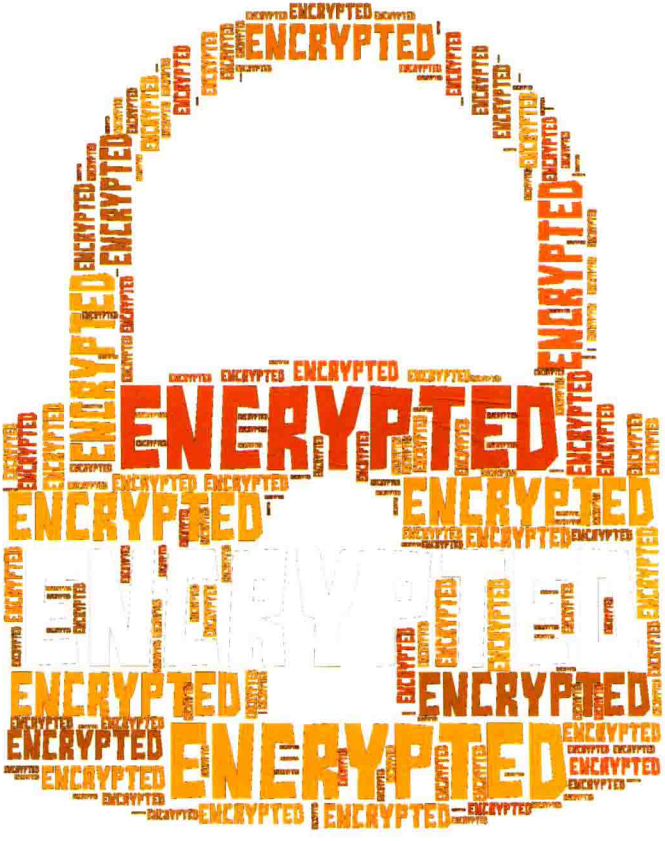
Python编程实现加密算法的初学者指南

Python 密码学编程

[美] Al Sweigart 著
李永伦 译

Hacking Secret Ciphers with Python

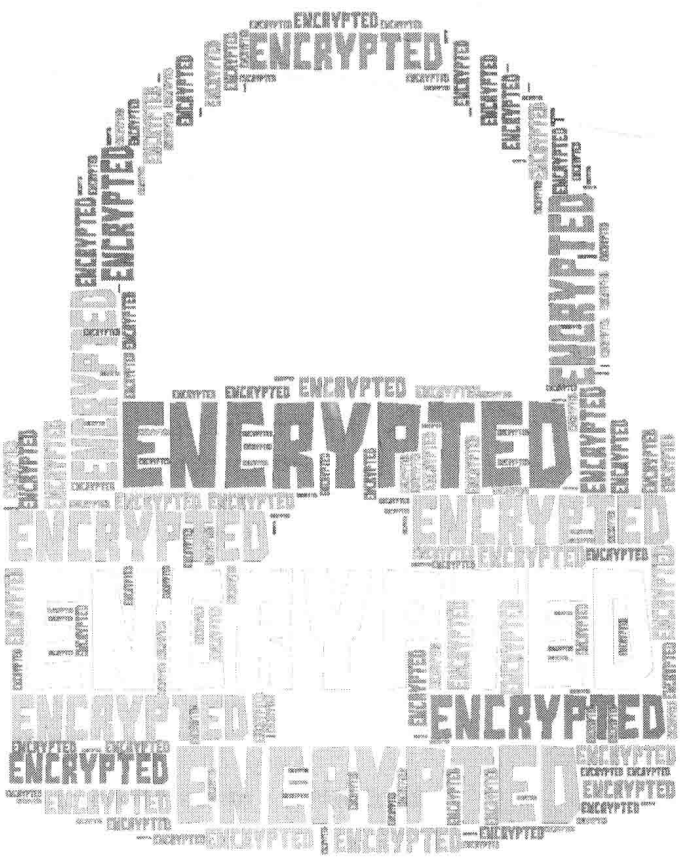


中国工信出版集团
人民邮电出版社
POSTS & TELECOM PRESS

Python
密码学编程

[美] Al Sweigart 著
李永伦 译

Hacking Secret Ciphers with Python



人 民 邮 电 出 版 社
北 京

图书在版编目（CIP）数据

Python密码学编程 / (美) 斯维加特 (Al Sweigart) 著 ; 李永伦译. -- 北京 : 人民邮电出版社, 2016.8
ISBN 978-7-115-42429-7

Ⅰ. ①P… Ⅱ. ①斯… ②李… Ⅲ. ①软件工具－程序设计 Ⅳ. ①TP311.56

中国版本图书馆CIP数据核字(2016)第157315号

◆ 著　[美] Al Sweigart
译　李永伦
责任编辑　陈冀康
责任印制　焦志炜

◆ 人民邮电出版社出版发行　北京市丰台区成寿寺路 11 号
邮编　100164　电子邮件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
北京艺辉印刷有限公司印刷

◆ 开本：800×1000　1/16
印张：21
字数：455 千字　2016 年 8 月第 1 版
印数：1－2 500 册　2016 年 8 月北京第 1 次印刷

著作权合同登记号　图字：01-2015-5084 号

定价：69.00 元

读者服务热线：(010)81055410　印装质量热线：(010)81055316
反盗版热线：(010)81055315
广告经营许可证：京东工商广字第 8052 号

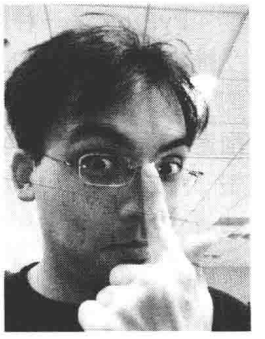
内容提要

Python 是一种高级程序设计语言，因其简洁、易读及可扩展性日渐成为程序设计领域备受推崇的语言。同时，Python 语言在算法领域也得到了很好的应用。

本书是面向初学者的 Python 密码学编程指南通过理论和实例相结合的方式介绍了多种加密算法及其破解方法。全书共分 24 章，由浅入深地介绍了与密码学编程相关的各类基础知识、编程技巧以及算法实现。除此之外，本书还提供了相应的源码下载资源，以供读者更好地进行探索和学习。

本书适合 Python 初学者和密码学的初学者，也适合信息安全从业人员。

作者简介



Albert Sweigart（你可以叫他 Al）在加州的旧金山从事有用软件开发，他喜欢去咖啡店。《Python 密码学教程》是他的第三本书。

他的前两本书分别是《Python 游戏编程快速上手》和《Python 和 Pygame 游戏开发指南》，它们可以在 http://inventwithpython.com 网站上免费阅读。

他的老家是得克萨斯州的休斯顿。他看到公园的松树就会大笑，这让人们觉得他是个傻瓜。

前言

有很多书教初学者如何使用加密法写秘密消息，有一些书教初学者如何破译加密法。据我所知，还没有书教初学者如何编写程序来破译加密法。这本书填补了这个空缺。

本书适合不懂加密、破译或密码学的初学者。本书的加密法（除了最后一章的 RSA 加密法）都有数百年历史了，现代计算机的计算能力可以破译使用它们加密的信息，现代组织或个人已经不再使用这些加密法了。有鉴于此，你不会因为本书里的内容而惹麻烦。

本书适合从来没有编过程序的初学者。本书使用 Python 编程语言讲解基本编程概念。Python 非常适合初学者学习编程：它是一种简单可读却又强大的编程语言，为专业软件开发者所用。Python 软件可以从 http://python.org 免费下载，可以在 Linux，Windows，OS X 和树莓派上运行。

“黑客”有两种定义。一种“黑客”是指通过学习来理解一个系统，并跳出系统原有的规则限制，有创造性地修改它，使之以新的方式来工作的人。另一种“黑客”也用来指入侵计算机系统，触犯个人隐私并造成伤害的罪犯。本书提到的“黑客”是第一种。**黑客很酷，罪犯则只是通过破坏来显摆智商的人**。就我个人而言，我的本职是一名软件开发者，和写病毒或网络诈骗相比，这份工作钱多活少。

还有一点要注意的，不要把本书里的任何加密程序用于你的实际文件。它们可以带来乐趣，但并不提供真正的安全。一般来说，你不应该信任你自己创造的加密法。正如传奇密码学家 Bruce Schneier 说的：“任何人，从最无能的外行到最好的密码学家，都能创建出他自己无法破译的算法。这并非难处。难处在于创建出别人无法破译的算法，即使经过数年分析，证明那点的唯一途径是通过各地最好的密码学家对这个算法进行长达数年的分析。”

如果你对这些程序如何工作有问题，可以随时给我发电子邮件：al@inventwithpython.com。

目录

第 1 章　制作纸质加密工具……1
1.1　密码学是什么……1
1.2　代码与加密法……2
1.3　制作纸质加密轮盘……2
1.4　虚拟加密轮盘……4
1.5　如何使用加密轮盘加密……4
1.6　如何使用加密轮盘解密……5
1.7　另一个加密法工具：St. Cyr 滑条…6
1.8　A 组练习……6
1.9　不用纸质工具做加密……7
1.10　B 组练习……9
1.11　双重强度加密……9
1.12　通过计算机编程进行加密……9
第 2 章　Pygame 基础知识……11
2.1　下载和安装 Python……11
2.1.1　Windows 安装步骤……11
2.1.2　OS X 安装步骤……12
2.1.3　Ubuntu 和 Linux 安装步骤…12
2.2　下载 pyperclip.py……12
2.3　启动 IDLE……12
2.4　特色程序……13
2.5　行号和空格……14
2.6　本书的文本换行……14
2.7　在线跟踪程序……15
2.8　使用在线比较工具检查输入的代码……15
2.9　复制粘贴文本……15
2.10　更多信息链接……15
2.11　编程和密码学……16
第 3 章　交互式 Shell……20
3.1　一些简单的数学知识……20
3.2　整数和浮点数……21
3.3　表达式……21
3.4　运算符顺序……22
3.5　计算表达式……22
3.6　错误是可以接受的……22
3.7　A 组练习……23
3.8　每个值都有一个数据类型……23
3.9　通过赋值语句把值存到变量里……23
3.10　重写变量……24
3.11　使用多个变量……25
3.12　变量名……26
3.13　驼峰式大小写……26
3.14　B 组练习……26
3.15　小结……26
第 4 章　字符串和写程序……28
4.1　字符串……28
4.2　使用+运算符的字符串连接……29
4.3　使用*运算符的字符串复制……30
4.4　使用 print()函数输出值……30
4.5　转义字符……31
4.6　引号和双引号……32
4.7　A 组练习……32
4.8　索引操作……33
4.9　负索引……33
4.10　分片操作……34
4.11　空分片索引……35
4.12　B 组练习……35
4.13　在 IDLE 的文件编辑器里写程序……35
4.14　Hello World!……36
4.15　Hello World 的源代码……36

4.16 保存你的程序 ····· 37
4.17 运行你的程序 ····· 37
4.18 打开你保存的程序 ····· 38
4.19 “Hello World”程序如何工作 ····· 38
4.20 注释 ····· 38
4.21 函数 ····· 39
4.22 print()函数 ····· 39
4.23 input()函数 ····· 39
4.24 结束程序 ····· 40
4.25 C 组练习 ····· 40
4.26 小结 ····· 40
第 5 章 反转加密法 ····· 41
5.1 反转加密法 ····· 41
5.2 反转加密法程序的源代码 ····· 41
5.3 运行反转加密法程序 ····· 42
5.4 用在线比较工具检查你的源代码 ····· 42
5.5 这个程序如何工作 ····· 43
5.6 len()函数 ····· 43
5.7 while 循环简介 ····· 44
5.8 布尔数据类型 ····· 44
5.9 比较运算符 ····· 45
5.10 条件 ····· 46
5.11 代码块 ····· 47
5.12 while 循环语句 ····· 47
5.13 “增长”一个字符串 ····· 48
5.14 一步一步跟踪程序 ····· 50
5.15 在我们的程序里使用 input() ····· 52
5.16 A 组练习 ····· 52
5.17 小结 ····· 52
第 6 章 凯撒加密法 ····· 53
6.1 实现程序 ····· 53
6.2 凯撒加密法程序的源代码 ····· 53
6.3 运行凯撒加密法程序 ····· 54
6.4 使用在线比较工具检查你的源代码 ····· 55
6.5 A 组练习 ····· 55
6.6 这个程序如何工作 ····· 55
6.7 使用 import 语句导入模块 ····· 55
6.8 常量 ····· 56
6.9 upper()和 lower()字符串方法 ····· 57
6.10 for 循环语句 ····· 58
6.11 相当于 for 循环的 while 循环 ····· 59
6.12 B 组练习 ····· 59
6.13 if 语句 ····· 59
6.14 else 语句 ····· 60
6.15 elif 语句 ····· 60
6.16 in 和 not in 运算符 ····· 61
6.17 find()字符串方法 ····· 62
6.18 C 组练习 ····· 62
6.19 回到代码 ····· 62
6.20 显示和复制加密/解密之后的字符串 ····· 64
6.21 加密非字母字符 ····· 65
6.22 小结 ····· 66
第 7 章 暴力破译凯撒加密法 ····· 67
7.1 破译加密 ····· 67
7.2 暴力破译 ····· 67
7.3 凯撒加密法破译程序的源代码 ····· 68
7.4 运行凯撒加密法破译程序 ····· 68
7.5 这个程序如何工作 ····· 69
7.6 range()函数 ····· 69
7.7 回到代码 ····· 70
7.8 字符串格式化 ····· 72
7.9 A 组练习 ····· 72
7.10 小结 ····· 72
第 8 章 使用换位加密法加密 ····· 73
8.1 换位加密法 ····· 73
8.2 A 组练习 ····· 74
8.3 换位加密法加密程序 ····· 74
8.4 换位加密法加密程序的源代码 ····· 75

8.5 运行换位加密法加密程序……76
8.6 这个程序如何工作……76
8.7 使用 def 语句创建你自己的函数…76
8.8 程序的 main()函数……77
8.9 形参……78
8.10 对形参的修改只存在于函数之内……79
8.11 全局作用域和本地作用域里的变量……79
8.12 global 语句……79
8.13 B 组练习……81
8.14 列表数据类型……81
8.15 使用 list()函数把区间对象转换成列表……82
8.16 重新赋值列表里的项……83
8.17 重新赋值字符串里的字符……83
8.18 列表的列表……83
8.19 C 组练习……84
8.20 在列表上使用 len()和 in 运算符…84
8.21 使用+和*运算符的列表连接和复制……85
8.22 D 组练习……85
8.23 换位加密算法……85
8.24 增强赋值运算符……88
8.25 回到代码……88
8.26 join()字符串方法……90
8.27 返回值和 return 语句……91
8.28 E 组练习……91
8.29 回到代码……91
8.30 特殊的__name__变量……92
8.31 密钥的大小和消息的长度……93
8.32 小结……93
第 9 章 使用换位加密法解密……94
9.1 在纸上使用换位加密法解密……94
9.2 练习 A 组……96
9.3 换位加密法解密程序……96
9.4 换位加密法解密程序的源代码……96
9.5 这个程序如何工作……97
9.6 math.ceil()、math.floor()和 round()函数……98
9.7 and 和 or 布尔运算符……101
9.8 B 组练习……102
9.9 真值表……102
9.10 and 和 or 运算符可以简化代码…103
9.11 布尔运算符的运算顺序……103
9.12 回到代码……103
9.13 C 组练习……105
9.14 小结……105
第 10 章 写一个程序测试我们的程序…106
10.1 换位加密法测试程序的源代码…106
10.2 运行换位加密法测试程序……107
10.3 这个程序如何工作……108
10.4 伪随机数和 random.seed()函数……108
10.5 random.randint()函数……109
10.6 引用……110
10.7 copy.deepcopy()函数……112
10.8 A 组练习……112
10.9 random.shuffle()函数……112
10.10 随机打乱一个字符串……113
10.11 回到代码……114
10.12 sys.exit()函数……114
10.13 测试我们的测试程序……115
10.14 小结……116
第 11 章 加密和解密文件……117
11.1 纯文本文件……117
11.2 换位加密法文件加密程序的源代码……118
11.3 运行换位加密法文件加密程序…120
11.4 读取文件……120

11.4.1 open()函数和文件对象……120
11.4.2 read()文件对象方法……120
11.4.3 close()文件对象方法……121
11.5 写入文件……121
write()文件对象方法……122
11.6 这个程序如何工作……122
11.7 os.path.exists()函数……123
11.8 startswith()和 endswith()字符串方法……123
11.9 title()字符串方法……124
11.10 time 模块和 time.time()函数…125
11.11 回到代码……126
11.12 A 组练习……126
11.13 小结……127
第 12 章 通过编程检测英文……128
12.1 计算机如何理解英文……128
12.2 A 组练习……130
12.3 检测英文模块……130
12.4 检测英文模块的源代码……130
12.5 这个程序如何工作……131
12.6 字典和字典数据类型……132
12.7 添加或修改字典里的项……132
12.8 B 组练习……133
12.9 在字典上使用 len()函数……133
12.10 在字典上使用 in 运算符……133
12.11 在字典上使用 for 循环……134
12.12 C 组练习……134
12.13 字典与列表之间的区别……134
12.14 在字典上查找项比在列表上更快……135
12.15 split()方法……135
12.16 None 值……136
12.17 回到代码……136
12.18 “除以零”错误……138
12.19 float()、int()和 str()函数以及整数除法……138
12.20 D 组练习……139
12.21 回到代码……139
12.22 append()列表方法……139
12.23 默认参数值……140
12.24 计算比例……141
12.25 E 组练习……142
12.26 小结……143
第 13 章 破译换位加密法……144
13.1 换位加密法破译程序的源代码…144
13.2 运行换位加密法破译程序……145
13.3 这个程序如何工作……146
13.4 使用三引号的多行字符串……146
13.5 回到代码……147
13.6 strip()字符串方法……148
13.7 A 组练习……150
13.8 小结……150
第 14 章 取模运算与乘数加密法和仿射加密法……151
14.1 噢，不，数学！……151
14.2 数学，噢耶！……151
14.3 取模运算（又名时钟运算）……151
14.4 取模运算符%……152
14.5 A 组练习……153
14.6 GCD：最大公约数（又名最大公因数）……153
14.7 使用古氏积木（Cuisenaire rods）可视化因数和 GCD……154
14.8 B 组练习……155
14.9 多重赋值……155
14.10 通过多重赋值交换值……156
14.11 找出两个数字的 GCD 的欧几里得算法……156
14.12 “互质”……157
14.13 C 组练习……157

14.14 乘数加密法……157
14.15 D 组练习……159
14.16 乘数加密法 + 凯撒加密法 = 仿射加密法……159
14.17 仿射密钥的第一个问题……159
14.18 使用仿射加密法解密……160
14.19 找出模逆……161
14.20 //整数除法运算符……161
14.21 cryptomath 模块的源代码……162
14.22 E 组练习……163
14.23 小结……163
第 15 章 仿射加密法……164
15.1 仿射加密法程序的源代码……164
15.2 运行仿射加密法程序……166
15.3 A 组练习……166
15.4 这个程序如何工作……166
15.5 把一个密钥分成两个密钥……167
15.6 元组数据类型……168
15.7 密钥的输入验证……168
15.8 仿射加密法加密函数……169
15.9 仿射加密法解密函数……170
15.10 生成随机密钥……171
15.11 仿射密钥的第二个问题：仿射加密法可以有多少个密钥……172
15.12 小结……173
第 16 章 破译仿射加密法……174
16.1 仿射加密法破译程序的源代码……174
16.2 运行仿射加密法破译程序……175
16.3 这个程序如何工作……176
16.4 仿射加密法破译函数……177
16.5 **指数运算符……177
16.6 continue 语句……178
16.7 A 组练习……180
16.8 小结……180
第 17 章 简单替代加密法……181
17.1 使用纸笔实现简单替代加密法…181
17.2 A 组练习……182
17.3 简单替代加密法的源代码……182
17.4 运行简单替代加密法程序……183
17.5 这个程序如何工作……184
17.6 程序的 main()函数……184
17.7 sort()列表方法……185
17.8 包装器函数……186
17.9 程序的 translateMessage()函数……187
17.10 isupper()和 islower()字符串方法……189
17.11 B 组练习……190
17.12 生成随机密钥……190
17.13 加密空格和标点符号……191
17.14 C 组练习……191
17.15 小结……192
第 18 章 破译简单替代加密法……193
18.1 计算单词模式……193
18.2 获取密词的候选单词列表……194
18.3 A 组练习……195
18.4 单词模式模块的源代码……195
18.5 运行单词模式模块……196
18.6 这个程序如何工作……197
18.7 pprint.pprint()和 pprint.pformat()函数……197
18.8 在 Python 里使用列表创建字符串……198
18.9 计算单词模式……199
18.10 单词模式程序的 main()函数…200
18.11 破译简单替代加密法……202
18.12 简单替代破译程序的源代码…202
18.13 破译简单替代加密法（理论）…205
18.14 使用交互式 Shell 探索

破译函数 ····· 205
18.15 这个程序如何工作 ····· 209
18.16 导入所有东西 ····· 209
18.17 正则表达式和 sub()正则方法简介 ····· 210
18.18 破译程序的 main()函数 ····· 211
18.19 部分破译加密法 ····· 211
18.20 空密字映射 ····· 212
18.21 把字母添加到密字映射 ····· 213
18.22 计算两个密字映射的交集 ····· 214
18.23 从密字映射移除已经破译的字母 ····· 215
18.24 破译简单替代加密法 ····· 217
18.25 从密字映射创建密钥 ····· 219
18.26 我们不能把空格也加密吗 ····· 220
18.27 小结 ····· 220
第 19 章 维吉尼亚加密法 ····· 221
19.1 不可破译的加密法 ····· 221
19.2 维吉尼亚密钥里的多个“密钥” ··· 221
19.3 维吉尼亚加密法程序的源代码 ··· 224
19.4 运行维吉尼亚加密法程序 ····· 226
19.5 这个程序如何工作 ····· 227
19.6 小结 ····· 230
第 20 章 频率分析 ····· 231
20.1 字母频率和 ETAOIN ····· 231
20.1.1 匹配字母频率 ····· 232
20.1.2 计算频率匹配分值的例子 ··· 233
20.1.3 另一个计算频率匹配分值的例子 ····· 233
20.1.4 破译每个子密钥 ····· 234
20.2 匹配字母频率的代码 ····· 234
20.3 这个程序如何工作 ····· 236
20.4 最常见的字母“ETAOIN” ····· 237
20.5 这个程序的 getLettersCount()函数 ····· 237
20.6 这个程序的 getItemAtIndexZero()函数 ····· 238
20.7 这个程序的 getFrequencyOrder()函数 ····· 238
20.8 sort()方法的 key 和 reverse 关键字参数 ····· 239
20.9 把函数作为值传递 ····· 240
20.10 通过 keys()、values()和 items()字典方法把字典转换成列表 ··· 241
20.11 对字典的项进行排序 ····· 242
20.12 这个程序的 englishFreqMatchScore()函数 ····· 243
20.13 小结 ····· 244
第 21 章 破译维吉尼亚加密法 ····· 245
21.1 字典攻击 ····· 245
21.2 维吉尼亚字典攻击程序的源代码 ··· 245
21.3 运行维吉尼亚字典破译程序 ····· 246
21.4 readlines()文件对象方法 ····· 247
21.5 巴贝奇攻击和卡西斯基试验 ····· 247
21.6 卡西斯基试验的第 1 步——找出重复序列的间距 ····· 247
21.7 卡西斯基试验的第 2 步——获取间距的因数 ····· 248
21.8 从字符串获取每隔 N 个字母 ····· 249
21.9 频率分析 ····· 249
21.10 暴力破译可能密钥 ····· 251
21.11 维吉尼亚破译程序的源代码 ··· 251
21.12 运行维吉尼亚破译程序 ····· 256
21.13 这个程序如何工作 ····· 258
21.14 找出重复序列 ····· 259
21.15 计算因数 ····· 260
21.16 通过 set()函数来移除重复值 ··· 261
21.17 卡西斯基测试算法 ····· 263
21.18 extend()列表方法 ····· 264
21.19 print()的 end 关键字参数 ····· 268

21.20 itertools.product()函数 ……269
21.21 break 语句 ……272
21.22 A 组练习 ……273
21.23 修改破译程序的常量 ……273
21.24 小结 ……274
第 22 章 一次一密加密法 ……275
22.1 牢不可破的一次一密加密法 ……275
22.2 为什么一次一密加密法是牢不可破的 ……275
22.3 小心伪随机 ……276
22.4 小心二次密码本加密法 ……277
22.5 二次密码本加密法就是维吉尼亚加密法 ……277
22.6 A 组练习 ……278
22.7 小结 ……278
第 23 章 寻找质数 ……279
23.1 质数 ……279
23.2 合数 ……280
23.3 质数筛选模块的源代码 ……280
23.4 这个程序如何工作 ……281
23.5 如何判断一个数字是不是质数 ……282
23.6 埃拉托色尼筛选法 ……283
23.7 primeSieve()函数 ……284
23.8 检测质数 ……285
23.9 拉宾米勒模块的源代码 ……285
23.10 运行拉宾米勒模块 ……287
23.11 这个程序如何工作 ……287
23.12 拉宾米勒算法 ……287
23.13 新的经过改进的 isPrime()函数 ……288
23.14 小结 ……289
第 24 章 公钥密码学和 RSA 加密法 ……291
24.1 公钥密码学 ……291
24.2 “课本” RSA 的危险 ……292
24.3 身份验证的问题 ……292
24.4 中间人攻击 ……293
24.5 生成公钥和私钥 ……293
24.6 RSA 密钥生成程序的源代码 ……294
24.7 运行 RSA 密钥生成程序 ……295
24.8 这个密钥生成程序如何工作 ……296
24.9 这个程序的 generateKey()函数 ……297
24.10 RSA 密钥文件格式 ……299
24.11 混合加密机制 ……300
24.12 RSA 加密法程序的源代码 ……300
24.13 运行 RSA 加密法程序 ……303
24.14 A 组练习 ……304
24.15 数字签名 ……304
24.16 RSA 加密法程序如何工作 ……306
24.17 ASCII：使用数字来表示字符 ……307
24.18 chr()和 ord()函数 ……308
24.19 B 组练习 ……308
24.20 区块 ……308
24.21 使用 getBlocksFromText()把字符串转成区块 ……311
24.22 encode()字符串方法和 bytes 数据类型 ……311
24.23 bytes()函数和 bytes 的 decode()方法 ……312
24.24 C 组练习 ……312
24.25 回到代码 ……313
24.26 min()和 max()函数 ……313
24.27 insert()列表方法 ……315
24.28 RSA 加密和解密的数学运算 ……316
24.29 pow()函数 ……317
24.30 从密钥文件读取公钥和私钥 ……318
24.31 完整的 RSA 加密流程 ……318
24.32 完整的 RSA 解密流程 ……320
24.33 D 组练习 ……321
24.34 我们为什么不能破译 RSA 加密法 ……321
24.35 小结 ……323

第 1 章　制作纸质加密工具

本章主要内容：

- 密码学是什么；
- 代码和加密法；
- 凯撒加密法；
- 加密轮盘；
- St. Cyr 滑条；
- 用纸笔做加密；
- “双重强度”加密。

我忍不住偷听，可能因为我在窃听。

——佚名

1.1　密码学是什么

看看以下两段文字：

```
“Zsijwxyfsi niqjsjxx gjyyjw. Ny
nx jnymjw ktqqd tw bnxitr; ny
nx anwyzj ns bjfqym fsi anhj ns
utajwyd. Ns ymj bnsyjw tk tzw
qnkj, bj hfs jsotd ns ujfhj ymj
kwznyx bmnhm ns nyx xuwnsl tzw
nsizxywd uqfsyji. Htzwynjwx tk
lqtwd, bwnyjwx tw bfwwntwx,
xqzrgjw nx ujwrnyyji dtz, gzy
tsqd zuts qfzwjqx.”
```

```
“Flwyt tsytbbnz jqtw yjxndwri
iyn fqq knqrqt xj mh ndyn
jxwqswbj. Dyi jjkxxx sg ttwt
gdhz js jwsn; wnjyiyb aijnn
snagdqt nnjwww, xstsxsu jdnxzz
xkw znfs uwwh xni xjzw jzwyjy
jwnmns mnyfjx. Stjj wwzj ti
fnu, qt uyko qqsbay jmwskj.
Sxitwru nwnqn nxfzfbl yy
hnwydsj mhnxytb myysyt.”
```

左边的文字是秘密消息。这段消息已被**加密**，或者说被变成了秘密代码。任何不知道如何**解密**（也就是把它变回普通英语消息）的人都无法阅读。本书将会教你如何加密和解密消息。

右边的消息只是随机乱码，没有包含任何有意义的内容。加密你写下来的消息是对其他人保密的一种方式，即使他们得到了加密之后的消息。这看起来和随机乱码完全一样。

密码学是使用秘密代码的科学。**密码编译者**是使用和研究秘密代码的人。本书会告诉你成为一名密码编译者需要知道什么。

当然，这些秘密消息并不总是保持秘密状态。**密码破译者**是能破译秘密代码并读取其他人的加密消息的人。密码破译者又称为**代码破译**者（code breaker）或**代码黑客**（hacker）。本书也会告诉你成为一名密码破译者需要知道什么。遗憾的是，你在本书里学到的破译方式

不会给你带来麻烦（我的意思是，幸亏如此）。

间谍、士兵、黑客、海盗、贵族、商人、暴君、政治激进分子、网购者以及任何要与可信好友分享秘密的人都依赖密码学，以确保他们的秘密还是秘密。

1.2　代码与加密法

19 世纪初发明的电报允许通过跨越大陆的电线进行即时通信，这比带着一袋信件骑马派送要快很多。然而，电报不能直接发送写在纸上的字母，它只能发送电子脉冲。短脉冲叫“点”，长脉冲叫“线”。

图 1-1　Samuel Morse

1791 年 4 月 27 日—1872 年 4 月 2 日

图 1-2　Alfred Vail

1807 年 9 月 25 日—1859 年 1 月 18 日

为了把这些点和线转成英文字母，需要一个编码（或**代码**）系统把英语翻译成电子脉冲代码（**编码**），另一边把电子脉冲翻译成英语（**解码**）。用于电报(后来也用于无线电)的代码叫摩斯代码(Morse Code)，由 Samuel Morse（见图 1-1）和 Alfred Vail（见图 1-2）发明。通过一个电报按钮敲打出点和线，电报员可以把英语消息发给世界另一端的某个人，几乎是实时的！（如果你想学习图 1-3 所示的摩斯代码，请到 http://invpy.com/morse。）

代码是可以理解的，而且是公开发布的。任何人都应该可以通过查找代码符号的含义解密已被加密的消息。

A	•—	T	—
B	—•••	U	••—
C	—•—•	V	•••—
D	—••	W	•——
E	•	X	—••—
F	••—•	Y	—•——
G	——•	Z	——••
H	••••		
I	••		
J	•———	1	•————
K	—•—	2	••———
L	•—••	3	•••——
M	——	4	••••—
N	—•	5	•••••
O	———	6	—••••
P	•——•	7	——•••
Q	——•—	8	———••
R	•—•	9	————•
S	•••	0	—————

图 1-3　国际摩斯代码，通过点和线表示字符

1.3　制作纸质加密轮盘

在学习通过计算机编程进行加密和解密之前，我们先来了解一下如何使用简单的纸质工

具手工完成这项任务。把可理解的英语文字（**明文**）变成隐藏秘密代码的乱码文字（**密文**）是很容易的。**加密法**（cipher）是一组转换明文和密文的规则。这些规则通常使用一个密钥。我们会在本书里学到多种不同的加密法。

我们来学一下凯撒加密法。这种加密法曾在两千年前被凯撒大帝用过。好消息是，它学起来很简单很容易。坏消息是，正因为它简单，密码破译者也很容易破译它。但我们可以把它看做一个简单的练习。Wikipedia 上有更多关于凯撒加密法的信息：http://en.wikipedia.org/wiki/Caesar_cipher。

要用凯撒加密法把明文转成密文，需要制作一个**加密轮盘**（又名**加密圆盘**）。你可以复印本书给出的加密轮盘（见图 1-4 和图 1-5），也可以打印 http://invpy.com/cipherwheel 上的那个加密轮盘。把这两个圆圈剪下来，然后把它们叠在一起，参考图 1-6 至图 1-8 所示的步骤。

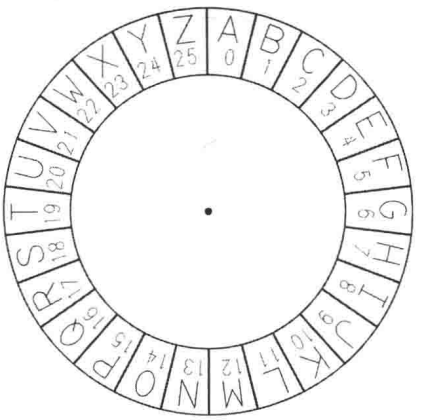


图 1-4 加密轮盘内圈

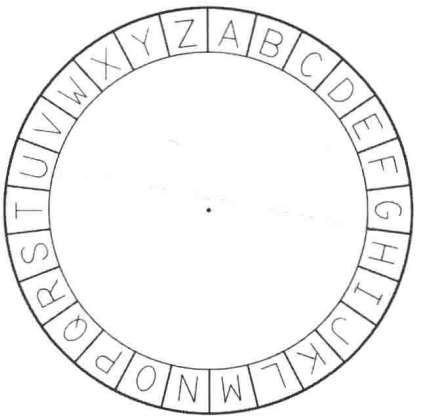


图 1-5 加密轮盘外圈

不要从本书上剪！复印本页或从 http://invpy.com/cipherwheel 上打印

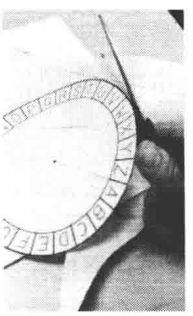


图 1-6 剪下加密轮盘

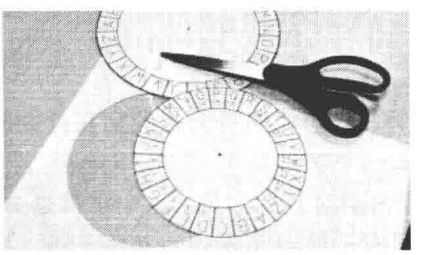


图 1-7 剪下来的圆圈

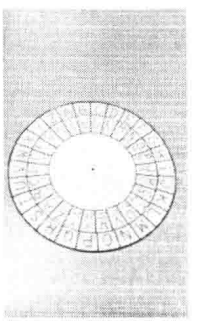


图 1-8 完成后的加密轮盘

剪下两个圆圈之后，把小的放在大的中间。在两个圆圈中间插一根大头针或曲头钉，这样你就可以在上面旋转了。现在，你拥有使用凯撒加密法加密信息所需的工具了。

1.4　虚拟加密轮盘

如果你手头没有剪刀和复印机，你也可以使用在线虚拟加密轮盘（见图 1-9）。用浏览器打开 http://invpy.com/cipherwheel，使用软件版的加密轮盘。

要旋转轮盘，用鼠标在上面点击一下，然后移动鼠标，直到你想要的密钥在适当的位置上。再次点击鼠标，就可以停止轮盘的旋转。

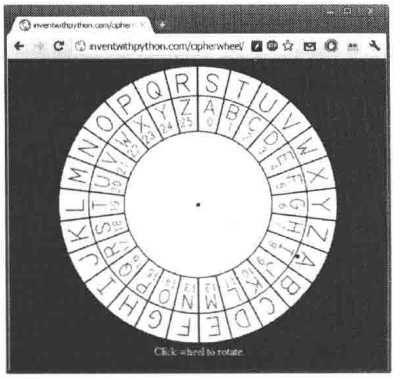


图 1-9　在线加密轮盘

1.5　如何使用加密轮盘加密

首先，在纸上用英语写下你的消息。在这个例子里，我们将会加密这条消息：“The secret password is Rosebud.”。接着，旋转内圈，直到它的字母匹配外圈的字母。值得注意的是，外圈的字母 A 下面有一个点。再看看外圈里的这个点对应的内圈里的数字，这个数字就是**密钥**。

这个密钥就是加密或解密消息的秘密所在。任何读过这本书的人都知道凯撒加密法，就像任何读过关于锁的书的人都知道门锁的工作原理。但是，就像平常的锁和钥匙，除非他们有密钥，否则他们不能解锁（也就是解密）已被加密的消息。在图 1-9 中，外圈的 A 在内圈的数字 8 上，这意味着我们将会使用 8 这个密钥来加密我们的消息。凯撒加密法使用的密钥范围是 0～25。我们的例子将会使用 8 这个密钥。保管好这个密钥，任何知道这条消息使用 8 这个密钥加密的人都能读懂密文。

```
T H E   S E C R E T   P A S S W O R D
↓ ↓ ↓   ↓ ↓ ↓ ↓ ↓ ↓   ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓
B P M   A M K Z M B   X I A A E W Z L

I S   R O S E B U D .
↓ ↓   ↓ ↓ ↓ ↓ ↓ ↓ ↓ .
Q A   Z W A M J C L .
```

对于我们的消息里的每个字母，我们将会找到它在外圈的位置，然后把它替换成内圈对应的字母。我们的消息的第一个字母是 T（“The secret…”里的第一个“T”），于是我们在外圈找到字母 T，然后找到内圈对应的字母。这个字母是 B，因此，我们总会把我们的秘密消息里的 T 替换成 B（如果我们使用 8 以外的其他密钥，那么明文里的 T 将被替换成别的字母）。

我们的消息里的下一个字母是 H，它会变成 P。字母 E 会变成 M。当我们加密完整条消息时，这条消息会从“The secret password is Rosebud.”变成“Bpm amkzmb xiaaewzl qa