密码学知识点

古典密码

•**代换密码**是明文中的每一个字符被替换成密文中的另一个字符。接受者对密文做反向替换就可以恢复出明文。

• 单表代换、流密码　　多表代换、分组密码

• 单表代换密码：

移位密码（如凯撒密码）可用穷搜索攻击

仿射密码

$$e_K(x) = (ax + b) \bmod 26$$

$$d_K(y) = a^{-1}(y - b) \bmod 26$$

例 凯撒(Caeser)密码是对英文26个字母进行移位代换的密码，其q=26。例如，选择密钥k=3，则有下述代换表：

A：a b c d e f g h i j k l m n o p q r s t u v w x y z

A'：D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

明文：　m = veni，vidi，vici

密文：c=E(m)= YHAL，YLGL，YLFL

（意思是“我来，我见，我征服”，曾经是恺撒征服本都王法那西斯后向罗马元老院宣告的名言）

解密运算为$D_3=E_{23}$，用密钥k=23的加密表加密就可恢复明文。

又称为加法密码(Additive Cipher)。

1.设由仿射变换对一个明文加密得到的密文为edsgickxhuklzveqzvkxwkzukvcuh，又已知明文的前两个字符是'if'，对该密文解密。

答：e=4 d=3 i=8 f=5　(26个字母下标从0开始) $E_{a,b}(m)=am+b(\mathrm{mod}\ 26)$

$E(i)=e,\ 4\equiv 8*a+b(\mathrm{mod}\ 26)$

$E(f)=d,\ 3\equiv 5*a+b(\mathrm{mod}\ 26)$

由上述两个式子可推出a=9，b=10，所以$m=9^{-1}(c-10)(\mathrm{mod}\ 26)$

2.设多表代换密码$C\equiv AM+B(\mathrm{mod}26)$中，A是2X2矩阵，B是0矩阵，又知明文'dont'被加密为'elni'，求矩阵A。

解：设矩阵 $A=\begin{bmatrix} a & b \\ c & d \end{bmatrix}$，

dont=(3,14,13,19)

elni=(4,11,13,8)

$$\begin{bmatrix} 4 \\ 11 \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 3 \\ 14 \end{bmatrix} (\mathrm{mod}\ 26)$$

$$\begin{bmatrix} 13 \\ 8 \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 13 \\ 19 \end{bmatrix} (\mathrm{mod}\ 26)$$

解得：$A=\begin{bmatrix} 10 & 13 \\ 9 & 23 \end{bmatrix}$

• 多表代换密码：以一系列（两个以上）代换表依次对明文消息字母进行代换的加密方法

维吉尼亚密码

密码体制 1.4　维吉尼亚密码

设 m 是一个正整数。定义$\mathcal{P}=\mathcal{C}=\mathcal{K}=(\mathbb{Z}_{26})^m$。对任意的密钥 $K=(k_1,k_2,\cdots,k_m)$，定义

$$e_K(x_1,x_2,\cdots,x_m)=(x_1+k_1,x_2+k_2,\cdots,x_m+k_m)$$

和

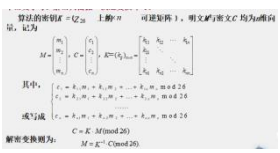
$$d_K(y_1,y_2,\cdots,y_m)=(y_1-k_1,y_2-k_2,\cdots,y_m-k_m)$$

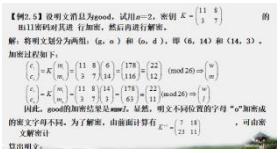
以上所有的运算都是在$\mathbb{Z}_{26}$上进行。

维吉尼亚密码

- 例令q=26，m=polyalphabetic cipher，密钥字k=RADIO，即周期d=5，则有
- 明文m=p o l y a l p h a b e t i c c i p h e r
- 密钥k=R A D I O R A D I O R A D I O R A D I O
- 密文$c=E_k(m)$=G O O G O C P K T P N T L K Q Z P K M F
- 其中，同一明文字母p在不同的位置上被加密为不同的字母G和P。
- 维吉尼亚密码是用d个凯撒代换表周期地对明文字母加密。

• **希尔密码**算法的基本思想时将 n 个明文字母通过线性变换，将特闷转换为 n 个密文字母，解密只需做一次逆变换即可。





• **矩阵求逆方法**

• 希尔密码可以有效抵御**唯密文攻击**，但不能抵御**已知明文攻击**

• **置换密码**又称**换位密码**，加密过程中明文的字母保持相同，但顺序被打乱了。

密码体制 1.6　置换密码

令 m 为一正整数。$\mathcal{P}=\mathcal{C}=(\mathbb{Z}_{26})^m$，$\mathcal{K}$ 由所有定义在集合 $\{1,2,\cdots,m\}$ 上的置换组成。对任意的密钥（置换）π，定义加密变换：

$$e_\pi(x_1,\cdots,x_m)=(x_{\pi(1)},\cdots,x_{\pi(m)})$$

相应的解密变换为：

$$d_\pi(y_1,\cdots,y_m)=(y_{\pi^{-1}(1)},\cdots,y_{\pi^{-1}(m)})$$

上式中 π^{-1} 为置换 π 的逆置换。

·置换密码在实质上是希尔密码的特例。

1. 由以下置换对明文 todayissunny 进行加密，结果应为 _____。

x	1	2	3	4	5	6
$\pi(x)$	3	5	1	6	4	2

·对称密码体制分类：**流密码、分组密码**；对称密码的两个基本运算：**代换、置换**；

·对称密码的两个基本设计原则

扩散：明文的统计结构被扩散小时到密文的长程统计特性，使得明文和密文之间的同级关系尽量复杂；

混乱：使得密文的统计特性与密钥的取值之间的关系尽量复杂。

·一个密码体制的明文必要分组长度 n 若为 1，则称该密码为流密码，否则（即 n>1）称该密码为分组密码。

·**同步流密码**，就是生成的密钥流独立于明文流；**自同步流密码**，每个密钥字符都从以前密文的固定 n 个字符中导出。

完善保密理论

·安全分类：计算安全、可证明安全、无条件安全

·完善保密的定义：给定密文 y，明文 x 的后验概率等于明文的先验概率。通俗地说，完善保密性就是攻击者不能通过观察密文获得明文的任何信息。

·shannon 定理：假设密码体制（P,C,K,E,D）满足|K|=|C|=|P|。该密码体制是完善保密的，当且仅当每个密钥被使用的概率都是 1/|K|，并且对于任意的 x∈P 和 y 属于 C，存在唯一的密钥 K 使得 e_K（x）=y。

·一次一密

密码体制2.1 一次一密

假设$n \geq 1$是正整数，$P = C = K = (\mathbb{Z}_2)^n$。对于$K \in (\mathbb{Z}_2)^n$，定义$e_K(x)$为$K$和$x$的模2向量和（或者说是两个相关比特串的异或）。因此，如果$x = (x_1, x_2, ..., x_n)$并且$K = (K_1, K_2, ..., K_n)$，则

$$e_K(x) = (x_1 + K_1, ..., x_n + K_n) mod 2$$

解密与加密是一样的。如果$y = (y_1, ..., y_n)$，则

$$d_K(y) = (y_1 + K_1, ..., y_n + K_n) mod 2$$

·乘积密码

·信息量和熵

分组密码

- **代换-置换网络 SPN**

密码体制3.1 （代换-置换网络）：
设l,m和Nr都是正整数，$\pi_s:\{0,1\}^l\to\{0,1\}^l$和$\pi_p:\{1,\ldots,lm\}\to\{1,\ldots,lm\}$都是置换。设$P=C=\{0,1\}^{lm}$，$K\subseteq(\{0,1\}^{lm})^{Nr+1}$是由初始密钥$K$用密码编排算法生成的所有可能的密钥编排方案之集。对一个密钥编排方案$K^1,\cdots,K^{Nr}$，我们使用下面的算法3.1来加密明文x。

- SPN 特色：无论从硬件还是软件角度来看，这种设计均简单有效；SPN 应该有更长的密钥长度和分组长度；SPN 有许多变体，比如使用不同的 S 盒。

- **堆积引理**

引理3.1（堆积引理）设$\mathbf{X}_{i_1},\ldots,\mathbf{X}_{i_t}$是独立随机变量，$\varepsilon_{i_1,i_2,\ldots,i_t}(i_1<i_2<\ldots<i_k)$表示随机变量$\mathbf{X}_{i_1}\oplus\mathbf{X}_{i_2}\oplus\ldots\oplus\mathbf{X}_{i_t}$的偏差，则

$$\varepsilon_{i_1,i_2,\ldots,i_t}=2^{k-1}\prod_{j=1}^{k}\varepsilon_{i_j}$$

- **线性密码分析**，（提出了对 DES 算法的一种新的攻击方法），是一种已知明文攻击。
  - 找出一组 S 盒的线性逼近
  - 导出整个 SPN（除最后一轮）的线性逼近
  - 利用已有的明密文对，测试候选密钥
  - 输出密钥
- **差分密码分析**，是一种选择明文攻击。

•两种分析的区别在于差分密码分析包含了将两个输入的异或与其相对应的两个输出的异或相比较。

- **CBC 模式（密码分组链接）模式：加密的输入时当前明文组和前一密文组的异或**
- 分组密码采用**混乱**原则和**扩散**原则来抵抗攻击者对该密码体制的统计分析。
- 有限域的特征一定是**素数**，有限域的元素个数一定是器特征的**整数次幂**。
- 5 种操作模式：

  ECB（电话本）模式 各明文组以同一密钥加密

  CBC 模式（密码分组链接）模式：加密的输入时当前明文组和前一密文组的异或

  CFB（密码反馈）模式 每次处理 j 位输入，上次密文加密产生伪随机再与当前明文异或

  OFB（输出反馈）模式 与 CFB 不同的是加密的输入时前一次加密的输出

私钥密码学：DES

• Feistel 网络：思想时把输入块分成左右两部分 L（i-1）和 R（i-1），变幻时在密码的第 1 轮只是用 R（i-1）；每个阶段有函数 g 工作，由第 1 个密钥控制（叫子密钥）

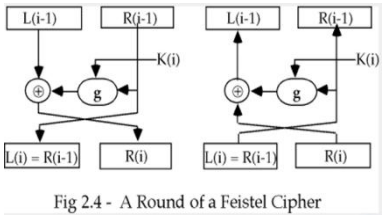
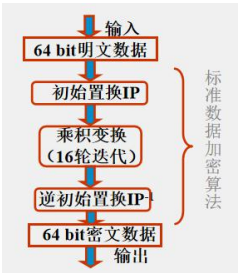


Fig 2.4 - A Round of a Feistel Cipher

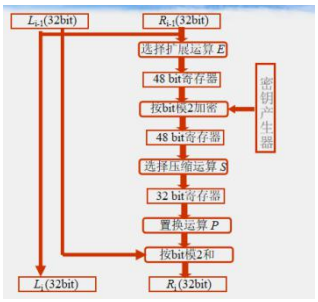
•DES 算法：分组长度为 64bits（8bytes），密文分组长度也是 64bits。密钥长度为 64bites，有 8bits 奇偶校验，有效密钥长度为 56bits。算法主要包括：初始置换 IP、16 轮迭代的乘积变换、逆初始置换 IP^{-1} 以及 16 个子密钥产生器。

• 初始置换 IP：将 64 bit 明文的位置进行**置换**，得到一个乱序的 64 bit 明文组，而后分成**左右两段**，每段为 32 bit，以 L0 和 R0 表示，IP 中各列元素位置号数相差为 8，相当于将原明文各字节**按列写出**，各列比特经过偶采样和奇采样置换后，再对各行进行逆序。将阵中元素**按行读出**构成置换输出。



• 逆初始置换 IP^{-1}：将 16 轮迭代后给出的 64 bit 组进行置换，得到输出的密文组。输出为阵中元素按行读得的结果。

• 乘积变换：它是 DES 算法的核心部分。将经过 IP 置换后的数据分成 32 bit 的左右两组，在迭代过程中彼此左右交换位置。每次迭代时只对右边的 32 bit 进行一系列的加密变换，在此轮迭代即将结束时，把左边的 32 bit 与右边得到的 32 bit 逐位模 2 相加，作为下一轮迭代时右边的段，并将原来右边未经变换的段直接送到左边的寄存器中作为下一轮迭代时左边的段。在每一轮迭代时，右边的段要经过**选择扩展运算 E、密钥加密运算、选择压缩运算 S、置换运算 P 和左右混合运算。**



• 加密过程和解密过程

加密过程:运算进行16次后就得到密文组。

$L_0R_0 \leftarrow IP$（〈64 bit 输入码〉）

$L_i \leftarrow R_{i-1}$　　$i=1,\ldots,16$

$R_i \leftarrow L_{i-1} \oplus f(R_{i-1}, k_i)$　　$i=1,\ldots,16$

〈64 bit密文〉$\leftarrow IP^{-1}(R_{16}L_{16})$

解密过程:DES的加密运算是可逆的，其解密过程可类似地进行。

$R_{16}L_{16} \leftarrow IP$（〈64 bit密文〉）

$R_{i-1} \leftarrow L_i$　　$i=16,\ldots,1$

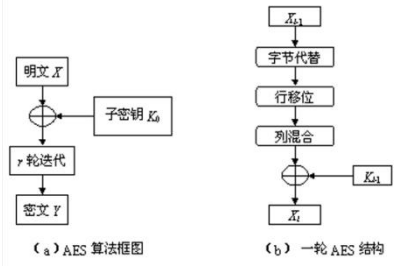
$L_{i-1} \leftarrow R_i \oplus f(L_{i-1}, k_i)$　　$i=16,\ldots,1$

〈64 bit明文〉$\leftarrow IP^{-1}(R_0L_0)$

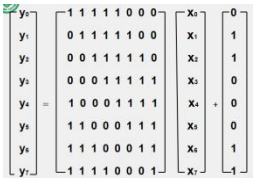
- 掷一对无偏的骰子，若告诉你得到的总的点数为 7，请问获得了多少信息量？
- 有限域GF(9)是利用GF(3)上不可约多项式 x^2+1构造的，x+1是GF(9)的一个元素，请计算出它的逆。
- 按照黑板上给出的S盒，计算随机变量$X_2 \oplus X_3 \oplus Y_2$的偏差。

分组密码：AES

- AES 算法结构



（a）AES 算法框图　　（b） 一轮 AES 结构



- **字节替代**是一个非线性的字节替代，独立地在每个状态字节上进行运算。它包括两个变换：在有限域 GF(2^8)上求乘法逆，‘00’映射到它自身；在 GF(2)上进行右面的仿射变换。

- **行移位**:状态阵列的后 3 行循环移位不同的偏移量。第 1 行循环移位 C1 字节，第 2 行循环移位 C2 字节，第 3 行循环移位 C3 字节。偏移量 C1、C2、C3 与分组长度 Nb 有关。

- **列混合**：将状态的列看作是有限域 GF(2^8)上的多项式 a(x)，与多项式 c(x) = 03 x^3 + 01 x^2 + 01 x +02 相乘(模 x^4＋1)

- **轮密钥加**：将轮密钥与状态按比特异或

- 在高级加密标准 AES 规范中，分组长度是 128 位，密钥的长度是 128 位。

- 密钥长度 128、192、256，迭代轮数 10、12、14。

1.对字节a=1011 0110字节替代变换，设a的逆为a^{-1} （必考！）

答：先求a的逆，再用仿射变换即可

1 由a得$(x^7+x^5+x^4+x^2+x)a^{-1} \equiv 1 (\mathrm{mod}\ x^8+x^4+x^3+x+1)$

所以$a^{-1}=x^6+x^5+x^4+x^3$ 即0111 1000 （二进制对应位数有1就代表有x的那一次方）

> 根据老师给的考试要点，仿射变换和m(x)题目会给出，a的逆试试就出来了

RSA

• 公钥密码体制：**RSA 密码体制（基于大整数分解）、ElGamal 密码体制（基于离散对数问题）公钥进行加密，私钥进行解密**

• 公钥密码学不能提供无条件安全性，我们仅仅讨论公钥密码体制的计算安全性。

• 公钥密码体制解决了私钥密码体制的**安全性、防伪性、鉴权性问题**

• **公钥密码体制可看成是陷门单向函数。**

• RSA 密码体制

设$n=pq$，其中p和q是素数。设$\mathscr{P}=\mathscr{C}=\mathbb{Z}_n$，且定义

$$\mathscr{K}=\{(n,p,q,a,b):ab\equiv 1(mod\phi(n))\}$$

对于$K=(n,p,q,a,b)$，定义

$$e_K(x)=x^b \mod n$$

和

$$d_K(y)=y^a \mod n$$

$(x,y\in\mathbb{Z}_n)$。值n和b组成了公钥，且值p,q,a组成了私钥。

算法5.4 RSA参数的生成
1.生成两个大素数，p和q，$p\neq q$;
2.$n\leftarrow pq$，且$\phi(n)\leftarrow(p-1)(q-1)$;
3.选择一个随机数b（$1<b<\phi(n)$），使得$gcd(b,\phi(n))=1$;
4.$a\leftarrow b^{-1} \mod \phi(n)$;
5.公钥为(n,b)，私钥为(p,q,a)。

1.RSA加密解密（必考！）

选俩素数p,q n=p*q φ(n)=(p-1)*(q-1) d*e ≡ 1 mod φ(n)

加密：c ≡ m^e mod n （e和n会给出）

解密：m ≡ c^d mod n (一般d要自己求)

2.可能会用到的公式：a*b (mod q) = a (mod q) * b (mod q)

> 推测是RSA中m^e或c^d可以拆成两个乘积的形式

• Pollard ρ 算法

• Diffie－Hellman（以下简称 DH)密钥交换是一个特殊的交换密钥的方法，目的是使得两个用户能够安全的交换密钥，得到一个共享的会话密钥，算法本身不能用于加密解密

1.在Diffie- Hellman密钥交换过程中，设大素数p=11,a=2是p的本原根。（必考！）
(1) 用户A的公开钥Y_A=9,求其秘密钥XA。
(2)设用户B的公开钥Y_B=3,求A和B的共享密钥K。

答：1 $Y_A=a^{XA} \mod p = a^{XA} \mod p$ 即 $9=2^{XA} \mod 11$ 所以 $X_A=6$ (考试时一个一个试就行)

2 $K=Y_B^{XA} \mod p = 3^6 \mod 11 = 3$

2.D-H密钥交换协议：（必考！）

已知私钥为X，公钥为Y，p是一大素数，a是p的本原根，a和p公开，K为共享密钥

用户A：计算$Y_A=a^{XA} \mod p$发送给B 共享密钥$K=Y_B^{XA} \mod p$

用户B：计算$Y_B=a^{XB} \mod p$发送给A 共享密钥$K=Y_A^{XB} \mod p$

> 通过上述操作求得的两个K值相等，这样就安全的求得了一个公共的密钥

1.RSA加密体制中，接收方的公开钥是(e,n)=(5,35)，接收到的密文是C=10，求明文M（必考！）

答：（m ≡ c^d mod n 所以得先算出d -----> d*e ≡ 1 mod φ(n)）

n=35 所以 p=5，q=7

φ(35) = (p-1)*(q-1) =4*6=24

因为 d*e ≡ 1 mod φ(n) 所以d=5 （这个到时候一个一个试就行）

m ≡ c^d mod n ≡ 10^5 mod 35 ≡ 5 （注意不是等号）

2.假设明文m=5，e=7，p=11，q=13，给出RSA的加密解密过程（必考！）

答：n=p*q=11*13=143 φ(143) = (p-1)*(q-1) = 10*12=120

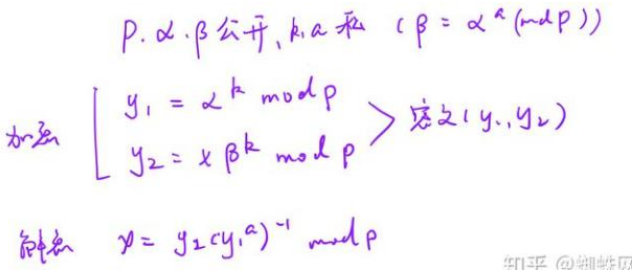
因为d*e ≡ 1 mod φ(n) 即d*7 ≡ 1 mod 120 所以 d=103

加密：c ≡ m^e mod n ≡ 5^7 ≡ 47 mod 143

解密：m ≡ c^d mod n ≡ 47^{103} ≡ 5 mod 143

ElGamal 算法

- **离散对数问题**：如果能找到 a，使得 α ^a=β，那么 a 成为 β 的离散对数。
- ElGamal 算法



1.ElGamal签名体制中，假设p=19，g=13。签名者A的私钥为x = 10，试计算公钥。设消息M=15，k=11，求签名过程并验证。（必考！）

答：公钥：$y \equiv g^x \bmod p \equiv 13^{10} \bmod 19 = 6$

签名：$r \equiv g^k \pmod p \equiv 13^{11} \bmod 19 = 2$

$s \equiv (H(m)-xr)k^{-1} \pmod{p-1} \equiv (m-xr)k^{-1} \pmod{p-1} \equiv (15-10*2)*11^{-1} \bmod 18 = 11$

所以(r,s)=(2,11)

验证：$y^r r^s \equiv 6^{2}*2^{11} \bmod 19 \equiv 8 \equiv g^m \pmod p \equiv 13^{15} \equiv 8 \bmod 19$

> 根据老师的重点中给出的a*b (mod q) = a (mod q) * b (mod q)应该是用在比如这题求$6^{2}*2^{11}$ mod 19就等同于求6^{2} mod 19 * 2^{11} mod 19 ≡ 17 *15 ≡ 8 mod 19

Hash 函数

• 找出任意两个不同的输入 x,y，使得 H(y)=H(x)，在计算上是不可行的，则称其为强单向哈希函数，用于抵抗生日攻击。

• 哈希函数有哪些？MD5，SHA1 ，SHA224，SHA256，SHA384，SHA512 国密：SM3

•**哈希函数**是一种公开函数，用于吧任意长度的输入通过散列算法，变换成固定长度的输出，该输出就是散列值/哈希值。

• hash 函数特点：输入任意长度，输出固定长度；计算 hash 值速度快；防碰撞特性；隐藏性；谜题友好。

• **消息认证码 MAC**（带密钥的 Hash 函数）：密码学中，通信实体双方使用的一种验证机制，保证消息数据完整性的一种工具。安全性依赖于 Hash 函数，故也称带密钥的 Hash 函数。消息认证码是基于密钥和消息摘要所获得的一个值，可用于数据源发认证和完整性校验。

• **消息认证码的作用是解决消息的完整性和发送者的正确性。**

• Hash 与 MAC 的区别，Hash 只能保证消息的完整性，MAC 不仅能够保证完整性，还能够保证真实性。

• 无条件安全&计算安全

• 加密算法满足下列两点则认为是计算上安全的

破译密文的代价超过被加密信息的价值

破译密文所花的时间超过信息的有用期

• 矩阵求逆

1. 如果 Hash 函数的散列值为 80 比特，那么生日攻击的代价为（ ）。

A. 2^{80}　　B. 2^{160}　　C. 2^{60}　　D. 2^{40}

• 密码体制包括：**私钥加密技术、公钥加密技术。**

• 数字签名包括**不可伪造性、认知性、不可重复性、不可修改性、不可否认性。**

• 四种攻击类型：

•唯密文攻击中密码分析者所能利用的数据资源仅为密文，这是对密码分析者最不利的情况。（最难，一般采取穷搜索法）

• 已知明文攻击中，密码分析者处理密文外，还有一些已知的“明文-密文对”来破译密码。

• 选择明文攻击中，密码分析者不仅可以得到一些“明文-密文对”，还可以选择被加密的明文，并获得相应的密文。

•选择密文攻击中，密码分析者可以选择一些密文并得到相应的明文。这种密码分析多用于攻击公钥密码体制。

• 主动攻击：**终端、篡改、伪造**。被动攻击：**消息内容泄露、业务流分析**。

•流密码和分组密码的区别：分组密码以一定大小作为每次处理的基本单元，而流密码是一一个元素作为基本处理单元。

1. 用中国剩余定理计算同余方程组：$\begin{cases} x \equiv 1 \bmod 3 \\ x \equiv 2 \bmod 5 \\ x \equiv 3 \bmod 7 \end{cases}$