

buu刷题 [ACTF新生赛2020]rome1

原创

元元努力向上 于 2022-01-11 23:17:30 发布 34 收藏

文章标签: [经验分享](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: https://blog.csdn.net/m0_56194555/article/details/122437461

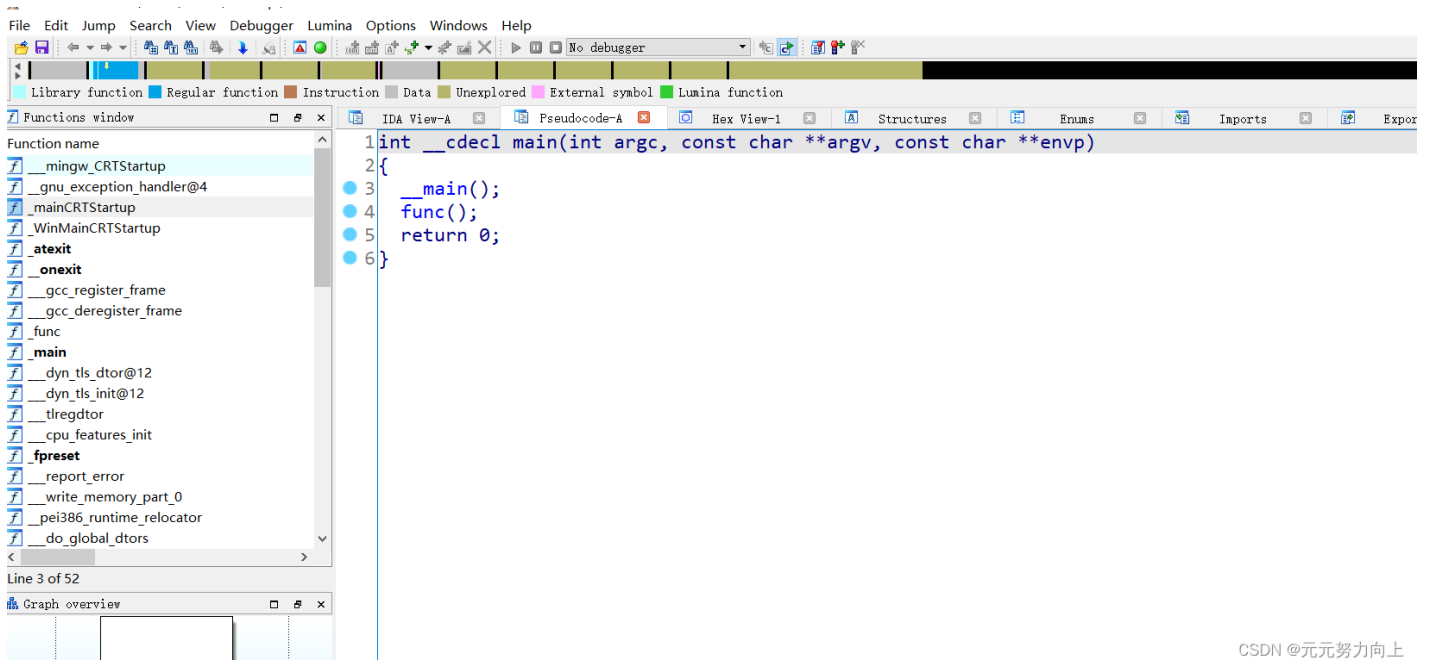
版权

先查壳:



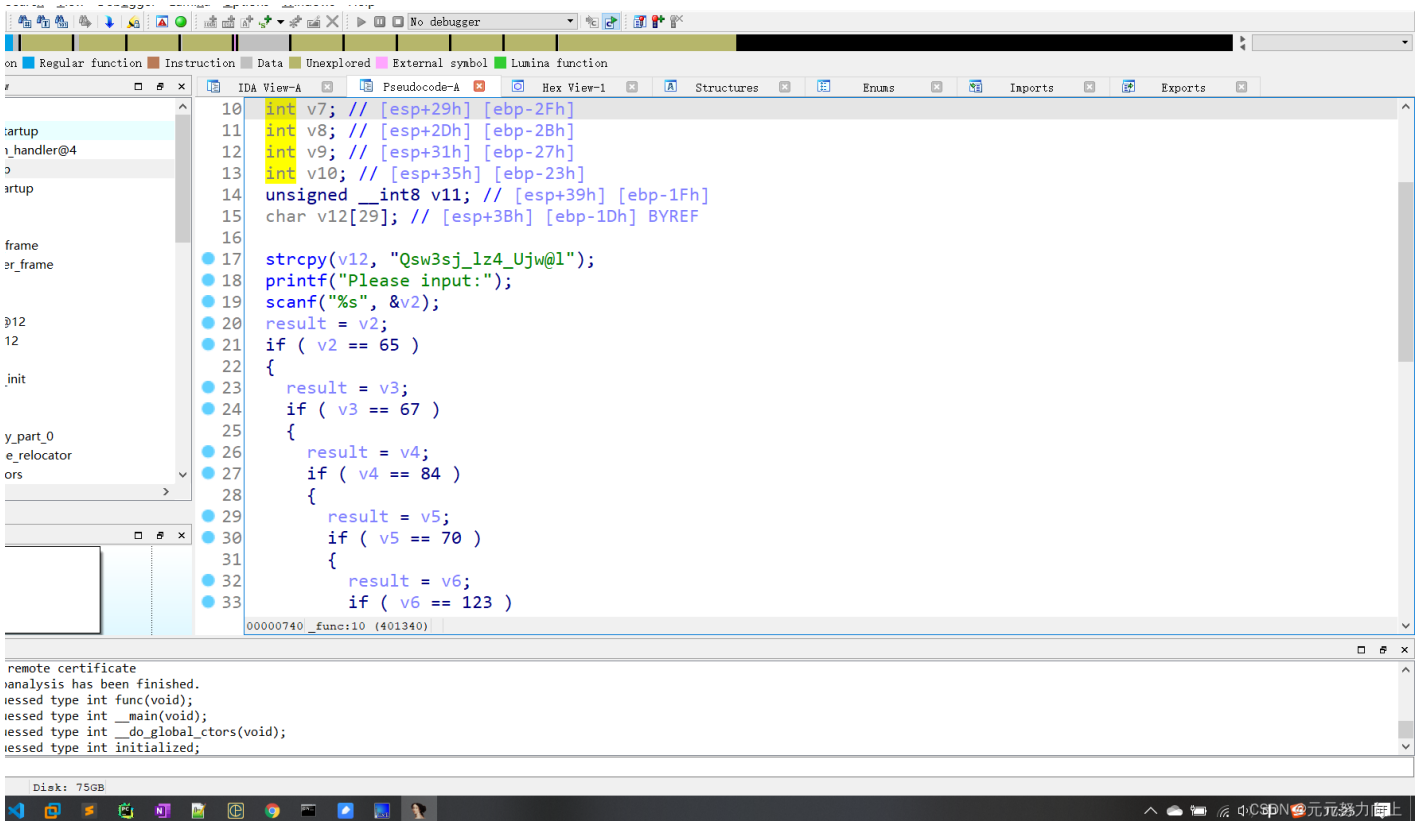
无壳 c++编译 32位

用ida打开



CSDN @元元努力向上

主函数没啥东西, 进入func()看看:



发现一串字符，然后把ASCII变成char看一下：

```

19  scanf("%s", &v2);
20  result = v2;
21  if ( v2 == 'A' )
22  {
23      result = v3;
24      if ( v3 == 'C' )
25      {
26          result = v4;
27          if ( v4 == 'T' )
28          {
29              result = v5;
30              if ( v5 == 'F' )
31              {
32                  result = v6;
33                  if ( v6 == '{' )
34                  {
35                      result = v11;
36                      if ( v11 == '}]' )
37                      {
38                          v1[0] = v7;
39                          v1[11] = v8.

```

CSDN @元元努力向上

这里肯定就是加密函数了：

```

int func()
{
    int result; // eax
    int v1[4]; // [esp+14h] [ebp-44h]

```

```

unsigned __int8 v2; // [esp+24h] [ebp-34h] BYREF
unsigned __int8 v3; // [esp+25h] [ebp-33h]
unsigned __int8 v4; // [esp+26h] [ebp-32h]
unsigned __int8 v5; // [esp+27h] [ebp-31h]
unsigned __int8 v6; // [esp+28h] [ebp-30h]
int v7; // [esp+29h] [ebp-2Fh]
int v8; // [esp+2Dh] [ebp-2Bh]
int v9; // [esp+31h] [ebp-27h]
int v10; // [esp+35h] [ebp-23h]
unsigned __int8 v11; // [esp+39h] [ebp-1Fh]
char v12[29]; // [esp+3Bh] [ebp-1Dh] BYREF

strcpy(v12, "Qsw3sj_lz4_Ujw@1");
printf("Please input:");
scanf("%s", &v2);
result = v2;
if ( v2 == 'A' )
{
    result = v3;
    if ( v3 == 'C' )
    {
        result = v4;
        if ( v4 == 'T' )
        {
            result = v5;
            if ( v5 == 'F' )
            {
                result = v6;
                if ( v6 == '{' )
                {
                    result = v11;
                    if ( v11 == '}' )
                    {
                        v1[0] = v7;
                        v1[1] = v8;
                        v1[2] = v9;
                        v1[3] = v10;
                        *&v12[17] = 0;
                        while ( *&v12[17] <= 15 )
                        {
                            if ( *(v1 + *&v12[17]) > 64 && *(v1 + *&v12[17]) <= 90 ) //如果字符在64-90之间 那么字符= (字
                                *(v1 + *&v12[17]) = *(v1 + *&v12[17]) - 51) % 26 + 65;
                            if ( *(v1 + *&v12[17]) > 96 && *(v1 + *&v12[17]) <= 122 //如果字符在96-122之间 那么字符= (字符
                                *(v1 + *&v12[17]) = *(v1 + *&v12[17]) - 79) % 26 + 97;
                            ++*&v12[17]; //如果字符等于它本身那么 就返回自己
                        }
                        *&v12[17] = 0;
                        while ( *&v12[17] <= 15 )
                        {
                            result = v12[*&v12[17]];
                            if ( *(v1 + *&v12[17]) != result )
                                return result;
                            ++*&v12[17];
                        }
                        result = printf("You are correct!");
                    }
                }
            }
        }
    }
}
}
}
}

```

```
    }  
  }  
  return result;  
}
```

脚本:

```
a= 'ACTF{'  
enc='Qsw3sj_lz4_Ujw@1'  
s=''  
for i in range(len(enc)):  
    for n in range(128):  
        j=n  
        if j >64 and j<=90:  
            j=(j-51)%26+65  
        if j>96 and j<=122:  
            j= (j-79)%26 +97  
        if j==ord(enc[i]):  
            a=a+chr(n)  
print(a+'}')  
  
ACTF{Cae3ar_th4_Gre@t}
```