东南大学网络空间安全学院
密码学与安全协议

第五讲 公钥密码算法

黄　　杰
信息安全研究中心

目录

- 公钥密码算法
- ElGamal算法
- RSA算法
- RSA算法的攻击方法

知识点

1、公钥密码算法的特点和原理

2、ElGamal和RSA等典型公钥算法

3、RSA的典型攻击方法

公钥密码算法

公钥密码算法

- 公钥密码和对称密码的区别
  - ✓对称密码是基于置换和扩散；
  - ✓非对称密码是基于数学难题；
  - ✓对称密码只使用一个密钥；
  - ✓公钥算法使用两个独立的密钥。

对称密码算法的两个难题

- 密钥分配问题。
- 数字签名问题。
- 问题：如何确保数字签名是出自某特定的人，并且各方对此均无异议？

- 如何理解以下问题
  - ✓公钥密码比对称密码安全吗？
  - ✓公钥密码可以取代对称密码吗？
  - ✓公钥密码实现密钥分配比对称密码容易吗？

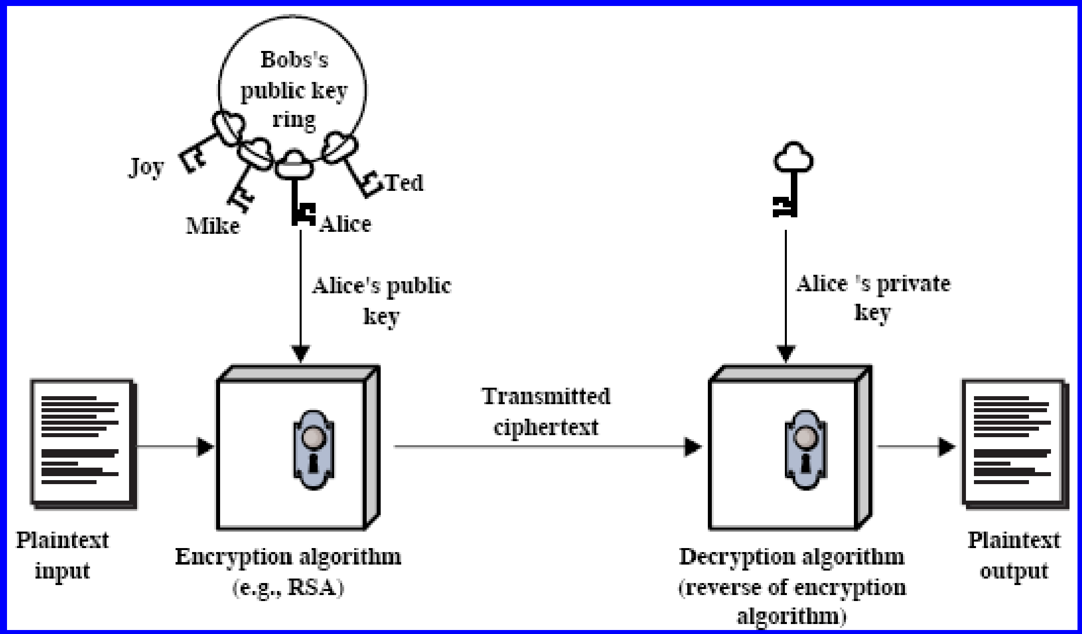
公钥密码体制的组成部分

- 特点
  - 仅根据密码算法和加密密钥不能确定解密密钥
  - 两个密钥中一个用于加密，另一个解密
- 六个组成部分
  - 明文
  - 加密算法
  - 公钥
  - 私钥
  - 密文
  - 解密算法

用公钥密码实现保密

- 用户拥有自己的密钥对(K_U,K_R)
- 公钥K_U公开,私钥K_R保密
- A-->B: $Y=E_{K_{Ub}}(X)$
- B: $D_{K_{Rb}}(Y)= D_{K_{Rb}}(E_{K_{Ub}}(X))=X$

Bob

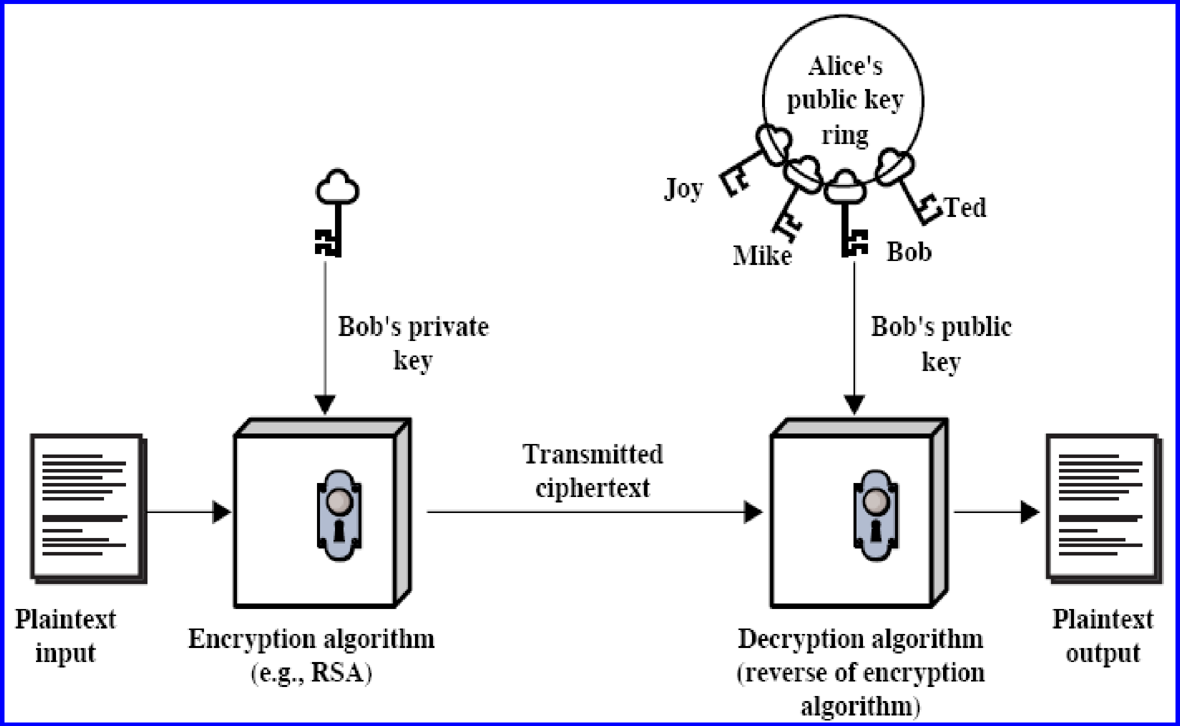


Alice

用公钥密码实现认证

- 认证:
  - A→ALL: $Y=D_{KRa}(X)$
  - ALL: $E_{KUa}(Y)=E_{KUa}(D_{KRa}(X))=X$

Bob

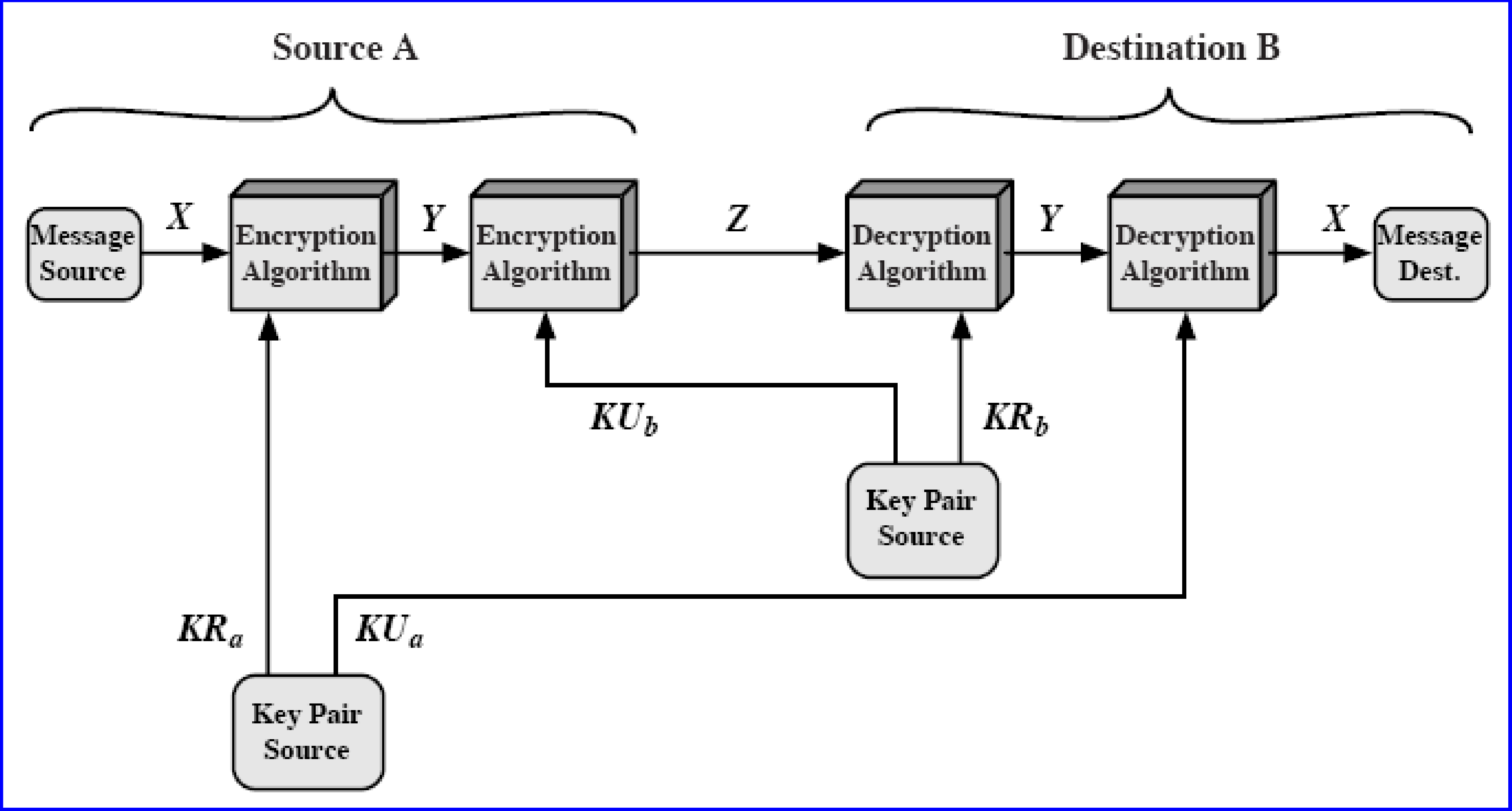


Alice

关于认证的讨论

- 认证算法
  - 加密整条消息
  - 加密部分消息(消息摘要)
- 认证不能保证消息的机密性。

- 需要：
  - 既能提供认证功能，又能保持机密性的方法


Source A
Destination B
Message Source
X
Encryption Algorithm
Y
Encryption Algorithm
Z
Decryption Algorithm
Y
Decryption Algorithm
X
Message Dest.
KUb
KRb
Key Pair Source
KRa
KUa
Key Pair Source

对称密码和公钥密码

对称密码	公钥密码
一般要求 1、加解密使用相同的密钥 2、收发双方必须共享密钥 **安全性要求** 1、密钥必须秘密保存 2、无密钥，解密消息不可行 3、知道算法和密文不影响密钥的安全性	**一般要求** 1、算法相同，但不同加解密密钥 2、发送方和接受方拥有不同密钥 **安全性要求** 1、有一个密钥必须保密 2、无密钥，解密信息不可行 3、知道算法、其中的一个密钥和若干密文，不影响另一个密钥的安全性

公钥密码体制的应用

- 加密/解密
- 数字签名
- 密钥交换

算 法	加密/解密	数字签名	密钥交换
RSA算法	是	是	是
椭圆曲线	是	是	是
ElGamal算法	是	是	是
Diffie-Hellman	否	否	是
DSS	否	是	否

对公钥密码的要求

- 公钥和私钥的产生在计算上是容易的；
- 已知公钥和明文，产生密文在计算上是容易的；
- 已知私钥和密文，恢复出明文在计算上是容易的；
- 已知公钥，确定私钥在计算上是不可行的；
- 已知公钥和密文，恢复明文在计算上是不可行的；
- 加密和解密函数的顺序可以交换。

公钥密码分析

- 穷举攻击——使用长密钥
  - 长密钥和短密钥矛盾
  - 仅用于数字签名和密钥管理中
- 从给定的公钥计算私钥
- 穷举消息攻击
  - 无论公钥体制的密钥有多长，这种攻击都可以转化为对56位密钥的穷举攻击。
  - 抗攻击的方法：在发送的消息后附加一个随机数。

ElGamal算法

1985年发表，既可用于数字签名又用于加密。其安全性依赖于离散对数难题。

离散对数问题：给定素数p及其p的一个本原根b和一个元素c，使得 $c = b^x \bmod p$。求解x，困难。

1.描述

选取素数$p>10^{150}$，一个模p的本原根g以及随机整数$x(1<x<p)$，计算$y = g^x \bmod p$，则公钥为(y,g,p)，私钥为x

2.ElGamal签名消息m

选择随机数k，且k与p-1互素，计算签名：

$$a=g^k \bmod p$$

$$b= k^{-1}（m - xa）\bmod (p-1)$$

签名（a, b）

注：k必须保密，每次签名k不同

验证签名：$y^a a^b \bmod p = g^m \bmod p$

3.ElGamal加密消息m

选择随机数k，且k与p-1互素，得到密文对(a,b)为：

$$a=g^k \bmod p, \quad b=m\cdot y^k \bmod p$$

解密消息：$b\cdot a^{-x} \bmod p=m\cdot y^k\cdot(g^k)^{-x} \bmod p=m$

3.举例（加解密）

已知：选Alice的私钥x=4，公钥p=11，g=2，$y=g^x \bmod p=5$

要求：Bob要将消息m=3加密传送给Alice。

(1)Bob选择随机数k=3，计算得到的密文：

$a=g^k \bmod p=2^3 \bmod 11=8$

$b=m \cdot y^k \bmod p=3 \cdot 5^4 \bmod 11=1$

(2)Alice对收到的密文(5,5)解密：

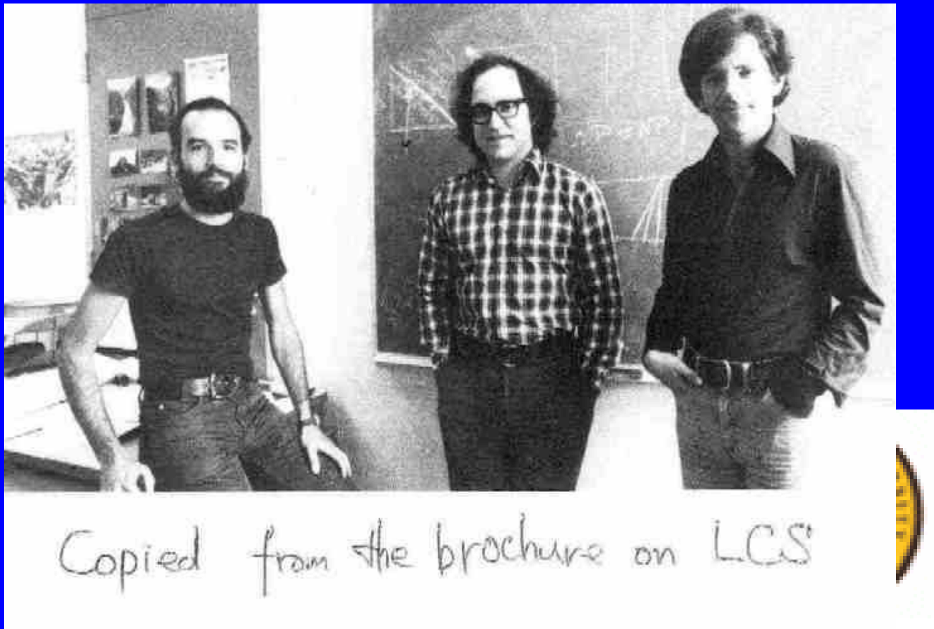
$b \cdot a^{-x} \bmod p=1 \cdot 8^{-4} \bmod 11=3$

RSA算法

RSA算法

- 1977年Rivest, Shamir和Adleman共同提出，是最早提出的满足要求的公钥算法之一，是被广泛接受且被实现的通用公钥加密方法。
- RSA是一种分组密码。明文和密文均为0到n-1的整数，通常n的大小为1024位二进制数或309位十进制数。

S、R、A



RSA算法描述

- 对于明文分组M，密文分组C，
- 加密过程：$C=M^e \bmod n$
- 解密过程：

  $M=C^d \bmod n =(M^e)^d \bmod n=M^{ed} \bmod n$
- 公钥用于加密$K_U=\{e,n\}$
- 私钥用于解密$K_R=\{d,n\}$
- 如何满足条件：$M=M^{ed} \bmod n$？

RSA算法中的元素

- 两个素数，p,q (保密)
- n=pq, （公开）
- e, gcd(Φ(n),e)=1,（公开）
- d, $d \equiv e^{-1} \bmod \phi(n)$ (保密)

Key Generation

Select p, q	p and q both prime, $p \quad q$
Calculate $n = p \times q$	
Calculate $\phi(n) = (p - 1)(q - 1)$	
Select integer e	$\gcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
Calculate d	$d \equiv e^{-1} \bmod \phi(n)$
Public key	$KU = \{e, n\}$
Private key	$KR = \{d, n\}$

Encryption

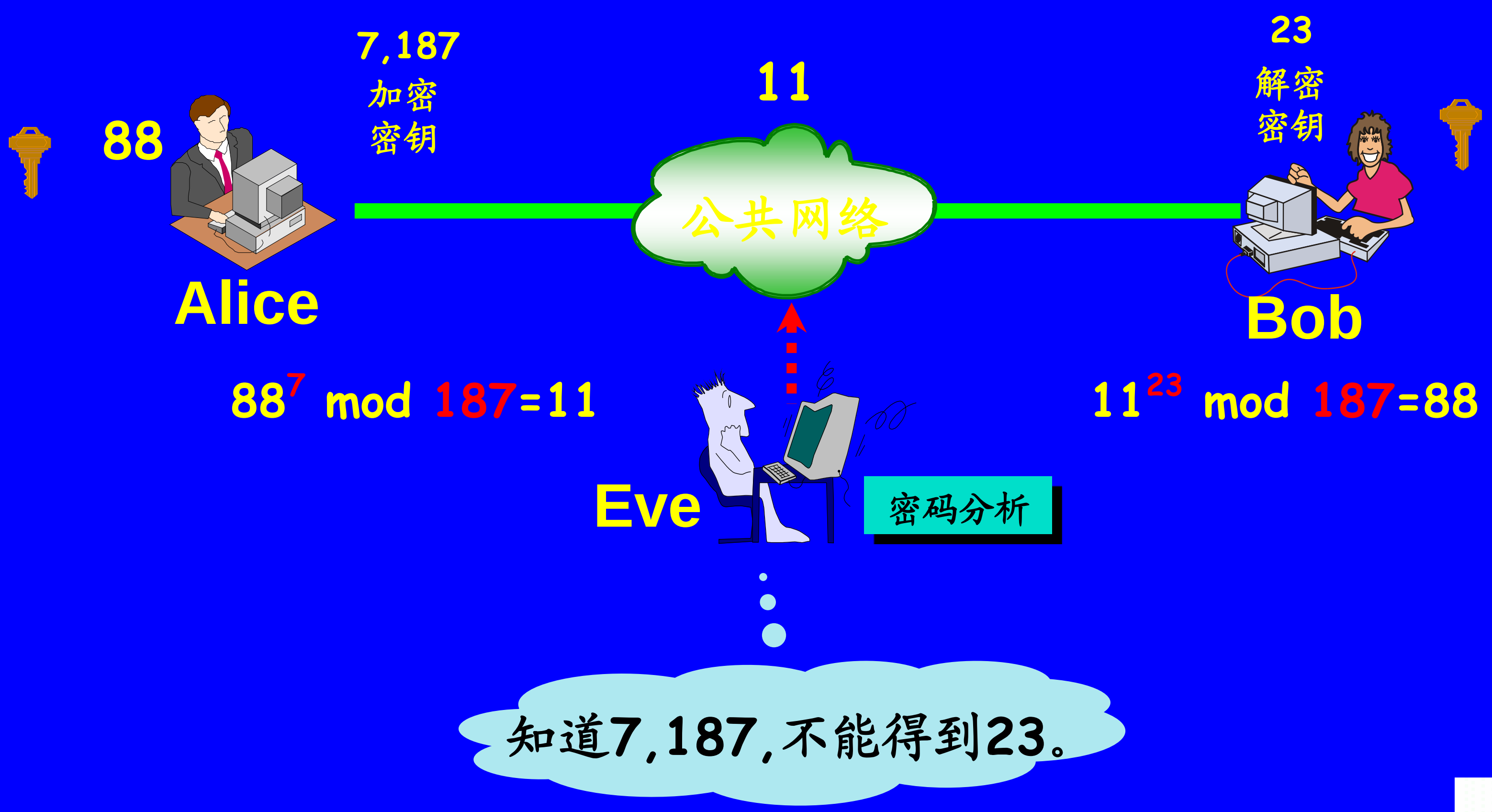
Plaintext:	$M < n$
Ciphertext:	$C = M^e \pmod{n}$

Decryption

Ciphertext:	C
Plaintext:	$M = C^d \pmod{n}$

RSA加解密举例

Bob的公钥为(7,187)，私钥为(23,187);
Alice要将保险柜密码88发送Bob。



可解析分析

(1)欧拉定理：如果gcd(x,n)=1，则有：

$x^{\varphi(n)}=1 \bmod n$

(2)明文x与模数n要互素，不互素的概率为：

$$1-\frac{(p-1)(q-1)}{pq}=\frac{1}{p}+\frac{1}{q}-\frac{1}{pq}$$

(3)e,d必须与φ(n)互素；

(4)具有形式为n=pq的整数称为Blum整数。

如何计算$a^b \bmod n$

- 先计算幂，再取模——中间结果非常大
- 幂运算的有效性问题

$$a^{16} = a \times a \times a \times a \times a \times a \times a \times a \times a \times a \times a \times a \times a \times a \times a \times a$$

- 解决方法：
- 利用模算术的性质

(axb) mod n =[(a mod n) x (b mod n)]mod n

- 重复计算中间结果的平方

a^2 a^4 a^8 a^{16},只需四次乘法

如何计算a^b mod n

更一般性的问题：a^m

m的二进制表示为$b_k b_{k-1} \dots b_0$, 则 $m=\sum_{i\neq 0} 2^i$

$$a^m = a^{(\sum_{b_i \neq 0} 2^i)} = \prod_{b_i \neq 0} a^{(2^i)}$$

$$a^m \bmod n = (\prod_{b_i \neq 0} a^{(2^i)}) \bmod n = (\prod_{b_i \neq 0} [a^{(2^i)} \bmod n]) \bmod n$$

计算a^b mod n的算法：

```
c ← 0; d ← 1
for i ← k downto 0
      do  c ← 2 × c
          d ← (d × d) mod n
          if   bi = 1
               then   c ← c + 1
                      d ← (d × a) mod n
return d
```

密钥产生

- 如何找到足够大的素数p和q？
- 选择e或d计算另外一个

素数选取

- 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。即p和q必须是大素数。
- 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。

因子分解的计算量

整数n的十进制位数	因子分解的运算次数	所需计算时间（每微秒一次）
50	1.4×10^{10}	3.9小时
75	9.0×10^{12}	104天
100	2.3×10^{15}	74年
200	1.2×10^{23}	3.8×10^{9}年
300	1.5×10^{29}	4.0×10^{15}年
500	1.3×10^{39}	4.2×10^{25}年

因子分解问题的进展情况

分解数	尺寸bits	分解日期	分解算法
RSA-100	330	1991.4	二次筛法
RSA-110	364	1992.4	二次筛法
RSA-120	397	1993.6	二次筛法
RSA-129	425	1994.4	二次筛法
RSA-130	430	1996.4	数域筛法
RSA-140	463	1999.2	数域筛法
RSA-155	512	1999.8	数域筛法

RSA的实现要求

- 若使RSA安全，p与q必为足够大的素数，使分析者没有办法在有效的时间内将n分解出来。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。

RSA的攻击方法

- 强力攻击（穷举法）：尝试所有可能的私有密钥
- 计时攻击
- 对RSA的选择密文攻击
- 对RSA的公共模数攻击
- 对RSA的低加密指数攻击
- 对RSA的低解密指数攻击
- 对RSA的加密和签名的攻击

对RSA选择密文攻击

- 目标：攻击者Eve能够得到$m=c^d \bmod n$。Eve窃听可以获取$c=m^e \bmod n$，m是明文，e是公钥。
- 方法：Eve随机选择r（$r<n$，且与n互素）

  计算：$x=r^e \bmod n$;

  生成待签名的文件：$y = xc \bmod n$

  Eve将y发送给Alice签名：$u=y^d \bmod n$

  Eve得到签名u后计算:

$$
\begin{aligned}
r^{-1} u \bmod n &= r^{-1} y^d \bmod n \\
&= r^{-1} (xc)^d \bmod n \\
&= r^{-1} r\, m^{ed} \bmod n \\
&= m
\end{aligned}
$$

注：不能对来历不明的数据签名。

对RSA的公共模数攻击

- 目标：如果两对密钥有相同的n，即：（d_1,e_1,n）和（d_2,e_2,n），则 容易恢复出明文m。其中，e_1和e_2互素.
- 方法：对相同的明文m分别计算密文，得:

$$c_1=m^{e_1} \bmod n \text{；} \quad c_2=m^{e_2} \bmod n$$

因此，可以找出r和s，满足:

$$re_1 + se_2= 1 \qquad \text{（扩展Euclid算法）}$$

然后，首先计算c_1^{-1}，则:

$$(c_1^{-1})^{-r} \times c_2^{s} \bmod n = m \bmod n$$

注：两组密钥不能有相同的模数。

问题

1、公钥密码算法能解决对称密码算法不能解决的哪些问题？

2、公钥密码算法比对称密码算法更安全？

3、为什么不能对来历不明的数据签名？

东南大学网络空间安全学院
密码学与安全协议

谢谢！