

CTF题目整理

原创

[起个名字被占了](#) 于 2015-08-23 10:56:00 发布 10418 收藏 13

分类专栏: [CTF](#) 文章标签: [CTF](#)

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/u013648937/article/details/47903993>

版权



[CTF 专栏收录该内容](#)

1 篇文章 0 订阅

订阅专栏

最近在整理一些CTF题目, 遇到很多很有意思的东西, 记录下来

1. 0x01: 栅栏加密

栅栏加密法是一种比较简单快捷的加密方法。栅栏加密法就是把要被加密的文件按照一上一下的写法写出来, 再把第二行的文字排列到例:

```
加密information
分行
i f r a i n
n o m t o
合并
ifrainnomto
完成~
```

题目给出了一串字符:

```
ifrainnomto
```

没有规律, 然后先统计一下字符多少

```
s = 'ifrainnomto'
a = []
for i in range(6):
    a.append(s[i::6])
print ''.join(a)
```

一共是12个字符(有一个空字符), 把这些字符整理成矩阵: 6*2

```
s = 'ifrainnomto'
a = []
for i in range(6):
    a.append(s[i::6])
print ''.join(a)
print len(s)

#information
```

1. 0x02:MD5加密

小明准备把加密好的字符串发给自己的同学，但是粗心的他在编辑的时候不小心按了一下键盘，改变了字符串，你能帮他还原吗

给出字符串：

```
21232f297a57a5a7438914a0e4a801fc3
```

题目很简单，熟悉的同学可能猜到这是一个MD5加密的字符串，但是怎么找出来混淆字符呢？开始我也没啥思路，不过后来看到有人说md5加密字符是0-f，所以，就瞬间可以解出来了。把字符串中的‘i’去掉就行了，还原之后的密文

```
21232f297a57a5a743894a0e4a801fc3
```

```
cmd5之后:admin
```

0x03:暂时就先写这么多吧，稍后更新