计算机技巧百科

万能钥匙传奇

加密与解密实例教程

加密和PKI基础
数字签名算法分析与Hash签名
公开密匙体系
加密工具的使用
文件加密
网络数据加密
80x86保护模式
常见破解应用实例

珠海出版社

计算机技巧百科

万能钥匙传奇

加密 解密实例教程

珠海出版社

图书在版编目（CIP）数据

万能钥匙传奇：加密与解密实例教程/网垠科技编，
—珠海：珠海出版社，2001.9（2004.4 重印）
（计算机技巧百科）

ISBN 7－80607－819－3

Ⅰ.万... Ⅱ.网... Ⅲ.①电子计算机-密码-加密教材
②电子计算机-密码-解密译码-教材 Ⅳ.TP309.7

中国版本图书馆 CIP 数据核字（2004）第 029566 号

计算机技巧百科

责任编辑：雷良波
选题策划：网垠
封面设计：A 平方视觉工厂
出版发行：珠海出版社
社　　址：珠海市银桦路 566 号报业大厦三层
邮政编码：519002
电　　话：（0756）2639330
印　　刷：郑州市毛庄印刷厂
开　　本：889×1194mm　1/16
印　　张：166
字　　数：3320 千字　　**印数**：10000~11000 册
版　　次：2004 年 5 月第 1 版第 2 次印刷
书　　号：ISBN 7－80607－819－3/TP・8
定　　价：200.00 元（全十册）

卷首语

当今计算机技术的发展日新月异，随着软件以共享方式在网络上发布的流行，软件保护和数据加密技术的迫切性也越来越突出地表现出来。一款优秀的软件，其技术秘密往往成为他人窃取的重点。作为软件开发人员，为了保护自己辛辛苦苦开发的软件不会轻易被他人“借鉴”，有必要了解软件的加密和破解技术。

加密与解密是相辅相成，不断发展的。本书编写以实用为目的，从保护（加密）和破解（解密）两方面进行详细而透彻的讲解，力求使读者能尽快掌握加密与解密技术。

第1章 加密和PKI基础知识，介绍了加密的概念，公钥基本结构的概念；**第2章 数字签名算法分析与Hash签名**，介绍了数字签名的实现方法；**第3章 公开密钥体系**，介绍了公开密钥算法-RSA、RSA算法和DES算法、椭圆曲线密码算法；**第4章 分组密码算法分析与改进**，介绍了分组密码-DES、IDEA算法及其他分组密码；**第5章 加密工具使用教程**，介绍了ABI-Coder、加密软件PGP、电子邮件加密工具A-lock、文件加密利器Fedt、情书加密Power Crypto等的使用方法；**第6章 文件加密**，介绍了隐藏高手Hide In Picture的使用、如何锁住图片、有效的软件锁、秘密信使、密码保管箱；**第7章 网络数据加密**，介绍了网络数据加密的三种技术以及常见的网络加密；**第8章 80x86保护模式教程**，介绍了分段管理机制、控制寄存器和系统地址寄存器、实模式与保护模式切换实例、任务状态段和控制门、控制转移、中断和异常、操作系统类指令、输入/输出保护、分页管理机制、虚拟8086模式；第9章介绍了常见破解实例。最后附2003常用加密工具软件。

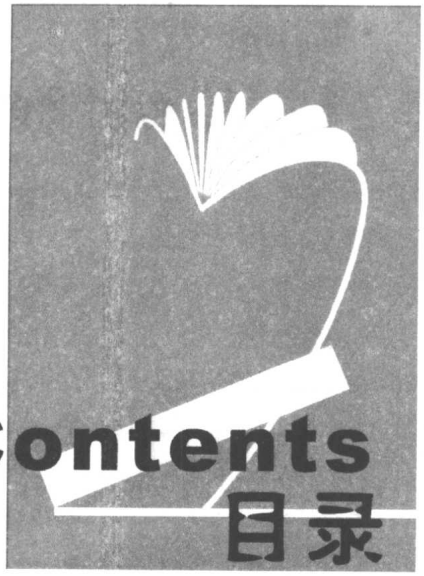
使用本书最好具备以下知识：汇编基础知识；C语言（不是必须）；Win32编程。不管研究加密与解密，还是编程，都必须了解Win32编程。Win32编程就是API方式的Windows程序设计。

本书适合对软件加密与解密、逆向工程或调试技术感兴趣的读者参考学习，也是软件开发人员不可多得的读物。

内容提要

加密与解密是软件开发过程中的重要环节。本书为照顾不同层次的读者，对内容作了精心编排，使从未接触过汇编和对编程知之甚少的读者看完本书后也能快速入门。主要内容包括：加密和PKI基础知识、数字签名算法分析与Hash签名、公开密钥体系、分组密码算法分析与改进、加密工具使用教程、文件加密、网络数据加密、80x86保护模式教程、常见破解实例。

本书适合对软件加密与解密、逆向工程或调试技术感兴趣的读者参考学习，也是软件开发人员不可多得的读物。



Contents 目录

第一章 加密和PKI基础知识

第二章 数字签名算法分析与Hash签名

Contents 目录

第三章 公开密钥体系

第四章 分组密码算法分析与改进

Contents
目录

Contents 目录

Contents
目录

Contents 目录

第九章　常见破解实例

第一章　加密和PKI基础知识

本章介绍了加密和公钥基本结构（PKI）的概念和使用Microsoft Windows 2000 Server操作系统中的证书服务的基础知识。

在Microsoft Windows 2000 Server的证书服务中提供了集成公钥基本结构（PKI），目的是使电子商务能够在安全的环境中进行。理解加密和PKI的相关概念是理解证书服务功能的先决条件，证书服务是Microsoft Windows 2000 Server操作系统中的一个组件。

1.1　加密概念

加密是通过Intranet、Extranet和Internet进行安全的信息交换的基础。从业务的角度来看，通过加密实现的安全功能包括：身份验证（使收件人确信发件人就是他或她所声明的那个人）、机密性（确保只有收件人能够阅读邮件）、完整性（确保邮件在传输过程中没有被更改）。从技术的角度来看，加密是利用数学方法将邮件转换为不可读格式从而达到保密数据的目的的一门学科。

本节介绍下列加密概念：

- 对称密钥加密：一个密钥
- 公钥加密：两个密钥
- 单向散列算法
- 数字签名：结合使用公钥与散列
- 密钥交换：结合使用对称密钥与公钥

在前三个小节中分别定义并说明对称密钥加密、公钥加密和单向散列算法。后两个小节中说明组合使用这些技术的方法，其中尤其重要的是将公钥算法与散列算法相结合以创建数字签名，以及将对称算法与公钥算法相结合使交换密（私）钥成为可能。

1.1.1　对称密钥加密：一个密钥

对称密钥加密也叫做共享密钥加密或机密密钥加密，使发件人和收件人共同拥有单个密钥。这种密钥既用于加密，也用于解密，叫做机密密钥（也称为对称密钥或会话密钥）。对称密钥加密是加密大量数据的一种行之有效的方法。

对称密钥加密有许多种算法，但所有这些算法都有一个共同的目的，就是以可还原的方式将明文（未加密的数据）转换为暗文。暗文使用加密密钥编码，对于没有解密密钥的任何人来说，它都是没有意义的。由于对称密钥加密在加密和解密时使用相同的密钥，因此这种加密过程的安全性取决于是否有未经授权的人获得了对称密钥。这就是它为什么也叫做机密密钥加密的原因。希望使用对称密钥加密通信的双方在交换加密数据之前，必须先安全地交换密钥。

衡量对称算法优劣的主要尺度是其密钥的长度。密钥越长，在找到解密数据所需的正确密钥之前必须测试的密钥数量就越多。需要测试的密钥数量越多，破解这种算法就越困难。有了好的加密算法和足够长的密钥后，如果有人想在一段实际可行的时间内逆转转换过程，并从暗文中推导出明文，从计算的角度来

讲，这种做法是行不通的。

1.1.2 公钥加密：两个密钥

公钥加密使用两个密钥，其中一个是公钥，另一个是私钥，这两个密钥在数学上是相关的。为了与对称密钥加密相对称，公钥加密有时也叫做不对称密钥加密。在公钥加密中，公钥可在通信双方之间公开传递，或在公用储备库中发布，但相关的私钥是保密的。只有使用私钥才能解开用公钥加密的数据。使用私钥加密的数据只能用公钥解密。发件人拥有收件人的公钥，并用它加密了一封邮件，但只有收件人掌握解密该邮件的有关私钥。

与对称密钥加密相似，公钥加密也有许多种算法。然而，对称密钥和公钥算法在设计上并无相似之处。你可以在程序内部使用一种对称算法替换另一种，而变化也不是很大，因为它们的工作方式是相同的。而另一方面，不同公钥算法的工作方式完全不同，因此它们之间不可以互换。

公钥算法是复杂的数学方程式，使用十分大的数字。公钥算法的主要局限在于，这种加密形式的速度相对较低。实际上，通常仅在关键时刻才使用公钥算法，如在实体之间交换对称密钥或者在签署一封邮件的散列（散列是通过应用一种单向数学函数获得的一个定长结果，对于数据而言，叫做散列算法）。将公钥加密与其他加密形式（如对称密钥加密）结合使用，可以优化性能。公钥加密提供了一种有效的方法，可用于为大量数据执行对称加密，使用机密密钥发送给某人。也可以将公钥加密与散列算法结合使用以生成数字签名。

若要进一步了解关于将公钥加密、对称密钥加密或散列算法结合使用的信息，请参考后面两小节：“数字签名：结合使用公钥与散列算法”和“密钥交换：结合使用对称密钥与公钥”

1.1.3 将公钥加密用于数字签名

数字签名是邮件、文件或其他数字编码信息的发件人将他们的身份与信息绑定在一起（即为信息提供签名）的方法。对信息进行数字签名的过程需要将信息与发件人掌握的秘密信息一起转换为签名的标记。数字签名用于公钥环境中，它通过验证发件人确实是他或她所声明的那个人，并确认收到的邮件与发送的邮件完全相同，以确保电子商务交易的安全。

通常，数字签名用于以明文（如电子邮件）分发数据的情形。在这种情况下，当邮件本身的敏感性可能无法保证加密的安全性时，确保数据处于其原始格式且并非由假冒者发送是非常重要的。

要了解如何结合使用公钥与散列算法来创建数字签名，请参考“数字签名：结合使用公钥与散列算法”一节。

1.1.4 常用公钥算法

下面是三种最常用的公钥算法：

● RSA适用于数字签名和密钥交换。Rivest-Shamir-Adleman（RSA）加密算法是目前应用最广泛的公钥加密算法，特别适用于通过Internet传送的数据。这种算法以它的三位发明者的名字命名：Ron Rivest、Adi Shamir和Leonard Adleman。RSA算法的安全性基于分解大数字时的困难（就计算机处理能力和处理时间而言）。在常用的公钥算法中，RSA与众不同的，它能够进行数字签名和密钥交换运算。Microsoft Base Cryptographic Service Provider（Microsoft Base CSP1）支持RSA加密算法，并且Microsoft Enhanced Cryptographic Service Provider（Microsoft Enhanced CSP2）已经内置到包括 Microsoft Internet Explorer在内

的许多软件产品中。

● DSA仅适用于数字签名。数字签名算法（Digital Signature Algorithm, DSA）是由美国国家安全署（United States National Security Agency，NSA）发明的，已经由美国国家标准与技术协会（National Institute of Standards and Technology，NIST）收录到联邦信息处理标准（Federal Information Processing Standard，FIPS）之中，作为数字签名的标准。DSA算法的安全性源自计算离散算法的困难。这种算法仅用于数字签名运算（不适用于数据加密）。Microsoft CSP支持DSA算法。

● Diffie-Hellman仅适用于密钥交换。Diffie-Hellman是第一个被发明的公钥算法，以其发明者Whitfield Diffie和Martin Hellman的名字命名。Diffie-Hellman算法的安全性源自于一个有限字段中计算离散算法的困难。Diffie-Hellman算法仅用于密钥交换。Microsoft Base DSS 3和Diffie-Hellman CSP都支持Diffie-Hellman算法。

1.1.5 单向散列算法

散列也称为散列值或消息摘要，是一种与密钥（对称密钥或公钥）的加密不同的数据转换类型。散列就是通过把一个叫做散列算法的单向数学函数应用于数据，将任意长度的一块数据转换为一个定长的、不可逆转的数字。所产生的散列值的长度应足够长，因此找到两块具有相同散列值的数据的机会很少。发件人生成邮件的散列值并加密它，然后将它与邮件本身一起发送。而收件人同时解密邮件和散列值，并由接收到的邮件产生另外一个散列值，然后将两个散列值进行比较。如果两者相同，邮件极有可能在传输期间没有发生任何改变。

1.1.6 常用的单向散列函数

下面是两个最常用的散列函数：

● MD5：MD5是由Ron Rivest设计的可产生一个128位的散列值的散列算法。MD5设计经过优化用于Intel处理器。这种算法的基本原理已经泄露，这就是为什么它不太受欢迎的原因。

● SHA-1：与DSA公钥算法相似，安全散列算法1（SHA-1）也是由NSA设计的，并由NIST将其收录到FIPS中，作为散列数据的标准。它可产生一个160位的散列值。SHA-1是流行的用于创建数字签名的单向散列算法。

1.1.7 数字签名：结合使用公钥与散列算法

可以结合使用公钥技术与散列算法来创建数字签名。数字签名可用于数据完整性检查并提供拥有私钥的凭据。

签署和验证数据（由启用PKI的应用程序如Microsoft Outlook完成）的步骤如下：

● 发件人将一种散列算法应用于数据，并生成一个散列值。
● 发件人使用私钥将散列值转换为数字签名。
● 发件人将数据、签名及发件人的证书发给收件人。
● 收件人将该散列算法应用于接收到的数据，并生成一个散列值。
● 收件人使用发件人的公钥和新生成的散列值验证签名。

对用户而言，这一过程是透明的。

散列算法处理数据的速度比公钥算法快得多。散列数据还缩短了要签名的数据的长度，因而加快了签

名过程。当创建或验证签名时，公钥算法必须转换散列值（128或160位的数据）。创建签名和验证签名的详细步骤取决于所采用的公钥算法。

1.1.8 密钥交换：结合使用对称密钥与公钥

对称密钥算法非常适合于快速并安全地加密数据。但其缺点是，发件人和收件人必须在交换数据之前先交换机密密钥。结合使用加密数据的对称密钥算法与交换机密密钥的公钥算法可产生一种既快速又灵活的解决方案。

基于公钥的密钥交换步骤如下：

- 发件人获得收件人的公钥。
- 发件人创建一个随机机密密钥（在对称密钥加密中使用的单个密钥）。在Windows 2000中，CryptoAPI4可用于创建机密密钥（有关CryptoAPI的详细信息，请参见1.2.10节。）
- 发件人使用机密密钥和对称密钥算法将明文数据转换为暗文数据。
- 发件人使用收件人的公钥将机密密钥转换为暗文机密密钥。
- 发件人将暗文数据和暗文机密密钥一起发给收件人。
- 收件人使用其私钥将暗文机密密钥转换为明文。
- 收件人使用明文机密密钥将暗文数据转换为明文数据。

同样，这些步骤是由启用 PKI 的应用程序（如 Microsoft Outlook）来完成的，并且对用户来说是透明的。

1.2 公钥基本结构的概念

公钥基本结构（PKI）用于描述管制或操纵证书与公钥及私钥的策略、标准和软件。实际上，PKI是指由数字证书、证书颁发机构（CA）以及对电子交易所涉及各方的合法性进行检查和验证的其他注册机构组成的一套系统。PKI的有关标准仍处于不断地发展之中，即使这些标准已被作为电子商务的要素而广泛实施。

本节帮助你理解什么是 PKI 以及创建 PKI 需要哪些服务。这些 PKI 概念将在以下几个小节中讨论：

- 证书
- 证书颁发机构（CA）
- 不可更改的CA层次结构
- 注册
- 证书登记
- 证书吊销
- 证书链验证

1.2.1 证书

公钥证书通常简称为证书，用在Internet、Extranet和Intranet上进行身份验证并确保数据交换的安全。证书的颁发者和签署者就是众所周知的证书颁发机构（CA），将在下一小节中介绍。颁发证书的实体是证书的主体。

公钥证书是以数字方式签名来声明的，它将公钥的值与持有相应私钥的主体（个人、设备和服务）的身份绑定在一起。通过在证书上签名，CA可以核实与证书上公钥相应的私钥为证书所指定的主体所拥有。

可以为各种目的颁发证书，如Web用户身份验证、Web服务器身份验证、使用安全/多用途 Internet邮件扩充协议（Secure/Multipurpose Internet Mail Extensions，S/MIME）的安全电子邮件、IP安全性（IP Security）、安全套接字协议层/事务层安全性（Secure Sockets Layer/Transaction Layer Security，SSL/TLS）和代码签名。如果在一个组织内部使用Windows 2000企业证书颁发机构，证书可用于登录到Windows 2000域。证书还可以由一个CA颁发给另一个CA，以建立证书层次结构。

可以通过多个名称来识别主体，如：用户主要名称（用于最终用户证书）、目录名、电子邮件名称和DNS域名等。证书还应包含下列信息：

- 证书的有效期。
- 证书的序列号，CA 应保证该序列号是惟一的。
- CA 的名称以及用于签署该证书的密钥。
- CA 所遵循的用来确定证书主体身份的策略的标识符。
- 在证书中标识的密钥对（公钥及相关的私钥）的用法。
- 证书吊销列表（CRL）的位置，这是一个由CA维护并发布的列出已被吊销的证书的文档。为确保其完整性，CRL是用CA的私钥签署的。

证书提供了一个在公钥和拥有相应私钥的实体之间建立关系的机制。目前最常用的证书格式通过ITU-T X.509版本3（X.509v3）国际标准定义。RFC 2459是X.509v3的一个配置文件，进一步阐明了X.509v3中定义的字段。Windows 2000 PKI采用X.509v3标准。Windows证书是按照RFC 2459 中的说明编程的，但仍然叫做X.509v3证书。

ITU-T X.509并非证书的惟一格式。例如，Pretty Good Privacy（PGP）安全电子邮件是依赖PGP所独有的一种证书。

1.2.2 证书颁发机构

证书颁发机构（CA）是一个向个人、计算机或任何其他申请实体颁发证书的可信实体。CA受理证书申请，根据该CA的策略验证申请人的信息，使用它的私钥把其数字签名应用于证书。然后，CA将该证书颁发给证书的主体，作为PKI内部的安全凭据。由于不同的CA使用不同的方法验证公钥与主体之间的绑定，在选择信任该颁发机构之前，理解该CA的策略是非常重要的。

CA可以是远程的第三方机构，如VeriSign。作为选择，也可以是你创建的供你所在组织使用的 CA，例如，通过安装Windows 2000证书服务即可创建一个CA。每个CA对证书申请人可能有完全不同的身份凭据要求，如Windows 2000域帐户、职员标记、驾驶执照、公证请求、实际住址等。

1.2.3 CA 策略

CA根据已确立的一套标准向申请人颁发证书。CA在受理证书请求（以及颁发证书、吊销证书和发布CRL）时所采用的一套标准被称为CA策略。通常，CA以一种叫做证书惯例声明（Certification Practice Statement，CPS）的文档发布其策略。

不应将CA策略与Windows 2000“组策略”相混淆，后者通常与域帐户和应用程序部署服务（如IntelliMirror）相关联。

1.2.4 证书颁发机构的类型

CA 的类型包括以下三种：

● 自签名CA：在自签名CA中，证书中的公钥和用于验证证书的密钥是相同的。一些自签名CA是根CA。

● 从属CA：在从属CA中，证书中的公钥和用于核实证书的密钥是不同的。一个CA向另一个CA颁发证书的过程叫做交叉认证。

● 根CA：根CA是一种特殊的CA，它受到客户无条件的信任，位于证书层次结构的最高层。所有证书链均终止于根CA。根颁发机构必须对它自己的证书签名，因为在证书层次结构中再也没有更高的认证机构了。

所有自签名CA都是根CA，因为到自签名CA时证书链就终止了。

Windows 2000只能指定一个自签名CA为根CA。将一个CA指定为根CA的决策由个人在企业级或本地做出。

1.2.5 不可更改的CA层次结构

管理员可以创建CA的层次结构，从根CA证书开始，然后添加中级CA，每一个CA都可以为其从属CA颁发证书。当CA向最终实体（用户）颁发证书时，证书链就终止了。

根CA证书的分发费用最高，因为如果你开始改变根证书，就必须重建整个PKI。如果根证书改变了，就必须吊销组织内所有客户端的旧的根证书，并添加新的根证书。另外，必须重新颁发由根CA颁发的、再由从属CA颁发给最终实体的所有证书。因此，在部署CA层次结构时，使用少量的长寿命根CA可提供最经济的解决方案。根CA非常重要，因为它们被无条件地信任，因为它们是证书链的顶点。因此，在分发证书时要有一个圈外的身份验证。也就是说，由于根CA是自签名的，所以必须有人来证明根证书是真品。

因为最终实体要比CA多得多，所以向最终实体颁发证书的CA使用私钥在大量的数据上签名。用来对数据签名的密钥使用得越频繁，加密数据受到攻击的可能性就越大。因此，为了保持安全，向最终实体颁发证书的联机CA必须经常更换其签名密钥。

向最终实体颁发证书的CA具有的吊销证书列表要比中级或根CA的列表大得多（这些CA仅向其他CA，更多的是从属CA颁发证书）。其部分原因是因为最终实体要比CA证书多得多。另外，有许多理由可以解释为什么必须吊销最终实体的证书，如：职员改变了工作或离开了公司。

CA发布吊销证书列表（CRL），其中列出了不应再使用的证书。被吊销证书的有关条目将一直保留在CRL列表中，直至证书的有效期结束之后，CA才可将该证书从列表中删除。CRL中的条目越多，CRL就越大，其下载时间就越长。通常，只有使用较慢的网络链路（如拨号连接）的用户才会遇到下载时间问题。CA还可以管理CRL列表的大小。一种方法是维护多个列表，称为分区CRL；另一种方法是缩短已颁发证书的有效期，从而加快CA从列表中删除吊销证书的速度（关于CRL的详细信息，请参见1.2.9节）。

许多应用程序必须能够查明证书最近的吊销状态信息。只有一个联机CA能够发布有关证书状态的当前信息。由脱机CA公布的吊销状态必须使用圈外的方法发布到联机位置。

大多数容易受到攻击的CA都是处于联机状态的、物理安全措施较差并签署了大量证书的CA。因此，建立根CA和从属CA时，应该平衡一下安全性和可用性。通常，建议采用三级层次结构，即一个脱机的独立根CA、一个脱机的独立从属策略CA和一个联机从属颁发企业CA。

● 脱机根CA：在设计CA的层次结构时，根CA的安全级别应设为最高。根CA应以脱机状态保存在安