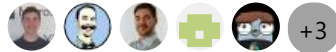


Mount a Linux disk in WSL 2 (preview)

09/27/2021 • 5 minutes to read •



In this article

[Prerequisites](#)

[Mounting an unpartitioned disk](#)

[Mounting a partitioned disk](#)

[Unmount the disk](#)

[Mount a VHD in WSL](#)

[Command line reference](#)

[Limitations](#)

If you want to access a Linux disk format that isn't supported by Windows, you can use WSL 2 to mount your disk and access its content. This tutorial will cover the steps to identify the disk and partition to attach to WSL2, how to mount them, and how to access them.

ⓘ Note

Administrator access is required to attach a disk to WSL 2. The WSL 2 `mount` command does not support mounting a disk (or partitions that belong to the disk) that is currently in use. `ws1 --mount` always attaches the entire disk even if only a partition is requested. You can't mount the Windows installation disk.

Prerequisites

You will need to be on Windows 11 Build 22000 or higher to access this feature. You can join the [Windows Insiders Program](#) to get the latest preview builds.

Mounting an unpartitioned disk

In this simplest case, if you have a disk that doesn't have any partitions, you can mount it directly using the `ws1 --mount` command. First you need to identify the disk.

1. **Identify the disk** - To list the available disks in Windows, run:

```
PowerShell
```



```
GET-WMIOBJECT -query "SELECT * from Win32_DiskDrive"
```

The disks paths are available under the 'DeviceID' columns. Usually under the `\\.\PHYSICALDRIVE*` format.

2. **Mount the disk** - Using PowerShell, you can mount the disk using the Disk path discovered above, run:

PowerShell



```
wsl --mount <DiskPath>
```

```
craig@Craig-Alienware: /mnt/v x + v
PS E:\wslDistroStorage\Ubuntu2004> GET-WMIOBJECT -query "SELECT * from Win32_DiskDrive"

Partitions : 1
DeviceID    : \\.\PHYSICALDRIVE0
Model       : Samsung SSD 970 EVO Plus 500GB
Size        : 500105249280
Caption     : Samsung SSD 970 EVO Plus 500GB

Partitions : 1
DeviceID    : \\.\PHYSICALDRIVE1
Model       : ST2000DM001-1CH164
Size        : 2000396321280
Caption     : ST2000DM001-1CH164

Partitions : 3
DeviceID    : \\.\PHYSICALDRIVE2
Model       : PM9A1 NVMe Samsung 256GB
Size        : 256052966400
Caption     : PM9A1 NVMe Samsung 256GB

Partitions : 0
DeviceID    : \\.\PHYSICALDRIVE3
Model       : Microsoft Virtual Disk
Size        : 322118415360
Caption     : Microsoft Virtual Disk

PS E:\wslDistroStorage\Ubuntu2004> wsl --mount \\.\PHYSICALDRIVE3
The disk \\.\PHYSICALDRIVE3 was successfully mounted under the name 'PHYSICALDRIVE3'. The mountpoint can be found under the path p
ointed to by the automount setting (default: /mnt/wsl).
To unmount and detach the disk, run 'wsl --unmount \\.\PHYSICALDRIVE3'.
PS E:\wslDistroStorage\Ubuntu2004> wsl
craig@Craig-Alienware:/mnt/e/wslDistroStorage/Ubuntu2004$ cd /mnt/wsl/PHYSICALDRIVE3/
craig@Craig-Alienware:/mnt/wsl/PHYSICALDRIVE3$ ls
bin  dev  home  lib  lib64  lost+found  mnt  proc  run  snap  sys  usr  wslHKjNMD  wslKEAFMJ  wslcnleED  wslolnend
boot  etc  init  lib32  libx32  media  opt  root  sbin  srv  tmp  var  wslJInHfN  wslKFeiGO  wslfCnNoM  wslpjNEiK
craig@Craig-Alienware:/mnt/wsl/PHYSICALDRIVE3$
```

Mounting a partitioned disk

If you have a disk that you aren't sure what file format it is in, or what partitions it has, you can follow the steps below to mount it.

1. **Identify the disk** - To list the available disks in Windows, run:

PowerShell



```
GET-WMIOBJECT -query "SELECT * from Win32_DiskDrive"
```

The disks paths are listed after 'DeviceID', usually in the `\\.\PHYSICALDRIVE*` format.

2. List and select the partitions to mount in WSL 2 - Once the disk is identified, run:

PowerShell

 Copy


```
wsl --mount <DiskPath> --bare
```

This will make the disk available in WSL 2. (In the case of our example, the

`<DiskPath>` is `\\.\PHYSICALDRIVE*`.

3. Once attached, the partition can be listed by running the following command inside WSL 2:

Bash

 Copy

```
lsblk
```

This will display the available block devices and their partitions.

Inside Linux, a block device is identified as `/dev/<Device><Partition>`. For example, `/dev/sdb3`, is the partition number 3 of disk `sdb`.

Example output:

Bash

 Copy

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
sdb	8:16	0	1G	0	disk	
├sdb2	8:18	0	50M	0	part	
├sdb3	8:19	0	873M	0	part	
└sdb1	8:17	0	100M	0	part	
sdc	8:32	0	256G	0	disk	/
sda	8:0	0	256G	0	disk	

Identifying the filesystem type

If you don't know the type of filesystem of a disk or partition, you can use this command:

PowerShell

 Copy

```
blkid <BlockDevice>
```

This will output the detected filesystem type (under the `TYPE="<Filesystem>"` format).

Mount the selected partitions

Once you have identified the partitions you want to mount, run this command on each partition:

PowerShell

 Copy

```
wsl --mount <DiskPath> --partition <PartitionNumber> --type <Filesystem>
```

ⓘ Note

If you wish to mount the entire disk as a single volume (i.e. if the disk isn't partitioned), `--partition` can be omitted.

If omitted, the default filesystem type is "ext4".

Access the disk content

Once mounted, the disk can be accessed under the path pointed to by the config value: `automount.root`. The default value is `/mnt/wsl`.

From Windows, the disk can be accessed from File Explorer by navigating to: `\\wsl$\\<Distro>\\<Mountpoint>` (pick any Linux distribution).

Unmount the disk

If you want to unmount and detach the disk from WSL 2, run:

PowerShell

 Copy

```
wsl --unmount <DiskPath>
```

Mount a VHD in WSL

You can also mount virtual hard disk files (VHD) into WSL using `wsl --mount`. To do this, you first need to mount the VHD into Windows using the [Mount-VHD](#) command in Windows. Be sure to run this command with administrator privileges. Below is an example where we use this command, and also output the disk path. Be sure to replace `<pathToVHD>` with your actual VHD path.

PowerShell

 Copy

```
Write-Output "\\.\PhysicalDrive$((Mount-VHD -Path <pathToVHD> -PassThru |
Get-Disk).Number)"
```

You can use the output above to obtain the disk path for this VHD and mount that into WSL following the instructions in the previous section.

You can also use this technique to mount and interact with the virtual hard disks of other WSL distros, as each WSL 2 distro is stored via a virtual hard disk file called: `ext4.vhdx`. By default the VHDs for WSL 2 distros are stored in this path: `C:\Users\[user]\AppData\Local\Packages\[distro]\LocalState\[distroPackageName]`, please exercise caution accessing these system files, this is a power user workflow. Make sure to run `wsl --shutdown` before interacting with this disk to ensure the disk is not in use.

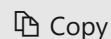
```
craig@Craig-Alienware: /mnt/v
PS E:\wslDistroStorage\Ubuntu2004> Write-Output "\\.\PhysicalDrive$((Mount-VHD -Path ./ext4.vhdx -PassThru |
Get-Disk).Number)"
\\.\PhysicalDrive3
PS E:\wslDistroStorage\Ubuntu2004> wsl --mount \\.\PhysicalDrive3
The disk \\.\PhysicalDrive3 was successfully mounted under the name 'PhysicalDrive3'. The mountpoint can be found under the path p
ointed to by the automount setting (default: /mnt/wsl).
To unmount and detach the disk, run 'wsl --unmount \\.\PhysicalDrive3'.
PS E:\wslDistroStorage\Ubuntu2004> wsl
craig@Craig-Alienware:/mnt/e/wslDistroStorage/Ubuntu2004$ cd /mnt/wsl/PhysicalDrive3/
craig@Craig-Alienware:/mnt/wsl/PhysicalDrive3$ ls
bin  dev  home  lib  lib64  lost+found  mnt  proc  run  snap  sys  usr  wslHKjNMD  wslKEAFMJ  wslcnleED  wslolnend
boot  etc  init  lib32  libx32  media  opt  root  sbin  srv  tmp  var  wslJInHfN  wslKFeiGO  wslfCNoM  wslpjNEiK
craig@Craig-Alienware:/mnt/wsl/PhysicalDrive3$
craig@Craig-Alienware:/mnt/wsl/PhysicalDrive3$
```

Command line reference

Mounting a specific filesystem

By default, WSL 2 will attempt to mount the device as ext4. To specify another filesystem, run:

PowerShell

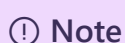


```
wsl --mount <DiskPath> -t <FileSystem>
```

For example, to mount a disk as fat, run:



```
wsl --mount <Diskpath> -t vfat
```



Note

To list the available filesystems in WSL2, run: `cat /proc/filesystems`

When a disk has been mounted via WSL2 (Linux file system), it is no longer available to mount via an ext4 driver on the Windows file system.

Mounting a specific partition

By default, WSL 2 attempts to mount the entire disk. To mount a specific partition, run:

```
ws1 --mount <Diskpath> -p <PartitionIndex>
```

This only works if the disk is either MBR (Master Boot Record) or GPT (GUID Partition Table). [Read about partition styles - MBR and GPT.](#)

Specifying mount options

To specify mount options, run:

```
ws1 --mount <DiskPath> -o <MountOptions>
```

Example:

```
ws1 --mount <DiskPath> -o "data=ordered"
```

ⓘ Note

Only filesystem specific options are supported at this time. Generic options such as `ro`, `rw`, `noatime`, ... are not supported.

Attaching the disk without mounting it

If the disk scheme isn't supported by any of the above options, you can attach the disk to WSL 2 without mounting it by running:

```
PowerShell
```

```
wsl --mount <DiskPath> --bare
```

This will make the block device available inside WSL 2 so it can be mounted manually from there. Use `lsblk` to list the available block devices inside WSL 2.

Detaching a disk

To detach a disk from WSL 2, run:

PowerShell



```
wsl --unmount [DiskPath]
```

If `Diskpath` is omitted, all attached disks are unmounted and detached.

ⓘ Note

If one disk fails to unmount, WSL 2 can be forced to exit by running `wsl --shutdown`, which will detach the disk.

Limitations

- At this time, only entire disks can be attached to WSL 2, meaning that it's not possible to attach only a partition. Concretely, this means that it's not possible to use `wsl --mount` to read a partition on the boot device, because that device can't be detached from Windows.
- USB flash drives and SD cards are not supported at this time and will fail to attach to WSL 2. USB disks are supported though.
- Only filesystems that are natively supported in the kernel can be mounted by `wsl --mount`. This means that it's not possible to use installed filesystem drivers (such as ntfs-3g for example) by calling `wsl --mount`.

Is this page helpful?

Yes No

Recommended content

[Troubleshooting Windows Subsystem for Linux](#)

Provides detailed information about common errors and issues people run into while running Linux on the Windows Subsystem for Linux.

[Manual installation steps for older versions of WSL](#)

Step by step instructions to manually install WSL on older versions of Windows, rather than using the wsl install command.

[Windows Subsystem for Linux Documentation](#)

Explore the Windows Subsystem for Linux documentation.

[Release Notes for WSL](#)

Read release notes for the Windows Subsystem for Linux. These release notes include fixed issues and are updated weekly.