


# BUUCTF Reverse Java逆向解密 WriteUp

原创

PlumpBoy  于 2021-09-11 16:12:57 发布  21  收藏

分类专栏: [BUUCTF 逆向题解](#) 文章标签: [java](#) [安全](#) [系统安全](#)

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/weixin\\_45723661/article/details/120227935](https://blog.csdn.net/weixin_45723661/article/details/120227935)

版权



[BUUCTF 逆向题解 专栏收录该内容](#)

18 篇文章 1 订阅

订阅专栏

## java逆向解密-WP

扔进jd-gui中进行反编译, 得到源代码

```

import java.io.PrintStream;
import java.util.ArrayList;
import java.util.Scanner;

public class Reverse
{
    public static void main(String[] args)
    {
        Scanner s = new Scanner(System.in);
        System.out.println("Please input the flag ");
        String str = s.next();
        System.out.println("Your input is ");
        System.out.println(str);
        char[] stringArr = str.toCharArray();
        Encrypt(stringArr);
    }

    public static void Encrypt(char[] arr)
    {
        ArrayList<Integer> Resultlist = new ArrayList();
        for (int i = 0; i < arr.length; i++)
        {
            int result = arr[i] + '@' ^ 0x20;
            Resultlist.add(Integer.valueOf(result));
        }
        int[] KEY = { 180, 136, 137, 147, 191, 137, 147, 191, 148, 136, 133, 191, 134, 140, 129, 135, 191, 65 };
        ArrayList<Integer> KEYList = new ArrayList();
        for (int j = 0; j < KEY.length; j++) {
            KEYList.add(Integer.valueOf(KEY[j]));
        }
        System.out.println("Result:");
        if (Resultlist.equals(KEYList)) {
            System.out.println("Congratulations");
        } else {
            System.err.println("Error");
        }
    }
}

```

上面代码很好理解，就是输入的字符串先加上0x40再与0x20进行异或，最终与KEY数组进行比较，一样就正确。

写出解密脚本

```

a = [180, 136, 137, 147, 191, 137, 147, 191, 148, 136, 133, 191, 134, 140, 129, 135, 191, 65]

flag = ""

for i in range(len(a)):
    flag += chr((a[i] ^ 0x20) - 0x40)
print(flag)

```