

Hgame 2024 week 1 wp

SiIence

Misc

1. SignIn

修改图片尺寸，把宽拉大。

2. 来自星尘的问候

题目描述里提到了六位弱加密，应该就是用的 steghide 用六位密码搞的隐写，所以放进去分离一下，爆破密码之前试了 123456 结果直接就成功了。

```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.22621.3007]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\wg-Cu>cd C:\Users\wg-Cu\Downloads\steghide-0.5.1-win32\steghide

C:\Users\wg-Cu\Downloads\steghide-0.5.1-win32\steghide>steghide extract -sf secret.jpg -p 123456
wrote extracted data to "secret.zip".

C:\Users\wg-Cu\Downloads\steghide-0.5.1-win32\steghide>
```

X=V=!{A!N=L=! !}

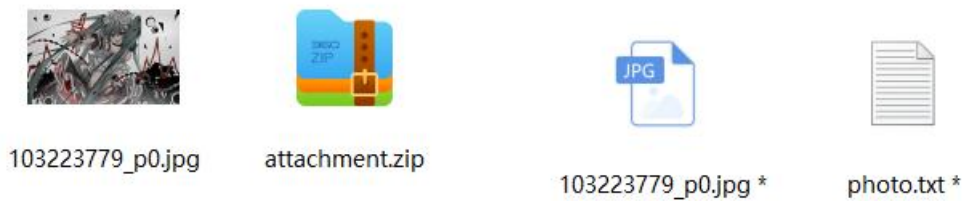
再根据题目里说的官网上找相关文字，就可以去一个个对应翻译。但更方便的是直接搜网上对这个文字的分析。然后又发现网上有人指出可以去官网扒 woff2 文件，跟大小写字母和数字逐一对应就好了。



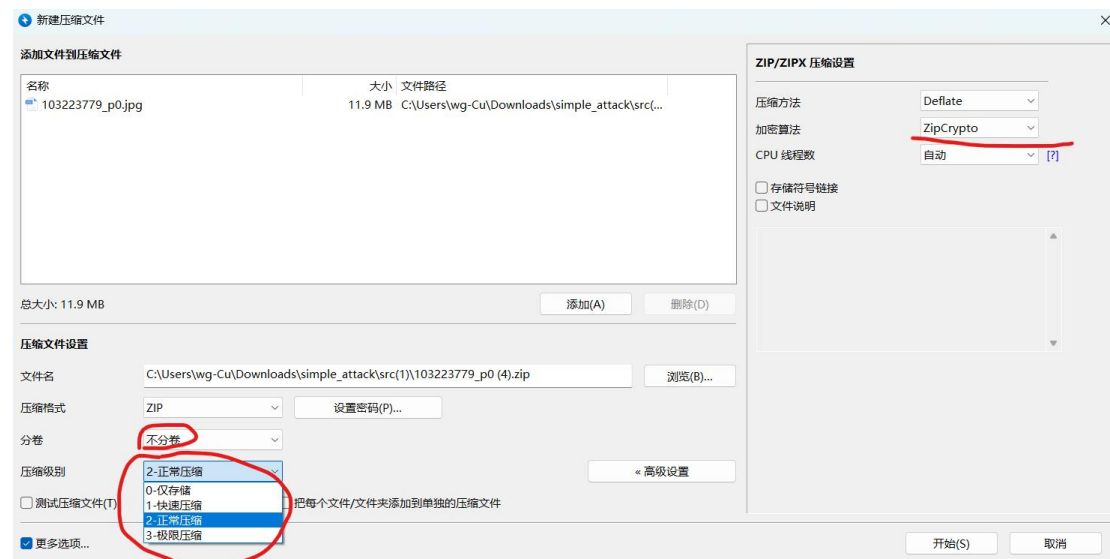
3. simple_attack

压缩包解密题，里面一张图片加另一个压缩包，包中包里有一个加密过的跟外面一样的图片（放 Bandizip 里看 crc 一致且命名一致）和加密过的 txt，压

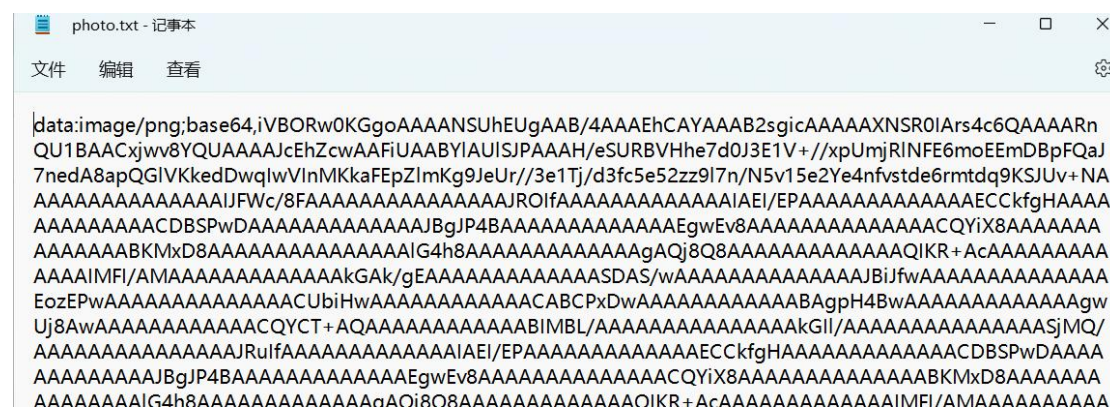
缩算法都是 ZipCrypto，那基本就是明文攻击的题型了。



所以把外面的那张未加密图片按压缩算法 ZipCrypto 压缩，其他项也要与压缩包内加密的图片一致，压缩级别逐个试过来就是正常压缩，然后放到 ARCHPR 里明文攻击，得到解密后的 attachment 压缩包。

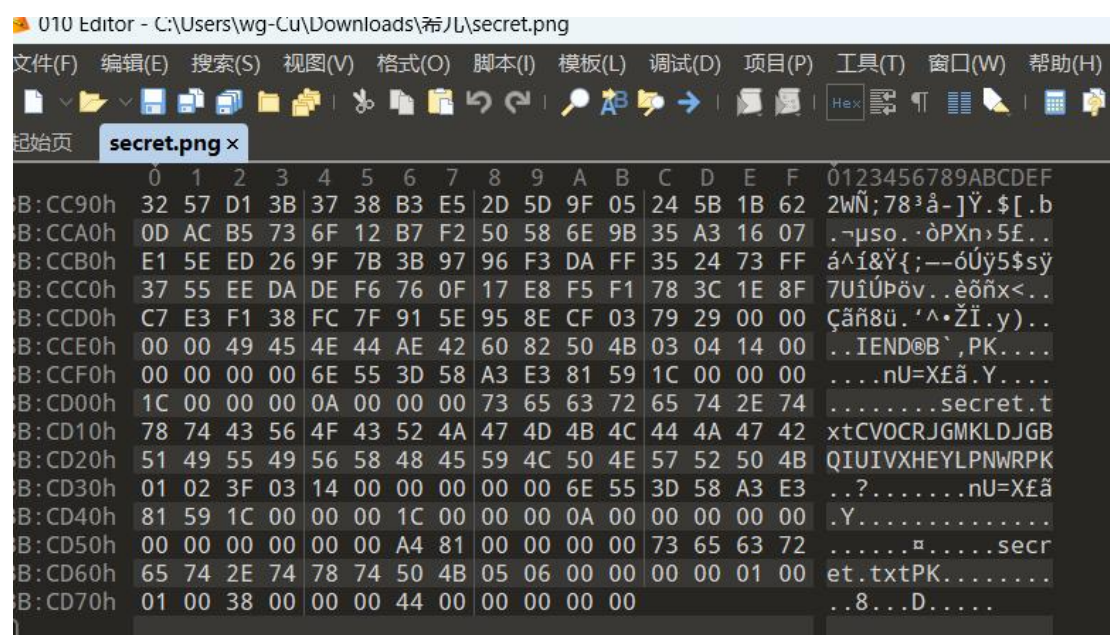


打开 photo.txt 里面是 Data URI scheme 的格式，放到浏览器地址栏里就可以查看图片，即 flag。



4. 希儿希儿希儿

拿到手是一个不正常显示的图片，题目说需要修复，所以在 010Editor 检查了一下，图片格式没有问题，但在末尾有 secret.txt 和 PK，说明图里藏有压缩包，直接改了拓展名，拿到压缩包里的 txt 文件。但没法直接解密，暂时还不知道有什么用。

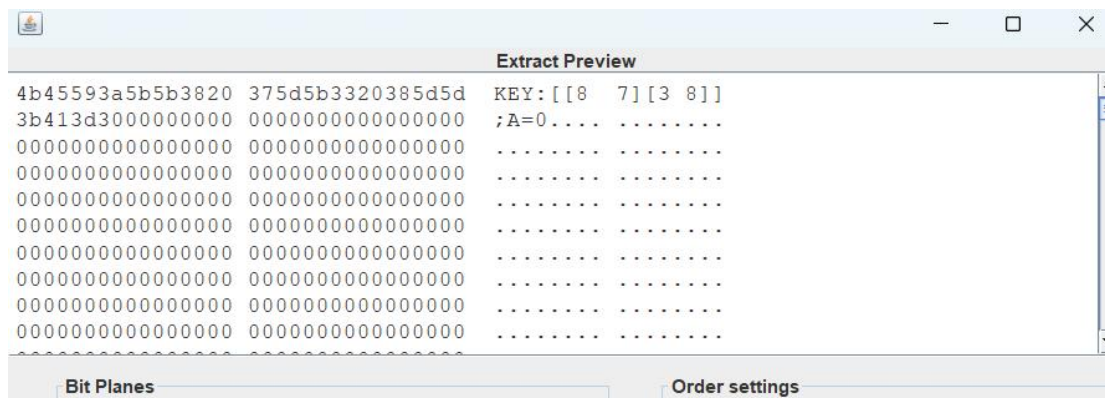


然后想把它放进 Stegsolve 里看看，结果打不开，看来应该是宽高被修改，需要 crc 校验。于是到 python 脚本里跑出正常宽高并在 010Editor 修改，得到希儿的正常图片。



接下来就可以在 Stegsolve 里正常打开，顺便检查了一下属性没什么问题，

但发现了 LSB 隐写藏着可疑的数据。



因为一开始没有好好看题，结果始终不知道这个到底是怎么用的，甚至后来以为这里只是我多想了转而去尝试其他隐写，直到我又看了一遍题：“希儿希儿希尔”最后一个“希尔”且“不过他似乎忘了这个加密的名字不是希儿了”，也就是说题目已经给出提示，去网上一搜“希尔加密”还真有，然后在线解密网站解决。

5. 签到

真真真 • 签到

Pwn

1. EzSignIn

连上去就完了《真真真 • 签到》

Reverse

1. ezASM

一开始去临时学习了一下汇编知识，后来感觉没必要，像 C 语言理解应该就可以，于是把上面 c 里的数和 0x22 异或一下，ASCII 转文字就是 flag 了。

```
section .data
c db 74, 69, 67, 79, 71, 89, 99, 113, 111, 125, 107, 81, 125, 107, 79, 82, 18, 80, 86, 22, 76, 86, 125, 22, 125, 112, 71, 84, 17, 80, 81, 17, 95, 34
flag db 33 dup(0) //定义了33个0
format db "plz input your flag: ", 0
success db "Congratulations!", 0
failure db "Sry, plz try again", 0

section .text
global _start

_start:
; Print prompt
mov eax, 4 //将4的内容移至eax(累加寄存器)，移动后4的内容仍在，eax的内容被4覆盖。
mov ebx, 1 // (基址寄存器)
mov ecx, format // (计数寄存器)
mov edx, 20 // (数据寄存器)
int 0x80 //int 表示中断,编号为0x80是中断号在 Linux 中,0x80中断处理程序是内核,用于其他程序对内核进行系统调用。

; Read user input
mov eax, 3
mov ebx, 0
mov ecx, flag
mov edx, 33
int 0x80

; Check flag
xor esi, esi
check_flag:
mov al, byte [flag + esi]
xor al, 0x22
cmp al, byte [c + esi]
jne failure_check
```

2. ezUPX

如题，是个 UPX，所以先用 upx 去壳即可

```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.22621.3007]
(c) Microsoft Corporation。保留所有权利。

C:\Users\wg-Cu>cd C:\Users\wg-Cu\Downloads\upx-4.2.2-win64

C:\Users\wg-Cu\Downloads\upx-4.2.2-win64>upx -d C:\Users\wg-Cu\Downloads\ezUPX\ezUPX.exe
      Ultimate Packer for eXecutables
      Copyright (C) 1996 - 2024
UPX 4.2.2      Markus Oberhumer, Laszlo Molnar & John Reiser      Jan 3rd 2024

  File size      Ratio      Format      Name
-----
  10752 <-      8192      76.19%     win64/pe     ezUPX.exe

Unpacked 1 file.

C:\Users\wg-Cu\Downloads\upx-4.2.2-win64>
```

```

1 int __cdecl main(int argc, const char **argv, const char **envp)
2 {
3     int v3; // edx
4     __int64 i; // rax
5     __int128 v6[2]; // [rsp+20h] [rbp-38h] BYREF
6     int v7; // [rsp+40h] [rbp-18h]
7
8     memset(v6, 0, sizeof(v6));
9     v7 = 0;
10    sub_140001020("plz input your flag:\n");
11    sub_140001080("%36s");
12    v3 = 0;
13    for ( i = 0i64; (*((__BYTE *)v6 + i) ^ 0x32) == byte_1400022A0[i]; ++i )
14    {
15        if ( (unsigned int)++v3 >= 0x25 )
16        {
17            sub_140001020("Cooool!You really know a little of UPX!");
18            return 0;
19        }
20    }
21    sub_140001020("Sry,try again plz...");
22    return 0;
23 }

```

输入的 flag 要与 0x32 异或后等于 byte_1400022A0 的内容，所以找到并异或回去就得到 flag。（新学的快捷键 shift+e 导出这些文本）

```

; BYTE byte_1400022A0[48]
; DATA XREF: main+36fo
'3 60 49 65 5D 45 13+byte_1400022A0 db 'd','{','v','s',' ','I','e','j','E','\x13k','\x02GmY','','\x02Em\x06m^\x03FF^\x01m\x02Imgbj\x1302'
'D 59 5C 02 45 6D 06+ ; DATA XREF: main+36fo
'4 66 5E 01 6D 02 54+db '\^', 2 dup('F'),'^', '\^', 'm',' ', 'T','m','g','b','j','\x3h','O','2', 0Bh dup(0)
; Size
dd 0 ; Time stamp
dw 2 dup(0) ; Version: 0.0

```

```

1 key='2'
2 pwd='d{vs`IejE\x13k\x02GmY\\ \x02Em\x06m^\x03FF^\x01m\x02Imgbj\x1302'
3 #解密
4 result=[]
5 #pwd为密文
6 for j in range(len(pwd)):
7     result.append(chr(ord(pwd[j])^ord(key))) #跟KEY异或回去就是原文
8 result=''.join(result)
9 print(result)
10

```

3. ezIDA

一进 IDA 差不多就看到了。

Web

1. Bypass it

一开始想偏了，以为要绕过别的什么，但其实就是绕过对注册的阻止就行，查看 html，可以看到注册相关的页面地址。

```
</li>
<li>
  <label> </label>
  <input type="submit" name="login" value="登录" />
  <a href="/register_page.php">注册</a>
</li>
</ul>
1 <html>
2 <head>
3   <meta charset="utf-8">
4   <title>用户注册</title>
5   <link rel="stylesheet" href="/css/bootstrap.min.css">
6   <script src="/js/jquery.min.js"></script>
7   <script src="/js/bootstrap.min.js"></script>
8 </head>
9 <body>
10 <div class="container">
11   <form action="register.php" method="post">
12     <fieldset>
13       <legend>用户注册</legend>
14       <ul>
15         <li>
16           <label>用户名:</label>
17           <input type="text" name="username" />
18         </li>
19         <li>
20           <label>密 码:</label>
21           <input type="password" name="password" />
22         </li>
23         <li>
24           <label> </label>
25           <input type="submit" name="register" value="注册" />
26         </li>
27       </ul>
28     </fieldset>
29   </form>
30   <script language="javascript" defer>alert('很抱歉，当前不允许注册');top.location.href='login.html'</script></div>
31 </body>
32 </html>
```

然后直接向 register.php 传参就好了 username=1&password=1®ister=注册。然后登录。

2. ezHTTP

先是“请从 vidar.club 访问这个页面”，Referer=vidar.club 即可

再是“请通过 Mozilla/5.0 (Vidar; VidarOS x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36 Edg/121.0.0.0 访问此页面”，User Agent=Mozilla/5.0 (Vidar; VidarOS x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36 Edg/121.0.0.0 即可

然后“请从本地访问这个页面”，我第一个想到的是 X-Forwarded-For，但是没反应，发现响应头里有提示“Hint: Not XFF”，所以需要更换其他等效的字段，最终 X-Real-IP: 127.0.0.1 可以发挥作用。

于是“Ok, the flag has been given to you ^-^”，去看响应头有“Authorization”，放到 https://jwt.io/解码即可。