

Justin Ellingwood  
May 12, 2014

♥ 46    💬 25

📄 Share



How To Install Node.js on an Ubuntu 14.04 server

Tags: Node.js

Distribution: Ubuntu

## Introduction

Node.js is a Javascript platform for server-side programming that allows users to build network applications quickly. By leveraging Javascript on both the front-end and the back-end, development can be more consistent and be designed within the same system.

In this guide, we'll show you how to get started with Node.js on an Ubuntu 14.04 server.

If you are looking to set up a production Node.js environment, check out this link: [How To Set Up a Node.js Application for Production](#).

## How To Install the Distro-Stable Version

Ubuntu 14.04 contains a version of Node.js in its default repositories that can be used to easily provide a consistent experience across multiple servers. The version in the repositories is 0.10.25. This will not be the latest version, but it should be quite stable.

In order to get this version, we just have to use the `apt` package manager. We should refresh our local package index prior and then install from the repositories:

```
sudo apt-get update
sudo apt-get install nodejs
```

If the package in the repositories suits your needs, this is all that you need to do to get set up with Node.js. In most cases, you'll also want to also install `npm`, which is the Node.js package manager. You can do this by typing:

```
sudo apt-get install npm
```

This will allow you to easily install modules and packages to use with Node.js.

Because of a conflict with another package, the executable from the Ubuntu repositories is called `nodejs` instead of `node`. Keep this in mind as you are running software.

Below, we'll discuss some more flexible methods of installation.

## How To Install Using a PPA

An alternative that can get you a more recent version of Node.js is to add a PPA (personal package archive) maintained by NodeSource. This will probably have more up-to-date versions of Node.js than the official Ubuntu repositories.

First, you need to install the PPA in order to get access to its contents:

```
curl -sL https://deb.nodesource.com/setup | sudo bash -
```

The PPA will be added to your configuration and your local package cache will be updated automatically. After running the setup script from nodesource, you can install the Node.js package in the same way that you did above:

```
sudo apt-get install nodejs
```

The `nodejs` package contains the `nodejs` binary as well as `npm`, so you don't need to install `npm` separately. However, in order for some `npm` packages to work (such as those that require building from source), you will need to install the `build-essentials` package:

```
sudo apt-get install build-essential
```

## How To Install Using NVM

An alternative to installing Node.js through `apt` is to use a specially designed tool called `nvm`, which stands for "Node.js version manager".

Using `nvm`, you can install multiple, self-contained versions of Node.js which will allow you to control your environment easier. It will give you on-demand access to the newest versions of Node.js, but will also allow you to target previous releases that your app may depend on.

To start off, we'll need to get the software packages from our Ubuntu repositories that will allow us to build source packages. The `nvm` script will leverage these tools to build the necessary components:

```
sudo apt-get update
sudo apt-get install build-essential libssl-dev
```

Once the prerequisite packages are installed, you can pull down the `nvm` installation script from the [project's GitHub page](#). The version number may be different, but in general, you can download and install it with the following syntax:

```
curl https://raw.githubusercontent.com/creationix/nvm/v0.16.1/install.sh | sh
```

This will download the script and run it. It will install the software into a subdirectory of your home directory at `~/.nvm`. It will also add the necessary lines to your `~/.profile` file to use the file.

To gain access to the `nvm` functionality, you'll need to log out and log back in again, or you can source the `~/.profile` file so that your current session knows about the changes:

```
source ~/.profile
```

Now that you have `nvm` installed, you can install isolated Node.js versions.

To find out the versions of Node.js that are available for installation, you can type:

```
nvm ls-remote
```

```
. . .  
v0.11.6  
v0.11.7  
v0.11.8  
v0.11.9  
v0.11.10  
v0.11.11  
v0.11.12  
v0.11.13
```

As you can see, the newest version at the time of this writing is v0.11.13. You can install that by typing:

```
nvm install 0.11.13
```

Usually, nvm will switch to use the most recently installed version. You can explicitly tell nvm to use the version we just downloaded by typing:

```
nvm use 0.11.13
```

When you install Node.js using nvm, the executable is called `node`. You can see the version currently being used by the shell by typing:

```
node -v
```

```
v.0.11.13
```

If you have multiple Node.js versions, you can see what is installed by typing:

```
nvm ls
```

If you wish to default one of the versions, you can type:

```
nvm alias default 0.11.13
```

This version will be automatically selected when a new session spawns. You can also reference it by the alias like this:

```
nvm use default
```

Each version of Node.js will keep track of its own packages and has `npm` available to manage these.

You can have `npm` install packages to the Node.js project's `./node_modules` directory by using the normal format:

```
npm install express
```

If you'd like to install it globally (available to the other projects using the same Node.js version), you can add the `-g` flag:

```
npm install -g express
```

This will install the package in:

```
~/ .nvm/node_version/lib/node_modules/package_name
```

Installing globally will let you run the commands from the command line, but you'll have to use link the package into your local sphere to require it from within a program:

```
npm link express
```

You can learn more about the options available to you with nvm by typing:

```
nvm help
```

## Conclusion

As you can see, there are a quite a few ways to get up and running with Node.js on your Ubuntu 14.04 server. Your circumstances will dictate which of the above methods is the best idea for your circumstance. While the packaged version in Ubuntu's repository is the easiest, the `nvm` method is definitely much more flexible.

By Justin Ellingwood

♥ Heart 46

📄 Share

Subscribe

Author:  
Justin Ellingwood

Spin up an SSD cloud  
server in under a minute.

Simple setup. Full root access.  
Straightforward pricing.

DEPLOY SERVER

## Related Tutorials

How To Use npm to Build and Publish Node.js Packages on a Linux Server

Geddy.JS: A No-Brainer MVC Node.js Framework

How To Use node.js, request and cheerio to Set Up Simple Web-Scraping

How To Install Express, a Node.js Framework, and Set Up Socket.io on a VPS

How To Deploy a DocPad Application on Ubuntu 14.04

## 25 Comments

**B** *I*      



Leave a comment...

Log In to Comment



Nerull *May 21, 2014*

been trying to run p2pool-scanner and it keeps throwing up this error

<http://pastebin.com/jQtpfwjG>



Reply



asb **MOD** *May 21, 2014*

@anthonymschimke: Looks like p2pool-scanner was written for express 3.x but you have express 4.x installed. Express 4.x does not have a configure method. This is a bug in p2pool-scanner. You might be able to work around it by, uninstalling express and then reinstalling it forcing the old version:

```
npm uninstall -g express
npm install -g express@3.8.0
```



Reply



Nerull *May 22, 2014*

thanks that seems to have done it



Reply



bjones1831 *June 2, 2014*

Thanks for the info.

When I use the sudo add-apt-repository ppa:chris-lea/node.js method, Node installs fine (0.10.28). And later I can install NPM ok (1.4.9). However when I try npm install express (or npm install -g express), I get errors (just a sample below):

```
npm ERR! Error: EACCES, mkdir '/usr/share/nodejs/node_modules'
npm ERR! { [Error: EACCES, mkdir '/usr/share/nodejs/node_modules']
npm ERR!   errno: 3,
```

npm ERR! Please try running this command again as root/Administrator.

-----

From what I can understand, using the ppa:chris-lea method means I can only install express using sudo. But I've read that this is not recommended.

I've tried chowning to my `whoami` username in several directories, but I'm still having no luck finding a solution.

Can you possibly help me?

Thanks

FWIW, I have a node file (test.js) in my /usr/share/node directory which runs fine in my terminal on port 8080.

♥ Reply



asb **MOD** June 3, 2014

@bjones1831: You should only need `sudo` if you are installing it globally with "`-g`" Are you running it in `/usr/share/nodejs/` ? If so, that's why you need root permissions. Try moving your project to a directory you have write access to, like your home directory.

♥ Reply



tejas-manohar June 8, 2014

Where is node path if you install it via the PPA?

♥ Reply



asb **MOD** June 9, 2014

The PPA also installs the binary /usr/bin/nodejs So you might want to run:

```
sudo ln -s /usr/bin/nodejs /usr/bin/node
```

♥ 1 Reply



jacquelinehoman7 July 13, 2014

I am really having a lot of difficulty in getting this to work for me. In the year and a half in to learning software development, I am finding that I most often have problems with having to fight with getting my dev environment set up correctly than I do actually learning the programming language(s). Anyway, I made up some gist files of my terminal output in the order I have tried getting this to work for me and would greatly appreciate any help with getting this to work for me.

First I tried this:

<https://gist.github.com/jacqueline-homan/ce96b7e21493ca13e734#file-unable-to-get-yoeman-and-bower>

Then I tried this (which worked for the npm part of it):

<https://gist.github.com/jacqueline-homan/d1a051750dc360d058bb#file-this-worked>

But when I tried to install nvm, it failed and I don't know how to get it to work for me:

<https://gist.github.com/jacqueline-homan/bebbb88250ee5f776e64#file-installing-nvm-did-not-work>

♥ Reply



kamaln7 **MOD** July 14, 2014

@jacquelinehoman7: <https://raw.githubusercontent.com/creationix/nvm/v.10.0/install.sh> is invalid. You can find the correct URL here: [creationix/nvm](https://raw.githubusercontent.com/creationix/nvm/):

<https://raw.githubusercontent.com/creationix/nvm/v0.10.0/install.sh>.

♥ Reply



robertchristian1985 July 23, 2014

These instructions are incomplete. Packages like grunt will attempt to execute scripts at the command line using "node", not "nodejs". I for one am looking for instructions about the correct way to fix this. I could remove the package ("node") but I don't know what it does. Or I could remove it from my path...

♥ Reply



kamaln7 **MOD** July 23, 2014

@robertchristian1985: You can link `node` to `nodejs` by running:

```
sudo ln -s "$(which nodejs)" /usr/bin/node
```

♥ Reply



kumiaradamhanp September 16, 2014

Thanks bro...It works!!

♥ Reply



ddascalescu+digiocean September 20, 2014

Chris Lea's repository is deprecated. Please see the latest node.js installation instructions at <https://github.com/joyent/node/wiki/Installing-Node.js-via-package-manager>

♥ Reply



kamaln7 **MOD** September 20, 2014

Thanks! I've updated the article.



tonyb67 December 27, 2014

Yeah I can play with Koa :D  
I use n on my local machine

♥ Reply



tonyb67 December 27, 2014

This is my preferred way of installing node / npm <https://www.digitalocean.com/community/tutorials/how-to-install-an-upstream-version-of-node-js-on-ubuntu-12-04>



NicholasFlint January 11, 2015

Running Ubuntu 14.04.1  
trying to install node.js via nvm

Hi there does anyone know offhand why when I type this for the nvm installation script

```
curl https://raw.githubusercontent.com/creationix/nvm/v0.16.1/install.sh | sh
```

I get this  
sh: 1: Not: not found

I tried this  

```
curl https://raw.githubusercontent.com/creationix/nvm/blob/v0.22.0/install.sh | sh
```

  
and I got the same result.  
I am not sure the the pipe at the end and the sh is for but is there something I am missing.  
I tried it once with bash as the ending, that didn't seem to work.

♥ Reply



jellingwood **MOD** January 12, 2015

@NicholasFlint: Can you try taking the "`| sh`" portion off for a moment? Does the `curl` command now download and display the raw script?



JacobsLadd3r January 13, 2015

Thanks so much for the great very in-depth guide @jellingwood! Really helped me for reference when writing a guide on how to install

Node.js myself.

Just as a heads up, although probably in the minority of cases, I ran into an issue where the old "node" amateur packet radio program was installed on a system. As some Node.js apps seem to call "node" instead of "nodejs", this was causing conflicts and had me scratching my head for a minute.

You can very easily check for "node" with "install" in its column, then remove it if you run into this:

```
$ dpkg --get-selections | grep node
ax25-node                                install
node                                    install
$ sudo apt-get remove --purge node
```

Then to make those Node.js apps that reference "node" functional symbolic link it up:

```
sudo ln -s /usr/bin/nodejs /usr/bin/node
```

Great stuff man!

♥ Reply



dpelletier February 10, 2015

When I install nodejs using apt-get on Ubuntu, the npm module is *not* included. Installing using nvm install script as described here -- is npm included?

♥ Reply



maxmckenzie February 10, 2015

For what i remember npm is not under the same name in the ubuntu software repo (apt-get).

See this post for more info <http://stackoverflow.com/a/21171188>

there it says to install via `sudo apt-get install nodejs-legacy`

however I think thats an out of date version. I think there is a way to include the repo URL for npm in the list of software sources in ubuntu. However this was a while ago i did this and the answers escape me hopefully this info will help.



dpelletier February 10, 2015

Thank you for the response maxmckenzie. Although I (ultimately) ended up going a different route mentioned below.



dpelletier February 10, 2015

Much to my great frustration, nowhere, could I really find a complete and concise method of installing the full node stack that did not involve using sudo or some drawn out tedious, solution (and neither the official node site or stackexchange helped here)...

As well, all of the sites I visited said that npm was included with the nodejs module yet the nodejs version existing in the default Ubuntu repository did not seem to have it.

The nvm module sounded very interesting and gives you the ability to easily install and switch node.js versions (among other features).

To save others similar frustrations, I thought I would sketch out the steps I took to install the nodejs stack onto Ubuntu 12.04, "Precise" (elementary):

Also, since nvm uses the name "node" instead of "nodejs", make sure that the unrelated program also named "node" -- the Amateur Packet Radio -- does not already exist on your system.

```
sudo apt-get install git build-essential libssl-dev curl git-core
```

```
wget -qO- https://raw.githubusercontent.com/creationix/nvm/v0.23.3/install.sh | bash
```

(I determined the latest nvm version available -- and a lot of other, really helpful information -- at: <https://github.com/creationix/nvm>)

To have nvm automatically sourced, accessible at login:



```
vi ~/.bashrc
```

And add this to the end:

```
source ~/.nvm/nvm.sh
```

Now, as root (of course):

```
vi /etc/bash.bashrc
```

And add this to the end:

```
export NVM_SYMLINK_CURRENT=true
```

(close your terminal sessions and log back in - not as root)

```
nvm install stable
```

```
nvm use stable
```

To set a default version to be used in any new shell, use the alias 'default':

```
nvm alias default stable
```

Notice that nvm installs everything to the ~/.nvm directory (circumventing the usual requirement and inherent danger of requiring sudo to be used for installing nodejs and its scripts):

```
/home/dmpelletier/.nvm/versions/node/v0.12.0/bin/node
```

```
/home/dmpelletier/.nvm/versions/node/v0.12.0/bin/npm
```

And if you want Ionic to be installed as well, as I did:

```
npm install ionic -g
```

♥ Reply



BlueX April 1, 2015

Hey ! Can you help me ? I installed Node.js on MongoDB server, all applications are running well, I just have problem with "forever" plugin, when I want to use it I just get :

```
node: no such file or directory
```

Is it because of usage of "nodejs" command instead of "node" command ? I had no problem with this on server created as Node.js server. Any help appreciated, thanks ^^

♥ Reply



jellingwood MOD April 2, 2015

@BlueX: If your other scripts are expecting the executable to be called `node`, you can create a symbolic link to the `nodejs` file like this:

```
sudo ln -s /usr/bin/nodejs /usr/bin/node
```

Try to see if that helps.

♥ 1

B I



Leave a comment...

Log In to Comment



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.



Copyright © 2015 DigitalOcean™ Inc.

[Community](#) [Tutorials](#) [Questions](#) [Projects](#) [Tags](#) [Terms, Privacy, & Copyright](#) [Security](#)