



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-config.1'

\$ man npm-config.1

NPM-CONFIG(1)

NPM-CONFIG(1)

NAME

npm-config - Manage the npm configuration files

Synopsis

npm config set <key>=<value> [<key>=<value> ...]

npm config get [<key> [<key> ...]]

npm config delete <key> [<key> ...]

npm config list [--json]

npm config edit

npm set <key>=<value> [<key>=<value> ...]

npm get [<key> [<key> ...]]

alias: c

Note: This command is unaware of workspaces.

Description

npm gets its config settings from the command line, environment variables, npmrc files, and in some cases, the package.json file.

See npm help npmrc for more information about the npmrc files.

See npm help config(7) for a more thorough explanation of the mechanisms involved, and a full list of config options available.

The npm config command can be used to update and edit the contents of the user and global npmrc files.

Sub-commands

Config supports the following sub-commands:

set

```
npm config set key=value [key=value...]
```

```
npm set key=value [key=value...]
```

Sets each of the config keys to the value provided.

If value is omitted, then it sets it to an empty string.

Note: for backwards compatibility, `npm config set key value` is supported as an alias for

```
npm config set key=value.
```

get

```
npm config get [key ...]
```

```
npm get [key ...]
```

Echo the config value(s) to stdout.

If multiple keys are provided, then the values will be prefixed with the key names.

If no keys are provided, then this command behaves the same as `npm config list`.

list

```
npm config list
```

Show all the config settings. Use `-l` to also show defaults. Use `--json` to show the set?

tings in json format.

delete

```
npm config delete key [key ...]
```

Deletes the specified keys from all configuration files.

edit

```
npm config edit
```

Opens the config file in an editor. Use the `--global` flag to edit the global config.

Configuration

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

json

? Default: false

? Type: Boolean

Whether or not to output JSON data, rather than the normal output.

? In `npm pkg set` it enables parsing set values with `JSON.parse()` before saving them to your `package.json`.

Not supported by all npm commands. <!-- automatically generated, do not edit manually -->

<!-- see lib/utis/config/definitions.js -->

global

? Default: false

? Type: Boolean

Operates in "global" mode, so that packages are installed into the prefix folder instead of the current working directory. See `npm help folders` for more on the differences in behavior.

? packages are installed into the `{prefix}/lib/node_modules` folder, instead of the current working directory.

? bin files are linked to `{prefix}/bin`

? man pages are linked to `{prefix}/share/man`

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

tions.js -->

editor

? Default: The `EDITOR` or `VISUAL` environment variables, or 'notepad.exe' on Windows, or 'vim' on Unix systems

? Type: String

The command to run for `npm edit` and `npm config edit`. <!-- automatically generated, do not edit manually --> <!-- see lib/utis/config/definitions.js -->

location

? Default: "user" unless `--global` is passed, which will also set this value to "global"

? Type: "global", "user", or "project"

When passed to `npm config` this refers to which config file to use. <!-- automatically generated, do not edit manually --> <!-- see lib/utis/config/definitions.js -->

long

? Default: false

? Type: Boolean

Show extended information in `ls`, `search`, and `help-search`. <!-- automatically generated, do not edit manually --> <!-- see lib/utis/config/definitions.js -->

<!-- AUTOGENERATED CONFIG DESCRIPTIONS END -->

See Also

? `npm help folders`

? `npm help config`

? npm help package.json

? npm help npmrc

? npm help npm

undefined NaN

NPM-CONFIG(1)