

# 第5章 公钥密码算法及应用

东南大学 网络空间安全学科  
胡爱群 教授

# 本章内容:

- 什么是公钥密码?
- 为什么需要公钥密码?
- 公钥密码体制的基本原理
- 单向函数
- 公钥算法
- 消息摘要
- 数字签名
- 公钥密码PKI中的应用

# 什么是公钥密码体制？

- 也称为非对称密码体制，即加密密钥和解密密钥不同，但又是唯一对应的一对；
- 从计算难度上看，不可能轻易从其中一个密钥推导出另一个密钥；
- 既然是非对称的，就可以把其中一个保密起来（私钥），把另一个公开（公钥）；
- 用私钥加密消息，用公钥解密消息，可实现数字签名；
- 用公钥加密消息，用私钥解密消息，可实现消息的保密传输。

# 为什么需要公钥密码算法？

- 传统密码体制只使用一个密钥；
  - 收发双方共享这个单一的密钥；
  - 密钥是对称的，双方是对等的；
  - 因此，不能确保接收方伪造信息，并声称是该信息是发送方发送的。
- 
- 密钥交换需要机密性通道；
  - 大规模群体中，密钥管理规模庞大复杂（1000个用户，任意两者之间需要一个密钥，整个群体约需保存50万个密钥）；
  - 未知实体间通信困难；
  - 难以实现非否认服务。

# 没有秘密的加密方式—密码学新方向

- **Diffie(1944-)**
  - 1965年，获得麻省理工学院数学学士学位
  - 1976年，和Hellman联合发表《密码学新方向》
  - 1991年，任职于Sun公司
  - 1992年，瑞士联邦理工学院授予博士头衔。



# Diffie考虑的问题

## 1. 加密过程中密钥分发问题

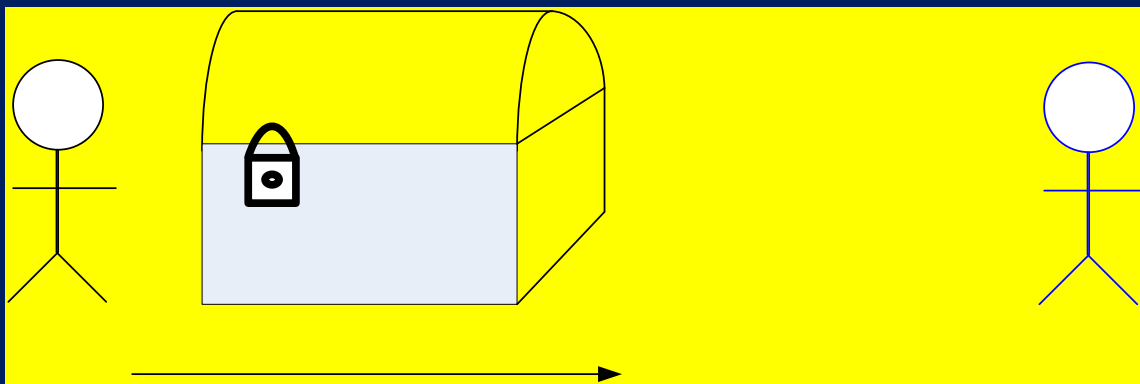
- 手工分发不符合实际情况，而KDC的存在意味着通信双方的隐私会被第三方监视。
- 用可以公开的信息作为密钥，就可以简化密钥分发问题。

## 2. “数字签名”问题：

- 寻找一个正确判断消息来源的方法，即象手写签名一样，以保证消息确是出自特定的人。

# 思考一下：双重加密方案

- 双重加密方案
  1. Alice把消息放到箱子里，用自己的锁锁上，发送给Bob;
  2. Bob收到箱子后加上自己的锁，把箱子返回给Alice;
  3. Alice除掉自己的锁，再把箱子寄给Bob;
  4. Bob打开自己的锁，读取消息。



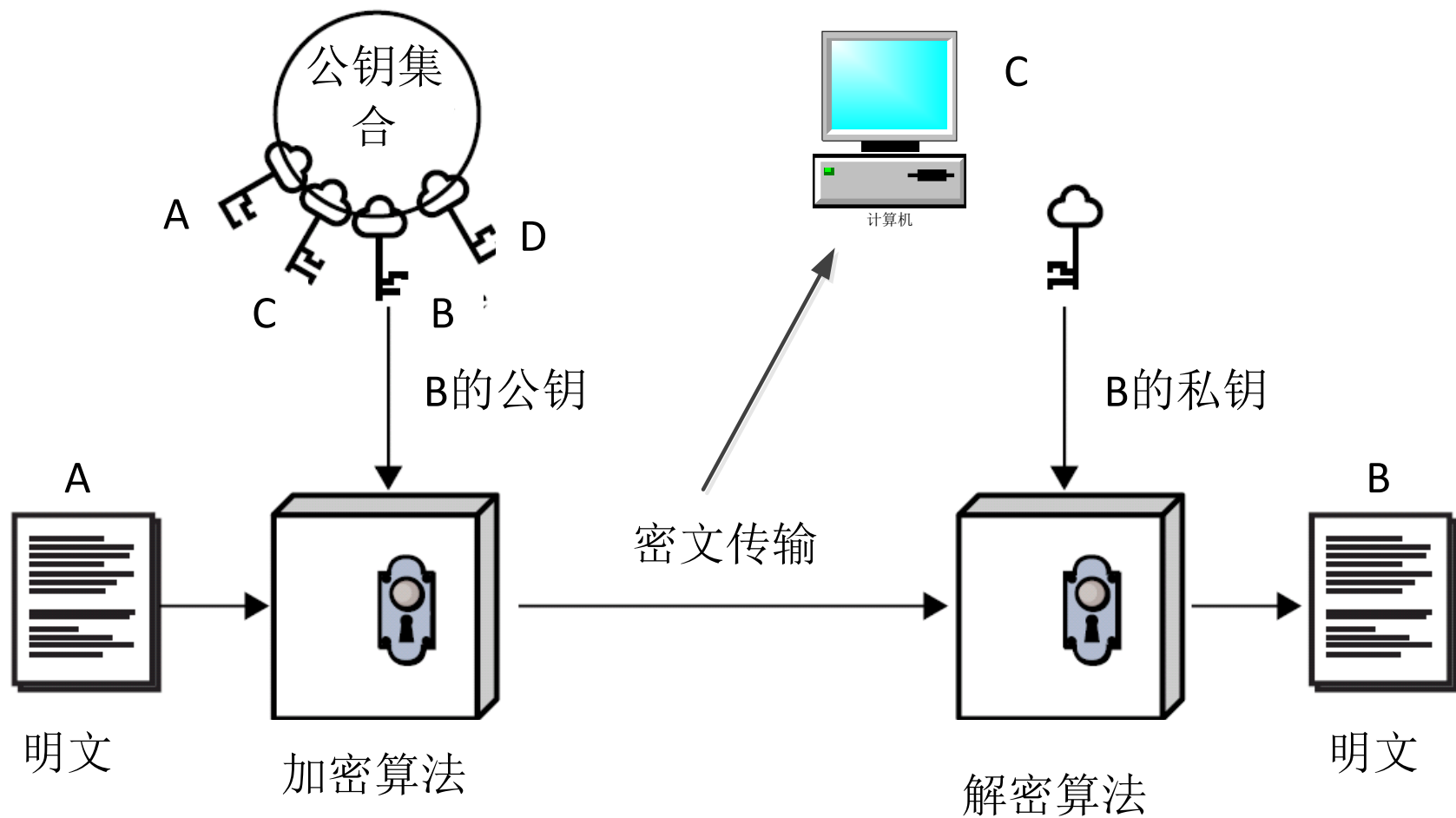
保证中间人不能看到消息，但A需要证明B的身份。  
与公钥加密思想一致。

# 理解公钥密码体制

- 公钥密码都是指使用两个密钥（一对）：
  - 公钥：可以对任何人公开的密钥，用于加密消息或验证签名。
  - 私钥：只能由接收者私存，用于解密消息或签名。
- 非对称
  - 密钥的非对称；
  - 用于加密消息或验证签名的人不能对该消息解密或对进行同样的签名（唯一性）。



# 公钥密码的基本模型



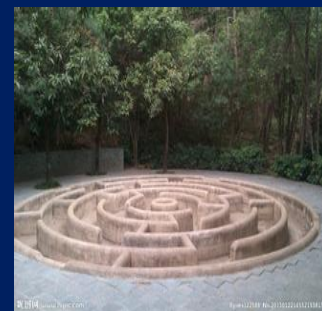
# 公钥密码体制特点:

公钥密码算法依赖于:

- 仅仅知道算法和加密密钥，推导解密密钥计算上是不可行的；
- 已知加解密密钥时，进行加解密运算计算上是容易的；
- 两个密钥中的任何一个都可以用来加密，而另一个用来解密。关键是保管好对应的密钥。

# 生活中的单向陷门函数：

- 道路的单行线：按照指示线路走，很容易从起点到终点；但如果没有指示牌，从终点很难找到起点。
- 合成饮料：很容易用原料合成饮料；但如果不掌握配方，想要分析出原料就很困难。



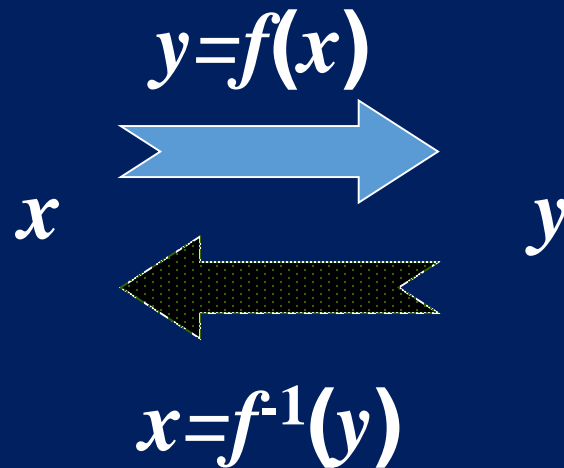
- 单向函数：
  - 函数值计算很容易
  - 逆计算是不可行的。
- 单向陷门函数：
  - 函数值计算很容易
  - 若知道某种附加的信息，则逆计算是可行的，否则不可行。

公钥密码基于  
单向陷门函数

# 限门单向函数

## Trapdoor One-way Function

- 一个函数正向容易计算，但求逆很困难。



- 在拥有特定的限门知识后容易求逆。

# 单向函数单向陷门函数

单向函数:

- $Y = f(X)$  容易;
- $X = f^{-1}(Y)$  不可行。

• 单向陷门函数

- $Y = f_k(X)$  给定 $k$ 和 $X$ , 容易;
- $X = f_k^{-1}(Y)$  给定 $k$ 和 $Y$ , 容易;
- $X = f_k^{-1}(Y)$  给定 $Y$ 但 $k$ 未知, 不可行。

# 常见的单向函数：

- 大整数分解的困难性

已知大整数 $n$ , 且 $n=p \cdot q$ , 试图求出 $p$ 、 $q$ 是困难的。  
一般推荐 $p$ 、 $q$ 为512比特的素数。

- 求解离散对数的困难性

就是给定正整数 $x$ 、 $y$ 和 $n$ , 求出正整数 $k$ （如果存在的话），使 $y \equiv x^k \pmod{n}$ 。就目前而言，人们还没有找到计算离散对数的快速算法。 $n$ 足够大。

# 常见单向函数(2)

- 多项式求根的困难性

$$y = x^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0 \bmod p$$

已知 $a_0, a_1, \dots, a_{n-1}$ 和 $p$ 、 $x$ 易于求 $y$ ，但若已知 $y$ 和 $a_0, a_1, \dots, a_{n-1}$ 及 $p$ 求 $x$ 是困难的。

- 背包问题

已知 $A = (a_1, a_2, \dots, a_N)$ ,  $a_i$ 为正整数

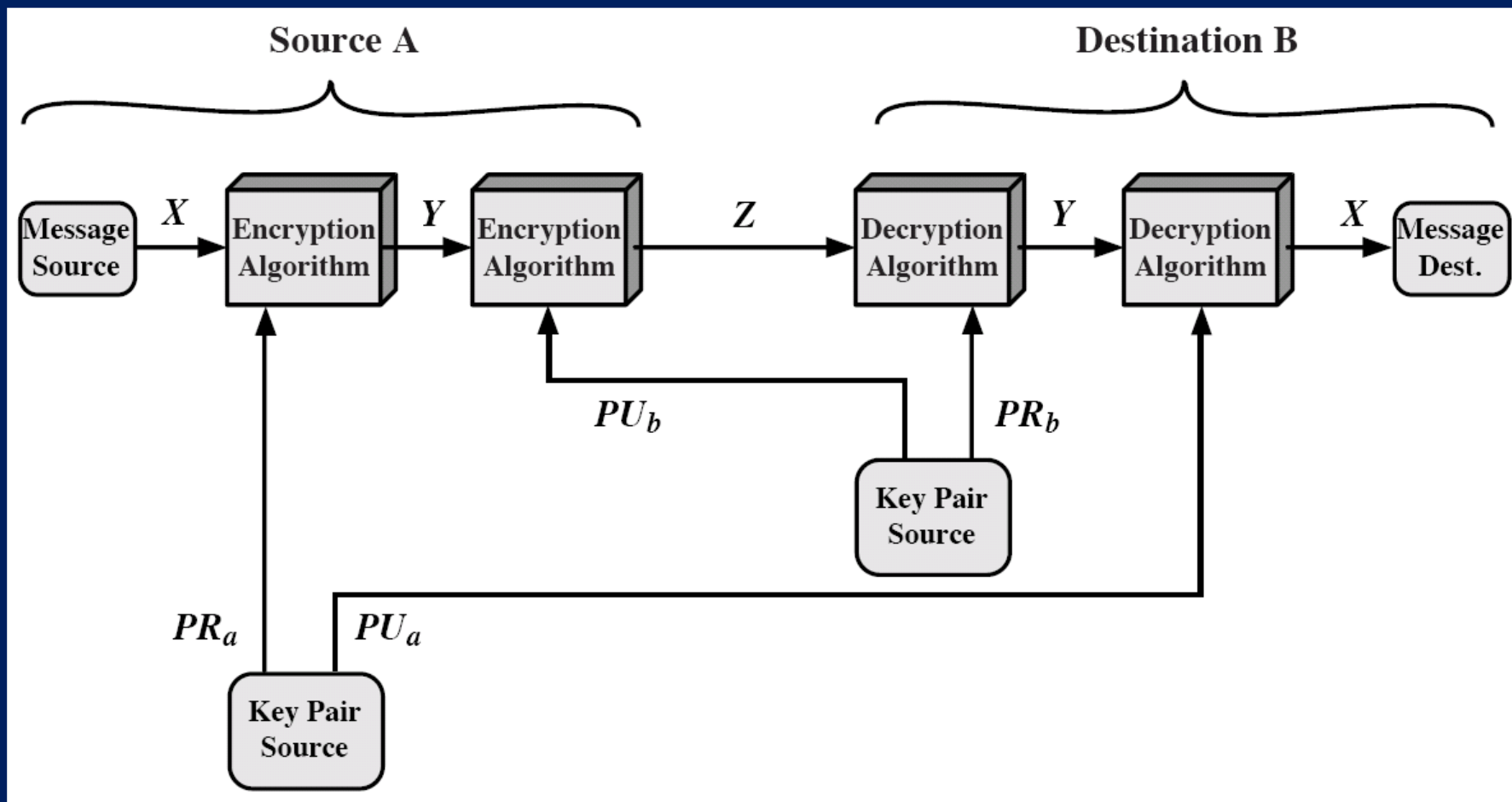
$$X = (x_1, x_2, \dots, x_N), x_i \in \{0, 1\}$$

求

$$S = f(x) = \sum_{i=1}^N x_i a_i$$

容易，但已知 $A$ 、 $S$ ，要求 $X$ 很困难，当 $N$ 很大时。矩阵求逆。

# 公钥密码体制：保密和认证

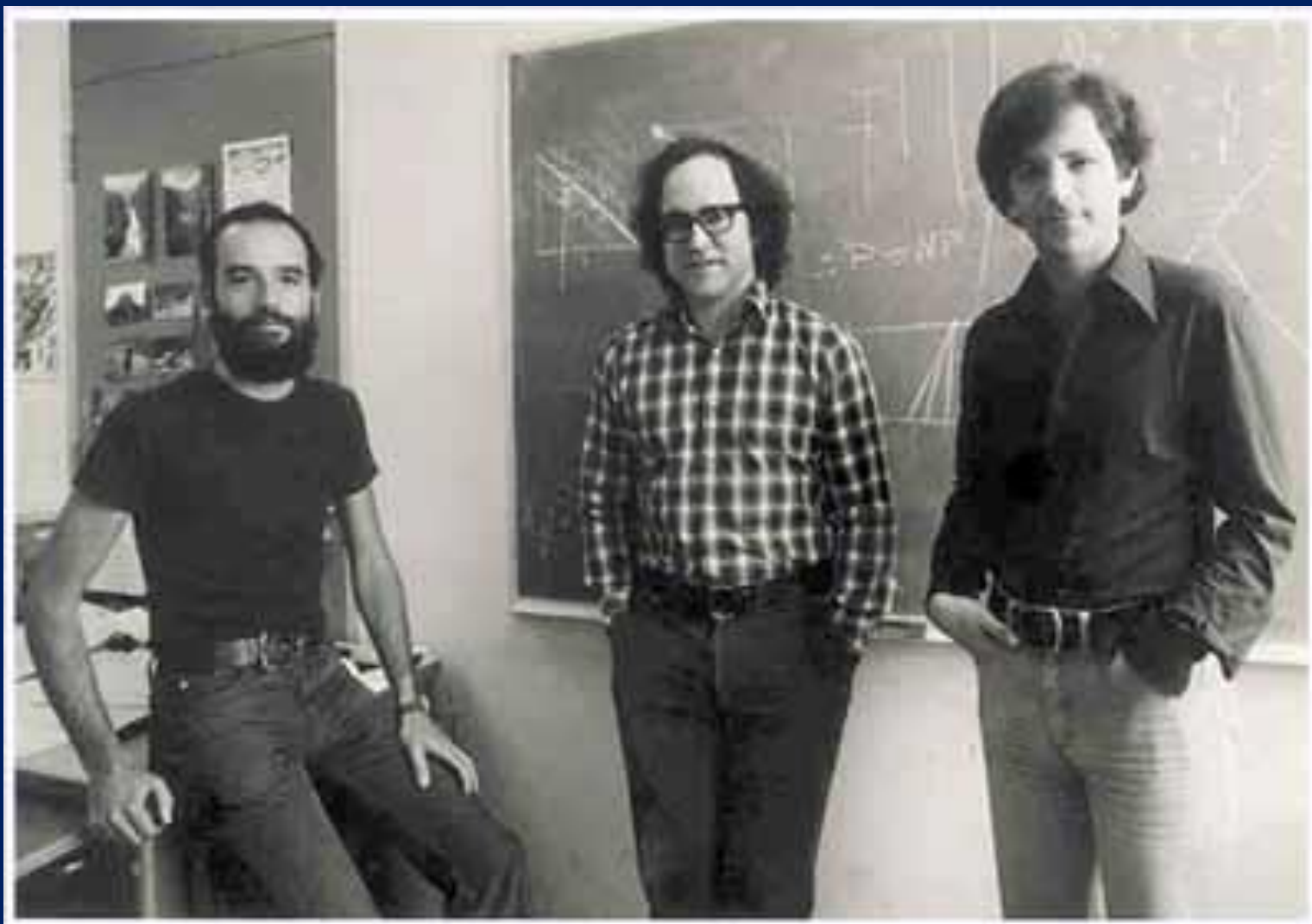




# 公钥密码体制的应用

- 分为三类:
  - 加密/解密 (提供保密性)
  - 数字签名 (提供认证)
  - 密钥交换 (会话密钥)
- 一些算法可用于上述三类, 而有些只适用于其中一类或两类。

# RSA算法介绍



# RSA算法

- 1977由MIT的Rivest, Shamir 和 Adleman 发明;
- 已知的且被广泛使用的公钥密码方案;
- 安全性基于大数的素因子分解的困难性。

# RSA算法描述

- 找素数：
  - 选取两个随机素数 $p$ 、 $q$ ;
- 计算模 $n$ 和Euler函数 $\varphi(n)$ :
  - $n=p \cdot q$
  - $\varphi(n)=(p-1) \cdot (q-1)$
- 随机选择加密密钥 $e$ , 使 $e$ 和 $(p-1)(q-1)$ 互素, 利用欧几里德扩展算法找 $e$ 的逆元 $d$ :
  - $ed \equiv 1 \pmod{\varphi(n)}$
  - $d = e^{-1} \pmod{\varphi(n)}$
- 发布
  - 发布 $(e, n)$ , 这是公钥 $k_e$
  - $d$ 保密,  $(d, n)$ 是私钥  $k_d$

# RSA加解密算法

- 加密
  - 明文分组 $m$ 做为整数，须小于 $n$

$$c = m^e \bmod n$$

- 解密

$$m = c^d \bmod n$$

$e$ 、 $n$ 为公钥， $d$ 私钥。在不知道 $p$ 、 $q$ 的情况下，不可能由 $e$ 推到 $d$ 。因为由 $n$ 分解得到 $p$ 、 $q$ 是困难的。

# 关于大素数的选取

- 为了避免攻击者用穷举法求出 $p$ 和 $q$ ，应该从足够大的集合中选取 $p$ 和 $q$ 。建议选择 $p$ 和 $q$ 大约是100位的十进制素数。
- 模 $n$ 的长度要求至少是512比特。在最近一段时间里， $n$ 取在1024到2048位是合适的。
- 没有产生任意的大素数的有用技术，通常的作法是随机选取一个需要的数量级的奇数并检验这个数是否是素数。若不是则挑选下一个随机数直至检测到素数为止。

# ECC公钥密码算法

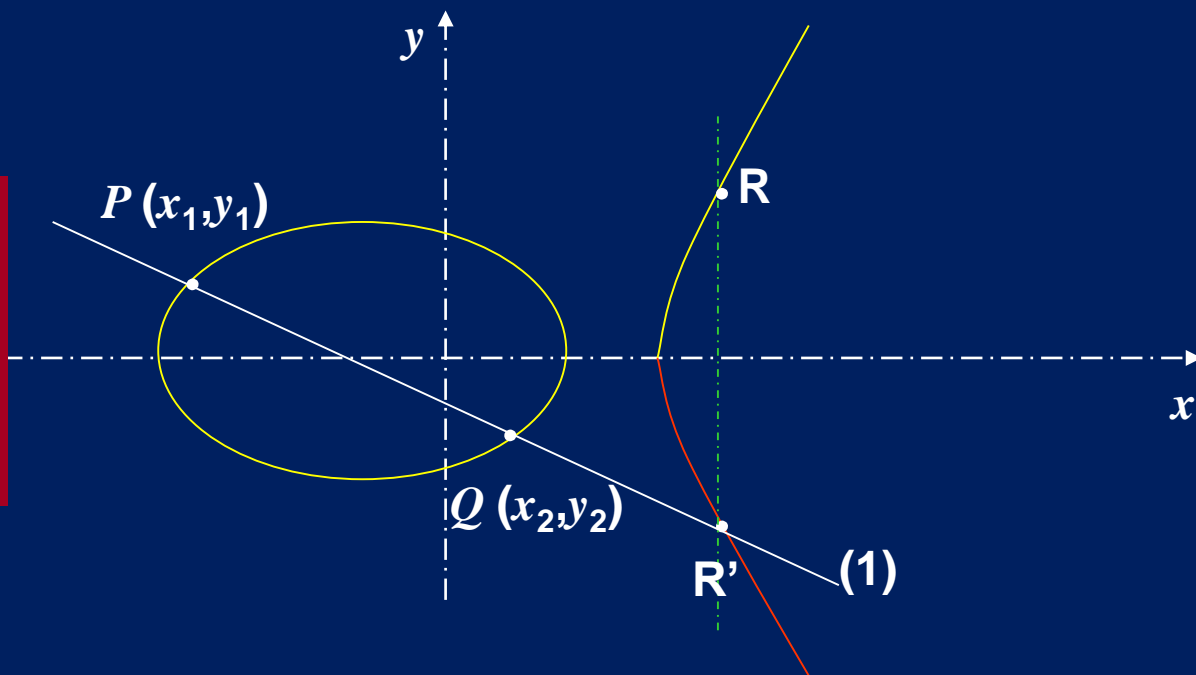
- 1985年由Koblitz和Miller提出
- 设 $a, b \in \mathbf{R}$ 是满足 $4a^2 + 27b^3 \neq 0$ 的实数。方程 $y^2 = x^3 + ax + b$ 的所有解 $(x, y) \in \mathbf{R} \times \mathbf{R}$ 的集合 $E$ ，加上一个无穷远点 $O$ ，组成了一个非奇异椭圆曲线。（充要条件）

三种情况:

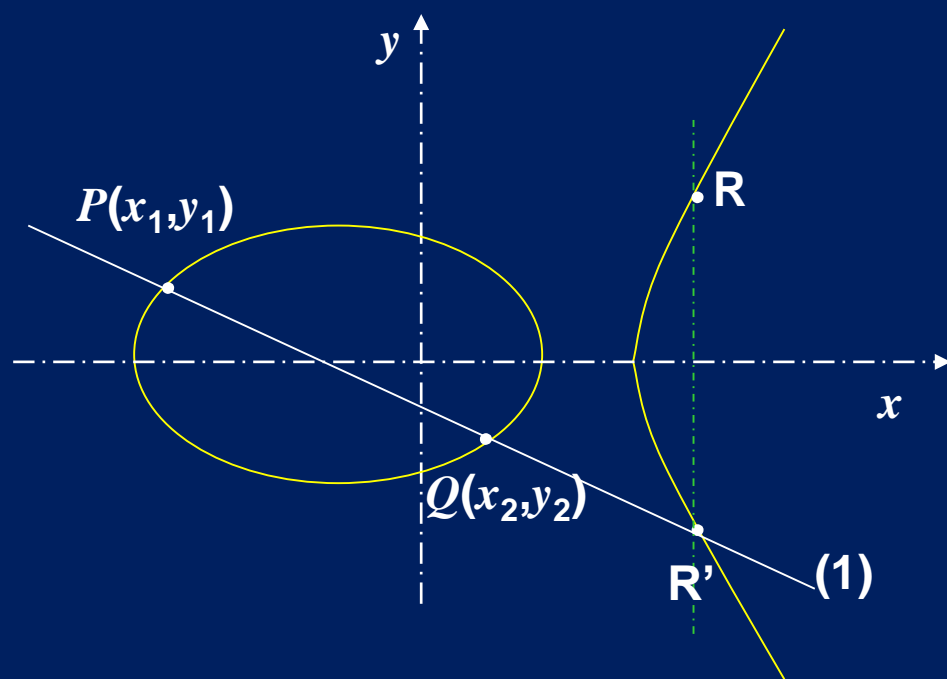
(1)  $x_1 \neq x_2$

(2)  $x_1 = x_2, y_1 = -y_2$

(3)  $x_1 = x_2, y_1 = y_2$



# ECC情形( 1 ): $x_1 \neq x_2$



定义:  $P+Q=R$

$$(x_1, y_1) + (x_2, y_2) = (x_3, y_3)$$

P、Q、R的坐标关系为:

$$(x_1, y_1) + (x_2, y_2) = (x_3, y_3)$$

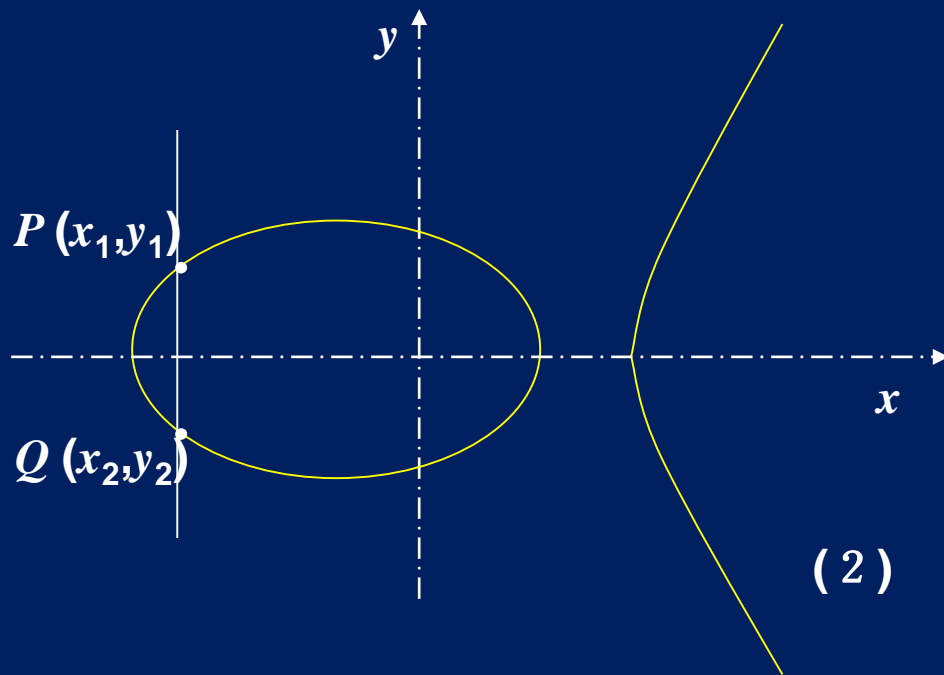
$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}$$

$$x_3 = \lambda^2 - x_1 - x_2$$

$$y_3 = \lambda(x_1 - x_3) - y_1$$



# ECC情形( 2 ): $x_1 = x_2, y_1 = -y_2$

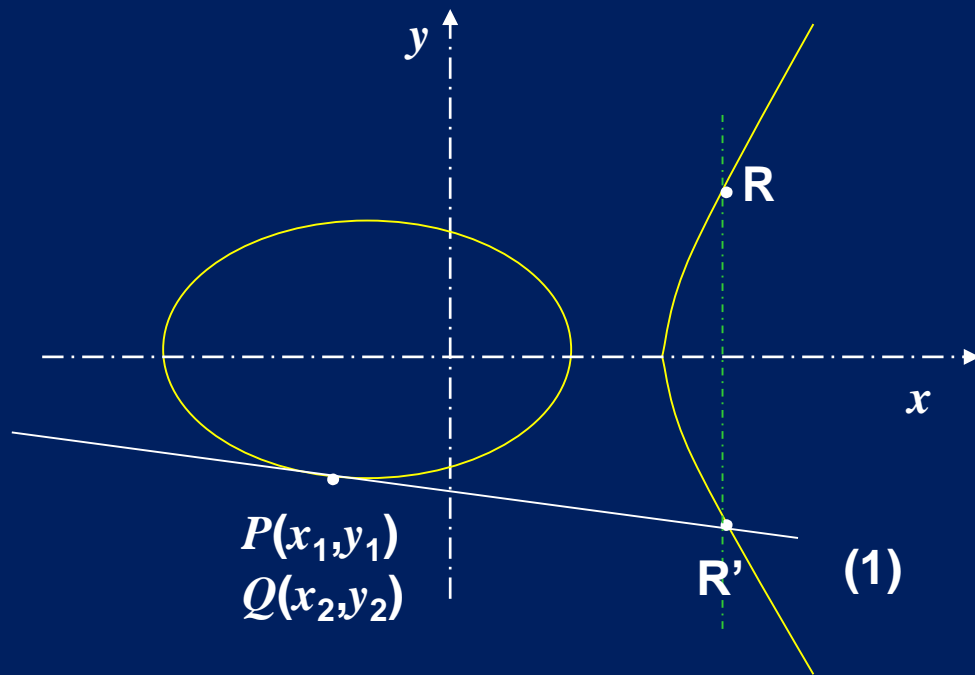


定义:  $P+Q=o$

$$(x, y) + (x, -y) = o$$

互逆运算。

# ECC情形( 3 ): $x_1=x_2, y_1=y_2$



定义:  $P=Q$ , 且为椭圆曲线的切点,  
此时切线的斜率 $\lambda$ 可通过微分求得。

由 $P(x_1, y_1)$ 点影射到 $R(x_3, y_3)$ :

$$\lambda = \frac{3x_1^2 + a}{2y_1}$$

$$x_3 = \lambda^2 - 2x_1$$

$$y_3 = \lambda(x_1 - x_3) - y_1$$

# ECC加密算法运算规则

- 加法规则

实际上是由P到R的位置映射关系：

(1)  $P+O=O+P=P, P+(-P)=O$

(2)  $P \neq Q$ : 用P、Q的连线求斜率 $\lambda$ ，找出 $R' \rightarrow R$ 点

(3)  $P=Q$ : 用P点的切线求斜率 $\lambda$ ，找出 $R' \rightarrow R$ 点

(4)  $(s+t)P=sP+tP$

- 乘法规则

$kP=P+P+\dots+P$  用位置映射关系计算

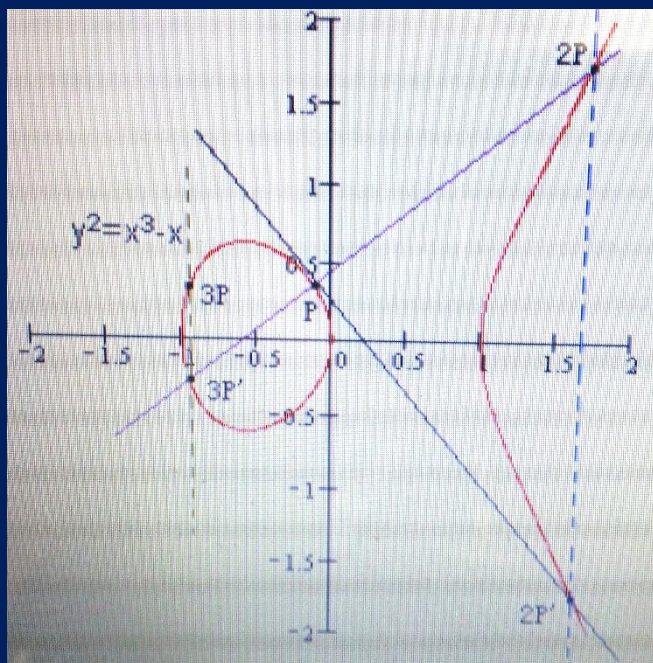
# ECC ElGamal加密算法

- 令信息 $M=(x,y)$ 为椭圆曲线上的点;
- 生成一个私钥  $\{u\}$ ;
- 取一个生成元 $\alpha = (c,d)$  , 计算公钥 $v = u\alpha$ ;
- 随机选择一常数 $k$ ;
- 加密:  $P = e_k(M, k, v) = \{y_1, y_2\}$ ;  
$$y_1 = k\alpha, y_2 = M + kv$$
- 解密:  $S = d_k(y_1, y_2) = y_2 - uy_1$ 。

# ECC加解密实例

- 已知明文  $M=(10,9)$  (椭圆上的一点)
- 私钥:  $u=7, \alpha=(2,7)$
- 公钥: 随机取  $k=3$ , 计算出:  $\mathbf{v}=u\alpha=(7,2)$ 。
- 加密运算:  $\mathbf{y}_1=k\alpha=3(2,7)=(8,3)$   
 $\mathbf{y}_2=M+k\mathbf{v}=(10,9)+3(7,2)=(10,2)$
- 解密运算:  $\mathbf{S}=\mathbf{y}_2-u\mathbf{y}_1$   
 $= (10,2)-7(8,3)$   
 $= (10,9)=M$

# ECC安全性



$$v = u\alpha = \underbrace{\alpha + \alpha + \dots + \alpha}_{u \uparrow}$$

私钥  $\{u, \alpha\}$  ; 公钥  $\{k, v\}$

(1) 已知  $k$ , 接收到  $k\alpha$ , 求  $\alpha$  很困难;

(2) 已知:  $v = u\alpha$ , 导出  $u$  是很困难的。

正向计算很容易, 当  $u$  较大时, 反向计算很困难。

# ECC与RSA的比较

破解所需时间/(MIPS 年)	RSA 密钥大小	ECC 密钥大小	RSA/ECC 密钥大小之比
104	512	106	5 : 1
108	768	132	6 : 1
1 011	1 024	160	7 : 1
1 020	2 048	210	10 : 1

**MIPS年：每秒执行1百万条指令，执行1年。**

# 消息摘要

篡改——对信息完整性的主动攻击

例：A送信给B，约定18点见面，M截获此信后将18点改为22点，再传给B；B收到信后未察觉信被改，故深信是A所约时间。结果使A白等一场。

抗击手段——用消息认证来防止攻击者篡改信息或破坏信息的完整性。



## 假冒——对信息来源真实性的主动攻击

例：M冒充A发信给B，B深信发信方是A。

抗击手段——用数字签名来防假冒，以保证消息来源的真实性。防止假冒和抵赖。

A handwritten signature in black ink on a white background. The signature is stylized and cursive, consisting of several loops and a long horizontal stroke at the end.

只有发信人才有的签名。

# 消息认证和数字签名的主要手段

- ❑ 用消息本身来产生消息认证信息或数字签名是费时费力的
- ❑ 考虑用消息的摘要（长度极短且固定）来产生消息认证信息或数字签名
- ❑ 要求由不同的消息产生出不同的摘要

消息认证和数字签名所依赖的主要手段  
——以Hash函数产生消息摘要

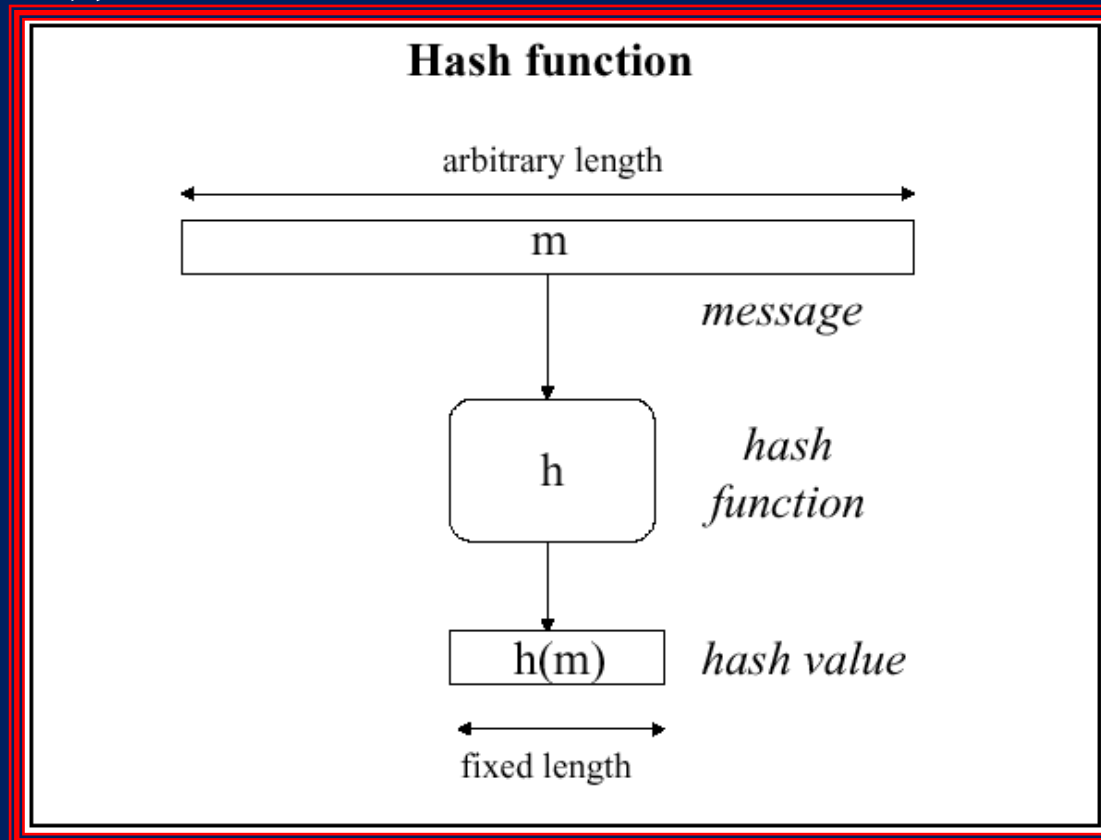
# 消息摘要： Hash函数



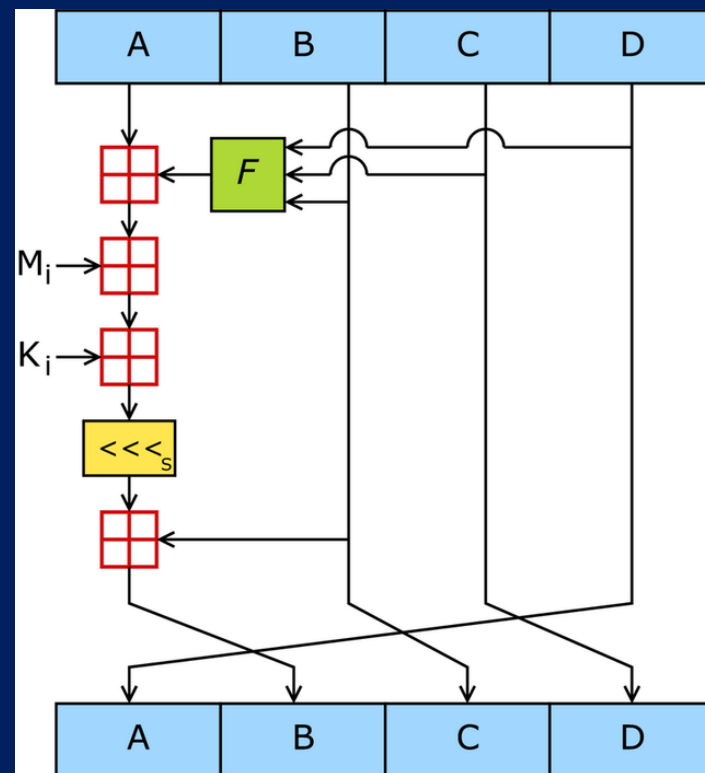
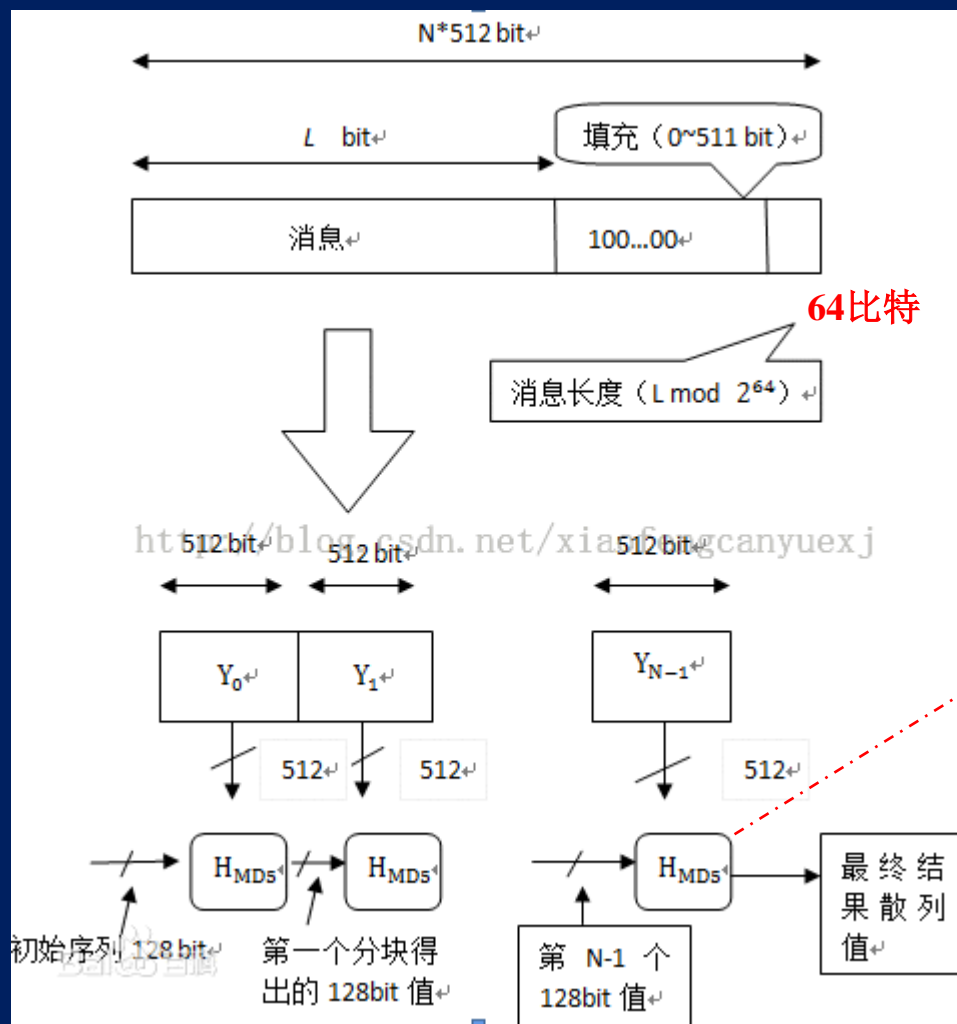
也称哈希函数、散列函数



是一种由不定长的自变量到定长的函数值的单向映射



# MD5 哈希（Hash）算法



$$F(X, Y, Z) = (X \wedge Y) \vee (\neg X \wedge Z)$$




# 关于MD5的碰撞现象

 Hash函数的值称为作为自变量的消息的“散列值”或“消息摘要”、“数字指纹”

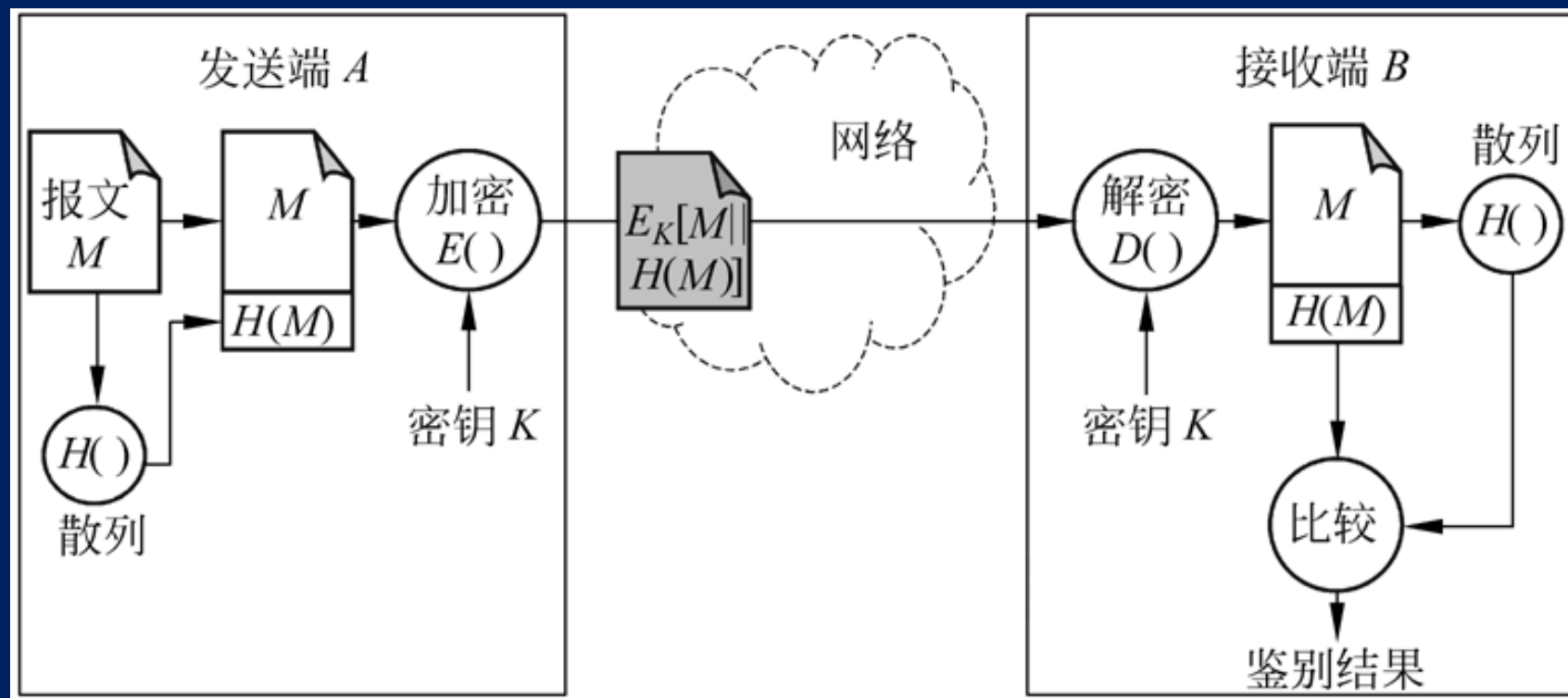


 不同的原像映射出相同的映像称为“碰撞”或“冲突”

# Hash算法的安全性要求

-  单向性——由消息的散列值倒算出消息在计算上不可行
-  抗弱碰撞性——对于任何给定消息及其散列值，不可能找到另一个能映射出该散列值的消息
-  抗强碰撞性——对于任何两个不同的消息，它们的散列值必定不同

# 基本的散列函数报文鉴别过程



# 散列函数的安全性—关于碰撞

- 在50个人当中，出现两个生日相同的人的概率有多大？高达97%。（碰撞概率）
- 在50人当中，有人与你同一天生日，这个概率有多大？ $50/365=7.3\%$ 。

- 生日悖论：在23个人当中出现相同生日的概率大于50%。
- 生日悖论的本质就是，随着元素增多，出现重复元素的概率会以惊人速度增长，而我们低估了它的速度。

从均匀分布的区间 $[1,d]$ 中，取出 $n$ 个整数，其中有两个相同的概率：

$$P(d, n) = 1 - \prod_{i=1}^{n-1} \left(1 - \frac{i}{d}\right)$$

当 $P=0.5$ 时， $n \approx \sqrt{d}$ . 上例中  $d=365, n=23$



# 采用生日攻击方法 可以大大减少攻击次数

- 考虑一个64位散列函数，它有 $2^{64}$ 种可能的散列值，要想100%地找到一组碰撞，就需要 $2^{64}$ 次攻击。但是基于生日攻击的原理，我们只需要 $2^{32}$ 次攻击，就有约50%的概率能够攻击成功。
- 生日攻击的前提就是存在碰撞。
- 为有效抵御生日攻击，必须使散列码的位数充分大，使得获得碰撞在计算上是不可能的。
- 著名的SHA-1散列算法的散列码取160位，由生日原理攻击者至少得算出 $2^{80} \approx 1.2 \times 10^{25}$ 个散列码才有机会遇到碰撞（如以每秒算1000万个散列码计算，需380亿年）。

# 数字签名的要求

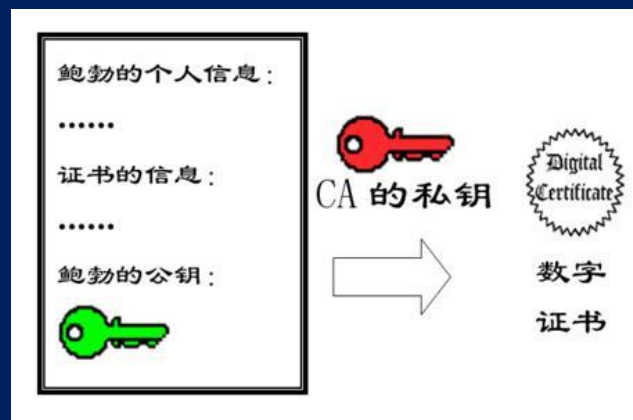
- 传统签名的基本特点：

- 能与被签的文件在物理上不可分割
- 签名者不能否认自己的签名
- 签名不能被伪造
- 容易被验证



- 数字签名是基本要求：

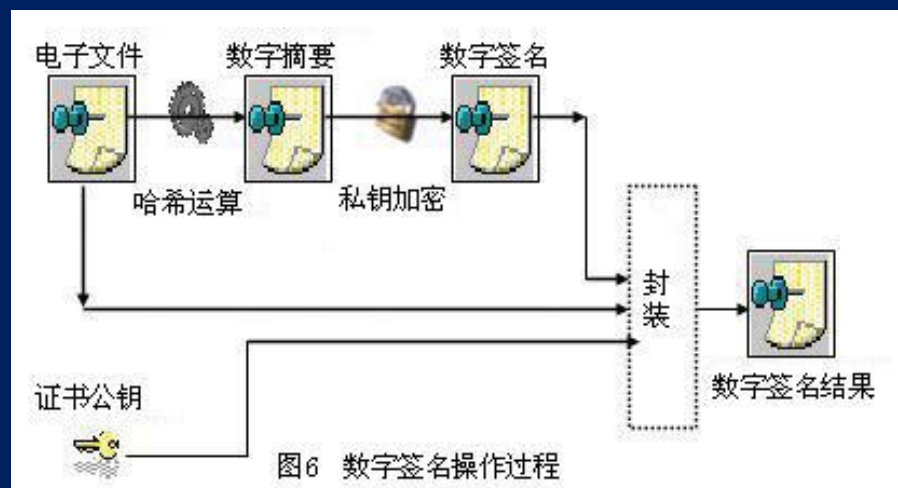
- 签名必须是与消息相关的二进制位串
- 签名者不能否认自己的签名
- 签名的产生和识别比较容易
- 收方对已收到的签名信息不能否认
- 伪造数字签名在计算上是不可行的



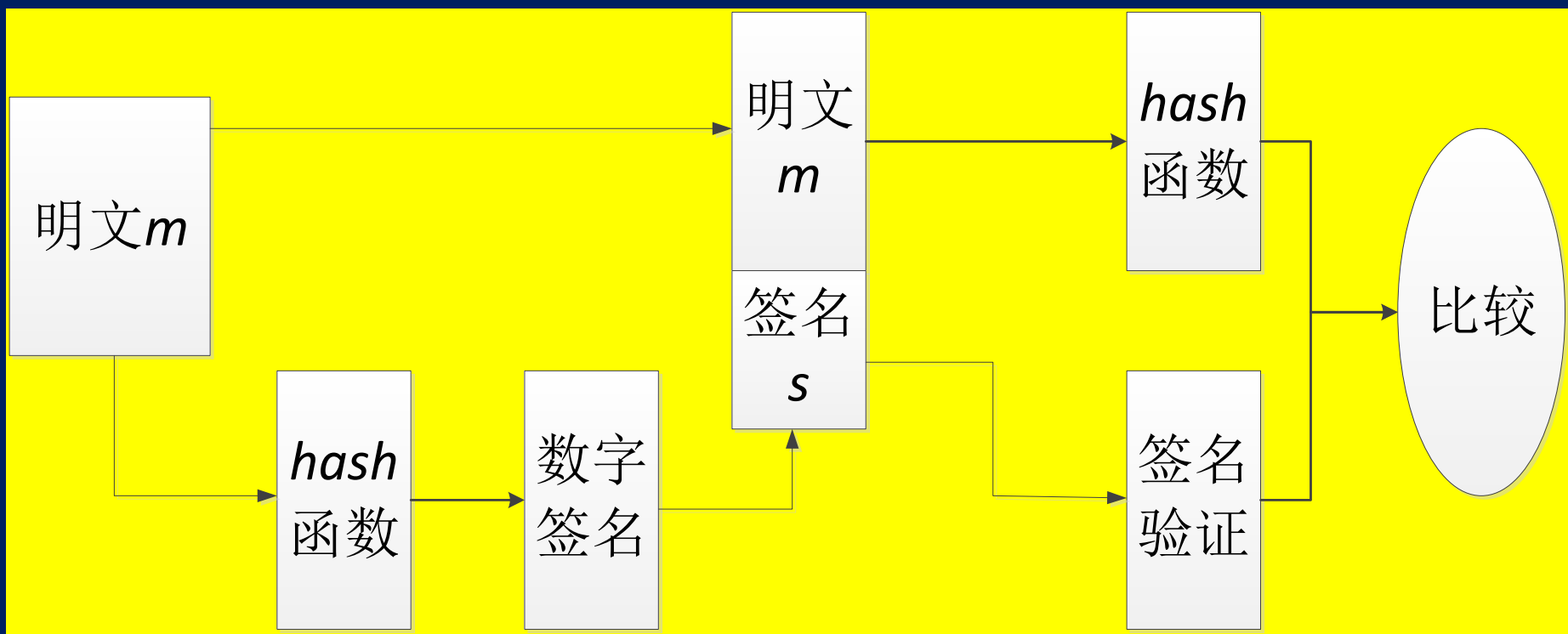
**CA：证书的认证机构，可信第三方**

# 数字签名体制

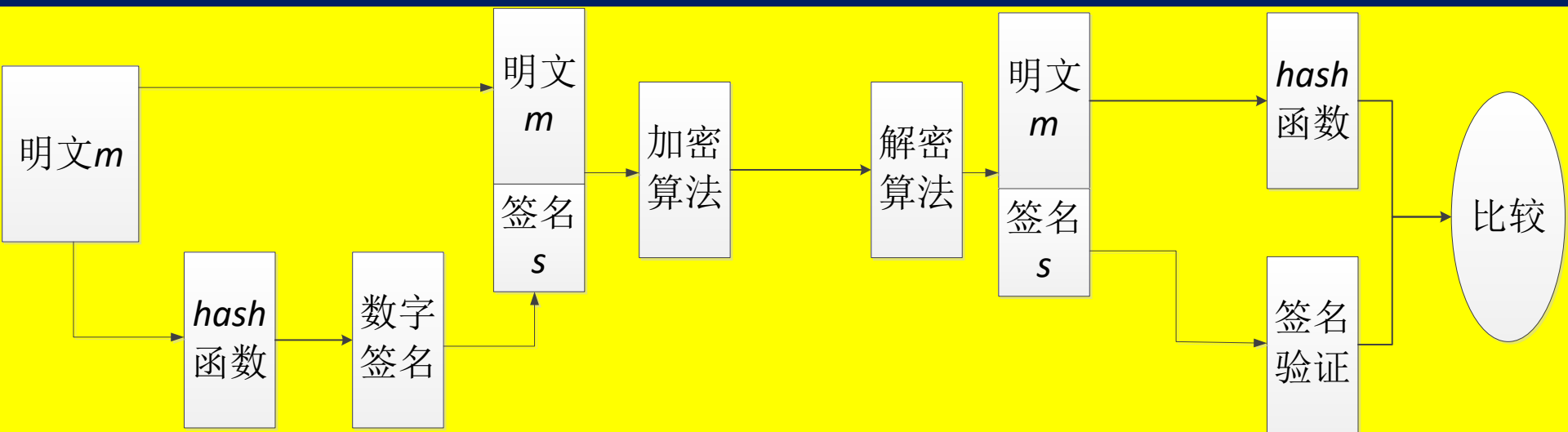
- 签名算法(Signature Algorithm)
  - $\text{Sig}(M)=S$
  - 签名算法或签名密钥K是秘密的，只有发信方掌握。
- 验证算法(Verification Algorithm)
  - $\text{Ver}(S)=\{0, 1\}=\{\text{真}, \text{伪}\}$
  - 验证算法公开，便于他人进行验证
- 签名体制的安全性在于，从M和其签名S难以推出签名密钥K或伪造一个M'使M'和S可被证实为真。



# Hash函数和数字签名的结合应用



# 保密性和数字签名的结合应用



# 基于公钥密码的基础设施PKI

- 目的：把基于公钥的安全服务独立于应用进行设计，使其标准化；
- 作用：提供证书的认证服务。



用户不用知道电是如何产生的，只是用电罢了。



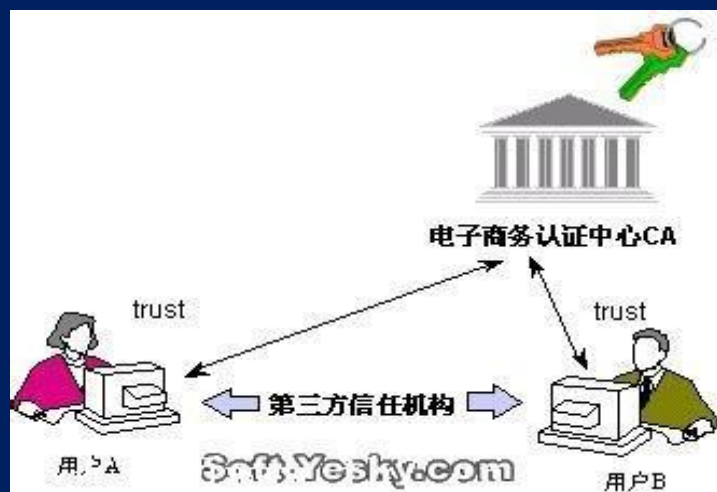
CA是PKI的运营者

# PKI的目的与作用

- Public Key Infrastructure
- 实现和提供具有普适性的安全基础设施，通过延伸到用户本地的接口，为网络应用提供全面的安全服务；
- 支持公开密钥管理，并支持真实性、保密性、完整性和不可否认性；
- 安全应用程序的开发者不用关心密码学的繁杂问题，直接使用标准接口享用PKI基础设施提供的服务；
- PKI与应用的分离，正如电力基础设施与电器的分离。

# PKI的目的与作用（续）

- 主要解决密钥属于谁，即密钥认证问题；
- 通过数字证书，PKI很好地证明密钥是谁的；
- PKI的核心技术主要是围绕数字证书的申请、颁发、使用、撤销等整个生命周期展开的。
- 多个PKI域之间的互联是关键问题，信任互联技术使得建立复杂的网络信任体系成为可能。



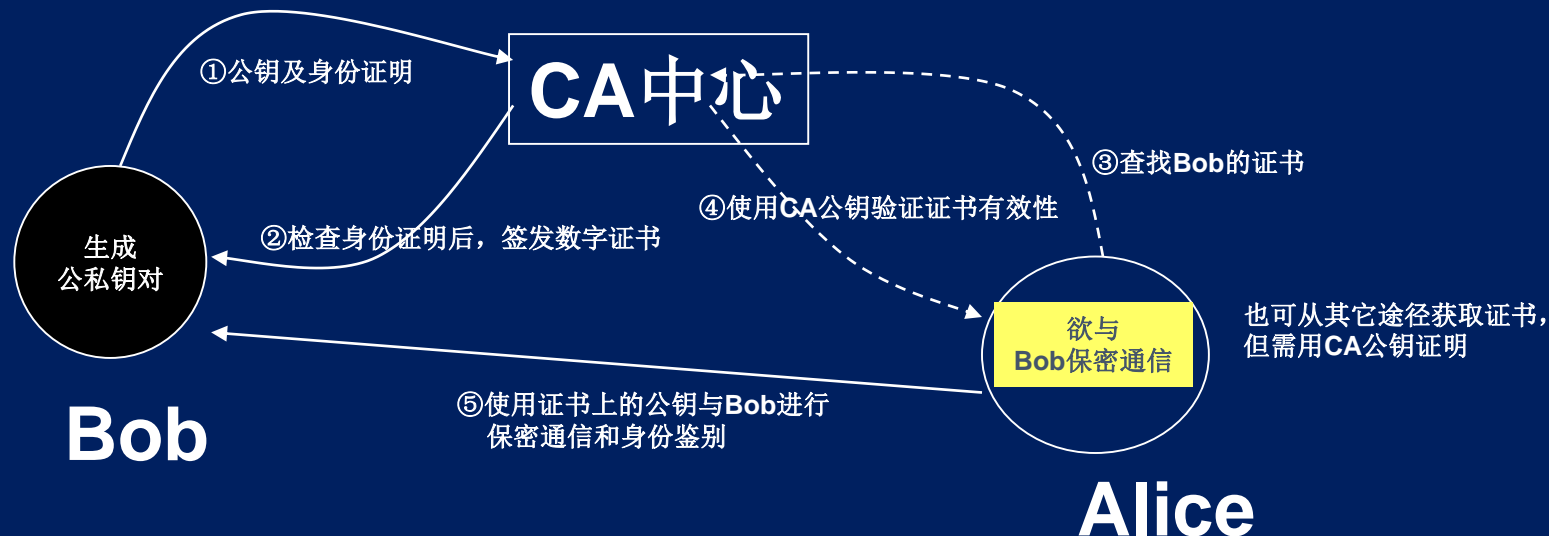


# PKI的技术优势

- 采用公钥密码技术
- 保护机密性（为实体间的保密通信提供支持）
- 采用数字证书方式进行服务
- 提供数字证书撤销机制
- 具有极强的互连能力

# PKI的基本结构

## • PKI的工作过程



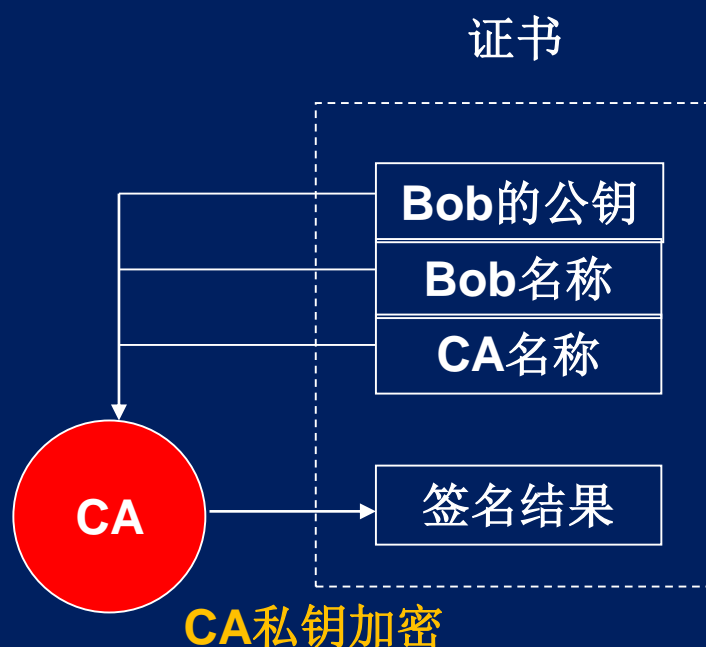
三种不同功能的实体:

- 1、证书认证中心 (CA)
- 2、证书持有者 (Bob)
- 3、证书依赖方 (Alice)

公钥对应的私钥持有者。

使用证书获取安全服务者

# CA签发证书过程



该证书必须从可靠的途径获得，如：  
用户申请CA服务时  
面对面直接获取；  
CA通过政府红头文件方式发布。

# 几种典型PKI标准

- X.509
- PKIX
- 中国商用PKI系统标准
- 美联邦MISPC PKI
- RSA公司数字证书解决方案
- Entrust Authority PKI
- 中科院ARP CA系统
- 微软公司PKI解决方案
- PGP

# X.509

- 国际电联电信标准部(ITU-T) 设计的PKI标准。
- ITU于1988年制定了X.500系列标准，提供公用网络用户目录信息服务。X.509则为X.500用户名称提供了通信实体鉴别机制，并规定了实体鉴别过程中广泛适用的证书语法和数据接口。
- X.509给出的鉴别框架是一种基于公开密钥体制的鉴别业务密钥管理。
- X.509 数字证书不仅包含用户名和公钥，而且还包含与用户有关的其他信息，包括电子邮件地址、授权对具有某种价值的文档进行签名、授权成为CA 并为其他证书签名等等。
- X.509 证书和许多其他证书都有有效期限。CA 保存并分发一个吊销证书的列表，即证书吊销列表 (CRL)。网络用户访问 CRL 以确定证书的有效性。
- 目前， X.509 标准已用于许多网络安全应用程序，其中包括IP 安全（IPsec）、安全套接层（SSL）、安全电子交易（SET）、安全多媒体INTERNET邮件扩展（S/MIME）等。

# X.509数字证书内容

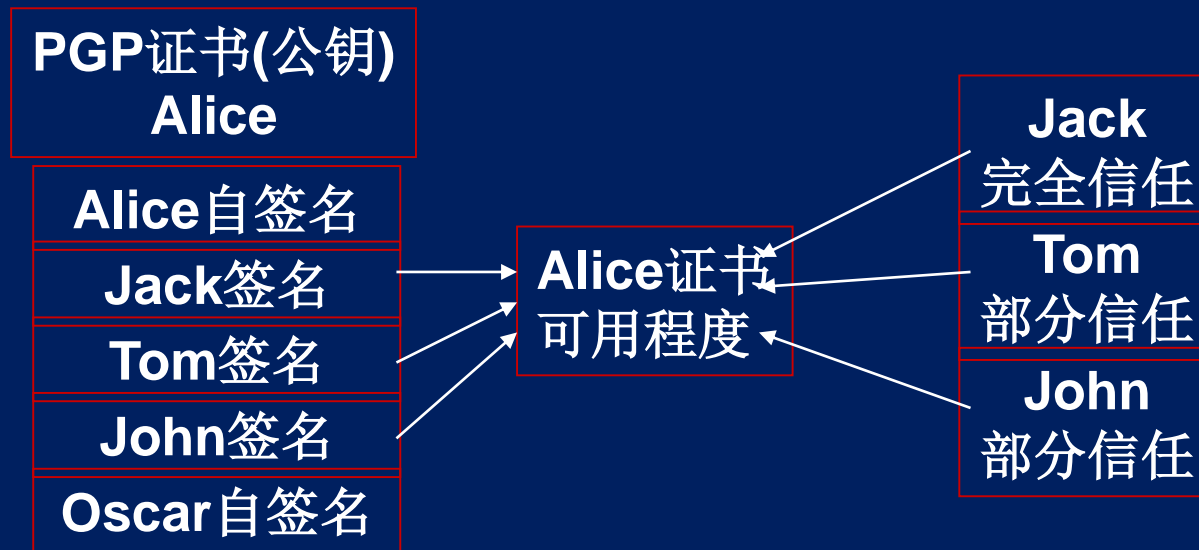
- 证书的版本信息;
- 证书的序列号, 每个证书都有一个唯一的证书序列号;
- 证书所使用的签名算法;
- 证书的发行机构名称, 命名规则一般采用X.500格式;
- 证书的有效期, 现在通用的证书一般采用UTC时间格式;
- 证书所有人的名称, 命名规则一般采用X.500格式;
- 证书所有人的公开密钥;
- 证书发行者对证书的签名。

## Ukey的内容

- MCU
- 数字证书
- 公钥密码算法
- 公私钥对
- 对称密码算法
- 随机数发生器
- 摘要算法
- 。 。 。

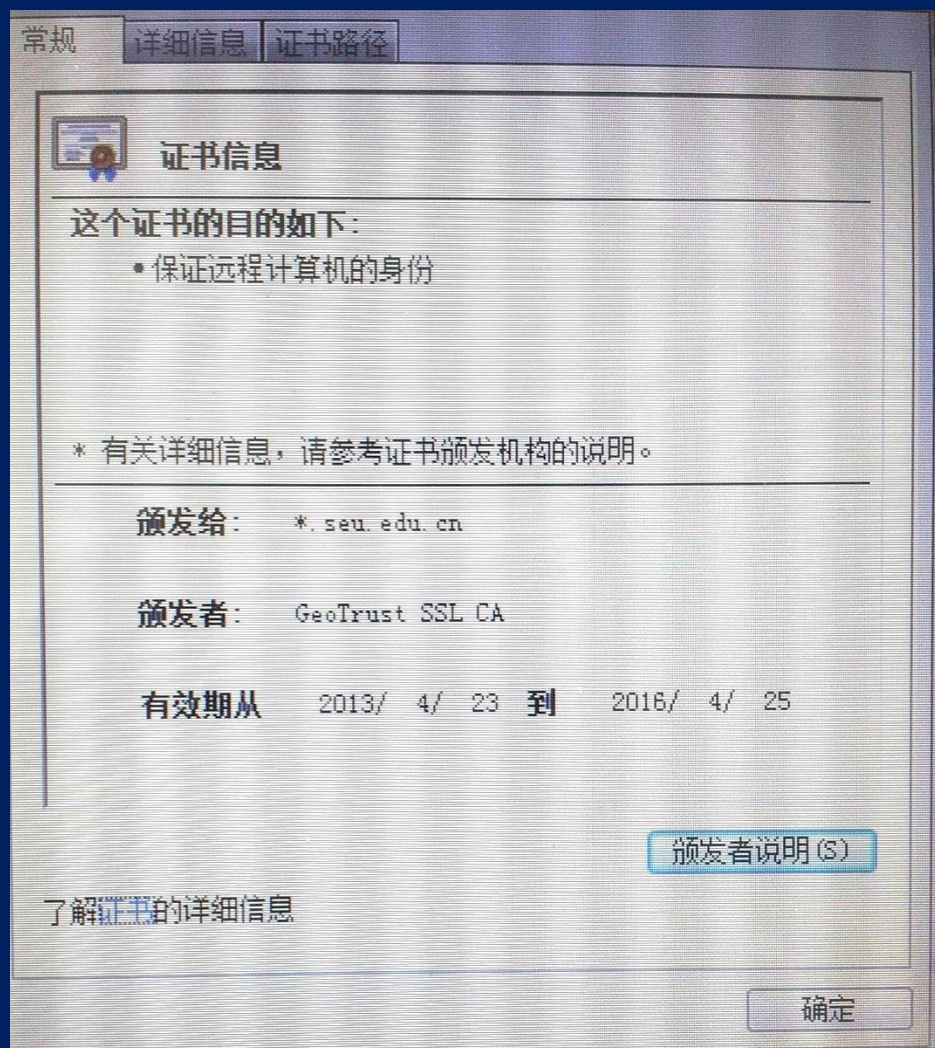
# PGP (Pretty Good Privacy)

- 基于公钥密码的电子邮件和文件存储应用系统



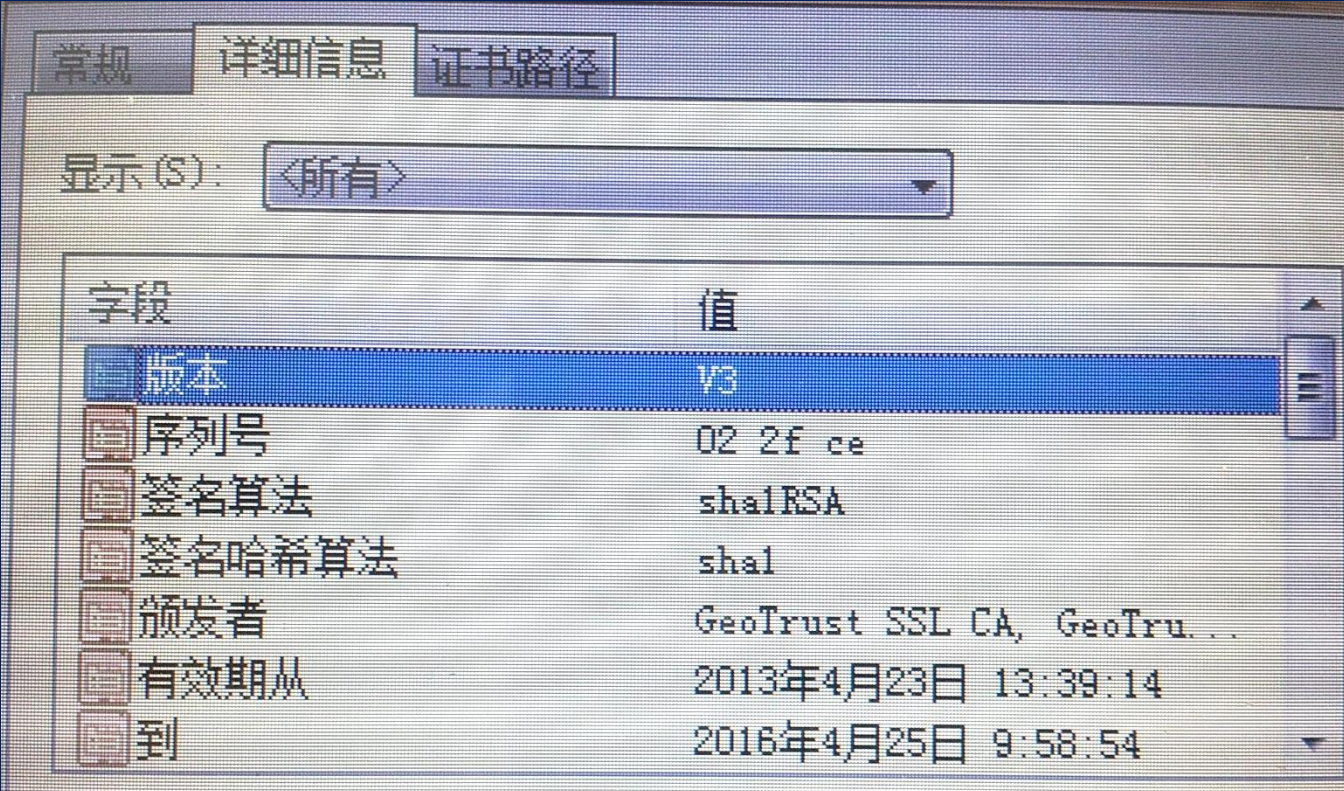
靠人之间的信任来自组管理, 无需CA服务

# Web证书:





# 证书内容:



The screenshot shows a window with three tabs: '常规' (General), '详细信息' (Details), and '证书路径' (Certificate Path). The '详细信息' tab is active. Below the tabs is a dropdown menu labeled '显示(S):' with the value '<所有>' (All). Below this is a table with two columns: '字段' (Field) and '值' (Value). The table contains the following entries:

字段	值
版本	V3
序列号	02 2f ce
签名算法	sha1RSA
签名哈希算法	sha1
颁发者	GeoTrust SSL CA, GeoTru...
有效期从	2013年4月23日 13:39:14
到	2016年4月25日 9:58:54



# 证书包含的信息：

## 证书概述

公钥证书，通常简称为证书，是一种数字签名的声明，它将公钥的值绑定到持有对应私钥的个人、设备或服务的标识。证书的主要好处之一是主机不必再为单个使用者维护一套密码，这些单个使用者进行访问的先决条件是需要通过身份验证。相反，主机只需在证书颁发者中建立信任。

大多数普通用途的证书基于 X.509 v3 证书标准。

通常，证书包含以下信息：

- 使用者的公钥值。
- 使用者标识信息（如名称和电子邮件地址）。
- 有效期（证书的有效时间）。
- 颁发者标识信息。
- 颁发者的数字签名，用来证明使用者的公钥和使用者的标识符信息之间的绑定的有效性。

证书只在对其指定的时间段内有效；每个证书都包含“有效起始日期”和“有效终止日期”，这两个值设置有效期的期限。一旦到了证书的有效期，到期证书的使用者就必须申请一个新的证书。

- [使用证书](#)



# 证书的用途：

## 使用证书

证书可以用于下列用途：

- 身份验证，验证某人或某物的身份。
- 隐私，确保该信息仅可用于指定用户。
- 加密，伪装信息，以使未授权的读者无法将其解密。
- 数字签名，提供非拒绝和消息完整性。

这些服务对于您的通信安全非常重要。此外，许多应用程序使用证书，例如电子邮件应用程序和 Web 浏览器。

## 身份验证

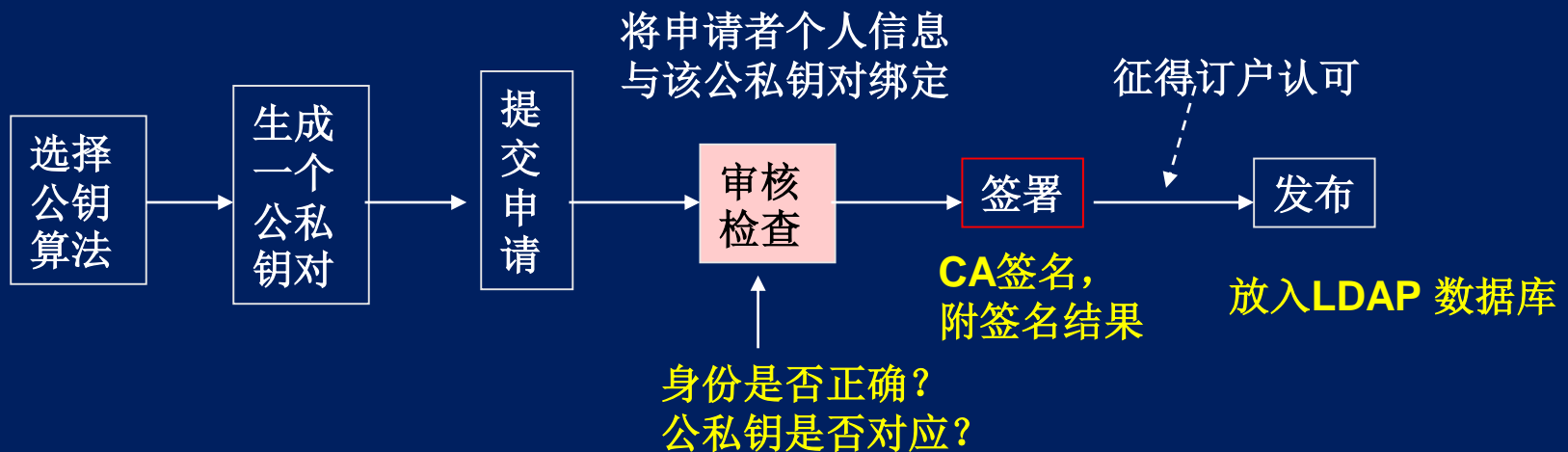
身份验证对于保证通信安全十分重要。用户必须可以向与其通信的人证明自己的身份，并且必须能够验证他人的身份。网络上的身份验证很复杂，因为通信各方在通信时物理上并不在一起。这会让不道德者有机会截获消息或冒充他人或实体。

# 证书的生命周期

- 随着时间的推移，密钥的安全性降低
- 订户个人的身份信息随时间变化
- 证书服务的时效性

# 证书的产生

- 提交证书申请材料
- CA(或RA)审核材料
- 签发证书



# 证书的存储

订户的证书一般存在于：

- 文件形式
- USB Key
- IC Card
- SD Card



# 密钥管理中心KMC

- KMC是非技术因素产生的，而是管理和法律的要求；
- 与CA相连
- 但CA的运营与KMC的运营是分开的，为安全；
- 支持多种加密算法，能够生成这些算法对应的公私钥对；
- 重要环节是 真随机数的生成：热噪声等

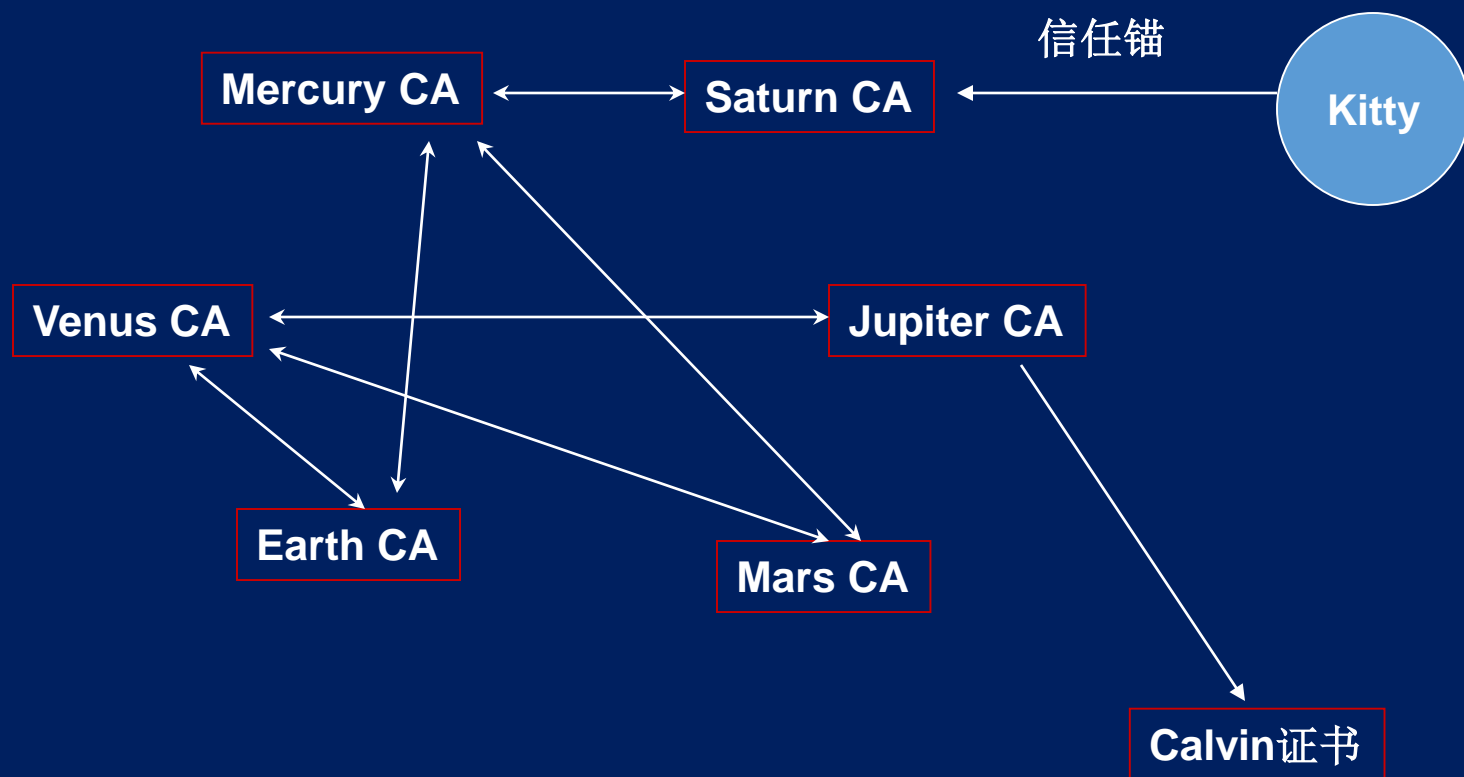
# PKI互联

## 解决信任互联问题

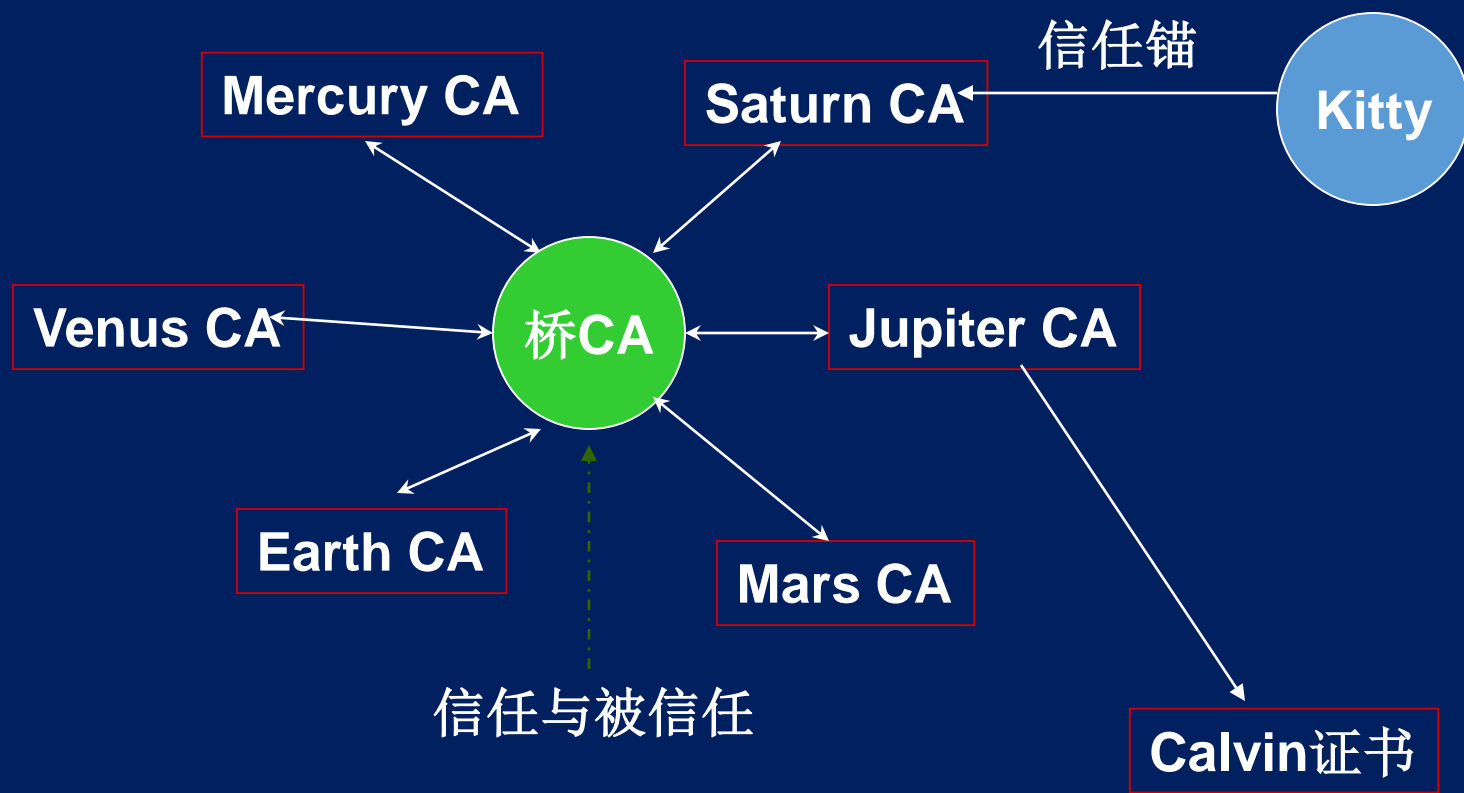
- 严格层次方案：把一个CA置于另一个CA之下，成为其子CA；
- 列表方案：用户添加多个根CA自签名证书，支持大规模应用；可以借用权威列表；
- 交叉认证：多个CA之间互相签发证书；
- 桥CA：多个CA之间通过建立桥CA，成为信任方和被信任方。



# 交叉认证



# 桥CA



# PKI核心服务

- 机密性：使用公钥加密对称密钥
- 数字签名：用私钥进行数字签名
- 数据完整性：用私钥加密摘要
- 数据源鉴别：用对方公钥解密摘要，验证数据源
- 身份鉴别：挑战/响应身份鉴别
- 非否认：数字签名
- 时间戳服务：时间信息与原文内容绑定，再进行签名
- ○ ○ ○

# PKI应用

- 支持可信计算
- 支持网上银行
- 支持电子商务/电子支付
- 支持电子政务
- 支持可信网站
- 支持智能卡
- 支持安全信息发布
- ○ ○ ○

# CA（数字证书认证中心）介绍

- 为保证网上数字信息的传输安全，除了在通信传输中采用加密算法等措施之外，必须建立一种信任及信任验证机制，即参加电子商务的各方必须有一个可以被验证的标识，这就是**数字证书**。
- 数字证书是各实体(持卡人/个人、商户/企业、网关/银行等)在网上信息交流及商务交易活动中的身份证明。该**数字证书具有唯一性**。
- 它将实体的公开密钥同实体本身联系在一起，同时数字证书的来源必须是可靠的。这就意味着应有一个网上各方都信任的机构，专门负责数字证书的发放和管理，确保网上信息的安全，这个机构就是**CA认证机构**。
- 各级CA认证机构的存在组成了整个电子商务的**信任链**。如果CA机构不安全或发放的数字证书不具有**权威性、公正性和可信赖性**。

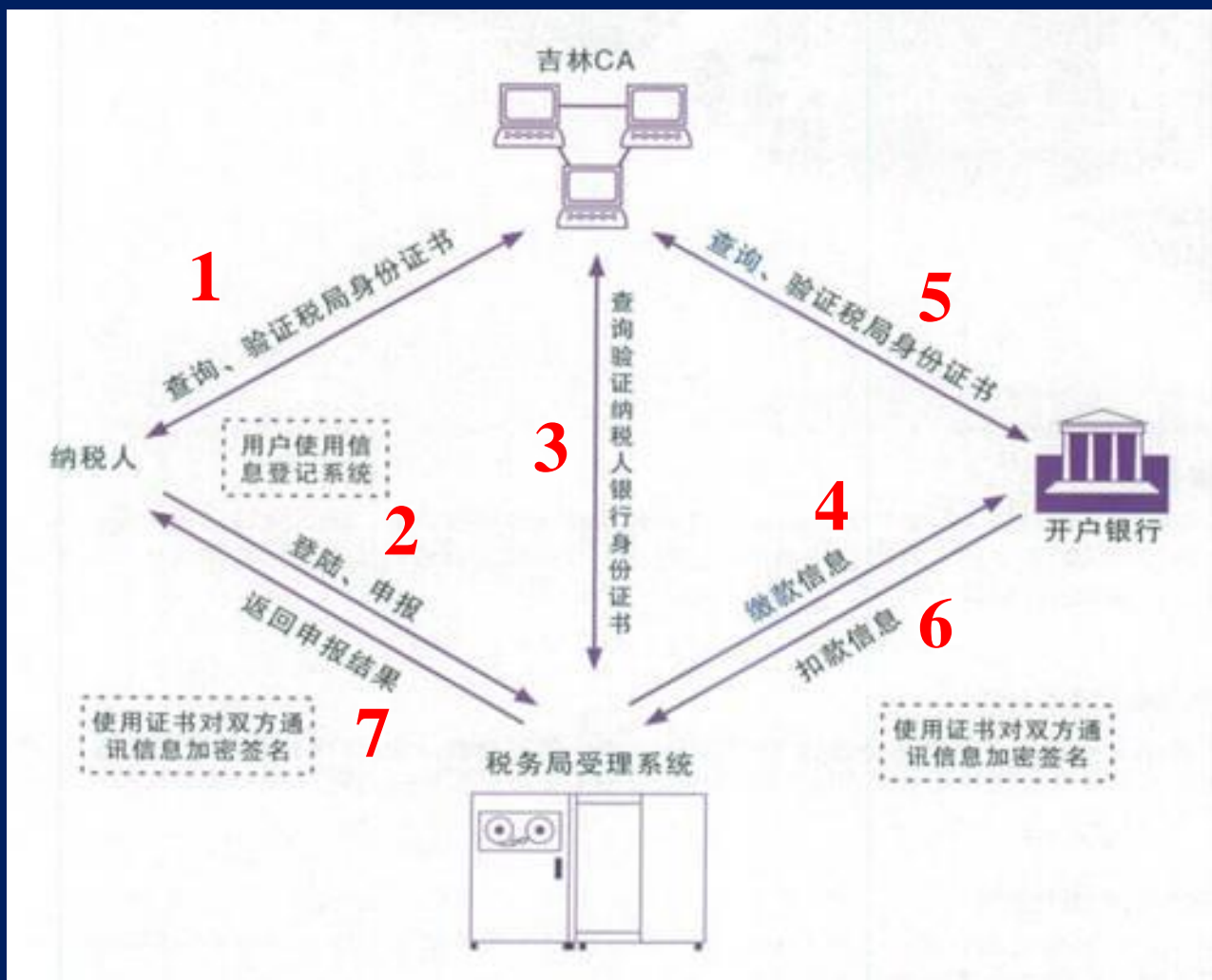


# CA（续）



- 它主要负责**产生、分配并管理**所有参与网上交易的实体所需的身份认证数字证书。
- 每一份数字证书都与上一级的数字签名证书相关联，最终通过安全链追溯到一个已知的并被广泛认为是安全、权威、足以信赖的机构-**根认证中心（根CA）**。
- 电子交易的各方都必须拥有合法的身份，即由数字证书认证中心机构（CA）签发的数字证书，在交易的各个环节，交易的各方都需检验对方数字证书的有效性，从而解决了用户信任问题。
- 数字证书认证解决了网上交易和结算中的安全问题。
- 认证中心（CA），通过自身的注册审核体系，检查核实进行证书申请的用户身份和各项相关信息，使网上交易的用户属性客观真实性与证书的真实性一致。

# 基于CA的报税纳税过程



# 习题

- 1、某用户选取 $p=11$ 和 $q=17$ 作为模数 $n=pq$ 的RSA公钥体制的两个素数，选取 $e=17$ 作为公钥。请给出用户的私钥，及对消息 $m=10$ 的加密结果。
- 2、设素数 $p=23$ ，有限域 $F_{23}$ 上的椭圆曲线方程 $y^2=x^3+x+4$ ，且已知 $P=(4,7)$ 、 $Q=(13,11)$ 为该椭圆曲线上的两点，试求 $P+Q$ 。