

Text

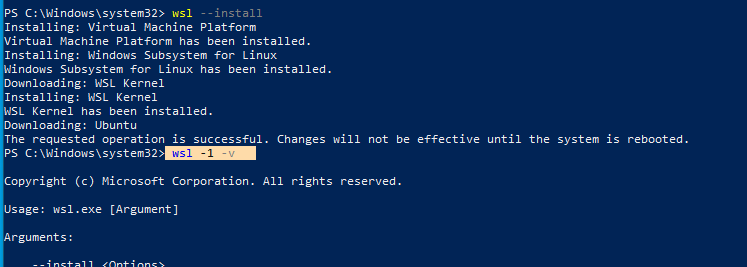
Description automatically generated

Text

Description automatically generated with medium confidence

Install Linux on Windows with WSL

wsl –install



**Install WSL command**

You can now install everything you need to run WSL with a single command. Open PowerShell or Windows Command Prompt in **administrator** mode by right-clicking and selecting "Run as administrator", enter the wsl --install command, then restart your machine.

PowerShellCopy

wsl --install

This command will enable the features necessary to run WSL and install the Ubuntu distribution of Linux. ([This default distribution can be changed](https://learn.microsoft.com/en-us/windows/wsl/basic-commands#install)).

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 machine. All future launches should take less than a second.

Note

The above command only works if WSL is not installed at all, if you run wsl --install and see the WSL help text, please try running wsl --list --online to see a list of available distros and run wsl --install -d <DistroName> to install a distro. To uninstall WSL, see Uninstall legacy version of WSL or unregister or uninstall a Linux distribution.

**Change the default Linux distribution installed**

By default, the installed Linux distribution will be Ubuntu. This can be changed using the -d flag.

* To change the distribution installed, enter: wsl --install -d <Distribution Name>. Replace <Distribution Name> with the name of the distribution you would like to install.
* To see a list of available Linux distributions available for download through the online store, enter: wsl --list --online or wsl -l -o.
* To install additional Linux distributions after the initial install, you may also use the command: wsl --install -d <Distribution Name>.

## Check which version of WSL you are running

You can list your installed Linux distributions and check the version of WSL each is set to by entering the command: wsl -l -v in PowerShell or Windows Command Prompt.

Installed Docker Desktop on Windows

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

LEFT TO DO:

Create a directory named learn-terraform-docker-container.

$ mkdir learn-terraform-docker-container

Navigate into the working directory.

$ cd learn-terraform-docker-container

In the working directory, create a file called main.tf and paste the following Terraform configuration into it.

Mac or LinuxWindows

terraform {

required\_providers {

docker = {

source = "kreuzwerker/docker"

version = ">= 2.13.0"

}

}

}

provider "docker" {

host = "npipe:////.//pipe//docker\_engine"

}

resource "docker\_image" "nginx" {

name = "nginx:latest"

keep\_locally = false

}

resource "docker\_container" "nginx" {

image = docker\_image.nginx.latest

name = "tutorial"

ports {

internal = 80

external = 8000

}

}

Text

Description automatically generated

Initialize the project, which downloads a plugin called a provider that lets Terraform interact with Docker.

$ terraform init

Copy

Provision the NGINX server container with apply. When Terraform asks you to confirm type yes and press ENTER.

$ terraform apply

Copy

Verify the existence of the NGINX container by visiting [localhost:8000](http://localhost:8000/) in your web browser or running docker ps to see the container.

Graphical user interface, application

Description automatically generated

$ docker ps

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

425d5ee58619 e791337790a6 "nginx -g 'daemon of…" 20 seconds ago Up 19 seconds 0.0.0.0:8000->80/tcp tutorial

To stop the container, run terraform destroy.

$ terraform destroy

Copy

You've now provisioned and destroyed an NGINX webserver with Terraform.

Graphical user interface, text, application, email

Description automatically generated