An Introduction to Linux

Moving from Windows
to the Linux Mint Desktop

Tom Adelstein

An Introduction to Linux

Moving from Windows to the Linux Mint Desktop

Tom Adelstein

An Introduction to Linux: Moving from Windows to the Linux Mint Desktop

Copyright © 2026 Tom Adelstein. All rights reserved.

Published by Paladin Publishers.

ISBN 979-8-9829152-0-7 (PDF)

Contents

About the author
Introduction
How we organized this book

- The parts
- How each chapter works

Part 1. See it, try it

1. A laptop to carry anywhere
2. Trusting a free system
3. Make the USB drive
4. Start the computer from the USB drive
5. Try it for an afternoon

Part 2. Move in

6. Copy the files first
7. Install Linux Mint
8. Find the way around Cinnamon
9. Seven things every computer does
10. Import the files and settings into Linux Mint

Part 3. The daily desktop

11. The browser and email
12. Documents, spreadsheets and slides
13. Printer, Wi-Fi, screens and sound
14. Software Manager and Update Manager
15. A week on the new desktop

Part 4. The terminal

16. Open the terminal
17. Files and folders by command
18. Users, sudo and permissions
19. Install and update software by command

Part 5. A secure computer

20. Check the encryption
21. The firewall on the desktop
22. Daily updates in Update Manager
23. A hardware key for sudo
24. Restore a file from a backup

25. Secure Boot and the TPM chip
26. Linux Mint beside Windows 11

Conclusion
Appendix A. Open Files on the drives
Appendix B. The history behind the desktop

Where the picture came from
How the computer got online
Why Linux came about
How Microsoft built its business
What the courts decided
The Linux desktop today

Index

About the author

Your computer works. Microsoft's managers say to replace it. Don't.

I have been getting people out from under decisions like that since 1998. That year I founded Bynari, worked out how Outlook talks to Exchange, and gave offices a Linux server to use in Exchange's place. IBM's managers made Bynari their first Linux Influencer Partner. I went on to write *Linux System Administration* for O'Reilly, an international best seller in 14 languages and a *Linux Journal* Readers' Choice winner. I have published more than 500 articles and was named one of the world's 20 Linux Luminaries.

Give me one afternoon and a USB drive. I'll show you the way out.

Introduction

People buy computers for various reasons, including writing, keeping their accounts current, running small businesses and staying in touch with family. For any of those tasks, each owner judges the PC by one test: Does it work? If it keeps working, a person has every reason to keep it. My Dell Inspiron 15 5567 passed that test. I kept it as a spare and let a student use it to assist me with a writing project. Microsoft's management left it off the Windows 11 compatible list. In spite of that, it still worked.

In October 2025, Microsoft's management ended support for Windows 10, and on many computers that run it, Settings shows a message that the PC doesn't meet the requirements for the upgraded version, Windows 11.

Every time I started the 5567, that message bothered me. I had used the computer for less than five years, and it worked fine. The message, I suspected, served the seller more than the owner. Then I bought a new laptop with Windows 11, a Dell G16 7630. I also needed a computer with an adequate graphics card to keep using Adobe Photoshop and Camera Raw. Dell's staff took weeks to move my files to the new laptop, and they did a poor job. I sold that 5567, and later I bought another one, unused, for this book's lab.

Afterward, I watched Microsoft's management extend the security updates for Windows 10 to October 2026, and then to October 2027. I didn't like that either. The same managers wrote that the extension gives people "more time to transition to a new Windows 11 PC." Who gave Microsoft's management the right to decide that the owner's working PC needs replacing?

New computers cost more than they did a year ago: in October 2026, the research firm IDC reported higher prices driven by the build-out of AI data centers and expected them to stay well above the year before. Before buying, the owner can ask whether a new one would make life better. Microsoft's answer, a new PC, turns a working computer into a bill at a time when prices are rising; with Linux Mint, which its makers give away, the owner keeps it doing the daily work for years, on a desktop we already know how to use.

My 5567 meets the security that Microsoft's management names for the upgraded version, Windows 11, and fails on the processor's age alone. A new PC comes with conditions too: to finish setting up the Home edition, a buyer needs an Internet connection and a Microsoft account. In chapter 1, we look at the rule and what it costs the owner.

The Linux Mint team sends updates for the current release, Linux Mint 22.3, until April 2029. In chapter 2, we show why its makers give it away. The Linux Mint desktop looks like the one the owner leaves behind, with the menu button at the lower left and the clock in the corner. Before the install, we try it for an afternoon from a USB drive.

The daily work moves with the owner. We copy each document, bookmark and password to a second drive, install Linux Mint and bring them back, and in LibreOffice, we open the Word, Excel and PowerPoint files that coworkers and schools send. We say plainly where Linux Mint falls short: a person whose work depends on Photoshop, which Adobe sells for Windows and macOS only, keeps Windows for that one job, as I did.

By the end of the book, we set up the protections, an encrypted disk, the firewall, updates and a backup, and place Linux Mint beside Windows 11, row by row. The computer on the desk keeps working, and the owner keeps the money.

Tom Adelstein, Paris TX USA

How we organized this book

We begin in chapter 1 with Windows 10 and Linux Mint desktops side by side. In chapter 2, we take up the question most people ask first, what makes Linux free. In chapter 9, after the install, we do seven things every computer does.

The parts

- **Part 1, See it, try it (chapters 1 to 5).** The first look, the trust question, the USB drive, and an afternoon's trial.
- **Part 2, Move in (chapters 6 to 10).** A copy of the files, the install, the way around the Cinnamon desktop, seven things every computer does, and the folders and settings imported into Linux Mint.
- **Part 3, The daily desktop (chapters 11 to 15).** The browser and email, documents, the printer, Wi-Fi, screens and sound, updates and software, and a week of work in Linux Mint.
- **Part 4, The terminal (chapters 16 to 19).** Opening the terminal, files and folders, users and permissions, and software by command.
- **Part 5, A secure computer (chapters 20 to 26).** An encrypted disk we have checked, the firewall, updates we install with Update Manager, a hardware key for administrator tasks, a backup we used to restore a file, and Secure Boot with Microsoft's 2023 certificate. At the end, we compare Linux Mint with Windows 11. The owner decides whether to add the key at sign-in.
- **Appendices.** Appendix A sets Files to open on the drives, the way File Explorer did; Appendix B tells the history behind the desktop.

How each chapter works

We do one job in each chapter. Where the reader did that job in Windows, we start from there and do the same job in Linux Mint. Most steps come with a picture from the lab. Where a fact needs a source, we name it in the paragraph, so a careful reader can check the fact. A short "Try it" list closes each chapter, with the chapter's steps in brief, to run again or to check that each one worked. It makes a good place to put the book down and pick it up the next day.

"In our lab" means we ran the steps on the Dell, first from a USB drive with a live copy of Linux Mint 22.3, then with Linux Mint installed, and in virtual machines, computers simulated inside another. We took the pictures from those runs. Where we didn't run a step in the lab, we say so.

Part 1. See it, try it

1. A laptop to carry anywhere

I own a Dell Inspiron 15 5567. It came with Windows 10, and today it runs Linux Mint. In October 2026, I encrypted its disk, the first step in locking it down for a stay at the coffee shop. By the end of this book, the reader can do the same with their computer.

The updates end

As we mentioned earlier, Microsoft's management ended support for Windows 10 on October 14, 2025. That day, the same managers stopped what they call "technical assistance, feature updates and security updates." Even so, my 5567 kept running, as does any computer still on Windows 10.

A home user can still get the security fixes through a separate program, Extended Security Updates. In Settings, under Windows Update, the words "Enroll now" appear; click them to enroll. An owner who syncs the computer's settings to Microsoft enrolls free; otherwise, the owner redeems 1,000 Microsoft Rewards points or pays $30. As we saw in the introduction, the program now ends on October 12, 2027.

> **Note.** My 5567 could have kept running Windows 10 after support ended. On their page for businesses, Microsoft's managers write, "Windows 10 PCs will continue to work," and the warnings about the end of Windows 10 don't apply the same way to everyone. A company can buy new Windows 11 computers now or pay for continued updates, $61 per device for the first year, a price that doubles each year for up to three years. The same managers include the updates "at no additional cost" for Windows 10 machines running in their cloud services, Windows 365 and Azure. The reader has a choice too: stay on Windows 10 until the date those managers set, or move to Linux Mint, a robust system with a familiar desktop.

On most computers, Microsoft's management wants Windows 11 in place of Windows 10, but they don't support it on my Dell. Their list names Intel processors from the 8th generation on. The Dell carries a 7th-generation Core i5.

The same managers chose those processors by age, not speed. In October 2024, Walmart sold a new Lenovo IdeaPad 1 for under $150. The Celeron N4500 chip came out in 2021, and Microsoft's list includes it. The Dell carries a Core i5 from 2016. By Intel's figures, the i5 runs at a higher speed and works on four tasks at once to the Celeron's two, yet the list leaves it out. As a former founder and chief executive, I would have asked my staff a plain question: did we draw that line by what the processors can do, or by what new computers earn us in Microsoft 365 subscriptions and Copilot?

So I kept the laptop and changed what runs on it.

The people who make Linux Mint give it away, and they say so on their site: "It's both free of cost and open source. It's community-driven." Donors and advertisers pay for the work. Some people grow wary of anything free, and a careful reader should ask. In chapter 2, we look at how that works. Before

installing it, we check that the downloaded copy came from the Linux Mint team.

A Windows 10 owner doesn't need a new computer for it. I installed it on the same Dell.

A familiar desktop

In Figure 1.1, we see Windows 10 above and Linux Mint below. Both screens come from computers running in the lab, not from drawings. Numbers 1 to 6 point to six parts that Microsoft's engineers and the Linux Mint team put in the same places:

1. The menu button at the lower left, where we find the programs. Windows calls it Start.
2. The file manager, where we open folders: File Explorer on Windows, Files on Linux Mint.
3. The web browser: Edge on Windows, Firefox on Linux Mint.
4. The bar along the bottom of the screen, with a button for each open program.
5. The network and sound controls at the lower right.
6. The clock in the corner.

A person who can find these parts on Windows 10 will see them in the same places on Linux Mint.



Figure 1.1. Windows 10 and Linux Mint: the same six parts in the same places.

A Windows user reads mail, looks things up on the web, writes a letter, keeps photos and prints a page. In later chapters, we do each one on Linux Mint, a step at a time.

Who makes Linux Mint

Clement Lefebvre started Linux Mint. A French developer known as Clem, he wrote reviews and tutorials for Linux websites, and, in the team's account, "To host his own articles he started a website called 'Linux Mint'." Over time, he learned what users wanted and began fixing common problems for them. In August 2006, he released Linux Mint 1.0. The team builds Linux Mint on Ubuntu and Debian, two other free Linux systems, and writes that millions of people use it today.

In January 2012, Clem announced Cinnamon, the desktop this book uses. He called the software behind Linux Mint's earlier screens "no longer an option," so he and his team designed a replacement. He wanted "a familiar layout," and he wrote, "Things aren't hidden away but easy to access." His list of Cinnamon's features opened with one panel along the bottom of the screen, a list of open windows and a menu, the parts marked 1 and 4 in Figure 1.1.

Clem leads the work. The Linux Mint team page lists him for the "Overall project and development team," and in October 2026, he wrote, "I've been working on Linux Mint for more than 20 years now."

Where this book takes the laptop

An owner takes more care with a computer carried out of the house than with one that stays on a desk. In Part 5, we lock down a computer like this laptop, a step at a time. From chapter 3 on, we work in our lab, on this Dell and on Linux Mint running in a virtual machine.

Try it

1. Find the model name of the computer.
2. On a Windows 10 computer, open Settings, then Update & Security, and read the message at the top of the Windows Update page. In Figure 1.2, we see the same page in the lab.
3. Look at Figure 1.1 and find the six parts on the computer's screen.

Windows 10: Settings, Update & Security, Windows Update



Figure 1.2. Windows Update on a Windows 10 computer after October 14, 2025.

A reader who wants the history of the desktop, the Internet and Linux, and of Microsoft's business and the antitrust case from 1975 to today, can find it at the back of the book, in Appendix B.

2. Trusting a free system

When someone offers a thing for free, a sensible person looks for the catch. A careful reader should look for it in Linux Mint too. In this chapter, we find the answer, with a source for each part so anyone can check it.

Pervasive and anonymous

The reader used Linux this morning without seeing its name. Google's engineers build the phone's Android on it. Millions of students carry Chromebooks to school, and the Chromium OS developers write that their system "uses the Linux Kernel." At lunch, Clover and Toast card terminals run Android. Toyota's managers wrote in October 2025 that the multimedia system in the 2026 RAV4 "remains on Automotive Grade Linux." In the evening, the TV may run LG's webOS, which LG's developers say "is based on Linux," or Roku OS, which Roku's engineers built "on top of an embedded Linux kernel." Valve tells Steam Deck buyers, "It's a Linux system, you know this!" At bedtime, a Kindle runs it too: Amazon's newest source download for that e-reader includes the Linux code, though Amazon's managers leave the word off the page that offers it.

An open license

The people who built Linux chose to give it away, under an open license. The authors of the GNU General Public License state its purpose in the first lines: "When we speak of free software, we are referring to freedom, not price." Anyone may read the code, change it and share it, as long as they pass the same rights on to the next person.

Linus Torvalds, the student who started Linux in 1991 (Appendix B), began with a stricter rule. In his first copyright, he forbade anyone to distribute Linux for money. With version 0.12, early in Linux's life, he removed that rule and moved Linux to the General Public License. In a 1997 interview, he said, "Making Linux GPL'd was definitely the best thing I ever did."

Who pays for the work

Engineers paid by companies and other organizations write most of the Linux kernel, the core of the system. On the desktop, developers get paid by firms such as Canonical, by donors, and by the German government, through its Sovereign Tech Fund. In 2017, Jonathan Corbet of LWN.net and Greg Kroah-Hartman of the Linux Foundation found that "well over 85 percent of all kernel development is demonstrably done by developers who are being paid for their work." For kernel 7.2, released in August 2026, Corbet counted 249 employers behind the changes, with Intel, Google and Red Hat at the top.

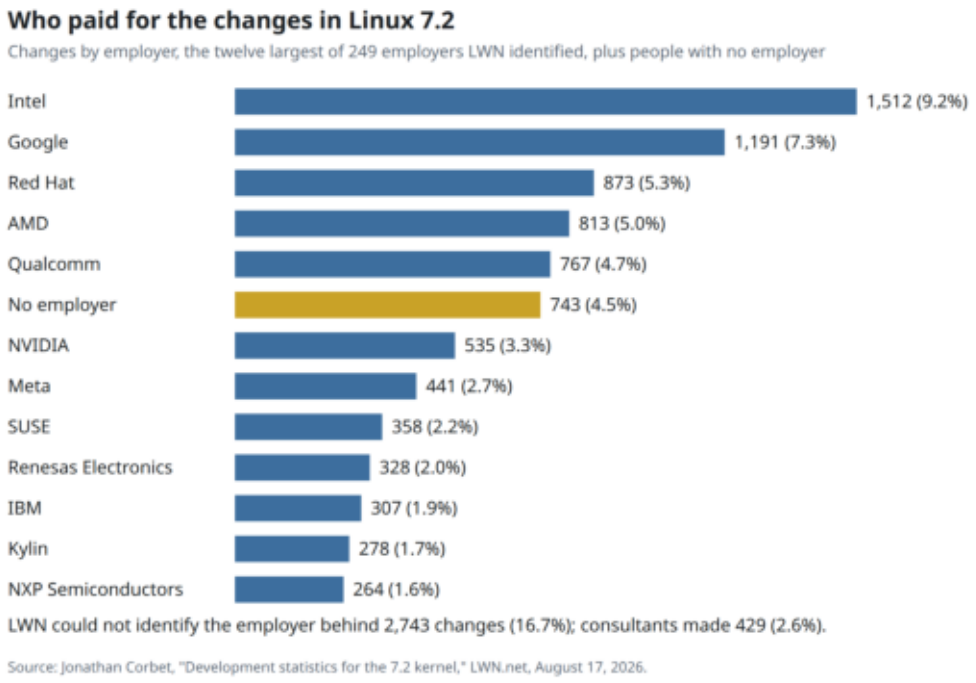


Figure 2.1. Who paid for the changes in Linux 7.2, by LWN's count.

The Linux Mint team pays its way through donations, sponsors, advertising and search partners, and its team publishes the figures. In its news for October 2026, the team reported $13,824 from 443 donors in September.

Microsoft's engineers now build a Linux, called Azure Linux, for the company's cloud. In Microsoft's words, it "is built exclusively for cloud and server workloads, it is not intended to support desktop usage or GUI applications." The desktop we use in this book comes from the Linux Mint team.

Why people give their time

Some people work on Linux without pay, and their reasons deserve respect. In a study published in 2005, researchers from MIT and the Boston Consulting Group asked the contributors. Of those who answered, 44.9 percent found the code "intellectually stimulating to write," 41.3 percent said it improves their skills, 33.1 percent said source code should be open, and 28.6 percent said they owe something back to the community. Some build a career this way. In a 2020 Linux Foundation report, 44 percent of hiring managers said they wanted people with open-source contributions on their record.

What AI tools add

People now use AI tools to find flaws in Linux. Roman Gushchin of Google built a review tool called Sashiko on the company's Gemini model, and the kernel's developers credit it in version 7.2 with reporting the problems behind 252 changes. A security researcher, Sean Heelan, used OpenAI's o3 model to find a flaw in the kernel's file-sharing code, then wrote the fix, and the fix shipped.

In December 2025, Sasha Levin, a kernel developer, wrote the policy: an AI may not sign off on a change, and the person who submits it carries the responsibility. In May 2026, Torvalds wrote that AI-generated reports made the kernel's security mailing list "almost entirely unmanageable." Two months later, as LWN reported, he put it plainly: "AI is a tool, just like other tools we use. And it's clearly a useful one."

In October 2026, Torvalds posted kernel patches, and ZDNET reported that he wrote them largely with Anthropic's Claude Code. They target a slowdown when many processors read the same file. In a synthetic test, he measured reads 23 times faster, though he wrote that "no real load" would see that figure, and in mid-October, the patches waited outside the main kernel. Each flaw found and fixed, with a person answering for the change, gives the reader one more reason to trust Linux.

Personal data

In April 2018, as the team prepared Linux Mint 19, Clem wrote that they would leave out a usage-reporting package that comes with Ubuntu: "This package won't be present in Linux Mint, no data will be collected or sent."

Firefox comes with Linux Mint. The people at Mozilla make it.

On personal data, the people at Mozilla made a different choice from the Linux Mint team. The first time we open Firefox, a bar across the top reads, "Firefox automatically sends some data to Mozilla so that we can improve your experience," and Mozilla's help page describes that data as "usage trends, crash reports, and system details." To choose what Firefox sends, open its menu, then **Settings**, then **Privacy & Security**, and under **Firefox Data Collection and Use**, clear the check mark beside each item to turn it off.

In chapter 3, we check that the copy we download came from Linux Mint, and we learn why the team asks for that step.

Any reader can test every reason in this chapter. Read the license, look up Linux Mint's donation reports, and look at the kernel's employer counts. The people who make Linux publish them in the open.

Try it

1. Read the first lines of the GNU General Public License at gnu.org.
2. Open Linux Mint's blog at blog.linuxmint.com and find the latest monthly news, with its donation figures.
3. Search for LWN's "development statistics" for the newest kernel and look at the list of employers.

3. Make the USB drive

How can a whole system run a computer from a small USB drive, without installing anything? The computer starts from the USB drive, not from Windows. A small program on it loads Linux Mint into memory. The drive holds Linux Mint as one compressed file, and the computer only reads that file.

Whatever we change stays in memory and disappears at shutdown. Windows and the files on the internal drive stay as they were until we choose to install.

People who wanted a first look at Linux kept asking one question: "How can I test Linux without having to change anything on my computer?" In 2000, a programmer named Klaus Knopper answered it with Knoppix, a Linux that ran from a CD, and he later said Knoppix "wasn't the first Live CD." Linux Mint's USB drive works the same way.

What we need

- A computer with a 64-bit processor, 2 GB of memory or more, a disk of 20 GB or more, and a screen of 1024×768 or more. The Linux Mint team gives those figures in its FAQ and recommends 4 GB "for a comfortable usage" and 100 GB of disk. In its user guide, the team writes that "new releases of Linux Mint, including 20, are only available in 64-bit," and 22.3 came after 20.
- A USB drive that holds 8 GB or more. The program we use to write Linux Mint erases everything on it. Find its size on the case, or in File Explorer after plugging it in.
- A Windows computer with an Internet connection.

To check the computer on Windows 10, select Start, then **Settings**, **System** and **About**. Under **Device specifications**, read the RAM installed and the line **System type**, with the kind of Windows first and the processor type after it. Microsoft's support writers note that 64-bit Windows requires "a CPU that's capable of running a 64-bit version of Windows," so 64-bit Windows tells us the processor qualifies. With 32-bit Windows, the processor type at the end of the same line gives the answer. For the disk, go back to **System** and pick **Storage**.

Download Linux Mint

Go to linuxmint.com and click **Download**. Beside **Cinnamon Edition**, click **Download** again. On the next page is a list of mirrors, copies of the file on servers around the world; click one in the same country as the computer. The Downloads folder then has one file of about 3 GB, called an ISO image, with the whole system in it: `linuxmint-22.3-cinnamon-64bit.iso`.

Check the download

The Linux Mint team writes on its download page, "Anyone can produce fake ISO images, it is your responsibility to check you are downloading the official ones." In February 2016, attackers broke into Linux Mint's website and, for one day, pointed the download link for one edition to a copy with a back door, hidden code they could use to get in. The team told its users what happened the next day, promised to show its checks "more prominently going forward," and today it puts a section called Integrity & Authenticity on that page.

To check the file, we compare its SHA256 sum, a long line of letters and numbers that a program computes from a file. If anyone alters the file, even by one character, the sum changes.

1. On Linux Mint's download page, right-click the **sha256sum.txt** button, choose **Save Link As**, and save the file in Downloads.
2. Open `sha256sum.txt` and find the line that ends with `linuxmint-22.3-cinnamon-64bit.iso`. Read Linux Mint's sum from the long line of letters and numbers at its start.
3. In Windows, open the Start menu, type PowerShell, and press Enter.
4. Type this, then press Enter:

```
Get-FileHash $HOME\Downloads\linuxmint-22.3-cinnamon-64bit.iso
```

5. Wait a minute. The sum of the downloaded file appears under **Hash** (Figure 3.1). Compare it with Linux Mint's sum, a few letters at a time. PowerShell writes in capitals and Linux Mint's file in lowercase; ignore that difference. If the two match, the Linux Mint team writes, "your ISO image was successfully downloaded." If they don't, delete the file and download it again. In the same guide, the team describes a second check, of its signature on `sha256sum.txt`, with tools beyond this book.

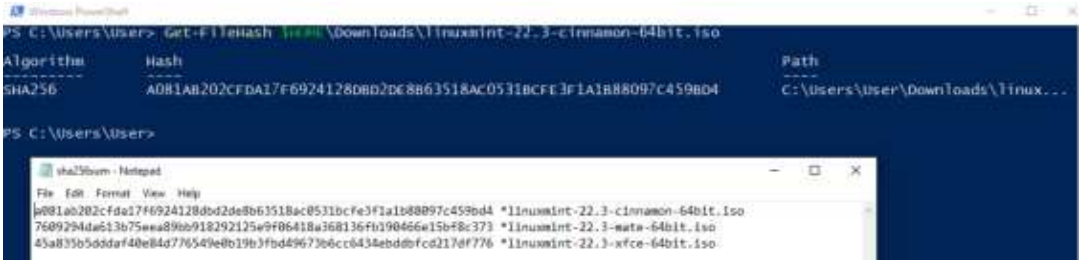


Figure 3.1. PowerShell's sum for the Linux Mint file, above the line for it in Linux Mint's `sha256sum.txt`.

Write the USB drive with Rufus

If we copy the ISO file to the USB drive the plain way, the computer can't start from it, so we use a small program to write it the right way.

1. Download **Rufus** from rufus.ie and open it.
2. Under **Device**, pick the USB drive. Compare its size and name with the one in hand. Rufus erases the drive under **Device**, so make sure no other disk appears there.
3. Click **SELECT** and choose the Linux Mint file.
4. Click **START**. Before Rufus writes the drive, we answer three questions and read one warning. In Rufus 4.15, the version current in October 2026, they come in this order:

- **ISOHybrid image detected.** Leave **Write in ISO Image mode (Recommended)** selected and click **OK** (Figure 3.2).
- **Download required**, about a file named Grub. Click **Yes**. Rufus downloads a small file it needs from rufus.ie, so keep the computer connected to the Internet.
- **Download required**, about Syslinux. Click **Yes** again. If we click **No** here, Rufus stops the copy.
- **WARNING: ALL DATA ON DEVICE ... WILL BE DESTROYED.** Click **OK**.

5. Wait for the green bar to read **READY**, then close Rufus.

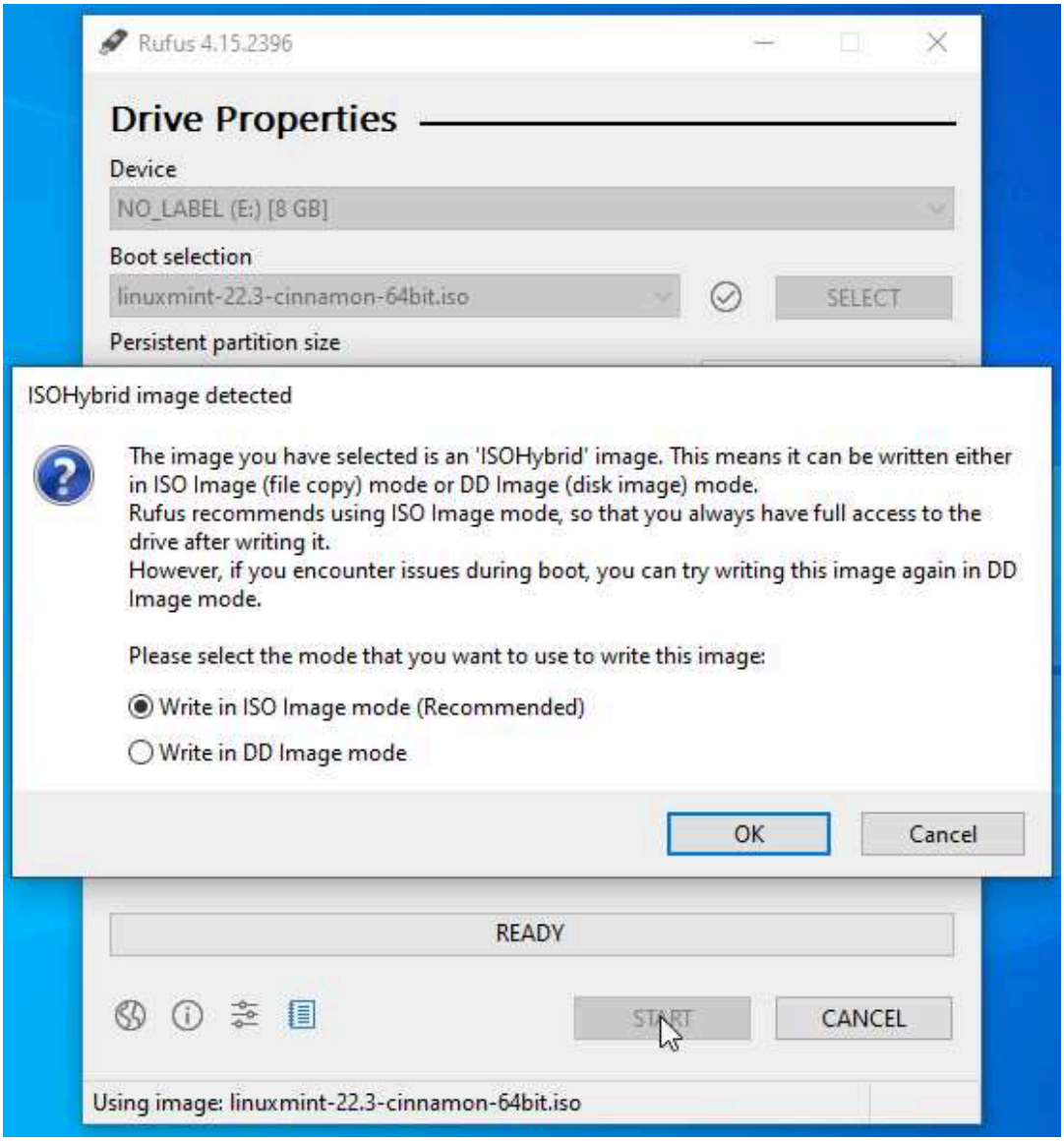


Figure 3.2. The first of Rufus's three questions. Leave ISO Image mode selected and click OK.

If the computer runs Windows 7

Microsoft's management ended support for Windows 7 on January 14, 2020, and two tools in this chapter run only on a newer Windows.

- **The check.** Windows 7's PowerShell lacks Get-FileHash. A tutorial on the Linux Mint forums uses a command in Command Prompt instead. Open the Start menu, type cmd, and press Enter. Then type these two lines, pressing Enter after each:

  ```
  cd Downloads
  CertUtil -hashfile linuxmint-22.3-cinnamon-64bit.iso SHA256
  ```

  In our lab on Windows 7, we ran that command and read the sum in pairs of letters and numbers with spaces between them, such as `a0 81 ab 20`, then "CertUtil: -hashfile command completed successfully." Read the sum pair by pair against the one on Linux Mint's page. The same tutorial adds that on older versions of Windows, CertUtil may be missing, and it recommends the 7-Zip program for the check.
- **The USB drive.** Rufus 4.15 needs Windows 8 or later. Make the USB drive on a computer with Windows 10 or later, then bring it to the Windows 7 computer.

Next, in chapter 4, we start the computer from the USB drive.

Try it

1. When READY appears, open the USB drive in File Explorer and look at the folders Rufus wrote there.
2. Keep `sha256sum.txt` with the ISO file, to check the file again before using it on another computer.

4. Start the computer from the USB drive

Each time we turn on a computer, its firmware, the start-up program the maker built in, picks a drive and starts the system it finds there. Most of the time, that means Windows on the hard disk. In this chapter, we tell it to pick our USB drive instead.

Before we leave Windows

First, check for BitLocker, which Microsoft's engineers build into some editions of Windows to lock the disk. On Windows 10, sign in at account.microsoft.com/devices, find this computer, and write down its recovery key if the page lists one. Then, if BitLocker Drive Encryption appears in Control Panel, open it and click **Suspend protection**. We explain why under the fourth setting below. Windows 10 Home has no BitLocker Drive Encryption, but it may have Device Encryption, which Microsoft's support writers say "enables BitLocker encryption automatically for the Operating System drive"; after a first sign-in

with a Microsoft account, "a recovery key is attached to that account." The recovery key from account.microsoft.com/devices covers it. On Windows 7, only the Ultimate and Enterprise editions have BitLocker.

Start from the USB drive

Each maker chooses its own keys and setup screens. Dell's support writers note that the setup screens differ by model, and other brands use different keys. In this chapter, we use Dell's names, on the lab's Dell Inspiron 5567.

For other brands, we took the keys below from each maker's support pages.

Maker	Choose the USB drive	Setup screen
Dell	F12	F2
HP laptops	Esc repeatedly, then F9	Esc repeatedly, then F10
Lenovo ThinkPad	F12	F1
Lenovo Yoga	F12	F2, or the Novo button
ASUS laptops	Not given on ASUS's page	Hold F2 as you press power

For any other computer, the Linux Mint team writes that the keys "can be Escape, F1, F2, F8, F10, F11, F12, or Delete," and adds, "That information is usually briefly written on the screen during the boot sequence."

Plug the USB drive into the computer and restart it. As soon as the Dell logo appears, tap the F12 key about once per second. In the list, use the arrow keys to pick the USB drive by its brand, sometimes after the word UEFI, and press Enter. At the next menu, with **Start Linux Mint 22.3 Cinnamon 64-bit** at the top (Figure 4.1), press Enter. A minute or two later, the Linux Mint desktop appears. Microsoft's support writers name two kinds of firmware: UEFI, short for Unified Extensible Firmware Interface, and the older BIOS.

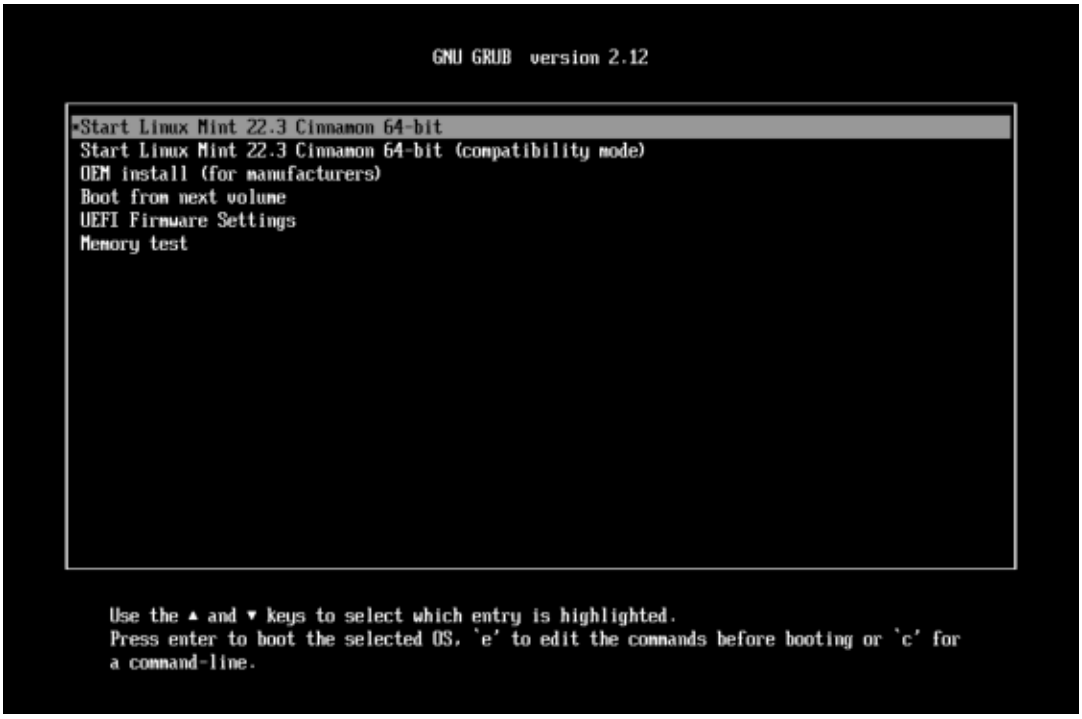


Figure 4.1. The USB drive's menu. Press Enter on the first line.

Dell's setup screen

To open the setup screen, turn on the computer and tap the F2 key about once per second as soon as the Dell logo appears. Find the list of headings on the left. Click the + beside one to open it, then pick a setting to show its choices on the right. To finish, select **Apply** at the bottom right, then **Exit**.

If the USB drive won't start: four settings

We may have to change one of four settings before the computer starts from the USB drive. Each of the four has a different sign on the screen (Figure 4.2). Change the setting in the setup screen, choose **Apply** and **Exit**, and tap F12 again.

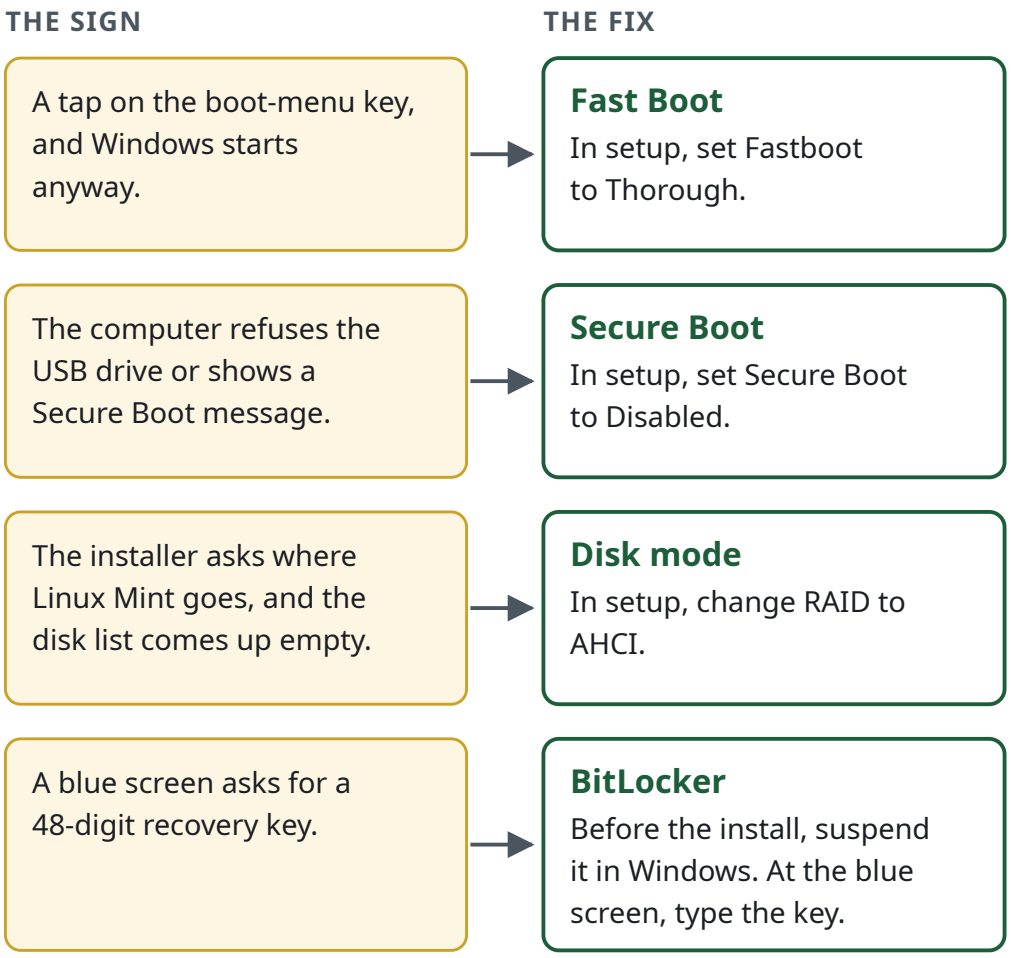


Figure 4.2. The four settings: the sign on the screen, and the fix.

Fast Boot. We tap F12, and Windows starts anyway, because the computer finishes starting before the keystroke reaches it. On Dell's setup screen, this setting appears as **Fastboot**, under **POST Behavior**, with three choices: Minimal, Thorough and Auto. Pick **Thorough**, which turns that shortcut off and runs every start-up check (Figure 4.3).

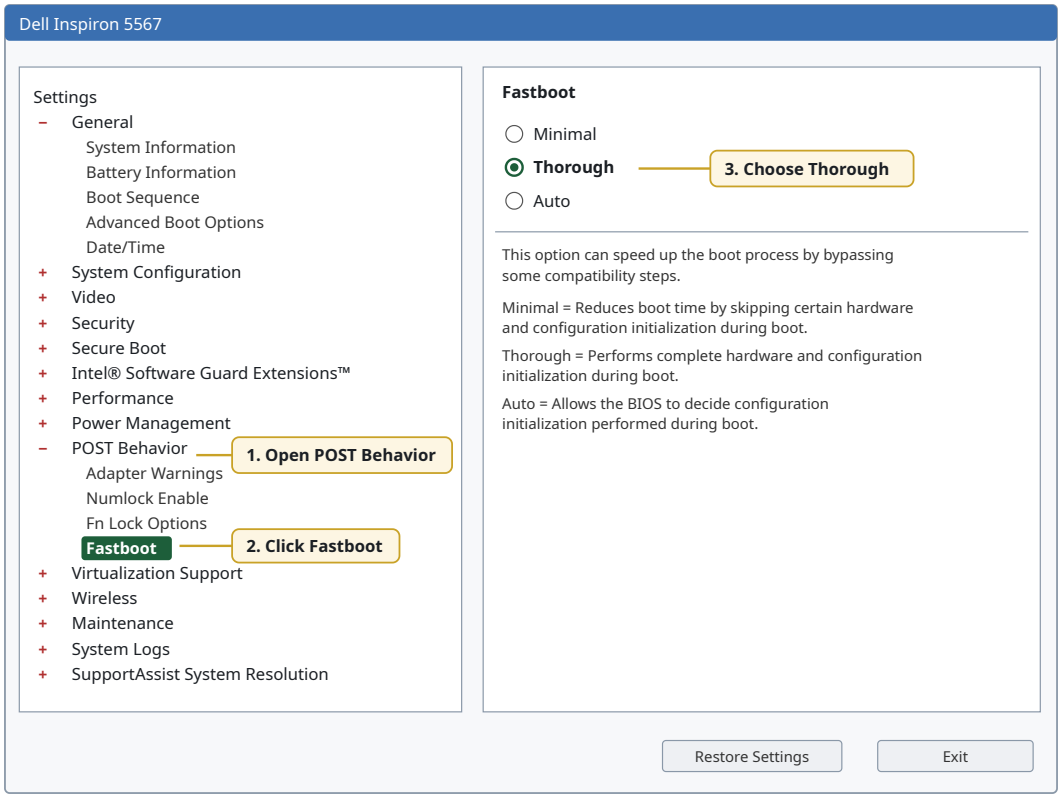


Figure 4.3. Dell setup on the Inspiron 5567: POST Behavior, Fastboot, Thorough.

Secure Boot. The maker stores a list of certificates in the computer's firmware. With Secure Boot on, the computer runs only programs signed by a certificate on that list. Leave it on. We started Linux Mint on the lab's Inspiron 5567 with Secure Boot on, though Microsoft's managers left the 5567 off the Windows 11 list. Microsoft's 2011 certificate for Linux expired in June 2026, and the company's engineers now sign the updates to Linux Mint's start-up program with its 2023 replacement. A computer that holds only the 2011 certificate can stop starting after such an update, so in chapter 25, we add the new one.

If the USB drive won't start, or a message about a Secure Boot violation appears, the Linux Mint team prints the fix in its installation guide: "Disable SecureBoot in the BIOS settings of your computer." On the Dell, open the **Secure Boot** heading, open **Secure Boot Enable**, and select **Disabled** (Figure 4.4).

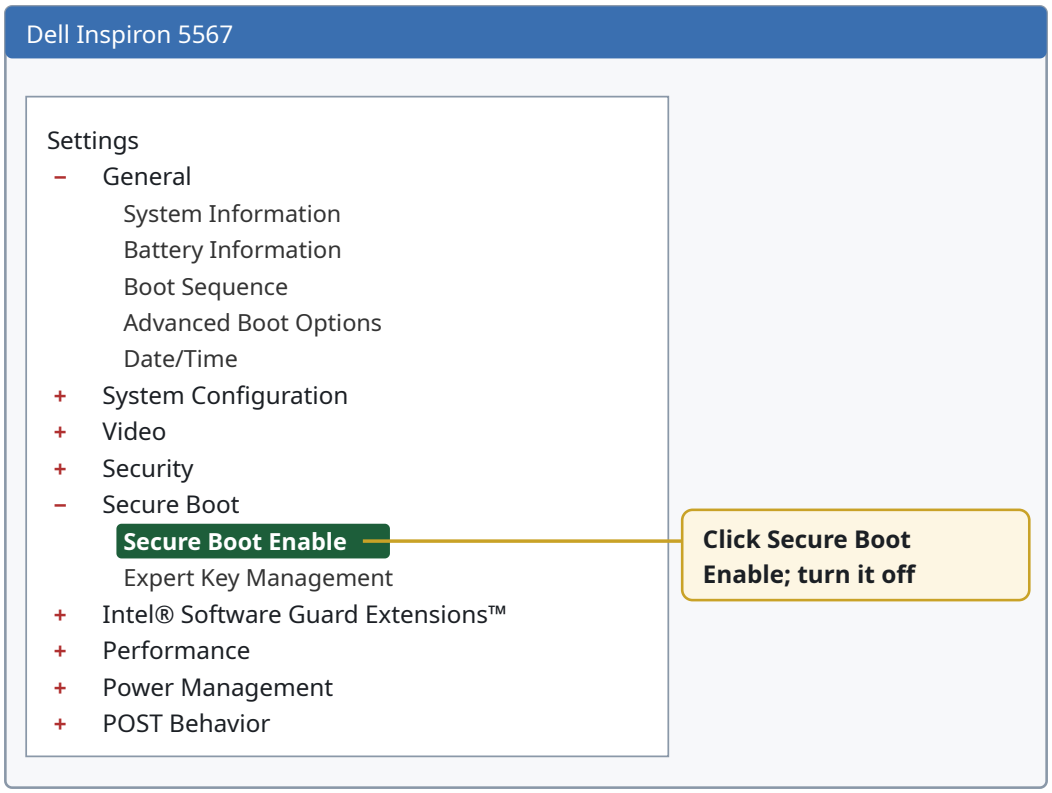


Figure 4.4. Dell setup on the Inspiron 5567: Secure Boot Enable.

Disk mode. If no disk appears on the chapter 7 installer screen where we choose where Linux Mint goes, come back to this setting. To follow Dell's fix, set the disk mode to AHCI: open **System Configuration**, then **SATA Operation**, and select **AHCI**. Dell's guide gives the reason: "Linux does not natively support SATA Operation in RAID On mode as of May 2022." Change this setting only when no disk appears in the installer, and leave it alone during the trial in chapter 5. Dell's support staff write that the change, with Windows on the disk, "could cause windows to present an INACCESSIBLE BOOT DEVICE (0X7B) error while booting," and that changing the setting back "can resolve the blue screen error."

BitLocker. Windows may lock its disk with BitLocker, and after a change in setup, it can ask for a 48-digit recovery key on a blue screen. If we suspend that lock beforehand, Windows skips the prompt. If the blue screen comes anyway, sign in at account.microsoft.com/devices from a second computer or a phone, find this one, and type its recovery key.

If the logo flashes past

If the Dell logo flashes past before the key takes, open the setup screen from Windows 10 instead: go to **Settings**, **Update & Security**, **Recovery**, and under **Advanced startup** click **Restart now**. Then choose **Troubleshoot**, **Advanced options**, **UEFI Firmware Settings** and **Restart**. Microsoft's support page "Enable virtualization on Windows" gives this route for Windows 10.

Try it

1. Write down the computer's BitLocker recovery key from account.microsoft.com/devices, and suspend BitLocker if it appears in Control Panel.
2. Try the USB drive with Secure Boot on. If the software does not start, turn Secure Boot off.
3. Plug in the USB drive, restart, and tap F12 about once per second at the Dell logo.
4. Pick the USB drive and press Enter on **Start Linux Mint 22.3 Cinnamon 64-bit**.
5. If Windows starts anyway, set Fastboot to Thorough in setup and try again.

5. Try it for an afternoon

We run Linux Mint from the USB drive, and the computer's disk stays as it is. The Linux Mint team calls this the live session, and it writes that the changes we make in it "are not written to the USB stick (or DVD) and they do not impact the system installed by the installer." By "USB stick," the team means the USB drive from chapter 3. Windows and our files stay untouched, so we try Linux Mint first and decide later.

The team also writes that "the Live session is slower," because we run Linux Mint from the USB drive instead of the computer's disk. An installed Linux Mint runs faster than the live session.

Connect to the network

Click the network icon at the lower right of the panel to see a list of Wi-Fi networks. Pick one by name and type its password. Or plug a network cable into the computer, and skip the password.

Play a sound

Click the speaker icon at the lower right, next to the network icon, and set the volume. Then open a video in the web browser and listen.

Open a web page and webmail

Open Firefox from the orange button on the panel. The first time, a tab with its privacy notice opens too; close it. Go to a familiar web page, then go to a webmail account and sign in (Figure 5.1).

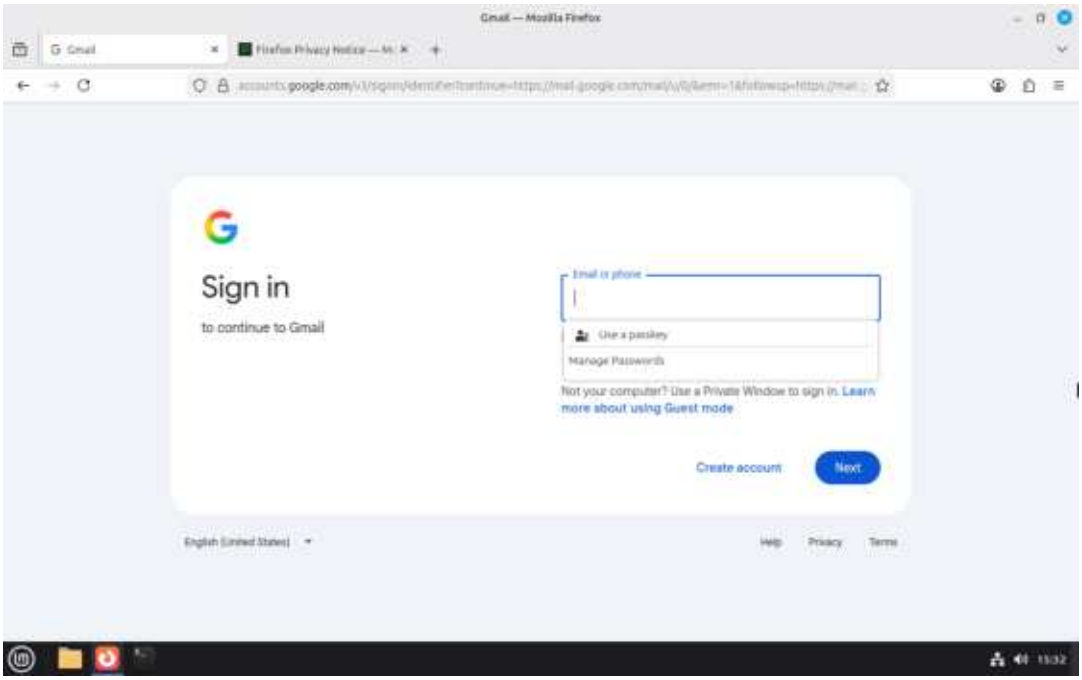


Figure 5.1. Gmail's sign-in page in Firefox, in the live session.

Open a photo or a document

If another USB drive holds a photo or a document, plug it in. Open Files from the yellow folder on the panel, click its name on the left, and double-click the file. Read it, and leave the contents as they are. Without a second drive, skip this step.

Leave the installer alone

One icon on the desktop, at the top left, reads **Install Linux Mint**. We use it in chapter 7. Leave it alone today.

Go back to Windows

To finish, open the menu, click the red power button, and choose **Shut Down**. The screen turns black, and the Linux Mint logo appears with one line: "Please remove the installation medium, then press ENTER:" (Figure 5.2). Pull the Linux Mint USB drive out and press Enter. When the computer turns off, start it again. Windows starts unchanged, with every file where we left it.

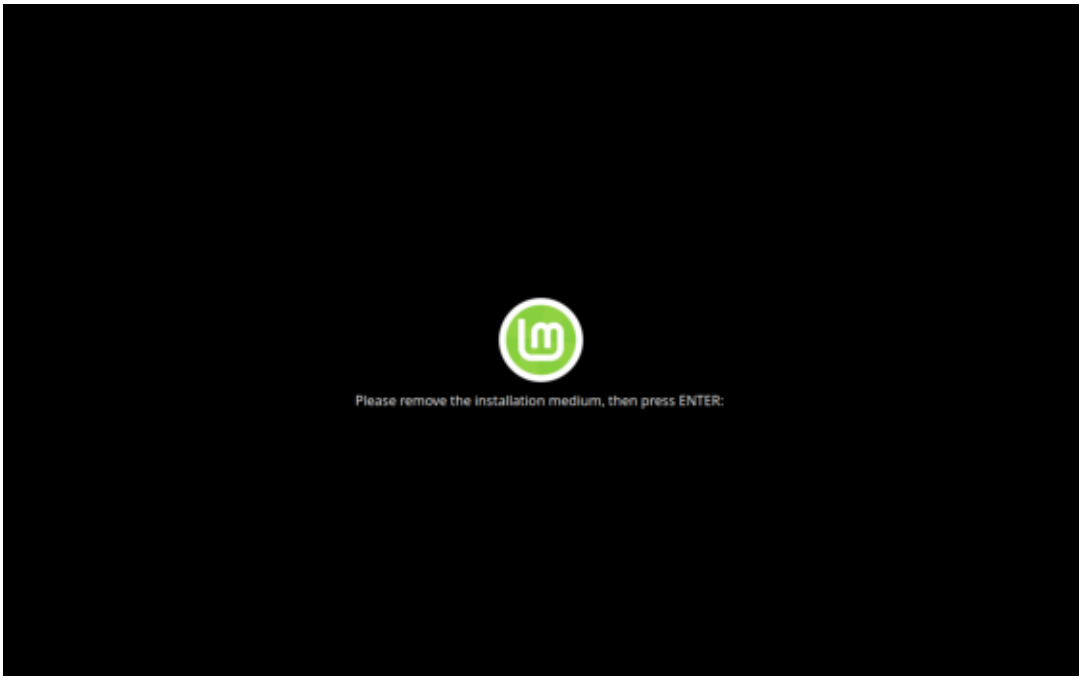


Figure 5.2. Shutting down from the USB drive.

A lighter edition: Xfce

Olivier Fourdan started Xfce in late 1996 as a separate project, and the Linux Mint team packages it as one edition of Linux Mint. The people who build Xfce today, the Xfce Development Team, write that it "aims to be fast and low on system resources, while still being visually appealing and user friendly." They make it from separate parts: a window manager arranges what is open on the screen, a panel switches among the windows and starts programs, and a file manager, a settings manager and an application finder do what their names say. A session manager handles signing in and power.

The team released Xfce 4.20 after what it calls "almost two years of work." If we want a desktop "lighter on resource usage," in the Linux Mint team's words, we can download the Xfce edition of Linux Mint and try it from a USB drive, the same way we tried Cinnamon in this chapter.

Decide

If the network, the sound, the web and the files all worked, the computer can run Linux Mint, and in chapter 6, we get the files ready to move. If something failed, write down what it was, and read the next section before going on; we set up the printer, Wi-Fi, screens and sound in chapter 13.

Before chapter 6

In chapter 7, the installer erases the disk with Windows on it, and this book has no steps to put it back. We take one path, Linux Mint alone on the disk, and leave out keeping Windows beside it. Before we copy our files in chapter 6, check three things.

The network. If Wi-Fi failed in the live session, plug in a network cable, start from the USB drive again, and check the connection. If the cable works, go on. After the install, Driver Manager in chapter 13 can add a Wi-Fi driver, and the Linux Mint team writes in its installation guide that it works offline too: "If you are offline, the Driver Manager will inform you that it cannot connect to the Internet," followed by "Insert your bootable Linux Mint USB stick (or DVD), wait for it to be mounted, and click OK." So keep the USB drive after the install. In our lab, we didn't test that route. If the cable fails as well, stop here and stay on Windows.

Programs. Write down each program we use every week, and look for a Linux version of each. Photoshop, AutoCAD and the Teams program, for example, have none (chapter 15), though Teams works in Chrome. If our work depends on one with no Linux version, we keep a Windows computer for that one job, or stay on Windows.

Outlook mail. Thunderbird can't read an Outlook data file, the .pst file (chapter 11), and the Outlook program goes with Windows. If we have mail only in a .pst file, move it into an account whose mail the service keeps on its server, before chapter 7. Microsoft's support writers list that among the reasons to import one: "You want to move Outlook items between local folders and an IMAP or Exchange server." An IMAP or Exchange account is the kind whose mail the service keeps on its server. Their steps, in classic Outlook:

1. In the list on the left, click a folder of that account, such as Outlook.com or a Microsoft 365 mailbox.
2. Choose **File**, then **Open & Export**, then **Import/Export**.
3. Select **Import from another program or file**, then **Next**.
4. Select **Outlook Data File (.pst)**, then **Next**.
5. Browse to the .pst file. Under **Options**, choose how to handle the mail and contacts, and click **Next**. If the file has a password, type it and click **OK**.
6. Choose that account's mailbox, or **Import items into the current folder**, then **Finish**.

Microsoft's writers add, "When the progress box disappears, the import is finished." In our lab, we didn't run this import. After the install, we read that mail from the server, in Thunderbird (chapter 11) or in the browser. For Outlook.com, Microsoft's support writers say it works "in either Firefox or Chrome on Linux" (chapter 15). Without an account of that kind, the mail in a .pst file opens only in Outlook; keep a Windows computer for it, or stay on Windows.

Try it

1. Connect to the network and open a web page.
2. Play a video and listen.
3. With a second USB drive, open a photo from it.
4. Shut down, pull out the Linux Mint USB drive, and check that Windows starts.

Part 2. Move in

6. Copy the files first

The installer in chapter 7 erases the disk. That disk holds years of our photos, letters and accounts, the reason we kept this computer, and without a copy, they go with Windows. So first, we copy our files to a second drive, a USB drive or an external disk big enough to hold them, and then we open a few to prove them. We set that drive aside until chapter 10, when we import the files into Linux Mint.

Copy the folders

Plug the second drive into the computer. On Windows 10, open File Explorer. On Windows 7, open Windows Explorer. Go to each folder in daily use: Documents, Pictures, Desktop, Downloads, Music and Videos. On Windows 7, four of them have My in front: My Documents, My Music, My Pictures and My Videos. In File Explorer, click **This PC** on the left to see them all, with the second drive listed below them (Figure 6.1). To check that the drive is big enough, right-click each folder and choose **Properties** to see its size, then compare the total with the free space shown under the drive in **This PC**.

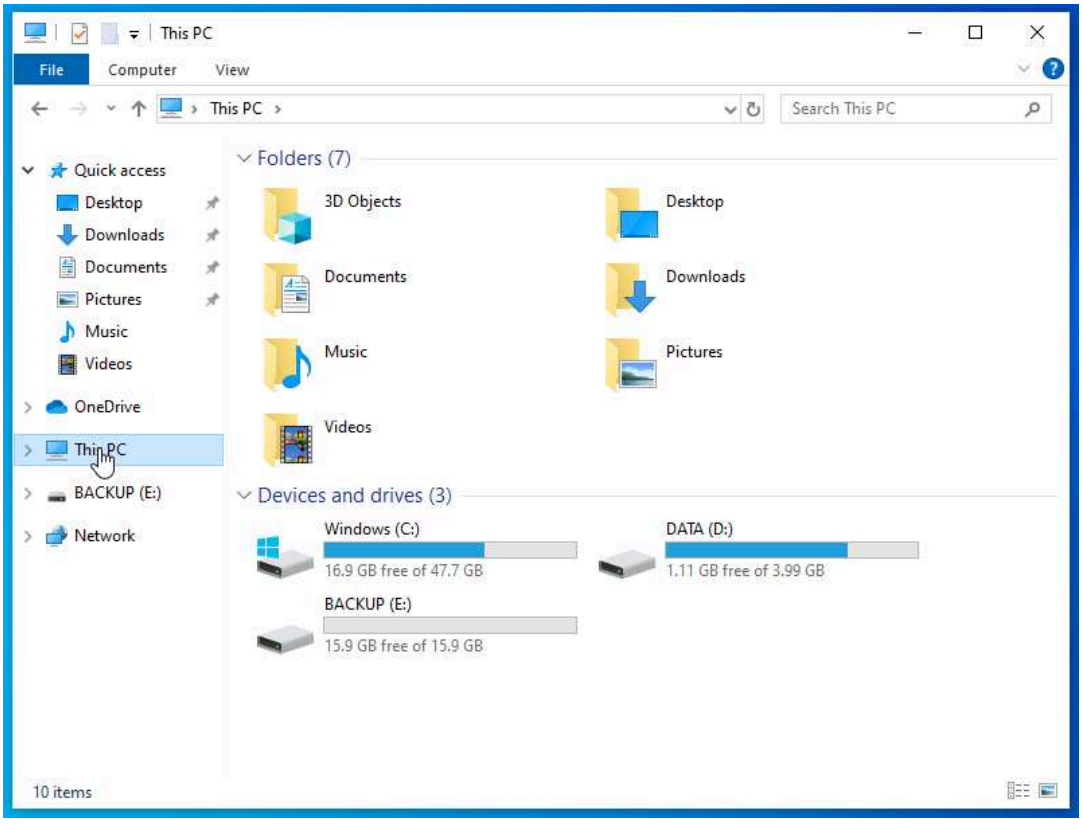


Figure 6.1. This PC in File Explorer, with the personal folders and the second drive, here named BACKUP.

Copy each folder to the second drive. Select one, right-click it, and choose **Copy** (Figure 6.2). Then open the second drive, right-click inside it, and pick **Paste** (Figure 6.3). Wait for the copy to finish before moving to the next folder. In their table of file systems, Microsoft's engineers list 4 GB as the largest file for FAT32, a format common on USB drives, and no such limit for exFAT or NTFS. A long video or an Outlook data file can pass 4 GB. If Windows says a file is too large for the drive, copy that file to an external disk formatted with exFAT or NTFS.

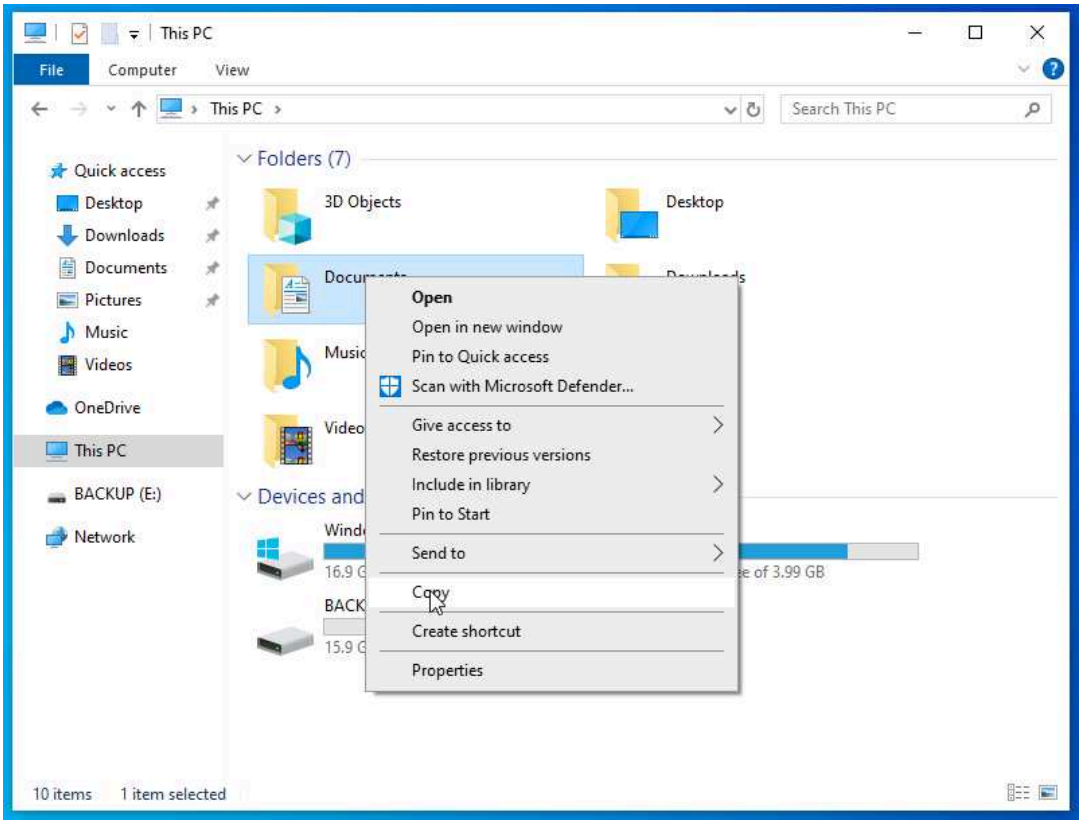


Figure 6.2. Right-click the Documents folder and choose Copy.

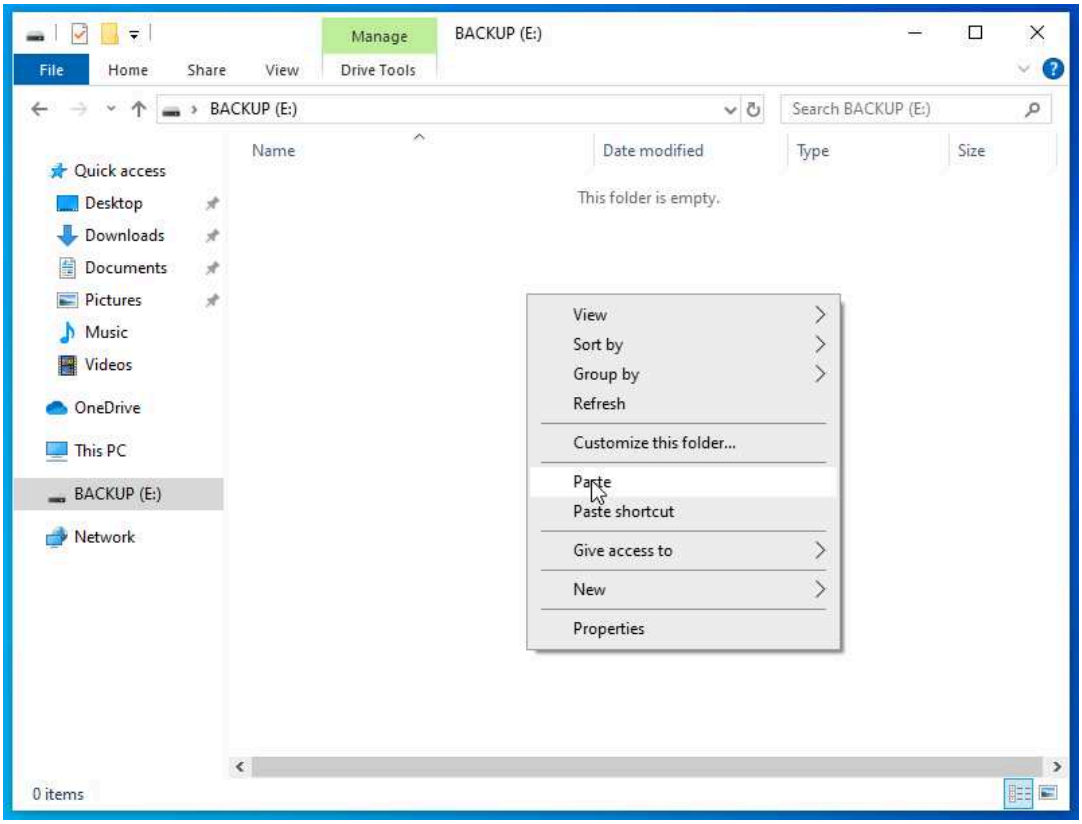


Figure 6.3. Right-click inside the second drive and choose Paste.

Now, prove the copies. Open the second drive and open a few files from it: a document, a photo, a song (Figure 6.4). If they open on the second drive, the copy worked. If a file does not open, copy that folder again.

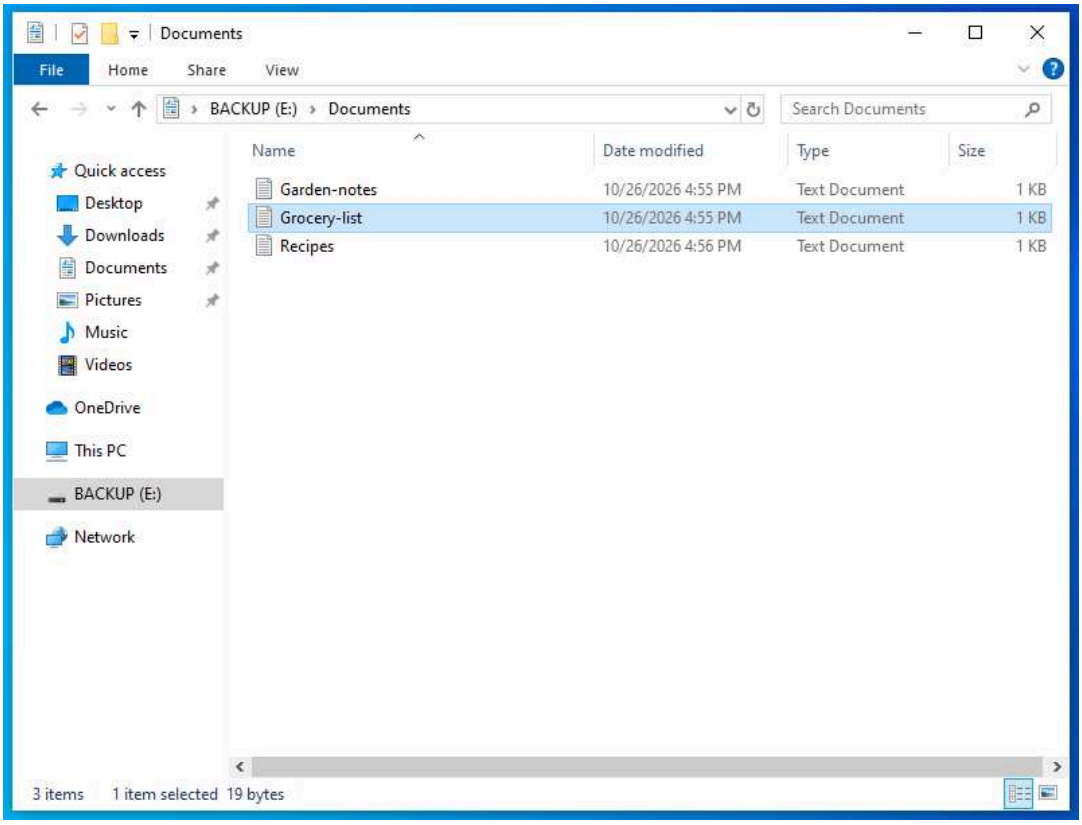


Figure 6.4. The Documents folder on the second drive, with the copied files.

Bookmarks

Bring the Chrome bookmarks along. These steps match the version current in October 2026; an older release, such as the last one for Windows 7, may name the menus differently. For Edge, see the steps after the saved passwords; for another browser, search its help for "export bookmarks." Open Chrome. At the top right, select **More**, the three dots, then **Bookmarks and lists**, then **Open bookmarks manager** (Figure 6.5). In the manager, select **More** again at its top right, then **Export bookmarks** (Figure 6.6). Chrome saves them as one HTML file. To pick a location, click the second drive on the left, then **Save** (Figure 6.7).

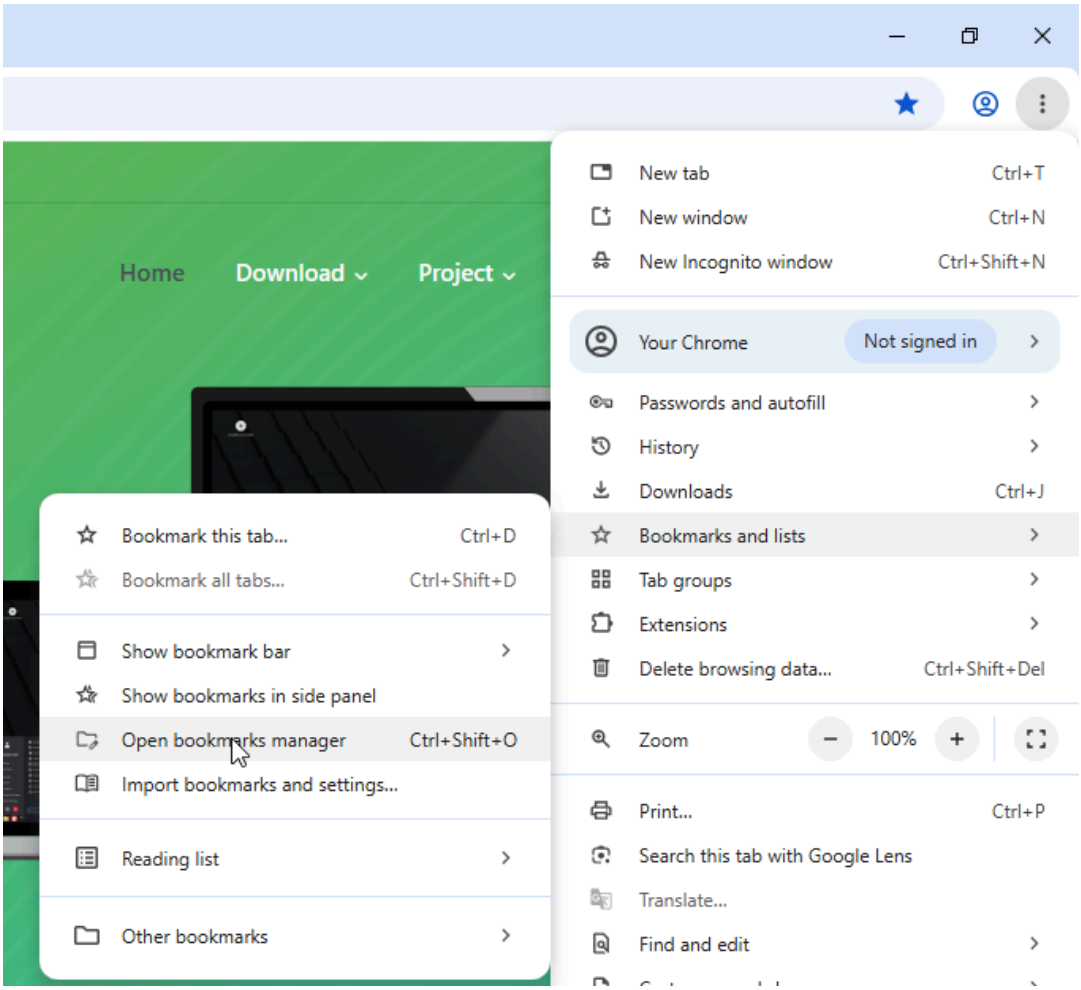


Figure 6.5. Chrome's More menu: Bookmarks and lists, then Open bookmarks manager.



Figure 6.6. In the bookmarks manager, More, then Export bookmarks.

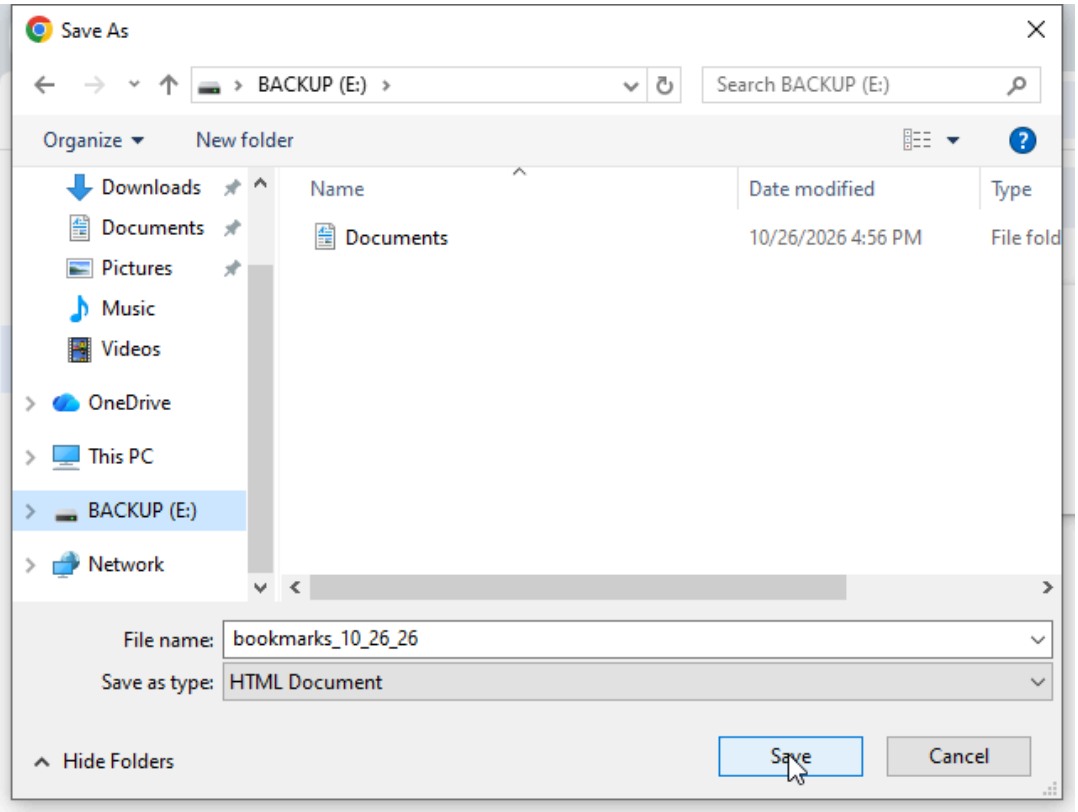


Figure 6.7. Save the bookmarks file on the second drive.

Saved passwords

Export the saved passwords from Chrome to a file. At the top right of the browser, select **More**, then **Passwords and autofill**, then **Google Password Manager**. On the left, choose **Settings**; in a narrow window, click the menu button at the top left first. To the right of "Export passwords," select **Download file**. A window titled Export Passwords From Chrome opens, with the file name Chrome Passwords. Click the second drive on the left, then **Save** (Figure 6.8).

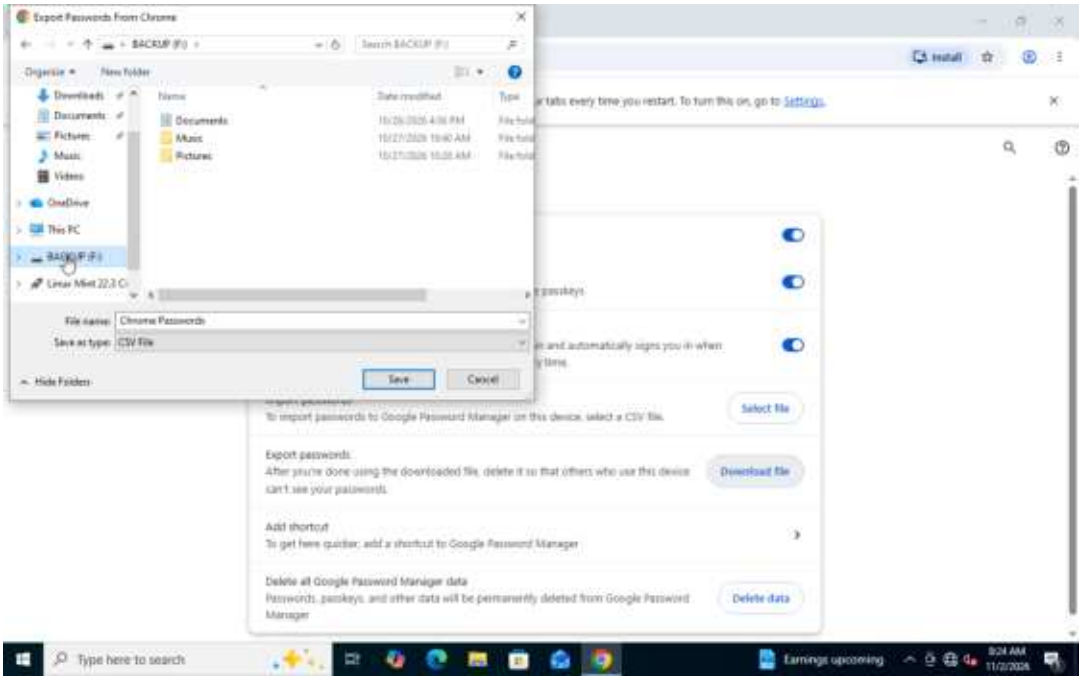


Figure 6.8. Save the passwords file on the second drive, here named BACKUP.

Treat this file like the passwords in it. Keep it on the second drive only, and keep it out of email. In chapter 10, we bring the passwords into Linux Mint and then delete the file.

Edge

For Edge, the steps differ. Open Edge. At the top right, select **Settings and more**, the three dots, then **Favorites**. In the Favorites panel, select **More options**, the three dots there, then **Export favorites** (Figure 6.9). Edge saves the favorites as one HTML file, named with the date. Click the second drive on the left, then **Save**.

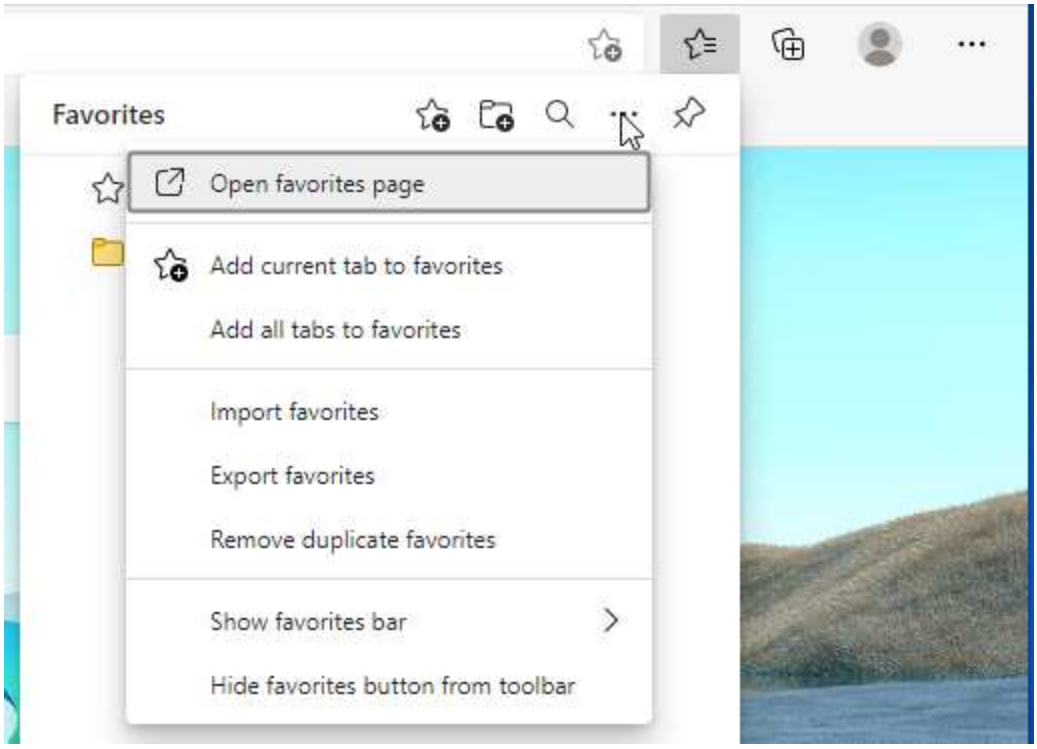


Figure 6.9. Edge's Favorites panel: More options, then Export favorites.

For the saved passwords, select **Settings and more**, then **Settings**, and on the page that opens, **Passwords**. Beside "Saved passwords," select **More actions**, then **Export passwords** (Figure 6.10). A window opens with the warning "Your passwords will be visible to anyone who can see the exported file." Select **Export passwords** again, click the second drive on the left, and save the file, Microsoft Edge Passwords. Treat it like Chrome's passwords file. In our lab, we took these steps in Edge 92 on Windows 10; a newer Edge may name the menus differently.

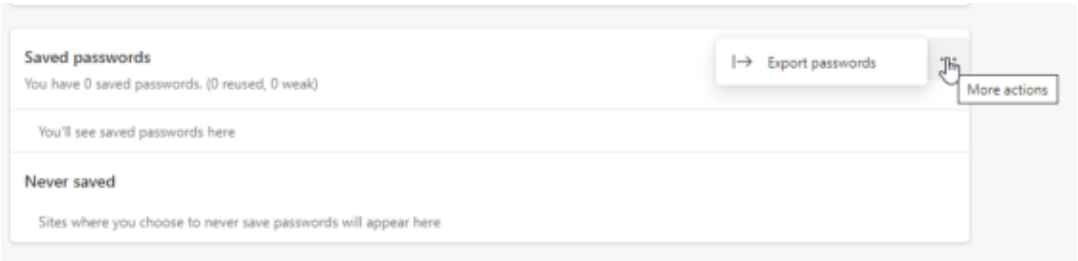


Figure 6.10. Beside Saved passwords, More actions, then Export passwords.

Outlook mail on this computer

If we read mail in Outlook, the program may keep some of it only on this computer. Microsoft's support page says that for a POP account, the kind that moves mail from the server onto the computer, "all of your information is already stored in a Personal Folders (.pst) file." In Outlook 2016 and newer, Outlook saves a new data file in `Documents\Outlook Files` unless someone moved it, so the copy of Documents usually includes it.

To check, open Outlook, choose **File**, then **Account Settings**, then **Account Settings** again, and click the **Data Files** tab. Each .pst file appears there with the account it belongs to and the folder that holds it. If Outlook keeps a .pst file outside Documents, open that folder in File Explorer and copy the file to the second drive. The same page adds, "You can move this file to a new computer and retain your information." Thunderbird can't read that file, so move the mail to the server first, with the steps at the end of chapter 5.

BitLocker

If chapter 4's BitLocker step was skipped, do it now.

Set the drive aside

When the folders, the bookmarks file, and the password file if we made one are on the second drive, and a few of the copies opened in a test, unplug it and set it aside. Keep it away from the computer during the install, so the only disk on offer, when choosing what to erase, is the internal one. We plug it back in for chapter 10. Before the erase, write down the Wi-Fi password and the password for each email account, since Windows may have typed them for years; the installer asks for the first in chapter 7, and Thunderbird asks for the second in chapter 11.

Try it

1. Plug in the second drive and copy the Documents folder to it.
2. Open one document from the second drive to prove the copy.
3. Export the Chrome bookmarks and copy the HTML file to the second drive.

7. Install Linux Mint

With the files copied in chapter 6 and the second drive unplugged, start the computer from the Linux Mint USB drive again, as in chapter 4, and wait for the live session. Now we install Linux Mint for good. We answer a few questions in the installer, and each answer has a reason.

Double-click **Install Linux Mint** on the desktop. Answer the screens in order, and click **Continue** after each one.

The screens

Welcome. Choose a language, because the installer and the desktop use it from here on.

Keyboard layout. Choose the one printed on the keys, so the typed letters match the ones on the screen.

Wireless. On Wi-Fi, choose the network and type its password. With a cable, the installer skips this screen.

Multimedia codecs. Tick **Install multimedia codecs**, so the computer plays common audio and video files. With Secure Boot on, as we left it in chapter 4, a password field for Secure Boot appears on the same screen. Pick a password and write it down. A prompt for it appears once, on the first restart (Figure 7.1).

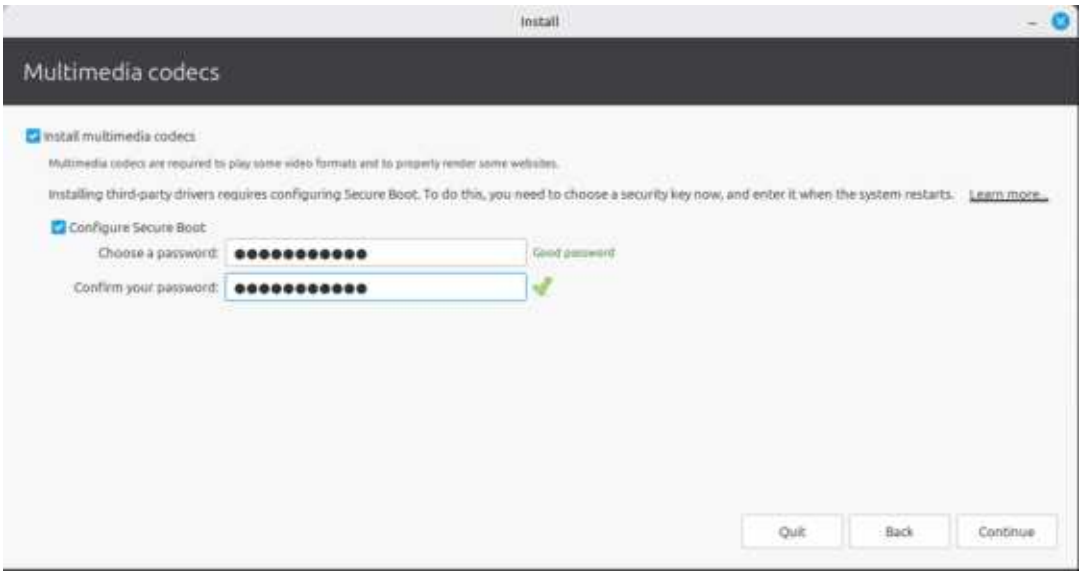


Figure 7.1. The codecs screen on a computer with Secure Boot on.

Installation type. Choose **Erase disk and install Linux Mint**. The Linux Mint team writes the rule for this choice: choose it "if Linux Mint is the only operating system you want to run on this computer and all data can be lost on the hard drive." Check the copies before going on (Figure 7.2). If no disk appears on this screen, turn to Disk mode in chapter 4. When we click **Continue** in step 5 below, the installer erases Windows and puts Linux Mint on the whole disk.



Figure 7.2. The installation type screen. Choose Erase disk and install Linux Mint.

Encrypt the disk. Before clicking Install Now, lock the disk. Under Erase disk and install Linux Mint, click **Advanced features...**. Choose **Use LVM with the new Linux Mint installation**, then tick the line below it, **Encrypt the new Linux Mint installation for security** (Figure 7.3). That line stays grey until after the LVM choice. LVM, short for Logical Volume Manager, divides the disk, and the installer's encryption builds on it. Click **OK**. The line "LVM and encryption selected." then appears on the installation type screen.

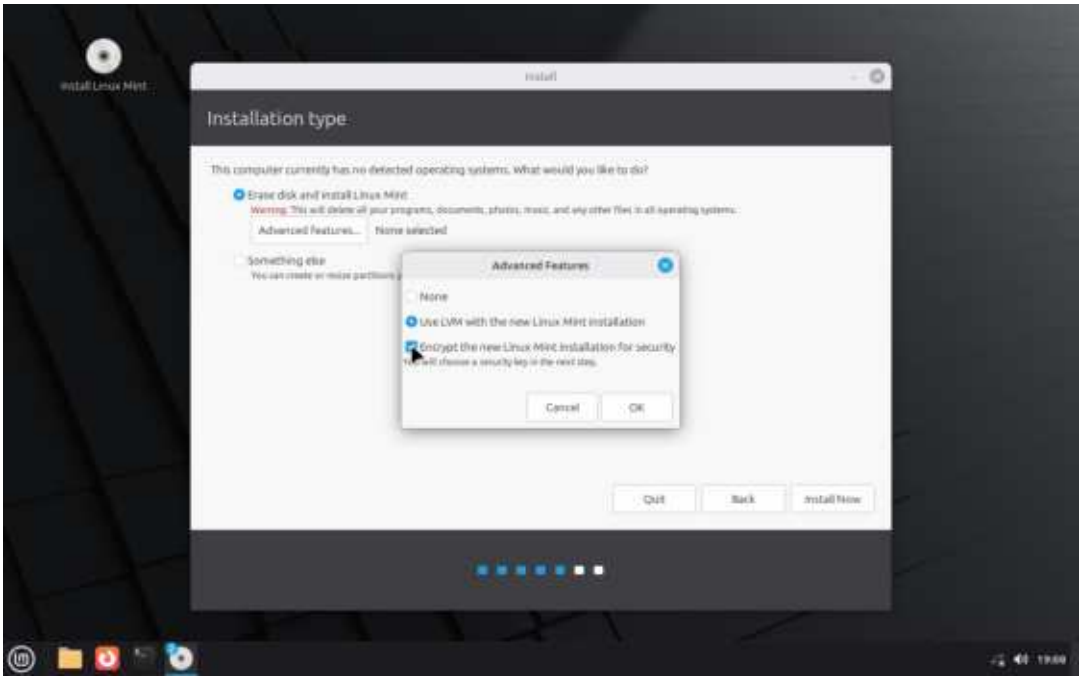


Figure 7.3. Advanced Features: LVM, then encryption.

1. Click **Install Now**. On the next screen, **Choose a security key** (Figure 7.4), the Linux Mint team gives the reason: "Disk encryption protects your files in case you lose your computer. It requires you to enter a security key each time the computer starts up."
2. Type a memorable key on both lines, and look for a green check when they match. Below them, the team adds a warning: "If you lose this security key, all data will be lost."
3. Write the key down and keep it away from the computer.
4. Leave **Enable recovery key** unticked; the written key does that job.
5. Then click **Install Now**, and in the window about writing the changes to disks, choose **Continue**.

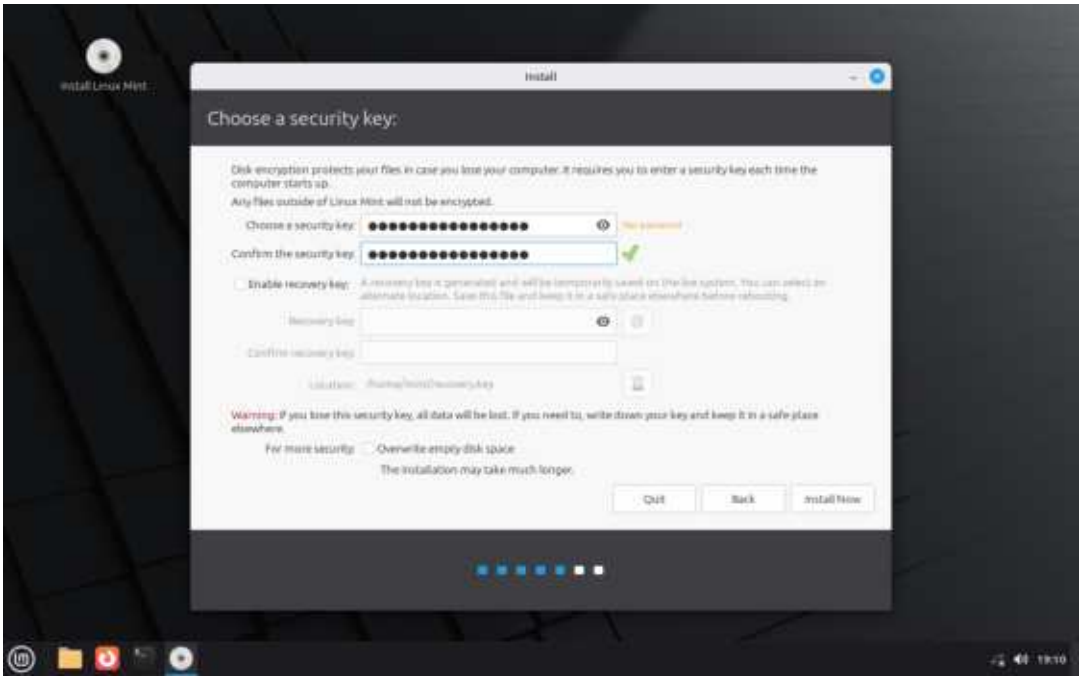


Figure 7.4. Choose a security key. Type it twice, and write it down.

Where are you? Click the location on the map, so the computer sets its clock to local time.

Who are you? Type the full name of the person who will use the computer. The Linux Mint team notes that this name "is only used locally, in the screensaver and on the login screen." The installer fills in a computer name and a username from it. Keep the username, and change the computer name to lowercase letters with no spaces or punctuation, such as samlaptop. Then type a strong password twice (Figure 7.5). Leave **Require my password to log in** chosen, and leave **Encrypt my home folder** unticked; the whole disk is already encrypted.

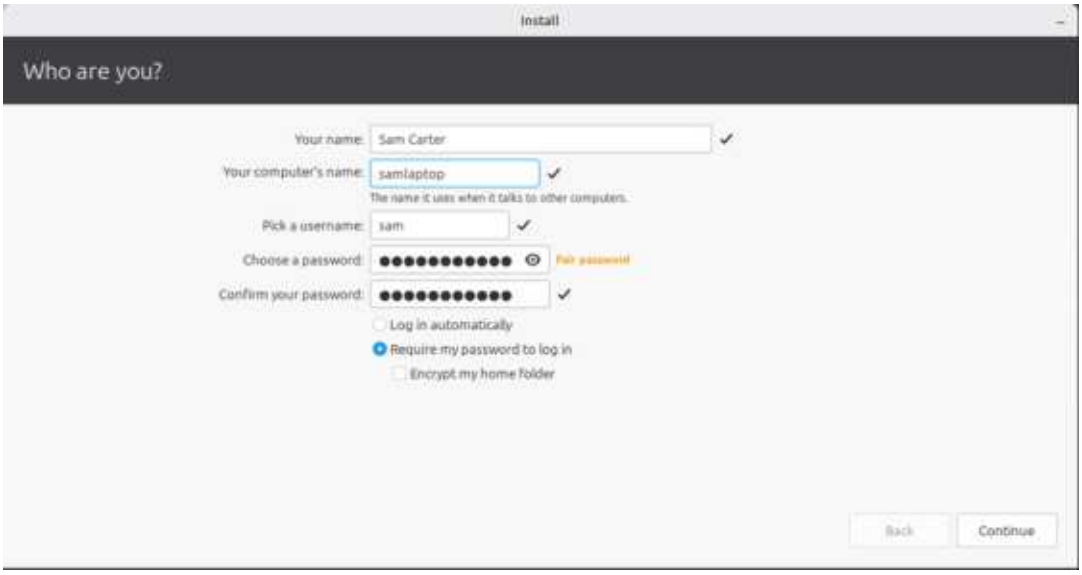


Figure 7.5. Who are you? A full name, the computer's name, a username and a password.

The wait and the restart

The installer copies Linux Mint to the disk while a slideshow runs. In our lab, the copy took about eight minutes. When it finishes, click **Restart Now**. At the message about removing the installation medium, pull out the USB drive and press Enter.

If Secure Boot stayed on, a blue screen titled **Perform MOK management** appears once, before Linux Mint starts. MOK stands for Machine Owner Key. On this screen, we approve the key for the drivers in Linux Mint, so the computer accepts them. Choose **Enroll MOK**, **Continue** and **Yes**, type the Secure Boot password from the codecs step, and select **Reboot** (Figure 7.6). With Secure Boot off, the computer starts without this screen.

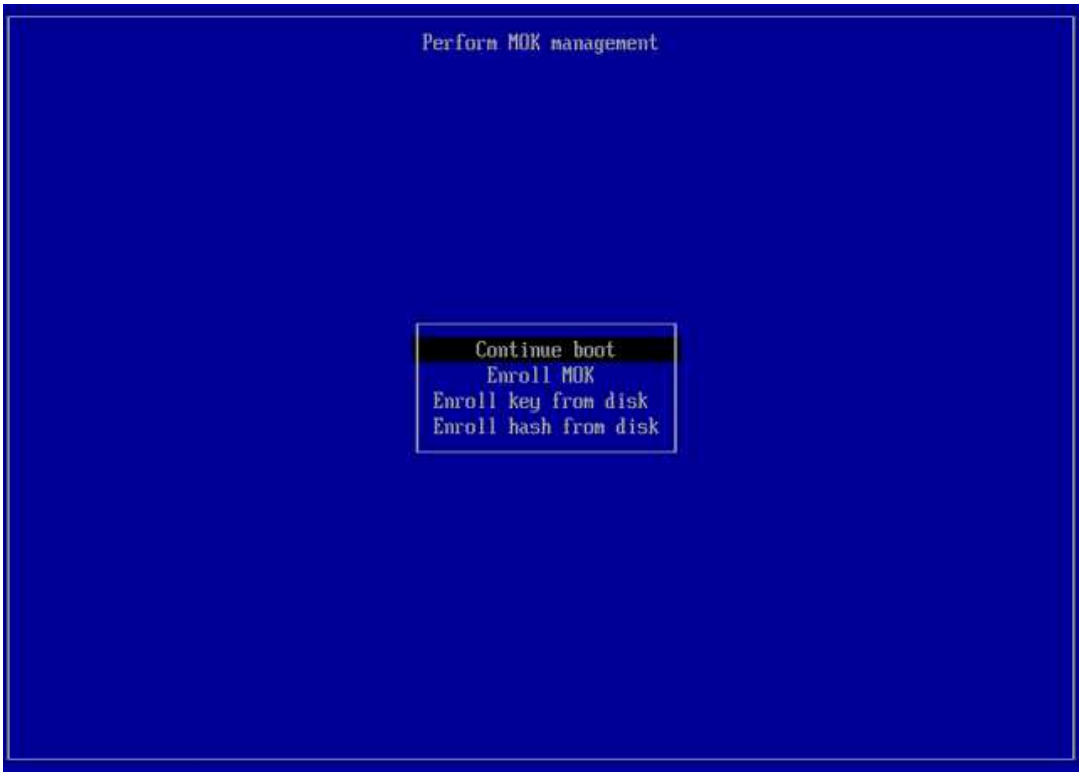


Figure 7.6. The blue key screen on the first restart, with Secure Boot on. Choose Enroll MOK.

Because we encrypted the disk, a prompt for the security key appears next, on this start and every one after: "Please unlock disk," followed by the disk's name (Figure 7.7). Type the key and press Enter. The usual sign-in screen follows.



Figure 7.7. The security key prompt at every start.

Sign in

The computer starts Linux Mint without the USB drive, and the sign-in screen appears with the new user's name on it. Sign in with the password. On the Welcome Screen, click **First Steps** to read what the Linux Mint team suggests doing first (Figure 7.8). One step is the **Update Manager**, with its shield icon beside the clock. We use it in chapter 14, after a snapshot, a saved copy of the system to return to. Until then, an orange dot on the shield marks the updates that wait. The Linux Mint team puts the snapshot first, before daily use, and we take it in chapter 14.

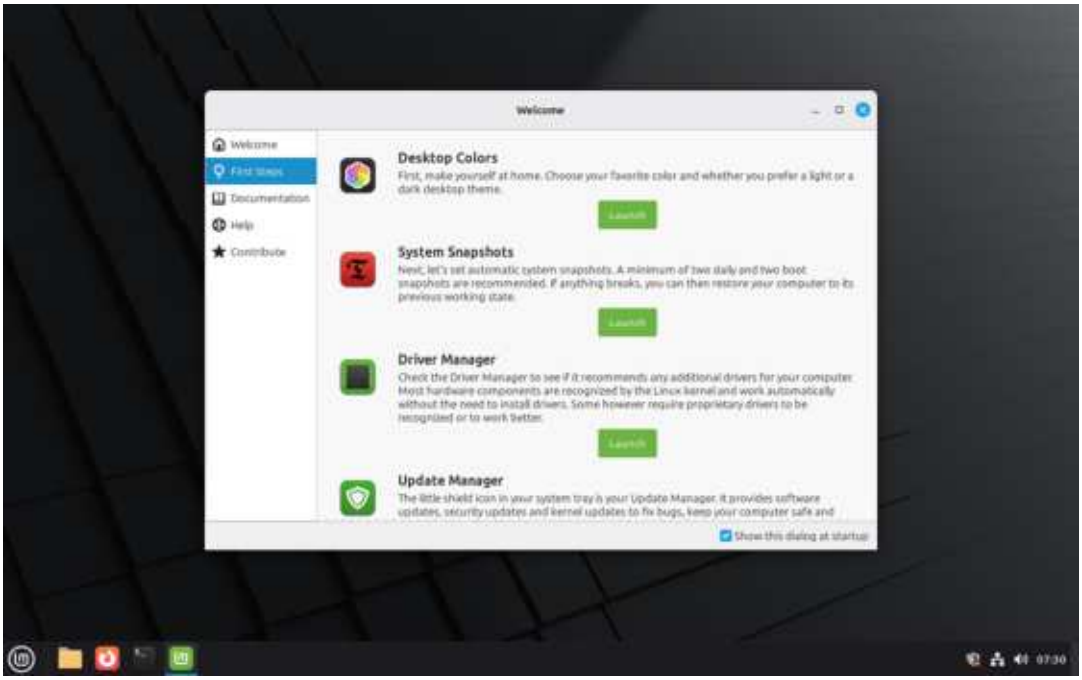


Figure 7.8. First Steps on the Welcome Screen.

Try it

1. Open the menu, click **Log Out**, and sign back in with the password.
2. Find the shield icon beside the clock. We open Update Manager in chapter 14.
3. Play a sound and a video to check the codecs.
4. Open the menu, type welcome, and open the Welcome Screen again to read First Steps.

8. Find the way around Cinnamon

The Linux Mint team calls its desktop Cinnamon. Most of it matches what we used on Windows. In this chapter, we find the Cinnamon versions of the Start menu, the taskbar, File Explorer and Settings.

The menu

Look at the lower left corner of the screen, where the menu button takes the place of Start on Windows. Click it to see the menu open on categories with readable names: Office, Internet, Sound & Video. Along the bottom of the menu is a search box, ready for typing. Type the name of a program, such as firefox, to see it in the menu (Figure 8.1). Click it to open it.

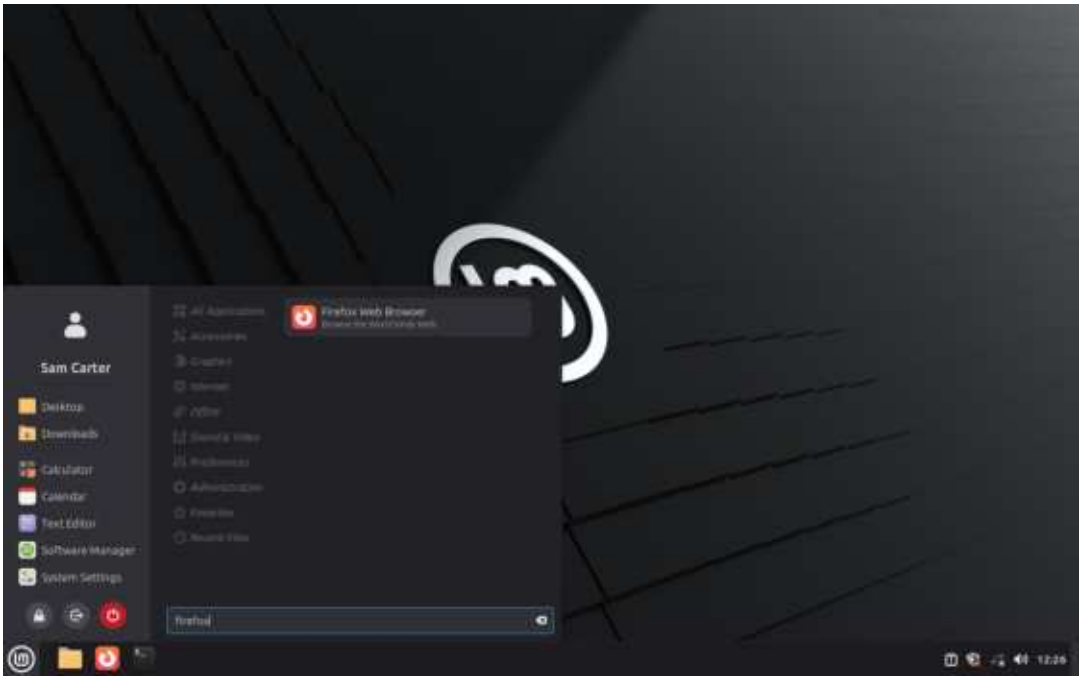


Figure 8.1. Type a program's name in the menu's search box.

The panel

The Linux Mint team calls the bar along the bottom of the screen the panel. We use it as we used the taskbar (Figure 1.1). At the left are buttons for Files, Firefox and Terminal, and each program we open gets one there too, beside them. The network, the volume and the clock are at the right.

To add a program to the panel for good, open the menu, type its name in the search box, right-click it in the results, and choose **Add to panel** (Figure 8.2). Its button joins the others at the left (Figure 8.3).

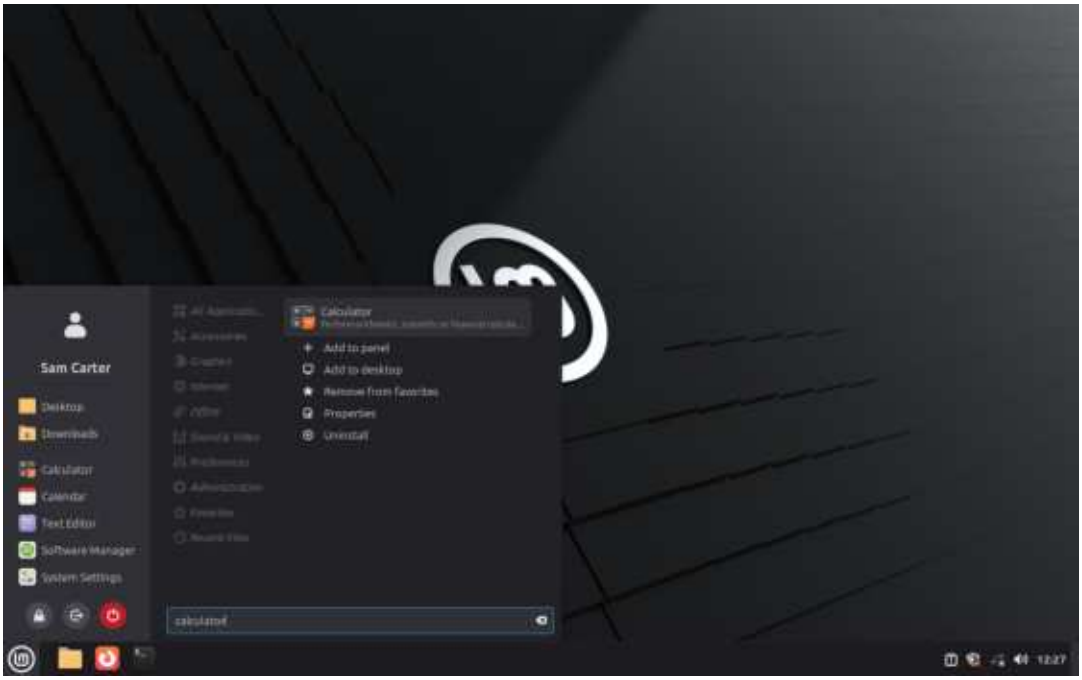


Figure 8.2. Right-click a program in the menu and choose Add to panel.



Figure 8.3. Calculator's button on the panel, beside Files, Firefox and Terminal.

Files

Open Files, the file manager, from the yellow folder at the left of the panel. We use it as we used File Explorer on Windows: folders open in a window, and on the left is a list of places, such as Home, Documents and Downloads. In Figure 8.4, Files appears beside File Explorer; to have Files open on the drives, the way File Explorer did, follow Appendix A after chapter 18, once Terminal is familiar. We keep our documents in the same named folders as before.

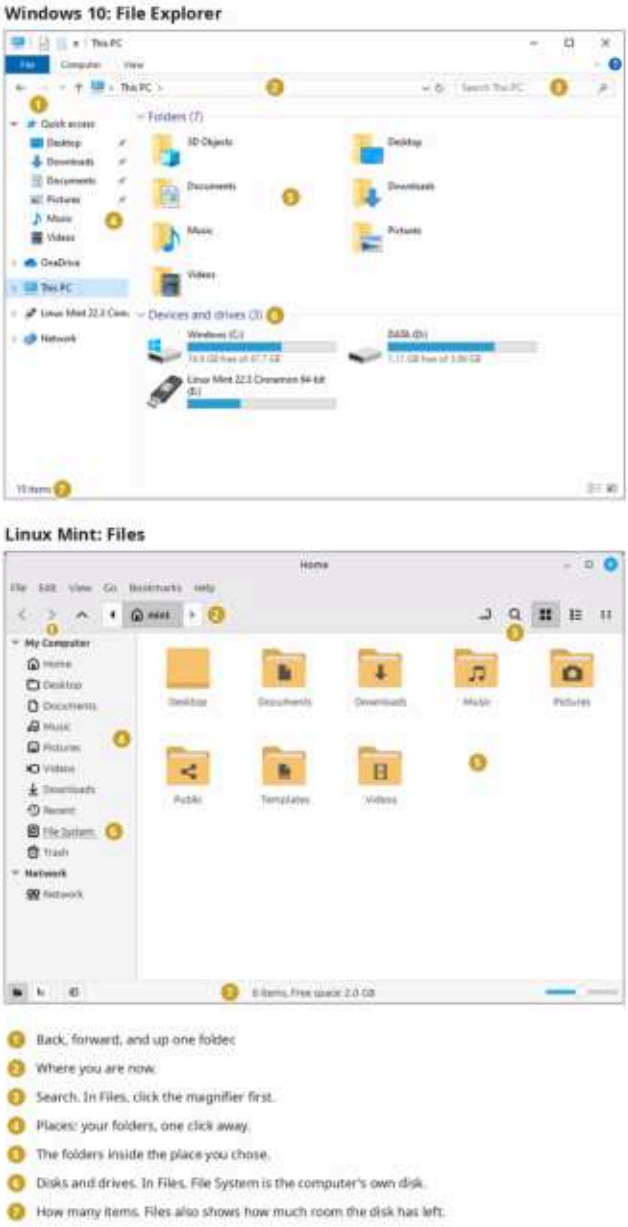


Figure 8.4. Files on Linux Mint beside File Explorer on Windows 10.

The window buttons

Each window has three buttons at the top right corner: minimize, maximize and close, in the same order as on Windows. Click **Minimize** to hide the window in its button on the panel. Click **Maximize** to make the window fill the screen. Click **Close** to shut it.

System Settings

Windows has Settings and the older Control Panel. Linux Mint has System Settings, at the lower left of the menu (Figure 8.5). Its choices appear under headings such as Appearance and Preferences; scroll down for more. To change how the computer looks or behaves, start there.

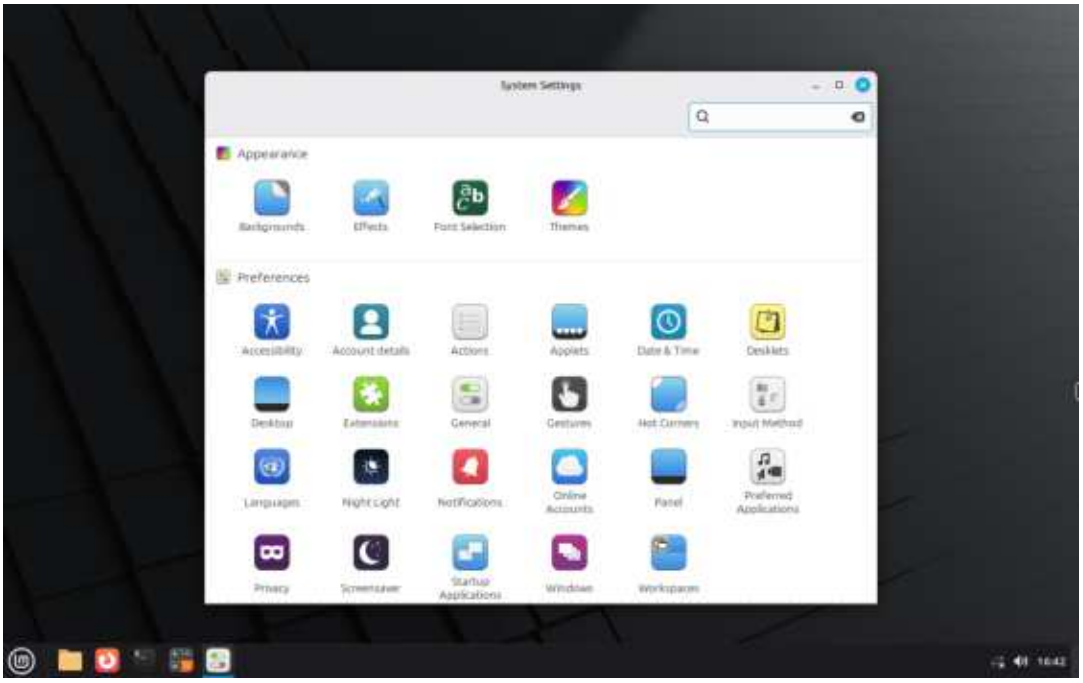


Figure 8.5. System Settings, under Appearance and Preferences.

Workspaces

In Linux Mint, we have more than one desktop, called workspaces. Press Ctrl+Alt+Up arrow to see four of them, labeled Workspace 1 to Workspace 4, in two rows (Figure 8.6). Click one to work there, or press Esc to go back. It is fine to keep working in one workspace and leave the others empty.



Figure 8.6. Workspace 1 to Workspace 4, after Ctrl+Alt+Up arrow.

System Monitor

On Windows, Task Manager ends a program that stops responding. On Linux Mint, System Monitor does that job. Open the menu, type system monitor, and open **System Monitor**. It opens on the **Processes** tab, a list of the programs running now.

Click the search button at the top, and type part of the program's name, such as calc for Calculator, whose row reads gnome-calculator. Select the row, and click **End Process** at the lower left (Figure 8.7). System Monitor asks first, and warns, "Ending a process may destroy data, break the session or introduce a security risk. Only unresponsive processes should be ended." Click **End Process** again (Figure 8.8). In our lab, Calculator's window closed. Any work in that program since the last save is lost.

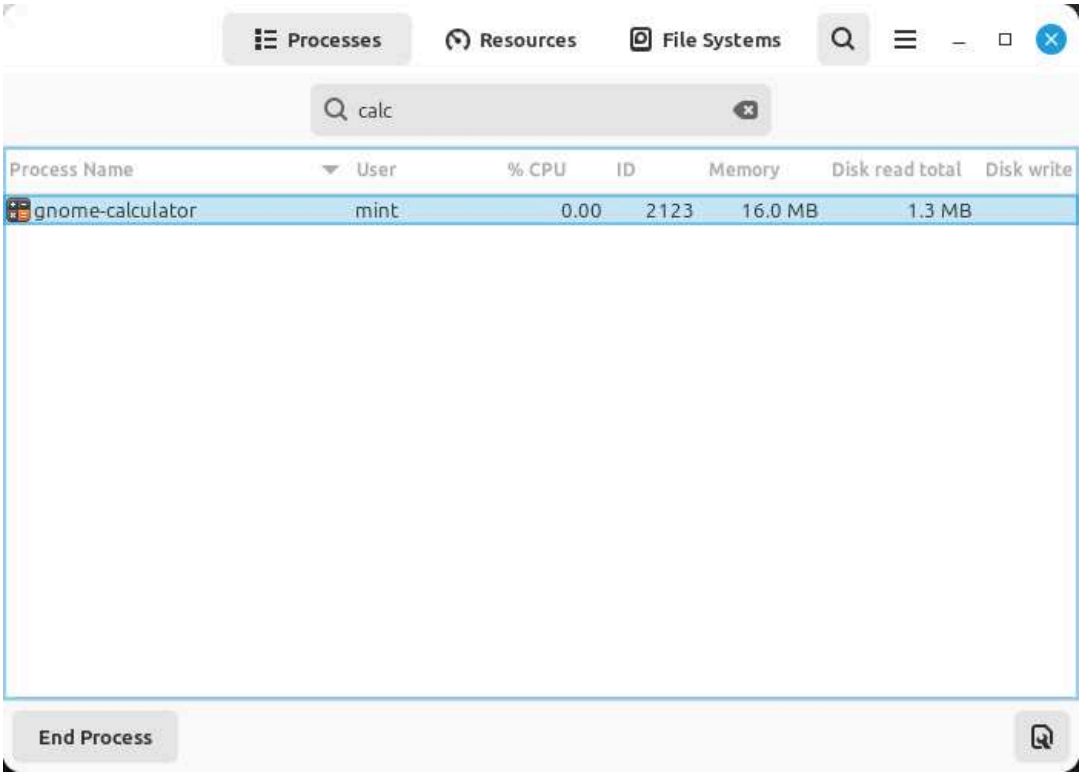


Figure 8.7. System Monitor, with Calculator selected. Click End Process.

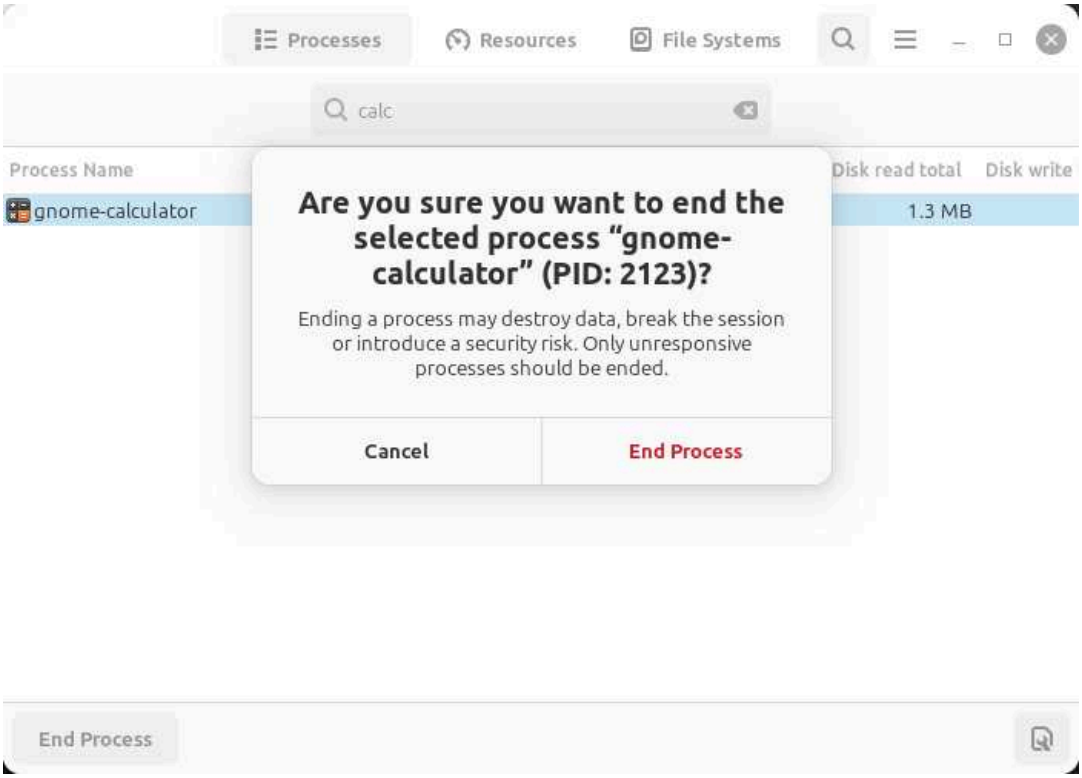


Figure 8.8. System Monitor asks before it ends a program.

Try it

1. Open the menu and type the name of a program in the search box.
2. Add Calculator to the panel: search for it in the menu, right-click it, and choose Add to panel.
3. Open Files and find Downloads in the list on the left.
4. Open System Settings from the menu and scroll through its headings.
5. Press Ctrl+Alt+Up arrow to see the workspaces, then press Esc.
6. Open Calculator, find it in System Monitor, and end it with End Process.

9. Seven things every computer does

In this chapter, we do seven jobs on Linux Mint: turn the computer on, open a program, write and save a letter, find it again, print it, send an email, and shut it down. We know each job from Windows.

Turn it on

Press the power button on the computer. At the first prompt, type the security key from chapter 7 and press Enter. At the next, type the password (Figure 9.1) and press Enter. The desktop appears (Figure 9.2).



Figure 9.1. Linux Mint's sign-in screen.



Figure 9.2. The Linux Mint desktop.

Open a program

Click the menu button at the lower left of the screen. Click the Office group to see LibreOffice Writer, a word processor compatible with Microsoft Word (Figures 9.3 and 9.4). Click **Writer** to open it.



Figure 9.3. The menu, open.



Figure 9.4. The Office group, with LibreOffice Writer.

Write and save a letter

The first time we open Writer, a Tip of the Day window appears (Figure 9.5). Click **OK** to close it. Type a few lines (Figure 9.6). When finished, press Ctrl+S to save. On the first save of a new document, the Save window opens on Documents. Type a name next to Name, then click **Save** (Figure 9.7).

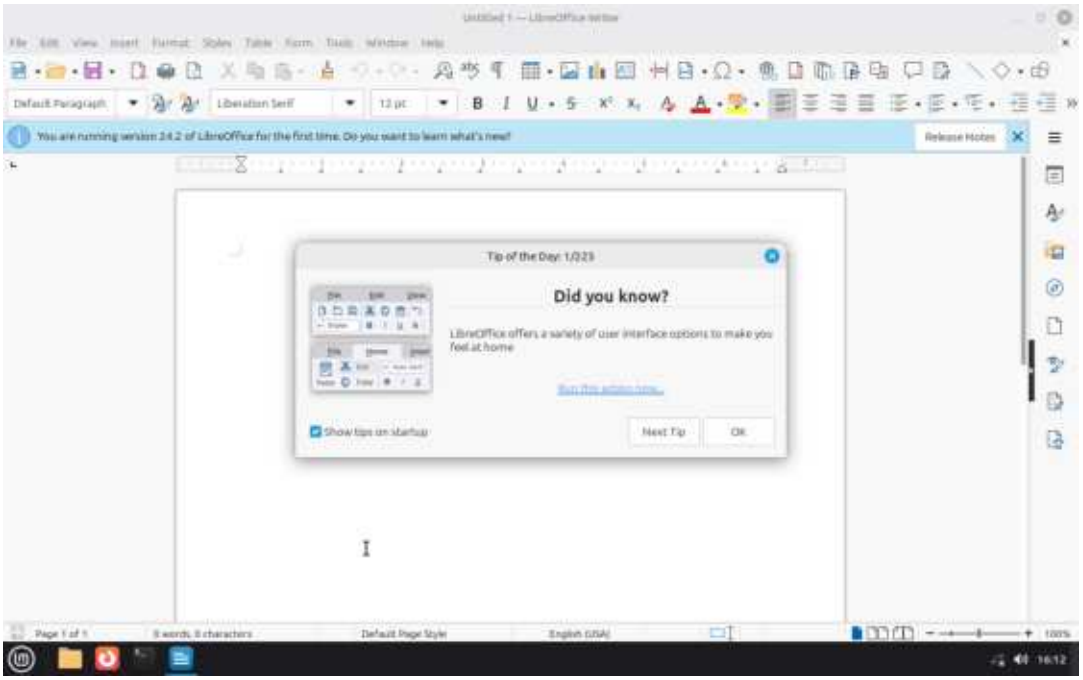


Figure 9.5. The Tip of the Day on opening Writer for the first time.

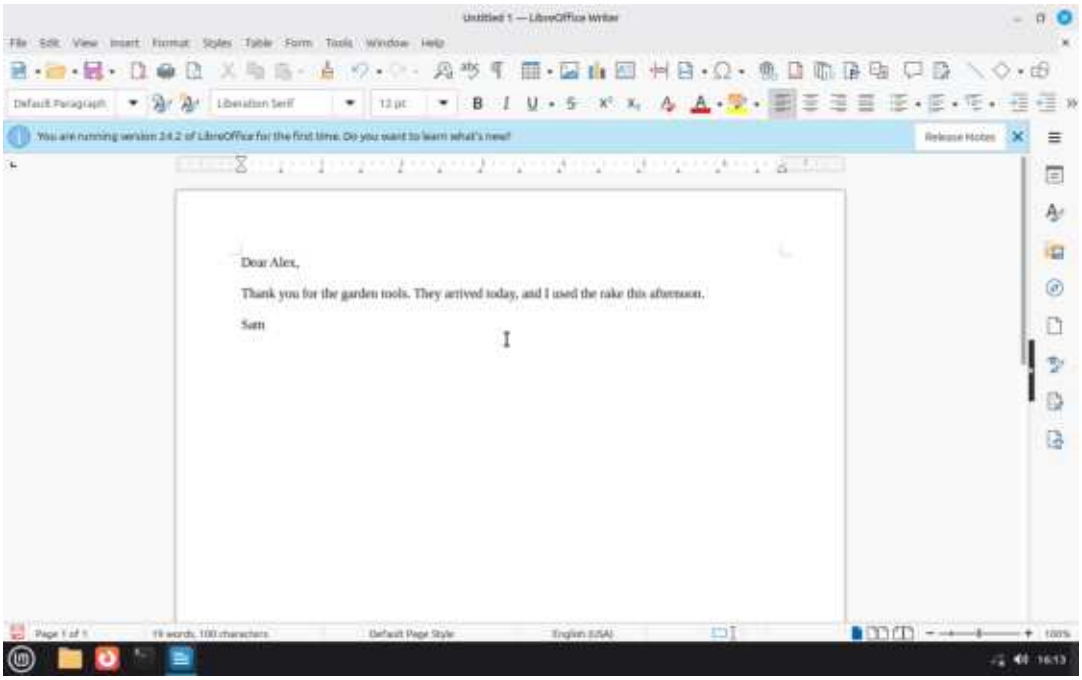


Figure 9.6. A letter in Writer.

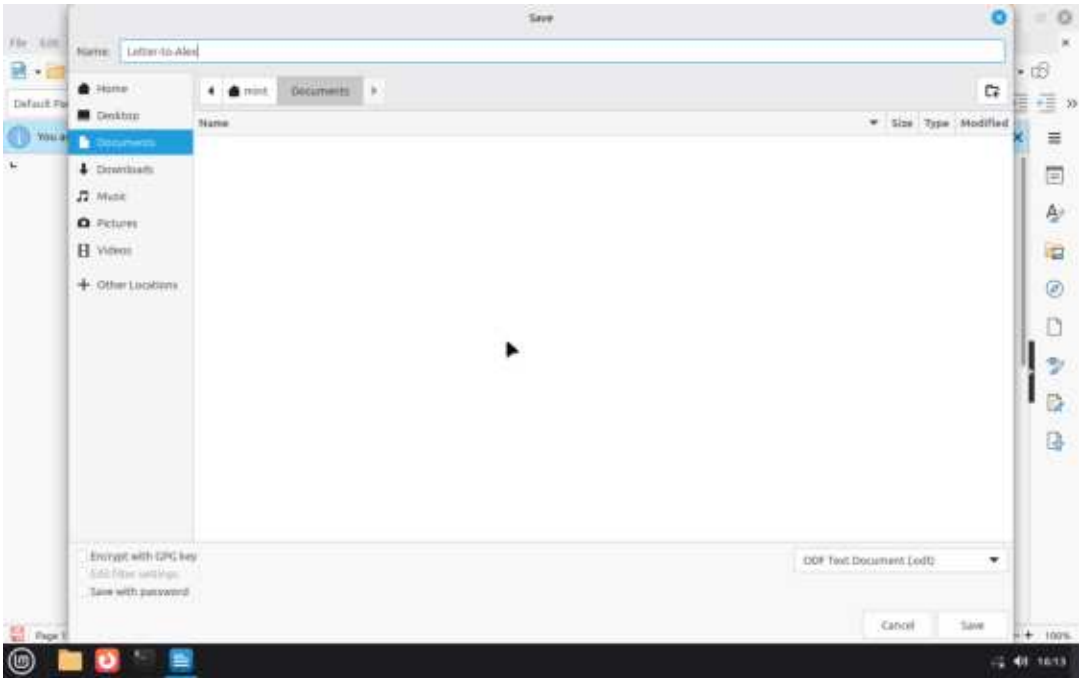


Figure 9.7. The Save window for a new document.

Find it again

Click the yellow folder on the panel to open Files, from chapter 8. On the left side, click **Documents** to see the files on the right, each as a small page with its name under it. Look for the name typed in the Save window (Figure 9.8). Double-click the letter to open it again.

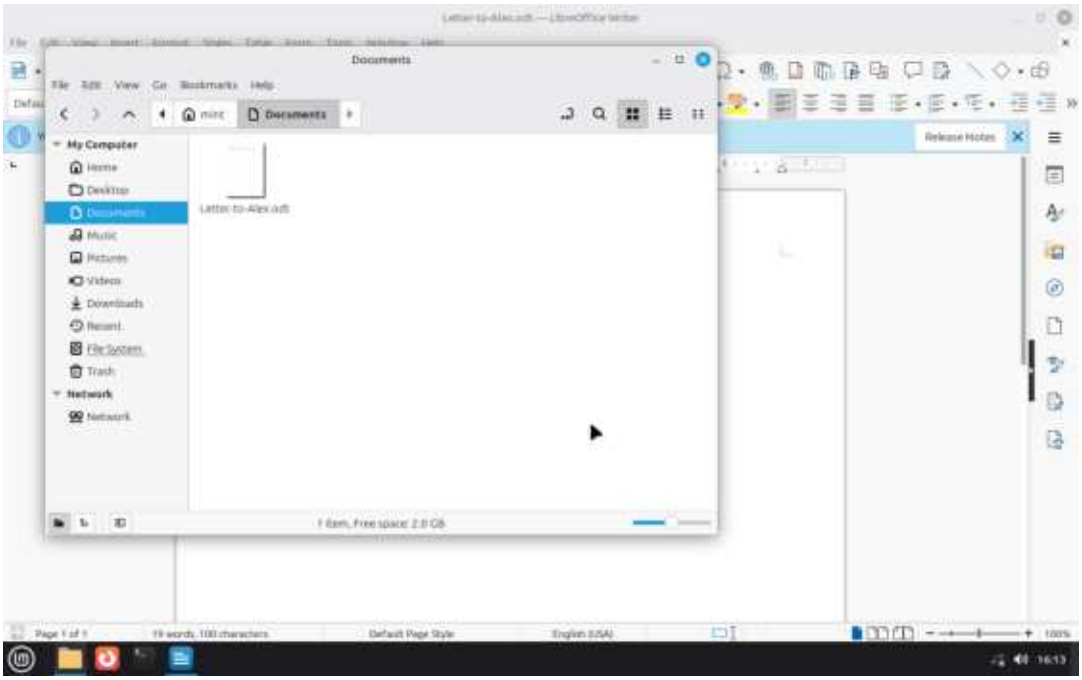


Figure 9.8. Files, in the Documents folder.

Print it

In Writer, click **File** at the top left, then Print. Or press Ctrl+P. In the Print window, pick the printer and click **Print** (Figure 9.9). For a computer with none yet, we add one in chapter 13.

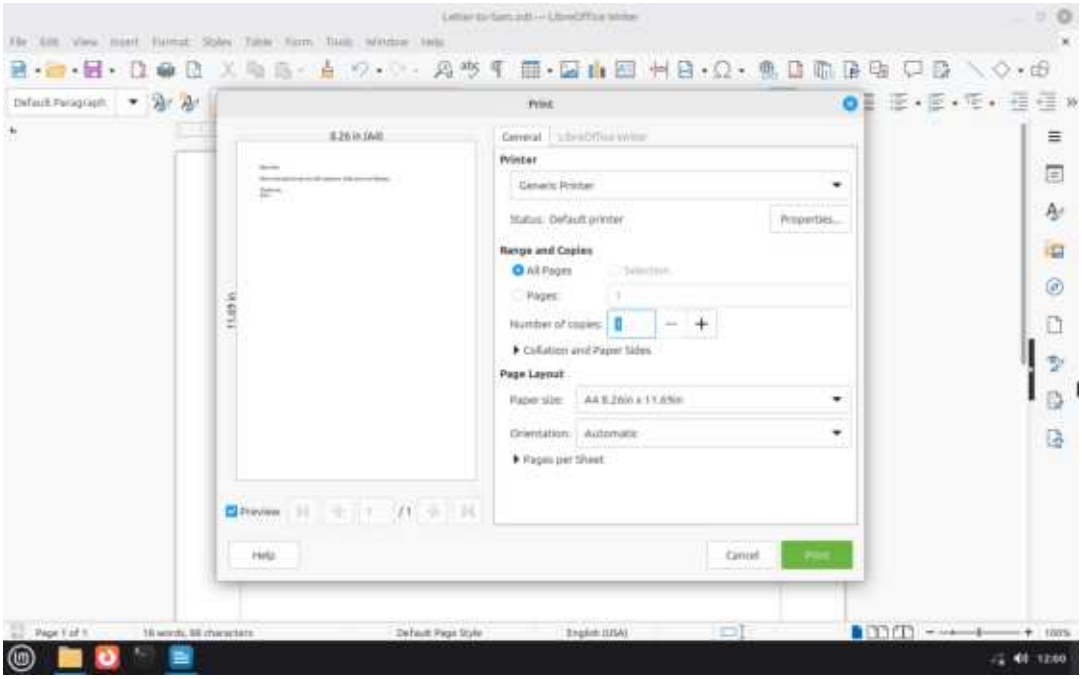


Figure 9.9. The Print window.

Send an email

Click the orange Firefox button on the panel to open the web browser. Go to the mail service's web page, such as mail.google.com for Gmail, and sign in with the email address and password for that account (Figure 9.10). The inbox opens (Figure 9.11). Click **Compose** at the top left. In the New Message window, type the address in Recipients, a few words in Subject, and the message in the large space below. Then click **Send** (Figure 9.12). Figures 9.11 and 9.12 show Gmail in Chrome, which we add to Linux Mint in chapter 11; the pages look the same in Firefox.

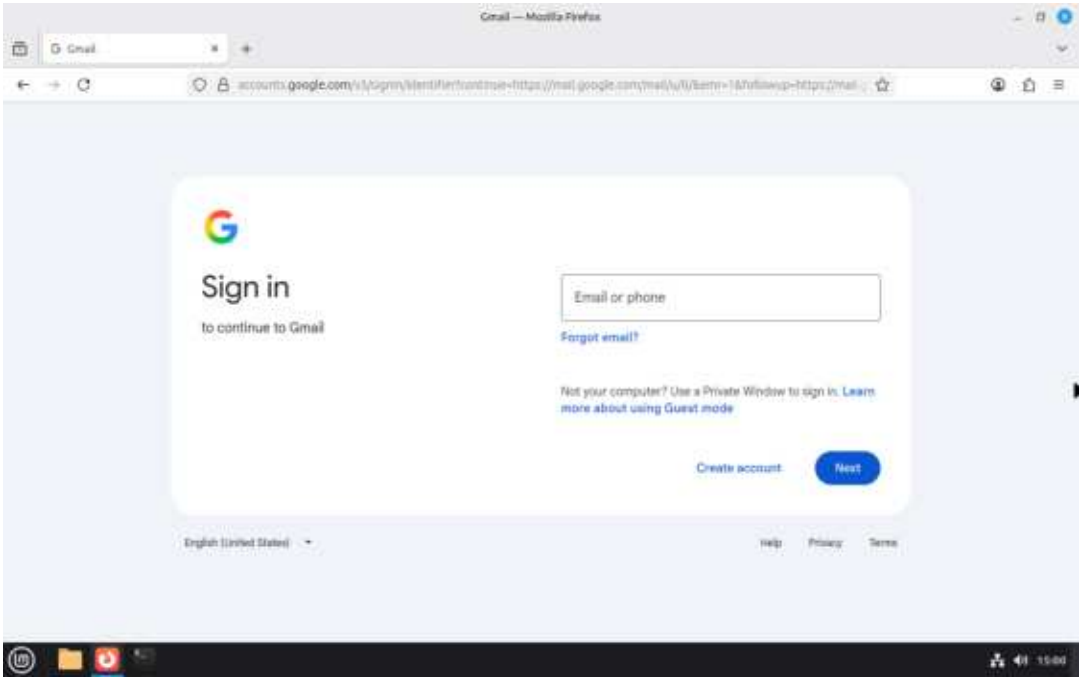


Figure 9.10. Gmail's sign-in page.

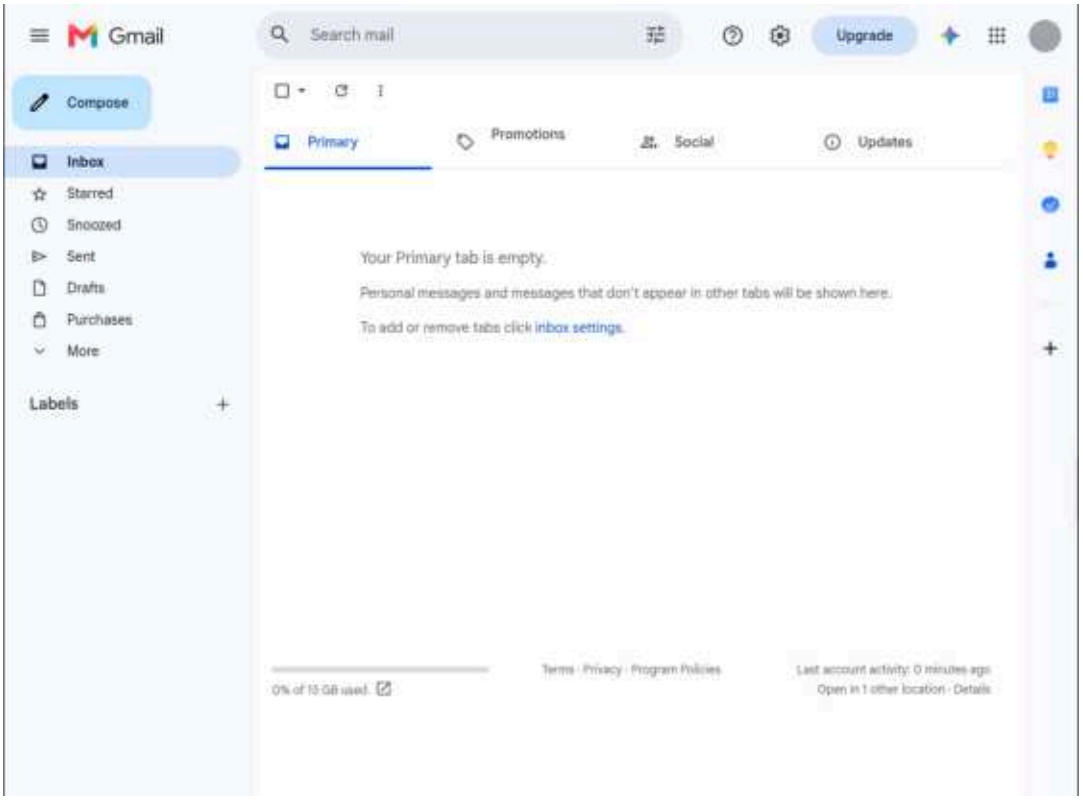


Figure 9.11. A Gmail inbox.

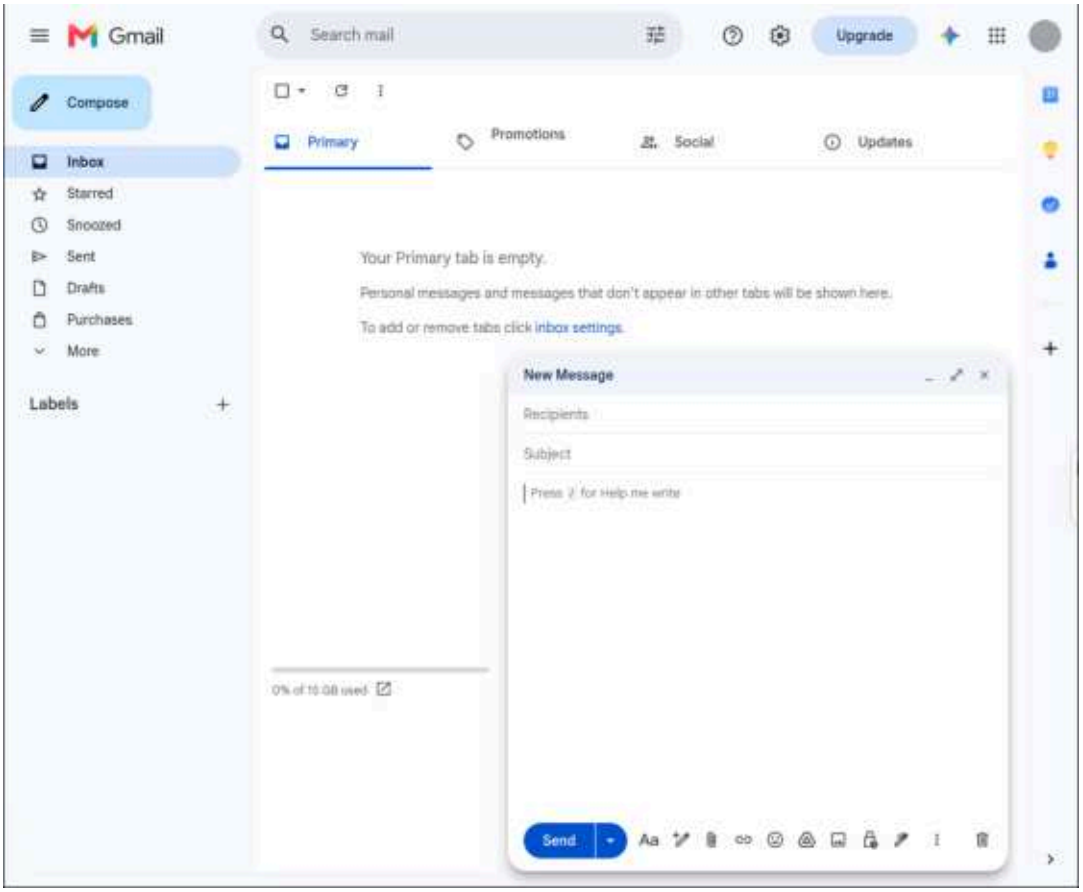


Figure 9.12. A new message in Gmail.

Shut it down

Click the menu button at the lower left. At the bottom left of the menu, click the red power button to see four choices: **Cancel**, **Suspend**, **Restart**, and **Shut Down** (Figure 9.13). Click **Shut Down**. The computer turns off.

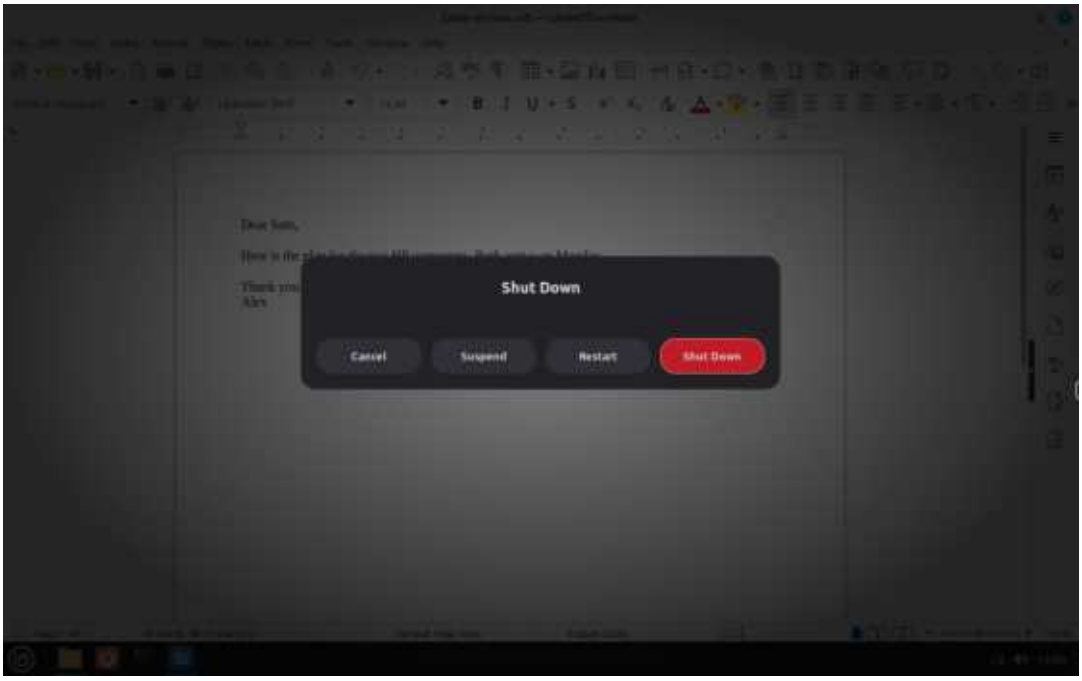


Figure 9.13. The Shut Down window.

Try it

1. Turn the computer on and wait for the desktop.
2. Open the menu and find LibreOffice Writer in the Office group.
3. Open Writer, type one sentence, and press Ctrl+S. Type a name in the Save window and click **Save**.
4. Open Files, click **Documents**, find the saved document, and double-click it.
5. Press Ctrl+P to see the Print window, then click **Cancel**.
6. Open Firefox and go to the mail page.
7. Open the menu, click the red power button, then **Shut Down**.

10. Import the files and settings into Linux Mint

In chapter 6, we copied the folders and the Chrome bookmarks to a second drive, and perhaps a file of saved passwords. Now, we import them into Linux Mint and put each one in its place.

The home folder

Click the yellow folder on the panel to see Home in Files, with eight folders in it (Figure 10.1). Six have the same names as on Windows: Desktop, Documents, Downloads, Music, Pictures and Videos. On Windows 7, four of them had My in front, as in chapter 6. The other two, Public and Templates, come with Linux Mint.

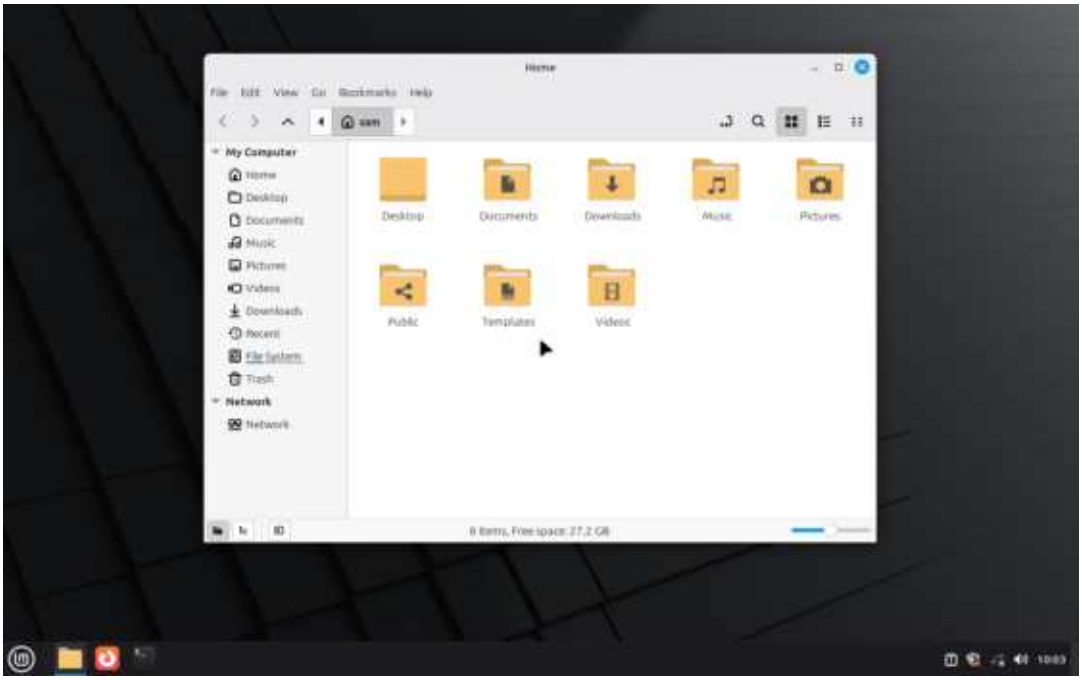


Figure 10.1. Files at Home, with its eight folders.

Copy the folders into Home

Plug the second drive into the computer. A window opens for it, and the drive's name appears under Devices on the left, with an eject button beside it (Figure 10.2). The drive may also hold a folder named System Volume Information. Copy only the folders from chapter 6.

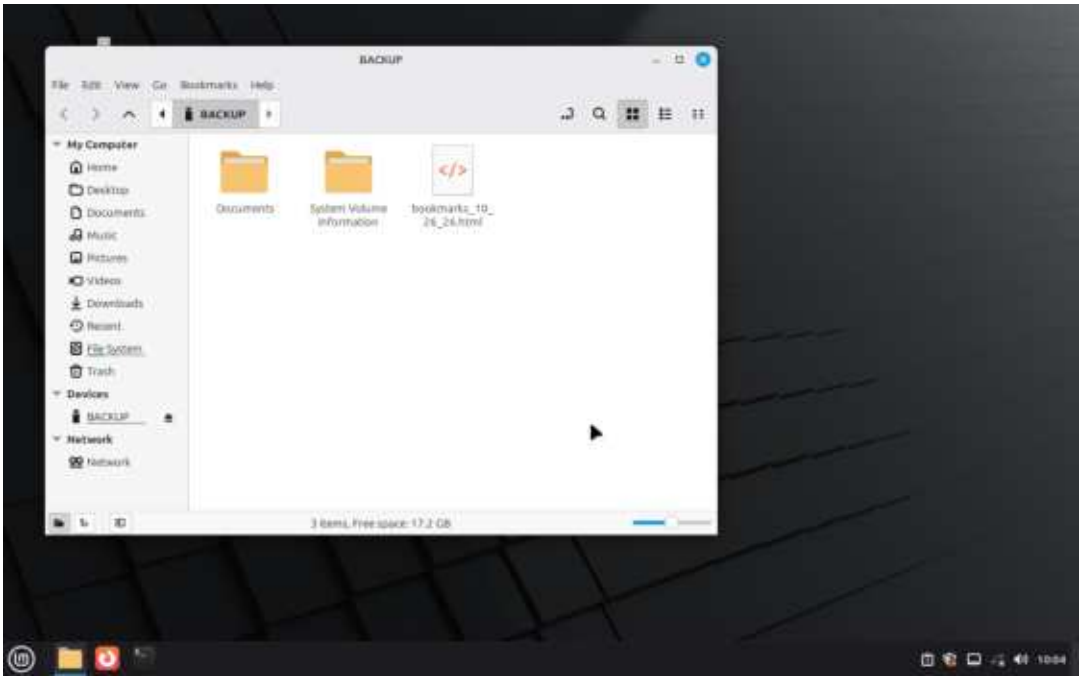


Figure 10.2. The second drive, here named BACKUP, open in Files.

Copy one folder's contents at a time. In the drive's window, open Documents (My Documents, if the copy came from Windows 7). Press Ctrl+A to select everything in it, right-click one of the selected files, and choose **Copy** (Figure 10.3). Then click **Documents** on the left, right-click an empty space, and choose **Paste** (Figure 10.4). To fetch the next folder, click the drive's name under Devices, and do the same for Pictures, Music, Videos, Downloads and Desktop.

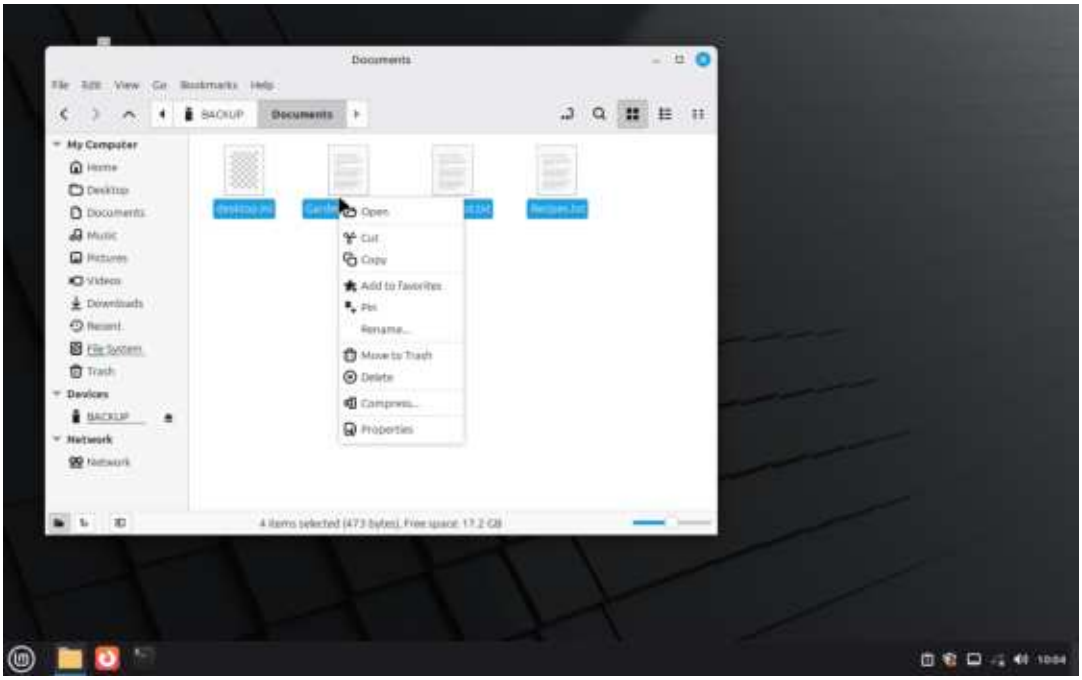


Figure 10.3. Documents on the drive, every file selected. Choose Copy.

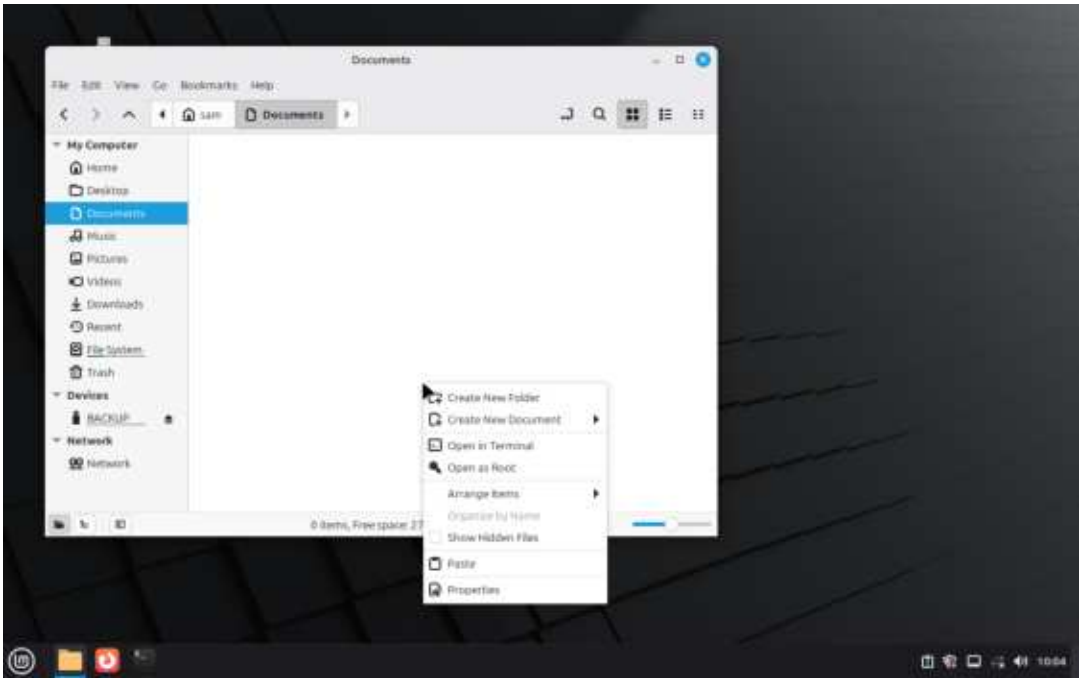


Figure 10.4. Documents in the home folder. Right-click an empty space and choose Paste.

A file named desktop.ini may be among the copied documents. Windows hides it, and Files lists it.

Check the copies

Open a few of the copies to check them: a document, a photo, a song. Files may treat a text document from the drive as a program, and a double-click on one brings up a window with a choice: run it or display it. Click **Display** to see it in the Text Editor (Figure 10.5). Double-click a photo to see it in Image Viewer (Figure 10.6). Double-click a song to hear it in Celluloid, Linux Mint's media player; its window stays black the whole time (Figure 10.7).

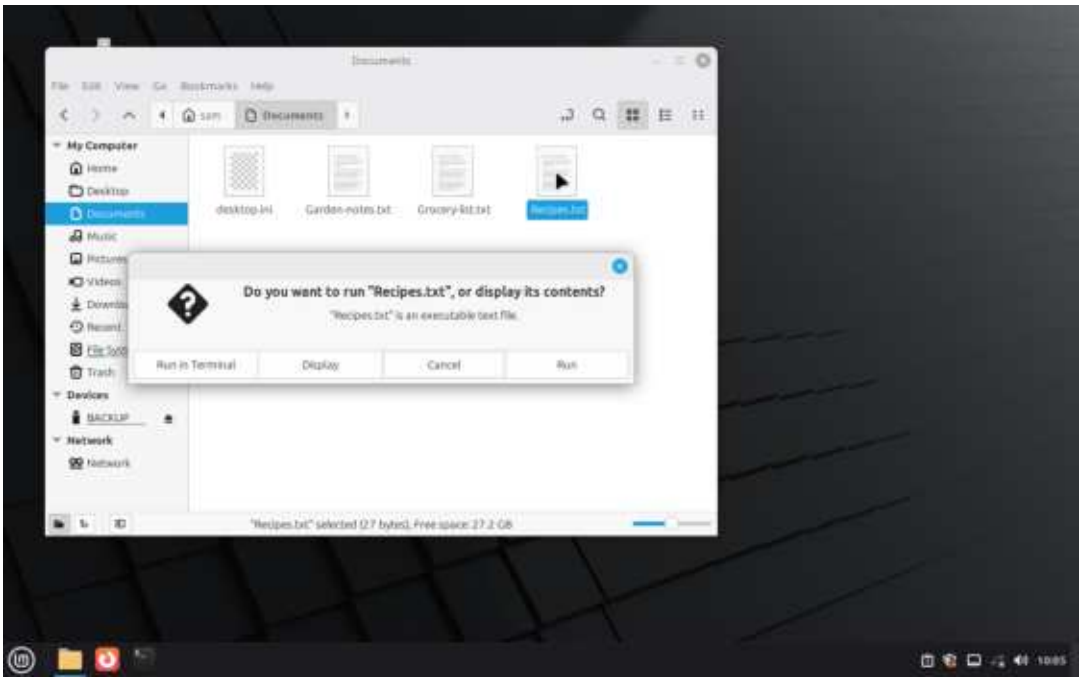


Figure 10.5. For a text file from the drive, click Display.

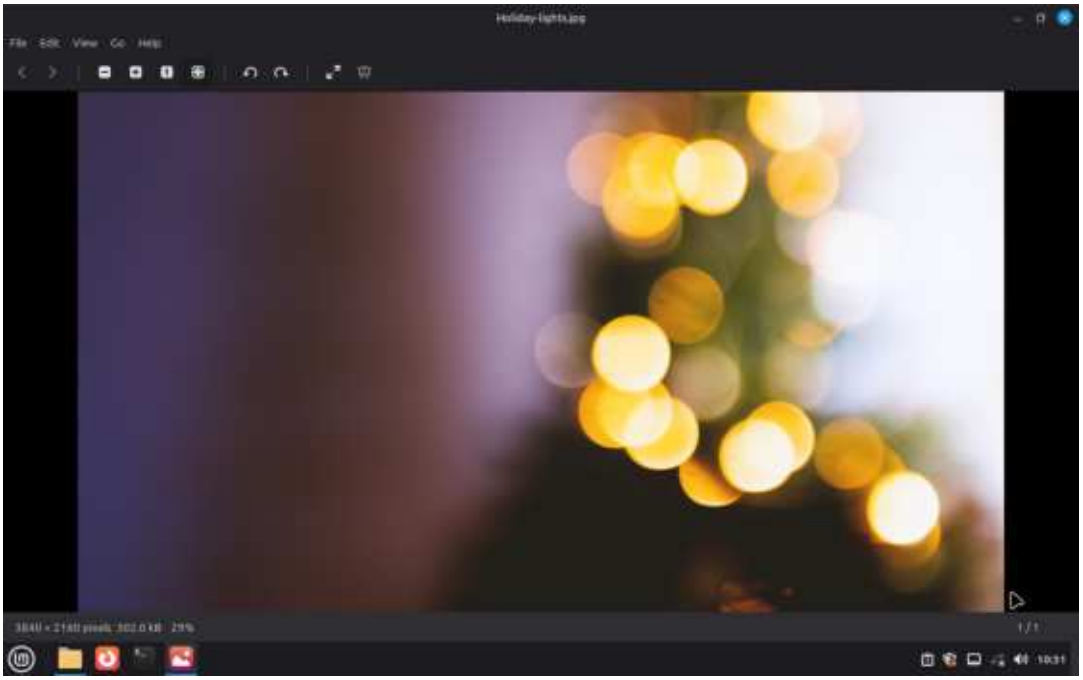


Figure 10.6. A photo from the drive, open in Image Viewer.

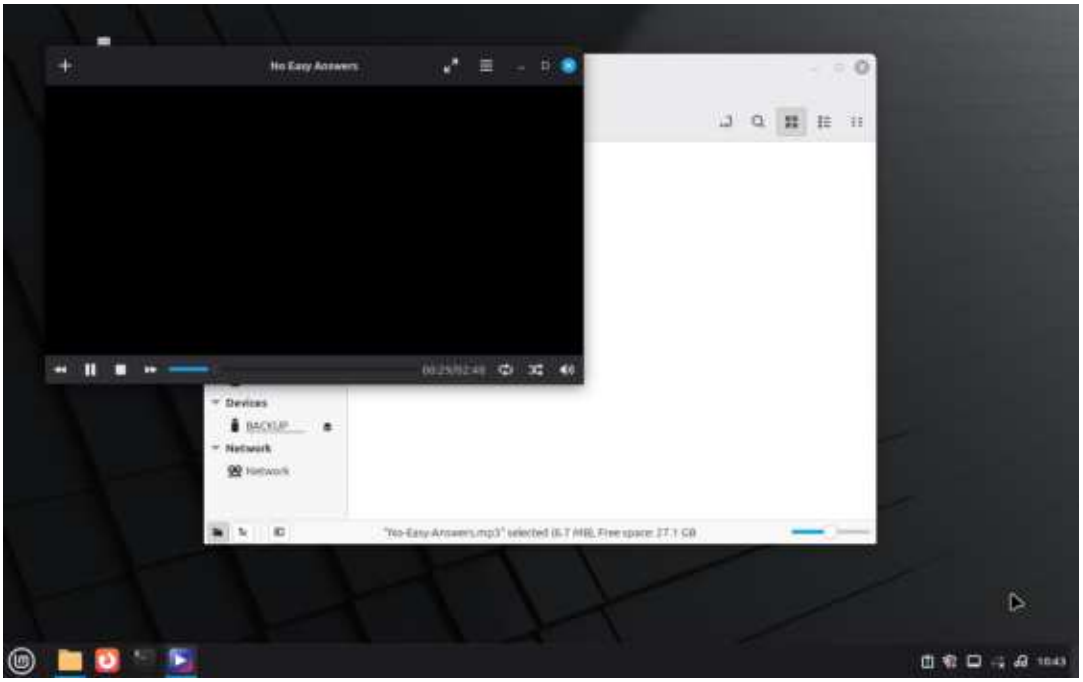


Figure 10.7. A song from the drive, playing in Celluloid.

Bookmarks

Open Firefox from the panel. On the welcome screen, click **Skip this step** until **Start browsing** appears, and choose it.

Click the menu button at the top right, the three lines, then **Bookmarks**, then **Manage bookmarks**. In the Library window, click **Import and Backup** and choose **Import Bookmarks from HTML** (Figure 10.8). Select the drive on the left and pick the HTML file from chapter 6: Chrome's name starts with bookmarks, and Edge's carries the date. Click **Open**. The bookmarks appear under Bookmarks Menu, with the bar from Chrome or Edge as a folder (Figure 10.9).

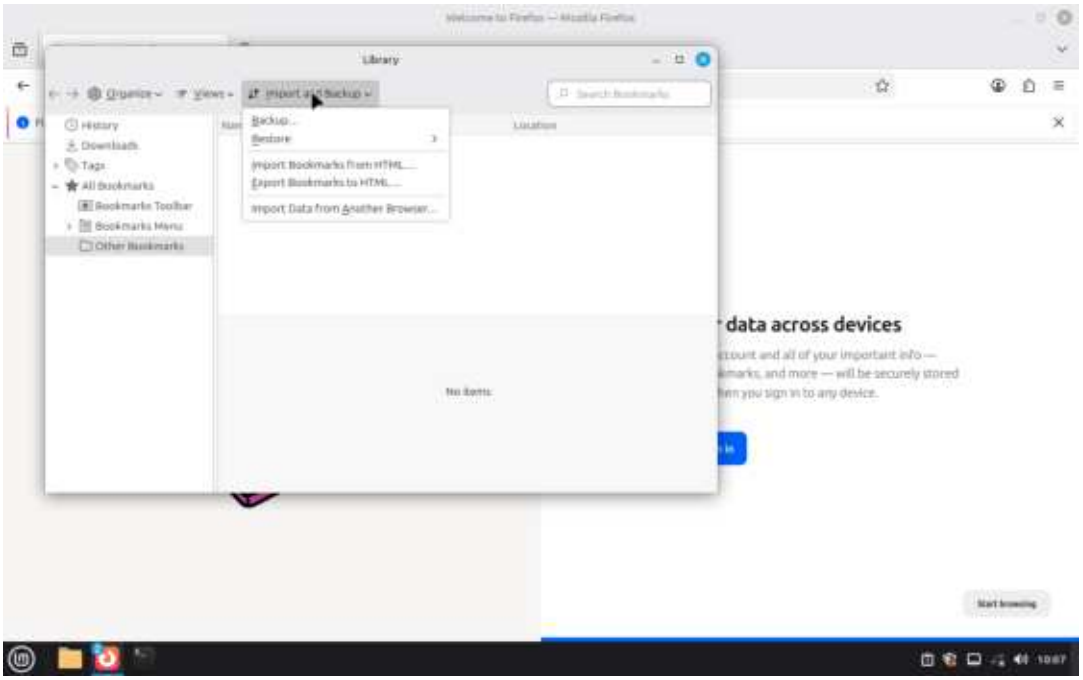


Figure 10.8. In the Library window, Import and Backup, then Import Bookmarks from HTML.

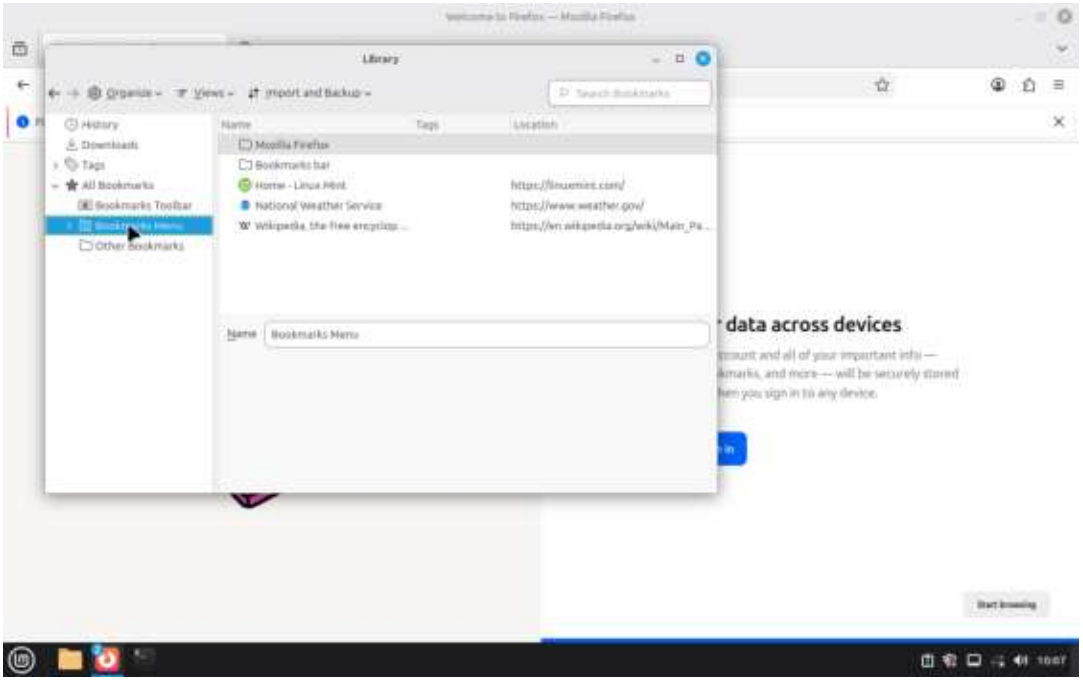


Figure 10.9. The bookmarks from Chrome, under Bookmarks Menu.

Passwords

If we exported the passwords from Chrome or Edge in chapter 6, we bring them in now. Click the menu button, then **Passwords**. At the top right, click the three dots and choose **Import from a File**. Select the drive on the left, pick the passwords file, Chrome's or Microsoft Edge Passwords, and click **Import** (Figure 10.10). Read the message for how many came in (Figure 10.11).

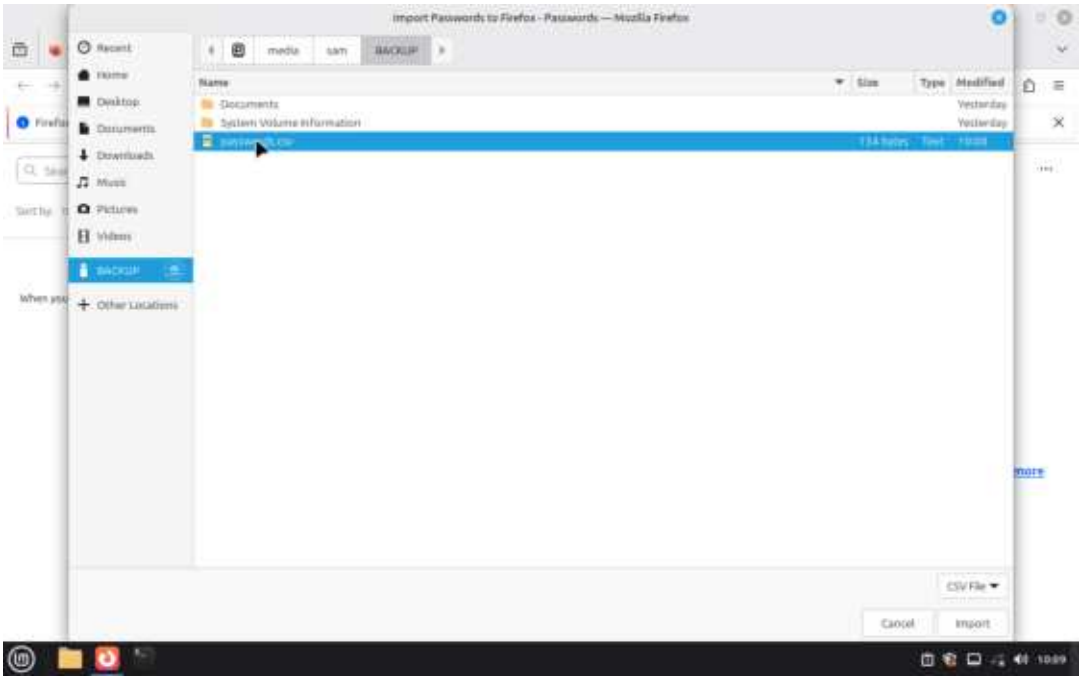


Figure 10.10. Pick the passwords file on the drive and click Import.

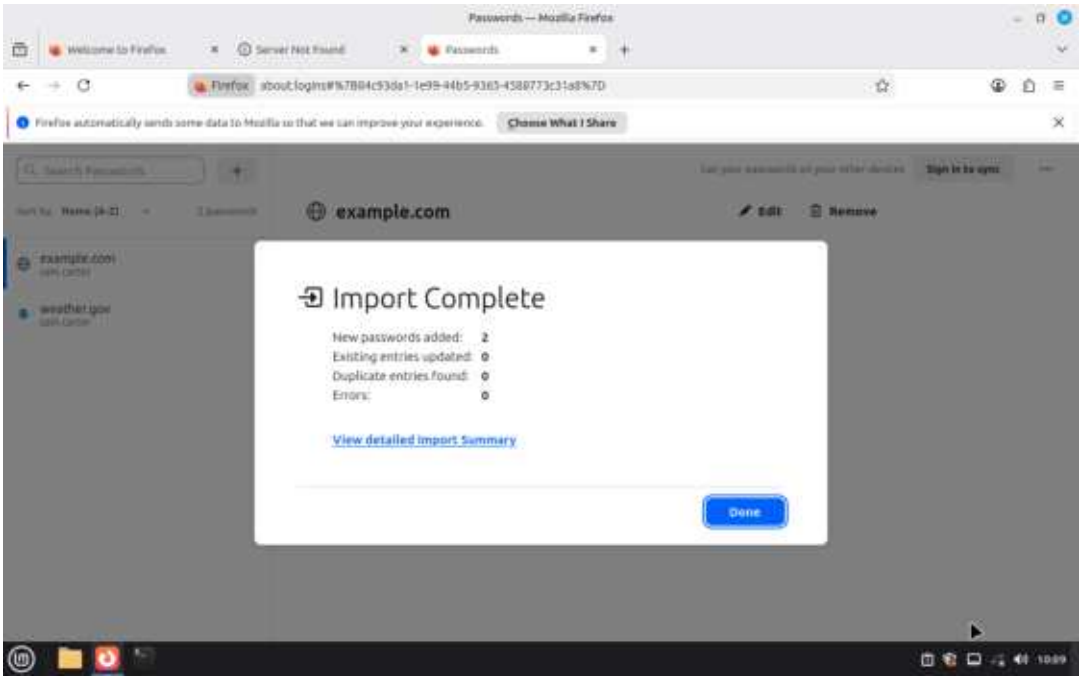


Figure 10.11. The count of imported passwords, in a message from Firefox.

Now, delete the passwords file from the drive. Mozilla's help page notes that Chrome writes them into this file "in a readable form," and Edge's warning in chapter 6 said as much of its file. Open Files, click the drive's name under Devices, right-click the file, and choose **Delete**, not Move to Trash. When asked whether to "permanently delete" it, click **Delete** (Figure 10.12).

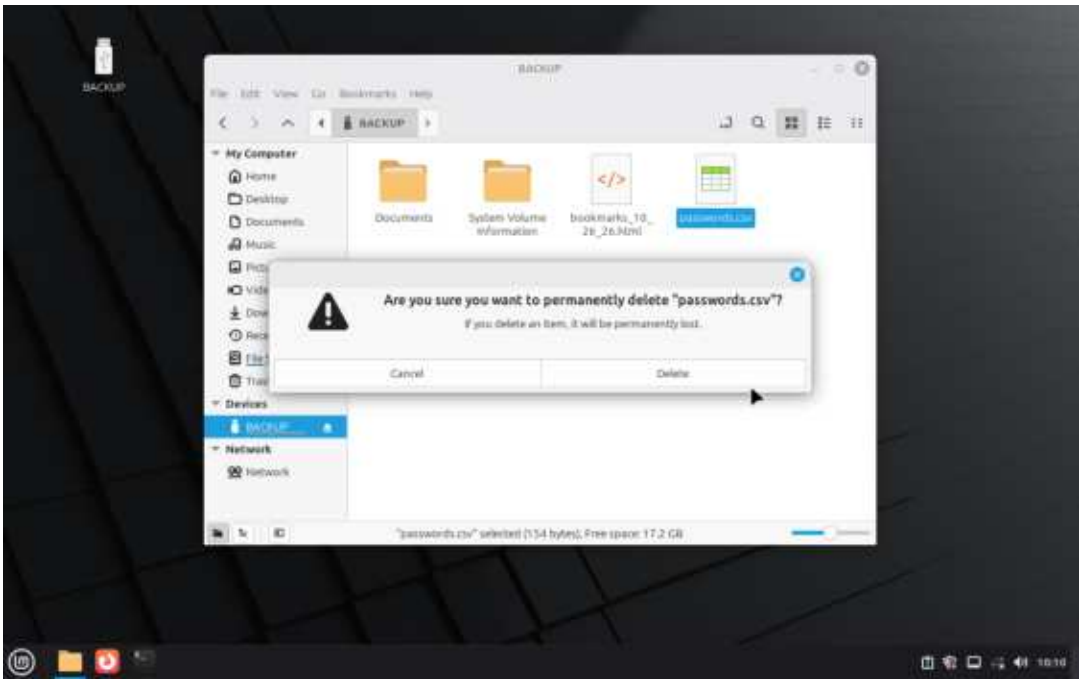


Figure 10.12. Delete the passwords file.

Unplug the drive

Click the eject button beside the drive's name under Devices, then unplug it. We now have the documents, bookmarks and passwords on Linux Mint.

Try it

1. Open Files and count the eight folders in Home.
2. Copy the files in Pictures on the second drive into Pictures in Home.
3. Open a copied text file, and click Display if that choice appears.
4. Import the bookmarks file into Firefox, then look under Bookmarks Menu.
5. Import the passwords file, then delete it from the drive with Delete, not Move to Trash.

Part 3. The daily desktop

11. The browser and email

We use the web on Linux Mint as we did on Windows, and the same goes for the mail an email service keeps on its server. We pick the browser and decide whether to read mail there or in a mail program.

Firefox

Firefox comes with Linux Mint, and in chapter 10, we brought the bookmarks, and any passwords, into it. Open it from the panel and go to any familiar page from Windows. To stay with Firefox, skip to "Mail in the browser."

Chrome, for those who want it

To keep using Chrome, get it from Google's site. In Firefox, go to google.com/chrome and click **Download Chrome**. The heading "Get Chrome for Linux" appears, with "64 bit .deb (For Debian/Ubuntu)" already selected (Figure 11.1). Above the last button, the page reads, "By downloading Chrome, you agree to the Google Terms of Service and Chrome and ChromeOS Additional Terms of Service." Read that line, then click **Accept and Install** (Figure 11.2).

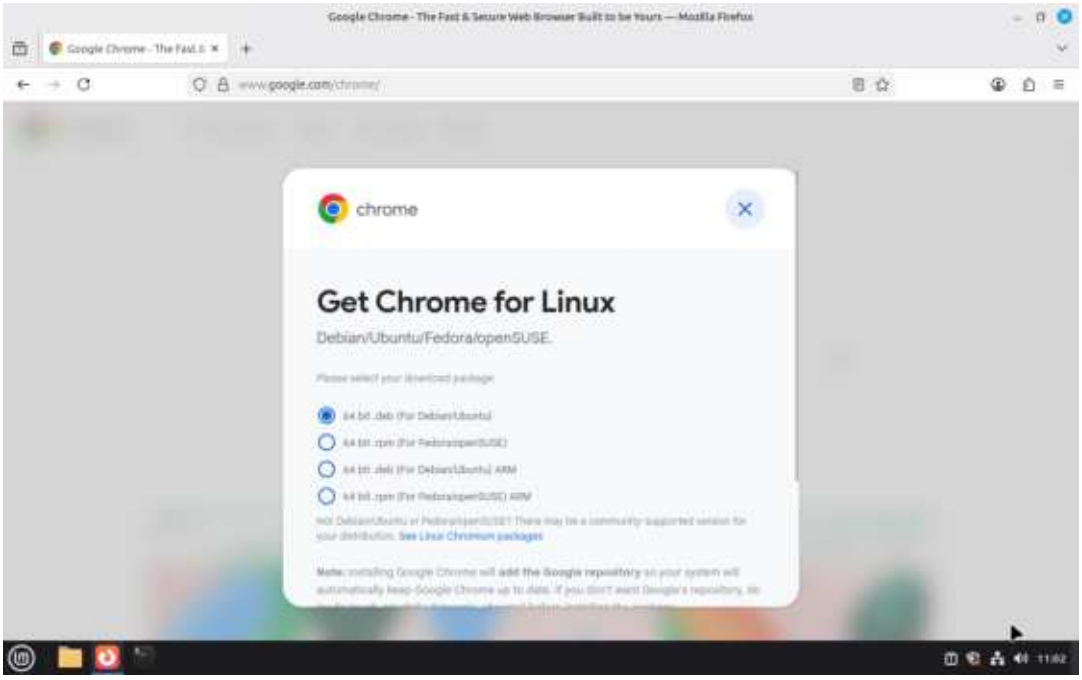


Figure 11.1. Get Chrome for Linux, with the right package already selected.

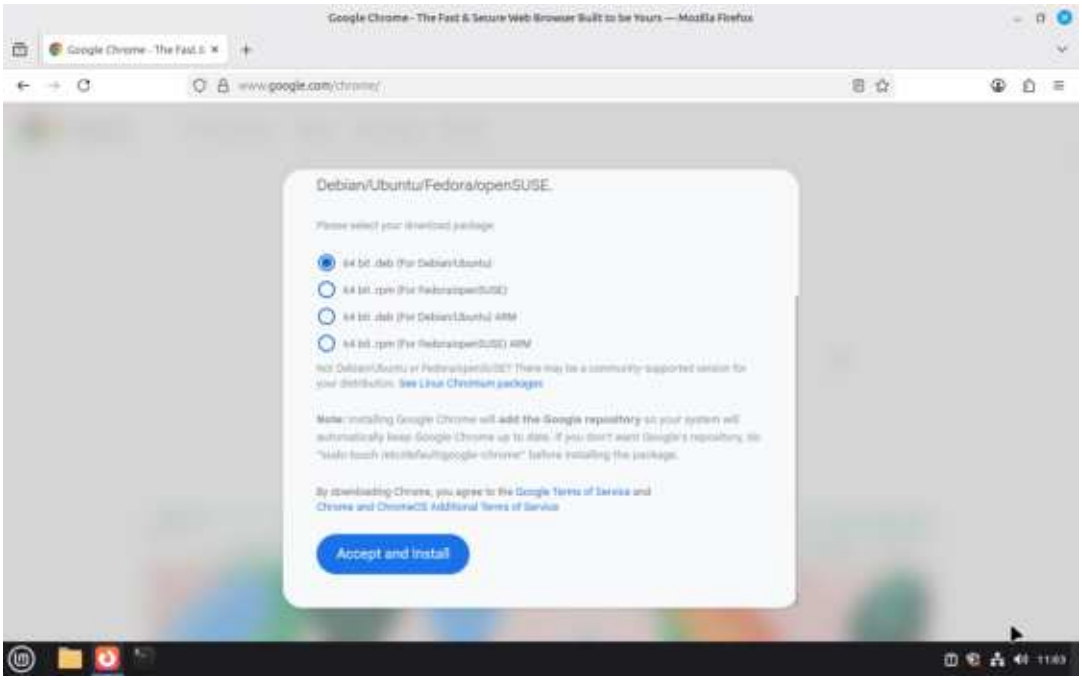


Figure 11.2. Google's terms line, and Accept and Install.

Once the download finishes, open Files, click **Downloads** on the left, and double-click the file, `google-chrome-stable_current_amd64.deb` (Figure 11.3). A .deb file is a package: a program and the parts it needs, packed in one file for installing. In the package window, click **Install Package** (Figure 11.4), and type the account password from chapter 7 in the Authentication Required window. Then the words "The installation was successful" appear in the window (Figure 11.5). Google's download page says that installing Chrome lets the computer "automatically keep Google Chrome up to date."

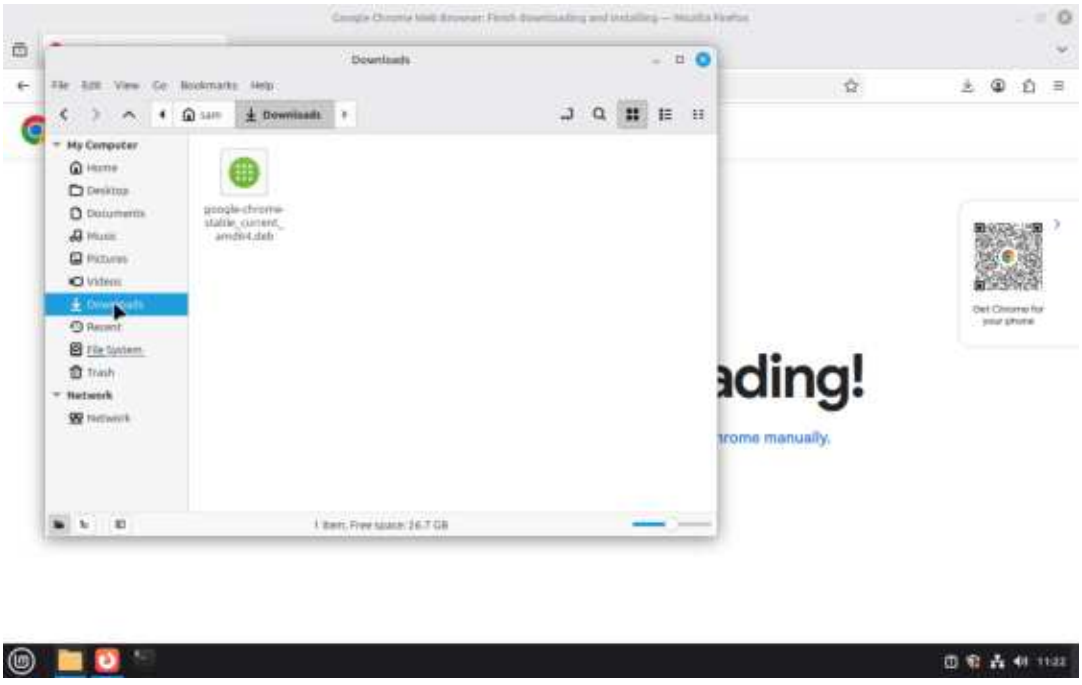


Figure 11.3. The Chrome file in Downloads.

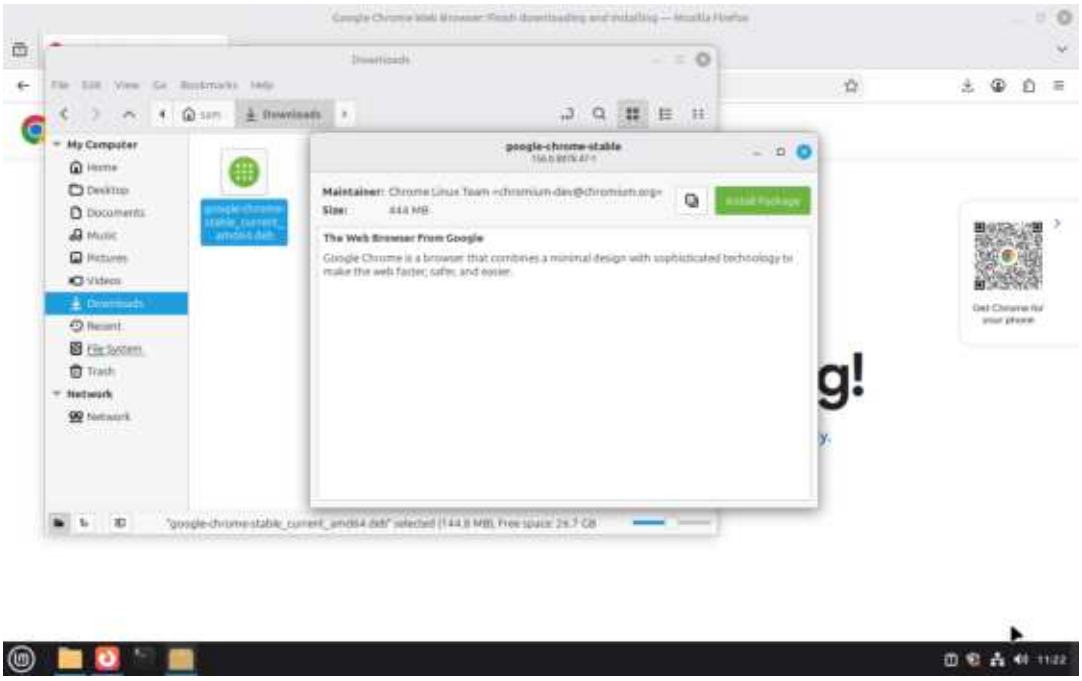


Figure 11.4. Linux Mint's package window. Click Install Package.

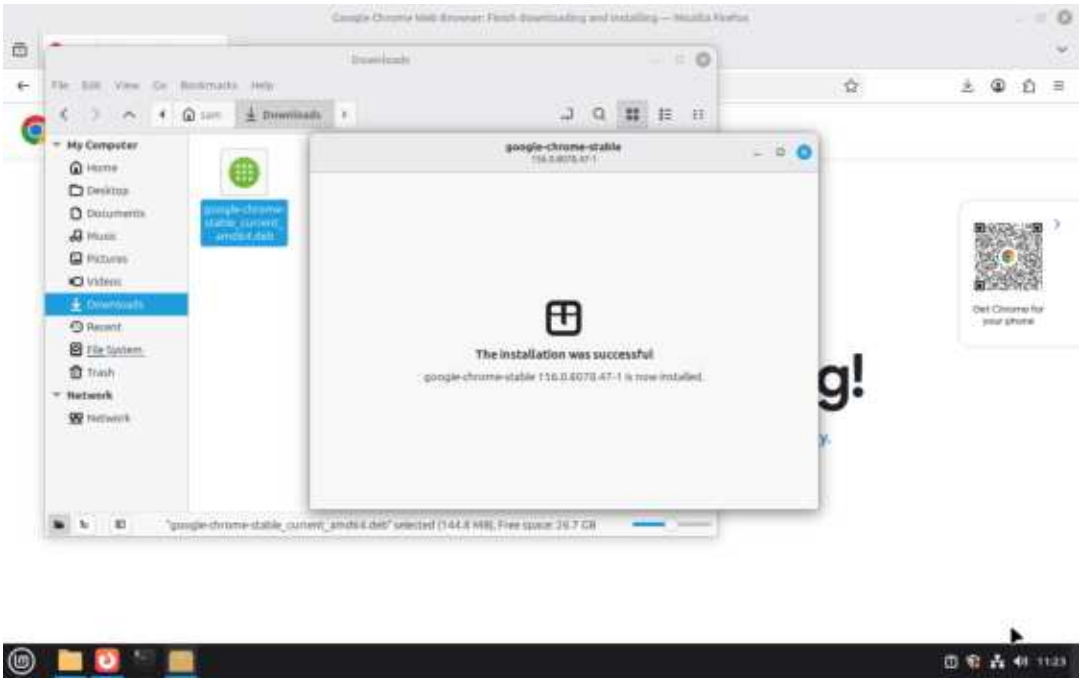


Figure 11.5. Chrome, installed.

To start Chrome, open the menu, type chrome in the search box, and click **Google Chrome**. The first time, Google's terms appear again; click **Accept**. A new tab opens with the words "Google Chrome isn't your default browser" in a bar, and a **Set as default** button (Figure 11.6). Click it to have links open in

Chrome, or leave Firefox as the default.

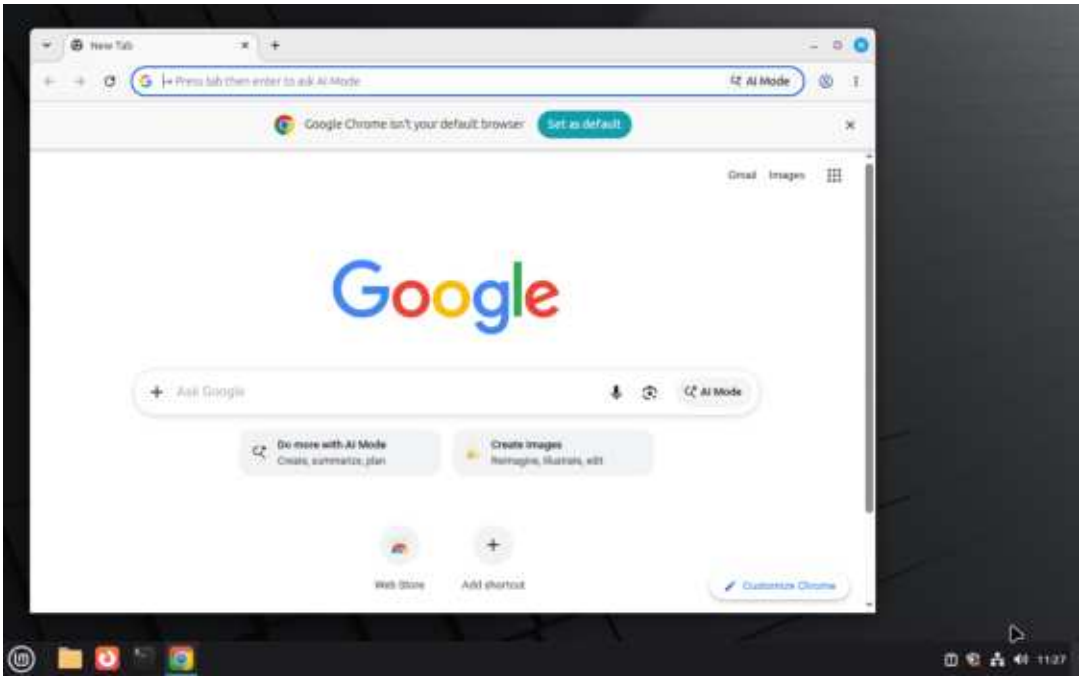


Figure 11.6. Chrome on Linux Mint, with the Set as default button.

If we used Chrome on Windows while signed in to a Google account, we get the bookmarks and passwords back by signing in here, with no export. Google's help writers say, "You can get your bookmarks, passwords, and more on all your devices." At the top right of Chrome, select **Profile**, then **Sign in to Chrome**. We didn't test this step in our lab.

Mail in the browser

Gmail, Outlook.com and other mail services work in either browser the way they did on Windows. Go to the service's page and sign in. Microsoft's management bought Hotmail in 1997 and renamed it Outlook.com in 2013, and the service still takes an @hotmail.com address.

Thunderbird

Anyone who read mail in a program on Windows, such as Outlook, can open Thunderbird, which comes installed with Linux Mint. Open the menu, type thunderbird, and click **Thunderbird Mail**.

The first time, a tab named Account Setup opens, headed "Set Up Your Existing Email Address."

1. Check the full name, type the email address and the password for the account, then click **Continue** (Figure 11.7). Thunderbird looks up the settings for that mail service. In our lab, we went as far as this screen and didn't connect a mail account, so the next steps come from Mozilla's guide.
2. "Depending on your webmail provider," Mozilla's guide says, "a new Thunderbird window may launch asking you to login via your provider's website."

3. In that window, the instruction is "Enter your login credentials and click Allow or equivalent."
4. For Gmail, the guide adds, "Click Done, sign in with your Google account if prompted."

Then "Thunderbird will now begin to retrieve your folders and messages from your provider."

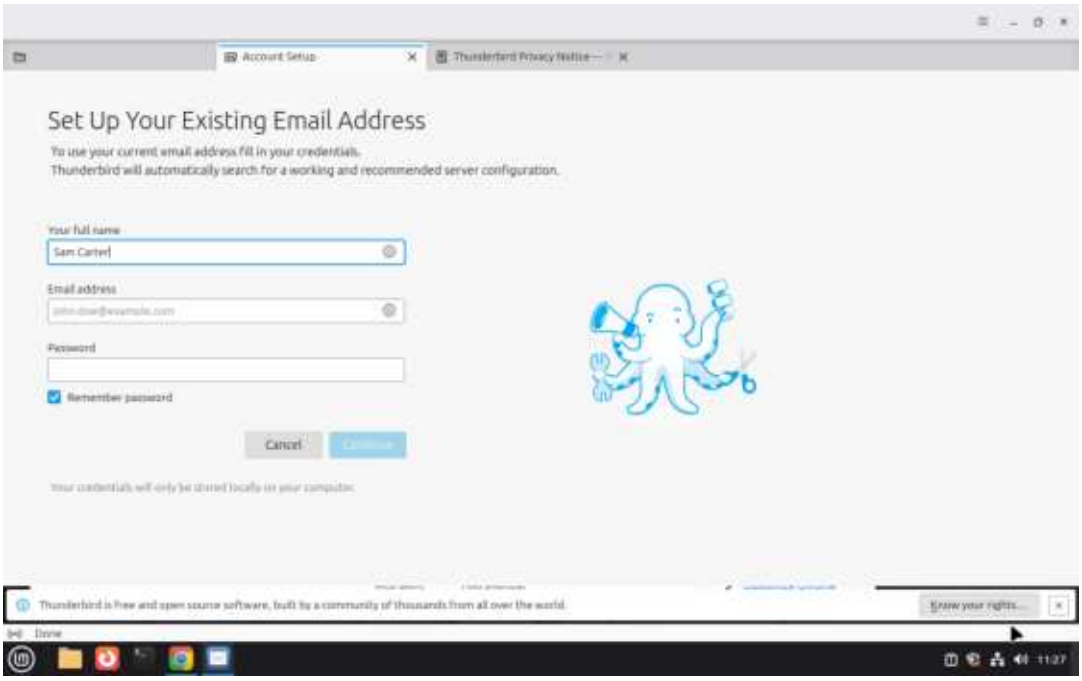


Figure 11.7. Thunderbird's first screen: Set Up Your Existing Email Address.

When we sign in, Thunderbird fetches the mail the service keeps on its server. Microsoft's help page says of Exchange, Microsoft 365 and Outlook.com accounts that "your email messages are backed up on your email server." Mozilla's guide suggests the kind of account that, in the Thunderbird team's words, lets us "store messages on the mail server and on your local machine at the same time."

Thunderbird can't read an Outlook data file, the .pst file. Mozilla's help page says that "Outlook import requires Outlook to be installed," and we erased Outlook along with Windows in chapter 7. A .pst file copied to the second drive in chapter 6 still holds that mail.

The calendar

Google Calendar and the calendar in Outlook.com open in the browser, as they did on Windows. For those who read Gmail in Thunderbird, Mozilla's guide says, "When adding a Google mail account to Thunderbird, you can easily add your existing Google calendars and address book."

Google Drive in Files

Google's help writers list no Drive desktop app for Linux and point Linux users to Drive on the web. Linux Mint 22.3 offers a second route, through Online Accounts. Open the menu and type online accounts, then click **Online Accounts**, "Connect to your data in the cloud" (Figure 11.8).

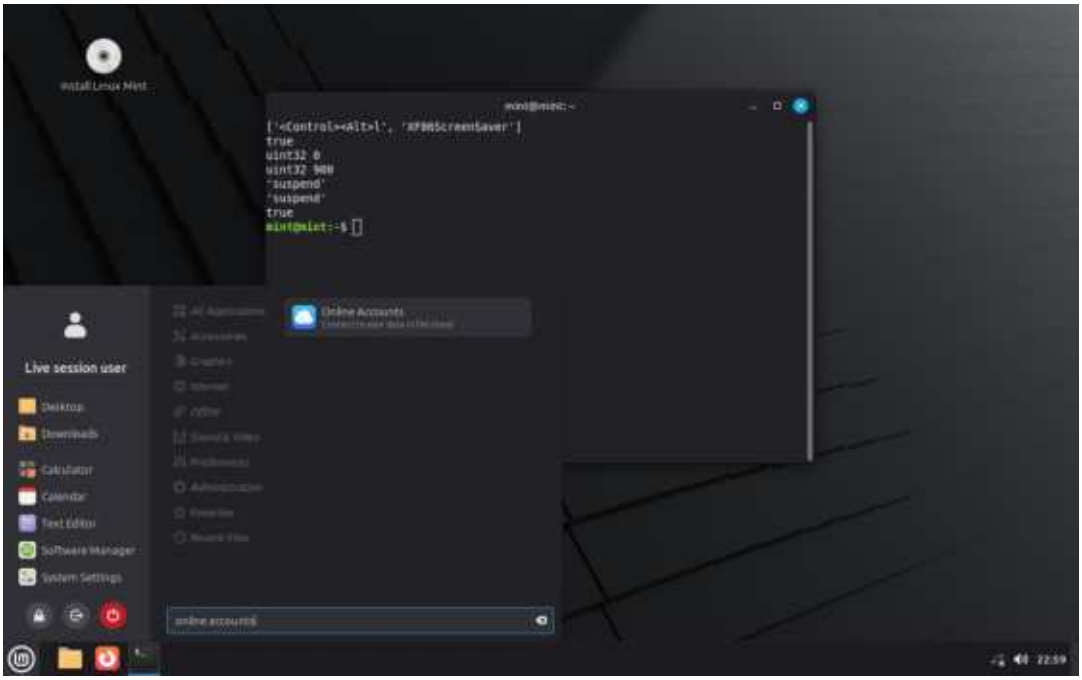


Figure 11.8. Online Accounts in the menu.

Its window lists Google first, then Microsoft, Nextcloud and others (Figure 11.9). With the computer online, click **Google**, and sign in with the account's address and password in the window that opens. Files then shows the account under Network in the left column; open it to find **My Drive** and **Shared with me** (Figure 11.10). In our lab, we didn't run the sign-in; I use Drive this way on my Linux Mint 22.3 computer.

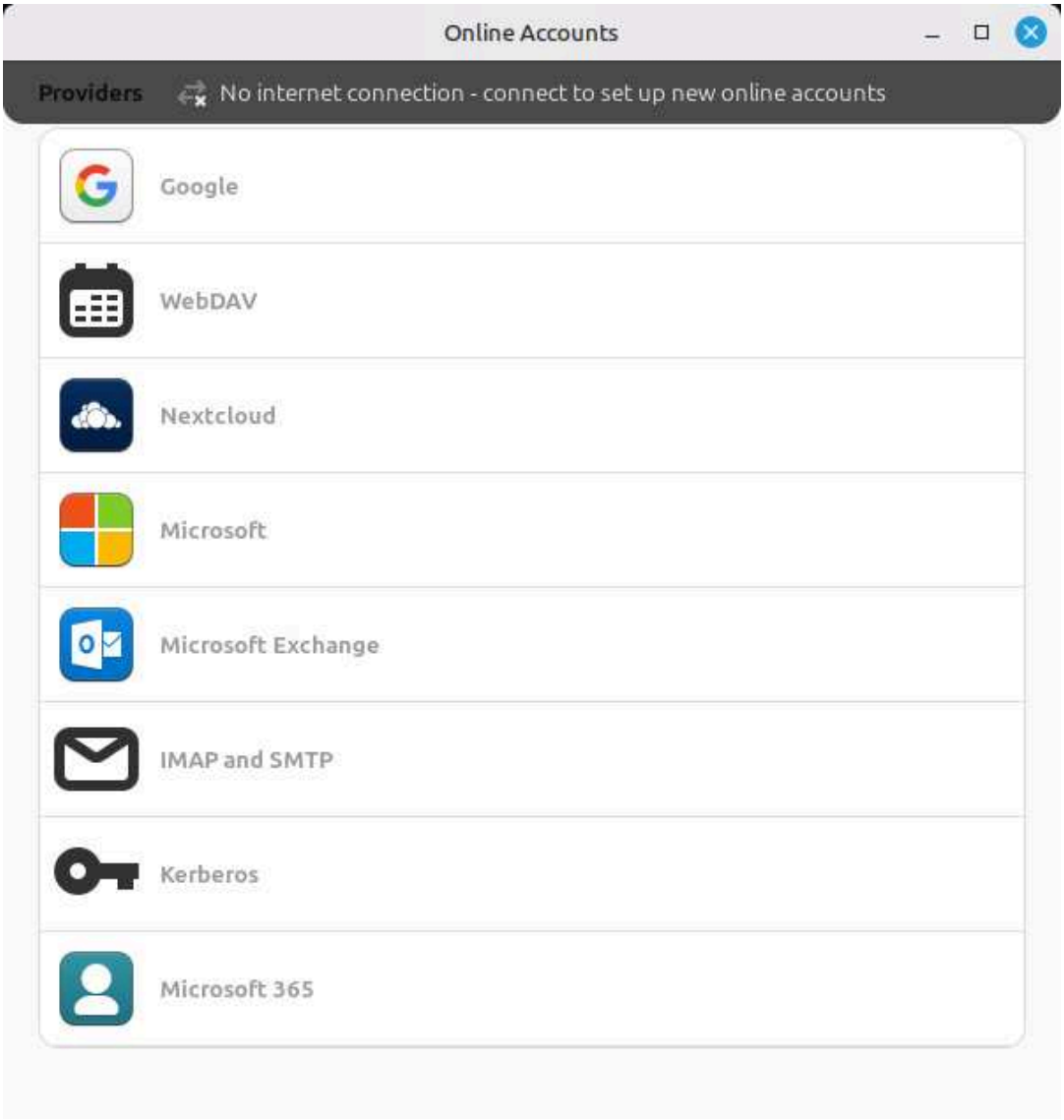


Figure 11.9. The Online Accounts window. Click Google.

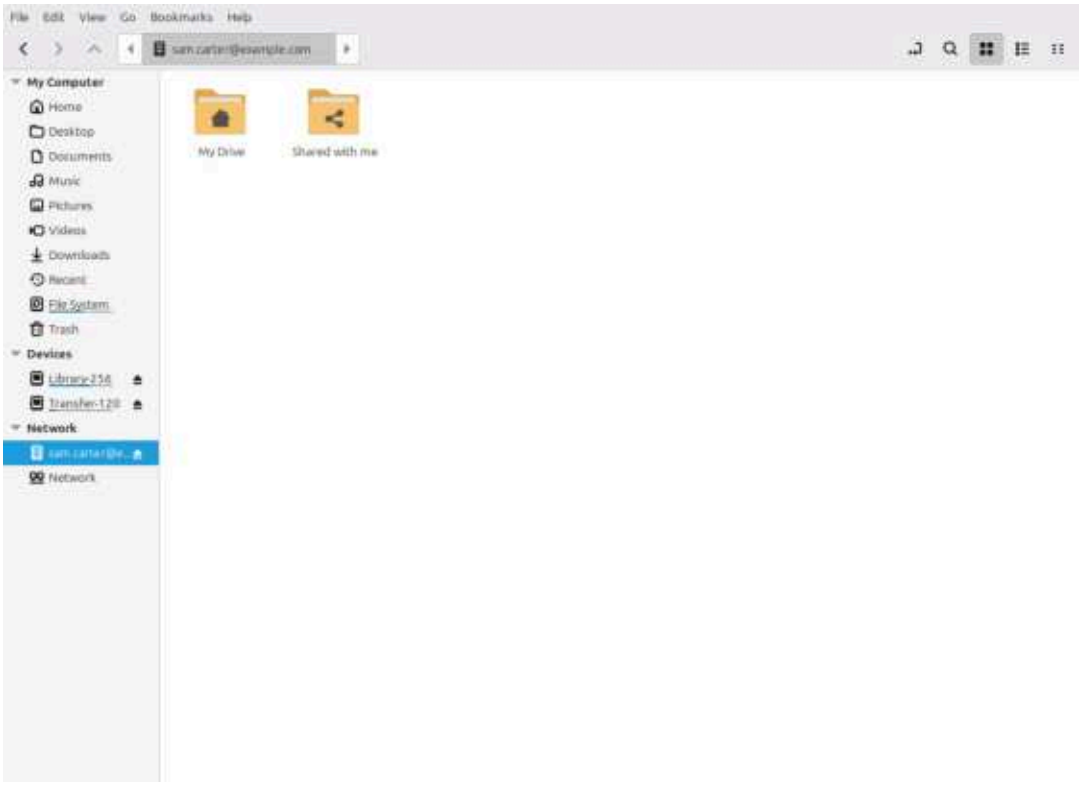


Figure 11.10. Google Drive in Files, under Network.

Try it

1. Open Firefox and go to a page from the bookmarks.
2. In Firefox, go to google.com/chrome, choose **Download Chrome**, read the terms line, and click **Accept and Install**.
3. Double-click the downloaded file in Downloads, click **Install Package**, and open Google Chrome from the menu.
4. Sign in to the mail service's page in Firefox or Chrome.
5. Open Thunderbird Mail and look at Set Up Your Existing Email Address.

12. Documents, spreadsheets and slides

Coworkers, contacts, schools and businesses send us Word, Excel and PowerPoint files. LibreOffice, which comes with Linux Mint, opens them and saves them back. We use Writer for Word files, Calc for Excel files and Impress for PowerPoint files. We opened Writer in chapter 9. Calc and Impress open with a double-click on the file.

A Word file

Open Files and find a Word file someone sent, such as `Letter-to-Dr-Lee.docx` in Downloads. Double-click it to open it in Writer (Figure 12.1).

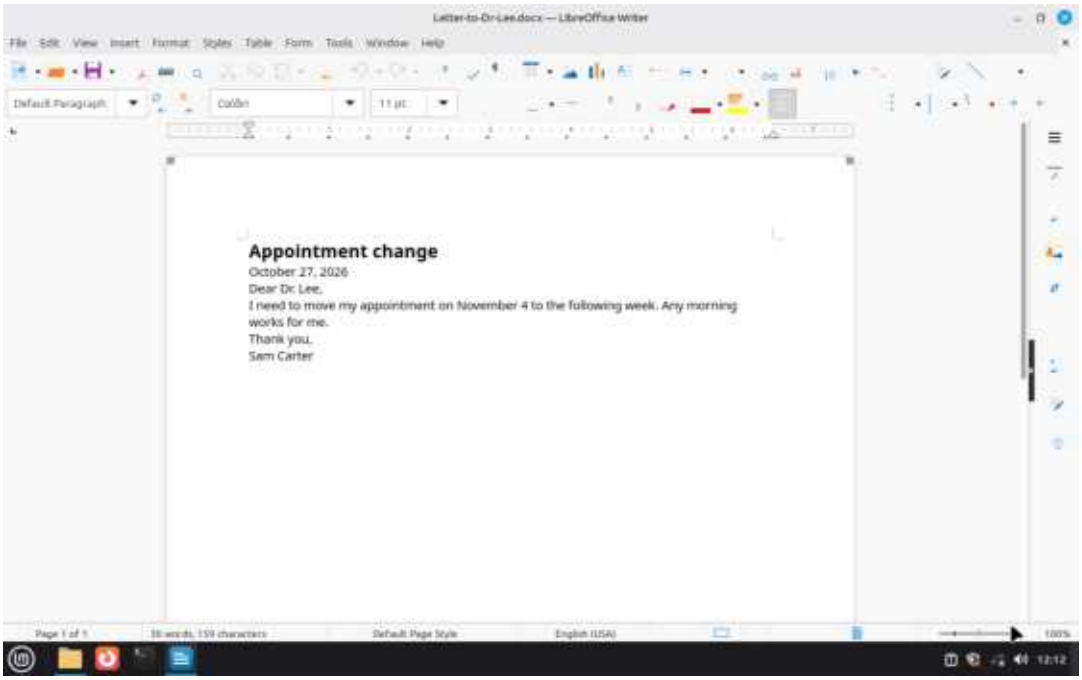


Figure 12.1. A Word letter, open in LibreOffice Writer.

Change a word, then press Ctrl+S. A window titled Confirm File Format opens. Its message begins, "This document may contain formatting or content that cannot be saved in the currently selected file format "Word 2007"." Click **Use Word 2007 Format** (Figure 12.2). Writer saves the change in the same .docx file.

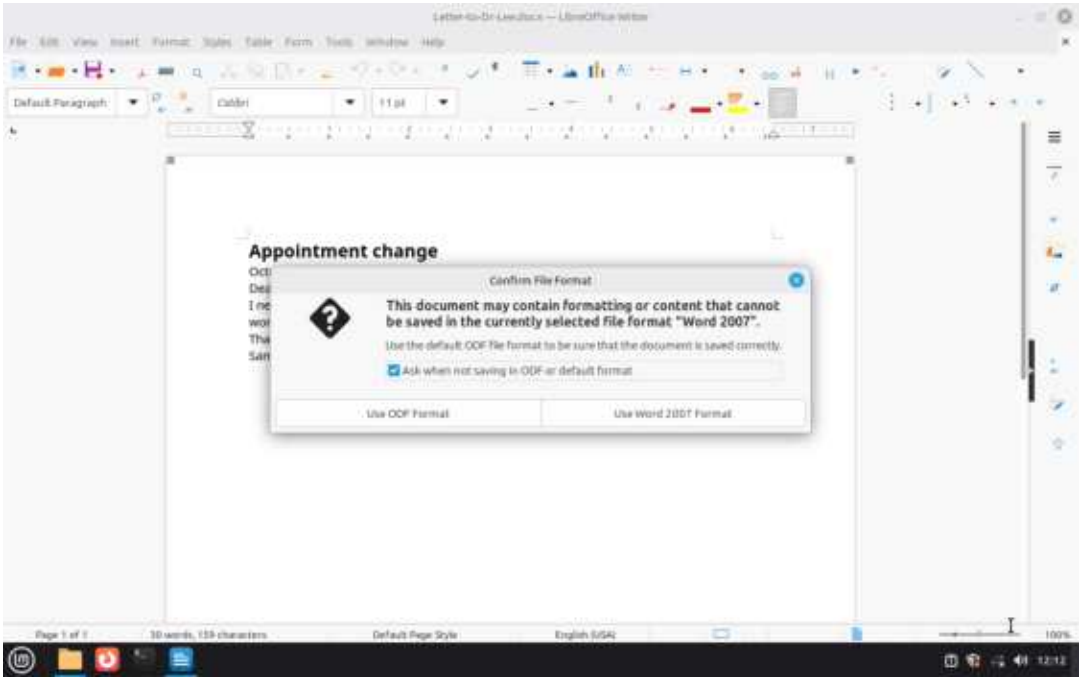


Figure 12.2. Confirm File Format. Click Use Word 2007 Format to keep the file as Word.

An Excel file

Double-click an Excel file, such as `Household-budget.xlsx`, to open it in Calc. The formulas stay in the file. Click a total to see the formula behind it in the bar above the columns, here =SUM(B2:B5) (Figure 12.3).

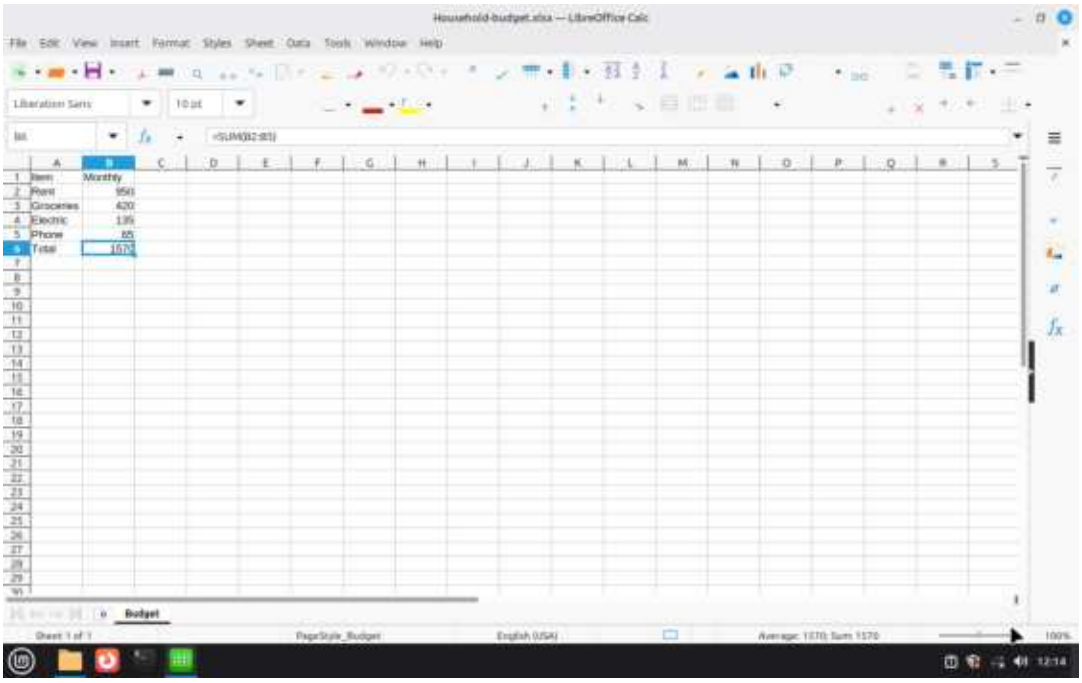


Figure 12.3. An Excel budget in Calc, with its formula.

A PowerPoint file

Double-click a PowerPoint file, such as `Garden-club.pptx`, to open it in Impress, with the slides listed down the left side (Figure 12.4).

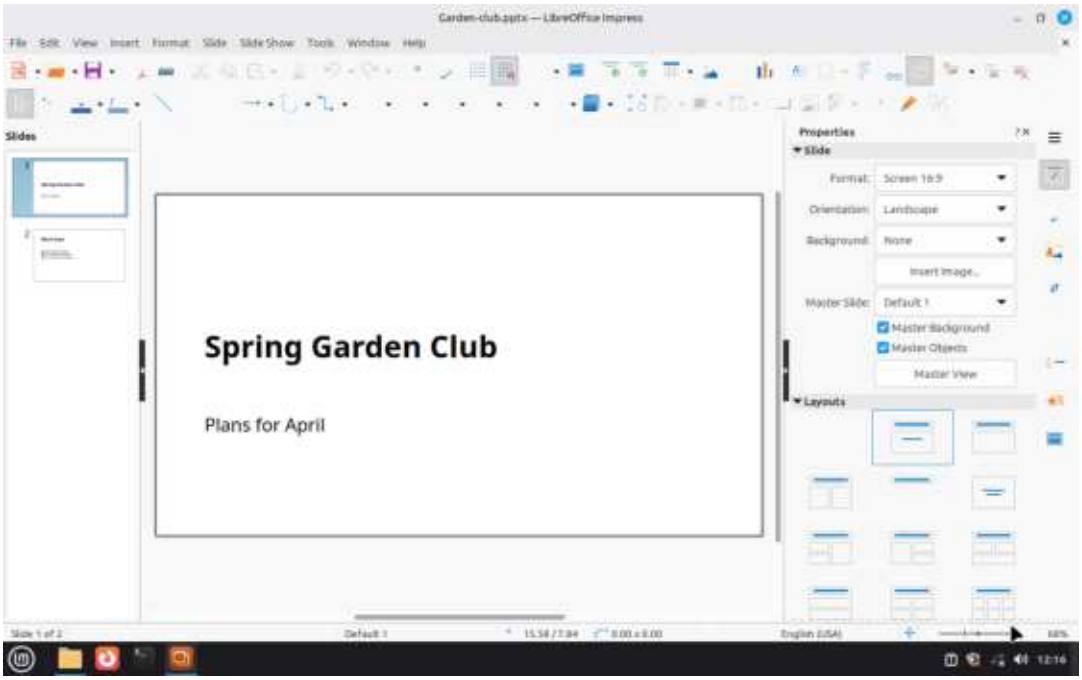


Figure 12.4. A PowerPoint file in Impress.

The LibreOffice team adds a caution about complex files: "converted files require some degree of manual reformatting. The amount of reformatting that can be expected is proportional to the complexity of the structure and formatting of the source document." In our lab, we opened the plain letter, budget and slides in this chapter, and their text, numbers and formula stayed in place.

A PDF

To send a copy that looks the same on any computer, save it as a PDF. In Writer, choose **File**, then **Export As**, then **Export Directly as PDF** (Figure 12.5). The Export window opens with the name filled in. Click **Export**. The PDF appears beside the original file.

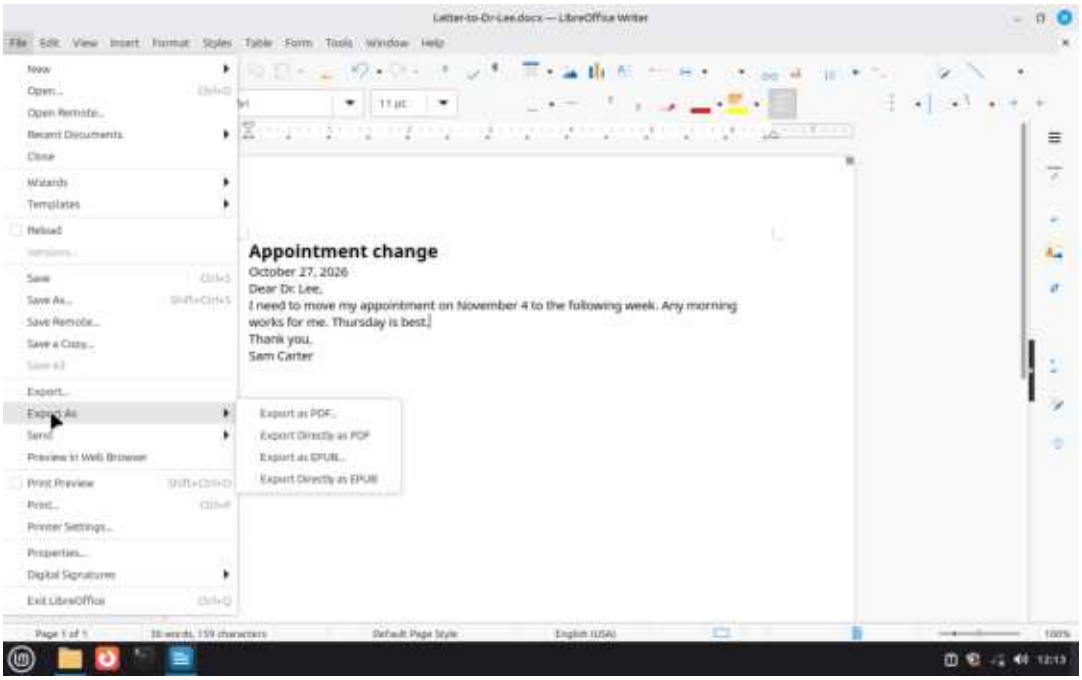


Figure 12.5. File, Export As, Export Directly as PDF.

Fonts

Many people write their Word files in Microsoft's Calibri font. When we open one, the text appears in Noto Sans, a font that comes with Linux Mint, with the name Calibri in slanted type in the font list at the top. Noto Sans runs wider than Calibri, so fewer words fit on each line than in Word.

We fix that by downloading two fonts from Software Manager, Linux Mint's store of free programs, over an Internet connection. In Software Manager, the package's maintainers describe Carlito as a "Sans-serif font metric-compatible with Calibri font," which means each letter takes the same width as in Calibri. Caladea does the same for Cambria.

Open the menu, type software manager, and open **Software Manager**. Search for carlito, choose **Fonts-crosextra-carlito**, click **Install**, and type the password in the Authentication Required window (Figure 12.6). Then search for caladea and install it the same way. Open the letter again. The line that begins "I need to move" now runs on through "works for me" (Figure 12.7). We still see Calibri in slanted type in the font list. Leave it.

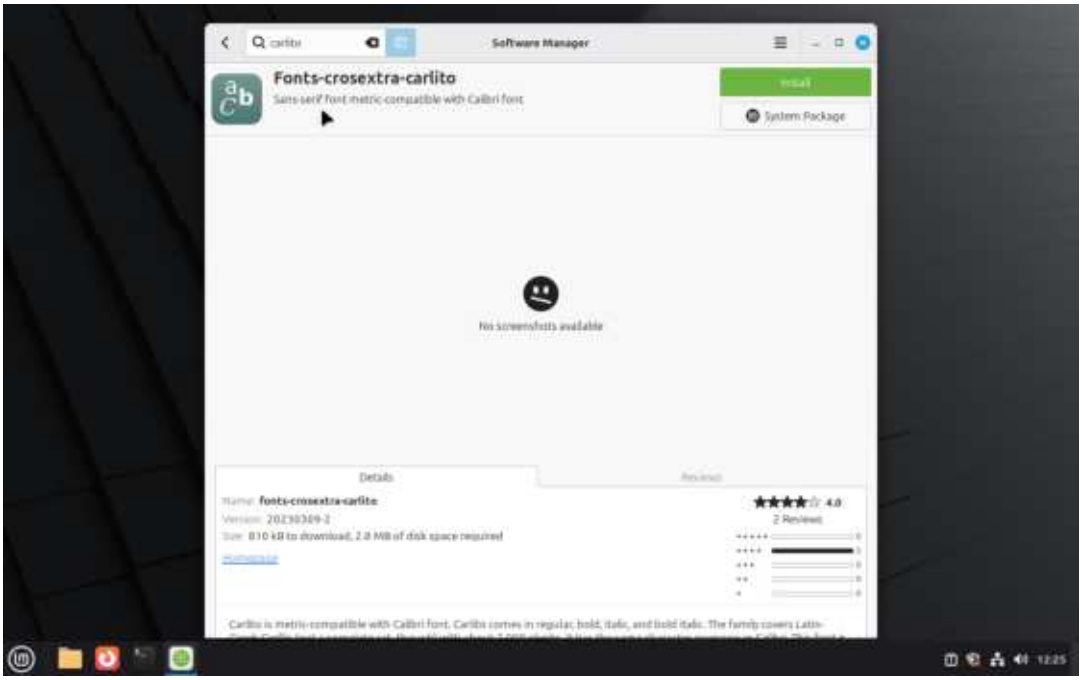


Figure 12.6. Carlito in Software Manager. Click Install.

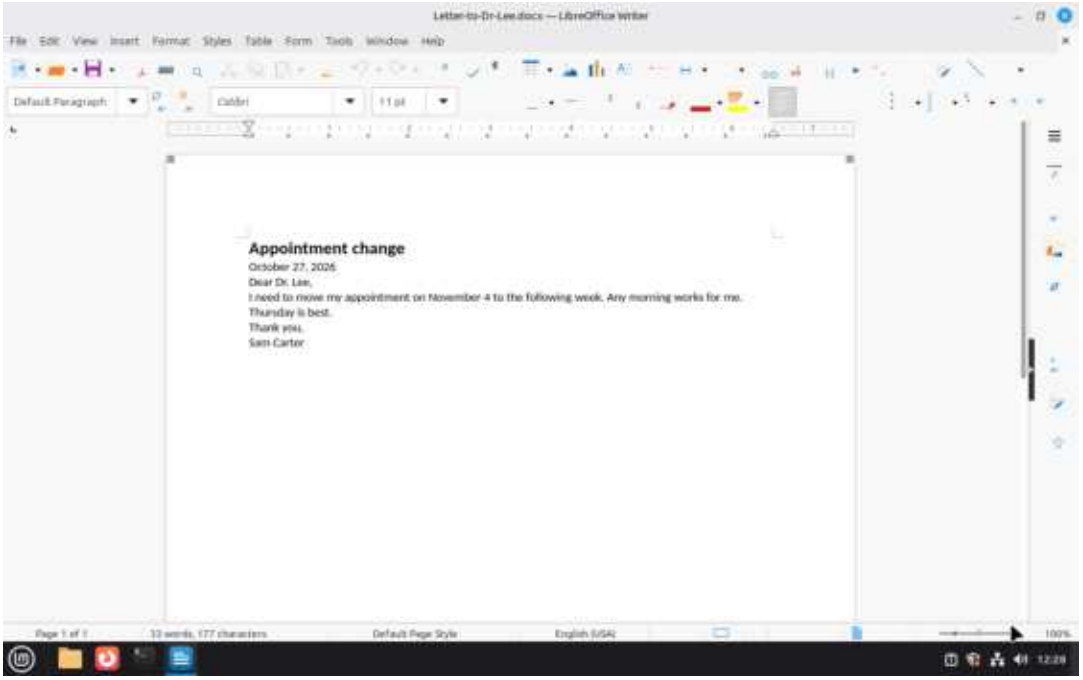


Figure 12.7. The same letter with Carlito installed. The line that begins "I need to move" now ends at "works for me."

Someone writing in a newer version of Office may use Aptos, which the type designer Steve Matteson drew for Microsoft as "a new default Office font to replace Calibri." On Linux Mint, Aptos text also appears in Noto Sans, and Carlito doesn't change that.

Try it

1. Double-click a Word file in Files, add a sentence, press Ctrl+S, and click Use Word 2007 Format.
2. Double-click an Excel file and click a total to see its formula.
3. Double-click a PowerPoint file and click through the slides at the left.
4. In Writer, choose File, Export As, Export Directly as PDF, then Export.
5. Install Carlito and Caladea from Software Manager.

13. Printer, Wi-Fi, screens and sound

For most printers, the Linux Mint team writes, "Everything just works out of the box." In this chapter, we check the network, add a printer that fails to appear, and set up a second screen, sound, drivers and a laptop's touchpad.

The network

We joined the Wi-Fi in chapters 5 and 7, from the network icon at the lower right of the panel and from the installer. Click that icon any time to join a different network or check the connection. With a network cable, the choices under the icon include **Wired**, with a switch to turn it on or off, and two more, **Network Settings** and **Network Connections**.

A printer

The Linux Mint team writes in its user guide that since Linux Mint 21, "Printers and scanners are detected and added automatically," and "No drivers are needed." For HP printers, the team adds that HP's drivers "already are installed by default in Linux Mint." To print, choose **File**, then **Print**, in the program at hand, as we did in chapter 9.

If the printer fails to appear in the Print window, add it in **Printers**. Have its address on the network ready, such as 192.168.1.25. For the lab's HP ENVY 4520, HP's user guide says that the address "can be found by pressing the Wireless icon or by printing a network configuration page." Open the menu, type printers, and open **Printers**. Click **Add** (Figure 13.1). Under **Network Printer**, choose **Find Network Printer**, type the printer's address next to Host, and click **Find**.

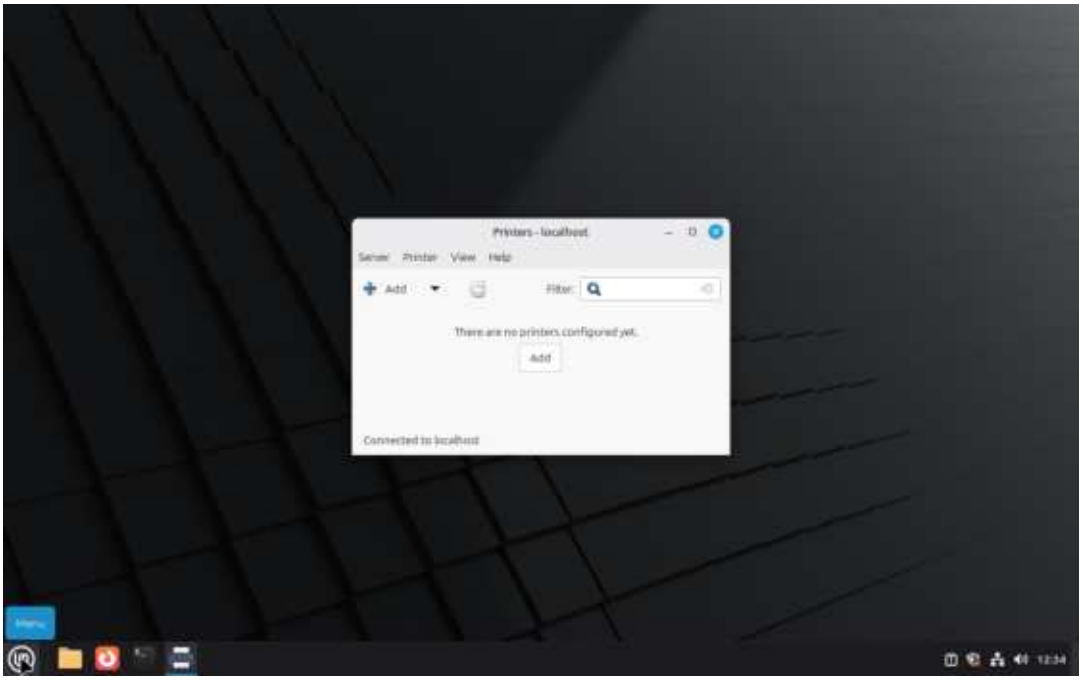


Figure 13.1. The Printers window, empty. Click Add.

In the New Printer window, find the list of the ways to reach the printer (Figure 13.2). Choose the line that begins **IPP**, and click **Forward**. In its guide, the Linux Mint team calls IPP "a standard protocol" and writes, "IPP takes priority so as long as it's installed, drivers won't be used." Describe Printer comes next, with the printer's name filled in (Figure 13.3). Click **Apply**. At the question "Would you like to print a test page?" click **Print Test Page** or **Cancel**. We then see the printer in the list with a green check (Figure 13.4).

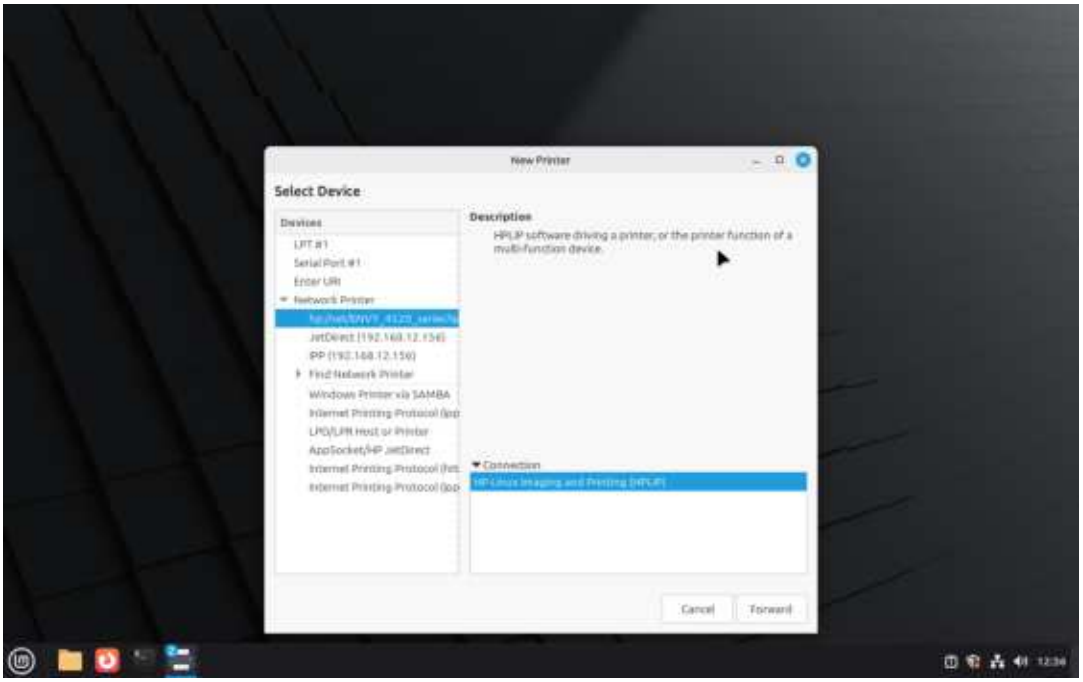


Figure 13.2. An HP ENVY 4520, found three ways. Choose IPP.



Figure 13.3. Describe Printer, with the name already filled in. Click Apply.

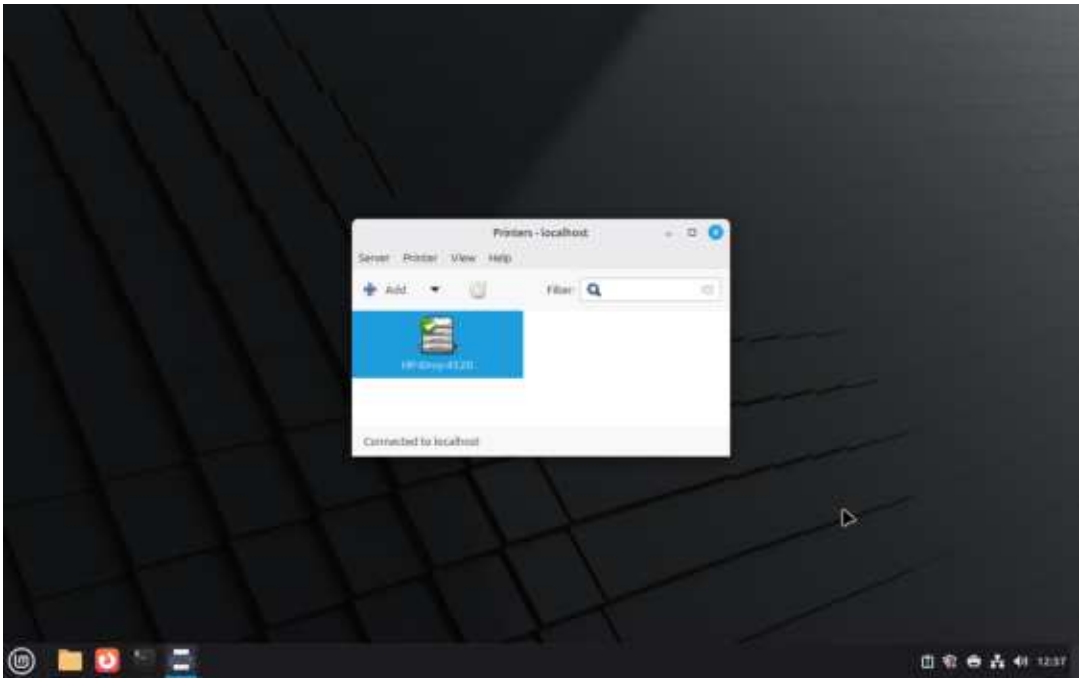


Figure 13.4. The printer, added and ready.

A second screen

With the second screen connected, open the menu, type display, and open **Display**. On the **Layout** tab, each screen appears as a numbered shape. Pick one from the list under that picture, and beside it is the **Set as Primary** button. Below are **Resolution**, **Rotation** and **User interface scale**. Click **Apply** to keep the changes.

Sound for calls

Open the menu, type sound, and open **Sound**. On the **Output** tab, pick the speakers or the headset, then click **Test sound**. Before a video call, open the **Input** tab and pick the microphone.

Drivers

The Linux Mint team writes, "One of the first things to do after installing Linux Mint is to check for available hardware drivers." Open the menu, type driver manager, and open **Driver Manager**. For an NVIDIA graphics card, the team writes in its release notes to install the driver from Driver Manager. Install it once, there. In our lab, a second NVIDIA driver installed on top of the first froze the computer, and it took several restarts to get it back.

On a laptop

Open the menu, type power, and open **Power Management** for the battery settings, including **When the lid is closed**, where we decide what happens when the lid shuts. Then type mouse in the menu, open **Mouse and Touchpad**, and choose **Touchpad**. On that tab are **Tap to click** and **Disable touchpad while typing**, and in the **Scrolling method** list, **Two-finger scrolling**.

If something failed in chapter 5

If the network or the sound failed in the live session, try Driver Manager first, as the team advises after any install. For audio, the team writes in its release notes, "If you're unable to get the sound working, you can try to go back to the older sound server called PulseAudio." For a touchpad that misbehaves, they name a second driver: "you can switch to another driver called 'synaptics'." If that doesn't fix the trouble, the team writes, "you can always try an earlier release" of Linux Mint.

Try it

1. Click the network icon and look at the connection.
2. In Writer, choose File, then Print, and see whether the printer is listed.
3. Open Display and look at the Layout tab.
4. Open Sound and click Test sound on the Output tab.
5. Open Driver Manager and look at what appears there.

14. Software Manager and Update Manager

We use two programs to keep Linux Mint supplied and current. Software Manager works like an app store: we search for a program and click Install. Update Manager works like Windows Update: we install fixes with it, and it posts a notice when it is time to restart.

Update Manager

In chapter 7, we waited to install the updates until now. Click the shield icon by the clock. From now on, an orange dot appears on it whenever new ones are ready. The first time we open Update Manager, a window titled "Welcome to the Update Manager" appears with the Linux Mint team's advice: "Linux Mint recommends to set up automated system snapshots and to apply all available updates" (Figure 14.1). Click **OK**. Then a message may appear: "A new version of the Update Manager is available." Click **Apply the Update**, and type the password in the Authentication Required window.

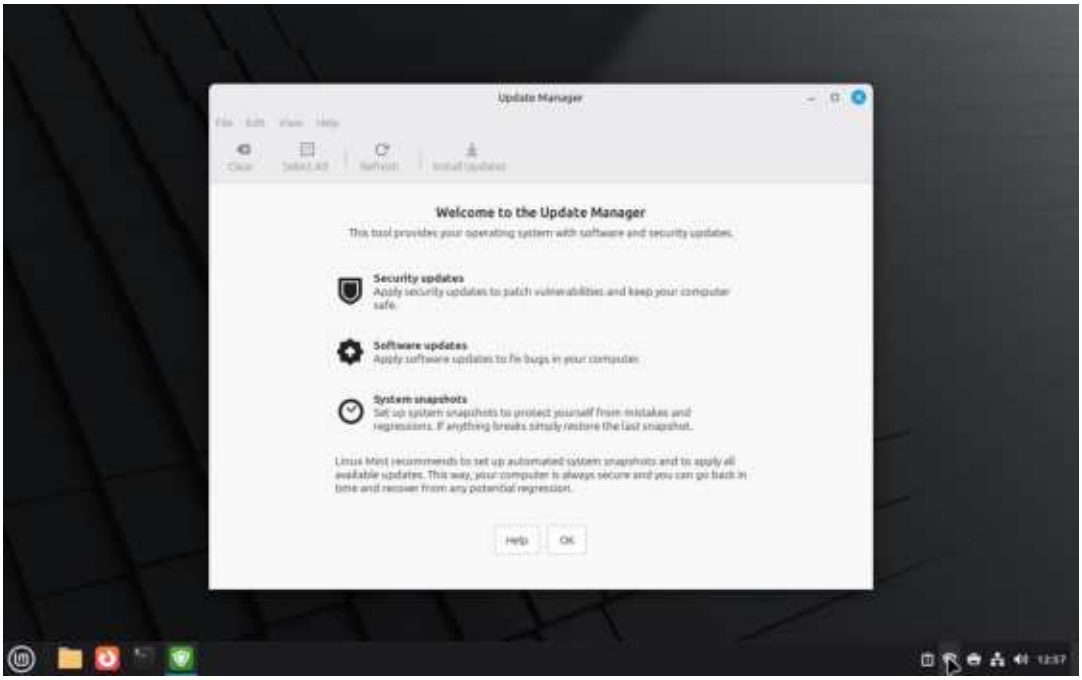


Figure 14.1. Update Manager's welcome, with the Linux Mint team's advice.

A snapshot first

The Linux Mint team writes in its installation guide, "Before you start using your operating system, set up system snapshots." With a snapshot, we can return Linux Mint to the way it was. In its user guide, the team writes that a restore brings the system back "as if it never happened." The team adds there that snapshots save Linux Mint, not our documents: "They do not include or affect your personal data."

Open the menu, type timeshift, open **Timeshift**, and type the password. In the Setup Wizard, leave RSYNC selected and click **Next**. Keep the default place and click **Next** again. That place is the disk Linux Mint runs from, so the snapshots take room there; the team writes that "the first snapshot takes a

significant amount of spaces." On the page of snapshot levels, keep **Daily** checked and click **Finish**. Then click **Create** for the first snapshot. That one took about ten minutes in our lab; later ones store only the files that changed (Figure 14.2).



Figure 14.2. Timeshift, with the first snapshot.

Install the updates

Back in Update Manager, every update in the list has a check mark (Figure 14.3). That matches the team's advice. Since Linux Mint 19, the team writes, "The Update Manager no longer promotes vigilance and selective updates," and the team relies on Timeshift instead. Click **Install Updates** and type the password.

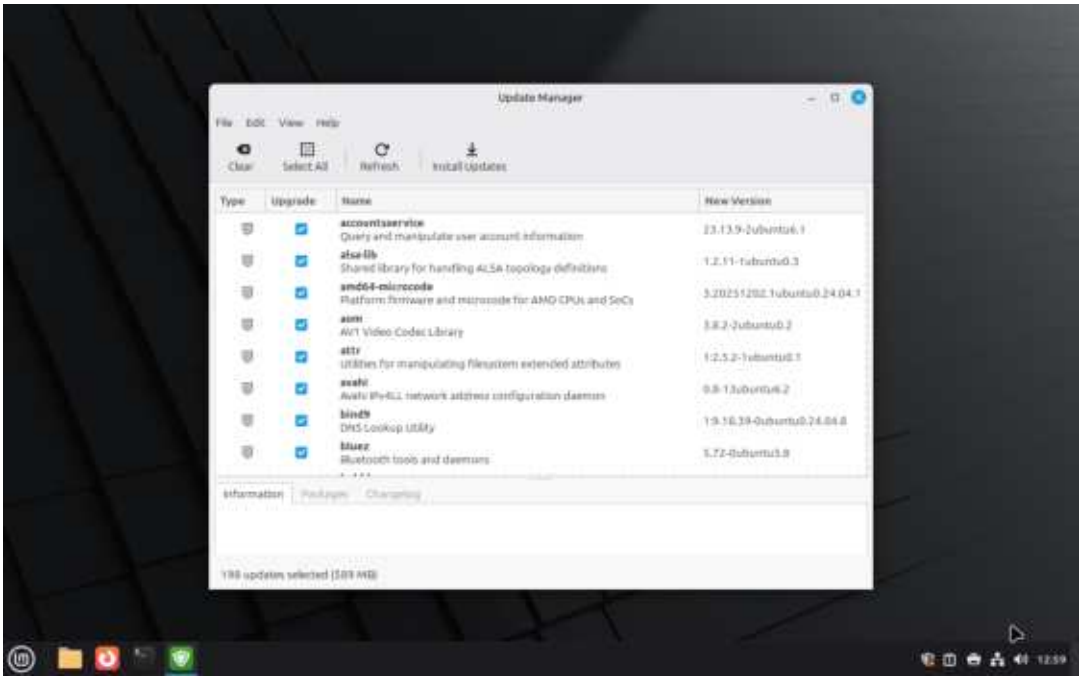


Figure 14.3. The waiting updates, all checked.

If a window appears with "Failed to download package files" and "Check your Internet connection," the connection may be fine. In our lab, it was. Update Manager's list had gone out of date while the first snapshot ran. Click **Refresh**, and Update Manager reloads the list. Then **Install Updates** works.

Restart

When the updates finish, an orange bar appears with "Restart required" and "You have installed updates that require a restart to take effect." Click **Restart**, then **Restart** again (Figure 14.4). Windows did the same; Microsoft's support page says, "To finish installing an update, your device will need to restart."

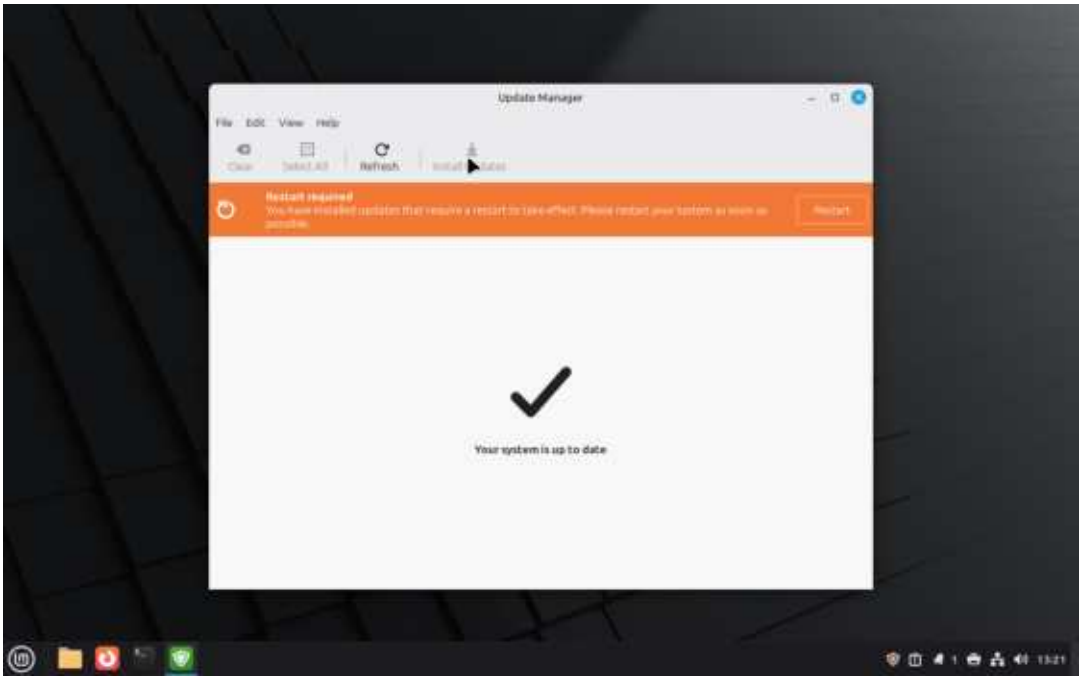


Figure 14.4. Restart required, and Your system is up to date.

Add a program

Open the menu, type software manager, and open **Software Manager**. Search for a program; here, VLC, a video player. Click **VLC**, then **Install**. When a window appears with "Additional software will be installed," click **Continue**, then type the password. Open the menu afterward, and **VLC media player** is in the list (Figure 14.5).

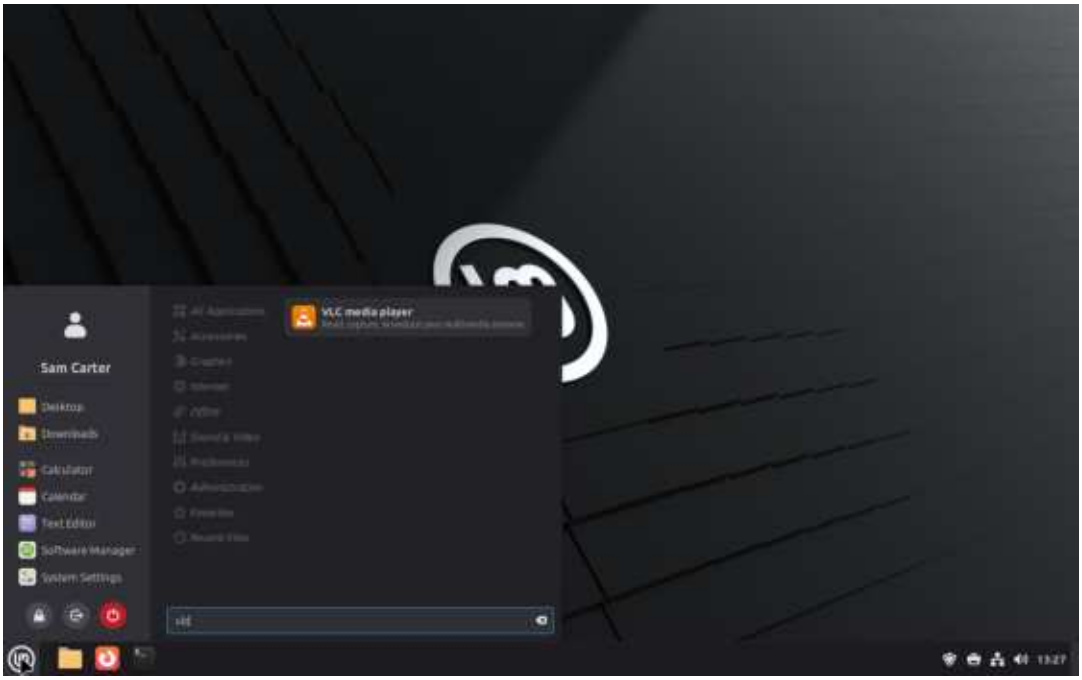


Figure 14.5. VLC media player in the menu after the install.

Remove a program

Open the menu and type the program's name, here vlc. Right-click **VLC media player** and choose **Uninstall** (Figure 14.6). In the window that lists the packages to remove, click **OK** and type the password. Search the menu for it again, and we no longer find it.

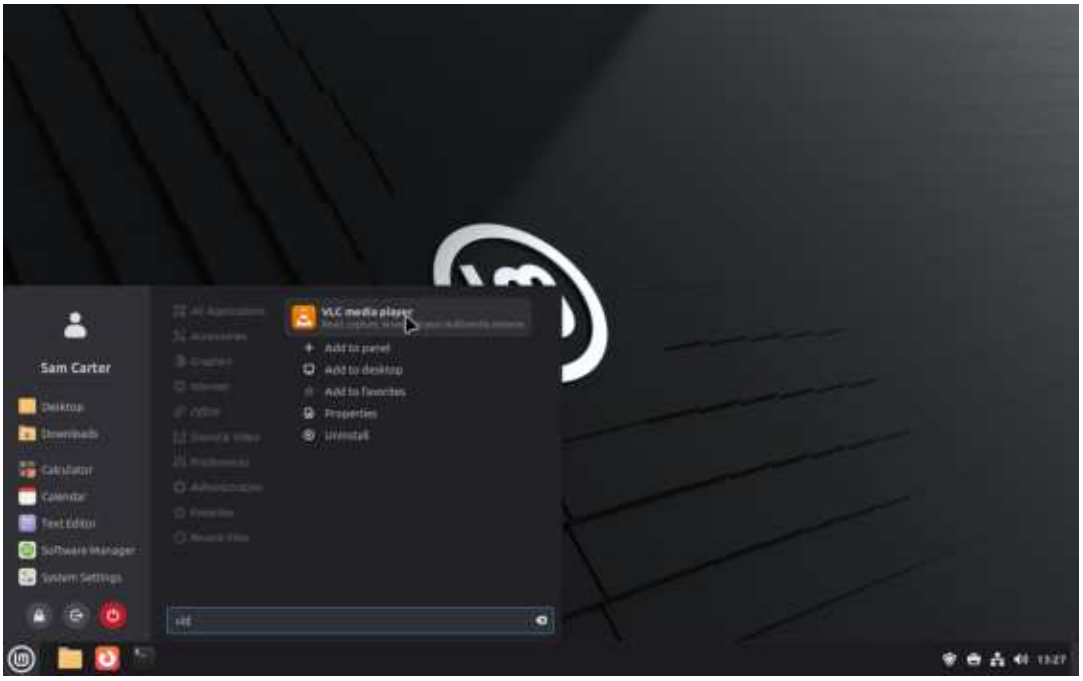


Figure 14.6. Right-click a program in the menu, then Uninstall.

Flatpaks

In Software Manager, we can also install Flatpaks, programs from a site called Flathub, which the people who run it call "The Linux App Store." In our lab's search for vlc, we saw the Flathub label under one result, GridPlayer. The Linux Mint team explains the site's term: a verified app is "an app that is published by its original developer or a third party approved by the developer." Since Linux Mint 22, the team writes, "Unverified Flatpaks are disabled by default," and "Verified Flatpaks now show their maintainer name."

Try it

1. Click the shield icon by the clock and look at the waiting updates.
2. Open the menu, type timeshift, open **Timeshift**, finish the Setup Wizard, and click **Create**.
3. Install the updates, and click Restart when the orange bar appears.
4. Install a program from Software Manager, then remove it from the menu.

15. A week on the new desktop

We have tried each part of Linux Mint, one at a time. Now give it a week of real work, and keep a list of what runs well and what doesn't. By Friday, the open lines on the list are the jobs still without a fix.

The checklist

Before Monday, write one line for each job from the Windows desktop. Then work through them on Linux Mint:

- Open a document, change it and print it (chapters 9, 12 and 13).
- Find the files from the second drive (chapter 10).
- Use the bookmarks, saved passwords and mail (chapters 10 and 11).
- Open a Word, Excel or PowerPoint file from someone else (chapter 12).
- Set the microphone and speakers for a video call (chapter 13).
- Install the week's updates after a snapshot (chapter 14).

Mark each line as it works. Write any job still out of reach at the bottom of the list.

What we may miss from Windows

Teams. Microsoft's managers retired the Teams program for Linux; the University of Waterloo's IT staff told their users it would go in "early December 2022." Teams now runs in the web browser instead, and Microsoft's documentation says that its web version "is available for both Edge and Chrome browsers running on Linux." Firefox works too, but the documentation adds, "users are required to have the OpenH264 plugin in Firefox for full support." The simplest path is Chrome, from chapter 11.

Other chat programs. An employer may choose a chat program other than Teams. Each of these five has a desktop version for Linux. Slack's help page says, "Slack for Linux is in beta," a test release, and names Ubuntu; the Linux Mint team builds Linux Mint 22.x on Ubuntu 24.04. Mattermost's install guide offers "a .deb package," the same kind of file as Chrome's in chapter 11, for "Ubuntu LTS releases 22.04 or later." Rocket.Chat's documentation lists "Linux: .AppImage, .deb, or .rpm." Element's download page gives install steps for Linux, and Zulip's help page says, "Download Zulip for Linux."

Outlook. Microsoft's help page says, "Outlook.com and Outlook on the web work in either Firefox or Chrome on Linux, but some features may not be available." Read mail in the browser, and add any missing feature to the list.

Excel macros. When we open a workbook with macros, a yellow bar appears: "Macros in this document are disabled due to the Macro Security settings." To run them, choose **Tools**, then **Options**, then **LibreOffice**, then **Security**, then **Macro Security**, select **Medium**, and open the workbook again. In the Security Warning window, click **Enable Macros** only on a workbook from someone we trust. In our lab, a test macro ran this way. The LibreOffice team writes of VBA, the programming language inside Excel workbooks, "Support for VBA is not complete, but it covers a large portion of the common usage patterns," and where a macro fails, "you may have to edit the VBA code."

Where a macro still fails, whoever wrote it can rewrite it in LibreOffice Basic, one of the scripting languages LibreOffice's help page lists. Another office suite, ONLYOFFICE, runs macros written in JavaScript; its documentation says, "ONLYOFFICE macros use the JavaScript syntax." A person who writes programs can also do a macro's job with a short program in the Python language, using pandas or openpyxl, two add-ons that read and write Excel files.

Programs with no Linux version. Adobe names Windows and macOS for Photoshop, and Autodesk lists "64-bit Microsoft® Windows® 11 and Windows 10" for AutoCAD 2026. If our work depends on a program like these, we keep a Windows computer for that one job and do the rest on Linux Mint.

In an office

Where an IT staff runs an office's computers from a central Windows server, ask the person in charge of the network before connecting Linux Mint to it. The Linux Mint team writes in its release notes, "Linux Mint 22.x is based on Ubuntu 24.04," and Ubuntu's developers publish a guide for that job, "How to set up SSSD with Active Directory."

The end of the week

Read the list. For each line still open, look in the sections above for a fix or a stand-in. Once every line has a cross through it, we do the whole week's work on the new desktop.

Where to ask

For a line still open at the end of the week, ask on the Linux Mint forums, at forums.linuxmint.com. The Welcome Screen calls them "the best place to find help," and a board named Beginner Questions takes, in its words, "Quick to answer questions about finding your way around Linux Mint as a new user." The forum's post "How To Get Help" asks us to "Search the Linux Mint forums and Google for questions before posting," and to "Use a title that is specific," not "HELP ME!!!"

The same post asks for the version of Linux Mint and the computer's details, and for 22.3 it gives the steps: "Open System Information app. Click the Copy button. Paste the copied information in your topic." Open the menu, type system information, and open **System Information** (Figure 15.1). Its window lists the version, "Linux Mint 22.3 - Cinnamon 64-bit," with the processor, memory and graphics card. Click **Copy**, and paste the lines into the question.

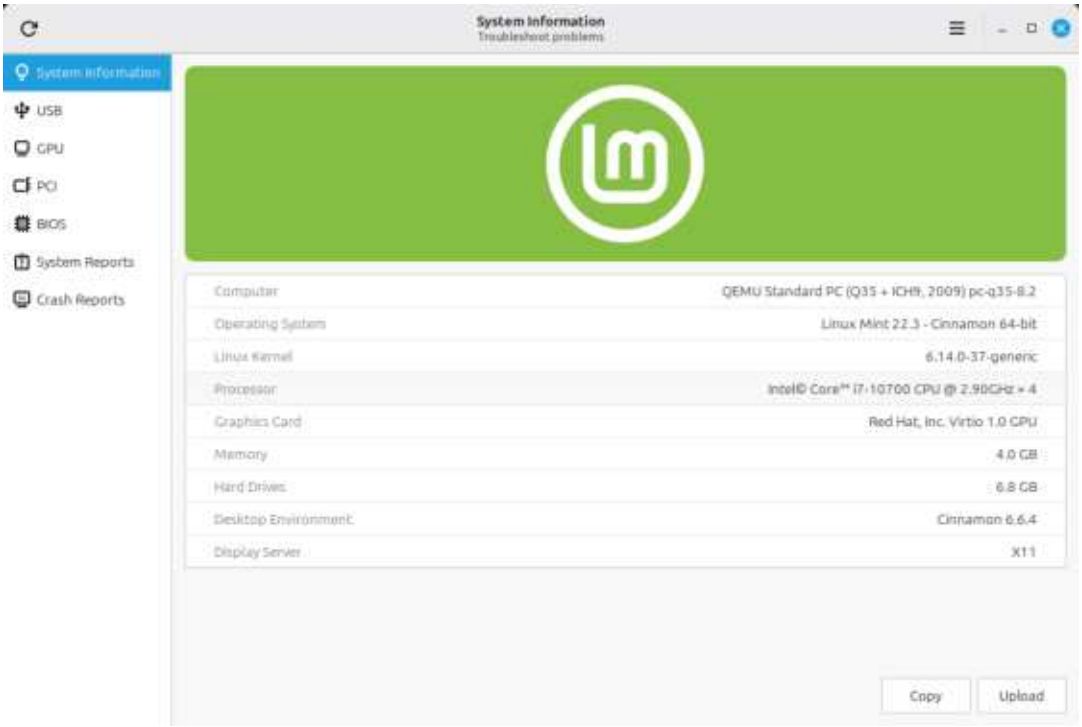


Figure 15.1. System Information, with the Copy button at the lower right.

Try it

1. Write the checklist before Monday, one line per job.
2. Work from Linux Mint for the whole week, and mark each line.
3. If a workbook from someone we trust has macros, set Macro Security to Medium, then open it and run them.

Part 4. The terminal

16. Open the terminal

So far we have clicked our way through Linux Mint. In the terminal, we ask the same computer questions by typing, and it prints the answers in text. In Part 5, we lock the computer down, and we take several of those steps by typing commands; here, we learn the commands we type there. Terminal's button is on the panel, beside Files and Firefox (chapter 8). Click it to see a window with a single line of text in it.

The prompt

The line reads `sam@samlaptop:~$`. In it are three names: the user, sam; the computer, samlaptop; and where the user is, with ~ standing for the home folder. Elsewhere, another user name and computer name appear in their place. The `$` ends the prompt.

Four questions

Type `whoami` and press Enter. Terminal prints the user name. Then ask three more questions: `hostname` gives the computer's name, `pwd` prints the current folder, and `ls` lists what it holds (Figure 16.1).

```
sam@samlaptop:~$ whoami
sam
sam@samlaptop:~$ hostname
samlaptop
sam@samlaptop:~$ pwd
/home/sam
sam@samlaptop:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
```

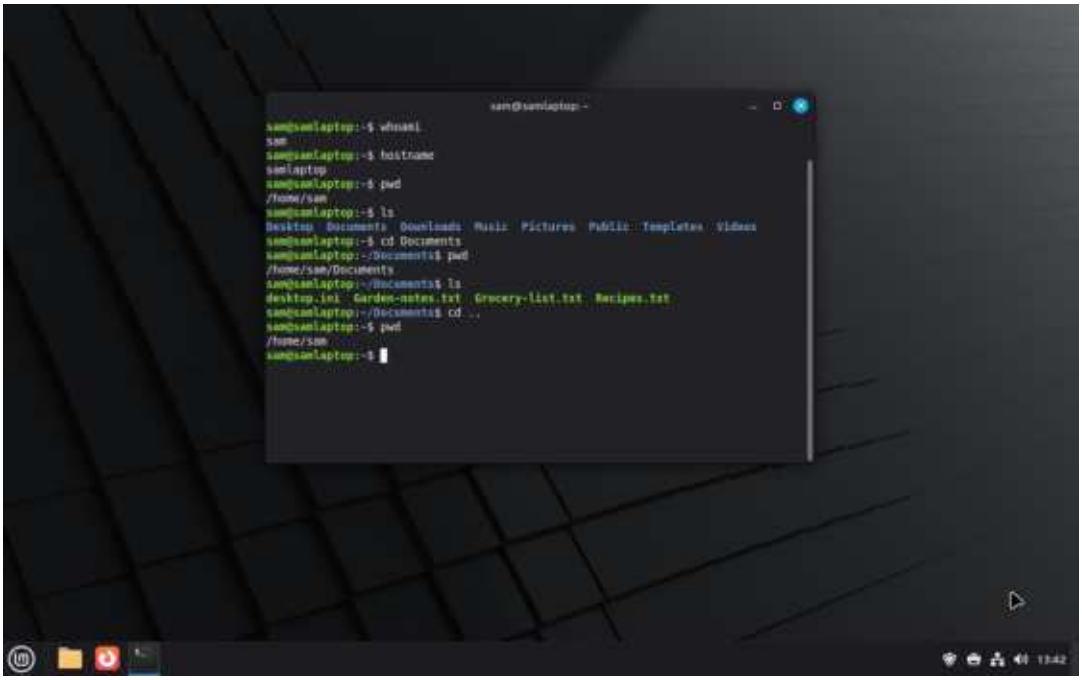


Figure 16.1. Who am I, which computer, which folder, and what's in it.

Move between folders

Type `cd Documents` and press Enter. The prompt changes to `sam@samlaptop:~/Documents$`, so we always see the current folder on that line. Now `pwd` prints `/home/sam/Documents`, and `ls` lists the files there. In our lab, the four files there came from the second drive, so Terminal showed them in green, the color it gives programs. That is also why, in chapter 10, we got the choice to run or display a text file in Files. To go back up one folder, type `cd ..` and press Enter.

The path `/home/sam/Documents` reads from the left. The first `/` is the top folder of the whole system, and every disk and folder hangs from it. Windows gives each drive a letter, such as C:, but Linux Mint uses no drive letters: the home folders are under `/home`, and a USB drive appears under `/media/sam/`, followed by the drive's name, such as `/media/sam/BACKUP` in chapter 24.

Commands on Windows

Microsoft's engineers built a program for typing commands into Windows, called Command Prompt, and in chapter 3, we used a related program, PowerShell, to check the Linux Mint download. In Command Prompt, we type the same questions in different words (Figure 16.2). There, `whoami` answered `desktop-pc\user`, `cd` alone printed the current location, and `dir` listed the folders. One difference matters right away: in the terminal, capital letters count. In our lab, `cd Documents` opened the folder, and `cd documents` answered "No such file or directory."

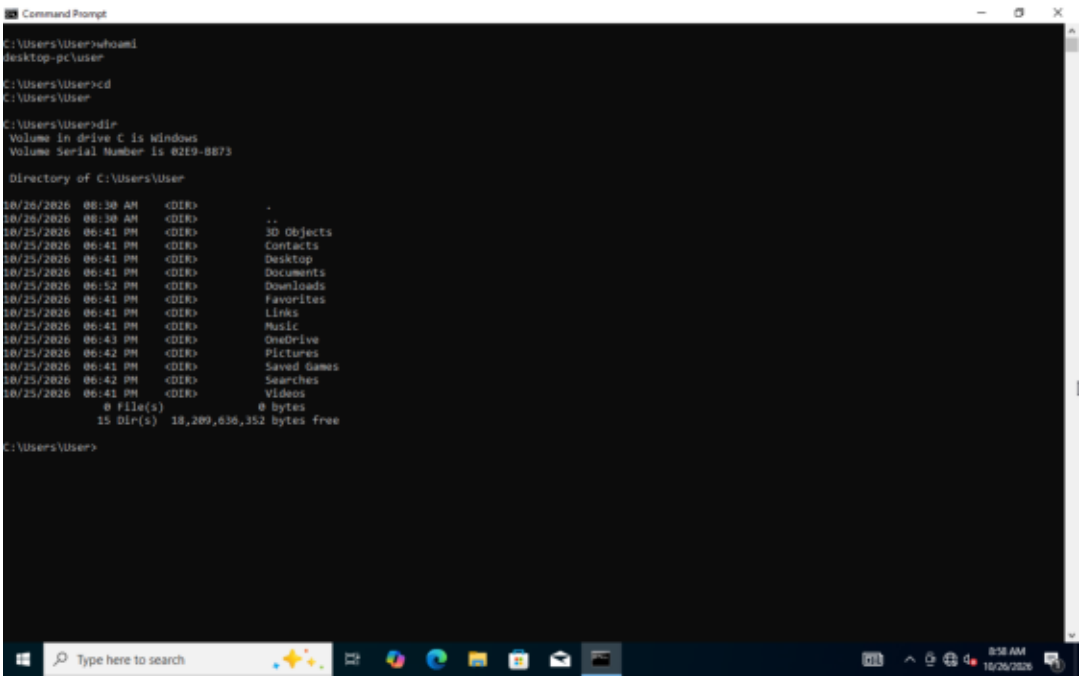


Figure 16.2. Command Prompt on Windows 10: whoami, cd and dir.

Less typing

Type part of a name, and press Tab to finish it. Type `cd Doc` and press Tab to see the line become `cd Documents/` (Figure 16.3). To run a command again, press the up arrow to bring the last one back to the line, ready for Enter.



Figure 16.3. Type cd Doc and press Tab to finish the name.

To copy text in Terminal, select it and press Ctrl+Shift+C; to paste, Ctrl+Shift+V. Terminal needs the Shift key because Ctrl+C has another job there: it stops the command that is running. When a command seems stuck, press Ctrl+C to get the prompt back. In our lab, Terminal on Linux Mint 22.3 listed those copy and paste keys in its settings.

Ask for help

To learn what a command does, add `--help`. Type `pwd --help` and press Enter. The answer opens with a usage line, `pwd: pwd [-LP]`, then the sentence "Print the name of the current working directory." and a list of the command's options (Figure 16.4). "Directory" is the terminal's word for a folder.



Figure 16.4. What pwd --help prints.

For the full manual, type `man ls`. It opens on the command's name and what it does: `ls - list directory contents`. The way out appears on the bottom line: "press h for help or q to quit" (Figure 16.5). Press q to return to the prompt.



Figure 16.5. The manual page for ls. Press q to quit.

Try it

1. Open Terminal from the panel and run `whoami`, `hostname`, `pwd` and `ls`.
2. Run `cd Documents`, then `ls`, then `cd ..` to go back.
3. Type `cd Doc`, press Tab, then Enter. Run `cd ..` to come back.
4. Run `hostname`, then press the up arrow and Enter to run it again.
5. Run `man ls`, and press q to quit.

17. Files and folders by command

In chapter 10, we copied and deleted files in Files with the mouse. At the terminal, we do the same jobs with short commands, and the up arrow from the last chapter runs any of them again. Open Terminal from the panel. The prompt ends in `~$`, so we start in the home folder.

Make a folder and a file

Type `mkdir Practice` and press Enter. The `mkdir` command makes a folder named Practice.

Now, make a file in it with nano, the editor that comes with Linux Mint. Type `nano Practice/list.txt` and press Enter. An empty page opens, with `[ New File ]` at the bottom. Type `Milk`, press Enter, and type `Bread` (Figure 17.1).



Figure 17.1. nano, with two lines typed. The keys run along the bottom; ^ means Ctrl.

The two lines at the bottom are a list of nano's keys, where ^ means the Ctrl key. Press Ctrl+O to save. At the bottom, nano prints `File Name to Write: Practice/list.txt`. Press Enter, then Ctrl+X to leave.

Read, copy and rename

Type `cat Practice/list.txt`. The `cat` command prints the file:

```
Milk
Bread
```

To copy it, type `cp Practice/list.txt Practice/list-copy.txt`, then `ls Practice`. Both files appear. To rename the copy, use `mv`, the move command:

```
mv Practice/list-copy.txt Practice/old-list.txt
```

Instead of typing `ls Practice` again, press the up arrow twice, then Enter. The list now reads `list.txt old-list.txt` (Figure 17.2).



Figure 17.2. cat, cp, mv, and ls brought back with the up arrow.

Read a long file

For a file longer than the window, use `less`. Copies of the GNU General Public License, the license family from chapter 2, come with Linux Mint in several versions. Type `less /usr/share/common-licenses/GPL-3` to open version 3. "GNU GENERAL PUBLIC LICENSE" appears on the first screen (Figure 17.3). Press Space for the next page, and q to return to the prompt.



Figure 17.3. The GNU General Public License in less.

Remove files and the folder

Type `rm Practice/old-list.txt`, then `ls Practice`. Only `list.txt` remains. Remove it with `rm Practice/list.txt`, and `ls Practice` returns an empty folder: the prompt comes straight back. In chapter 10, when we deleted a file in Files, a question came up whether to "permanently delete" it. `rm` deletes at once, and it skips the Trash. In our lab, after these commands, Linux Mint still had no Trash folder for the user.

To remove the empty folder, type `rmdir Practice` (Figure 17.4).



Figure 17.4. rm and rmdir. rm deletes without a question.

Try it

1. Make a folder named Notes with `mkdir`.
2. Make a file in it with `nano`, type two lines, save with Ctrl+O and Enter, and leave with Ctrl+X.
3. Print it with `cat`, copy it with `cp`, and rename the copy with `mv`.
4. Read the license with `less`, press Space, then q.
5. Remove both files with `rm`, then the folder with `rmdir`.

18. Users, sudo and permissions

Each account on Linux holds certain rights to each file. When we try to open or change a file without the right to do so, the message `Permission denied` appears, with the name of the program and the file, and we run `ls -l` to learn why.

Who the user is

Open Terminal. We start in the home folder, at the prompt that ends in `~$`. Type `whoami`; in our lab it printed `sam`. Now type `groups`, which lists the groups the account belongs to. Each file has one group, as we learn below. In our lab, the list read:

```
sam adm cdrom sudo dip plugdev users lpadmin sambashare
```

Look for `sudo` in the list. Members of the sudo group can run a command as root, the administrator's account. In our lab, the administrator settings file, `/etc/sudoers`, had this comment above that rule: "# Allow members of group sudo to execute any command."

Read a file's permissions

Make two small files with nano, as in chapter 17, this time right in the home folder: `note.txt` with the word Hello, and `draft.txt` with Goodbye. Then type `ls -l note.txt`. In our lab, it printed:

```
-rw-rw-r-- 1 sam sam 6 Oct 27 14:00 note.txt
```

Read the first ten characters as a dash, then three sets of three. The dash marks a plain file. The first set, `rw-`, belongs to the file's owner, the second to its group, and the third, `r--`, to everyone else. In each set, r stands for read, w for write, and x for run. Next come a count we can skip, the owner, sam, the group, also sam, and the size, 6 bytes.

A Permission denied, and the fix

Type `chmod u-w note.txt`. The `chmod` command changes permissions, and with `u-w`, we take write away from the owner. Run `ls -l note.txt` again, and the line now starts with `-r--rw-r--`. Now try to copy over the file:

```
cp draft.txt note.txt
```

The answer reads:

```
cp: cannot create regular file 'note.txt': Permission denied
```

Read it in parts. `cp:` names the program that was refused. `cannot create regular file 'note.txt'` is what it tried, and on which file. The reason is `Permission denied`, and `ls -l note.txt` shows it: we took away our right to write. Type `chmod u+w note.txt` to give it back, and run the copy again. This time it works, and `cat note.txt` prints Goodbye (Figure 18.1).



Figure 18.1. Taking write away with u-w, the Permission denied, and u+w to fix it.

The same command ends the run-or-display question from chapter 10. Files asks it because each file copied from the second drive has the x, the permission to run. For a text file in Documents, type `chmod a-x` and its name; the `a` stands for all three sets, and `-x` takes the run permission away. After that, a double-click opens the file in the Text Editor. In our lab, we ran `chmod a-x` on a scratch file, and `ls -l` showed the x gone; we didn't test the double-click afterward.

sudo

Root's folder is `/root`. Type `ls -a /root` (`-a` lists the hidden files too), and `ls` prints that it `cannot open directory '/root': Permission denied`. Now put `sudo` in front:

```
sudo ls -a /root
```

A password prompt appears: `[sudo] password for sam:`. Type the password for that account, and an asterisk marks each key (Figure 18.2). Press Enter to see the listing (Figure 18.3).



Figure 18.2. The sudo password prompt, with an asterisk for each key.



Figure 18.3. Permission denied without sudo, the listing with it.

Microsoft's engineers built something similar into Windows. Microsoft's documentation says that when someone signed in as an administrator starts a task that needs those rights, "Windows automatically prompts the user for approval," and the person at the keyboard selects Yes or No. On Linux Mint, we type a password at a prompt instead. In our lab, after one password, `sudo` ran a second command a few minutes later without another prompt.

chown

A file we copy with `sudo` belongs to root. Type:

```
sudo cp note.txt boss-note.txt
```

If the sudo prompt appears again, type the password. Then type `ls -l boss-note.txt`. `root root` appears as the owner and group.

Try `cp draft.txt boss-note.txt`, and `cp` prints `Permission denied`. To take the file over, type `sudo chown sam boss-note.txt`, with the account's user name in place of sam. The `chown` command changes a file's owner. Run `ls -l boss-note.txt`, and the owner now reads `sam` (Figure 18.4). The copy works.



Figure 18.4. A file owned by root, and sudo chown to take it over.

To clean up, type `rm note.txt draft.txt boss-note.txt`.

Try it

1. Run `groups`, and find sudo in the list.
2. Make `note.txt` and `draft.txt` with nano, then read `ls -l note.txt`.
3. Run `chmod u-w note.txt`, try the copy, read the error in parts, then run `chmod u+w note.txt`.
4. Run `ls -a /root`, then `sudo ls -a /root`.
5. Make a copy with `sudo cp`, take it over with `sudo chown`, and remove the three files with `rm`.

19. Install and update software by command

In chapter 14, we updated Linux Mint and installed a program with clicks. At the terminal, we keep the software from Linux Mint's lists current with two short commands, and we add and remove a program with two more. Each `apt` command that changes the computer starts with `sudo`, so expect the password prompt from chapter 18; in our lab, it didn't come back for a few minutes after the first time.

Update

Type `sudo apt update`. The command reads the lists of software set up on the computer (Figure 19.1). In our lab, Google's list for Chrome was on one of the lines. Google's download page, from chapter 11, says that "Installing Google Chrome will add the Google repository":

```
Hit:2 https://dl.google.com/linux/chrome-stable/deb stable InRelease
```

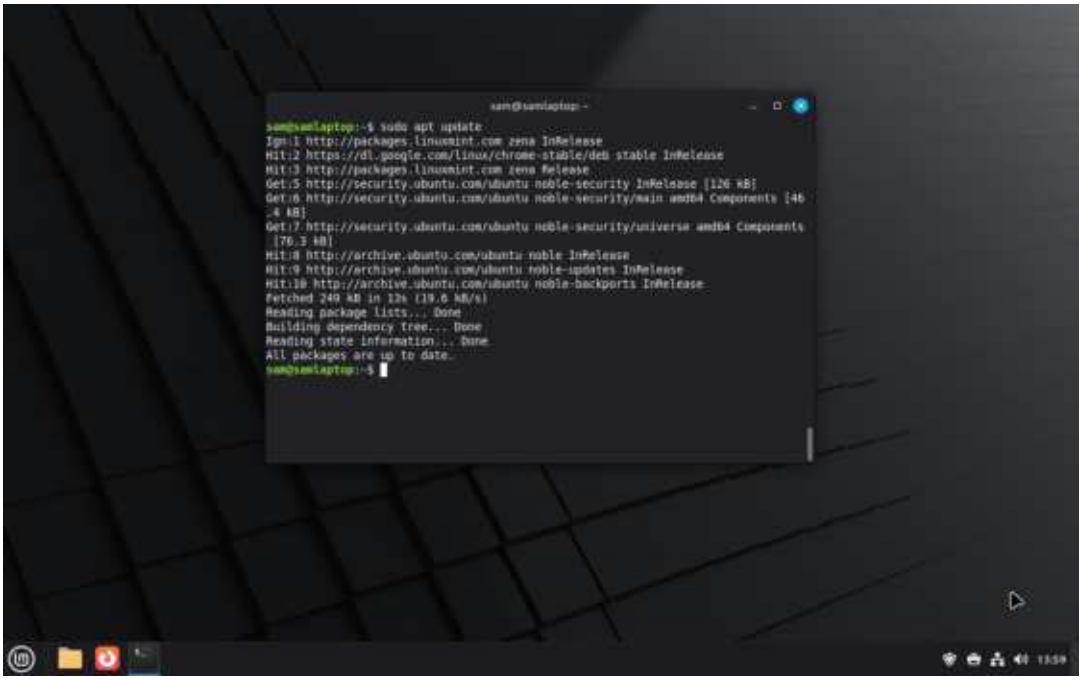


Figure 19.1. sudo apt update reads the lists, Google's for Chrome among them.

Then type `sudo apt upgrade`. It installs the new versions that `apt update` found for programs from those lists, much as **Install Updates** did in chapter 14. In our lab, Update Manager had already installed everything, so the command printed:

```
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Install a program

Type `sudo apt install tree`. The tree program draws folders as a branching picture. `apt` lists what it will add and how much space it will take, then fetches and sets it up (Figure 19.2). In our lab, no question came up first.



Figure 19.2. sudo apt install tree.

Try it: type `tree -d -L 1` in the home folder. With `-d`, tree lists folders only, and with `-L 1`, it goes one level deep. The program draws the folders and counts them.

Remove a program

Type `sudo apt remove tree`. This time a question comes before any change:

```
The following packages will be REMOVED:
  tree
0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
After this operation, 111 kB disk space will be freed.
Do you want to continue? [Y/n]
```

Type y and press Enter (Figure 19.3). Afterward, in the same Terminal window, type `tree`. The shell, the program in Terminal that reads what we type, prints `bash: /usr/bin/tree: No such file or directory`. The shell remembers where it found `tree` earlier in that window; in a new window, it says it can't find `tree` and offers the command to install it.



Figure 19.3. sudo apt remove tree, answered with y. Then tree is gone.

The Linux Mint version of apt

Type `type apt`. In our lab, the shell printed `apt is /usr/local/bin/apt`. To read the top of that file, type `head -20 /usr/local/bin/apt`. In the help text, the Linux Mint team opens with the line `This is the Linux Mint "apt" command.` and explains: "This command acts as a wrapper for the APT package manager" (Figure 19.4). In other words, the Linux Mint team's apt runs the usual APT tools for us.



Figure 19.4. The top of Linux Mint's apt command.

Try it

1. Run `sudo apt update`, then `sudo apt upgrade`.
2. Install tree, and run `tree -d -L 1`.
3. Remove tree, and answer y.
4. Run `type apt`, then `head -20 /usr/local/bin/apt`.

Part 5. A secure computer

In Part 5, we lock the computer down. Every owner works through chapters 20, 21, 22 and 24: we confirm the disk encryption and keep the security key on paper, turn on the firewall, set daily updates in Update Manager, and restore a file from a backup. Every owner also does the first two sections of chapter 25, "Check the certificates" and "Add the 2023 certificate," since LWN names "hardware where the vendor no longer supports the model" as the case that needs attention, and a computer left off the Windows 11 list may be one. The last section of chapter 22, updates without the password, stays optional, since it removes a prompt. In chapter 23 and the rest of chapter 25, an owner who wants more protection buys a hardware key, edits a system file and changes settings in the setup screen. In chapter 26, we set the result beside Windows 11, row by row.

20. Check the encryption

At the coffee shop from chapter 1, we step away from the table for a refill. The laptop stays behind, with years of our work and photos on it, and someone else could pick it up. The encryption we turned on in chapter 7 keeps those files locked. Here, we confirm the encryption is in place and keep the security key where we can find it. From here on, we call it the disk key, to keep it apart from the hardware key in chapter 23. A reader who skipped that step in chapter 7 can turn to the last section.

The sign at every start

Each time the computer starts, before the sign-in screen, a request to unlock the disk appears: "Please unlock disk," followed by the disk's name (Figure 7.7, in chapter 7). That prompt is the everyday sign of an encrypted disk. Type the disk key and press Enter, and the sign-in screen follows.

Check it by command

Open Terminal and type:

```
lsblk -o NAME,SIZE,TYPE,FSTYPE,MOUNTPOINT
```

The `lsblk` command lists the disks and their parts, and `-o` picks the columns it prints. In our lab, the lines for the disk read (Figure 20.1):

```
vda                30G disk
├─vda1            512M part  vfat        /boot/efi
├─vda2            1.7G part  ext4        /boot
└─vda3           27.8G part  crypto_LUKS
  └─vda3_crypt   27.8G crypt LVM2_member
    ├─vgmint-root 24.8G lvm   ext4        /
    └─vgmint-swap_1  3G lvm   swap        [SWAP]
```

Look for `crypto_LUKS` beside the largest part of the disk, and a line ending in `_crypt` under it. LUKS is the encryption. The line marked `/` holds Linux Mint, and it appears under the `_crypt` line, inside the lock. On another computer, `lsblk` may print another name than `vda`; on our lab's Inspiron 5567, the same layout appeared under `sda`. That laptop runs the Xfce edition of Linux Mint 22.3, which the Linux Mint team calls "a lightweight desktop environment" with fewer features than Cinnamon; the commands in this part are the same in Cinnamon.



Figure 20.1. lsblk on an encrypted Linux Mint: crypto_LUKS, and Linux Mint inside it.

Two small parts, `/boot/efi` and `/boot`, have no `crypto_LUKS` beside them, so those two stay unencrypted.

Keep the disk key

In chapter 7, the team warned on the installer's screen that a lost key means lost files. Keep the disk key written down, away from the computer, with other papers that matter. The copies we made in chapter 6 aren't encrypted, so store that second drive somewhere safe too. In chapter 24, we build a backup and restore from it on purpose.

A disk without encryption

If `lsblk` shows no `crypto_LUKS`, the disk isn't encrypted. In chapter 7, we turned on encryption in the installer under Advanced features, and we reach that choice again only with a new install. Copy the home folders to a second drive with Files, as in chapter 10 but in the other direction, and export the Firefox bookmarks first. Then install again with encryption (chapter 7), import the files into Linux Mint (chapter 10), and add back any programs installed before (chapter 14).

Lock the screen

The disk encryption protects a computer that is switched off. At the coffee shop from chapter 1, the laptop stays on while we step away for a refill. Press Ctrl+Alt+L to lock the screen, and type the password to come back.

Linux Mint also locks the screen after a set time. Open the menu, type screensaver, and open **Screensaver** (Figure 20.2). On the Settings tab, the screensaver starts after 15 minutes, and both switches under Lock settings are on: **Lock the computer when put to sleep** and **Lock the computer after the screensaver starts**, with **Lock immediately** beside Delay before locking. Closing a laptop's lid puts it to sleep, so the closed lid locks it too. For a laptop carried out of the house, pick a shorter delay, such as 5 minutes. In our lab, we read these settings in the live session of Linux Mint 22.3, where they start this way.

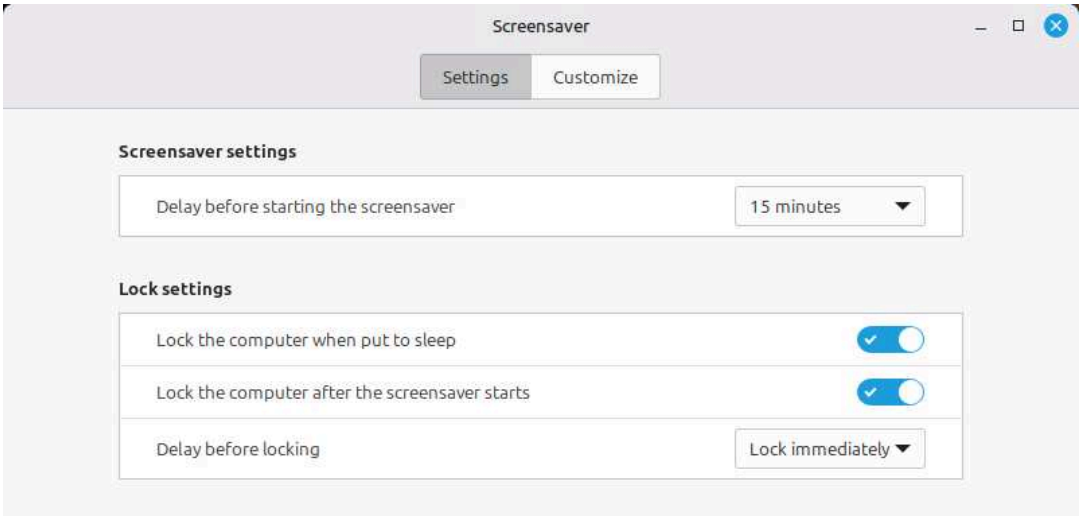


Figure 20.2. The Screensaver window, with both lock switches on.

Try it

1. Restart the computer, and type the disk key at the unlock prompt.
2. Run the `lsblk` command above, and find `crypto_LUKS`.
3. Find the written disk key without signing in to the computer.

21. The firewall on the desktop

In chapter 20, we made sure the disk stays locked if we lose the computer or someone steals it. Now we guard the computer's link to the network, and use a firewall to choose which connections may reach it. The Linux Mint developers left the firewall off in the copy we installed.

Check it first

Open Terminal and type:

```
sudo ufw status verbose
```

`ufw` is the firewall program that comes with Linux Mint. After the password, ufw printed `Status: inactive` on our lab's computer (Figure 21.1).



Figure 21.1. The firewall, off.

Turn it on

Open the menu, type firewall, and open **Firewall Configuration** (Figure 21.2). A window opens with the words "Authentication is required to run the Firewall Configuration." This time the password goes there, not into the terminal. Type it and click **Authenticate**.

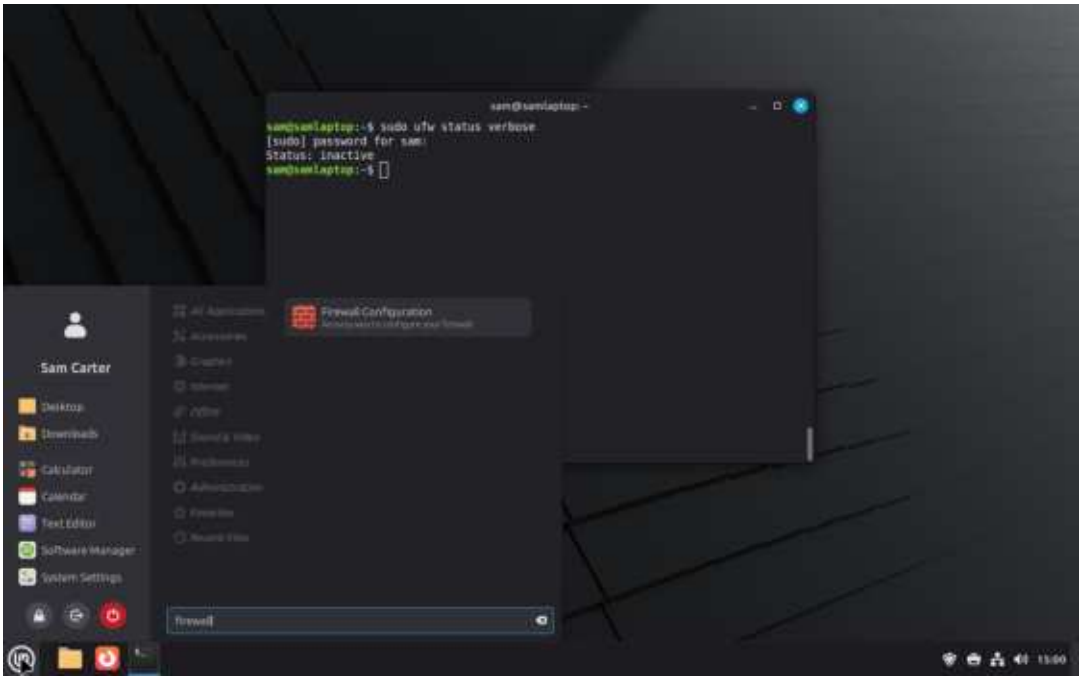


Figure 21.2. The menu search for Firewall Configuration.

In the Firewall window are a Profile, a Status switch, and two settings, Incoming and Outgoing (Figure 21.3). On its Getting started page is the advice to follow: "If you are a normal user, you will be safe with this setting (Status=On, Incoming=Deny, Outgoing=Allow)."

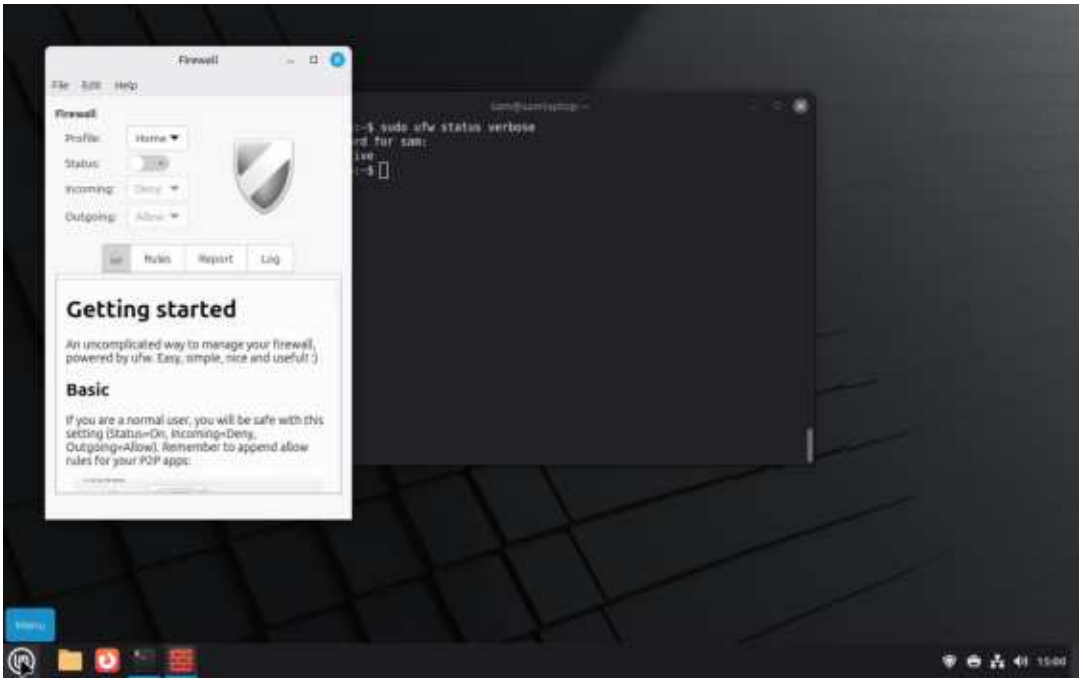


Figure 21.3. The Firewall window, with Status off.

Leave Incoming at **Deny** and Outgoing at **Allow**, and click the **Status** switch. It turns blue, and the words "Firewall enabled" appear at the bottom of the window (Figure 21.4).

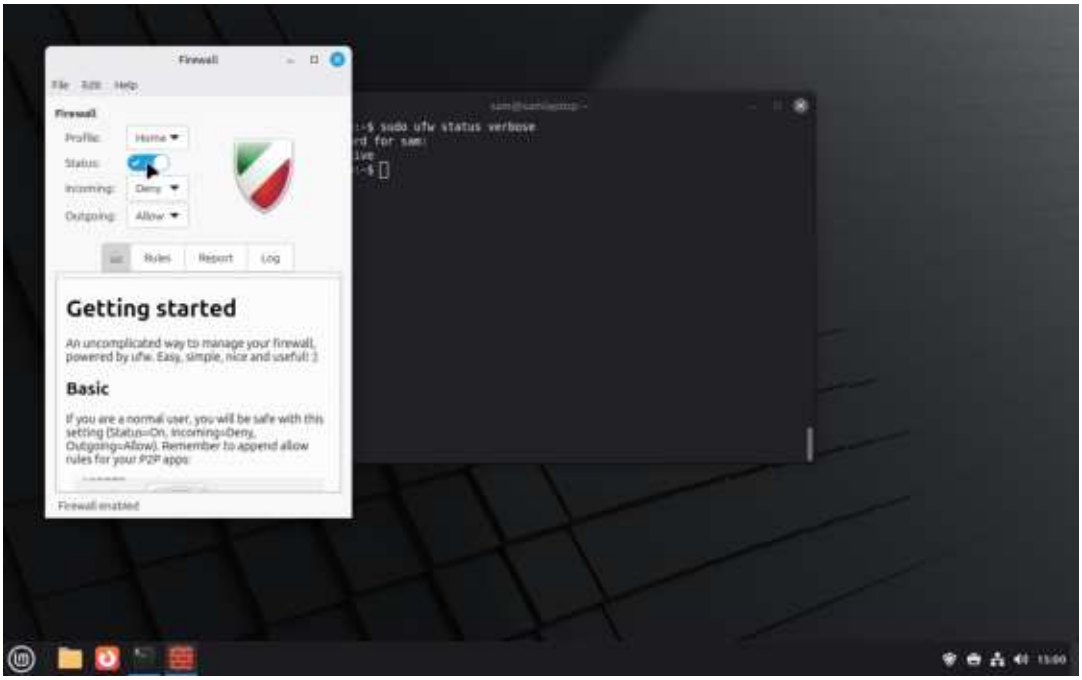


Figure 21.4. Status on. Firewall enabled.

Check it again

Back in Terminal, run `sudo ufw status verbose` once more (Figure 21.5). In our lab, it printed:

```
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), disabled (routed)
New profiles: skip
```

The third line holds the two settings we chose. With Deny (incoming), the firewall blocks connections that begin outside the computer. On a server in our lab, with the same defaults, a connection from another computer timed out, and the firewall's log recorded `[UFW BLOCK]`. With Allow (outgoing), our connections to other computers go through: with the firewall on, we opened linuxmint.com in Firefox as before. Leave the other lines as they are.

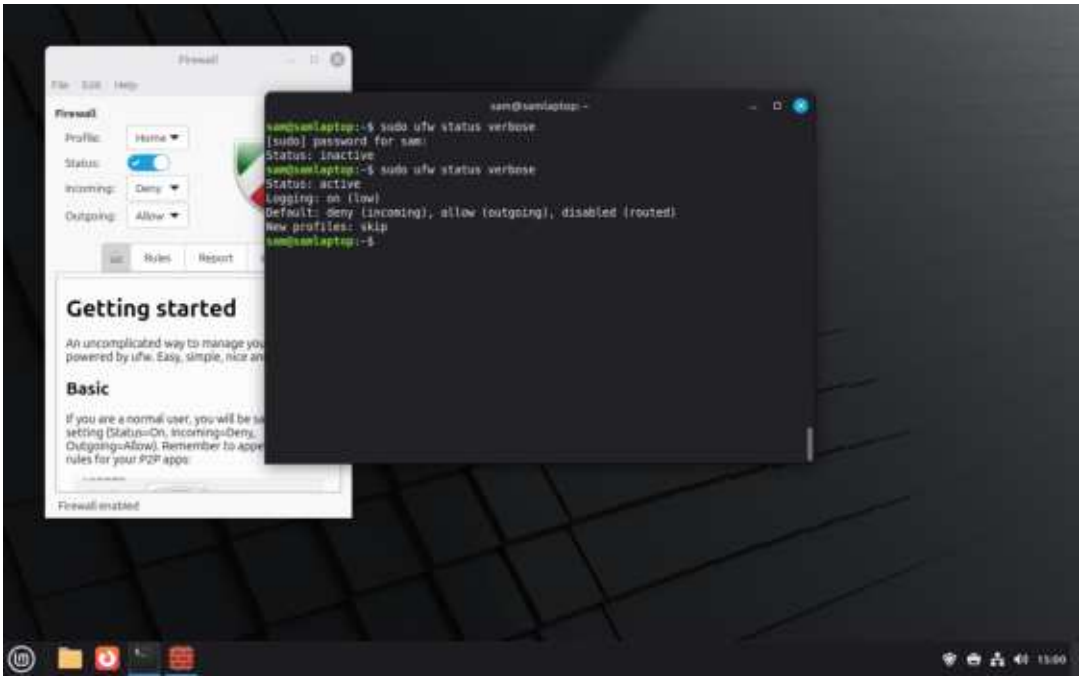


Figure 21.5. The firewall, active.

The rules

To see any rules beyond the two settings, type `sudo ufw status numbered`. In our lab, it printed `Status: active` and no numbered rules, so ufw follows the two defaults and no others.

Try it

1. Run `sudo ufw status verbose`, and read the Status line.
2. Open Firewall Configuration and turn Status on, with Incoming at Deny and Outgoing at Allow.
3. Run the command again, and find `deny (incoming)`.
4. Run `sudo ufw status numbered`.
5. Open a web page in Firefox.

22. Daily updates in Update Manager

In chapters 20 and 21, we locked the disk and switched on the firewall. Next, we keep the programs patched with updates. The Linux Mint team writes in its user guide, "'Security updates' are updates which patch vulnerabilities," and in Update Manager's welcome window from chapter 14, they give the reason to install them all: "This way, your computer is always secure and you can go back in time and recover from any potential regression." In chapter 14, we took a snapshot, then clicked **Install Updates**. Here, we turn on one setting on Update Manager's Automation tab, **Apply updates automatically**.

The Linux Mint team ties that setting to the snapshots we set up in chapter 14: "Thanks to Timeshift, which makes it easy for anyone to work around regressions by restoring snapshots, automatic updates can now be enabled easily, in the preferences."

Find the setting

Click the shield icon by the clock to open Update Manager. Choose **Edit**, then **Preferences** (Figure 22.1), and open the **Automation** tab.

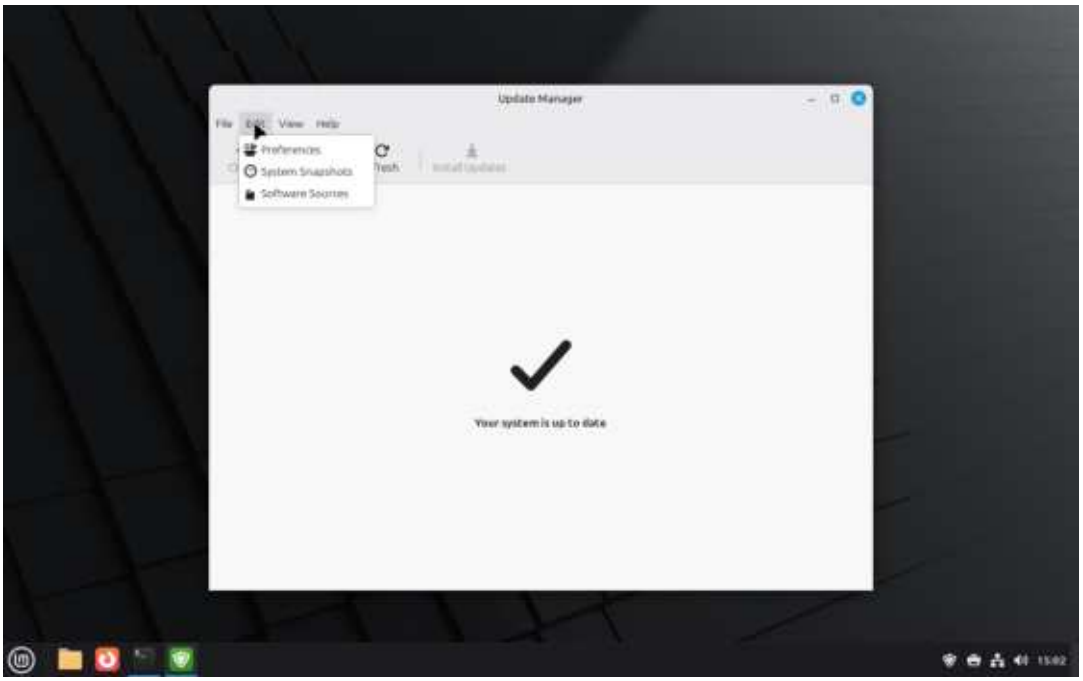


Figure 22.1. Update Manager's Edit menu. Choose Preferences.

Across the top is an orange bar with the same condition: "Make sure your system snapshots are properly configured before enabling automation." To check, choose **Edit**, then **System Snapshots**, in the same Update Manager window; Timeshift should list the snapshot from chapter 14.

Turn it on

Under **Package Updates** is the note "Performed as root on a daily basis," which means the job runs once a day with the administrator rights we used through sudo in chapter 18. Click **Apply updates automatically**. Type the password at the prompt, and the switch turns blue.

If the computer has any Flatpaks, the Flathub programs from chapter 14, also turn on **Update Flatpaks automatically**, under Other Updates, "Performed when you log in" (Figure 22.2).

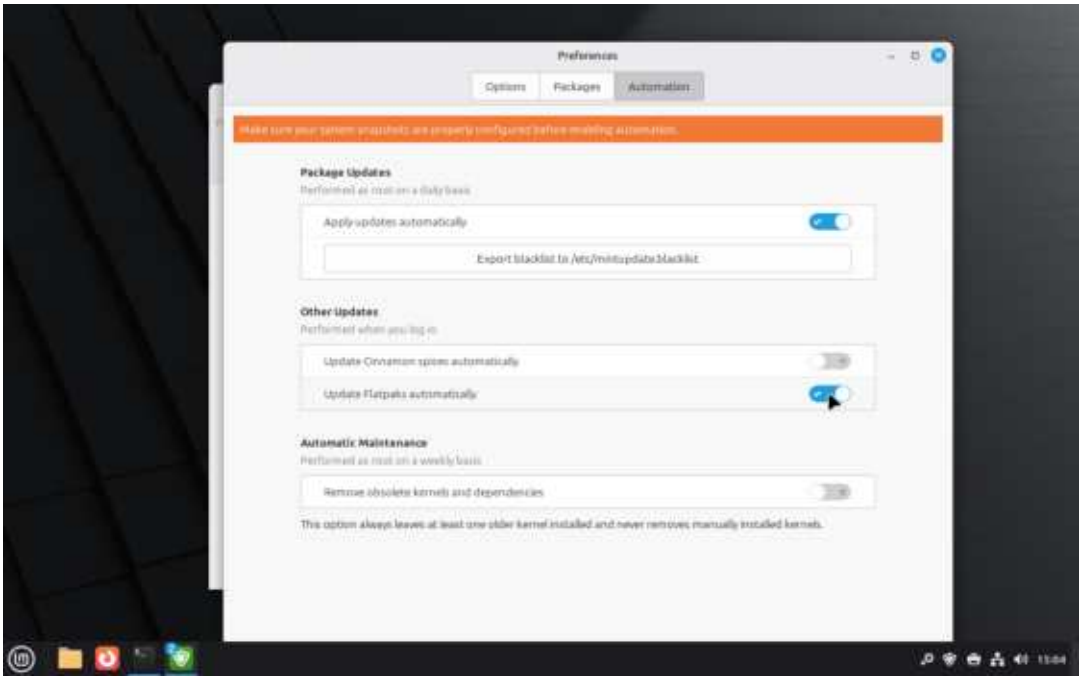


Figure 22.2. The Automation tab: daily package updates and Flatpaks, both on.

Other settings in the window

Two other switches on the Automation tab stay off unless we turn them on: **Update Cinnamon spices automatically** and **Remove obsolete kernels and dependencies**. On the **Options** tab, the Linux Mint developers set Update Manager to "Only show notifications for security and kernel updates," and to refresh its list of updates every two hours.

We turned the setting on in our lab and saw the window change. We didn't wait a day to see the result.

Restarts

When we turned the setting on in our lab, we saw a gear icon by the clock. The Linux Mint team writes that its update monitor "places an icon in your system tray" while an automatic task runs.

After some updates, the team recommends a restart. The team writes, "The Update Manager's system tray shows a warning icon if a restart is recommended following the installation of updates." When the icon appears, save any open work and restart, as we did in chapter 14.

After April 2029

The Linux Mint team sends updates for 22.3 until April 2029. Before then, we follow the team's User Guide page for the next major release, as the team published "How to upgrade to Linux Mint 22" for the last one.

Updates without the password (optional)

On a computer one person uses at home, we may want Update Manager and Software Manager to install software without the password window. We do that with one short file. The trade: once the file goes in, a program we run can install, update or remove software from the sources set up on the computer without a prompt.

The Linux Mint team built Update Manager and Software Manager to check with polkit, a program that runs in the background, before they change the system. Polkit takes its rules from files in the folder `/etc/polkit-1/rules.d/`. We add one rule there with our file: for an administrator account signed in at the computer, installs and updates go through without a prompt. The account we made while setting up Linux Mint counts as one; on the Inspiron 5567, ours did.

Make the file.

Open the menu, type text editor, and open **Text Editor**. Type these lines exactly; a missing quote or brace stops the rule from working:

```
// Update Manager and Software Manager install and update
// software without the password, for a person in the sudo group
// at the computer. Everything else still asks.
polkit.addRule(function(action, subject) {
    if ((action.id == "org.aptkit.upgrade-packages" ||
         action.id == "org.aptkit.install-or-remove-packages" ||
         action.id == "org.aptkit.install-packages.high-trust-repo" ||
         action.id == "org.freedesktop.packagekit.package-install") &&
        subject.active == true && subject.local == true &&
        subject.isInGroup("sudo")) {
            return polkit.Result.YES;
    }
    return polkit.Result.NOT_HANDLED;
});
```

Save it in the Downloads folder with the name `50-mint-updates-no-password.rules`.

Put it in place.

Open Terminal and type:

```
sudo cp ~/Downloads/50-mint-updates-no-password.rules /etc/polkit-1/rules.d/
```

Type the password at the sudo prompt, the last one for installs and updates. In our lab, the next installation went ahead without a restart.

To check it, open Software Manager and install a small program. If the Authentication Required window appears, compare the file with the lines above, fix it, and copy it again.

To take the file out:

```
sudo rm /etc/polkit-1/rules.d/50-mint-updates-no-password.rules
```

What we saw in the lab.

On a lab virtual machine running Linux Mint 22.3 Cinnamon, we checked each task before we put the file in, and every one asked for the password. With the file in place, we installed the game Mines in Software Manager without a password window, and we applied an update in Update Manager the same way. On the Inspiron 5567, the file had the same effect on installs and updates, while installing a downloaded file, removing an application through Software Manager and running a program as administrator still asked. Then we took the file out, and the prompt came back for each task.

Try it

1. Open Update Manager, then Edit, Preferences, Automation.
2. Choose Edit, System Snapshots, and find the snapshot.
3. Turn on **Apply updates automatically**, and the Flatpak switch on a computer with Flatpaks.

23. A hardware key for sudo

In chapter 18, `sudo` showed a password prompt before it let us change the computer. A person can learn a password from a distance. The United States Cybersecurity and Infrastructure Security Agency (CISA) describes the common trick in its fact sheet on phishing: "a threat actor sends an email to a target that convinces the user to visit a threat actor-controlled website that mimics a company's legitimate login portal. The user submits their username, password, as well as the 6-digit code from their mobile phone's authenticator app." If we use the same password at the sign-in screen, the stolen one opens this computer too. In another guide, CISA's staff warn, "If attackers guess your password, they would have access to your other accounts with the same password." In this chapter, we add a hardware security key to `sudo`. After that, to run a command with `sudo`, we type our password and touch the key in hand. The key goes beside the password, not in place of it.

The key

A hardware security key looks like a small USB drive with a spot to touch. Look for one marked FIDO2. CISA calls FIDO "the only widely available phishing-resistant authentication" and notes that "FIDO2-compliant tokens are available from a variety of vendors." In our lab, we used a Thetis FIDO2 Security Key, USB-A, sold in a two-pack. The maker printed Windows and Mac on the package, and Linux recognized the key with no extra setup.

Buy two. Register both, carry one, and keep the spare in a drawer.

Plug in one key, open Terminal, and type `lsusb`. The command lists each device plugged into the computer's USB ports. In our lab, we found the key's name on one line:

```
Bus 001 Device 003: ID 1ea8:fc25 ExcelSecu FIDO2 Security Key
```

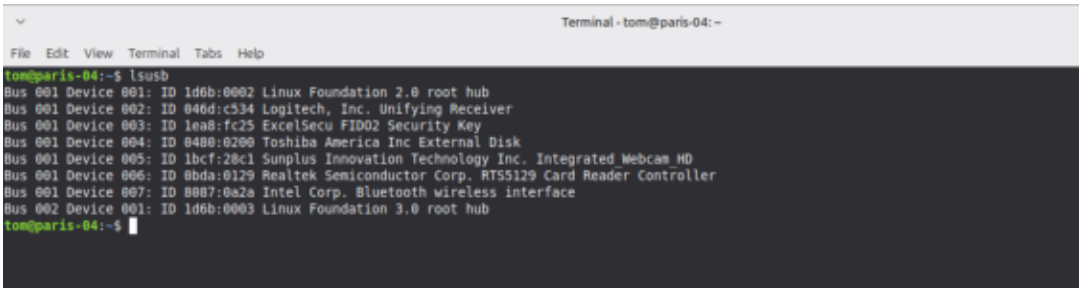


Figure 23.1. lsusb on the lab's Inspiron 5567, with the key on the third line.

Install two programs

Type:

```
sudo apt install libpam-u2f pamu2fcfg
```

The first program adds the key to the checks `sudo` makes. The second, pamu2fcfg, registers the key with the computer. Yubico, which publishes both, gives the same command on its pam-u2f page. In our lab, we installed the first by name, and apt added the second with it.

Register the key

With one key plugged in, type:

```
pamu2fcfg > u2f_keys
```

Touch the key when the command pauses. If we set a PIN for the key, a PIN prompt from pamu2fcfg appears first. pamu2fcfg prints one line, and the `>` puts it into a new file, `u2f_keys`, in the home folder. The line starts with the user name, then a long code for the key.

The program from libpam-u2f looks for that file in `.config/Yubico`, inside the home folder. A name that starts with a dot stays out of sight in Files and in a plain `ls`; `ls -a`, with a for all, shows it. Type these three commands, one at a time:

```
mkdir -p ~/.config/Yubico
mv u2f_keys ~/.config/Yubico/
chmod 600 ~/.config/Yubico/u2f_keys
```

The first makes the folder, and `-p` adds `.config` too if the computer lacks it. The second moves the file in. The third uses `chmod` from chapter 18 with numbers: with the 6, the file's owner may read and write it, and with the two zeros, the group and everyone else have no access. Type `ls -l ~/.config/Yubico/u2f_keys`, and the line starts with `-rw-------`.

Keep a way back

We are about to change the file `sudo` reads before it lets anyone in. Yubico's guide warns, "An erroneous PAM configuration may lock some or all users out of the system or prevent you from gaining root privileges." PAM is the part of Linux that checks a user's identity. Yubico's advice follows: open a second window as root first, and keep it open until the test works.

Open a second Terminal window and type:

```
sudo -i
```

Type the password. The prompt now ends in #, the mark of root. Leave this window open. If the change goes wrong, fix it from here.

Add the key to sudo

In the root window, type:

```
nano /etc/pam.d/sudo
```

Find the line `@include common-auth`. Put the cursor at the end of that line, press Enter, and type the new line below it. In our lab, a script added the same text, and the three lines then looked like this:

```
@include common-auth
auth required pam_u2f.so
@include common-account
```

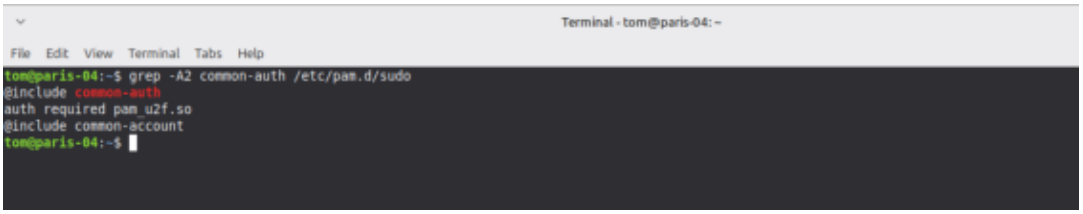


Figure 23.2. The three lines in `/etc/pam.d/sudo` on the Inspiron.

With the first line in place, we get a password prompt. With the new line, we also touch the key. Save with Ctrl+O and Enter, and leave with Ctrl+X, as in chapter 17.

Test it

Go back to the first window. In chapter 18, `sudo` kept our password for a few minutes, so clear it:

```
sudo -k
```

Now, type `sudo ls -a /root`. Enter the password, then touch the key. If the screen seems to pause at that point, `sudo` is waiting for the touch. After the touch, the listing appears. In our lab, on the Inspiron, we typed the password, touched the key, and `sudo` ran the command (Figures 23.3 and 23.4).



Figure 23.3. After we type the password, sudo waits for a touch of the key.



Figure 23.4. After the touch, sudo lists the files in `/root`.

If the listing doesn't appear, go to the root window, open the file with nano again, delete the new line, and save. We then type the password alone for `sudo` once more. Check each step above, and try again.

Register the spare

Keep the root window open. Unplug the first key and plug in the spare. In the first window, type:

```
pamu2fcfg -n >> ~/.config/Yubico/u2f_keys
```

Touch the spare. Two marks matter here. `-n` leaves the user name off, because the file already starts with it. `>>` adds the spare's code to the end of the file. A single `>` would replace the file and erase the first key, so type both.

Test the spare the same way: `sudo -k`, then `sudo ls -a /root`, the password, and a touch of the spare. When it works, type `exit` in the root window, and close it. Then put the spare in the drawer, away from the computer, beside the paper with the disk key from chapter 20. Keep the two keys apart, so the loss of one still leaves the other.

In our lab, we registered one key. The steps for the spare come from the pamu2fcfg manual, which describes `-n` as "Useful for appending," and from the program's code.

If we lose both keys, we need another. Buy one and register it the way we registered the spare. The key file belongs to our account, in the home folder, so that step needs no `sudo`.

Commands that take the key

With the key, we added a second lock to `sudo`, the word we type before a command in chapter 18. Two other routes to the administrator's rights still open with the password and no key. In password windows, such as Update Manager's, we type the password alone; in our lab, those windows didn't ask for the key. The `sudo -i` command, which opened our root window, follows a different file, `/etc/pam.d/sudo-i`, and skips the key too. Anyone who learns our password can still use those two routes, so we guard it as before. At the sign-in screen, we still type the password alone, and we still open the disk with the disk key. The lock screen from chapter 20 follows its own file, `/etc/pam.d/cinnamon-screensaver`, so a locked screen opens with the password alone, even after the optional step below.

An option: the hardware key at sign-in

We can add the hardware key to the screen where we sign in, too. In our lab, we added one line, and we then signed in with the password and a touch. Take this step for a key carried every day.

First, save a copy of the sign-in file:

```
sudo cp /etc/pam.d/lightdm /root/lightdm.before
```

Then open the file with `sudo nano /etc/pam.d/lightdm`, put the cursor at the end of the line `@include common-auth`, press Enter, and type:

```
auth required pam_u2f.so nouserok
```

Save and leave, as in chapter 17. With `nouserok`, a user without a registered key signs in with the password alone; with a registered key, we also touch it to sign in.

Now, set a timer that puts the saved copy back in ten minutes:

```
sudo systemd-run --on-active=600 --unit=undo-key-login cp /root/lightdm.before
/etc/pam.d/lightdm
```

In our lab, we kept a timer like this one running while we tested. Don't restart the computer while the timer runs; that cancels it.

Open the menu and click **Log Out**, the middle of the three round buttons at its lower left. Type the password, touch the key, and the desktop reappears. That matches our lab's test. Then cancel the timer:

```
sudo systemctl stop undo-key-login.timer
```

If the sign-in fails, wait ten minutes for the timer to put the old file back, and the password alone works again.

Try it

1. Plug in one key and run `lsusb`.

2. Install the two programs, and register the key with `pamu2fcfg`.
3. Move the file into `~/.config/Yubico`, and run `chmod 600` on it.
4. Open a root window with `sudo -i`, and add the line to `/etc/pam.d/sudo`.
5. Run `sudo -k`, then `sudo ls -a /root`, and touch the key.
6. Register the spare with `pamu2fcfg -n >>`, test it, close the root window, and put that key in the drawer.
7. For a key carried every day, add it to the sign-in screen.

24. Restore a file from a backup

One wrong click can delete a photo we kept for years. We learn whether a backup works on the day a file comes back from it, so in this chapter, we make that day happen on purpose: we make a backup, delete a file, and bring it back. In chapter 14, we took a Timeshift snapshot of Linux Mint. The Linux Mint team writes in its user guide that snapshots "do not include or affect your personal data." For documents, pictures and music, we make a separate backup, on the second drive from chapter 6.

Back up the files

Plug in the second drive. Open the menu, type backup, and open **Backup Tool**. With it, we save and restore copies of the files in the home folder; the Linux Mint team also calls it mintbackup. The window has two kinds of backup (Figure 24.1). Under **Personal data**, "Your files, settings and the content of your home directory," click **Back Up Now**.

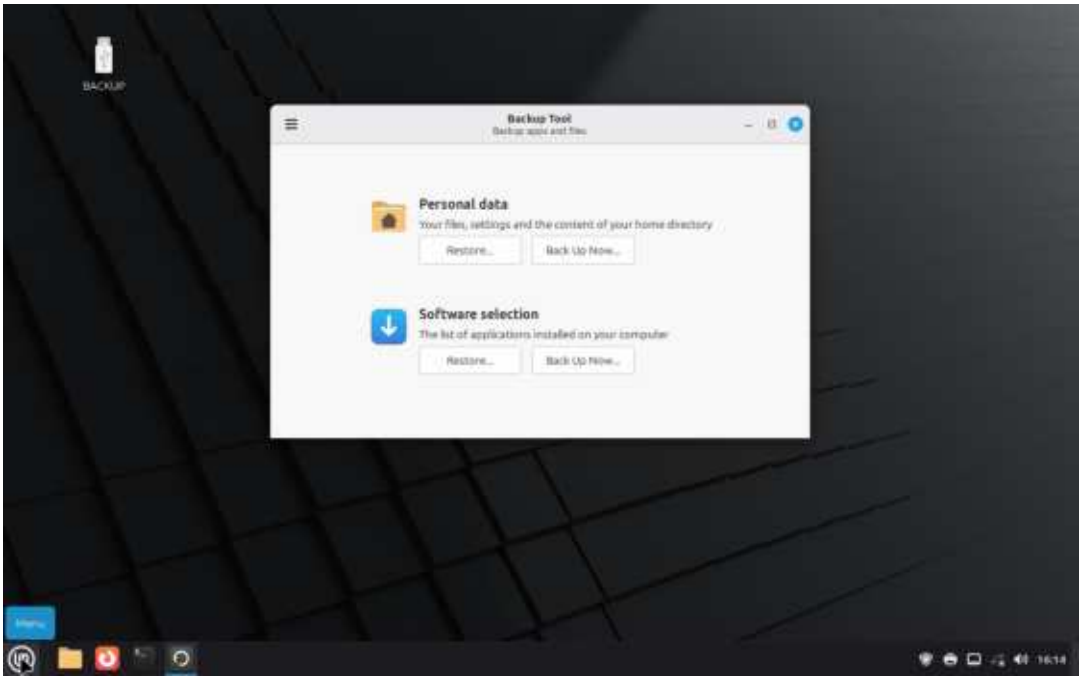


Figure 24.1. Backup Tool. Personal data: the home folder.

Next comes the prompt "Please select where you want to save the backup file," with a folder named Backups inside Documents as the default. If that disk fails, the copy is lost with it, so pick the second drive instead. Click the list, choose **Other**, select the drive's name on the left (ours is BACKUP), and press **Open**. We now see the name in the list (Figure 24.2). Click **Forward**.

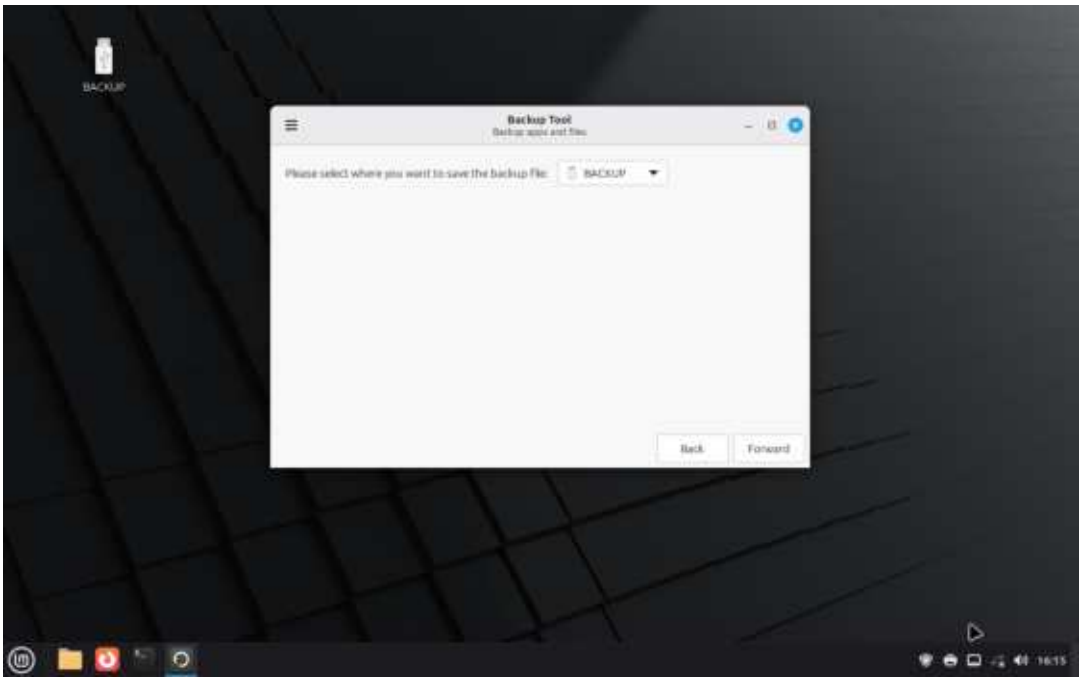


Figure 24.2. The backup goes to the second drive.

On the next page is a place to leave out files or folders. The tool already excludes Documents/Backups. Click **Forward**.

On the last page, the Linux Mint team explains what else stays out: "Hidden files and hidden folders located at the root of your home directory are not included by default" (Figure 24.3). These are the folders whose names start with a dot. The team adds that they "usually contain temporary data or preferences for your applications," and two of them matter more than that. In our lab, Firefox 159 kept our bookmarks and saved passwords in `.config`, and Thunderbird keeps the mail and the account settings in `.thunderbird`. The same page reads, "If you want to include some of them in the backup, please add them to the list below." The tool looks only at the top of the home folder, so we add all of `.config`, and the key file from chapter 23, if we added the key, goes along. Click **Include directories**, choose `.config`, and do the same for `.thunderbird`. Then click **Apply**. We didn't run a backup with these two folders added.

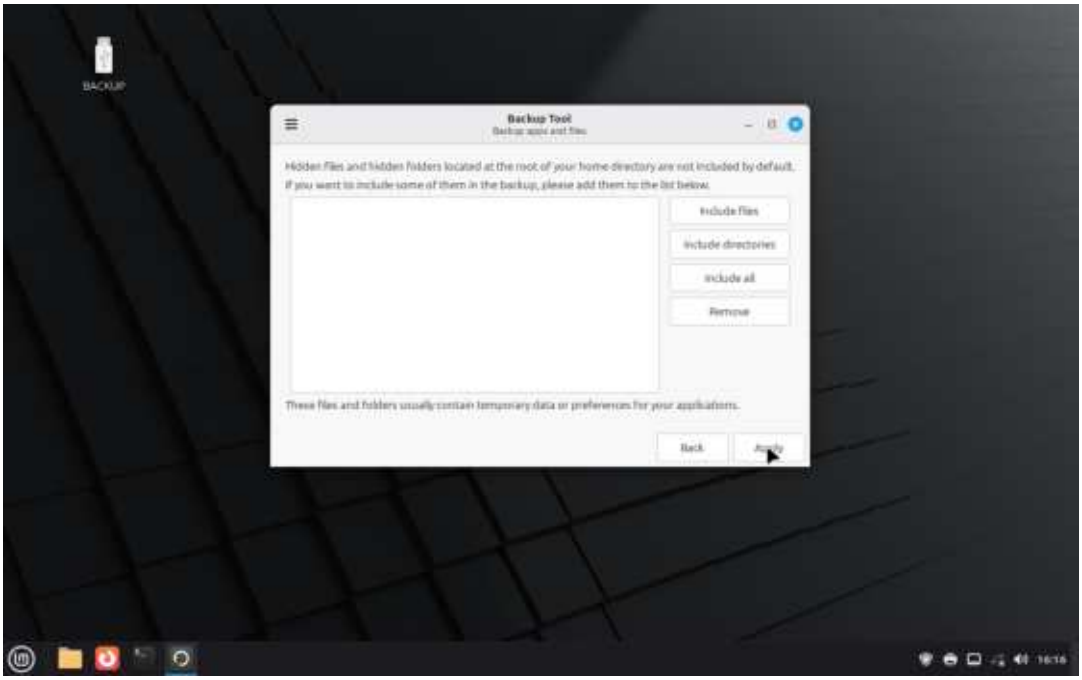


Figure 24.3. Hidden folders stay out of the backup unless we add them.

In our lab, we saw the message "Your files were successfully saved in /media/sam/BACKUP/2026-10-27-1616-backup.tar" (Figure 24.4). The file name includes the date and time. Our backup came to 151.9 MB. Backup Tool writes everything into one .tar file. As in chapter 6, a FAT32 drive takes files up to 4 GB, and years of photos can pass that, so for a large backup, use a disk formatted with exFAT or NTFS.

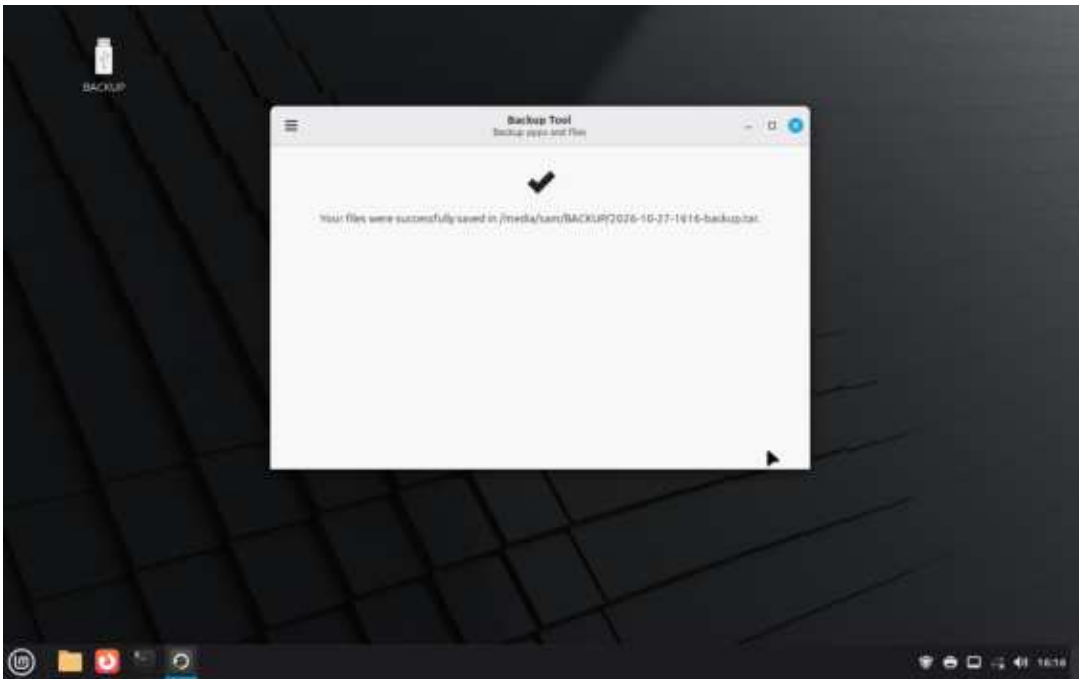


Figure 24.4. The backup, saved on the drive under the date and time.

Lose a file on purpose

Pick a small file that is safe to lose. In our lab, it was `Recipes.txt` in Documents, one of the files brought home in chapter 10. Open Terminal and look at it first:

```
ls -l Documents
cat Documents/Recipes.txt
```

Note the file's size and date. Ours were 27 bytes and Oct 26 16:56, and `cat` printed `Cornbread: 1 cup cornmeal`. Now delete it with `rm Documents/Recipes.txt`, and type `ls Documents`. `ls` no longer lists the file. `rm` skips the Trash, so the file is gone from the computer.

Restore it

Open Backup Tool again, and under **Personal data**, click **Restore**. Press the folder button beside "Please select the backup to restore," choose the drive on the left, pick the file whose name ends in `backup.tar`, and click **Open**.

The question "Overwrite existing files?" appears. Keep the first answer, "No, only restore missing files," which the Linux Mint team describes in the window as "safe," since it "does not affect your existing files" (Figure 24.5). The other choice puts every file back the way it was on the day of the backup, and beside it is the warning, "The changes you made to the files since the backup will be lost." Click **Forward**.



Figure 24.5. Restore only the missing files.

Then we saw this message: "Your files were successfully restored."

Check it

A message on the screen is not proof that the file came back. Open it. In Terminal, type:

```
ls -l Documents
cat Documents/Recipes.txt
```

In our lab, `Recipes.txt` was back, with the same size, 27 bytes, and the same date as before, and `cat` printed the cornbread line (Figure 24.6). Now we know the backup works: we used it to restore a file.



Figure 24.6. `Recipes.txt`*, back with its date and words.*

Timeshift and Backup Tool

We save different things with the two programs. Open Timeshift from the menu and type the password. In our lab, the snapshot from chapter 14 appeared in its list, with the space left on `/dev/vda2` under Available, where our lab's test computer, installed without encryption, keeps Linux Mint (Figure 24.7). On an encrypted disk, and on other computers, the name differs, as with `lsblk` in chapter 20. The team writes in its guide what a snapshot does: "If an update, a mistake, a bug or a malicious program breaks something on your computer, you can restore the operating system from any snapshot." In our lab, we didn't run a system restore.



Figure 24.7. Timeshift, with the snapshot from chapter 14, kept on the same disk as Linux Mint.

We save Linux Mint with Timeshift and personal files with Backup Tool, on a drive to unplug and carry away from the computer. In our lab, we clicked **Back Up Now** once. Pick a regular day, and click it again; each run makes a new file, named with the date and time. Store the drive in a safe place. As we saw in chapter 20, the copies on it aren't encrypted.

Try it

1. Back up the personal data with Backup Tool to the second drive.
2. Delete one small file with `rm`.
3. Restore with "No, only restore missing files."
4. Open the file and read it.
5. Open Timeshift, and find the snapshot.

25. Secure Boot and the TPM chip

In chapter 4, we left Secure Boot on, and a few readers had to turn it off so Linux Mint could start. With Secure Boot on, a computer runs only programs signed by a certificate on the list in its firmware. Microsoft's engineers sign Linux Mint's first start-up program, called shim, so the computer starts Linux Mint.

The certificate behind that signature, the Microsoft Corporation UEFI CA 2011, expired in June 2026. LWN, a Linux news site, summed up what that means: a shim signed with the 2011 certificate "will continue to work as it did before," but a computer "can fail to boot if it does not have the new certificate in its trust store and receives a new shim." In this chapter, we add Microsoft's 2023 certificate,

so Linux Mint still starts with Secure Boot on. Every owner works through the first two sections; the last two, on turning Secure Boot back on and the TPM chip, stay optional. In chapter 26, we compare it with Windows 11, row by row.

Check the certificates

The tool `mokutil`, named for the Machine Owner Keys from chapter 7, reports on Secure Boot. Open Terminal and type:

```
mokutil --sb-state
```

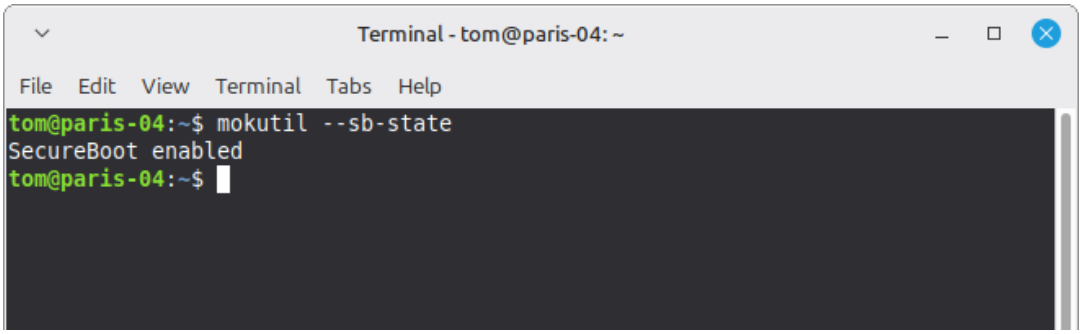


Figure 25.1. Secure Boot on, on the lab's Inspiron 5567.

If the answer is `SecureBoot enabled`, go on. `SecureBoot disabled` means we turned it off in chapter 4. In our lab, we ran the update below only with Secure Boot on, so the steps for a computer with it off haven't been tested. In our lab, on a computer without Secure Boot at all, the answer was `This system doesn't support Secure Boot`, and the rest of this section doesn't apply to it.

Now, list the certificates in the firmware's trust store:

```
mokutil --db | grep Subject:
```

The `mokutil` program prints the list. The `|`, called a pipe, sends that list to `grep`, a program that keeps only the lines with a given word, here `Subject:`, which comes before each certificate's name. On our lab's Dell Inspiron 5567, the list held three: a certificate from its maker, the Microsoft Corporation UEFI CA 2011, and the Microsoft Windows Production PCA 2011, which Microsoft's engineers use to sign Windows. Look for `Microsoft UEFI CA 2023`. If it appears, the computer has the new certificate already.

Add the 2023 certificate

Linux Mint comes with the firmware updater, fwupd. With it, we install the update that Microsoft's engineers prepared and signed with a key already in the trust store. Type:

```
fwupdmgr get-updates
```

Find the entry named **UEFI CA**, from 2011 to 2023, and copy its device ID, the long string of letters and numbers beside it. On our lab's Inspiron and on our OptiPlex 7080, the ID was the same: `5bc922b7bd1adb5b6f99592611404036bd9f42d0`. Install that one entry:

```
sudo fwupdmgr update 5bc922b7bd1adb5b6f99592611404036bd9f42d0
```

For `sudo`, type the password, and touch the hardware key if we registered one in chapter 23. Then a question appears: "Upgrade UEFI CA from 2011 to 2023?" Answer y, then restart. The firmware applies the update while the computer starts again.

Run `mokutil --db | grep Subject:` again. In our lab, the list grew to five: the three from before, plus `Microsoft UEFI CA 2023` and `Microsoft Option ROM UEFI CA 2023`. The update added two and removed none.

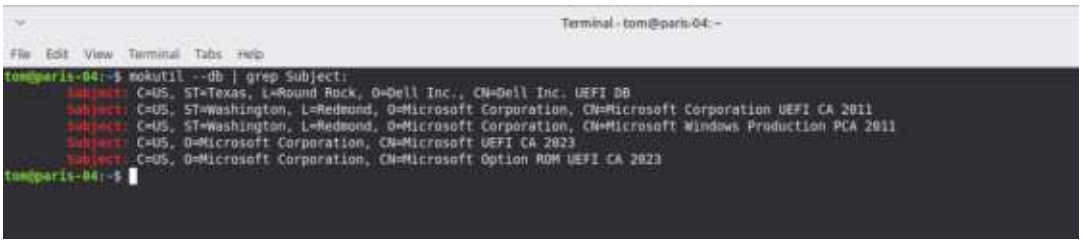


Figure 25.2. The Inspiron's certificates after the update, with Microsoft UEFI CA 2023 fourth.

Two cautions. fwupd gives that entry the same ID everywhere, so run the command on the intended computer, and read the prompt before answering. In fwupd, we may also see an update to a second list, called dbx, the list of blocked start-up programs. Leave that one for another day.

Turn Secure Boot back on

On a computer where Secure Boot was turned off in chapter 4, restart, open the setup screen with the same key as there (F2 on the Dell), and turn **Secure Boot** on again. Save and restart. If Linux Mint doesn't start, open setup again and turn Secure Boot off; the files stay as they were. In our lab, we ran the certificate update on two Dells, the Inspiron 5567 and an OptiPlex 7080, both with Secure Boot on, and reinstalled the Inspiron with Secure Boot on. We didn't test turning Secure Boot back on after an install.

Turn on the TPM chip

Microsoft's security documentation lists a TPM 2.0 chip, short for Trusted Platform Module, among the baseline items for Windows 11, and the table in chapter 26 has a row for it. The same documentation says that Windows keeps keys "provisioned in the TPM" for signing in, and that "The TPM works with BitLocker to ensure that a device hasn't been tampered with while the system is offline." On Dells with Intel processors, the setting takes its name from Intel's Platform Trust Technology, or PTT. The company's support pages describe PTT as meeting Microsoft's TPM requirement and working much like a separate TPM chip.

Check first. In Terminal, type:

```
cat /sys/class/tpm/tpm0/tpm_version_major
```

An answer of `2` means the chip is on and Linux Mint can reach it. Any other result, or an error, means Linux Mint can't reach a TPM 2.0 chip. On the lab's Inspiron 5567, before PTT was on, the check answered `cat: /sys/class/tpm/tpm0/tpm_version_major: No such file or directory`.



Figure 25.3. The TPM check on the Inspiron answers 2.

To turn it on, restart and tap F2 about once per second when the Dell logo appears, as in chapter 4. Under **Security**, turn on **PTT Security**, then save and restart. Dell's manual for the Inspiron 15 5570, another model in the Inspiron 15 5000 line, lists that setting there as the switch that makes PTT visible "to the operating system." Dell's support writers note that "BIOS menu layouts vary by model." When Linux Mint starts again, run the check once more. On the lab's Inspiron 5567, on October 29, 2026, the answer was `2`.

The chip is on, and at this writing, the Linux kernel detects it. In the future, we expect Linux Mint to use it. In this book, we still type the disk key, the one kept on paper since chapter 7, at each start. To tie the disk key to the chip, so the computer unlocks the disk at start-up without the key typed, we would need tools beyond this book.

Try it

1. Check Secure Boot with `mokutil --sb-state`.
2. List the certificates, and look for `Microsoft UEFI CA 2023`.
3. If it isn't there, add it with fwupd, restart, and list the certificates again.
4. Check the TPM chip with `cat /sys/class/tpm/tpm0/tpm_version_major`.
5. If the answer isn't `2`, turn on PTT Security in the Inspiron's setup screen, and check again.

26. Linux Mint beside Windows 11

On a Windows 11 computer, Microsoft's servers hold the account, a copy of the disk key where Windows turned on encryption, and any folders backed up to OneDrive. On the Linux Mint computer, we keep the disk key on paper and the backup on a drive, as in chapters 20 to 24. In this chapter, we trace that difference through Microsoft's pages, then set the two systems side by side in a table.

Security people call a computer that signs in to a network an endpoint. A laptop that signs in to Gmail or to a Microsoft account counts as one.

In October 2025, in the notes for a Windows 11 test build, Microsoft's Windows Insider team wrote that they were "removing known mechanisms for creating a local account in the Windows Setup experience (OOBE)," its name for the setup screens, and that "Users will need to complete OOBE with internet and a Microsoft account." On a computer set up that way, we sign in with the account on Microsoft's servers.

Microsoft's servers hold more than the account. On computers that qualify, Windows turns on disk encryption, and Microsoft's BitLocker page says it does so "with the recovery key automatically backed up to Microsoft Entra ID, AD DS, or the user's Microsoft account." A company's IT staff run Entra ID and AD DS, so on a home computer, Windows backs the key up to our Microsoft account. Microsoft's support page adds that "Microsoft Support doesn't have the ability to retrieve, provide, or recreate a lost BitLocker recovery key."

In January 2026, TechCrunch, citing Forbes, reported that Microsoft gave the FBI the recovery keys for three laptops in a fraud case in Guam, under a warrant. TechCrunch also wrote that "Microsoft told Forbes that the company sometimes provides BitLocker recovery keys to authorities, having received an average of 20 such requests per year."

On Copilot+ PCs, Windows computers built with a neural processing unit, a kind of chip, and at least 16 GB of memory, Microsoft's engineers added a program called Recall. The people who make Signal, a messaging app, wrote that Recall "takes screenshots of your apps every few seconds as you use your computer and then stores them in an easily searchable database." Microsoft's Recall documentation says, "Snapshots aren't sent to Microsoft." The pictures stay on the computer.

Microsoft's managers announced Recall in May 2024. The developers of the Brave web browser wrote that the first version kept the pictures "in a local plaintext database," a file Microsoft's engineers left without encryption, so anyone who could reach the computer could read it. They added that Microsoft's engineers rolled back the feature and reworked it.

In June 2024, Pavan Davuluri, Microsoft's executive vice president for Windows and Devices, wrote, "we encrypted the search index database," and "If you don't proactively choose to turn it on, it will be off by default." In May 2025, Signal's developers set their app to block Recall on Windows 11 and wrote, "Microsoft has simply given us no other option." The Brave browser blocks it in every tab, with a setting to turn it back on.

We can choose. A Windows 11 computer works as an endpoint on Microsoft's cloud. A Linux Mint computer becomes one too, whenever we sign in to Google or Microsoft, and we set up each protection on it, as in chapters 20 to 24.

Microsoft's guide for Windows 11 Secured-core PCs sets a floor: "Baseline Windows security is supported by Secure Boot, Bitlocker device encryption, Microsoft Defender, Windows Hello and a TPM 2.0 chip." In the table below are that list and a few more items from the company's pages, beside what we did in this book.

Windows 11, from Microsoft's pages	Linux Mint, from this book
Secure Boot	On, with the 2023 certificate (chapter 25)

Windows 11, from Microsoft's pages	Linux Mint, from this book
BitLocker device encryption, "enabled automatically" on qualifying computers, with the recovery key "automatically backed up to Microsoft Entra ID, AD DS, or the user's Microsoft account"	Disk encrypted at install; we hold the disk key, on paper (chapters 7 and 20)
Windows Firewall, "two-way network traffic filtering"	On: incoming denied, outgoing allowed (chapter 21)
Windows Update; for its Smart App Control, Microsoft's documentation tells users to keep "their devices up to date via Windows Update every month"	Installed once a day, set up in Update Manager (chapter 22)
For administrator tasks, "Windows automatically prompts the user for approval," Yes or No	For `sudo` before a command, the password, and a touch of the hardware key if we added it (chapter 23); for `sudo -i` and in Update Manager's window, the password with no key (chapters 22 and 23)
Windows Hello, a PIN or a face, so "people no longer need to carry external hardware like a security key"	The password; the hardware key at sign-in, if we take the option in chapter 23
Microsoft Defender Antivirus, "real-time, behavior-based, and heuristic antivirus protection"	The Linux Mint team's statement: Linux Mint "requires very little maintenance (no regressions, no antivirus, no anti-spyware...etc)." We don't add an antivirus program in this book; the choices follow the table
A TPM 2.0 chip, and protections built on it	On: on the lab's Inspiron 5567, we turned on PTT, the TPM 2.0 setting in Dell's setup screen, and the check in chapter 25 answers `2`. We still type the disk key, kept on paper (chapters 7 and 20)
Windows Backup, which "syncs the folders you specify to your OneDrive account"	A backup we used to restore a file (chapter 24)

For Secure Boot, disk encryption, the firewall and updates, we set up a version of each item on the Linux Mint computer. On one row, we chose differently from Microsoft: we keep the disk key on paper. On the lab's Inspiron 5567, which Microsoft's managers left off the Windows 11 list, we ran the disk encryption, the hardware key, the 2023 certificate and the TPM chip.

Microsoft's engineers build an antivirus program, Microsoft Defender Antivirus, into Windows 11. For Linux desktops, Microsoft's managers offer Defender for Endpoint to companies under a business license, in a test version their documentation calls a public preview, and leave Linux Mint off its list of supported systems. Bitdefender's support page says the company "does not offer an antivirus for Linux home users," and ESET stopped distributing updates to its home Linux program after August 3, 2022. For a home computer, the Linux Mint software sources carry ClamTk, a free window for ClamAV, a scanner whose makers write that they designed it "especially for e-mail scanning" and that it "is not a traditional anti-virus or endpoint security suite." If we want an antivirus program, we can install ClamTk from Software Manager and scan a file whenever we choose.

Try it

1. Read the table, and check each row on the Linux Mint computer.

Conclusion

We do the daily work on a computer Microsoft's managers left off the Windows 11 list: mail, letters, photos, a printed page and the web. We held on to a computer already paid for, put Linux Mint on it at no cost, and kept the money a new PC would have cost. On the desktop, the menu button is at the lower left and the clock in the corner, as on Windows 10.

Then we locked the computer down and checked each lock. We encrypted the disk and kept the disk key on paper, away from any account. We turned on the firewall, set Update Manager to install the updates each day, and brought back a deleted file from the backup on the second drive. Those of us who took chapters 23 and 25 added a hardware key to `sudo` and Microsoft's 2023 certificate.

We keep the routine from here. Pick a regular day for Backup Tool, keep the spare hardware key, if we bought one, in a drawer and the second drive in a safe place, and restart when Update Manager asks. In the next book, *Running a Linux Server*, we turn a second computer into a server, then reach it from this desk, lock it down, back it up and run it with commands.

Appendix A. Open Files on the drives

On Windows, we saw File Explorer open on This PC, with a picture of each drive (Figure 8.4). On Linux Mint, we see Files open on the home folder, and we click Go, then Computer, to see the drives. A reader who wants Files to open the way File Explorer did can make that change after chapter 18, with seven commands. Click Terminal on the panel. Type each line exactly as printed, and press Enter after each one.

1. Make two folders. One holds a small program, and the other a copy of the file behind the Files button.

```
mkdir -p ~/.local/bin ~/.local/share/applications
```

`mkdir` makes both folders, and `-p` adds `.local` too if the computer lacks it.

2. Write the small program. We name it files-open, and it asks Files for Computer.

```
cd ~/.local/bin
printf '#!/bin/sh\nexec nemo "${@:-computer:///}"\n' > files-open
chmod +x files-open
```

Nemo is the name behind Files. The `printf` line writes a two-line program into `files-open`, and it starts Nemo on Computer, or on a folder another app asks for. `chmod +x` gives the file the x, the permission to run, from chapter 18.

3. Point Files at the program. We copy that file and change the copy to start files-open.

```
cd ~/.local/share/applications
cp /usr/share/applications/nemo.desktop .
sed -i "0,/^Exec=.*/s##Exec=$HOME/.local/bin/files-open %U#" nemo.desktop
```

The file `nemo.desktop` tells the menu and the panel how to open Files. We put a copy in our folder under `.local`, where Linux Mint looks first, and `sed` changes its Exec line, the line that names what to run, to `files-open`.

Close Files, then open it again to see Computer, with a picture of each drive; we did not restart to get there. If we still see the home folder, type the seven lines again, since a repeat does no harm. Home stays one click away in the list on the left, and a program that asks for a certain folder still gets it.

To undo the change, type this line in Terminal and press Enter:

```
rm ~/.local/share/applications/nemo.desktop
```

From then on, we see the home folder when we open Files. The small program stays in its folder, unused.

Appendix B. The history behind the desktop

Linux runs most websites and most phones, almost always without its name. W3Techs reported in October 2026 that "Linux is used by 63.3% of all the websites whose operating system we know," and in September 2026, StatCounter counted Android, which Google's engineers build on Linux, on 69 percent of the mobile web traffic it measured worldwide. At the desk, the count turns around. In October 2026, StatCounter counted Windows on 76.65 percent of desktop use worldwide and Linux on 5.63 percent.

How did Windows come to hold the desk while Linux took the rest? Six stories answer that question, and this appendix tells them in order: where the picture on the screen came from, how the home computer got online, why Linux came about, how Microsoft's management built a business on the desktop, what the courts decided about it, and who builds the free desktop today. A reader can follow every step in the book without it; the history explains why the steps matter.

Where the picture came from

The picture on the screen, with its pointer, desk and folders, came from a handful of engineers over about 30 years.

The mouse and the desk

The pointer came first. Bill English, an engineer at the Stanford Research Institute, built the first mouse in 1964 from Douglas Engelbart's work. On December 9, 1968, Engelbart showed it to about a thousand people at a conference in San Francisco.

Engineers at Xerox's research center in Palo Alto drew the desk. In April 1981, Xerox's managers put a computer called the Star on sale. On its screen, a document looked like a sheet of paper and a folder looked like a folder. A person moved the mouse and clicked the picture they wanted, instead of typing a command from memory.

Apple's engineers carried the picture to more people. In 1983, the Lisa went on sale, a computer Steve Jobs named for his daughter, Lisa Brennan. On January 24, 1984, Apple's managers announced the Macintosh for $2,495. In January 1985, the company's engineers added the LaserWriter printer, and that July, Paul Brainerd's team at a company called Aldus released PageMaker. Adobe's PostScript, the language inside the LaserWriter, tied the two together, and Brainerd, the head of Aldus, named the result desktop publishing. He aimed it at small businesses, churches and schools.

Windows and the laptop

Microsoft's engineers brought the same picture to the inexpensive copies of the IBM PC that people called clones. Windows 1.0, released in 1985, placed each program in a tile, side by side. In 1990, Windows 3.0 added the Program Manager, with a small picture for each program. Five years later,

Windows 95 brought in the Start button and the taskbar, the same two parts we found on Windows 10 in Figure 1.1.

Other makers shrank the computer at the same time. In February 1982, three executives left Texas Instruments and started Compaq in Houston. Compaq's Portable went on sale in March 1983 and ran the same programs as the IBM PC. In 1989, Compaq's engineers fit a PC into the LTE, a notebook of 6.7 pounds, for $2,399 and up. Prices kept falling. A friend of mine bought a new Lenovo laptop with Windows 11 at Walmart for $150, much like the one in chapter 1.

Who owns the picture

In March 1988, Apple sued Microsoft and Hewlett-Packard and argued that Windows copied the look of the Macintosh. A federal appeals court in San Francisco decided the case on September 19, 1994, and the judges ruled against Apple: "Apple cannot get patent-like protection for the idea of a graphical user interface, or the idea of a desktop metaphor which concededly came from Xerox." A company could own its drawing of a desktop. It could not own the idea of one.

That ruling left the picture free for anyone. Sit a person in front of Windows, a Mac or Linux Mint, and they know where to click. The people who built Linux desktops drew windows, icons and menus freely, and the Linux Mint team draws the same picture today.

How the computer got online

People brought the home computer online in stages. First came services such as CompuServe, each with a separate network. Then a person dialed over the phone line and put the computer on the Internet. Then the phone and cable companies sold faster lines, and at last the makers took the wire away.

CompuServe

In 1983, a CompuServe advertisement in BYTE called the service "The Personal Communications Network For Every Computer Owner," with electronic mail for letters to friends and business associates. In 1984, Apple's modems came with a CompuServe starter pack in the box. CompuServe's engineers ran a network apart from the Internet. A Computer History Museum timeline for 1989 says that CompuServe, AOL and the services like them "have their own networks."

Bulletin boards, then a Netcom disk

Bulletin boards were computers that people reached by phone, and their operators ran dating sites on real software. A program called Dial-Your-Match, sold to bulletin-board owners for $199, had each caller fill out a questionnaire, then looked for probable matches among the others. The boards drew a crowd. In September 1994, InfoWorld's publisher, Bob Metcalfe, reported an estimate from two people in the bulletin-board business: "17 million people regularly dial into one or more of 60,000 BBSes in the United States."

Netcom's staff brought the Internet to people at home. Their service connected to the Internet in February 1991, and in July 1994 they released NetCruiser, a program that made the connection easy for a person at home. The company's filing counted about 73,000 accounts at the end of 1994 and about 308,000 a year later.

Faster lines, then no line

The phone company offered ISDN, which carried 128 kilobits per second over the ordinary phone line. It cost a great deal. In 1996, Bell Atlantic charged $249 a month for unlimited use, and in 1999, one home user priced unlimited ISDN at $283.95 a month against $49.99 for cable.

People could also buy Internet service from cable TV companies. @Home's cable service started in September 1996, and in its annual report, @Home claimed peak speeds "over 100 times faster" than a dial-up modem. The cable installers ran "one piece of coax" to many homes, though, and every customer on it shared its speed. A 1995 trade report traced early failures to cold weather, Christmas tree lights wrapped around a TV cable, and a microwave oven. @Home filed for bankruptcy reorganization in September 2001, and that December, @Home cut off 850,000 AT&T customers.

The Census Bureau counted the change. In 1997, 18 percent of American households used the Internet at home. By 2000, 42 percent did, and by 2003, 55 percent.

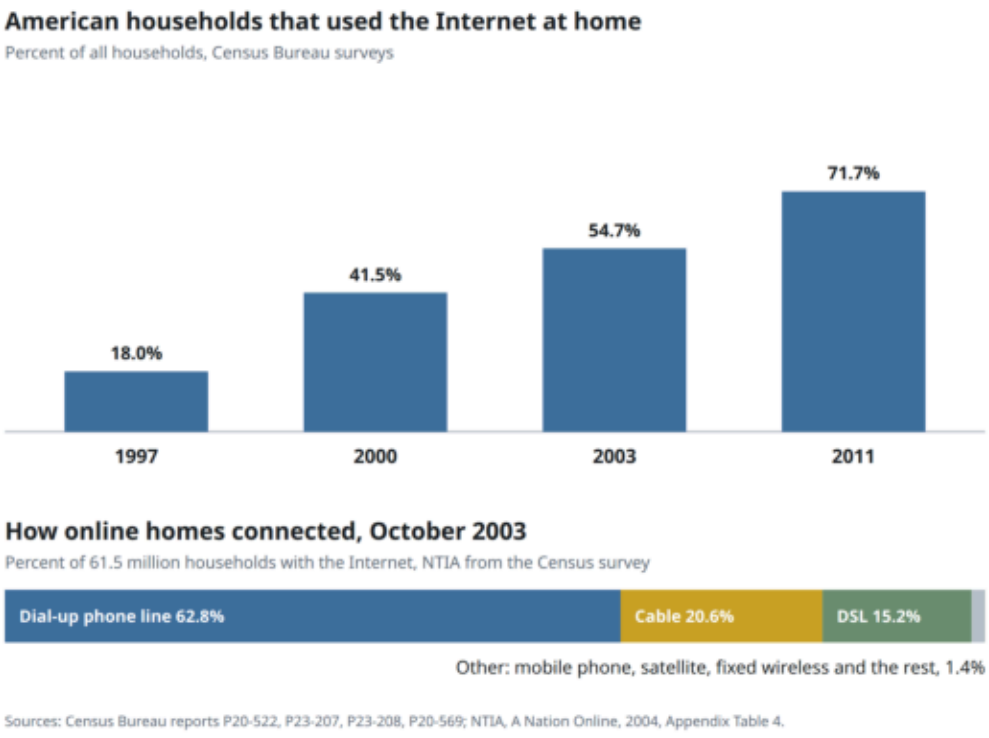


Figure B.1. American households that used the Internet at home, 1997 to 2011, and how online homes connected in 2003.

Then the makers took the wire away. Steve Jobs, then Apple's interim chief executive, unveiled Apple's AirPort wireless networking in July 1999, and in April 2000 the Wireless Ethernet Compatibility Alliance's testers certified the first products that carried the Wi-Fi name. In August 2002, T-Mobile's managers put Wi-Fi in about 1,200 Starbucks stores. From March 2003, Intel's engineers put wireless into mainstream laptops with their Centrino chips. In June 2010, Howard Schultz, Starbucks' chief executive, announced free Wi-Fi in its U.S. stores from July 1.

Where Windows went next

Once most American homes had a connection, Microsoft's management moved Windows toward it. Satya Nadella ran Microsoft's cloud business before he became chief executive in February 2014. In July 2021, when he announced Windows 365, he said, "we are now bringing the operating system to the cloud." As we saw in the introduction, a buyer now needs an Internet connection and a Microsoft account to finish setting up Windows 11 Home.

The same network carried the start of Linux, the story of the next section.

Why Linux came about

On August 25, 1991, someone at the University of Helsinki posted a short note to a Usenet group called comp.os.minix. Linus Torvalds, a student there, wrote: "I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones." He asked the group what they liked and disliked about MINIX, the small system he ran at home. People wrote back, and over the next few years, students and programmers around the world built Linux together over telephone lines.

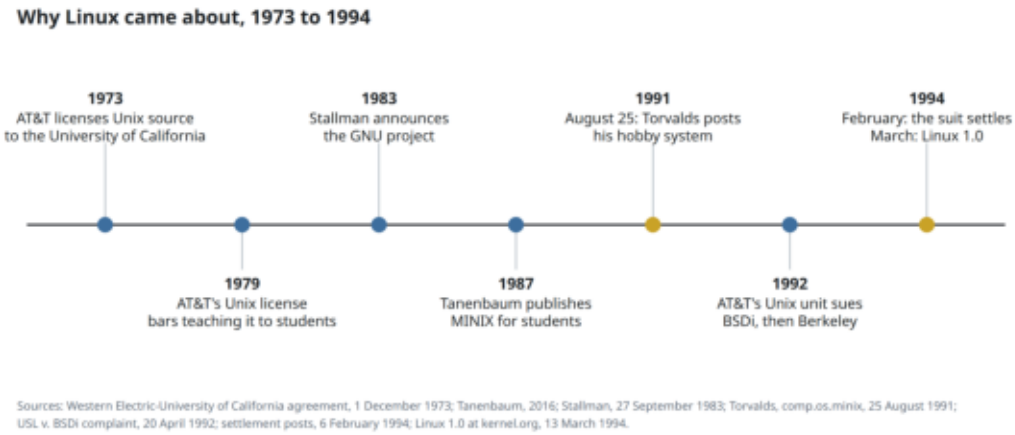


Figure B.2. Why Linux came about, 1973 to 1994.

Why students needed it

A computer science student had a lot to learn. From 1978 on, the people who wrote the curriculum reports of the Association for Computing Machinery, the field's professional society, put an operating systems course at the core of the degree. Teachers wanted their students to read a real, working system.

John Lions, who taught at the University of New South Wales and wrote a commentary on the Unix source code for his courses, put it this way: "In our opinion, it is highly beneficial for students to have the opportunity to study a working operating system in all its aspects."

Western Electric, which administered Unix for AT&T, licensed its source code to universities from 1973, and by Andrew Tanenbaum's account hundreds of schools taught from Lions' commentary. Then AT&T changed the terms. Tanenbaum, a professor in Amsterdam, wrote in 2016 that AT&T's 1979 license "explicitly forbade anyone from writing a book about it or teaching it to students." He went on: "Operating systems courses went back to theory-only mode or had to use toy simulators." Students kept the commentary alive anyway. A 1999 Salon article recalled that "fourth- and fifth-generation photocopies became treasured possessions."

Tanenbaum answered by writing a small system, with none of AT&T's code in it. He published MINIX in 1987 and, in his words, designed it to "run on cheap hardware so students could afford it." Torvalds took his first course on Unix and C in the fall of 1990, and in the spring of 1991 he ran MINIX at home on his new 386 PC. Microsoft's managers offered Windows source code for teaching much later, in 2006, to academics only and for non-commercial use.

AT&T's Unix unit holds up the other choice

Students had one more possible choice. Programmers at the University of California, Berkeley rewrote much of Unix and gave their version away, and that code became the base of 386BSD, a free Berkeley Unix for PCs. A small company, BSDi, sold a low-cost version built from the same free code.

In 1992, lawyers for AT&T's Unix unit sued BSDi and then the University of California. AT&T's unit kept the Berkeley code tied up in court until the two sides settled in February 1994, the same years Linux grew.

The writers at Klara Systems, in their history of FreeBSD, report that during the suit, many programmers who would have worked on BSD sent code and bug fixes to Linux instead. In a 2002 interview with KernelTrap, Jordan Hubbard, a co-founder of FreeBSD, said that the suit, together with troubles inside the 386BSD project, "effectively ceded the territory to Linux during a critical period." In 1993, Torvalds said, "If 386BSD had been available when I started on Linux, Linux would probably never had happened."

GNU, and the name

Torvalds wrote the kernel, the core of the system. Richard Stallman started the GNU project in 1983, and its programmers built many of the other programs a working system needs. In his first post, Torvalds named two GNU programs already running on his PC: "bash(1.08) and gcc(1.40)." I credit GNU here and call the whole system Linux.

In the Linux 1.0 credits of March 1994, 42 of the 80 people listed gave an email address at a university or a school, which shows where they worked or studied.

Students and teachers needed a system they could afford and read. AT&T's management closed off Unix with the 1979 license, the company's Unix unit held up the free Berkeley version in court, and people working together wrote Linux to fill the gap. Chapter 2 tells why the people who built Linux give

it away and why its users can trust it.

How Microsoft built its business

Microsoft's management built the company on one idea: a person pays for software. The story runs from a license signed in 1975 to an annual report filed in 2026.

A license, a letter and a language

Paul Allen and Bill Gates licensed Microsoft's first product. On July 22, 1975, in the words of the company's timeline, "Paul Allen and Bill Gates sign a licensing agreement with MITS regarding the Basic Interpreter." MITS made the Altair, an early home computer sold as a kit, and with BASIC, an owner could write programs. On November 18, 1977, "Microsoft terminates an exclusive license to MITS."

Altair owners copied BASIC from one another, and Gates answered them in print. The Homebrew Computer Club Newsletter of January 31, 1976, carried his "An Open Letter to Hobbyists." Gates wrote that "less than 10% of all Altair owners have bought BASIC." He quoted the hobbyists' view back to them: "Hardware must be paid for, but software is something to share." Then he gave his reply: "Most directly, the thing you do is theft." In that letter, Gates states the idea behind Microsoft's business.

Home computer makers then licensed the BASIC and built it in. Commodore's managers paid a flat $25,000 in 1977. In a 2025 article, Scott Hanselman and Stacey Haffner of Microsoft noted that, with the deal, Commodore placed BASIC "at the heart of Commodore's PET computers and, later, the VIC-20 and Commodore 64." Apple's 1978 manual for Applesoft, the Apple II's version of the language, reads "Copyright 1977 Microsoft Inc."

Xenix, Microsoft's Unix

For a time, Microsoft's managers sold Unix, the operating system that programmers at AT&T's Bell Laboratories wrote. On August 25, 1980, they announced Xenix, which Microsoft's history page calls "a portable, UNIX-based operating system for 16-bit microprocessors." They licensed Unix from Western Electric and sold Xenix to computer makers; in January 1985, Computerworld named IBM and Tandy among those adopting it. In February 1989, the trade newsletter Unigram/X reported that Microsoft had "effectively handed its Xenix low-end variant of Unix to the Santa Cruz Operation Inc a couple of years ago." That month, Microsoft's management announced that Microsoft would buy a share of less than 20 percent in the Santa Cruz Operation, a publisher of Unix software.

A fee for every PC

In 1994, lawyers at the U.S. Department of Justice challenged Microsoft's per-processor licenses for MS-DOS and Windows. Under them, a PC maker paid "a fee to Microsoft for each computer shipped, whether or not the computer contains Microsoft operating system software." Microsoft's management denied any violation and agreed to a settlement the court could enforce, called a consent decree. A federal judge approved it on August 21, 1995.

Novell and WordPerfect

In June 1994, Novell bought WordPerfect. Robert Frankenberg, its chief executive, later testified that Novell's plan was to "take advantage of the capabilities in Windows 95." Microsoft's engineers shared with software companies a set of programming interfaces, ways for their programs to work with Windows. One part let a person see and open documents in places such as the desktop and the Recycle Bin from inside a program.

On October 3, 1994, Bill Gates withdrew support for that part in the test version of Windows 95. In an email that day, as a federal court later quoted it, Gates wrote: "I have decided that we should not publish these extensions. We should wait until we have a way to do a high level of integration that will be harder for the likes of Notes [a Lotus product], WordPerfect to achieve, and which will give Office a real advantage."

Novell's programs for Windows 95 came out in May 1996, nine months after Windows 95 came out, and Novell's lawyers said that delay made all the difference. At trial, Microsoft's lawyers argued that the company had concerns about stability, and the technical expert Novell called agreed that a software company that wrote a bad program for those interfaces could crash Windows 95. Novell sued in November 2004. In 2013, a federal appeals court ruled for Microsoft, and Judge Neil Gorsuch wrote, "With respect to Novell at least, Microsoft did nothing unlawful."

The deal with Apple

On August 6, 1997, at the Macworld trade show in Boston, Steve Jobs, Apple's co-founder, announced an agreement with Microsoft on stage, and Bill Gates joined him on an overhead screen. Apple's press release that day listed the terms. Microsoft would "develop and ship future versions of its popular Microsoft Office productivity suite, Internet Explorer and other Microsoft tools for the Mac platform." Internet Explorer was Microsoft's program for viewing web pages, and the deal put it on every new Mac: "Apple will bundle the Microsoft Internet Explorer browser with the Mac OS, making it the default browser in future operating system software releases." A new Mac would open the web in that browser unless its owner chose another. The two firms "agreed to a broad patent cross-licensing agreement," and "Microsoft will invest $150 million in non-voting Apple stock."

The New York Times reported that Microsoft agreed to keep the stock "for at least three years." Matt Richtel, who wrote the report, added that Apple's share of the personal computer market fell "from 10 percent to less than three percent," and that the decline came "at the hands of Microsoft, which Apple has long accused of stealing the look and feel of the Macintosh operating system." The crowd answered with "gasps and boos." Fred Anderson, Apple's chief financial officer, said the agreement "strengthens Apple's viability." Chris Le Tocq, a principal analyst at Dataquest, said, "It's not that Apple is losing, but Microsoft is definitely winning."

Friends in the states and in Washington

When the states sued, Microsoft's executives called on their attorneys general. In the New York Times's account, they were "often accompanied by close friends or local colleagues of the attorneys general," and "Microsoft had just put these people on the payroll." Texas, "the state that started the multistate coalition, dropped out a few days before the suit was filed" in the spring of 1998. "At the end of 1998, the South Carolina attorney general dropped out, saying Microsoft faced plentiful competition in the market."

Ralph Reed, the former head of the Christian Coalition, signed on with Microsoft in the fall of 1998 and, Salon reported, "stayed on the company's payroll after he became a senior advisor to Bush's presidential campaign in 1999." In South Carolina's Republican primary, Reed "had Christian Coalition phone banks blasting McCain," the New York Post wrote. In April 2000, the New York Times reported that Microsoft hired Reed to win George W. Bush to its side in the antitrust case, and that Reed's aides "had been encouraging influential Bush supporters to write to the Republican nominee-to-be saying the lawsuit is misguided." Reed's firm, Century Strategies, apologized and said Reed had not asked Bush to take a position on the case, and Bush's spokesman, Scott McClellan, said, "The matter is closed."

In 2000, Microsoft and its employees gave $4.66 million to federal candidates and parties, "more than two-thirds of it to Republicans," the New York Times reported, and the company paid "at least $6 million more to lobbyists." In Salon, Joshua Micah Marshall asked whether readers were surprised that Microsoft, "home of those hard-working but culturally freewheeling computer nerds up in Redmond," hired Reed, and answered, "You probably weren't the only one."

Bill Gates gave in a different pattern. In the Federal Election Commission's records from 1989 to 2000, his contributions went to both parties, mostly to Washington state's members of Congress: Republicans such as Slade Gorton and Rick White, and Democrats such as Patty Murray and Tom Foley. On November 20, 1997, he gave $1,000 to Leahy for U.S. Senator. In those records, his gifts went to members of Congress; the bet on Bush came from the company's executives.

A guitar string maker

In 2000, federal marshals raided Ernie Ball, the guitar string maker in San Luis Obispo, California, for the Business Software Alliance (BSA), an industry association of software companies. The BSA's account says the case began with a call to its anti-piracy hotline; Sterling Ball, who runs the company, said a "disgruntled ex-employee" made the call. The BSA sued, and a judge in Los Angeles ordered an unannounced audit of the company's computers. "I was sued under federal seal. There was no warning. We were raided at ten o'clock on a Friday," Ball told Doc Searls of Linux Journal in 2003.

Ernie Ball paid a $90,000 fine over eight unlicensed copies of Microsoft Office, which the company said it used without knowing. USA Today reported that Microsoft, which co-founded the BSA, then clipped a news story about the raid and mailed it to area businesses with a letter warning that they could face similar fines; Microsoft wouldn't discuss the case but said it no longer sent such letters. Ball called the letter "fear-based marketing, with government help."

He moved the company to Red Hat Linux on its servers and OpenOffice for documents. "I didn't care if we used ten thousand abacuses," he said. "We were not going to do business with Microsoft." In 2003, Ball put the savings at $80,000.

Linux on Microsoft's servers, Windows on the desk

Today, Microsoft's customers run Linux at a large scale inside Azure, Microsoft's cloud. In May 2026, a Microsoft blog post said that "more than two-thirds of customer cores in Azure run Linux." The engineers build a Linux for those servers, Azure Linux, and Microsoft's managers offer it at the desk only inside Windows, for developers, through the Windows Subsystem for Linux, which arrived in March 2016.

In the annual report for the year ended June 30, 2026, management listed its plan for Windows: "Leveraging Windows to fuel our cloud business, grow our share of the PC market, and drive increased engagement with our services like Microsoft Edge, Bing, Copilot, Microsoft Teams, Microsoft 365 Consumer, XBOX Game Pass, and more." Revenue that year came to $331.8 billion. Customers paid about $111 billion of it for Microsoft 365, which counts the Windows sold to businesses, and $17.1 billion for Windows and Devices. The license of 1975 has become a desk that leads to the cloud.

What the courts decided

The government's case against Microsoft came to trial in 1998. Two pieces of evidence gave it a phrase and a plan, and the judges' decisions shaped the Windows that owners use today.

A phrase and a memo

On November 9, 1998, in the antitrust trial United States v. Microsoft, Steven McGeady of Intel testified about a meeting at Intel's site in Oregon in November 1995. Paul Maritz, a Microsoft executive, attended. McGeady said someone at that meeting described Microsoft's strategy as "embrace, extend, extinguish." He explained the phrase: "they would embrace Internet standards, extend them, presumably, in incompatible ways that others couldn't follow, and thereby extinguish the competition."

The next morning, the government's lawyer read a line from McGeady's notes of a meeting. He said the line recorded the words of a Microsoft participant: "kill HTML by extending it." HTML is the language of web pages. Under oath, McGeady gave one Intel executive's account of those meetings.

Three months before that testimony, in August 1998, Vinod Valloppillil, an engineer at Microsoft, wrote an internal memo on open-source software, the kind of project Linux is. He put it this way: "By extending these protocols and developing new protocols, we can deny OSS projects entry into the market." Protocols are the agreed rules that let programs on different computers talk to each other, and OSS means open-source software. Eric Raymond published the memo after it leaked and named it the Halloween document. The New York Times reported on November 3, 1998, that "Microsoft executives acknowledged on Monday that the document was authentic." In a statement, Microsoft called it "a technical analysis written by an engineer in a staff capacity," not an official position.

The verdict and the breakup that failed

In 1999, the trial judge, Thomas Penfield Jackson, found that Microsoft held a monopoly. He wrote that putting Internet Explorer in Windows at no separate charge was "benefitting consumers" in some ways, and in the next paragraph he named harm "to the detriment of consumers." On April 3, 2000, he found that Microsoft violated the Sherman Antitrust Act, and on June 7, he ordered the company split in two: one business for operating systems, one for applications. That January, days after reports that the Justice Department would propose a breakup, Bill Gates handed the chief executive's job to Steve Ballmer and stayed on as chairman; both men said the two events weren't connected.

On June 28, 2001, the appeals court upheld the finding that Microsoft illegally kept its monopoly but threw out the breakup order. The judges wrote that Jackson held "secret interviews with members of the media" and made "numerous offensive comments about Microsoft officials" outside the courtroom, though they found "no evidence of actual bias." In the 2002 final judgment, the judge required Microsoft to share technical details with other firms and barred retaliation against PC makers. The company admitted no wrongdoing. In 2019, Gates said, "There's no doubt the antitrust lawsuit was bad for Microsoft," and that he was "just too distracted" to win the phone; Google's Android won it.

An office suite for both systems

In the late 1990s, programmers at a German company, Star Division, made an office suite for Windows and Linux, StarOffice. The Los Angeles Times called it "the most popular office software among Linux users." Sun Microsystems' management bought the company in 1999 and gave StarOffice away as a download.

I watched people in the industry look for other answers, and to them, an office suite that ran on both Windows and Linux looked like a winner. In July 2000, Sun's management announced that Sun would release StarOffice's code as OpenOffice.org, and it did so that October. In 2010, members of that project started LibreOffice, which comes with Linux Mint.

What happened to Intel

In his 1999 findings, Judge Jackson told what happened to Intel in 1995. Intel, he wrote, "is engaged principally in the design and manufacture of microprocessors," the chips that run a PC. Intel's programmers also wrote software at its Intel Architecture Labs. By early 1995, people at the labs had Native Signal Processing (NSP) software in the advanced stages of development, and it "would endow Intel microprocessors with substantially enhanced video and graphics performance." The labs also had versions under way for operating systems other than Windows, and "Microsoft reacted to Intel's NSP software with alarm."

The judge gave Microsoft's side. Its managers worried that a version for Windows 95 would not be ready by the time PC makers began loading the new system, and they complained that Intel had not tested the software enough. The judge wrote that "Microsoft's concerns with compatibility and quality

were genuine." Beneath those concerns, he found a "more abiding fear at Microsoft that the NSP software would render ISVs, device manufacturers, and (ultimately) consumers less dependent on Windows." By ISVs, he meant the companies that sell programs.

In the first week of July 1995, Bill Gates met Andrew Grove, who ran Intel. Afterward he told his senior staff in a memo that he had tried to convince Grove "to basically not ship NSP." Microsoft's executives offered to build NSP's functions into Windows if Intel would stop promoting NSP's interfaces, the points where programmers connect their work to a system. Microsoft's people also pressed the PC makers not to install NSP until the code stopped exposing them, and "by the end of July 1995, Intel had agreed to stop promoting its NSP software."

On August 2, 1995, Gates told Grove that Intel could not count on Microsoft to support its next processors while Grove's company built programs that competed with Windows. "Faced with Gates' threat, Intel agreed to stop developing platform-level interfaces that might draw support away from interfaces exposed by Windows." Platform-level code runs beneath the programs a person uses, as an operating system does.

From Intel's story and those of other firms, the judge drew a general finding: "it is Microsoft's corporate practice to pressure other firms to halt software development that either shows the potential to weaken the applications barrier to entry or competes directly with Microsoft's most cherished software products." By the "applications barrier," he meant the many programs written for Windows: buyers stay with the system to keep running them.

The settlement

On June 28, 2001, after the appeals court ruled, John Ashcroft, George W. Bush's attorney general, said the court "unanimously found that Microsoft engaged in unlawful conduct to maintain its dominant position in computer operating systems. This is a significant victory." On September 6, Justice Department officials announced that the department "will not seek a break-up of the company," citing "prompt, effective and certain relief for consumers." CBS News reported that David Boies, who led the government's case at trial, said a breakup might have been easier for Microsoft, and that Boies doubted Ashcroft decided on politics alone. On November 2, with nine states, the department's lawyers reached a proposed settlement with Microsoft, and Ashcroft announced "a strong, historic settlement." He said his department "has fully and completely addressed the anti-competitive conduct that was outlined by the Court of Appeals."

Before a federal judge approved the 2002 judgment, the public had its say. Under the Tunney Act, a federal law on antitrust cases, a judge decides whether such an agreement serves "the public interest," and people and groups had until the end of January 2002 to send comments.

On December 12, 2001, Senator Patrick Leahy of Vermont, who chaired the Senate Judiciary Committee, opened a hearing with the test he said courts use for a remedy: "The remedy must end the anticompetitive practices, it must deprive the wrongdoer of the fruits of the wrongdoing, and it must ensure that illegality never recurs." A company uses anticompetitive practices to shut out competitors.

Charles A. James, the assistant attorney general over the Antitrust Division, defended the agreement. "We think that it is," he said of the public interest. He called the government's case "a fairly narrow challenge" from the start, and he said the appeals court left the liability findings "drastically curtailed."

On January 28, 2002, Ralph Nader, with James Love of the Consumer Project on Technology, sent their comments. They wrote that the order "addresses important areas where Microsoft has abused its monopoly power," and also: "It is astonishing that the agreement fails to provide any penalty for Microsoft's past misdeeds." In their view, "there is ample evidence that Microsoft is focused on strategies to cripple the free software movement," meaning the people who write programs anyone may copy and change, Linux among them. The two named GNU/Linux "the most important rival to Microsoft in the server market," the market for the computers that run networks and websites. In the same letter, they quoted Steve Ballmer, then Microsoft's chief executive, from the Chicago Sun-Times of June 1, 2001: "Linux is a cancer that attaches itself in an intellectual property sense to everything it touches."

Europe, and the browser

Europe went further. The European Commission fined Microsoft €497.2 million in 2004, ordered it to disclose how Windows exchanges information with other programs, and ordered it to offer a version of Windows without Media Player. A European court upheld the decision in 2007. In 2009, Microsoft committed to show each user a choice of web browsers. In 2013, the Commission fined it €561 million for leaving that screen out of a Windows 7 update. On September 12, 2025, the Commission accepted Microsoft's commitment to sell its office suites without Teams, at a lower price.

Mozilla, which makes Firefox, is what remains of Netscape, the browser company Marc Andreessen and Jim Clark started in 1994. In 1998, people at Netscape released the source code of their browser, and I compiled that code on Linux.

In the end, users settled the browser war. They left Internet Explorer for a browser from Google. Figures from StatCounter, a company that measures web traffic, showed that Chrome passed Internet Explorer for a full day in March 2012. In May 2012, for the first full month, Chrome led, with 32.43 percent of the world's page views to Internet Explorer's 32.12 percent.

A seller of licenses and subscriptions has a reason to keep a customer's files and sign-in inside its programs. Judges in the United States and the European Commission decided where Microsoft's methods broke the law, and Microsoft changed its products and terms under those decisions. On a free desktop, the reader keeps documents and photos in another place.

The Linux desktop today

The courts settled what Microsoft's management could do with Windows. They left open the question a Windows 10 owner faces today: what runs on the other desktop, who pays to build it, and where it works.

Programs that skip Linux

Adobe's Photoshop, InDesign, Premiere Pro and Acrobat, and Microsoft's Word, Excel and PowerPoint, come as desktop programs for Windows and macOS and no other system. On Linux Mint, a person uses Outlook and Teams in a web browser (chapter 15).

The managers of the companies that sell cloud storage make the same choice. Microsoft's OneDrive requirements page, updated June 23, 2026, names Windows, macOS, iOS, Android and web browsers, and Linux stays off the list. Google's help page for Drive reads: "Linux: The desktop version isn't available. Instead, you can use Google Drive on the web." Apple's iCloud requirements page of September 30, 2026, lists "iCloud for Windows 15," with Linux left unmentioned. Microsoft's Teams page reads, "Teams is no longer supported on Linux," and in 2022, Microsoft wrote, "We will be retiring the Microsoft Teams desktop client on Linux," and pointed Linux users to the web app. Dropbox's developers, by contrast, offer a desktop app for Linux.

All the same, Microsoft's engineers keep their programming interface open, the set of commands other programs use to reach Microsoft's services. Named Graph, it reaches the 365 services, OneDrive and OneNote, and it answers any program a developer registers. Microsoft's Work IQ server, which lets AI agents "read, create, update, and delete Microsoft 365 entities," runs on Linux, according to its GitHub page.

That open door let outside developers write the Linux apps for people at the desk. The "onedrive" program, written by a developer who signs as abraunegg and free under the GNU General Public License (chapter 2), syncs OneDrive on Linux. GNOME's developers added OneDrive to the desktop's Online Accounts on March 20, 2024, and the Linux Mint team brought Online Accounts back to Cinnamon in Linux Mint 22 with a separate app.

Why a free desktop matters

Most people at a desk choose another system, even when they could have Linux for free. Satya Nadella, Microsoft's chief executive, counts it as competition all the same. Among the risks in the annual report he signed: "Competing with operating systems licensed at low or no cost may decrease our PC operating system margins." Linux Mint is one of those systems.

The free desktop matters for a second reason: the files stay on the disk, where the person who bought the computer controls them. On free Outlook.com mail, attachments stay on Microsoft's servers. A user gets 15 GB for mail, and the attachments count against 5 GB of online storage shared with OneDrive files and photos. When that 5 GB fills, Microsoft's support page says, "You will not be able to send or receive email in Outlook.com."

In this book's lab, we run Linux Mint on a Dell Inspiron 5567 bought unused for $173.15, and in chapter 12, we open Word, Excel and PowerPoint files in LibreOffice on it.

Who builds the Linux desktop

Companies pay many of the people who work on Linux (chapter 2), and donors support the desktop projects. GNOME's 2025-26 budget came to about $435,000, with three full-time staff. KDE e.V. took in about seven of every ten euros of its 2025 income from members and donors. The Linux Mint team counts 2,434 patrons who give $5,258 a month, and in September, 443 donors gave $13,824. Germany's Sovereign Tech Fund granted KDE €1,285,200 in May 2026 and GNOME €1 million in 2023.

Set those numbers beside the Linux Foundation. Its managers spent $284.8 million in 2025, about 650 times GNOME's budget. Its annual report sorts the projects by segment, among them cloud, containers and virtualization at 23 percent, networking and edge at 14, AI and data at 12, and the Linux kernel at 2. A search of its pages for the word desktop finds one use, in the name of a finance-industry standard for trading programs. The foundation's 2025 board seats directors from Microsoft, Google, Meta, Intel, Oracle, Samsung, Huawei, Red Hat, OpenAI and nine other companies, and leaves GNOME, KDE and Linux Mint off it. Developers receive pay from Canonical and other firms to work on the desktop, and we found no published figure for it. The people who decide where that money goes answer to their bosses, and most of them send it to servers. Linus Torvalds, who started Linux, said in a 2025 video, "I don't care about anything other than the kernel."

Where it works

In Germany, the Schleswig-Holstein State Chancellery reported, in December 2025, that nearly 80 percent of the state's administrative workplaces outside the tax authorities had moved to an open-source office suite. In September 2026, it counted more than 200 of its staff working on Linux. Munich's council went the other way in 2017 and voted to return to Windows, and ZDNET reported in October 2026 that the city is not bringing its Linux desktop back. In the report the council acted on, consultants at Accenture traced staff complaints about their computers to old and mixed operating systems, programs that didn't work with the operating systems, and hard-to-use office software, and The Register reported that the consultants "found the most important issues were organisational." Some officials keep the free system and others leave it, and a reader who installs Linux Mint adds one more computer to the count.

Index

AHCI, 22, 24
Andreessen, Marc, 148
Android, 13, 137, 146, 149
antitrust case, Microsoft, 12, 144, 145, 146, 147
Apple, 137, 138, 140, 142, 143, 149
apt, 104, 105, 106, 107, 119
ASUS, 20
AT&T, 139, 141, 142
Authentication Required window, 68, 78, 84, 117
Azure Linux, 14, 145
Backup Tool, 123, 125, 126, 127, 128, 135
backups, 63, 123, 124, 125, 126, 128
BASIC, 142
BIOS, *see* setup screen
BitLocker, 19, 20, 24, 25, 37, 132
bookmarks, 32, 33, 37, 63, 64, 66
boot menu, 20, 21, 22, 25
bulletin boards, 138
Calc, 74, 76
calendar, 71
Calibri, 78, 79
cat, 97, 126, 127, 131
cd, 93, 94, 96
certificates, Secure Boot, 23, 108, 128, 129, 130, 131
chmod, 100, 101, 103, 119, 123, 136
chown, 103
Chrome, 32, 34, 67, 69, 70, 89
Chromebooks, 13
ChromeOS, 67
Cinnamon, 12, 16, 20, 25, 43, 109
Claude Code, 15
Command Prompt, 19, 93, 94
CompuServe, 138
Computer view, in Files, 136
Control Panel (Windows), 19, 25, 47
cp, 97, 100, 103, 117, 122, 136
Dell, 6, 9, 20, 23, 24, 131
Device Encryption, 19
disk key, 39, 108, 109, 110, 131, 133
donors, 13, 14, 150
download, checking, 17, 19
Driver Manager, 28, 83
drivers, 28, 41, 80, 81, 83
Edge, 10, 32, 35, 36, 63, 64
encryption, 19, 39, 108, 109, 132, 133
Excel, *see* Calc
FAT32, 30, 125
File Explorer (Windows), 10, 16, 19, 29, 45, 136
Files, 10, 45, 58, 74, 96, 136
Firefox, 10, 15, 55, 67, 74, 89
firewall, 110, 111, 112, 113, 114, 133
firmware, 19, 20, 23, 128, 129, 130
Flatpaks, 88, 115, 116, 118
fonts, 78, 79, 80
Gates, Bill, 142, 143, 144, 146, 147
Gmail, 26, 55, 56, 57, 70, 71
GNOME, 149, 150
GNU, 13, 15, 98, 99, 141, 148
GNU General Public License, 13, 15, 98, 99, 149
Google, 13, 67, 68, 69, 71, 104
Google Drive, 71, 74, 149
hardware key, 108, 119, 120, 121, 123, 133
hidden folders, 119, 121, 123, 124, 125
home folder, 40, 59, 100, 119, 123, 136
HP, 20, 80, 82
IBM, 137, 138, 142
IMAP, 28
Impress, 74, 76, 77
installing Linux Mint, 26, 37, 39
installing programs, 78, 84, 88, 105, 117, 118
Intel, 9, 13, 130, 145, 146, 147
KDE, 150
kernel, 13, 14, 15, 116, 131, 150
Kindle, 13
Lenovo, 9, 20, 138
less, 98, 99
LibreOffice, 6, 51, 52, 58, 89, 90
Linux Foundation, 13, 14, 150
Linux Mint, 11, 15, 16, 17, 84, 90
live session, 25, 26, 28, 37, 83, 110

lock screen, 110, 122
ls, 97, 99, 100, 101, 103
lsblk, 108, 109, 110, 127
macros, Excel, 89, 90, 91
man, 95, 96
menu button, 6, 10, 34, 43, 51, 57
Microsoft, 132, 142, 143, 145, 148, 149
Microsoft account, 6, 20, 131, 132, 133, 140
Microsoft Defender, 132, 133
Microsoft Teams, 28, 89, 145, 148, 149
Microsoft Word, *see* Writer
mkdir, 96, 99, 119, 136
MOK (Machine Owner Key), 41, 129
mokutil, 129, 130, 131
Mozilla, 15, 65, 70, 71, 148
Munich, 150
mv, 97, 98, 99, 119
nano, 96, 97, 99, 100, 103, 120
Nemo, 136
Netscape, 148
network, connecting to, 25, 27, 28, 37, 80, 140
NVIDIA, 83
OneDrive, 131, 133, 149
Online Accounts, 71, 72, 73, 149
open-source software, 9, 14, 145, 150
Outlook, 28, 36, 70, 71, 89, 149
panel, 25, 44, 45, 50, 96, 136
passwords, saved, 32, 34, 35, 36, 64, 65
PDF, saving as, 77, 78, 80
permissions, 99, 100, 101, 102, 103, 136
PowerPoint, *see* Impress
PowerShell, 17, 19, 93
printers, 27, 54, 80, 81, 82, 83
prompt, terminal, 92, 95, 102, 108, 117, 118
pwd, 92, 93, 95, 96
Recall, 132
recovery key, BitLocker, 19, 20, 22, 24, 25, 132
restart, 24, 41, 57, 86, 87, 88
rm, 99, 103, 117, 126, 128, 136
rmdir, 99
Roku, 13
Rufus, 17, 18, 19
Secure Boot, 22, 23, 24, 41, 129, 130
setup screen, 20, 21, 22, 24, 108, 130
snapshots, 84, 85, 115, 123, 127, 128
Software Manager, 78, 79, 84, 88, 117, 118
Stallman, Richard, 141
Start (Windows), *see* menu button
Steam Deck, 13
sudo, 100, 103, 117, 120, 121, 122
System Monitor, 48, 49, 50
System Settings, 47, 50
terminal, 45, 92, 93, 95, 96, 105
Thunderbird, 28, 37, 70, 71, 74, 124
Timeshift, 84, 85, 88, 115, 127, 128
Torvalds, Linus, 13, 15, 140, 141, 150
Toyota, 13
TPM chip, 128, 129, 130, 131, 132, 133
Trash, 65, 66, 99, 126
Ubuntu, 11, 15, 67, 89, 90
UEFI, 20, 24, 128, 129, 130, 131
Unix, 141, 142
Update Manager, 84, 85, 86, 114, 115, 117
updates, Linux Mint, *see* Update Manager
upgrading Linux Mint, 6, 116
USB drive, making, 16, 17, 18, 19
Windows 7, 19, 20, 29, 32, 58, 60
Windows 10, 6, 9, 10, 12, 19, 24
Windows 11, 6, 9, 108, 131, 132, 133
Windows Update, 9, 12, 84, 133
Word, *see* Writer
workspaces, 47, 48, 50
Writer, 51, 52, 53, 58, 74, 75
Xfce, 27, 109

Your computer works.
Keep it.

Microsoft's managers ended support for Windows 10 and say to replace computers left off the Windows 11 list. This book shows another way: put Linux Mint on the computer you already own, free, on a desktop that looks like the one you know.

- Try Linux Mint for an afternoon from a USB drive, without changing Windows.
- Copy your files, install, and bring back your documents, bookmarks and passwords.
- Write, print, browse, email, and open Word, Excel and PowerPoint files.
- Lock it down with an encrypted disk, a firewall, daily updates, and a backup you've tested.

Tom Adelstein wrote Linux System Administration for O'Reilly, an international best seller in 14 languages, and founded Bynari, IBM's first Linux Influencer Partner.

linuxhello.org

ISBN 979-8-9829152-0-7