BugKu CTF(Crypto篇)---where is flag 5

原创

肖萧然 于 2022-04-15 19:05:45 发布

分类专栏： MyCTF # CRYPTO 文章标签： crypto

BugKu CTF(Crypto篇)—where is flag 5

文章目录

题目描述

```
Gx8EAA8SCBIfHQARCxMUHwsAHRwRHh8BEQwaFBQfGwMYCBYRHx4SBRQdGR8HAQ0QFQ==
```

首先一看就是base64

但是使用解密网站没有结果

> 使用 python base64解密 看看效果

```
import base64
str = "Gx8EAA8SCBIfHQARCxMUHwsAHRwRHh8BEQwaFBQfGwMYCBYRHx4SBRQdGR8HAQ0QFQ=="
str = base64.b64decode(str)
print(str)
```

```
b'\x1b\x1f\x04\x00\x0f\x12\x08\x12\x1f\x1d\x00\x11\x0b\x13\x14\x1f\x0b\x00\x1d\x1c\x11\x1e\x1f\x01\x11\x0c\x1a\x14\x14\x1f\x1
\x01\r\x10\x15'
```


> 发现是字节的形式,所以才不会显示

解码内容好像16进制,python提取出来

```
l = [hex(i).replace("0x", "") for i in str]
l = ["0"*(2-len(i))+i if(len(i) != 2) else i for i in l]
print(l)
```

```
['1b', '1f', '04', '00', '0f', '12', '08', '12', '1f', '1d', '00', '11', '0b', '13', '14', '1f', '0b', '00', '1d', '1c', '11', '1e', '1f', '01', '11', '0c', '1a', '1
'1b', '03', '18', '08', '16', '11', '1f', '1e', '12', '05', '14', '1d', '19', '1f', '07', '01', '0d', '10', '15']
```

但是 1b ,1f ,04 等这些数又不在 ascll字母范围

尝试提取16进制每个第一个的数字

因为第一个只有 1或0 所以试试

```
str="".join([i[0] for i in l])
print(len(str),str)
print(*[chr(int(str[i:i+7], 2)) for i in range(0, len(str), 7)])
```

```
49 1100010111010111001111101011111010111110111100011
b u g k u { c
```

由于 长度为 49 ,所以尝试 每7个转 ascll

推测出16进制转二进制后 竖着 提取 转 ascll

转 二进制

```
l = [bin(i).replace("0b", "") for i in str]
print(l)
```

0.8s

```
['11011', '11111', '100', '0', '1111', '10010', '1000', '10010', '11111', '11101', '0', '10
'11111', '1', '10001', '1100', '11010', '10100', '10100', '11111', '11011', '11', '11000',
'11111', '111', '1', '1101', '10000', '10101']
```

补全0

```
l = ["0"*(5-len(i))+i if(len(i) != 5) else i for i in l]
print(l)
```

0.1s

```
['11011', '11111', '00100', '00000', '01111', '10010', '01000', '10010', '11111', '11101', '00000', '10001', '01011', '10011', '10100', '11111', '01011', '00000', '11101', '11100',
'10001', '11110', '11111', '00001', '10001', '01100', '11010', '10100', '10100', '11111', '11011', '00011', '11000', '01000', '10110', '10001', '11111', '11110', '10010', '00101',
'10100', '11101', '11001', '11111', '00111', '00001', '01101', '10000', '10101']
```

解出

```
for i in range(5):
    str = "".join([ii[i] for ii in l])
    print(*[chr(int(str[i:i+7], 2))
            for i in range(0, len(str), 7)], sep="", end="")
```

0.7s

```
bugku{ce26f61d40fea75fc0b980d7588e}
```

总结

```
import base64
str = "Gx8EAA8SCBIfHQARCxMUHwsAHRwRHh8BEQwaFBQfGwMYCBYRHx4SBRQdGR8HAQ0QFQ=="
str = base64.b64decode(str)
l = [bin(i).replace("0b", "") for i in str]
l = ["0"*(5-len(i))+i if(len(i) != 5) else i for i in l]

for i in range(5):
    str = "".join([ii[i] for ii in l])
    print(*[chr(int(str[i:i+7], 2)) for i in range(0, len(str), 7)], sep="", end="")
```