i春秋网鼎杯网络安全大赛Beijing题目writeup

原创

iqiqiya 于 2018-08-20 17:31:37 发布 5869 收藏 4
分类专栏： 我的CTF之路 我的逆向之路 ------2018i春秋网鼎杯 我的CTF进阶之路 文章标签： i春秋网鼎杯网络安全大赛Beijing题目writeup Beijing writeup reverse

我的CTF之路 同时被 3 个专栏收录
92 篇文章 5 订阅
订阅专栏

我的逆向之路
108 篇文章 10 订阅
订阅专栏

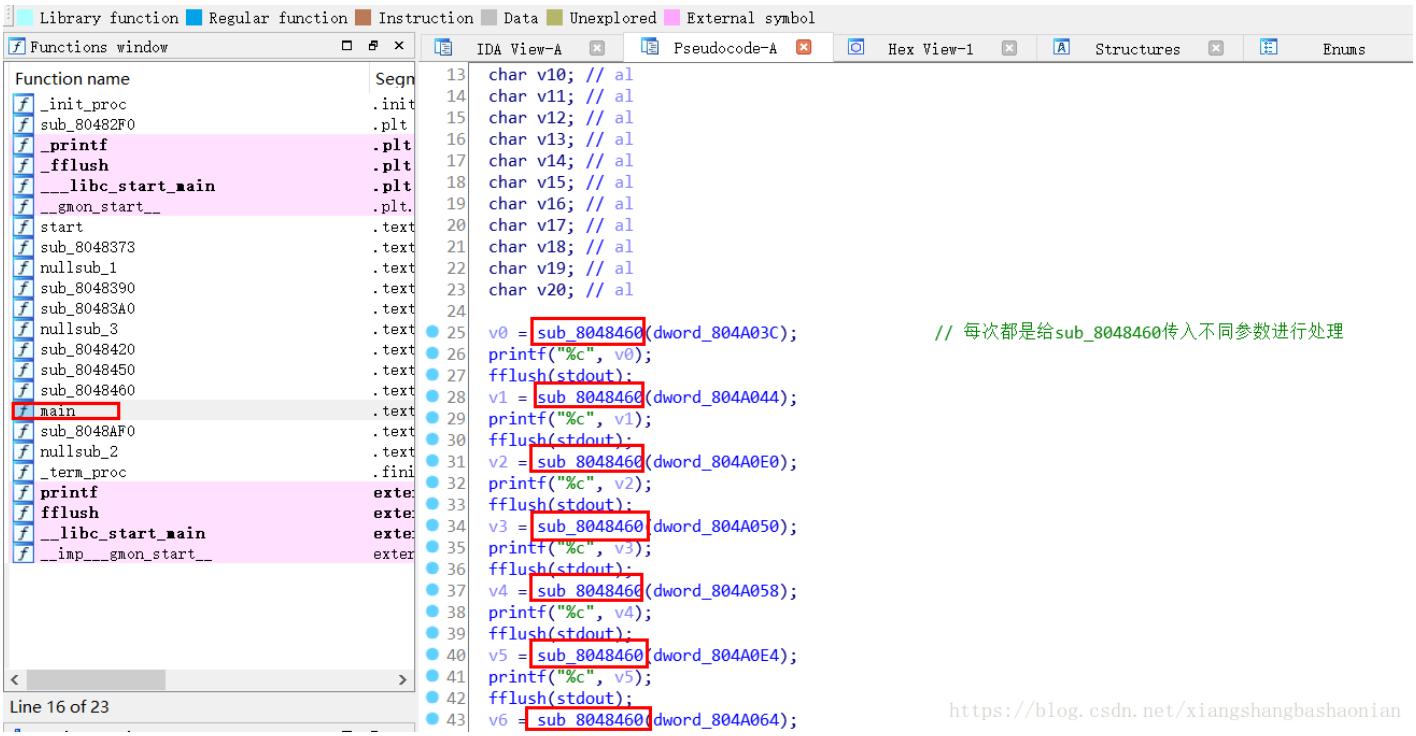
------2018i春秋网鼎杯
6 篇文章 0 订阅
订阅专栏

题目信息：

下载链接：https://pan.baidu.com/s/1jZ36MzyiHslLxkJ0LwsOXg 密码: 37k3

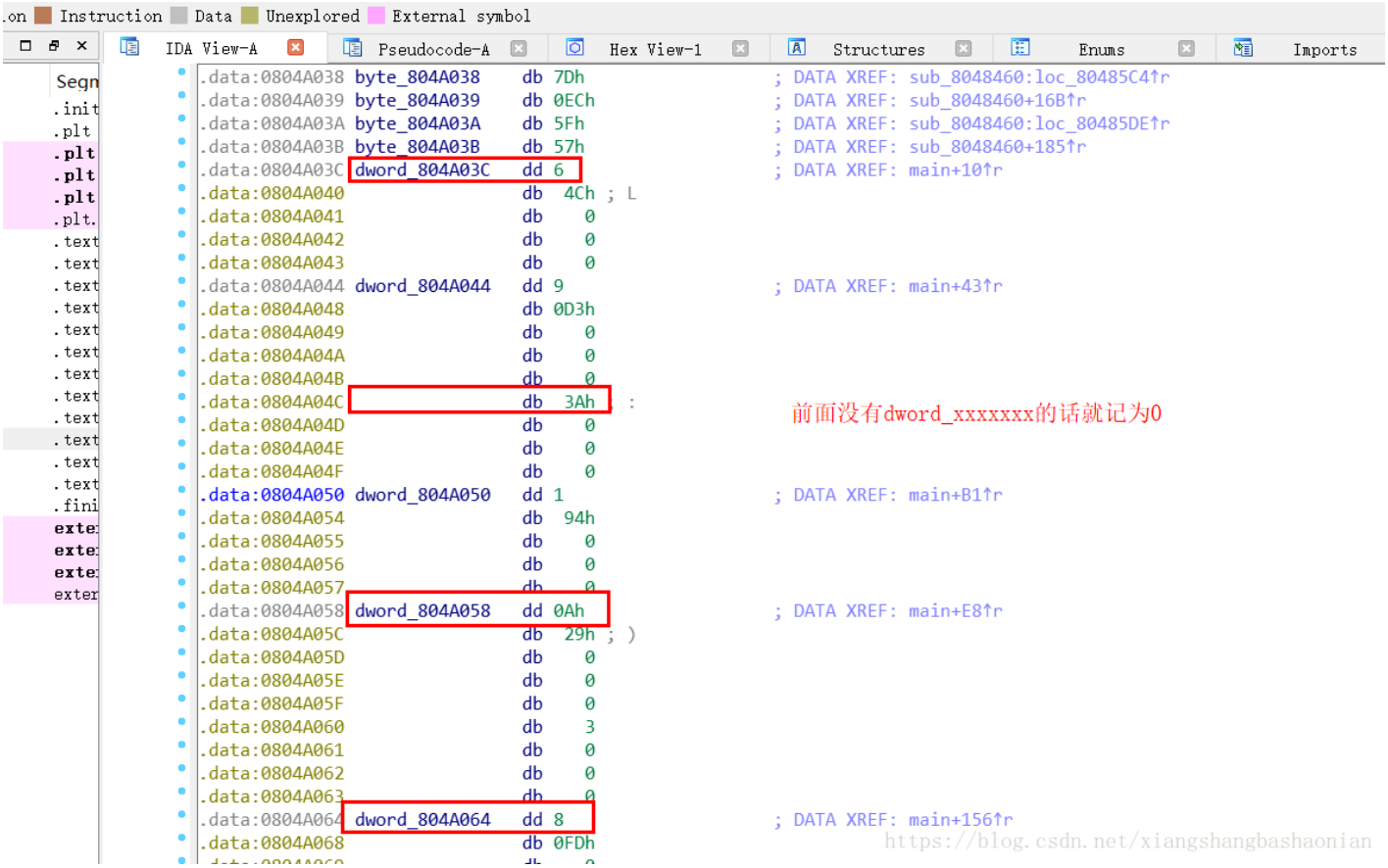
IDA载入 发现函数很少 也没有啥可以字符串 那就直分析

先看main函数 可以直接F5变成c伪代码



双击任意dword_804Axxx

接着向下翻到dword_段



最后得到顺序6 9 0 1 10 0 8 0 11 2 3 1 13 4 5 2 7 2 3 1 12

从v0到v20

```
.data:0804A020 byte_804A020    db 61h          把光标移到上边，按r就可以转换成字符  ; DATA XREF: sub_8048460:loc_804848C↑r
.data:0804A021 byte_804A021    db 4Ch                                              ; DATA XREF: sub_8048460+33↑r
.data:0804A022 byte_804A022    db 67h                                              ; DATA XREF: sub_8048460:loc_80484A6↑r
.data:0804A023 byte_804A023    db 59h                                              ; DATA XREF: sub_8048460+4D↑r
.data:0804A024 byte_804A024    db 69h                                              ; DATA XREF: sub_8048460:loc_80484C0↑r
.data:0804A025 byte_804A025    db 29h                                              ; DATA XREF: sub_8048460+67↑r
.data:0804A026 byte_804A026    db 6Eh                                              ; DATA XREF: sub_8048460:loc_80484DA↑r
.data:0804A027 byte_804A027    db 42h                                              ; DATA XREF: sub_8048460+81↑r
.data:0804A028 byte_804A028    db 62h                                              ; DATA XREF: sub_8048460:loc_80484F4↑r
.data:0804A029 byte_804A029    db 0Dh                                              ; DATA XREF: sub_8048460+9B↑r
.data:0804A02A byte_804A02A    db 65h                                              ; DATA XREF: sub_8048460:loc_804850E↑r
```



```
.data:0804A020 byte_804A020    db 'a'          0   ; DATA XREF: sub_8048460:loc_804848C↑r
.data:0804A021 byte_804A021    db 4Ch              ; DATA XREF: sub_8048460+33↑r
.data:0804A022 byte_804A022    db 'g'          1   ; DATA XREF: sub_8048460:loc_80484A6↑r
.data:0804A023 byte_804A023    db 59h              ; DATA XREF: sub_8048460+4D↑r
.data:0804A024 byte_804A024    db 'i'          2   ; DATA XREF: sub_8048460:loc_80484C0↑r
.data:0804A025 byte_804A025    db 29h              ; DATA XREF: sub_8048460+67↑r
.data:0804A026 byte_804A026    db 'n'          3   ; DATA XREF: sub_8048460:loc_80484DA↑r
.data:0804A027 byte_804A027    db 42h              ; DATA XREF: sub_8048460+81↑r
.data:0804A028 byte_804A028    db 'b'          4   ; DATA XREF: sub_8048460:loc_80484F4↑r
.data:0804A029 byte_804A029    db 0Dh              ; DATA XREF: sub_8048460+9B↑r
.data:0804A02A byte_804A02A    db 'e'          5   ; DATA XREF: sub_8048460:loc_804850E↑r
.data:0804A02B byte_804A02B    db 71h              ; DATA XREF: sub_8048460+B5↑r
.data:0804A02C byte_804A02C    db 'f'          6   ; DATA XREF: sub_8048460:loc_8048528↑r
.data:0804A02D byte_804A02D    db 34h              ; DATA XREF: sub_8048460+CF↑r
.data:0804A02E byte_804A02E    db 'j'          7   ; DATA XREF: sub_8048460:loc_8048542↑r
.data:0804A02F byte_804A02F    db 0C6h             ; DATA XREF: sub_8048460+E9↑r
.data:0804A030 byte_804A030    db 'm'          8   ; DATA XREF: sub_8048460:loc_804855C↑r
.data:0804A031 byte_804A031    db 8Ah              ; DATA XREF: sub_8048460+103↑r
.data:0804A032 byte_804A032    db 'l'          9   ; DATA XREF: sub_8048460:loc_8048576↑r
.data:0804A033 byte_804A033    db 7Fh              ; DATA XREF: sub_8048460+11D↑r
.data:0804A034 byte_804A034    db '{'          10  ; DATA XREF: sub_8048460:loc_8048590↑r
.data:0804A035 byte_804A035    db 0AEh             ; DATA XREF: sub_8048460+137↑r
.data:0804A036 byte_804A036    db 'z'          11  ; DATA XREF: sub_8048460:loc_80485AA↑r
.data:0804A037 byte_804A037    db 92h              ; DATA XREF: sub_8048460+151↑r
.data:0804A038 byte_804A038    db '}'          12  ; DATA XREF: sub_8048460:loc_80485C4↑r
.data:0804A039 byte_804A039    db 0ECh             ; DATA XREF: sub_8048460+16B↑r
.data:0804A03A byte_804A03A    db '_'          13  ; DATA XREF: sub_8048460:loc_80485DE↑r
.data:0804A03B byte_804A03B    db 57h              ; DATA XREF: sub_8048460+185↑r
.data:0804A03C dword_804A03C   dd 6                ; DATA XREF: main+101↑r
```


得到对应关系

0 1 2 3 4 5 6 7 8 9 10 11 12 13

a g i n b e f j m l { z } _

然后就是按照上边得到的顺序进行一一对应

```
1  6 9 0 1 10 0 8 0 11 2 3 1 13 4 5 2 7 2 3 1 12
2  f l a g {  a m a z  i n g _  b e i j i n g }
3
4  0 1 2 3 4 5 6 7 8 9 10 11 12 13
5  a g i n b e f j m l {  z  }  _
```


最后得到**flag{amazing_beijing}**