



Curl 工具简介

By 陈炜健

🕒 发表于 2015-10-19

前言

最近正在上分布式计算的课程，第一阶段我们需要学习基于 Web 服务的编程，而第一节课作业中有要求我们了解 Curl 工具的使用，因此顺便写一篇博客简单地介绍 Curl 工具的用法。

定义

- **WIKI**

cURL is a computer software project providing a library and command-line tool for transferring data using various protocols. The cURL project produces two products, libcurl and cURL. It was first released in 1997. The name originally stood for “see URL”.

- **官方**

curl is an open source command line tool and library for transferring data with URL syntax, supporting DICT, FILE, FTP, FTPS, Gopher, HTTP, HTTPS, IMAP, IMAPS, LDAP, LDAPS, POP3, POP3S, RTMP, RTSP, SCP, SFTP, SMB, SMTP, SMTPS, Telnet and TFTP. curl supports SSL certificates, HTTP POST, HTTP PUT, FTP uploading, HTTP form based upload, proxies, HTTP/2, cookies, user+password authentication (Basic, Plain, Digest, CRAM-MD5, NTLM, Negotiate and Kerberos), file transfer resume, proxy tunneling and more.

根据解释，**Curl** 是一个开源的命令行工具和库（**Library**），利用 **URL** 语法，实

文章目录

- 1. 前言
- 2. 定义
- 3. 用法
 - 3.1. 抓取网页信息
 - 3.2. 发送表单数据
 - 3.3. HTTP 认证
 - 3.4. 设置 Referer
 - 3.5. 设置 User Agent
 - 3.6. 设置 Cookies
 - 3.7. 重定向 Redirects
- 4. P.S.
- 5. 参考资料

现对 URL 进行访问操作及传输数据的功能。它支持的通讯协议有 DICT、FILE、FTP、FTPS、Gopher、HTTP、HTTPS、IMAP、IMAPS、LDAP、LDAPS、POP3、POP3S、RTMP、RTSP、SCP、SFTP、SMB、SMTP、SMTPS、Telnet 和 TFTP。

依我看，Curl 是一个很成功的开源项目（[License](#)），从 1997 年发布以来，已经有上千名开发者对该项目作出过贡献，适用于各式各样的操作系统。具体代码可在 [Github](#) 上参阅。

用法

既然 Curl 是一个实用的工具，那我还是不讲消遣的话，直接列举一些常用的命令用法吧。接下来我将介绍 Curl 的七个功能，主要参考自官方文档「[Using cURL to automate HTTP jobs](#)」这一篇 Tutorial 文章。

抓取网页信息

- 不添加任何参数，直接在终端显示 HTML 源码：

```
curl [URL]
```

- 添加 `-i` 或 `--include` 参数，同时取得源码与 HTTP-header 的信息：

```
curl -i [URL]
```

- 添加 `-I` 或 `--head` 参数，只显示 HTTP-header 的信息：

```
curl -I [URL]
```

- 添加 `-v` 或 `--verbose` 参数，可显示 Curl 工具与 Server 通信过程的信息：

```
curl -v [URL]
```

- 如果还是觉得以上方法显示的信息不够...还可以选择添加 `--trace` 或 `--trace-ascii` 参数，显示更多细节，且可保存在本地：

```
curl --trace output.txt [URL]
```

```
curl --trace-ascii output.txt [URL]
```

- 添加 `-o` 或 `--output` 参数以及文件名，自定义文件名保存源码在本地（亦可用于下载文件）：

```
curl [URL] -o index.html
```

- 添加 `-O` 或 `--remote-name` 参数，提取 URL 文件名并下载文件到本地：

```
curl -O [URL]
```

- 可同时抓取多个 URL 地址，依次显示：

```
curl [URL-1] [URL-2]
```

发送表单数据

- Get 形式，把数据信息附着在 URL 的后面（注意添加 `"`）：

形式：

```
curl "http://www.example.com/test-form?data1=XXX&data2"
```

例子：

```
curl "http://www.hotmail.com/when/junk.cgi?birthyear=1905&press=OK"
```

- Post 形式，数据与 URL 是分开的，我们需要添加 `-d` 或 `--data` 参数：

形式：

```
curl --data "data1=xxx&data2=xxx" [URL]
```

例子：

```
curl --data "birthyear=1905&press=%20OK%20" http://www.example.com/when.cgi
```

添加 `--data-urlencode`，还能让 Curl 对表带数据进行编码，比如发送数据为「Hello World」时，需把空格变为「%20」。

```
curl --data-urlencode "data1=xxx&data2=xxx" [URL]
```

- 上传文件（以 Post 形式），假设有如下表单：

```
1 <form method="POST" enctype='multipart/form-data' action="upload.cgi">
2 <input type=file name=upload>
3 <input type=submit name=press value="OK">
4 </form>
```

那么 Curl 命令格式应为：

```
curl --form upload=@localfilename --form press=OK [URL]
```

HTTP 认证

HTTP 认证是 Web 服务器对客户端的权限进行认证的一种方式，能够为 Web 应用提供一定程度的安全保障。HTTP 认证采用「质询-响应」（challenge-response）的机制。「质询」是服务器端对客户端的质询，即要求客户端发送认证信息；「响应」是客户端对「质询」的响应，即发送带有认证信息的 HTTP 请求。

- 需添加 `--user` 参数：

```
curl --user name:password [URL]
```

设置 Referer

HTTP Referer 是 header 的一部分，当浏览器向 web 服务器发送请求的时候，一般会带上 Referer，告诉服务器我是从哪个页面链接过来的，服务器籍此可以获得一些信息用于处理。

有时候出于安全考虑，如防盗链，服务器会根据 Referer 来决定是否响应请求（尽管很容易被伪造...），所以添加 Referer 信息就显得很重要了。而 Curl 是支持这一功能的。

- 添加 `-e` 或 `--referer` 参数以及 Referer：

```
curl -e [URL-referer] [URL]
```

设置 User Agent

就像 Referer 一样，很多 HTTP 请求中都会包含 User Agent 信息，它表示客户端的设备信息。现在很多网页的服务器都会针对不同设备返回不同格式的页面，以达到页面显示的最好效果。如手机移动端的页面与桌面网页端的页面。

- 添加 `-A` 或 `--user-agent` 参数以及设备信息：

```
curl -A "[User Agent]" [URL]
```

设置 Cookies

Cookie 对象使用 key-value 属性对的形式保存用户状态，一个 Cookie 对象保存一个属性对，一个 request 或者 response 同时使用多个 Cookie。

- 添加 `-b` 或 `--cookie` 参数，指定cookie，多个 cookies 时用分号 `;` 间隔：

```
curl -b "user=xxx;pass=xxx" [URL]
```

- 添加 `-D` 或 `--dump-header` 参数，保存通常会含有 Cookie 信息的 HTTP-header 数据到本地：

形式：

```
curl -D [File-name] [URL]
```

例子：

```
curl -D header.txt eleveneat.com
```

可在 `-b` 参数后加上保存 HTTP-header 数据的文件，Curl 便会自动从文件中读取 Cookie 信息：

```
curl -b [File-name] [URL]
```

重定向 Redirects

有时候利用 Curl 访问一些网页，会显示出「3XX Moved Permanently」的信息反馈，其实这代表网页移动到别的地方了，通常在 HTTP 响应中会有 Location 字段提供。重定向出现的原因一般是因为网站结构调整后，避免用户访问原来页面会出现 404 错误。

- 添加 `-L` 或 `--location` 参数，自动追踪重定向：

```
curl -L [URL]
```

P.S.

其实除了以上提到的功能，Curl 还有不少强大的命令，待我日后使用过程中再慢慢探索，争取补充上去。

BTW，也许以后我应该写一篇博客对比一下 wget 与 curl 的区别，两者似乎有很多共同点。

END.

参考资料

- [cURL From WIKI](#)
- [Curl From Curl](#)
- [Curl License From Curl](#)
- [Curl Code From Github](#)
- [Using cURL to automate HTTP jobs From Curl](#)
- [curl网站开发指南](#) By 阮一峰
- [Curl 使用总结](#) By 文刀旋子

工具



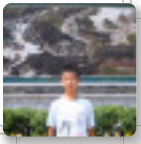
下一篇:

➤ [密码学简介 \(未完成\)](#)

喜欢

2 条评论

最新 最早 最热



小郭
炜健，那个-v的那条命令的例子是不是有问题？
5小时前 回复 顶 转发



陈炜健
回复 小郭: 我的错...已改~
1分钟前 回复 顶 转发

陈炜健 帐号管理



☒ 分享到:

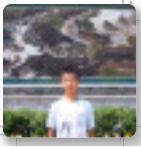
发布

eleveneat正在使用多说

喜欢

2 条评论

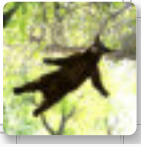
最新 最早 最热



小郭

炜健，那个-v的那条命令的例子是不是有问题？

5小时前 回复 顶 转发

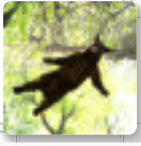


陈炜健

回复 小郭: 我的错...已改~

1分钟前 回复 顶 转发

陈炜健 帐号管理



🔍 写点什么吧

☒ 分享到:

发布

eleveneat正在使用多说

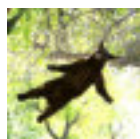
分类

- Coursera²
- Web³
- 工具¹
- 计算机原理³
- 随笔²

标签

- Software Security³
- Web安全²
- 东野圭吾¹
- The Beetles¹
- Hexo¹

新浪微博



[_陈炜健_](#) 广东 广州

✓ 我自己

哈哈哈哈哈//@马伯庸:这个梗太棒，仅次于那个反严刑拷打的海报

谷大白话 **V**：【上帝欢迎斯诺登】斯诺登开推特之后，“上帝”发推表示：“爱德华斯诺登，欢迎你来推特。我看你只关注了一个账号(NSA美国国家安全局)。我也是。但我关注的那个要邪恶得多。”😂



RSS 订阅



Hi, I'm Eleveeat, an undergraduate studying at Sun Yat-sen university.
Hope you enjoy my blog.



知



Powered by [hexo](#) and Theme by [Jacman](#) © 2015 陈炜健