2020 VidarTeam Hgame Writeup By ITFS

Web

Cosmos的新语言

首先给了段源码

```php
<?php
highlight_file(__FILE__);
$code = file_get_contents('mycode');
eval($code);
```

c9541539cfc6f296dg2f7c8dd:6g4db9

然后代码审计发现包含了一个"mycode"文件，于是改一下URL

```php
function encrypt($str){
    $result = '';
    for($i = 0; $i &lt; strlen($str); $i++){
        $result .= chr(ord($str[$i]) + 1);
    }
    return $result;
}

echo(encrypt(base64_encode(base64_encode(strrev(strrev(strrev(strrev(encrypt(str_rot13(base64_encode($_SERVER['token'])))))))))));

if(@$_POST['token'] === $_SERVER['token']){
    echo($_SERVER['flag']);
}
```

发现是一段加密代码，将flag加密了，根据前面的代码可以知道需要对那段字符串解密得到`token`，然后post回去才能拿到flag

一开始想的是手动解码，但是后来发现这段加密代码一直在不断的变化，解出来的token也在不断变化，大约5秒左右变化一次，于是想到用Python拆解加密代码然后进行分析，直接得到解密步骤，拿到`token`再直接post过去。exp代码如下

```python
import requests
import base64
import string
def decrypt(token):
    result = ""
    for x in token:
        result += chr(ord(x)-1)
    return result
def strrev(token):
    return token[::-1]
def b64decode(token):
    return base64.b64decode(token.encode()).decode()
def rot13(token):
    result = ""
    for x in token:
        if x in string.ascii_lowercase:
            result += chr(((ord(x)-ord('a')+13)%26)+ord('a'))
```

```
        elif x in string.ascii_uppercase:
            result += chr(((ord(x)-ord('A')+13)%26)+ord('A'))
        else:
            result += x
    return result
url = "http://33cb992b0c.php.hgame.n3ko.co/"
php = "http://33cb992b0c.php.hgame.n3ko.co/mycode"
r = requests.get(url)
code = requests.get(php)
token = r.text[625:-21]
en = code.text[164:-75]
de = en.split('(')
de.pop()
for step in de:
    if step == "encrypt":
        token = decrypt(token)
    elif step == "base64_encode":
        token = b64decode(token)
    elif step == "str_rot13":
        token = rot13(token)
    elif step == "strrev":
        token = strrev(token)
flag = requests.post(url, data={"token":token})
print(flag.text)
```

最后拿到flag

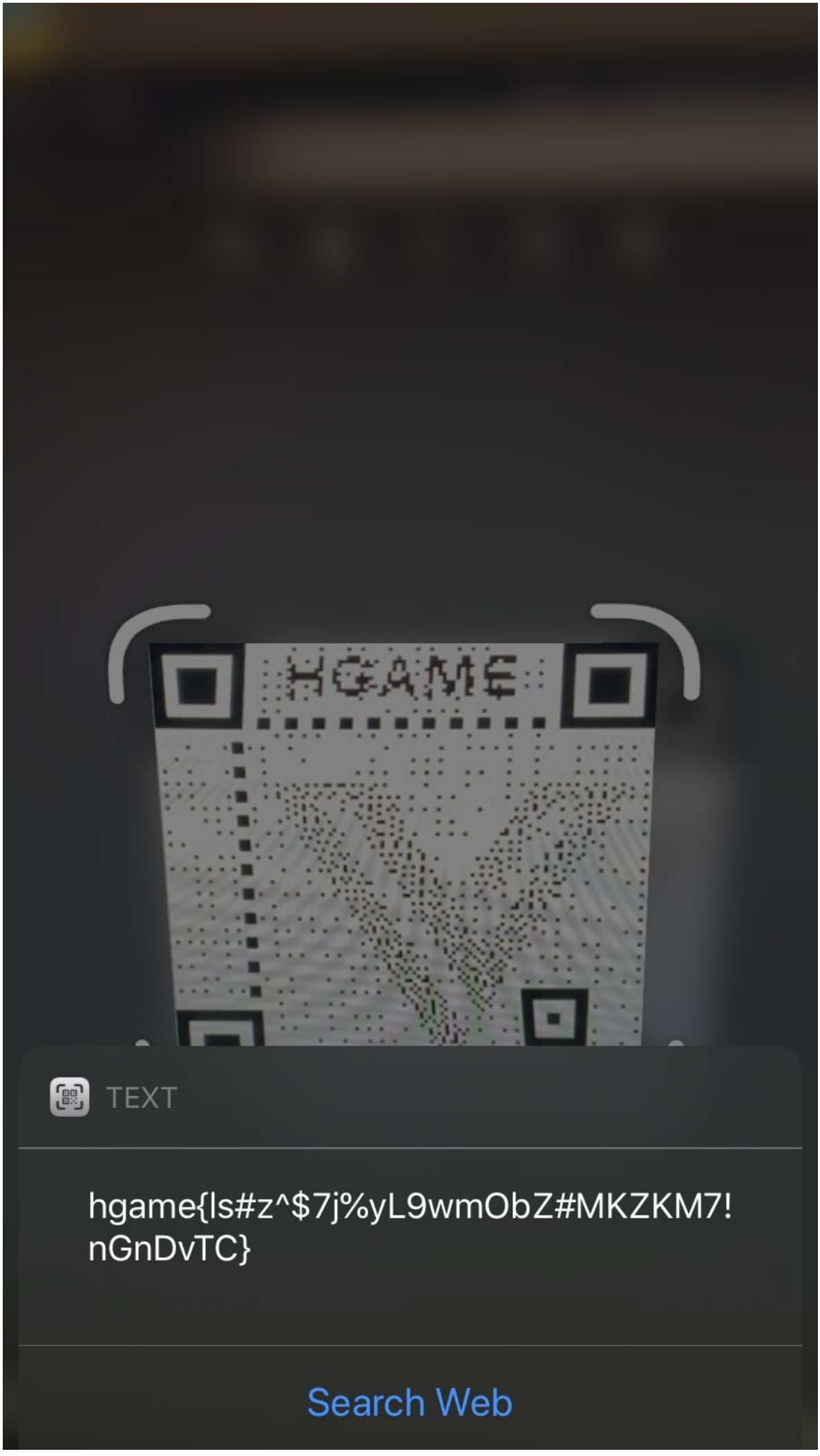
```
</span>
</code><br>
Pqy|PN[~PqVqFXE{Q{R8EKu}FUx7E~Q|PKH9Eqx8^{LA<br>
hgame{$iMpLe~$CrIPt~WITH-pythoN~0r-pHp}
</body>
</html>
```

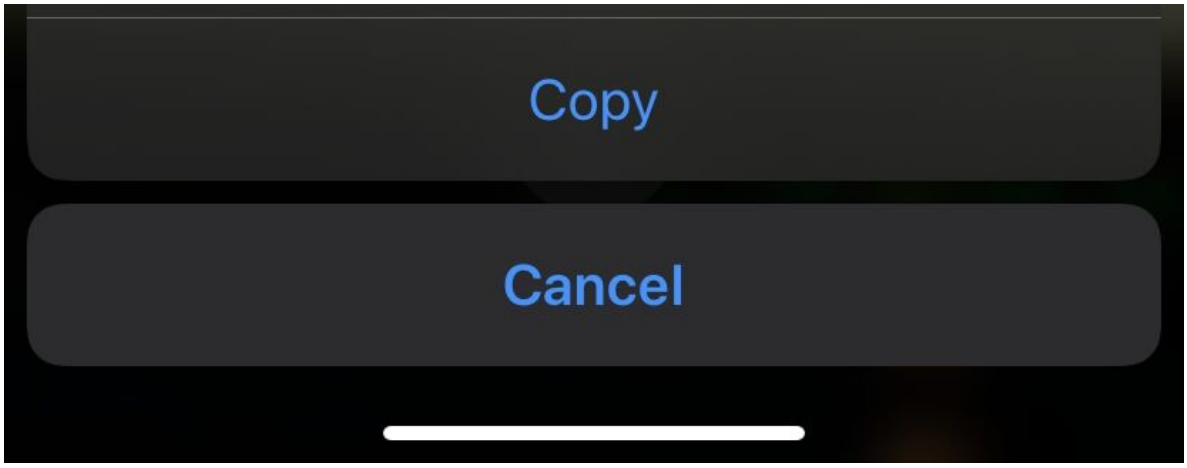
Misc

Cosmos的午餐

拿到的是一个pacpng文件和一个ssl通信的log文件，于是可以知道是HTTPS流量分析

首先用wireshark打开，导入log得到明文通信内容，然后追踪流发现一个压缩包文件，里面有个`Outguess with key.jpg`，可以知道是outguess隐写，于是在网上找了相应的解密工具，得到隐写内容是一个url，下载后是一个压缩包，里面有张二维码，扫描拿到flag


HGAME
TEXT
hgame{ls#z^$7j%yL9wmObZ#MKZKM7!
nGnDvTC}
Search Web



所见即所假

拿到的是一个压缩包，里面有张图片

提示F5 key，于是得知是F5隐写，在网上找了对应的工具，用key解密得到一串hex

在线转字符串拿到flag

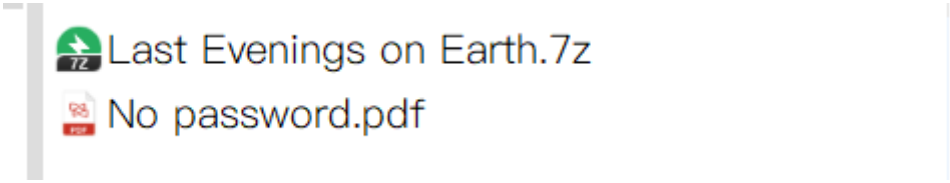
Rar!□□□3□□□

□□□□□□□□□Rz□$□□□□□□ ◆[◆-◆□flag.txt

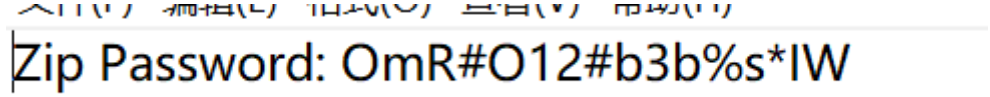
□□□□le□□□□hgame{4087^z#msw34ERtnFUyqpKUk2dmLPW60}□wVQ□□□

地球上最后的夜晚

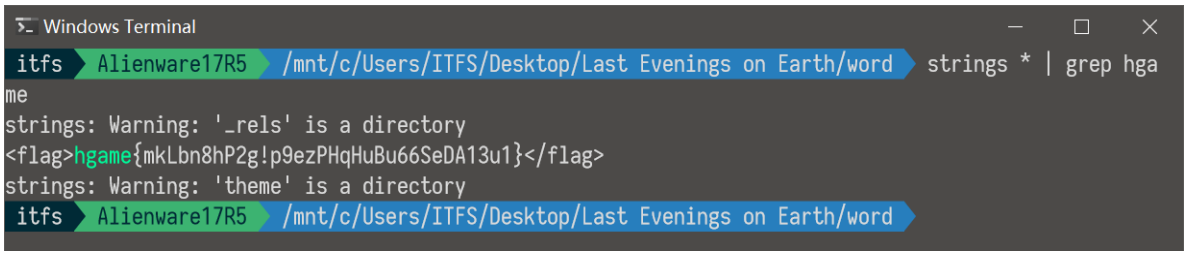
拿到一个加密的压缩包和一个pdf



于是猜想是`wbStego`隐写，试了下果然得到了压缩包密码



解压后是一个doc，里面是篇小说，没有找到隐藏文字，于是把这个文件解压到word文件夹里去找



搜索字符串`hgame`，拿到flag

Crypto

签到题

直接给了一段Python代码，分析了一下发现有两个关键的函数

```python
def proof_of_work(self):
    random.seed( os.urandom(8) )
    proof = ''.join([ random.choice(string.ascii_letters+string.digits) for _ in range(20) ])
    _hexdigest = sha256(proof.encode()).hexdigest()
    self.send(str.encode( "sha256(XXXX+%s) == %s" % (proof[4:],_hexdigest) ))
    x = self.recv(prompt=b'Give me XXXX: ')
    if len(x) != 4 or sha256(x+proof[4:].encode()).hexdigest() != _hexdigest:
        return False
    return True
```

```python
self.send(b'The secret code?')
_code = self.recv()
if _code == b'I like playing Hgame':
    self.send(b'Ok, you find me.')
    self.send(b'Here is the flag: ' + FLAG)
    self.send(b'Bye~')
else:
    self.send(b'Rua!!!')
```

看起来需要找到所给字符串的前四位，用了sha256算法，然后输入`I like playing Hgame`即可

打开`hashcat`，先用Python连接上打开交互模式，拿到sha256密文和残缺的明文，然后构造一个掩码字符集，用`hashcat`进行sha256掩码攻击

```
E:\tools\Misc\hashcat-4.1.0\hashcat64.exe -a 3 --force -c
abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789 -m 1400
cbd0841aed15ae0ae263b7a1863046b1880e6f6290033a234e96814b325c0c10
!c!c!c!cIO7mztG5K9Xn3fUo
```

得到前四位`nmCz`

```
itfs Alienware17R5 /mnt/c/Users/ITFS/Desktop python 1.py
[+] Opening connection to 47.98.192.231 on port 25678: Done
[*] Switching to interactive mode
sha256(XXXX+IO7mztG5K9Xn3fUo) == cbd0841aed15ae0ae263b7a1863046b1880e6f6290033a234e96814b325c0c10
Give me XXXX: $ nmCz
The secret code?
> $ I like playing Hgame
Ok, you find me.
Here is the flag: hgame{It3Rt0O|S+I5_u$3fu1~Fo2_6rUtE-f0Rc3}
Bye~
[*] Got EOF while reading in interactive
$
```

拿到flag