

# 【逆向】i春秋入门实战——crack\_me

原创

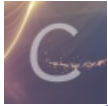
Scorpionbelle 于 2020-07-24 13:56:08 发布 291 收藏

分类专栏: [ctf](#) 文章标签: [反编译](#) [java](#) [安全](#)

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/qq\\_30303857/article/details/107558771](https://blog.csdn.net/qq_30303857/article/details/107558771)

版权



[ctf](#) 专栏收录该内容

1 篇文章 0 订阅

订阅专栏

作为小白, 刚开始看题目确实非常懵, 在网上一步步搜索, 终于摸索到了flag, 在此做些许记录。但是guid还是没有搜索到

1. (1.5分) 本实验中, 逆向分析C#程序的程序为 ()

- Reflector
- xspy
- ImportREC
- LoadPE

2. (1.5分) 本实验中, 程序的全局统一标识符 (GUID) 为 ()

- 9f1c5a2b-1252-4695-8673-93a811cff353
- 9f1c5a2b-1252-4695-8673-93a811cff354
- 9f1c5a2b-1252-4695-8673-93a811cff356
- 9f1c5a2b-1252-4695-8673-93a811cff355

3. (2分) 本实验中逆向程序flag字符串为 \_\_.

TSCTF{A\_S1MPL3\_C#\_CR4CKME}

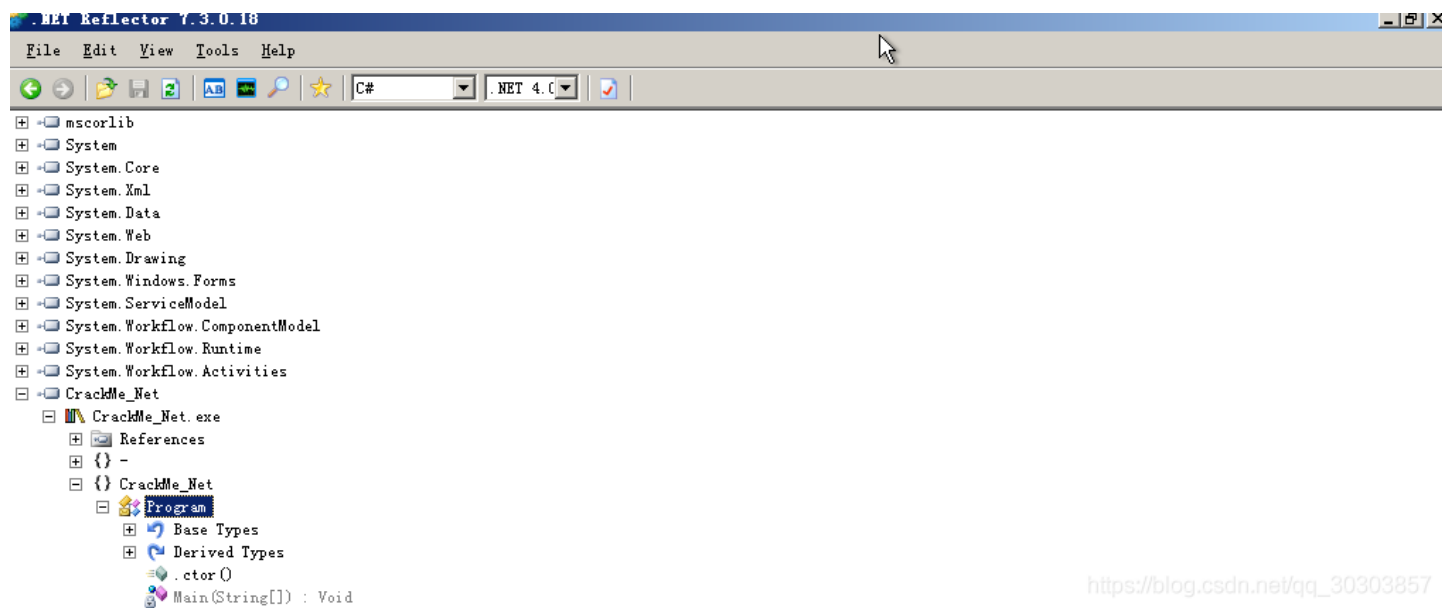
[https://blog.csdn.net/qq\\_30303857](https://blog.csdn.net/qq_30303857)

## 一.c#逆向工具

### 1.reflector

发现实验工具有写这个工具, 显而易见  
reflector用法详解参考这篇文章

打开reflector 并打开crackme.exe文件,界面如下



[https://blog.csdn.net/qq\\_30303857](https://blog.csdn.net/qq_30303857)

## 二.guid

### 第一次尝试

刚开始看到该篇博客反编译.netz压缩过的exe文件

但好像这个实验中没法用这种方法（还是我操作不对？

### 第二次尝试

发现题目中有9f等一串，只是后面数字些许不同，于是在reflector里面搜索，但是也没有搜索到此类字符串。

privateimplemtationdetails

但guid也没有出现答案中的一个

（dream一个大佬解答

## 三.flag

打开main函数，直接被反编译出来相应代码

```
Program.Main(String[]) : Void x
private static void Main(string[] args)
{
    int[] numArray = new int[] {
        0x30, 0x2c, 9, 0x18, 0x1f, 0x49, 0x2b, 0x20, 0x30, 12, 30, 50, 0x29, 0x29, 0x2b, 0x2c,
        15, 0x1b, 9, 0x3d, 20, 0x2b, 0x25, 0x30, 0x3b, 30
    };
    int[] numArray2 = new int[] {
        0x24, 0x27, 0x3a, 60, 0x27, 50, 0x16, 0x3f, 0x23, 0x25, 0x2f, 30, 0x23, 10, 0x34, 0x17,
        20, 0x44, 0x3a, 0x15, 0x20, 0x18, 0x26, 0x1d, 10, 0x5f
    };
    Console.WriteLine("Please input your flag!");
    string str = Console.ReadLine();
    if (str.Length != 0x1a)
    {
        Console.WriteLine("Wrong flag length!");
    }
    else
    {
        for (int i = 0; i <= 0x19; i++)
        {
            if (str[i] != (numArray[i] + numArray2[i]))
            {
                Console.WriteLine("Wrong flag!");
                return;
            }
        }
        Console.WriteLine("Right flag!");
    }
}
```

[https://blog.csdn.net/qq\\_30303857](https://blog.csdn.net/qq_30303857)

先写代码算法得到的十进制数字

```
public static void main(String[] args){
    int[] numsArray = new int[] {
        0x30,0x2c,9,0x18,0x1f,0x49,0x2b,0x20,0x30,12,30,50,0x29,0x29,0x2b,0x2c,15,0x1b,9,0x3d,20,0x2b,0x25,0x30,0x3b,30
    };
    int[] numsArray2 = new int[] {
        0x24,0x27,0x3a,60,0x27,50,0x16,0x3f,0x23,0x25,0x2f,30,0x23,10,0x34,0x17,20,0x44,0x3a,0x15,0x20,0x18,0x26,0x1d,10,0x5f
    };

    for(int i = 0;i<=0x19;i++)
    {
        System.out.print(numsArray2[i]+numsArray[i]+" ");
    }
}
```

最后将得到的值转码为字符串，即为flag

ASCII转换到 ASCII (例: a b c)

TSCTF {A\_S1MPL3\_C#\_CR4CKME}

添加空格

删除空格

将空白字符转换

十六进制转换到16进制(例:0x61或61或61/62)  删除 0x

0x540x530x430x540x460x7b0x410x5f0x530x310x4d0x500x4c0x330x5f0x430x230x5f0x430x520x340x430x4b0x4d0x450x7d

十进制转换到 10进制 (例: 97 98 99)

848367847012365958349778076519567359567825267757769125

二进制转换到 2进制(例:01100001 01100010 01100011)

0101010001010011010000110101010001000110011110110  
1000001010111110101001100110001010011010101000001  
0011000011001101011111010000110010001101011111010  
0001101010010001101000100001101001011010011010100

[https://blog.csdn.net/qq\\_36363857](https://blog.csdn.net/qq_36363857)