

qiyesafe.com

七夜的故事

个人博客:<http://qiyesafe.com>
出版书籍:《Python爬虫开发与项目实战》 购买链接:<https://item.jd.com/12206762.html>
Python中文社区专栏作者
七夜安全博客_微信公众平台: [qiye_safe](#),及时更新技术内容

昵称: 七夜的故事
园龄: 2年6个月
粉丝: 339
关注: 85
+加关注

<2018年8月>

日	一	二	三	四	五	六
29	30	31	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

🔍 搜索

随笔分类

android
C(1)
C++(1)
Django项目实战(5)
FPGA
html(2)
java
linux(2)
PHP
python(15)
Python Web开发(3)
Python爬虫开发(16)
seo
Sublime插件开发教程(1)
windows(2)
汇编
漏洞挖掘(1)
逆向工程(3)
爬虫(2)
算法(1)
网络安全(13)
无线局域网安全教程(6)
心情(1)

博客园 首页 □□□ 联系 管理 □□ XML

随笔- 57 文章- 0 评论- 254

BaseProxy:异步http/https中间人

BaseProxy

异步http/https代理,可拦截并修改报文,可以作为中间人工具.仅支持**py3.5+**.项目地址:[BaseProxy](#)。

意义

BaseProxy项目的本意是为了使HTTP/HTTPS拦截更加纯粹,更加易操作,学习成本更低。

在Python领域,中间人工具非常强大和成功的是MitmProxy,但是有些地方不是很喜欢。

- Windows上安装比较费时费力
- 功能太多了,可惜我用不到这么多(似乎不是它的错,哈哈)
- 随着版本升级,采用插件化框架,需要定制功能,需要写个插件成为它的一部分(我只是想集成它而已)。

因此BaseProxy就诞生了,不仅支持HTTPS透明传输,还支持HTTP/HTTPS拦截,简单易用,可以很好地集成到你们的项目中。

安装

安装非常简单,本项目已经发布到PyPI中...

```
pip3 install baseproxy
```

使用配置

启动baseproxy

在test文件夹下, 有很多测试用例。以startserver.py为例。

```
from baseproxy.proxy import AsyncMitmProxy

baseproxy = AsyncMitmProxy(https=True)

baseproxy.serve_forever()
```

使用上述代码,就可以将HTTPServer运行起来了.对代码的解释如下:

- `https=True` 是对https进行解密; `https=False` 是对于https实行透传
- baseproxy默认运行在8788端口,如果想改变端口的话,修改为
`AsyncMitmProxy(server_addr=(' ',port),https=True)` .

运行结果如下:

```
[2018-06-22 18:46:32] INFO HTTPServer is running at address( , 8788 ).....
```

随笔档案

- 2018年8月 (2)
- 2018年7月 (1)
- 2018年6月 (1)
- 2018年5月 (9)
- 2017年10月 (1)
- 2017年9月 (1)
- 2017年8月 (2)
- 2017年6月 (5)
- 2017年5月 (3)
- 2017年3月 (1)
- 2017年1月 (1)
- 2016年12月 (1)
- 2016年11月 (2)
- 2016年10月 (2)
- 2016年9月 (2)
- 2016年8月 (3)
- 2016年7月 (2)
- 2016年6月 (3)
- 2016年5月 (3)
- 2016年4月 (3)
- 2016年3月 (8)
- 2016年2月 (1)

积分与排名

积分 - 92813
排名 - 4031

最新评论

- 1. Re:百度登录加密协议分析(上)
@ | Mars引用现在登录多了好多参数 traceid, dv等变参, 太坑了traceid这个是JS里面的, 很容易就可以获取, dv猜测是一开始产生的链接里面的tk、as、ds有关, 我也没找到破解的方法.....
--admin09
- 2. Re:百度登录加密协议分析(上)
@ | Mars带着cookie就没事...
--admin09
- 3. Re:纯手工打造简单分布式爬虫(Python)
控制节点在通知爬虫节点结束的时候, 当存在多个爬虫节点时, 一个url_q.put('end')只能结束第一个获取到end的爬虫节点; 当第一个爬虫节点get到end时, 对result_q.put('en.....
--小白还是菜鸟
- 4. Re:BaseProxy:异步http/https中间人
收藏先
--sparkdev
- 5. Re:纯手工打造简单分布式爬虫(Python)
@gaoyulin请问您这个问题解决了么, 我也卡在这里了, 好痛苦T.T.
..
--GarVicker

阅读排行榜

- 1. python开源IP代理池--IPProxys(

安装CA证书

1.将chrome浏览器代理服务器设置为127.0.0.1:8788,推荐使用SwitchyOmega插件.



2.设置好代理,并将baseproxy运行后,访问www.baidu.com.



3.这时候访问被拒绝,需要安装证书.在当前网页访问 baseproxy.ca,下载证书.

- 47833)
2. python爬虫:爬取慕课网视频(18024)
3. 无线安全专题_破解篇02--kali破解pin码(14242)
4. “永恒之蓝”漏洞的紧急应对--毕业生必看(13322)
5. 纯手工打造简单分布式爬虫(Python)(8960)

评论排行榜

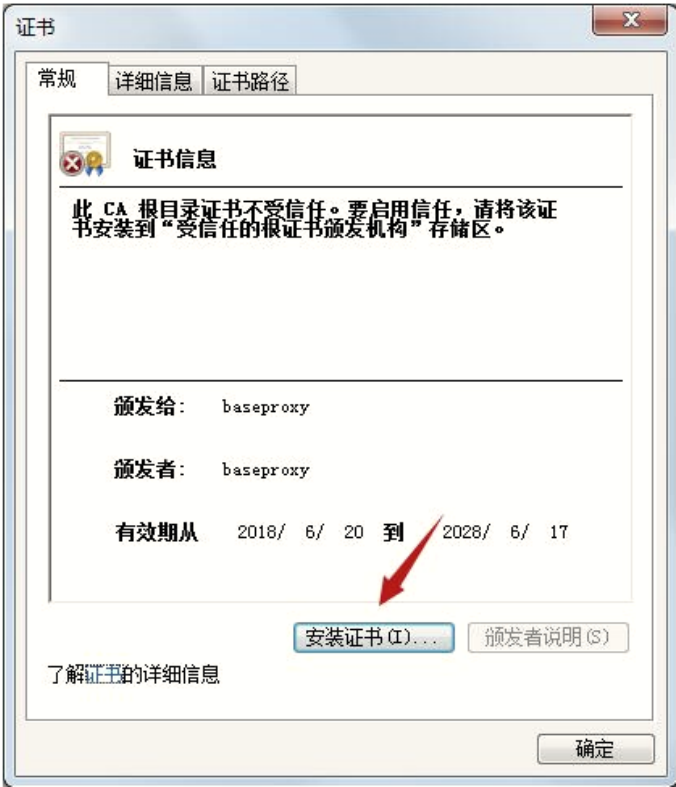
1. python开源IP代理池--IPProxys(59)
2. 纯手工打造简单分布式爬虫(Python)(32)
3. 百度登录加密协议分析(上)(17)
4. Scrapy爬取美女图片第三集 代理ip(下)(16)
5. Scrapy爬取美女图片第四集 突破反爬虫(上)(16)

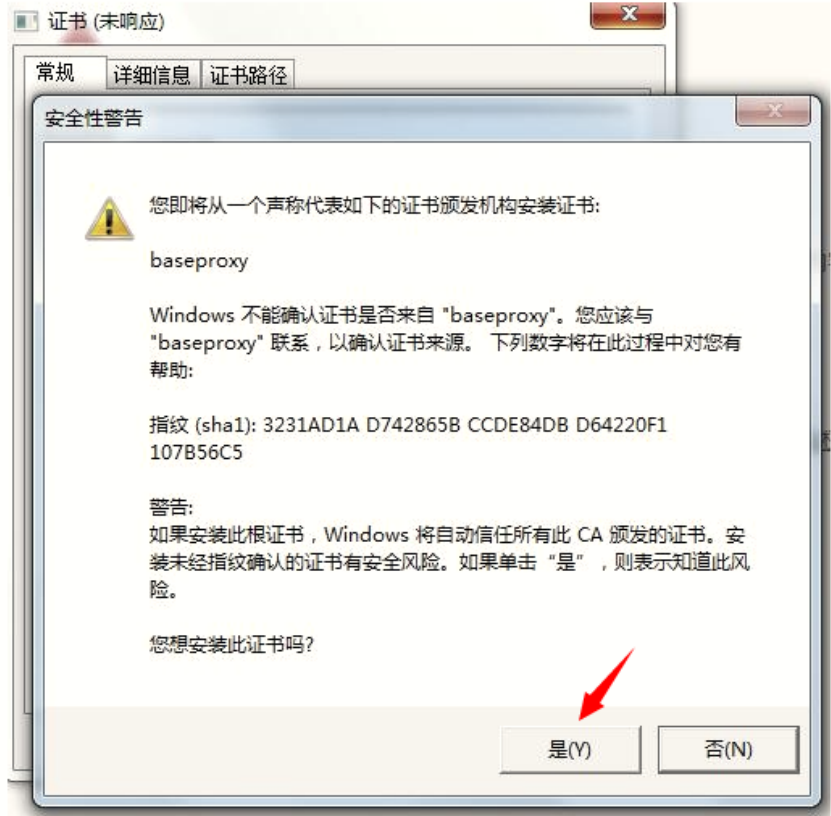
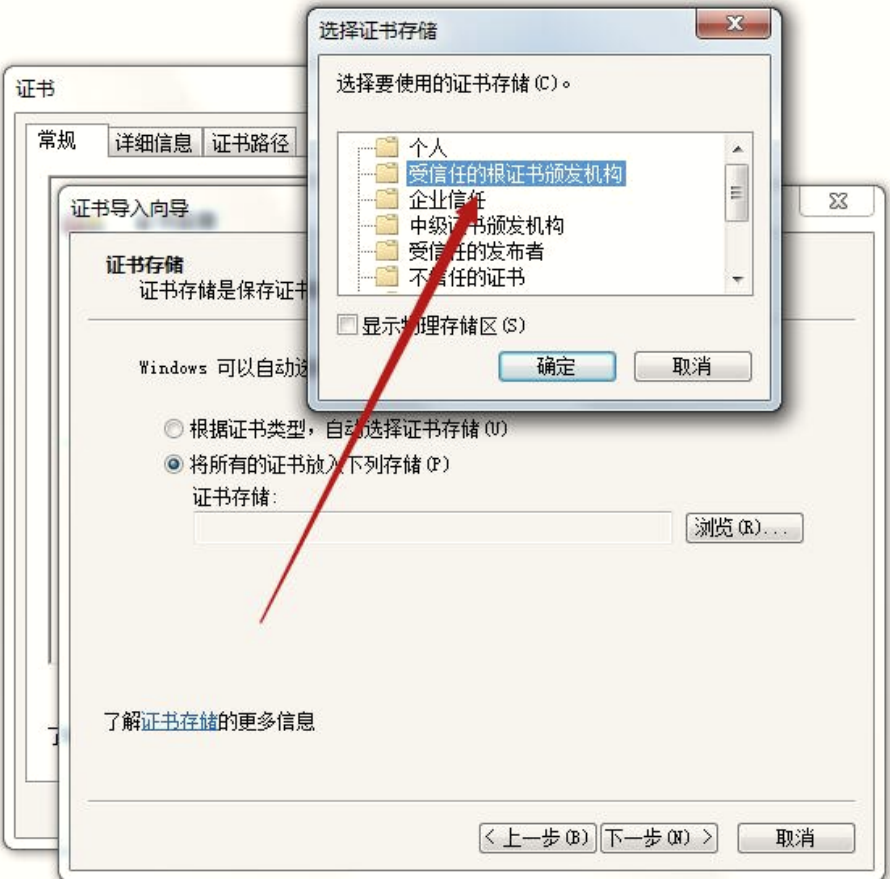
推荐排行榜

1. python开源IP代理池--IPProxys(22)
2. 练手之经典病毒熊猫烧香分析(上)(12)
3. “永恒之蓝”漏洞的紧急应对--毕业生必看(11)
4. Scrapy爬取美女图片第三集 代理ip(下)(10)
5. Scrapy爬取美女图片第三集 代理ip(上) (原创)(9)



4.双击下载的证书,并安装到合法机构中.





5.接着访问百度就可以了.



注意：只有 `https=True` 时,才需要安装CA证书。

开发

经过上一步的使用配置,baseproxy已经可以正常运行了,但是这样是远远不够的.baseproxy还提供了接口,方便开发者对http请求和响应进行修改.

接口

baseproxy提供了两个接口,一个是修改请求,一个是修改响应.

拦截请求

```
class ReqIntercept(InterceptPlug):  
  
    def deal_request(self,request):  
        pass
```

对于请求的拦截,需要继承ReqIntercept类,并重写其中的deal_request函数.在deal_request函数的最后,需要将修改后的request参数返回出去.

如果想抛弃这个请求,直接返回None.

request参数

deal_request函数中的request参数类型为Request类

成员变量

Name	□□	□□
hostname	str	□□
port	int	□□
command	str	□□□□
path	str	□□□□
request_version	str	HTTP□□□□

成员函数

```
def set_headers(self,headers)  
- headers:类型为dict  
- 用于设置头部
```

```
def get_header(self,key):  
- key:类型为str  
- 用于获取指定头部,返回str
```



```
def get_headers(self):  
    - 用于获取整个头部,返回为dict
```

```
def set_header(self,key,value):  
    - 头部 key,类型str  
    - 头部 value,类型str  
    - 用于设置头信息
```

```
def get_body_data(self):  
    - 获取请求体内容,返回类型为bytes
```

```
def set_body_data(self,body):  
    - 设置请求体内容,body类型为bytes
```

拦截响应

```
class RspIntercept(InterceptPlug):  
  
    def deal_response(self,response):  
        pass
```

对于响应的拦截,需要继承RspIntercept类,并重写其中的deal_response函数.在deal_response函数的最后,需要将修改后的response参数返回出去.
如果想抛弃这个响应,直接返回None.

response参数

deal_response函数中的response参数类型为Response类

成员变量

Name	□□	□□
hostname	str	□□
port	int	□□
status	int	□□□
reason	str	□□□□
response_version	str	HTTP□□□□
request	Request	□□□□□□□□

成员函数

```
def set_headers(self,headers)  
    - headers:类型为dict  
    - 用于设置头部
```

```
def get_header(self,key):  
    - key:类型为str  
    - 用于获取指定头部,返回str
```

```
def get_headers(self):  
    - 用于获取整个头部,返回为dict
```

```
def set_header(self,key,value):  
    - 头部 key,类型str  
    - 头部 value,类型str  
    - 用于设置头信息
```

```
def get_body_data(self):  
    - 获取响应体内容,返回类型为bytes
```

```
def set_body_data(self,body):  
    - 设置响应体内容,body类型为bytes
```

```
def get_body_str(self,decoding=None):  
    - decoding:编码,默认为None,内部采用chardet探测  
    - 返回响应体,类型为str.如果无法解码,返回None
```



```
def set_body_str(self,body_str,encoding=None):
- encoding:编码,默认为None,内部采用chardet探测
- 设置响应体,body_str类型为str
```

注册拦截插件

将拦截类完成后，需要注册到baseproxy中,需要调用AsyncMitmProxy的register函数.示例如下:

```
from baseproxy.proxy import ReqIntercept, RspIntercept, AsyncMitmProxy
__author__ = 'qiye'
__date__ = '2018/6/21 23:35'

class DebugInterceptor(ReqIntercept, RspIntercept):
    def deal_request(self, request):
        return request

    def deal_response(self, response):
        return response

if __name__=="__main__":

    baseproxy = AsyncMitmProxy(https=False)
    baseproxy.register(DebugInterceptor)
    baseproxy.serve_forever()
```

小例子

将淘宝中的所有产品图片换成我公众号的二维码.代码在test文件夹的replace_image.py中,内容如下:

```
from baseproxy.proxy import RspIntercept, AsyncMitmProxy

class ImageInterceptor( RspIntercept):

    def deal_response(self, response):
        if response.get_header("Content-Type") and 'image' in response.get_header("Content-Type"):
            with open("../img/qiye2.jpg",'rb') as f:
                response.set_body_data(f.read())
            return response

if __name__ == "__main__":
    baseproxy = AsyncMitmProxy(https=True)
    baseproxy.register(ImageInterceptor)
    baseproxy.serve_forever()
```

效果如下:

淘宝网
Taobao.com

宝贝 连接机干扰器 新款

搜索

品质生活家



亲宝贝 享受育儿新方式



淘宝教育 给你未来的学习体验



极有家 过我想要的生活



淘宝花鸟 园艺宠物爱好者集中营



聚名品 甄选大牌好货



特色玩味控



淘宝众筹 脑洞创意坊



中国质造 懂得好生活



阿里汽车 冰点低价



参考项目

MitmProxy
proxy2

福利大放送

关注公众号：七夜安全博客

- 回复【1】：领取 Python数据分析 教程大礼包
- 回复【2】：领取 Python Flask 全套教程
- 回复【3】：领取 某学院 机器学习 教程
- 回复【4】：领取 爬虫 教程

知识星球已经50人了，随着人数的增多，价格之后会上涨，越早关注越多优惠。星球的福利有很多：

- 比如上面的教程，已经提前在知识星球中分享
- 可以发表一些问题，大家一块解决

http://www.cnblogs.com/qiyeboy/p/9216433.html[2018/8/31 星期五 22:13:04]

- 我之后写的电子书，录制的教学视频，对于知识星球的朋友都是优惠的（基本上免费）
- 一些节假日会给大家发个红包或者赠书



本文章属于原创作品,欢迎大家转载分享，禁止修改文章的内容。尊重原创,转载请注明来自:七夜的故事 <http://www.cnblogs.com/qiyeboy/>

分类: [python](#),[网络安全](#)

标签: [网络安全](#), [python](#), [中间人](#), [https代理](#)

好文要顶

关注我

收藏该文

七夜的故事

关注 - 85

粉丝 - 339

加关注

1

0

推荐

反对

« 上一篇: [百度云满速下载原理与方法](#)
 » 下一篇: [基于HTTPS的中间人攻击-BaseProxy](#)

posted @ 2018-06-23 10:14 [七夜的故事](#) 阅读(310) 评论(1) [编辑](#) [收藏](#)

评论

#1楼 2018-06-23 11:03 | [sparkdev](#)



收藏先

支持(0) 反对(0)

[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】超50万VC++源码: 大型组态工控、电力仿真CAD与GIS源码库！
- 【推荐】企业SaaS应用开发实战，快速构建企业运营/运维系统
- 【推荐】ActiveReports 报表控件，全面满足 .NET开发需求

腾讯云

学生服务器体验套餐

10元/月

· 1核2G · 1M带宽 · 50GB存储

立即抢购

最新IT新闻:

- 顺风车事件后 我们探访了滴滴司机群体
 - 乐融致新完成工商变更：腾讯京东入股 注册资本提升
 - 51信用卡营收同比增50.5% CFO赵轲：收入更多元化
 - 第三大股东对东芝表现不满 提议改组董事会
 - 今年蒸发个平安，今天蒸发个携程，马化腾被拉下首富神坛
- » 更多新闻...



最新知识库文章:

- 如何招到一个靠谱的程序员
 - 一个故事看懂“区块链”
 - 被踢出去的用户
 - 成为一个有目标的学习者
 - 历史转折中的“杭派工程师”
- » 更多知识库文章...

历史上的今天:

2017-06-23 CoolBlog开发笔记第1课:项目分析

