4. 报文完整性

4.1 数字签名

数字签名类比于手写签名

发送方 (Bob) 数字签署了文件，前提是他(她)是文件的拥有者/创建者.

可验证性，不可伪造性，不可抵赖性

谁签署：接收方 (Alice)可以向他人证明 是 Bob，而不是其他人签署了这个文件 (包括Alice)

签署了什么：这份文件，而不是其它文件

可验证性：

给Bob一个m，Bob用私钥对m进行加密，Alice用Bob的公钥解密，确认是Bob

不可伪造性：

别人没有私钥

不可抵赖性：

同可验证性，（只能说明私钥是同一个）

- 假设Alice收到报文m, 以及数字签名$K_B^-(m)$
- Alice 使用Bob的公钥K_B^+对$K_B^-(m)$进行验证， 判断$K_B^+(K_B^-(m)) = m$是否成立.
- 如 $K_B^+(K_B^-(m)) = m$成立, 那么签署这个文件的人一定拥有Bob的私钥.

Alice 可以验证:

- ✓ Bob 签署了m.
- ✓ 不是其他人签署了m.
- ✓ Bob签署了m 而不是m'.

不可抵赖性:

- ✓ Alice可以拿着m,以及数字签名$K_B^-(m)$到法庭上，来证明是Bob签署了这个文件 m.

4.2 报文摘要

对长报文进行公开密钥加密算法的实施需要耗费大量的时间

Goal: 固定长度，容易计算的“fingerprint”

对m使用散列函数H，获得固定长度的 报文摘要H(m).

散列函数的特性：

多对1

结果固定长度

给定一个报文摘要x，反向计算出原报文在计算上是不可行的x = H(m)

Internet校验和：弱的散列函数

产生报文m的固定长度的摘要 (16-bit sum)

多对1的
但是给定一个散列值，很容易计算出另外一个报文具有同样的散列值:见图

散列函数算法:
MD5散列函数(RFC 1321)被广泛地应用
4个步骤计算出128-bit的报文摘要
给定一个任意的128-bit串x，很难构造出一个报文m具有相同的摘要x.
SHA-1也被使用.
US标准 [NIST, FIPS PUB 180-1]
160-bit报文摘要

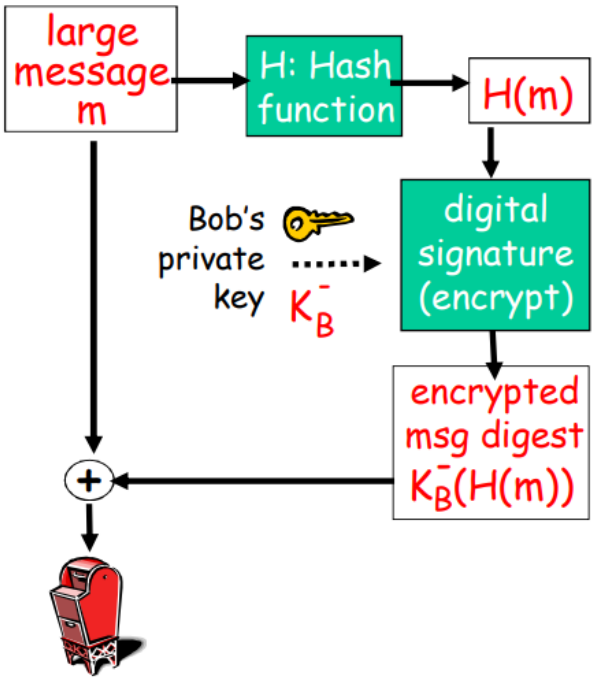
message	ASCII format
I O U 1	49 4F 55 31
0 0 . 9	30 30 2E 39
9 B O B	39 42 D2 42
	B2 C1 D2 AC

message	ASCII format
I O U 9	49 4F 55 39
0 0 . 1	30 30 2E 31
9 B O B	39 42 D2 42
	B2 C1 D2 AC

不同的报文
但是相同的校验和!

数字签名 = 对报文摘要进行数字签署

Bob 发送数字签名的报文:



Alice校验签名和报文完整性: