现代密码学

中国海洋大学　信息安全实验室

第7章

数字签名

本章内容

7.1 数字签名概述

7.2 RSA签名方案

7.3 ElGamal签名方案

7.4 公钥基础设施(PKI)简介

7.1 数字签名概述

对称密码技术（MAC算法）的局限性

只能实现	数据完整性
不能实现	非否认

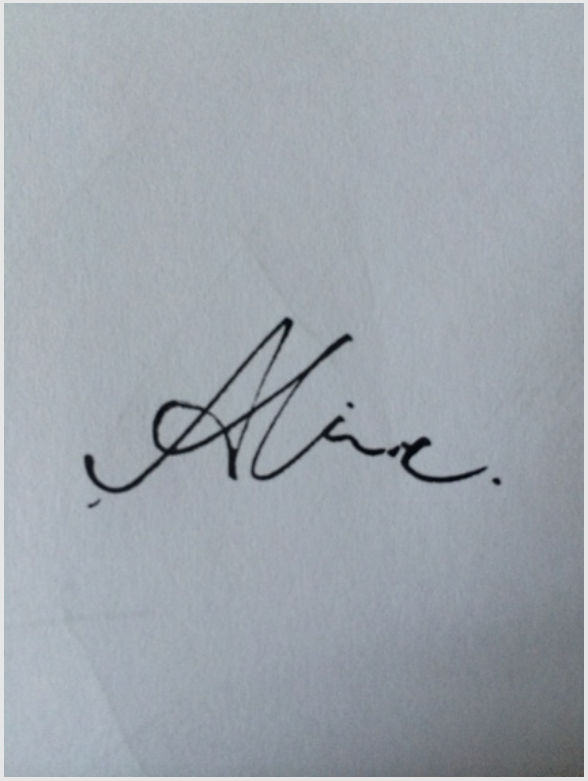
什么是 否认

要赖，不承认曾经参与过某次通信

Q: 为什么对称密码技术不能实现非否认？

- 因为双方都持有相同的密钥（信息是对称的）
  - 接收方可以产生相同的消息，所以发送方可以诬赖消息是接收方伪造的

数字签名是基于公钥思想的数据完整性技术



手写签名：Alice 对一份文件签名后

① 别人可以 **验证** 她的签名

② 其他人 **很难模仿** 她的签名

① 可验证性

② 不可伪造性

数字签名：利用电子手段对电子文档进行签名，数字签名至少要满足手写签名的两个基本性质

① 别人可以 **验证** 数字签名

② 其他人 **很难模仿** 数字签名

① 可验证性

② 不可伪造性

- 由于数字签名技术对政府、企事业、一般团体和个人的重要影响，世界各国都加强了对它的研究。
  - · 1994年，美国正式颁发美国数字签名标准DSS
  - · 1995年，我国制定自己的签名标准(GB15851-1995)
  - · 1999年，美国参议院已通过了立法，规定电子数字签名与手写签名的文件、邮件在美国具有同等的法律效力
  - · 2004年，我国颁发《中华人民共和国电子签名法》

数字签名的基本思想：

发送者利用 自己的私钥SK 产生消息的认证码 (类似于MAC)

只有发送者掌握 SK，所以该认证码只有发送者才能产生

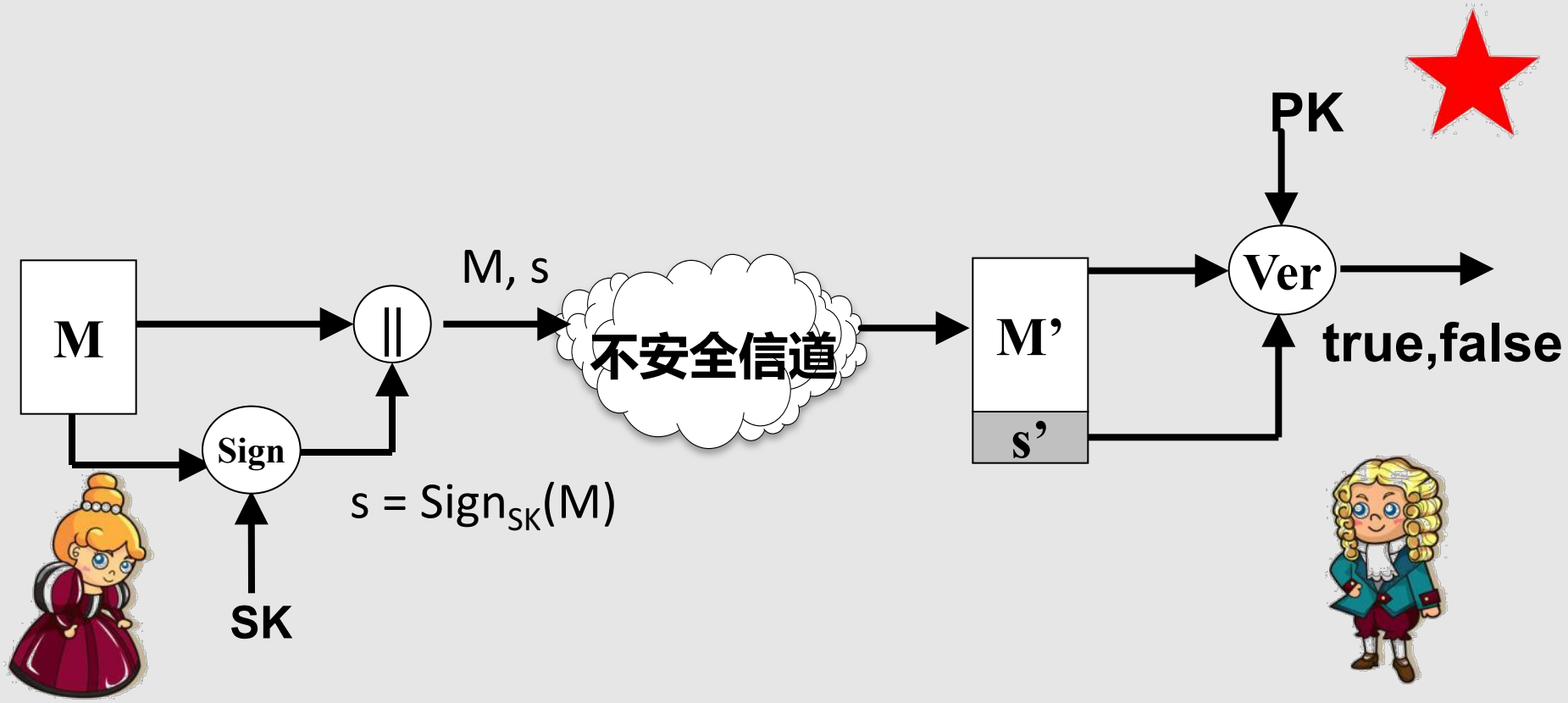
任何人都可以用相应的公钥 PK 验证认证码的合法性

只要通过验证，就可以确信发送者产生了该消息

认证码相当于发送者在消息上做的“签名”，故而称作 数字签名

- 数字签名方案包括三个组成部分：
  - 密钥生成 Setup：产生公钥/私钥
  - 签名算法 Sign：利用私钥对消息产生数字签名
  - 验证算法 Ver：利用公钥对数字签名进行验证

Setup	**$Setup(1^k) = PK / SK$**
签名	**$Sign_{SK}(M) = s$**
验证	**$Ver_{PK}(M, s) = true/false$**
$Ver_{PK}(M, Sign_{SK}(M)) = true/false$	



类似 MAC 算法

数字签名原理图

- **Sign 是签名算法**
- **Ver 是验证算法**

数字签名 主要安全特性

抗伪造

任何人都不能
伪造他人的签名

防篡改

任何人无法篡改
已签名的消息

非否认

签名者事后无法
否认自己的签名

消息认证

接收者可以确信
消息发送者的身份

相当于在电子文件上签自己的名字

抗伪造是数字签名的核心安全性要求

什么叫 伪造签名

在不知道私钥 SK 的情况下，产生签名 s，使得

$Ver_{PK}(M,s) = true$

则称 s 是对 M 伪造的签名 (PK 是 SK 对应的公钥)

也即 找到一对能通过验证算法校验的 (M,s)

什么叫 抗伪造

给定消息 M，在不知道私钥 SK 的情况下，产生签名 s，使得 $Ver_{PK}(M,s) = true$ 是计算上不可行的（就算已知一大堆消息以及其对应的签名也是如此）

数字签名 安全模型

① **完全攻破**：攻击者能找到私钥 SK

（UB: Unbreakability，不可完全攻破）

② **泛伪造**：攻击者可以对任何消息产生合法的签名

（UU: Universal Unforgeability，泛不可伪造）

③ **选择性伪造**：对别人选择的消息，攻击者能以不可忽略的概率产生一个合法的签名

（SU: Selective Unforgeability，选择性不可伪造）

④ **存在性伪造**：攻击者能至少为一条消息产生合法的签名

（EU: Existential Unforgeability，不可存在性伪造）

攻击者最容易实现的伪造

数字签名 攻击分类

依据攻击者获得信息的多少，对数字签名的攻击分类：

①唯密钥攻击: 攻击者只知道公钥

(KOA: Key only attack)

②已知消息攻击: 攻击者拥有一系列用私钥签过的消息和相应的签名

(KMA: Known message attack)

③选择消息攻击: 攻击者任意选择一系列消息，并可获得相应的签名

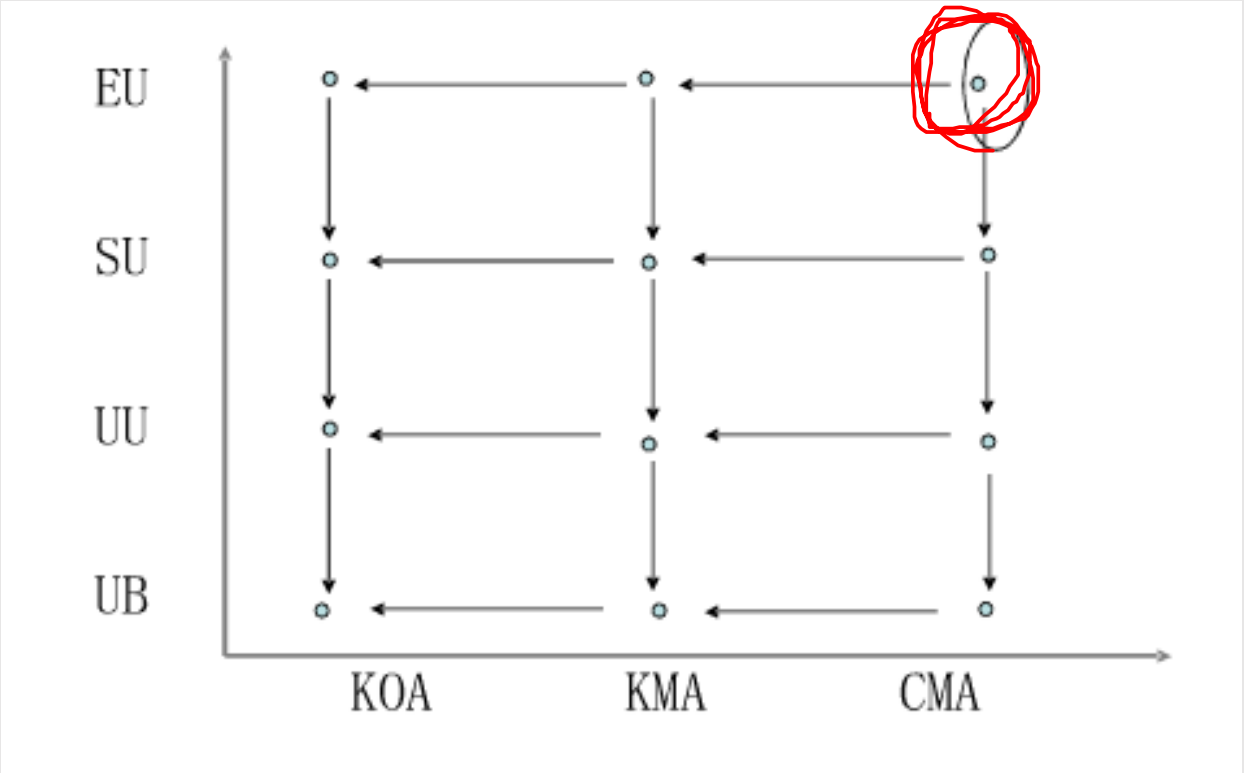
(CMA: Chosen message attack)

攻击者的灵活性最大

攻击者的目标	恢复私钥(但很难实现)，或伪造签名

数字签名 安全模型关系图

攻击者最容易实现的伪造
而且 灵活性最大



数字签名方案的最高安全性要求：

选择消息攻击下 不可存在性伪造（EU-CMA）

7.2 RSA签名方案

- RSA算法不仅可用于加密，还可用于数字签名
- 注意
  - 绝大多数算法只能用于加密或签名二者之一

RSA签名方案 描述

① **系统建立：**

· 随机选择大素数p、q，计算n = pq

· 随机选取e < ϕ(n)，且gcd(e, ϕ(n))=1

· 计算d，使 ed≡1 (mod ϕ(n))

· (e，n)为公钥

· d 为私钥

(与RSA加密方案的系统建立过程完全一样)

② **签名：** $s = m^d \bmod n,\ m \in Z_n^*$

③ **校验：** $m \stackrel{?}{=} s^e \bmod n$

RSA签名方案 安全性

- **安全性原理**
  - 只有签名者知道私钥d，所以他是产生签名s的唯一人
  - 公钥e是公开的，任何人都可以验证签名s的合法性
- 但上述基本的RSA签名方案有安全漏洞 —— 存在性伪造

RSA签名方案　存在性伪造

先看一下数字签名的原理

- 给定消息m, 计算签名 $s=Sig_{sk}(m)$
- 给定 m 和 s, 验证合法性 $Ver_{pk}(m,s)?=true$

存在性伪造的原理：将产生签名的思路反过来

- 先选择s，再构造相应的消息m, 使得 $Ver_{pk}(m,s)=true$
- 这样不知道私钥sk，也可以产生满足验证算法的消息和签名，这种伪造称为 存在性伪造

RSA签名方案　RSA签名方案的存在性伪造

伪造的过程

- 攻击者随机选择s
- 用签名者公钥(e,n)计算 $M \leftarrow s^e \bmod n$
- 将(M,s)作为消息/签名输出

Q: (M,s)能否通过验证算法的校验呢？

- 很明显，(M,s) 满足验证算法 $M = s^e \bmod n$ 的要求
- 因此，(M,s) 是一对合法的伪造

RSA签名方案　其他漏洞举例

- 可以利用两个消息的签名，产生新消息的签名
  - 如果攻击者知道消息m_1和m_2的签名，设分别是

    $s_1 = m_1^d \bmod n$，$s_2 = m_2^d \bmod n$

    则可以伪造消息$m=m_1m_2$的签名$s=s_1s_2$
  - 因为RSA签名方案存在以下性质：

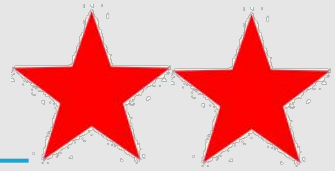
    $(m_1m_2)^d \equiv m_1^d m_2^d \pmod n$

    所以，$s^e \equiv (s_1s_2)^e \equiv (m_1^d m_2^d)^e$

    $\equiv ((m_1m_2)^d)^e \equiv m_1m_2$

    $\equiv m \pmod n$

HASH函数在数字签名中的重要作用



- **重要作用**
  - ① 抵抗存在性伪造
  - ② 加快计算速度
- **方法**
  - “对消息m签名”改为“对H(m)签名”
- **原理**
  - ① 抵抗存在性伪造：利用Hash函数的单向、抗碰撞等性质
  - ② 加快计算速度：H(m)比m短得多

利用Hash函数改进RSA签名方案

- **改进后的方案**
  - 签名：$s = H(m)^d \bmod n$
  - 验证：$H(m) \stackrel{?}{=} s^e \bmod n$
- **可抵抗“存在性伪造”**
  - 攻击者随机选择s, 用签名者公钥计算$h=s^e \bmod n$
  - 但计算一个m，使得$H(m)=h$在计算上不可行
- **可抵抗“利用两个消息的签名，产生新消息的签名”**
  - 因为$(H(m_1)H(m_2))^d \neq H(m_1m_2)^d \pmod n$

利用Hash函数改进RSA签名方案

- **注意**
  - 目前没有严格证明表明其在 选择消息攻击下 不可存在性伪造 (EU-CMA)
  - 一些RSA签名方案的变形在 某些假设下 能被证明在 选择消息攻击下 不可存在性伪造 (EU-CMA)

7.3 ElGamal签名方案

ElGamal签名方案

- 1985年提出
- 其变型已被NIST采纳为数字签名算法(DSA)
- 安全性基于“离散对数”问题

Taher Elgamal

ElGamal签名方案 描述

① **系统建立**

- 随机选择大素数p, 及生成元$g \in Z^*_p$
- 随机选取$0 < x \le p-2$，计算 $y=g^x \bmod p$
- 公钥是(p,g,y)
- 私钥是x

(与ElGamal加密方案的系统建立过程完全一样)

目前，p的长度至少应为512比特

ElGamal签名方案 描述

② **签名**

对消息m，随机选择$0 < r \le p-2$，然后计算：

$u = g^r \bmod p$

$s = r^{-1}(m - xu) \bmod (p-1)$

m的签名为(u,s)

③ **验证**

对于消息/签名 (m,(u,s))，如果：

$y^u u^s \equiv g^m \pmod p$

则(u,s)是m的有效签名

ElGamal签名方案 描述

- **正确性：**

$y=g^x$、$u=g^r$

$$y^u u^s = g^{xu} g^{rs}$$

$$= g^{xu+rs}$$

$$= g^m \bmod p$$

$$xu+rs = xu + r(r^{-1}(m - xu))$$
$$= xu + m - xu$$
$$= m$$

ElGamal签名方案 存在性伪造

- **对ElGamal签名的存在性伪造**
  - 攻击者随机选择$0 < r,v \le p-2$，且$\gcd(v,p-1)=1$
  - 计算 $u = g^r y^v \bmod p$

    $s = -uv^{-1} \bmod (p-1)$

    $m = -ruv^{-1} \bmod (p-1)$

  则(u,s)是对伪造的消息m的有效签名
- **伪造的正确性**
  - $y^u u^s = y^u (g^r y^v)^s = g^{xu}(g^r g^{xv})^s$

    $= g^{xu+rs+xvs}$

    $= g^m \bmod p$

$$
\begin{aligned}
xu+rs+xvs &= xu+rs+xv(-uv^{-1}) \\
&= xu+rs-ux \\
&= rs \\
&= r(-uv^{-1}) \\
&= -ruv^{-1} \\
&= m
\end{aligned}
$$

ElGamal签名方案　利用HASH函数改进

- **签名**

  $u=g^r \bmod p$,

  $s=r^{-1}(H(m,u)-x\,u) \bmod (p-1)$

  其中Hash函数$H:\{0,1\}^* \rightarrow Z_p$

  则m的签名为(u,s)

- **验证**

  对于消息/签名(m,(u,s))，如果：

  $$y^u u^s = g^{H(m,u)} \bmod p$$

  则(u,s)是m的有效签名。

ElGamal签名方案

- **注意**

① 每次签名时，必须选择不同的 r，否则私钥 x 可能会泄露

② 效率不如RSA签名高，而且数据长度有扩张

③ 有很多种变形

④ 有的ElGamal签名方案的变形能在 某些假设下 被证明在选择消息攻击下 是安全的

ElGamal签名方案 DSA

- DSA (数字签名算法) 是 NIST 在 1991年选定的数字签名标准
- DSA类似于ElGamal，但具有明显的优势

  ① 效率更高

  ② 签名更短

具有特殊功能的数字签名方案

Bob 想让 Alice 对一个消息进行签名，但又不想让 Alice 知道消息的内容，且当Bob揭示签名和消息后， Alice 无法知道这就是当初她签过的消息。如何实现？

—— 盲签名 (Blind Signature)

- 由 Chaum 于1983年提出，并申请了专利
- 应用：电子投票

具有特殊功能的数字签名方案

扩展阅读

- Group Signature
- Online/Offline Signature
- Undeniable Signature
- One-time Signature
- Fail-stop Signature
- Proxy Signature

7.4 公钥基础设施(PKI)

- 公钥的分发仍是个严重的问题
  - 当你要与Alice通信时，首先你必须获得她的公钥
  - 但是，你从网络上接收到的公钥只是一个杂乱无章的比特串，你能确定它就是Alice的公钥吗？
- 解决方案
  - 以公钥和用户身份作为消息，产生 **数字签名(目的是抗伪造)**，将两者捆绑在一起
- 现在的问题是，该签名由谁产生？
  - 其他用户？—— 缺乏公信力
  - 应该由可信第三方承担，称之认证机构(CA)

“公钥证书”与“身份证”的类比

数字世界

CA

签发

证书

证明公钥所有者的真实性

证明持有者的真实性

现实世界

公安局

签发

身份证

公钥证书的逻辑形式

Name:	Tom
Serial number:	484865
Issued by:	GA-CA
Issue date:	1997 01 02
Expiration date:	2003 01 02
Public key:	84A3796301C…

用户身份和用户公钥的结合体；

由CA审核用户身份后签发

公钥基础设施(PKI)

- 为配合公钥证书的签发，需要十分复杂的管理机构，CA只是其中一个组成部分而已
- 由此，引出PKI的概念

公钥基础设施(PKI) 概述

通过网络进行交流和商业活动，面临的最大问题是

- 如何建立相互间的信任关系
- 如何确保信息的真实性、完整性、机密性和非否认

PKI是解决这一系列问题的技术基础

它是电子商务、电子政务的关键和基础技术

目的

- 为了提供可信任的高效密钥和证书管理，以支持众多依赖于公钥的安全技术

公钥基础设施(PKI) 概述

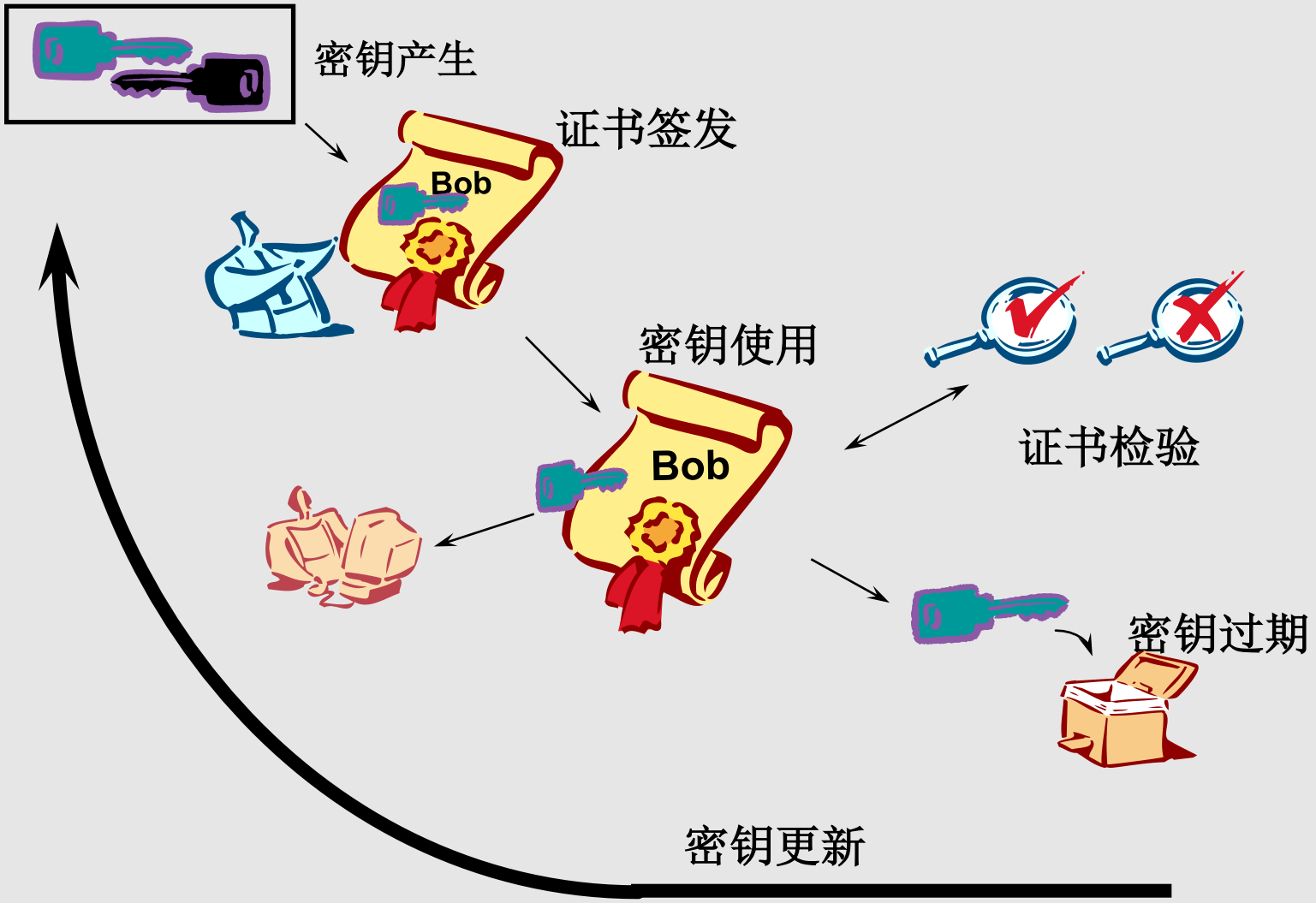
PKI的定义

一个使用公钥概念和密码技术实施和提供安全服务的具有普适性的安全基础设施的总称

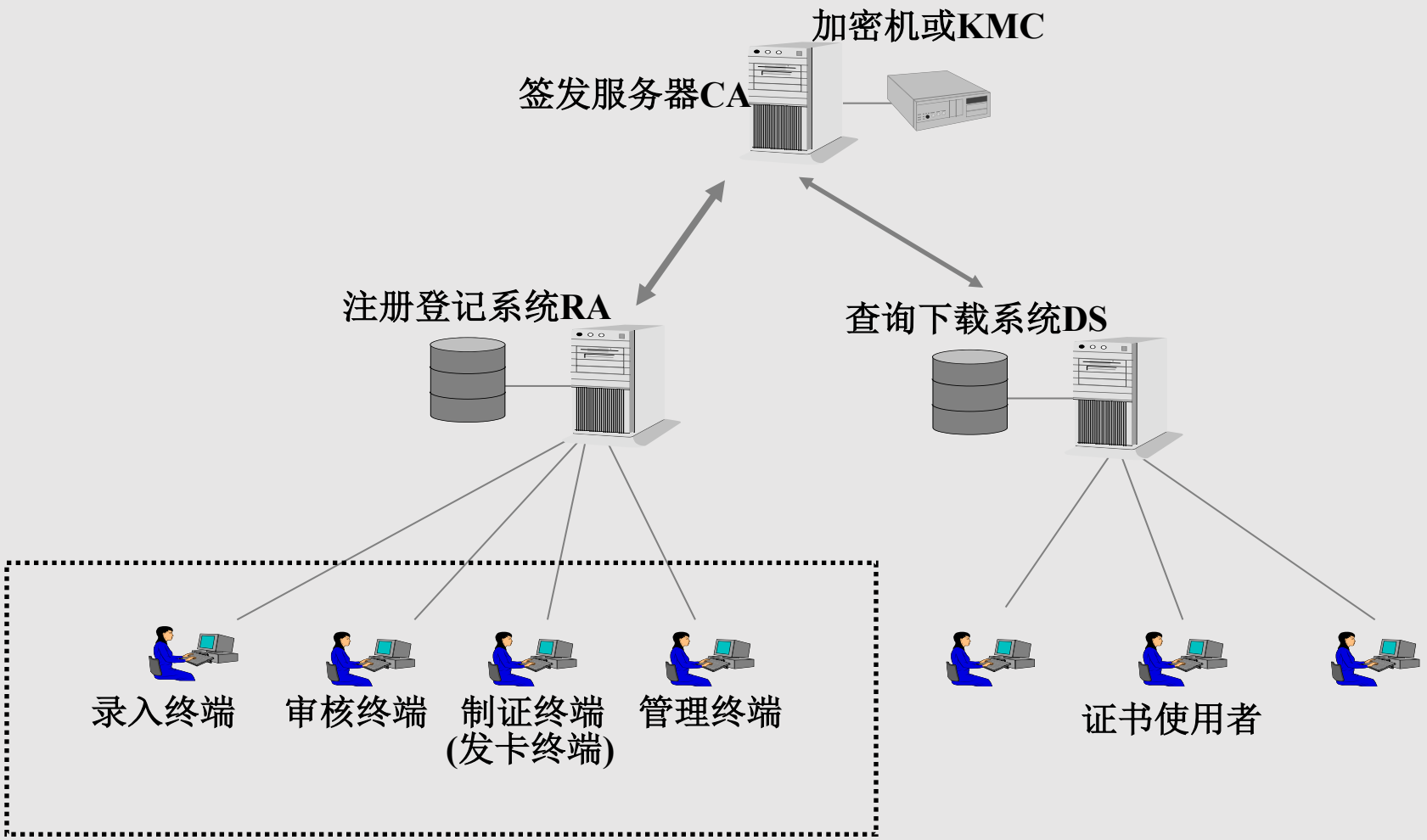
PKI不是特指某一个密码设备和管理设施，它是

- 生成、管理、存储、颁发和撤销公钥证书所需要的软硬件、人员、策略和规程的总和
- 提供密钥管理和数字签名服务的平台

公钥基础设施(PKI)　密钥生成周期

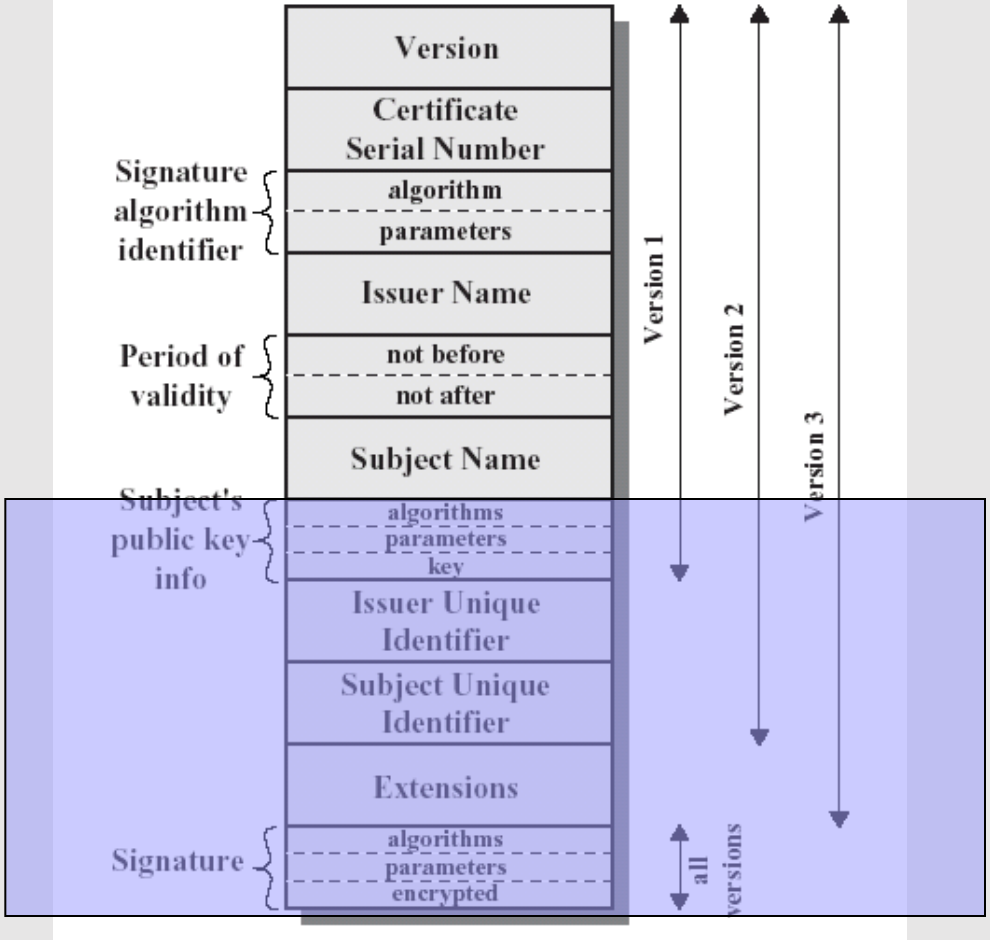


公钥基础设施(PKI) 系统架构



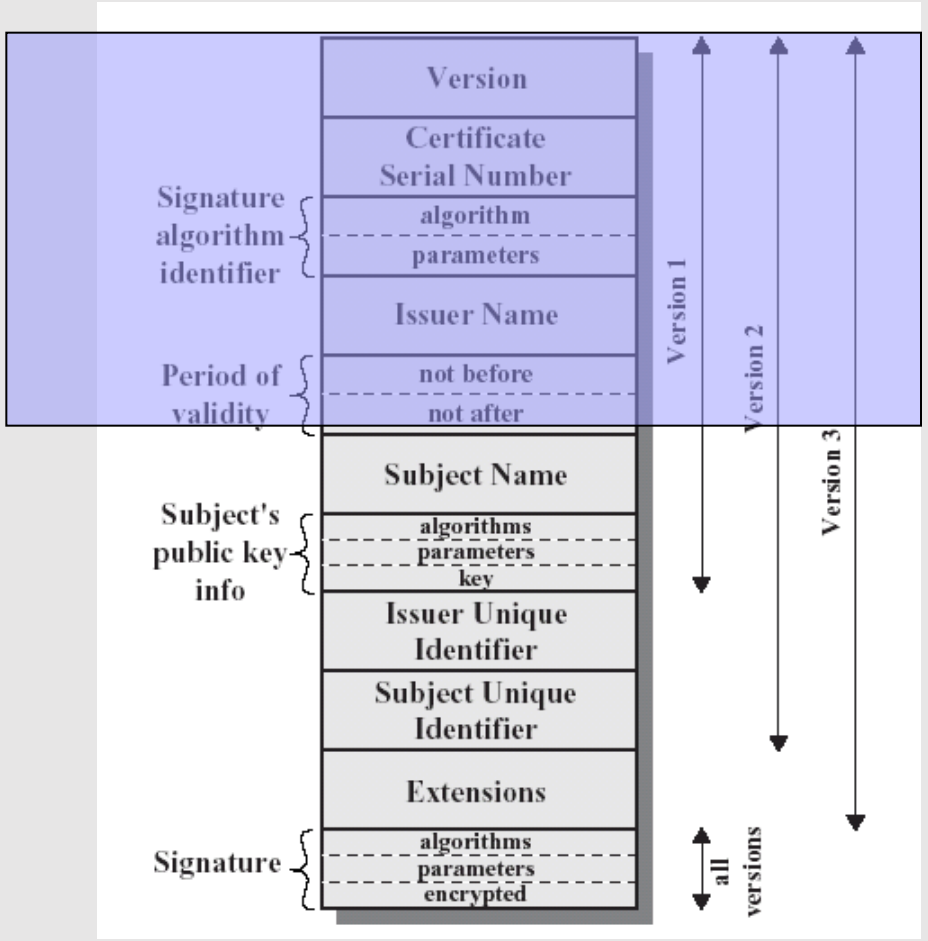
PKI 相关标准　X.509证书格式

- 版本1、2、3
- 序列号
  - 在CA内部唯一
- 签名算法标识符
  - 指该证书中的签名算法
- 签发人名字
  - CA的名字
- 有效时间
  - 起始和终止时间
- 实体名字

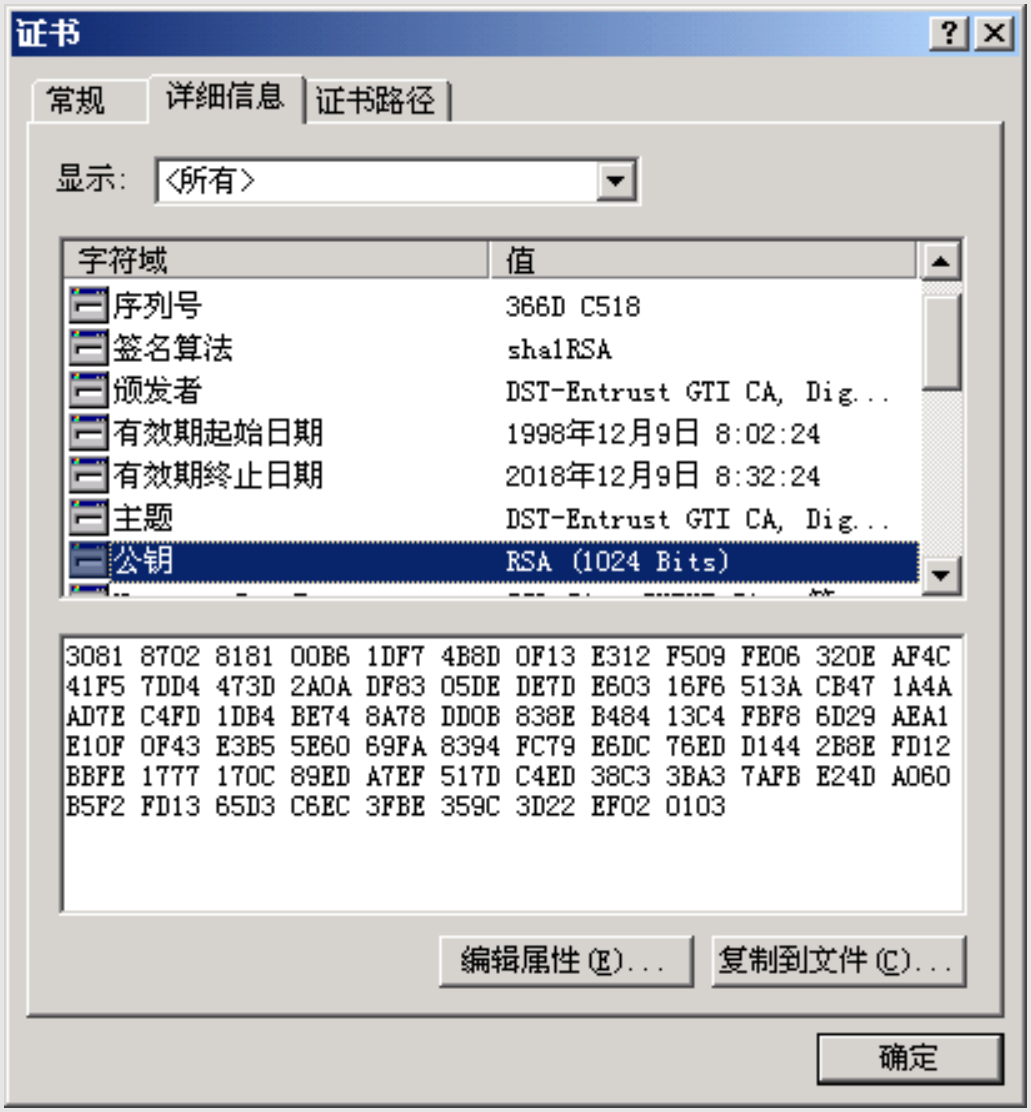


PKI 相关标准　X.509证书格式

- 实体的公钥信息
  - 算法
  - 参数
  - 密钥
- 签发者唯一标识符
- 实体唯一标识符
- 扩展域
- 签名



PKI 相关标准　X.509证书格式



PKI 相关产品

① **VeriSign (www.verisign.com)**

- 最大的公共CA，最早推广PKI的公司之一
- 最可信的公共CA之一，开发了一些使用工具

② **Entrust 公司——世界一流**

- 产品：Entrust/PKI 5.0
- 优点：
  - ① 管理和安全做得都很出色
  - ② 与其它产品兼容
  - ③ 支持各种标准:X.509等

本章小结

1. 掌握数字签名的含义、特性、存在性伪造的含义
2. 掌握RSA、ElGamal签名方案
3. 掌握Hash函数在数字签名中的重要作用
4. 掌握提出PKI 的意义、CA的作用
5. 掌握4种信任模型的工作原理
6. 了解数据库和目录服务的不同之处
7. 了解X.509、LDAP的用途

练习题

1. 数字签名无法提供的特性是（ D ）

A. 抗伪造　　B. 非否认　　C. 不可重用性　　D. 保证可用性

2. 哪个不是对数字签名的攻击方法（ D ）

A. 唯密钥攻击　　B. 已知消息攻击

C. 选择消息攻击　　D. 已知明文攻击

3.下列哪个是公钥证书格式的标准（ B ）

A. X.500　　B. X.509　　C.LDAP　　D. OCSP