

Gatsby on Linux

This guide assumes you already have a native installation of Linux on your machine. The following steps walk through how to install Node.js and associated dependencies.

Ubuntu, Debian, and other apt based distros

Begin by updating and upgrading.

```
sudo apt update
sudo apt -y upgrade
```

Install cURL which allows you to transfer data and download additional dependencies.

```
sudo apt install curl
```

Once `curl` is installed, you can use it to install `nvm`, which will manage `node` and all its associated versions.

```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh | bash
```

Note that this is the current stable release of `nvm`. Full installation instructions and troubleshooting can be found at the `nvm` GitHub page

When `nvm` is installed, it does not default to a particular `node` version. You'll need to install the version you want and give `nvm` instructions to use it. This example uses the latest release of version 16, but more recent version numbers can be used instead.

```
nvm install 16
nvm use 16
```

To confirm this has worked, use the following command.

```
node -v
```

Note that `npm` comes packaged with `node`

Finally, install `git` which will be necessary for creating your first Gatsby project based on a starter.

```
sudo apt install git
```

Fedora, RedHat, and other dnf based distros

These distros come installed with `curl`, so you can use that to download `nvm`.

```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh | bash
```

Note that this is the current stable release of `nvm`. Full installation instructions and troubleshooting can be found at the `nvm` GitHub page

When `nvm` is installed, it does not default to a particular `node` version. You'll need to install the version you want and give `nvm` instructions to use it. This example uses the latest release of version `16`, but more recent version numbers can be used instead.

```
nvm install 16
nvm use 16
```

To confirm this has worked, use the following command.

```
node -v
```

Note that `npm` comes packaged with `node`

Finally, install `git` which will be necessary for creating your first Gatsby project based on a starter.

```
sudo dnf install git
```

Arch Linux and other pacman based distros

Begin by updating.

```
sudo pacman -Sy
```

These distros come installed with `curl`, so you can use that to download `nvm`.

```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh | bash
```

Note that this is the current stable release of `nvm`. Full installation instructions and troubleshooting can be found at the `nvm` GitHub page

Before using `nvm`, you need to install additional dependencies.

```
sudo pacman -S grep awk tar git
```

When `nvm` is installed, it does not default to a particular `node` version. You'll need to install the version you want and give `nvm` instructions to use it. This example uses the latest release of version `16`, but more recent version numbers can be used instead.

```
nvm install 16
nvm use 16
```

To confirm this has worked, use the following command.

```
node -v
```

Note that **npm** comes packaged with **node**

Windows Subsystem Linux (WSL)

This guide assumes that you already have WSL installed with a working Linux distro. If you don't, follow this guide from Microsoft's site to install WSL and a Linux distro of your choice. Make sure that you installed **WSL 2**.

As of October 17th 2017, Windows 10 ships with WSL and Linux distributions are available via the Microsoft Store, there are several different distributions to use which can be configured via **wslconfig** if you have more than one distribution installed.

```
# set default distribution to Ubuntu
wslconfig /setdefault ubuntu
```

Please note that if you have used the Gatsby on Windows setup without WSL, then you have to delete any existing **node_modules** folder in your project and re-install the dependencies in your WSL environment.

Using Windows Subsystem Linux: Ubuntu

If you have a fresh install of Ubuntu then update and upgrade:

```
sudo apt update
sudo apt -y upgrade
```

Build tools

To compile and install native addons from npm you may also need to install build tools for **node-gyp**:

```
sudo apt install -y build-essential
```

Install node

Following the install instructions on nodejs.org leaves a slightly broken install (i.e. permission errors when trying to **npm install**). Instead try installing node versions using **n** which you can install with **n-install**:

```
curl -L https://git.io/n-install | bash
```

There are other alternatives for managing your node versions such as **nvm** but this is known to slow down bash startup on WSL.

Using Windows Subsystem Linux: Debian

Debian setup is nearly identical to Ubuntu except for the additional installs of `git` and `libpng-dev`.

Begin by updating and upgrading.

```
sudo apt update
sudo apt -y upgrade
```

Additional dependencies need to be installed as well. `build-essential` is a package that allows other packages to compile to a Debian package. `git` installs a package to work with version control. `libpng-dev` installs a package that allows the project to manipulate images.

```
sudo apt install build-essential
sudo apt install git
sudo apt install libpng-dev
```

Or to install all at the same time and approve (y) all installs:

```
sudo apt update && sudo apt -y upgrade && sudo apt install build-essential && sudo apt install git && sudo apt install libpng-dev
```

Additional links and resources

- Super detailed guide to making VSCode work with WSL from VSCode's docs website
- Microsoft Store page for downloading Ubuntu on Windows
- `n`
- `nvm`
- `n-install`
- `bash` startup