buuctf 逆向 findit

原创

菜逼的ctf之路 于 2020-10-06 18:10:13 发布 507 收藏 1

本文链接：https://blog.csdn.net/weixin_45701079/article/details/108941757

版权

buuctf 逆向 findit

第一次做这种题，记录一下

文件下载后发现是apk逆向，用jeb打开

```
00A4  .array-data 2 x 0x26
          0x70
          0x76
          0x6B
          0x71
          0x7B
          0x6D
          0x31
          0x36
          0x34
          0x36
          0x37
          0x35
          0x32
          0x36
          0x32
          0x30
          0x33
          0x33
          0x6C
          0x34
          0x6D
          0x34
          0x39
          0x6C
          0x6E
          0x70
          0x37
          0x70
          0x39
          0x6D
          0x6E
          0x6B
          0x32
          0x38
          0x6B
          0x37
          0x35
```


发现字符串，字符找到规律，比如第五个数据是“{”的asscii，最后一个是“}”的ascii，所以大胆猜想一下，这串数据和flag有关，用脚本跑出字符串,贴上脚本。

```
a=[ 0x70,0x76,0x6B,0x71,0x7B,0x6D,0x31,
    0x36,0x34,0x36,0x37,0x35,0x32,0x36,
    0x32,0x30,0x33,0x33,0x6C,0x34,0x6D,
    0x34,0x39,0x6C,0x6E,0x70,0x37,0x70,
    0x39,0x6D,0x6E,0x6B,0x32,0x38,0x6B,
    0x37,0x35,0x7D]
s=''
for i in a:
    s+=chr(i)
print(s)
```

跑出结果

```
pvkq{m16467526203314m491np7p9mnk28k75}
```

结果和flag很像，但不是，查数据，第一个数据是0x70,第二个是0x76,相差6，众所周知，"f"和"l"也差6，大胆猜测，凯撒加密(不懂凯撒加密的百度，很简单)，网络工具直接用，得到结果。

位移长度，自行计算。

总结：这道题我个人感觉有点偏，对于数据不敏感的人很难发现flag，不过这种题做一遍下次就知道怎么做了。