

How to Install a Node.js Application

Overview

Install a Node.js application

Pre-installation settings.

Install the application.

Test the application.

Register the application.

Restart the application.

Additional documentation

Overview

This document describes how to install a Node.js web application. Node.js is a JavaScript runtime that allows you to build scalable web applications.



Warning:

- cPanel Technical Support does **not** provide assistance with the steps in this document.
- We recommend that **only** experienced system administrators perform these steps.
- We are **not** responsible for any data loss.



Note:

- We recommend that you perform the steps in this document via the command line as a cPanel user unless otherwise specified.
- You can also perform these steps in cPanel's *Terminal* interface (*cPanel* >> *Home* >> *Advanced* >> *Terminal*).
- In this document, `nodejsapp` represents the application's name and `3000` represents an open firewall port.

For more information, read the [Node.js](#) documentation.

Install a Node.js application



Pre-installation settings.

Before you begin, make **certain** that your hosting provider installed the following EasyApache 4 RPMs on your server:

- `ea-ruby24-mod_passenger`



Note:

This module disables [Apache's mod_userdir module](#).

- `ea-apache24-mod_env`
- `ea-nodejs10`



Note:

We also recommend that your hosting provider install the `ea-ruby24-ruby-devel` module.

1

Install the application.

To install an application, perform the following steps:

1. Log in to the server via SSH as a cPanel user.
2. Create the application's directory, relative to your home directory. To do this, run the following command:

```
mkdir nodejsapp
```

3. Change to the application's directory.
4. Create the `app.js` file with a text editor.
5. Add the application's configuration to the `app.js` file. This will resemble the following example:

```
const http = require('http')
const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World! NodeJS \n');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

2

Test the application.

After you install the application, we recommend that you confirm it's active. To do this, perform the following steps:

1. Run the following command:

```
/opt/cpanel/ea-nodejs10/bin/node app.js
```

The output will resemble will the following example:

```
Server running at http://127.0.0.1:3000
```

2. Open another terminal window and log in to the server via SSH as the same cPanel user.
3. Run the following command:

```
curl http://127.0.0.1:3000
```

The output will resemble the following example:

```
Hello World! NodeJS
```

**Note:**

If you wish to export the `/opt/cpanel/ea-nodejs10/bin` path to your environment, add the following line to your `.bashrc` file:

```
export PATH=/opt/cpanel/ea-nodejs10/bin/:$PATH
```

**Register the application.**

After you install the application, register it. To do this, use cPanel's [Application Manager](#) interface (*cPanel >> Home >> Software >> Application Manager*).

After you register the application, you can access the application in a web browser with the following url:

```
http://example.com/nodejsapp
```

**Note:**

In this example, `example.com` represents your website.

**Restart the application.**

When you want your application to restart after you edit it, [create the `restart.txt` touch file](#) in the application's `/tmp` directory. This file directs `mod_passenger` to restart the application after you modify it. This action applies your changes to the application.

**Important:**

You **must** touch the `restart.txt` touch file each time that you want `mod_passenger` to restart the application.

Additional documentation

- [Guide to Node.js Installations](#)
- [How to Install a Node.js Application](#)