

Web

Week1 Web1

题目名字：谁吃了我的 flag

Description：

呜呜呜，Mki 一起床发现写好的题目变成这样了，是因为昨天没有好好关机吗 T_T hint:
据当事人回忆，那个夜晚他正在用 vim 编写题目页面，似乎没有保存就关机睡觉去了，
现在就是后悔，十分的后悔。

URL：

<http://118.25.111.31:10086/index.html>

来自 <<https://hgame.vidar.club/#/challenge/list>>

这个题我一开始一筹莫展。由于是第一次体验 ctf，看到网页打开就有 flag 非常开心，
提交，果然是错的（怎么可能因为是第一次就白送 50 分啊！）

补上原题目的缺失的右括号？无理！

抓包、查看页面源代码，一点用都没有。。。出题人一开始没给提示，我以为提示是“昨天晚上”，所以还去改 If-Modified-Since，当然是没用的。

后来根据“昨晚没有关机”，以及 3eek_diSclosure (Seek disclosure 注意泄露) 这个提示，查了一下相关资料，原来存在**隐藏文件**这种操作吗？

查了一下资料，是这样：

网站备份压缩文件

在网站的使用过程中，往往需要对网站中的文件进行修改、升级。此时就需要对网站整站或者其中某一页面进行备份。当备份文件或者修改过程中的缓存文件因为各种原因而被留在网站web目录下，而该目录又没有设置访问权限时，便有可能导致备份文件或者编辑器的缓存文件被下载，导致敏感信息泄露，给服务器的安全埋下隐患。

漏洞成因及危害：

该漏洞的成因主要有以下两种：

1. 服务器管理员错误地将网站或者网页的备份文件放置到服务器web目录下。
2. 编辑器在使用过程中自动保存的备份文件或者临时文件因为各种原因没有被删除而保存在web目录下。

漏洞检测：

该漏洞往往会导致服务器整站源代码或者部分页面的源代码被下载，利用。源代码中所包含的各类敏感信息，如服务器数据库连接信息，服务器配置信息等会因此而泄露，造成巨大的损失。被泄露的源代码还可能会被用于代码审计，进一步利用而对整个系统的安全埋下隐患。

```
.rar .zip .7z .tar.gz .bak .swp .txt .html
```

于是乎，我把原址的 index.html 的后缀改成了：.swp

。。。 404 Not Found

好，我再试试.txt, .bak, .doc, .php, .zip, .tar, .tar.gz, .7z! —————

结果全部 404 Not Found：



WTF? ! 怎么会这样?

于是暂时放下这道题去做别的了。

过了几天，vim 的提示出来了，结合关机提示，后缀名就基本锁定是 swp 了，但是，

你妈的，为什么？会 404?

最后不得已问了某教授：原来是 `"/.index.html.swp"`，不是 `"/index.html.swp"`，slash

后面有一个点!!!

卒。



哦哦，下载下来了。

linux 虚拟机启动！

打开文件！



看到被吃掉的 flag 是 hgame{3eek_diSc10Sure_from+wEbsit@}

Week1 Web2

题目名字：换头大作战

<http://120.78.184.111:8080/week1/how/index.php>

来自 <<https://hgame.vidar.club/#/challenge/list>>

打开页面，写打开页面检查器看看有没有什么猫腻，貌似没有。。。

想要flag嘛：

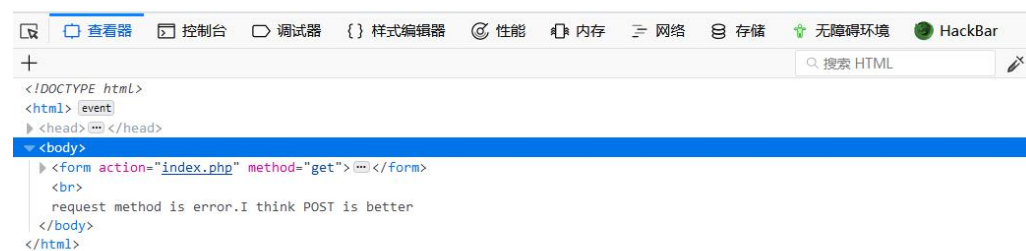


```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>换头大作战</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" type="text/css" media="screen" href="main.css">
    <script src="main.js"></script>
  </head>
  <body>
    <form action="index.php" method="get"></form>
    <br>
  </body>
</html>
```

hmm，点一下 submit 试试看。

想要flag嘛：

request method is error.I think POST is better



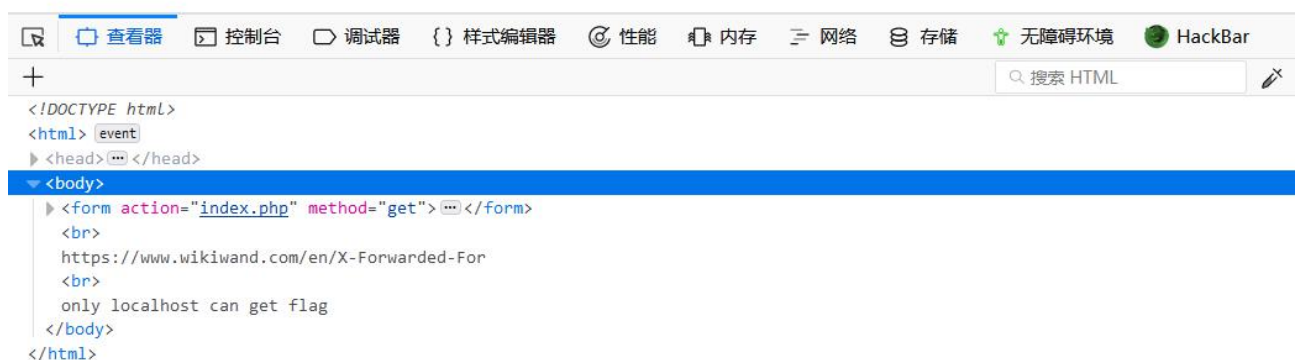
```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    <form action="index.php" method="get"></form>
    <br>
    request method is error.I think POST is better
  </body>
</html>
```

啊，换成 post 方法是吧，那就把 method 里的 “get” 改成 “post” 呗。改完以后再

按一下 submit:

想要flag嘛:

https://www.wikiwand.com/en/X-Forwarded-For
only localhost can get flag



```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    <form action="index.php" method="get">
    <br>
    https://www.wikiwand.com/en/X-Forwarded-For
    <br>
    only localhost can get flag
  </body>
</html>
```


哦？还有学习资料？根据我从《图解 http》和题目名字的提示来看，是要在请求头里加

一个 X-Forwarded-For:127.0.0.1 啊, burpsuite 抓一下包, 把 XFF 加进去, 再 forward

一下:

想要flag嘛:

https://www.wikiwand.com/en/X-Forwarded-For
only localhost can get flag



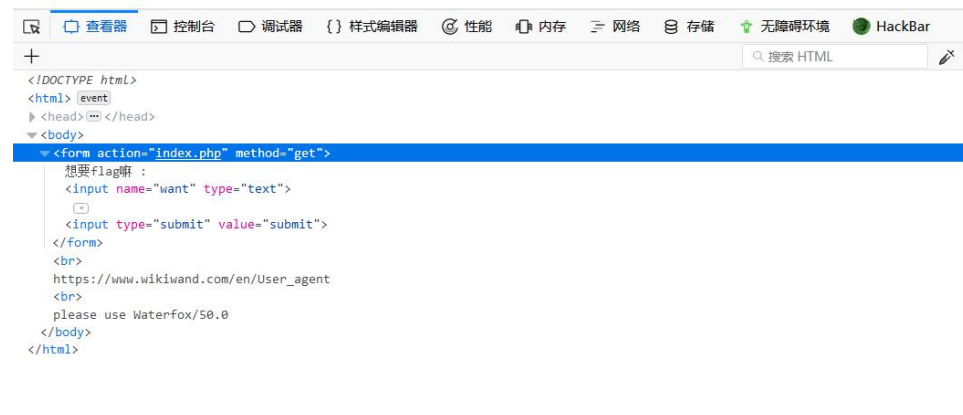
```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    <form action="index.php" method="get">
    <br>
    https://www.wikiwand.com/en/X-Forwarded-For
    <br>
    only localhost can get flag
  </body>
</html>
```

??? 怎么还是这个界面? 我当时卡了半天。。。经过不断的努力尝试, 我发现光加

XFF 还不够, 还得把 method 改成 post。再试一次:

想要flag嘛:

https://www.wikiwand.com/en/User_agent
please use Waterfox/50.0



```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
  </head>
  <body>
    <form action="index.php" method="get">
      想要flag嘛:
      <input name="want" type="text">
      <input type="submit" value="submit">
    </form>
    <br>
    https://www.wikiwand.com/en/User_agent
    <br>
    please use Waterfox/50.0
  </body>
</html>
```

接下来是改 User-Agent 吗。。。我还专门去百度了, 原来还真有 waterfox 浏览器啊。。。

那就接着改咯, XFF 和 post 别忘了:

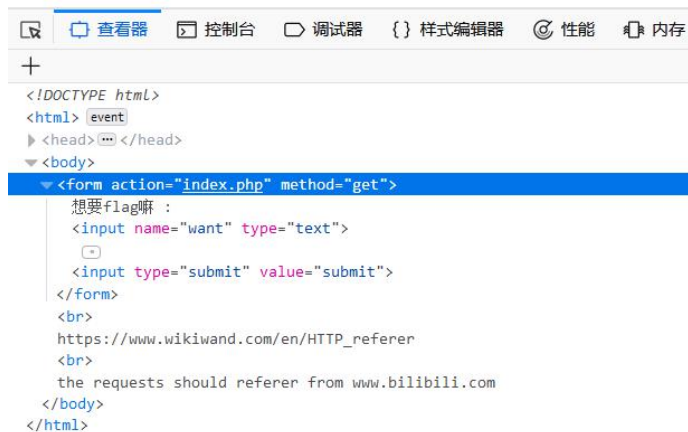
```
POST /week1/how/index.php HTTP/1.1
Host: 120.78.184.111:8080
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:64.0) Gecko/20100101 Waterfox/50.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Referer: http://120.78.184.111:8080/week1/how/index.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 5
Connection: close
Cookie: admin=0; admin=0
Upgrade-Insecure-Requests: 1
X-Forwarded-For: 127.0.0.1

want=
```

forward 一下:

想要flag嘛 :

https://www.wikiwand.com/en/HTTP_referer
the requests should referer from www.bilibili.com

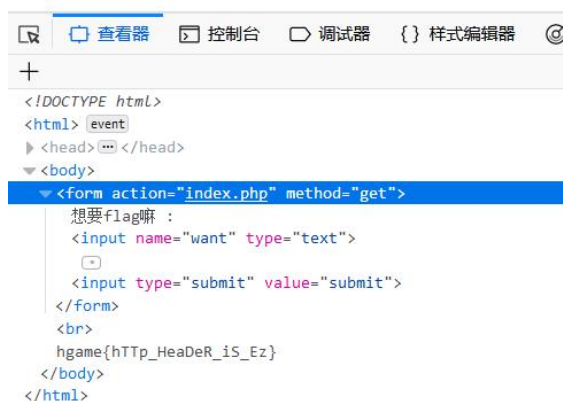


```
<!DOCTYPE html>
<html> event
  <head>...</head>
  <body>
    <form action="index.php" method="get">
      想要flag嘛 :
      <input name="want" type="text">
      <input type="submit" value="submit">
    </form>
    <br>
    https://www.wikiwand.com/en/HTTP_referer
    <br>
    the requests should referer from www.bilibili.com
  </body>
</html>
```

继续。。。referer 是哔哩哔哩海星：

想要flag嘛 :

hgame{hTTP_HeaDeR_iS_Ez}



```
<!DOCTYPE html>
<html> event
  <head>...</head>
  <body>
    <form action="index.php" method="get">
      想要flag嘛 :
      <input name="want" type="text">
      <input type="submit" value="submit">
    </form>
    <br>
    hgame{hTTP_HeaDeR_iS_Ez}
  </body>
</html>
```

flag 就出来了

Week1 Web3

题目名字: very easy web

Description

代码审计初♂体验

URL

http://120.78.184.111:8080/week1/very_ez/index.php

来自 <<https://hgame.vidar.club/#/challenge/list>>

代码审计题啊，没学过 php，所以全靠百度，bing 和 google 做的：

```
<?php
error_reporting(0);
include("flag.php");

if(strpos("vidar", $_GET['id'])!==FALSE)
    die("<p>干巴爹</p>");

$_GET['id'] = urldecode($_GET['id']);
if($_GET['id'] === "vidar")
{
    echo $flag;
}
highlight_file(__FILE__);
?>
```

这题的关键是 urldecode

第一个条件判断 id 参数里有没有 vidar，如果参数是 vidar，直接返回 “干巴爹”

第二个条件是判断 id 参数 urldecode 是否等于 “vidar”，如果等于则显示 flag；

按照网上别人所说，我先把 v 用 urlencode 得到%76，然后再对%76 进行 urlencode，

得到%2576，这样就能绕过判断：

① 120.78.184.111:8080/week1/very_ez/index.php?id=%2576idar|

在 url 栏输入结果得到 flag:

```
hgame{urlDecode_Is_GoOd} <?php
error_reporting(0);
include("flag.php");

if(strpos("vidar", $_GET['id'])!==FALSE)
    die("<p>干巴爹</p>");

$_GET['id'] = urldecode($_GET['id']);
if($_GET['id'] === "vidar")
{
    echo $flag;
}
highlight_file(__FILE__);
?>
```

Week1 Web4

题目名字: can u find me?

Description

为什么不问问神奇的十二姑娘和她的小伙伴呢

URL

<http://47.107.252.171:8080/>

来自 <<https://hgame.vidar.club/#/challenge/list>>

the gate has been hidden

can you find it? xixixi



```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    <p>the gate has been hidden</p>
    <p>can you find it? xixixi</p>
    <a href="f12.php"></a>
  </body>
</html>
```

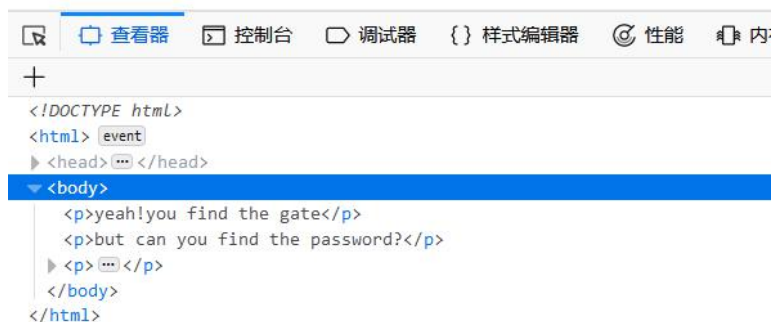
gate 被藏起来了, 但是打开页面审查看到有 f12.php 跳转, 所以直接在 url 的斜杠后

面加 f12.php, 回车

yeah!you find the gate

but can you find the password?

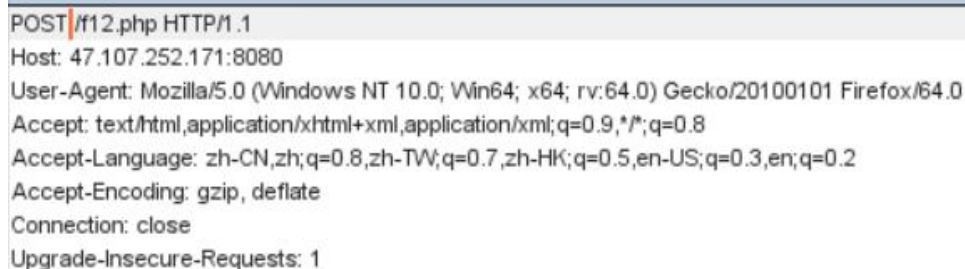
please post password to me! I will open the gate for you!



```
<!DOCTYPE html>
<html> event
  <head> ... </head>
  <body>
    <p>yeah!you find the gate</p>
    <p>but can you find the password?</p>
    <p>please post password to me! I will open the gate for you!</p>
  </body>
</html>
```

好嘞，接下来找密码！审查元素貌似没有有用的提示了，burpsuite 抓个包试试？题目

还暗示了：POST 一下密码，说明 get 不行：



```
POST /f12.php HTTP/1.1
Host: 47.107.252.171:8080
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:64.0) Gecko/20100101 Firefox/64.0
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
Connection: close
Upgrade-Insecure-Requests: 1
```

forward 一下，看一下：

yeah!you find the gate

but can you find the password?

please post password to me! I will open the gate for you!

no no no,your password isn't right



```
<!DOCTYPE html>
<html> <event></html>
<head></head>
<body>
  <p>yeah!you find the gate</p>
  <p>but can you find the password?</p>
  <p>please post password to me! I will open the gate for you!</p>
  <p>no no no,your password isn't right</p>
</body>
</html>
```

页面上没有什么，但是 burpsuite 的 response 却暗藏玄机：



```
HTTP/1.1 200 OK
Server: nginx/1.15.8
Date: Sun, 27 Jan 2019 16:23:14 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
X-Powered-By: PHP/7.2.14
password: woyaoflag
Content-Length: 283

<!DOCTYPE html>
<html>
<head>
  <title>can u find me?</title>
</head>
<body>
  <p>yeah!you find the gate</p>
  <p>but can you find the password?</p>
  <p>please post password to me! I will open the gate for you!</p>
  <p>no no no,your password isn't right</p></body>
</html>
```

password 写在响应头里还行？

那就再 post 一遍，请求内容里加上 password=woyaoflag

```
POST /f12.php HTTP/1.1
Host: 47.107.252.171:8080
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:64.0) Gecko/20100101 Firefox/64.0
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://47.107.252.171:8080/f12.php?password=woyaoflag
Content-Type: application/x-www-form-urlencoded
Content-Length: 18
Connection: close
Upgrade-Insecure-Requests: 1
```

password=woyaoflag

forward:

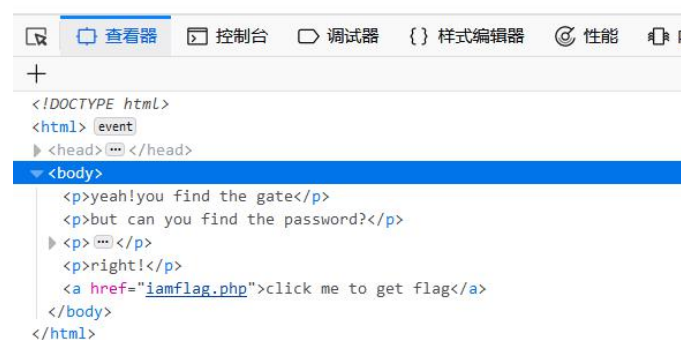
yeah!you find the gate

but can you find the password?

please post password to me! I will open the gate for you!

right!

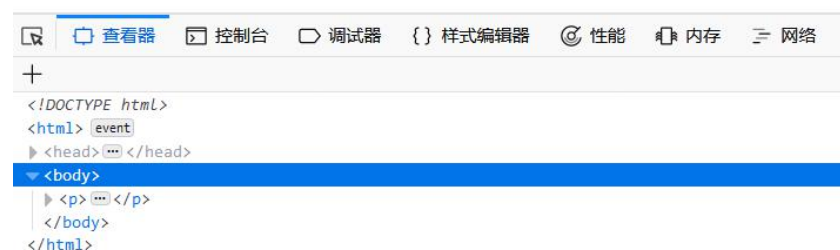
[click me to get flag](#)



```
<!DOCTYPE html>
<html> event
<head>...</head>
<body>
  <p>yeah!you find the gate</p>
  <p>but can you find the password?</p>
  <p>...</p>
  <p>right!</p>
  <a href="iamflag.php">click me to get flag</a>
</body>
</html>
```

哇，胜利就在眼前!点击 click me to get flag 链接，结果跳转到 toofast.php:

aoh,your speed is sososo fast,the flag must have been left in somewhere



```
<!DOCTYPE html>
<html> event
<head>...</head>
<body>
  <p>...</p>
</body>
</html>
```

搞我? Too fast 可还行。提示了 left somewhere。

还能 left where? 回到前面那个界面, 在跳转到 iamflag.php 的时候抓包截胡:

```
HTTP/1.1 302 Found
Server: nginx/1.15.8
Date: Sun, 27 Jan 2019 16:34:13 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
X-Powered-By: PHP/7.2.14
location: toofast.php
Content-Length: 132

<html>
  <head>
    <title>can you find me?</title>
  </head>
  <body>
    <p>flag:hgame{f12_1s_aMazing111}</p>
  </body>
</html>
```

302Found 和 location 都在疯狂暗示强制跳转, 怪不得。再一看 response 的响应内容,

flag 果然在这里!

RE 部分题

汇编知识=0, 只会最后一题的 python。。。.

Week1 Re5:

Description

Easiest Python Challenge!

URL

<http://plqbnxx54.bkt.clouddn.com/first.py>

来自 <<https://hgame.vidar.club/#/challenge/list>>

```
import base64
import hashlib

enc1 = 'hgame{'
enc2 = 'SGVyZV8xc18zYXN5Xw=='
enc3 = 'Pyth0n}'

print 'Welcome to Processor\'s Python Classroom!\n'
print 'Here is Problem One.'
print 'There\'re three parts of the flag.'

print '-----'

print 'Plz input the first part:'
first = raw_input()
if first == enc1:
    pass
else:
    print 'Sorry , You\'re so vegatable!'
    exit()

print 'Plz input the secend part:'
secend = raw_input()
secend = base64.b64encode(secend)
if secend == enc2:
    pass
else:
    print 'Sorry , You\'re so vegatable!'
    exit()

print 'Plz input the third part:'
third = raw_input()
third = base64.b32decode(third)
if third == enc3:
    pass
else:
    print 'Sorry , You\'re so vegatable!'
    exit()

print 'Oh, You got it !'
```

这题代码逻辑并不难, first=enc1=hgame{不用动,

enc2 经过 base64 解密后得到 secend, third...我想都没想直接觉得肯定是 enc3 自身了。。。

拼接 first, secend 和 third 就得到 flag 了。。。

MISC 部分题

Week1 MISC1

题目名字: Hidden Image

Here are some magic codes which can hide information in an ordinary picture, can you extract the hidden image in the provided picture?

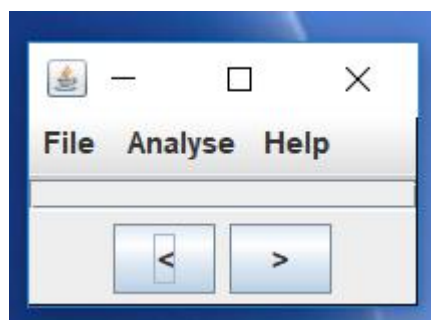
来自 <<https://hgame.vidar.club/#/challenge/list>>

题目给了这个图片:



Hidden image, 也就是隐写咯?

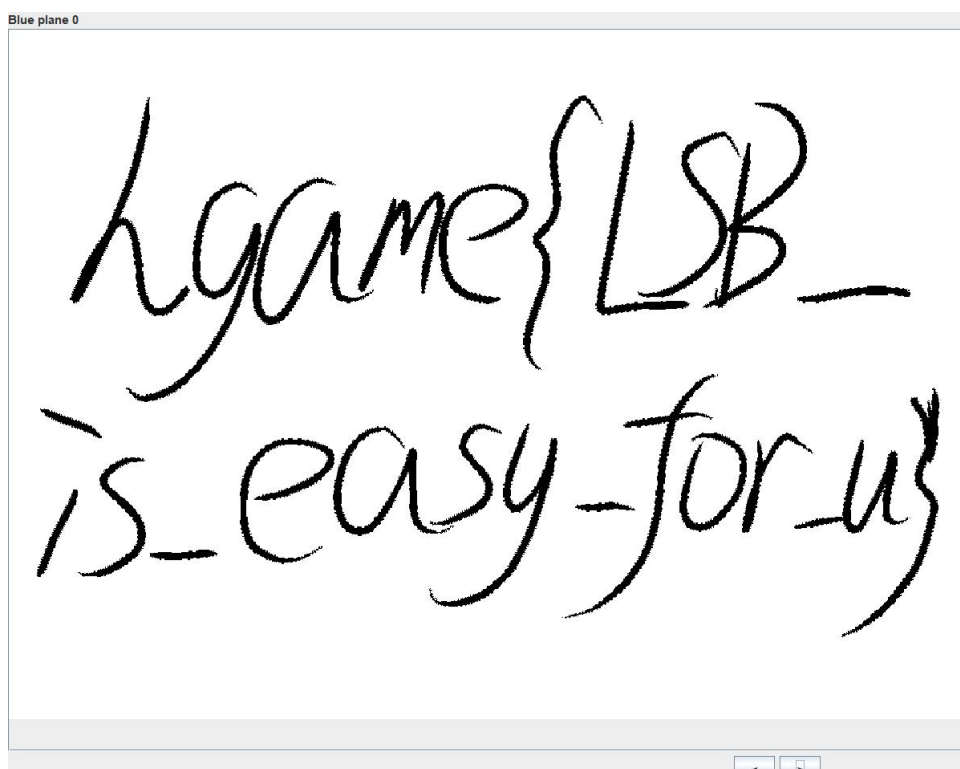
学长推荐了一个神器 stegsolve, 到 github 上下载 (就是个 jar)



FileOpen 一下把图片打开:



按一下图片下方的箭头切换模式：



在 blueplane 模式下看到了隐写的 flag

Week1 MISC2

题目名字: Broken Chest

Description

这个箱子坏掉了! 快用你无敌的[疯狂钻石]想想办法啊!

学习资料 <https://ctf-wiki.github.io/ctf-wiki/misc/archive/zip/>

URL

<http://plqfgjy5a.bkt.clouddn.com/Broken-Chest.zip>

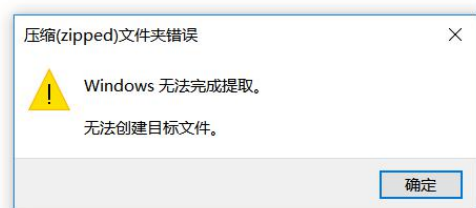
来自 <<https://hgame.vidar.club/#/challenge/list>>

下载下来是一个 zip 文件



打开看看, 发现里面有一个 flag 文件, 单击打开试试:

名称	类型	压缩大小	密码保护	大小	比率
flag	文本文档	1 KB	是	1 KB	0%



纳尼? zip 解压貌似也不行。。。看一眼学习资料, hmm。。。。

查看一下题目里 flag.txt 的 CRC32 的值:



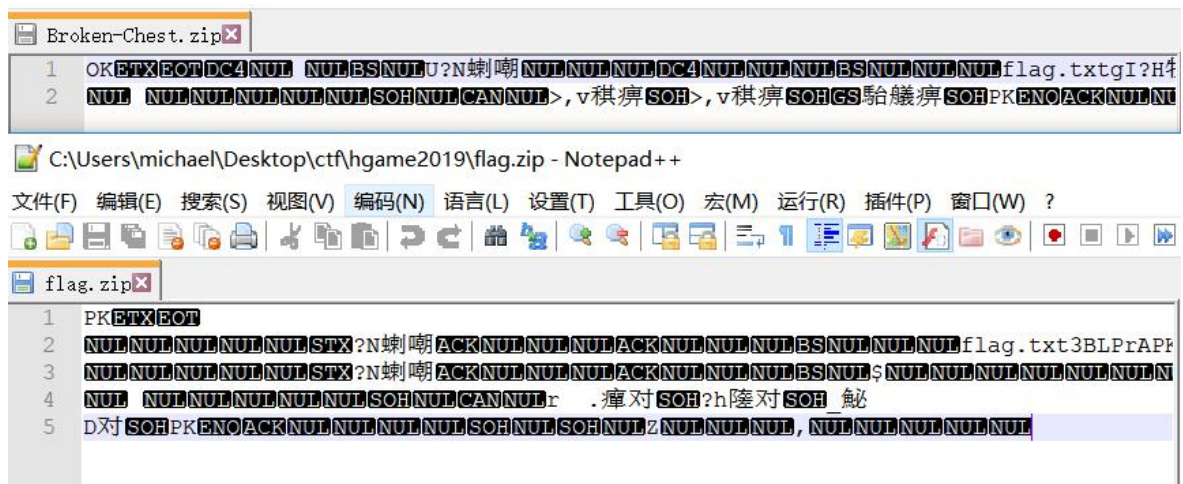
然后我自己新建了一个 flag.txt, flag 的内容用 CRC32.py 解出来:

```
$ python crc32.py reverse 0xB0B37CCE
4 bytes: {0x3f, 0x07, 0x12, 0x6c}
verification checksum: 0xb0b37cce (OK)
alternative: 13dsFT (OK)
alternative: 3BLPrA (OK)
alternative: 57yrG7 (OK)
alternative: B3QYTP (OK)
alternative: Bb388L (OK)
alternative: F7LXU3 (OK)
alternative: QsSYgG (OK)
alternative: VjTgMl (OK)
alternative: XeKWqb (OK)
alternative: _laT6A (OK)
alternative: d9k3_s (OK)
alternative: k652xd (OK)
alternative: nBUsgN (OK)
alternative: wyi3ld (OK)
alternative: xjxnJg (OK)
alternative: xv72Ks (OK)
alternative: yKg082 (OK)
```

随便选一个当内容, 然后用 7z 压缩成.zip;

接下来对比原题的 Broken_Chest.zip 和 flag.zip 的头:

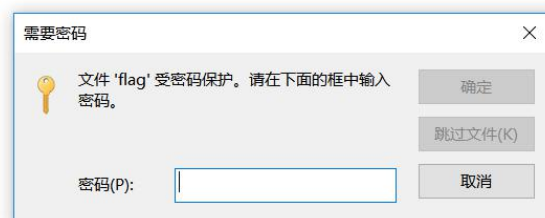
我用 notepad++ 打开两个 zip:



一个是 OK。。。一个是 PK。那就把 Broken-Chest 的头也改成 PK 好了（先把属性里只读选项的勾去掉）

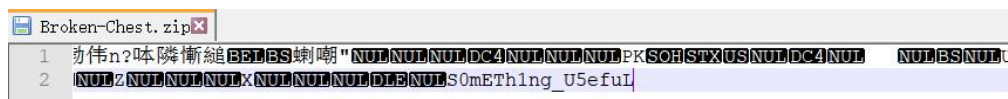
改好了以后再打开:

名称	类型	压缩大小	密码保护	大小	比率
 flag	文本文档	1 KB	是	1 KB	0%



要密码？

等一下，等一下，回到用 notepad++ 打开的 Broken Chest.zip!结果看到了这个：




我一开始以为这是 flag，试着提交发现不对。Something Useful。。。估计是密码？

试试先。

复制到密码栏，还真是啊！

flag.txt:

 flag - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

hgame{Cra2y_D1aM0nd}

我知道了，出题人是 JOJO 厨哒！（Description 竟然暗示了 flag!这就是你的出题计划吗？）

Week1 MISC2

题目名字：打字机

Description：Aris(划掉)牌打字机，时尚时尚最时尚~

URL

<http://plps4kyke.bkt.clouddn.com/打字机.zip>

来自 <<https://hgame.vidar.club/#/challenge/list>>

zip 打开是两张照片，stegsolve 看一下有没有隐写，结果都没有：



My_violet_typerwriter

这个花括号里的一定就是 flag 了

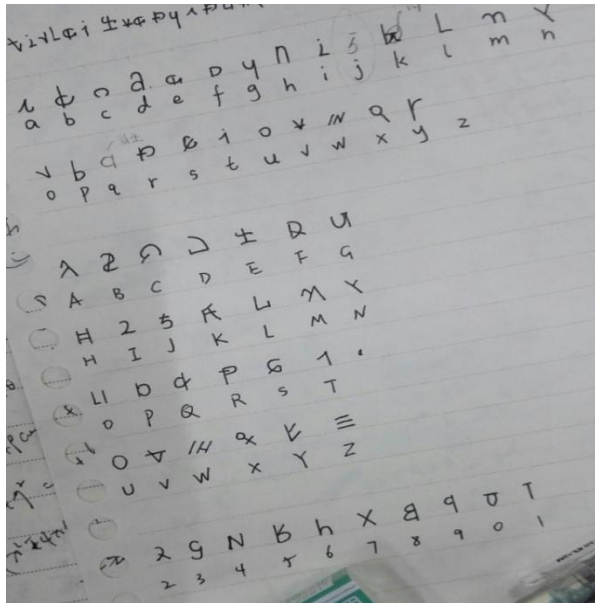
这题不怎么需要脑筋转弯，对照着英文键盘半猜半试就能把结果试出来（我试了 40 分钟左右）：

其中字母 e,r,v,m,p,w,o 还有 l，很快就可以从 “hgame” 和图片对应写出来。

最终 flag 是 hgame{My_violet_typerwriter}

???这个 flag 有什么深意吗？百度一下，violet typewriter，原来是《紫罗兰永恒花园》

里的打印机和文字么？！哔哩哔哩上甚至还有网友自己做的字体集和字母对照表：



说不出话？我还试了 40 分钟？？？（侧面说明了解出题人的偏好有多么重要）

Crypto 部分题

Week1 Crypto1:

题目名字: Mix

Description

--.../....-/...-/--.../...-/..../.-./-....-/..-/..../.-./--.../---./....-/....-/--.../..---/-....-/.../-
-.../-.../-..../.---/..../.---/-..../-..../...--/....-/...--/----/-..../..../---.../-.. So easy

来自 <<https://hgame.vidar.club/#/challenge/list>>

这道题真的是各种密码的 mix...

题目明显是摩斯密码，/应该是表示间隔用在线摩斯密码解密结果得到

众果搜首页>>摩尔斯电码转换

英文字母:

744B735F6D6F7944716B7B6251663430657D

转换为摩斯电码

清除

生成摩斯代码的分隔方式: ☐ 空格分隔 ☒ 单斜杠/分隔

摩斯电码: (格式要求: 可用空格或单斜杠/来分隔摩斯电码, 但只可用一种, 不可混用)

-. . . / . . . - / . . . - / - . . / - - . . / . . - - / / . . - / - . . . / - . . / -
/ . . - / - . . / - - - . / . . . - / . . . - / - - . . / . - - - / - . . . / - . . / - . . .
/ - . . / - . . . / . - - / / - - - / - . . . / - . . . / . . - - / . . . - / . . - -
/ - - - - / - . . . / / - . . . / - . .

转换为英文字母

是 36 位的字符串 744B735F6D6F7944716B7B6251663430657D，一开始没数位数，直接 MD5 和 Base64 去了，才发现位数不对。如果是 MD5 加密，多出一位还能理解（排除 F 以后的字母即可），但是多出 4 位？而且都满足 MD5 加密字符串在 0-9，A-F 之间的定例啊？？然后我想了想，A-F？16 进制？？等一下，我两两分组转成 ASCII

码试试？然后查表：74=t，4B=K，73=s....以此类推，最后得到这样一个字符串：

tKs_moyDqk{bQf40e}

我一开始惊了，有_和{}！这肯定跟 flag 有关了！

但是这个字符串里没有“hgame”的 h，g 和 a 啊？肯定是凯撒移位过了，于是用了个

在线凯撒移位，把所有的移位可能打印出来：

解密

tKs_moyDqk{bQf40e}

解密 使用英文字典智能分析

第1次解密:tks_moydqk{bqf40e}
第2次解密:sjr_lnxcpj{ape40d}
第3次解密:riq_kmwboi{zod40c}
第4次解密:qhp_jlvanh{ync40b}
第5次解密:pgo_ikuzmg{xmb40a}
第6次解密:ofn_hjtylf{wla40z}
第7次解密:nem_gisxke{vkz40y}
第8次解密:mdl_fhrwjd{ujy40x}
第9次解密:lck_egqvic{tix40w}
第10次解密:kbj_dfpuhb{shw40v}
第11次解密:jai_ceotga{rgv40u}
第12次解密:izh_bdnsfz{qfu40t}
第13次解密:hyg_acmrey{pet40s}
第14次解密:gxf_zblqdx{ods40r}
第15次解密:fwe_yakpcw{ncr40q}
第16次解密:evd_xzjobv{mbq40p}
第17次解密:duc_wyinau{lap40o}
第18次解密:ctb_vxhmzt{kzo40n}
第19次解密:bsa_uwglys{jyn40m}
第20次解密:arz_tvfkxr{ixm40l}
第21次解密:zqy_suejwq{hw140k}
第22次解密:ypx_rtdivp{gvk40j}
第23次解密:xow_qschuo{fuj40i}
第24次解密:wnv_prbgtn{eti40h}
第25次解密:vmu_oqafsm{dsh40g}
第26次解密:ult_npzerl{crg40f}

简介：通过把字母移动一定的位数来实现加密和解密。

找一找哪一个有 hgame 的各个字母呢？啊，第十三个就是：hyg_acmrey{pet40s}

flag 的提交格式是 hgame{}，但是现在{}外面是 hyg_acmrey。这个时候发现 h 和 g，

g 和 a 刚好都隔了一个字符。。。

那还用说，栅栏密码咯？把它再两两分组，横着写下：

h y

g _

a c

m r

e y

{ p

e t

4 0

s }

竖着读，flag 就出来了：hgame{e4sy_crypt0}

提交，嗯？怎么不对？？？

哦，原来是凯撒移位之前的 ASCII 码是大写的，凯撒移动完以后全变小写了，所以要把对应的大写改成小写提交就行了。