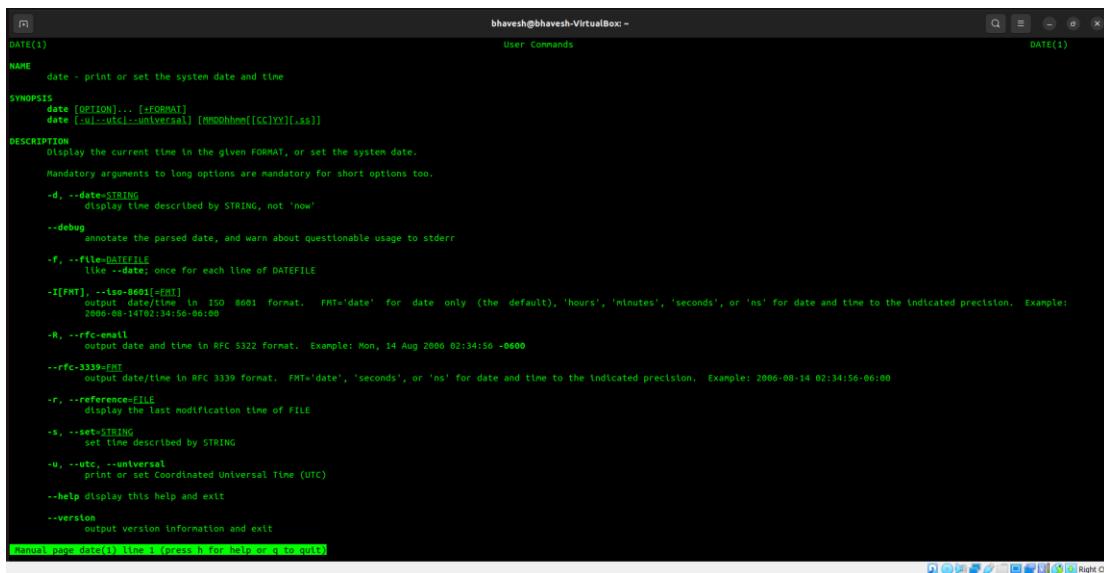


A screenshot of a terminal window showing the output of the "traceroute" command. The window title is "bhavesh@bhavesh-VirtualBox ~". The command "traceroute www.iit.edu" was run, and the output shows the path to the target host, www.iit.edu (50.19.226.237), with a maximum of 64 hops. The first three hops are explicitly listed: 1 10.0.2.2 0.344ms 0.225ms 0.182ms, 2 \* \* \*, and 3 \* \*. The terminal window has a dark background and light-colored text.

```
bhavesh@bhavesh-VirtualBox:~$ traceroute www.iit.edu
traceroute to www.iit.edu (50.19.226.237), 64 hops max
 1  10.0.2.2  0.344ms  0.225ms  0.182ms
 2  * * *
 3  * *
```

vv. date : This command displays the date and time. It can also be used to set the date and time if you are a superuser.

```
$ man date
```



The screenshot shows a terminal window with the title "DATE(1)" and the subtitle "User Commands". The content of the window is the man page for the "date" command. The man page starts with a brief description of what the command does: "print or set the system date and time". It then provides a synopsis of the command: "date [OPTION]... [-FORMAT]" and "date [-u|--utc|--universal] [MMDDhhmm[CC]YY][.ss]". The "DESCRIPTION" section is quite detailed, explaining various options like "-d", "--date=STRING", "--debug", and "--reference=FILE". It also covers output formats such as ISO 8601, RFC 3339, and RFC 822. The man page ends with a note about the "date" command being part of the "coreutils" package.

```
bhavesh@bhavesh-VirtualBox:~
```

```
DATE(1)
```

```
NAME
    date - print or set the system date and time

SYNOPSIS
    date [OPTION]... [-FORMAT]
    date [-u|--utc|--universal] [MMDDhhmm[CC]YY][.ss]

DESCRIPTION
    Display the current time in the given FORMAT, or set the system date.
    Mandatory arguments to long options are mandatory for short options too.

    -d, --date=STRING
        display time described by STRING, not 'now'

    --debug
        annotate the parsed date, and warn about questionable usage to stderr

    -f, --file=DATEFILE
        like --date; once for each line of DATEFILE

    -I[FMT], --iso-8601[=FMT]
        output date/time in ISO 8601 format. FMT='date' for date only (the default), 'hours', 'minutes', 'seconds', or 'ns' for date and time to the indicated precision. Example: 2006-08-14T02:34:56-06:00

    -R, --rfc-email
        output date and time in RFC 5322 format. Example: Mon, 14 Aug 2006 02:34:56 -0600

    --rfc-3339[=FMT]
        output date/time in RFC 3339 format. FMT='date', 'seconds', or 'ns' for date and time to the indicated precision. Example: 2006-08-14 02:34:56-06:00

    -r, --reference=FILE
        display the last modification time of FILE

    -s, --set=STRING
        set time described by STRING

    -u, --utc, --universal
        print or set Coordinated Universal Time (UTC)

    --help display this help and exit

    --version
        output version information and exit

    date(1) | iso-8601(7) | rfc3339(7) | rfc822(7)

```

```
$ date
```

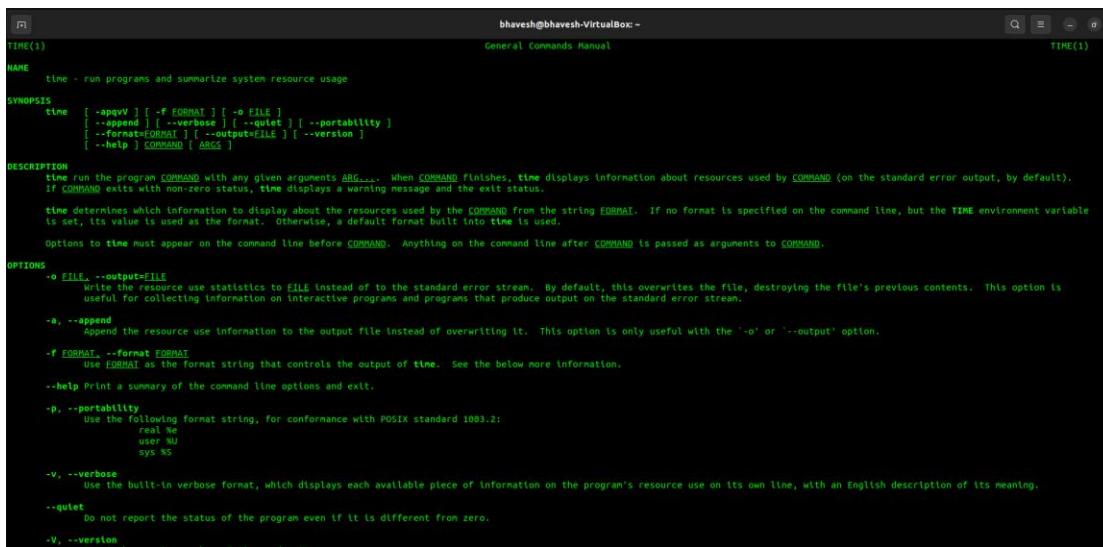


The screenshot shows a terminal window with the title "bhavesh@bhavesh-VirtualBox:~". The command "date" was run, and the output was "Wed Aug 31 11:02:26 PM CDT 2022". The terminal window has a dark background with light-colored text.

```
bhavesh@bhavesh-VirtualBox:~$ date
Wed Aug 31 11:02:26 PM CDT 2022
bhavesh@bhavesh-VirtualBox:~$
```

ww. time : This command returns the time required for an command/process to run.

```
$ man time
```



The screenshot shows a terminal window with the title bar "bhavesh@bhavesh-VirtualBox: ~" and the window title "TIME(1)". The content of the window is the man page for the "time" command, titled "General Commands Manual". The man page includes sections for SYNOPSIS, DESCRIPTION, and OPTIONS. The SYNOPSIS section shows the command usage: "time [ -o FILE ] [ -f FORMAT ] [ -a ] [ --append ] [ --verbose ] [ --quiet ] [ --portability ] [ --format=FORMAT ] [ --output=FILE ] [ --version ] [ --help ] COMMAND [ ARGS ]". The DESCRIPTION section explains the purpose of the command: "time runs the program COMMAND with any given arguments ARG... . When COMMAND finishes, time displays information about resources used by COMMAND (on the standard error output, by default). If COMMAND exits with non-zero status, time displays a warning message and the exit status." It also describes the "FORMAT" option and how it interacts with the TIME environment variable. The OPTIONS section lists various flags and their descriptions, such as "-o" for writing to a file, "-f" for specifying a format string, and various verbosity and portability options.

```
TIME(1)
bhavesh@bhavesh-VirtualBox: ~
General Commands Manual
TIME(1)

NAME
    time - run programs and summarize system resource usage

SYNOPSIS
    time [ -o FILE ] [ -f FORMAT ] [ -a ]
    [ --append ] [ --verbose ] [ --quiet ] [ --portability ]
    [ --format=FORMAT ] [ --output=FILE ] [ --version ]
    [ --help ] COMMAND [ ARGS ]

DESCRIPTION
    time runs the program COMMAND with any given arguments ARG... .
    When COMMAND finishes, time displays information about resources used by COMMAND (on the standard error output, by default).
    If COMMAND exits with non-zero status, time displays a warning message and the exit status.
    time determines which information to display about the resources used by the COMMAND from the string FORMAT.
    If no format is specified on the command line, but the TIME environment variable is set, its value is used as the format.
    Otherwise, a default format built into time is used.
    Options to time must appear on the command line before COMMAND.
    Anything on the command line after COMMAND is passed as arguments to COMMAND.

OPTIONS
    -o FILE, --output=FILE
        Write the resource use statistics to FILE instead of to the standard error stream.
        By default, this overwrites the file, destroying the file's previous contents.
        This option is useful for collecting information on interactive programs and programs that produce output on the standard error stream.

    -a, --append
        Append the resource use information to the output file instead of overwriting it.
        This option is only useful with the '-o' or '--output' option.

    -f FORMAT, --format FORMAT
        Use FORMAT as the format string that controls the output of time.
        See the below more information.

    --help Print a summary of the command line options and exit.

    -p, --portability
        Use the following format string, for conformance with POSIX standard 1003.2:
            real Ns
            user NU
            sys NS

    -v, --verbose
        Use the built-in verbose format, which displays each available piece of information on the program's resource use on its own line, with an English description of its meaning.

    --quiet
        Do not report the status of the program even if it is different from zero.

    -V, --version
```

```
$ time
```



```
bhavesh@bhavesh-VirtualBox:~$ time sleep 1.2
real    0m1.202s
user    0m0.001s
sys     0m0.000s
bhavesh@bhavesh-VirtualBox:~$
```

xx. wget : wget is a command line utility which can be used to download from the web hence the name “get from web” (wget). It uses HTTPS, FTP, and few other protocols.

```
$ man wget
```

```

bhavesh@bhavesh-VirtualBox: ~
GNU Wget
MGET(1)

NAME      Wget - The non-interactive network downloader.

SYNOPSIS  wget [option]... [URL]...

DESCRIPTION
  GNU Wget is a free utility for non-interactive download of files from the Web. It supports HTTP, HTTPS, and FTP protocols, as well as retrieval through HTTP proxies.

  Wget is non-interactive, meaning that it can work in the background, while the user is not logged on. This allows you to start a retrieval and disconnect from the system, letting Wget finish the work. By contrast, most of the Web browsers require constant user's presence, which can be a great hindrance when transferring a lot of data.

  Wget can follow links in HTML, XHTML, and CSS pages, to create local versions of remote web sites, fully recreating the directory structure of the original site. This is sometimes referred to as "recursive downloading." While doing that, Wget respects the Robot Exclusion Standard (/robots.txt). Wget can be instructed to convert the links in downloaded files to point at the local files, for offline viewing.

  Wget has been designed for robustness over slow or unstable network connections; if a download fails due to a network problem, it will keep retrying until the whole file has been retrieved. If the server supports regetting, it will instruct the server to continue the download from where it left off.

OPTIONS
  Option Syntax
    Since Wget uses GNU getopt to process command-line arguments, every option has a long form along with the short one. Long options are more convenient to remember, but take time to type. You may freely mix different option styles, or specify options after the command-line arguments. Thus you may write:
      wget -r --tries=10 http://fly.srk.fer.hr/ -o log
    The space between the option accepting an argument and the argument may be omitted. Instead of -o log you can write -olog.
    You may put several options that do not require arguments together, like:
      wget -drc <URL>
    This is completely equivalent to:
      wget -d -r -c <URL>
    Since the options can be specified after the arguments, you may terminate them with --. So the following will try to download URL -x, reporting failure to log:
      wget -o log -- -x
    The options that accept comma-separated lists all respect the convention that specifying an empty list clears its value. This can be useful to clear the .wgetrc settings. For instance, if your .wgetrc sets "exclude_directories" to /cgi-bin, the following example will first reset it, and then set it to exclude /nobody and /somebody. You can also clear the lists in .wgetrc.
      wget -X "" -X /nobody,/somebody
  Manual page wget(1) line 3 (press h for help or q to quit)

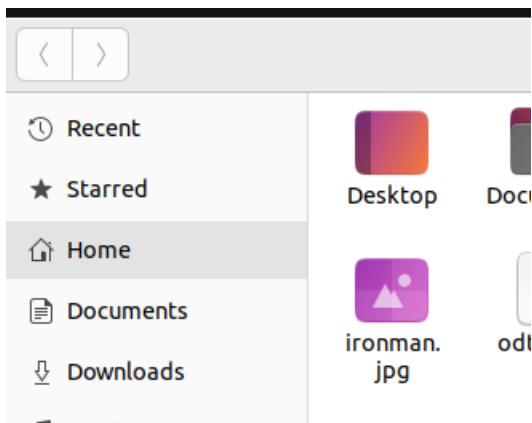
```

\$ wget

```

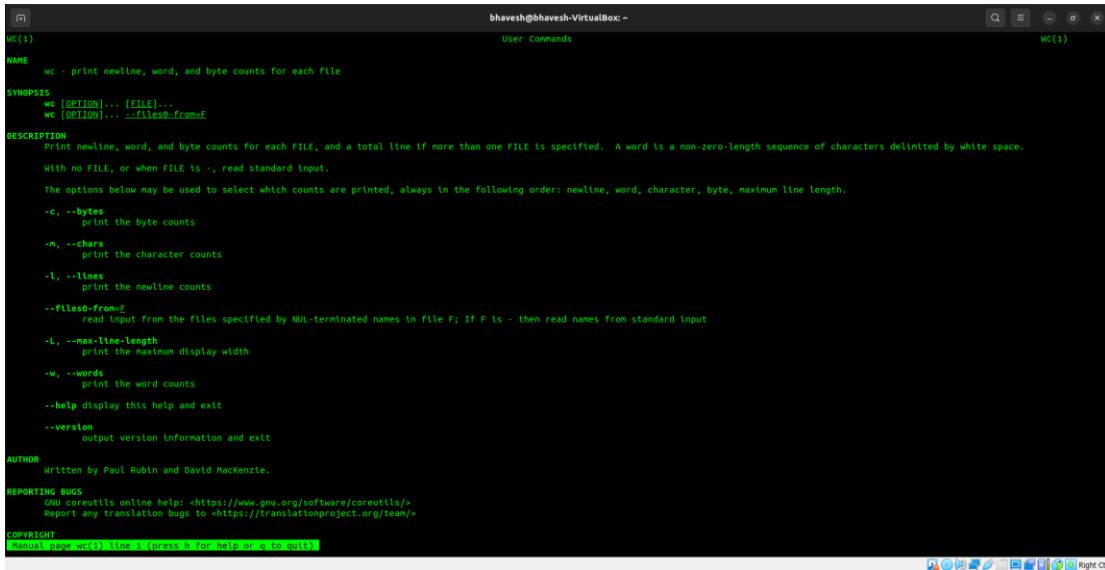
bhavesh@bhavesh-VirtualBox: ~
bhavesh@bhavesh-VirtualBox: ~$ wget -o ironman.jpg https://www.pinterest.com/pin/298363544063644704/
bhavesh@bhavesh-VirtualBox: ~

```



yy. **wc** : The wc command aka word count is used to get number of lines, words, and character count in a file.

```
$ man wc
```



The screenshot shows a terminal window titled "User Commands" with the command "wc(1)" in the title bar. The window displays the man page for the "wc" command. The text is as follows:

```
NAME      wc - print newline, word, and byte counts for each file
SYNOPSIS  wc [OPTION]... [FILE]...
          wc [OPTION]... --files0-from=F
DESCRIPTION
Print newline, word, and byte counts for each FILE, and a total line if more than one FILE is specified. A word is a non-zero-length sequence of characters delimited by white space.
With no FILE, or when FILE is -, read standard input.
The options below may be used to select which counts are printed, always in the following order: newline, word, character, byte, maximum line length.
-c, --bytes      print the byte counts
-m, --chars      print the character counts
-l, --lines      print the newline counts
--files0-from=F  read input from the files specified by NUL-terminated names in file F; If F is - then read names from standard input
-L, --max-line-length  print the maximum display width
-w, --words      print the word counts
--help           display this help and exit
--version        output version information and exit
AUTHOR     Written by Paul Rubin and David MacKenzie.
REPORTING BUGS
  GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
  Report any translation bugs to <https://translationproject.org/team/>
COPYRIGHT
  Copyright © 2010 Free Software Foundation, Inc.
  License GPLv3+: https://gnu.org/licenses/gpl.html
  This is free software: you are free to change it and/or redistribute it
  under the terms of the GNU General Public License version 3 or later.
  There is NO WARRANTY, to the extent permitted by law.
  You should have received a copy of the GNU General Public License
  along with this program.  If not, see <https://gnu.org/licenses/gpl.html>.
```

```
$ wc
```

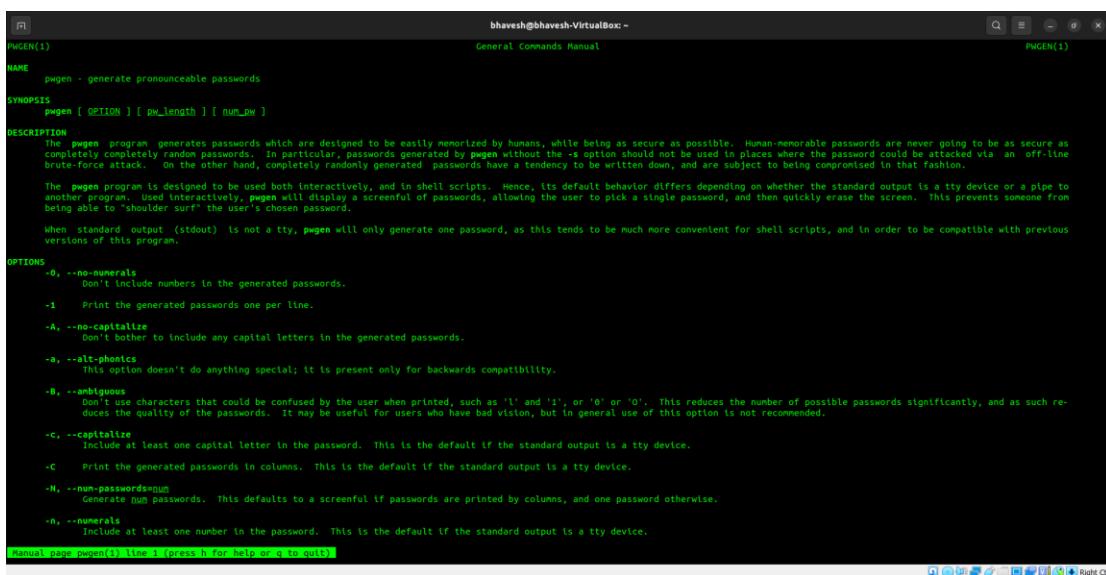


The screenshot shows a terminal window with the command "bhavesh@bhavesh-VirtualBox:~\$ wc file\_grep.txt" entered. The output is:

```
bhavesh@bhavesh-VirtualBox:~$ wc file_grep.txt
6 6 30 file_grep.txt
bhavesh@bhavesh-VirtualBox:~$
```

zz. pwgen : This command is used to generate random passwords. By default, it gives a list of 160 randomly generated passwords.

```
$ man pwgen
```



```
bhavesh@bhavesh-VirtualBox:~$ man pwgen
PwGEN(1)                                         General Commands Manual                                         PwGEN(1)

NAME
    pwgen - generate pronounceable passwords

SYNOPSIS
    pwgen [ OPTION ] [ pw_length ] [ num_pw ]

DESCRIPTION
    The pwgen program generates passwords which are designed to be easily memorized by humans, while being as secure as possible. Human-memorable passwords are never going to be as secure as completely completely random passwords. In particular, passwords generated by pwgen without the -s option should not be used in places where the password could be attacked via an off-line brute-force attack. On the other hand, completely randomly generated passwords have a tendency to be written down, and are subject to being compromised in that fashion.

    The pwgen program is designed to be used both interactively, and in shell scripts. Hence, its default behavior differs depending on whether the standard output is a tty device or a pipe to another program. Used interactively, pwgen will display a screenful of passwords, allowing the user to pick a single password, and then quickly erase the screen. This prevents someone from being able to "shoulder surf" the user's chosen password.

    When standard output (stdout) is not a tty, pwgen will only generate one password, as this tends to be much more convenient for shell scripts, and in order to be compatible with previous versions of this program.

OPTIONS
    -0, --no-numerals
        Don't include numbers in the generated passwords.

    -1      Print the generated passwords one per line.

    -A, --no-capitalization
        Don't bother to include any capital letters in the generated passwords.

    -a, --alt-phonics
        This option doesn't do anything special; it is present only for backwards compatibility.

    -B, --ambiguous
        Don't use characters that could be confused by the user when printed, such as 'l' and '1', or '0' or 'O'. This reduces the number of possible passwords significantly, and as such reduces the quality of the passwords. It may be useful for users who have bad vision, but in general use of this option is not recommended.

    -C, --capitalize
        Include at least one capital letter in the password. This is the default if the standard output is a tty device.

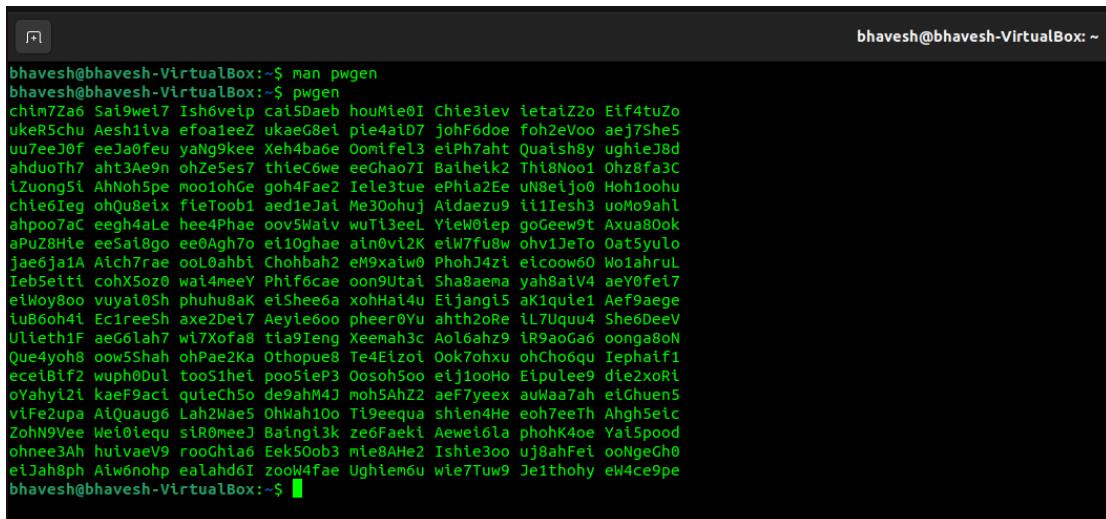
    -c      Print the generated passwords in columns. This is the default if the standard output is a tty device.

    -N, --num-passwords=N
        Generate num passwords. This defaults to a screenful if passwords are printed by columns, and one password otherwise.

    -n, --numerals
        Include at least one number in the password. This is the default if the standard output is a tty device.

Manual page pwgen(1)-line 1 (press h for help or q to quit)
```

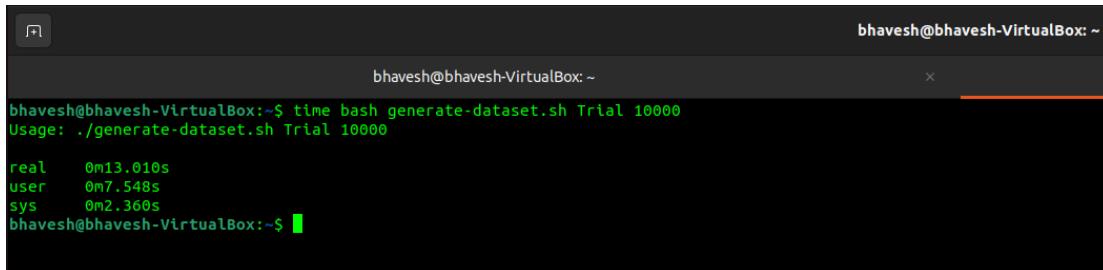
```
$pwgen
```



```
bhavesh@bhavesh-VirtualBox:~$ man pwgen
bhavesh@bhavesh-VirtualBox:~$ pwgen
chim7za6 Sai9wei7 Ish6veip cai5Daeb houMie0I Chie3iev ietaiz2o Eif4tuZo
ukerSchu Aeshiiva efoaiez ukaeG8ei ple4aiD7 johF6doe foh2eVoo aej7She5
uu7teeJ0f eeJaofeu yaNg9kee Xeh4ba6e Oomifel3 elPh7aht Quaish8y ughie3Bd
ahduoTh7 aht3Ae9n ohZe5es7 thieC6we eeGhao7I Baiheik2 Th18Noo1 Ohz8fa3C
izuong5i AhNoh5pe moolohGe goh4Fae2 Iele3tue ePhia2Ee uNBeijo0 Hoh1oohu
chie6Teg ohQubeix fieToobi aediejal Me30ohuj Alidaezu9 lliiTesh3 uoMo9ahl
ahpo07aC eegh4aLe hee4Phae oov5Waiv wuTl3eel YieW0iep goGeew9t Axua80ok
aPuZ8Hte eeSa18go ee0Agh7o eli0guae aln0v12k elW7fu8w ohv1JeTo Oat5yu0
jae6ja1A Aich7rae ooL0ahbi Chohbah2 em9xa1w0 PhohJ4zi eicoow60 WoiahruL
Ieb5Seit cohX5oz0 wai4meeY Phif6cae oon9Utal Sha8ema yah8aiV4 aeY0fei7
eiWoy8oo vuya0Sh phuhu8ak elShee6a xohHai4u Eljangi5 aKiqule1 Aef9aäge
iuB6oh4i Ec1reeSh axe2Dei7 Aeyle6oo pheer0Yu ahtth2oRe il7Uquu4 She6DeeV
Ulith1F aeG6lah7 wl7Xofa8 tla9Ieng Xeemah3c Ao16ahz9 iR9aoGa6 oonga8oN
Que4yoH8 oow55shah ohPae2Ka Othopue8 Te4Elzol Ook7ohxu ohCh6qu Iephalf1
ece1bif2 wuph00ul tooSihei poo5leP3 Oosoh5oo ej1ooHo Eipulee9 die2xoRi
oYahyl2i kaeF9ac1 quieCh5o de9ahM4J moh5AhZ2 aeF7yeex auWaa7ah eighuen5
vlFeZupa AlQuaug6 Lah2Wae5 OhWah10o Tl9eequa shien4He eoh7eeTh AghhSeic
Zohn9Vee Wei0tequ siR0meeJ Baingi3k ze6Faek1 Aewe1ela phohK4oe Yai5poood
ohnee3Ah uitvaaeV9 rooGhia6 Eek5Oob3 mle8AHe2 Ishie3oo uj8ahFei ooNgeGh0
e1Jah8ph AlW6nohp ealahd6I zoorW4fae Ughiem6u wle7Tuw9 Jeithohy ew4ce9pe
bhavesh@bhavesh-VirtualBox:~$
```

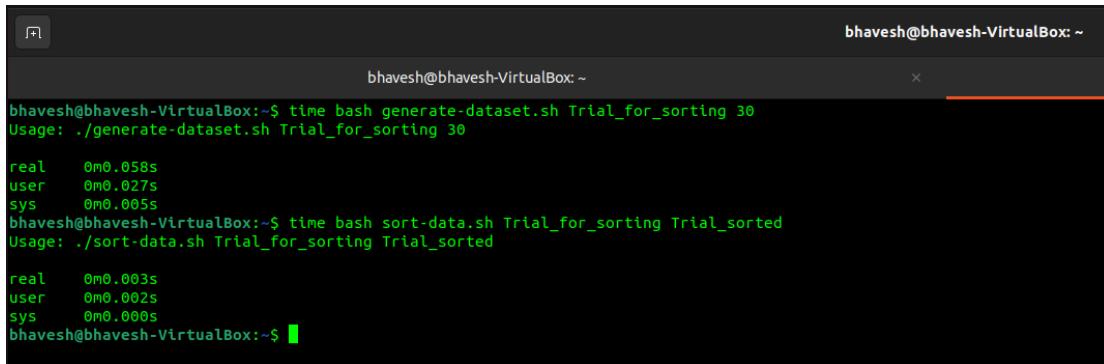
3. (21 points) Write bash scripts to do the following:

a. Write a script called “generate-dataset.sh <filename> <num\_records>” with two command line arguments specifying the file name to output and the number of records, where each record is separated by new line character, and each has the following format: <integer> <integer> <ASCII\_string>. The integer should be any random number up to a 32-bit integer. The ASCII\_string should be any string using ASCII of exactly 100 bytes long. Use the “time” command to show how long the benchmark took to complete. The benchmark should run for at least 10 seconds, and it should complete even if the ssh (or bash) session is terminated. How many records does your file contain after running it? You must write this script entirely with existing Linux commands (which you can install if they don’t exist on your system), without using any other programming languages like C, C++, Java, or Python



A screenshot of a terminal window titled "bhavesh@bhavesh-VirtualBox: ~". The terminal shows the command "time bash generate-dataset.sh Trial 10000" being run. The output includes the usage information "Usage: ./generate-dataset.sh Trial 10000", followed by the time statistics: "real 0m13.010s", "user 0m7.548s", and "sys 0m2.360s". The terminal window has a dark background with light-colored text.

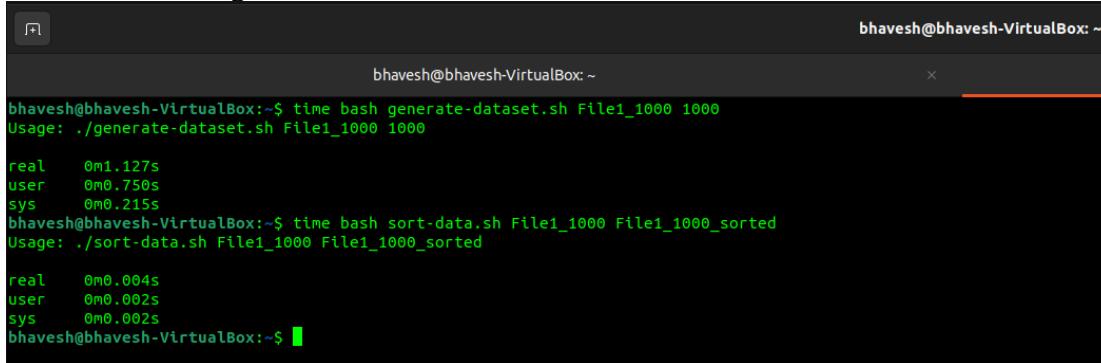
b. Write a script called “sort-data.sh” that takes input a file from part (a) above and sorts the file based on the first column data (make sure to only sort based on the first column data, and not on the entire line of data; also make sure you are treating the data in column 1 as numbers and not text). Use the “time” command to show how long the sort script took to complete.



```
bhavesh@bhavesh-VirtualBox:~$ time bash generate-dataset.sh Trial_for_sorting 30
Usage: ./generate-dataset.sh Trial_for_sorting 30
real    0m0.058s
user    0m0.027s
sys     0m0.005s
bhavesh@bhavesh-VirtualBox:~$ time bash sort-data.sh Trial_for_sorting Trial_sorted
Usage: ./sort-data.sh Trial_for_sorting Trial_sorted
real    0m0.003s
user    0m0.002s
sys     0m0.000s
bhavesh@bhavesh-VirtualBox:~$
```

- c. Use the script in part (a) and generate 3 data files with different number of records (1000, 100000, 1000000); measure time taken to generate these records. Sort the data with your script from part (b) and measure the time. Write a Python matplotlib script to generate a graph for the time taken to generate the data and the time taken to sort the data at the 3 different scales. The graph should automatically adjust to the number of entries, and the scale of the data.

1000 records file generate and sort

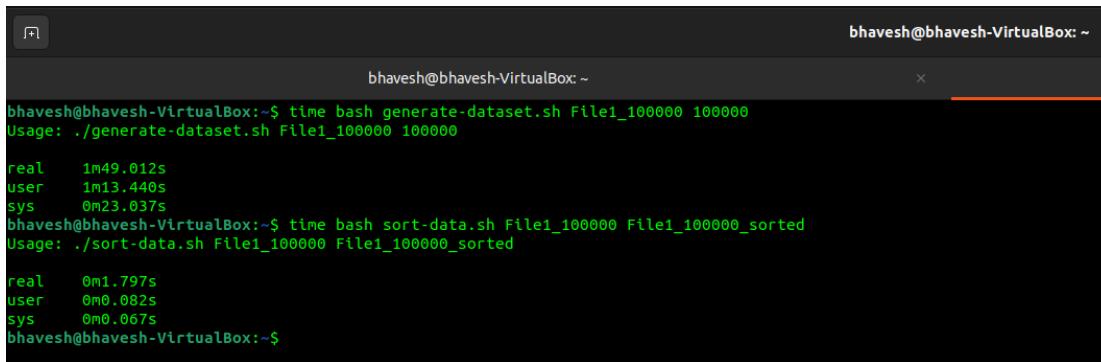


```
bhavesh@bhavesh-VirtualBox:~$ time bash generate-dataset.sh File1_1000 1000
Usage: ./generate-dataset.sh File1_1000 1000

real    0m1.127s
user    0m0.750s
sys     0m0.215s
bhavesh@bhavesh-VirtualBox:~$ time bash sort-data.sh File1_1000 File1_1000_sorted
Usage: ./sort-data.sh File1_1000 File1_1000_sorted

real    0m0.004s
user    0m0.002s
sys     0m0.002s
bhavesh@bhavesh-VirtualBox:~$
```

100000 records file generate and sort

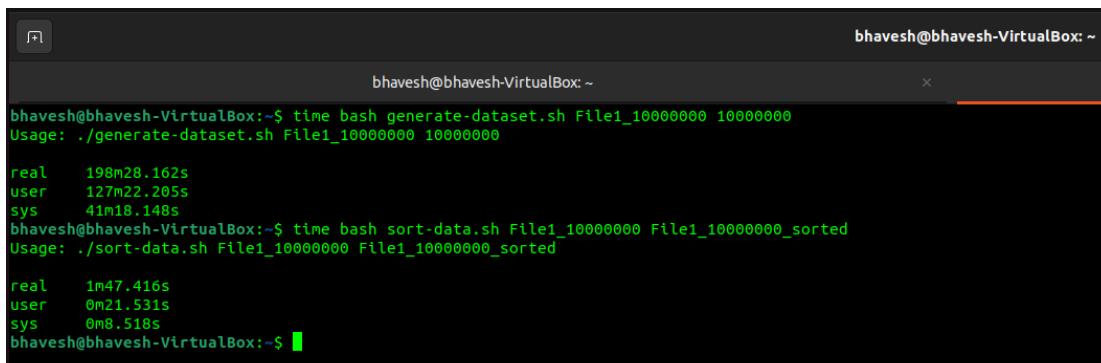


```
bhavesh@bhavesh-VirtualBox:~$ time bash generate-dataset.sh File1_100000 100000
Usage: ./generate-dataset.sh File1_100000 100000

real   1m49.012s
user   1m13.440s
sys    0m23.037s
bhavesh@bhavesh-VirtualBox:~$ time bash sort-data.sh File1_100000 File1_100000_sorted
Usage: ./sort-data.sh File1_100000 File1_100000_sorted

real   0m1.797s
user   0m0.082s
sys    0m0.067s
bhavesh@bhavesh-VirtualBox:~$
```

10000000 records file generate and sort



```
bhavesh@bhavesh-VirtualBox:~$ time bash generate-dataset.sh File1_10000000 10000000
Usage: ./generate-dataset.sh File1_10000000 10000000

real   198m28.162s
user   127m22.205s
sys    41m18.148s
bhavesh@bhavesh-VirtualBox:~$ time bash sort-data.sh File1_10000000 File1_10000000_sorted
Usage: ./sort-data.sh File1_10000000 File1_10000000_sorted

real   1m47.416s
user   0m21.531s
sys    0m8.518s
bhavesh@bhavesh-VirtualBox:~$
```

**4. (15 points) Answer the following questions about VMs:**

- a. In the system configuration of the VM, explain how changing the number of processors changes the behavior of your VM. Explain a scenario where you want to set this to the minimum, and a scenario where you want to set it to the maximum. Why is setting it to the maximum potentially a bad idea?**

Answer: The virtual cores that are assigned to a VM are used in multi-processing within the client OS. If only one core is assigned to the VM, even though rest of the cores will be there for the host machine to use, the VM will suffer as it won't be able to multi-process.

Considering the situation where we want to assign minimum number of cores to the guest, the guest OS will be able to utilize one core. Every thread will be able to utilize one core only. Running a guest OS in this setup which fully loads the CPU will yield a host OS CPU load of 1/nth capacity of the cores.

Of course you can still have CPU-consuming tasks running within your host OS and make use of the remaining n-1 cores.

Considering the other situation where we want to assign all the cores to the guest OS, it will be able to utilize the full capacity of the cores but the host OS will reach 100% memory usage.

Even if we assign all the cores to the guest OS, if the guest OS remains idle, it won't have any effect on the host OS.

**b. In the system configuration of the VM, under the Acceleration Tab, explain the difference between the paravirtualization options: None, Legacy, Minimal, Hyper-V, and KVM. Explain which one would be best to use with Ubuntu Linux, and why.**

Answer: Paravirtualization is a technique by which a guest OS is modified prior to the installation inside a virtual machine. It enhances and enables the use of different operating systems using the same set of hardware, resources such as processors and memory.

Default: The default paravirtualization attempts to pick the best paravirtualization for the virtual machine based on the host operating system.

Legacy: The legacy paravirtualization is used when we have to import our virtual machine from an older version of virtual box.

Minimal: The minimal paravirtualization indicates the presence of a virtual environment. It reports TSC and APIC frequency to the guest OS. It is used for Mac OS 10 guests. It is mandatory to use minimal for that. Even if we use default, it should choose minimal for virtualization to work.

KVM: The KVM paravirtualization allows the Linux KVM as the hypervisor interface and it supports paravirtualized clocks and SMP spinlocks. It is recommended for Linux and Linux based OS.

Hyper-V: The Hyper-V paravirtualization enables the Microsoft Hyper-V as the hypervisor interface which is recognized by Windows OS. It supports paravirtualized clocks, APIC, guest debugging, guest crashing report, and real time checks. It is best suited for Windows virtual machines.

None: It is not recommended to use None paravirtualization. The only scenario where one should use None is when the default or recommended ones are not working properly.

As conclusion, we must use KVM paravirtualization in case of manual selection for Ubuntu and Linux as the virtualization module in the linux kernel recognizes it and allows the linux kernel to be the hypervisor.

**c. In storage devices when configuring the VM, there are multiple types of storage controllers: explain the difference between the IDE, SATA, and NVMe controller. Give an example for each type of storage controller of a scenario where you may want to use this type of controller.**

Answer: The storage we have in VM is used to load the OS in it and also any files that we create will be in that space. It is basically a memory allocation from the host memory.

IDE is an interface standard used for connecting of storage devices such as Hard Disk Drives (HDD), Solid State Drives (SSD) and CD/DVD drives to the computer. In IDE data transfer speed ranges from 100 MB/s to 133 MB/s. It is used in a parallel connection and gives a slower performance and does not support of hot plugging. We want to use the IDE in the situation where we want more memory and adjust with lower speed, we want to consider the IDE for VM.

SATA is a computer bus interface or standard hardware interface used for connecting hard drives, Solid State Drives (SSD) and CD/DVD drives to the computer. In SATA data transfer speed ranges from 150 MB/s for SATA I and 300 MB/s for SATA II. It is used in a serial connection and gives a faster performance with support of hot plugging. SATA cables are better than IDE. We want to use the SATA in the situation where we want less memory and adjust with faster speed, we want to consider the IDE for VM.

NVMe (nonvolatile memory express) is a new storage access and transport protocol for flash and next-generation solid-state drives (SSDs) that delivers the highest throughput and fastest response times yet for all types of enterprise workloads.

To help deliver a high-bandwidth, low-latency user experience, the NVMe protocol accesses flash storage via a PCI Express (PCIe) bus, which supports tens of thousands of parallel command queues and thus is much faster than hard disks and traditional all-flash architectures, which are limited to a single command queue. NVMe memory is considered in the situation where we want to carry prolonged operations i.e. in a non volatile memory and have the fastest access.

**d. In the network configuration of the VM, there are multiple types of network adapters: explain the difference between NAT, Bridged Adapter, Internal Network, and Host-only Network. Give an example for each type of network of a scenario where you may want to use this type of network.**

Answer: In Virtualbox, each VM that we create can have a total of 8 network adapters. Out of 8, 4 of the network adapters can be configured by application interface and the remaining 4 can be configured by command line utility VboxManage. As mentioned in the question, virtual box supports various network modes. The modes are NAT, NAT network, Bridged Adapter, Internal Network, Host-Only network, and Generic driver.

1) NAT (Network Address Translation): This mode is the default selection for all VirtualBox adapters. In NAT mode, virtual box shields the VM behind the host IP. The VM connects to the outside world and access internet by translating its IP to the host IP using it as a proxy. In this mode, VM cannot be accessed from outside network, from the host, from the other VM.

Scenario: As a default configuration, we can use NAT to communicate between host and virtual machine using Port Forwarding.

2) NAT Network: This mode is similar and extended concept of NAT. All the virtual machines on the Virtual box can communicate with each other via a NAT network. We need to configure that first. Each VM will interact with host via NAT only.

Scenario: We can form a NAT network and allow two or more machines to communicate with each other.

3) Bridged Adapter: This mode uses the ‘net-filter’ driver on the host to directly intercept data packets from the physical network port. This creates a new virtualized interface along with the host interface. From outside, it will look like two separate hosts have connected to the network.

Scenario: We can use the bridged adapter mode when we want our VM to directly communicate with the host and also take part in bridging and routing. Bridged adapters are suited for Hadoop setup as they allow VM to communicate with VM using bridged network, access outside internet, communicate with host, and the VM can be accessed via outside network.

4) Internal Network: In this mode, we can create an internal virtual box network of all the VMs where they can talk with each other. All the VMs have to use the internal network as the adapter setting for it to work. In this mode, VMs cannot be accessed and cannot access the outside network. Even the host cannot access the VM.

Scenario: We can use this internal network when we want the VMs to talk with each other but not with the host and not with the outside world.

5) Host-only Network: This mode is an extension of the Internal Network mode with host being part of it. In this way, the VMs can also connect to the host. However, VMs cannot connect to any outside network and cannot access the internet.

Scenario: We can use this host only network when we want the VMs to talk with each other and the host but not with the outside world or avoid access to the internet.

**e. For the USB configuration of the VM, explain the difference between USB 1.1, 2.0, and 3.0 controllers.**

Answer: USB is a specification which allows the host to connect to devices. The USB controller in the VM configuration allows the virtual box a dedicated USB support. Via this configuration the VM can directly access the USB device plugged in the host machine.

USB 1.1 : It is a earliest version of the USB which was widely distributed and put to use. USB 1.1 controller on the virtual box allows the VM to interact with the USB 1.1 supported device plugged into the host machine. The bandwidth allowed by USB 1.1 is 12 Mbits/s. By default, it is the only controller allowed in the virtual machine.

USB 2.0 : It is an upgraded version of the USB 1.1 which substantially increased the bandwidth upto 480 Mbits/s. It introduced a wide range of applications in technology where it was used. In these modern times, USB 2.0 is also used to transfer power and also allows two USB devices to communicate with each other without use of any separate USB host. USB 2.0 controller on the virtual box allows the VM to interact with the USB 2.0 supported device plugged into the host machine. USB 2.0 is also know as hi-speed.

USB 3.0 : It is a more recent version of the USB also known as the SuperSpeed USB. The bandwidth permitted by this is 5Gbits/s. USB 3.0 controller on the virtual box allows the VM to interact with the USB 3.0 supported device plugged into the host machine. USB 3.0 is also known for fast power transfer in rapid charging supported devices.