第一章 古典密码学

苏 明

密码学原理与实践
（第三版）
Cryptography Theory and Practice
Third Edition
CRYPTOGRAPHY
THEORY AND PRACTICE
[加] Douglas R. Stinson 著
冯登国 等译

概览

- **1．1 几个简单的密码体制**

  1．1．1 移位密码
  1．1．2 代换密码
  1．1．3 仿射密码
  1．1．4 维吉尼亚密码
  1．1．5 希尔密码
  1．1．6 置换密码
  1．1．7 流密码

概览

- **1．2 密码分析**

  1．2．1 仿射密码的密码分析

  1．2．2 代换密码的密码分析

  1．2．3 维吉尼亚密码的密码分析

  1．2．4 希尔密码的密码分析

  1．2．5 LFSR流密码的密码分析

简单的密码体制

- 密码学设计的初衷： 解决安全通信

- Alice, Bob 安全传输信息
  - ✓ 攻击者(Oscar/Eve)
  - ✓ 明文(Plaintext)
  - ✓ 密文(Ciphertext)
  - ✓ 密钥(Key)

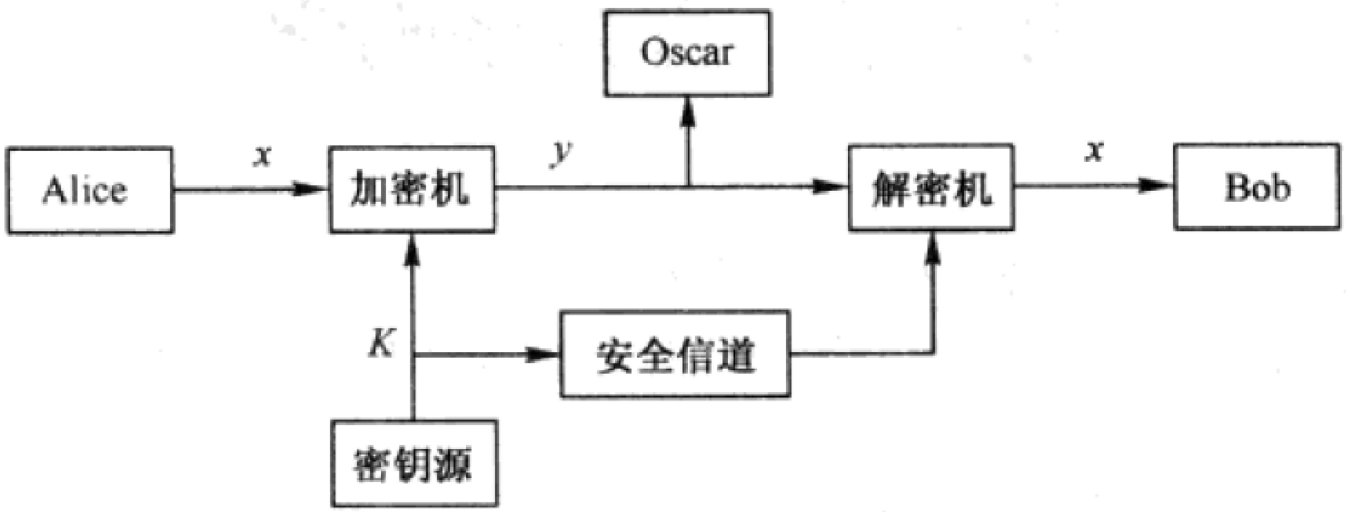
密码体制定义

定义 1.1 一个密码体制是满足以下条件的五元组$(\mathcal{P},\mathcal{C},\mathcal{K},\mathcal{E},\mathcal{D})$：

1. $\mathcal{P}$ 表示所有可能的明文组成的有限集。
2. $\mathcal{C}$ 表示所有可能的密文组成的有限集。
3. $\mathcal{K}$ 代表密钥空间，由所有可能的密钥组成的有限集。
4. 对每一个 $K\in\mathcal{K}$，都存在一个加密规则 $e_K\in\mathcal{E}$ 和相应的解密规则 $d_K\in\mathcal{D}$。并且对每对 $e_K:\mathcal{P}\to\mathcal{C}$，$d_K:\mathcal{C}\to\mathcal{P}$，满足条件：对每一个明文 $x\in\mathcal{P}$，均有 $d_K(e_K(x))=x$。

通信信道模型

记号：Plaintext Ciphertext $\boldsymbol{y} = y_1 y_2 \cdots y_n$



移位密码

$Z_m = \{0,1,\ldots,m-1;+,*\}$ 是一个环

1. 对加法运算封闭：对任意的$a,b\in \mathbb{Z}_m$，有$a+b\in \mathbb{Z}_m$
2. 加法运算满足交换律：对任意的$a,b\in \mathbb{Z}_m$，有$a+b=b+a$
3. 加法运算满足结合律：对任意的$a,b,c\in \mathbb{Z}_m$，有$(a+b)+c=a+(b+c)$
4. 0 是加法单位元：对任意的$a\in \mathbb{Z}_m$，有$a+0=0+a=a$
5. 任何元素存在加法逆元：a的逆元为$m-a$，因为$a+(m-a)=(m-a)+a=0$
6. 对乘法运算封闭：对任意的$a,b\in \mathbb{Z}_m$，有$ab\in \mathbb{Z}_m$
7. 乘法运算满足交换律：对任意的$a,b\in \mathbb{Z}_m$，有$ab=ba$
8. 乘法运算满足结合律：对任意的$a,b,c\in \mathbb{Z}_m$，有$(ab)c=a(bc)$
9. 1 是乘法单位元：对任意的$a\in \mathbb{Z}_m$，有$a\times 1=1\times a=a$
10. 乘法和加法之间存在分配律：对任意的$a,b,c\in \mathbb{Z}_m$，有$(a+b)c=(ac)+(bc)$，$a(b+c)=(ab)+(ac)$

移位密码

密码体制 1.1　移位密码

令 $\mathcal{P}=\mathcal{C}=\mathcal{K}=\mathbb{Z}_{26}$。对 $0\leqslant K\leqslant 25$，任意 $x,y\in\mathbb{Z}_{26}$，定义

$$e_K(x)=(x+K)\bmod 26$$

和

$$d_K(y)=(y-K)\bmod 26$$

凯撒密码(Caesar Cipher)

移位密码

- 明文用小写字母表示，密文用大写字母

例 1.1 wewillmeetatmidnight $K=11$

HPHTWWXPPELEXTOYTRSE

移位密码

例 1.2 JBCRCLQRWCRVNBJENBWRWN **K=?**

穷举攻击

```
jbcrclqrwcrvnbjenbwrwn
iabqbkpqvbqumaidmavqvm
hzapajopuaptlzhclzupul
gyzozinotzoskygbkytotk
fxynyhmnsynrjxfajxsnsj
ewxmxglmrxmqiweziwrmri
dvwlwfklqwlphvdyhvqlqh
cuvkvejkpvkogucxgupkpg
btujudijoujnftbwftojof
astitchintimesavesnine
```

密钥空间小：expected trials：26/2=13 次

代换密码

- Substitution Cipher

密码体制 1.2　代换密码

令 $\mathcal{P}=\mathcal{C}=\mathbb{Z}_{26}$。$\mathcal{K}$是由 26 个数字 0,1⋯,25 的所有可能的置换组成。对任意的置换 $\pi\in\mathcal{K}$，定义

$$e_\pi(x)=\pi(x)$$

和

$$d_\pi(y)=\pi^{-1}(y)$$

这里 π^{-1} 表示置换 π 的逆置换。

代换密码

- 一个密钥对应于26个字母的一个置换

密钥空间的大小？

$$26! \approx 4 * 10^{26}$$

Attacks?

仿射密码

- Affine Cipher

加密函数 $e(x) = ax + b \; mod \; 26, \; a, b \in Z_{26}$

为了保证**加解密的可逆性**，要求Z_{26}上定义的映射$\psi_a: x \rightarrow ax$是一一映射.

因此，要求$\gcd(a, 26) = 1$

仿射密码

- 一般的，Z_m中所有与m互素的数的个数记为$\phi(m)$(欧拉函数)

定理 1.2 假定

$$m=\prod_{i=1}^{n} p_i^{e_i}$$

这里 p_i 均为素数且互不相同，$e_i>0, 1\leqslant i\leqslant n$。则

$$\phi(m)=\prod_{i=1}^{n}(p_i^{e_i}-p_i^{e_i-1})$$

仿射密码

密码体制 1.3　仿射密码

令 $\mathcal{P}=\mathcal{C}=\mathbb{Z}_{26}$，且

$$\mathcal{K}=\{(a,b)\in\mathbb{Z}_{26}\times\mathbb{Z}_{26}:\underline{\gcd(a,26)=1}\}$$

对任意的 $K=(a,b)\in\mathcal{K}$， $x,y\in\mathbb{Z}_{26}$，定义

$$e_K(x)=(ax+b)\bmod 26$$

和

$$d_K(y)=a^{-1}(y-b)\bmod 26$$

维吉尼亚密码

- Vigenère Cipher

密码体制 1.4　维吉尼亚密码

设 m 是一个正整数。定义 $\mathcal{P}=\mathcal{C}=\mathcal{K}=(\mathbb{Z}_{26})^m$。对任意的密钥 $K=(k_1,k_2,\cdots,k_m)$，定义

$$e_K(x_1,x_2,\cdots,x_m)=(x_1+k_1,x_2+k_2,\cdots,x_m+k_m)$$

和

$$d_K(y_1,y_2,\cdots y_m)=(y_1-k_1,y_2-k_2,\cdots,y_m-k_m)$$

以上所有的运算都是在 $\mathbb{Z}_{26}$ 上进行。

维吉尼亚密码

- 有“分组”加密的味道

一个字母可以被映射为m个字母中的某一个：***多表代换密码体制***

- Key Space？

  26^m

希尔密码

- Hill Cipher（1929, by Leser S. Hill)

- 线性变换：K取一个m*m的矩阵

$$y = e_K(x) = (y_1, y_2, \cdots, y_m)$$

$$(y_1, y_2, \cdots, y_m) = (x_1, x_2, \cdots, x_m)\begin{pmatrix} k_{1,1} & k_{1,2} & \cdots & k_{1,m} \\ k_{2,1} & k_{2,2} & \cdots & k_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ k_{m,1} & k_{m,2} & \cdots & k_{m,m} \end{pmatrix}$$

希尔密码

- 加密： $y = xK$
- 解密： $x = yK^{-1}$

密码体制 1.5 希尔密码

设 $m \geqslant 2$ 为正整数，$\mathcal{P} = \mathcal{C} = (\mathbb{Z}_{26})^m$，且

$$\mathcal{K} = \{ 定义在 \mathbb{Z}_{26} 上的 m \times m 可逆矩阵 \}$$

对任意的密钥 $\boldsymbol{K}$，定义：

$$e_K(x) = x\boldsymbol{K}$$

和

$$d_K(y) = y\boldsymbol{K}^{-1}$$

以上运算都是在 $\mathbb{Z}_{26}$ 上进行的。

希尔密码

注记：

- 涉及到矩阵求逆

定理 1.3 设 $\boldsymbol{K}=(k_{i,j})$ 是一个定义在 $\mathbb{Z}_n$ 上的 $m\times m$ 矩阵。若 $\boldsymbol{K}$ 在 $\mathbb{Z}_n$ 上可逆，则有 $\boldsymbol{K}^{-1}=(\det\boldsymbol{K})^{-1}\boldsymbol{K}^{*}$，这里 $\boldsymbol{K}^{*}$ 为矩阵 $\boldsymbol{K}$ 的伴随矩阵。

- **KeySpace?**

$\approx O(p^{m^2})$ over Z_p

置换密码

定义在有限集 X 上的一个置换是一个双射函数 $\pi: X \to X$

密码体制 1.6 置换密码

令 m 为一正整数。$\mathcal{P}=\mathcal{C}=(\mathbb{Z}_{26})^m$，$\mathcal{K}$ 是由所有定义在集合 $\{1, 2, \cdots, m\}$ 上的置换组成。对任意的密钥(即置换) π，定义：

$$e_\pi(x_1, x_2, \cdots, x_m) = (x_{\pi(1)}, x_{\pi(2)}, \cdots, x_{\pi(m)})$$

和

$$d_\pi(y_1, y_2, \cdots, y_m) = (y_{\pi^{-1}(1)}, y_{\pi^{-1}(2)}, \cdots, y_{\pi^{-1}(m)})$$

其中 π^{-1} 为置换 π 的逆置换。

置换密码

- 置换密码是希尔密码的特殊情形

可以定义与置换π 相关的置换矩阵$K_\pi = \left(k_{i,j}\right)_{m*m}$

$$k_{i,j} = \begin{cases} 1 & 若 i = \pi(j) \\ 0 & 其他 \end{cases}$$

使用矩阵K_π为密钥的希尔密码 **等价于** 使用置换π为密钥的置换密码

流密码

- 连续的明文元素使用相同的密钥K加密

$$y = y_1 y_2 \cdots = e_K(x_1) e_K(x_2) \cdots$$

→分组加密(Block Cipher)

- 如果产生一个密钥流$z = z_1 z_2 \ldots$

$$y = y_1 y_2 \cdots = e_{z_1}(x_1) e_{z_2}(x_2) \cdots$$

→流加密(Stream Cipher)

流密码

定义 1.6 同步流密码是一个六元组$(\mathcal{P},\mathcal{C},\mathcal{K},\mathcal{L},\mathcal{E},\mathcal{D})$和一个函数$g$，并且满足如下条件：

1. $\mathcal{P}$是所有可能明文构成的有限集。
2. $\mathcal{C}$是所有可能密文构成的有限集。
3. 密钥空间$\mathcal{K}$为一有限集，由所有可能密钥构成。
4. $\mathcal{L}$是一个称之为密钥流字母表的有限集。
5. g是一个密钥流生成器。g使用密钥K作为输入，产生无限长的密钥流$z=z_1z_2\cdots$，这里$z_i\in\mathcal{L}$，$i\geqslant 1$。
6. 对任意的$z\in\mathcal{L}$，都有一个加密规则$e_z\in\mathcal{E}$和相应的解密规则$d_z\in\mathcal{D}$。并且对每个明文$x\in\mathcal{P}$，$e_z:\mathcal{P}\to\mathcal{C}$和$d_z:\mathcal{C}\to\mathcal{P}$是满足$d_z(e_z(x))=x$的函数。

教材观点：分组密码可看作是流密码的特殊情况

流密码

- 线性反馈移位寄存器（LFSR）
  高性能的软硬件实现

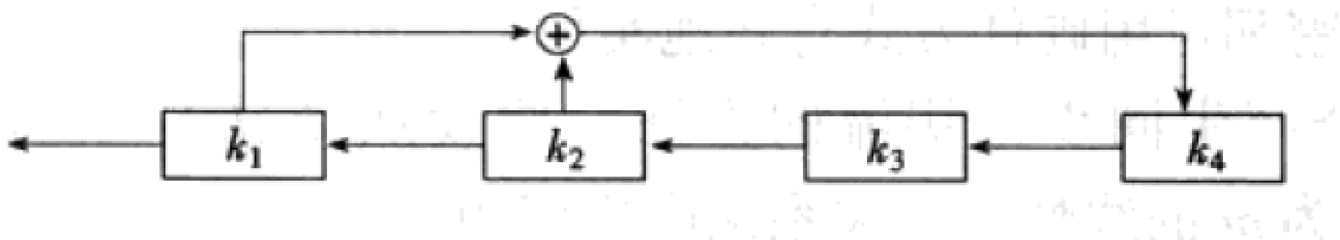


图 1.2　线性反馈移位寄存器

流密码

异步流密码： 密钥流的产生不但与密钥K有关，而且还与明文元素或者密文元素有关

密码体制 1.7 自动密钥密码

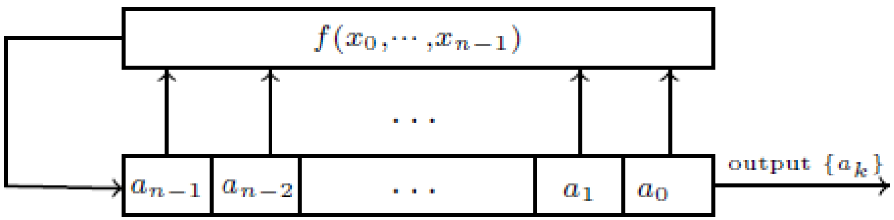
设 $\mathcal{P}=\mathcal{C}=\mathcal{K}=\mathcal{L}=\mathbb{Z}_{26}$，$z_1=K$，定义 $z_i=x_{i-1}$，$i\geqslant 2$。对任意的 $0\leqslant z\leqslant 25$，$x,y\in\mathbb{Z}_{26}$，定义

$$e_z(x)=(x+z)\bmod 26$$

和

$$d_z(y)=(y-z)\bmod 26$$

Feedback Shift Register Sequences



Feedback Function: $f(x_0, x_1, \cdots, x_{n-1}) = \sum c_{i_1 i_2 \cdots i_t} x_{i_1} x_{i_2} \cdots x_{i_t}$, $c_{i_1 i_2 \cdots i_t} \in \mathcal{F}_2$ where the sum runs through all subsets $\{i_1, \cdots, i_t\}$ of $\{0, 1, \cdots, n-1\}$ (a boolean function in n variables)

Initial state	$(a_0, a_1, \cdots, a_{n-1})$
State transition	$(a_0, a_1, \cdots, a_{n-1}) \longrightarrow (a_1, a_2, \cdots, a_n)$
Feedback bit	$a_n = f(a_0, a_1, \cdots, a_{n-1})$
Outputs	$a_0, a_1, \cdots, a_n, \cdots$
The recursive relation	$a_{k+n} = f(a_k, a_{k+1}, \cdots, a_{k+n-1}), k = 0, 1, \cdots$
The kth state in the FSR (after clocked k times)	$(a_k, a_{k+1}, \cdots, a_{k+n-1})$

Feedback Shift Register Sequences

- *Degree* of a boolean monomial

- *Degree* of a boolean function:
  maximum degree among the monomials in the boolean function

- Question: what is the total number of boolean functions with *n* variables?

Feedback Shift Register Sequences

- The output sequence of an FSR is called a linear feedback shift register sequence if

$$f(x_0, x_1, \cdots, x_{n-1}) = c_0 x_0 + c_1 x_1 + \cdots + c_{n-1} x_{n-1}, c_i \in \mathcal{F}_2.$$

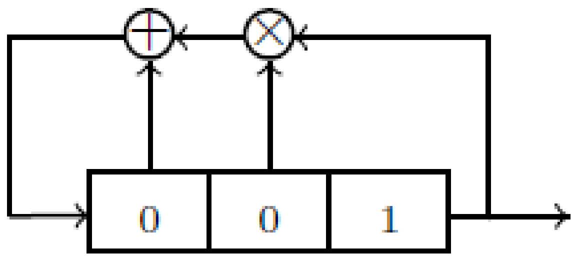
$$a_{k+n} = \sum_{i=0}^{n-1} c_i a_{k+i}, k = 0, 1, \cdots$$

- Otherwise, it is called a *nonlinear* feedback shift register (NLFSR) sequence.

A state Diagram of an FSR

- A state graph(diagram) of an n-stage FSR is a graph with all the possible states, represented as n-bit vectors, as vertexes and each edge is drawn from one state to its successor.
- *The feedback function* dominates the randomness **behavior** of the FSR, and an *initial state* determines randomness of the **output** sequence

A state Diagram of an FSR- Example 1



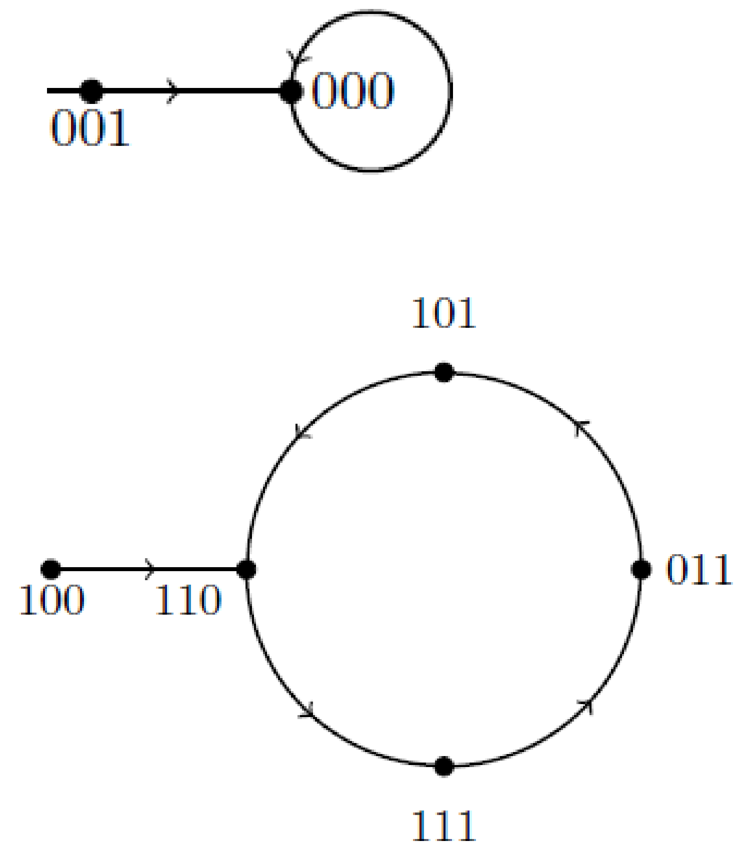
What is the corresponding **Feedback Function**?

$$x_0 x_1 + x_2$$

A state Diagram of an FSR- Example 1

- Transition States—$X_0X_1+X_2$

Current state	Next state
(x_2, x_1, x_0)	(x_2, x_1, x_0)
000	000
001	000
010	001
011	101
100	110
101	110
110	111
111	011

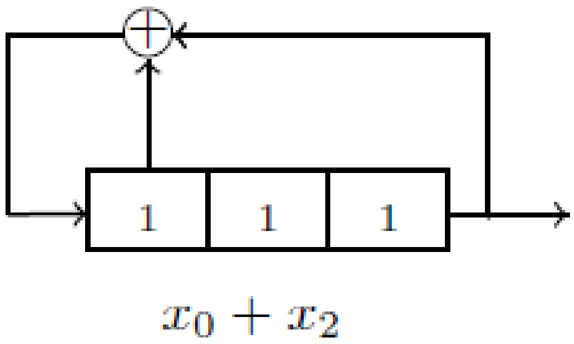


A state Diagram of an FSR- Example 1

- Initial State is closely related to **Output Sequence**

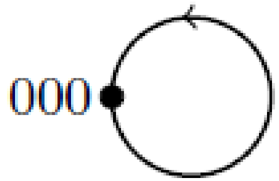
Initial state	Output sequence
(a_2, a_1, a_0)	$a_0, a_1, \cdots$
010	$0100000\cdots$
100	$0011101110111\cdots$
111	$111011101110\cdots$

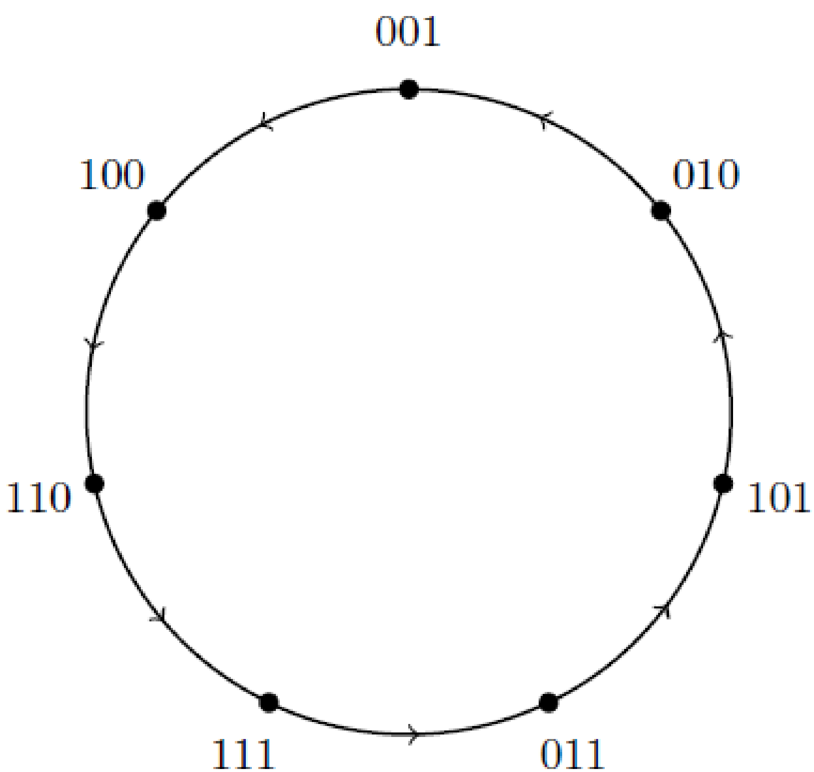
A state Diagram of an FSR-Example 2



Current state	Next state
(x_2, x_1, x_0)	(x_2, x_1, x_0)
000	000
001	100
010	001
011	101
100	110
101	010
110	111
111	011

A state Diagram of an FSR-Example 2





Initial state	Output sequence
(a_2, a_1, a_0)	$a_0, a_1, \cdots$
000	$000\cdots$
111	$11101001110100\cdots$

FSRs with Maximum Periods

- Facts 1: The output bits are closely related to **feedback function**, and **internal states**.
- Facts 2: The output bits are repeated once the internal states are repeated.
- Property 1: Any output of FSR is either ultimately periodic or periodic

FSRs with Maximum Periods

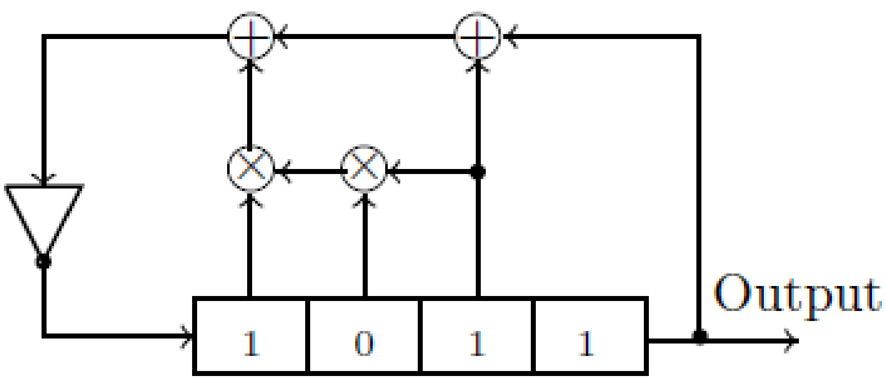
- What is the maximum period of an n-stage FSR sequence?

  2^n, referred to as a *de Bruijn sequence*

- What is the maximum period of an n-stage LFSR sequence?

  2^n-1, referred to as a *maximal length sequence*, or *m-sequence, or pseudonoise (PN) sequence.*

FSRs with Maximum Periods



Nonlinear feedback function	$f(x_0, x_1, x_2, x_3) = x_0 + x_1 + x_1x_2x_3 + 1$
The initial state	$(a_3, a_2, a_1, a_0) = 1011$
The output sequence	$a_0, a_1, \cdots = 1101100101000011\cdots$
Period	16

Research Problem: How to construct De bruijn Sequences, and how to construct m-sequences?

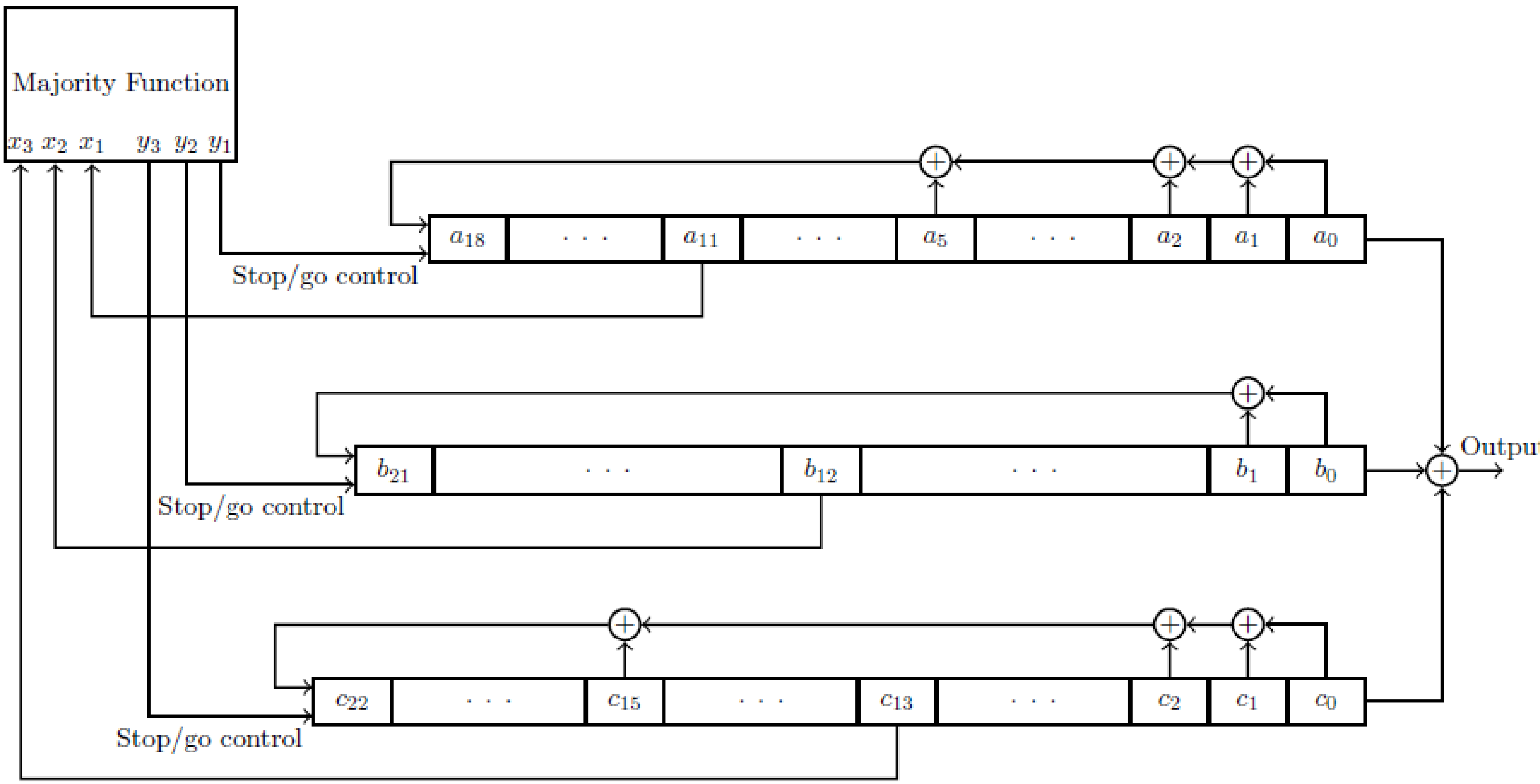
LFSR and m-sequence

- For an n-stage LFSR

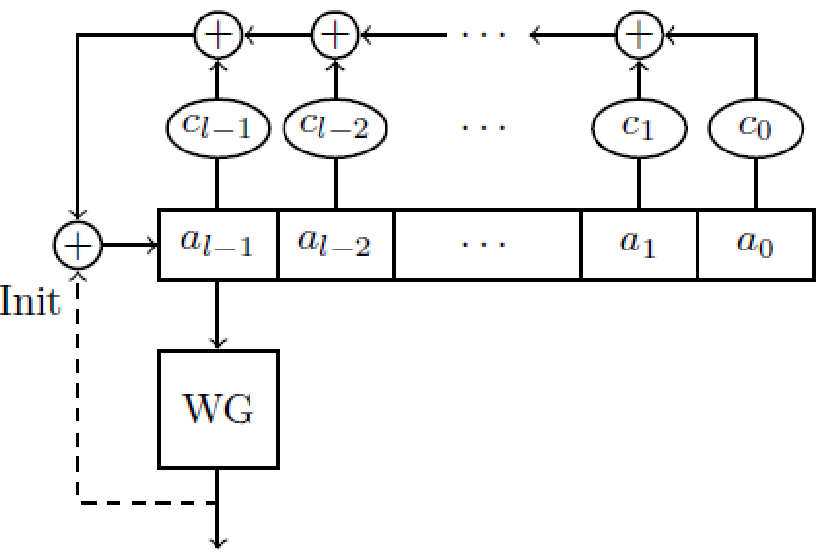
$$
\begin{array}{ccccccccccccc}
f(x_0, x_1, \cdots, x_{n-1}) & = & & & c_{n-1}x_{n-1} & + & \cdots & + & c_1x_1 & + & c_0x_0 \\
\updownarrow & & & & & & & & & & \\
f(x) & = & x^n & + & c_{n-1}x^{n-1} & + & \cdots & + & c_1x & + & c_0
\end{array}
$$

f(x) is called a characteristic polynomial of the LFSR

A5/1 in GSM system



WG Stream Cipher



Mathematical parameters	
m	Bit-width of LFSR
$g(x)$	Generating polynomial for $GF(2^m)$
$p(x) = \sum_{i=0}^{l} c_i x^i$	Primitive polynomial for LFSR
l	Degree of $p(x)$.

$$\begin{aligned} WGperm(x) &= t(x+1)+1 \\ t(x) &= x + x^{r_1} + x^{r_2} + x^{r_3} + x^{r_4} \end{aligned}$$

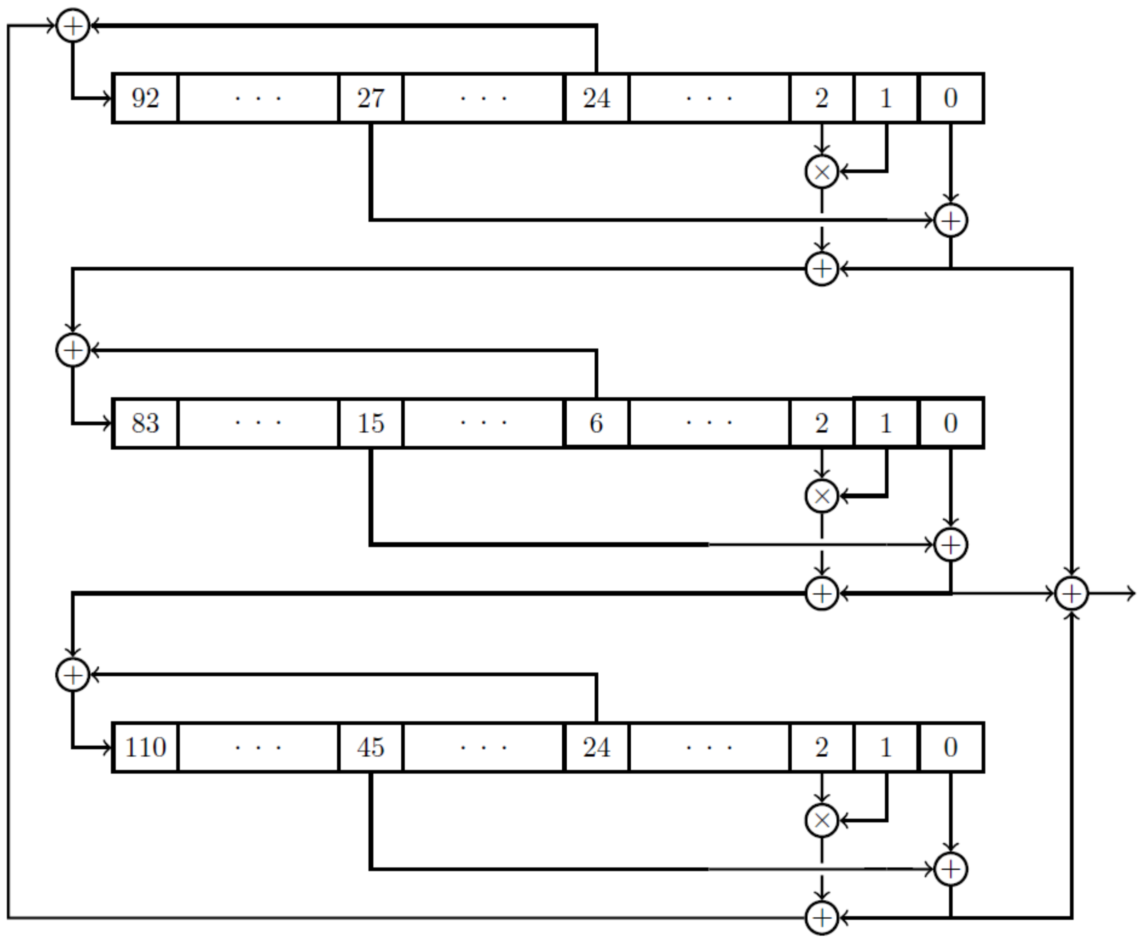
where $x \in GF(2^m)$

$$WG(x) = Tr(WGperm(x))$$

Find k such that $3k \equiv 1 \mod m$.

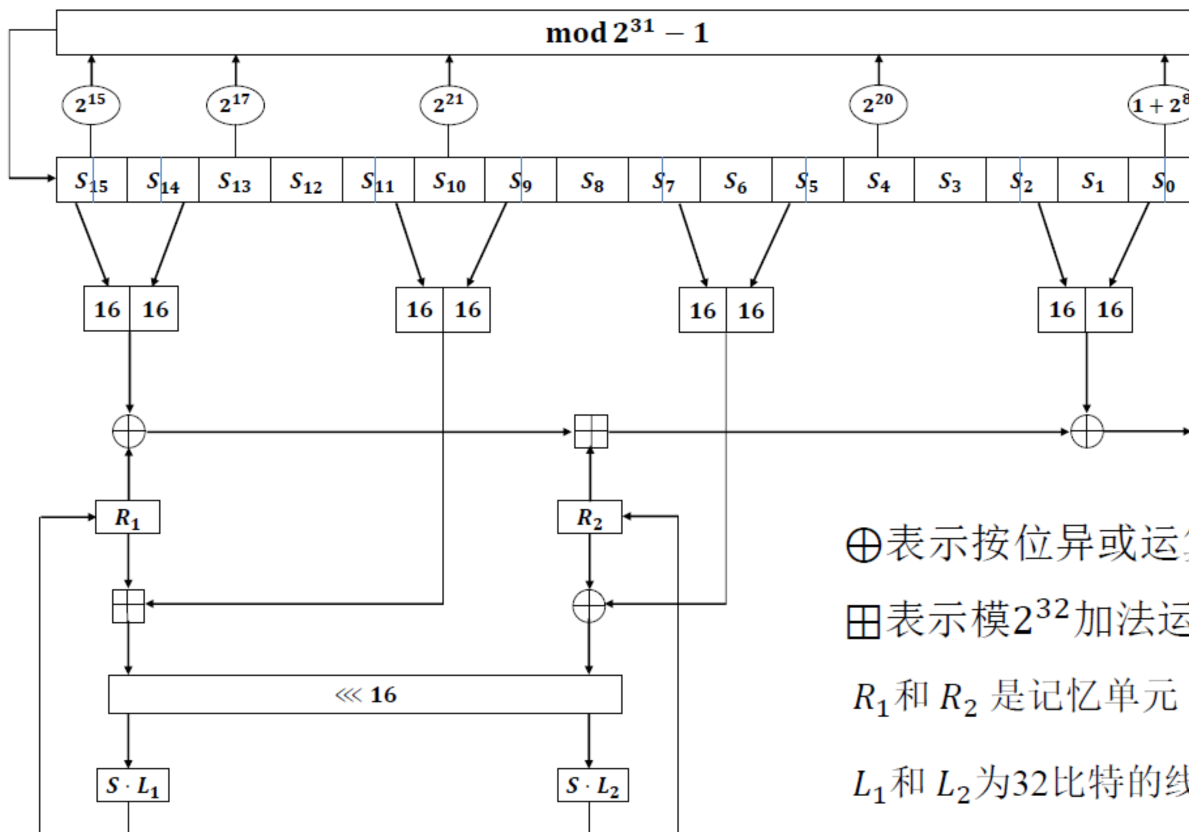
$$\begin{aligned} r_1 &= 2^k + 1 \\ r_2 &= 2^{2k} + 2^k + 1 \\ r_3 &= 2^{2k} - 2^k + 1 \\ r_4 &= 2^{2k} + 2^k - 1 \end{aligned}$$

Typical Stream Ciphers



Trivium generator

Typical Stream Ciphers



$\oplus$表示按位异或运算

$\boxplus$表示模2^{32}加法运算

R_1和 R_2 是记忆单元

L_1和 L_2为32比特的线性变换

S 为32比特的S-盒,

ZUC算法结构图

密码分析

- Kerckhoff假设：假设敌手知道所使用的密码体制

  - **唯密文攻击**(Ciphertext only attack):敌手只拥有密文串y;
  - **已知明文攻击**(Known plaintext attack): 敌手拥有明文串x以及对应的密文串y;
  - **选择明文攻击**(Chosen plaintext attack): 敌手可获得对*加密机*的临时访问权限，这样他能够选择一个明文串x，并可获得对应的密文串y;
  - **选择密文攻击**(Chosen ciphertext attack): 敌手可获得对*解密机*的临时访问权限，这样他能够选择一个密文串y，并可获得对应的明文串x;

密码分析

- 敌手的目标：确定使用的密钥

- 密码分析的出发点：
  往往利用了英文语言的统计特性

  推而广之: 利用对象先验的统计信息

密码分析

26 个英文字母出现的概率

字　母	概　率	字　母	概　率
A	0.082	N	0.067
B	0.015	O	0.075
C	0.028	P	0.019
D	0.043	Q	0.001
E	0.127	R	0.060
F	0.022	S	0.063
G	0.020	T	0.091
H	0.061	U	0.028
I	0.070	V	0.010
J	0.002	W	0.023
K	0.008	X	0.001
L	0.040	Y	0.020
M	0.024	Z	0.001

密码分析

- 二字母、三字母的频数
- 利用英文语言的统计特性，可以容易攻破： **移位密码 、代换密码**

仿射密码的密码分析

- 基于字母统计频率

例 1.10　利用仿射密码中获得如下密文:

FMXVEDKAPHFERBNDKRXRSREFMORUDSDKDVSHVUFEDKAPRKDLYEVLRHHRH

字　母	频　数	字　母	频　数
A	2	N	1
B	1	O	1
C	0	P	2
D	7	Q	0
E	5	R	8
F	4	S	3
G	0	T	0
H	5	U	2
I	0	V	4
J	0	W	0
K	5	X	2
L	2	Y	1
M	2	Z	0

仿射密码的密码分析

- e, t出现**频率最高**的字符
- 推测e-→R, t---→D; E, H, K

---→ 4a+b=17

19a+b=3, ***

---→ Valid K=(3,5)

---→ 验证解密结果

algorithmsarequitegeneraldefinitionsofarithmeticprocesses

代换密码的密码分析

- 频率分析（单、双字母）+经验判断

YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ

NDIFEFMDZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ

NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ

XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR

密文中出现的 26 个字母的频数统计

字母	频数	字母	频数
A	0	N	9
B	1	O	0
C	15	P	1
D	13	Q	4
E	7	R	10
F	11	S	3
G	1	T	2
H	4	U	5
I	5	V	5
J	11	W	8
K	1	X	6
L	0	Y	10
M	16	Z	20

代换密码的密码分析

- 推测出 $d_K(Z) = e$
- 双字母组DZ, ZW(4 times);
- 猜测$d_K(W)$=d
- R在密文中频繁出现， 猜测$d_K(R) = n$

```
------end---------e----ned---e------------
YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ

--------e----e---------n--d---en----e----e
NDIFEFMDZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ

-e---n------n------ed---e---e--ne-nd-e-e--
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ

-ed-----n-----------e----ed-------d---e--n
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR
```

代换密码的密码分析

- 根据双字母、三字母的搭配频率，依次推测

```
-----iend-----a-i-e-a-inedhi-e------a---i-
YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ

h-----i-ea-i-e-a---a-i-nhad-a-en--a-e-hi-e
NDIFEFMDZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ

he-a-n-----in-i----ed---e---e-ineandhe-e--
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ

-ed-a--inhi--hai--a-e-i--ed-----a-d--he--n
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR

o-r-riend-ro--arise-a-inedhise--t---ass-it
YIFQFMZRWQFYVECFMDZPCVMRZWNMDZVEJBTXCDDUMJ
hs-r-riseasi-e-a-orationhadta-en--ace-hi-e
NDIFEFMDZCDMQZKCEYFCJMYRNCWJCSZREXCHZUNMXZ

he-asnt-oo-in-i-o-redso-e-ore-ineandhesett
NZUCDRJXYYSMRTMEYIFZWDYVZVYFZUMRZCRWNZDZJJ

-ed-ac-inhischair-aceti-ted--to-ardsthes-n
XZWGCHSMRNMDHNCMFQCHZJMXJZWIEJYUCFWDJNZDIR
```

维吉尼亚密码的密码分析

首先猜测m

- Kasiski测试法(Friedrich Kasiski, 1863)

如果两个*相同的明文*段将**加密**成*相同的密文*段，假设它们的间距为δ，那么*很可能*

$$\delta \equiv 0 \ (mod\ m).$$

搜索长度为3的相同的密文段，得到类似的$\delta_1, \delta_2, \ldots,$ 那么**猜测m**为这些δ_i的*最大公因子的因子*。

维吉尼亚密码的密码分析

- Example

Plaintext:	the	man	and	the	woman	retrieved	the	letter	from	the	post	office
Key:	bea	dsb	ead	sbe	adsbe	adsbeadsb	ead	sbeads	bead	sbe	adsb	eadsbe
Ciphertext:	ULE	PSO	ENG	LII	WREBR	RHLSMEYWE	XHH	DFXTHJ	GVOP	LII	PRKU	SFIADI

the greatest common divisor of the distances between repeated sequences will yield the key length

维吉尼亚密码的密码分析

- 重合指数法(Index of Coincidence Method, William Friedman)

定义 1.7 设 $\boldsymbol{x} = x_1x_2\cdots x_n$ 是一条 n 个字母的串，$\boldsymbol{x}$ 的重合指数记为 $I_c(\boldsymbol{x})$，定义为 $\boldsymbol{x}$ 中两个随机元素相同的概率。

$$I_c(\boldsymbol{x}) = \frac{\sum_{i=0}^{25}\binom{f_i}{2}}{\binom{n}{2}} = \frac{\sum_{i=0}^{25} f_i(f_i-1)}{n(n-1)}$$

维吉尼亚密码的密码分析

- 假设x是英语文本串，设A,B, ...Z出现的概率为$p_0, p_1, \ldots, p_{25}$，那么

$$I_c(x) \approx \sum_{i=0}^{25} p_i^2 = 0.065$$

- 如果x是通过***单表代换***，那么$I_{c(x)}$不变。
- 但是对于定义在Z_{26}上的随机串，重合指数为

$$I_c \approx 26\left(\frac{1}{26}\right)^2 = \frac{1}{26} = 0.038$$

维吉尼亚密码的密码分析

- 因此猜测m, 分割串y为

$$\begin{aligned} \boldsymbol{y}_1 &= y_1 y_{m+1} y_{2m+1} \cdots \\ \boldsymbol{y}_2 &= y_2 y_{m+2} y_{2m+2} \cdots \\ &\vdots \\ \boldsymbol{y}_m &= y_m y_{2m} y_{3m} \cdots \end{aligned}$$

Related to k_1

......

Related to k_m

- 如果m正确，那么每一个$I_c(y_i) \approx 0.065$, $i = 1,2,\ldots,m$;　否则 $I_c(y_i) \approx 0.038$

维吉尼亚密码的密码分析

例 1.12 利用维吉尼亚密码获得如下密文：

```
CHREEVOAHMAERATBIAXXWTNXBEEOPHBSBQMQEQERBW
RVXUOAKXAOSXXWEAHBWGJMMQMNKGRFVGXWTRZXWIAK
LXFPSKAUTEMNDCMGTSXMXBTUIADNGMGPSRELXNJELX
VRVPRTULHDNQWTWDTYGBPHXTFALJHASVBFXNGLLCHR
ZBWELEKMSJIKNBHWRJGNMGJSGLXFEYPHAGNRBIEQJT
AMRVLCRREMNDGLXRRIMGNSNRWCHRQHAEYEVTAQEBBI
PEEWEVKAKOEWADREMXMTBHHCHRTKDNVRZCHRCLQOHP
WQAIIWXNRMGWOIIFKEE
```

密文串 CHR 共出现在 5 个位置，起始位置分别为 1，166，236，276 和 286，其距离分别为 165，235，275 和 285。

因此推测m=5；

当 $m=5$ 时重合指数分别为 0.063, 0.068, 0.069, 0.061, 0.072。

维吉尼亚密码的密码分析

- 接下来确定 $K=(k_1,k_2,\cdots,k_m)$

??

- 一种方法是通过移位的方式，匹配26个英文字符出现的频率；
- 但还有一种方法可以自动确定

维吉尼亚密码的密码分析

- y_i的长度为$n' = n/m$

- 频率分布 $\frac{f_0}{n'}, \frac{f_1}{n'}, \cdots, \frac{f_{25}}{n'}$ **移动k_i** $\frac{f_{k_i}}{n'}, \cdots, \frac{f_{25+k_i}}{n'}$

假设$0 \leqslant g \leqslant 25$，定义数值

$$M_g = \sum_{i=0}^{25} \frac{p_i f_{i+g}}{n'}$$

如果$g = k_i$，类似于前面重合指数的讨论，应该有

$$M_g \approx \sum_{i=0}^{25} p_i^2 = 0.065$$

如果$g \neq k_i$，则M_g一般应该小于 0.065

维吉尼亚密码的密码分析

- 利用M_g自动分析K

i	$M_g(y_i)$的值								
1	0.035	0.031	0.036	0.037	0.035	0.039	0.028	0.028	0.048
	$\boxed{0.061}$	0.039	0.032	0.040	0.038	0.038	0.044	0.036	0.030
	0.042	0.043	0.036	0.033	0.049	0.043	0.041	0.036	
2	$\boxed{0.069}$	0.044	0.032	0.035	0.044	0.034	0.036	0.033	0.029
	0.031	0.042	0.045	0.040	0.045	0.046	0.042	0.037	0.032
	0.034	0.037	0.032	0.034	0.043	0.032	0.026	0.047	
3	0.048	0.029	0.042	0.043	0.044	0.034	0.038	0.035	0.032
	0.049	0.035	0.031	0.035	$\boxed{0.066}$	0.035	0.038	0.036	0.045
	0.027	0.035	0.034	0.034	0.036	0.035	0.046	0.040	
4	0.045	0.032	0.033	0.038	$\boxed{0.060}$	0.034	0.034	0.034	0.050
	0.033	0.033	0.043	0.040	0.033	0.029	0.036	0.040	0.044
	0.037	0.050	0.034	0.034	0.039	0.044	0.038	0.035	
5	0.034	0.031	0.035	0.044	0.047	0.037	0.043	0.038	0.042
	0.037	0.033	0.032	0.036	0.037	0.036	0.045	0.032	0.029
	0.044	$\boxed{0.072}$	0.037	0.027	0.031	0.048	0.036	0.037	

$$K = (9, 0, 13, 4, 19)$$

希尔密码的密码分析

- **已知明文攻击**

1. 尝试m值
2. 获取 $x_j=(x_{1,j},x_{2,j},\cdots,x_{m,j})$ 对应的 $y_j=(y_{1,j},y_{2,j},\cdots,y_{m,j})$
3. 如果K不可逆，重新选择m个明-密文对。

希尔密码的密码分析

例 1.13 假设明文 friday 利用 $m=2$ 的希尔密码加密，得到的密文为 PQCFKU。

首先我们有 $e_K(5,17)=(15,16)$, $e_K(8,3)=(2,5)$, $e_K(0,24)=(10,20)$ 。使用头两个明-密文对，可得到矩阵方程

$$\begin{pmatrix} 15 & 16 \\ 2 & 5 \end{pmatrix} = \begin{pmatrix} 5 & 17 \\ 8 & 3 \end{pmatrix} \boldsymbol{K}$$

利用推论 1.4，容易计算

$$\begin{pmatrix} 5 & 17 \\ 8 & 3 \end{pmatrix}^{-1} = \begin{pmatrix} 9 & 1 \\ 2 & 15 \end{pmatrix}$$

因此

$$\boldsymbol{K} = \begin{pmatrix} 9 & 1 \\ 2 & 15 \end{pmatrix} \begin{pmatrix} 15 & 16 \\ 2 & 5 \end{pmatrix} = \begin{pmatrix} 7 & 19 \\ 8 & 3 \end{pmatrix}$$

可以使用第三个明-密文对进行验证。

LFSR流密码的密码分析

- 求解线性方程系统

$$z_{m+i} = \sum_{j=0}^{m-1} c_j z_{i+j} \bmod 2 \qquad i \geqslant 1$$

- 本质上需要确定m个系数

  m个线性方程可以用矩阵表达为

$$(z_{m+1}, z_{m+2}, \cdots, z_{2m}) = (c_0, c_1, \cdots, c_{m-1}) \begin{pmatrix} z_1 & z_2 & \cdots & z_m \\ z_2 & z_3 & \cdots & z_{m+1} \\ \vdots & \vdots & \ddots & \vdots \\ z_m & z_{m+1} & \cdots & z_{2m-1} \end{pmatrix}$$

如果n(所需比特数)>=2m, 理论上就可以确定这m个系数

LFSR流密码的密码分析

例 1.14 假设 Oscar 得到密文串

1011010111110010

和相应的明文串

0110011111111000

那么他能计算出密钥流比特

1101001000001010

如果攻击者知道密钥流采用*5级*（*m=5*）LFSR生成, 那么可以利用前10个比特得到

$$(0,1,0,0,0)=(c_0,c_1,c_2,c_3,c_4)\begin{pmatrix}1&1&0&1&0\\1&0&1&0&0\\0&1&0&0&1\\1&0&0&1&0\\0&0&1&0&0\end{pmatrix}$$

$$(c_0,c_1,c_2,c_3,c_4)=(1,0,0,1,0)$$

Berlekamp-Massey Algorithm

Procedure of BM Algorithm

Input: A sequence $\mathbf{a}^N = a_0, a_1, \cdots, a_{N-1}$.

Output: LFSR (f_{N-1}, l_{N-1})

Procedure $LFSR(\mathbf{a}^N)$

Initial Setting

Find k such that $a_i = 0, i = 0, 1, \cdots, k-1$ and $a_k = 1$.

Set

$$f(x) = x^{k+1} + 1$$
$$l = k + 1$$
$$g(x) = 1$$
$$a = k$$
$$b = 0$$
$$T(x) = 0$$

Main loop

for n **from** $k+1$ **to** $N-1$ **do**

(a) Compute: $d = a_n + \sum_{i=0}^{l-1} c_i a_{n-l+i}$

(b) **if** $d = 0$ **then** $b = b + 1$

(c) **if** $d \neq 0$ **and** $2l > n$ **then**

$$f(x) = f(x) - x^{a-b} g(x)$$
$$b = b + 1$$

(d) **if** $d \neq 0$ **and** $2l \leq n$ **then**

$$T(x) = f(x)$$
$$f(x) = x^{b-a} f(x) - g(x)$$
$$l = n + 1 - l$$
$$g(x) = T(x)$$
$$a = b$$
$$b = n - l + 1$$

return $(f(x), l)$

BM Algorithm & Attack

- Computational Complexity for a sequence of length *N*, $O(N^2)$

If $2l$ bits(l the *linear span*) are known in an N bit long sequence, we can recover the rest of bits.

For example, we can recover the whole key stream of a Period=2^{127}-1 *m-sequence* by only 2*127 bits!