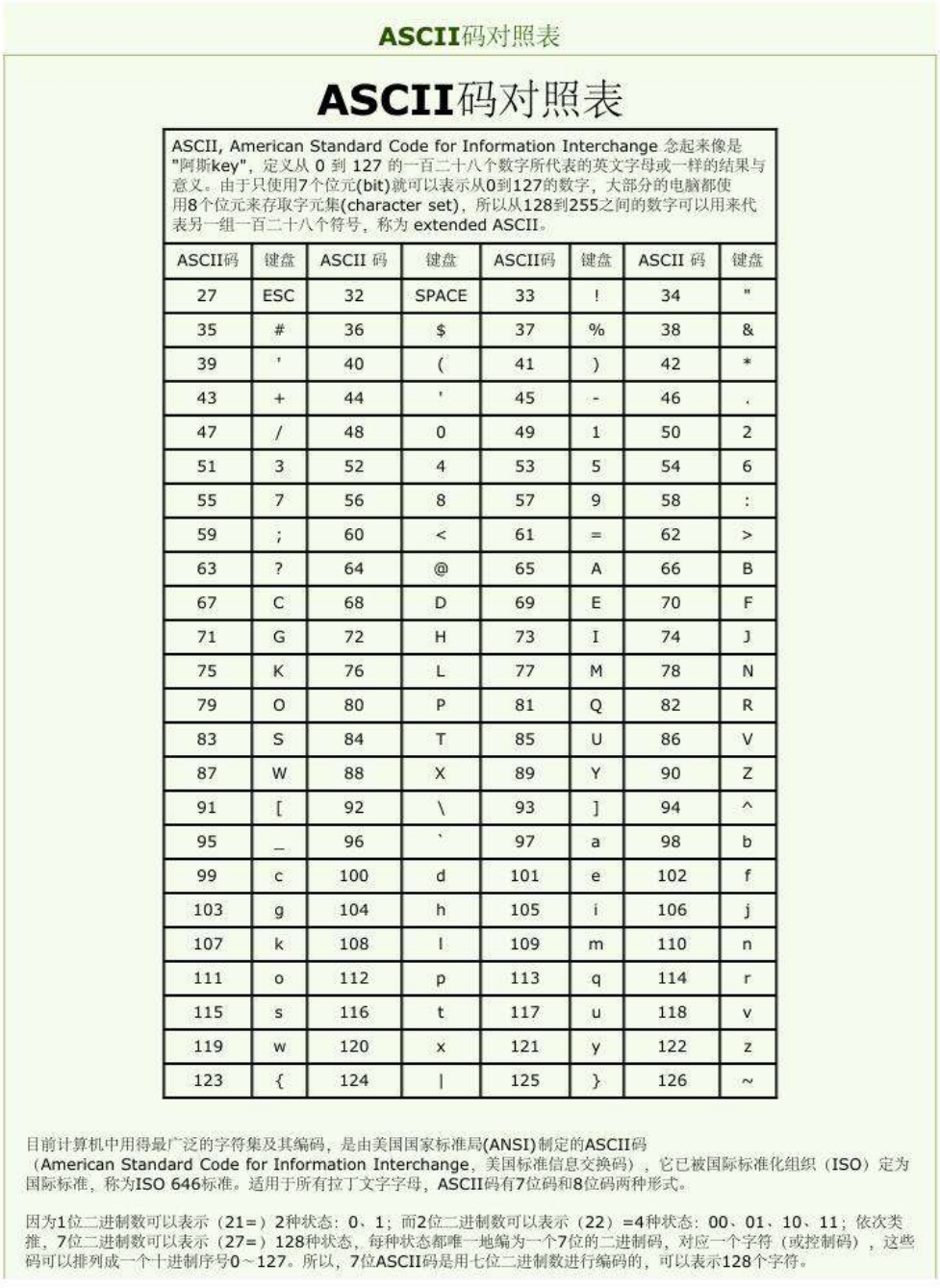
密码

笔记本: CTF
创建时间: 2020/3/27 14:57 **更新时间:** 2020/3/31 10:07
作者: Cecilia
URL: https://ctf.bugku.com/challenges#%E5%AF%8C%E5%BC%BA%E6%B0%91%E4%...

【ASCII码】

ASCII码对照表 - Linux/Unix编程 - 工作中用到的东西



ASCII码对照表

ASCII码对照表

ASCII, American Standard Code for Information Interchange 念起来像是"阿斯key"，定义从 0 到 127 的一百二十八个数字所代表的英文字母或一样的结果与意义。由于只使用7个位元(bit)就可以表示从0到127的数字，大部分的电脑都使用8个位元来存取字元集(character set)，所以从128到255之间的数字可以用来代表另一组一百二十八个符号，称为 extended ASCII。

ASCII码	键盘	ASCII 码	键盘	ASCII码	键盘	ASCII 码	键盘
27	ESC	32	SPACE	33	!	34	"
35	#	36	$	37	%	38	&
39	'	40	(	41	)	42	*
43	+	44	'	45	-	46	.
47	/	48	0	49	1	50	2
51	3	52	4	53	5	54	6
55	7	56	8	57	9	58	:
59	;	60	<	61	=	62	>
63	?	64	@	65	A	66	B
67	C	68	D	69	E	70	F
71	G	72	H	73	I	74	J
75	K	76	L	77	M	78	N
79	O	80	P	81	Q	82	R
83	S	84	T	85	U	86	V
87	W	88	X	89	Y	90	Z
91	[	92	\	93	]	94	^
95	_	96	`	97	a	98	b
99	c	100	d	101	e	102	f
103	g	104	h	105	i	106	j
107	k	108	l	109	m	110	n
111	o	112	p	113	q	114	r
115	s	116	t	117	u	118	v
119	w	120	x	121	y	122	z
123	{	124	|	125	}	126	~

目前计算机中用得最广泛的字符集及其编码，是由美国国家标准局(ANSI)制定的ASCII码（American Standard Code for Information Interchange，美国标准信息交换码），它已被国际标准化组织（ISO）定为国际标准，称为ISO 646标准。适用于所有拉丁文字字母，ASCII码有7位码和8位码两种形式。

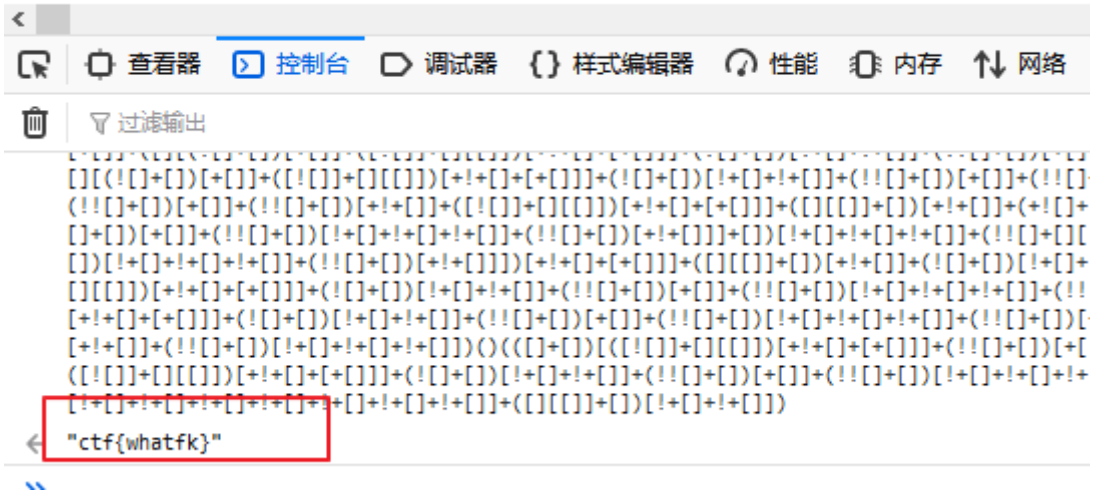
因为1位二进制数可以表示（21=）2种状态：0、1；而2位二进制数可以表示（22）=4种状态：00、01、10、11；依次类推，7位二进制数可以表示（27=）128种状态，每种状态都唯一地编为一个7位的二进制码，对应一个字符（或控制码），这些码可以排列成一个十进制序号0～127。所以，7位ASCII码是用七位二进制数进行编码的，可以表示128个字符。

http://blog.chinaunix.net/u/6593/showart_304002.html[2010-4-27 17:40:03]

【ok密码】
https://www.splitbrain.org/services/ook
https://tool.bugku.com/brainfuck/
1、Ook.ok
2、... .... !?!!. ?

【Brainfuck密码】
https://tool.bugku.com/brainfuck/
1、+++++ ++[-> +++++ ++<]> ++++. +.--. ...<

【JSfuck】
1、(!![]+[])[+[]]+(!![]+[])[!+[]+!+[]+!+[]]+(!![]+[])[+!+[]]+[])[!+[]+!+[]+!+[]+(![]+[])[!+[]+!+[]+!+[]]()[+[]])[+[]]+[!+[]+!+[]+!+[]+!+[]+!+[]+!+[]+!+[]+!+[]+!+[]+([][[]]+[])[!+[]+!+[]])
使用firefox F12控制台回车。



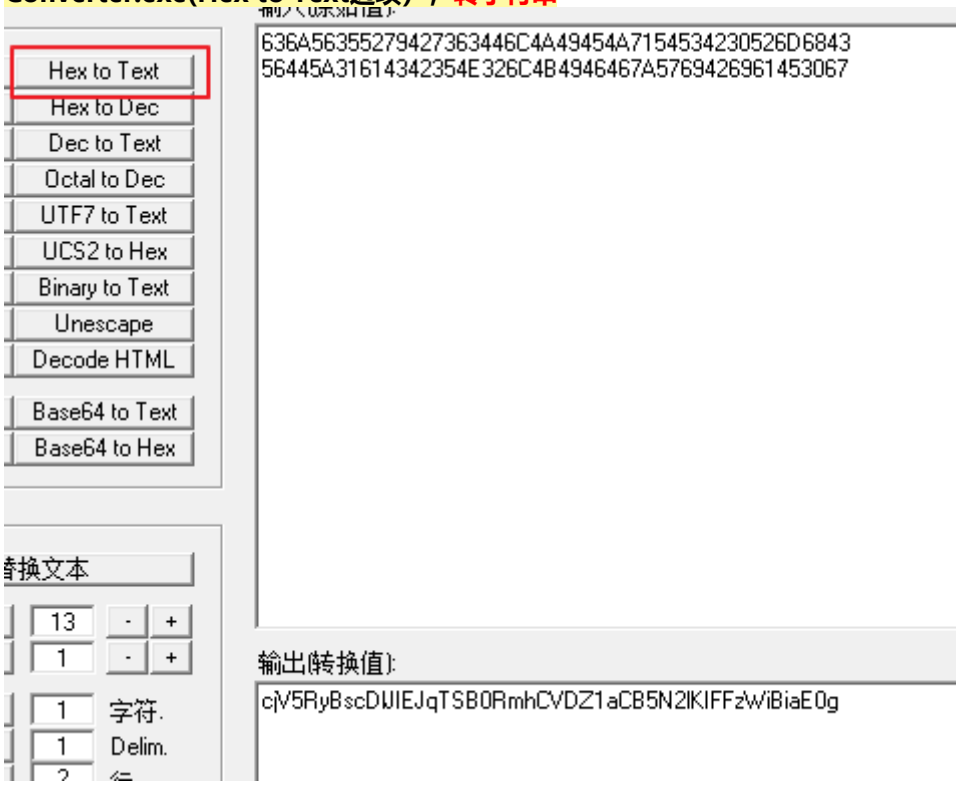
【base91密码】
http://ctf.ssleye.com/base91.html
1、 @iH<,{bdR2H;i6*Tm,Wx2izpx2!
【base91需要91个字符来表示用ASCII编码的二进制数据。在94个可打印的ASCII字符（0x21-0x7e）中，以下三个字符被省略以构建base91字母表-（破折号，0x2d）\\（反斜杠，0x5C）'（撇号，0x27）】

【进制转换】
O 八进制（逢八进一），Octal，缩写OCT或O，一种以8为基数的计数法，用O表示，为了不与数字0混淆，有时也用Q来表示八进制数。
D 十进制（逢十进一），英文为Decimal System，缩写Dec或D，来源于希腊文Decem，用D表示。
0x十六进制（逢十六进一），英文名称：Hexadecimal，是计算机中数据的一种表示方法。它由0-9，A-F组成，字母不区分大小写，前缀用0x表示。
B 二进制（逢二进一），Binary，用b表示。
E:\ctf\3-bugku\3-密码\进制转换\trans.py，需要python2环境。在线编译网站
https://c.runoob.com/compile/6

【八进制】
1、147145163 （三个一组，不超过8的数字序列）
E:\ctf\3-bugku\1-MISC\100-普通的二维码\misc80\rev.py （转ASCII码）
2、147 145 163 （三个一组，不超过8的数字序列，中间有空格）

八进制 十进制 ASCII相互转换.exe　转ASCII码
3、\134\170\65\143\134\170\67\65\134
Converter.exe(unescape选项）转十六进制

【十六进制】
1、73646E6973635F32303138
Converter.exe(Hex to Text选项），转字符串



2、\x5c\x75\x30\x30\x35\x33\x5c\x75\x30\x30\x37
Converter.exe(Hex to Text选项）转unicode编码

【Unicode 】
1、&#102（此种编码方式不是HTML编码，需要继续转换）
Converter.exe(Decode HTML选项）,转网页Unicode码
2、KEY
采用CTFCrackTools.jar，转ASCII码
3、\u0053\u0074\u0072\u0069\u006e\u0067\u002e
Converter.exe(unescape选项）转CharCode码

【ASCII偏移】
1、e6Z9i~]8R~U~QHE{RnY{QXg~QnQ{^XVlRXlp^XI5Q6Q6SKY8jUAA
字符串结尾为AA，对应ASCII 65，base64结尾为==，对应ASCII 61。偏移四位。因此判断为ASCII偏移密码。
E:\ctf\3-bugku\3-密码\简单加密\caesar.py，偏移后，显示字符串为base64编码

```
>>>
================ RESTART: E:\ctf\3-bugku\3-密码\简单加密\caesar.py =============
===
a2V5ezY4NzQzMDAwNjUwMTczMjMwZTRhNThlZTE1M2M2OGU4fQ==
```

2、 gndk€rlqhmtkwwp}z
gndk的10进制的ASCII码分别是：103 110 100 107，
flag的10进制的ASCII码分别是： 102 108 97 103。发现ASCII以此减少 1 2 3 4，所以以此类推后面会继续减少，按规律减少的ASCII偏移。
E:\ctf\3-bugku\3-密码\奇怪的密码\reascii.py，偏移后，展现字符串

```
>>>
========== RESTART: E:\ctf\3-bugku\3-密码\奇怪的密码\reascii.py ==========
====
>>>
flagClei_ci_jiami
>>>
```

【URL编码】
1、%07D%07B

【未知码 CharCode】
1、String.fromCharCode(38,35,120,50)
javascript函数，括号里面为参数
删掉解释和括号，Converter.exe(Dec to Text选项）转unicode编码

【base64】
首先：base64编码后的字符串的长度一定会被4整除，包括用作后缀的等号吧；如果明文字符数不能被3整除，余1时，1个字符转为2个，补2个等号，共4个字符；余2时，2个字符转为3个字符，补1个等号，共4个字符；其实归根结底就是一句话：经过base64编码后的字符串长度一定会被4整除（包括后缀等号）

简单来说：

1. 标准base64只有64个字符（英文大小写、数字和+、/）以及用作后缀等号；

2.base64是把3个字节变成4个可打印字符，所以base64编码后的字符串一定能被4整除（不算用作后缀的等号）；

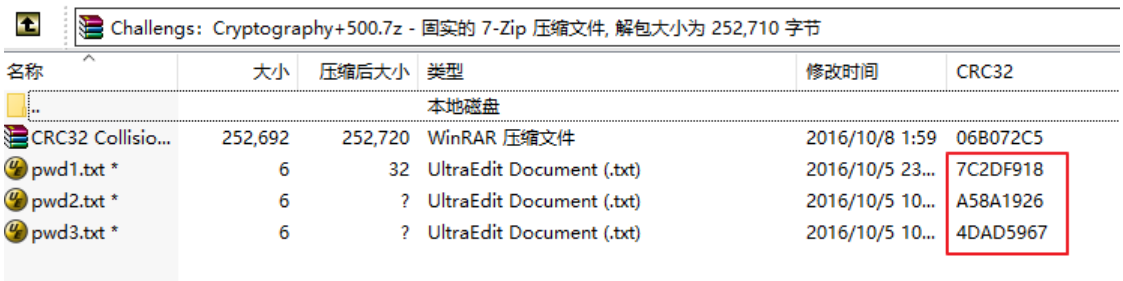
3.等号一定用作后缀，且数目一定是0个、1个或2个。这是因为如果原文长度不能被3整除，base64要在后面添加\0凑齐3n位。为了正确还原，添加了几个\0就加上几个等号。显然添加等号的数目只能是0、1或2；

（ps：当然~ 看到这种类似字符组合的字符串，你可能分不清哪些是base64，哪些是base32，那么我们直接可以放进在线加解密的网站去试试~ 见多了就可以自己大概判断出来啦~）
1、iVBORw0KGgoAAAANSUhEUgA==
采用CTFCrackTools.jar，转字符串。

【base64转图片】
http://tool.chinaz.com/tools/imgtobase/
1、

【morse】
1、 0010 0100 01 110 1111011 11 11111 010 000 0 001101 1010 111 100 0 001101 01111 000 001101 00 10 1 0 010 0 000 1 01111 10 11110 101011 1111101
长短不一的01序列
E:\ctf\3-bugku\3-密码\easy_crypto\mose2ascii.py，转ASCII码

【CRC32】
1、多见于加密压缩包。



2、CRC32解码（依次爆破CRC码，得到部分解压密码，连接起来即为完整密码）
E:\ctf\4-长亭工具\crc32-master> python crc32.py reverse 0x7c2df918

```
PS E:\ctf\4-长亭工具\crc32-master> python crc32.py reverse 0x7c2df918
4 bytes: {0x1c, 0x00, 0x1c, 0xa1}
verification checksum: 0x7c2df918 (OK)
alternative: 5EJeBD (OK)
alternative: 74bFvQ (OK)
alternative: D4W1dU (OK)
alternative: Jvea5S (OK)
alternative: OSgAFe (OK)
alternative: WtU1WB (OK)
alternative: XgD1qA (OK)
alternative: _3n26b (OK)
alternative: _CRC32 (OK)
alternative: a5KHAn (OK)
alternative: dvIh2X (OK)
alternative: fJLvkE (OK)
alternative: hESFWK (OK)
alternative: l1r6Sx (OK)
alternative: pbakF1 (OK)
alternative: uGcK5Z (OK)
alternative: vgh8vJ (OK)
alternative: xt8TKP (OK)
alternative: ytyePI (OK)
```

【维吉尼亚密码】
E:\ctf\3-bugku\1-MISC\300-就五层\Challengs：Cryptography+500\CRC32 Collision\pwd.py
依赖于ciphertext.txt和keys.txt两个文件，产生pwd.txt文件。

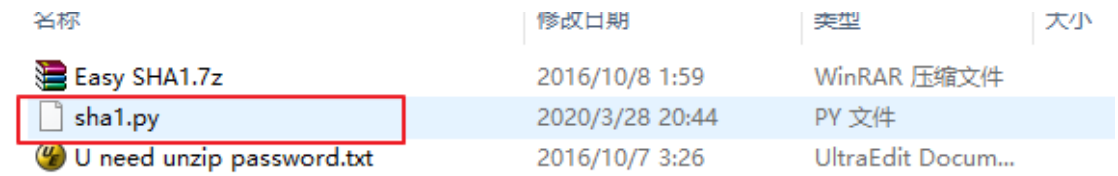
【sha1密码】

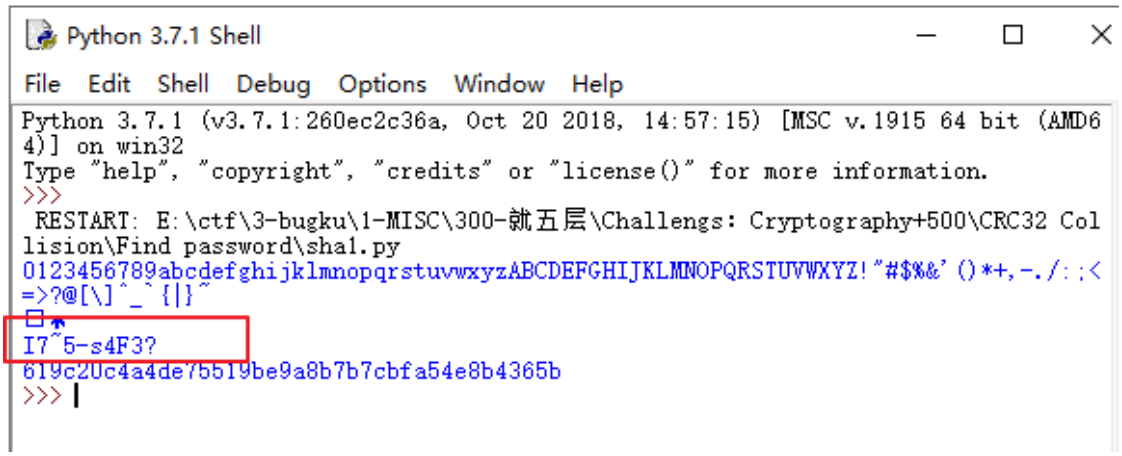
不完整的密码："*7*5-*4*3?" *代表可打印字符

不完整的sha1："619c20c*a4de755*9be9a8b*b7cbfa5*e8b4365*" *代表可打印

人生苦短，我用Python。

Congratulations!
E:\ctf\3-bugku\1-MISC\300-就五层\Challengs：Cryptography+500\CRC32 Collision\Find password\sha1.py



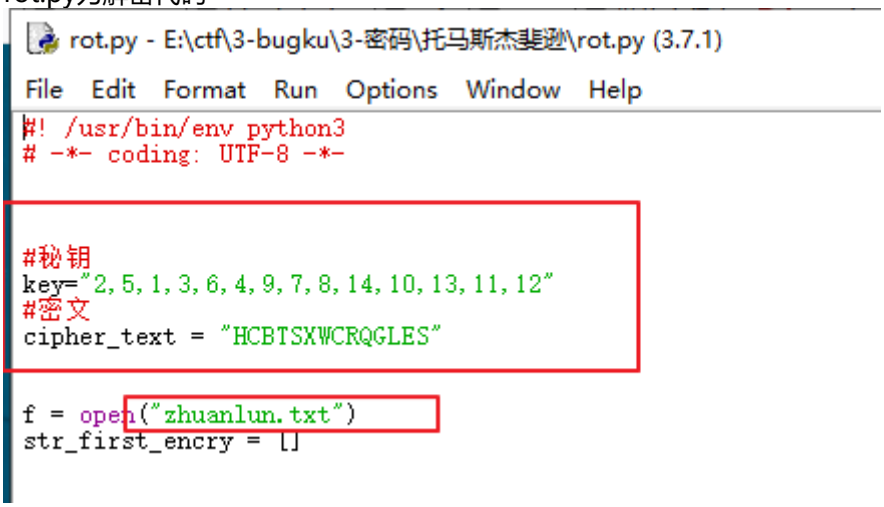


【MD5】
【RSA】
【DES】
【AES】

【键盘密码】
r5yG lp9l BjM tFhB T6uh y7iJ QsZ bhM 开脑洞，在键盘上画一下，标志是 3-4个字母为一组，中间有空格。

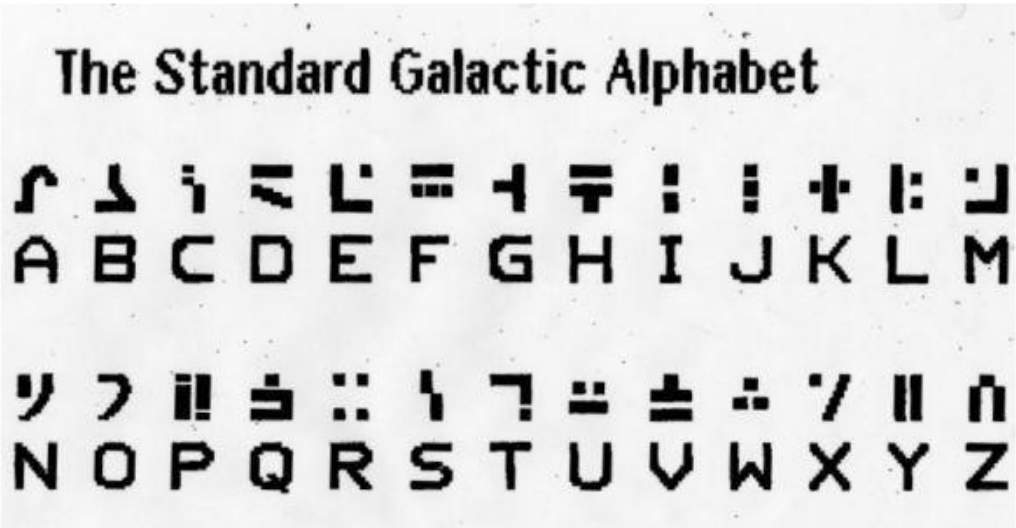
【核心价值观编码】
http://ctf.ssleye.com/cvencode.html

【 杰弗逊转轮加密】
https://www.cnblogs.com/0yst3r-2046/p/11810574.html
E:\ctf\3-bugku\3-密码\托马斯杰斐逊
zhuanlun.txt 为密文
rot.py为解密代码



1、银河标准字母

没办法，上网搜索，发现是银河标准字母，如图



2、zip伪加密
一个 ZIP 文件由**三个部分**组成：

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标志

压缩源文件数据区：
50 4B 03 04：这是头文件标记（0x04034b50）
14 00：解压文件所需 pkware 版本
00 00：全局方式位标记（有无加密）
08 00：压缩方式
5A 7E：最后修改文件时间
F7 46：最后修改文件日期
16 B5 80 14：CRC-32校验（1480B516）
19 00 00 00：压缩后尺寸（25）
17 00 00 00：未压缩尺寸（23）
07 00：文件名长度
00 00：扩展记录长度

压缩源文件目录区：
50 4B 01 02：目录中文件文件头标记(0x02014b50)
3F 00：压缩使用的 pkware 版本
14 00：解压文件所需 pkware 版本
00 00：全局方式位标记（有无加密，这个更改这里进行伪加密，改为09 00打开就会提示有密码了）
08 00：压缩方式
5A 7E：最后修改文件时间
F7 46：最后修改文件日期
16 B5 80 14：CRC-32校验（1480B516）
19 00 00 00：压缩后尺寸（25）
17 00 00 00：未压缩尺寸（23）
07 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 00 00 00：外部文件属性
00 00 00 00：局部头部偏移量

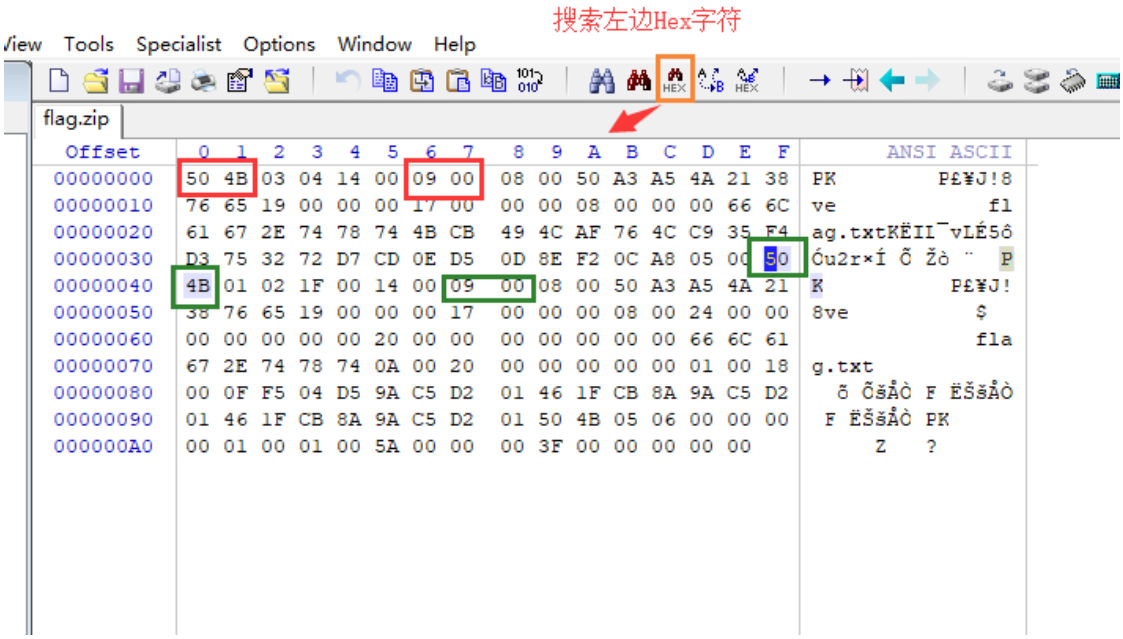
压缩源文件目录结束标志：
50 4B 05 06：目录结束标记
00 00：当前磁盘编号
00 00：目录区开始磁盘编号
01 00：本磁盘上纪录总数
01 00：目录区中纪录总数
59 00 00 00：目录区尺寸大小
3E 00 00 00：目录区对第一张磁盘的偏移量
00 00：ZIP 文件注释长度

因此，根据此题目来看：

压缩源文件数据区：50 4B 03 04：这是头文件标记

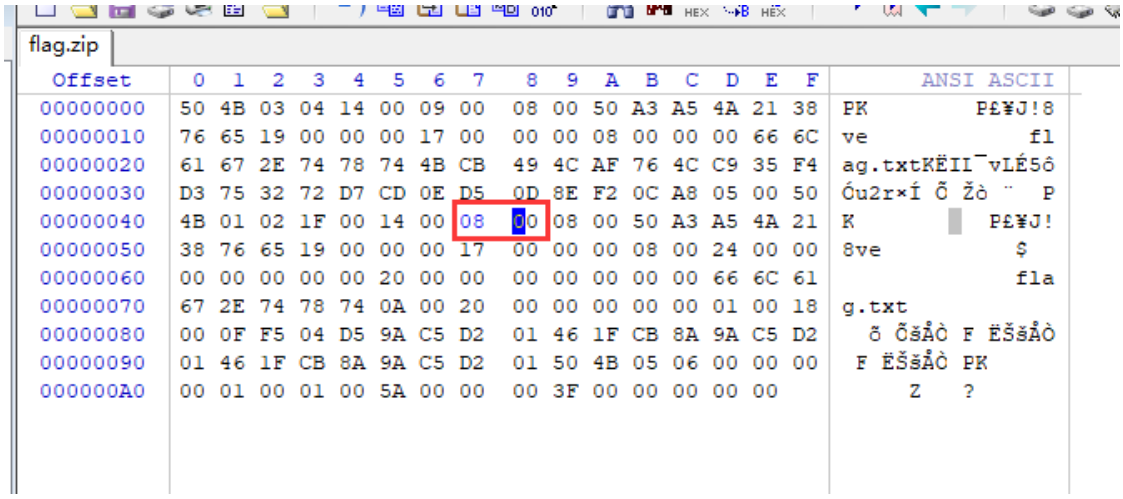
压缩源文件目录区：
50 4B 01 02：目录中文件文件头标记
3F 00：压缩使用的 pkware 版本
14 00：解压文件所需 pkware 版本
00 00：全局方式位标记（有无加密，这个更改这里进行伪加密，改为09 00打开就会提示有密码了）
压缩源文件目录结束标志 ： 50 4B 05 06：目录结束标记

我们用winhex打开压缩包，搜索504B，点击第二个504B（压缩源文件目录区）



软件：winhex
我们看到上图，红色框住的50 4B 是压缩源文件数据区的头文件标记，它对应的红色框柱的 09 00 并不影响加密属性。
绿色框住的50 4B 是压缩源文件目录区 ， 它对应的绿色框柱的 09 00 影响加密属性，当数字为奇数是为加密，为偶数时不加密。

因此我们更改标志位保存即可。

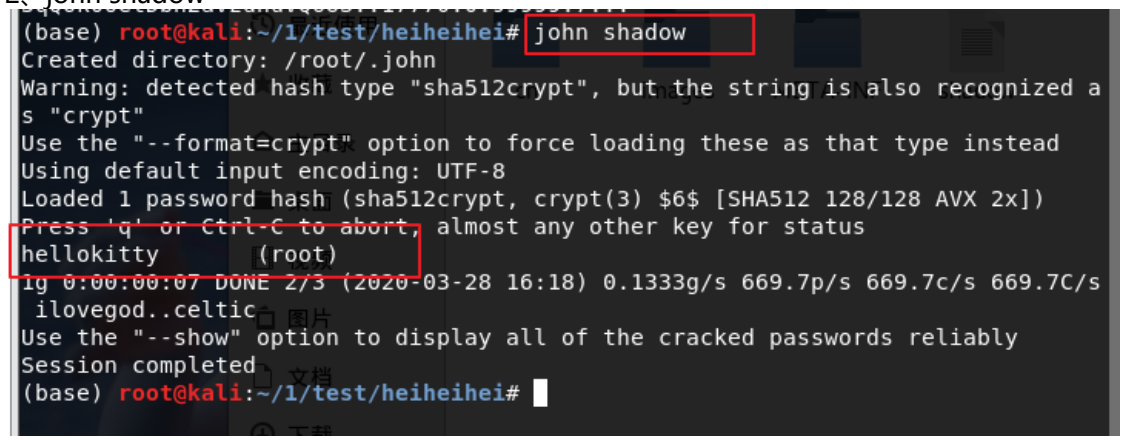


【散乱的密文】
lf5{ag024c483549d7fd@@1}
2 1 6 5 3 4
6个字符分组，依次调整顺序：flag{52048c453d794df1}@@

或将全部字符都写到表内，按列依次取出：**f25dl03fa4d1g87}{c9@544@**
栅栏密码解密flag{52048c453d794df1}@@

【shadow】
1、 more shadow（与拖到winhex里看是一样的结果）

2、john shadow



全称叫“John the Ripper password cracker（约翰开膛手密码破解）”。

John the Ripper是一个快速的密码破解程序，目前可用于Unix，Windows，DOS和OpenVMS的许多口味。其主要目的是检测弱Unix密码。

除了在各种Unix系统上最常见的几种crypt（3）密码哈希类型之外，现在支持的还有Windows LM哈希，以及社区增强版本中的大量其他哈希和密码。

John the Ripper是免费的开源软件，主要以源代码形式发布。

网址：http://www.openwall.com/john/

--single[=SECTION]]“单裂”模式

--wordlist[=FILE] --stdin 单词表模式，从FILE或stdin读取单词

--pipe 像--stdin一样，但批量读取，并允许规则

--loopback[=FILE] 像 --wordlistg一样, 但是从.pot文件中获取单词

--dupe-suppression 压制wordlist中的所有模糊（并强制预加载）

--prince[=FILE] PRINCE模式，从FILE中读取单词

--encoding=NAME 输入编码（例如，UTF-8，ISO-8859-1）。 也可以看看doc / ENCODING和--list = hidden-options。

--rules[=SECTION] 为单词表模式启用单词修改规则

--incremental[=MODE] “增量”模式[使用部分模式]

--mask=MASK 掩码模式使用MASK

--markov[=OPTIONS] “马尔可夫”模式（参见doc / MARKOV）

--external=MODE 外部模式或字过滤器

--stdout[=LENGTH] 只是输出候选人密码[在LENGTH切]

--restore[=NAME] 恢复被中断的会话[名为NAME]

--session=NAME 给一个新的会话NAME

--status[=NAME] 打印会话的状态[名称]

--make-charset=FILE 制作一个字符集文件。 它将被覆盖

--show[=LEFT] 显示破解的密码[如果=左，然后uncracked]

--test[=TIME] 运行测试和每个TIME秒的基准

--users=[-]LOGIN|UID[,..] [不]只加载这个（这些）用户

--groups=[-]GID[,..] 只加载这个（这些）组的用户

--shells=[-]SHELL[,..] 用[out]这个（这些）shell来加载用户

--salts=[-]COUNT[:MAX] 用[out] COUNT [到MAX]散列加载盐

--save-memory=LEVEL 启用内存保存，级别1..3

--node=MIN[-MAX]/TOTAL 此节点的数量范围不在总计数中

--fork=N 叉N过程

--pot=NAME 锅文件使用

--list=WHAT 列表功能，请参阅--list = help或doc / OPTIONS

--format=NAME 强制使用NAME类型的散列。 支持的格式可以用--list=formats和--list=subformats来看

【仿射密码】！！！！！！什么鬼啊！
https://www.cnblogs.com/0yst3r-2046/p/12172757.html

【Python一题】
题解 https://www.cnblogs.com/0yst3r-2046/p/12123653.html