Installation troubleshooting

Redirecting to http://loopback.io/doc/en/lb2/Installation-troubleshooting.html ...

Make sure you have the latest version of Node

Ensure you have the latest stable version of Node as stated on http://nodejs.org/.

Uninstall old versions

If you installed StrongLoop software prior to Aug. 6, 2014, you must uninstall and then re-install the software. See Updating to the latest version.

Make sure you have sufficient file privileges

> ⊘ Changing privileges like this is appropriate *only* on your local development system. Never do this on a server system.

To install Node and StrongLoop , you need permissions to write to directories:

- `/usr/local/bin`
- `/usr/local/lib/node_modules`

If you see errors such as:

```
npm ERR! Error: EACCES, mkdir '/usr/local/lib/node_modules/strongloop'
...
npm ERR! Please try running this command again as root/Administrator
...
```

Then you don't have the required rights to create files or directories. Either change the rights for the specified directories, or run the command using `sudo`. In general, it's better to fix the directory rights as follows:

```
$ sudo chown -R $USER /usr/local
```

This command makes your user account the owner of the `/usr/local` directory. Then you won't ever have to use `sudo` to install Node or install packages globally with `npm`. For more information, see How to Node.

> ⊘ **DO NOT** use the above `chown` command on the `/usr/bin` directory. Doing so can severely misconfigure your system.

If you have to use `sudo`, use the following command:

```
$ sudo npm install -g --unsafe-perm install strongloop
```

Windows issues

Cygwin not supported

LoopBack does not support Cygwin (Windows bash shell emulator), because Node does not support it for interactive prompts. Use Windows command shell instead.

Installation errors

If you want to use LoopBack and the [slc loopback](#) command-line tool on Windows, but encounter installation errors running `npm install -g strongloop`, use the following workaround:

```
$ npm install -g yo
$ npm install -g generator-loopback
```

Then, instead of `slc loopback`, use `yo loopback` instead, including for all the generators, for example `yo loopback:model`. For more information on Yeoman, see http://yeoman.io/.

Xcode license issues

If you see errors such as:

```
Agreeing to the Xcode/iOS license requires admin privileges, please re-run as root via sudo.
```

Then you recently upgraded or installed Xcode and haven't agreed to the license yet.

Enter the following command to validate your Xcode license, then reinstall StrongLoop:

```
$ sudo xcode-select
```

Errors with npm

Peer dependency errors

If you encounter `peerDependency` conflicts with modules that are already installed globally either through `npm install -g` or `npm link`, you may have conflicting versions of modules installed globally. Inspect your global Node module directory (typically `/usr/local/lib/node_modules`), remove the conflicting modules manually, then re-install.

Firewall issues

A firewall may block npm installation because it blocks `git://` URLs. You can configure Git to use HTTPS instead as follows:

```
$ git config --global url."https://".insteadOf git://
```

See [Git is blocked, how to install npm modules (StackOverflow)](#) for more details.

Alternatives to public npm registry

Occasionally, the npm package manager will be down or otherwise generate error messages during installation. When this occurs, you can wait for the npm problem to be resolved, or:

- Use an alternative registry
- Install modules from GitHub

Use alternative registry

In addition to the official npm registry at http://registry.npmjs.org, there are several alternative npm registries:

- http://registry.npmjs.org.au (Australia)
- https://registry.nodejitsu.com (US East)
- http://registry.cnpmjs.org (China)

The unofficial registries are not always updated in realtime but may work when the official npm registry is down or overloaded. You can use [npmrc](#) to manage different configurations.

To install StrongLoop software from a different registry:

```
$ npm —registry <registry_URL> install strongloop
```

One you have installed `slc`, you can do more sophisticated registry management using the `slc registry` command. See Using multiple package registries for more information.

Install modules from Github

As a last resort, you can search the name of your package on GitHub and either clone with `git clone` or download the tarball, unzip and copy it in your **node_modules** folder.

Errors on Ubuntu

You may see the following errors when installing on Ubuntu:

```
sqlite3@3.1.1 install
/usr/local/lib/node_modules/strong-pm/node_modules/minkelite/node_modules/sqlite3
node-pre-gyp install --fallback-to-build

/usr/bin/env: node: No such file or directory
npm WARN This failure might be due to the use of legacy binary "node"
npm WARN For further explanations, please read
/usr/share/doc/nodejs/README.Debian
npm ERR! weird error 127
npm ERR! not ok code 0
```

To fix this, enter the following command:

```
$ update-alternatives --install /usr/bin/node node /usr/bin/nodejs 99
```