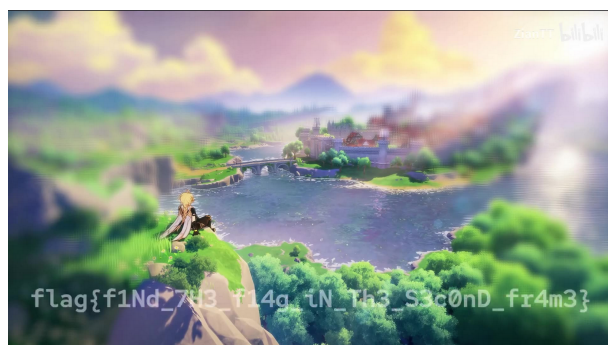


X...什么？

根据题目说明是chrome提示，结合版本号chrome-114.0.5735.90直接google搜索漏洞发现此链接<https://github.com/xcanwin/CVE-2023-4357-Chrome-XXE>，接着就是改一下代码去读/flag就可以了，记得改响应头哦（打了半天没打通。。。发现最后没改）改成svg格式即可。

非常好忽悠混合

第二阶段使用工具ffmpeg去分离视频帧即可，flag就在前几帧。



心中无码，自然高清



阅读生成马赛克代码可以发现，是将字体所在的上下区块进行了平均值进行填充的，那么就可以直接跑一遍所有的字符的平均值，然后再读取要破解的马赛克的平均值，进行一次对比即可得到flag。




bepbep

第一段音频很明显是摩斯电码。找一个在线的识别网站即可。

<https://morsecode.world/international/decoder/audio-decoder-adaptive.html>

Use the microphone: Or analyse an audio file containing Morse code:

Listen  Stop 

Upload  Play  Stop 

Filename: "morse.wav"

FLAGG3TF14GFRoMMOR3ECAD3ADDUNDERLINEBETWEENEACHWORDGOTOH
TTPS://STORAGE.WEAROS.FANS/AMARKET/NAME.WAV<KN>ALLLOWERCASE)

得到提示后进行下一步的音频，听了一下想到之前的一个外星人音频的题。



得知应该是sstv慢扫描电视，那么直接去下载软件。配置一下虚拟声卡，将输出接入到输入中播放音频即可得到flag。

我朝，大盒

Flag1去exiftool查看一下照片的信息可以得到gps信息然后多试几个就出来了

Flag2的话根据短信的信息去列车过站查询，多试几次也差不多

躺平问答

1. hit也就是哈尔滨工业大学
2. 这个直接google搜索可以得到中文页面，然后删去后面的cn就是英文页面



- 3.
- 可以找到一個問卷調查就是我們需要的答案
4. 這個直接google大不了把那幾個全試一遍也行
5. 這個我直接爆破的 因為感覺沒那么多。。。
6. 基礎的域名查詢
7. 企查查

躺平论坛

Flag1就f12即可

Flag2想的太复杂了 最后才注意到题目是找不到的 那也就是意味着找起来很麻烦, 注意到url中<https://forum.tpcup.org/d/1048-random>后面有数字也就是1-1048发现1为第一道题, 那么直接从2-1048开始爆破即可。

```
import requests

for i in range(257,1049):
    url = 'https://forum.tpcup.org/d/' + str(i) + '-random'
    re = requests.get(url)
    if 'flag{' in re.text:
        print(re.text)
        break
    else:
        print(i)
```

中途掉了我就从257开始的。


躺平聊天室2.0

基础的安卓逆向

Flag1 jadx打开直接搜字符串flag{即可

Flag2 搜一下flag2或者flag3字样，发现flag3搜索到

```
ConstraintLayout constraintLayout = (ConstraintLayout) c.o(inflate, R.id.flag3);
if (constraintLayout != null) {
    i4 = R.id.imageView;
    ImageView imageView = (ImageView) c.o(inflate, R.id.imageView);
    if (imageView != null) {
        i4 = R.id.imageView2;
        ImageView imageView2 = (ImageView) c.o(inflate, R.id.imageView2);
        if (imageView2 != null) {
            i4 = R.id.imageView3;
            ImageView imageView3 = (ImageView) c.o(inflate, R.id.imageView3);
            if (imageView3 != null) {
                ConstraintLayout constraintLayout2 = (ConstraintLayout) inflate;
                this.f1822u = new f4(constraintLayout2, button, editText, constraintLayout imageView, imageView2, imageView3);
                setContentView(constraintLayout2);
                try {
                    new t2.a(this, new String(Base64.getDecoder().decode("ZmxhZ3RmbGFuYmFzZTY0fQ=="), "UTF-8")).start();
                    return;
                } catch (UnsupportedEncodingException e4) {
                    throw new RuntimeException(e4);
                }
            }
        }
    }
}
```



Base64解码即可

Flag3那么就是只能看看flag2了（如果你下载了apk运行过 你就会知道flag2生成flag3 所以搜索flag2去看看哪里运算了）。

在flag2的代码下可以看见一个按钮，这就是flag3的触发。

```
public void onCreateFlag3Click(View view) {
    String obj = ((EditText) this.f1822u.f400c).getText().toString();
    if (obj.length() != 16) {
        p2.m.f(view).g();
    } else {
        new t2.b(this, obj, view).start();
    }
}
```

可以看见初始化了一个b对象并调用了start函数点进去看看。

```
@Override // java.lang.Thread, java.lang.Runnable
public final void run() {
    try {
        SecretKeySpec secretKeySpec = new SecretKeySpec(this.f4475b.getBytes("UTF-8"), "AES");
        Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");
        cipher.init(2, secretKeySpec);
        String str = new String(cipher.doFinal(Base64.decode("e+Fy/ONEqo3VBIXzCLZ6Kx+vjukEgXfk0aet9ti3hrc=", 0)), "UTF-8");
        MainActivity mainActivity = this.f4477d;
        MainActivity.n(mainActivity, (ImageView) mainActivity.f1822u.f404g, "3.".concat(str));
    } catch (Exception unused) {
        m.f(this.f4476c).g();
    }
}
```

发现aes加密key为flag2，那么直接去在线解密即可。

AES加密模式: ECB
填充: pkcs5padding
密钥长度: 128位
密钥: flag {flagbase64}
输出: base64

e+Fy/ONEqoJVBIXzCLZ6Kx+vjukEgXFkOaet9ti3hrc=

AES加密
AES解密
复制结果
清空所有

flag {alldone!!!}

HIT! 准入认证!

Flag1就直接字符串列表里可以看见

Flag2是里面最麻烦的，我直接用python实现了其中的两个加密函数，然后去进行爆破因为flag的格式为flag{...}。写好函数后将前五个值赋值为flag{ 的ascii码，然后运行一下看看得到的结果是不是和%%xv\$V^DlcLABnMaxNF^idm*OVr^r^?v-AJoQJczCo\$.前面对得上，在爆破中发现并不是单字节的，会影响到后面的，于是我直接没有细看，自己写脚本看看到底如何影响的。然后手动去爆破。不过手动进行的时候会有多个值都可以对得上后续的，那么就再用这多个值再跑一次，其中一般只会有一个值跑的通，然后再选定所给的值，最后看看能否组成有意义的话即可，整个过程还是很麻烦的，也许是我的方法不对吧，脚本写的也很繁琐了就。就直接贴下面了，这个就不用参考了，有思路直接自己去写就可以。

```
tap =list("flag{ZI4N7T_a7t3nd_ycJh_w1nt3r_c4mp_1N_h1t}.")
```

```
aAsdfghjklqwerty=list("asdfghjklqwertyuiopzxcvbnmQWERTYUIOPZXCVBNMASDFGHJKL$%^)!)@#&*(-?H")
```

```
import string
```

```
a2 = list('*****')
```

```
a3 = 32
```

```

def sub_125E(a1,a2,a3):

    v14 = 0

    for i in range(0,a3,3):

        v3 = v14

        v15 = v14 + 1

        a2[v3] = aAsdfghjklqwert[a1[i] >> 2]

        if (a3 <= i + 1):

            v4 = 0

        else:

            v4 = a1[i + 1] >> 4

        v5 = v4 + ((16 * a1[i]) & 0x30)

        v6 = v15

        v16 = v15 + 1

        a2[v6] = aAsdfghjklqwert[v5]

        if (a3 <= i + 1):

            v8 = 46

        else:

            if (a3 <= i + 2):

                v7 = 0

            else:

                v7 = a1[i + 2] >> 6

            v8 = aAsdfghjklqwert[((4 * a1[i + 1]) & 0x3C) + v7]

        v9 = v16

        v17 = v16 + 1

        a2[v9] = v8

        if (a3 <= i + 2):

```

```

        v10 = chr(46)

    else:

        v10 = aAsdfghjklqwerty[a1[i + 2] & 0x3F]

    v11 = v17

    v14 = v17 + 1

    a2[v11] = v10

    return ('.join(a2))

```

```

a3 = [

    1, 9, 205, 18, 122, 158, 121, 241, 116, 25,

    25, 12, 219, 111, 37, 20, 221, 97, 19, 226,

    139, 188, 196, 38, 131, 190, 184, 112, 170, 74,

    144, 88];

```

```

s2 = list("%%xv$V^DlcLABnMaxNF^idm*OVr^r^?v-AJoQJczCo$.")

```

```

ppp =[102, 108, 97, 103, 123, 55, 104, 51, 95, 72, 73, 84, 95, 51, 55, 59, 44, 21,
62, 35, 57, 98, 76, 48, 41, 47, 69, 30, 28, 29, 75, 125]

```

```

# for i in range(400,415):

#     ppp[13] = i

#     for j in range(1,500):

#         ppp[14] = j

#         tmp = sub_125E(ppp,a2,32)

```

```
#         if tmp[18] == 'F':  
  
#         print(tmp, i,j)
```

```
def part_2(a1,a2,a3,a4,a5):  
  
    for i in range(a5):  
  
        a4[i] = (a2 * a1[i] + a3[i]) & 0xff  
  
    return a4
```

```
flag = list('flag{*****}')  
  
text = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'a', 'b', 'c', 'd', 'e',  
'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u',  
'v', 'w', 'x', 'y', 'z', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K',  
'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z', '_']
```

```
compare = "%%xv$V^DlcLABnMaxNF^idm*OVr^r^?v-AJoQJczCo$."
```

```
# for i in range(5,31):  
  
#     for j in text:  
  
#         flag[i] = j  
  
#         print(sub_125E(flag,a2,32))
```

```
def translate(op):  
  
    for i in range(len(op)):  
  
        op[i] = ord(op[i])  
  
    return op
```



```

# for i in text:

#     ppp[5] = ord(i)

#     ppp = part_2(ppp,233,a3,ppp,32)

#     tmp = sub_125E(ppp,a2,32)

#     print(tmp[:5])


# for i in range(5,31):

#     for j in text:

#         ppp[i] = ord(j)

#         tmp1 = ppp.copy()

#         asd = part_2(tmp1,233,a3,tmp1,32)

#         tmp = sub_125E(asd,a2,32)

#         if tmp[:i] == compare[:i]:

#             print(tmp)

#             break


# print(ppp)


# ml = part_2(ppp,233,a3,ppp,32)

# print(sub_125E(ml,a2,32))

ppp[13] = 76

ppp[14] = 105

ppp[15] = 76

```

```
ppp[16] = 97
```

```
ppp[17] = 99
```

```
ppp[18] = 95
```

```
ppp[19] = 105
```

```
ppp[20] = 115
```

```
ppp[21] = 95
```

```
ppp[22] = 115
```

```
ppp[23] = 48
```

```
ppp[24] = 95
```

```
ppp[25] = 78
```

```
ppp[26] = 49
```

```
ppp[27] = 67
```

```
ppp[28] = 51
```

```
tmp_exet =
```

```
['6','7','8','b','c','d','m','n','o','y','z','A','B','C','L','M','N','W','X','Y']
```

```
# for j in text:
```

```
#     ppp[30] = ord(j)
```

```
#     tmp1 = ppp.copy()
```

```
#     asd = part_2(tmp1,233,a3,tmp1,32)
```

```
#     tmp = sub_125E(asd,a2,32)
```

```
#     # if tmp[39] == 'z':
```

```
#         print(tmp,j)
```

```
for i in string.printable:
```

```
for j in string.printable:

    ppp[29] = ord(i)

    ppp[30] = ord(j)

    tmp1 = ppp.copy()

    asd = part_2(tmp1,233,a3,tmp1,32)

    tmp = sub_125E(asd,a2,32)

    if tmp[39] == 'z' and tmp[38] == 'c' and tmp[40] == 'C':

        print(tmp,i,j)
```

```
for i in ppp:

    print(chr(i),end="")
```