IP代理

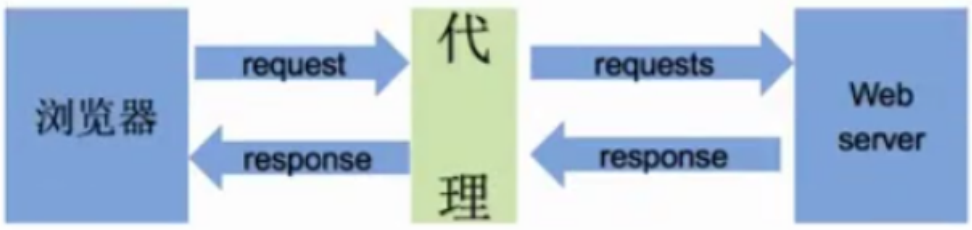
> proxy代理参数通过指定代理ip，让代理ip对应的正向代理服务器转发我们发送的请求，那么我们首先来了解一下代理ip以及代理服务器

学习目标：

- 掌握 代理ip参数proxies的使用
- 掌握 使用verify参数忽略CA证书
- 掌握 requests模块发送post请求
- 掌握 利用requests.session进行状态保持

理解使用代理的过程----<代理相当于中介>

1. 代理ip是一个ip，指向的是一个代理服务器
2. 代理服务器能够帮我们向目标服务器转发请求



1.浏览器发送请求给代理>>>>2.代理接收请求，转发请求>>>>3.服务器接收请求，返回响应给代理>>>>4.在由代理把响应转发给浏览器，在这里，代理相当于扮演中介的角色

正向代理和反向代理的区别

> 前边提到proxy参数指定的代理ip指向的是正向的代理服务器，那么相应的就有反向服务器；现在来了解一下正向代理服务器和反向代理服务器的区别

1. 正向代理：对于浏览器知道服务器的真实地址，例如vpn(虚拟专用网络)

   反向代理：浏览器不知道服务器的真实地址，例如nginx(一个高性能的HTTP和反向代理web服务器)

   - 从用途上来讲
     - 正向代理-为局域网客户端向外访问Internet服务，可以使用缓冲特性减少网络使用率。
     - 反向代理-为局域网服务器向外提供Internet服务，可以使用负载平衡提高客户访问量，还可以基于高级URL策略和管理技术对服务进行高质量管控。
     - 从安全性来讲
       - 正向代理-必须采取安全措施确保内网客户端通过它访问外部网站，隐藏客户端的身份。

- 反向代理-对外提供服务是透明的，客户端并不知道自己访问的是一个代理，隐藏服务端的身份。

1.代理ip（代理服务器）的分类

1. 根据代理ip的匿名程度，代理IP可以分为下面三类：
    - 透明代理(Transparent Proxy)：透明代理虽然可以直接“隐藏”你的IP地址，但是还是可以查到你是谁。目标服务器接收到的请求头如下：

```
REMOTE_ADDR = Proxy IP
HTTP_VIA = Proxy IP
HTTP_X_FORWARDED_FOR = Your IP
```

- 匿名代理(Anonymous Proxy)：使用匿名代理，别人只能知道你用了代理，无法知道你是谁。目标服务器接收到的请求头如下：

```
  REMOTE_ADDR = proxy IP
  HTTP_VIA = proxy IP
  HTTP_X_FORWARDED_FOR = proxy IP
```

- 高匿代理(Elite proxy或High Anonymity Proxy)：高匿代理让别人根本无法发现你是在用代理，所以是最好的选择。**毫无疑问使用高匿代理效果最好。**目标服务器接收到的请求头如下：

```
  REMOTE_ADDR = Proxy IP
  HTTP_VIA = not determined
  HTTP_X_FORWARDED_FOR = not
determined
```

2. 根据网站所使用的协议不同，需要使用相应协议的代理服务。从代理服务请求使用的协议可以分为：
   - http代理：目标url为http协议
     - https代理：目标url为https协议
     - socks隧道代理（例如socks5代理）等：
       1. socks 代理只是简单地传递数据包，不关心是何种应用协议（FTP、HTTP和HTTPS等）。
       2. socks 代理比http、https代理耗时少。
       3. socks 代理可以转发http和https的请求

2.proxies代理参数的使用

- 查看ip地址

1. ipconfig（内网的ip——私有地址）
2. ipip.net(查看外网的ip)（封的是外网的ip)
3. 目标网站：https://httpbin.org/ip

> 为了让服务器以为不是同一个客户端在请求；为了防止频繁向一个域名发送请求被封ip，所以我们需要使用代理ip；让服务器以为不是同一个客户在请求，防止我们的真实地址被泄露，防止被追究

- 用法：

```
  response = requests.get(url,
proxies=proxies)
```

- proxies的形式：字典
- 例如：

```
  proxies = {
      "http": "http://12.34.56.79:9527",
      "https":
"https://12.34.56.79:9527",
  }
```

- 注意：如果proxies字典中包含有多个键值对，发送请求时将按照url地址的协议来选择使用相应的代理ip

快代理国内代理平台：https://www.kuaidaili.com/

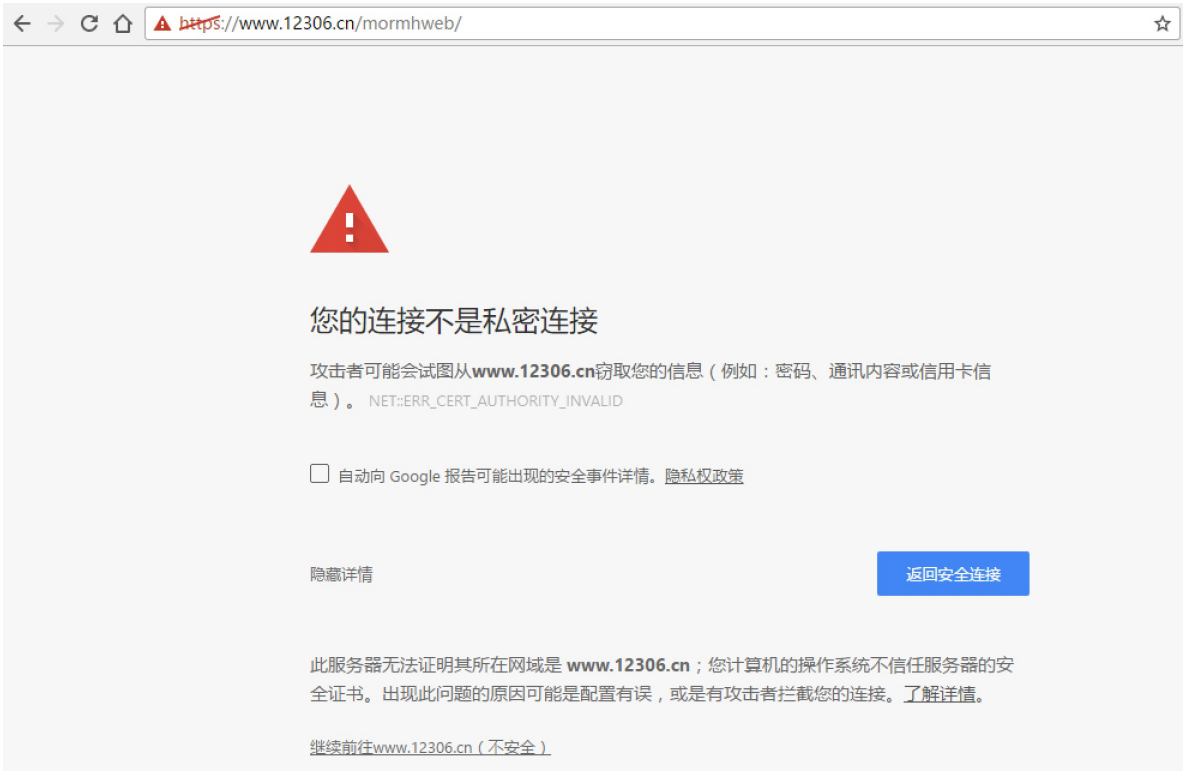
代理平台：https://www.hailiangip.com/personal/flow

教程：https://www.yuque.com/g/qiananshiguang-vakmf/hwh58w/aurg369iwxasd745/collaborator/join?token=mwLYLFAku4yCeeqS&source=doc_collaborator# 《代理ip生成API链接》

极客云国外代理平台：

https://jike138.com/user##https://cloud.tencent.com/developer/article/1937190

3.使用verify参数忽略CA证书



- 原因：该网站的CA证书没有经过【受信任的根证书颁发机构】的认证
  - **关于CA证书以及受信任的根证书颁发机构点击了解更多**，课上我们不做展开

运行代码查看代码中向不安全的链接发起请求的效果

网站：https://inv-veri.chinatax.gov.cn/

> 运行下面的代码将会抛出包含`ssl.CertificateError ...`字样的异常

```
  import requests
  url =
"https://sam.huat.edu.cn:8443/selfservice/"
  response = requests.get(url)
```

解决方案

> 为了在代码中能够正常的请求，我们使用`verify=False`参数，此时requests模块发送请求将不做CA证书的验证：verify参数能够忽略CA证书的认证

```
  import requests
  url =
"https://sam.huat.edu.cn:8443/selfservice/"
  response = requests.get(url,
verify=False)
```

4.禁止响应页面重定向

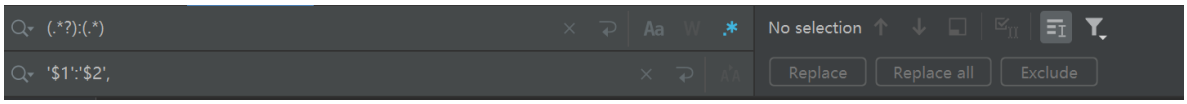
- 当用户访问某一个网站时：服务器将请求转发到其他页面的过程
- 会重复发请求（里面有些数据进行的重定向)

> 重定向状态码：2类
>
> 301: 永久重定向，浏览器会记住这个状态码，之后访问这个url都会跳转到新的url
>
> 302:临时重定向，不会记得改状态码

```
    allow_redirects=False       禁止重定向
    # 使用方法
    response = requests.get(url=start_url, headers=headers, allow_redirects=False)
```

5.requests模块发送post请求

> 思考：哪些地方我们会用到POST请求?
>
> 1. 登录注册（在web工程师看来POST 比 GET 更安全，url地址中不会暴露用户的账号密码等信息）
> 2. 需要传输大文本内容的时候（POST 请求对数据长度没有要求）
>
> 所以同样的，我们的爬虫也需要在这两个地方回去模拟浏览器发送post请求
>
> 利用正则快速转字典格式
>
> 1. 选中要转为字典的内容
> 2. 按ctrl+r快捷键
>
> . 任意匹配一个字符
>
> *可以让表达式出现0次或者任意次
>
> .*: 可以任意多个字符



requests发送post请求的方法

```
# data参数接收一个字典
response = requests.post(url, data)
```

- **requests模块发送post请求函数的其它参数和发送get请求的参数完全一致**

POST请求练习

下面面我们通过360翻译的例子看看post请求如何使用：

地址：https://fanyi.so.com/#

> 需求：抓取翻译的数据，做一个翻译小程序

请求参数调试工具：https://apifox.com/compare/postman-vs-apifox

使用教程：看课件资料

思路分析

发现响应数据是json格式，中文可能是unicode编码

1. 抓包确定请求的url地址

   https://fanyi.so.com/index/search?eng=1&validate=&ignore_trans=0&query=hello

2. 确定请求的参数

请求方式：post请求

参数单独携带：

```
eng: 1
validate:
ignore_trans: 0
query: hello  # 翻译关键参数，决定了可以翻译什么样的内容
```

3. 确定返回数据的位置

4. 模拟浏览器获取数据

抓包分析的结论

1. url地址：
2. 请求方法：POST
3. 请求所需参数：

```

```

4. **pc端User-Agent**:

`Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36`

代码实现

目标url：http://www.whggzy.com/site/category?parentId=66007&childrenCode=PoliciesAndRegulations&utm=site.site-PC-49434.959-pc-websitegroup-navBar-front.4.d4b3a230bc4d11eea13d9399f144d651

> 需求：获取前五页的标题跟时间
>
> 携带参数：字典，字符串

1. 确定url