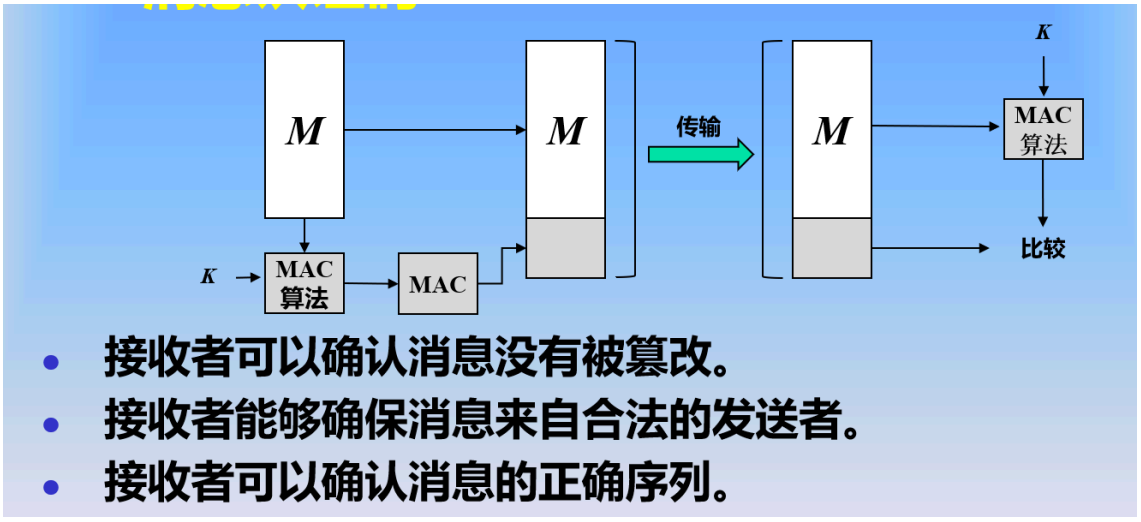
消息认证码

消息认证方法

- 加密可以防止被动攻击（窃听），而消息认证方法可以防止主动攻击（伪造数据和业务）。
- 作用
  - 验证消息的内容有没有被篡改和验证来源是否可信
  - 验证消息的时效性以及两实体之间消息流的相对顺序
- 方法：
  - 基于常规加密的消息认证
  - 非加密的消息认证：消息认证与消息加密是两个独立的功能，存在无须保密的消息认证情况

消息认证码

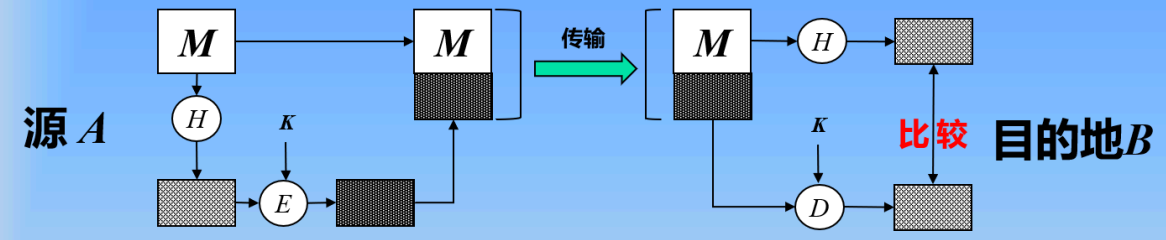
- 定义：消息认证码是一种认证技术，利用私钥产生一小块数据，并将其附在消息上。
  - 通信双方（ A和B ）共享公共密钥KAB。
  - 当某一方（A和B ）有消息M要发送时，则计算消息认证码（MACM=F(KAB, M) ），消息M连同MAC一起发送。
  - 接收方同样计算消息认证码（MACM=F(KAB, M) ），并与收到的消息认证码进行比较。



- 安全分析：解密算法需要可逆，然而认证算法并不需要可逆。因此，由于认证函数的数学特性，与加密相比更难攻破

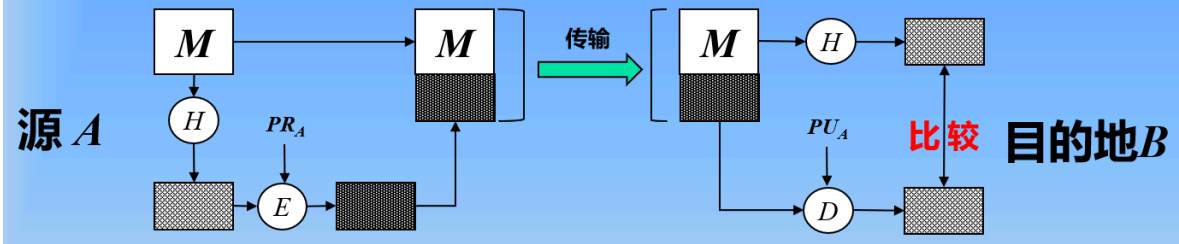
采用Hash的MAC

3.2 传统密码加密Hash方法



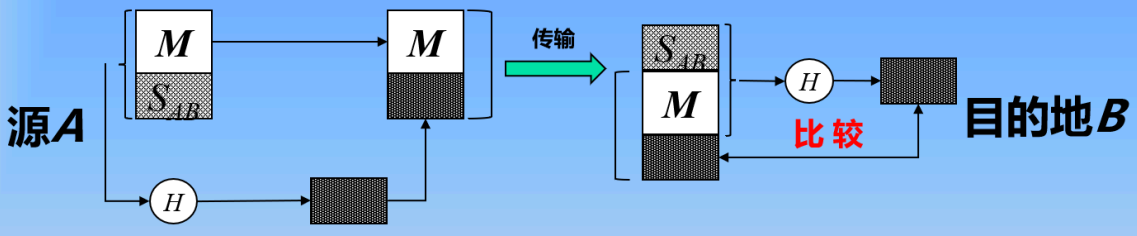
- 发送者和接收者共享同一密钥，可以保证私密性，实现消息认证功能。
- 只需要加密消息摘要，效率更快。

3.2 公钥密码加密Hash方法



- 发送者用私钥进行加密，只有自己的公钥可以解密，提供数字签名和消息认证功能。
- 不需要在通讯各方之间发分密钥。

3.2 没有使用加密的Hash方法



- 发送者A和接收者B共享秘密值S_{AB}。A计算S_{AB}与消息的散列函数值发送给B，由B计算散列值进行比较。
- 实际没有发送S_{AB}，攻击者无法得到S_{AB}并篡改消息。

HMAC（重点）

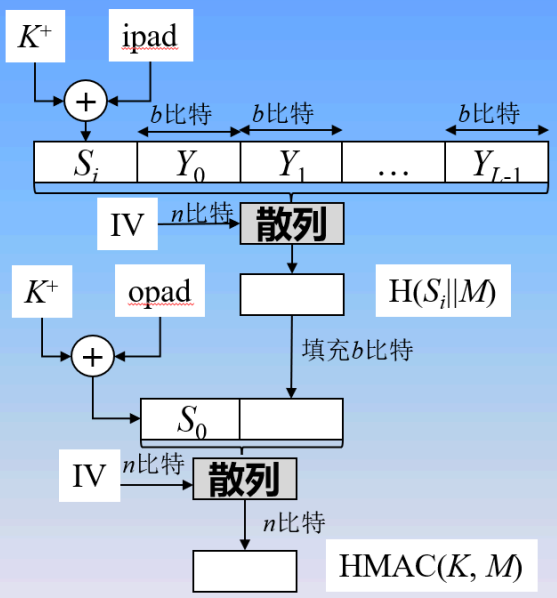
HMAC算法可以用下面式子表示：

$$HMAC(K, M) = H[(K^+ \oplus opad) \| H[(K^+ \oplus ipad) \| M)]]$$

其中K^+是为使K为b位长而在K左边填充0后得到的结果，opad(01011100), ipad(00110110)是指定的填充常量。H是嵌入的散列函数，例如： MD5, SHA-2等。

4.2 HMAC算法

HMAC总体流程图：



HMAC算法描述：

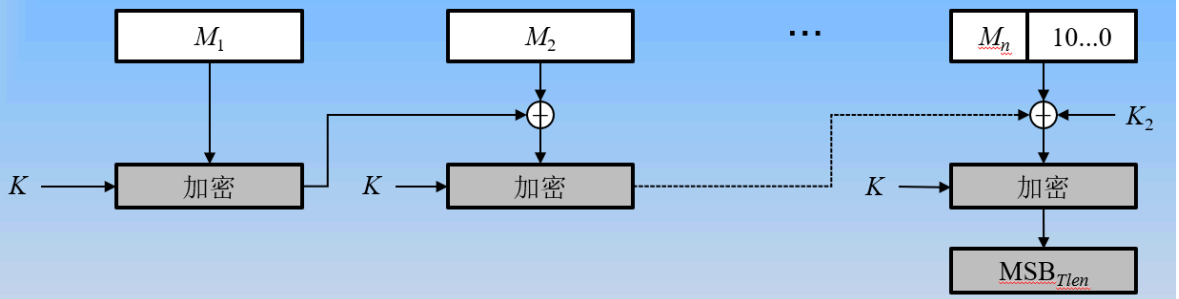
（1）在K的左端追加0构成b比特的字符串K^+。
（2）ipad与K^+进行异或操作生成b比特的分组S_i。
（3）将消息M追加在S_i上。
（4）将H应用于步骤（3）所产生的数据流。
（5）opad与K^+进行异或操作生成b比特的分组S_o。
（6）将步骤（4）产生的散列结果追加在S_o上。
（7）将H应用于步骤（6）所产生的数据流。

CMAC

- 根据消息长度分情况处理，使用2个密钥和数据填充
  - 当消息长度是密码块b的整数倍数n时，采用k比特的加密密钥K和n比特的密钥K_1。
  - 当消息长度不是密码块b的整数倍时，对最后一块数据进行填充（一位1和若干0组成），然后采用k比特的加密密钥K和n比特的密钥K_2。

- 根据消息长度分情况处理，使用2个密钥和数据填充
  - 当消息长度是密码块b的整数倍数n时，采用k比特的加密密钥K和n比特的密钥K_1。
  - 当消息长度不是密码块b的整数倍时，对最后一块数据进行填充（一位1和若干0组成），然后采用k比特的加密密钥K和n比特的密钥K_2。

- 当消息长度不是密码块b的整数倍数



CCM