# **[一个简单的零配置命令行HTTP服务器 - http-server (nodeJs)](http://www.cnblogs.com/lucker/p/4108838.html)**

http-server 是一个简单的零配置命令行HTTP服务器, 基于 nodeJs.

如果你不想重复的写 nodeJs 的 web-server.js, 则可以使用这个.

**安装** (全局安装加 -g) :

npm install http-server

**Windows 下使用**:

在站点目录下开启命令行输入

http-server

访问: [http://localhost:8080](http://localhost:8080/) or [http://127.0.0.1:8080](http://127.0.0.1:8080/" \t "http://www.cnblogs.com/lucker/p/_blank)

使用于package.json

"scripts": { "start": "http-server -a 0.0.0.0 -p 8000",

}

**参数** :

[IMG_256](http://www.cnblogs.com/lucker/p/javascript:void(0);)

-p 端口号 (默认 8080)

-a IP 地址 (默认 0.0.0.0)

-d 显示目录列表 (默认 'True')

-i 显示 autoIndex (默认 'True')

-e or --ext 如果没有提供默认的文件扩展名(默认 'html')

-s or --silent 禁止日志信息输出

--cors 启用 CORS via the Access-Control-Allow-Origin header

-o 在开始服务后打开浏览器

-c 为 cache-control max-age header 设置Cache time(秒) , e.g. -c10 for 10 seconds (defaults to '3600'). 禁用 caching, 则使用 -c-1.

-U 或 --utc 使用UTC time 格式化log消息  
  
-P or --proxy Proxies all requests which can't be resolved locally to the given url. e.g.: -P [http://someurl.com](http://someurl.com/)  
  
-S or --ssl 启用 https  
  
-C or --cert ssl cert 文件路径 (default: cert.pem)  
  
-K or --key Path to ssl key file (default: key.pem).  
  
-r or --robots Provide a /robots.txt (whose content defaults to 'User-agent: \*\nDisallow: /')  
  
-h or --help 打印以上列表并退出

[IMG_257](http://www.cnblogs.com/lucker/p/javascript:void(0);)

 引用文档: [a simple zero-configuration command-line http server](https://www.npmjs.org/package/http-server" \o "http-server" \t "http://www.cnblogs.com/lucker/p/_blank)