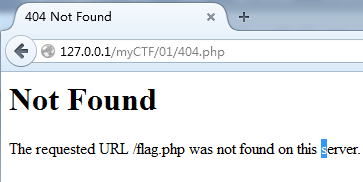
# 目录



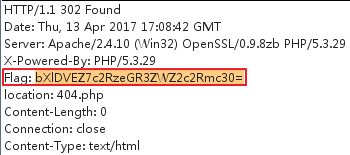
# 第一题：flag在这里

页面中提示flag在这里，但是点击这个链接之后，并没有跳转到有flag的页面。而是404页面。使用burp suite抓包看看。发现有一个flag.php文件重定向了。



点开flag.php的http头信息，发现多了一个Flag，其值一看就知是base64加密。



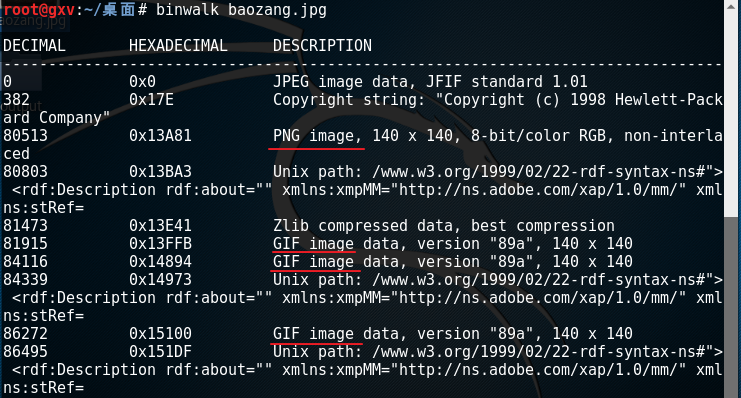
解密就得到flag为：myCTF{sdsxdwefvsdfs}

# 第二题：快打开这个宝盒

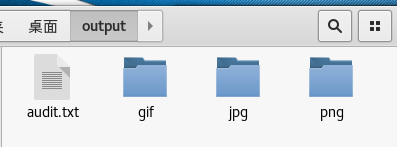


将这个图片下载，复制到kali中。使用binwalk工具对图片进行分析，可以发现，图片中还有四张图片。

命令：binwalk baozang.jpg



使用foremost工具对图片进行自动提取出来。命令：foremost baozang.jpg。分解成功后的文件保存在output文件夹里。



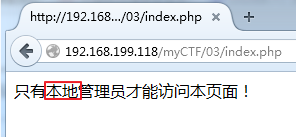
将这些图片复制到一个文件夹，发现有4张疑似二维码的残部。



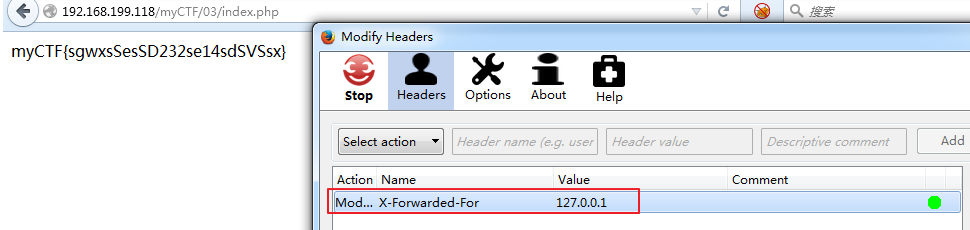
将这四张图片用图片编辑软件拼起来，然后用手机扫描拼成的二维码就可以得到了flag为myCTF{WERsdsd2342sdse}



# 第三题：管理员本地访问



关键字：本地。可以使用firefor插件modify headers修改X-Forwarded-For为127.0.0.1。就可以突破这个限制，得到flag为myCTF{sgwxsSesSD232se14sdSVSsx}



# 第四题：对方不想和你说话，并向你扔了一段代码



通过查看图片中的代码，发现存在变量覆盖。构造利用url如下：

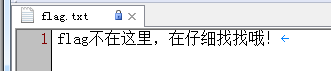
http：//xxx/index.php?a=&b=

得到flag为myCTF{23efwwwsdgweew121efwfwfwf}

# 第五题：下载下载

点击下载flag文件，可以下载到flag.txt。但是里面没有flag





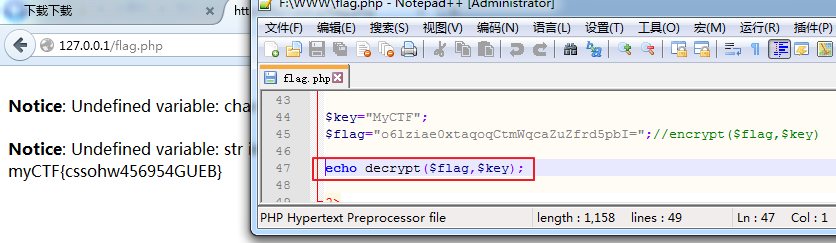
从查看源码发现存在文件flag.php，于是试着构造如下下载url。看看存不存在任意文件下载。

url：http：//xxx/index.php?file=flag.php

发现可以下载，打开代码如下：

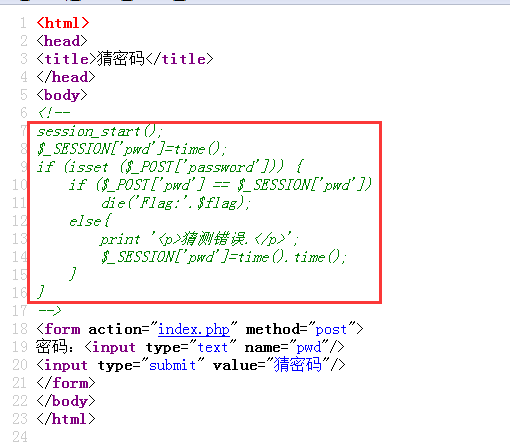


由注释可知道，这个flag是由encrypt函数加密了。但是我们下载的文件提供了加解密函数。我们可以编写代码调用解密函数对密文解密。

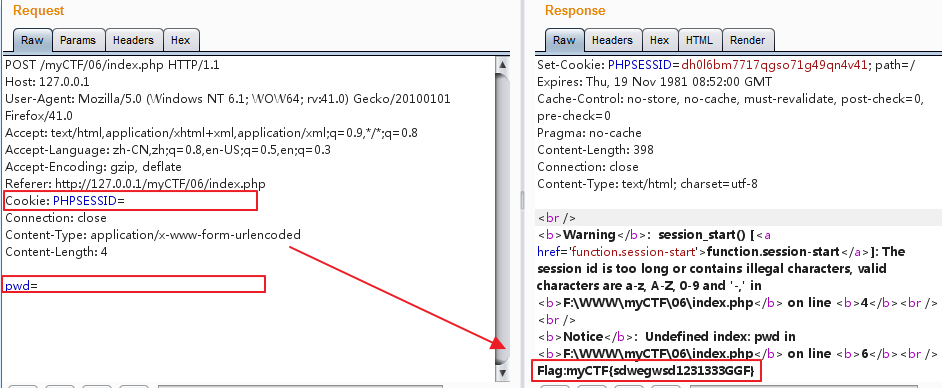


最后得到flag为：myCTF{cssohw456954GUEB}

# 第六题：猜密码

查看页面源码可知，这个密码是两个当下时间戳拼接。我们可以尝试把为了的时间戳都做成字典，让后用burpsuite去爆破。但是这个字典制作比较繁琐。

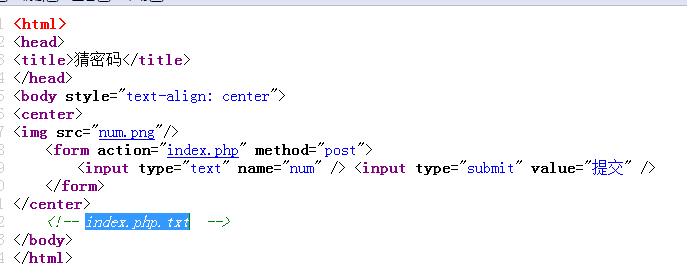
不过我们可以尝试以下方法：将cookie中的PHPSESSID随便改掉（我直接全部删除）。使得服务器无法找到对应的session。这样$\_session['pwd']为空，然后我们再使得提交的参数pwd也为空。这样就可以使得它们相等。



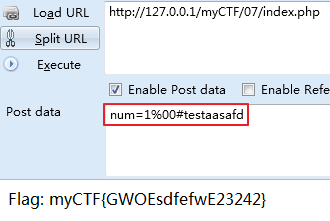
可以得到flag为：Flag:myCTF{sdwegwsd1231333GGF}

# 第七题：你给我一个满意的数字，我就给你flag！

查看页面源码可发现有文件index.php.txt

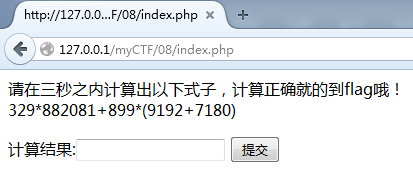


通过index.php.txt文件知道index.php文件的源码。ereg的%00截断来达到if语句的第二个条件，从而执行die ( 'Flag: ' . $flag );。故构造如下post数据：

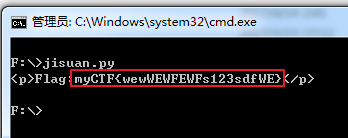


最后得到flag为：myCTF{GWOEsdfefwE23242}

# 第八题：快速计算



这个使用python脚本读取页面进行计算，可得到flag为：myCTF{wewWEWFEWFs123sdfWE}



脚本代码如下：

#coding=utf-8

import requests,re

s = requests.Session()

url = 'http://192.168.199.118/myCTF/08/index.php'

html = s.get(url).content

reg = r'([0-9].+)<'

pattern = re.compile(reg)

match = re.findall(pattern,html)

payload = {'result': eval(match[0])}

print s.post(url, data=payload).content

# 第九题：该网站已经被黑

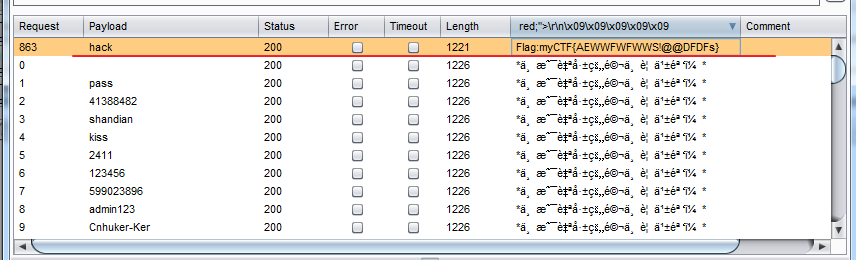


该网站被黑，说明有可能存在黑客留下的webshell。我们可以利用御剑目录扫描，扫除webshel的路径。



访问该页面是一个webshell



使用burp抓包暴力破解，字典尤为重要。网上公布有黑客们常用的webshell密码，将其收集为字典。发现密码为hack。

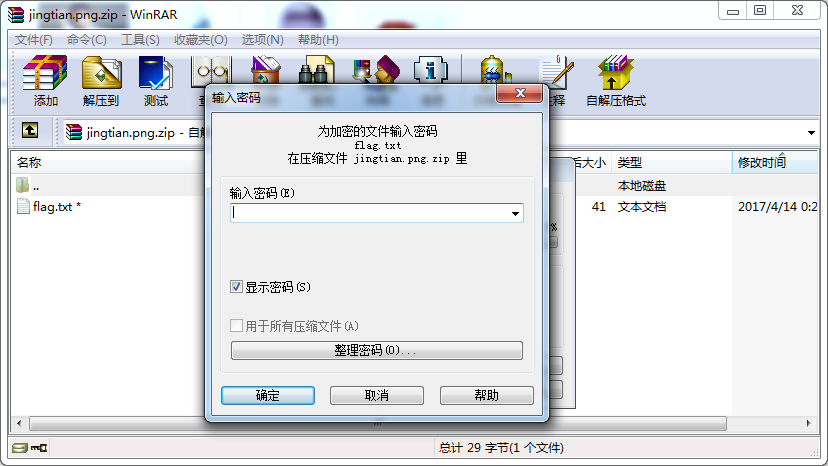
最终得到flag为myCTF{AEWWFWFWWS!@@DFDFs}

# 第十题：厉害了，word景甜

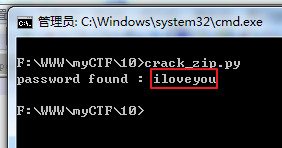


下载这个图片，使用binwalk分析，发现图片中隐藏着一个zip文件。故将后缀改为zip

。并用winrar软件解压。但是有密码。



可以考虑很多人zip的解压软件，加载一些字典来破解!这里使用python脚本配合12306top100肉口令。进行破解得到密码：iloveyou。使用该密码解压就发现flag在flag.txt里。



脚本代码如下：

#coding: utf-8

import zipfile

import threading

def zipbp(zfile, pwd):

try:

zfile.extractall(pwd=pwd)

print 'password found : %s' % pwd

except:

return

def main():

zfile = zipfile.ZipFile('jingtian.zip')

pwdall = open('12306\_password\_top100.txt')

for pwda in pwdall.readlines():

pwd = pwda.strip('\n')

t = threading.Thread(target=zipbp, args=(zfile, pwd))

t.start()

t.join()

if \_\_name\_\_ == '\_\_main\_\_':

main()