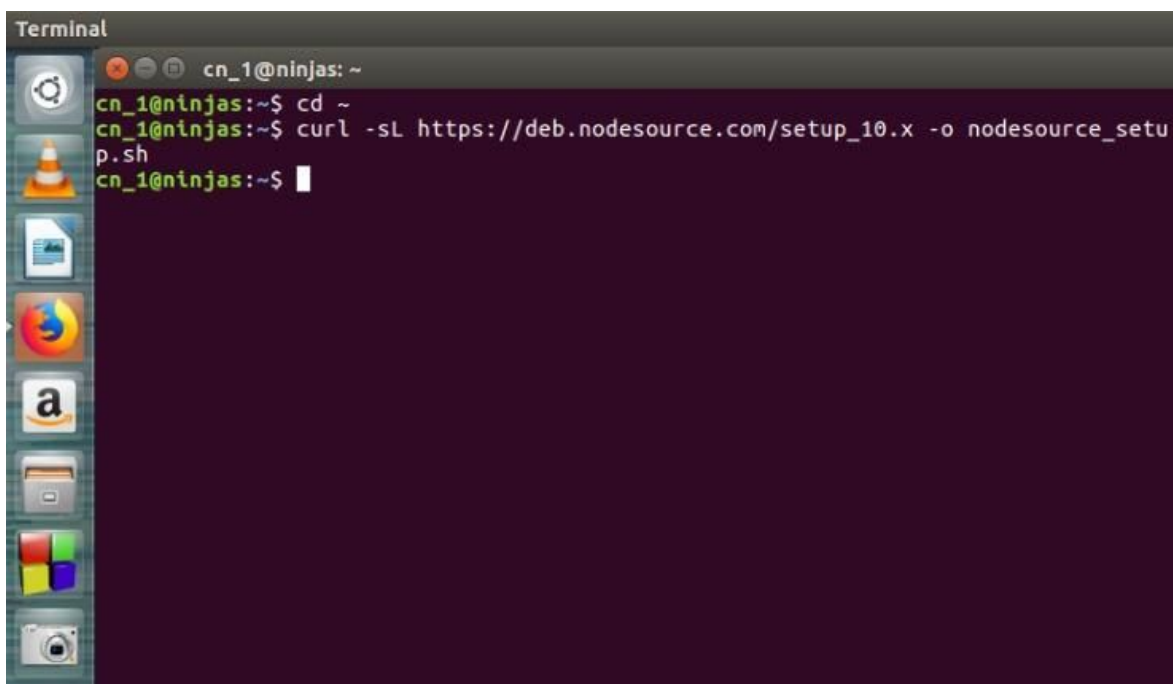


Install Node.js and npm using a PPA

- 1) First, install the PPA in order to get access to its contents. From your home directory, use curl to retrieve the installation script for your preferred version, making sure to replace **10.x** with your preferred version string (if different):

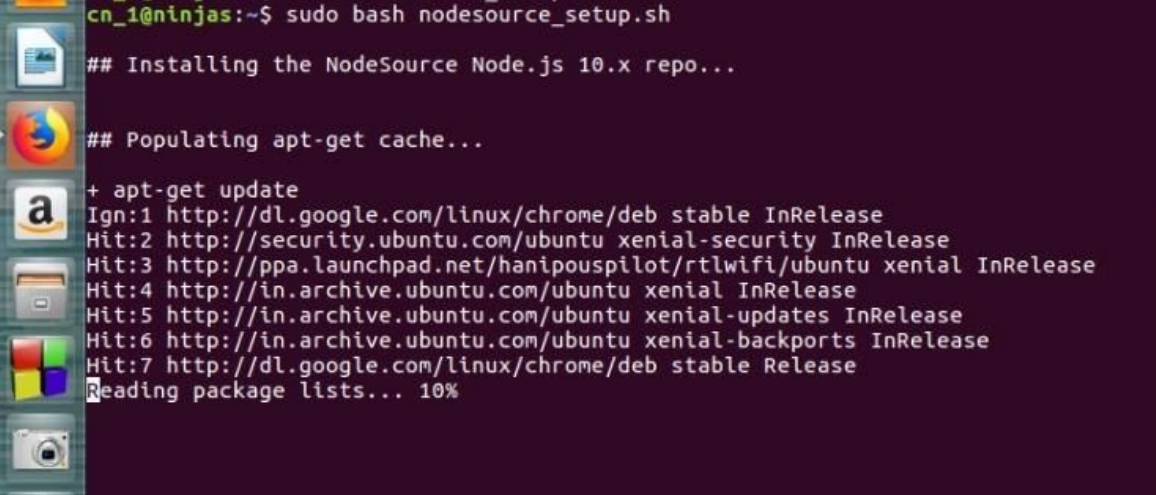
```
cd ~
```

```
curl -sL https://deb.nodesource.com/setup_10.x -o nodesource_setup.sh
```



- 2) Run the script under sudo:

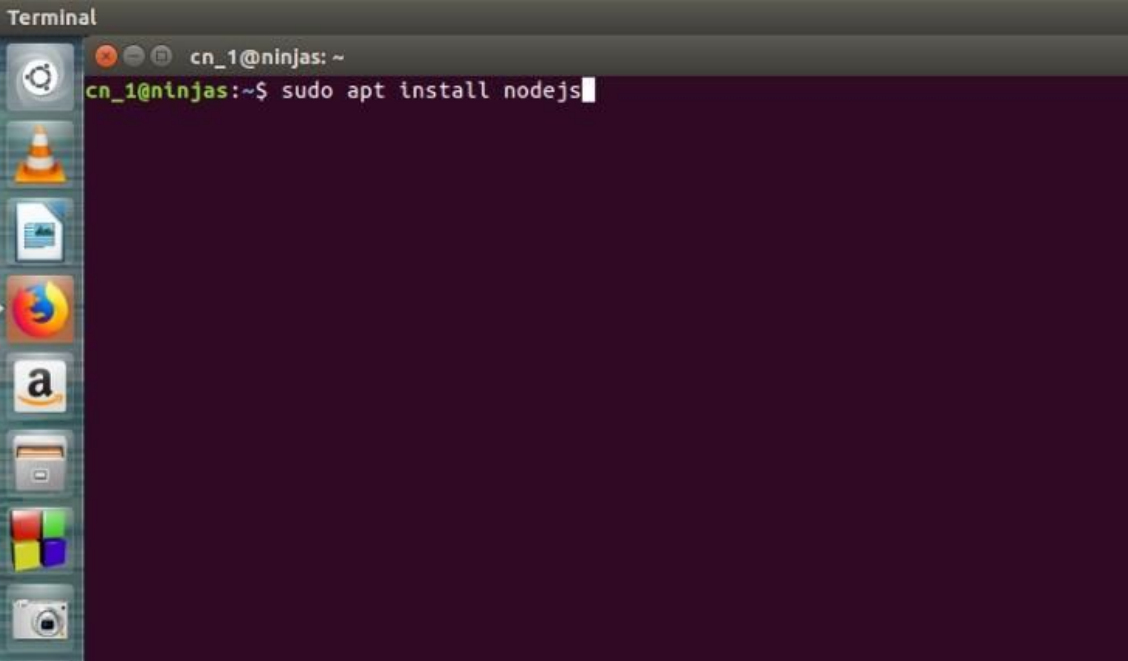
```
sudo bash nodesource_setup.sh
```

A terminal window with a dark purple background. The prompt is 'cn_1@ninjas:~\$'. The user has run 'sudo bash nodesource_setup.sh'. The script outputs: '## Installing the NodeSource Node.js 10.x repo...', '## Populating apt-get cache...', and '+ apt-get update'. It then lists several package sources with 'Hit' status and ends with 'Reading package lists... 10%'.

```
cn_1@ninjas:~$ sudo bash nodesource_setup.sh
## Installing the NodeSource Node.js 10.x repo...
## Populating apt-get cache...
+ apt-get update
Ign:1 http://dl.google.com/linux/chrome/deb stable InRelease
Hit:2 http://security.ubuntu.com/ubuntu xenial-security InRelease
Hit:3 http://ppa.launchpad.net/hanipouspilot/rtlwifi/ubuntu xenial InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu xenial InRelease
Hit:5 http://in.archive.ubuntu.com/ubuntu xenial-updates InRelease
Hit:6 http://in.archive.ubuntu.com/ubuntu xenial-backports InRelease
Hit:7 http://dl.google.com/linux/chrome/deb stable Release
Reading package lists... 10%
```

- 3) 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 you did above:

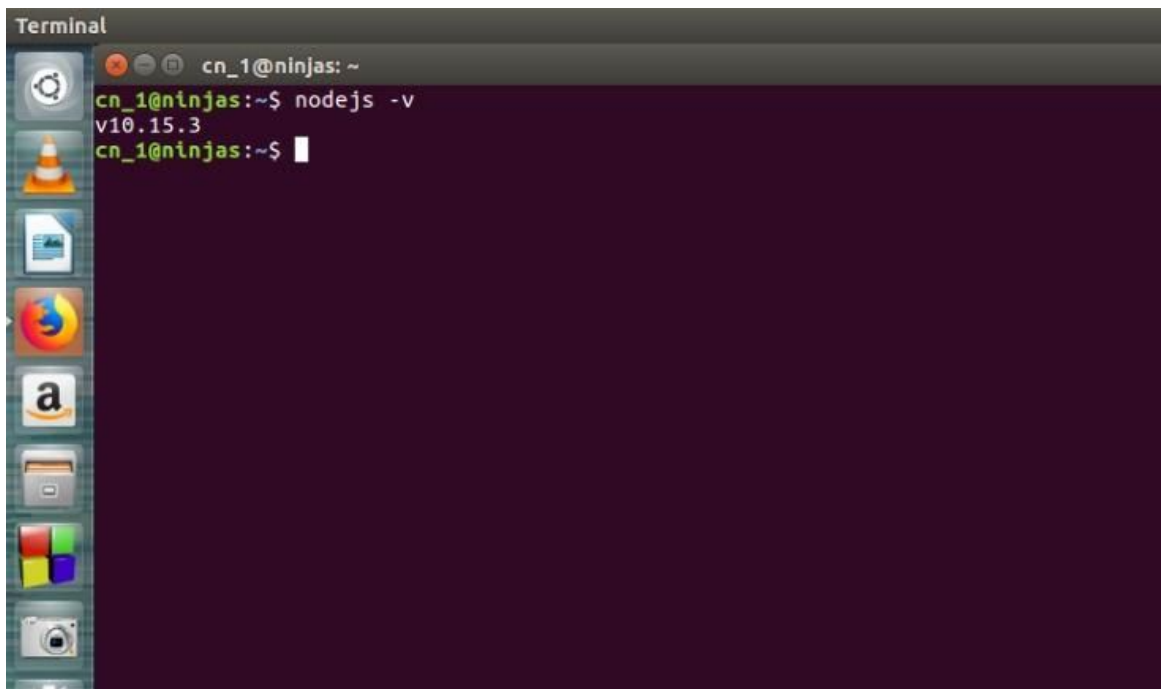
`sudo apt install nodejs`

A terminal window titled 'Terminal' with a dark purple background. The prompt is 'cn_1@ninjas: ~'. The user has entered the command 'sudo apt install nodejs' and the cursor is at the end of the line.

```
Terminal
cn_1@ninjas: ~
cn_1@ninjas:~$ sudo apt install nodejs
```

- 4) To check which version of Node.js you have installed after these initial steps, type:

`nodejs -v`

A terminal window titled "Terminal" with a dark purple background. The window shows the user "cn_1@ninjas" in the home directory. The command "nodejs -v" has been executed, resulting in the output "v10.15.3". The prompt "cn_1@ninjas:~\$" is visible on the next line. On the left side of the terminal, there is a vertical dock with several application icons: a gear, a traffic cone, a document, a Firefox logo, an Amazon logo, a folder, a Windows logo, and a camera.

```
Terminal
cn_1@ninjas: ~
cn_1@ninjas:~$ nodejs -v
v10.15.3
cn_1@ninjas:~$
```

- 5) npm uses a configuration file in your home directory to keep track of updates. It will be created the first time you run npm. Execute this command to verify that npm is installed and to create the configuration file:

```
npm -v
```

- 6) In order for some npm packages to work (those that require compiling code from source, for example), you will need to install the build-essential package:

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

```
Terminal
cn_1@ninjas: ~
cn_1@ninjas:~$ npm -v
6.4.1
cn_1@ninjas:~$ sudo apt install build-essential
Reading package lists... Done
Building dependency tree
Reading state information... Done
build-essential is already the newest version (12.1ubuntu2).
build-essential set to manually installed.
The following packages were automatically installed and are no longer required:
  gnome-software-common gyp javascript-common libgtkspell3-3-0 libjs-inherits
  libjs-jquery libjs-node-uuid libjs-underscore libssl-dev libssl-doc libuv1
  libuv1-dev node-abbrev node-ansi node-ansi-color-table node-archy node-async
  node-block-stream node-combined-stream node-cookie-jar node-delayed-stream
  node-forever-agent node-form-data node-fstream node-fstream-ignore
  node-github-url-from-git node-glob node-graceful-fs node-gyp node-inherits
  node-ini node-json-stringify-safe node-lockfile node-lru-cache node-mime
  node-minimatch node-mkdirp node-mute-stream node-node-uuid node-nopt
  node-normalize-package-data node-npmlog node-once node-osenv node-qjs
  node-read node-read-package-json node-request node-retry node-rimraf
  node-semver node-sha node-sigmund node-slide node-tar node-tunnel-agent
  node-underscore node-which python-pkg-resources zlib1g-dev
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 499 not upgraded.
cn_1@ninjas:~$
```

You now have the necessary tools to work with npm packages that require compiling code from source.