

CURL 发送POST请求

© 发表于 2020-02-17 13:16 ■ 阅读次数:280688 评论次数:10

LINUX知识积累

上面命令添加 HTTP 标头 `Accept-Language: en-US`。

```
1 $ curl -H 'Accept-Language: en-US' -H 'Secret-Message: xyzzy' https://google.com
2
```

上面命令添加两个 HTTP 标头。

```
1 $ curl -d '{"login": "emma", "pass": "123"}' -H 'Content-Type: application/json' https://google.com/login
2
```

上面命令添加 HTTP 请求的标头是 `Content-Type: application/json`，然后用 `-d` 参数发送 JSON 数据。

-i

`-i` 参数打印出服务器回应的 HTTP 标头。

```
1 $ curl -i https://www.example.com
2
```

上面命令收到服务器回应后，先输出服务器回应的标头，然后空一行，再输出网页的源码。

-I

`-I` 参数向服务器发出 HEAD 请求，然会将服务器返回的 HTTP 标头打印出来。

```
1 $ curl -I https://www.example.com
2
```

上面命令输出服务器对 HEAD 请求的回应。

`--head` 参数等同于 `-I`。

```
1 $ curl --head https://www.example.com
2
```

-k

`-k` 参数指定跳过 SSL 检测。

```
1 $ curl -k https://www.example.com
2
```

上面命令不会检查服务器的 SSL 证书是否正确。

-L

`-L` 参数会让 HTTP 请求跟随服务器的重定向。curl 默认不跟随重定向。

```
1 $ curl -L -d 'tweet=hi' https://api.twitter.com/tweet
2
```

--limit-rate

`--limit-rate` 用来限制 HTTP 请求和回应的带宽，模拟慢网速的环境。

```
1 $ curl --limit-rate 200k https://google.com
2
```

上面命令将带宽限制在每秒 200K 字节。

-o

`-o` 参数将服务器的回应保存成文件，等同于 `wget` 命令。

```
1 $ curl -o example.html https://www.example.com
2
```

上面命令将 `www.example.com` 保存成 `example.html`。

-O

`-O` 参数将服务器回应保存成文件，并将 URL 的最后部分当作文件名。

```
1 $ curl -O https://www.example.com/foo/bar.html
2
```

上面命令将服务器回应保存成文件，文件名为 `bar.html`。

-s

-s 参数将不输出错误和进度信息。

```
1 | $ curl -s https://www.example.com
2 |
```

上面命令一旦发生错误，不会显示错误信息。不发生错误的话，会正常显示运行结果。

如果想让 `curl` 不产生任何输出，可以使用下面的命令。

```
1 | $ curl -s -o /dev/null https://google.com
2 |
```

-S

-S 参数指定只输出错误信息，通常与 **-s** 一起使用。

```
1 | $ curl -s -o /dev/null https://google.com
2 |
```

上面命令没有任何输出，除非发生错误。

-u

-u 参数用来设置服务器认证的用户名和密码。

```
1 | $ curl -u 'bob:12345' https://google.com/login
2 |
```

上面命令设置用户名为 `bob`，密码为 `12345`，然后将其转为 HTTP 标头 `Authorization: Basic Ym9lQjEyMzQ1`。

`curl` 能够识别 URL 里面的用户名和密码。

```
1 | $ curl https://bob:12345@google.com/login
2 |
```

上面命令能够识别 URL 里面的用户名和密码，将其转为上个例子里面的 HTTP 标头。

```
1 | $ curl -u 'bob' https://google.com/login
2 |
```

上面命令只设置了用户名，执行后，`curl` 会提示用户输入密码。

-v

-v 参数输出通信的整个过程，用于调试。

```
1 | $ curl -v https://www.example.com
2 |
```

--trace 参数也可以用于调试，还会输出原始的二进制数据。

```
1 | $ curl --trace - https://www.example.com
2 |
```

-X

-X 参数指定 HTTP 请求的代理。

```
1 | $ curl -x socks5://james:cats@myproxy.com:8080 https://www.example.com
2 |
```

上面命令指定 HTTP 请求通过 `myproxy.com:8080` 的 socks5 代理发出。

如果没有指定代理协议，默认为 HTTP。

```
1 | $ curl -x james:cats@myproxy.com:8080 https://www.example.com
2 |
```

上面命令中，请求的代理使用 HTTP 协议。

-X

-X 参数指定 HTTP 请求的方法。

```
1 | $ curl -X POST https://www.example.com
2 |
```

上面命令对 `https://www.example.com` 发出 POST 请求。

参考链接

- [Curl Cookbook](#)

__EOF__



本文作者: [Fantasy](#)
本文链接: <https://www.cnblogs.com/fan-gx/p/12321351.html>
关于博主: 评论和私信会在第一时间回复。
版权声明: 本文为博主呕心力作，喜欢请打赏，转载请注明出处。
声援博主: 如果您觉得文章对您有帮助，可以点击文章右下角【推荐】一下。您的鼓励是博主的最大动力！

分类: [Linux知识积累](#)

好文要顶

关注我

收藏该文

🔖

👤



博客萨斯州

粉丝 - 44 关注 - 33

+加关注

• 上一篇: [mysql的my.cnf配置参考](#)

• 下一篇: [Maven —— 命令行清除编译打包](#)

11

👍 推荐

0

👎 反对

登录后才能查看或发表评论 · 立即 [登录](#) 或者 [注册](#) 博客园首页

编辑推荐:

• [单标签实现复杂的棋盘布局](#)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

- C# 非托管资源中 HEAP_ENTRY 的 Size 对不上是怎么回事？
- CSS之垂直水平居中中的跨后
- 记一次 .NET 某打印服务 非托管内存泄漏分析
- 使用 Three.js 实现一个创意纪念页面

最新文章：

- 索尼预热 PlayStation VR2：新机将于明年初发布
- 羊了个羊，无法穿越的狭沟
- 在Web3，大厂是要被鄙视的
- 小马智行、曹操出行、吉利智驾中心达成战略合作，联合打造Robotaxi车队
- 普通人用了上瘾，艺术家看了流泪：这个爆火的AI真能一键挑战大师画作？

• 更多新闻...



[搞科学是需要人的全部生命 ♡八小时工作制是行不通的]

This blog has running : 2208 d 8 h 26 m 52 s (日) / 0
Copyright © 2022 博客萨斯州 Powered by .NET 6 on Kubernetes
Theme version: v1.2.6 / Loading theme version: v1.2.6

