Bahir Dar University

Software engineering department

Report on Puppy Linux

Name: Dagmawit Amsalu

ID No.1601218

Course: operating system and system programming

Submitted to: Lecturer Wendimu B.

Table of contents

1.Introduction.....3
2.objectives.....4
3.Requierments.....5
4.Installation steps.....6
5.Issues(problem faced).....11
6.why I chose ext4 for my OS installation.....15
7.advanatges and disadvantages of the OS.....16
8.implementing a system call.....18
9.outlook & recommendation 19
10.what is a virtual machine.....20
References.....23

1. Introduction

The concept of virtualization refers to the ability to create virtual versions of computers, resources, or storage systems. Enabling different OS to run together on a single piece of hardware provides flexibility, optimal use of resources, and simplified experiments. Oracle VirtualBox is one of the virtual machine applications among many virtualization tools. It is a powerful solution that allows us to run, test, and manage virtual machines on a large scale from Windows, Linux and macOS host operating systems.

Historical Context of Puppy Linux

Puppy Linux is a family of lightweight Linux distributions designed for speed, efficiency, and ease of use. It can run entirely from RAM, allowing users to remove the boot medium after startup. This design makes it ideal for older or low- resource systems. The system footprint is remarkably small—around 300 MB for 32-bit and 600 MB for 64-bit versions—yet it comes bundled with useful applications like AbiWord, Gnumeric, MPlayer, and lightweight web browsers.

The distribution was originally created by Barry Kauler in 2003. Kauler developed it in response to the growing system requirements of mainstream Linux distributions. He began with guidance from the "Boot Disk HOWTO" and gradually built the system file-by-file. While the first versions were based on Vector Linux, Puppy Linux eventually became a fully independent distribution.

Kauler led the project until his retirement in 2013, after which the community continued development. A key tool in Puppy Linux's evolution is Woof, which

allows developers to build new Puppy versions using the binary packages of other major distributions like Debian or Ubuntu.

With Puppy Linux booted in Oracle VirtualBox, the users can experiment with this light-weight OS without reserving hardware resources or making permanent modifications to their primary OS. The blend of VirtualBox's strong virtualization feature and the light-weight status of Puppy Linux makes this platform suitable for hands-on training, system testing and debugging in a secure, light-footed virtual environment.

What sets Puppy Linux apart is that it is concerned with performance, portability, and simplicity. It is optimized to be run completely in RAM, meaning that when it is booted, it can execute without a hard disk, thereby being extremely fast and efficient. Puppy Linux is typically used by enthusiasts who want to breathe new life into old computers or by users who prefer a highly customizable, fast, and secure OS.

2. Objectives

This document's primary objective is to assist users in installing Puppy Linux on oracle VirtualBox. Following the instructions laid out in this document, users will be able to

- Understand the prerequisites
- Understand the hardware and software requirements for installing and running Puppy Linux in a virtualization system.
- Create a virtual machine and configure it to run Puppy Linux properly.
- Install Puppy Linux in the virtual machine by following step-by-step instructions to ensure the operating system is installed correctly.
- Test and learn about a lightweight operating system in a safe environment
- Test and play with Puppy Linux in a virtual machine without accessing the host operating system and learn more about lightweight operating systems.

Users will be trained in this process and have the ability to run Puppy Linux in oracle VirtualBox providing them the ability to experiment, test, and learn without requiring additional, usable, physical hardware.

3. Requirements

1. Hardware Requirements

The following hardware requirements should be considered to ensure Puppy Linux is able to execute in a virtual machine without error:

- Processor: CPU - A 64-bit Intel or AMD processor with virtualization support (e.g., VT-x/AMD-V) enabled. This will improve the performance of Puppy Linux while running in a virtual machine.
- RAM - At least 1 GB of RAM minimum (recommended: 2 GB or higher!). Puppy Linux is lightweight but having enough RAM allocated will help provide more graceful performance if not running other programs in the VM.
- Storage - A minimum of 5 GB of available disk space that will be used in the new virtual machine running Puppy Linux OS. You may need more storage space if you want to store files or install more programs inside the VM.
- Graphic Display- a graphics display of at least 1024x768 is recommended, which will help our ability to see and interact with a graphical interface to the operating system.
- CPU Virtualization Support - Ensure that the CPU has supported and enabled virtualization technology (VT-x or AMD-V) in BIOS / UEFI.

2. Software Requirements

In addition to VMware Workstation and Puppy Linux there are other software to utilize or download for a successful install.

- Virtual Workstation - This software is the primary software you will use to create and manage the virtual machines. It must be installed on your host operating system (Windows, Linux, or MacOS). This allows you to configure the resources (CPU, Memory and storage) provided to the

virtual machine. For example, oracle VirtualBox can be downloaded from the official VMware website.

- Puppy Linux ISO File - This is the installation media that you will use to install Puppy Linux in the virtual machine. Go to the official Puppy Linux website and from there download the latest stable version of Puppy Linux. The installation media will be an ISO file, which will be the media you will mount to install the OS in the virtual machine.
- Host Operating System: This is the operating system, either Windows, Linux, or macOS, that will have VMware Workstation installed on it. Your host operating system must also meet the minimum system requirements for Oracle VirtualBox to operate effectively.
- Internet Connection: A stable internet connection may be needed to download the Puppy Linux ISO file and depending on your needs, once Puppy Linux is installed, may also require access to install additional software packages or updates.
- Optional Software for Disk Image Control: If you want to manage or resize your virtual disk after installation, there are tools (e.g., VMware vDisk Manager or GParted (available in Puppy Linux) that can modify disk partitions and resize or clone disks.

4. Installation Steps

Step1: install a virtual machine of your choice(in my case I am downloading oracle virtual Box

Here are the simple steps to download **Oracle VirtualBox**:

1. Go to the Official Website:
    a. https://www.virtualbox.org
2. Click on "Downloads" (on the left sidebar or top menu).

3. Choose Your Host Operating System:
   Click the link that matches your system:

(i) Windows hosts
(ii) macOS hosts
(iii) Linux distributions

4. Download the Installer: Your browser will start downloading the .exe (for Windows) or .dmg (for macOS), etc.

How to Install Oracle VirtualBox (Windows Version Example)

1. Run the Installer
   - Double-click the downloaded VirtualBox-x.x.x-xxxxx-Win.exe file.
   - Click "Next" through the setup wizard.

2. Choose Installation Options
   - Leave defaults unless you have specific needs.

- Click Next and then Yes if prompted for network interfaces.

- Click Install.
- If Windows asks for permission to install device software, click Install or Allow.

4. Finish
    - After installation, leave "Start Oracle VM VirtualBox after installation" checked.
    - Click Finish.

Step 2: Download Puppy Linux ISO

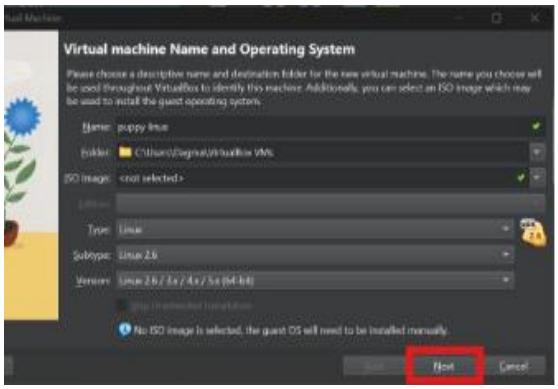
1. Visit the official Puppy Linux website.
2. Download the latest stable ISO file.

I will be downloading BookwormPup64 10.0. BookwormPup64 10.0 was chosen for its stability, compatibility, and performance. Based on Debian Bookworm (Debian 12), it provides long-term support and access to a vast software repository. As a 64-bit system, it ensures optimal performance on modern hardware. Its strong compatibility with VMware Workstation makes it ideal for virtualization, offering a lightweight yet efficient computing environment.

Step 2: open VirtualBox & create a new VM

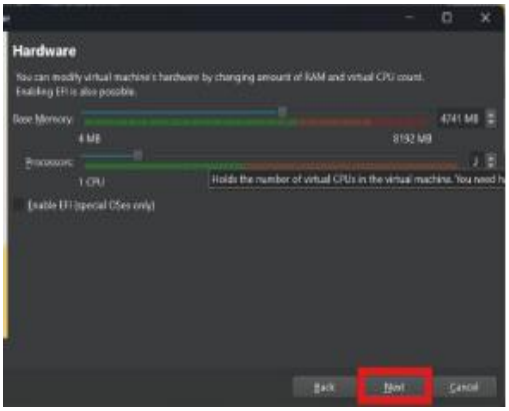
1. Open Oracle VirtualBox
2. Click **"New"**.

3. Set:
    a. Name: Puppy Linux
    b. Type: Linux
    c. Version: Other Linux (64-bit)
4. Click Next.



Step 3: Set Memory Size

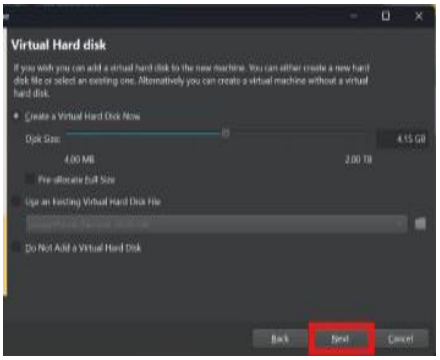
- Allocate at least 512 MB – 1 GB RAM (1024 MB recommended).
- CPU(s): 1 is usually enough for Puppy Linux since it's lightweight.
- If you have a multi-core CPU and want snappier performance, you can use 2 CPUs.
- Click Next.



Step 4: Create a Virtual Hard Disk

1. Choose "Create a virtual hard disk now" → Click Create.

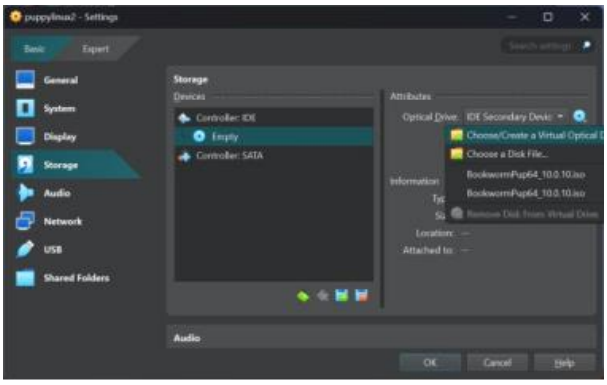
2. Choose VDI (VirtualBox Disk Image) → Click Next.



3. Choose Dynamically allocated → Click Next.
4. Set disk size to 4 GB or more → Click Create.

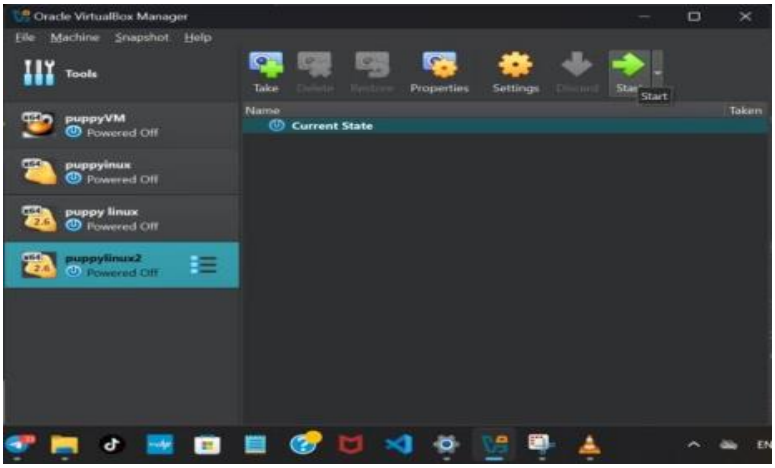
Step 5: Attach the Puppy Linux ISO

1. Select your VM → Click Settings.
2. Go to Storage tab.
3. Under Controller: IDE, click the empty disk icon.
4. On the right side, click the CD icon → Choose a disk file.
5. Select the Puppy Linux ISO you downloaded.



Step 6: Boot & Run Puppy Linux

1. Click Start to boot the VM.
2. Puppy Linux will load into RAM (live mode).
3. You'll see a setup screen — follow the prompts (keyboard, time Zone, resolution).
4. You can run Puppy without installing (I didn't know this at first and I will be telling you the obstacles I encountered)



Issues (Problem Faced)

Initially, I was unaware that Puppy Linux could be run in *Live Mode* without requiring full installation. I assumed that downloading the ISO and booting it would automatically install the operating system to the virtual machine. Because of this misunderstanding, I proceeded to use the terminal and attempted to create and save files, thinking they would be permanently stored.

However, after shutting down the system and booting it again, I noticed that the files I had created were missing. This led to confusion, as I couldn't locate any of my previous work. Upon further investigation, I realized that I had been running Puppy Linux in Live Mode, which loads the OS into RAM and does not retain changes after shutdown unless explicitly configured to do so.Once I understood this, I proceeded to perform a proper installation of Puppy Linux to the virtual hard drive. After installation, I was able to save files and retain changes across sessions, resolving the issue.

How to Install Puppy Linux to the Virtual Hard Drive in VirtualBox Rather than just using the live mode

Step 1: Boot into Live Mode

Start your VirtualBox VM with the Puppy Linux ISO. Once you're in the desktop (livesession):

- Puppy is running entirely from RAM at this point.

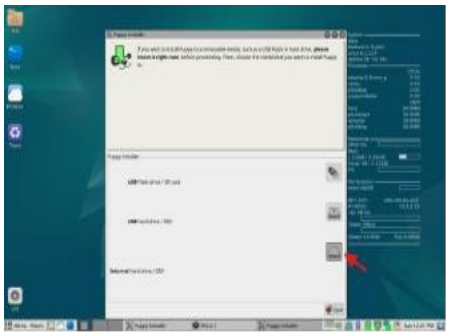
- You'll see an "Install" icon on the desktop.

Step 2: Click the "Install" Icon

1. Double-click Install on the desktop.
2. A window will pop up with several installation options.
3. Choose "Universal Installer."

Step 3: Select Installation Type

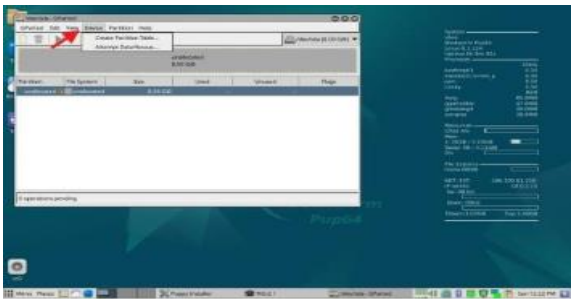
1. In the Universal Installer, choose "internal (IDE or SATA) hard drive".
2. It will list available drives like sda — that's your virtual hard drive.
3. Select sda and click OK.



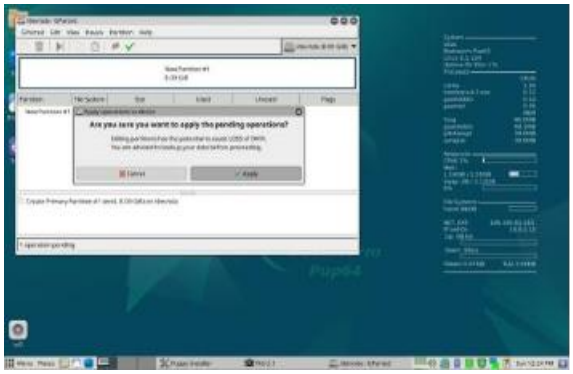
Step 4: Partition the Drive (If needed)

If no partitions exist, you'll be asked to run GParted:

1. In GParted:
    a. Select your virtual drive (usually /dev/sda)
    b. Go to Device → Create Partition Table (choose msdos)
    c. Right-click the unallocated space → New



        i. Set:
            1. Filesystem: ext4
            2. Size: use the full space
            3. Label: optional
        ii. Click Add
    d. Click Apply (green check mark)

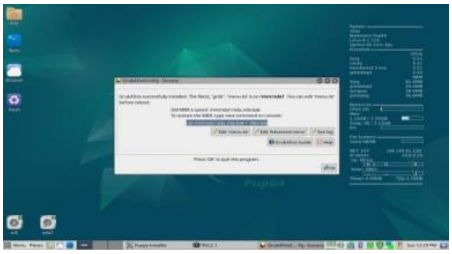


2. Close GParted and go back to the installer.
3. Select the new partition (like /dev/sda1).

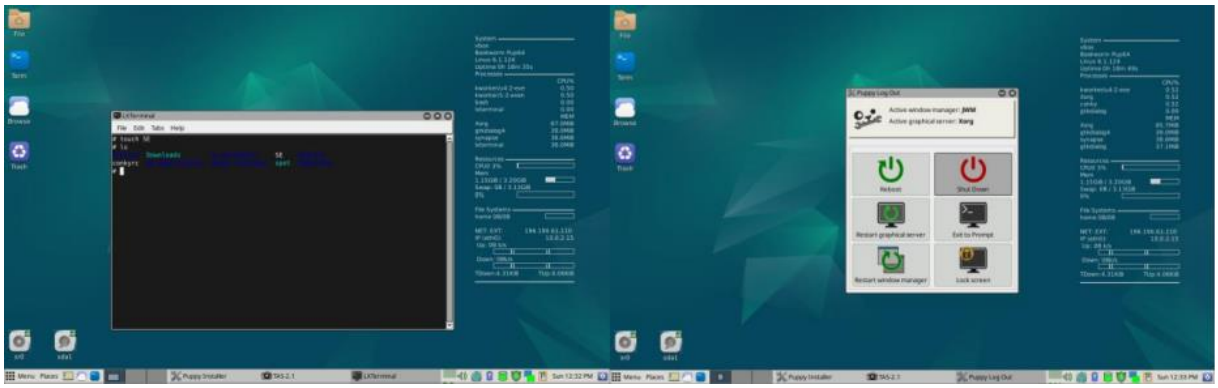
- It will copy necessary files and prompt to install GRUB bootloader.

Step 6: Install GRUB Bootloader

1. Say Yes to install GRUB.
2. Choose GRUB4DOS (it works well with Puppy).
3. Confirm installation to the MBR of the virtual disk.
4. When done, close all windows.



Now that we have installed our OS successfully just to make sure we are still not using the live mode we will be creating a text file and shut down our OS. After we run it again if the text file, we have created it still there we have successfully downloaded our puppy Linux.



We have successfully downloaded our puppy linux.

5. Conclusion

By following this guide, you have successfully installed Puppy Linux in oracle VirtualBox. This setup allows you to explore and test Puppy Linux in a safe, virtualized environment without modifying your primary operating system. Puppy Linux is a great option for lightweight computing and can be used for educational and experimental purposes.

5.Why I Chose ext4 for My Puppy Linux Installation

Selecting an appropriate filesystem is very important when installing an operating system because the filesystem will dictate how data is stored and accessed while also providing protection. In my case, I selected ext4 (Fourth Extended Filesystem) for my installation of Puppy Linux for the above reasons and to achieve an acceptable balance of speed, reliability, and efficiency. This choice of filesystem also works well because Puppy Linux is a 'lightweight' Linux distribution.

Advantages of ext4 for Puppy Linux

1. Journaling for Data Protection: A journaling filesystem logs changes made to the file prior to the to the disk, thus lessening the possibility of corruption due to a system crash or power loss. Puppy Linux, because it is usually run on older, low resources, machines, could be unpredictable to some extent, and therefore, recording data changes via journaling is needed for data safety.
2. Ext4 is a file system that makes storing and opening files faster by using two main features: extents, which keep file data in big, connected pieces instead of small scattered ones, and delayed allocation, which waits to save data until it knows exactly how much to write, helping reduce extra work and improve speed.
3. Backward and forward compatibility: ext4 has the ability to read filesystems such as ext3 or ext2, and thus upgrading from the older systems is easy. also Since ext4 is the default filesystem for many Linux distributions, future kernel updates will continue to support it, ensuring long-term usability and stability.

4. Low CPU and Memory Usage: Since Puppy Linux is designed to be lightweight and run efficiently on older hardware, it is important to choose a filesystem that does not consume excessive resources. ext4 is optimized for low CPU and memory usage, making it an excellent choice for a fast and responsive Puppy Linux installation.
5. Built-in Error Detection and Recovery: ext4 includes an automatic filesystem check (fsck) feature that detects and repairs corrupted files during system boot. This self-healing capability helps maintain system stability and reduces the risk of losing important data.

6.Advantages and disadvantages

Advantages of Puppy Linux

1. Lightweight and Fas: Puppy Linux is one of the lightest Linux distributions that is available. An install of Puppy Linux often occupies less than 300 MB of disk space, and it uses very little RAM. Puppy Linux is a perfect operating system for old computers and devices that have low specifications.
2. Runs Completely in RAM: Puppy Linux doesn't act like a standard operating system. Unlike installing a standard operating system image, Puppy Linux loads into RAM during boot up. Puppy Linux can be extremely fast and responsive as programs open directly in RAM. The elimination of read/write operations is also a significant benefit for storage devices and prolongs the lifespan of storage devices.
3. Portable and Live Mode:Puppy Linux is run directly from a USB, CD, or SD card, making it a great option for users who are looking for a portable OS. Puppy Linux supports Live Mode, which allows users to boot and run the OS without permanent installations or changes to the OS.
4. Simple and Customizable: Puppy Linux has a simple but powerful interface. The user will find all essential applications like web browsers, text editors, and multimedia applications. Puppy Linux is easy to use, even by a novice user. Puppy Linux offers a full range of customization for all users.
5. Low Power Consumption: Since Puppy Linux is designed to be lightweight, it consumes very little power, making it a great choice for old laptops and embedded systems where energy efficiency is crucial.

Disadvantages of Puppy Linux

While Puppy Linux has many advantages, it also has some limitations:

1. Limited Software Support Compared to mainstream Linux distributions such as Ubuntu or Fedora, Puppy has a smaller selection of pre-installed and officially supported applications, as it uses Ubuntu repositories, some software may be buggy, and some applications may work but not as its intended.
2. Not Suitable for Resource Intensive Tasks Puppy Linux is a lightweight distribution and is not built for heavy programs like video editing, gaming system, or enterprise-grade software development.
3. Persistence Requires Setup Since Puppy can be run from Live Mode, the user will have to manually prepare persistence and save on startup across reboots which can make this challenging for newcomers.
4. Security Risks Puppy Linux (by default) runs as root (administrator) on a networked system, which can pose some security risks. While experienced users can set Puppy Linux to run as a regular user, it will require an extra configuration effort.
5. No Automatic Updates In contrast, to Ubuntu or Fedora Puppy Linux does not offer an automatic update mechanism, so users will have to remember to check and install updates manually.
6. Limited Multiuser Support Puppy Linux is single user computing centric, and its multi-user support beyond that is shallow and not enterprise grade.

Conclusion

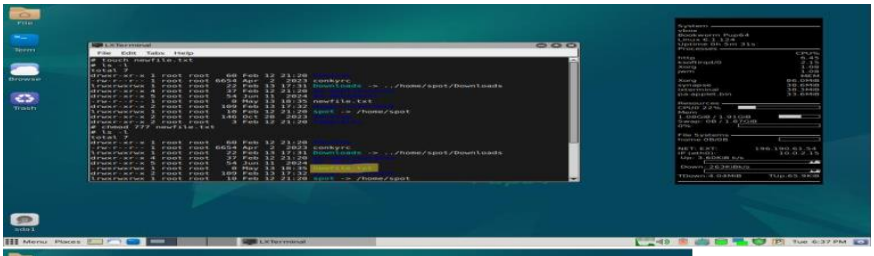
Puppy Linux is an excellent choice for users who need a fast, lightweight, and portable operating system. Its ability to run on minimal hardware, load entirely into RAM, and be used from a USB drive makes it highly versatile. However, it may not be the best option for those needing a fully featured desktop OS for high-performance computing.

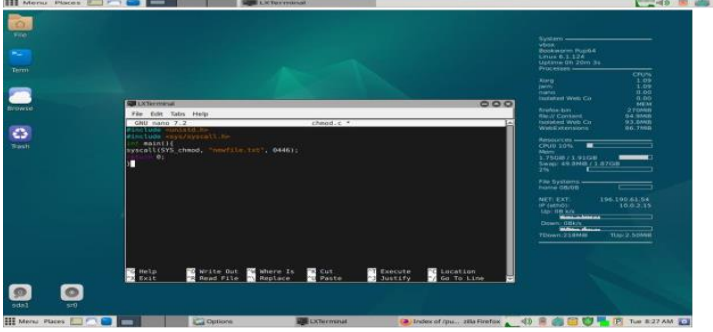
7. implementing a system call

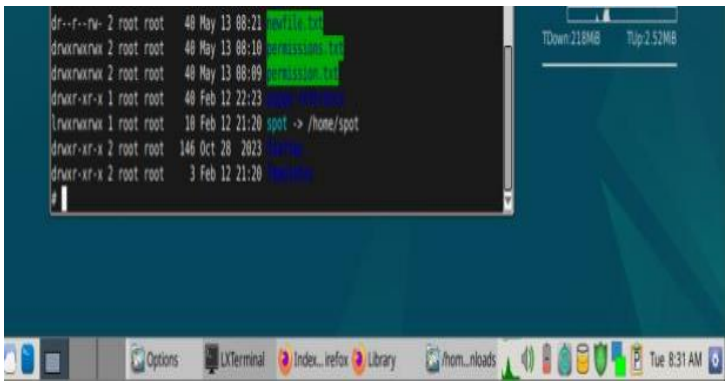
A system call is how a program talks to the operating system.Programs can't do important things like opening files, reading from the keyboard, or using the internet directly. They have to ask the operating system (like Linux) to do it for them. That asking is done using system calls. Common System Calls:

- open() – open a file
- read() – read from a file
- write() – write to a file
- close() – close a file
- fork() – create a new process
- exit() – end a program

The system call i will be demonstrating is chmod() changing the permission of a file.







This C program uses the syscall function to directly invoke the SYS_chmod system call, which changes the permissions of a file. The program attempts to modify the permissions of a file named newfile.txt, setting the permissions to 0446, which means the owner has read and write permissions, the group has read permissions, and others also have read permissions. The system call is part of the low-level interface provided by the operating system, bypassing higher-level C library functions like chmod(). This demonstrates a more manual and low-level way of manipulating file permissions in Unix-like systems.

8.Outlook & Recommendations

Even though I am still learning about operating system from what I have learnt till now these are the updates or recommendations I think puppy Linux should improve especially for beginners like me:

1. Stronger Security: Puppy Linux runs everything as the main "root" user, which isn't very safe. It would be better if users could easily create and switch to a normal user account with fewer permissions, especially when connected to the internet.
2. More App Choices:Right now, finding and installing apps can be tricky. Adding support for popular tools like Flatpak, Snap would make it easier to install modern software with just a few clicks.
3. A Modern Look That's Still Light:Puppy works great on old computers, but the interface feels outdated. A cleaner, more modern design (while still lightweight) would make it more enjoyable and easier to use.
4. Automatic Updates:Right now, you have to update apps and security patches by yourself. A small and simple auto-update feature would keep things up to date without needing to do it manually.
5. Better Support for New Hardware: Some new laptops or Wi-Fi don't work well with Puppy. It'd be great if it could support newer hardware like graphics cards and Wi-Fi so more people can use it on their newer computers.
6. Beginner Guides & Tutorials:New users can feel lost at first. Adding step-by-step guides or interactive tutorials inside Puppy would help people learn how things work in a simple, friendly way.

9.What is a Virtual Machine?

A virtual machine (VM) is software-based representation of a physical computer allowing them to run on a single physical machine. Virtualization allows for hardware resources processors, memory, storage and network connectivity, to be abstracted into logical objects capable of running independently from the physical hardware. A virtual machine monitor (VMM), more commonly referred to as a hypervisor, is the software used to create and maintain virtual machines. A hypervisor allocates system resources to VMs while ensuring there is strict separation between VMs. The concepts of virtualization were formally published by Gerald J. Popek and Robert P. Goldberg in 1974 with their paper Formal Requirements for Virtualizable Third Generation Architectures, which outlined three required properties a VMM must possess

1. Fidelity – Where a VM environment behaves nearly identical to a physical machine ensuring that software appears to run in real hardware.
2. Isolation (Safety) – Where the VMM has complete control over all system resources so that all VMs can run separates of each other.
3. Performance – A VMM needs to be designed so that the VMs operate with a minimal performance detriment to a physical system. There have key advantages of virtual machines over physical machines in regard to efficiency in resource utilization, flexibility, safety, and cost efficiency.

Virtual machines provide significant advantages, such as improved resource utilization, flexibility, security, and cost efficiency. By abstracting hardware, VMs allow organizations to deploy applications more efficiently while reducing reliance on physical infrastructure.

A virtual machine (VM) functions like a physical server, supporting an operating system (OS) and applications. The primary difference is that, unlike a physical server where only one OS and a limited number of applications run, a single physical machine can host multiple VMs, each running its own OS and supporting various applications. In fact, a VM is not a physical entity but a set of files that

defines the virtualized server, making it more flexible and efficient than traditional servers.

The main components that make up a VM include the configuration file and virtual disk files. The configuration file describes the virtual hardware resources assigned to the VM, such as the CPU, memory, storage, and networking. It essentially tells the system which resources to allocate, making the VM resemble a new physical server just waiting for an operating system and applications to be installed. The virtual disk files represent the storage of the VM, similar to a hard drive in a physical server, but they exist in a virtualized form.

Virtualization is widely used in modern operating systems for several important reasons. Here's why it's very popular;

1. Efficient Resource Utilization: Virtualization allows multiple virtual machines (VMs) to run on a single physical machine. Each VM operates as if it's a separate computer, sharing the physical resources (CPU, memory, storage) efficiently. This maximizes hardware utilization and reduces the need for physical servers, leading to better resource allocation
2. Cost Savings: Since multiple VMs can run on a single physical server, organizations save money on hardware, energy, and space. The ability toconsolidate workloads onto fewer machines lowers both capital expenditure and operational costs for businesses.
3. Flexibility and Scalability:Virtualization enables the creation, deletion, and movement of VMs quickly. This provides a high level of flexibility and scalability, allowing systems to scale up or down depending on demand. Businesses can also easily deploy new applications or environments without needing additional physical machines.
4. Isolation and Security:Each VM operates in its own isolated environment, meaning that one VM crashing or being compromised doesn't affect others on the same host. This isolation improves security, as VMs can be used to test potentially harmful software or isolate sensitive applications.
5. Easy Management and Disaster Recovery:VMs can be easily backed up, cloned, and moved between physical hosts. This makes managing systems simpler and supports disaster recovery, as VMs can be quickly restored or migrated in case of failure.

References

Mathur, A., Cao, M., Bhattacharya, S., Dilger, A., Tomas, M., & Vivier, L. (2007). *The new ext4 filesystem: Current status and future plans*. In *Proceedings of the Ottawa Linux Symposium*.

Portnoy, M. (2020). *Virtualization essentials* (2nd ed.). Wiley.

Puppy Linux. (n.d.). *Puppy Linux official website*. Retrieved from https://puppylinux.com

Wikipedia contributors. (n.d.). *Puppy Linux*. Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Puppy_Linux