npm Docs

Try the latest stable version of npm

▶ Details

See what version of npm you're running

```
npm -v
```

Upgrading on `*nix` (OSX, Linux, etc.)

(You may need to prefix these commands with `sudo`, especially on Linux, or OS X if you installed Node using its default installer.)

You can upgrade to the latest version of npm using:

```
npm install -g npm@latest
```

Upgrading on Windows

By default, npm is installed alongside node in

`C:\Program Files (x86)\nodejs`

npm's globally installed packages (including, potentially, npm itself) are stored separately in a user-specific directory (which is currently

`C:\Users\<username>\AppData\Roaming\npm`).

Because the installer puts

`C:\Program Files (x86)\nodejs`

before

```
C:\Users\<username>\AppData\Roaming\npm
```

on your `PATH` , it will always use the version of npm installed with node instead of the version of npm you installed using `npm -g install npm@<version>` .

To get around this, you can do **one** of the following:

- Option 1: edit your Windows installation's `PATH` to put `%appdata%\npm` before `%ProgramFiles%\nodejs` . Remember that you'll need to restart `cmd.exe` (and potentially restart Windows) when you make changes to `PATH` or how npm is installed.
- Option 2: remove both of
    - `%ProgramFiles%\nodejs\npm`
    - `%ProgramFiles%\nodejs\npm.cmd`
- Option 3: Navigate to `%ProgramFiles%\nodejs\node_modules\npm` and copy the `npmrc` file to another folder or the desktop. Then open `cmd.exe` as an administrator and run the following commands:

```
cd %ProgramFiles%\nodejs
npm install npm@latest
```

If you installed npm with the node.js installer, after doing one of the previous steps, do the following.

- Option 1 or 2
    - Go into `%ProgramFiles%\nodejs\node_modules\npm` and copy the file named `npmrc` in the new npm folder, which should be `%appdata%\npm\node_modules\npm` . This will tell the new npm where the global installed packages are.
- Option 3
    - Copy the npmrc file back into `%ProgramFiles%\nodejs\node_modules\npm`

(See also the point below if you're running Windows 7 and don't have the directory `%appdata%\npm` .)

A brief note on the built-in Windows configuration

The Node installer installs, directly into the npm folder, a special piece of Windows-specific configuration that tells npm where to install global packages. When npm is used to install itself, it is supposed to copy this special `builtin` configuration into the new install. There was a bug in some versions of npm that kept this from working, so you may need to go in and fix that up by hand. Run the following command to see where npm will install global packages to verify it is correct.

```
npm config get prefix -g
```

If it isn't set to `<X>:\Users\<user>\AppData\Roaming\npm` , you can run the below command to correct it:

```
npm config set prefix %APPDATA%\npm -g
```

Incidentally, if you would prefer that packages not be installed to your roaming profile (because you have a quota on your shared network, or it makes logging in or out from a domain sluggish), you can put it in your local app data instead:

```
npm config set prefix %LOCALAPPDATA%\npm -g
```

...as well as copying `%APPDATA%\npm` to `%LOCALAPPDATA%\npm` (and updating your `%PATH%` , of course).

Everyone who works on npm knows that this process is complicated and fraught, and we're working on making it simpler. Stay tuned.

Edit this page on GitHub