

# yanmiao

博客园 :: 首页 :: 新随笔 :: 联系 :: 订阅  :: 管理

posts - 21, comments - 2, trackbacks - 0

## 公告

昵称: yanmiao  
园龄: 6年8个月  
粉丝: 1  
关注: 9  
[+加关注](#)

2017年5月								
<	日	一	二	三	四	五	六	>
	30	1	2	3	4	5	6	
	7	8	9	10	11	12	13	
	14	15	16	17	18	19	20	
	21	22	23	24	25	26	27	
	28	29	30	31	1	2	3	
	4	5	6	7	8	9	10	

## 搜索


  



## 常用链接

- [我的随笔](#)
- [我的评论](#)
- [我的参与](#)
- [最新评论](#)
- [我的标签](#)

## 我的标签

- [rails\(3\)](#)
- [ruby\(3\)](#)

## 随笔分类

[移动ui](#)

## 随笔档案

[2013年10月 \(1\)](#)

## mac curl 使用方法(转载)

[查看原文章](#)

CURL? 嗯, 说来话长了~~~~

这东西现在已经是苹果机上内置的命令行工具之一了, 可见其魅力之一斑

1)  
二话不说, 先从这里开始吧!

```
curl http://www.yahoo.com
```

回车之后, www.yahoo.com 的html就稀里哗啦地显示在屏幕上了~~~~~

2)  
嗯, 要想把读过来页面存下来, 是不是要这样呢?  
curl http://www.yahoo.com > page.html

当然可以, 但不用这么麻烦的!  
用curl的内置option就好, 存下http的结果, 用这个option: -o  
curl -o page.html http://www.yahoo.com

这样, 你就可以看到屏幕上出现一个下载页面进度指示。等进展到100%, 自然就OK咯

3)  
什么什么?! 访问不到? 肯定是你的proxy没有设定了。  
使用curl的时候, 用这个option可以指定http访问所使用的proxy服务器及其端口: -x  
curl -x 123.45.67.89:1080 -o page.html http://www.yahoo.com

4)  
访问有些网站的时候比较讨厌, 他使用cookie来记录session信息。

2013年8月 (1)  
 2013年7月 (1)  
 2013年6月 (1)  
 2012年11月 (1)  
 2012年10月 (2)  
 2012年5月 (3)  
 2012年4月 (3)  
 2012年3月 (1)  
 2012年2月 (1)  
 2011年11月 (1)  
 2011年9月 (1)  
 2011年8月 (1)  
 2011年7月 (1)  
 2010年10月 (2)

#### 最新评论

1. Re:Excel2003和excel2007读取方法

@Robert·Hou那你是否注册"microsoft.ACE.oledb.12.0"了呢? ...

--yanmiao

2. Re:Excel2003和excel2007读取方法

貌似不太可以哦, 我打开了一个excel2007的文件, Conn.Open();就错了

--Robert·Hou

#### 阅读排行榜

1. centos安装与卸载 postgresql(3079)  
 2. 在ASP.NET MVC 中获取当前URL、controller、action(转载)(1254)  
 3. mac curl 使用方法(转载)(698)  
 4. rails处理时间以及MYSQL处理时间(675)  
 5. rails中获取url信息的一些方法(590)

#### 评论排行榜

1. Excel2003和excel2007读取方法(2)

#### 推荐排行榜

1. 在ASP.NET MVC 中获取当前URL、controller、action(转载)(1)

像IE/NN这样的浏览器, 当然可以轻易处理cookie信息, 但我们的curl呢? ....

我们来学习这个option: -D <-- 这个是把http的response里面的cookie信息存到一个特别的文件中

```
curl -x 123.45.67.89:1080 -o page.html -D cookie0001.txt
http://www.yahoo.com
```

这样, 当页面被存到page.html的同时, cookie信息也被存到了cookie0001.txt里面了

5)

那么, 下一次访问的时候, 如何继续使用上次留下的cookie信息呢? 要知道, 很多网站都是靠监视你的cookie信息, 来判断你是不是不按规矩访问他们的网站的。

这次我们使用这个option来把上次的cookie信息追加到http request里面去: -b

```
curl -x 123.45.67.89:1080 -o page1.html -D cookie0002.txt -b
cookie0001.txt http://www.yahoo.com
```

这样, 我们就可以几乎模拟所有的IE操作, 去访问网页了!

6)

稍微等等~~~~~我好像忘记什么了~~~~~  
 对了! 是浏览器信息~~~~~

有些讨厌的网站总要我们使用某些特定的浏览器去访问他们, 有时候更过分的是, 还要使用某些特定的版本~~~~~

NND, 哪里有时间为了它去找这些怪异的浏览器呢! ?

好在curl给我们提供了一个有用的option, 可以让我们随意指定自己这次访问所宣称的自己的浏览器信息: -A

```
curl -A "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)" -x
123.45.67.89:1080 -o page.html -D cookie0001.txt
http://www.yahoo.com
```

这样, 服务器端接到访问的要求, 会认为你是一个运行在Windows 2000上的IE6.0, 嘿嘿嘿, 其实也许你用的是苹果机呢!

而"Mozilla/4.73 [en] (X11; U; Linux 2.2; 15 i686)"则可以告诉对方你是一台PC上跑着的Linux, 用的是Netscape 4.73, 呵呵呵

7)

另外一个服务器端常用的限制方法, 就是检查http访问的referer。比如你先访问首页, 再访问里面所指定的下载页, 这第二次访问的 referer地址就是第一次访问成功后的页面地址。这样, 服务器端只要发现对下载页面某次访问的 referer地址不是首页的地址, 就可以断定那是个盗连了~~~~~

讨厌讨厌~~~我就是要盗连~~~~~!!

幸好curl给我们提供了设定referer的option: -e

```
curl -A "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)" -x
123.45.67.89:1080 -e "mail.yahoo.com" -o page.html -D
cookie0001.txt http://www.yahoo.com
```

这样，就可以骗对方的服务器，你是从mail.yahoo.com点击某个链接过来的了，呵呵呵

8)

写着写着发现漏掉什么重要的东西了！----- 利用curl 下载文件

刚才讲过了，下载页面到一个文件里，可以使用 -o ，下载文件也是一样。比如， curl -o 1.jpg http://cgi2.tky.3web.ne.jp/~zzh/screen1.JPG 这里教大家一个新的option： -O 大写的O，这么用： curl -O http://cgi2.tky.3web.ne.jp/~zzh/screen1.JPG 这样，就可以按照服务器上的文件名，自动存在本地了！

再来一个更好用的。

如果screen1.JPG以外还有screen2.JPG、screen3.JPG、.....、screen10.JPG需要下载，难不成还要让我们写一个script来完成这些操作？不干！

在curl里面，这么写就可以了：

```
curl -O http://cgi2.tky.3web.ne.jp/~zzh/screen[1-10].JPG
```

呵呵呵，厉害吧？！~~~

9)

再来，我们继续讲解下载！

```
curl -O http://cgi2.tky.3web.ne.jp/~/[001-201].JPG
```

这样产生的下载，就是

```
~zzh/001.JPG
```

```
~zzh/002.JPG
```

```
...
```

```
~zzh/201.JPG
```

```
~nick/001.JPG
```

```
~nick/002.JPG
```

```
...
```

```
~nick/201.JPG
```

够方便的了把？哈哈

咦？高兴得太早了。

由于zzh/nick下的文件名都是001，002....，201，下载下来的文件重名，后面的把前面的文件都给覆盖掉了~~~

没关系，我们还有更狠的！

```
curl -o #2_#1.jpg http://cgi2.tky.3web.ne.jp/~/[001-201].JPG
```

--这是.....自定义文件名的下载？

--对头，呵呵！

#1是变量，指的是这部分，第一次取值zzh，第二次取值nick

#2代表的变量，则是第二段可变部分---[001-201]，取值从001逐一加到201

这样，自定义出来下载下来的文件名，就变成了这样：

原来： ~zzh/001.JPG ---> 下载后： 001-zzh.JPG

原来: ~nick/001.JPG ---> 下载后: 001-nick.JPG

这样一来, 就不怕文件重名啦, 呵呵

9)

继续讲下载

我们平时在windows平台上, flashget这样的工具可以帮我们分块并行下载, 还可以断线续传。

curl在这些方面也不输给谁, 嘿嘿

比如我们下载screen1.JPG中, 突然掉线了, 我们就可以这样开始续传

```
curl -c -O http://cgi2.tky.3wb.ne.jp/~zzh/screen1.JPG
```

当然, 你不要拿个flashget下载了一半的文件来糊弄我~~~~别的下载软件的半截文件可不一定能用哦~~~

分块下载, 我们使用这个option就可以了: -r

举例说明

比如我们有一个http://cgi2.tky.3web.ne.jp/~zzh/zhao1.mp3 要下载  
(赵老师的电话朗诵)

我们就可以用这样的命令:

```
curl -r 0-10240 -o "zhao.part1"  
http://cgi2.tky.3web.ne.jp/~zzh/zhao1.mp3 &\  
curl -r 10241-20480 -o "zhao.part1"  
http://cgi2.tky.3web.ne.jp/~zzh/zhao1.mp3 &\  
curl -r 20481-40960 -o "zhao.part1"  
http://cgi2.tky.3web.ne.jp/~zzh/zhao1.mp3 &\  
curl -r 40961- -o "zhao.part1"  
http://cgi2.tky.3web.ne.jp/~zzh/zhao1.mp3
```

这样就可以分块下载啦。

不过你需要自己把这些破碎的文件合并起来

如果你用UNIX或苹果, 用 `cat zhao.part* > zhao.mp3` 就可以

如果用的是Windows, 用 `copy /b` 来解决吧, 呵呵

上面讲的都是http协议的下载, 其实ftp也一样可以用。

用法嘛,

```
curl -u name:passwd ftp://ip:port/path/file
```

或者大家熟悉的

```
curl ftp://name:passwd@ip:port/path/file
```

10)

说完了下载, 接下来自然该讲上传咯

上传的option是 -T

比如我们向ftp传一个文件: `curl -T localfile -u name:passwd  
ftp://upload_site:port/path/`

当然, 向http服务器上传文件也可以

比如 `curl -T localfile http://cgi2.tky.3web.ne.jp/~zzh/abc.cgi`

注意, 这时候, 使用的协议是HTTP的PUT method

刚才说到PUT，嘿嘿，自然让老服想起来了其他几种method还没讲呢！GET和POST都不能忘哦。

http提交一个表单，比较常用的是POST模式和GET模式

GET模式什么option都不用，只需要把变量写在url里面就可以了  
比如：

```
curl http://www.yahoo.com/login.cgi?
user=nickwolfe&password=12345
```

而POST模式的option则是 -d

```
比如， curl -d "user=nickwolfe&password=12345"
http://www.yahoo.com/login.cgi
就相当于向这个站点发出一次登陆申请~~~~~
```

到底该用GET模式还是POST模式，要看对面服务器的程序设定。

一点需要注意的是，POST模式下的文件上的文件上传，比如  
<form method="POST" enctype="multipart/form-data"  
action="http://cgi2.tky.3web.ne.jp/~zzh/up\_file.cgi">  
<input type=file name=upload>  
<input type=submit name=nick value="go">  
</form>

这样一个HTTP表单，我们要用curl进行模拟，就该是这样的语法：

```
curl -F upload=@localfile -F nick=go
http://cgi2.tky.3web.ne.jp/~zzh/up_file.cgi
```

罗罗嗦嗦讲了这么多，其实curl还有很多很多技巧和用法  
比如 https的时候使用本地证书，就可以这样  
curl -E localcert.pem https://remote\_server

再比如，你还可以用curl通过dict协议去查字典~~~~~  
curl dict://dict.org/d:computer

好文要顶

关注我

收藏该文



yanmiao

关注 - 9

粉丝 - 1

0

0

+加关注

« 上一篇：解决 Flash 盖住弹出目录的方法(转载)

» 下一篇：Linux SSH远程文件/目录传输命令scp

posted on 2012-02-09 10:26 yanmiao 阅读(698) 评论(0) 编辑 收藏

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】50万行VC++源码：大型组态工控、电力仿真CAD与GIS源码库

【推荐】中铁、中石油等大型企业的复杂报表解决方案

【活动】 阿里云海外云服务全面降价助力企业全球布局

【实用】 40+篇云服务器操作及运维基础知识!



**最新IT新闻:**

- 离职者说: 两载从生到死, ZUK做错了什么
  - 易到回应老员工群发邮件炮轰乐视: 邮箱系统被黑
  - Buzz Aldrin呼吁NASA尽快退役国际空间站, 转向火星
  - 小蓝单车接入支付宝十天后用户数激增至1300万
  - 随着生物辨识技术兴起, 密码的「死期」可能即将到来
- » 更多新闻...



**最新知识库文章:**

- 软件开发为什么很难
  - 唱吧DevOps的落地, 微服务CI/CD的范本技术解读
  - 程序员, 如何从平庸走向理想?
  - 我为什么鼓励工程师写blog
  - 怎么轻松学习JavaScript
- » 更多知识库文章...

Powered by:



Copyright ©2017 yanmiao