CTF密码学-加解密总结

原创

zkzq 于 2022-03-04 14:38:23 发布 3057 收藏 3
文章标签： web安全 ctf 渗透测试 网络安全

密码学基本简介

密码学（在西欧语文中，源于希腊语kryptós“隐藏的”，和gráphein“书写”）是研究如何隐密地传递信息的学科。

在现代特别指对信息以及其传输的数学性研究，常被认为是数学和计算机科学的分支，和信息论也密切相关。

著名的密码学者Ron Rivest解释道：“密码学是关于如何在敌人存在的环境中通讯”，自工程学的角度，这相当于密码学与纯数学的异同。

密码学是信息安全等相关议题，如认证、访问控制的核心。密码学的首要目的是隐藏信息的涵义，并不是隐藏信息的存在。密码学也促进了计算机科学，特别是在于电脑与网络安全所使用的技术，如访问控制与信息的机密性。密码学已被应用在日常生活：包括自动柜员机的芯片卡、电脑使用者存取密码、电子商务等等。
密码是通信双方按约定的法则进行信息特殊变换的一种重要保密手段。依照这些法则，变明文为密文，称为加密变换；变密文为明文，称为脱密变换。密码在早期仅对文字或数码进行加、脱密变换，随着通信技术的发展，对语音、图像、数据等都可实施加、脱密变换。

密码学的发展

第一个阶段是从古代到19世纪末——古典密码（classical cryptography）
第二个阶段从20世纪初到1949年——近代密码
第三个阶段从C.E.Shannon（香农）于1949年发表的划时代论文“The Communication Theory of Secret Systems”开始——现代密码
第四个阶段从1976年W.Diffie和M.Hellman创造性地发表了论文”New Directions in Cryptography”开始——公钥密码

对称和非对称算法

对称密码算法（Symmetric cipher）：加密密钥和解密密钥相同，或实质上等同，即从一个易于推出另一个。又称传统密码算法（Conventional cipher）、秘密密钥算法或单密钥算法。DES、3DES、IDEA、AES
非对称密码算法（Asymmetric cipher）：加密密钥和解密密钥不同，从一个很难推出另一个。又叫公钥密码算法（Public-key cipher）。其中的加密密钥可以公开，称为公开密钥（public key），简称公钥；解密密钥必须保密，称为私人密钥（private key），简称私钥。RSA、ECC、ElGamal

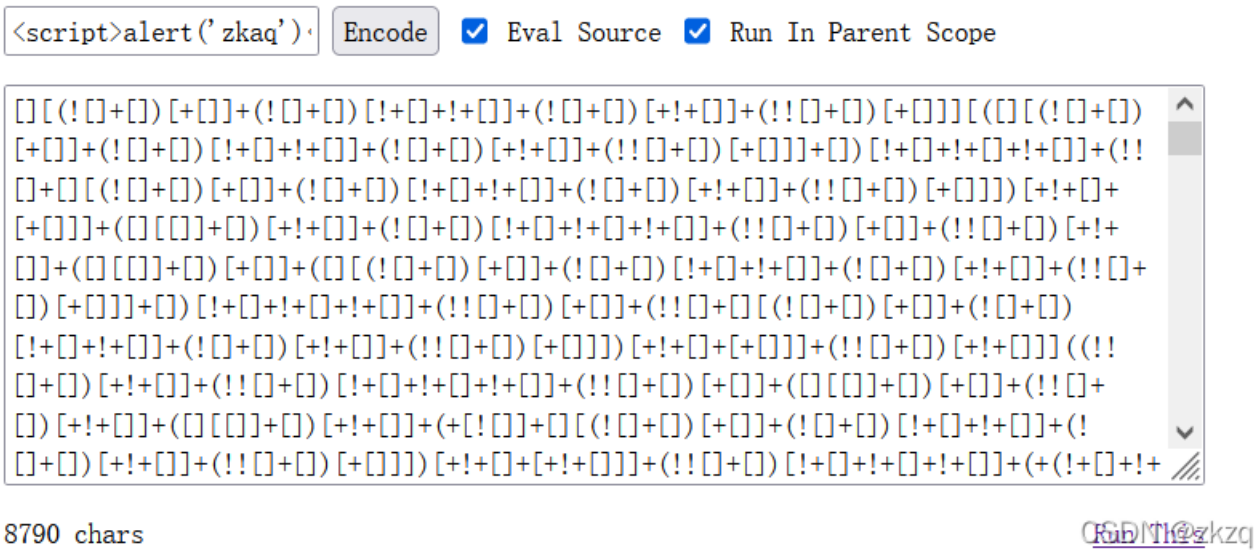
JS混淆

有些时候开发者为了保护劳动成果可以通过对javascript的变量名称和过程名称进行编码，从而起到混淆is代码的作用，通常使用eval函数进行混淆处理，该函数可以计算字符串，并执行其中的JS代码。
如，对进行16进制转换，然后使用eval函数进行读取

特点：通常在S脚本里使用eval与function函数进行混淆。
JS在线解混淆：http://www.atool.org/jscompression.php

JSFuck

JSFuck 是用6个字符[0（0！+来编写JavaScript程序，如下图所示，经过加密后便使用了[00！+进行编写，点击run this 可进行解密，或将密文放在浏览器的console控制台上进行解密。



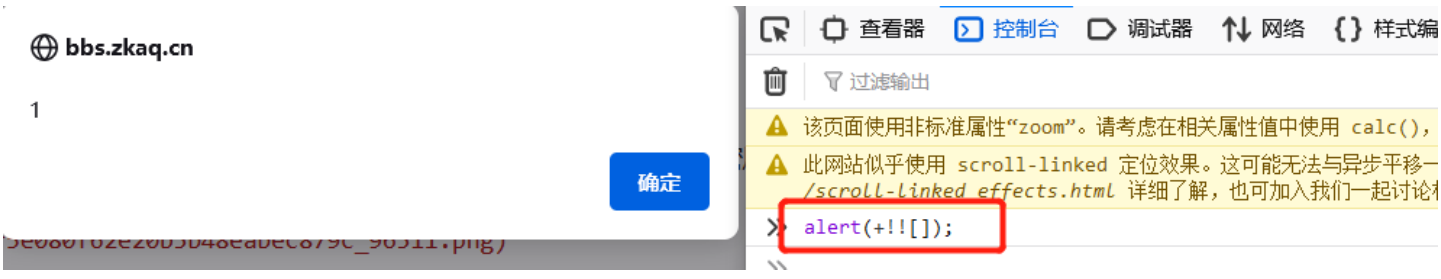
JSFuck在线加解密：http://www.jsfuck.com/

Jother

Jother是一种运用于javascript语言中利用少量字符构造精简的匿名函数方法对于字符串进行的编码方式。其中8个少量字符包括：！+（）[]{}。只用这些字符就能完成对任意字符串的编码（可以在浏览器的console控制台上直接解密）。例如0到9

```
"+[]",                                          //0
"+![]",                                         //1
"!![]+!![]",                                    //2
"!![]+!![]+!![]",                               //3
"!![]+!![]+!![]+!![]",                          //4
"!![]+!![]+!![]+!![]+!![]",                     //5
"!![]+!![]+!![]+!![]+!![]+!![]",                //6
"!![]+!![]+!![]+!![]+!![]+!![]+!![]",           //7
"!![]+!![]+!![]+!![]+!![]+!![]+!![]+!![]",      //8
"!![]+!![]+!![]+!![]+!![]+!![]+!![]+!![]+!![]"  //9
```

CSDN @zkzq



aaencode

aaencode使用的表情符号对js代码进编码，可以直接在命令行中继续解码，或者在以下的链接网站进行解码。

aaencode demo

aaencode - Encode any JavaScript program to Japanese style emoticons (^_^)

Enter JavaScript source:

```
alert("zkaq")
```

aaencode

```
ﾟωﾟﾉ= /｀ｍ´）ﾉ ~┻━┻   //*´∇｀*/ ['_']; o=(ﾟｰﾟ)  =_=3; c=(ﾟΘﾟ) =(ﾟｰﾟ)-(ﾟｰﾟ); (ﾟДﾟ) =(ﾟΘﾟ)= (o^_^o)/
(o^_^o);(ﾟДﾟ)={ﾟΘﾟ: '_' ,ﾟωﾟﾉ : ((ﾟωﾟﾉ==3) +'_') [ﾟΘﾟ] ,ﾟｰﾟﾉ :(ﾟωﾟﾉ+ '_')[o^_^o -(ﾟΘﾟ)] ,ﾟДﾟﾉ:((ﾟｰﾟ==3)
+'_')[ﾟｰﾟ] }; (ﾟДﾟ) [ﾟΘﾟ] =((ﾟωﾟﾉ==3) +'_') [c^_^o];(ﾟДﾟ) ['c'] = ((ﾟДﾟ)+'_') [ (ﾟｰﾟ)+(ﾟｰﾟ)-(ﾟΘﾟ) ];(ﾟДﾟ)
['o'] = ((ﾟДﾟ)+'_') [ﾟΘﾟ];(ﾟoﾟ)=(ﾟДﾟ) ['c']+(ﾟДﾟ) ['o']+(ﾟωﾟﾉ +'_')[ﾟΘﾟ]+ ((ﾟωﾟﾉ==3) +'_') [ﾟｰﾟ] + ((ﾟДﾟ)
+'_') [(ﾟｰﾟ)+(ﾟｰﾟ)]+ ((ﾟｰﾟ==3) +'_') [ﾟΘﾟ]+((ﾟｰﾟ==3) +'_') [(ﾟｰﾟ) - (ﾟΘﾟ)]+(ﾟДﾟ) ['c']+((ﾟДﾟ)+'_') [(ﾟｰﾟ)+
(ﾟｰﾟ)]+ (ﾟДﾟ) ['o']+((ﾟｰﾟ==3) +'_') [ﾟΘﾟ];(ﾟДﾟ) ['_'] =(o^_^o) [ﾟoﾟ] [ﾟoﾟ];(ﾟεﾟ)=((ﾟｰﾟ==3) +'_') [ﾟΘﾟ]+ (ﾟДﾟ)
.ﾟДﾟﾉ+((ﾟДﾟ)+'_') [(ﾟｰﾟ) + (ﾟｰﾟ)]+((ﾟｰﾟ==3) +'_') [o^_^o -ﾟΘﾟ]+((ﾟｰﾟ==3) +'_') [ﾟΘﾟ]+ (ﾟωﾟﾉ +'_') [ﾟΘﾟ];
(ﾟｰﾟ)+=(ﾟΘﾟ); (ﾟДﾟ)[ﾟεﾟ]='¥¥'; (ﾟДﾟ).ﾟΘﾟﾉ=(ﾟДﾟ+ ﾟｰﾟ)[o^_^o -(ﾟΘﾟ)];(oﾟｰﾟo)=(ﾟωﾟﾉ +'_')[c^_^o];(ﾟДﾟ) [ﾟ
oﾟ]='¥"';(ﾟДﾟ) ['_'] ( (ﾟДﾟ) ['_'] (ﾟεﾟ+(ﾟДﾟ)[ﾟoﾟ]+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ (ﾟΘﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+
((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟｰﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ ((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o) +(o^_^o))+
((o^_^o) - (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o) +(o^_^o))+ (ﾟｰﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+((ﾟｰﾟ) + (ﾟΘﾟ))+ (c^_^o)+ (ﾟДﾟ)
[ﾟεﾟ]+(ﾟｰﾟ)+ ((o^_^o) - (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((ﾟｰﾟ) + (o^_^o))+ ((o^_^o) - (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+
((ﾟｰﾟ) + (ﾟΘﾟ))+ (o^_^o)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ (ﾟｰﾟ)+ (ﾟΘﾟ)+ (ﾟДﾟ)[ﾟεﾟ]+(ﾟΘﾟ)+ ((o^_^o) +(o^_^o))+ (ﾟΘﾟ)+ (ﾟ
Дﾟ)[ﾟεﾟ]+(ﾟｰﾟ)+ ((o^_^o) - (ﾟΘﾟ))+ (ﾟДﾟ)[ﾟεﾟ]+((ﾟｰﾟ) + (ﾟΘﾟ))+ (ﾟΘﾟ)+ (ﾟДﾟ)[ﾟoﾟ]) (ﾟΘﾟ)) ('_');
```

[eval] [Permalink]

加解密地址：http://utf-8.jp/public/aaencode.html

换位加密

换位加密：栅栏密码、曲路密码、列位移密码

栅栏密码

栅栏密码（Rail-fence Cipher）就是把要加密的明文分成N个一组，然后把每组的第1个字符组合，每组第2个字符组合…每组的第N（最后一个分组可能不足N个）个字符组合，最后把他们全部连接起来就是密文。

明文：The quick brown fox jumps over the lazy dog
去空格：Thequickbrownfoxjumpsoverthelazydog
分组：Th eq ui ck br ow nf ox ju mp so ve rt he la zy do g
第一组：Teucbonojmsvrhlzdg
第二组：hqikrwfxupoeteay4
密文：Teucbonojmsvrh1zdghqikrwfxupoeteayo.

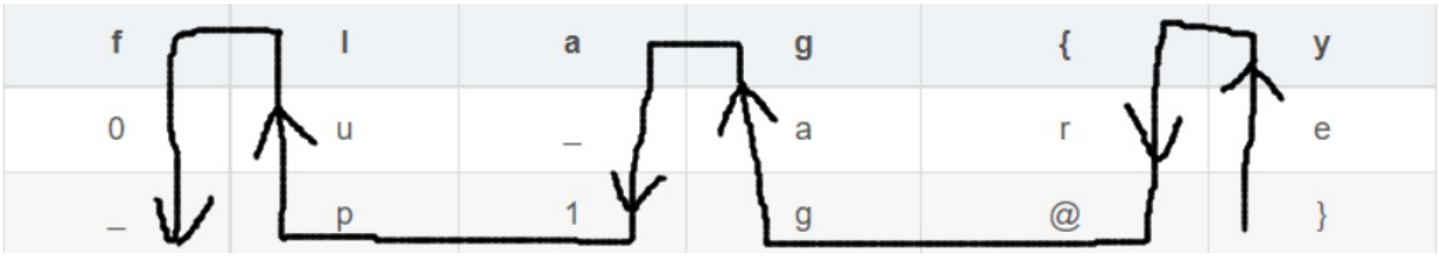
栅栏密码在线加解密：http://www.qqxiuzi.cn/bianma/zhalanmima.php

曲路密码

曲路密码（Curve Cipher）是一种换位密码需要事先双方约定密钥（也就是曲路路径）。
例如：秘钥就是行列数，但我感觉只需要列数就好了（可能是我理解有问题），使用行列数构成一个表格，将明文依次填入，例如：行列为6列3行， 明文为: “flag{y0uare_p1g@}”

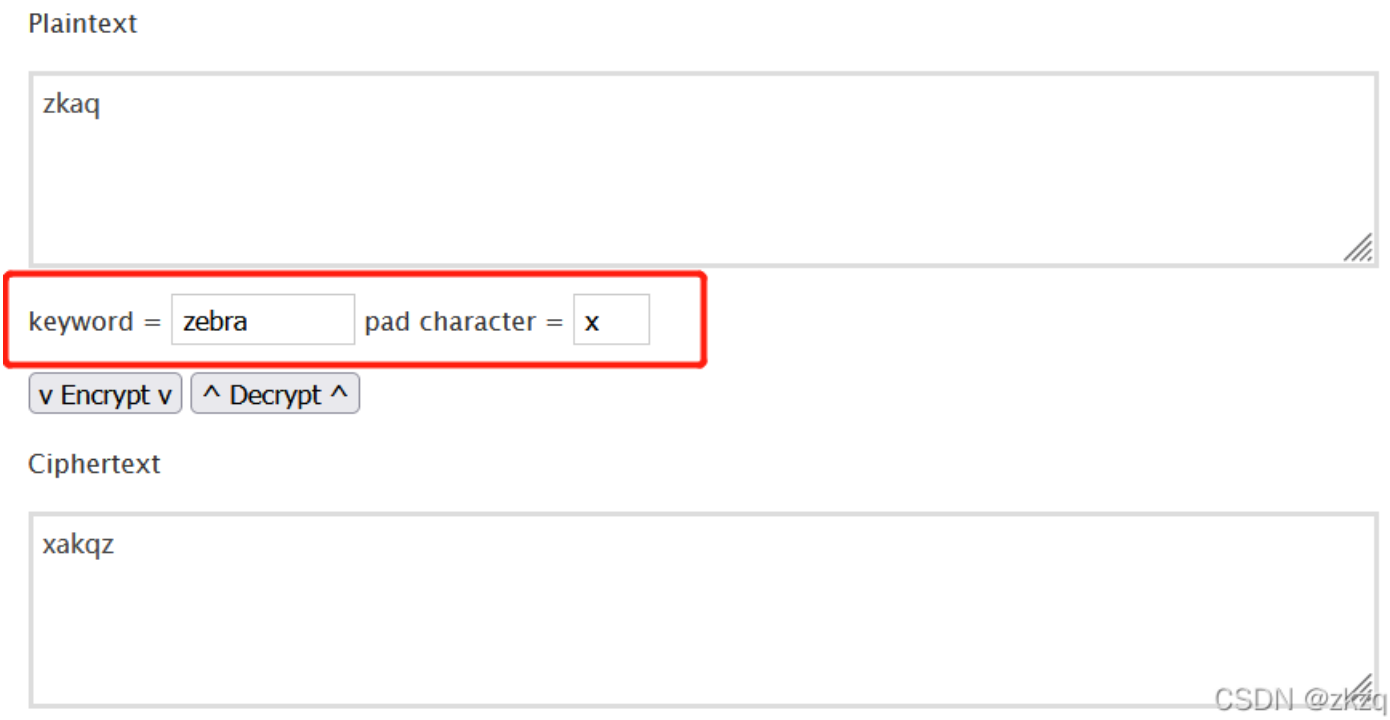
f	l	a	g	{	y
0	u	_	a	r	e
_	p	1	g	@	}

然后从最后一个字符像如下方式串起来即构成了密文：}ey{r@gaga_1pulf0



列位移密码

列移位密码（Columnar Transposition Cipher）是一种比较简单，易于实现的换位密码，通过一个简单的规则将明文打乱混合成密文。



列位移密码在线加解密：http://www.practicalcryptography.com/ciphers/classical-era/columnar-transposition/

替换加密

替换加密：凯撒密码、摩斯密码、ROT5/13/18/47、维吉尼亚密码、培根密码
凯撒密码
凯撒密码（Caesar Cipher或称恺撒加密、恺撒变换、变换加密、位移加密）是一种替换加密，明文中的所有字母都在字母表上向后（或向前）按照一个固定数目进行偏移后被替换成密文。例，当偏移量是3的时候，所有的字母A将被替换成D，B变成E，以此类推。

> 明文：The quick brown fox jumps over the lazy do
>
> 偏移量：1
>
> 密文：Uif rvjdl cspxo gpy kvnqt pwfs uif mbaz ephe

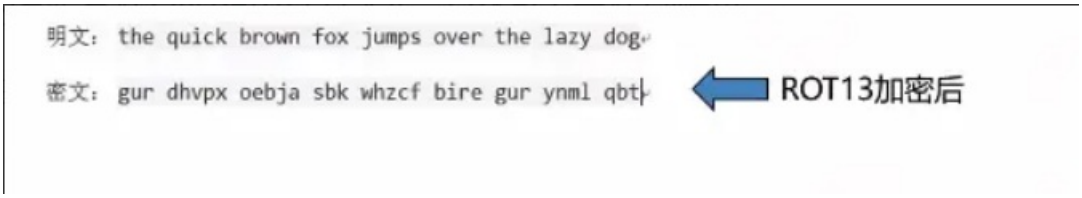
摩斯密码

摩尔斯电码（Morse Code）是由美国人萨缪尔·摩尔斯在1836年发明的一种时通时断的且通过不同的排列顺序来表达不同英文字母、数字和标点符号的信号代码，摩尔斯电码主要由以下5种它的代码组成：
1.点（.）2.划（-）3.每个字符间短的停顿（通常用空格表示停顿）4.每个词之间中等的停顿（通常用/划分）5.以及句子之间长的停顿

源文本: THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG
编码后:- … . / --.- …- … -.-. -.- / -… .-. — .-- -. / …-. — -…- / .— …- – .–. … / — …- . .-. / - … . / .-… .- --… -.-- / -… — --.
在线编码解码：http://ctf.ssleye.com/

ROT5/13/18/47

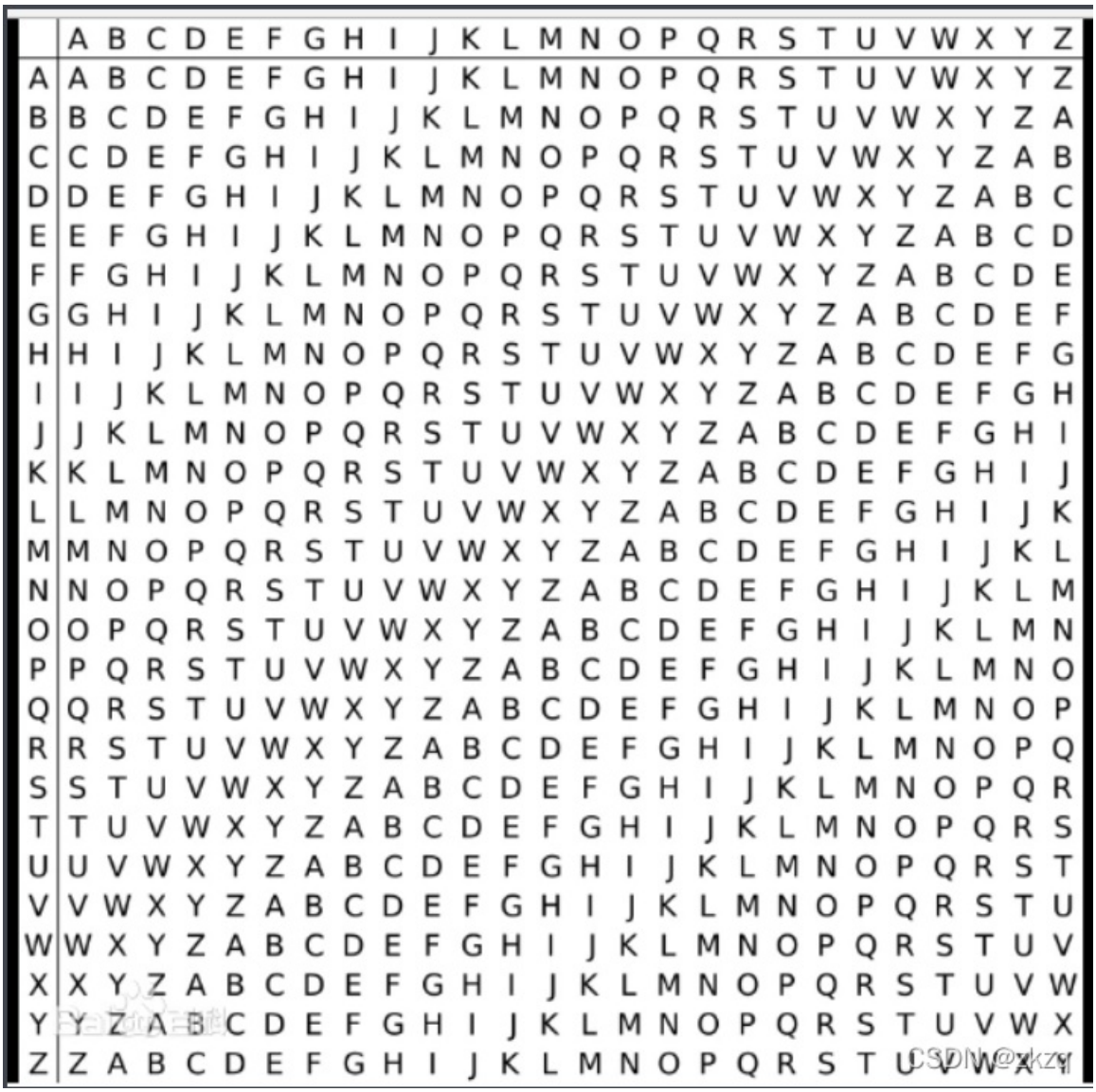
ROT5：只对数字进行编码，用当前数字往前数的第5个数字替换当前数字，例如当前为0，编码后变成5，当前为1，编码后变成6，以此类推顺序循环。
ROT13：只对字母进行编码，用当前字母往前数的第13个字母替换当前字母，例如当前为A，编码后变成N，当前为B，编码后变成O，以此类推顺序循环。
ROT18：这是一个异类，本来没有，它是将ROT5和ROT13组合在一起，为了好称呼，将其命名为ROT18。
ROT47：对数字、字母、常用符号进行编码，按照它们的ASCII值进行位置替换，用当前字符ASCII值往前数的第47位对应字符替换当前字符，例如当前为小写字母z，编码后变成大写字母K，当前为数字0，编码后变成符号。
用于ROT47编码的字符其ASCII值范围是33-126，具体可参考ASCII编码。



ROT5/13/18/47在线加解密：http://www.qqxiuzi.cn/bianma/ROT5-13-18-47.php

维吉尼亚密码

维吉尼亚密码（Vigenere Cipher）是在单一恺撒密码的基础上扩展出多表代换密码，根据密钥（当密钥长度小于明文长度时可以循环使用）来决定用哪一行的密表来进行替换，以此来对抗字频统计。
在一个凯撒密码中，字母表中的每一字母都会作一定的偏移，例如偏移量为3时，A就转换为了D、B转换为了E……而维吉尼亚密码则是由一些偏移量不同的恺撒密码组成。
为了生成密码，需要使用表格法。这一表格（如下图所示）包括了26行字母表，每一行都由前一行向左偏移一位得到。具体使用哪一行字母表进行编译是基于密钥进行的，在过程中会不断地变换。



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

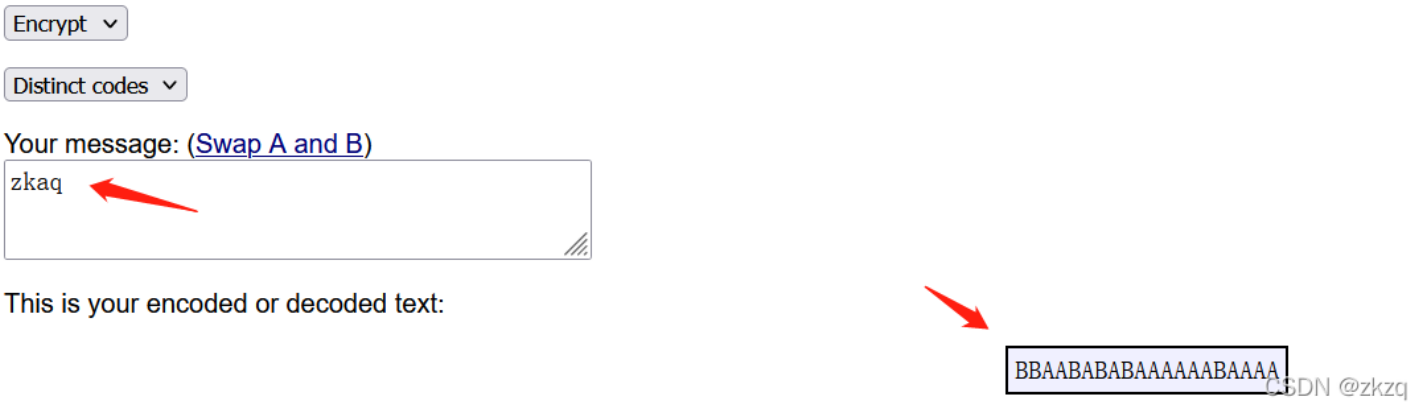
例如，假设明文为：
ATTACKATDAWN
选择某一关键词并重复而得到密钥，如关键词为LEMON时，密钥为：
LEMONLEMONLE
对于明文的第一个字母A，对应密钥的第一个字母L，于是使用表格中L行字母表进行加密，得到密文第一个字母L。类似地，明文第二个字母为T，在表格中使用对应的E行进行加密，得到密文第二个字母X。以此类推，可以得到：
明文：ATTACKATDAWN密钥：LEMONLEMONLE密文：LXFOPVEFRNHR
解密的过程则与加密相反。
例如：根据密钥第一个字母L所对应的L行字母表，发现密文第一个字母L位于A列，因而明文第一个字母为A。密钥第二个字母E对应E行字母表，而密文第二个字母X位于此行T列，因而明文第二个字母为T。以此类推便可得到明文。

培根密码

培根密码（Baconian Cipher）是一种替换密码，每个明文字母被一个由5字符组成的序列替换，最初的加密方式就是由AA和‘B’组成序列替换明文（所以你当然也可以用别的字母），比如字母‘D’替换成“aaabb”，以下是全部的对应关系（另一种对于关系是每个字母都有唯一对应序列，I和J与U/V各自都有不同对应序列）。

A/a	aaaaa	H/h	aabbb	O/o	abbba	V/v	babab
B/b	aaaab	I/i	abaaa	P/p	abbbb	W/w	babba
C/c	aaaba	J/j	abaab	Q/q	baaaa	X/x	babbb
D/d	aaabb	K/k	ababa	R/r	baaab	Y/y	bbaaa
E/e	aabaa	L/l	ababb	S/s	baaba	Z/z	bbaab
F/f	aabab	M/m	abbaa	T/t	baabb		
G/g	aabba	N/n	abbab	U/u	babaa		

加密者需使用两种不同字体，分别代表A和B。准备好一篇包含相同AB字数的假信息后，按照密文格式化假信息，即依密文中每个字母是A还是B分别套用两种字体。
解密时，将上述方法倒转。所有字体一转回A，字体二转回B，以后再按上表拼回字母。



培根密码在线加解密：http://rumkin.com/tools/cipher/baconian.php

其他密码

其他密码：MD5、SHA、仿射密码、当铺密码、playfair密码、曼彻斯特编码

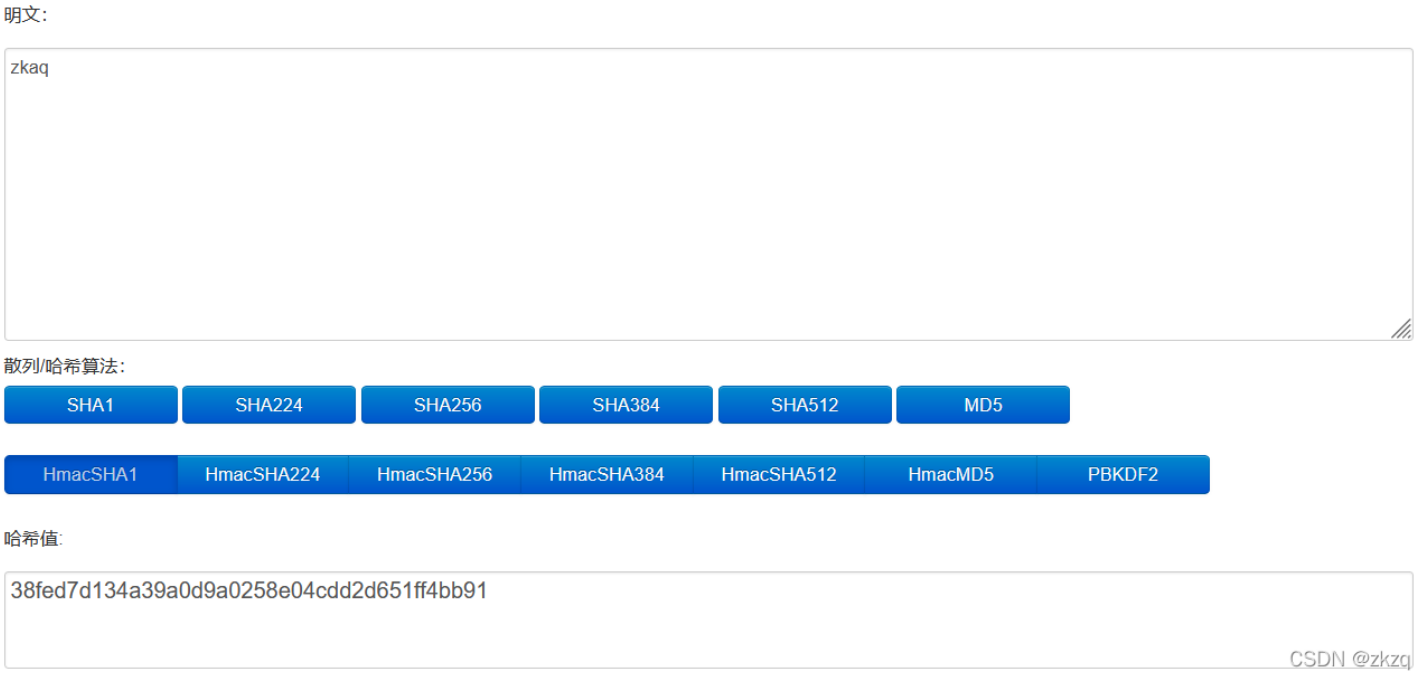
MD5

MD5（哈希算法）MD5以512位分组来处理输入的信息，且每一分组又被划分为16个32位子分组，经过了一系列的处理后，算法的输出由四个32位分组组成，将这四个32位分组级联后将生成一个128位散列值。MD5值分为16位和32位，通常MD5的值中最大是F，如，603F52D844017E83CA267751FEE5B61B。

MD5在线加解密地址：http://www.cmd5.com/

SHA

SHA（安全哈希算法）SHA-1是一种数据加密算法，该算法的思想是接收一段明文，然后以一种不可逆的方式将它转换成一段（通常更小）密文，也可以简单的理解为取一串输入码（称为预映射或信息），并把它们转化为长度较短、位数固定的输出序列即散列值（也称为信息摘要或信息认证代码）的过程。SAH的值通常是40位，最大值是F对强行攻击的安全性：最显著和最重要的区别是SHA-1摘要比MD5摘要长32位。使用强行技术，产生任何一个报文使其摘要等于给定报摘要的难度对MD5是2^{128}数量级的操作，而对SHA-1则是2^{160}数量级的操作。这样，SHA-1对强行攻击有更大的强度。



在线加密解密：http://tool.oschina.net/encrypt?type=2

仿射密码

概述：基本上和数学上的仿射变换类似y=ax+b，通过如此达到一一对应加密。
仿射变换加密加密过程：
加密算法：c=a*m+ b(mod n)
加密过程：
1.获取a，b（密钥），n（字符个数）
2.获取明文。
3.加密成密文，明文转换成各个字符所对应的数字，将所得数字带入上面的算法公式，得到数字再转换成对应的字符

解密过程：
算法：m=a^-1（m-b）（mod n）这里a^-1不是指倒数，而是a关于字符数量模的乘法可逆元，下面介绍一下乘法可逆元
乘法可逆元定义；
群G中任意一个元素a,都在G中有唯一的逆元a‘,具有性质aa’=a’a=e,其中e为群的单位元
简单来说就是
加密函数：Y=（AX+B）%26
解密函数：Y=（AX+B）%26，得到：X=（A的逆元）*（Y-B）%26

在线加密解密：https://aliyunvi.com/affine

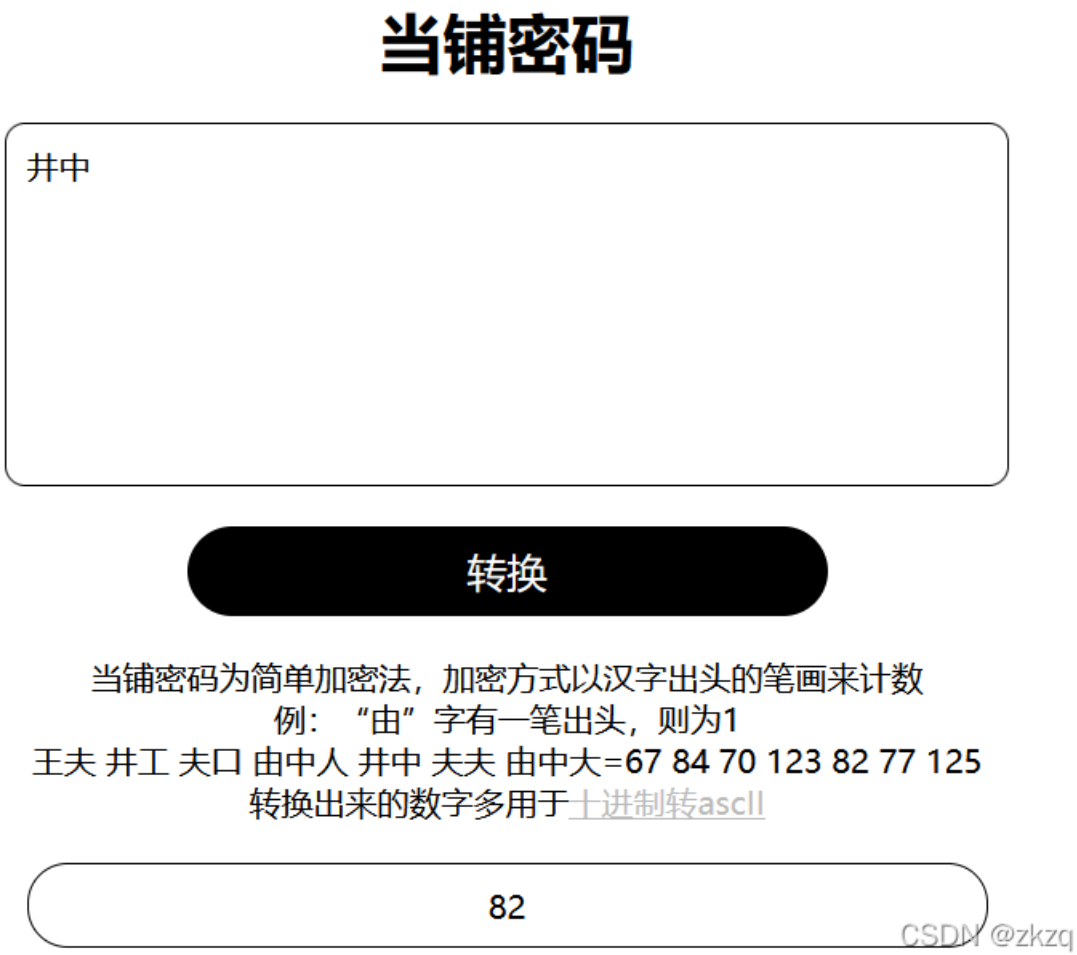
当铺密码

当铺密码是一种很有意思的密码，专门用来加密数字的，不需要密钥，明文信息包含在加密后的密文中。

它通过一个汉字中隐藏的信息：笔画数，来将汉字和数字关联起来，将汉字定义为明文，将数字定义为密文，加密是将数字映射到对应笔画的汉字，解密是将汉字按照笔画映射回数字。

有很多汉字的笔画数是相同的，所以可能会有多个明文（汉字）对应同一个密文（数字），当然这个主要是看汉字笔画映射表的选择，如果映射表只准备了9个汉字，每种笔画有一个汉字对应则是一对一的，否则是一对多的。一对一的话有个缺点就是如果要加密的明文中有重复数字，比如33，转换为“飞马”比“三三”更难总结出规律，而这种没有秘钥的加密方式重要的就是隐藏自己的规律，所以一对多会更难被破译。

这个加密算法比较简单，实现的话只需要一个汉字笔画对照表，参考链接中会附上一个1~10笔画的汉字笔画映射表。



在线加密解密：http://dsb.ink/ctf/dangpu.html

参考：https://www.cnblogs.com/cc11001100/p/9357263.html

playfair密码

普莱费尔密码（英文：Playfair cipher 或 Playfair square）是一种使用一个关键词方格来加密字符对的加密法，1854年由一位名叫查尔斯·惠斯通（Charles Wheatstone）的英国人发明。
经莱昂·普莱费尔提倡在英国军地和政府使用。它有一些不太明显的特征：密文的字母数一定是偶数；任意两个同组的字母都不会相同，如果出现这种字符必是乱码和虚码。它使用方便而且可以让频度分析法变成瞎子，在1854到1855年的克里米亚战争和1899年的布尔战争中有广泛应用。但在1915年的一战中被破译了。编写分三步：1.编制密码表 2.整理明文 3.编写密文 构成部分：1.密钥 2.明文3.密文4.注明的某个字母代替的另一个字母。

C	D	F	M	T
R	O	H	N	U
A	G	I(J)	P	V
Z	B	K	Q	W
Y	E	L	S	X

它依据一个5*5的正方形组成的密码表来编写，密码表里排列有25个字母。如果一种语言字母超过25个，可以去掉使用频率最少的一个。如，法语一般去掉w或k，德语则是把i和j合起来当成一个字母看待。英语中z使用最少，可以去掉它。

第一步是编制密码表。在这个5*5的密码表中，共有5行5列字母。第一列（或第一行）是密钥,其余按照字母顺序。密钥是一个单词或词组，若有重复字母，可将后面重复的字母去掉。当然也要把使用频率最少的字母去掉。如：密钥是Live and learn,去掉后则为liveandr。如果密钥过长可占用第二列或行。同时字母I和J会被当成一个字母。
如密钥crazy dog，可编制成

第二步整理明文。将明文每两个字母组成一对。如果成对后有两个相同字母紧挨或最后一个字母是单个的，就插入一个字母X（或者Q）。
如，communist，应成为co,mx,mu,ni,st。
最后编写密文。对明文加密规则如下：1、若p1 p2在同一行，对应密文c1 c2分别是紧靠p1 p2 右端的字母。其中第一列被看做是最后一列的右方。如，按照前表，ct对应dc；2 、若p1 p2在同一列，对应密文c1 c2分别是紧靠p1 p2 下方的字母。其中第一行被看做是最后一行的下方；3 、若p1 p2不在同一行，不在同一列，则c1 c2是由p1 p2确定的矩形的其他两角的字母（至于横向替换还是纵向替换要事先约好，或自行尝试）。如，按照前表，wh对应ku或uk。
如，依照上表，明文where there is life,there is hope.
可先整理为：WH ER ET HE RE IS LI FE TH ER EI SH OP EX
然后密文为：KU YO XD OL OY PL FK DL FU YO LG LN NG LY
将密文变成大写，然后几个字母一组排列。
如5个一组就是KUYOX DOLOY PLFKD LFUYO LGLNN GLY

解密：Playfair解密算法首先将密钥填写在一个5*5的矩阵中（去Q留Z），矩阵中其它未用到的字母按顺序填在矩阵剩余位置中，根据替换矩阵由密文得到明文。对密文解密规则如下：1 、若c1 c2在同一行，对应明文p1 p2分别是紧靠c1 c2 左端的字母。其中最后一列被看做是第一列的左方。2 、若c1 c2在同一列，对应明文p1 p2分别是紧靠c1 c2 上方的字母。其中最后一行被看做是第一行的上方。3、 若c1 c2不在同一行，不在同一列，则p1 p2是由c1 c2确定的矩形的其他两角的字母。其实就是反其道而行之。另外PlayFair解密算法不能解决字母中连续出现'XX'的情况，但是在英语中很少有连续两个XX的字母，所以不太影响PlayFair算法的使用。

一： 密钥：boys and girls are students (按列填充密钥, 不在同一行或列的密文，采用纵向替换)
密文GUUID BCYZC YOETX UUGAB EPBCE TDIUV LDDSB KRPRD IRUW
明文(原文)：It is not a problem. It is a challenge. Enjoy facing it.
二：密钥：father (按列填充密钥, 不在同一行或列的密文，采用横向替换)
UIHEK INREL EFPVI CMRYM ORROQ GQCLV OEPOH UXHPO IDKIH C
明文(原文)：Nothing in the world is difficult, if you set your mind to it.

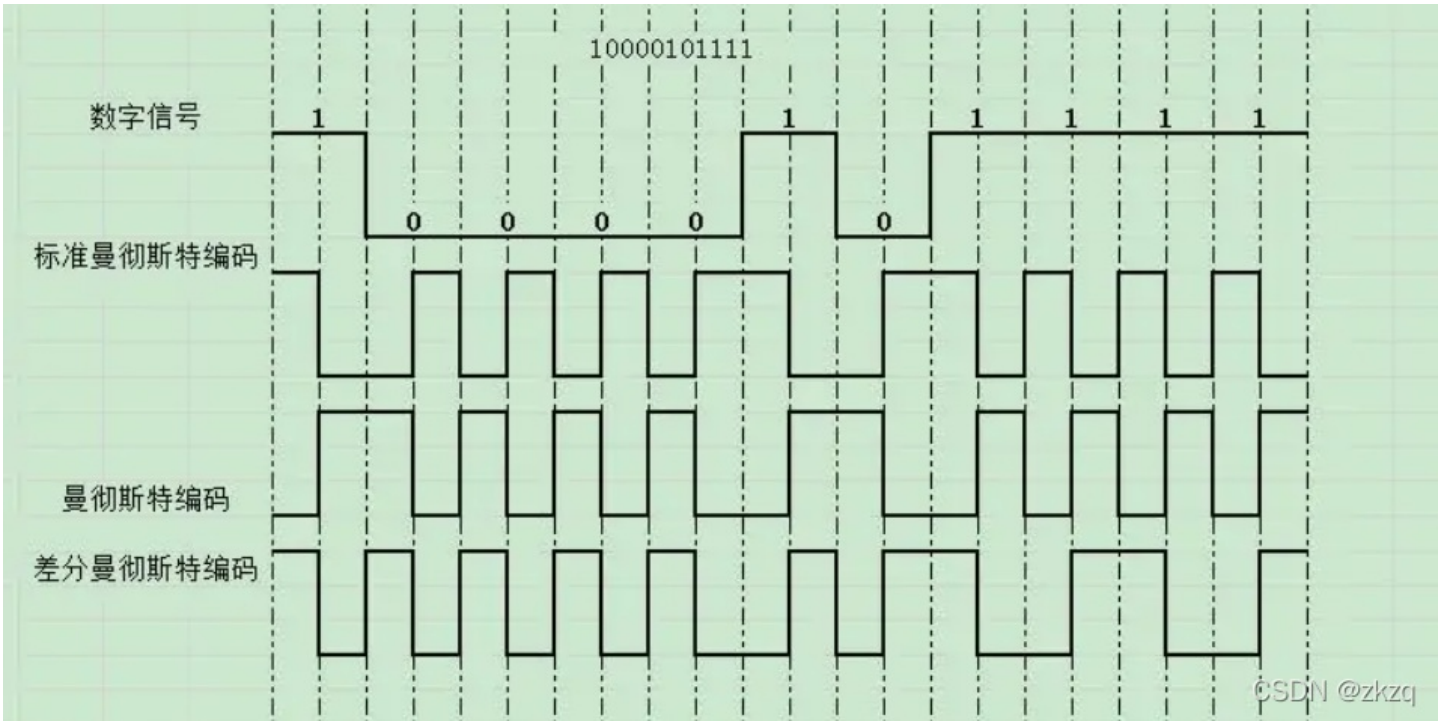
曼彻斯特编码

曼彻斯特编码(Manchester)又称裂相码、同步码、相位编码，是一种用电平跳变来表示1或0的编码方法，其变化规则很简单，即每个码元均用两个不同相位的电平信号表示，也就是一个周期的方波，但0码和1码的相位正好相反。由于曼彻斯特码在每个时钟位都必须有一次变化，因此，其编码的效率仅可达到50%左右

编码规则：在曼彻斯特编码中，每一位的中间有一跳变，位中间的跳变既作时钟信号，又作数据信号。曼彻斯特编码有两种相反的约定。其中的第一种约定由1949年由GE托马斯（GE Thomas）首次出版，随后有众多作家使用，例如，安迪·塔南鲍姆（Andy Tanenbaum）。它指定对于0位，信号电平将为低高电平（假设对数据进行幅度物理编码）-在位周期的前半段为低电平，在后半段为高电平。对于1位，信号电平将为高-低。第二种约定也被众多作者使用（例如William Stallings），IEEE 802.4（令牌总线）和IEEE 802.3（以太网）标准的低速版本所遵循。它指出逻辑0由高-低信号序列表示，逻辑1由低-高信号序列表示。其中非常值得注意的是，在每一位的”中间”必有一跳变，根据此规则，可以得出曼彻斯特编码波形图的画法。例如：传输二进制信息0，若将0看作一位，我们以0为中心，在两边用虚线界定这一位的范围，然后在这一位的中间画出一个电平由高到低的跳变。后面的每一位以此类推即可画出整个波形图。

编码原理：曼彻斯特编码是将时钟和数据包含在信号流中，在传输代码信息的同时，也将时钟同步信号一起传输到对方。曼彻斯特编码的每一个码元都被调制成两个电平，所以数据传输速率只有调制速率的1/2。

解码：有保证的跳变的存在使信号可以自计时，也可以使接收器正确对准。接收器可以识别它是否在半个比特周期内未对齐，因为在每个比特周期内将不再总是存在过渡。与更简单的NRZ编码方案相比，这些好处的代价是带宽需求增加了一倍。




安全实战技能学习# 配套攻防靶场hack.zkaq....
录播 1#学黑客难？安全黑客工程师零基础入...
65分钟
录播 2#漏洞之王-实战教你获得管理员账号...
71分钟
录播 3#xss还能这么用！-无密码登陆目标账户
73分钟
录播 4#学会这些小技巧，萌新也能轻松找漏洞
75分钟
录播 5#实战-手把手教你写木马控制对方的...
75分钟
录播 6#手把手教学解密黑客如何拿下目标最...
67分钟
录播 7#手把手教学解密黑客如何拿下目标最...
67分钟
录播 8#黑客的就业宝典&职业分析推荐.mp4
77分钟
https://blog.csdn.net/hackzkaq