



Linux Ubuntu 26.04 Manual Pages on command 'npm-publish.1'

\$ man npm-publish.1

NPM-PUBLISH(1)

General Commands Manual

NPM-PUBLISH(1)

NAME

npm-publish

Synopsis

```
<!-- AUTOGENERATED USAGE DESCRIPTIONS -->
```

Description

Publishes a package to the registry so that it can be installed by name.

By default npm will publish to the public registry. This can be overridden by specifying a different default registry or using a scope in the name, combined with a scope-configured registry (see package.json).

A package is interpreted the same way as other commands (like npm install and can be:

- ? a) a folder containing a program described by a package.json file
- ? b) a gzipped tarball containing (a)
- ? c) a url that resolves to (b)
- ? d) a <name>@<version> that is published on the registry (see registry) with (c)
- ? e) a <name>@<tag> (see npm dist-tag) that points to (d)
- ? f) a <name> that has a "latest" tag satisfying (e)

? g) a <git remote url> that resolves to (a)

The publish will fail if the package name and version combination already exists in the specified registry.

Once a package is published with a given name and version, that specific name and version combination can never be used again, even if it is removed with `npm unpublish`.

As of `npm@5`, both a `sha1sum` and an `integrity` field with a `sha512sum` of the tarball will be submitted to the registry during publication. Subsequent installs will use the strongest supported algorithm to verify downloads.

Similar to `--dry-run` see `npm pack`, which figures out the files to be included and packs them into a tarball to be uploaded to the registry.

Files included in package

To see what will be included in your package, run `npx npm-packlist`. All files are included by default, with the following exceptions:

? Certain files that are relevant to package installation and distribution are always included. For example, `package.json`, `README.md`, `LICENSE`, and so on.

? If there is a "files" list in `package.json`, then only the files specified will be included. (If directories are specified, then they will be walked recursively and their contents included, subject to the same ignore rules.)

? If there is a `.gitignore` or `.npmignore` file, then ignored files in that and all child directories will be excluded from the package. If both files exist, then the `.gitignore` is ignored, and only the `.npmignore` is used.

`.npmignore` files follow the same pattern rules

as `.gitignore` files

? If the file matches certain patterns, then it will never be included, unless explicitly added to the "files" list in `package.json`, or un-ignored with a `!` rule in a `.npmignore` or `.gitignore` file.

? Symbolic links are never included in npm packages.

See developers for full details on what's

included in the published package, as well as details on how the package is built.

Configuration

<!-- AUTOGENERATED CONFIG DESCRIPTIONS -->

See Also

? package spec

? npm-packlist package

? npm registry

? npm scope

? npm adduser

? npm owner

? npm deprecate

? npm dist-tag

? npm pack

? npm profile

9.2.0

March 2026

NPM-PUBLISH(1)