Hash 函数的性质
Hash 函数实现的基本过程
消息认证实现的基本过程
数字签名实现的基本过程
数字证书包含内容及安全性
消息认证码和数字签名的对比

Hash 的性质：hash 是一种单向密码体制
总结：单向特征，输出数据长度固定特征
1. 输入的消息是任意长度的
2. 输出的哈希值是固定长度的
3. 容易计算哈希值
4. 单向性，是从明文到密文的不可逆映射
5. 抗弱碰撞性：有 M1,H（M1）难找到第二个 M2 和 M1 的哈希值相等的
6. 抗强碰撞性：任意一对 M1M1 想使哈希值相等不可行
7. 雪崩效应

Hash 实现的基本过程

常见的 hash 函数：MD 算法家族，SHA 算法家族
MD5 算法输出哈希值长度是 128 位，主要用于确保信息完整性
目前计算能力，安全哈希值长度为 160 位

哈希函数的应用:
消息认证，数字签名，口令的安全性，文件的完整性，密码协议的应用

对称密码体制和公钥密码体制都可以提供消息认证服务，但用于消息认证的最常见密码技术是基于哈希函数的消息认证码

消息认证实现的基本过程:
消息认证码 MAC 是与密钥相关的单向哈希函数，不同密钥会产生不同的散列函数，因此功能：验证消息没有经过篡改+知道发送者是谁（原本约定好的都有 K 的两个人）
但 MAC 的生成一方与检测一方持有相同密钥，所以不能确定消息是被谁生成的，如果有人捏造的信息他可以否认是对方捏造的，不足，用数字签名可以解决
发送方：消息+对称密钥 K ，经过哈希函数生成哈希值作为消息认证码 MAC 后，消息+MAC 经过公开信道被接受，接收者根据消息+对称密钥再经哈希函数生成哈希值，与 MAC 进行对比，若相等则认证有效，若不相等则认证无效（消息被篡改或者发送方不正确）

数字签名
数字签名的复制是有效的，而手写签名的复制品无效，数字签名不仅与签名者有关，也因消息而异

数字签名实现的基本过程：
签名者 A 将消息经 hash 函数转化为哈希值，再将 A 的私钥和哈希值经过签名算法转化成签名值附在消息后面经过公开信道，验证者拿消息算得的哈希值和 A 的公钥经过验证算法判断签名是否有效。RSA 数字签名方案

“公开密钥加密的加密和解密都比较耗时，为了节约运算时间，实际上不会对消息直接进行加密，而是求得消息的哈希值，再对哈希值进行加密，然后将其作为签名来使用。”

不足：X 可能冒充 A，把 A 的公钥换成自己的公钥，因为无法确定公开密钥的制作者是谁

公钥密码管理：公钥的私密性不用确保，真实性、完整性必须严格保护
私钥的私密性、真实性、完整性都必须保护

数字证书的内容：版本号、序列号、认证机构标识（发证机构的名称，一般是 CA）、主体标识、主体公钥、证书有效期、证书用途、扩展内同、发证机构签名（用发证机构的私钥生成的数字签名

数字证书的安全性：
1. 证书以文件形式存在，公开可复制
2. 任何具有 CA 公钥的用户都可以验证证书的有效性
3. 除了 CA 外，任何人无法伪造、修改证书
4. 证书的安全性依赖于 CA 的私钥

·消息认证码和数字签名的对比

消息认证码和数字签名的对比

	消息认证码	数字签名
发送者	用对称密钥计算MAC	用私钥生成签名
接受者	用对称密钥计算MAC	用公钥验证签名
密钥分发问题	存在	不存在，但公钥需要另外认证
效率	高	低
完整性	支持	支持
认证性	支持(仅限通信双方)	支持(可适用于任何第三方)
不可否认性	不支持	支持