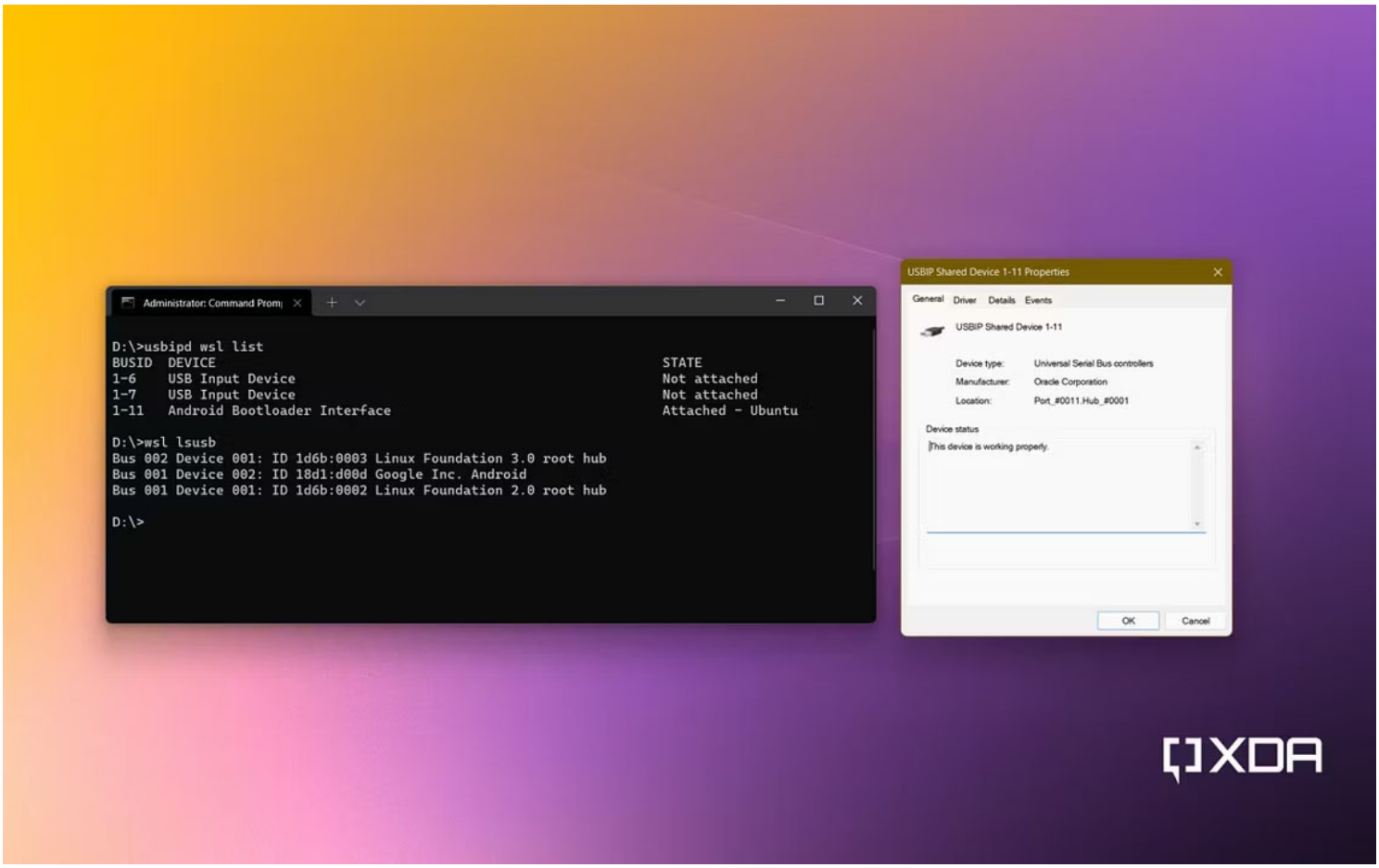
How to connect USB devices in Windows Subsystem for Linux under Windows 11

It's possible to connect USB devices in Windows Subsystem for Linux under Windows 11 and work with them directly.

BY **SKANDA HAZARIKA** UPDATED APR 3, 2023



Quick Links

Microsoft introduced Windows Subsystem for Linux (WSL) a while back, which brought support for running Linux apps straight from your Windows installation without the need to set up a virtual machine. Being such a complex compatibility layer, WSL had

some limitations. The initial implementation didn't even use a full-fledged Linux kernel, so several applications faced compatibility issues. Version 2 made a lot of improvements, but something was still missing: support for connecting physical USB devices.

If you're an active Windows Subsystem for Linux user and want to access USB devices right from the Linux instance running under WSL, then you'll be happy to know that there has been significant progress on this front. Thanks to Microsoft's contributions to the open-source usbipd-win project, you can now enjoy USB pass-through access in WSL. It means end users can perform tasks like flashing Android devices under Windows Subsystem for Linux, which are otherwise not supported by the Linux subsystem.

What is usbipd-win?

The usbipd-win project is all about creating a solution for sharing locally connected USB devices to other machines, including Hyper-V guests and WSL2. However, the official WSL2 Linux kernel used to lack support for the USB/IP protocol, requiring users to build the kernel themselves.

Since WSL kernel version 5.10.60.1, the required parameters are configured out of the box, so no manual recompilation is needed. Microsoft has also contributed a command line interface to the project to both enable a USB device for sharing on Windows and seamlessly attach it from within WSL.

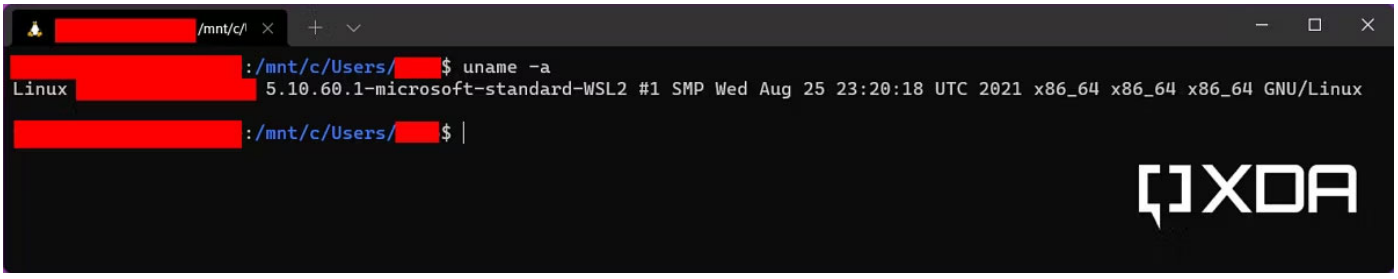
Connect USB devices under Windows Subsystem for Linux

Before starting, make sure that you're running Windows 11 and have installed all WSL updates. As mentioned earlier, you need a WSL 2 Linux distro for this to work.

1. Run the following command from within WSL:

```
uname -a
```

   If it shows a kernel version of 5.10.60.1 or later (as shown below), then you're good to go.



   - If the installed kernel version is older than 5.10.60.1, then it can be updated using the following commands:

```
wsl --shutdown
wsl --update
```

2. Head over to the GitHub repo of the usbipd-win project. Then download and install the latest release.

   - The latest version of usbipd-win supports silent driver installation, thanks to the Windows Hardware Compatibility Program certified drivers. It also supports Dev channel builds of Windows 11.
   - In addition to the release page on GitHub, you can install and update the app using Windows Package Manager CLI (aka winget).

```
winget install --interactive --exact dorssel.usbipd-win
```

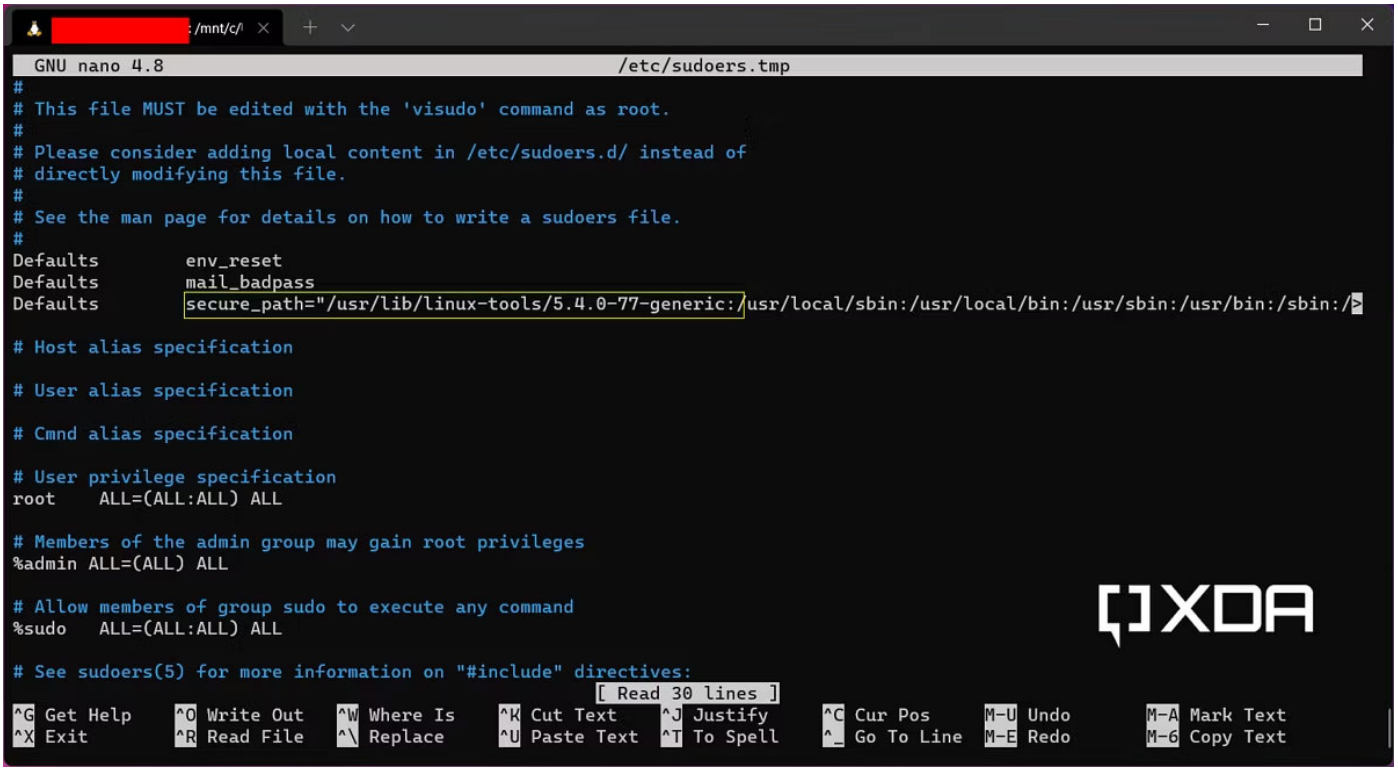
3. Install the user space tools for USB/IP and a database of USB hardware identifiers. If you're running Ubuntu on WSL, execute the following commands:

```
sudo apt install linux-tools-virtual hwdata
sudo update-alternatives --install /usr/local/bin/usbip usbip `ls /usr/lib/linux-tools/*/usbip | tail -n1` 20
```

   - You can opt for different meta packages (e.g. `linux-tools-generic`) as well, depending on the installed distro. However, you might need to adjust the path variable in the next step, according to the version number of the meta package.
   - For other distros, consult the documentation of the relevant package management tool.

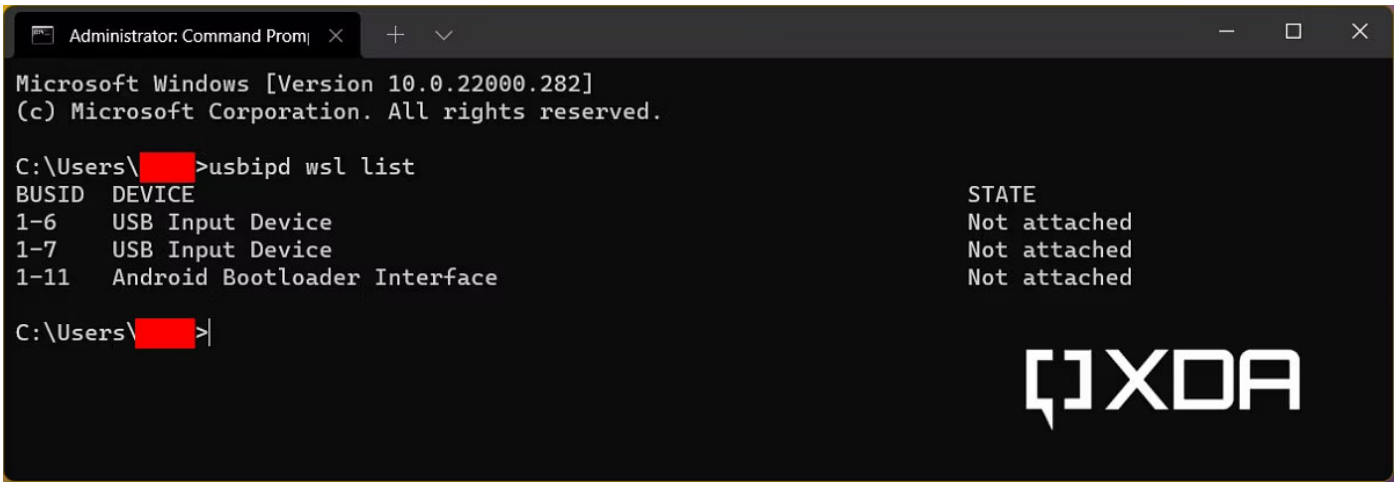
4. In case the symbolic link created using the `update-alternatives` command in the previous step didn't work for you, you can manually modify the sudo options to allow the root user to find the `usbip` command. To do so, edit the `/etc/sudoers` file using `sudo visudo` and add `/usr/lib/linux-tools/<version-number>-generic` to the beginning of `secure_path`. It should look like the following:

```
Defaults secure_path="/usr/lib/linux-tools/5.4.0-77-generic:/usr/local/sbin:..."
```



5. Open a new Terminal instance as Administrator and type the following command to list all the USB devices connected to Windows.

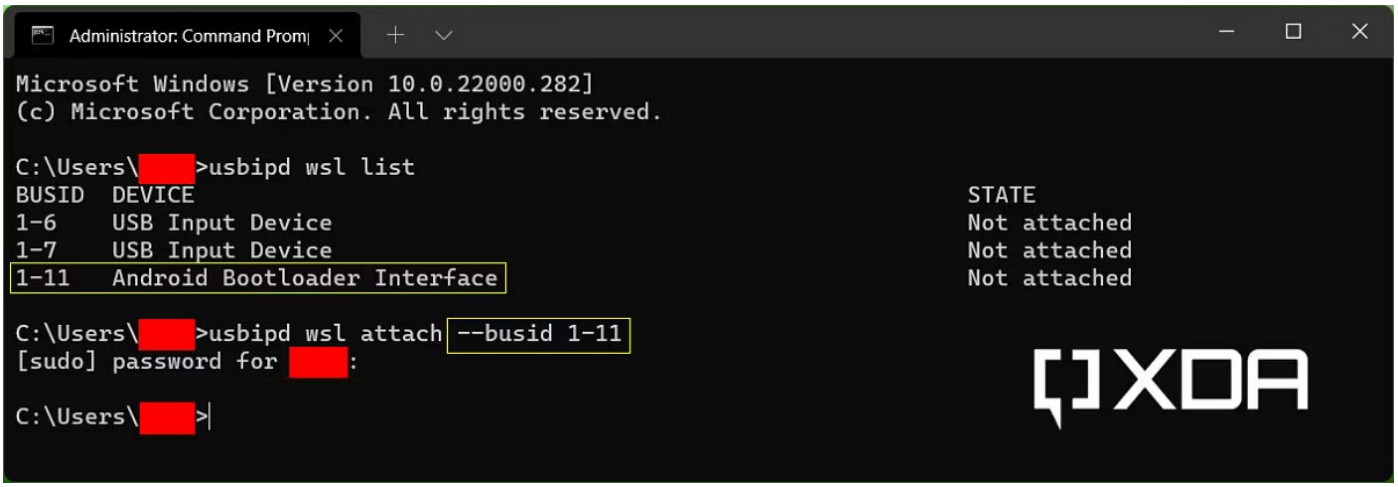
Link copied to clipboard



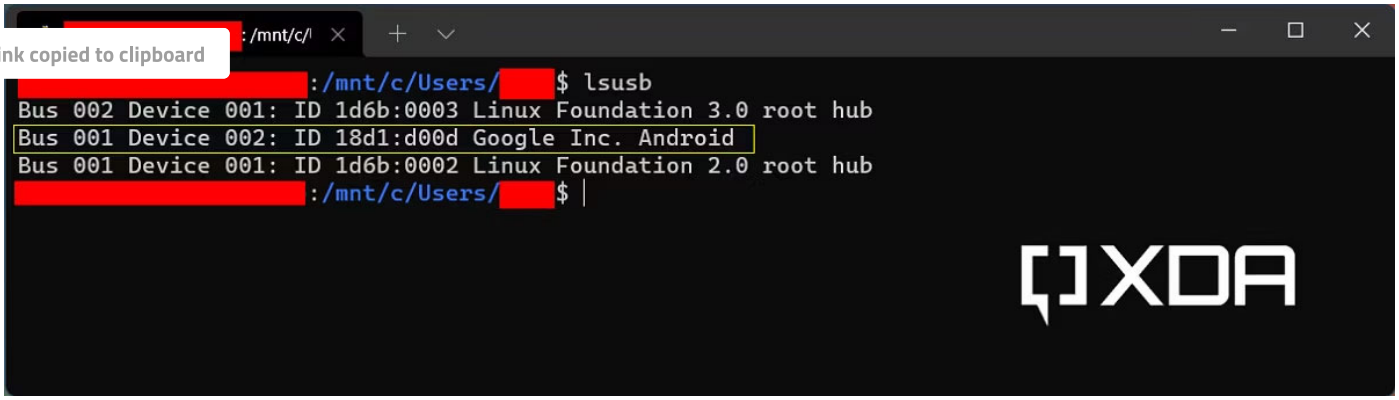
6. Note down the BUSID value of the USB device you want to attach to WSL. Next, run the following command to initiate the attachment process. You'll be prompted by WSL for a password to run a sudo command.

```
usbipd wsl attach --busid <busid>
```

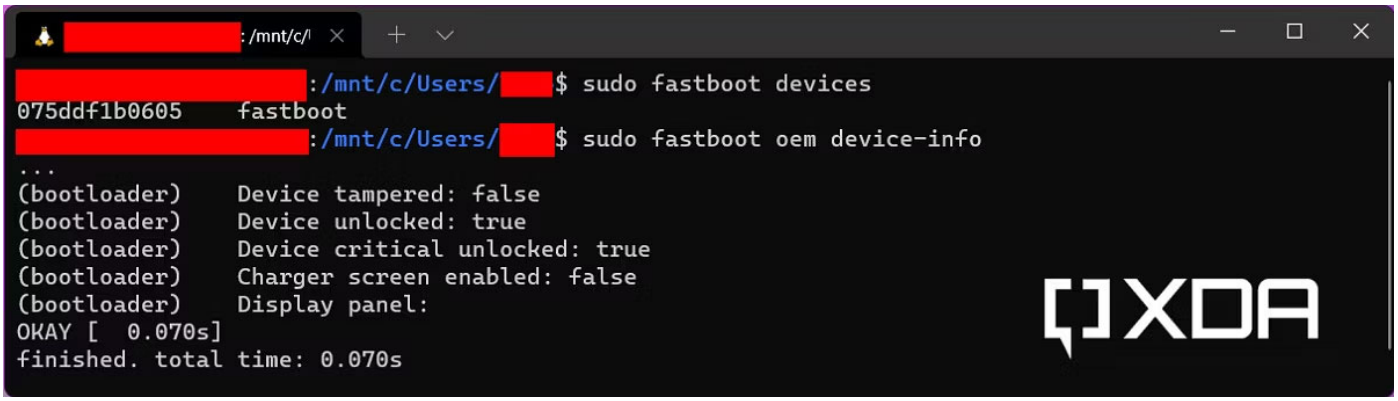
For example, here we're attaching an Android device's Fastboot interface to WSL from the host Windows 11 instance:



7. Now open a new WSL 2 instance and run the `lsusb` command to see available USB devices. It should list the attached device(s).



8. That's it! You can now work with your USB device right from WSL.



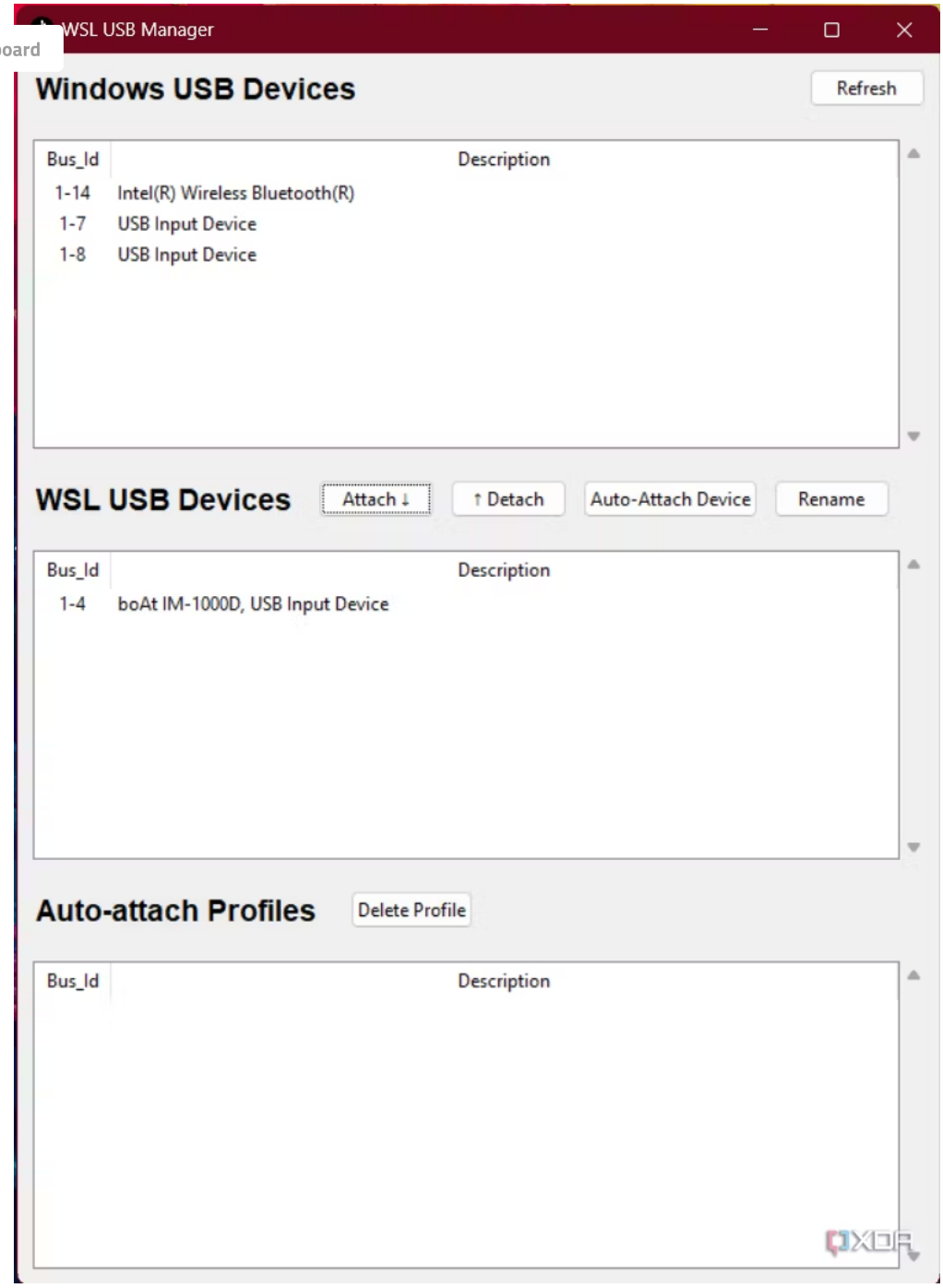
When you're done, you can detach the device using the command shown below. Once again, run it from a Terminal instance running as administrator.

```
usbipd wsl detach --busid <busid>
```

Is there any GUI alternative?

In case the command line interface isn't your forte, you can give wsl-usb-gui a try. It's an unofficial GUI wrapper around the usbipd-win binary, which greatly simplifies the USB device listing and interoperability between the host OS and the WSL instance.

1. Make sure the latest version of usbipd-win is installed and the WSL kernel is up-to-date.
2. Download the latest installer of wsl-usb-gui from the project's GitLab releases page.
3. Run the MSI file you downloaded.
   - Before installing the program, the installer also checks for the prerequisites and try to set them up if necessary. However, it would be better to keep everything updated by yourself to avoid any unforeseen compatibility issues.
4. Upon a successful installation, you should be able to find a newly created shortcut named **WSL USB** in the Start menu. Run the app from there.
5. Click on the **Refresh** button to populate the list of USB devices connected to the host OS.
6. Use the **Attach** and **Detach** buttons to unleash the full potential of USB pass-through.



From here on out, you're ready to go forth and tinker with all your USB devices you want in WSL. Our full guide will help you along the way, but once the USB passthrough is set up how you like it, it just fades into the background. To learn more, take a look at Microsoft's official documentation on connecting USB devices under WSL and usbipd-win's GitHub Wiki section.

Link copied to clipboard ct USB devices

Thanks to XDA Senior Member Some_Random_Username for the tip!

XDA Newsletter

Email Address

SUBSCRIBE

By subscribing, you agree to our **Privacy Policy** and may receive occasional deal communications; you can unsubscribe anytime.

Comments 6

Related Topics

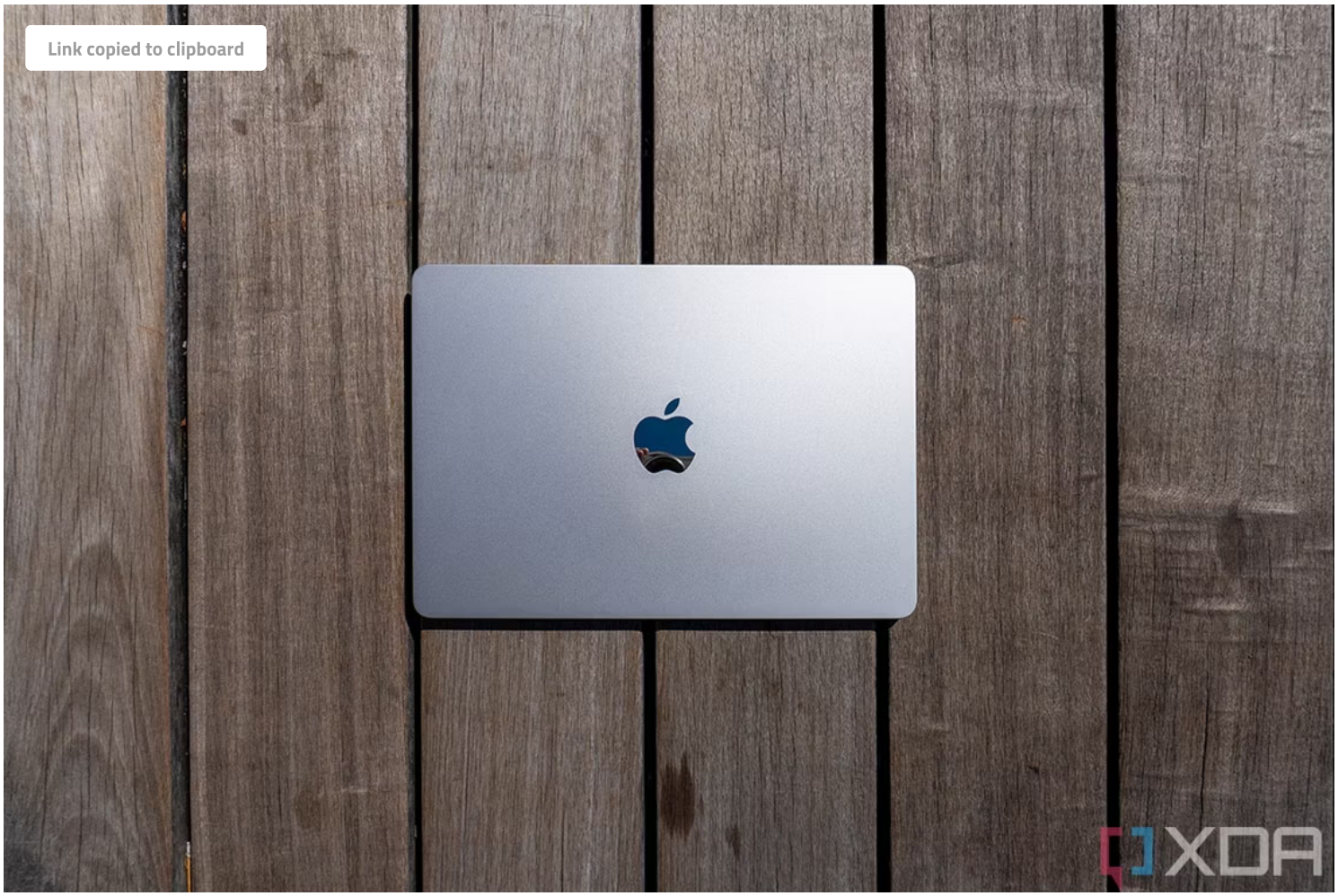
XDA TUTORIALS LINUX USB WINDOWS 11 WINDOWS SUBSYSTEM FOR LINUX

About The Author

Link copied to clipboard

(854 Articles Published)

DIY enthusiast (i.e. salvager of old PC parts). An avid user of Android since the Eclair days, Skanda also likes to follow the recent development trends in the world of single-board computing.



KEY TAKEAWAYS

- **Apple's M1 chip in the MacBook Air is powerful enough even in 2023, making an upgrade unnecessary for many users.**
- **The return of the Touch Bar would be a welcome addition to the MacBook as long as it exists alongside the function row.**
- **A touchscreen MacBook would be a natural evolution and should be considered, as it is already common in the Windows laptop space.**

Apple's M1 MacBook Air was announced back in November 2020, alongside the Mac Mini and the 13-inch MacBook Pro model. All these great Macs were among the first to be powered by Apple's M1 chip, which is plenty capable and powerful enough for most users even in 2023. I was one of the first ones to buy the M1 MacBook Air back in the day, and I still use it almost every day.

I've thought about upgrading to a new MacBook occasionally, but I never found a compelling reason to do so. A performance gain alone isn't convincing enough, especially when my current MacBook can already keep up with my needs well. Apple really needs to look at the competition and find new ways to make its existing users upgrade to newer products. The company does a decent job developing new features and upgrades for iPhones every other year, but MacBook upgrades have mostly been about performance for quite some time. I refuse to upgrade to a new MacBook unless Apple starts checking things off my wishlist.

3 Bring back the Touch Bar

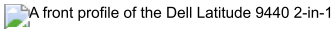
But make it better

Link copied to clipboard

I'm one of the few who liked the idea of having a Touch Bar on the MacBook. I completely agree with my colleague Brady Snyder on this, who rightly pointed out that the Touch Bar should've existed alongside the function row instead of entirely replacing it. I can exist without a function row on my keyboard, but I'm not a fan of how the physical escape key had to go to make space for the Touch Bar. I am rooting for the return of the Touch Bar as well, and I hope Apple realizes the mistake and brings it back the same way it brought the ports back on the new MacBook Pros.

2 Give me a touchscreen

Natural evolution



It may never happen, but the next MacBook Air or Pro model should have a touchscreen. They've already become quite common in the Windows laptop space, and I don't see why it's still not a thing on the MacBook. Using a 12.9-inch iPad Pro model is the closest thing you can have to a touchscreen MacBook, but iPadOS doesn't have a lot of macOS's most crucial features, and it's safe to say that it's not replacing a MacBook anytime soon.

There's definitely space for both products to co-exist on the market, so it's not like a touchscreen MacBook will affect iPad sales. If anything, the iPad's revenue is already dwindling, so you might as well make it easy for the Pro users to switch to a touchscreen MacBook while retaining the relatively smaller iPads for casual users who want a tablet. You can also still sell a non-touch MacBook for those who prefer a traditional laptop.

1 Switch up the design

Windows laptops are way cooler





There's nothing inherently wrong with the MacBook's design, but it could use some changes to keep things fresh. A facelift would certainly make an upgrade more compelling than packing the new internals in the same chassis, similar to how the "trash can" Mac Pro was replaced by the "cheese grater" option. It's not something that would make me want to go out of my way to buy one immediately, but it's just another thing to look forward to when considering an upgrade. Adding new colors is nice, but they're not nearly enough to warrant an upgrade.

The MacBook's design looks particularly boring and when you compare it with some cool Windows laptops that feature new form factors, like the Lenovo Yoga Book 9i or the HP Spectre Foldable. The Yoga Book 9i, in particular, made me go "wow" with its dual-screen design. I only expect Windows laptops to get better in 2024, and I would love to see some significant design changes making their way to the MacBook. If not a completely new design, at least give us some new textures or finishes like the unique forged carbon lid you get on the Legion 9i.

Closing thoughts

I love how Apple constantly improves MacBooks to make them more powerful. The M-series chips have been truly revolutionary, but I am rooting for more features and meaningful upgrades. The performance jump from an M1 to an M3 is quite significant, but that won't be enough for most people. I know I'm not alone when I say the M1 chip is plenty capable for everything I do on my MacBook Air. I doubt if I'll be upgrading anytime soon for the performance gains, so it's time Apple got started on other changes.

Related Topics

COMPUTING COMPUTING EDITORIAL APPLE MACBOOK AIR APPLE MACBOOK PRO

About The Author

Karthik Iyer

(795 Articles Published)

An evergreen writer at XDA Developers serving up reviews, buying guides, deals, and more on laptops, PC hardware, and smartphones that you care about. He's been covering the computing and smartphone industry for over seven years now and has worked with various brands to establish an authority in the technology space. Before joining XDA...