

攻防世界-MISC:base64stego

原创

LY613313 于 2021-11-25 17:48:14 发布 197 收藏

分类专栏: [攻防世界](#) 文章标签: [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/m0_56161093/article/details/121524866

版权



[攻防世界 专栏收录该内容](#)

21 篇文章 1 订阅

订阅专栏

这是攻防世界新手练习区的第十一题, 题目如下:

base64stego

👍 214 最佳Writeup由CTFshow • zEr0_0提供

难度系数: ★★★★★ 5.0

题目来源: [olympicCTF](#)

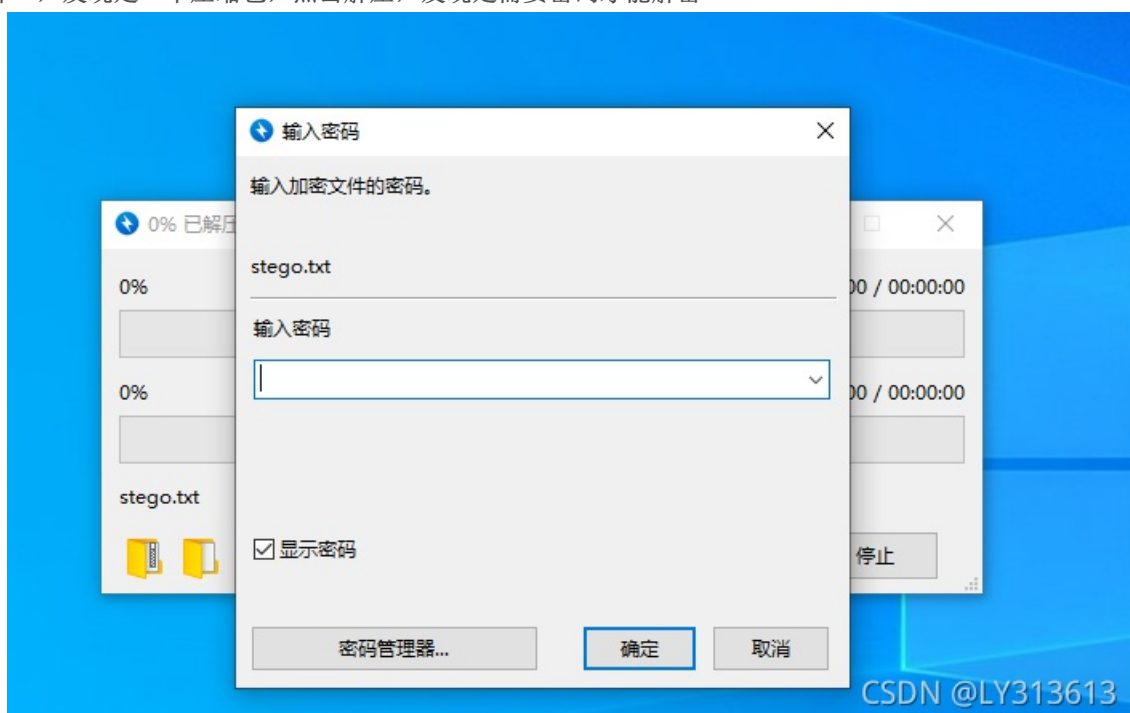
题目描述: 菜狗经过几天的学习, 终于发现了如来十三掌最后一步的精髓

题目场景: 暂无

题目附件: [附件1](#)

CSDN @LY313613

点击下载附件一, 发现是一个压缩包, 点击解压, 发现是需要密码才能解密



先用U10editor打开这个压缩包，这里需要知道zip压缩包的组成部分，包括压缩源文件数据区、压缩源文件目录区、压缩源文件目录结束区，具体组成如下：

| 压缩源文件数据区(组成/长度) | |
|--------------------------|---------------------|
| 文件头标记 | 4 bytes(0x04034b50) |
| 解压文件所需的 <u>pkware</u> 版本 | 2 bytes |
| 全局方式位标记(判断真伪加密) | 2 bytes |
| 压缩方式 | 2 bytes |
| 最后修改文件时间 | 2 bytes |
| 最后修改文件日期 | 2 bytes |
| CRC-32 校验 | 4 bytes |
| 压缩后尺寸 | 4 bytes |
| 未压缩尺寸 | 4 bytes |
| 文件名长度 | 2 bytes |
| 扩展记录长度 | 2 bytes |
| 文件名 | 不定长度 |
| 扩展字段 | 不定长度 |

CSDN @LY313613

| 压缩源文件目录区(组成/长度) | |
|-------------------------|---------------------|
| 目录中文件的文件头标记 | 4 bytes(0x02014b50) |
| 压缩使用的 <u>pkware</u> 版本 | 2 bytes |
| 解压缩所需的 <u>pkware</u> 版本 | 2 bytes |
| 全局方式位标记(判断真伪加密) | 2 bytes |
| 压缩方式 | 2 bytes |
| 最后修改文件时间 | 2 bytes |
| 最后修改文件日期 | 2 bytes |
| CRC-32 校验 | 4 bytes |
| 压缩后尺寸 | 4 bytes |
| 未压缩尺寸 | 4 bytes |
| 文件名长度 | 2 bytes |
| 扩张字段长度 | 2 bytes |
| 文件注释长度 | 2 bytes |
| 磁盘开始号 | 2 bytes |
| 内部文件属性 | 2 bytes |
| 外部文件属性 | 4 bytes |
| 局部文件偏移量 | 4 bytes |
| 文件名 | 不定长度 |
| 扩展字段 | 不定长度 |
| 文件注释 | 不定长度 |

CSDN @LY313613

压缩源文件目录结束标志(组成/长度)

| | |
|---------------|---------------------|
| 目录结束标记 | 4 bytes(0x06054b50) |
| 当前磁盘编号 | 2 bytes |
| 目录区开始磁盘编号 | 2 bytes |
| 本磁盘上记录总数 | 2 bytes |
| 目录区中记录总数 | 2 bytes |
| 目录区尺寸大小 | 4 bytes |
| 目录区对第一张磁盘的偏移量 | 4 bytes |
| ZIP 文件注释长度 | 2 bytes |
| ZIP 文件注释 | 不定长度 |

CSDN @LY313613

压缩包是否为伪加密的判断

1、无加密:

- (1) 压缩源文件数据区的全局加密标志应当为00 00
- (2) 且压缩源文件目录区的全局方式位标记应当为00 00

2、伪加密:

- (1) 压缩源文件数据区的全局加密标志应当为00 00
- (2) 且压缩源文件目录区的全局方式位标记应当为**09 00**

3、真加密

- (1) 压缩源文件数据区的全局加密标志应当为09 00
- (2) 且压缩源文件目录区的全局方式位标记应当为09 00

现在看一下这道题，该压缩包的压缩源文件的数据区如下

010 Editor - C:\Users\ly313\Desktop\2eb7ceaf5ab49f7acb33de2e7eed74a.zip

File Edit Search View Format Scripts Templates Debug Tools Window Help

Startup a2eb7ceaf5ab49f7acb33de2e7eed74a.zip x

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F | 0123456789ABCDEF |
|--------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|--------------------|
| 0000h: | 50 | 4B | 03 | 04 | 14 | 03 | 00 | 00 | 08 | 00 | 68 | BF | 9B | 48 | FE | 32 | PK.....hZ>Hb2 |
| 0010h: | 7D | 4B | E9 | 0D | 00 | 00 | B5 | 1B | 00 | 00 | 09 | 00 | 00 | 00 | 73 | 74 | }Ké...μ.....st |
| 0020h: | 65 | 67 | 6F | 2E | 74 | 78 | 74 | 7D | 59 | C9 | 76 | E2 | 48 | 10 | BC | EB | ego.txt}YÉvâH.¼è |
| 0030h: | 57 | E6 | 22 | 24 | 33 | CF | 1C | 38 | 8C | 68 | 83 | C4 | 18 | 7A | 00 | A3 | Wæ"\$3I.8ChfĀ.z.£ |
| 0040h: | ED | A6 | C5 | 0F | 01 | 12 | 30 | 0D | 08 | C4 | D7 | 4F | 44 | 56 | 09 | 68 | i!Ā...0..A×ODV.h |
| 0050h: | 77 | BF | 39 | F8 | B1 | A9 | AA | B2 | 32 | 23 | 22 | 17 | 2F | ED | 79 | 19 | w;9ø±@ª²#"./iy. |
| 0060h: | 5B | C3 | 73 | 6A | E5 | 4D | 14 | 3A | FB | CF | 85 | 73 | C8 | 1A | C7 | 4C | [Āsjãm.:ûĪ...sĒ.ĈL |
| 0070h: | 46 | CB | 55 | 14 | 8E | 4D | 6F | 34 | 3C | C7 | 03 | E7 | 16 | 59 | 65 | 99 | FĒU.ŽMo4<Ç.ç.Ye™ |
| 0080h: | 56 | D3 | D2 | 1B | F5 | 2A | C3 | 73 | F3 | 26 | 09 | E7 | 87 | B4 | CA | 56 | VÓ0.õ*Āsó&.ç†'ĒV |
| 0090h: | C9 | A8 | DC | C6 | 23 | FF | EC | 8D | 3A | 65 | 66 | 4F | 8B | D8 | F2 | 6F | É"Ū#yĪ.:ef0<ðòo |
| 00A0h: | DE | A8 | 3C | 7B | EE | B4 | 13 | 59 | AB | 55 | B4 | 70 | EC | 28 | DC | AE | þ"<{i'.Y«U`pì(Ū@ |
| 00B0h: | F2 | 51 | 51 | E4 | 03 | E7 | 9C | 36 | 4E | 9D | 56 | 7E | DF | 78 | 1F | 38 | òQQä.çø6N.V~Bx.8 |

Template Results - ZIP.bt

| Name | Value | Start | Size | Color |
|---------------------------------|-----------|-------|------|---------|
| > struct ZIPFILERECD record | stego.txt | 0h | E10h | Fg: Bg: |
| > struct ZIPDIRENTRY dirEntry | stego.txt | E10h | 5Bh | Fg: Bg: |
| > struct ZIPENDLOCATOR endLo... | | E6Bh | 16h | |

CSDN @LY313613

- 50 4B 03 04: 文件头标记 (0x04034B50)
- 14 03: 解压文件所需 pkware 版本
- 00 00: 全局方式位标记 (判断真伪加密的重要标志)**
- 08 00: 压缩方式
- 68 BF: 最后修改文件时间
- 9B 48: 最后修改文件日期
- FE 32 7D 4B: CRC-32校验
- E9 0D 00 00: 压缩后尺寸
- B5 1B 00 00: 未压缩尺寸
- 09 00: 文件名长度
- 00 00: 扩展记录长度

该压缩包的压缩源文件的目录区如下:

```

0h: 36 7A C7 00 3A B1 B6 F5 2F AD E8 CC FC DB F8 0F 6zÇ.:±¶ö/-èïüŦ.
0h: 50 4B 01 02 3F 03 14 03 09 00 08 00 68 BF 9B 48 PK..?.....h¿>H
0h: FE 32 7D 4B E9 0D 00 00 B5 1B 00 00 09 00 24 00 p2}Ké...µ....$.
0h: 00 00 00 00 00 00 20 80 ED 81 00 00 00 00 73 74 ..... €í.....st
0h: 65 67 6F 2E 74 78 74 0A 00 20 00 00 00 00 00 01 ego.txt.....
0h: 00 18 00 80 0B 49 BF 9D A0 D1 01 80 A7 42 38 B7 ...€.I¿. Ñ.€$B8.
  
```

- 50 4B 01 02: 目录中文件文件头标记(0x02014B50)
- 3F 03: 压缩使用的 pkware 版本
- 14 03: 解压文件所需 pkware 版本
- 09 00: 全局方式位标记 (数据区的加密标志为00 00, 所以判断这是伪加密, 将09 00 改为00 00即可)
- 08 00: 压缩方式
- 68 BF: 最后修改文件时间
- 9B 48: 最后修改文件日期
- FE 32 7D 4B: CRC-32校验 (1480B516)
- E9 0D 00 00: 压缩后尺寸 (25)
- B5 1B 00 00: 未压缩尺寸 (23)
- 09 00: 文件名长度
- 24 00: 扩展字段长度
- 00 00: 文件注释长度
- 00 00: 磁盘开始号
- 00 00: 内部文件属性
- 20 80 ED 81: 外部文件属性
- 00 00 00 00: 局部头部偏移量

该压缩包源文件目录结束标志如下:

```

Startup a2eb7ceaf5ab49f7acb33de2e7eed74a.zip x
0E50h: 00 18 00 80 0B 49 BF 9D A0 D1 01 80 A7 42 38 B7 ...€.I¿. Ñ.€$B8.
0E60h: 2F D4 01 00 11 AA 37 B7 2F D4 01 50 4B 05 06 00 /Ŧ...ª7·/Ŧ.PK...
0E70h: 00 00 00 01 00 01 00 5B 00 00 00 10 0E 00 00 00 ..... [.....
0E80h: 00
  
```

Template Results - ZIP.bt

| Name | Value | Start | Size | Color | Comment |
|------|-------|-------|------|-------|---------|
|------|-------|-------|------|-------|---------|

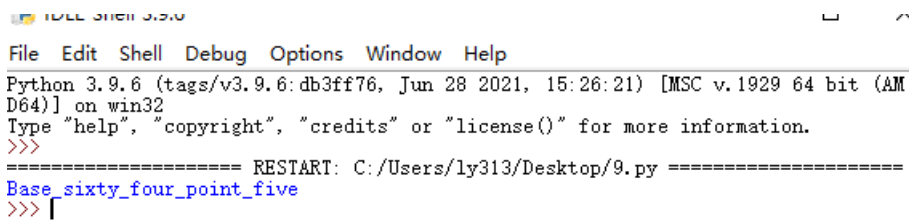
CSDN @LY313613

50 4B 05 06: 目录结束的标记 (0x06054B50)
00 00: 当前磁盘编号
00 00: 目录区开始磁盘编号
01 00: 本磁盘上纪录总数
01 00: 目录区中纪录总数
5B 00 00 00: 目录区尺寸大小
10 0E 00 00: 目录区对第一张磁盘的偏移量
00 00: ZIP 文件注释长度

所以，我们只要将该压缩包的压缩源文件的全局方式位标记由09 00 改为00 00即可解压该文件，通过解压文件我们可以得到一个文本文档。打开后可以看到是一些base64字符，我们写个脚本跑一下

```
import base64
bin_str=''
b64chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/'
with open('stego.txt','r') as f: # 这里要改成你的文件路径
    for line in f.readlines():
        s64="".join(line.split())
        r64="".join(str(base64.b64encode(base64.b64decode(s64)), 'utf-8').split())
        offset=abs(b64chars.index(s64.replace('=','')[-1])-b64chars.index(r64.replace('=','')[-1]))
        equal=line.count('=')
        if equal:
            bin_str += bin(offset)[2:].zfill(equal * 2)
print(''.join([chr(int(bin_str[i:i + 8], 2)) for i in range(0,len(bin_str),8)]))
```

代码运行结果如下：



```
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/ly313/Desktop/9.py =====
Base_sixty_four_point_five
>>> [
```

所以这道题的flag如下：

```
flag{Base_sixty_four_point_five}
```

以上就是我对这道题的解法。因本人菜鸡一只，如果有什么不对的地方，实属正常。还请各位大佬予以指正，谢谢！