# 第5章

# 公钥基础设施

## 5.1 概　　述

**公钥基础设施**(Public Key Infrastructure,PKI)技术引起了极大关注,成为 Internet 上现代安全机制的焦点中心。PKI 不是一个简单的思想,而是需要大量经费、精力和决心才能建立、维护与促进的技术方案。PKI 是几乎所有加密系统的必经之路。

本章将详细介绍 PKI。PKI 与非对称密钥加密密切相关,包括消息摘要、数字签名和加密服务,这些已经在前面介绍。要支持所有这些服务,最主要的要求是数字证书技术。数字证书是 Web 上的护照,将在本章详细介绍。

数字证书带来了许多问题、担心和争议,本章将会详细进行介绍。我们介绍证书机构(CA)、注册机构(RA)的作用,CA 之间的关系和根 CA、自签名证书、交叉证书等概念。

验证数字证书既关键又复杂,可以用 CRL、OCSP 与 SCVP 之类特殊协议进行处理,我们将逐一进行介绍和比较。

CA 的其他要求是维护与建档用户密钥和在必要时提供出来供用户使用,漫游证书之类思想也得到重视,也将在本章介绍。

PKIX 与 PKCS 是两个最常用的数字证书与 PKI 标准,本章将详细介绍 PKIX 与 PKCS 及其关键领域。XML 安全性近来备受重视,因为其特性丰富,加上 XML 已成为全球信息交换标准。本章也会介绍 XML 安全的各个方面。

## 5.2 数 字 证 书

### 5.2.1 简介

我们已经详细介绍了密钥协定或密钥交换的问题,介绍了 Diffie-Hellman 密钥交换算法之类的算法如何处理这个问题,存在哪些缺点。非对称密钥加密是很好的解决方案,但也有未能解决的问题,就是双方(消息发送方与接收方)如何相互交换公开密钥? 显然,不能公开交换,否则很容易在公开密钥时受到中间人攻击。

因此，密钥交换的问题是个难题，事实上也是设计任何计算机加密方案时最大的难题。经过大量思考，人们提出了**数字证书**(digital certificates)的革命性思想，详见下面介绍。

在概念上，数字证书相当于护照、驾驶证之类的证件。护照和驾照可以帮助证明身份。例如，笔者的护照至少可以证明以下几个方面：

- 姓名
- 国籍
- 出生日期和地点
- 照片与签名

同样，数字证书也可以证明一些关键信息，详见下面几节介绍。

## 5.2.2　数字证书的概念

数字证书其实就是一个小的计算机文件。例如，笔者的数字证书就是一个计算机文件，文件名为 atul.cer(其中.cer 是单词 certificate 的前三个字母。当然，这只是一个示例，实际上可以使用不同的文件扩展名)。护照证明笔者与姓名、国籍、出生日期和地点、照片与签名等的关联，而数字证书证明我与公开密钥的关联。图 5.1 显示了数字证书的概念。

我们没有指定用户与数字证书之间的关联由谁批准。显然，这要有某个机构，而且是各方都信任的。假设护照不是政府发的，而是一个小店主发的，你能相信吗？同样，数字证书也要由信任实体签发，否则很难让人相信。

前面曾介绍过，数字证书建立了用户与公钥的关联。因此，数字证书要包含用户名和用户的公钥，证明特定公钥属于某个用户。除此之外，数字证书还包含什么信息呢？图 5.2 显示了数字证书的示例。

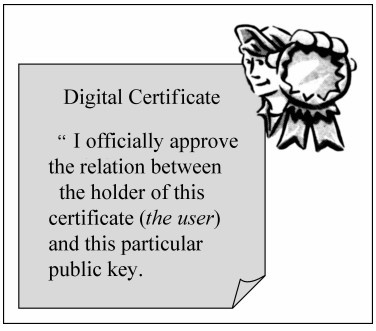


图 5.1　数字证书的直观概念

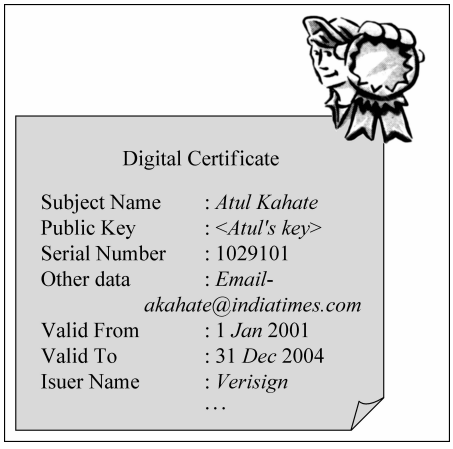


图 5.2　数字证书的示例

这里有几个有趣的事实。首先，笔者的姓名显示为主体名(subject name)。事实上，数字证书中任何用户名都称为主体名(因为数字证书可以发给个人、组织或小组)。还有另一个有趣的信息，称为序号(serial number)，稍后将介绍其含义。证书还有其他信息，如证书有效期和签发者名(issuer name)。可以把这些信息与护照中的项目做一个比较，如图 5.3

所示。

| 护照项目 | 数字证书项目 |
|---|---|
| 姓名(Full name) | 主体名(Subject name) |
| 护照号(Passport number) | 序号(Serial number) |
| 起始日期(Valid from) | 相同 |
| 终止日期(Valid to) | 相同 |
| 签发者(Issued by) | 签发者名(Issuer name) |
| 照片与签名(Photograph and signature) | 公钥(Public key) |

图 5.3 数字证书与护照的相似性

从图 5.3 中可以看出，数字证书与护照很相似。每个护照有一个护照号，而数字证书则具有唯一序号。我们知道，同一签发者签发的护照不会有重号，同样，同一签发者签发的数字证书也不会有重号。谁来签发这些数字证书？稍后将回答这个问题。

### 5.2.3 证书机构

**证书机构**(Certification Authority，CA)就是可以签发数字证书的信任机构。谁是证书机构？显然不能随便找一个阿狗阿猫作为证书机构，而应该是每个人都信任的机构。因此，各国政府会确定谁有资格成为证书机构(但并不是每个人都信任政府！这是另一个问题)。通常，证书机构是一些著名组织，如邮局、财务机构、软件公司，等等。世界上最著名的证书机构是 VeriSign 与 Entrust。Safescrypt 公司是 Satyam 信息公司的子公司，2002 年 2 月成为印度第一家证书机构。

这样，证书机构有权向个人和组织签发数字证书，使其可以在非对称密钥加密应用程序中使用这些证书。

### 5.2.4 数字证书技术细节

下面看看数字证书的技术细节。我们已经从概念上做了介绍，下面要从技术角度进行介绍。读者可以跳过本节，而不失连贯性。

数字证书的结构在 Satyam 标准中定义。国际电信联盟(ITU)于 1988 年推出这个标准，当时放在 X.500 标准中。后来，X.509 标准于 1993 年和 1995 年做了两次修订。这个标准的最新版本是 X.509V3。1999 年，Internet 工程任务小组(IETF)发表了 X.509 标准的 RFC2459，图 5.4 显示了 X.509V3 数字证书的结构。

图 5.4 不仅显示了 X.509 标准指定的数字证书字段，还指定了字段对应的标准版本。可以看出，X.509 标准第 1 版共 7 个基本字段，第 2 版增加了 2 个字段，第 3 版增加了 1 个字段。增加的字段分别称为第 2 版和第 3 版的扩展或扩展属性。当然，这些版本的末尾还有 1 个共同字段。图 5.5(a)、图 5.5(b) 与图 5.5(c) 列出了这些字段。第 2 版增加了两个字段，处理不小心重复签发者名(即 CA 名)和主体名(证书持有者名)的情形。但是，数字证

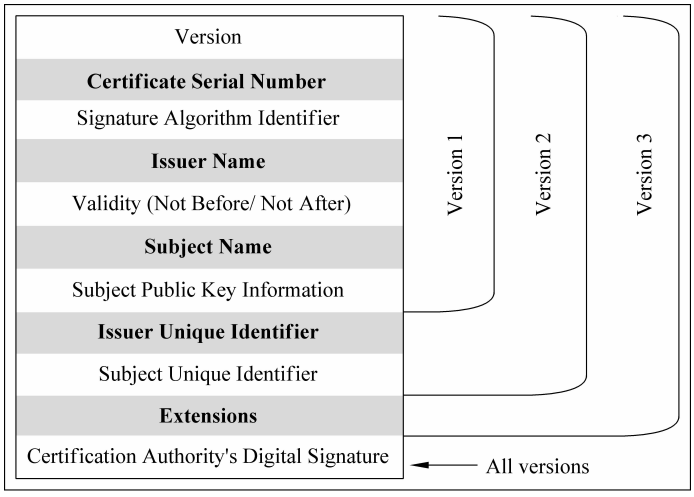


图 5.4　数字证书的结构

书标准(RFC2459)要求签发者名和主体名不能重复,因此尽管第 2 版增加了这些字段,但这些字段是可选的,使用时可以区别不小心重复的签发者名和主体名。

X.509V3 在数字证书结构中增加了许多扩展,如图 5.5(c)所示。

| 字　　段 | 描　　述 |
|---|---|
| 版本(Version) | 标识本数字证书使用的 X.509 协议版本,目前可取 1、2、3 |
| 证书序号(Certificate Serial Number) | 包含 CA 产生的唯一整数值 |
| 签名算法标识符(Signature Algorithm Identifier) | 标识 CA 签名数字证书时使用的算法(见稍后介绍) |
| 签发者名(Issuer Name) | 标识生成与签名数字证书的 CA 区别名(DN) |
| 有效期(之前/之后)(Validity (Not Before/Not After)) | 包含两个日期时间值(之前/之后),指定数字证书有效的时间范围。这些值通常指定日期与时间,精度到秒或毫秒 |
| 主体名(Subject Name) | 标识数字证书所指实体(即用户或组织)的区别名(DN)除非 V3 扩展中定义了替换名,否则这个字段必须有值 |
| 主体公钥信息(Subject Public Key Information) | 包含主体的公钥和与密钥相关的算法,这个字段不能为空 |

(a) X.509 数字证书字段描述——第 1 版

| 字　　段 | 描　　述 |
|---|---|
| 签发者唯一标识符(Issuer Unique Identifier) | 在两个或多个 CA 使用相同签发者名时标识 CA |
| 主体唯一标识符(Subject Unique Identifier) | 在两个或多个主体使用相同签发者名时标识 CA |

(b) X.509 数字证书字段描述——第 2 版

图 5.5　X.509 数字证书字段的描述

| 字　段 | 描　述 |
|---|---|
| 机构密钥标识符(Authority Key Identifier) | 一个证书机构可能有多个公钥/私钥对，这个字段定义这个证书的签名使用哪个密钥对(用相应密钥验证) |
| 主体密钥标识符(Subject Key Identifier) | 主体可能有多个公钥/私钥对(原因见稍后介绍)。这个字段定义这个证书的签名使用哪个密钥对(用相应密钥验证) |
| 密钥用法(Key Usage) | 定义这个证书的公钥操作范围。例如，可以指定这个公钥可用于所有密码学操作或只能用于加密，或只能用于Diffie-Hellman密钥交换，或只能用于数字签名，等等 |
| 扩展密钥用法(Extended Key Usage) | 可以补充或取代密钥用法字段，指定这个证书可以采用哪些协议(见稍后介绍)，这些协议包括TLS(传输层安全协议)、客户端认证、服务器认证、时间标志，等等 |
| 私钥使用期(Private Key Usage Period) | 可以对这个证书对应的公钥/私钥对定义不同的使用期限。如果这个字段是空的，则这个证书对应的公钥/私钥对定义相同的使用期限 |
| 证书策略(Certificate Policies) | 定义证书机构对某个证书指定的策略和可选限定信息，这里不进行介绍 |
| 证书映射(Policy Mappings) | 在某个证书的主体也是一个证书机构时使用，即一个证书机构向另一证书机构签发证书，指定认证的证书机构要循环哪些策略 |
| 主体替换名(Subject Alternative Name) | 对证书的主体定义一个或多个替换名，但如果主证书格式中的主体名字段是空的，则这个字段不能为空 |
| 签发者替换名(Issuer Alternative Name) | 可选定义证书签发者的一个或多个替换名 |
| 主体目录属性(Subject Directory Attributes) | 可以提供主体的其他信息，如主体电话/传真、电子邮件地址，等等 |
| 基本限制(Basic Constraints) | 表示这个证书的主体可否作为证书机构。这个字段还指定主体可否让其他主体作为证书机构。例如，如果证书机构X向证书机构Y签发这个证书，则X不仅能指定Y可否作为证书机构，向别的主体签发证书，还可以指定Y可否指定别的主体为证书机构 |
| 名称限制(Name Constraints) | 指定名字空间，这里不进行介绍 |
| 策略限制(Policy Constraints) | 只用于CA证书，这里不进行介绍 |

(c) X.509数字证书字段描述——第3版

图5.5 （续）

### 5.2.5 生成数字证书

#### 1. 涉及的各方

了解数字证书的概念和技术方面后，下面介绍生成数字证书的典型过程。参与各方是谁？要干些什么？我们知道，这个过程有两方要参与，即主体(最终用户)和签发者(证书机构)。证书生成与管理还涉及(可选)第三方。

由于证书机构的任务很多，如签发新证书、维护旧证书、吊销因故无效的证书，等等，因此可以将一些任务转交给第三方——注册机构(RA)。从最终用户角度看，证书机构与注册机构差别不大。技术上，注册机构是用户与证书机构之间的中间实体，帮助证书机构完成日常工作，如图 5.6 所示。

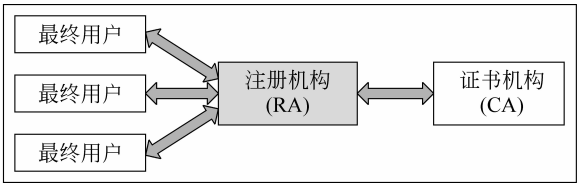


图 5.6　注册机构

注册机构通常提供下列服务：

- 接收与验证最终用户的注册信息；
- 为最终用户生成密钥；
- 接收与授权密钥备份与恢复请求；
- 接收与授权证书吊销请求。

在证书机构与最终用户间加上注册机构的另一重要影响是证书机构成为隔离实体，更不容易受到安全攻击。由于最终用户只能通过注册机构与证书机构通信，因此可以将注册机构与证书机构通信高度保护，使这部分连接很难被攻击。

但是，值得注意的是，注册机构主要是为了帮助证书机构与最终用户间交互，注册机构不能签发数字证书，证书只能由证书机构签发。此外，签发证书后，证书机构要负责所有证书管理工作，如跟踪证书状态，对因故无效的证书发出吊销通知，等等。

### 2. 证书生成步骤

生成数字证书需要几个步骤，如图 5.7 所示。

下面介绍这些证书生成步骤。

**■ 第 1 步：密钥生成**

首先是主体(用户/组织)要取得证书，可以使用两种方法。

(1) 主体可以用某个软件生成的公钥/私钥对，这个软件通常是 Web 浏览器或 Web 服务器的一部分，也可以使用特殊软件程序。主体要使生成的私钥保密，然后把公钥和其他信息与身份证明发送给注册机构，如图 5.8 所示。

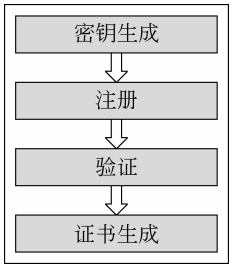


图 5.7　数字证书生成步骤

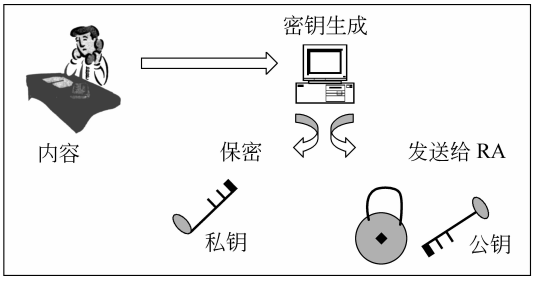


图 5.8　主体生成密钥对

（2）注册机构也可以为主体（用户）生成密钥对，可能用户不知道生成密钥对的技术，或特定情况要求注册机构集中生成和发布所有密钥，便于执行安全策略和密钥管理。当然，这个方法的主要缺点是注册机构知道用户的私钥，发送给用户时也可能中途暴露给别人，如图5.9所示。

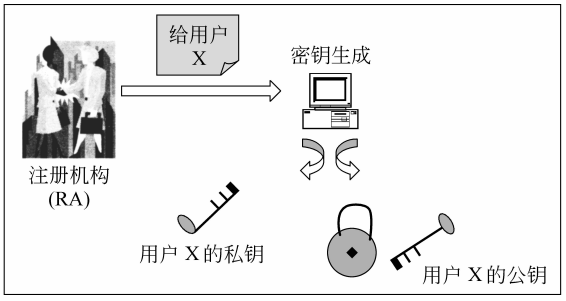


图5.9 注册机构为主体（用户）生成密钥对

■ **第2步：注册**

这一步只在第1步由用户生成密钥对时才需要。如果注册机构为主体（用户）生成密钥对，则这一步已经在第1步中完成。

假设用户生成密钥对，则要向注册机构发送公钥和相关注册信息（如主体名，要放在数字证书中）以及关于自己的所有证明材料。为此，软件提供了一个向导，可以让用户输入数据，并在所有数据正确时提交数据。然后数据通过网络。

Internet传递到注册机构。证书请求格式已经标准化，称为**证书签名请求**（Certificate Signing Request，CSR）。这是一个**公钥加密标准**（Public Key Cryptography Standards，PKCS），见稍后介绍。CSR也称为PKCS#10。

但是，证明材料不一定是计算机数据，有时是纸质文档（如护照、营业执照、收入/税收报表复印件等），如图5.10所示。

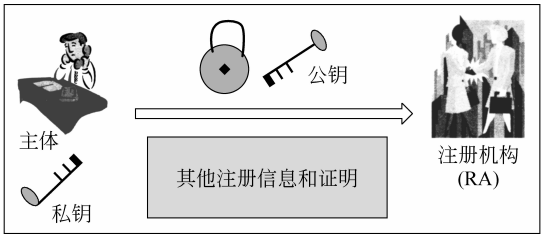


图5.10 主体将公钥与证明材料发给注册机构

注意，用户不能把私钥发给注册机构，而要将其保密。事实上，私钥最好根本不要离开用户的计算机。

图5.11显示了实际的数字证书请求页面。

然后，用户通常取得一个请求标识符，用于跟踪证书请求进展，如图5.12所示。

■ **第3步：验证**

注册过程完成后，注册机构要验证用户的材料，这个验证分为两个方面。

（1）首先，RA要验证用户材料，如提供的证据，保证它们可以接受。如果用户是一个组

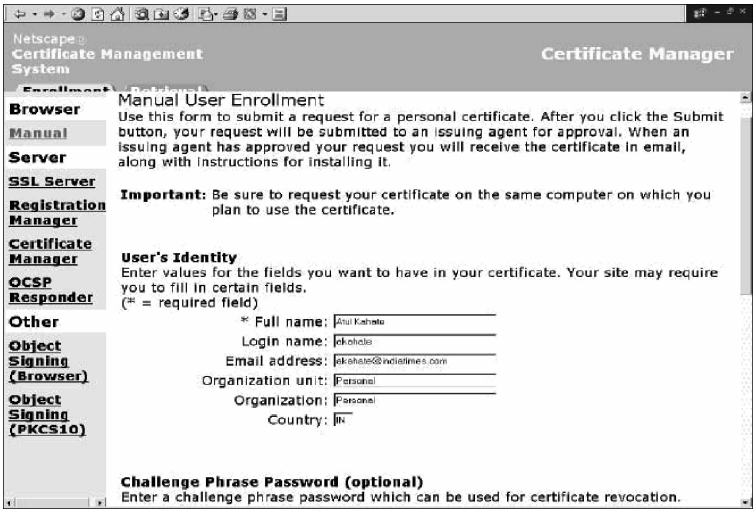


图 5.11　数字证书请求页面

织，则 RA 可能要检查营业记录、历史文件和信用证明。如果是个人用户，则只要简单证明就够了，如验证邮政地址、电子邮件地址、电话号码或者护照与驾照等。


图 5.12　CA 收到证书请求的收据

（2）第二个检查是保证请求证书的用户持有向 A 的证书请求中发送的公钥所对应的私钥。这一点很重要，因为我们必须证明用户拥有与这个公钥对应的私钥，否则可能造成法律问题（如果用户没有这个私钥，则用他的私钥签名的文档必然造成法律问题）。这个检查称为检查私钥的**拥有证明**（Proof Of Possession，POP）。RA 怎么进行这个检查呢？可以用许多方法，主要如下：

- RA 可以要求用户用私钥对证书签名请求进行数字签名。如果 RA 能用这个用户的公钥验证签名正确性，则可以相信这个用户拥有该私钥。

- 在这个阶段，RA也可以生成随机数挑战，用这个用户的公钥加密，将加密挑战发给用户。如果用户能用其私钥解密，则也可以相信这个用户拥有该私钥。
- 第三，RA可以对用户生成一个哑证书，用这个用户的公钥加密，将其发给用户。用户要解密这个加密证书才能取得明文证书。

■ **第4步：证书生成**

假设上述所有步骤成功，则RA把用户的所有细节传递给证书机构。证书机构进行必要的验证，并对用户生成数字证书。可以用程序生成X.509标准格式的证书。证书机构将证书发给用户，并保留一份证书记录。证书机构的证书记录放在**证书目录**（certificate directory）中，这是证书机构维护的中央存储地址。证书目录的内容与电话目录相似，帮助用一个点访问证书管理与发布。

没有一个描述证书目录结构的标准，但X.500标准已经成为普遍的选项，不仅可以存储数字证书，还可以以可控方式在中央地址存储服务器、打印机、网络资源和用户个人信息，如电话号码/分机号、电子邮件地址，等等。目录客户机可以用目录访问协议从这个中央仓储库请求和访问信息，如**轻量级目录访问协议**（Lightweight Directory Access Protocol，LDAP）。LDAP使用户和应用程序可以根据权限访问X.500目录。

然后证书机构将证书发给用户，可以附在电子邮件中，也可以向用户发一个电子邮件，通知其证书已生成，让用户从CA站点下载。后者如图5.13所示，用户访问一个屏幕，告知证书已生成，让用户从CA站点下载。

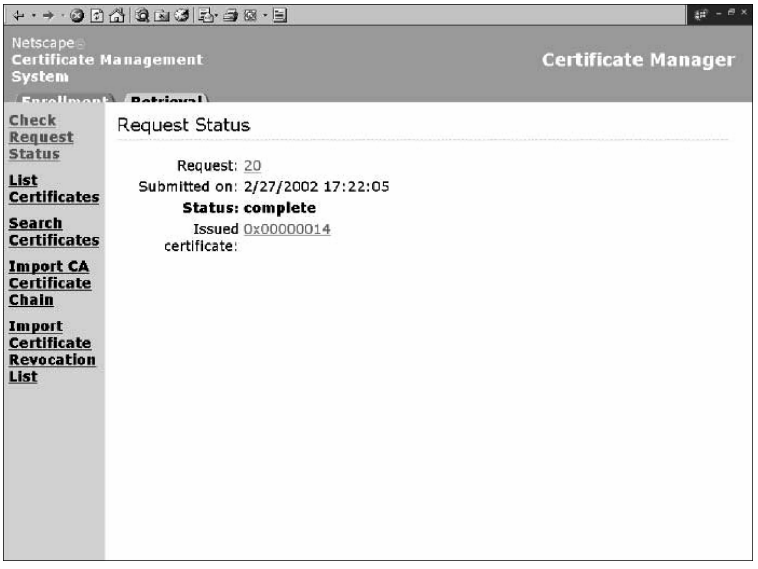


图5.13 CA告知证书已好，可以下载

用户下载证书时，得到图5.14所示屏幕，这个屏幕显示计算消息摘要和签名证书的版本、序号与算法，以及签发者、有效期、主体细节等。

数字证书的格式实际上是不可读的，如图5.15所示，从中看不出任何名堂，但应用程序可以分析或解释数字证书，以可读格式显示证书细节。

打开Internet Explorer浏览器（一种应用程序）浏览证书时，可以看到可读格式的证书

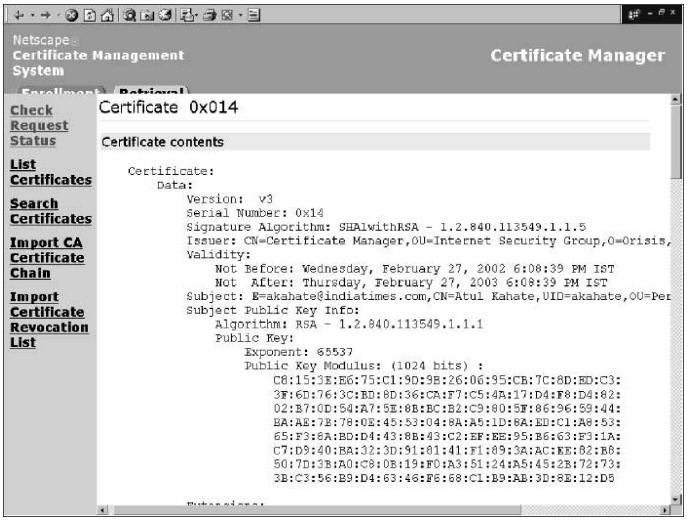


图 5.14　证书内容


图 5.15　数字证书的格式实际上是不可读的

细节，如图 5.16 所示。

## 5.2.6　为何信任数字证书

### 1. 简介

我们故意没有回答一个极其重要的问题，就是为何信任数字证书。要回答这个问题，想想为什么要信任护照。我们信任护照，不是因为其中包含关于护照持有者的信息，而是因为它是固定格式的，而且有权威机构的印章与签名。因此，信任数字证书不是因为其中包含用户的某些信息（特别是公钥），数字证书不过是个计算机文件，我们也可以用任何公钥生成数