171126 Misc-湖湘杯部分

原创

奈沙夜影 于 2017-11-28 23:20:46 发布 503 收藏

分类专栏： CTF

本文链接：https://blog.csdn.net/whklhhhh/article/details/78660514

1625-5 王子昂 总结《2017年11月26日》【连续第422天总结】

A. 湖湘杯部分Misc-WriteUp

B.

流量分析

Wireshark分析数据包，转为HTTP对象后查看，发现有一个文件名很明显：

2967	qex.f.360.cn	multipart/form-data	1267 bytes	qexquery
2969	qex.f.360.cn	application/octet-stream	197 bytes	qexquery
3100	qex.f.360.cn	multipart/form-data	1291 bytes	qexquery
3102	qex.f.360.cn	application/octet-stream	197 bytes	qexquery
3202	192.168.199.203	application/zip	54 kB	flag.zip
3262	qex.f.360.cn	multipart/form-data	1267 bytes	qexquery
3264	qex.f.360.cn	application/octet-stream	197 bytes	qexquery
3322	x.jd.com		1440 bytes	exsites?spread_type=2&ad_ids=1!

下载下来解压发现是RGB值，于是用脚本显示

对行数98457进行质因子分解，得到3 x 37 x 887，猜测111x887：

```python
from PIL import Image

x = 887     #x坐标  通过对txt里的行数进行整数分解
y = 111     #y坐标  x * y = 行数

im = Image.new("RGB", (x, y))    #创建图片
file = open('ce.txt')     #打开rbg值的文件

#通过每个rgb点生成图片

for i in range(0, x):
    for j in range(0, y):
        line = file.readline()  #获取一行的rgb值
        rgb = line.split(", ")  #分离rgb，文本中逗号后面有空格
        im.putpixel((i, j), (int(rgb[0]), int(rgb[1]), int(rgb[2])))     #将rgb转化为像素

im.show()    #也可用im.save('flag.jpg')保存下来
```


Encryptor.apk

反编译，发现加密函数为循环异或md5(key)

```
private byte[] encryptData(byte[] arg7, byte[] arg8) {
    byte v3 = ((byte)arg8.length);
    byte[] v0 = new byte[arg7.length];
    int v2;
    for(v2 = 0; v2 < arg7.length; ++v2) {
        v0[v2] = ((byte)(arg7[v2] ^ arg8[v2 % v3]));
    }

    return v0;
}
```



```
public void onEncryptClicked(View arg11) {
    File v4;
    byte[] v0;
    String v5 = this.findViewById(2131296323).getText().toString();
    if(v5.equals("")) {
        Toast.makeText(this.getApplicationContext(), "You must enter a password!", 0).show();
        return;
    }

    byte[] v3 = this.md5(v5);
    if(this.file.toString().length() == 0) {
        Toast.makeText(this.getApplicationContext(), "You must select a file!", 0).show();
        return;
    }

    try {
        v0 = this.encryptData(this.readUri(this.file), v3);
    }
    catch(IOException v1) {
        Toast.makeText(this.getApplicationContext(), "Could not read file", 0).show();
        return;
    }
```


而题目没有给出key相关的内容，也不知道源文件的格式来反推key
但是运行程序发现password中默认有一些字符，猜测是这些默认key，寻找R.smali，发现可疑字符串：

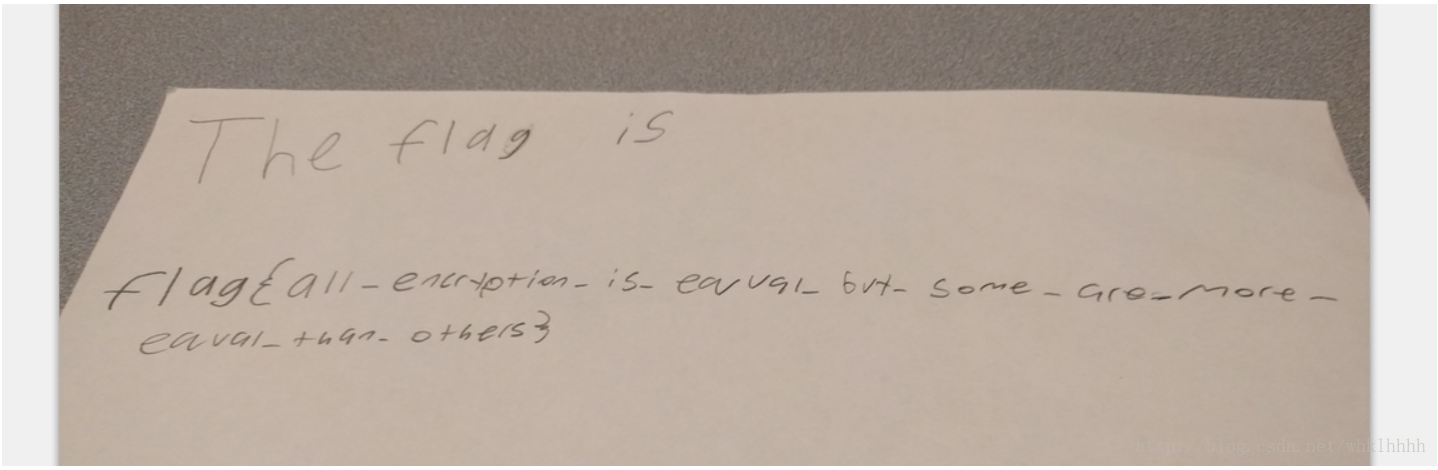
```
<string name="hello_world">
    Hello world!</string>
<string name="password_text">
    Password</string>
```


自己加密了一张图片后验证key是Password
脚本解密得到原图片：

```python
import hashlib
key = "hello_world"
key = "Password"
key = hashlib.md5(key.encode('utf-8')).hexdigest()
print(key)
key = bytes.fromhex(key)
f = open("flag.encrypted", "rb")
c = f.read()
f.close()


f = open("flag.jpg", "wb")
for i in range(len(c)):
    k = bytes((c[i] ^ (key[i%len(key)]), ))
    f.write(k)

f.close()

```

```
运行 misc3
E:\Users\hasee\AppData\Local\Programs\Python\Python35-32\python.exe F:/
dc647eb65e6711e155375218212b3964

进程已结束,退出代码0
```


（也可以直接把加密文件加上后缀名jpg后放入手机里再次加密，因为异或是对称密钥加密，所以可以直接解密得到图片）



C. 明日计划

上海CTF线下赛总结