密码学引论补充提纲

> 来自🦖密码学补充提纲计划
> latest update：2024/12/11
> 作者：🦖 chy xqh Samdy
> 本题纲是针对《密码学引论（第四版）》的补充提纲，主要整理书上缺失或不完整而PPT讲解的重点内容。

乘法逆元求法

```
整个求解的过程就是使用辗转相除法求两个数的公约数，逆推出ax+py=gcd(a,p)=1即可。
例：a = 3,p = 17
2 = 17 - 3 * 5
1 = 3 - 2 * 1

然后我们通过等式的代换就可以得到a与p的关系
1=3-(17-3*5)*1
1=3-17*1+3*5
1=3*6-17*1
得到结果为6。
```

第二讲 密码学的基本概念

密码体制的分类

从加密钥与解密钥是否相等划分：

(1) 传统密码/对称密码/单密钥密码： Ke = Kd 典型密码：DES，AES，SM4，ZUC，RC4

(2) 公开密钥密码/非对称密码/双密钥密码： Ke ≠ Kd 且由Ke 不能计算出 Kd 于是可将Ke公开，这样也不会危害Kd的安全 典型密码：RSA，ELGAMAL，ECC

从密钥的使用方式划分：

(1) 序列密码： 明文、密文、密钥以位（字符）为单位加解密 核心密码的主流 （回顾：核心、普通、商密、个人） 典型密码：RC4，祖冲之密码（ZUC）

(2) 分组密码： 明文、密文、密钥以块（分组）为单位加解密 商用密码的主流 典型密码：DES，AES，SM4

从密码算法是否变化划分： (1)固定算法密码 密码在工作过程中算法固定不变，密钥可变 迄今为止的绝大多数密码都是固定算法密码 典型密码：AES，DES，SM4，RC4，ZUC，RSA，ELGAMAL，ECC
(2)演化密码 借鉴生物进化，将密码学与演化计算结合 密码算法不断演化变化，而且越变越好 实现密码设计与密码分析自动化的一种方法 密码系统智能化的一种成功实践

从是否基于数学划分 (1)基于数学的密码：前面所有的密码

(2)基于非数学的密码

①基于物理学的密码：量子密码（量子密钥分发QKD） 利用量子力学产生真随机数作密钥，利用量子通信的保密性传输密钥，利用模2加进行加密，而且按一次一密方式工作 在唯密文攻击下无条件安全的密码 安全基于量子的物理属性
②基于生物学的密码：DNA密码 已有各种密码方案 安全性基于生物学中的困难问题 由于不基于计算，所以无论计算机的计算能力多么强大，与DNA密码都是无关的 尚不成熟：缺少理论，技术实现复杂

第三讲 数据加密标准（DES）

1、可逆性证明

① 定义 变换T是把64位数据的左右两半交换位置：

$T(L,R) = (R,L)$ 计算：$T^2(L,R) = (L,R) = I$,其中 I为恒等变换。 因为$T^2 = I$,所以有$T = T^{-1}$

所以 T 变换是对合运算。

②记 DES第 i 轮中的主要运算为,即

$F_i(L_i-1, R_i^{-1}) = (L_i^{-1} \oplus f(R_i^{-1}, K_i), R_i^{-1})$

$F_i^2 = F_i(L_i^{-1} \oplus f(R_i^{-1}, K_i), R_i^{-1}) = (L_i^{-1} \oplus f(R_i^{-1}, K_i) \oplus f(R_i^{-1}, K_i), R_i^{-1}) = (L_i^{-1}, R_i^{-1}) = I$
所以，$F_i = F_i^{-1}$

所以 Fi 变换也是对合运算。

③ 结合①、②，便构成DES的轮运算

$H_i = F_iT$ 因为$(F_iT)(TF_i)=(F_i(TT)F_i)= F_iF_i = I$

所以 $(F_iT)^{-1} = (TF_i)$

$(F_iT) = (TF_i)^{-1}$

④加解密表示

(1) $DES(M) = (M)IP\ (F1T)\ (F2T) \ldots (F15\ T)\ (F16)\ IP^{-1} = C$ (2) $DES^{-1}(C) = (C)IP\ (F16\ T)\ (F15\ T) \ldots (F2\ T)\ (F1)\ IP^{-1} = M$ 把 (1) 式代入 (2) 式可证： $DES^{-1}(DES(M)) = M$
所以，DES是可逆的。

2、对合性证明

$DES = IP\ (F1T)\ (F2T) \ldots (F15\ T)\ (F16)\ IP^{-1}$

$DES^{-1} = IP\ (F16\ T)\ (F15\ T) \ldots (F2\ T)\ (F1)\ IP^{-1}$

DES和DES^{-1}除了子密钥的使用顺序相反之外是相同的。

不考虑子密钥的使用顺序：
$DES = IP\ (F)\ (TF)\ (TF) \ldots (TF)\ (TF)\ (TF)\ IP^{-1}$

$DES^{-1} = IP\ (F)\ (TF)\ (TF) \ldots (TF)\ (TF)\ (TF)\ IP^{-1}$

显然：$DES = DES^{-1}$ 所以DES的运算是对合运算。

DES的安全性

①攻击

穷举攻击。这是目前最有效的方法。
差分攻击。
线性攻击。

②安全弱点

密钥太短。这是最主要的弱点。
存在弱密钥（如输入为全0和全1，输出也是全0全1。因为操作全为移位、置换并不改变内容）。
存在互补对称性（DES互补性：DES加密中若将明文消息X和加密密钥K都逐比特位取反，则加密的密文也是原密文逐比特位取反）。这使得攻击复杂度下降一半。
设C=DES（M，K），则有C=DES（M，K）。

第四讲 高级数据加密标准（AES）

③**不是对合运算**:
加解密使用不同的算法。
(AES)-1≠AES
解密算法的结构与加密算法的结构相同
解密中的变换为加密算法变换的逆变换，且密钥扩展策略稍有不同。

⑤整体结构:SP结构，基本轮函数迭代，迭代轮数可变（≥10）

AES的GF(2^8)表示
加法：两个元素多项式的系数按位模2加
(x6+x4+x2+x+1)⊕(x7+x+1)= x7+x6+x4+x2

乘法：两个元素多项式相乘，模 m (x),当除法不断减x^n倍求余数就行
(x6+x4+x2+x+1)×(x7+x+1)=x7+x6+1 mod m (x)
转换成二进制运算也可以

S盒变换 ByteSub(State)
①S盒变换是AES的唯一的非线性变换，是AES安全的关键，起密码学的混淆作用
②AES使用16个相同的S盒，DES使用8个不相同的S盒。
③AES的S盒有8位输入8位输出，是一种非线性置换。DES的S盒有6位输入4位输出，是一种非线性压缩。

第五讲 中国商用密码SM4与分组密码的应用技术

重点内容与书上重复

第6讲 序列密码基础

重点内容与书上重复

第7讲 祖冲之密码

重点内容与书上重复

第8讲 Hash函数的概念

Hash函数的定义

*一般，h的长度小于M的长度，因此HASH函数是一种压缩变换。

②**实用性**：对于给定的数据M，计算h=Hash (M)是高效的。
③**安全性**（这部分与课本一样，课本有详细解读）

Hash函数的类型

ppt上对基于序列密码的Hash函数多了一条： 目前有许多强分组密码，都可设计Hash函数。基于序列密码设计Hash函数较少，**ZUC的MAC是一个例子**

第十讲 公钥密码基础

一、公钥密码的基本思想

传统密码的优缺点:

优点

- 理论和实践都很成熟
- 安全容易把握
- 加解密速度快

缺点

- 收发双方持有相同密钥，Ke=Kd，密钥分配困难，网络环境更突出。
- 不能方便地实现数字签名，商业等应用不方便。

公钥密码的理论模型

（一）**单向函数**

设函数 y=f(x)，如果满足以下两个条件，则称为单向函数:

- 如果对于给定的 x，要计算出 y=f(x)很容易；
- 而对于给定的 y，要计算出 x=f -1(y)很难

（二）**利用单向函数构造密码**

- 用正变换作加密，加密效率高；
- 用逆变换作解密，安全，敌手不可破译；
- 但是合法收信者也无法解密。

（三）**单向陷门函数**

设函数 y=f(x)，且 f 具有陷门，如果满足以下两个条件，则称为单向陷门函数:

- 如果对于给定的 x，要计算出 y=f(x) 很容易；

- 而对于给定的 y，如果不掌握陷门要计算出x=f -1(y)很难，而如果掌握陷门要计算出 x=f -1(y)就很容易。

（四）**利用单向陷门函数构造密码**

- 用正变换作加密，加密效率高；
- 用逆变换作解密，安全；
- 把陷门信息作为密钥，且只分配给合法用户。确保合法用户能够方便地解密，而非法用户不能破译。

（五）**单向函数的研究现状**

理论上：尚不能证明单向函数一定存在； **实际上**：密码学认为只要函数单向性足够应用就行了；

第十一讲 中国商用公钥密码SM2加密算法

一、椭圆曲线

1、素域上的椭圆曲线

椭圆曲线特性：曲线的所有点都没有两个或者两个以上的不同的切线。

2、 GF(2^m)上的椭圆曲线

除了GF(p)上的椭圆曲线，还有定义在GF(2^m)上的椭圆曲线。 基于这两种椭圆曲线都可以设计出安全的椭圆曲线密码。

注意：GF(2^m)上的椭圆曲线与GF(p)上的椭圆曲线的加法定义不同。 （见书）

三、椭圆曲线公钥密码

1、椭圆曲线密码的一般情况

我国商用密码采用了椭圆曲线密码，并颁布了椭圆曲线密码标准算法SM2。
椭圆曲线密码已成为除RSA密码之外呼声最高的公钥密码之一。 它密钥短，软件实现规模小、硬件实现节省电路。 **由于椭圆曲线离散对数问题尚没有发现亚指数算法，所以普遍认为，椭圆曲线密码比RSA和ElGamal密码更安全。**
ElGamal密码建立在有限域GF(p)的乘法群的离散对数问题的困难性之上。而椭圆曲线密码建立在椭圆曲线群的离散对数问题的困难性之上。**两者的主要区别是其离散对数问题所依赖的群不同。** 因此两者有许多相似之处。 基于GF(p)和GF(2^m)上的椭圆曲线，都可以构成安全的椭圆曲线密码。

3、GF(p)上的椭圆曲线密码和ElGamal密码的对比

(1)密钥生成

椭圆： 用户选择一个随机数d作为私钥，d∈{1,2,⋯,n-1}。 用户计算 Q=dG ，以Q点为自己的公开钥。

ElGamal： 用户随机地选择一个整数d作为自己保密的解密钥，2≤d ≤p-2 。 用户计算 y =α^d mod p，以y为自己的公开钥。

(2)加密

椭圆： 设明文数据为M，0≤ M ≤ n-1 。
加密过程：

- 选择一个随机数k，k∈{1,2,⋯,n-1}。
- 计算点X1(x1 , y1) = kG 。
- 计算点X2(x2 ， y2) =kQ（公钥），
- 如果分量x2=0，则转①。
- 计算密文C=Mx2 mod n 。
- 以(X1，C)为最终密文。

ElGamal： 设明文消息M（0≤M≤p-1）
加密过程：

- 随机地选取一个整数k，2≤k≤p-2。
- 计算: $C1 = \alpha^k \bmod p$； $U = y^k \bmod p$； $C2 = UM \bmod p$；
- 取C = (C1 ， C2)为最终密文

(3)解密：

椭圆

- 用私钥d求出点X2： dX1 = d(kG)= k(dG) = kQ = X2(x2 ， y2)
- 对C 解密：利用x2计算得到明文 $M = C\, x_2^{-1} \bmod n$

ElGamal

- 计算$V = C_1^{\ d} \bmod p = (\alpha^k)^d \bmod p = (\alpha^d)^k \bmod p = (y)^k \bmod p = U$
- 计算$M = C_2 V^{-1} \bmod p$

4、GF(p)上椭圆曲线密码的实现

难点：

- 安全椭圆曲线的产生；
- 倍点运算比较麻烦。

四、中国商用椭圆曲线公钥密码SM2

使用256位素域GF(p)上的椭圆曲线： $y^2 = x^3 + ax + b$

5、解密正确性

- 证明： $d_BC_1=d_B(kG) =k(d_BG) =kP_B = (x_2，y_2)$； 如果$(x_2，y_2)$是正确的，则$t = KDF(x_2 \| y_2, klen)$也将是正确的。 又因为加密时$C_2 = M \oplus t$，所以解密时 $M' = C_2 \oplus t$ 。
- 验证： 根据解密得到的x2、y2和M′重新计算C3，并于接收到的C3比较，若两者相等则说明密文和解密正确，否则说明密文或解密不正确。

6、比较

传统椭圆曲线密码

- 计算点X2（x2 ， y2）=kQ。
- 计算密文 C = Mx2 mod n 。
- 最终密文是< X1，C >

SM2

- 计算点kPB=(x2，y2)；
- 计算t =KDF(x2 || y2, klen)；
- 计算C2 = M ⊕ t；
- 最终密文是< C1， C2， C3 >

传统椭圆曲线密码

- 利用分量x2作密钥进行加密：C = m x2 mod n，加密运算是乘法比较复杂。
- 分量y2没有利用。
- （X1，C）为密文。

SM2

- 利用分量x2和y2经过密钥派生函数产生中间密钥t，再用t进行加密：C2 = M ⊕ t，加密运算是模2加，因此效率更高，密钥派生函数提高了安全性，却增加了时间消耗。
- C = C1 || C2 || C3为密文，密文数据扩张较前者严重。
- SM2 采取了许多检错措施，从而提高了密码系统的数据完整性和系统可靠性，进而提高了密码系统的安全性。

对于SM2所使用的椭圆曲线，h=1。因此，步骤③对于保密来说是非本质的。但是，如果h或P_B发生了错误或PB选得不好，致使S=hP_B=O，则它可以把错误检查出来。 在解密算法中加入了更多的检错功能，这是因为解密的密文是经过信道传输过来的，由于信道干扰的影响和对手的篡改，在密文中含有错误或被篡改的可能性是存在的。采取措施把错误和篡改检测出来，对提高密码系统的数据完整性、系统可靠性和安全性是有益的。

第十二讲 数字签名基础

二、数字签名模型

一个数字签名体制包括两个方面的处理：
施加签名： 为数据产生签名
验证签名： 验证签名的真伪

三、利用RSA密码实现数字签名

1、利用公钥密码实现数字签名的一般方法

凡是能够构成安全SIG和VER的公钥密码都可实现数字签名 RSA密码、ElGamal密码、椭圆曲线密码、许多其他密码

为了实施数字签名，应成立管理机构； 制定规章制度，统一技术标准，用户登记注册，纠纷的仲裁，其它。

2、利用RSA密码实现数字签名：

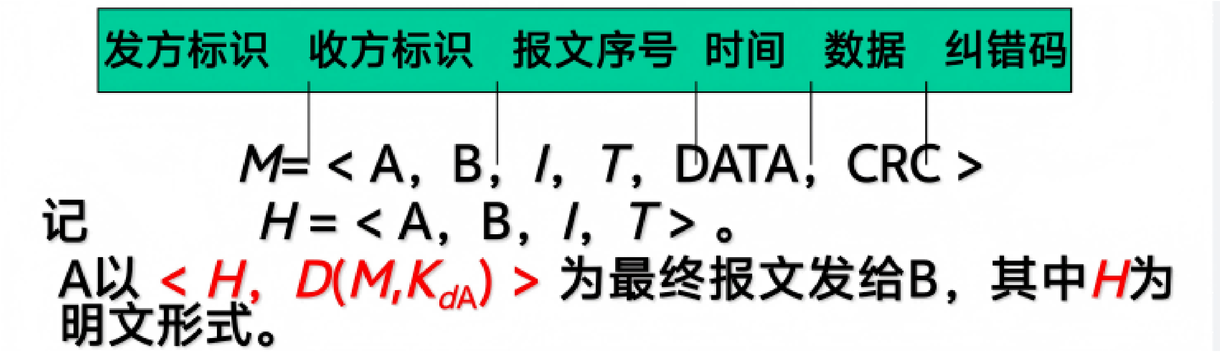
对于RSA密码， $D(E(M))=(M^e)^d=M^{ed}=(M^d)^e=E(D(M)) \bmod n$，所以RSA可同时确保数据的秘密性和真实性。在这里，SIG=D，VER=E 因此利用RSA密码可以同时实现数据加密和数字签名。

（1）签名算法

设M为明文，K_{eA} =<e,n>是A的公开加密钥，K_{dA} =<d,p,q,φ(n)>是A的保密的解密钥，则A对M的签名过程是，$S_A = D(M, K_{dA}) = (M^d) \bmod n$，$S_A$ 便是A对M的签名。 验证签名的过程是， $E(S_A, K_{eA}) = (M^d)^e \bmod n = M$ 如果收信者验证得到正确的数据M，则签名为真。

上述验证中，如果收信者验证得到正确的数据M，则判定签名为真。有时收信者事前不知道M，如何判定？

- 方法一：合理设计明文的数据格式



只要用A的公钥验证签名并恢复出正确的附加信息H = < A，B，I，T >，便可断定明文M是否正确。记接收到的附加信息为H，恢复出的为H'，仅当H=H'时判定签名为真。 设附加信息 H=<A，B，I，T> 的二进制长度为L，则错判概率 $p_e \le 2^{-L}$ 。

- 方法二：对Hash（M）签名

（2）对RSA数字签名的攻击 ①一般攻击: 因为e和n是用户A的公开密钥，所以任何人都可以获得并使用e和n。攻击者可随意选择一个数据Y，并用A的公钥计算

$X = (Y)^e \bmod n$

因为 $Y = (X)^d \bmod n$ ，于是可以用Y 伪造A的签名。因为Y 是A对X 的一个有效签名。
注意：这样的 X 往往无正确语义！因此，这种攻击在实际上有效性不大！

②利用已有的签名进行攻击: 攻击者选择随机数据 M3，且M3=M1M2 mod n 。

攻击者设法让A对M1和M2签名：$S1 = (M1)^d \bmod n$， $S2 = (M2)^d \bmod n$

于是可以由S1和S2计算出A对M3的签名。因为$S1S2=(M1)^d(M2)^d \bmod n = (M3)^d \bmod n = S3$ **对策：A不直接对数据M 签名，而是对HASH(M)签名。** 此时：$S1 = (HASH(M1))^d \bmod n$， $S2 = (HASH(M2))^d \bmod n$

而，$(HASH(M1))^d (HASH(M2))^d \neq (HASH(M1M2))^d \bmod n$
所以：S3≠S1S2 ，于是不能由S1和S2计算出A对M3的签名。

③攻击签名获得明文: 攻击者截获C，$C = (M)^e \bmod n$ 。

攻击者选择小的随机数r，计算：$x = r^e \bmod n$，$y = xC \bmod n$， $t = r^{-1} \bmod n$

攻击者让A对y签名， $S = y^d \bmod n$ ，于是攻击者又可截获S

攻击者计算$tS=r^{-1}y^d=r^{-1}x^dC^d=C^d =M \bmod n$
对策：A不直接对数据M签名，而是对HASH(M)签名

结论

- 不直接对数据M签名，而是对HASH(M)签名。
- 使用时间戳
- 对于同时确保秘密性和真实性的通信，应当先签名后加密（致使无法对签名进行攻击）。

四、利用ElGamal密码实现数字签名

利用ELGamal密码可以构建安全的SIG和VER，所以可以实现数字签名
如果知道了k重复使用签名的m1和m2，便可求出k，进而求出保密的解密钥。 **由此可知，不要随便给别人签名。不要直接对m签名，而是对HASH(m)签名。**
签名时需要使用随机数k，所以需要有良好的随机数产生器。 **缺点**：由于取（r，s）作为m的签名，所以数字签名的长度是明文的两倍，数据扩张一倍。

第十三讲 中国商用公钥密码SM2签名算法

传统椭圆曲线签名直接使用m产生签名；
而SM2使用M* =ZA||M， e = Hash(M*)
SM2使用了用户参数和系统参数，起到一定的认证作用，提高了安全性：
IDA是A的标识。ENTLA是IDA的长度。基点是G = (xG, yG)
A的私钥是dA，A的公钥是PA=dAG = (xA, yA)
ZA=Hash (ENTLA||IDA||a||b|| xG||yG||xA||yA)
传统椭圆曲线签名算法计算：点R（x R ,y R） = kG，并记 r = x R ；
SM2计算：点G1(x1, y1)=kG，且计算r = (e+x1) mod n；
传统椭圆曲线签名算法计算：
s = (m - dr)k-1 mod n ；
SM2计算：s = ((1 + dA)−1 · (k − r · dA)) mod n。M 没有直接出现，而是通过r参与其中；私钥dA 作用了两次。
SM2增加了合理性检查，确保签名正确，提高安全性。
例如第⑤中检查r+k=n是否等于n。
如果r+k=n，则k = -r mod n，会使s = ((1 + dA)−1 · (k − r · dA)) mod n = ((1 + dA)−1 · (-r)(1 + dA)) = -r mod n 。 s = -r mod n，显然是不合适的。

第十四讲 密码协议

重点内容与书上重复