

How to Install Node.js on Ubuntu 16.04 Xenial Xerus Linux server

Lubos Rendek Ubuntu 12 December 2016

Objective

The objective is to install Node.js the cross-platform JavaScript runtime environment on Ubuntu 16.04 Xenial Xerus Linux

Requirements

Privileged access to your Ubuntu System as root or via `sudo` command is required.

Difficulty

EASY

Conventions

- # - requires given command to be executed with root privileges either directly as a root user or by use of `sudo` command
- \$ - given command to be executed as a regular non-privileged user

Instructions

There are multiple ways on how to install Node.js on your Ubuntu 16.04 Xenial Xerus Linux server. The below steps will show you how to install Node.js using a standard Ubuntu repository, PPA repository, Node.js native setup script and by use of Node Version Manager. The easiest installation is by using standard Ubuntu repository, however it yields lower Node.js version. If you need a bleeding-edge Node.js version you better opt for an automatic installation using the Node.js native setup script.

node.js installation from Ubuntu Repository

Installation of Node.js using Ubuntu's standard repository cannot be simpler:

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

You may also want to install Node.js package manager `npm` :

Contents

1. Objective
2. Requirements
3. Difficulty
4. Conventions
5. Instructions
 - 5.1. node.js installation from Ubuntu Repository
 - 5.2. Using PPA repository
 - 5.3. Node.js setup script
 - 5.4. nvm installation

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

Check for installed versions:

```
$ nodejs --version  
v4.2.6  
$ npm --version  
3.5.2
```

Using PPA repository

If applicable, first make `add-apt-repository` command available on your system:

```
$ sudo apt-get install python-software-properties
```

Next, add PPA repository:

```
$ sudo add-apt-repository -y -r ppa:chris-lea/node.js  
$ sudo curl --silent https://deb.nodesource.com/gpgkey/nodesource.gpg.key | sudo apt-key add -
```

Set a version number of Node.js you wish to install:

```
VERSION=node_7.x  
DISTRO="$(lsb_release -s -c)"
```

Configure, Node.js repositories with the above settings:

```
$ sudo echo "deb https://deb.nodesource.com/$VERSION $DISTRO main" | sudo tee /etc/apt/sources.list.d/n  
$ sudo echo "deb-src https://deb.nodesource.com/$VERSION $DISTRO main" | sudo tee -a /etc/apt/sources.li
```

At this point simple execute the below `apt-get` commands to install Node.js:

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

Check correctness of the installation:

```
$ nodejs --version  
v7.2.1  
$ npm --version  
3.10.10
```

Node.js setup script

Using a native Node.js setup script is probably the most easiest way how to install latest Node.js version on your Ubuntu 16.04 Linux server:

```
$ curl -sL https://deb.nodesource.com/setup_7.x | sudo -E bash -  
$ sudo apt-get install nodejs
```

All done. Check versions:

```
$ nodejs --version  
v7.2.1  
$ npm --version  
3.10.10
```

nvm installation

In case none of the above Node.js install fit your environment, the below manual installation using `nvm` might prove helpful. First, install all prerequisites:

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

Install `nvm` using its native installation script. Correct the version number within the below URL if necessary:

```
# curl -o- https://raw.githubusercontent.com/creationix/nvm/v0.32.1/install.sh | bash
```

Source new NVM settings:

```
$ . ~/.profile
```

Find an appropriate version number. For example the below command will list all LTS Node.js releases so far:

```
$ nvm ls-remote | grep -i lts  
v4.2.0 (LTS: Argon)  
v4.2.1 (LTS: Argon)  
v4.2.2 (LTS: Argon)  
v4.2.3 (LTS: Argon)  
v4.2.4 (LTS: Argon)  
v4.2.5 (LTS: Argon)  
v4.2.6 (LTS: Argon)  
v4.3.0 (LTS: Argon)  
v4.3.1 (LTS: Argon)  
v4.3.2 (LTS: Argon)  
v4.4.0 (LTS: Argon)  
v4.4.1 (LTS: Argon)  
v4.4.2 (LTS: Argon)  
v4.4.3 (LTS: Argon)  
v4.4.4 (LTS: Argon)
```

```
v4.4.5 (LTS: Argon)
v4.4.6 (LTS: Argon)
v4.4.7 (LTS: Argon)
v4.5.0 (LTS: Argon)
v4.6.0 (LTS: Argon)
v4.6.1 (LTS: Argon)
v4.6.2 (LTS: Argon)
v4.7.0 (Latest LTS: Argon)
v6.9.0 (LTS: Boron)
v6.9.1 (LTS: Boron)
v6.9.2 (Latest LTS: Boron)
```

Use `nvm` command to install your desired Node.js version. For example the below command will install a latest LTS Node.js release:

```
$ nvm install 6.9.2
##### 100.0%
Computing checksum with sha256sum
Checksums matched!
Now using node v6.9.2 (npm v3.10.9)
Creating default alias: default -> 6.9.2 (-> v6.9.2)
```

Check versions:

```
$ node --version
v6.9.2
$ npm --version
3.10.9
```