北京郵電大學
BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

《现代密码学》第七章

公钥加密体制（一）

信息安全中心

上讲内容回顾

- Hash函数的定义及安全目标
- Hash函数的发展现状
- Hash函数的构造
- 消息鉴别码的定义及安全目标
- 消息鉴别码的发展现状
- 消息鉴别码的构造
- 认证加密模式

本章主要内容

- 对称密码体制面临的问题
- 公钥密码体制的发展
- 单向陷门函数及构造
- RSA加密算法及其应用
- ElGmal加密算法
- 椭圆曲线加密码体制简介

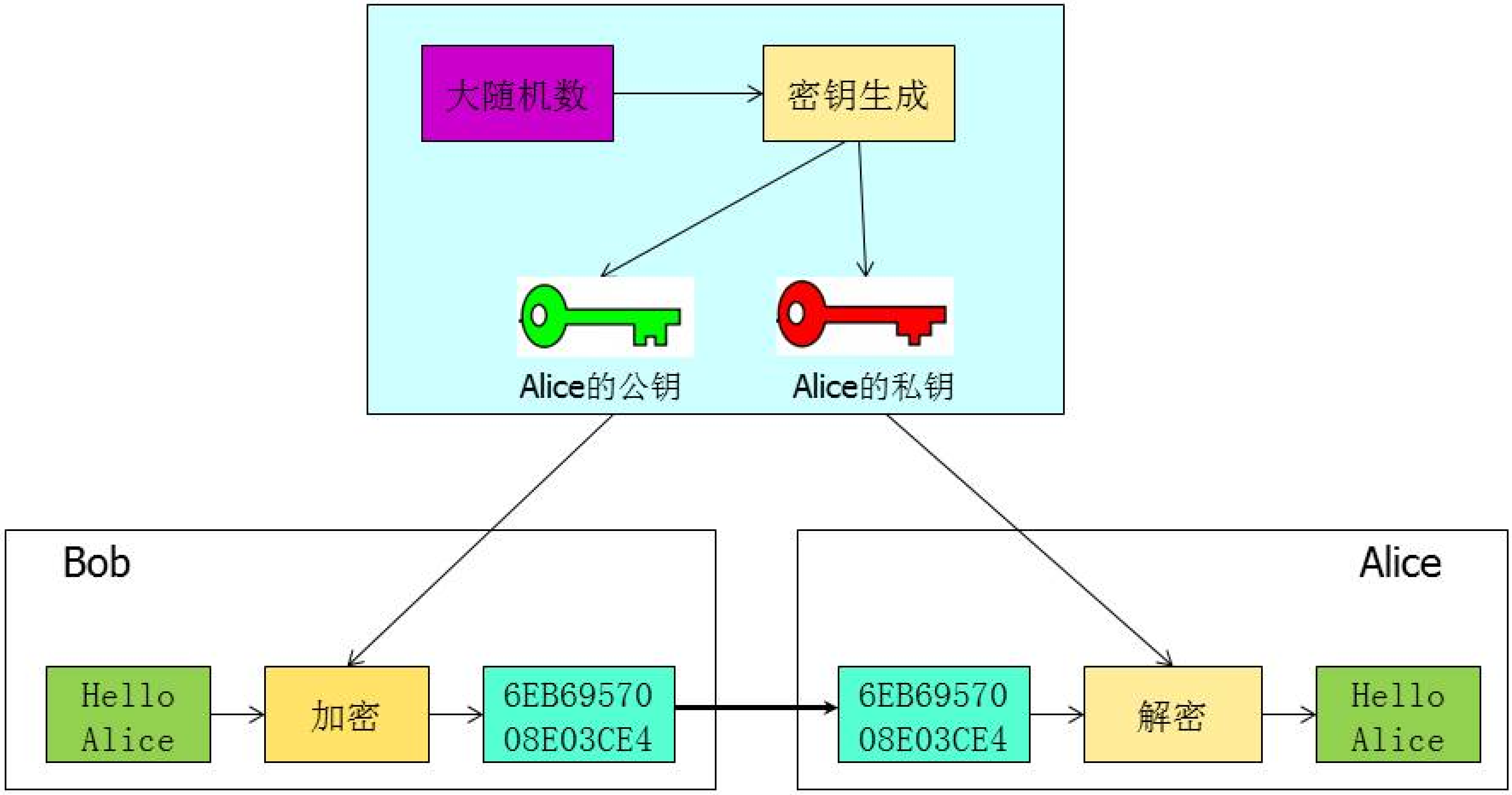
本章主要内容

- 对称密码体制面临的问题
- 公钥密码体制的发展
- 单向陷门函数及构造
- RSA加密算法及其应用
- ElGmal加密算法
- 椭圆曲线加密码体制简介

对称密码体制面临问题

- **初始密钥分配**：对称密码体制，发送方指定一个（种子）密钥后，必须得想方设法把密钥告知接收方，怎样确保“告知过程”密钥不泄露？
- **密钥管理**：在有n个用户的网络中，若需要两两用户安全通信，则每对用户需要共享独立的秘密密钥，网络中需要管理的密钥总数是 n*（n-1）/2
- **不可抵赖性**：当主体A收到主体B的电子文档（电子数据）时，无法向第三方证明此电子文档确实来源于B。

公钥加密模型


大随机数
密钥生成
Alice的公钥
Alice的私钥
Bob
Hello Alice
加密
6EB69570 08E03CE4
Alice
6EB69570 08E03CE4
解密
Hello Alice

公钥加密模型

- 密钥生成过程：接收消息的端系统（如图中的接收者Alice）产生一对密钥(PK_A, SK_A)，PK_A是公开钥（用于加密），SK_A是秘密密钥（用于解密）.

- 加密过程：Bob想向Alice发送消息m，则获取Alice的公开密钥PK_A，加密得密文$c = E_{PK_A}[m]$，其中E是加密算法.

- 解密过程：Alice收到密文c后，用自己的秘密钥SK_A解密，表示为$m = D_{SK_A}[c]$，其中D是解密算法.

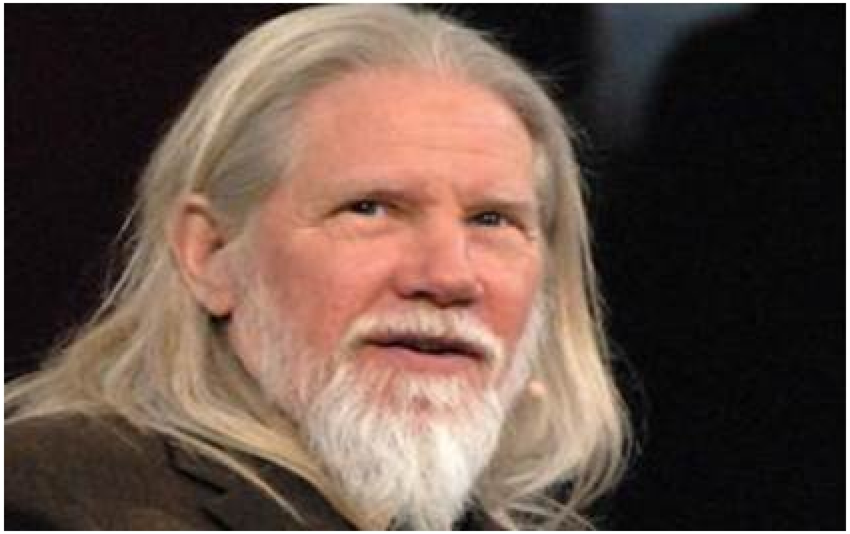
本章主要内容

- 对称密码体制面临的问题
- 公钥密码体制的发展
- 单向陷门函数及构造
- RSA加密算法及其应用
- ElGmal加密算法
- 椭圆曲线加密码体制简介

公钥密码体制的发展

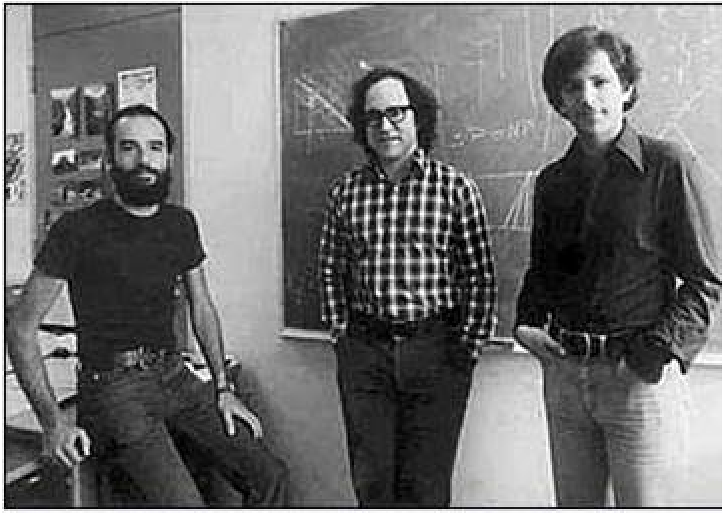
- 1976年 Diffie和Hellman 在《密码学的新方向》中首次公开提出了非对称密码算法的思想，但是没有实现加密方案，只给出一个密钥协商协议；
- 1978年 Rivest,Shamir和Adleman提出应用广泛的RSA算法；
- 1984年 Shamir提出基于身份的密码体制，没有实现加密体制，只给出一个基于身份的数字签名算法；
- 2001年 Boneh,Franklin和Cocks分别独立提出基于身份的加密算法
- 2003年 Al-Riyami提出的无证书的密码体制。

公钥密码体制的发展



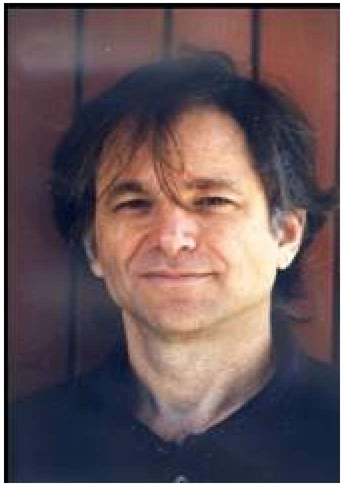
Diffie和Hellman

公钥密码体制的发展



Ronald Rivest, Adi Shamir, and Len Adleman





公钥密码体制的发展

北京邮电大学 BEIJING UNIVERSITY OF POSTS AND TELECOMMUNICATIONS

基于格的密码体制

- Jeffrey Hoffstein (de), Jill Pipher, and Joseph H. Silverman，NTRU, 1996.
- Oded Goldreich, Shafi Goldwasser, and Shai Halevi. "Public-key cryptosystems from lattice reduction problems". In *CRYPTO '97*, pages 112–131, London, UK, 1997.

其它特性

- Sahai, Amit; Brent Waters. "**Fuzzy Identity-Based Encryption**" . Proceedings of Eurocrypt 2005.
- Jump up; Boneh, Dan; Amit Sahai; Brent Waters. "**Functional Encryption**: Definitions and Challenges". Proceedings of Theory of Cryptography Conference (TCC) 2011.
- Jump up ^ Gorbunov, Serge; Hoeteck Wee; Vinod Vaikuntanathan. "**Attribute-Based Encryption** for Circuits". Proceedings of STOC, 2013.
- Craig Gentry. **Fully Homomorphic Encryption** Using Ideal Lattices. In *the 41st ACM Symposium on Theory of Computing (STOC)*, 2009.

本章主要内容

- 对称密码体制面临的问题
- 公钥密码体制的发展
- 单向陷门函数及构造
- RSA加密算法及其应用
- ElGmal加密算法
- 椭圆曲线加密码体制简介

单向陷门函数及构造

定义：一个定义在（X,Y）空间上的单向陷门函数包含3个“有效”算法（G，F，F^{-1}）

- G()：随机算法，输出对(pk, sk)
- F(pk, ·)：确定算法，定义X $\longrightarrow$ Y的运算
- F^{-1}(sk, ·)：F(pk, ·)的逆函数，定义Y $\longrightarrow$ X的运算

单向陷门函数及构造

对于任意生成的一对密钥G →(pk, sk), 单向陷门函数满足:

$$\forall x \in X: \; F^{-1}(sk, \; F(pk, x)) = x$$

定义:(G, F, F^{-1}) 是一个安全的的TDF当对于所有有效的算法A:

$$Adv_{OW}[A,F] = \mathbf{Pr[\, x = x' \,]} < \text{可忽略}$$

单向陷门函数及构造

背包问题： 设背包向量$A=(a_1,a_2,\ldots,a_n)$，s是背包容积. 求A的子集A'，使子集中的元素a_i的和恰好等于s.

例.

A=(43, 129, 215, 473, 903, 302, 561, 1165, 697, 1523)，s=3231. 求A的子集A'使其和为s.

解：3231=129+473+903+561+1165.

所以满足要求的子集合

$$A'=\{129, 473, 903, 561, 1165\}.$$

单向陷门函数及构造

原则上讲，通过检查A的所有子集，总可找出问题的解（如果有解的话）.

如果A中元素个数n很大，子集个数2^n将非常大.

上例中，A的子集共有$2^{10}=1024$个（包括空集）.

目前，寻找满足要求的A的子集没有比穷搜索更好的算法. 因此只要n足够大，那么求解背包问题计算不可行.

单向陷门函数及构造

1）单向函数 F：X $\longrightarrow$ Y.

令函数 f: $\{0,1\}^n \longrightarrow S$，$S$是所有可能的背包容积集合

$\forall\, x \in \{0,1\}^n$，$x$二进制表示为$x=(x_1,x_2,\cdots,x_n)$，$x_i \in \{0,1\}$，$1 \le i \le n$

则，$f(x)$ 定义为：

$$f(x) = A \cdot x = \sum_{i=1}^{n} x_i a_i$$

上例中 f: $\{0,1\}^{10} \longrightarrow S$

$f(364_{10}) = f(0101101100) = 129+473+903+561+1165 = 3231$，

$f(609_{10})=2942$,　$f(686_{10})=3584$,　$f(32_{10})=903$,

$f(46_{10})=3326$,　$f(128_{10})=215$,　$f(261_{10})=2817$.

单向陷门函数及构造

2）单向陷门函数F：X ⟶ Y

超递增背包问题：背包向量$B=(b_1,b_2,\ldots,b_n)$中的元素满足下列性质：

$$b_j > \sum_{i=1}^{j-1} b_i \quad j = 2,\cdots,n$$

超递增背包算法有多项式时间解法，记为PT(B, ·)

解：对A从大到小检查每一元素，以确定是否在A'.

① 若$s \geq a_n$，则a_n在解中，令x_n=1；若$s<a_n$，则a_n不在解中，令x_n=0. 然后令

$$s = \begin{cases} s, & s < a_n \\ s - a_n, & s \geq a_n \end{cases}$$

② 对$a_{n-1}, a_{n-2}, \ldots, a_1$重复执行上述过程，直到$x_1$被赋值.

③ 检查$xA=S$是否成立，成立则解为$x=(x_1x_2\ldots x_n)$；否则该问题实例无解

单向陷门函数及构造

2）单向陷门函数F：X $\longrightarrow$ Y

G(n)： 输出对$(A, sk(B))$

$$f(A, x): s = f(A, x) = A \cdot x = \sum_{i=1}^{n} x_i a_i$$

$$f^{-1}(sk(B), s): \quad f^{-1}(sk, s) \to PT(B, s') \to x$$

单向陷门函数及构造

3）基于背包问题的密码算法

G(n)：输出密钥对(A, sk(B))

① 生成长度为n的超递增背包向量 $B=(b_1,b_2,\ldots,b_n)$;

② 取整数k和t，满足$k>\sum b_i$，且$\gcd(t,k)=1$.

$$a_i \equiv t\cdot b_i \bmod k,\ i=1,2,\ldots,n.$$

得一般背包向量$A=(a_1,a_2,\ldots,a_n)$，满足

$$A\equiv t\cdot B \bmod k.$$

③公开密钥为一般背包向量$A=(a_1,a_2,\ldots,a_n)$；t、t^{-1}和k为解密密钥（即陷门信息）.

单向陷门函数及构造

3）基于背包问题的密码算法

$f(A,x)$： $s=f(A,x)=A\cdot x=\sum_{i=1}^{n}x_i a_i$

①令二进制表示明文为m，将其分成长为n的分组$m=x^{(1)}\|x^{(2)}\|\ldots$；

②分别求每一分组的函数值$s_1=f(A,x^{(1)})$，$s_2=f(A,x^{(2)}),\cdots$；

③以函数值s_1，$s_2,\cdots$作为密文，发送给接收方.

单向陷门函数及构造

3）基于背包问题的密码算法

$f^{-1}(sk(B), s)$： $f^{-1}(sk,s) \rightarrow PT(B,s') \rightarrow x$

① 计算$s_i' \equiv t^{-1} s_i \bmod k$，$i$=1,2,⋯;

②解背包问题即得$x^{(1)}=(x^{(1)}{}_1 x^{(1)}{}_2 \ldots x^{(1)}{}_n)$，$x^{(2)}=(x^{(2)}{}_1 x^{(2)}{}_2 \ldots x^{(2)}{}_n)$，⋯;

③ 消息$m = x^{(1)} \| x^{(2)} \| \cdots$

因$s_i' \equiv$为$t^{-1} s_i \bmod k \equiv t^{-1} t B x^{(i)} \bmod k \equiv B x^{(i)} \bmod k$，而由$k > \sum a_i$，知$B x^{(i)} < k$，所以$s_i' \equiv B x^{(i)} \bmod k$ 是惟一的s_i作为超递增背包向量B的容积.

单向陷门函数及构造

4）基于背包问题的密码算法例

密钥生成 G(n)：

① 生成长度为10的超递增背包向量 B=(1, 3, 5, 11, 21, 44, 87, 175, 349, 701)；

② 取整数k=1590和 t=43,满足$k>\sum b_i$, gcd(43,1590)=1.

$$a_i \equiv t\cdot b_i \bmod k,\ i=1,2,\ldots,10.$$

得一般背包向量A=(43, 129, 215, 473, 903, 302, 561, 1165, 697, 1523).

③公开密钥为一般背包向量A；t、t^{-1}和k为解密密钥.

$$t^{-1} \equiv 37 \bmod 1590.$$

单向陷门函数及构造

4）基于背包问题的密码算法例

加密$f(A,x)$：①明文是SAUNA AND HEALTH，因为n=10，将明文分组SA，UN，A ` ´，AN，D ` ´，HE，AL，TH（其二进制序列为：$m=x^{(1)} \| x^{(2)} \| x^{(3)} \| x^{(4)} \| x^{(5)} \| x^{(6)} \| x^{(7)} \| x^{(8)}$=1001100001 ‖ 1010101110 ‖ 0000100000 ‖ 0000101110 ‖ 0010000000 ‖ 0100000101 ‖ 0000101100 ‖ 1010001000）；

②分别求每一分组的函数值$s_1=f(A,x^{(1)})=2942$，$s_2=f(A,x^{(2)})=3584$，$s_3=903$，$s_4=3326$，$s_5=215$，$s_6=2817$，$s_7=2629$，$s_8=819$；

③以函数值s_1，s_2，…，s_8作为密文，发送给接收方。

单向陷门函数及构造

4）基于背包问题的密码算法例

解密$f^{-1}(sk(B), s)$：

密文是(2942, 3584, 903, 3326, 215, 2817, 2629, 819)

① 计算$s_i' \equiv t^{-1} s_i \bmod k$，$i$=1,2,⋯,8;

$37\times2942\equiv734 \bmod 1590$,　$37\times3584\equiv638 \bmod 1590$,
$37\times903\equiv21 \bmod 1590$,　$37\times3326\equiv632 \bmod 1590$,
$37\times215\equiv5 \bmod 1590$,　$37\times2817\equiv879 \bmod 1590$,
$37\times2629\equiv283 \bmod 1590$,　$37\times819\equiv93 \bmod 1590$;
$(s_1', s_2', \cdots, s_8')$=(734, 638, 21, 632, 5, 879, 283, 93)

单向陷门函数及构造

4）基于背包问题的密码算法例

解密$f^{-1}(sk(B), s)$:

$(s_1', s_2', \cdots, s_8')$=(734, 638, 21, 632, 5, 879, 283, 93)

②解背包问题

取s_1'=734，由734>701，得$x^{(1)}_{10}$=1；

令s_1'= 734-701=33，由33<349，得$x^{(1)}_9$=0；

以此类推得$x^{(1)}$= 1001100001，对应的英文是SA；

类似地得其他明文分组$m = x^{(1)} \| x^{(2)} \| x^{(3)} \| x^{(4)} \| x^{(5)} \| x^{(6)} \| x^{(7)} \| x^{(8)}$ =1001100001 ‖ 1010101110 ‖ 0000100000 ‖ 0000101110 ‖ 0010000000 ‖ 0100000101 ‖ 0000101100 ‖ 1010001000);

③消息m：SAUNA AND HEALTH

单向陷门函数及构造

背包密码体制是继Diffie和Hellman 1976年提出公钥密码体制设想后的第一个公钥密码算法.

上述方案由Merkle和Hellman 于1978年提出. 两年后该体制即被破译, 破译的基本思想是找出任意模数k′和乘数t’, 使得用k’和t’去乘公开的背包向量B时, 能够产生超递增的背包向量即可.

本节要点小结

- 对称密码体制面临的问题
- 公钥密码体制的发展
- 单向陷门函数及构造

THE END！