密码学复习

2021年1月5日　22:38

第二章

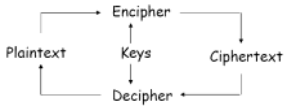
1.对网络安全的攻击主要有哪些?

唯密文攻击，已知明文攻击，选择明文攻击，选择密文攻击，选择文本攻击

2.网络安全服务的目标是什么?

3. 什么是密码？简单加密系统的模型?

密码：含有参数K的变换E



4. 什么是理论安全？什么是实际安全?

理论安全：攻击者无论截获多少密文，都无法得到足够的信息来唯一的决定明文。香农理论证明，理论安全需要加密密钥长度必须大于等于明文长度，密钥只用一次

实际安全（计算安全）：如果攻击者拥有无限资源，任何密码系统都是可以被破译的，但是在有限的资源范围内，攻击者都不能通过系统的分析方法来破解系统。

5. 什么是密码体制？有哪几类?

密码体制：加密系统采用的基本工作方式

分类：

1）对称密码体制（加密密钥和解密秘钥相同或其一可由另一导出）和非对称密码体制（加密密钥和解密秘钥不同，其一导出另一在计算上不可行）

2）序列密码体制（如果密文不仅与最初给定的算法和密钥有关，同时和明文位置有关）和分组密码体制（经过加密所得到的仅和算法、密钥有关）

3）确定型和概率型

4）单向函数型和双向变换型

6. 现代密码学的基本原则是什么?

设计加密系统，算法是可以公开的，需要保密的是密钥，安全性不在算法保密而在密钥。

7.加密系统应满足哪些具体要求?

1）系统是实际安全的

2）加密解密算法适用于密钥空间中的所有元素

3）系统易于实现，使用方便

4）满足现代密码学基本原则

5）不应使通信网络的效率过分降低

第三章

1.单表代换密码

2. 置换密码

3.DES算法详细

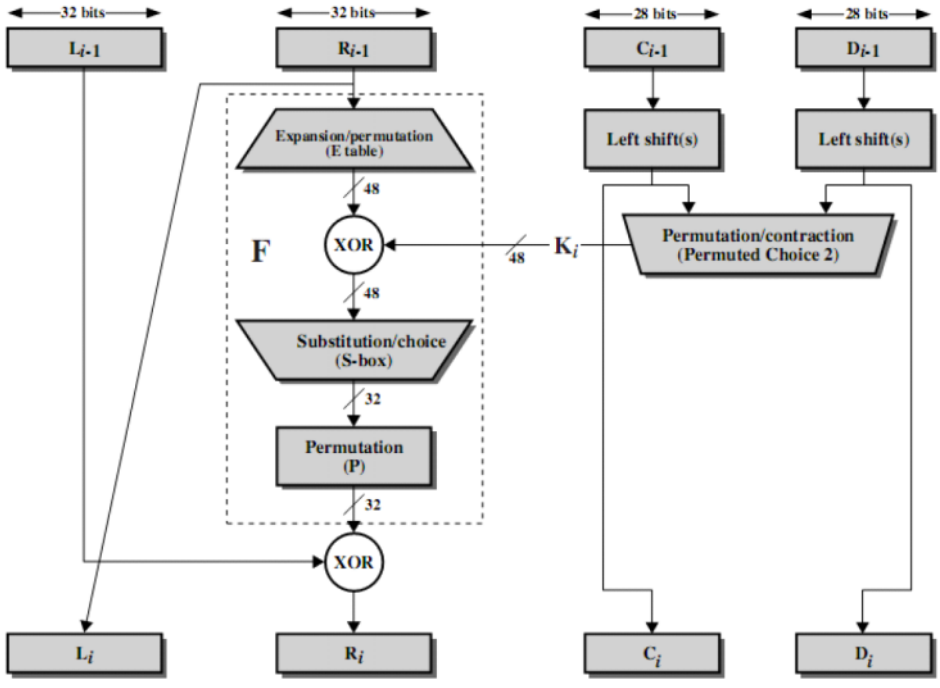


Figure 3.5　Single Round of DES Algorithm

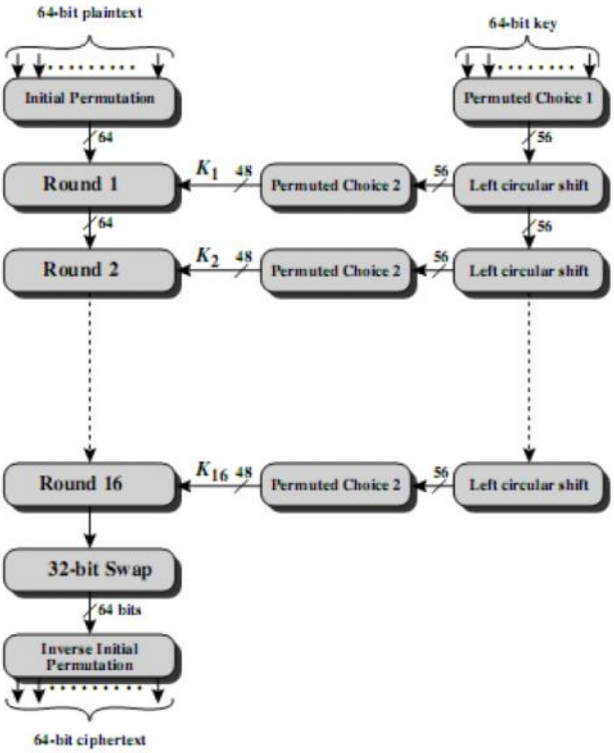
DES分组和密钥分别是64位和56位

1）流密码(Stream Cipher)和分组密码(Block Cipher)：

如果密文不仅与最初给定的算法和密钥有关，同时也与明文位置有关(是所处位置的函数)，则称为流密码体制。加密以明文比特为单位，以伪随机序列与明文序列模2加后，作为密文序列，一次一比特/字节

如果经过加密所得到的密文仅与给定的密码算法和密钥有关，与被处理的明文数据在整个明文中的位置无关，则称为分组密码体制。通常以大于等于64位的数据块为单位，加密得相同长度的密文。

2）算法流程一般描述：



(1) 首先进行一步初始置换和逆置换

(2) 将明文分成左右两部分

Li = Ri–1

Ri = Li–1 xor F(Ri–1, Ki)

1.使用置换表E，将32位右半部R扩展成48位

2.与48位子密钥做异或

3.48位结果送给8个替换盒S-boxes，得到32位结果

4.最后使用32位置换表P，把32位结果再进行一次置换处理

(3) S-box

有8个将6位数据映射成4位数据的S盒

6到4的映射规则是

外侧的第1位和第6位用作行选择

其余4位(2-5bit)用作列选择

这样每盒就有4行16列，输出4位，8个S盒输出32位

3)DES加密的雪崩效应

明文或密钥的一比特的变化，引起密文许多比特的改变。如果变化太小，就可能找到一种方法减小有待搜索的明文和密文空间的大小。

4)DES的安全强度

强力搜索(brute force search) 似乎很困难，20世纪70年代估计要1000 - 2000年，技术进步使穷举搜索成为可能

S-box问题

其设计标准没有公开，但是迄今没有发现S盒存在致命弱点

计时攻击

计时攻击利用的事实是加密或解密算法对于不同的输入所花的时间有细微的差别

DES能够很好地抵抗计时攻击

差分密码分析攻击问题

DES对差分分析攻击有较好的免疫力

差分密码分析攻击方法

关注一对明文在加密过程中通过轮函数的演变情况，而不是观测单个明文分组的演变。

4.AES高级加密标准

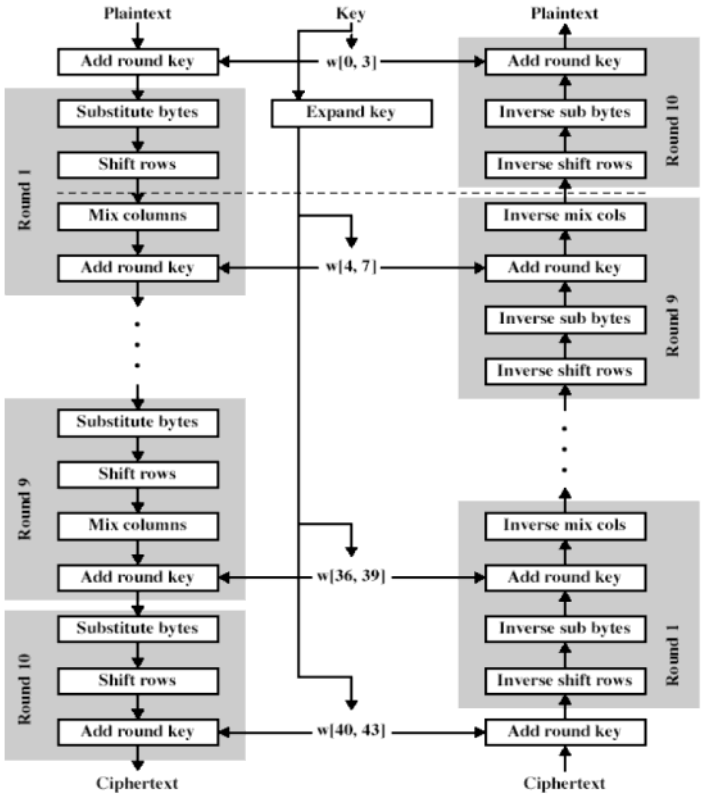
1）分组长度为128位，密钥长度为128位、192位或256位。

2）AES评估准则的三大类别

安全性：指密码分析方法分析一个算法所需的代价

成本：期望AES能够广泛应用于各种实际应用，计算效率要高

算法和执行特征：算法灵活性、适合于多种硬件和软件方式的实现、简洁性，便于分析安全性



3.一个混淆和三个代换

字节代换：用一个S盒完成分组中的按字节代换

行移位：一个简单的置换

列混淆：利用在域GF(28)上的算术特性的代换

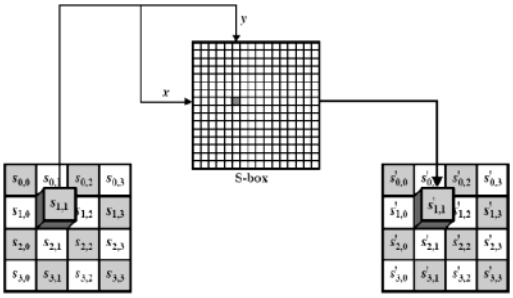
轮密钥加：利用当前分组和扩展密钥的一部分进行按位XOR

4.具体每步的操作

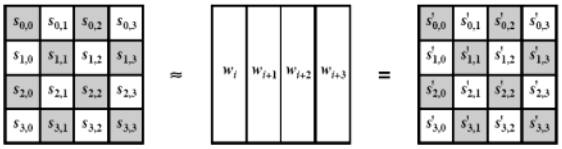
128位数据分成4组，每组4字节(state)

每个state执行9/11/13轮操作:

1.字节代换(每个字节使用一个S盒)：字节代替变换和轮密钥加变换

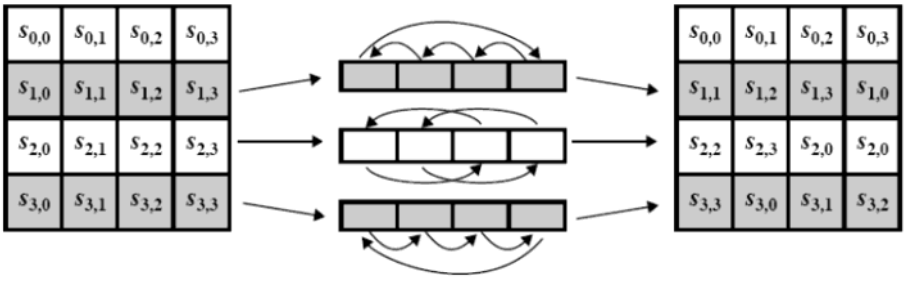


(a) Substitute byte transformation



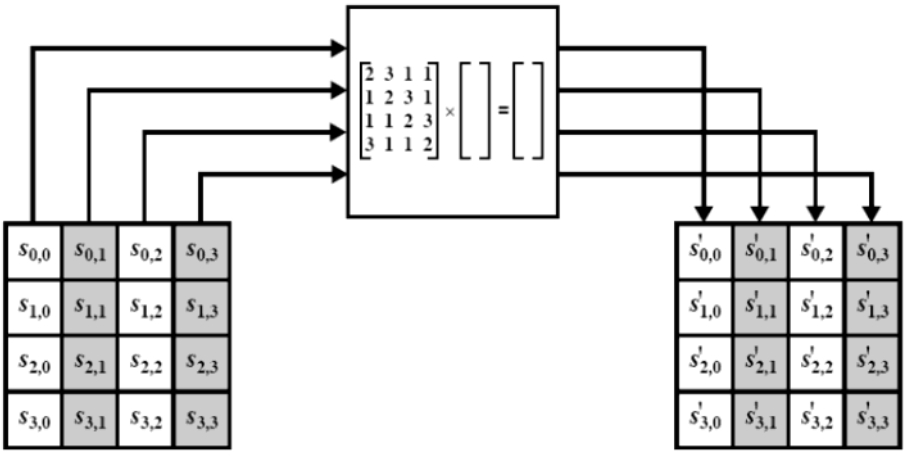
(b) Add Round Key Transformation

2.行移位变换(在组和列之间变换)：State的第1行保持不变，State的第2行循环左移1个字节，State的第3行循环左移2个字节,State的第4行循环左移3个字节



(a) Shift row transformation

3.列混淆变换(使用组的矩阵乘):正向列混淆变换对每列独立进行操作,每列中的每个字节被映射为一个新值，由该列中的4个字节通过函数变换得到.这个变换由以下基于State的矩阵乘法表示，使用素多项式m(x) =x8+x4+x3+x+1, GF(2^8),逆向列混淆变换也由矩阵乘法定义



(b) Mix column transformation

4.轮密钥加(用密钥异或state)：正向轮密钥加变换中128位的State按位与128位的秘密钥异或(XOR)
所有运算均可通过异或和查表来完成，因此非常快而且效率高。
密钥扩展

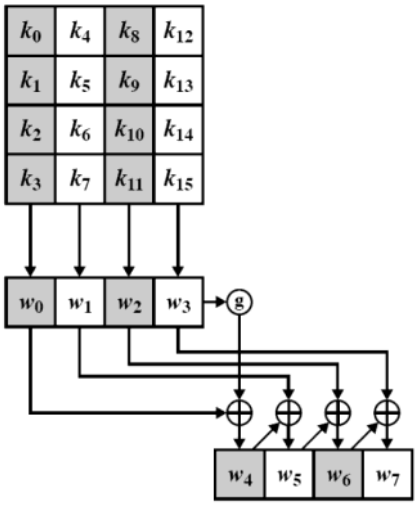
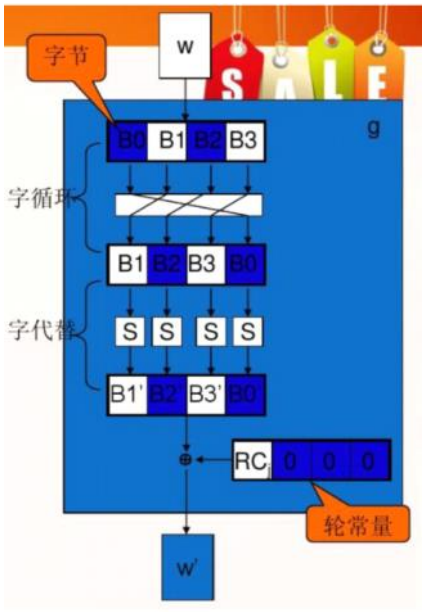


Figure 5.6 AES Key Expansion



5.AES的优点（可以在8位和32位的处理器上非常有效地实现）

字节代换是在字节级别上进行操作的，只要求一个256字节的表

行移位是简单的移字节操作

轮密钥加是按位异或操作

列混淆变换要求在域GF(28)上的乘法，所有的操作都是基于字节的，只要简单地查表即可

第六章 多重加密

1）工作模式：电码本模式（明文分成64位的分组进行加密，必要时填充，每个分组用同一密钥加密，同样明文分组加密得相同密文）

密文分组链接模式(加密输入是当前明文分组和前一密文分组的异或，形成一条链，使用相同的密钥， 这样每个明文分组的加密函数输入与明文分组之间不再有固定的关系)

密文反馈模式(加密函数的输入是一个64位的移位寄存器，产生初始向量IV。加密函数高端j位与明文P1的第一单元异或，产生j位密文C1进入移位寄存器低端，继续加密，与P2输入异或，如此重复直到所有明文单元都完成加密。)

输出反馈模式(结构上类似CFB，但是OFB中加密函数输出被反馈回移位寄存器，CFB中是密文单元被反馈回移位寄存器。优点是传输中的比特差错不会传播，缺点是比CFB更容易受报文流篡改攻击)

计数器模式(与OFB很像，但是加密的是计数器的值而不是任何反馈回来的值,每一个明文分组都必须使用一个不同的密钥和计数器值，决不要重复使用,Ci = Pi XOR Oi ,Oi = DESK1(i),可以用于高速网络加密中)

2）流密码：

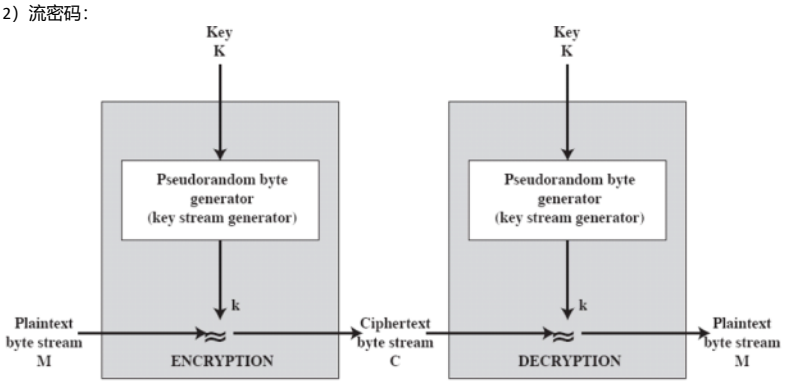


Figure 6.8 Stream Cipher Diagram

- 从数组S开始，S为{0..255}
- 使用密钥充分进行变换
- S形成密码的内在状态**internal state**
- 密钥k的长度为l字节

```
for i = 0 to 255 do
  S[i] = i
j = 0
for i = 0 to 255 do
  j = (j + S[i] + k[i mod l]) (mod 256)
  swap (S[i], S[j])
```

数论：

费马定理：若p 是素数，a 是正整数且不能被p 整除，则有a^p−1 ≡ 1(modp).

若p 是素数且a 是任意正整数，则a^p ≡ a mod p。

欧拉函数：定义欧拉函数φ(n) 为比n 小且与n 互素的正整数的个数。习惯上φ(1) = 1。

欧拉函数的性质：p 和q 是素数，n = pq，则φ(n) = φ(p)φ(q) = (p−1)(q−1)。

重要问题：公钥密码体制和传统密码体制的区别

传统密码的困难：

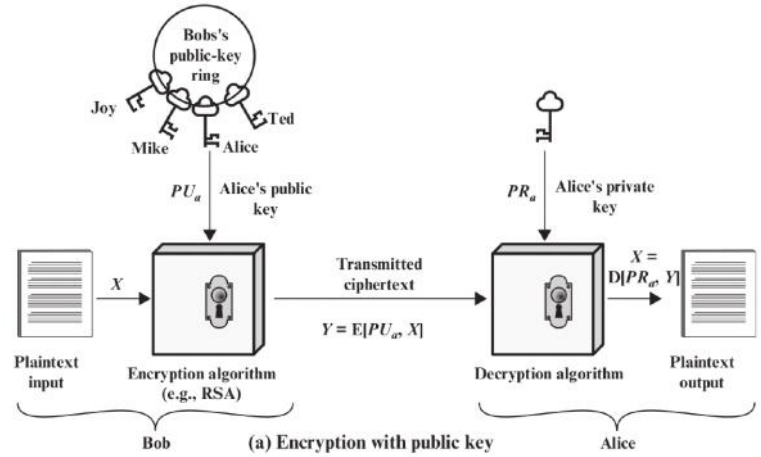
1、对称密码的加密解密是捆绑在一起的

2、对称密码更换，传递，交换需要可靠信道，秘钥分发比较困难

3、密钥数量多，管理困难

4、无法满足不相识的人之间通信的保密要求

5、不能实现数字签名



公钥密码体制基本特点：
1、加密解密能力分开
2、密钥分发简单，需要保存密钥量大大减少
3、可满足不相识的人之间保密通信
4、可以实现数字签名

公钥密码体制的分析
• 穷举攻击：公钥密码易受穷举攻击，解决方法是使用长密钥；同时为了便于实现加密和解密，又希望密钥足够短。目前公钥密码仅限于密钥管理和签名。
• 从给定的公钥计算出私钥：尚未在数学上证明对一特定公钥算法这种攻击是不可行的。因此包括RSA 在内的任何算法都是值得怀疑的。
• 穷举消息攻击：攻击者用公钥对所有可能的消息加密，并与传送的密文匹配，从而解密任何消息。抵抗的方法是在要发送的消息后附加随机数。

RSA算法流程
• 随机选择两个秘密大素数p 和q；
• 计算公开模数n = pq；
• 计算秘密的欧拉函数φ(n) = (p − 1)(q − 1)；
• 选择一个与φ(n) 互素的数，作为e 或d；
• 用扩展Euclid 算法计算模φ(n) 的乘法逆元素，即根据ed mod φ(n) = 1, 求d 或e；
• 加密：C = Me mod n，e 为公钥
• 解密：M = Cd mod n，d 为私钥

RSA计算实例：
• 选择p = 17, q = 11，则n = pq = 187，φ(n) = (p − 1)(q − 1) = 160；
• 选择e = 7 满足gcd(7, 160) = 1。因为23 × 7 = 161，所以d = 23；
• 公钥PU = 7, 私钥PR = 23；
• 明文M = 88；
• 加密计算C = 887 mod 187 = 11；
• 解密计算M = 1123 mod 187 = 88。

RSA选择密文攻击步骤
• 攻击者的目标是在不知道目标对象私钥的情况下解密密文C = Me mod n；
• 攻击者选择一个随机数r 并计算X = (Cre) mod n；
• 攻击者将X 发送给目标对象，欺骗目标对象对X 签名，得到Y = Xd mod n；
• 注意到X = (rM)e mod n，因此攻击者收到的Y = rM mod n，从而得到M = Y r−1 mod n，其中r−1 mod n 为r 模n 的乘法逆元。
• 为防止CCA 攻击，需要让RSA 在加密之前对明文进行随机填充，破坏RSA 的乘法同态性。

几种公钥分配方案：
• 公开发布
• 公开可访问目录
• 公钥授权
• 公钥证书

Diffie-Hellman 密钥交换协议

- 通信双方约定一个大素数 p, 和模 p 的一个素根 α;
- 各方产生公开密钥:
  - 选择一个秘密钥（数值），如 $x_A < p$，$x_B < p$;
  - 计算公钥 $y_A = \alpha^{x_A} \bmod p$, $y_B = \alpha^{x_B} \bmod p$, 并相互交换。
- 双方共享的会话密钥 K 可以如下算出:

用户 A	用户 B
$K = y_B^{x_A} \bmod p$ $= \alpha^{x_B x_A} \bmod p$	$K = y_A^{x_B} \bmod p$ $= \alpha^{x_A x_B} \bmod p$

- K 是双方用对称密码通信时共享的密钥;
- 攻击者如果想要获得 K, 则必须解决 DLP 问题。

ECC Diffie-Hellman 密钥交换

- 类似于 D-H，ECC 也可以实现密钥交换
- 用户选择合适的 ECC, $E_p(a,b)$
- 选择基点 $G = (x_1, y_1)$, 满足 $nG = O$ 的最小 n 是一个大整数
- A 和 B 之间的密钥交换如下
  - A 和 B 选择私钥 $n_A < n$，$n_B < n$;
  - 计算公钥 $P_A = n_A G$，$P_B = n_B G$;
  - A 与 B 交换 P_A 和 P_B;
  - 计算共享密钥 $K = n_A P_B = n_B P_A$，因为 $K = n_A n_B G$，所以这两个密钥是一样的。

椭圆曲线加密

- 首先将明文消息 m 编码为 (x, y) 的点 P_m，点 P_m 就是要进行加密的点。注意不能简单地把消息编码为点的 x 坐标或 y 坐标，因为并不是所有的坐标都在 $E_p(a,b)$ 中。
- 类似于 D-H 密钥交换，加解密系统也需要点 G 和椭圆群 $E_p(a,b)$ 这些参数。
- 用户 A 选择私钥 $n_A < n$，并产生公钥 $P_A = n_A G$;
- 用户 B 选择私钥 $n_B < n$，并产生公钥 $P_B = n_B G$;
- A 要给 B 发送 P_m;
- A 随机选择一个正整数 k，加密点 P_m 产生密文: $C = \{kG, P_m + kP_B\}$;
- B 解密 C，计算: $P_m + kP_B - n_B(kG) = P_m + k(n_B G) - n_B(kG) = P_m$。

公钥加密作为认证手段

• 若要提供认证，发送方用自己的私钥对消息加密，接收方用发送方的公钥解密(验证)，就提供了认证功能。

• 如果发送方用私钥加密消息，再用接收方的公钥加密，就实现了既保密又认证的通信。

• 既保密又 两次。

消息认证码MAC

• 又称密码校验和或MAC：利用密钥K 生成消息M 的一个定长短数据块MAC = C(K,M)，并将MAC 附加在消息后。

• 接收方对收到的消息重新计算MAC，并与接收到的MAC比较。因为只有双方知道密钥，如果两个MAC 匹配，则:

• 接收方可以确信报文未被更改;

• 接收方可以确信报文来自声称的发送者;

• 接收方可以确信报文序号正确，如果有的话。

• MAC 函数类似加密，但非数字签名，也无需可逆;

• 将MAC 直接与明文并置，然后加密传输比较常用。

消息认证码的特点

• MAC 是一种多对一函数，类似于加密函数，但不要求可逆。
• 定义域由任意长的消息组成，值域由所有可能的MAC 组成。
• 若使用n 位长的MAC，则有2n 个可能的MAC。有N 条可能的消息，$N \gg 2n$。若密钥长度为k，则有2k 种可能的密钥，即2k 种可能的映射关系。
• 如N 为2100，n 为10，共有210 种不同的MAC，平均每一个MAC 可由2100/210 = 290 条不同的消息产生。若密钥长度为5，则从消息集合到MAC 值的集合有25 = 32 种不同映射。
• 与加密相比，认证函数更不荣易被攻破。

对消息认证码的要求

• 若攻击者已知M 和C(K,M)，则构造满足C(K,M′) = C(K,M) 的消息M′ 在计算上是不可行的；
• C(K,M) 应该是均匀分布的，即对任何随机选择的消息M 和M′，C(K,M′) = C(K,M) 的概率是2^{-n}, 其中n 是MAC 的位数；
• 设M′ 是M 的某个已知的变换，即M′ = f(M)，如f 可能表示逆转M 的一位或多位，那么

$$Pr[C(K,M) = C(K,M')] = 2^{-n}$$

直观理解：只要两消息不相同，其MAC 值相同的概率都只是2^{-n}。

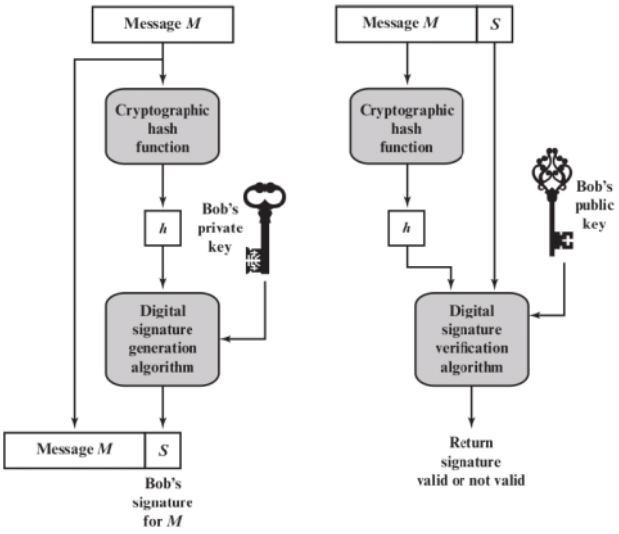
MAC 的安全性

• 攻击者如何用穷举的方法找到密钥?
• 假设攻击者可以访问明文及其MAC；
• k 为密钥长度，n 为MAC 长度，k > n；
• 对满足T1 = C(K,M1) 的M1 和T1，攻击者需要穷举所有可能的密钥值Ki，计算Ti = C(Ki,M1)，那么至少会找到一个密钥使得Ti = T1。
• 这里一共产生2k 个MAC，只有2n 个不同MAC；
• 许多密钥会产生相同MAC，而攻击者不知道哪个是正确密钥；
• 平均来说，有$2k/2n = 2^{k-n}$ 个密钥会产生正确MAC，因此攻击者对这些密钥必须做重复攻击；
• 攻击者需要用另一个（消息MAC）对：T2 = C(K,M2)，对找到的2^{k-n} 个密钥计算Ti = C(Ki,M2)，得到2^{k-2n} 个密钥。
• 若$k = \alpha n$，则需α 次循环，才能找到正确密钥。

密码学Hash 函数的安全性要求

1. H 可以应用于任意大小的数据块；
2. H 产生固定长度的输出；
3. 对任意给定的明文x，计算H(x) 容易；
4. 单向性：对任意给定的哈希码h，找到满足H(x) = h 的x，在计算上不可行（即由哈希码找消息不可行）；
5. 抗弱碰撞性：对任何给定的消息x，找到满足$y \neq x$ 且H(x) = H(y) 的消息y，在计算上不可行（即给定一个消息，找另外一个消息，使它们有相同的哈希码不可行）；
6. 抗强碰撞性：找到任何满足H(x) = H(y) 的消息对(x, y)，在计算上不可行。

数字签名的一般模型



(a) Bob signs a message

(b) Alice verifies the signature