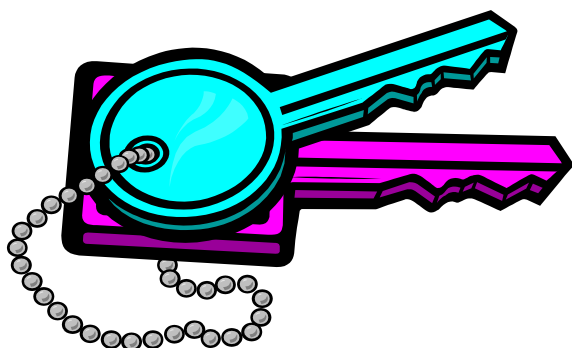


# 第二章 密码学

杨帆  
2021春



信息安全概论



# 大纲

1

密码学基础

2

古典密码

3

公钥密码





# 1 密码学基础

## 1.1 密码学历史



## 1.2 密码体制



# 1.1 密码学历史

## 1949以前

### 古典加密

在计算机出现以前，密码学与其说是一门科学不如说是一门艺术。

出现一些密码算法，加密机和简单的密码分析手段。

主要的加密对象是英文字母。

数据的安全依赖于算法的保密。



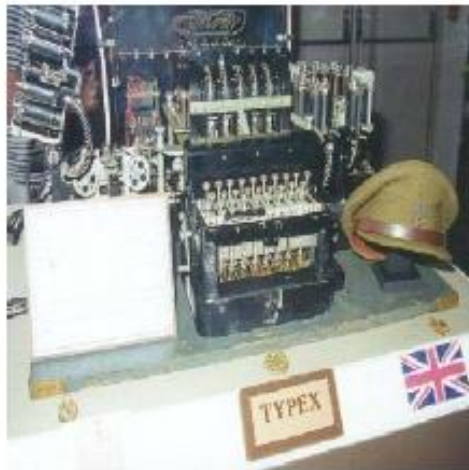
# Phaistos圆盘 (1700 BC)



TechWeb



# 20世纪早期的加密机





## ○ 1949~1976

- 香农在1949年发表了“保密系统的通信理论”，密码学正式成为一门学科。

计算机的发展使得密码由复杂运算构成。

数据的安全依赖于密钥的保密而不是算法的保密。







## ○ 1976以后

- Diffie和Hellman在1976年发表了“密码学的新方向”，提出了非对称密码体制。
- Rivest, Shamir和Adleman提出了RSA公钥密码算法。

公钥密码体制的保密通信不需要在发送者和接收者之间分配密钥，非常适用于数字签名。





## 1.2 密码体制

### ○ 定义

- 密码编制学

研究加密的方法

- 密码分析学

研究在不知道密钥的前提下解密密文的方法

- 密码学

包括密码编制学和密码分析学

### ○ 原理

伪装信息，防止非授权用户知道其真实含义



# 密码体制组成

- 明文 (Plaintext, Message)

- 原始信息

- 密文 Ciphertext

- 加密后的信息

- 密钥 Key

- 用于控制加解密的信息，只被发送者和接收者共享，必须被保密

$$K = \langle K_e, K_d \rangle$$

- 加密算法 (Encrypt)

- 将明文转换为密文

$$C = E(M, K_e)$$

- 解密算法 (Decrypt)

- 从密文中恢复明文

$$M = D(C, K_d)$$





密码分析者

明文 M

密文 C

密文 C

明文 M

源地址

love

加密

xlcm

信道

xlcm

解密

love

目的地址



$K_e$

加密密钥

安全信道

$K_d$

解密密钥

密钥



# 密码体制分类

## ○ 根据保密的内容

- 受限算法: 对算法的保密—古典密码
- 基于密钥的算法: 对密钥的保密—现代密码

## ○ 根据密钥的数量

- 哈希函数: 无密钥
- 秘密钥密码: 单密钥(对称密码/ 传统密码/ 单钥密码)
- 公钥密码: 双密钥- 公钥, 私钥 (非对称密码/ 公钥密码/ 双钥密码)

## ○ 根据明文的处理方式

- 分组密码: 输入和输出按照定长的块处理
- 流密码/序列密码: 输入和输出按照bit位或字符处理



# 对称密码& 非对称密码

## 对称密码

$$K_e = K_d$$

## 非对称密码

$$K_e \neq K_d \quad K_e \neq K_d$$

公开  $K_e$ , 保密  $K_d$



# ● ● ● | 优缺点

## ○ 传统密码优点

- 加解密速度快

## ○ 传统密码缺点

- 难以分配和管理密钥，实现数字签名困难

## ○ 公钥密码优点

- 易于分配和管理密钥，实现数字签名容易

## ○ 公钥密码缺点

- 难以产生密钥，加解密速度慢



## ● ● ● | 2 古典密码

**20.8.1.14.11/25.15.21/9/12.15.22.5/  
25.15.21/**

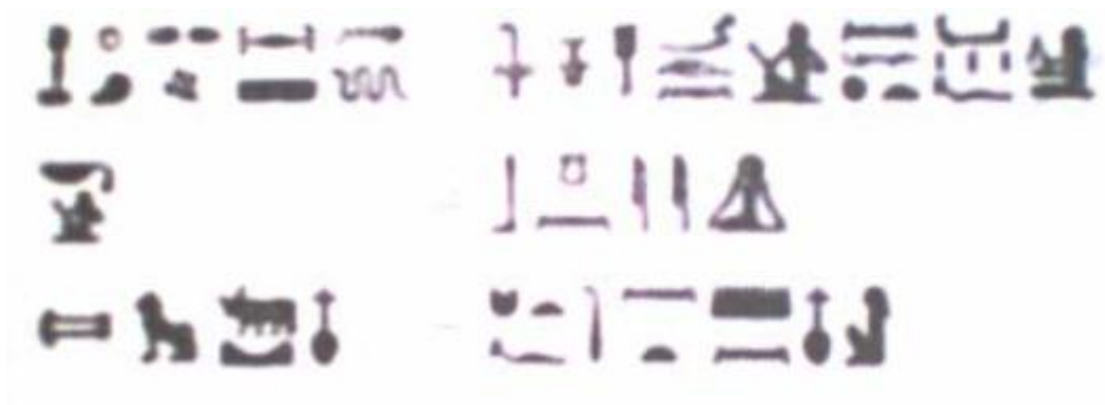
试着解密





## ○代替

- 明文字母被其他字母、数字或符号代替
- 明文被看作bit流，代替将明文bit流替换为密文bit流



被修改的象形文字  
1900 B.C. 古埃及



## ○ 置换

- 不改变实际使用的字母，而是通过重新排列字母顺序来隐藏信息



斯巴达棍  
500 B.C. 古希腊

# 古典密码分类

## ○ 代替密码

### ● 单字母代替

#### ➤ 单表

✓ 移位密码（加法密码）

✓ 置乱密码

#### ➤ 多表

✓ 维吉尼亚密码

✓ 转轮机

### ● 多字母代替

✓ Playfair

## ○ 置换密码

✓ 栅栏密码

✓ 列置换密码





**2.1 代替密码**



**2.2 置换密码**



## 2.1 代替密码

2.1.1 定义

2.1.2 移位密码

2.1.3 置乱密码

2.1.4 Vigenere密码

2.1.5 Playfair

2.1.6 转轮机



## 2.1.1 定义

### ○单字母代替

将每个明文字母独立的映射为密文字母

- 单表代替

一个明文字母表对应一个密文字母表

- 多表代替

一个明文字母表对应多个密文字母表

### ○多字母代替

字母映射以组的方式进行，依赖于其上下文的位置



## 2.1.2 移位密码

### 凯撒密码

- 2000年前，以凯撒大帝命名，最早的单表代替密码
- 将每个明文字母以其后的第三个字母代替
- 示例

meet me after the toga party  
PHHW PH DIWHU WKH WRJD SDUWB







- 定义变换如下

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
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

C: LQIRUPDWLRQ VHFUXULWB

P: ?

- 给每个字母定义一个序列号

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
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25


- 定义通用凯撒密码如下

$$C = E(m) = (m + k) \bmod 26$$

$$m = D(C) = (C - k) \bmod 26$$

$$K = 1 \sim 25$$



- 
- 只有26种可能的变换 (其中25种可用)
    - A被映射为B~Z (密钥空间=25)
  - 给定密文，尝试所有的移位可能
  - 通过25种穷举搜索能够攻破通用凯撒密码



# 穷举攻击

KEY	PHHW	PH	DIWHU	WKH	WRJD	SDUWB
1	oggv	og	chvgt	vjg	vqic	rctva
2	nffu	nf	bgufs	uif	uphb	qbsuz
3	meet	me	after	the	toga	party
4	ldds	ld	zesdq	sgd	snfz	ozqsx
5	kccr	kc	ydrpc	rfc	rmey	nyprw
6	jbbq	jb	xcqbo	qeb	qldx	mxoqv
7	iaap	ia	wbpan	pda	pkcw	lwnpu
8	hzzo	hz	vaozm	ocz	objv	kvmot
9	gyyn	gy	uznyl	nby	niau	julns
10	fxxm	fx	tymxk	max	mhzt	itkmr
11	ewwl	ew	sxlwj	lzw	lgys	hsjlg
12	dvvk	dv	rwkvi	kyv	kfxr	grikp
13	cuuj	cu	qvjuh	jxu	jewq	fghjo
14	btti	bt	putg	iwt	idvp	epgin
15	assh	as	othsf	hvs	hcuo	dofhm
16	zrrg	zr	nsgr	gur	gbtn	cnegl
17	yqqf	yq	mrfqd	ftq	fasm	bmdfk
18	xppe	xp	lqepc	esp	ezrl	alcej
19	wood	wo	kpdob	dro	dyqk	zkbdi
20	vnnc	vn	jocna	cqn	cxpj	yjach
21	ummb	um	inbmz	bpm	bwoi	xizbg
22	tlla	tl	hmaly	aol	avnh	whyaf
23	skkz	sk	glzcx	znk	zumg	vgxze
24	rjyy	rj	fkyjw	ymj	ytlf	ufwyd
25	qiix	qi	ejxiv	xli	xske	tevxc

- 仅知密文攻击
- 攻击成功的特征
  - 加解密算法已知
  - 只需尝试25种密钥
  - 明文已知或易于识别



## 2.1.3 置乱密码

- 与平移字母表不同，随意打乱字母表中字母的顺序，每一个明文字母映射到一个随意的密文字母上
- 明文 & 密文——英文字母表A~Z
- 密钥——明文字母表和密文字母表的映射关系
- 示例

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

Message: DANGERHELPME

Ciphertext: ZBWUOEAOQCQRO

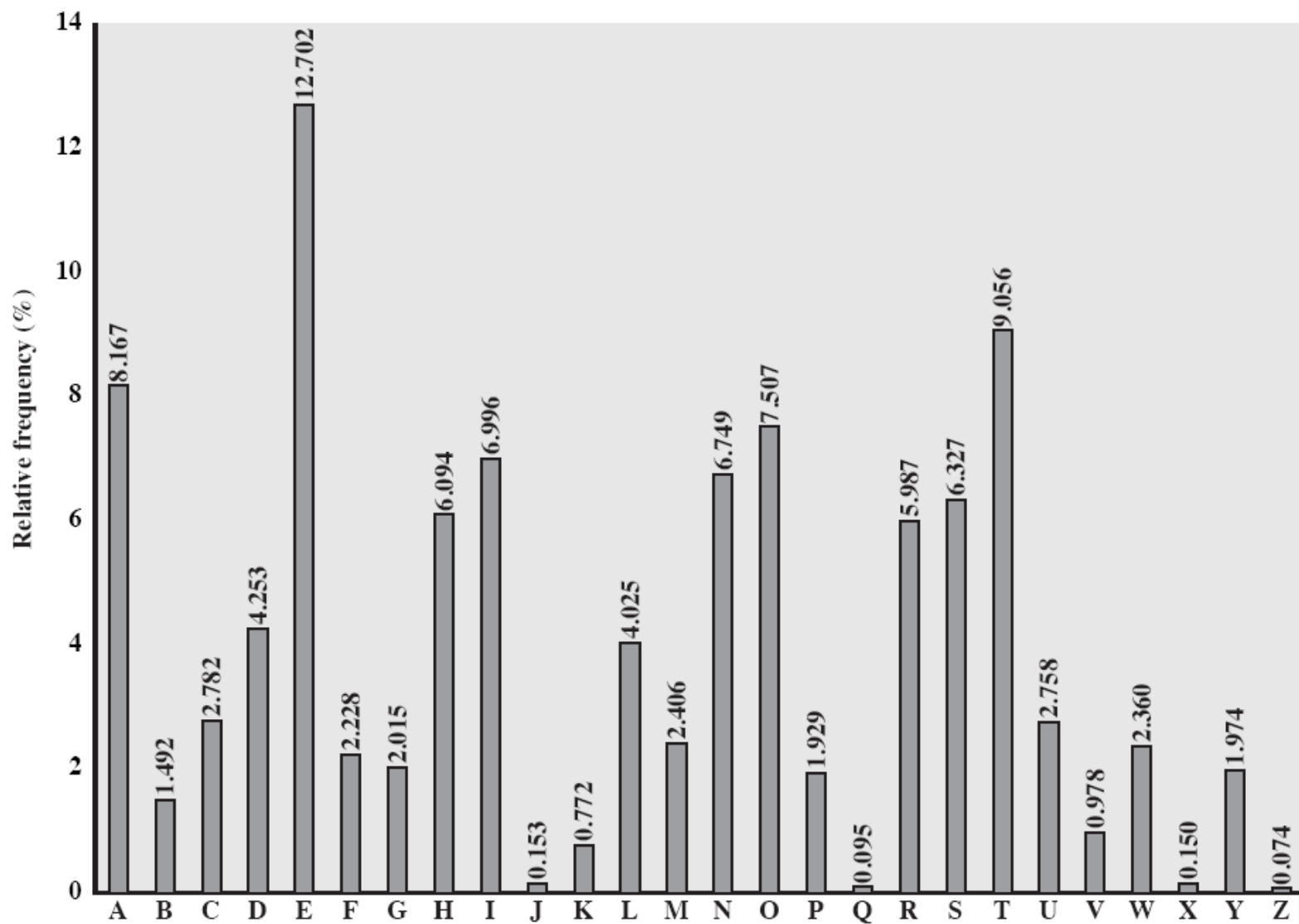


# ● ● ● | 针对置乱密码的统计分析攻击

- 密钥长度= 26
- 密钥空间=  $26! \approx 4 \times 10^{26}$
- 在历史上很长时间内被认为是安全的，但很容易被频率分析攻击破解  
(公元9世纪的阿拉伯数学家)
- Why?  
语言特征
  - 人类语言中各个字符的使用不是均衡的



# 英文文本中各字母的相对使用频率



# ● ● ● | 英文的频率统计特征

- 除了单字母外，双字母(digram)和三字母(trigram) 组合的统计规律也应用于密码分析破解
- 最常见的双字母组合
  - TH, HE, IN, ER, AN, RE, ED, ON, ES, ST, EN, AT, TO, NT, HA, ND, OU, EA, NG, AS, OR, TI, IS, ET, IT, AR, TE, SE, HI, OF
- 最常见的三字母组合
  - THE, ING, AND, HER, ERE, ENT, THA, NTH, WAS, ETH, FOR, DTH





# 密码破解实例

## · 给定密文

UZ QSO VUOHXMOPV GPOZPEVSG ZWSZ OPFPESX UDBMETSX AIZ  
VUEPHZ HMDZSHZO WSFP APPD TSVP QUZW YMXUZUHSX  
EPYEPOPDZSZUFPO MB ZWP FUPZHMDJ UD TMOHMQ

## · 统计字母频率

单字母	P	16	→	E
	Z	14	→	T
	S	10	→	A
	M	9	→	O
双字母	ZW	3	→	TH
	UZ	3	→	IT
三字母	ZWP	1	→	THE



# 密码破解实例

- 经过多次尝试和失败，最终得到:

UZ QSO VUOHXMOPV GPOZPEVSG ZWSZ OPFPESX UDBMETSX **AIZ**  
 VUEPHZ HMDZSHZO **WSFP** **APPD** TSVP QUZW **YMXUZHXSX** **BUT**  
 EPYEPOPDZSZUFPO MB ZWP FUPZHMDJ UD TMOHMQ  
**HAVE BEEN POLITICAL**

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



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

# 密码破解实例

- Ciphertext

UZ QSO VUOHXMOPV GPOZPEVSG ZWSZ OPFPESX UDBMETSX AIZ  
VUEPHZ HMDZSHZO WSFP APPD TSVP QUZW YMXUZUHSX  
EPYEPOPDZSZUFPO MB ZWP FUPZHMDJ UD TMOHMQ

- Message

it was disclosed yesterday that several informal but  
direct contacts have been made with political  
representatives of the vietcong 越共 in moscow

- 情报内容

据悉昨天在莫斯科有一些非正式但是直接与越南共产党的政治  
代表的接触



## 2.1.4 Vigenère密码(1553)

- 最著名的多表代替密码
- 每个密钥字母选取了26个凯撒(移位)密码之一
- 每个密钥用完后再从开头开始重复使用
- 有多个密文表，平滑了频率分布，从而使得密码分析更加困难





		Plaintext																									
		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
Key	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





- 多重(26) 凯撒密码
- 密钥词重复使用使得密钥跟明文一样长
- $C_i = (p_i + k_{i \bmod m}) \bmod 26$   $k = \text{a} \sim \text{z}$
- 示例

Key:	deceptive	deceptive	deceptive
Plaintext:	wearediscovered	saveyourself	
Ciphertext:	ZICVTWQNGRZG	VTWAVZH	CQYGLMGJ

- 密钥长度为m, 密钥空间为 $26^m$



# Kasiski测试 (1863)

- 每个明文字符可能对应26个密文字符，因此统计规律不明显
- 但由于密钥比较短且重复使用，统计规律并未完全消失

Key:	deceptivedeceptivedeceptive
Plaintext:	wearediscoveredsaveyourself
Ciphertext:	ZICVTWQNGRZGVTWAVZHCQYGLMGJ

- 给定足够多的密文，出现重复的密文序列能暴露密钥长度/周期（密码破解的目标之一）
- 如果两个相同明文序列的间距是密钥长度的整数倍，则能得到两个相同的密文序列
- 相同的VTW暗示着密钥长度为3或9





## 2.1.5 Playfair

1854年，由Charles Wheatstone发明，以他朋友Baron Playfair的名字命名

- 最著名的多字母代替密码（一次加密多个字母）
- 双字母代替 (i.e., 通过基于密钥词的5x5转换表  
 $E(p_i p_{i+1}) = c_i c_{i+1}$ )



# Playfair矩阵

- 基于密钥词的**5X5**字母矩阵
- 首先填充密钥词字母
- 接着将剩余字母填充进矩阵
- 举例： 密钥词 **MONARCHY**

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

密钥词 = monarchy

明文:  
密文:

H	S	E	A	A	R	M	U
B	P	I	M	R	M	C	M
	J	M					



# 加密算法

## ○ 一次加密两个字母

1. 如果双字母是重复的，插入一个填充字母，如 'X', 'Z'

例如. "balloon" 被填充为 "ba lx lo on"

2. 如果双字母位于同一行，用每个字母右边的字母去代替原字母（当行结束后回滚到行首）

例如. "ar" 被加密为 "RM"

3. 如果双字母位于同一列，用每个字母下面的字母去代替原字母（当列结束后回滚到列顶部）

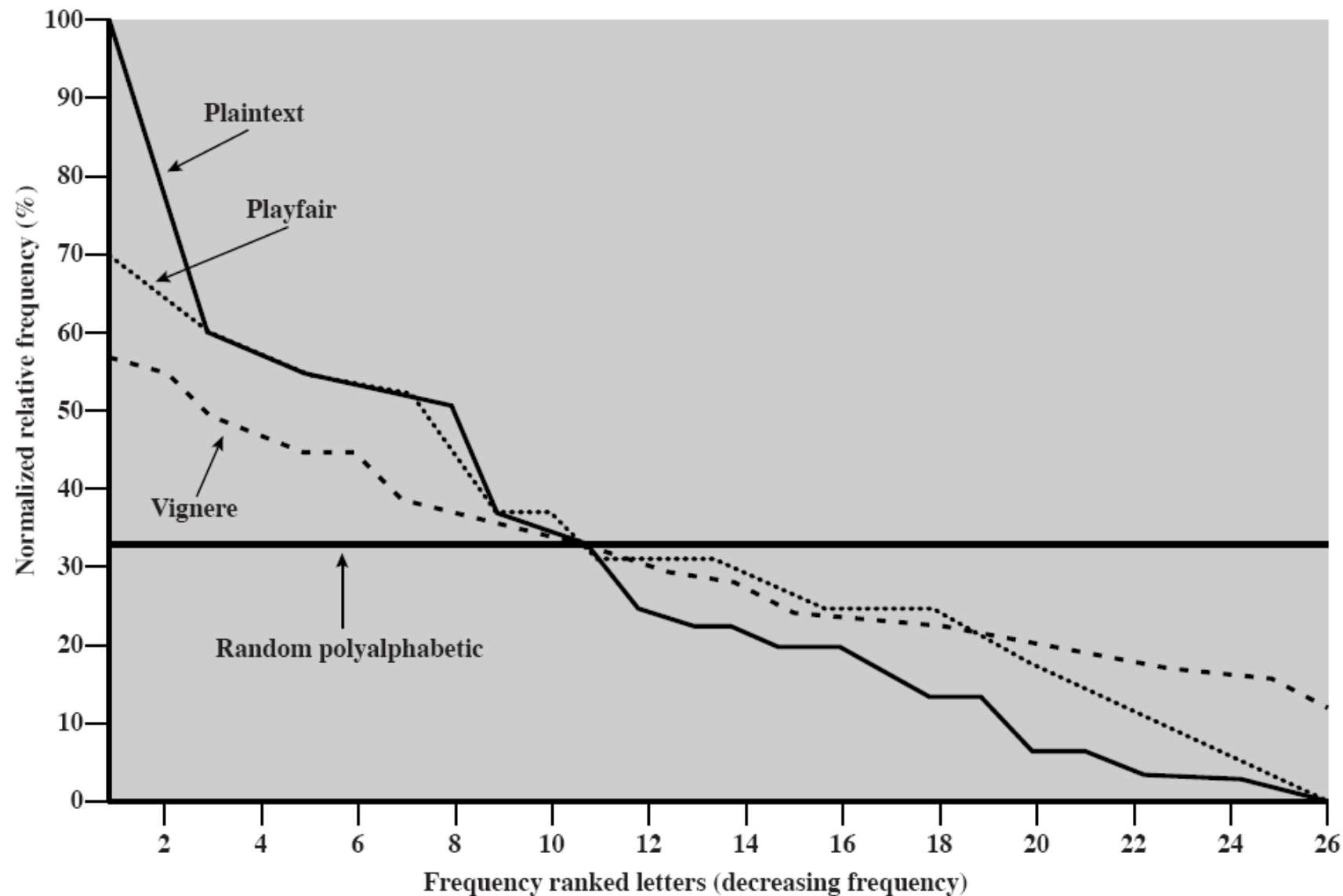
例如. "mu" 被加密为 "CM"

4. 否则，每个字母用与它同一行，与另一字母同一列交叉点的字母代替

例如. "hs" 被加密为 "BP", "ea" 被加密为 "IM" 或 "JM"



# 字母的相对出现频率



## 2.2 置换密码

### ○ 栅栏密码

- 将明文字母以对角线的方式写入多行
- 逐行读出密文字母
- 例如. 将明文写成如下方式

m e m a t r h t g p r y  
e t e f e t e o a a t

- 得到密文

**MEMATRHTGPRYETEFETEOAAT**



# 列置换密码

- 明文以矩形方式一行行写入，但一列列的读出
- 密钥是矩形的长度和列读出顺序

密钥:           4 3 1 2 5 6 7

明文:           a t t a c k p

                o s t p o n e

                d u n t i l t

                w o a m x y z

密文: TTNAAPTMTSUOAODWCOIXKNLYPETZ



# 多步置换

## 多步置换难以反向重构

密钥: 4 3 1 2 5 6 7

输入: t t n a a p t

m t s u o a o

d w c o i x k

n l y p e t z

输出: NSCYAUOPTTWLTMDNAOIEPAXTTOKZ

字母的原始顺序:

01 02 03 04 05 06 07 08 09 10 11 12 13 14

15 16 17 18 19 20 21 22 23 24 25 26 27 28

第一次置换后:

03 10 17 24 04 11 18 25 02 09 16 23 01 08

15 22 05 12 19 26 06 13 20 27 07 14 21 28

第二次置换后:

17 09 05 27 24 16 12 07 10 02 22 20 03 25

15 13 04 23 19 14 11 01 26 21 18 08 06 28



# ● ● ● | 乘积密码

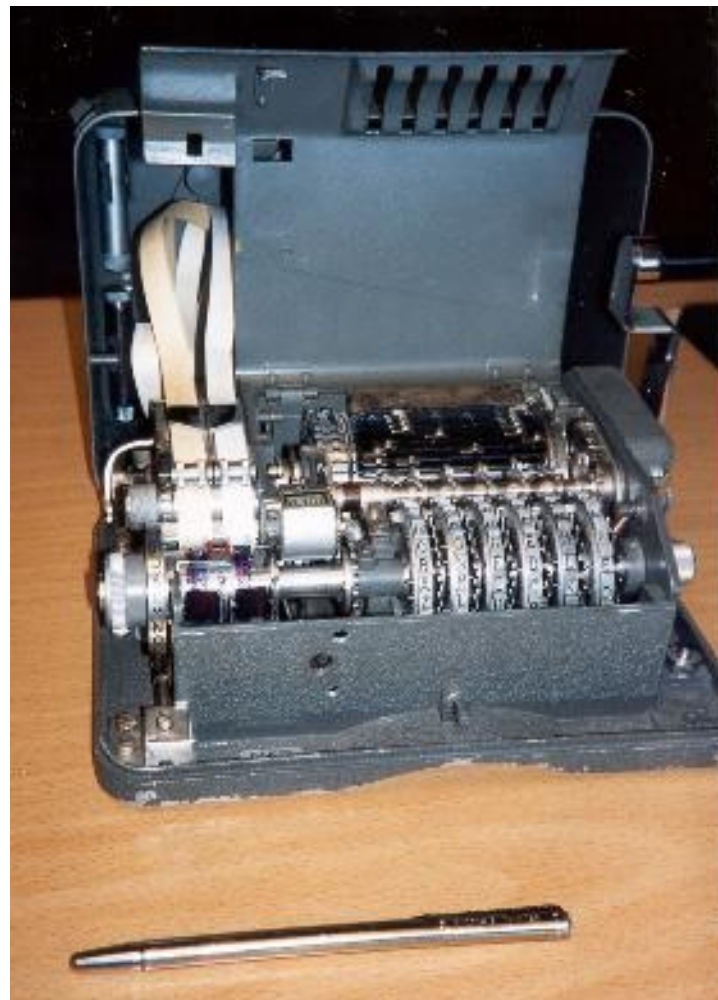
- 由于语言的统计特征，单纯使用代替或置换是不够安全的
- 连续多次使用密码使得算法更安全
  - 两次代替得到更为复杂的代替密码
  - 两次置换得到更为复杂的置换密码
  - 对比前两种，一次代替加一次置换得到安全性高得多的密码
- 从古典密码过渡到现代密码的桥梁





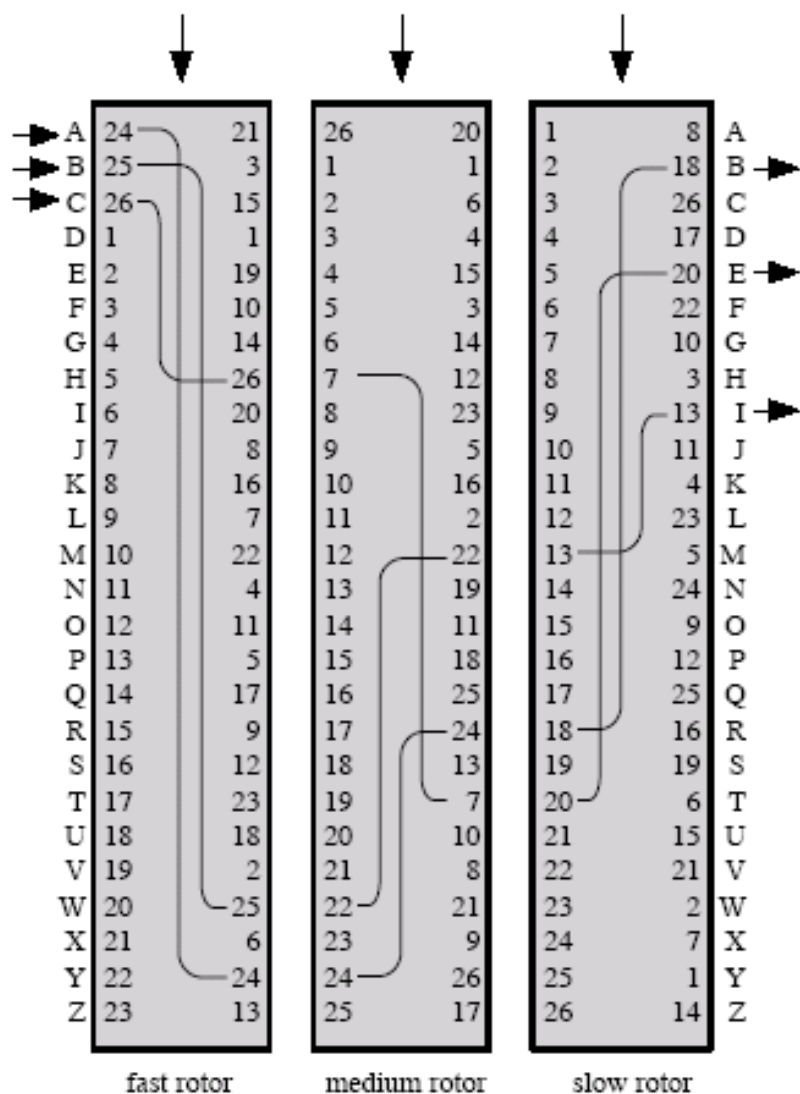
## 2.1.6 转轮机

- 在现代密码之前，转轮机是最经典的多重代替的乘积密码
- 机械密码装置，广泛的应用于一战二战：德国 (Enigma), 日本 (Purple), 瑞典 (Hagelin)



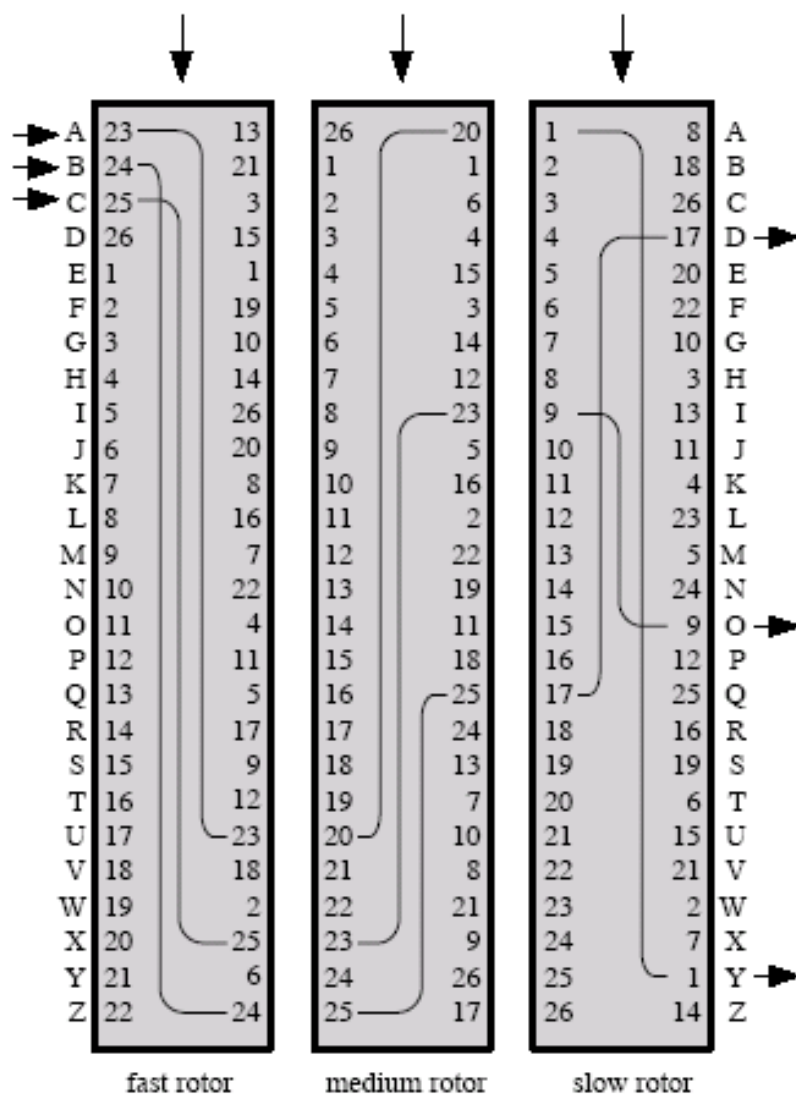
# 3-转子

direction of motion




(a) Initial setting

direction of motion



(b) Setting after one keystroke



- 
- 每个转子对应一个代替密码
  - 每个转子产生一个对应着26个密文字母表的多表代替密码
  - 每次按键后，转子旋转一个刻度
  - 每个转子的输出是下个转子的输入
  - 在每个转子循环完一个完整周期后，相邻的下个转子旋转一个刻度（类似里程表）
    - 一个3-转子转轮机产生 $26^3=17576$ 个密文表



# 3 公钥密码

## 3.1 公钥密码原理



## 3.2 RSA算法



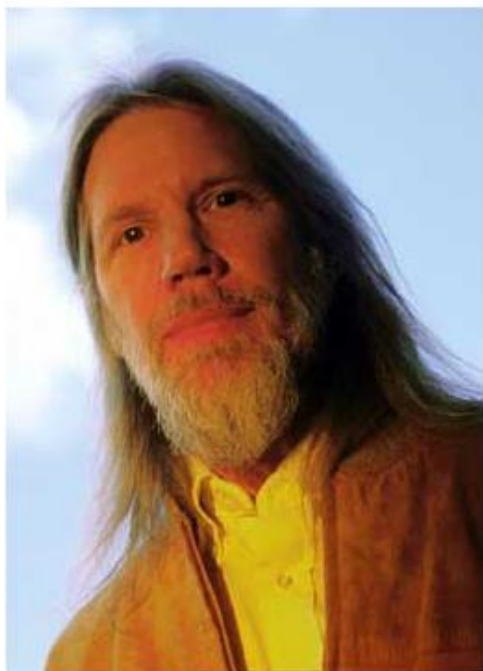
# 误解

- 公钥密码比对称密码更安全
- 公钥密码的出现使得对称密码过气
- 使用公钥密码进行密钥分配很容易



## 3.1 公钥密码原理

- 1976年，Diffie和Hellman首次公开提出公钥密码的概念
- 他们因此赢得了2016年的图灵奖



Whitfield Diffie



Martin Hellman



- 公钥算法依赖于一个密钥加密，另一个不同但是有关联的密钥解密 ( $PR \rightarrow PU$ )

- 要求

- ✓ 加解密可以还原(基本条件)

$$D(E(M, PU), PR) = M \text{ or } D(E(M, PR), PU) = M$$

- ✓ 给定加密密钥，想得到解密密钥从计算上不可行  
(安全条件)

$$PU \not\rightarrow PR$$

- ✓ 当相关的加解密密钥已知，做相应加解密操作是快速高效的

(工程使用条件)

- 可选条件

- ✓ 加解密的顺序可以互换

$$D(E(M, PU), PR) = D(E(M, PR), PU) = M$$



# 对称密码和公钥密码

## 对称密码

- 加解密使用相同的密钥和算法
- 收发双方共享密钥
- 密钥必须保密
- 如没有其他信息，解密消息是不可行的
- 知道算法和若干密文不足以确定密钥

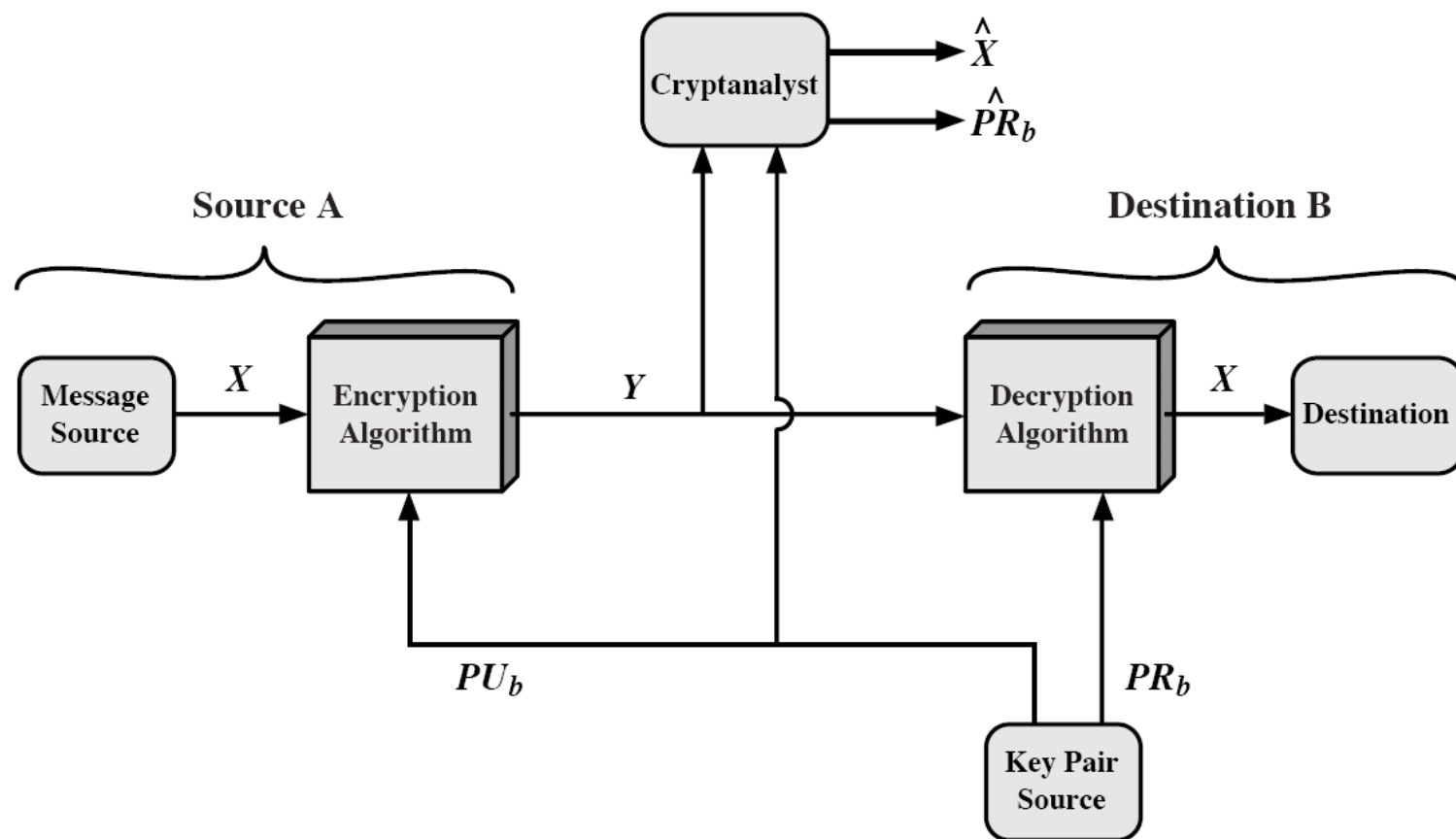
## 公钥密码

- 同一算法用于加解密，但加解密使用不同密钥
- 发送方拥有PU或PR，而接收方拥有另一个
- PR必须保密
- 如没有其他信息，解密消息是不可行的
- 知道算法、公钥PU和若干密文不足以确定私钥PR





# 公钥密码体制：保密性



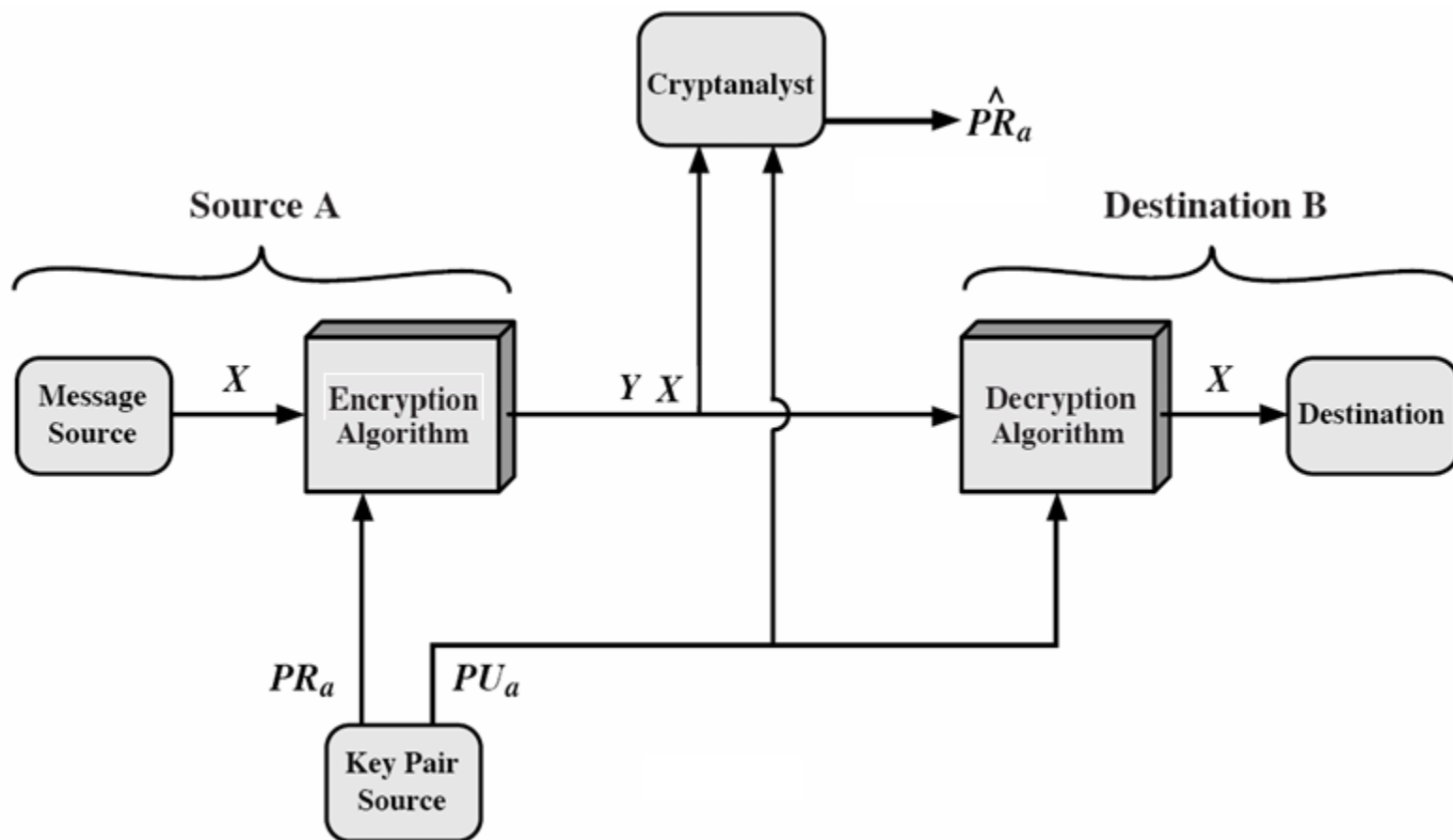
$$Y = E_{PU_b}(X)$$
$$X = D_{PR_b}(Y)$$

不能保证认证和真实性

$PU_b$ : B的公钥  
 $PR_b$ : B的私钥



# 公钥密码体制：认证（数字签名）

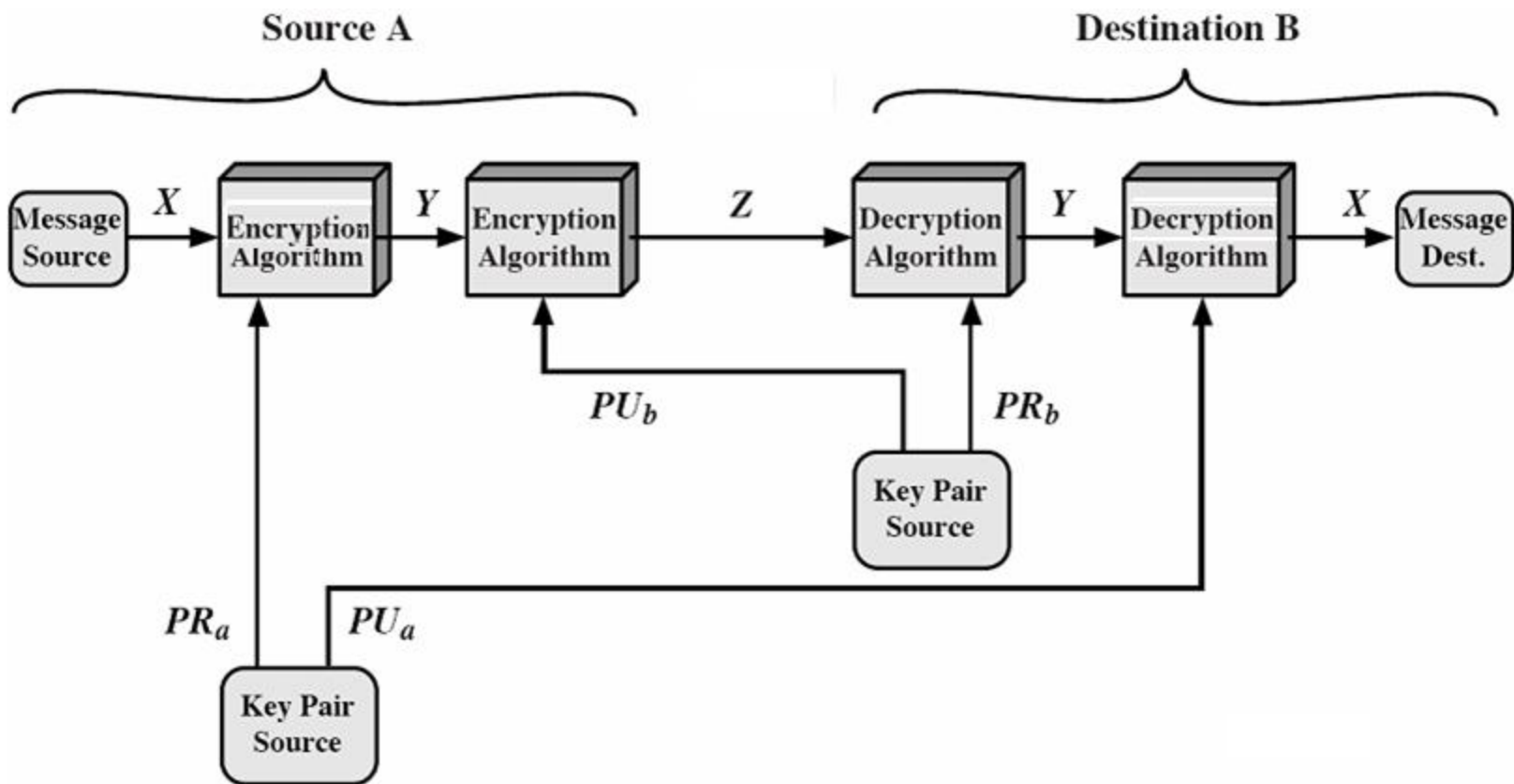


$$Y = E_{PR_a}(X)$$
$$X = D_{PU_a}(Y)$$

保证认证，不可否认性和完整性  
不能保证保密性



# 公钥密码体制：保密性和认证



保证保密性，认证，不可否认性和完整性

$$Z = E_{PU_b}[E_{PR_a}(X)]$$
$$X = D_{PU_a}[D_{PR_b}(Z)]$$



# 公钥密码应用

- 加解密（提供保密性）
  - 数字签名（提供认证、完整性和不可否认性）
  - 密钥交换（分配会话密钥）
- 有些算法适合三种应用，有些只能提供一种



# ● ● ● | 单向函数&单向陷门函数

## ○ 单向函数

$Y = f(X)$  容易 (多项式时间)

$X = f^{-1}(Y)$  计算上不可行 (非多项式时间)

## ○ 单向陷门函数

$Y = f(X)$  已知 $X$ , 容易

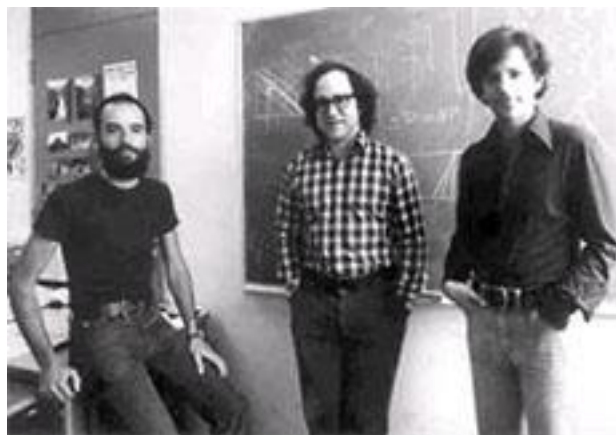
$X = f_{PR}^{-1}(Y)$  已知 $PR$ 和 $Y$ , 容易

$X = f^{-1}(Y)$  已知 $Y$ 但不知道 $PR$ , 计算上不可行



## 3.2 RSA算法

- 1977年，由MIT的Ron Rivest, Adi Shamir和Len Adleman三人设计
- 既支持公钥加密，又支持数字签名
- 明密文长度 $\leq \lg(n)$  bit





# 理论基础

计算两个大素数的乘积容易，而将一个大的乘积因子分解为两个素数困难



# 模算术

- RSA基于有限域的模运算
- 模加
- 模乘
- 模幂





# 模加

- $x+y \bmod (n=10)$

	0	1	2	3	4	5	6	7	8	9
0	0	1	2	3	4	5	6	7	8	9
1	1	2	3	4	5	6	7	8	9	0
2	2	3	4	5	6	7	8	9	0	1
3	3	4	5	6	7	8	9	0	1	2

- 加法逆元: 和模 $n$ 为0



# 模乘

- $x * y \bmod (n=10)$

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9
2	0	2	4	6	8	0	2	4	6	8
3	0	3	6	9	2	5	8	1	4	7

- 乘法逆元: 乘积模 $n$ 为1
- 不是所有数都有乘法逆元



- 只有与n互素的数有模n的乘法逆元
- x, n互素: 除了1以外, 没有其他公因子
- 例.  $n=10$ , 1、3、7、9与10互质

$$1*1=1 \bmod 10$$

$$3*7=1 \bmod 10$$

$$7*3=1 \bmod 10$$

$$9*9=1 \bmod 10$$



# 模幂

$X^y \bmod n$ ( $n=10$ )	0	1	2	3	4	5	6	7	8	9
0		0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1
2	1	2	4	8	6	2	4	8	6	2
3	1	3	9	7	1	3	9	7	1	3
4	1	4	6	4	6	4	6	4	6	4
5	1	5	5	5	5	5	5	5	5	5
6	1	6	6	6	6	6	6	6	6	6
7	1	7	9	3	1	7	9	3	1	7
8	1	8	4	2	6	8	4	2	6	8
9	1	9	1	9	1	9	1	9	1	9



# RSA算法

## 密钥生成

选择 $p, q$	$p$ & $q$ 均为大素数且 $p \neq q$	(保密)
计算 $n = p * q$		(公开)
计算 $\Phi(n) = (p-1)(q-1)$		(保密)
选择 $e$	$\gcd(\Phi(n), e) = 1$ 且 $1 < e < \Phi(n)$	(公开)
选择 $d$	$d \equiv e^{-1} \pmod{\Phi(n)}$	(保密)
公钥	$PU = \{e, n\}$	
私钥	$PR = \{d, p, q, \Phi(n)\}$	

## 加密 & 验证签名

明文:	$M < n$
密文:	$C = M^e \pmod n$

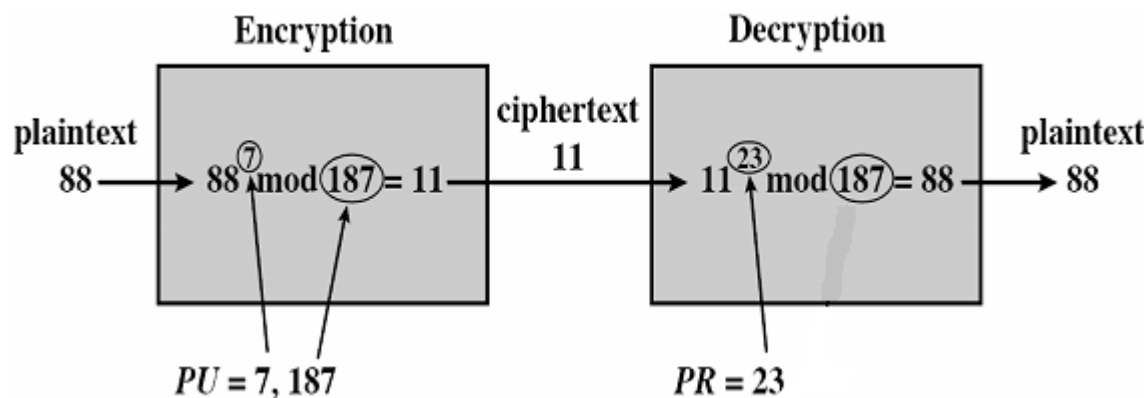
## 解密 & 签名

密文:	$C$
明文:	$M = C^d \pmod n$



# RSA例子

1. 选择两个素数,  $p = 17$   $q = 11$
2. 计算  $n = pq = 17 \times 11 = 187$
3. 计算欧拉函数  $\phi(n) = (p-1)(q-1) = 16 \times 10 = 160$
4. 选择  $e$ ,  $e$  小于  $\phi(n)$  且与  $\phi(n)$  互素;  $e = 7$
5. 确定  $d$ ,  $de \equiv 1 \pmod{160}$  且  $d < \phi(n)$ .  $d = 23$  ( $7 \times 23 = 161 = 160 + 1$ )
6.  $PU = \{7, 187\}$ ,  $PR = \{23, 17, 11, 160\}$



# ● ● ● | 安全性分析

- $PU = \{e, n\}, PR = \{d, p, q, \phi(n)\}$
- $d \leftarrow \phi(n) \leftarrow p, q \leftarrow$  对 $n$ 进行因子分解



# 可交换性

- $D(E(M)) \equiv (M^e)^d \equiv M^{ed} \equiv (M^d)^e \equiv E(D(M)) \pmod n$
- 加解密的顺序可以交换，因此可以保证保密性和认证，即加密和数字签名





# 可逆性证明

- $e^*d \equiv 1 \pmod{\phi(n)}$ , 即  $e^*d = 1 + k^*\phi(n)$
  - $C^d = (M^e)^d = M^{1+k^*\phi(n)}$
  - 要证明加解密还原, 即  $M^{1+k^*\phi(n)} \equiv M \pmod{n}$ 
    - ① 如果  $(M, n) = 1$ ,  $M^{\phi(n)} \equiv 1 \pmod{n}$  (欧拉定理)
- 左 =  $M^*(M^{\phi(n)})^k \equiv M^*(1)^k \pmod{n} \equiv M \pmod{n} =$   
 $M =$  右

② 如果  $(M, n) \neq 1$ , 假设  $(M, n) = p$  则  $M = ap$

$\because M < n$  且  $q$  是素数  $\therefore (M, q) = 1$

$M^{\Phi(q)} \equiv 1 \pmod q$  (欧拉定理)

$\therefore M^{k(p-1)\Phi(q)} \equiv 1 \pmod q$

$\because \Phi(q) = q - 1$

$\therefore M^{k\Phi(n)} = 1 \pmod q \quad M^{k\Phi(n)} = bq + 1$

左右同时乘以  $M$ :

$M^{k\Phi(n)+1} = bqM + M = abpq + M = abn + M$

$\pmod n: M^{k\Phi(n)+1} \equiv M \pmod n$  故得证



# 计算方面

## ○ 加解密

- 使用反复平方乘算法，做模幂运算
- 使用中国剩余定理CRT，做计算加速cf.

9.2.2 P204

## ○ 密钥产生

- 确定两个大素数 $p$ 和 $q$  (Miller-Rabin测试) cf. 网上
- 选择 $e$ ，计算逆元 $d$ (扩展Euclid算法)



# 模幂

- $(a*b) \bmod n = [(a \bmod n)*(b \bmod n)] \bmod n$
- $a^b \bmod n = (a \bmod n)^b \bmod n$
- 使用反复平方乘算法，快速有效的计算模幂
- 参照幂指数的二进制表示
- 对于幂指数**b**，模幂的计算复杂性为 $O(\log_2 b)$



模幂  $(a, b, n)$  之反复平方乘算法

1.  $c \leftarrow 0$  (初始幂指数)  $f \leftarrow 1$  (初始模幂)

2. 将  $b$  表示为二进制  $b_k b_{k-1} \dots b_0$

3. for  $i \leftarrow k$  downto 0 do

$c \leftarrow 2 \times c$  (幂指数的中间计算结果)

$f \leftarrow (f \times f) \bmod n$  (模幂的中间计算结果)

    if  $b_i = 1$  then  $c \leftarrow c + 1$

$f \leftarrow (f \times a) \bmod n$

4. 返回  $f$



# 示例

- $a^b \bmod n$ ,  $a=7$ ,  $b=560=1000110000$ ,  $n=561$

i	10	9	8	7	6	5	4	3	2	1
bi	1	0	0	0	1	1	0	0	0	0
c	1	2	4	8	17	35	70	140	280	560
f	7	49	157	526	160	241	298	166	67	1



# 求逆

- 用扩展欧几里得算法（辗转相除法）求 $b(e)$ 关于 $a(\Phi(n))$ 的乘法逆元

扩展欧几里得定理：

对于正整数 $a, b$ ，必然存在整数对  $X, Y$ ，

使得  $X*a + Y*b = \gcd(a, b)$

如果 $a, b$ 互素，则 $X*a + Y*b = 1$

两边同时 $\text{mod } a$ ，则 $Y*b \equiv 1 \pmod a$ ， $Y$ 就是 $b$ 关于 $a$ 的乘法逆元

Extended Euclid ( $a, b$ )算法：

```
int Gcd(int a, int b)
{
    if(b == 0)
        return a;
    return Gcd(b, a % b);
}
```



# 示例

- $a=35, b=13$ , 求 $b$ 关于 $a$ 的乘法逆元

$a \bmod b$	$a$	$b$	
$35=2*13+ 9$	35	13	
$13=1*9+ 4$	13	9	
$9=2*4+ 1$	9	4	$1=9-2*4$
$4=4*1+ 0$	4	1	
	1	0	

$1=9-2*4=9-2*(13-1*9)=3*9-2*13=3*(35-2*13)-2*13=3*35-8*13$   
 $1= 3*35-8*13$   
两边同时mod35:  
 $1 \equiv (-8)*13 \bmod 35 \equiv (35-8)*13 \bmod 35 = 27*13 \bmod 35$   
得到, 27是13关于35的乘法逆元

$$(27*13) \bmod 35 = 351 \bmod 35 = 1$$





# 求逆算法

- 用扩展欧几里得算法求**b**关于**a**的乘法逆元

Extended Euclid (a, b)

```
1.(A1,A2,A3)←(1,0,a); (B1,B2,B3)←(0,1,b);  
2.If B3=0 return 'No Inverse';  
3.If B3=1 return B2;  
4.Q←[A3/B3]  
5.(Temp1,Temp2,Temp3)←(A1-QB1,A2-QB2,A3-QB3)  
6.(A1,A2,A3)←(B1,B2,B3)  
7.(B1,B2,B3)←(Temp1,Temp2,Temp3)  
8.goto 2
```



# 示例

- $a=35, b=13$ , 求 $b$ 关于 $a$ 的乘法逆元

轮	A1	A2	A3	B1	B2	B3	Q	Temp1	Temp2	Temp3
1	1	0	35	0	1	13	2	1	-2	9
2	0	1	13	1	-2	9	1	-1	3	4
3	1	-2	9	-1	3	4	2	3	-8	1
4	-1	3	4	3	-8	1				

$$-8 \equiv (35-8) = 27 \pmod{35}$$

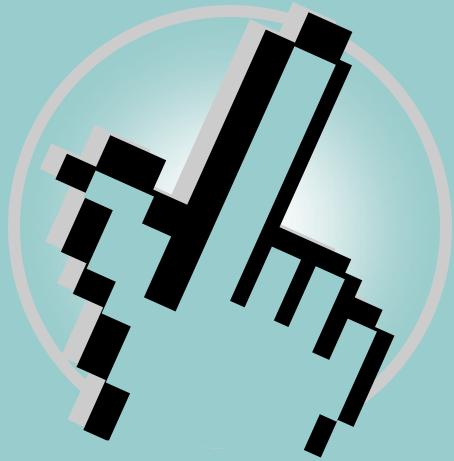
$$(27 \cdot 13) \pmod{35} = 351 \pmod{35} = 1$$



# 公钥密码的密码分析

- 针对PR的穷举攻击
- 数学分析攻击:  $PU \rightarrow PR$
- 穷举消息攻击





| 2021 |

Thank you...