BUUCTF-Crypto学习笔记(三)

原创

晓德 于 2020-12-29 22:42:00 发布 506 收藏

文章标签： 安全 密码学

本文链接：https://blog.csdn.net/weixin_42271850/article/details/111875910

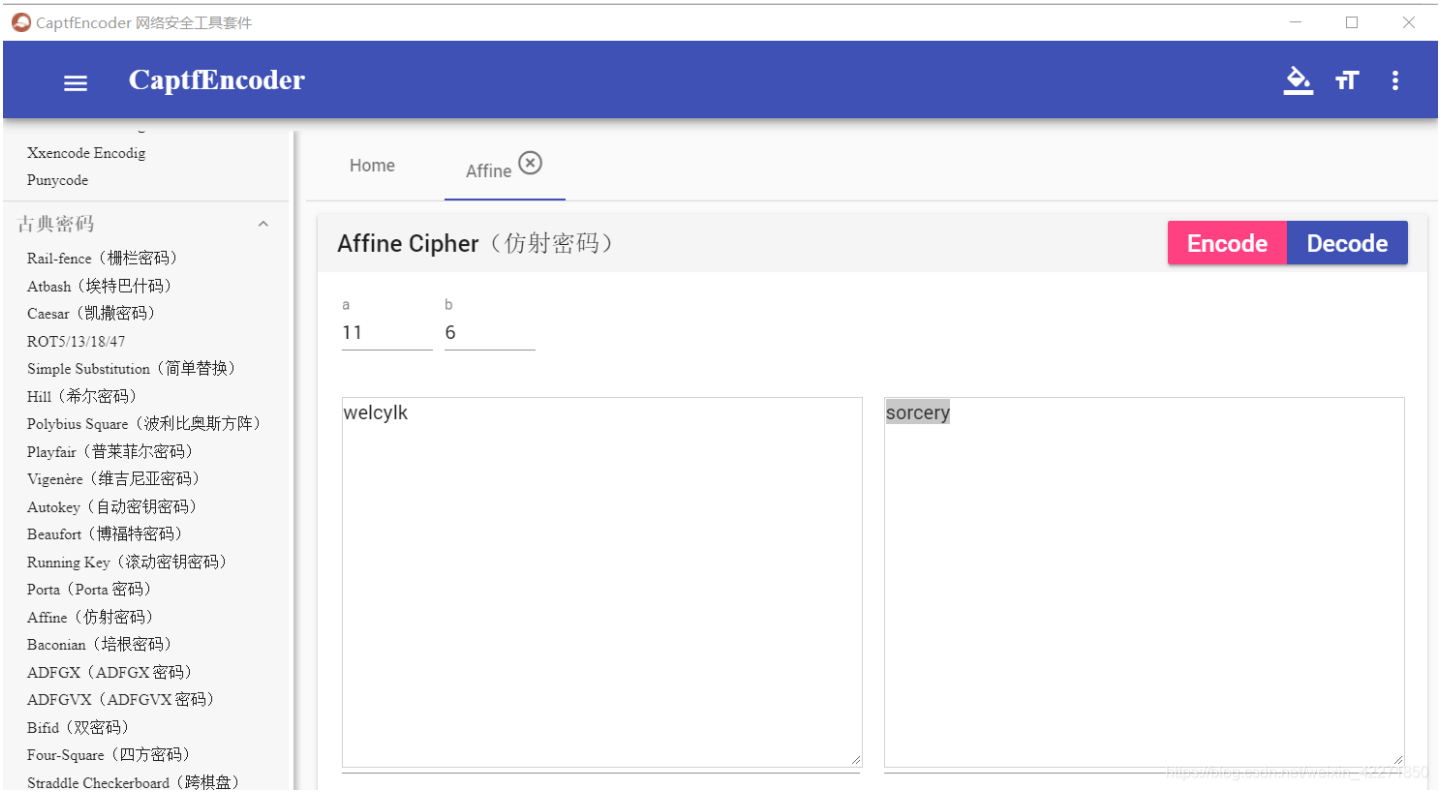
这是第三篇的BUUCTF-Crpto学习笔记了，希望能坚持下去

一、[GKCTF2020]小学生的密码学

打开题目，得到信息如下：

```
#题目原文
e(x)=11x+6(mod26)
密文：welcylk
（flag为base64形式）
```

如果你有做过类似的题目，看到题目这个格式很容易就能猜到这是仿射密码。其中a是11，b是6，直接通过在线的解密工具进行解密就能得到明文sorcery，然后再拿到在线base64的网站进行编码，最后得到答案为c29yY2VyeQ==。



二、传统知识+古典密码

打开题目需要下载文件，文件信息如下：

```
#题目原文
小明某一天收到一封密信，信中写了几个不同的年份
辛卯，癸巳，丙戌，辛未，庚辰，癸酉，己卯，癸巳。
信的背面还写有“+甲子”，请解出这段密文。
key值：CTF{XXX}
```

看到很明显就是天干地支纪年法，网上百度一下对应的数字

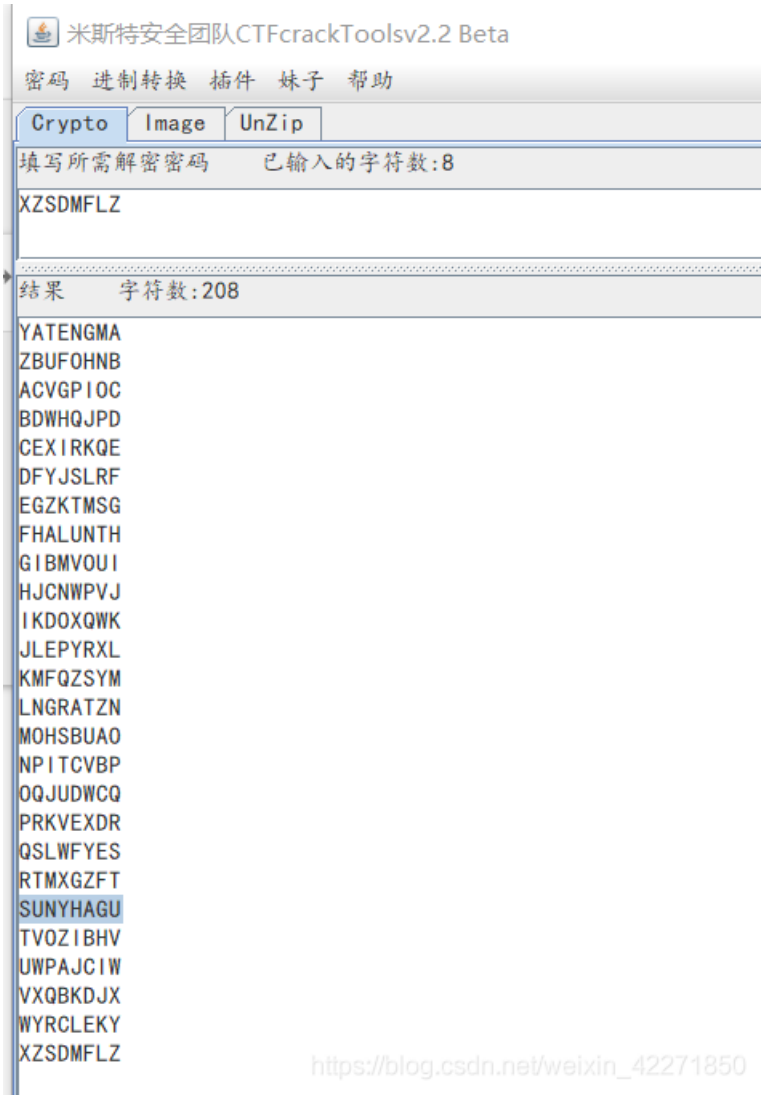
1	2	3	4	5	6	7	8	9	10
甲子	乙丑	丙寅	丁卯	戊辰	己巳	庚午	辛未	壬申	癸酉
11	12	13	14	15	16	17	18	19	20
甲戌	乙亥	丙子	丁丑	戊寅	己卯	庚辰	辛巳	壬午	癸未
21	22	23	24	25	26	27	28	29	30
甲申	乙酉	丙戌	丁亥	戊子	己丑	庚寅	辛卯	壬辰	癸巳
31	32	33	34	35	36	37	38	39	40
甲午	乙未	丙申	丁酉	戊戌	己亥	庚子	辛丑	壬寅	癸卯
41	42	43	44	45	46	47	48	49	50
甲辰	乙巳	丙午	丁未	戊申	己酉	庚戌	辛亥	壬子	癸丑
51	52	53	54	55	56	57	58	59	60
甲寅	乙卯	丙辰	丁巳	戊午	己未	庚申	辛酉	壬戌	癸亥

```
得到对应如下：
辛卯，癸巳，丙戌，辛未，庚辰，癸酉，己卯，癸巳
28    30    23    8    17   10   16  30
甲子
1
```

考虑了如果把甲子当成1的话，得到的数值完全达不到ASCII的范围。所以考虑将一甲子当成60一个轮回，+60。写以下脚本，得到XZSDMFLZ。

```
a = [28,30,23,8,17,10,16,30]
flag = ''
for j in a:
    flag += chr(j+60)
print(flag)
```

然后题目讲了除了传统知识外，还有一种古典加密方式。那这种没有其他密钥的古典加密，很容易想到栅栏和凯撒。但栅栏只是顺序改变，里面的字母不会变。这里面的字母这么拼都感觉不像一个单词，所以用凯撒解密，最后找到一个比较像单词的答案SUNYHAGU。



三、信息化时代的步伐

题目打开需要下载文件，文件信息如下：

```
#题目原文
也许中国可以早早进入信息化时代，但是被清政府拒绝了。
附件中是数十年后一位伟人说的话的密文。
请翻译出明文(答案为一串中文！) 注意：得到的 flag 请包上 flag{} 提交
60604615262360081783121612162119638 6
```

查看长度是36位，那么应该是每三个数字一组或者每两个数字一组。但如果是三个数字一组的话，第一组是606感觉不太像。那应该就是每两个一组。一开始尝试过先转成ASCII，然后进行凯撒解密。但发现行不通，百度了一下后才发现有个叫中文电码的编码方式。使用在线网站**http://code.mcdvisa.com**，查到结果为计算机要从娃娃抓起。

四、RSA1

打开题目需要下载文件，文件信息如下：

```
#题目原文
p = 863763376725700856709965348654109117132049150943361544753916243791124417588566780639841179052408355344515811
350222774520620532769093950403299469990205322 9
q = 126406749739964727691760479371708834209270508214800105815931371353724738805956137373376306297525773461470392
840300825934907766305725849599542053368802284 69
dp = 650079570221683462110904235119326153065004384105625293093094966335862501688183284072806602615026469307610 93
548740998413804548817160977783072681169105829 29
dq = 783472263673553449019532580386470672380574033551303889137911760438881683674556098098256795673512201963002 17
543876276751696804359958252753916081112055004 1
c = 247223054038873820735673164676490806626315529059602293990791079956021544181760563358006388875276141640735304
376570850796761573502053519452229893513160764865735995760419783398722659250627643185360890073102702785261596789 3
743190386289240074791552511898395997060793414297473667578432599344594203137210734210385 2
```

题目里面有dp、dq、p、q、c这几个参数，其中dp不是d*p而是d%(p-1)，dq为d%(q-1)，很容易想到是RSA题目中的dp泄露，直接通过脚本计算，得到16进制然后转成字符串为noxCTF{W31c0m3_70_Ch1n470wn}。

```
import gmpy2
p = 863763376725700856709965348654109117132049150943361544753916243791124417588566780639841179052408355344515811
350222774520620532769093950403299469990205322 9
q = 126406749739964727691760479371708834209270508214800105815931371353724738805956137373376306297525773461470392
840300825934907766305725849599542053368802284 69
dp = 650079570221683462110904235119326153065004384105625293093094966335862501688183284072806602615026469307610 93
548740998413804548817160977783072681169105829 29
dq = 783472263673553449019532580386470672380574033551303889137911760438881683674556098098256795673512201963002 17
543876276751696804359958252753916081112055004 1
c = 247223054038873820735673164676490806626315529059602293990791079956021544181760563358006388875276141640735304
376570850796761573502053519452229893513160764865735995760419783398722659250627643185360890073102702785261596789 3
743190386289240074791552511898395997060793414297473667578432599344594203137210734210385 2
I = gmpy2.invert(q,p)
mp = pow(c,dp,p)
mq = pow(c,dq,q)
m = (((mp-mq)*I)%p)*q+mq
print(hex(m))
```

五、凯撒？替换？呵呵！

点击题目，得到题目信息如下：

```
MTHJ{CUBCGXGUGXWREXIPOYAOEYFIGXWRXCHTKHFCOHCFDUCGTXZOHIXOEOWMEHZO}
注意：得到的 flag 请包上 flag{} 提交，flag{小写字母}
```

看题目不是应该不是普通的凯撒和替换，但是又没有给我们密码的对照表。所以扔到词频分析网站**https://quipqiup.com**。然后在Clues中填入MTHJ = FLAG这样能更加准确的还原。最后得到flag substitution cipher decryption is always easy just like a piece of cake。

六、old-fashion

点击题目需要我们下载文件，文件信息如下：

```
#题目原题
Os drnuzearyuwn, y jtkjzoztzoes douwlr oj y ilzwex eq lsdexosa kn pwodw tsozj eq ufyoszlbz yrl rlufydlx pozw dou
wlrzlbz, ydderxosa ze y rlatfyr jnjzli; mjy gfbmw vla xy wbfnsy symmyew (mjy vrwm qrvvrf), hlbew rd symmyew, meb
hsymw rd symmyew, vbomgeyw rd mjy lxrzy, lfk wr dremj. Mjy eyqybzye kyqbhjyew mjy myom xa hyedrevbfn lf bfzyewy
wgxwmbmgmbrf. Wr mjy dsln bw f1_2jyf-k3_jg1-vb-vl_l
```

题目应该是一大段英文，但相关字母完全不对。所以第一时间就考虑拿去词频分析，看最后一段话应该是XX the flag is xxxxxx。所以在词频分析网站**https://quipqiup.com**中Puzzle:填入密文，Clues填入**mjy=the dsln=flag bw=is**。最后得到分析后的明文n1_2hen-d3_hu1-mi-ma_a。其实好像填Puzzle也能分析出来，但是填的话会正确很多。

七、权限获得第一步

点击题目需要我们下载文件，文件信息如下：

```
Administrator:500:806EDC27AA52E314AAD3B435B51404EE:F4AD50F57683D4260DFD48AA351A17A8:::
```

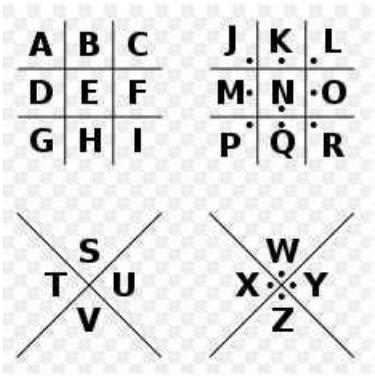
这个一看就是Windows系统下保存用户名和密码里的文件，直接拿后面的md5去网站**https://www.cmd5.com**进行解密，最后得到结果是3617656。

八、萌萌哒的八戒

点击题目提示如下，并要我们下载一个附件里面是一个图片：

```
萌萌哒的八戒原来曾经是猪村的村长，从远古时期，猪村就有一种神秘的代码。
请从附件中找出代码，看看萌萌哒的猪八戒到底想说啥。
注意：得到的 flag 请包上 flag{} 提交
```

如果做过类似题目的，很容易就猜到这个其实是猪圈密码。直接拿对照表解密，得到的结果是whenthepigwanttoeat。



九、[BJDCTF 2nd]灵能精通-y1ng

点击题目提示如下，并要我们下载一个附件里面是一个图片：

```
身经百战的Y1ng已经达到崇高的武术境界，以自律克己来取代狂热者的战斗狂怒与传统的战斗形式。
Y1ng所受的训练也进一步将他们的灵能强化到足以瓦解周遭的物质世界。借由集中这股力量，Y1ng能释放灵能能量风暴来摧毁敌人的心智、肉体与器械。
```

如果做过类似题目的，很容易就猜到这个其实是圣堂武士。直接拿对照表解密，得到的结果是IMKNIGHTSTEMPLAR。

十、RSA3

点击题目需要我们下载文件，文件信息如下：

```
#题目原题
c1=22322035275663237041646893770451933509324701913484303338076210603542612758956262869640822486470121149424485571361007421293675516338822195280313794991136048140918842471219840263536338886250492682739436410013436651161720725855484866690084788721349555662019879081501113222996123305533009325964377798892703161521852805956811219563883312896330156298621674684353919547558127920925706842808914762199011054955816534977675267395009575347820387073483928425066536361482774892370969520740304287456555508933372782327506569010772537497541764311429052216291198932092617792645253901478910801592878203564861118912045464959832566051361
n=22708078815885011462462049064339185898712439277226831073457888403129378547350292420267016551819052430779004755846649044001024141485283286483130702616057274698473611149508798869706347501931583117632710700787228016480127677393649929530416598686027354216422565934459015161927613607902831542857977859612596282353679327773303727004407262197231586324599181983572622404590354084541788062262164510140605868122410388090174420147752408554129789760902300898046273909007852818474030770699647647363015102118956737673941354217692696044969695308506436573142565573487583507037356944848039864382339216266670673567488871508925311154801
e1=11187289
c2=18702010045187015556548691642394982835669262147230212731309938675226458555210425972429418449273410535387985931036711854265623905066805665751803269106880746769003478900791099590239513925449748814075904017471585572848473556490565450062664706449128415834787961947266259789785962922238701134079720414228414066193071495304612341052987455615930023536823801499269773357186087452747500840640419365011554421183037505653461286732740983702740822671148045619497667184586123657285604061875653909567822328914065337797733444640351518775487649819978262363617265797982843179630888729407238496650987720428708217115257989007867331698397
e2=9647291
```

可以看到题目给了n、c1、e1、c2、e2，看到条件很容易想到这是共模攻击。就是给定一个n，然后用不同e生成不同的公私钥，加密同一段明文得到两段不同的密码C1、C2。使用共模攻击的脚本解密，得到16进制，然后放到在线16进制转字符串的网站进行转换，得到lag{49d91077a1abcb14f1a9d546c80be9ef}。

```python
from gmpy2 import invert

def gongmogongji(n, c1, c2, e1, e2):
    def egcd(a, b):
        if b == 0:
            return a, 0
        else:
            x, y = egcd(b, a % b)
            return y, x - (a // b) * y
    s = egcd(e1, e2)
    s1 = s[0]
    s2 = s[1]

    # 求模反元素
    if s1 < 0:
        s1 = - s1
        c1 = invert(c1, n)
    elif s2 < 0:
        s2 = - s2
        c2 = invert(c2, n)
    m = pow(c1, s1, n) * pow(c2, s2, n) % n
    return m


def hex_to_str(s):
  return ''.join([chr(i) for i in [int(b, 16) for b in s.split(r'/x')[1:]]])

n= 22708078815885011462462049064339185898712439277226831073457888403129378547350292420267016551819052430779004755846649044001024141485283286483130702616057274698473611149508798869706347501931583117632710700787228016480127677393649929530416598686027354216422565934459015161927613607902831542857977859612596282353679327773037270044072621972315863245991819835726224045903540845417880622621645101406058681224103880901744201477524085541297897609023008980462739090078528184740307706996476473630151021189567376739413542176926960449696953085064365731425655734875835070373569448480398643823392162666706735674888715089253111548014e1= 11187289e2= 9647291c1= 22322035275663237041646893770451933509324701913484303338076210603542612758956262869640822486470121149424485571361007421293675516338822195280313794991136048140918842471219840263536338886250492682739436410013436651161720725855484866690084788721349555662019879081501113222996123305533009325964377798892703161521852805956811219563883312896330156298621674684353919547558127920925706842808914762199011054955816534977675267395009575347820387073483928425066536361482774892370969520740304287456555508933372782327506569010772537497541764311429052216291198932092617792645253901478910801592878203564861118912045464959832566051361c2= 18702010045187015556548691642394982835669262147230212731309938675226458555210425972429418449273410535387985931036711854265623905066805665751803269106880746769003478900791099590239513925449748814075904017471585572848473556490565450062664706449128415834787961947266259789785962922238701134079720414228414066193071495304612341052987455615930023536823801499269773357186087452747500840640419365011554421183037505653461286732740983702740822671148045619497667184586123657285604061875653909567822328914065337797733444640351518775487649819978262363617265797982843179630888729407238496650987720428708217115257989007867331698397result = gongmogongji(n, c1, c2, e1, e2)result = hex(result)print(result)
```