

registry

Description

To resolve packages by name and version, npm talks to a registry website that implements the CommonJS Package Registry specification for reading package info.

npm is configured to use the **npm public registry** at <https://registry.npmjs.org> by default. Use of the npm public registry is subject to terms of use available at <https://docs.npmjs.com/policies/terms>.

You can configure npm to use any compatible registry you like, and even run your own registry. Use of someone else's registry may be governed by their terms of use.

npm's package registry implementation supports several write APIs as well, to allow for publishing packages and managing user account information.

The npm public registry is powered by a CouchDB database, of which there is a public mirror at <https://skimdb.npmjs.com/registry>.

The registry URL used is determined by the scope of the package (see **scope**). If no scope is specified, the default registry is used, which is supplied by the **registry** config parameter. See **npm config**, **npmrc**, and **config** for more on managing npm's configuration.

When the default registry is used in a package-lock or shrinkwrap it has the special meaning of "the currently configured registry". If you create a lock file while using the default registry you can switch to another registry and npm will install packages from the new registry, but if you create a lock file while using a custom registry packages will be installed from that registry even after you change to another registry.

Does npm send any information about me back to the registry?

Yes.

When making requests of the registry npm adds two headers with information about your environment:

- **Npm-Scope** – If your project is scoped, this header will contain its scope. In the future npm hopes to build registry features that use this information to allow you to customize your experience for your organization.
- **Npm-In-CI** – Set to “true” if npm believes this install is running in a continuous integration environment, “false” otherwise. This is detected by looking for the following environment variables: `CI`, `TDDIUM`, `JENKINS_URL`, `bamboo.buildKey`. If you’d like to learn more you may find the original PR interesting. This is used to gather better metrics on how npm is used by humans, versus build farms.

The npm registry does not try to correlate the information in these headers with any authenticated accounts that may be used in the same requests.

How can I prevent my package from being published in the official registry?

Set `"private": true` in your `package.json` to prevent it from being published at all, or `"publishConfig":{"registry":"http://my-internal-registry.local"}` to force it to be published only to your internal/private registry.

See `package.json` for more info on what goes in the `package.json` file.

Where can I find my own, & other’s, published packages?

<https://www.npmjs.com/>

See also

- `npm config`
- `config`
- `npmrc`
- `npm developers`