

Internet 和 数据库加密

Cryptography for Internet and Database Applications



[美] Nick Galbreath 著

曾振宇 白克壮 尹 喆 等译

Internet 和数据库加密

Cryptography for Internet and Database Applications

[美] Nick Galbreath 著

曾振宇 白克壮 尹 喆 等译

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书介绍了如何用Java的加密技术提高数据的安全性。重点介绍了存储、消息完整性和身份验证密码学。全书包括7章:第1~4章介绍了密码学的基础知识,第5章介绍了Java密码API,第6章介绍了短消息的编码和加密,第7章是前面6章内容的实际应用。

本书的特点是用简单实用的示例程序来阐述理论问题和实现方法,并为数据库和Web应用程序的安全提供了详细的Java语言实现指南和代码示例。全书内容翔实、清晰、概念性强、论述深入浅出,适合于从事密码系统研究的软件工程师阅读,也可作为大专院校相关专业的教学参考书。



Copyright ©2002 by Publishing House of Electronics Industry. Original English language edition copyright ©2002 by Wiley Publishing, Inc. All rights reserved including the right of reproduction in whole or in part in any form. This translation published by arrangement with Wiley Publishing, Inc.

本书中文简体专有翻译出版权由美国Wiley Publishing, Inc.授予电子工业出版社及其所属今日电子杂志社。未经许可,不得以任何手段和形式复制或抄袭本书内容。该专有出版权受法律保护,侵权必究。

著作权合同登记号 图字:01-2002-5162

图书在版编目(CIP)数据

Internet和数据库加密/(美)加尔布雷恩(Galbreath,N.)著;曾振宇等译.

—北京:电子工业出版社,2003.6

书名原文:Cryptography for Internet and Database Applications

ISBN 7-5053-8760-X

I.I... II.①加...②曾... III.①因特网-基本知识②因特网-安全技术 IV.TP393.4

中国版本图书馆CIP数据核字(2003)第040379号

责任编辑:王春宁

印 刷:北京大中印刷厂

出版发行:电子工业出版社 www.phei.com.cn

北京市海淀区万寿路173信箱 邮编:100036

经 销:各地新华书店

开 本:787×980 1/16 印张:22.75 字数:524千字

版 次:2003年6月第1版 2003年6月第1次印刷

定 价:39.00元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系调换。联系电话:(010)88211980 68279077

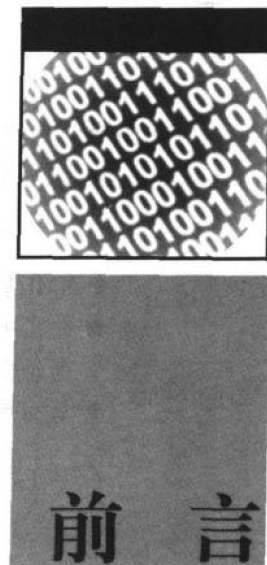
译 者 序

Java从版本 1.1 开始引入加密体系结构 (JCA), 它是访问和开发 Java 平台密码功能的框架, 其中包括数字签名和消息摘要 API。版本 1.2 对该结构进行了较大的扩展。Java 密码扩展 (JCE) 是一组包, 提供了加密框架及其实现、密钥生成和消息认证码 (MAC) 的算法, 现已成为 Java SDK 1.4 的核心组成部分。

本书介绍了如何用 Java 的加密方法提高数据的安全性, 重点介绍了存储、消息完整性和身份验证密码学。作者 Nick Galbreach 是一位著名的安全问题专家, 曾长期从事敏感信息存储和转换的加密策略研究, 相信对欲了解 Java 密码 API 的读者会有所帮助。

与纯粹的密码学书籍不同, 本书用简明实用的示例代码来阐述理论问题和具体的实现方法, 并为数据库和 Web 应用程序的安全提供了详细的 Java 语言实现指南和代码示例。全书包括 7 章。第 1 章介绍了基本的数值理论, 这一章是理解全书内容的基础。第 2 章和第 3 章介绍了公开密钥和保密密钥密码学标准。第 4 章所介绍的随机数和随机数发生器是密码学的核心。第 5 章介绍了 Java 密码 API, 密码包中的类按字母顺序 (而不是包顺序) 列在了附录 A 中。第 6 章介绍了短消息的编码和加密, 这对数据库的查找和索引很有意义。第 7 章是前面几章内容的实际运用, 介绍了如何在应用程序中使用和实现数据编码和加密, 内容包括应用程序和数据库设计、口令和记号的使用、密钥管理和日志等等。

本书主要由曾振宇、白克壮、尹喆、纪宁、梅开、文达、杨开开、成文、颜先杰、任映梅翻译, 参与翻译的还有苏泳民、单力、汪梓鸣、魏莲方、戚叶等, 梁敏、史荣光、李昕怡对本书进行了全面的审校。不当和疏漏之处敬请广大读者批评指正。



此书是专为那些接触过一点或没接触过密码学的软件工程师编写的。在此，作者尽量避免写成本书的天书。这类书皆容易陷入两种误区：百科全书式的和记述式的或者纯粹是API式的描述。作者的目的是通过对密码学的整体介绍，试图在两者之间架起桥梁，同时提供应用实例和用法来达到这一目的。另外，许多书都只是大量地关注公开密钥技术，而以作者的经验，公开密钥密码系统最常用的是第三方应用程序（虚拟专用网、电子邮件）或现行的协议（如SSL和SSH）。

虽然在C和C++语言中有许多优秀的密码库，然而Java扮演着参考书目的角色，因为：

- ◆ 在新型商务和服务端应用程序中非常流行。
- ◆ 自动管理内存，消除了各种类型的错误（堆栈破碎和缓冲区溢出）。
- ◆ 它提供了标准密码API，虽然不完美，但这是我们能够得到的“通用”API。

Java密码API分散在几个包中。附录A按字母顺序列出了这些类，而不是像通常的做法那样按包列出这些类。作者发现这比在不同的包间翻来翻去有用多了。作者试图对示例源代码加以限制，使它相对简单或能说明要点或本身就是实用程序。更复杂（也可能更有用）的示例没有被采用，这是因为对一个安全应用程序

来说, 错误和异常处理是相当烦琐的, 并且实际上没有增加多少价值。作者一直不喜欢将 CD-ROM 中的内容大量写进书中, 同样, 也从没发现源代码章节对说明问题特别有用。相关的 Web 站点 (www.wiley.com/compbooks.galbreath) 提供了全部的源代码和更多的示例。

由于时间和篇幅的限制, 许多主题不能讨论。特别是:

- ◆ XML 密钥管理、加密和签名模式。
- ◆ 秘密项目, 包括秘密“密封”、Gramm-Leech-Bliley 1999 法案和 HIPPA 保密条例。
- ◆ 具有更多细节性的数据库技巧和技术。

希望在后续版本的书中会详细论述这些主题。



绪 论

本书的目的是针对 Java 密码学及其应用，向系统程序员做一个全面的介绍。与诸多课本不同，此书重点不在“传输”（让两方秘密通信）密码学上。对于大多数应用程序来说，那些需求可由 SSL 或 SSH 协议处理，或者由第三方应用程序解决。本书把重点放在存储、消息的完整性和身份验证密码学上（即所谓的“单点”技术，它在客户应用程序中用得更为普遍）。除了纯密码学外，本书还介绍了许多使该密码学有用的“辅助”技术。

第 1 章论述了基本逻辑和数值运算，对两者的介绍都局限在理论层面上，并有 Java 数值模型的规范。对许多读者来说，这可能是个温习，但我发现许多系统程序员恰恰很少接触此类底层细节。他们或者从没学过这方面内容，或者更可能是因为它不是每天工作常用的，所以被遗忘了。那些对 C 和 C++ 语言更熟悉的读者会发现，第 1 章对于翻译 Java 代码非常有用。

接下来的两章介绍科学、数学以及密钥和公钥密码学的标准。在这里读者将会学到为什么算法可工作以及它们之间的种种运算变换。虽然数学不是日常编程任务常用的，但目的是让读者熟悉常出现的可能影响信息判断的词汇。

第 4 章讨论随机数和随机数生成器。虽然从技术上讲不是密码学，但随机数是它的本质。那些开发游戏系统的读者会发现，此章内容非常有用。算法示例是用 Java 编写的，但也很容易用其他编程语言实现。

第 5 章将介绍 Java 密码 API。Java SDK 已经带有一些很好的文档，但是我想

655194107

把它们整理一下，并将 JCA 和 JCE 合并成协调的形式。同样，当编程需要时，我制作了附录 A —— Java 密码类参考。附录中的类按字母顺序（而不是包）排列。可以发现，这种组织方式使读者不必总是在各个包间翻来翻去，并且可以更好地理解 API。

第6章介绍短消息编码。像第4章一样，从技术上说，它不是密码学范畴，但当它涉及到密码时总是个问题。在此，读者可以学到如何将二进制数据（也许加密了）转换成各种 ASCII 码形式；当将数据嵌入 URL、生成密码和密码记号时，这很重要。

第7章将所有内容综合在一起，并讨论许多主题：应用程序和数据库设计、密码和记号的使用、密钥管理和日志。

本书使用了大量的源代码例子，可以在 Web 站点 www.modp.com 查到这些源代码。本书中许多例子需要稍加修改以后才能作为产品使用，读者也肯定愿意按自己的需要修改错误。某些情况下，如果代码太长也不很明了，那么我不在书中列出，而是让读者到 Web 站点查看。我发现一页一页地印源代码也不是特别有意义。

遗憾的是，由于时间和篇幅的限制，许多重要主题不能一一包括。数字签名也只是提了一下，虽然它值得写得更多些，但我发现，对于基本应用程序来说，数字签名不是很有用。或者因为大多数人不愿使用数字签名或已经存在信任关系，或者因为人们更愿意用散列或 MAC 进行加密。最后，当我确实需要使用数字签名时，通常可以用第三方应用程序来处理它。还有其他主题也没有得到应有的论述，包括用来创建包含加密数据文档的新 XML 和 SAML 标准、数据库中的嵌入式密码（如最新的 Oracle 数据库可做到这一点）。密钥管理和数据库设计也可多加论述，也许将来的版本会加以补救。

目 录

前言

绪论

第 1 章 位和字节	1
1.1 一般运算	1
1.1.1 数的基数	1
1.1.2 位和字节	3
1.1.3 有符号字节	3
1.1.4 位运算符	4
1.1.5 字群压缩	9
1.1.6 整数和结尾表示法	10
1.2 Java 数值	11
1.2.1 基本类型	11
1.2.3 使用字节	19
1.2.4 BigInteger	24
第 2 章 保密密钥	29
2.1 对称分组密码	29
2.1.1 密码特性	29
2.1.2 常用的分组密码	32
2.1.3 不应使用的密码	39
2.1.4 填充	41
2.1.5 运算模式	42
2.1.6 电子密码本模式 (ECB).....	43
2.1.7 密码分组链接模式 (CBC).....	44
2.1.8 密钥包装	50

2.1.9 把密码转变成密钥	52
2.2 散列	53
2.2.1 密码散列	54
2.2.2 运算法则	57
2.2.3 散列函数标准和实践	60
2.3 散列式信息鉴定代码 (HMAC)	61
2.3.1 标准 HMAC	62
2.3.2 HMAC 标准和实践	63
2.4 小结	63
第 3 章 公开密钥	65
3.1 公开密钥密码	65
3.1.1 其他系统	66
3.2 公开密钥安全性分类	67
3.3 数学基础	69
3.3.1 素数	69
3.3.2 初等数值理论	74
3.4 公开密钥加密和主要 PKCS 分类	76
3.4.1 RSA 和整数因子分解	77
3.4.2 离散对数系统	84
3.4.3 椭圆曲线	89
3.4.4 其他公开密钥密码系统	95
3.5 小结	97
第 4 章 随机数	99
4.1 随机和安全性	101
4.1.1 随机性实验	102
4.2 伪随机数发生器	103
4.2.1 密码系统 PRNG	103
4.2.2 流密码	105
4.3 使用随机性	106
4.3.1 游戏随机数的生成	106
4.3.2 生成某个范围内的随机数	107
4.3.3 洗牌	109
4.3.4 生成随机排列	111

4.3.5 随机取样	112
4.4 访问熵	114
4.4.1 操作系统服务	114
4.4.2 “用户方”服务	116
4.4.3 TrueRand 库	120
4.4.4 远程服务	120
4.5 Java 和随机数	122
4.5.1 类 Random 和 SecureRandom	122
4.5.2 开发者问题	125
4.5.3 重新设置种子	126
4.5.4 收集熵	127
第5章 Java 密码学	137
5.1 组织方式	138
5.1.1 提供者和引擎类	139
5.1.2 参数、密钥和证书	140
5.1.3 错误处理	140
5.2 提供者	141
5.2.1 标准名称	141
5.2.2 标准 Sun 和 SunJCE 提供者	144
5.2.3 其他提供者	144
5.2.4 初始化提供者	145
5.2.5 编写自己的提供者	146
5.3 核心引擎类	146
5.3.1 MessageDigest	147
5.3.2 MAC	148
5.3.3 SecureRandom	150
5.3.4 Cipher	152
5.3.5 Signature	157
5.3.6 密钥一致协议	159
5.4 参数、密钥和证书	160
5.4.1 算法参数	160
5.4.2 密钥	163
5.5 小结	175

第 6 章 短消息编码和加密	177
6.1 预处理	177
6.1.1 把数字转换成字节	177
6.1.2 把 7 位数据压缩成 8 位	179
6.1.3 通用压缩和 java.util.zip.Deflater	180
6.1.4 添加奇偶校验位	182
6.2 短消息加密	184
6.2.1 单分组加密	184
6.3 短消息编码	187
6.3.1 对客户所用数据编码	187
6.3.2 机器和客户可见的应用程序的编码	202
第 7 章 应用程序和数据框架	213
7.1 加密数据的数据库框架	213
7.1.1 选择密码	215
7.1.2 数据	216
7.1.3 查找、索引和约束	221
7.1.4 不对称数据的使用	224
7.1.5 空值和数据库应用程序	225
7.2 Java 中的安全内存管理	227
7.2.1 灵巧的数组类	227
7.2.2 字符数组	231
7.2.3 使用类 SecureRandom	231
7.3 保密密钥管理	232
7.3.1 保密密钥数据	232
7.3.2 密钥的生产	234
7.3.3 对密钥的加密	235
7.3.4 存储	235
7.3.5 密钥访问和分发	236
7.3.6 通过 Cipher、MAC 使用密钥	236
7.4 口令	240
7.4.1 启动口令	240
7.4.2 成员名和口令	242
7.5 日志	245

7.5.1	嵌入式加密日志	246
7.5.2	完全加密的日志文件	246
7.5.3	公开密钥日志文件	248
7.5.4	拆分日志文件	248
7.5.5	基于网络的日志	248
7.6	密钥记号和应用程序	249
7.6.1	记号设计	249
7.6.2	URL 记号	251
7.6.3	Cookie 记号	263
7.6.4	访问控制记号	265
7.7	小数值和货币值的计算	266
7.7.1	双精度值和浮点值	267
7.7.2	BigDecimal	268
附录 A	Java 密码类参考	271



位和字节

在进入密码算子细节前，我们先复习一下位和数值基数的一些基本内容。对于使用操作系统、底层协议、嵌入式系统的读者，本章的内容可能是复习一下以前所学的知识；对于其他每天都与信息技术和基本软件开发打交道，而不直接与字节信息打交道的人来说，这些信息是为理解本书余下部分所包含信息的主要基础。

1.1 一般运算

大多数密码功能直接作用于数字的机器表示。接下来的内容是关于数值如何以不同的进制表示、电脑如何用一组位的形式表示数、如何直接操作位。这部分是用计算机和语言类方式表达的，下节将着重论述 Java 模型。

1.1.1 数的基数

在日常操作中，我们用十进制表示数，数字是从 0~9；这样给出一个十进制数字串 $d_n d_{n-1} \cdots d_2 d_1 d_0$ ，数值是：

$$10^n d_n + 10^{n-1} d_{n-1} + \cdots + 10^2 d_2 + 10 d_1 + d_0$$

推而广之可将基数换成任何其他数 x ，它有 x 个不同的数字，数值可表示为：

$$x^n d_n + x^{n-1} d_{n-1} + \cdots + x^2 d_2 + x d_1 + d_0$$

在计算机领域，最主要的基数是 2，或者说数值的二进制表示，即每个数字或位不是 0 就是 1。十进制数 30 可用二进制表示成 11110 或 $16+8+4+2$ 。十六进制（或者叫基数 16）也经常用，数字是 0~9 以及分别代表 10~15 的 A, B, C, D, E 和 F。数 30 可以用十六进制表示成 1E 或 $16+14$ 。二进制、十进制和十六进制数位间关系列在表 1.1 中。

表 1.1 二进制、十进制和十六进制表示法

二进制	十进制	十六进制
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	10	A
1011	11	B
1100	12	C
1101	13	D
1110	14	E
1111	15	F

使用不同进制时，数的基数也许是模糊的。例如，99 是二进制还是十六进制（ $=9 \times 16+9$ ）？在这种情况下，通常在十六进制数前加 0x 或者只加 x（例如：99 写成 0x99 或 x99）。二进制数也可能发生同样的问题，101 是十六进制 101 还是二进制 101（ $4+1=5$ ）？为避免二进制数与其他进制数混淆，通常在数字串后加个下标 2，例如 101_2 。

任何数都可以用另一个正的基数 b 表示出来；但是转换不是很容易，我们将在较后的章节中讨论一般的基数转换。但是，如果知道一个基数是另一个基数的幂，那么两者之间的转换就容易多了。例如，十进制数 1234 用基数 10 可表示为： $1 \times 1000 + 2 \times 100 + 3 \times 10 + 4$ 。但是，1234 也可以看成是基数为 100 的两个“数字”（12）和（34），值为 $12 \times 100 + 34 \times 1$ 。数相同，值也相同，只是位被重组了。这

些特点对底为2的数很重要，一个二进制数字串，将其分为每4位一组，这样易于转换成十六进制数，计算成十六进制值为：

$$10011100 = 1001\ 1100 = 9C$$

1.1.2 位和字节

位是计算机可赖以工作的最小信息单位，它可采用两个值“1”和“0”，虽然根据上下文有时用“是”和“否”。在多数现代计算机中，我们不直接处理“位”，而是一串位，有时称做一个“字”，最小的一个字是一个“字节”。今天，一个字节默认为8位，但技术上它可以是4~10个位。除了旧的或实验用CPU系统还在用外，奇数值在实际应用中已无人用了。为了精确，许多标准用“8位组”(octet)来表示8位。但是我们更愿意用通常的“字节”(byte)。现代CPU一般在大得多的字长上运行：术语“32位微处理器”也就是说，在一个时钟周期内，CPU基本上在32位字上运行。当然也可在8位字上运行，但不意味着运行得更快。许多CPU也有特殊的说明，可在更大的字上工作，如SSE和有类似指令的多媒体系统，还有PowerPC上的矢量处理。

每个0~255的整数都可用基数为2的一个字节来表示。位 n 表示值 2^n ，字节的值就是这些位的和。一个数的表示可用位展开，位0在右，位7在左：

$$(b_7b_6b_5b_4b_3b_2b_1b_0) = 2^7b_7 + 2^6b_6 + 2^5b_5 + 2^4b_4 + 2^3b_3 + 2^2b_2 + 2^1b_1 + 2^0b_0$$

或用十进制表示：

$$(b_7b_6b_5b_4b_3b_2b_1b_0) = 128b_7 + 64b_6 + 32b_5 + 16b_4 + 8b_3 + 4b_2 + 2b_1 + b_0$$

例如：00110111 = 32 + 16 + 4 + 2 + 1 = 55

左边的位称为“最有意义的位”，因为它们对整个数值的贡献最大。同样，右边的位叫“意义最小的位”。这种排列方式叫做“大尾”(Big-Endian)，我们将在后面讨论。

1.1.3 有符号字节

负数可以用多种方式表示。最简单的是将一个位取反来代表这个数的符号，不管是正还是负。0~6的位表示数，位7代表正负。这样可表示-127~127的数，但有两个零值：一个正零(+0)和一个负零(-0)。虽然有两个零值很奇怪，但工作起来很方便。较大的问题是何时发生溢出，如：127 + 2在无符号运算时为129(或1000001)，而在有符号运算时，其值却是-1。

最常用的表示法是“2的补码”。给定 x ，它的负数是将所有的位反转（1 变成 0，0 变成 1）并加 1，或处理成 $-1-x$ （值不变）。例如，在表 1.2 中，1+127 值是 -128。虽然这种方法有些怪，但有很多好处。微处理器只能给一个加循环编码，而与之互补的循环可作为加和减两种运算（事实上，许多 CPU 在做减法时处理原值的补码）。另外的好处体现在发生转换时，即将一个字节转换成较大的字时（见下节）。

表 1.2 2 的补码表示法

无符号值	有符号值	十六进制表示	二进制表示
0	0	00	00000000
1	1	01	00000001
2	2	02	00000010
126	126	7d	01111110
127	127	7f	01111111
128	-128	80	10000000
129	-127	81	10000001
130	-126	82	10000010
253	-3	fd	11111101
254	-2	fe	11111110
255	-1	ff	11111111

1.1.4 位运算符

常用的数学运算功能，如加法和乘法，将字解释成数值，并执行合适的运算。其他运算直接作用于位而不考虑它们的数值表示。这些运算就是位运算或逻辑运算。下节中的例子虽然是用 8 位字节表示的，但实际上它们可扩展到任何字长。

如表 1.3 所示，根据上下文是编程还是书面表示，运算用不同的符号来表示。需要时，本书将采用编程符号，除了 XOR，因为脱字符号（^）在一些系统中用来表示指数。

表 1.3 位逻辑运算和符号

运算	C 风格表示法	C 风格的自赋值	书面表示
非 a	$\sim a$		$\neg a$
a 与 b	$a \& b$	$a \&= b$	$a \wedge b$
a 或 b	$a b$	$a = b$	$a \vee b$
a 异或 b	$a \wedge b$	$a \wedge= b$	$a \oplus b$