密码学题型总结

第一章 概论与古典密码

概论

- 五类安全业务
    - 鉴别
    - 访问控制
    - 数据机密性
    - 数据完整性
    - 抗抵赖性
- 攻击：主动攻击、被动攻击

古典密码

- 基本思想：置换和代换（现代密码学中也有体现）

计算

- 仿射变换：知道明文M，密文C，求A
    1. $C = AM + B$
    2. 列方程
    3. 求解
- 仿射变换：加密及解密验证
    1. 加密：$C = AM + B$
    2. 解密：$M = A^{-1} \cdot (C - B)$

第二章 流密码

- 流密码可分为同步和异步流密码
    - 同步流密码：状态独立于明文字符流
    - 异步（自同步）：状态依赖于明文字符流

计算

LFSR

- 给生成多项式/方程 + 初态，算输出
    - 根据初态+方程算下一个输出
- 周期的计算
    - m序列为2^n-1
        - 特征多项式为本原多项式
    - 周期r为特征多项式周期p的因子
- 破译密码系统
    1. 密钥串 = 明文串 + 密文串（m级LFSR需要 2m bit）
    2. 建立方程

$$(a_{m+1} \ \cdots \ a_{2m}) = (c_m \ \cdots \ c_1) \cdot X$$

X每列为对应的状态向量

3. 解方程得到 C 向量

- 求钟控序列输出及周期
  - 周期
    1. 分别求出a_k, b_k 的周期 p_1, p_2
    2. c_k 周期 $p = p_1 p_2$
  - 输出：a=0，b不动；a=1，b往后一个

特殊题型

- 特定比特对$s_1 s_2$
  - 思路一：讨论长度为i的1游程个数个数
  - 思路二：考察 $s_1 s_2 * \cdots *$ 状态数目
- 判断下一状态
  - 从状态转移入手
- 知道 2n-2 bit的明密文对，破译密钥流生成器
  1. 穷举**密钥流**余下的 2bit
  2. 建立方程 $S_{n+1}^T = C \cdot X$
  3. 排除X不可逆
  4. 求得 $C = S_{n+1}^T \cdot X^{-1}$
  5. *验证C是否为m序列（要求输出为m序列时）

第三章 分组密码

DES

- 输入：明文64bit，密钥56/64bit（PC-1会压缩）
- 输出：密文64bit
- 多轮迭代密码
- 二重DES安全性
  - 存在中间相遇攻击 $X = E_{K1}[P] = D_{K2}[C]$
- EDE（三重DES）
  - 密钥长度：168bit（三个密钥时）
  - 分组长度：64bit
  - 输出密文长度：64bit

证明

- 证明取反特性
  1. 说明异或计算中，一项取反，结果取反（有时间可以真值表证明）
  2. 置换、循环移位可保持取反
  3. K取反，子密钥全部取反
  4. F中，进入S盒之前：$E(\overline{R_{i-1}}) \oplus \overline{K_i} = \overline{E(R_{i-1})} \oplus \overline{K_i} = E(R_{i-1}) \oplus K_i$

     即K，R同时取反，F的输出保持不变
  5. L取反了之后与F异或，导致结果时取反的

6. 输入L，R，输出的L，R是反的；每一轮如此，则最后输出是$\overline{Y}$

- ECB模式错误传播
    - ECB的分组传播出错，只影响下一分组
    - 加密前的错误可能传播到后续所有分组
        - 但是不影响接收者解密
- CFB模式错误传播
    - 对于接收者而言，各密文分组C_i是相互独立的
    - 1比特出错，移入下一分组的SR，并不断左移
    - 经过 64/j 轮后移出寄存器
- IDEA的模数选择
    - 乘法：每个元素需存在逆元，模为素数比较方便
    - 加法：每个元素存在逆元，且方便处理

计算

- S盒计算：输入6 bit，输出4 bit
    1. 横坐标x_1x_6
    2. 纵坐标$x_2x_3x_4x_5$
    3. 查表输出
- IP置换使用
    - 表里的数字是源分组中对应位的值

AES

- 分组长度：128bit（固定）
- 密钥长度：128，192，256bit
- 状态：开始输入的**明文分组**、**中间分组**、**密文分组**
- 输出：以字节为单位按列优先顺序从状态整列取出（128bit）
- 状态用 $4 \times N_b$矩阵表示（状态矩阵）
    - N_b=分组长度/32，AES中$N_b = 4$
    - 字节为单位
    - 列优先顺序摆放
- 计算部件
    - 字节代换（非线性）
        - 逆元映射
        - 仿射变换（可逆）
    - 行移位
    - 列混合
    - 密钥相加
- 密钥编排
    - 密钥扩展
    - 轮密钥选取
- 加解密
    - 相似不相同
    - 运算部件使用顺序不同

计算

- AES多项式相乘（系数为多项式）
    1. 列方程（下标和=对应项下标）
    2. 计算每一项
- 两个小多项式相乘
    1. 分解成8位中只有一位为1之和
    2. X乘
- X乘计算
    1. 左移1位，末尾补0
    2. 原来的最高位为1的话，与'1B'做异或（为0不用）
- *密钥扩展的计算
    1. 前4个字为种子密钥（一个字32bit）
    2. 分是不是4的倍数计算后面的字
- 验证S盒正确性
    1. 找逆元
    2. 做仿射变换（二元运算）
- *差异的扩散：1bit变5bit

第四章 公钥密码体制

数学部分

- 扩展欧几里得算法求模n下a的逆元
    1. Q {X1 X2 X3} {Y1 Y2 Y3}
    2. - {1 0 n} {0 1 a}
    3. 除、减、移
- 本原根
    - 验证——定义法（遍历1至$\phi(n)$）
    - 求本原根
        1. 分解素因数 $\phi(p) = k_1^{n_1} k_2^{n_2} \cdots$
        2. 验证 $a^{\frac{\phi(p)}{k_i}} \neq 1 \ mod \ p$ 对所有 $k_i \mid \phi(p)$ 成立

RSA

- RSA密钥原理
    - 密钥生成
    - 加密
    - 解密

计算

- 求明文/密文
    1. 求私钥d（求模$\phi(n)$逆元）
    2. m^e或m^d

ElGamal

- 基于离散对数困难问题

- 密文长度是明文长度的两倍

计算

- 给 {p，g，y}及k，求密文
    1. $C_1 = g^k \ mod \ p$
    2. $C_2 = y^k M \ mod \ p$
    3. $C = \{C_1, C_2\}$
- 给C1，C2，x，求密文M
    - $M = C2/(C_1)^x \ mod \ p$
- 给k，C1，求C2
    1. 求k，$C_1 = g^k \ mod \ p$
    2. 求C2

ECC

计算

- 给a，b，p求所有的点
    1. $E_p(a,b) = x^3 + ax + b \ mod \ 11$
    2. $x:0 \rightarrow p-1, \ n = y^2 = f(x)$
    3. 验 n = f(x) 是否是平方剩余，$n^{\frac{p-1}{2}} = 1 \ mod \ p$ 则为平方剩余
    4. 把是平方剩余的n对应的点算出来（一对），最后加上无穷远点O
- 加法计算
    1. 判断是否和为0，不为0继续
    2. 计算λ
    3. 计算x_3
    4. 计算y_3
- 公钥计算
    1. $P_A = n_A \cdot G$
- 给消息P_m，随机数k，公钥P_A，求密文C_m
    1. 计算$C_1 = kG$
    2. 计算kP_A
    3. 计算$C_2 = P_m + kP_A$
    4. C = {C1，C2}
- 从密文恢复明文（都是 第二项 - 私钥*第一项）
    - $P_m = C_2 - n_A C_1$

第五章 密钥分配与密钥管理

单钥加密体制

公钥加密体制

- 对D-H密码交换协议的攻击
    1. Z拦截 $A \rightarrow B$
    2. Z冒充A，给B发Y_Z
    3. Z冒充B，给A发Y_Z
    4. $K_{AZ} = K_{ZA} = {Y_A}^Z$

$$K_{BZ} = K_{ZB} = {Y_B}^Z$$

5. Z在A，B之间转发消息，实现窃听

- D-H密钥交换，给素数p，本原根a，公钥Y，计算私钥
  - 即求 $a^x = Y \ mod \ p$
- D-H求共享密钥K
  - $K = {Y_B}^{X_A}$

秘密分割

- 给子密钥，求多项式
  1. 选k个秘密
  2. 分别计算 $f(i)\frac{x-x_i}{i_j-i} \ mod \ q$
  3. 求和，得到$f(x)$
- 恢复秘密
  - $s = f(0)$

第六章 消息认证及哈希函数

算法特征

- 安全需求
  - 抗原像攻击
  - 抗弱碰撞攻击
  - 抗强碰撞攻击
  - *伪随机性

MD5

- 迭代型哈希算法一般结构
- 输入：任意长消息
- 输出：128 bit消息摘要
- 分组：512 bit
- 轮数：4轮，每轮16步迭代运算
- 缓冲区：4个32bit寄存器（little-endian存储）
- 抗穷搜索攻击
  - 抗弱碰撞性：$O(2^{80})$
  - 抗强碰撞性：$O(2^{64})$（比较弱）

SHA

- 结构与MD5非常类似
- 输入：小于2^{64} bit
- 输出：160 bit消息摘要
- 分组：512 bit
- 轮数：4轮，每轮20步迭代
- 缓冲区：5个32bit寄存器

- 抗穷搜索攻击
    - 抗弱碰撞性：$O(2^{160})$
    - 抗强碰撞性：$O(2^{80})$

简答题

- CBC与CFB（用CFB获得与CBC相同的输出）
    1. CBC递推：$O_{i+1}=E_k(O_i\oplus D_{i+1})$

       CBC输出：$MAC=S_M(O_N)$（取最左M比特）
    2. CFB下，取 j=64（即全部移入下一寄存器）
    3. CFB递推：$O_{i+1}=E(O_i\oplus P_{i+1})$

       CFB输出：$MAC=S_M(E_K(O_{N-1}))$（最后一轮加密结束后，直接取M比特输出）
- CBC模式，篡改分组
    - 利用取反特性，将对应分组盒子密钥一起取反
    - 最后得到一样的哈希值
- RSA构造哈希函数及攻击

第七章 数字签名算法

RSA

- 没引入随机数：不能防重放

DSA

计算

- 给p，q，g，x，y，求签名（r, s）
    1. $r=(g^k\ mod\ p)\ mod\ q$
    2. 扩展欧几里得求 $k^{-1}\ mod\ q$
    3. $s=k^{-1}(H(M)+xr)\ mod\ q$
- 给（r，s），p，q，g，y，验证签名
    1. $w=s^{-1}\ mod\ 41$
    2. $u_1=H(M')w\ mod\ q$
    3. $u_2=rw\ mod\ q$
    4. $v=(g^{u_1}y^{u_2}\ mod\ p)\ mod\ q$
- 为什么不能泄露参数k?
    1. 已知（r，s）及M（公开）
    2. 获取了k
    3. 可由s的算式推出x的算式，从而计算私钥x
    4. 进而冒充用户伪造签名