



Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'npm-ci.1'

\$ man npm-ci.1

NPM-CI(1) NPM-CI(1)

NAME

npm-ci - Install a project with a clean slate

Synopsis

npm ci

Description

This command is similar to `npm help install`, except it's meant to be used in automated environments such as test platforms, continuous integration, and deployment -- or any situation where you want to make sure you're doing a clean install of your dependencies.

npm ci will be significantly faster when:

? There is a `package-lock.json` or `npm-shrinkwrap.json` file.

? The `node_modules` folder is missing or empty.

In short, the main differences between using `npm install` and `npm ci` are:

? The project must have an existing `package-lock.json` or `npm-shrinkwrap.json`.

? If dependencies in the package lock do not match those in `package.json`, `npm ci` will exit with an error, instead of updating the package lock.

? `npm ci` can only install entire projects at a time: individual dependencies cannot be added with this command.

? If a `node_modules` is already present, it will be automatically removed before `npm ci` begins its install.

? It will never write to `package.json` or any of the `package-locks`: installs are essentially frozen.

Example

Make sure you have a package-lock and an up-to-date install:

```
$ cd ./my/npm/project
```

```
$ npm install
```

```
added 154 packages in 10s
```

```
$ ls | grep package-lock
```

Run npm ci in that project

```
$ npm ci
```

```
added 154 packages in 5s
```

Configure Travis to build using npm ci instead of npm install:

```
# .travis.yml
```

```
install:
```

```
- npm ci
```

```
# keep the npm cache around to speed up installs
```

```
cache:
```

```
  directories:
```

```
    - "$HOME/.npm"
```

Configuration

```
<!-- AUTOGENERATED CONFIG DESCRIPTIONS START --> <!-- automatically generated, do not edit manually --> <!-- see lib/utils/config/definitions.js -->
```

audit

```
? Default: true
```

```
? Type: Boolean
```

When "true" submit audit reports alongside the current npm command to the default registry and all registries configured for scopes. See the documentation for `npm help audit` for details on what is submitted. <!-- automatically generated, do not edit manually --> <!-- see lib/utils/config/definitions.js -->

ignore-scripts

```
? Default: false
```

```
? Type: Boolean
```

If true, npm does not run scripts specified in package.json files.

Note that commands explicitly intended to run a particular script, such as `npm start`, `npm stop`, `npm restart`, `npm test`, and `npm run-script` will still run their intended script if `ignore-scripts` is set, but they will not run any pre- or post-scripts. <!-- automatically

generated, do not edit manually --> <!-- see lib/utis/config/definitions.js -->

script-shell

? Default: '/bin/sh' on POSIX systems, 'cmd.exe' on Windows

? Type: null or String

The shell to use for scripts run with the npm exec, npm run and npm init <pkg> commands.

<!-- automatically generated, do not edit manually --> <!-- see lib/utis/config/defini?

tions.js -->

<!-- AUTOGENERATED CONFIG DESCRIPTIONS END -->

See Also

? npm help install

? npm help package-lock.json

undefined NaN

NPM-CI(1)