

NAME**npm-adduser** – Add a registry user account**Synopsis**`npm adduser [--registry=url] [--scope=@orgname] [--auth-type=legacy]`

aliases: login, add-user

Note: This command is unaware of workspaces.

Description

Create or verify a user named **<username>** in the specified registry, and save the credentials to the **.npmrc** file. If no registry is specified, the default registry will be used (see `npm help config`).

The username, password, and email are read in from prompts.

To reset your password, go to <https://www.npmjs.com/forgot>

To change your email address, go to <https://www.npmjs.com/email-edit>

You may use this command multiple times with the same user account to authorize on a new machine. When authenticating on a new machine, the username, password and email address must all match with your existing record.

npm login is an alias to **adduser** and behaves exactly the same way.

Configuration

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

registry

- Default: "https://registry.npmjs.org/"
- Type: URL

The base URL of the npm registry. <!-- automatically generated, do not edit manually --> <!-- see lib/utils/config/definitions.js -->

scope

- Default: the scope of the current project, if any, or ""
- Type: String

Associate an operation with a scope for a scoped registry.

Useful when logging in to or out of a private registry:

```
# log in, linking the scope to the custom registry
npm login --scope=@mycorp --registry=https://registry.mycorp.com
```

```
# log out, removing the link and the auth token
npm logout --scope=@mycorp
```

This will cause **@mycorp** to be mapped to the registry for future installation of packages specified according to the pattern **@mycorp/package**.

This will also cause **npm init** to create a scoped package.

```
# accept all defaults, and create a package named "@foo/whatever",
# instead of just named "whatever"
npm init --scope=@foo --yes
```

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

<!-- AUTOGENERATED CONFIG DESCRIPTIONS END -->

See Also

- npm help registry
- npm help config
- npm help npmrc
- npm help owner
- npm help whoami
- npm help token
- npm help profile