数字内容安全

曹 刚

gangcao@cuc.edu.cn

课程内容

第1章 概述

第2章 消息认证与数字签名

第3章 数字指纹/图像Hash

第4章 信息隐藏

第5章 数字取证

第2章 消息认证与数字签名

2.0 密码学基础知识

2.1 消息认证概念

2.2 消息认证技术（原理与方法）

2.3 数字签名概念

2.4 数字签名技术（原理与方法）

主要内容

- 消息认证
  - 功能：完整性验证、消息来源验证
  - 消息认证码 MAC=$C_K(M)$ (要求带密钥的加密)
  - Hash：MD5 、SHA
  - 基于带密钥Hash的消息认证
  - 基于Hash后再加密的消息认证
  - 带保密功能的消息认证
- 数字签名
  - 功能：身份认证（消息来源验证？）、完整性验证
  - 签名算法、验证算法
  - RSA、DSS、DSA (Hash+公钥加密体制)
  - 数字证书：公钥证书（公钥分发）

第2章 消息认证与数字签名

2.0 密码学基础知识

2.1 消息认证概念

2.2 消息认证技术（原理与方法）

2.3 数字签名概念

2.4 数字签名技术（原理与方法）

密码学分类

- 按操作分类
  - 代换密码（替换密码）
  - 置换密码
- 按使用密钥数量分类
  - 对称密码（单钥密码）
    - 序列密码（流密码）
    - 分组密码
  - 公钥密码（双钥密码，非对称密码）

典型对称密码算法

- 序列密码（流密码）：RC4，……
- 分组密码：DES，IDEA，AES

算法	密钥长度(bit)	明文组长度(bit)	密文组长度(bit)	加密变换轮数	加密步骤
DES	56	64	64	16	① IP ② 16轮乘积变换 ③ IP^{-1}
IDEA	128	64	64	8	① 8轮迭代 ② 输出变换
AES	128/192/256	128/192/256	128/192/256	Nr	① 初始密钥加（异或） ② Nr轮变换（字节代换，行移变换，列变换，轮密钥加）

典型公钥密码算法

	RSA	ElGamal	ECC
数论基础	欧拉定理	离散对数	离散对数
安全性基础	大整数的素因子分解的困难性	有限域上离散对数问题的难解性	椭圆曲线离散对数问题的难解性
当前安全密钥长度	**1024**位	**1024**位	**160**位
用途	加密、数字签名	加密、数字签名	加密、数字签名
是否申请专利	是	否	否

两种密码体制的优缺点

- 对称密码技术
  - 优点：效率高，算法简单，系统开销小，适合加 密大量数据
  - 缺点：进行安全通信前需要以安全方式进行密钥交换，且规模复杂
- 公钥密码技术
  - 优点：密钥管理方便，密钥尺寸大
  - 缺点：加解密速度慢，发展历史较短

第2章 消息认证与数字签名

2.0 密码学基础知识

2.1 消息认证概念

2.2 消息认证技术（原理与方法）

2.3 数字签名概念

2.4 数字签名技术（原理与方法）

回顾

信息安全的基本属性

- 保密性（Confidentiality）
- 完整性（Integrity）
- 可用性（Availability）
- 不可抵赖性（Non-Repudiation）
- 可控性（Controllability）

消息认证

➢ **需要验证：**

- 信息发送方是否真实？接收方是否真实？
- 信息在传输过程中是否被改变？
- 信息的到达时间是否在指定的期限内？

消息认证概念

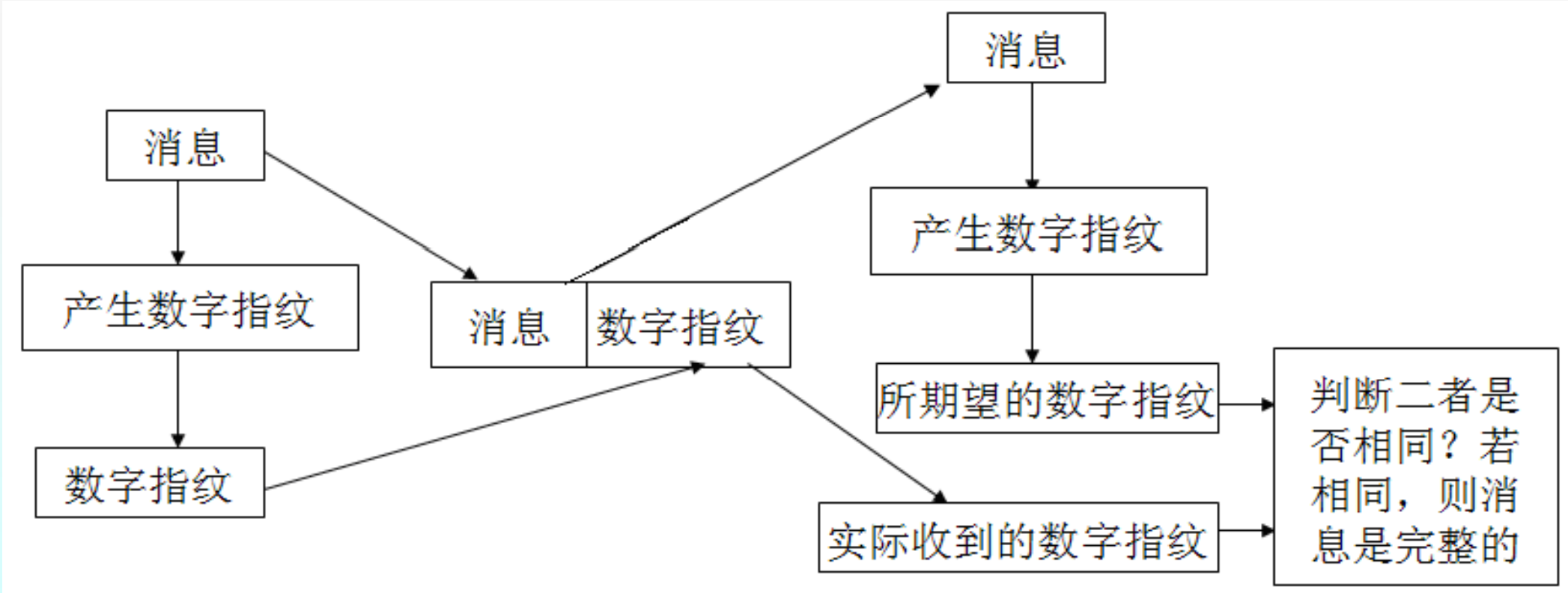
- 消息认证
  - 指通过对消息（或与消息有关的信息）进行加密或签名变换而实现的认证；
  - 目的是为了防止传输和存储的消息被篡改，包括
    - 消息内容认证（即消息完整性认证）
    - 消息源宿认证（即身份认证）
    - 消息的序号和操作时间认证等

消息认证的两层含义

- 消息认证具有两层含义
  - 1) 检验消息的完整性，即验证消息在传送或存储过程中未被修改（如篡改、删除或插入等）；
  - 2) 检验消息来源的真实性，即对消息的发送者的身份进行认证（即身份认证）；

消息完整性检验

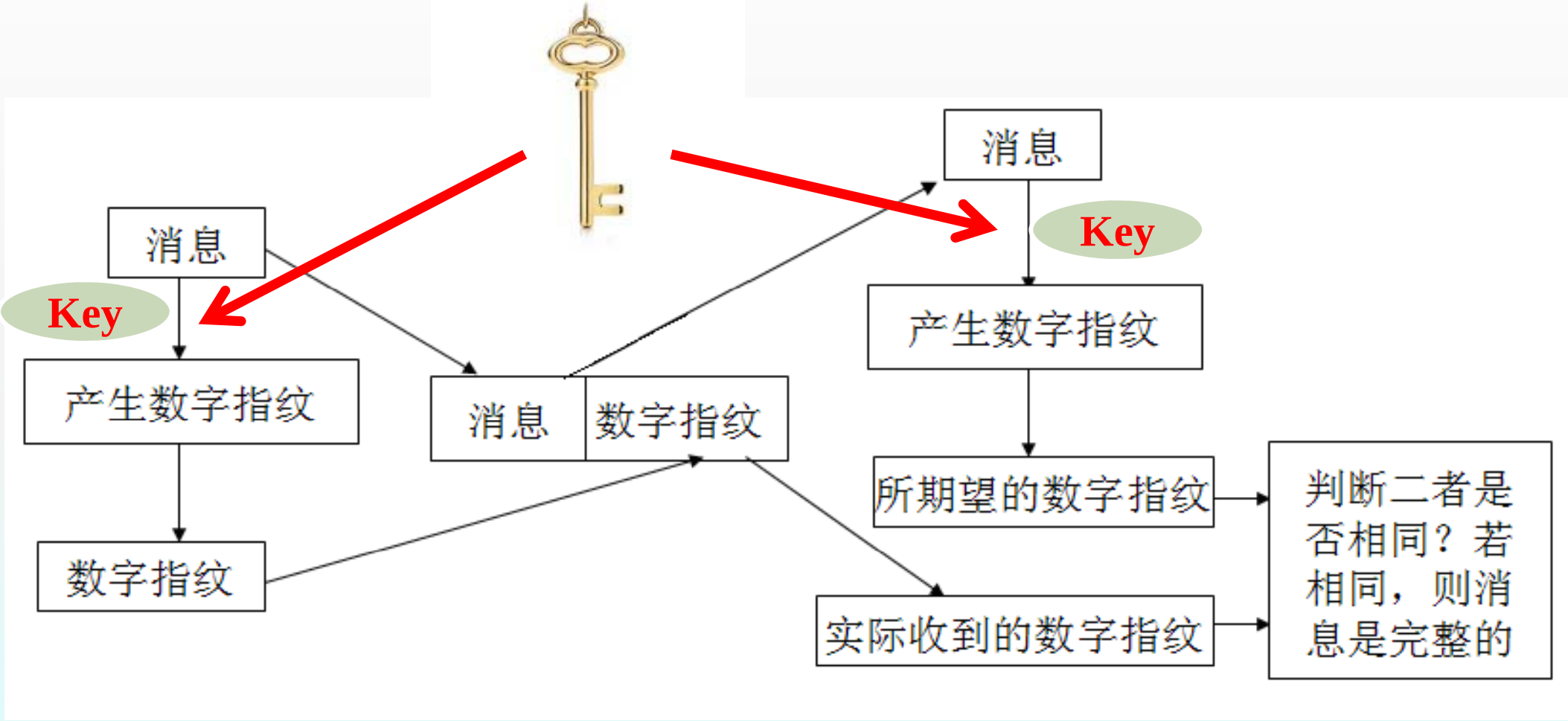
- 消息完整性检验的一般机制：无论是存储文件还是传输文件，都需要同时存储或发送该文件的数字指纹；验证时，对于实际得到的文件重新产生其数字指纹，再与原数字指纹对比；如果一致，则说明文件是完整的。否则，是不完整的。



消息完整性检验

- 一般方法：传统的单向Hash函数
  - **MD5**
    - **1990年10月，Ron Rivest，MD4**
    - **1992年4月，改进的MD4称为MD5**
    - **输入分组为512比特，输出的摘要为128比特**
  - **SHA**
    - **1993年，NIST，SHA0**
    - **1995年修订，称之为SHA-1**
    - **输入分组为512比特，输出的摘要为160比特**
    - **2002年，NIST发布修正版，增加SHA-256, SHA-384, SHA-512**

消息身份认证



- 当需要进行消息认证时，需将消息和密钥K同时作为输入，这就是消息认证的原理。能否认证，关键在于信息发送者或信息提供者是否拥有密钥K。

第2章 消息认证与数字签名

2.0 密码学基础知识

2.1 消息认证概念

2.2 消息认证技术（原理与方法）

2.3 数字签名概念

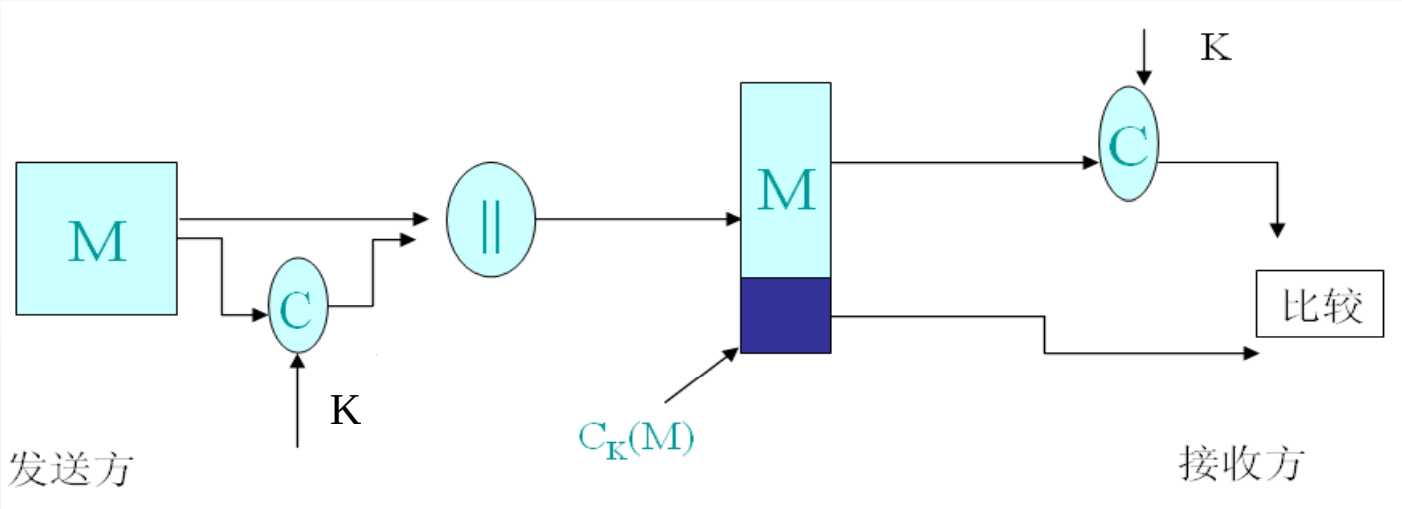
2.4 数字签名技术（原理与方法）

消息认证码

- 消息认证码(Message Authentication Code，MAC)是一种鉴别函数，其实现鉴别的原理是：用公开函数和密钥产生一个固定长度的值作为认证标识，用这个标识鉴别消息的完整性。接收方利用共享密钥进行身份认证。
- 消息认证码通常表示为 $MAC=C_K(M)$
  - 其中M是可变长的消息，K是收发双方共享的密钥，函数值$C_K(M)$是定长的认证码，也称为密码校验和。
- MAC就是带密钥的消息摘要函数，其实就是一种带密钥的数字指纹，它与不带密钥的数字指纹是有本质区别的。

消息认证

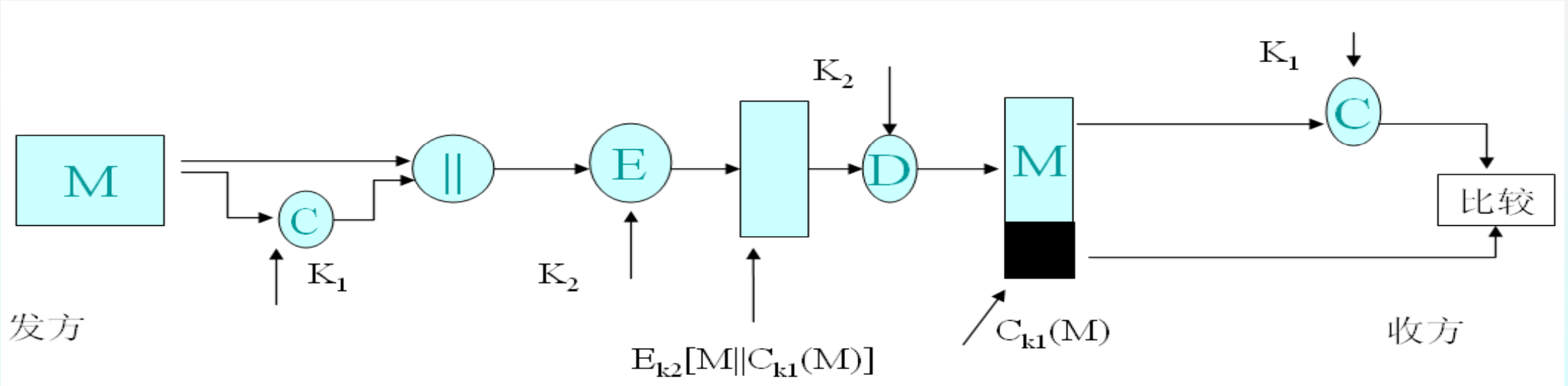
- 认证码被附加到消息后以M||MAC方式一并发送，收方通过重新计算MAC以实现对M的认证。



- 假定收发双方共享密钥K，如果收到的MAC与计算得出的MAC一致，那么可以得出如下结论：
  - ① 接收方确信消息M未被篡改。此为完整性验证。
  - ② 接收方确信消息来自所声称的发送者，因为没有其他人知道这个共享密钥，其他人也就不可能为消息M附加合适的MAC。此为消息源验证。

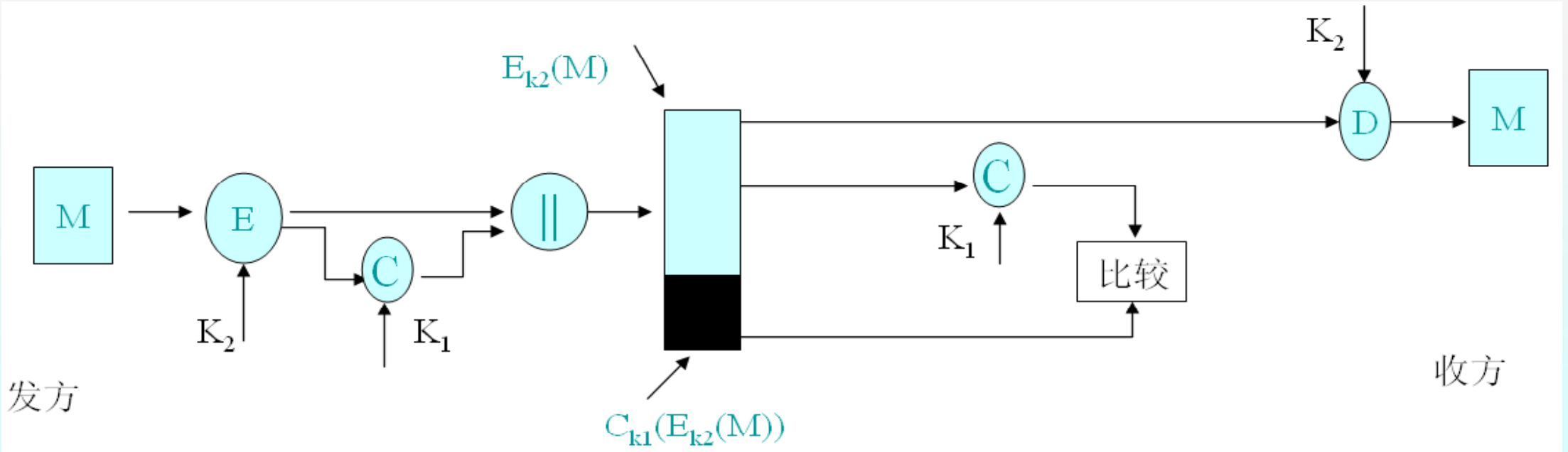
消息认证与保密

- 在**MAC**认证中，如果消息以明文方式传送，就只提供认证而不具备保密性。下图所示提供一种既加密又认证的方式，发送方发送$E_{k2}[M||C_{k1}(M)]$。该种处理方式除具备消息认证功能外，还具有保密性。



消息认证与保密

- 改变加密的位置，得到另外一种消息保密与认证方式。该种处理方式先对消息进行加密，然后再对密文计算MAC，传送$E_{k2}(M)||C_{k1}(E_{k2}(M))$给接收方。接收方先对收到的密文进行认证，认证成功后再解密。



MAC的构造方法

基于带密钥的Hash函数的MAC

- 要求所使用的Hash函数具有迭代结构（如MD5/SHA-1等）迭代结构是指反复地使用压缩函数 *f* 将长消息映射为短消息
- *f*有两个输入，长度为*l*的链变量*k*和长度为*b*的数据块*x*，*fk*=*f*(*k*,*x*)。 在MD5中*b*=512，*l*=128。Hash函数*F*(*x*)。IV是初始向量，长度为*l*
- 如使用密钥K作为初始向量，Hash函数就成为了带密钥的Hash函数
- 被证明是安全的

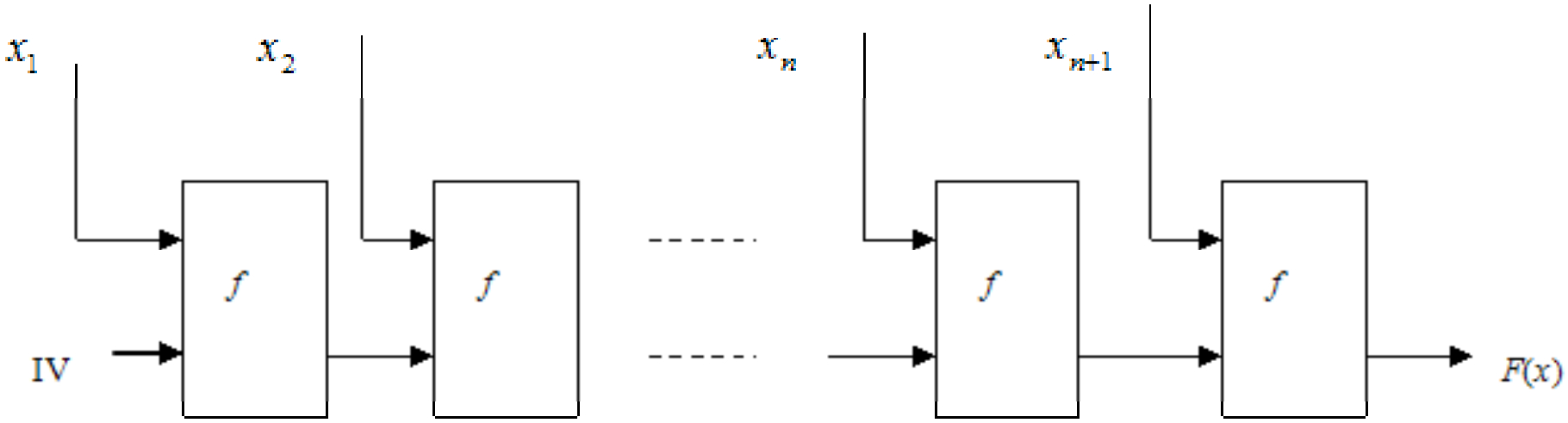


图 3-10 具有迭代结构的单向 Hash 函数

MAC的构造方法

- 基于带密钥的Hash函数的MAC
- 优点
  - 在软件实现上，比使用分组密码构造的MAC快，效率更高
  - 基于带密钥的Hash函数的构造方法将MAC的安全性归结到所使用的Hash函数上
- 不足
  - 所使用的Hash 函数有些没有安全性证明， 不能保证其安全性
  - 由于压缩函数是串行的，构造方法不支持并行

MAC的构造方法

- 基于分组密码的**MAC**
  - **CBC-MAC**、
  - **XOR-MAC**、
  - **PMAC**、
  - **XECB-MAC**
  - **OCB**

第2章 消息认证与数字签名

2.0 密码学基础知识

2.1 消息认证概念

2.2 消息认证技术（原理与方法）

2.3 数字签名概念

2.4 数字签名技术（原理与方法）

问题的提出

- 假定A向B发送一个带数字指纹的报文，可能会出现如下的争执：
  - ✓ B可能伪造不同的报文，并声称它来自A。B只要简单地生成一个报文，并附加使用由A和B所共享的密钥生成的认证码即可。

  - ✓ A可以否认发送过该报文。因为B伪造一个报文是可能的，无法证明A发送过该报文这一事实。

数字签名概念

- 数字签名（**Digital Signature**），又称公钥数字签名、电子签章，是一种使用了公钥加密技术，用于鉴别数字信息的方法。一套数字签名通常定义为两种互补的运算，一个用于签名，另一个用于验证。
- 通常的方法是数据单元上附加一些数据，或是对数据单元进行密码变换，使得接收者能够确认数据单元的来源和数据单元的完整性并保护数据，防止被人进行伪造。
- 基于公钥密码体制和私钥密码体制都可以获得数字签名，目前主要是基于公钥密码体制的数字签名。

第2章 消息认证与数字签名

2.0 密码学基础知识

2.1 消息认证概念

2.2 消息认证技术（原理与方法）

2.3 数字签名概念

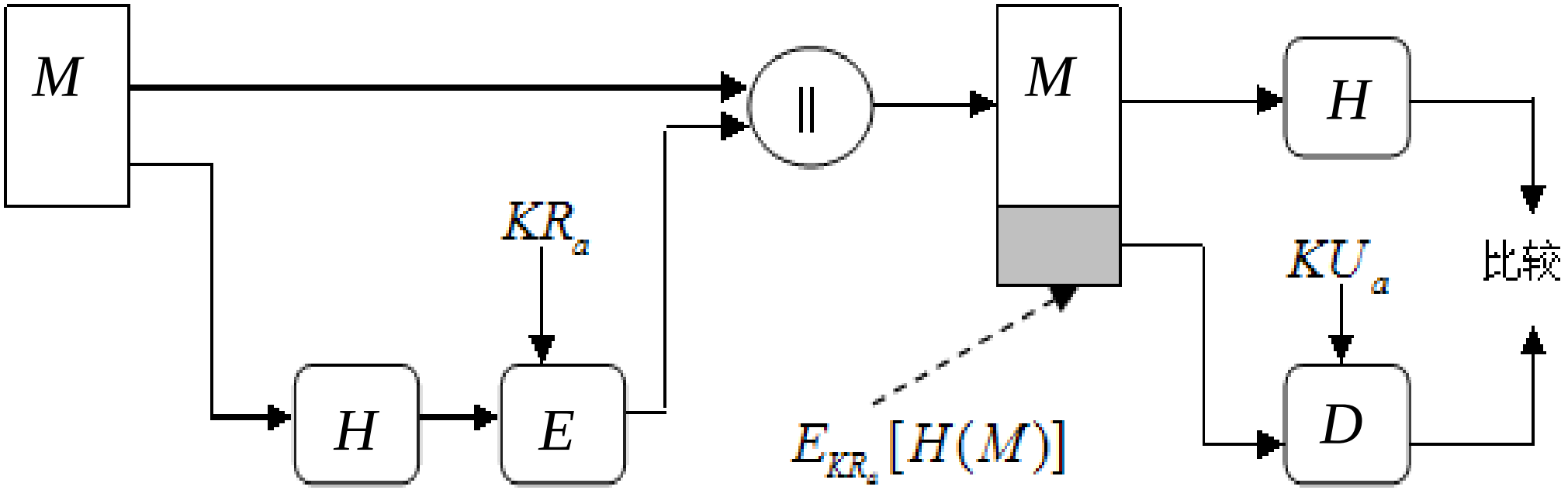
2.4 数字签名技术（原理与方法）

数字签名技术

- 普通数字签名算法：**RSA**、**DSS**、**ElGamal**、**Fiat-Shamir**、**DES/DSA**、椭圆曲线(ECC)数字签名算法等。
- 特殊数字签名有盲签名、代理签名、群签名、不可否认签名、公平盲签名、门限签名、具有消息恢复功能的签名等，它与具体应用环境密切相关。
- 数字签名技术是非对称加密算法的典型应用，是在网络系统虚拟环境中确认身份的重要技术。数字签名应用中，发送者的公钥可以很方便地得到，私钥则需要严格保密。

数字签名技术

- **RSA数字签名算法**



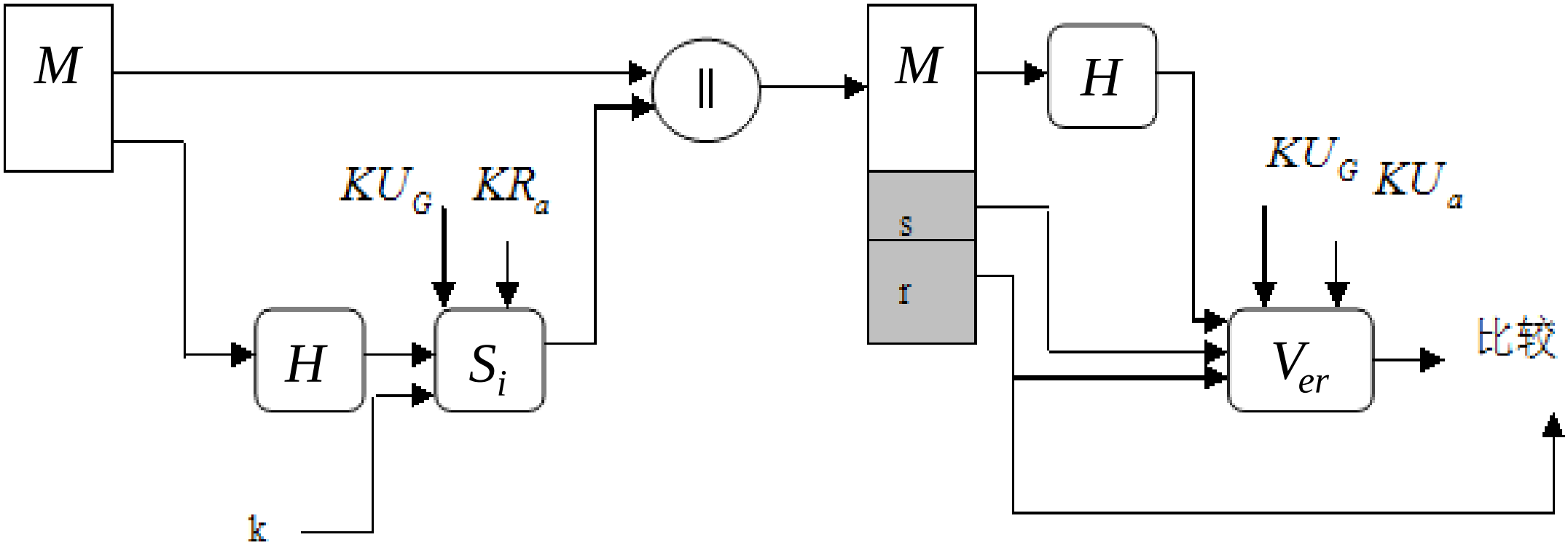
  - ***H*：Hash函数，输出定长的Hash码**

数字签名技术

- **RSA数字签名算法**
  - 签名：**Hash函数的输入是要签名的消息，输出是定长的Hash码，用发送方的私钥对该Hash码加密形成签名，然后发送消息及签名，**
  - 验证：接收方用发送方的公钥对签名进行解密，如果计算出的**Hash码与解密出的结果相同，则认为签名是有效的。**

数字签名技术

- **DSS数字签名算法**



- S_i：签名函数，其输入为Hash码和为此次签名而产生的随机数k、受控于发送方的私钥KR_a和一组参数即全局公钥KU_G

数字签名技术

- **DSS数字签名算法**
  - 签名由两部分组成，分别记为**s**和**r**。
  - 接收方对接收到的消息产生**Hash**码，这个**Hash**码和签名一起作为验证函数的输入，验证函数依赖于全局公钥和发送方公钥，若验证函数的输出等于签名中的**r**成分，则签名是有效的。

数字签名技术

- **DSA数字签名算法**
  - 建立在求离散对数的困难性以及**ElGamal**和**Schnorr**最初提出的方法之上。
  - 全局公钥（一组公开参数）：***p, q, g***
  - 用户私钥：***x***
  - 用户公钥：***y***

• DSA

全局公钥组成

P 为素数，其中 $2^{L-1} < P < 2^{L}$，$2^{9} \le P \le 2^{10}$，且 L 是 64 的倍数

$q(p\text{-}1)$的素因子，其中 $2^{159} \le q \le 2^{160}$，即位长为 160 位

$g = h^{(p-1)/q} \bmod p$，其中 h 是满足 1<h<(p-1) 并且 $h^{(p-1)/q} \bmod p > 1$ 的任何整数

用户的私钥

x 为随机或伪随机整数且 $0<x<q$

用户的公钥

$$y = g^{x} \bmod p$$

与用户每条消息相关的秘密值

$k=$ 随机或伪随机整数且 $0<k<q$

签名

$$r = (g^{k} \bmod p) \bmod q$$

$$s = [k^{-1}(H(M) + xr)] \bmod q$$

签名 $= (r, s)$

验证

$$w = (s')^{-1} \bmod q$$

$$u_1 = [H(M')^{w}] \bmod q$$

$$u_2 = (r')^{w} \bmod q$$

$$v = [(g^{u1} y^{u2}) \bmod p] \bmod q$$

检验： $v = r'$

M: 要答名的消息

$H(M)$ 使用 SHA-1 计算的 M 的 Hash 值

M', r', s' 接收到的 M, r, s

Hash函数的用法

对称密钥加密[报文+消息摘要] 提供保密、鉴别	对称密钥加密[消息摘要] 提供鉴别
$E_k[M\|\|H(M)]$	$M\|\|E_k[H(M)]$
发方私钥加密[消息摘要] 提供鉴别、数字签名	对称密钥加密[发方私钥加密消息摘要的结果] 提供鉴别、数字签名、保密
$M\|\|E_{ka}[H(M)]$	$E_k[M\|\|E_{ka}[H(M)]]$
共享密值 提供鉴别	共享密值、对称加密 提供鉴别、数字签名、保密
$M\|\|H(M\|\|S)$	$E_k[M\|\|H(M)\|\|S]$

数字签名技术的应用(1)

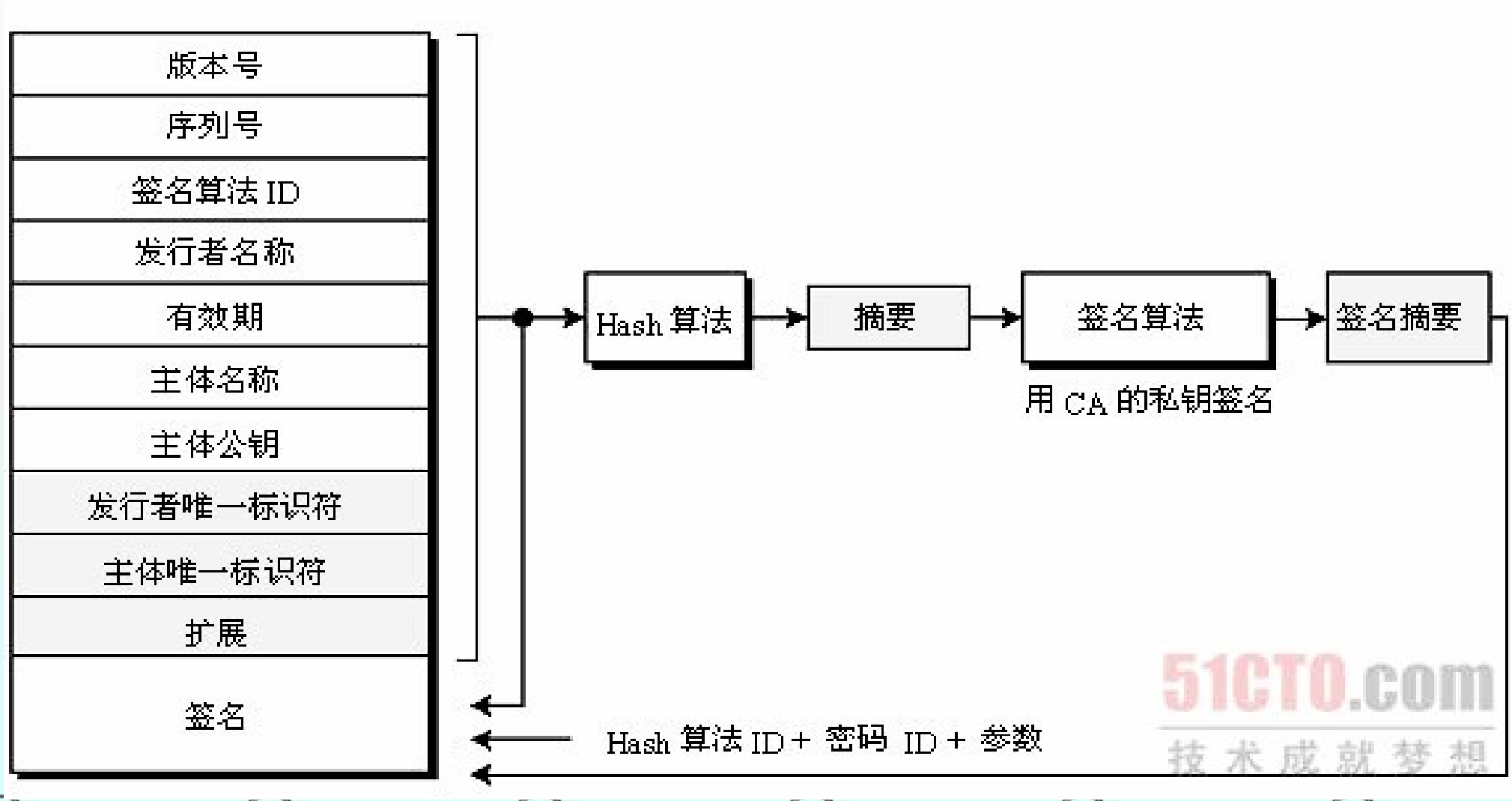
- 数字证书

数字证书

- 数字证书是一段包含用户身份信息、用户公钥信息以及身份验证机构数字签名的数据。身份验证机构的数字签名可以确保证书信息的真实性，传递的用户公钥可用于后续通信与应用。
- 数字证书是各类终端实体和最终用户在网上进行信息交流及商务活动的身份证明，在电子交易的各个环节，交易的各方都需验证对方数字证书的有效性，从而解决相互间的信任问题。
- 数字证书是一个经证书认证中心（CA）数字签名的包含公开密钥拥有者信息以及公开密钥的文件。认证中心（CA）作为权威的、可信赖的、公正的第三方机构，负责为各种认证需求提供数字证书服务。数字证书遵循X.509 标准。

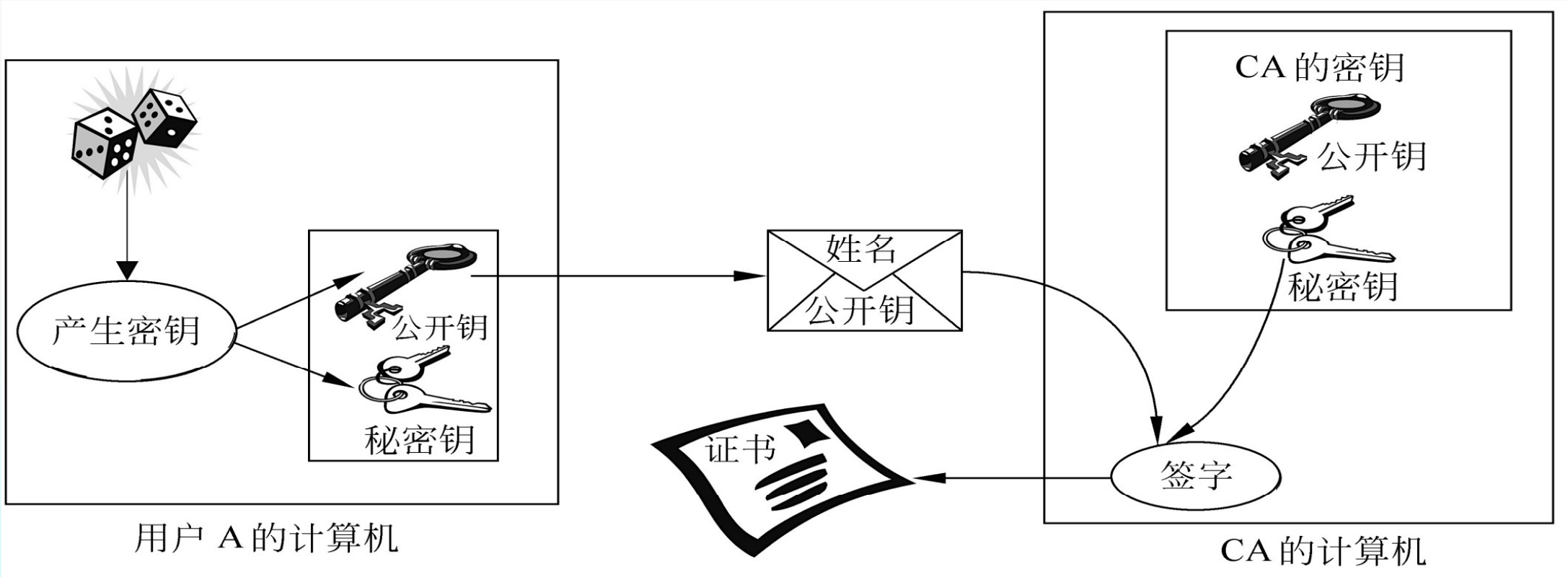
密钥管理

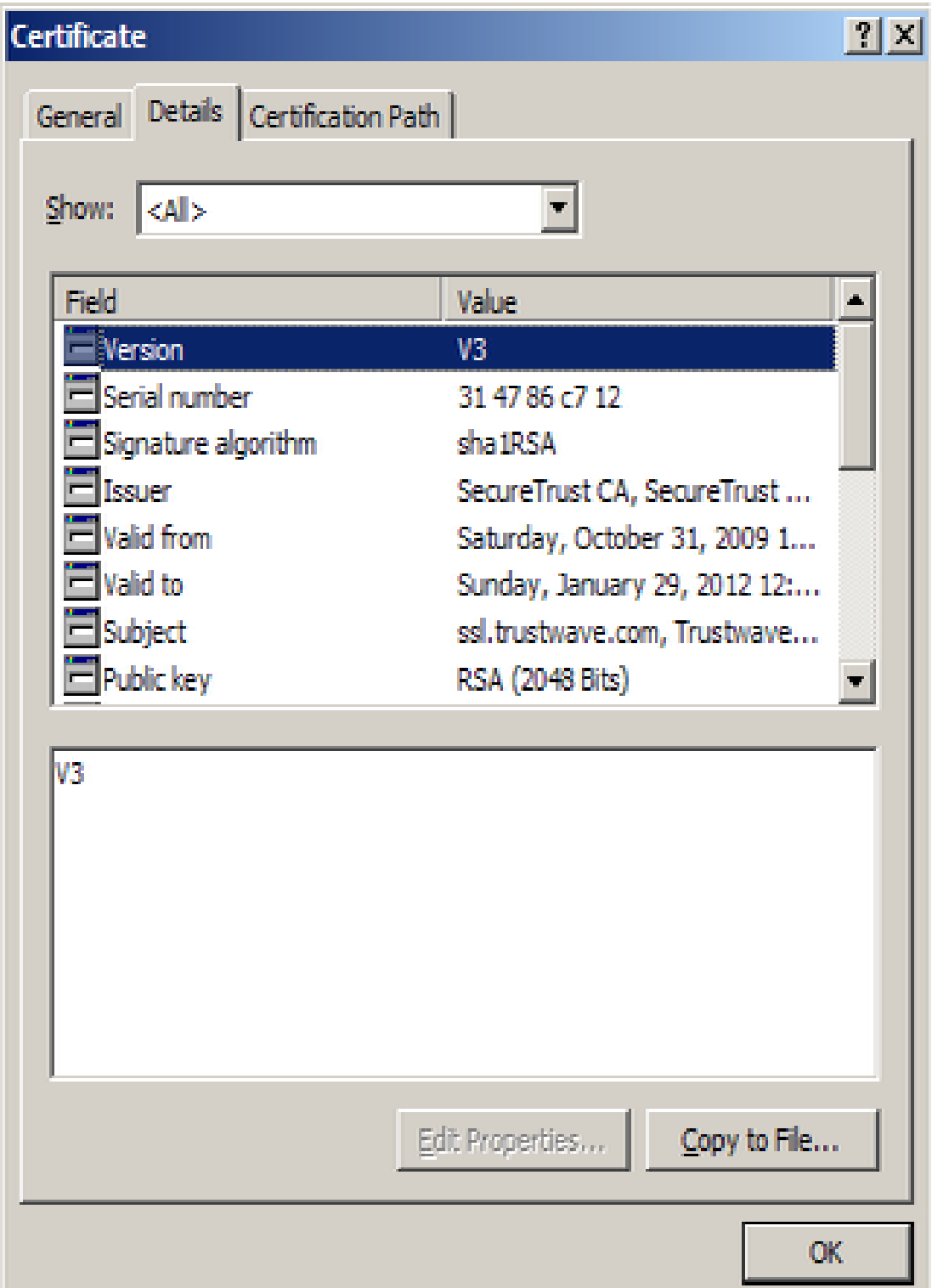
- 公钥分发——公钥证书（数字证书）

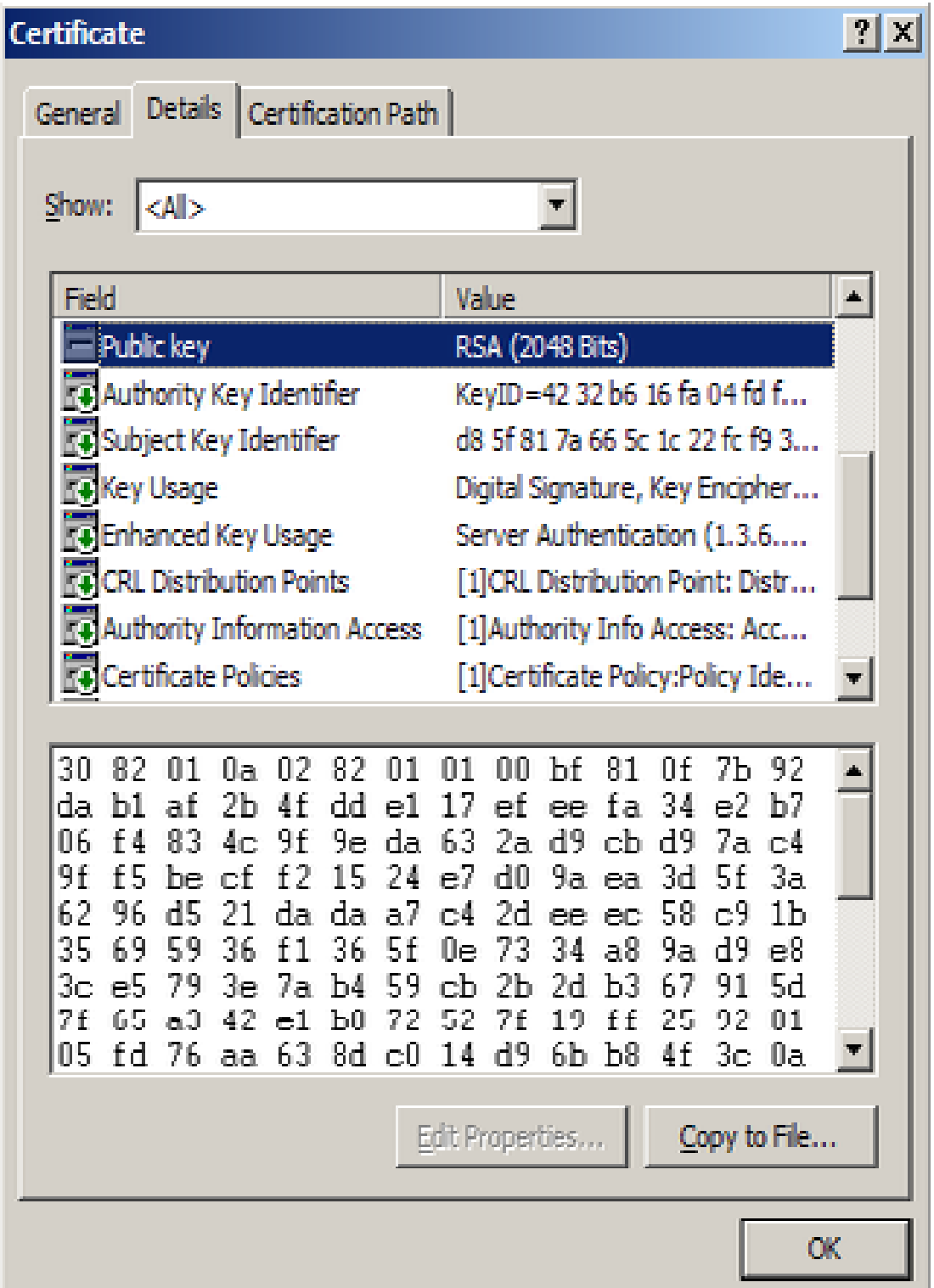


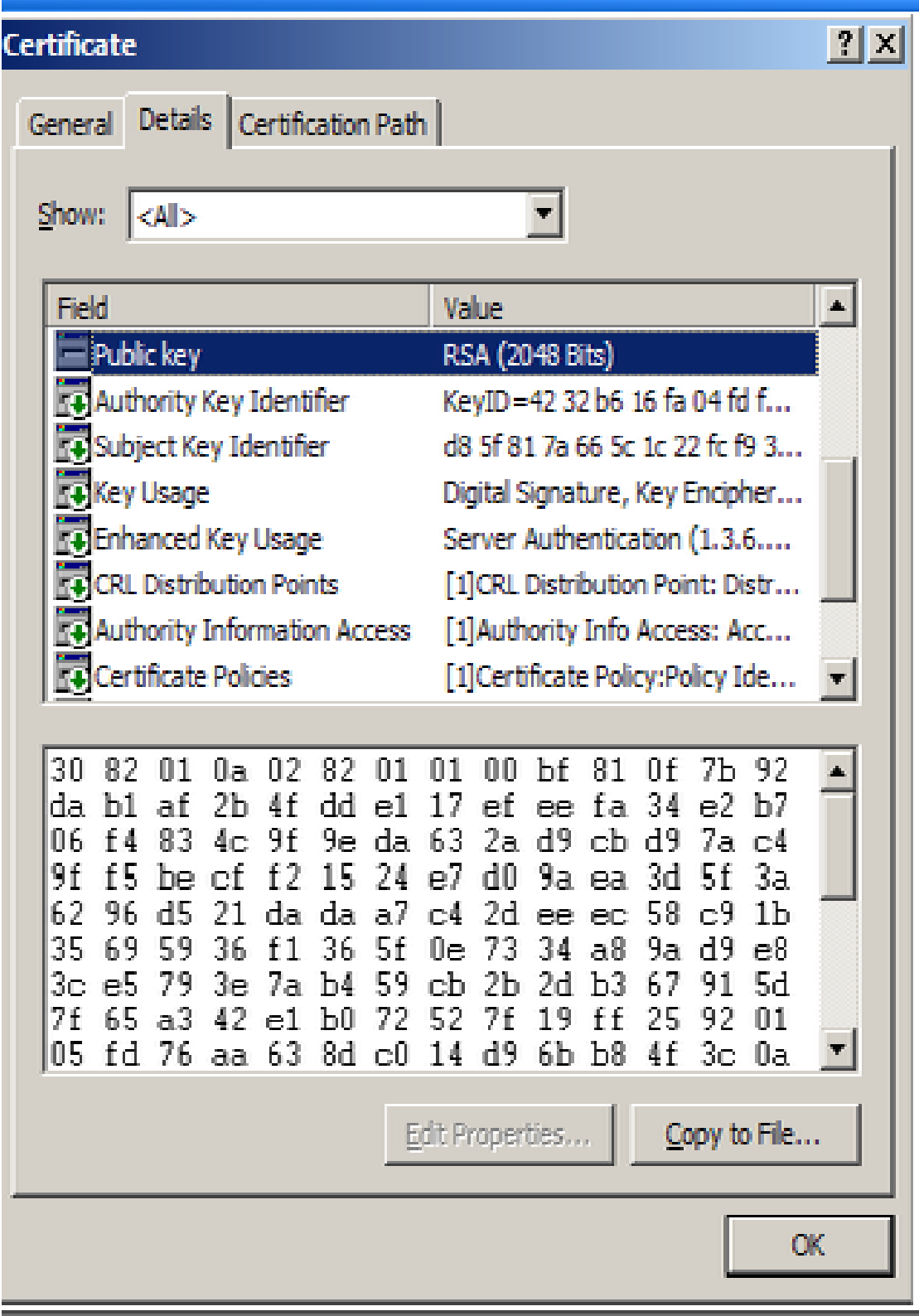
密钥管理

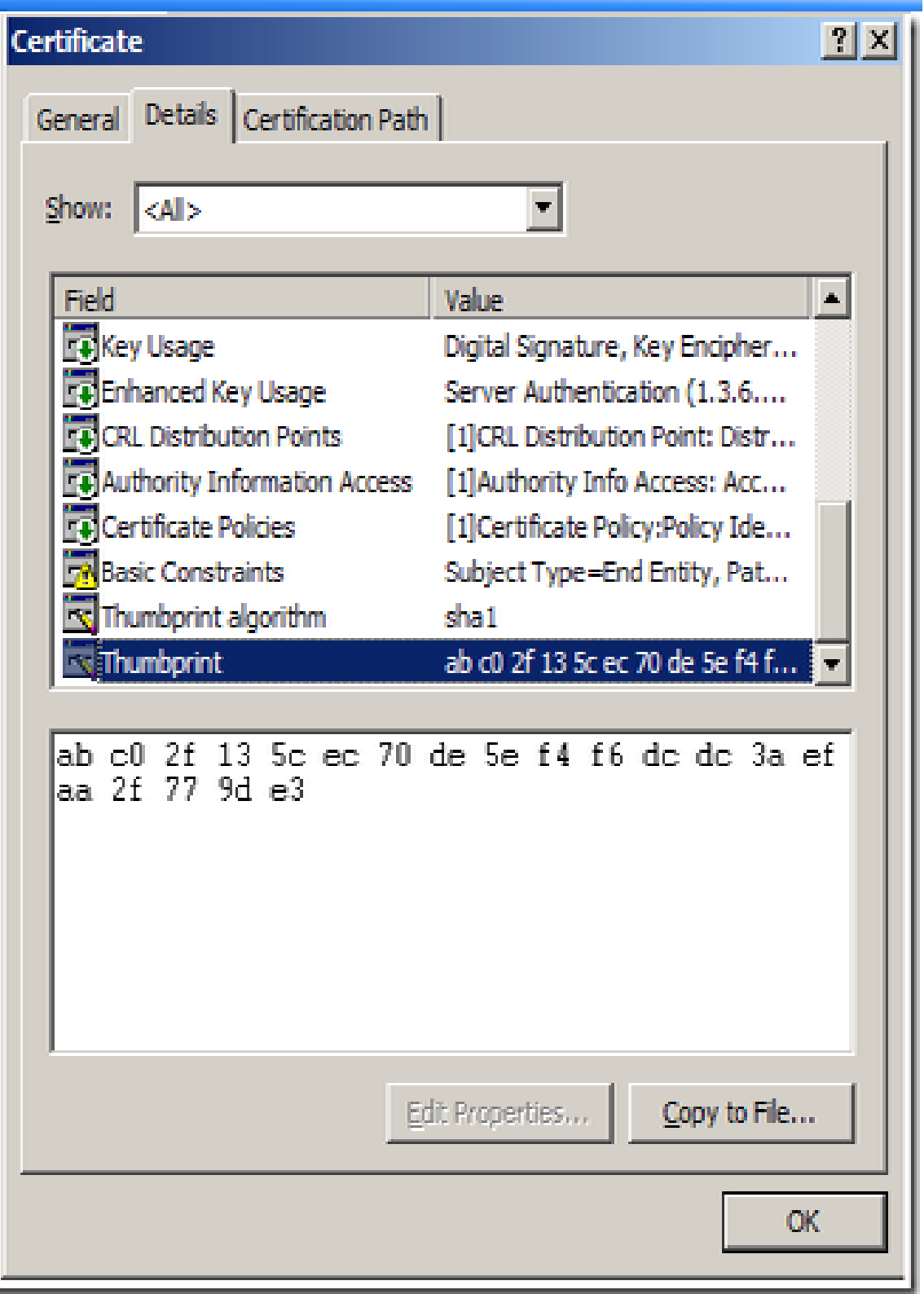
- 公钥分发——公钥证书（数字证书）




Certificate
General
Details
Certification Path
Show:
<All>
Field
Value
Version
V3
Serial number
31 47 86 c7 12
Signature algorithm
sha1RSA
Issuer
SecureTrust CA, SecureTrust ...
Valid from
Saturday, October 31, 2009 1...
Valid to
Sunday, January 29, 2012 12:...
Subject
ssl.trustwave.com, Trustwave...
Public key
RSA (2048 Bits)
V3
Edit Properties...
Copy to File...
OK


Certificate
General
Details
Certification Path
Show:
<All>
Field
Value
Public key
RSA (2048 Bits)
Authority Key Identifier
KeyID=42 32 b6 16 fa 04 fd f...
Subject Key Identifier
d8 5f 81 7a 66 5c 1c 22 fc f9 3...
Key Usage
Digital Signature, Key Encipher...
Enhanced Key Usage
Server Authentication (1.3.6....
CRL Distribution Points
[1]CRL Distribution Point: Distr...
Authority Information Access
[1]Authority Info Access: Acc...
Certificate Policies
[1]Certificate Policy:Policy Ide...
30 82 01 0a 02 82 01 01 00 bf 81 0f 7b 92
da b1 af 2b 4f dd e1 17 ef ee fa 34 e2 b7
06 f4 83 4c 9f 9e da 63 2a d9 cb d9 7a c4
9f f5 be cf f2 15 24 e7 d0 9a ea 3d 5f 3a
62 96 d5 21 da da a7 c4 2d ee ec 58 c9 1b
35 69 59 36 f1 36 5f 0e 73 34 a8 9a d9 e8
3c e5 79 3e 7a b4 59 cb 2b 2d b3 67 91 5d
7f 65 a3 42 e1 b0 72 52 7f 19 ff 25 92 01
05 fd 76 aa 63 8d c0 14 d9 6b b8 4f 3c 0a
Edit Properties...
Copy to File...
OK


Certificate
General
Details
Certification Path
Show:
<All>
Field
Value
Public key
RSA (2048 Bits)
Authority Key Identifier
KeyID=42 32 b6 16 fa 04 fd f...
Subject Key Identifier
d8 5f 81 7a 66 5c 1c 22 fc f9 3...
Key Usage
Digital Signature, Key Encipher...
Enhanced Key Usage
Server Authentication (1.3.6....
CRL Distribution Points
[1]CRL Distribution Point: Distr...
Authority Information Access
[1]Authority Info Access: Acc...
Certificate Policies
[1]Certificate Policy:Policy Ide...
30 82 01 0a 02 82 01 01 00 bf 81 0f 7b 92
da b1 af 2b 4f dd e1 17 ef ee fa 34 e2 b7
06 f4 83 4c 9f 9e da 63 2a d9 cb d9 7a c4
9f f5 be cf f2 15 24 e7 d0 9a ea 3d 5f 3a
62 96 d5 21 da da a7 c4 2d ee ec 58 c9 1b
35 69 59 36 f1 36 5f 0e 73 34 a8 9a d9 e8
3c e5 79 3e 7a b4 59 cb 2b 2d b3 67 91 5d
7f 65 a3 42 e1 b0 72 52 7f 19 ff 25 92 01
05 fd 76 aa 63 8d c0 14 d9 6b b8 4f 3c 0a
Edit Properties...
Copy to File...
OK


Certificate
General
Details
Certification Path
Show:
<All>
Field
Value
Key Usage
Digital Signature, Key Encipher...
Enhanced Key Usage
Server Authentication (1.3.6....
CRL Distribution Points
[1]CRL Distribution Point: Distr...
Authority Information Access
[1]Authority Info Access: Acc...
Certificate Policies
[1]Certificate Policy:Policy Ide...
Basic Constraints
Subject Type=End Entity, Pat...
Thumbprint algorithm
sha1
Thumbprint
ab c0 2f 13 5c ec 70 de 5e f4 f...
ab c0 2f 13 5c ec 70 de 5e f4 f6 dc dc 3a ef
aa 2f 77 9d e3
Edit Properties...
Copy to File...
OK

数字证书的组成

◆**Issuer (证书的发布机构)**

- 指出是什么机构创建和发布这个证书，不是指证书的使用者

◆**Valid from , Valid to (证书的有效期)**

◆**Public key (公钥)**

◆**Subject (主体)**

- 这个证书是发布给谁的，或者说证书的所有者，一般是某个人或者某个公司名称、机构的名称、公司网站的网址等。

◆**Signature algorithm (签名所使用的算法)**

- 指这个数字证书的数字签名所使用的加密算法，这样就可以使用证书发布机构的证书里面的公钥，根据这个算法对指纹进行解密。指纹的加密结果就是数字签名。

数字证书的组成

◆**Thumbprint, Thumbprint algorithm (指纹以及指纹算法)**

- 用来保证证书的完整性的，也就是说确保证书没有被修改过。

- 其原理就是在发布证书时，发布者根据指纹算法(hash)计算整个证书的hash值(指纹)并和证书放在一起。使用者在打开证书时，自己也根据指纹算法计算一下证书的hash值(指纹)，如同刚开始的值一致，就说明证书未被修改过

- 注意，这个指纹会使用"SecureTrust CA"这个证书机构的私钥用签名算法(Signature algorithm)加密后和证书放在一起。

CA自己的数字证书

- 为了保证安全，在证书的发布机构发布证书时，证书的指纹和指纹算法，都会加密后再和证书放到一起发布，以防有人修改指纹后伪造相应的数字证书。
- 证书的指纹和指纹算法用什么加密呢？用证书发布机构的私钥进行加密。可用证书发布机构的公钥对指纹和指纹算法解密，即证书发布机构除了给别人发布证书外，他本身也有自己的证书。
- 证书发布机构的证书是哪里来的呢？微软会根据权威评估选取信誉好并且通过一定安全认证的证书发布机构，把这些证书发布机构的证书默认就安装在操作系统里，并设置为信任的数字证书。
- 证书发布机构持有与他自己的数字证书对应的私钥，他会用这个私钥加密所有他发布的证书的指纹作为数字签名。

数字签名技术的应用(2)

- 加密通信

Reference: http://www.cnblogs.com/JeffreySun/archive/2010/06/24/1627247.html

加密通信的完整过程

Step1： “客户”->“服务器”：你好

Step2： “服务器”->“客户”：你好，我是服务器，这里是我的数字证书

Step3： “客户”->“服务器”：向我证明你就是服务器，这是一个随机字符串 //验证证书为真后

“服务器”->“客户”：**{**一个随机字符串**}[**私钥**|RSA]** //验证证书到底是不是服务器的

step4： //验证“服务器”的身份后，“客户”生成一个对称加密算法和密钥，用于后面的通信

“客户”->“服务器”：**{**我们后面的通信过程，用对称加密来进行，这里是对称加密算法和密钥**}[**公钥**|RSA]**

“服务器”->“客户”：**{OK**，已经收到你发来的对称加密算法和密钥！有什么可以帮到你的？**}[**密钥**|**对称加密算法**]**

“客户”->“服务器”：**{**我的帐号是**aaa**，密码是**123**，把我的余额的信息发给我看看**}[**密钥**|**对称加密算法**]**

“服务器”->“客户”：**{**你好，你的余额是**100**元**}[**密钥**|**对称加密算法**]**

…… //继续其它的通信

小结

2.0 密码学基础知识

2.1 消息认证概念

2.2 消息认证技术（原理与方法）

2.3 数字签名概念

2.4 数字签名技术（原理与方法）

2.5 数字签名技术的应用：数字证书

- 作业 (下次课交)
- 1. 名词解释：

  消息认证、数字签名、数字证书。
- 3. 简述带保密功能的消息认证技术的实现原理。
- 2. 简述RSA数字签名算法。