安全丛书

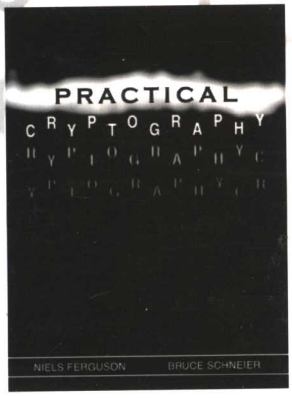
WILEY

★ 2003年度Jolt大奖技术类入围图书

★ *Applied Cryptography* 姊妹篇

密码学实践

Practical Cryptography



[美] Niels Ferguson
Bruce Schneier 著

张振峰 徐 静 李红达 等译

電子工業出版社
Publishing House of Electronics Industry
www.phei.com.cn

密码学实践

Practical Cryptography

信息安全丛书

密码学实践

Practical Cryptography

[美] Niels Ferguson
Bruce Schneier 著

张振峰 徐 静 李红达 等译

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内容简介

本书从工程实践的角度讲述了如何实现密码系统。作者结合自己丰富的实践经验，从特定算法的选取、关键部件的实现到基础设施的建设，详细讲述了如何在现实世界中正确地实现密码系统，探讨了如何把密码系统的安全性转化为实际的安全性。全书共分为五部分。第一部分介绍了密码系统的设计原理、密码学的基本内容和各种攻击方法；第二部分讲述了消息的安全性，包括分组密码、Hash 函数、消息认证码等基本密码模块及其选取和实现；第三部分讲述了公钥密码系统与密钥协商；第四部分探讨了密钥管理问题；最后一部分介绍了标准与专利方面的问题。

本书是第一部从工程的角度论述如何正确实现密码系统以及如何把它整合在实际安全系统中的著作。可作为密码学专业的大学生、研究生的教材，或作为密码学研究人员以及各类设计和实现密码系统的工程师的参考书。

Niels Ferguson, Bruce Schneier: **Practical Cryptography.**

ISBN 0-471-22357-3

图书在版编目（CIP）数据

密码学实践/（美）弗格森（Ferguson, N.）等著；张振峰等译. -北京：电子工业出版社，2005.8
（信息安全丛书）
书名原文：Practical Cryptography
ISBN 7-121-01628-1

Ⅰ.密… Ⅱ.①弗… ②张… Ⅲ.密码-理论 Ⅳ.TN918.1

中国版本图书馆 CIP 数据核字（2005）第 087755 号

责任编辑：周宏敏　　特约编辑：王　崧
印　　刷：北京市顺义兴华印刷厂
出版发行：电子工业出版社
　　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：16　　字数：358 千字
印　　次：2005 年 8 月第 1 次印刷
定　　价：33.00 元

译 者 序

在过去的十年里，信息安全的重要性得到了空前的重视，而密码技术作为信息安全中的关键技术也受到了极大的关注，甚至被看做是一种具有魔力的强大工具而广为流行。各种各样的密码学专著，尤其是Bruce Schneier所著的《应用密码学》（“Applied Cryptography”），极大地推动了密码技术的研究与应用的发展。

多年的实践表明，密码学是一门独特的工程学科。如何利用密码系统来设计和构建实际的安全系统，如何将密码系统的安全性转化为实际的安全性，绝非一件容易的事情。Niels Ferguson和Bruce Schneier所著的本书是第一部面向工程实践的密码学专著，旨在向设计和实现密码系统的人们提供具体的指导性建议。两位密码学专家以其丰富的实践经验，分析了安全系统所独具的特点，探讨了安全系统的设计原理，从特定算法的选取、关键部件的实现，到基础设施的建设，详细讲述了如何在实际系统中正确地实现和应用密码系统。本书不是一门基础教程，也不是一部全面而详尽的密码学百科全书，而是在有限的范围内，利用简单易懂的事实讨论密码系统的实现细节和安全系统的设计原理。作为《应用密码学》的后续作品，这是第一部实用的密码系统设计与密码产品实现指南，填补了理论密码学与密码学实践之间的鸿沟。我们相信本书不仅对密码实践工程人员具有重要的指导作用，而且对于密码学的理论研究也有一定的借鉴意义。

本书由张振峰、徐静、李红达等人翻译。其中，第一部分由张振峰等人翻译，第二部分由李红达翻译，第三部分由张振峰、徐静和乔鹏等人翻译，第四部分由徐静翻译。本书在翻译过程中得到了电子工业出版社的大力支持，在此深表谢意！

本书的出版得到了国家自然科学基金（编号60373039）的支持，在此表示感谢。

特别感谢电子工业出版社的周宏敏编辑为本书的出版提出的宝贵意见，感谢所有帮助和支持我们的人！

前　言

在过去的十年里，密码学对于数字系统的安全带来了更多的危害，而不是更好地提高了它的安全性。密码学早在20世纪90年代就作为因特网的保护者登上了历史舞台。有人认为，密码学是一个强大的均衡器，一个强大的数学工具，让寻求保护的普通人具有和国家情报局一样的地位。有人把它看做是一种武器，当政府在电脑空间里丧失统治能力时将给国家带来灾难。也有人认为它是毒品贩卖者、恐怖分子以及儿童色情作品发行者的完美而又可怕的工具，他们可以用它进行完善保密的通信。即使那些持有现实主义看法的人也把密码学想像为能够使全球化的商务在这个全新的在线领域内成为现实的一门技术。

十年过去了，所有这些都没有发生。不管密码学有多流行，因特网的国界比任何时候都明显。侦察和检测通信犯罪的能力更多地与政治和管理领域有关，而与数学无关。同强大的有良好资助的情报局相比，个人仍然没有什么机会。而且全球化贸易的兴起与密码学的流行也没有什么关系。

在很大程度上讲，密码学除了给因特网用户一个错误的安全意识之外什么也没做，它承诺了安全性却没有提供安全性。除了攻击者之外，对任何人都没有什么益处。

为什么会这样？这与密码学作为一门数学科学没有多少关系，而与密码学作为一门工程学科有更大的关系。在过去的十年内，我们开发、实现并实际应用了密码系统，它没有起到多大作用的原因在于如何把数学上承诺的密码系统的安全性转化成实际的安全性。而事实证明，做到这一点并非易事。

很多工程师把密码学看做是一种有魔力的安全金粉，可以撒在他们的硬件和软件上，并将浸透那些产品而带来神话般的“安全”性质。太多的消费者读到类似于“加密”的产品声明时，就相信其中存在神奇的安全金粉。评论家也好不了多少，他们在此基础上对密钥长度之类的事情进行比较，然后就断言一个产品比另外一个更安全。

安全性的强度体现在最薄弱的环节，而密码系统中的数学绝不是最薄弱的环节。密码学的基本原理是非常重要的，但更为重要的是如何实现和使用这些基本原理。讨论一个密钥应该是112比特还是128比特，就好比把一段巨大的树桩栽在地面上，而希望攻击者恰好会撞上它。你可以讨论这个树桩应该是一英里高还是一英里半高，但是攻击者只需要绕着树桩走过去就行了。安全性是一堵宽大的围墙，正是密码系统周围的东西才使得密码系统真正有效。

过去十年里的密码学书籍给这个魔法带上了光环。一本又一本的书都在赞美112比特的三重DES的优点，而没有讲到应该如果产生和使用它的密钥。一本又一本的书提出了这种或者

那种复杂的协议，而一点儿也没有提到那些协议必须面对的商业和社会约束。一本又一本的书都把密码学作为纯数学进行解释，而没有考虑现实世界和实际情况的限制。然而，正是现实世界和实际情况的限制，构成了承诺的密码学魔力与数字安全现实之间的差异。

本书也是一本关于密码学的书，但它是一本关于密码学“缺陷”的书。我们的目的是明确地描述现实世界的约束和密码学的真实情况，讨论如何设计安全的密码系统。在某些方面，本书是Bruce Schneier的第一本书《应用密码学》（“Applied Cryptography”）的续篇，那本书在十年以前就出版了。《应用密码学》广泛地概括了密码学以及密码学能够带来的无数的可能性，本书讨论的范围是有限的、集中的。我们没有给出许多选择，而是只给出一个选项并告诉读者如何正确地实现。《应用密码学》展示了密码学作为一门数学科学的惊人潜力——什么是可能的，什么是可以得到的；而本书则向设计和实现密码系统的人提出了具体的建议。

在本书中，我们努力填补密码学承诺与密码学的实际情况之间的鸿沟。我们尝试着教会工程师们如何用密码学来提高安全性。

我们是有资格写这本书的，因为我们两人都是经验丰富的密码学家。Bruce由于他的《应用密码学》和《秘密与谎言》（“Secrets and Lies”）以及他的时事通信“Crypto-Gram”而广为人知。Niels Ferguson很早就在阿姆斯特丹的CWI（研究数学和计算机科学的荷兰国家研究院）开始构建密码支付系统，后来在荷兰一家称做DigiCash的公司工作。Bruce设计了Blowfish加密算法，并且我们两个人都是Twofish的设计团队的成员。Niels的研究引出了当代高效匿名支付协议的第一个范例。我们二人的学术论文加在一起达到了三位数。

最重要的是，我们在设计和构建密码系统方面都有着非常丰富的经验。自1991年到1999年，Bruce的咨询公司Counterpane Systems为一些世界上最大的计算机和金融公司提供了设计和分析建议。Niels在创办自己的咨询公司MacFergus之前也为Counterpane Systems工作。我们在现实世界里看到了密码学的生命，感觉到了它的气息，看到了它在工程实践甚至是商业现实面前的艰难。我们有资格写这本书，因为我们需要为咨询的客户一次又一次地写这些东西。

如何阅读本书

本书是一本记叙体式的书，而不是一本参考书。它讲述了密码系统的设计，从特定算法选取开始，由关键部分延伸到正常工作所需要的基础设施。我们讨论了一个单纯的密码问题——两个人进行安全通信的建立方式——这几乎是所有密码应用的核心问题。通过关注一个问题以及解决该问题的一条设计原理，我们相信能告诉读者很多关于密码工程的现实。

我们以前都曾经出过书，也知道出版是一门不完美的科学。我们尽了很大的努力，但本书仍不可能没有错误。我们很抱歉，但事情往往就是这样。（非常奇怪，密码系统也有类似的问题，本书中我们会谈到这一点。）不过，我们会努力使这本书尽可能地完美，我们有一条途径来保证使不可避免的错误得到修正。

- 阅读本书之前，请访问 http://www.macfergus.com/pc 并下载当前的更正列表。
- 如果发现了本书的错误，请检查它是否已经在列表中得到更正。
- 如果列表中没有，请通过 practical-cryptography@macfergus.com 通知我们。我们将把它添加到列表中。

我们认为密码学是数学中最大的乐趣所在。我们尽力使本书充满这种乐趣，并希望你能享受其中的乐趣。

Niels Ferguson
荷兰，阿姆斯特丹
niels@macfergus.com

Bruce Schneier
美国明尼苏达州，明尼阿波利斯
schneier@counterpane.com

2003 年 1 月

目　录

第 1 章　我们的设计原理 1
1.1　追求性能带来的恶果 1
1.2　追求性能带来的恶果 3

第 2 章　密码学背景 4
2.1　密码系统的作用 4
2.2　最脆弱的链接性质 5
2.3　敌手设置 6
2.4　实用的妄想狂 7
2.5　威胁模型 9
2.6　密码系统不是解决方案 10
2.7　棘手的密码系统 11
2.8　易于实现的密码系统 11
2.9　背景阅读材料 12

第 3 章　密码学简介 13
3.1　加密 13
3.2　认证 14
3.3　公钥加密 16
3.4　数字签名 17
3.5　PKI 18
3.6　攻击 19
3.7　安全等级 22
3.8　性能 23
3.9　复杂性 24

第一部分　消息的安全性

第 4 章　分组密码 28
4.1　分组密码简述 28

4.2 攻击类型 28
4.3 理想的分组密码 29
4.4 分组密码安全性的定义 30
4.5 实际的分组密码 32

第5章 分组密码模式 42
5.1 填充 42
5.2 ECB 43
5.3 CBC 43
5.4 OFB 45
5.5 CTR 46
5.6 一些较新的模式 47
5.7 模式的选择 48
5.8 信息的泄露 49

第6章 Hash 函数 52
6.1 Hash 函数的安全性 52
6.2 实际的 Hash 函数 54
6.3 Hash 函数的缺陷 56
6.4 修正缺陷 57
6.5 Hash 的选择 59
6.6 进一步的工作 59

第7章 消息认证码 60
7.1 MAC 的作用 60
7.2 理想的 MAC 60
7.3 MAC 的安全性 60
7.4 CBC-MAC 61
7.5 HMAC 63
7.6 UMAC 64
7.7 选择哪一个 MAC 66
7.8 MAC 的使用 67

第8章 安全信道 69
8.1 问题陈述 69
8.2 认证与加密的次序 71
8.3 概述 72

8.4 详细说明 74
8.5 选择 78
8.6 结论 79

第 9 章 实现问题（I） 80
9.1 创建正确的程序 81
9.2 创作安全的软件 83
9.3 保守秘密 84
9.4 代码质量 89
9.5 侧信道攻击 92
9.6 结论 93

第二部分 密 钥 协 商

第 10 章 随机性 96
10.1 真随机 96
10.2 PRNG 的攻击模型 98
10.3 Fortuna 99
10.4 发生器 100
10.5 累加器 104
10.6 种子文件管理 110
10.7 应该做什么 113
10.8 选取随机元素 113

第 11 章 素数 115
11.1 可除性与素数 115
11.2 生成小素数 117
11.3 素数的模运算 118
11.4 大素数 123

第 12 章 Diffie-Hellman 129
12.1 群 129
12.2 基本的 DH 130
12.3 中间相遇攻击 131
12.4 缺陷 132
12.5 安全素数 133
12.6 使用较小的子群 134

12.7 p 的长度 134
12.8 实践准则 136
12.9 可能出错的地方 136

第 13 章 RSA 139
13.1 引言 139
13.2 中国剩余定理 139
13.3 模 n 乘法 142
13.4 RSA 详细说明 143
13.5 使用 RSA 的缺陷 147
13.6 加密 148
13.7 签名 150

第 14 章 密码协商协议介绍 153
14.1 角色 153
14.2 信任 153
14.3 动机 155
14.4 密码协议中的信任 156
14.5 消息和步骤 157

第 15 章 密钥协商协议 163
15.1 背景 163
15.2 初试 163
15.3 永久性的协议 165
15.4 认证约定 165
15.5 第二次尝试 166
15.6 第三次尝试 167
15.7 最终协议 168
15.8 看待协议的不同观点 170
15.9 协议的计算复杂性 172
15.10 协议的复杂性 173
15.11 善意的警告 174
15.12 使用口令的密钥协商 174

第 16 章 实现问题（II） 175
16.1 大整数算术 175
16.2 更快的乘法 179

16.3 侧信道攻击 180
16.4 协议 182

第三部分 密 钥 管 理

第 17 章 时钟 186
17.1 使用时钟 186
17.2 使用实时时钟 187
17.3 安全隐患 188
17.4 产生可靠的时钟 189
17.5 相同状态的问题 190
17.6 时间 191
17.7 结论 191

第 18 章 密钥服务器 192
18.1 基本概念 192
18.2 Kerberos 192
18.3 更简单的方案 193
18.4 选择 195

第 19 章 PKI 之梦 196
19.1 PKI 的简短回顾 196
19.2 PKI 的例子 196
19.3 其他细节 197
19.4 结论 199

第 20 章 PKI 的现实问题 200
20.1 名字 200
20.2 权力 201
20.3 信任度 202
20.4 间接授权 202
20.5 直接授权 203
20.6 信用系统 204
20.7 修正的梦想 205
20.8 撤销 205
20.9 PKI 的优势是什么 207
20.10 选择 208

第 21 章　PKI 的实用性 209
21.1　证书模式 209
21.2　密钥的生命周期 210
21.3　密钥作废的原因 211
21.4　根据应用采取行动 212

第 22 章　存储秘密 213
22.1　磁盘 213
22.2　人脑记忆 213
22.3　便携式存储 216
22.4　安全 token 217
22.5　安全 UI 217
22.6　生物统计学 218
22.7　单点登录 219
22.8　丢失的风险 219
22.9　秘密分享 220
22.10　清除秘密 221

第四部分　其 他 事 宜

第 23 章　标准 224
23.1　标准过程 224
23.2　SSL 226
23.3　AES：由竞争带来的标准化 227

第 24 章　专利 228
24.1　先知部分 228
24.2　后续 228
24.3　含糊性 229
24.4　读专利 229
24.5　授权 229
24.6　专利保护 230
24.7　完善专利系统 231
24.8　不予承诺 231

第 25 章　专家 232

致谢 235

参考文献 236

第1章　我们的设计原理

本书是一本关于安全的书——是一本如何构建安全密码系统的书。本书体现出我们对于密码学的狂热，这是因为在这一领域工作的这些年来，我们已经看到了安全的整体状况。的确是这样的，我们分析过的每一个系统都以某种方式被攻破了。其中总是有一些部件是好的，但是它们总是以不安全的方式被使用。

假如整个社会希望未来的数字世界是安全的，那么就需要进行改进。我们期待本书能够对此有所贡献。

本书给出了大量的关于密码系统的实践信息，但是这些都不重要，除非我们能够说服你：正确地部署安全性非常重要。正确部署在其他领域意味着不留情面。这是很难做到的，我们需要花费多年的时间才能做到足够的不留情面。只有一点点安全性是不够的，这就像在院子周围修建一半的围墙，或者只把前门锁起来而把后门敞开着。安全性是不能妥协的系统性质。围墙上有一个洞就意味着不安全。所以，我们要放弃其他任何事情，给安全性留下足够的空间。根据经验，我们知道这在IT行业很难被接受。然而，如果希望使我们的数字世界安全，就必须这样做。

1.1　追求性能带来的恶果

苏格兰的Forth海湾大桥作为19世纪的工程奇迹，同从上面穿过的列车相比，它超乎想像地庞大（因而昂贵）。它是一个不可思议的超级工程，使我们很难相信自己的眼睛。而它的设计者做了一件正确的事情。他们面临的是一个之前从没有成功解决的问题：修建一座庞大的钢铁大桥。他们做出了令人震惊的工作。他们的成功是令人注目的：在100多年后的今天，这座大桥仍然在使用。一个成功的工程看起来就是这样的。

多年来，桥梁设计师已经知道了如何以更低的代价和更高的效率来修建类似的大桥。但是，首先需要考虑的是要建造一座安全而且实用的大桥，以缩减成本的方式带来的效率则是次要的问题。

在计算机安全中我们则颠倒了这些先后顺序。首要的设计目标时常是严格的效率需求，第一优先考虑的事情始终是速度，即使在速度并不重要的地方也是如此，这样就削减了安全方面的花费。而在安全性这个工程领域中，我们没有构建安全系统的技巧，即使拥有无限的预算。其结果总是构建一个高效的系统，但也不可避免地是一个不安全的系统。

我们从另一个角度来看一看苏格兰Forth海湾大桥的故事。在1878年，Thomas Bouch建成了当时世界上最长的大桥，它横跨敦提的泰河海湾。Bouch采用了铸铁和熟铁，使用了一种新的设计方法，这座大桥被认为是一个工程奇迹。建成不到两年，在1879年12月28日的夜里，大桥在一场暴风雪中倒塌了，当时正好有一列承载着75个人的列车经过，列车上所有的人都遇难了。这在当时是一场巨大的工程灾难①，所以，几年后当设计Forth海湾大桥的时候，设计者使用了更多的钢铁，这样做不仅使得大桥更安全，而且使公众看起来也觉得安全。

我们知道，有时候工程师的设计是错误的，尤其是当他们进行一些新的尝试的时候。而当他们设计出错的时候就会有人为此而丧生。然而，维多利亚的工程师们有一条很好的经验：如果失败了，就退回去并且采用更加保守的方法。计算机行业忘记了这个经验。在计算机系统中，当遭遇了非常严重的安全失败的时候（几乎每周都有这类事情发生），我们只是迈着沉重的脚步向前走，并把它看做是注定要发生的事情来接受。我们不会回到制图板并更加保守地进行设计，只是持续不断地抛出一些补丁并期望它们会解决问题，这样做是很不光彩的。

现在已经非常清楚了，任何时候我们都要选择安全性而不是效率。我们愿意在安全性上花费多少CPU的时间呢？——几乎所有的时间。如果在可靠的安全系统上花费90%的CPU时间，我们也不会在意。对于我们以及大多数用户来说，缺乏安全性是真正的伤害。这就是人们为何仍然通过传送几页纸来进行签名，为何必须担心病毒以及针对计算机的其他攻击的原因。未来的数字骗子知道的事情会更多，装备也会更好，从而计算机安全将成为一个越来越大的问题。我们所看到的只是数字犯罪潮的开始。如果想使用因特网来进行商业交易，就必须更好地保护我们的计算机。

当然，有很多方式可以实现安全性。但是，正如Bruce在《秘密与谎言》一书中用大量资料详细描述的那样，有效的安全性总要与防护、检测和响应相结合[89]②。密码系统的作用属于防护部分，它必须非常好，从而保证检测和响应部分（它们能够并且应该相互干涉）不会被控制。不管怎样，本书是关于密码学的，所以我们将把注意力集中在这上面。

是的，我们知道，你还对前面提到的90%感到吃惊。但是，我们的计算机还应该做什么事情呢？我们每秒只能打出大约10个字符——指的是现在——而即使在一个世纪之前的低速机器上做到这一点也没有什么问题。如今的机器要快上1000多倍。如果我们把90%的CPU用于安全性，计算机的速度看起来只剩下十分之一了。这大约是五年以前的计算机速度，而那些计算机对我们完成工作来说已经足够快了。

只有少数情况我们必须坐在计算机前面等待，包括等待打开网页、打印数据、启动某些程序、启动机器等。好的安全系统不应该使这些活动慢下来。现代的计算速度如此之快，很难估计如何以有益的方式来利用这种高速度。当然，我们可以在图像处理、三维动画乃至语音识别中使用，但是这些应用的关键部分不会执行任何与安全有关的动作，所以它们不会由于引入安

① William McGonagall为此写了一首有名的诗，结尾这样写道：建造的房子越坚固，受到伤害的可能性就越小。这样的建议在今天也是非常中肯的。

② 表示参见参考文献[n]。这里，n为89。

全系统而慢下来。正是其余的一些系统会占用开销，但它们已经尽可能快了。如果它们减慢了10倍，我们并不在意。大多数时间你甚至没有注意到这些开销。即使对于开销很重要的情况，换取的也只是商业代价。

如果你曾经想以效率的名义切下安全的一角，就需要对自己不断地说："我们已经有了足够快速但却不安全的系统，我们不需要另外一个"。我们发现人们乐意接受这种解释，而不是长篇大论的关于安全等级的理由。

现在已经很清楚了，我们考虑的优先顺序是：安全性第一，安全性第二，安全性第三，而性能则在列表下面的某个地方。当然，我们还是希望系统的效率尽可能地高，但它不是以安全性为代价换来的。我们知道这一设计原理在现实世界里并不总是可行的，通常市场需求会胜过对安全的需求。很少有系统是从零开始开发的，常常需要逐渐地增加保护，或者是在部署之后加以保护。系统需要与现有的不安全系统反向兼容。我们两个人都在这些限制条件下设计了很多安全系统。可以告诉读者的是，按照那种方式几乎不可能构建一个好的安全系统。本书的设计原理是安全性第一，安全性至上。我们愿意看到这一原理在商业系统中被更多地采用。

1.2 追求性能带来的恶果

没有一个复杂的系统是安全的。复杂性是安全性的最大敌人，而复杂性总以特性或者选项的形式出现。

我们将在以后更详细地讨论为什么说复杂性不能与安全性结合在一起，不过这里先给出一个基本的理由。设想一个有着20个不同选项的计算机程序，每一个选项可以选中也可以不选，那么就会有100多万个不同的组合。为了使程序能够工作，只需要对最常用的选项组合进行测试。而要保证程序是安全的，就必须对程序可能具有的100多万个组合中的每一个进行评估，并检查各个组合在每一种可能的攻击下是安全的，这是不可能做到的。而多数程序都不止20个选项。如果想要构建安全的东西，就必须把它简化。

简单的系统不一定是小系统。我们既可以构建大系统，同时也可以使其相当简单。复杂性用来度量在任何一个点有多少事件相互作用。如果一个选项的影响只限于程序的一小部分，那么它就不可能与一个影响范围局限于程序另一部分的选项相互影响。为了构建大的、简单的系统，必须在系统的不同部件之间提供一个很清晰而又很简单的接口。程序设计师把它称为模块化。一个好的简单接口可以把系统的其余部分与模块的细节隔离开来。

我们其实不必写下所有这些内容，它们都是基本的软件工程方面的知识。遗憾的是，在现实世界的系统中我们很少看到这样的系统。

在本书中，我们尽力要做到的一件事情是为密码模块定义简单的接口。没有特性、没有选项、没有特殊情况，也没有其他的东西需要记忆，只有我们提出来的最简单的定义。其中有一些定义是全新的，是在写这本书的时候提出来的。它们有助于对我们的安全系统进行思考，并且希望对你也有所帮助。