Windows Subsystem for Linux Installation Guide for Windows 10

09/06/2021 • 11 minutes to read • +27

In this article

Simplified Installation for Windows Insiders
Manual Installation Steps
Step 1 - Enable the Windows Subsystem for Linux
Step 2 - Check requirements for running WSL 2
Step 3 - Enable Virtual Machine feature
Step 4 - Download the Linux kernel update package
Step 5 - Set WSL 2 as your default version
Step 6 - Install your Linux distribution of choice
Install Windows Terminal (optional)
Set your distribution version to WSL 1 or WSL 2
Troubleshooting installation

There are two options available for installing Windows Subsystem for Linux (WSL):

- **Simplified install** *(preview release)*: `wsl --install`

  The `wsl --install` simplified install command requires that you join the Windows Insiders Program and install a preview build of Windows 10 (OS build 20262 or higher), but eliminates the need to follow the manual install steps. All you need to do is open a command window with administrator privileges and run `wsl --install`, after a restart you will be ready to use WSL.

- **Manual install**: Follow the six steps listed below.

  The manual install steps for WSL are listed below and can be used to install Linux on any version of Windows 10.

> **Note**
>
> If you run into an issue during the install process, check the **Troubleshooting installation** section at the bottom of this page.

Simplified Installation for Windows Insiders

The installation process for Windows Subsystem for Linux has been significantly improved in the latest Windows Insiders preview builds of Windows 10, replacing the manual steps below with a

single command.

In order to use the `wsl --install` simplified install command, you must:

- Join the Windows Insiders Program
- Install a preview build of Windows 10 (OS build 20262 or higher).
- Open a command line windows with Administrator privileges

Once those requirements are met, to install WSL:

- Enter this command in the command line you've opened in Admin mode: `wsl.exe --install`
- Restart your machine

The first time you launch a newly installed Linux distribution, a console window will open and you'll be asked to wait for files to de-compress and be stored on your PC. All future launches should take less than a second.

You will then need to create a user account and password for your new Linux distribution.

CONGRATULATIONS! You've successfully installed and set up a Linux distribution that is completely integrated with your Windows operating system!

The --install command performs the following actions:

- Enables the optional WSL and Virtual Machine Platform components
- Downloads and installs the latest Linux kernel
- Sets WSL 2 as the default
- Downloads and installs a Linux distribution *(reboot may be required)*

By default, the installed Linux distribution will be Ubuntu. This can be changed using `wsl --install -d <Distribution Name>`. *(Replacing `<Distribution Name>` with the name of your desired distribution.)* Additional Linux distributions may be added to your machine after the initial install using the `wsl --install -d <Distribution Name>` command.

To see a list of available Linux distributions, enter `wsl --list --online`.

Manual Installation Steps

If you are not on a Windows Insiders build, the features required for WSL will need to be enabled manually following the steps below.

Step 1 - Enable the Windows Subsystem for Linux

You must first enable the "Windows Subsystem for Linux" optional feature before installing any Linux distributions on Windows.

Open PowerShell as Administrator and run:

PowerShell Copy

```
dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all
/norestart
```

We recommend now moving on to step #2, updating to WSL 2, but if you wish to only install WSL 1, you can now **restart** your machine and move on to Step 6 - Install your Linux distribution of choice. To update to WSL 2, **wait to restart** your machine and move on to the next step.

Step 2 - Check requirements for running WSL 2

To update to WSL 2, you must be running Windows 10.

- For x64 systems: **Version 1903** or higher, with **Build 18362** or higher.
- For ARM64 systems: **Version 2004** or higher, with **Build 19041** or higher.
- Builds lower than 18362 do not support WSL 2. Use the Windows Update Assistant to update your version of Windows.

To check your version and build number, select **Windows logo key + R**, type **winver**, select **OK**. Update to the latest Windows version in the Settings menu.

> **Note**
>
> If you are running Windows 10 version 1903 or 1909, open "Settings" from your Windows menu, navigate to "Update & Security" and select "Check for Updates". Your Build number must be 18362.1049+ or 18363.1049+, with the minor build # over .1049. Read more: **WSL 2 Support is coming to Windows 10 Versions 1903 and 1909**. See the **troubleshooting instructions**.

Step 3 - Enable Virtual Machine feature

Before installing WSL 2, you must enable the **Virtual Machine Platform** optional feature. Your machine will require virtualization capabilities to use this feature.

Open PowerShell as Administrator and run:

PowerShell Copy

```
dism.exe /online /enable-feature /featurename:VirtualMachinePlatform /all /norestart
```

Restart your machine to complete the WSL install and update to WSL 2.

Step 4 - Download the Linux kernel update package

1. Download the latest package:

   - WSL2 Linux kernel update package for x64 machines

   > **Note**
   >
   > If you're using an ARM64 machine, please download the **ARM64 package** instead. If you're not sure what kind of machine you have, open Command Prompt or PowerShell and enter: `systeminfo | find "System Type"`. **Caveat:** On non-English Windows versions, you might have to modify the search text, for example, in German it would be `systeminfo | find "Systemtyp"`.

2. Run the update package downloaded in the previous step. (Double-click to run - you will be prompted for elevated permissions, select 'yes' to approve this installation.)

Once the installation is complete, move on to the next step - setting WSL 2 as your default version when installing new Linux distributions. (Skip this step if you want your new Linux installs to be set to WSL 1).

> **Note**
>
> For more information, read the article **changes to updating the WSL2 Linux kernel**, available on the **Windows Command Line Blog**.

Step 5 - Set WSL 2 as your default version

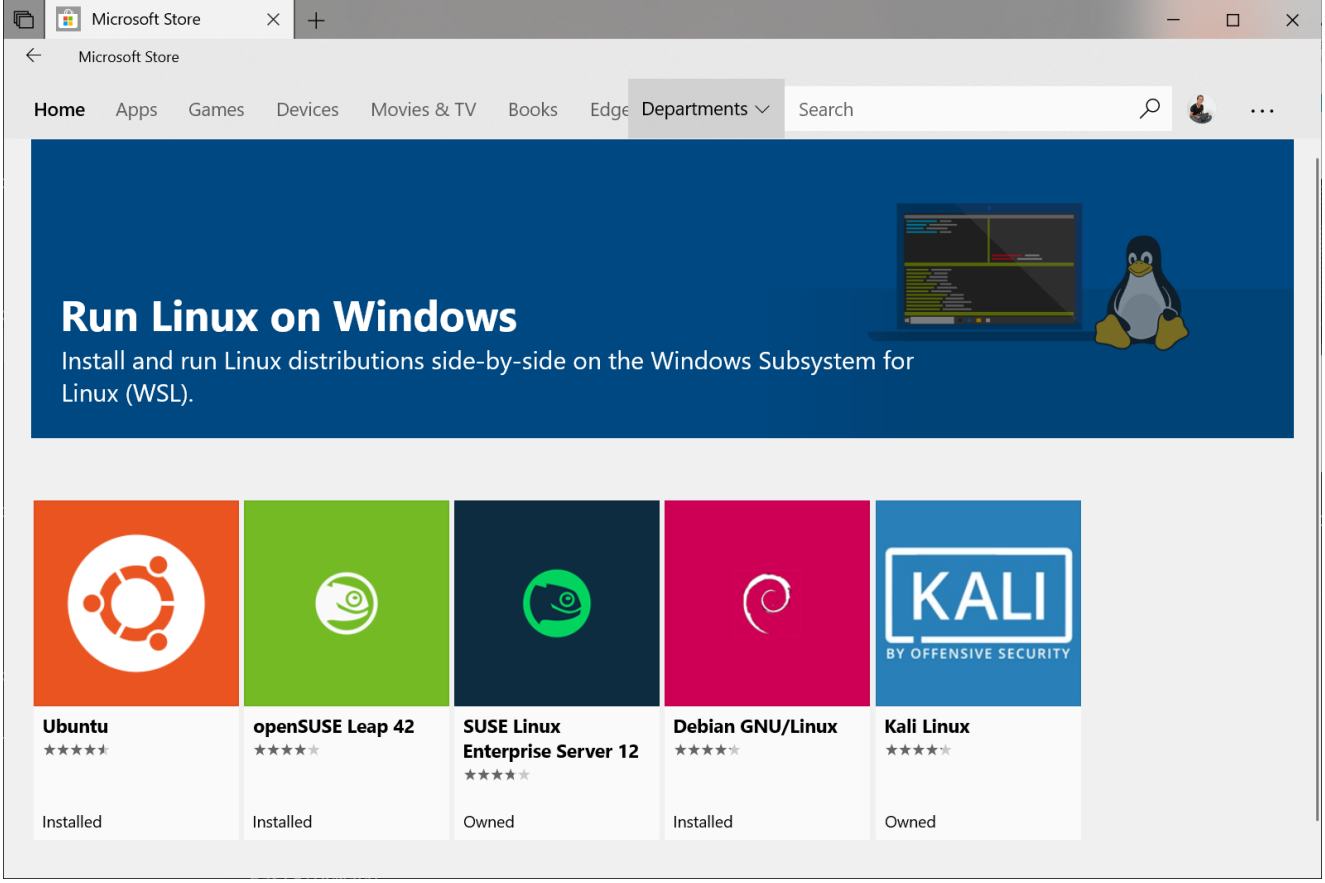
Open PowerShell and run this command to set WSL 2 as the default version when installing a new Linux distribution:

PowerShell | Copy

```
wsl --set-default-version 2
```

Step 6 - Install your Linux distribution of choice

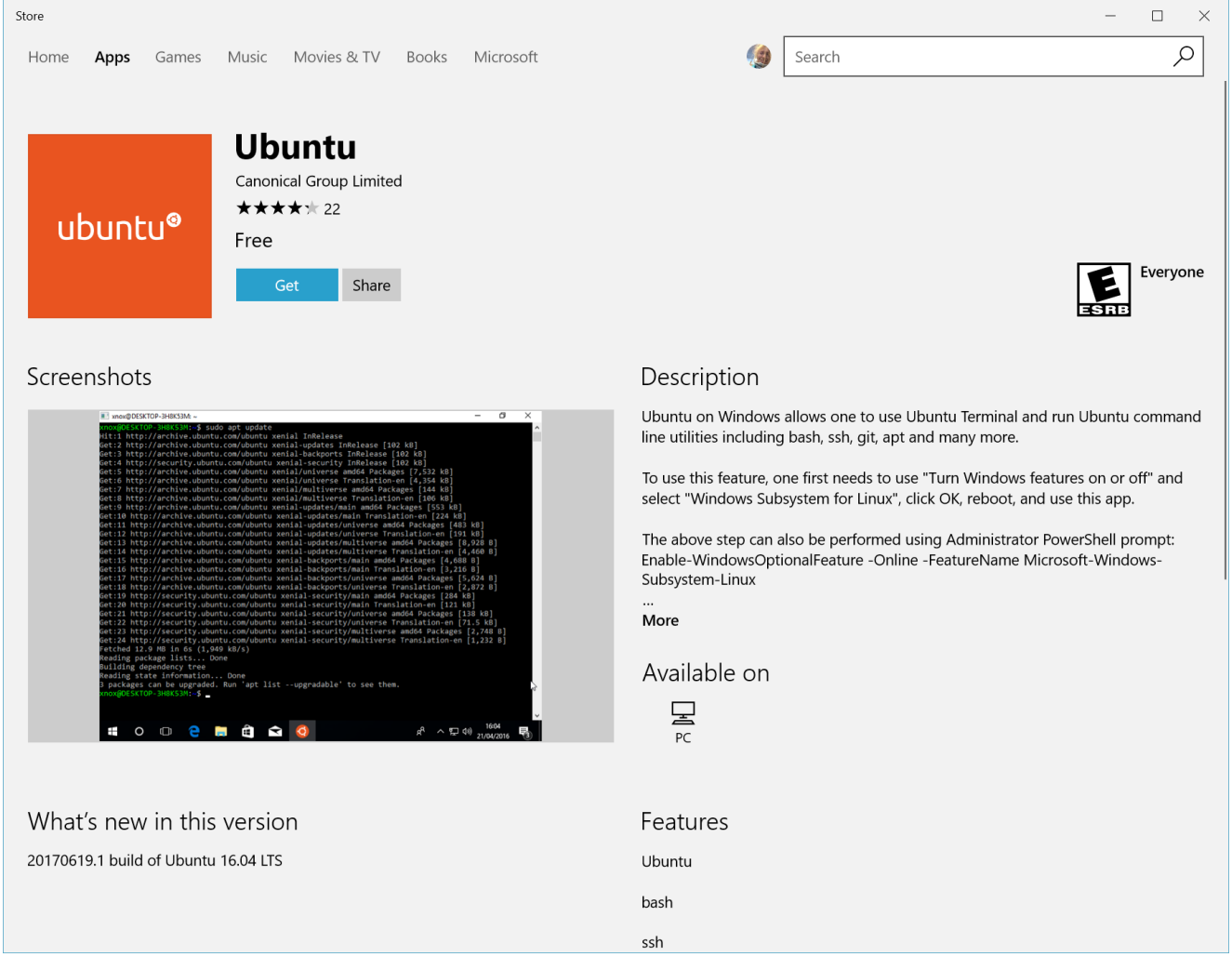
1. Open the Microsoft Store and select your favorite Linux distribution.



The following links will open the Microsoft store page for each distribution:

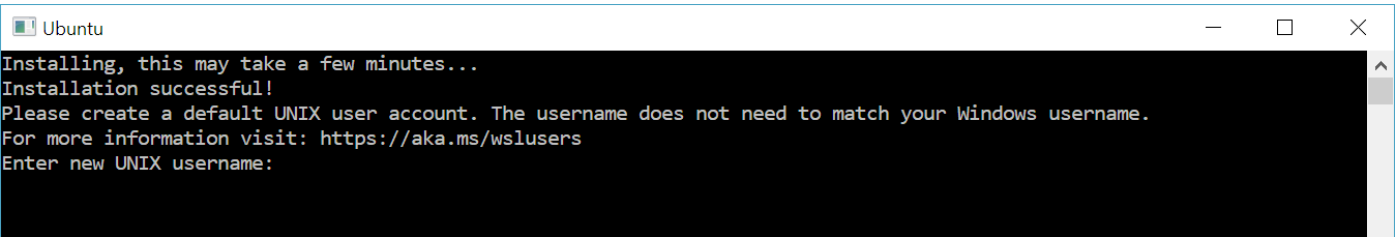
- Ubuntu 18.04 LTS
- Ubuntu 20.04 LTS
- openSUSE Leap 15.1
- SUSE Linux Enterprise Server 12 SP5
- SUSE Linux Enterprise Server 15 SP1
- Kali Linux
- Debian GNU/Linux
- Fedora Remix for WSL
- Pengwin
- Pengwin Enterprise
- Alpine WSL

2. From the distribution's page, select "Get".



The first time you launch a newly installed Linux distribution, a console window will open and you'll be asked to wait for a minute or two for files to de-compress and be stored on your PC. All future launches should take less than a second.

You will then need to create a user account and password for your new Linux distribution.

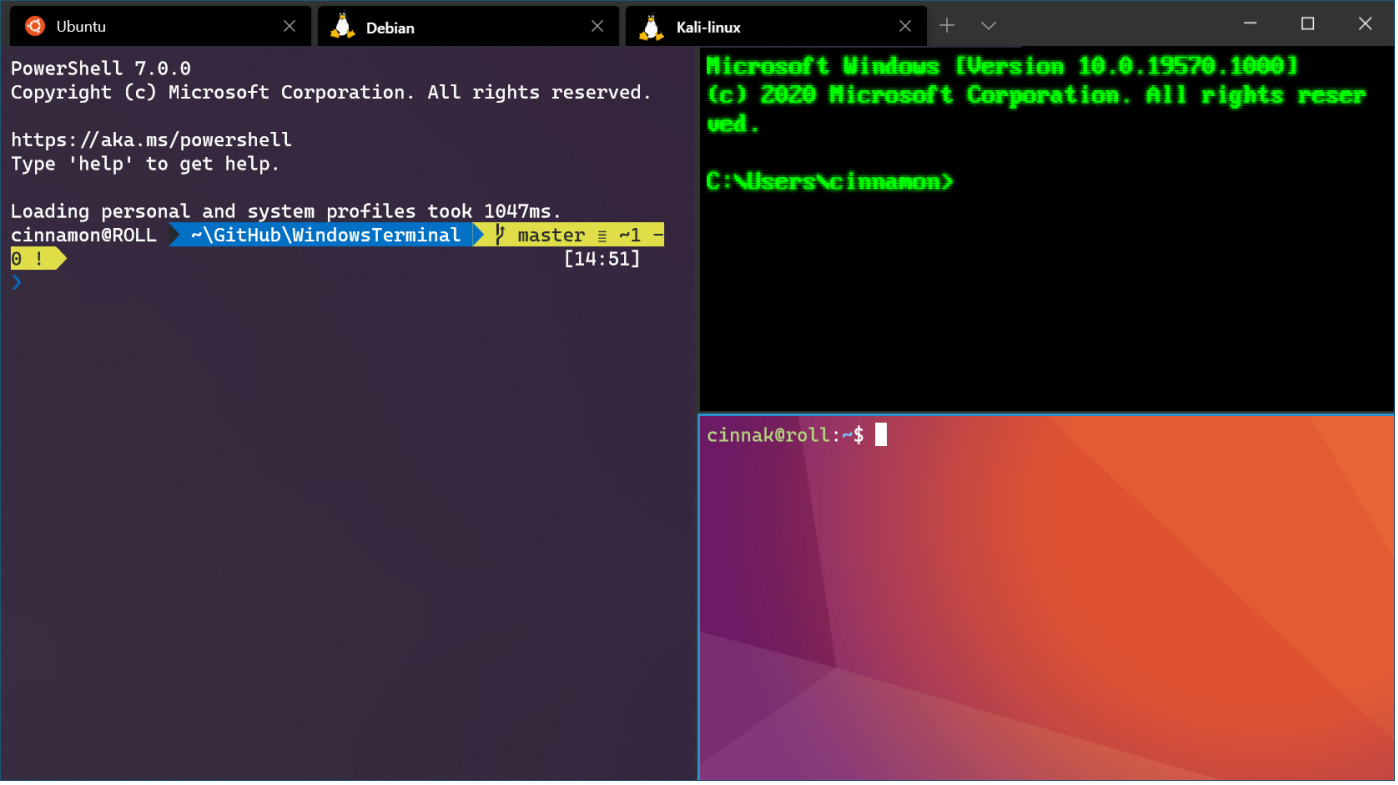


CONGRATULATIONS! You've successfully installed and set up a Linux distribution that is completely integrated with your Windows operating system!

Install Windows Terminal (optional)

Windows Terminal enables multiple tabs (quickly switch between multiple Linux command lines, Windows Command Prompt, PowerShell, Azure CLI, etc), create custom key bindings (shortcut keys for opening or closing tabs, copy+paste, etc.), use the search feature, and custom themes (color schemes, font styles and sizes, background image/blur/transparency). Learn more.

Install Windows Terminal.



Set your distribution version to WSL 1 or WSL 2

You can check the WSL version assigned to each of the Linux distributions you have installed by opening the PowerShell command line and entering the command (only available in Windows Build 18362 or higher): `wsl -l -v`

PowerShell Copy

```
wsl --list --verbose
```

To set a distribution to be backed by either version of WSL please run:

PowerShell Copy

```
wsl --set-version <distribution name> <versionNumber>
```

Make sure to replace `<distribution name>` with the actual name of your distribution and `<versionNumber>` with the number '1' or '2'. You can change back to WSL 1 at anytime by running the same command as above but replacing the '2' with a '1'.

> **Note**
>
> The update from WSL 1 to WSL 2 may take several minutes to complete depending on the size of your targeted distribution. If you are running an older (legacy) installation of WSL 1 from Windows 10 Anniversary Update or Creators Update, you may encounter an update error. Follow these instructions to **uninstall and remove any legacy distributions**.

If `wsl --set-default-version` results as an invalid command, enter `wsl --help`. If the `--set-default-version` is not listed, it means that your OS doesn't support it and you need to update to version 1903, Build 18362 or higher. If you are on Build 19041 for ARM64, this command may fail when using PowerShell in which case you can use a *Command Prompt* instead to issue the `wsl.exe` command.

If you see this message after running the command: `WSL 2 requires an update to its kernel component. For information please visit https://aka.ms/wsl2kernel`. You still need to install the MSI Linux kernel update package.

Additionally, if you want to make WSL 2 your default architecture you can do so with this command:

PowerShell | Copy

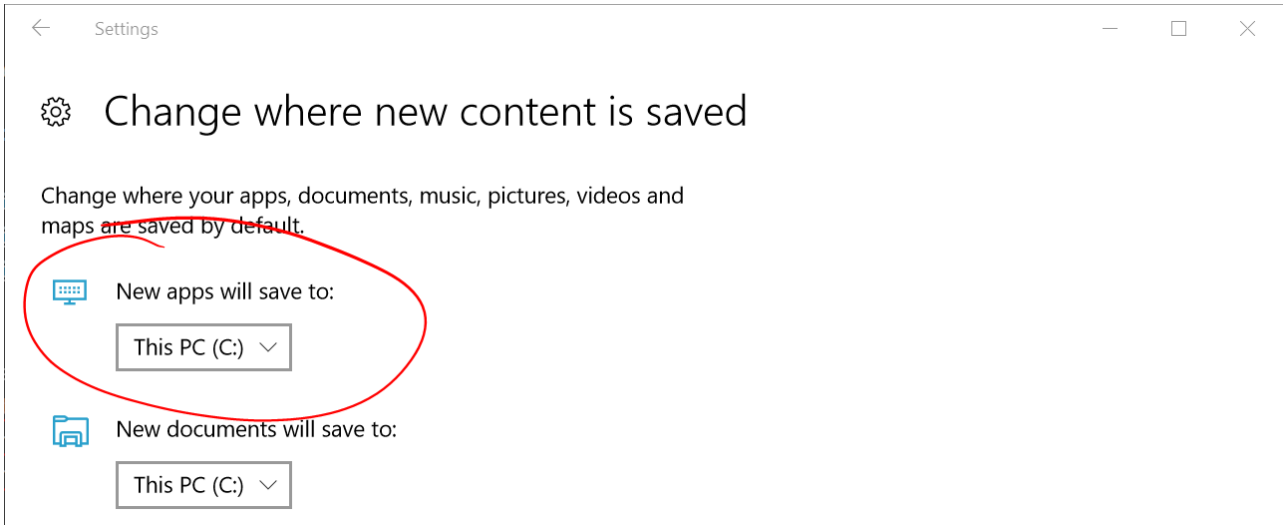
```
wsl --set-default-version 2
```

This will set the version of any new distribution installed to WSL 2.

Troubleshooting installation

Below are related errors and suggested fixes. Refer to the WSL troubleshooting page for other common errors and their solutions.

- **Installation failed with error 0x80070003**
  - The Windows Subsystem for Linux only runs on your system drive (usually this is your `C:` drive). Make sure that distributions are stored on your system drive:
  - Open **Settings** -> **System** --> **Storage** -> **More Storage Settings: Change where new content is saved**



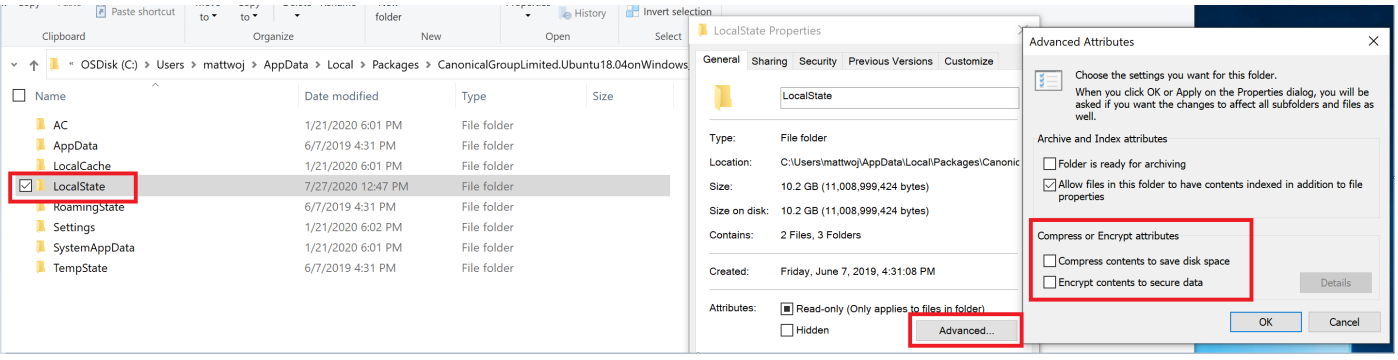
- **WslRegisterDistribution failed with error 0x8007019e**
  - The Windows Subsystem for Linux optional component is not enabled:
  - Open **Control Panel** -> **Programs and Features** -> **Turn Windows Feature on or off** -> Check **Windows Subsystem for Linux** or using the PowerShell cmdlet mentioned at the

beginning of this article.

- **Installation failed with error 0x80070003 or error 0x80370102**
  - Please make sure that virtualization is enabled inside of your computer's BIOS. The instructions on how to do this will vary from computer to computer, and will most likely be under CPU related options.
  - WSL2 requires that your CPU supports the Second Level Address Translation (SLAT) feature, which was introduced in Intel Nehalem processors (Intel Core 1st Generation) and AMD Opteron. Older CPUs (such as the Intel Core 2 Duo) will not be able to run WSL2, even if the Virtual Machine Platform is successfully installed.

- **Error when trying to upgrade: `Invalid command line option: wsl --set-version Ubuntu 2`**
  - Ensure that you have the Windows Subsystem for Linux enabled, and that you're using Windows Build version 18362 or higher. To enable WSL run this command in a PowerShell prompt with admin privileges: `Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux.`

- **The requested operation could not be completed due to a virtual disk system limitation. Virtual hard disk files must be uncompressed and unencrypted and must not be sparse.**
  - Deselect “Compress contents” (as well as “Encrypt contents” if that’s checked) by opening the profile folder for your Linux distribution. It should be located in a folder on your Windows file system, something like:
    `USERPROFILE%\AppData\Local\Packages\CanonicalGroupLimited...`
  - In this Linux distro profile, there should be a LocalState folder. Right-click this folder to display a menu of options. Select Properties > Advanced and then ensure that the “Compress contents to save disk space” and “Encrypt contents to secure data” checkboxes are unselected (not checked). If you are asked whether to apply this to just to the current folder or to all subfolders and files, select “just this folder” because you are only clearing the compress flag. After this, the `wsl --set-version` command should work.



ⓘ Note

In my case, the LocalState folder for my Ubuntu 18.04 distribution was located at C:\Users<my-user-name>\AppData\Local\Packages\CanonicalGroupLimited.Ubuntu18.04onWindows_79rhkp1fndgsc

Check **WSL Docs GitHub thread #4103** where this issue is being tracked for updated information.

- **The term 'wsl' is not recognized as the name of a cmdlet, function, script file, or operable program.**
  - Ensure that the Windows Subsystem for Linux Optional Component is installed. Additionally, if you are using an ARM64 device and running this command from PowerShell, you will receive this error. Instead run `wsl.exe` from PowerShell Core, or Command Prompt.

- **Error: Windows Subsystem for Linux has no installed distributions.**
  - If you receive this error after you have already installed WSL distributions:

    1. Run the distribution at least once before invoking it from the command line.
    2. Check whether you may be running separate user accounts. Running your primary user account with elevated permissions (in admin mode) should not result in this error, but you should ensure that you aren't accidentally running the built-in Administrator account that comes with Windows. This is a separate user account and will not show any installed WSL distributions by design. https://docs.microsoft.com/windows-hardware/manufacture/desktop/enable-and-disable-the-built-in-administrator-account
    3. The WSL executable is only installed to the native system directory. When you're running a 32-bit process on 64-bit Windows (or on ARM64, any non-native combination), the hosted non-native process actually sees a different System32 folder. (The one a 32-bit process sees on x64 Windows is stored on disk at \Windows\SysWOW64.) You can access the "native" system32 from a hosted process by looking in the virtual folder \Windows\sysnative. It won't actually be present on disk, mind you, but the filesystem path resolver will find it.

- **Error: This update only applies to machines with the Windows Subsystem for Linux.**
  - To install the Linux kernel update MSI package, WSL is required and should be enabled first. If it fails, it you will see the message: `This update only applies to machines with the Windows Subsystem for Linux.`
  - There are three possible reason you see this message:

    1. You are still in old version of Windows which doesn't support WSL 2. See step #2 for version requirements and links to update.

    2. WSL is not enabled. You will need to return to step #1 and ensure that the optional WSL feature is enabled on your machine.

    3. After you enabled WSL, a reboot is required for it to take effect, reboot your machine and try again.

- **Error: WSL 2 requires an update to its kernel component. For information please visit https://aka.ms/wsl2kernel .**

- If the Linux kernel package is missing in the %SystemRoot%\system32\lxss\tools folder, you will encounter this error. Resolve it by installing the Linux kernel update MSI package in step #4 of these installation instructions. You may need to uninstall the MSI from 'Add or Remove Programs', and install it again.

Is this page helpful?

Yes No