公钥密码

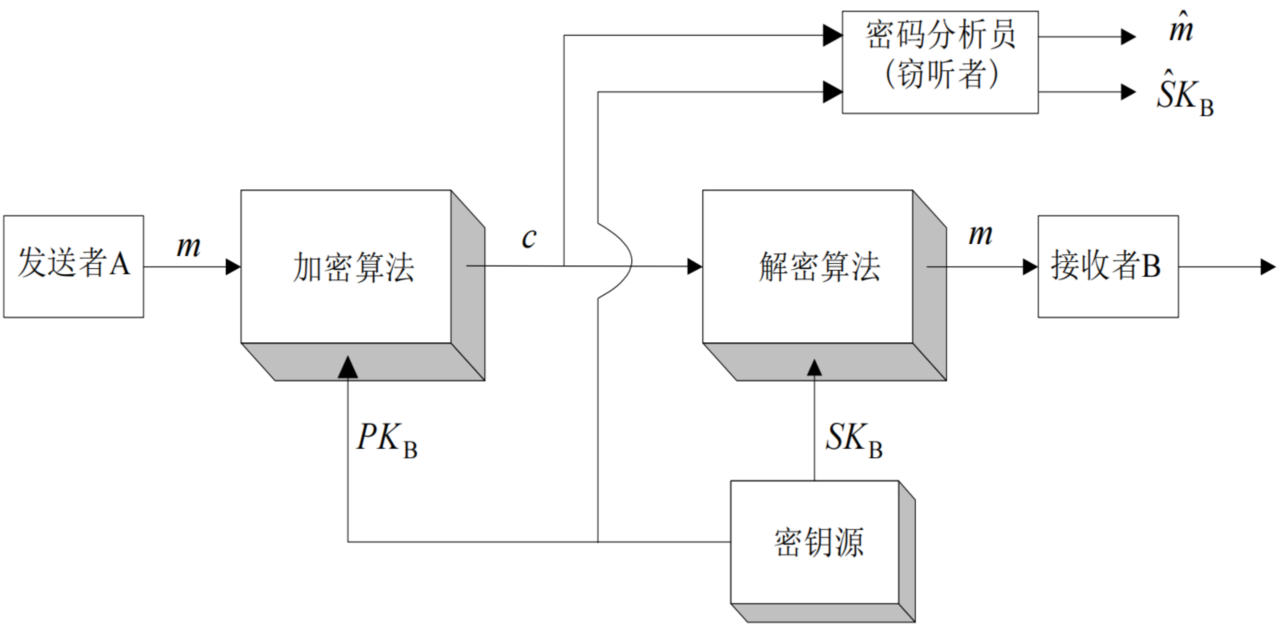
1 概述

公钥密码系统的应用：

- 加解密
- 密钥分配
- 数字签名

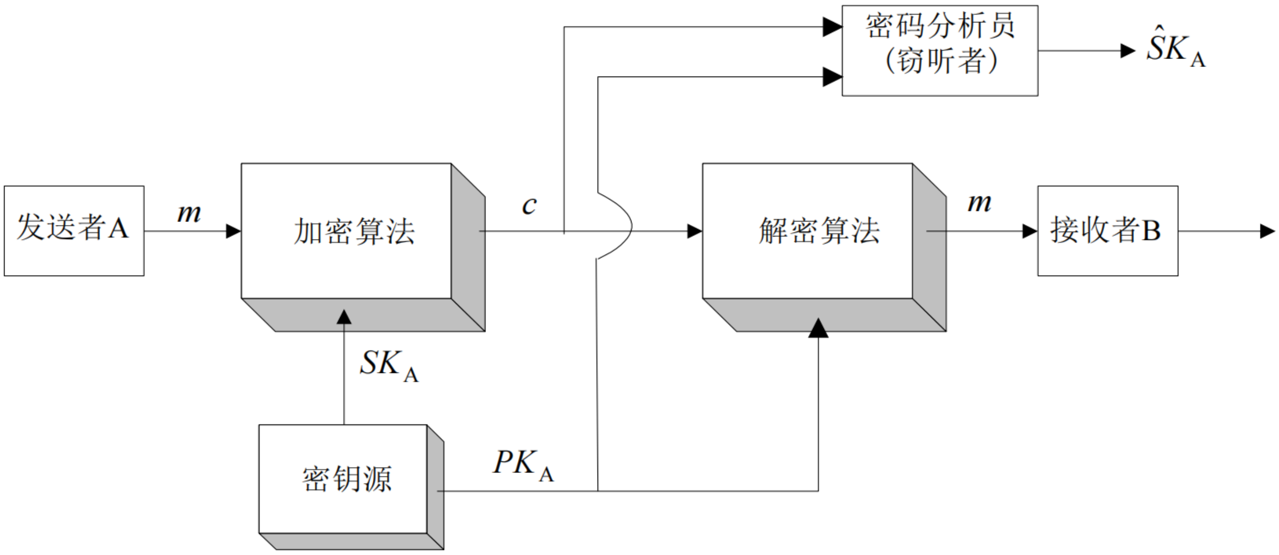
公钥密码体制的特点：
两个密钥将加解密的能力分开。一个是**公开**的，即为**公钥**，用于**加密**；另一个是**用户专用**的，即为**私钥**，用于**解密。**
已知密码算法和公钥（加密密钥），求解解密密钥（私钥）在计算上是不可行的。

公钥体制加密过程框图

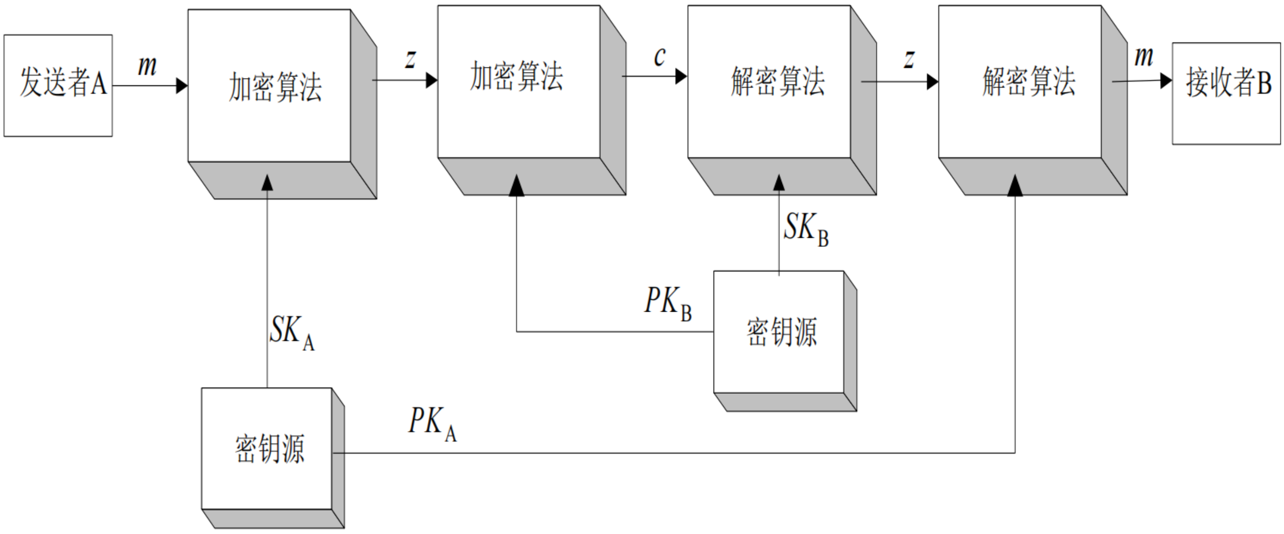


注意：发送者使用**接收者的公钥**对消息进行加密；接收者使用**接收者的私钥**对消息进行解密。

公钥体制消息认证框图



注意：发送者使用**发送者的私钥**对消息进行加密；接收者使用**发送者的公钥**进行认证。
上图中，消息不会被修改，但可以被窃听（公钥公开）。为了保密性，再用**接收者的私钥**对签名进行加密。



陷门单向函数
函数本身易于计算，但**求其逆不可行**，除非再已知某些附加信息。即f_k满足：

- 当已知k和X时，$Y=f_k(X)$易于计算
- 当已知k和Y时，$X=f_k^{-1}(Y)$易于计算
- 当已知Y但未知k时，$X=f_k^{-1}(Y)$在计算上不可行
  研究公钥密码算法就是找出合适的陷门单向函数。

2 公钥密码算法

2.1 RSA

算法描述

1. **密钥产生**

- 选取两个大素数p，q
- 计算$n=p\times q, \phi (n) = (p-1)(q-1)$，其中，$ \phi (n)$为n的欧拉函数值。
- 选一个整数e，满足$1<e<\phi(n)$，且$gcd(e,\phi(n)) = 1$
- 计算d，满足$d\cdot e \equiv 1 \bmod \phi(n)$
- 以(e,n)为公钥，(d,n)为私钥

2. **加密与解密**
   将明文比特串分组，使得每个组对应的十进制数小于n（*对应的十进制表示与n长度相同*），即分组长度小于$log_2 n$，然后对每个明文分组m做加密运算：$$c \equiv m^e \bmod n$$ 对应的，解密运算为：$$m \equiv c^d \bmod n$$

重复平方乘
求a^m时，可按如下进行：
将m表示为二进制形式$b_kb_{k-1}...b_0$，即$m=b_k 2^k + b_{k-1} 2^{k-1} + ... + b_1 2+b_0$，因此$$a^m=(...(((a^{b_k})^2a^{b_{k-1}})^2a^{b_{k-2}})^2...a^{b_1})^2a^{b_0}$$

基于中国剩余定理改进的RSA实现
解密方计算：$$d_p \equiv d \bmod (p-1)$$ $$d_q \equiv d\bmod (q-1)$$ $$m_p \equiv c^{d_p} \bmod p$$ $$m_q \equiv c^{d_q} \bmod q$$ 解方程，并根据中国剩余定理得到m $$ \begin{cases} m_p \equiv c^{d_p} \bmod p \equiv c^d \bmod p \equiv m \bmod p\ m_q \equiv c^{d_q} \bmod q\equiv c^d \bmod q \equiv m

\bmod q \end{cases} $$ 具体推导过程如下：

$m \equiv c^d \bmod n$ （正常计算时）

$$CRT \xrightarrow{n=p\cdot q} \begin{cases} m \equiv c^d \bmod p \\ m \equiv c^d \bmod q \end{cases}$$

为了使 d 减小，令 $d_p \equiv d \bmod \varphi(p)$.

故 $\exists k$, s.t. $d_p = k\varphi(p) + d. \Rightarrow d = d_p - k\varphi(d)$.

$\therefore c^d \bmod p \equiv c^{d_p - k\varphi(p)} \bmod p \equiv c^{d_p} \cdot (c^{\varphi(p)})^{-k} \bmod p$

由Euler定理，$\gcd(c,p)=1$ 则 $c^{\varphi(p)} \bmod p \equiv 1$.

故 $c^d \bmod p \equiv c^{d_p} \bmod p$.

同理，$c^d \bmod q \equiv c^{d_q} \bmod q$. 其中 $d_q = d \bmod \varphi(q)$

综上，方程组转化为 $\begin{cases} m \equiv c^{d_p} \bmod p \\ m \equiv c^{d_q} \bmod p \end{cases}$

RSA的安全性
RSA的安全主要基于大素数分解的困难性。同时，对p和q有以下要求：

- $\lvert p-q \rvert$ 要大
- $p-1$ 和 $q-1$要有大素因子（防止重复加密攻击）

对RSA的攻击
两种攻击方法：

- 共模攻击
  给不同用户相同的模数n，攻击者截获c_1, c_2后，通过求解$re_1 + se_2 = 1$得到 r 和 s 后，$(c_1^{-1})^{-r}c_2^s\equiv m \bmod n$即可获得明文。
- 低指数攻击
  多个用户的加密指数相同且都很小，通过解方程后开方。

2.2 Rabin

Rabin密码体制是对RSA的修正，特点如下：

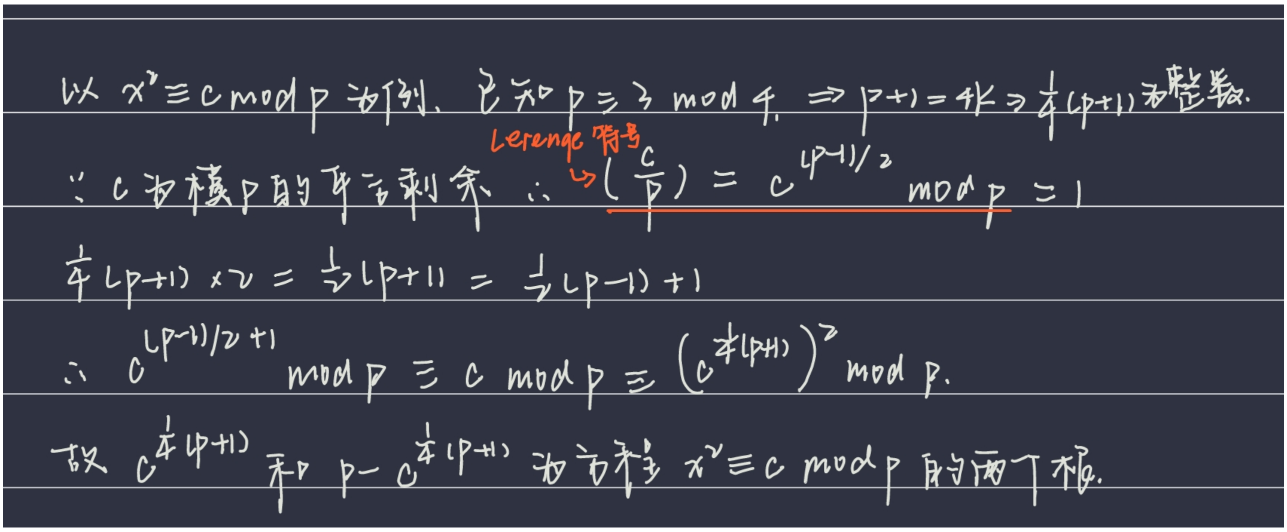
- 不以一一对应的单项陷门函数为基础，**同一密文可能对应不同明文。**
- 破译该密码体制等价于对大整数的分解。

公开钥e的选取

- RSA：$1<e<\phi (n)$，且$gcd(e,\phi(n))=1$
- Rabin：$e=2$

算法描述

1. 密钥的产生
   两个大素数p、q，满足$p\equiv q\equiv 3\bmod4$；计算$n=p\times q$，以**n为公钥**，**p、q为私钥。**
2. 加密与解密
   加密： $$c\equiv m^2 \bmod n$$ 解密：求c模n的平方根，即解$x^2\equiv c \bmod n$，等价于方程组 $$ \begin{cases} x^2 \equiv c \bmod p \ x^2 \equiv c \bmod q \end{cases} $$ 求解过程如下：



（图中写错了，应为Legendre符号）
结论：若p模4余3，则方程的解为$c^{(p+1)/4}与p-c^{(p+1)/4}$

2.3 椭圆曲线密码体制

椭圆曲线的曲线方程：$$y^2+axy+by=x^3+cx^2+dx+e$$ 定义中包括一个无穷远点O。
密码中普遍采用**有限域**上的椭圆曲线（即所有系数均为某一有限域$GF(p)$中的元素）。其中，最为常见的是由方程$$y^2=x^3+ax+b (a,b \in GF(p), 4a^3+27b^2 \neq 0)$$定义的曲线。下文未特殊指出时，椭圆曲线方程均为上述等式。

椭圆曲线上的点集
$E_p(a,b)={(x,y)|0\leq x <p, 0\leq y <p, 且x，y均为整数}\cup{O} $
计算方法

1. 对每一个x（$0\leq x <p$），计算$x^3+ax+b \bmod p$
2. 判断1中计算结果是否为模p的平方剩余，是则保留并开平方，不是则舍弃

$E_p(a,b)$上的加法
设$P,Q\in E_p(a,b)$，则：

1. $P+O=P$

2. $P=(x,y)$的加法逆元$-P=(x,-y)$
3. $P=(x_1,y_1),Q=(x_2,y_2), P\neq Q$，则$P+Q=(x_3,y_3)$由以下规则确定：$$x_3=\lambda^2-x_1-x_2$$ $$y_3=\lambda(x_1-x_3)-y_1$$ $$\lambda=\begin{cases} \frac{y_2-y_1}{x_2-x_1}, P\neq Q \ \frac{3x_1^2+a}{2y}, P=Q \end{cases}$$ 倍数运算看作重复加法。

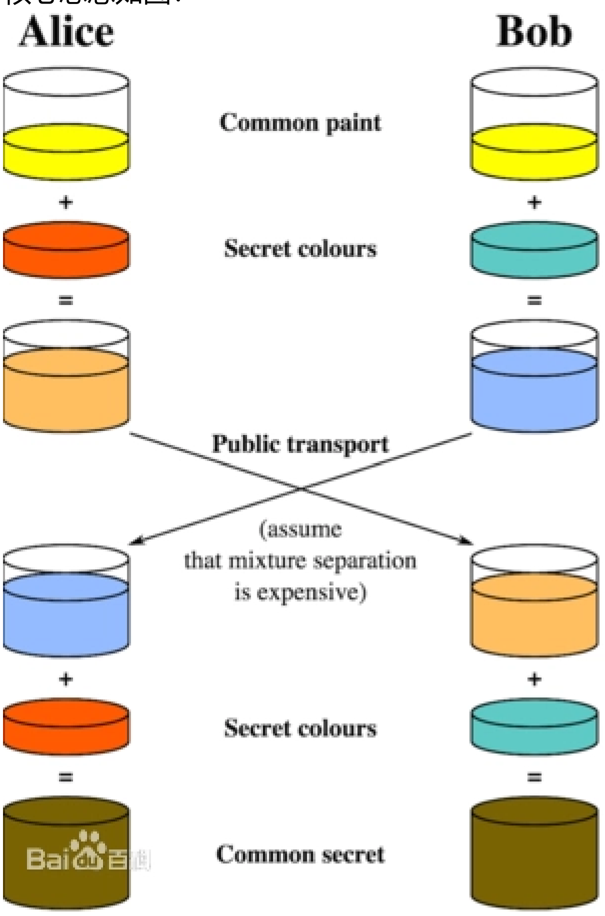
椭圆曲线上的点数
第一象限中的整数点加无穷远点O共有$1+p+\epsilon$个，其中，$\lvert \epsilon \rvert\leq2\sqrt{p}$。

将明文消息嵌入到椭圆曲线
选取足够大的整数k（以k取30为例），对明文消息m，如下计算一系列x：$$x={mk+j, j=0,1,2\dots}={30m,30m+1,\dots}$$直到x^3+ax+b为模p的平方剩余，即得到椭圆曲线上的点。

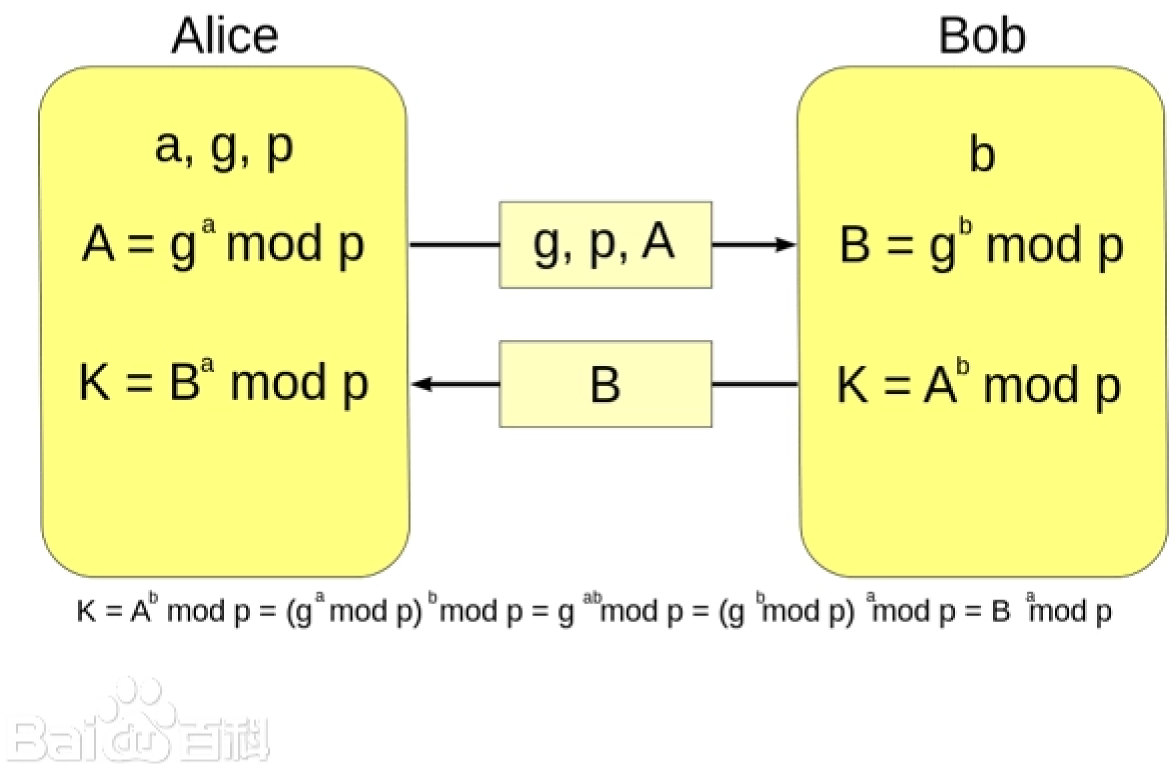
3 Diffie-Hellman密钥交换

3.1 Diffie-Hellman

作用： 密钥交换
核心思想如图：



具体至基于离散对数困难的Diffie-Hellman密钥交换的过程如下：

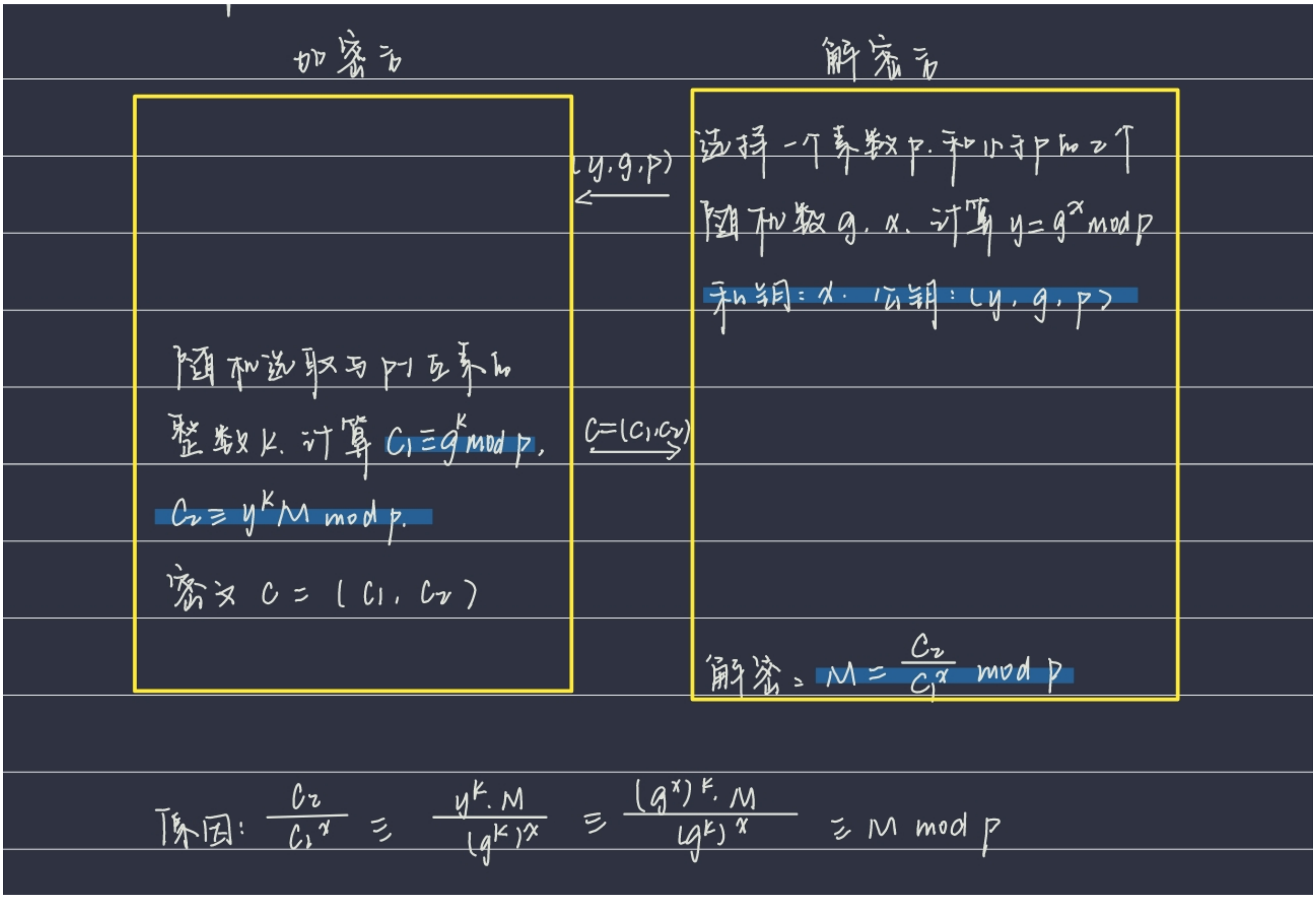


其中，g和p为公开参数，A和B分别为Alice和Bob的公钥，a和b为对应的私钥，约定共享的密钥为K。
若用椭圆曲线来写，则椭圆曲线$ E_p(a,b), G$为公开参数，其中$G$为一个生成元。生成公钥的过程为$A=nG$，共享密钥为$K=aB=Ab$。

3.2 利用Diffie-Hellman思想的密码体制

ElGamal密码体制
作用： 加密

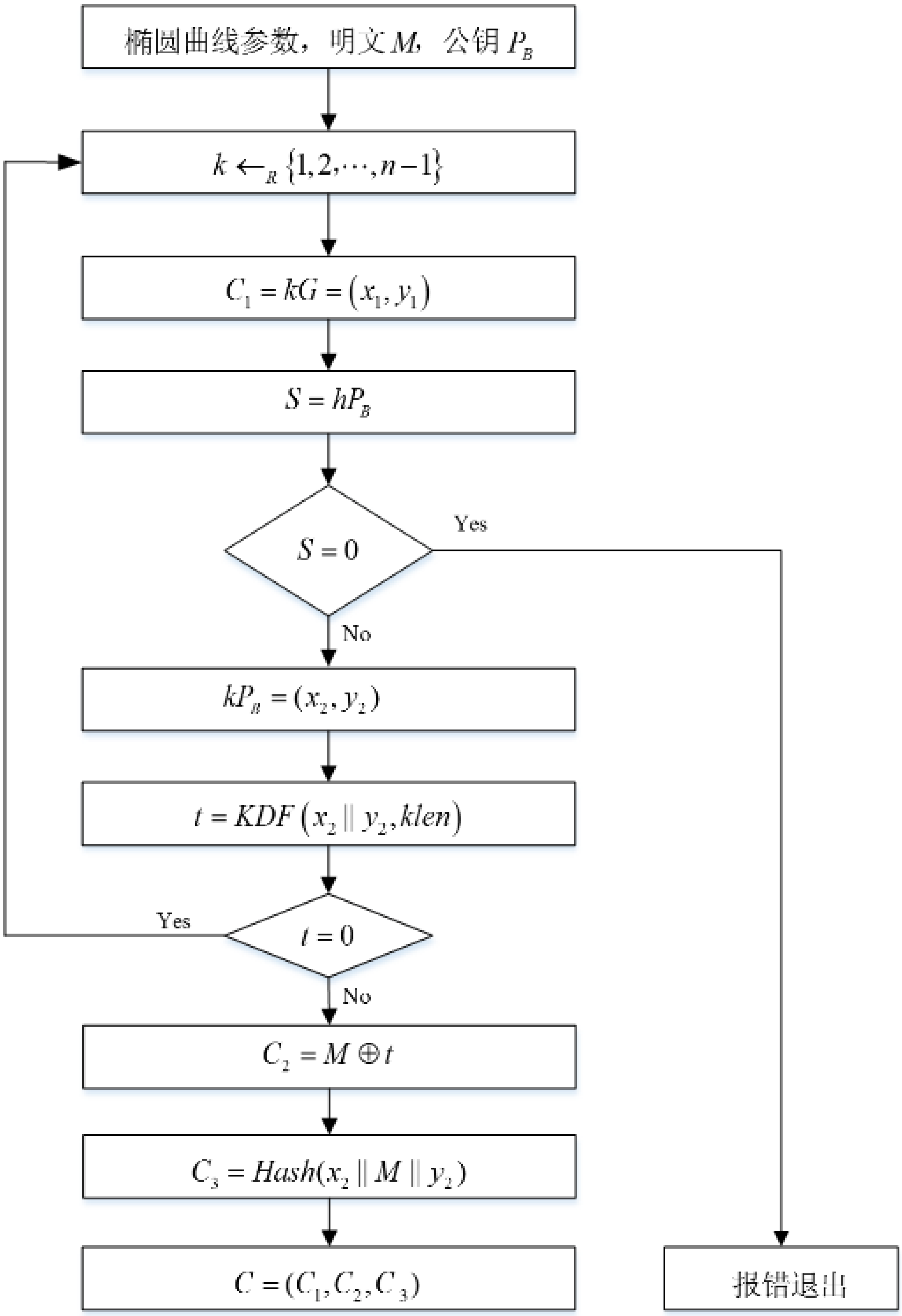
过程如下图所示：



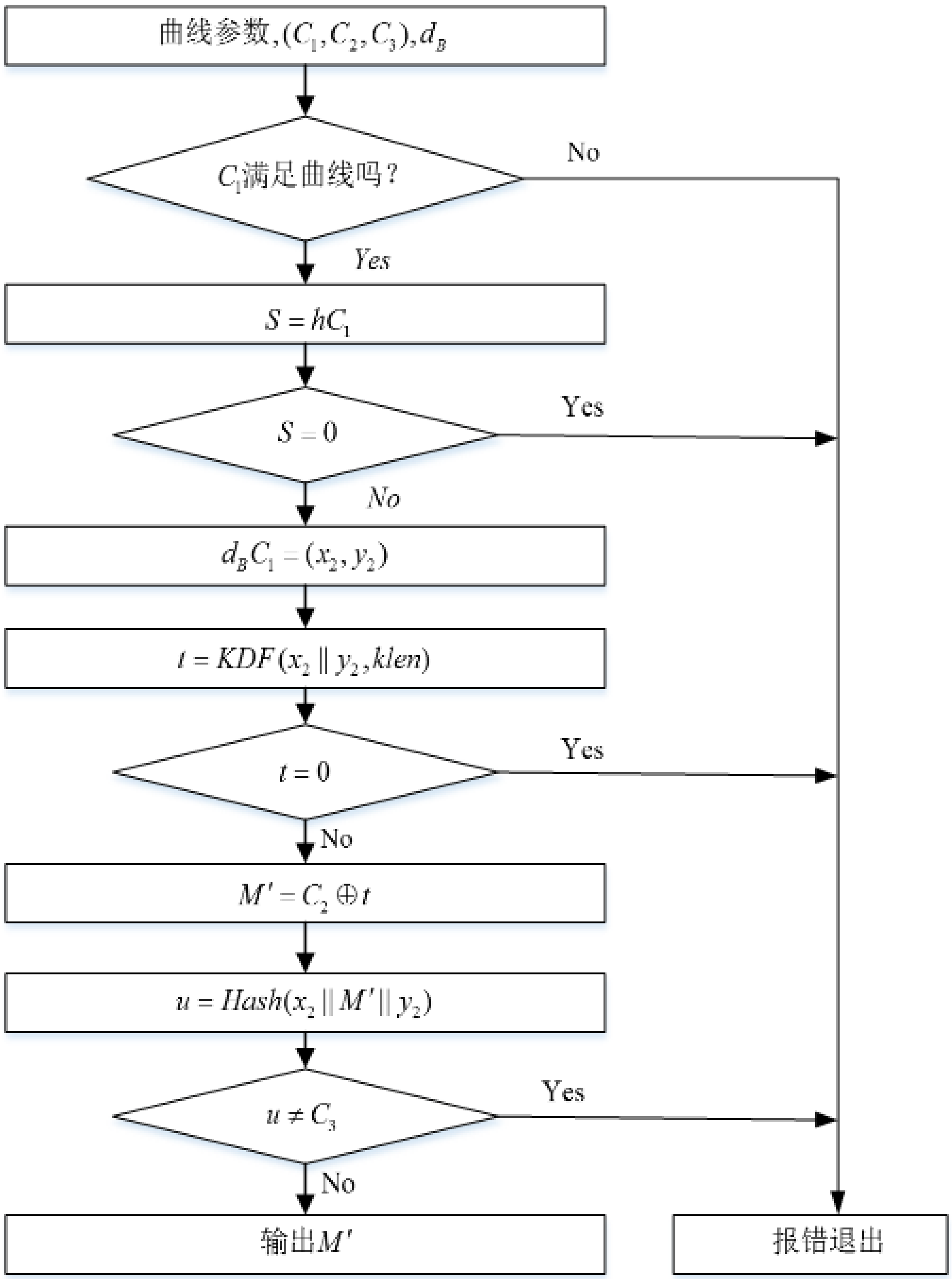
SM2加解密流程图

$P_B=d_BG$，其中，P_B为公钥，d_B为私钥。

加密


椭圆曲线参数，明文 M，公钥 P_B
$k \leftarrow_R \{1,2,\cdots,n-1\}$
$C_1 = kG = (x_1, y_1)$
$S = hP_B$
$S = 0$
Yes
No
$kP_B = (x_2, y_2)$
$t = KDF(x_2 \| y_2, klen)$
$t = 0$
Yes
No
$C_2 = M \oplus t$
$C_3 = Hash(x_2 \| M \| y_2)$
$C = (C_1, C_2, C_3)$
报错退出

解密



解密的正确性：$$d_BC_1=d_B(kG)=k(d_BG)=kP_B=(x_2,y_2)$$