

## Nvm

开源地址: <https://github.com/coreybutler/nvm-windows/releases>

The screenshot shows the GitHub repository for `nvm-windows` by `coreybutler`. The release `v1.1.11` is highlighted as the latest version, dated April 13. A terminal window displays the output of the `nvm debug` command, showing the current configuration and no problems detected. The 'What's Changed' section lists updates to reference links and a fix for the `install.cmd` PATH bug. Below the release details, a table lists the repository's files and their commit history.

File	Commit Message	Time Ago
<code>.github</code>	Bump actions/stale from 7 to 8	3 months ago
<code>bin</code>	Merge pull request #459 from noelhibbard/installer-improvements	4 months ago
<code>buildtools</code>	Fix Buildtools for 64bit systems by updating binaries	2 years ago
<code>examples</code>	Added examples for manual installation	9 years ago
<code>src</code>	Updated go-where dependency and cleaned up debug output.	2 months ago
<code>updater</code>	Updated version in updater.exe. Closes #874.	8 months ago
<code>utilities/pathfinder</code>	Added pathfinder notes	7 months ago
<code>.gitattributes</code>	executables should be binary	5 years ago
<code>.gitignore</code>	Added extra debug details.	2 months ago
<code>CONTRIBUTING.md</code>	Update CONTRIBUTING.md	4 years ago
<code>LICENSE</code>	Updated elevation approach	8 months ago
<code>README.md</code>	Added new check command for v1.1.11	5 months ago
<code>SUPPORT.md</code>	Create SUPPORT.md	3 years ago
<code>build.bat</code>	Add clearer logging to buildfile	2 years ago

The screenshot shows the 'Contributors' and 'Assets' sections of the `nvm-windows` repository. The 'Contributors' section lists `dauidribyrne`, `noelhibbard`, and 4 other contributors. The 'Assets' section lists various files for download, including `nvm-noinstall.zip`, `nvm-noinstall.zip.checksum.txt`, `nvm-setup.exe` (highlighted with a red box), `nvm-setup.zip`, `nvm-setup.zip.checksum.txt`, `nvm-update.zip`, `nvm-update.zip.checksum.txt`, `Source code (zip)`, and `Source code (tar.gz)`.

Asset	Size	Date
<code>nvm-noinstall.zip</code>	4.58 MB	Apr 13
<code>nvm-noinstall.zip.checksum.txt</code>	34 Bytes	Apr 13
<code>nvm-setup.exe</code>	5.47 MB	Apr 13
<code>nvm-setup.zip</code>	4.97 MB	Apr 13
<code>nvm-setup.zip.checksum.txt</code>	34 Bytes	Apr 13
<code>nvm-update.zip</code>	4.07 MB	Apr 13
<code>nvm-update.zip.checksum.txt</code>	34 Bytes	Apr 13
<code>Source code (zip)</code>		Apr 13
<code>Source code (tar.gz)</code>		Apr 13

只需点击第三个，下载安装包，下载之后点击安装，无需配置就可以使用，十分方便。

简单说明：

- `nvm-noinstall.zip`：这个是绿色免安装版本，但是使用之前需要配置
- `nvm-setup.zip`：这是一个安装包，下载之后点击安装，无需配置就可以使用，方便。
- `Source code(zip)`：zip压缩的源码
- `Source code(tar.gz)`：tar.gz的源码，一般用于\*nix系统

## ✧ 安装

- 第一安装目录：这里没有使用默认的安装目录，修改安装目录为 `D:\nvm\`；读者可自行决定安装目录
- 第二安装目录：添加 `node.js` 的安装目录 这里用 `添加 node.js 的安装目录` 的说法其实不太准确，准确的说法应该是 `配置 node.js 的安装目录`；`默认` 会使用 `C:\Program Files\nodejs` 目录，`C:\Program Files\nodejs` 目录也是 `node.js` 默认的安装目录；如果读者在安装 `nvm` 前就已经安装了 `node.js`，且修改了 `node.js` 的安装目录，那么此处的目录要修改为 `node.js` 的安装目录；如果读者在安装 `nvm` 前没有安装 `node.js`，那么此处的目录可自行定义
- 第三点击安装，在安装过程中会弹出：由于已经安装了 `node`，所以此时提示“你希望nvm管理已经安装的 `node` 版本吗”，点击 是待安装完成后测试是否安装成功

## 验证

```
1 nvm -v
```

```

Microsoft Windows [版本 10.0.22621.1848]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\AyoUser>nvm -V

Running version 1.1.11.

Usage:

  nvm arch                : Show if node is running in 32 or 64 bit mode.
  nvm current              : Display active version.
  nvm debug                : Check the NVM4W process for known problems (troubleshooter).
  nvm install <version> [arch] : The version can be a specific version, "latest" for the latest current version, or "lts"
for the                    : most recent LTS version. Optionally specify whether to install the 32 or 64 bit version
(defaults                  : to system arch). Set [arch] to "all" to install 32 AND 64 bit versions.
                           : Add --insecure to the end of this command to bypass SSL validation of the remote downlo
ad server.
  nvm list [available]     : List the node.js installations. Type "available" at the end to see what can be installe
d. Aliased as ls.
  nvm on                   : Enable node.js version management.
  nvm off                  : Disable node.js version management.
  nvm proxy [url]          : Set a proxy to use for downloads. Leave [url] blank to see the current proxy.
                           : Set [url] to "none" to remove the proxy.
  nvm node_mirror [url]    : Set the node mirror. Defaults to https://nodejs.org/dist/. Leave [url] blank to use def
ault url.
  nvm npm_mirror [url]     : Set the npm mirror. Defaults to https://github.com/npm/cli/archive/. Leave [url] blank
to default url.
  nvm uninstall <version> : The version must be a specific version.

```

## ✧ 命令使用

**nvm for windows** 是一个 **命令行工具**，在控制台输入 **nvm**，就可以看到它的命令用法。常用命令有：

命令	说明
<b>nvm list available</b>	显示可以安装的所有 <b>node.js</b> 的版本
<b>nvm list</b>	显示所有已安装的 <b>node.js</b> 版本
<b>nvm use</b>	切换到指定的 <b>nodejs</b> 版本
<b>nvm install</b>	安装指定版本的 <b>node.js</b> ，例如： <b>nvm install 8.12.0</b>
<b>nvm uninstall</b>	卸载指定版本的 <b>node.js</b> ，例如： <b>nvm uninstall 8.12.0</b>
<b>nvm on</b>	启用 <b>node.js</b> 版本管理
<b>nvm off</b>	禁用 <b>node.js</b> 版本管理(不卸载任何东西)

**i** 注意：在使用**nvm-window**时得环境变量的配置，尤其**Path**是否正确。

修改**nvm**下载源

在**nvm**目录中找到 **setting.txt** 添加以下代码

```

1 node_mirror: https://npm.taobao.org/mirrors/node/
2 npm_mirror: https://npm.taobao.org/mirrors/npm/

```

## \* npm管理

使用 `nvm` 时，默认的 `prefix` 是当前激活的 Node.js 版本的安装路径。

带来一个问题是：切换版本之后，之前安装全局命令模块需要重新安装，非常不方便。

解决方案是配置统一的全局模块安装路径。

新建 `npm_global` 和 `npm_cache` 文件夹，分别用于 npm 包的全局安装路径和全局 cache 路径

### npm 查看各种全局路径的命令

- 查看当前 npm 包的全局安装路径

```
1 npm prefix -g
```

- 查看当前 npm 包的全局 cache 路径

```
1 npm config get cache
```

- 查看配置列表

```
1 npm config ls
```

- 查看配置列表的全部信息

```
1 npm config ls -l
```

**【注】** 每次使用 `nvm` 切换 node 版本，最好都查看一下 npm 全局配置路径是否失效

### npm 修改全局路径命令

- 修改 npm 的包的全局安装路径

```
1 npm config set prefix "E:\NodeJs\npm\npm_global"
```

- 修改 npm 的包的全局 cache 位置

```
1 npm config set cache "E:\NodeJs\npm\npm_cache"
```

### 配置环境变量

将 `npm` 包全局安装路径配置在环境变量中

此电脑 -> 属性 -> 高级系统设置 -> 环境变量 -> 系统变量 -> path -> 编辑 -> 新增路径 - `E:\NodeJs\npm\npm_global` (路径可以根据 `npm prefix -g` 查看)

## \* yarn 管理

---

- 安装 yarn

```
1 npm install yarn -g
2 yarn -v
```

【注】如果首次安装 `yarn` 后，运行 `yarn -v` 不能显示 `yarn` 的版本，可以重启一下终端再尝试

yarn 的默认缓存和存储包的路径都在 C 盘，所以最好在安装后也进行修改

### yarn 查看各种路径命令

- 查看 yarn 全局 bin 位置(prefix)

```
1 yarn global bin
```

- 查看 yarn 全局安装位置(folder)

```
1 yarn global dir
```

- 查看 yarn 全局 cache 位置(cache)

```
1 yarn cache dir
```

- 查看配置列表

```
1 yarn config list
```

### yarn 修改路径命令

- 改变 yarn 全局 bin 位置(prefix)

```
1 yarn config set prefix "E:\NodeJs\npm\yarn_bin"
```

- 改变 yarn 全局安装位置(folder)

```
1 yarn config set global-folder "E:\NodeJs\npm\yarn_dir"
```

- 改变 yarn 全局 cache 位置(cache)

```
1 yarn config set cache-folder "E:\NodeJs\npm\npm_cache"
```

## 配置环境变量

将 `E:\NodeJs\npm\yarn_bin` 添加到环境变量的`path`变量中，若该目录下自动生成了 `bin` 目录，则添加 `E:\NodeJs\npm\yarn_bin\bin` 到环境变量中