WP 4 i春秋_internetwache-ctf-2016

原创

segOt 于 2017-04-16 22:12:10 发布 2040 收藏

分类专栏： CTF writeup 文章标签： wp CTF

The hidden message

> 辨识八进制，ascii转换，base64

Description: My friend really can’t remember passwords. So he uses some kind of obfuscation. Can you restore the plaintext?

将附件下载之后，发现内容如下

```
0000000 126 062 126 163 142 103 102 153 142 062 065 154 111 121 157 1130000020 122 155 170 150 132 172 157 147 1
23 126 144 067 124 152 102 1460000040 115 107 065 154 130 062 116 150 142 154 071 172 144 104 102 1670000060 130
 063 153 167 144 130 060 113 0120000071
```

本来的文件是一行下来的，稍微找一下规律就能发现，文件应该是如下形式

```
0000000 126 062 126 163 142 103 102 153 142 062 065 154 111 121 157 113
0000020 122 155 170 150 132 172 157 147 123 126 144 067 124 152 102 146
0000040 115 107 065 154 130 062 116 150 142 154 071 172 144 104 102 167
0000060 130 063 153 167 144 130 060 113 012
0000071
```

左面第一列是计数，后面的列是数据，并且一行16个数据，所以从第二列起始的计数是`0000020`，所以看出是**八进制**，所以推测后面的数据也是八进制，考虑将数据转化成字符形式。Python代码如下（先将行号删除）：

```
#!/usr/bin/python
with open("README.txt") as f:
    s = f.read()

s = s.split()
l = ''

for i in s:
    n = int(i)
    sum = 0
    mi = 0
    while n != 0:
        n1 = n % 10
        sum += n1 * 8 ** mi
        n = int(n / 10)
        mi += 1
    l += chr(sum)

print l
```

得到字符串

```
V2VsbCBkb25lIQoKRmxhZzogSVd7TjBfMG5lX2Nhbl9zdDBwX3kwdX0K
```

进行BASE64解码，得到

```
Well done!

Flag: IW{N0_0ne_can_st0p_y0u}
```

得到FLAG~

Quick run

> base64，二维码

Description: Someone sent me a file with white and black rectangles. I don't know how to read it. Can you help me?

这道题目的文件内容显然是base64编码，直接对其解码，发现是一串二维码。代码如下（不能将内容直接解码，否则只会解码第一个base64编码）：

```
#!/usr/bin/python

import base64

with open('README.txt', 'r') as f:
    s = f.read().split()
    base64Str = ''

    for i in s:
        base64Str += i
        if len(i) < 76:
           print  base64.b64decode(base64Str)
           base64Str = ''
```

大佬都自动解码，小弱只能一个一个扫描，扫描最后的结果是

```
Flagis:IW{QR_C0DES_RUL3}
```

---

rock-with-the-wired-shark

> Wireshark数据包分析，HTTP基本认证

Description: Sniffing traffic is fun. I saw a wired shark. Isn't that strange?

题目给了一个抓包的结果，使用Wireshark分析，可以比较容易地看到传输了一个.zip的压缩包，如下图：

```
GET /flag.zip HTTP/1.1
Host: 192.168.1.41:8080
Connection: keep-alive
Authorization: Basic ZmxhZzphenVsY3JlbWE=
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/46.0.2490.80 Safari/537.36
DNT: 1
Referer: http://192.168.1.41:8080/
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8,ht;q=0.6

HTTP/1.0 200 OK
Server: servefile/0.4.4 Python/2.7.10
Date: Fri, 13 Nov 2015 18:41:09 GMT
Content-Length: 222
Connection: close
Last-Modified: Fri, 13 Nov 2015 18:41:09 GMT
Content-Type: application/octet-stream
Content-Disposition: attachment; filename="flag.zip"
Content-Transfer-Encoding: binary

PK..
.....x.mG....(...........flag.txtUT     ...-FV.-FVux.............;......q.........9.....H.!...     >B.....+:PK......(.......PK....
.....x.mG....(.........................flag.txtUT....-FVux.............PK.........N...z.....
```


将传输内容导出，发现解压文件需要密码。仔细观察HTTP头部发现，使用了HTTP基本认证，猜测密码就在其中。

HTTP BASIC AUTHORIZATION其实在传输过程中内容没有加密，只是进行了BASE64编码。编码内容如下

```
ZmxhZzphenVsY3JlbWE=
```

进行解码，得到如下字符串

```
flag:azulcrema
```

其中`flag`为用户名，`azulcrema`为密码。尝试使用此密码来解开压缩包，结果正确，得到FLAG

```
IW{HTTP_BASIC_AUTH_IS_EASY}
```

---

404 Flag not found

> ascii转换，数据包分析

仍然是给了数据包，打开发现是一系列DNS请求，请求的域名非常奇怪，最前面似乎是一串十六进制数。尝试对其解码，使用ASCII解码发现了有意义的字符，把所有域名的十六进制数连在一起。得到

```
In the end, it's all about flags.
Whether you win or lose doesn't matter.
{Ofc, winning is cooler
Did you find other flags?
Noboby finds other flags!
Superman is my hero.
_HERO!!!_
Help me my friend, I'm lost in my own mind.
Always, always, for ever alone.
Crying until I'm dying.
Kings never die.
So do I.
}!
```

这段话比较奇怪，仔细研究发现，每行的首字母拼起来，就是FLAG。

```
IW{DNS_HACKS}
```