

1. 命令

- npm link
 - 在全局环境下，生成一个符号链接文件，该文件的名字就是package.json文件中指定的模块名
 - npm unlink pkg-name - 取消link
- npm adduser
 - 格式：npm adduser [--registry=url][--scope=@orgname][--always-auth][--auth-type=legacy]
 - To reset your password, go to <https://www.npmjs.com/forgot>
 - To change your email address, go to <https://www.npmjs.com/email-edit>
- npm login
 - 和adduser一样

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

2. npm配置优先级

2.1 命令行

2.1.1 命令行格式

- `--flag1 --flag2` will set both configuration parameters to `true`
- `--flag1 --flag2 bar` will set `flag1` to `true`, and `flag2` to `bar`
- `--flag1 --flag2 -- bar` will set both configuration parameters to `true`, and the `bar` is taken as a command argument

2.2 环境变量

- 所有以 `npm_config_` 开头的环境变量都会被读取，eg: `npm_config_foo=bar`
 - 如果没设置，则默认赋值 `true`
 - 环境变量是大小写不敏感的，但是npm-scripts习惯使用小写，并覆盖你设置的大写的环境变量

However, please note that inside `npm-scripts` npm will set its own environment variables and Node will prefer those lowercase versions over any uppercase ones that you might set. For details see [this issue](#).

- 单词之间使用下划线分隔（而不是连字符）

Notice that you need to use underscores instead of dashes, so `--allow-same-version` would become `npm_config_allow_same_version=true`.

2.3 npmrc File

- per-project configuration file (`/path/to/my/project/.npmrc`)
- per-user configuration file (defaults to `$HOME/.npmrc` ; configurable via CLI option `--userconfig` or environment variable `$NPM_CONFIG_USERCONFIG`)
- global configuration file (defaults to `$PREFIX/etc/npmrc` ; configurable via CLI option `--globalconfig` or environment variable `$NPM_CONFIG_GLOBALCONFIG`)
- npm's built-in configuration file (`/path/to/npm/npmrc`)

2.4 默认配置

2.4.1 查看命令

- `npm config ls -l`

3. 概念

3.1 Scope

3.1.1 格式

符号@'开始，/'结束

```
@somescope/somepackagename
```

3.2 机制介绍

- 任何 **用户 / 组织** 都可以拥有自己的scope，只有自己可以发布package到自己的scope下
 - 如此，也是有利于区分官方包
- scope包和非scope包，可以相互依赖
- **npm client** 可以逆向兼容非scope包（同时支持scope包）