

《深入浅出CryptoPP密码学库》

作者：韩露露、杨波

日期：2019年3月1日

说明

本电子文档为《深入浅出CryptoPP密码学库》的目录，它最初被存放于GitHub上。



简介

《深入浅出CryptoPP密码学库》内容简介：

本书向读者介绍密码学库CryptoPP（或Crypto++）的使用方法和设计原理。CryptoPP是一个用C++语言编写的、开源的、免费的密码程序库，它最初由Wei Dai开发，现由开源社区维护。CryptoPP库广泛应用于学术界、开源项目、非商业项目以及商业项目，它几乎包括了目前已经公开的所有密码算法，支持当前主流的多种系统平台，并且具有良好的设计结构和较高的执行效率。

全书共15章，主要内容包括随机数发生器、Hash函数、流密码、分组密码、消息认证码、密钥派生和基于口令的密码、公钥加密系统、数字签名、密钥协商等，本书涵盖C++程序设计、设计模式、数论和密码学等知识。

本书最大的特点就是以应用为导向、以解决实际工程问题为目标，理论结合实践，将抽象的密码学变成保障信息安全的实际工具。

本书可以作为密码学、网络安全等专业在校学生的上机实验教材，也可以作为信息安全产品开发、科研人员、密码算法实现者的参考手册。



资源

本书更多示例代码：<https://github.com/locomotive-crypto>

Crypto++网站：<https://www.cryptopp.com/>

Crypto++库GitHub地址：<https://github.com/weidai11/cryptopp>

Crypto++库SourceForge地址：<https://sourceforge.net/projects/cryptopp/>

Crypto++库Google论坛：

⇒公告通知地址：<https://groups.google.com/forum/#!forum/cryptopp-announce>

⇒用户群组地址：<https://groups.google.com/forum/#!forum/cryptopp-users>

目录

1	第1章CryptoPP库简介	1
1.1	CryptoPP库简介	1
1.2	CryptoPP库作者简介	1
1.2.1	Wei Dai 简介	1
1.2.2	Jeffrey Walton 简介	1
1.3	CryptoPP库内容简介	1
1.4	CryptoPP库的历史版本	1
1.5	其他的密码程序库	1
1.6	小结	1
2	第2章安装和配置CryptoPP库	2
2.1	下载CryptoPP库	2
2.2	在Windows系统下安装CryptoPP库	2
2.3	在Linux系统下安装CryptoPP库	2
2.4	小结	2
3	第3章程序设计基础	3
3.1	C/C++基础知识	3
3.1.1	面向对象程序设计的常用概念	3
3.1.2	类 (Class) 和对象 (Object)	3
3.1.3	类的数据成员 (Data Member) 和成员函数 (Member Function)	3
3.1.4	继承 (Inheritance)	3
3.1.5	类成员的访问属性 (Access Property)	3
3.1.6	重载 (Overloading)	3
3.1.7	构造函数 (Constructor) 和析构函数 (Destructor)	3
3.1.8	类型转换 (Type Cast)	3
3.1.9	多态性 (Polymorphism) 和虚函数 (Virtual Function)	3
3.1.10	纯虚函数 (Pure Virtual Function) 和抽象类 (Abstract Class)	3
3.1.11	传引用 (By Reference)、传值 (By Value) 和传指针 (By Pointer)	3
3.1.12	友元函数 (Friend Function) 和友元类 (Friend Class)	3
3.1.13	内存分配 (Allocate) 和释放 (Free)	3
3.1.14	模板 (Template)	3
3.1.15	异常处理 (Exception Handling)	3
3.1.16	命名空间 (Namespace)	3
3.2	数据结构和算法	3
3.2.1	使用SecByteBlock类	3
3.3	面向对象的程序设计原则和设计模式	3
3.3.1	创建型模式 (Creational Pattern)	3
3.3.2	结构型模式 (Structural Pattern)	3
3.3.3	行为型模式 (Behavioral Pattern)	3
3.3.4	其他模式 (Other Pattern)	3
3.4	小结	3

4	第4章初识CryptoPP库	4
4.1	使用帮助文档	4
4.2	CryptoPP库的源代码文件	4
4.3	数据编码	4
4.3.1	整数的b进制表示	4
4.3.2	Base编码	4
4.3.3	ASN.1编码	4
4.3.4	编码与加密的区别	4
4.4	Pipeling范式数据处理	4
4.4.1	Pipeling范式数据处理概念	4
4.4.2	Pipeling范式数据处理原理	4
4.4.3	使用Pipeling范式数据处理技术	4
4.4.4	以自动方式使用Pipeling范式技术	4
4.4.5	以手动方式使用Pipeling范式技术	4
4.4.6	以半手动或半自动方式使用Pipeling范式技术	4
4.4.7	一个特殊的BufferedTransformation类-ByteQueue	4
4.4.8	单链型到多链型Pipeling范式数据处理	4
4.5	计时器工具	4
4.6	秘密分割工具	4
4.7	Socket网络工具	4
4.8	压缩工具	4
4.9	小结	4
5	第5章随机数发生器 (Random Number Generator)	5
5.1	基础知识	5
5.2	CryptoPP库中的随机数发生器算法	5
5.3	使用CryptoPP库中的随机数发生器算法	5
5.3.1	示例一：使用LC_RNG算法	5
5.3.2	示例二：使用AutoSeededX917RNG算法	5
5.3.3	示例三：以Pipeling范式方式使用AutoSeededX917RNG算法	5
5.4	小结	5
6	第6章Hash函数 (Hash Function)	6
6.1	基础知识	6
6.2	CryptoPP库中的Hash函数算法	6
6.3	使用CryptoPP库中的Hash函数算法	6
6.3.1	示例一：计算字符串的Hash值	6
6.3.2	示例二：计算文件的Hash值	6
6.3.3	示例三：以Pipeling范式方式使用Hash函数	6
6.4	小结	6
7	第7章流密码 (Stream Cipher)	7
7.1	基础知识	7
7.2	CryptoPP库中的流密码算法	7
7.3	使用CryptoPP库中的流密码算法	7
7.3.1	示例一：使用XSalsa20算法加解密字符串	7
7.3.2	示例二：使用XSalsa20算法加解密文件	7

7.3.3 示例三：以Pipelining范式方式使用ChaCha12算法	7
7.4 小结	7
8 第8章分组密码 (Block Cipher)	8
8.1 基础知识	8
8.1.1 分组密码的运行模式	8
8.2 CryptoPP库中的分组密码算法和操作模式	8
8.3 使用CryptoPP库中的分组密码算法	8
8.3.1 示例一：以CBC模式运行分组密码Camellia	8
8.3.2 示例二：以EAX模式运行分组密码Camellia	8
8.4 小结	8
9 第9章消息认证码 (Message Authentication Code)	9
9.1 基础知识	9
9.1.1 消息认证码的构造	9
9.2 CryptoPP库中的消息认证码算法	9
9.3 使用CryptoPP中的消息认证码算法	9
9.3.1 示例一：使用HMAC算法	9
9.3.2 示例二：利用Hash函数自定义消息认证码算法	9
9.4 小结	9
10 第10章密钥派生和基于口令的密码 (Key Derivation and Password-based Cryptography)	10
10.1 基础知识	10
10.1.1 密钥派生函数的其他参数	10
10.1.2 利用派生函数实现数据保护的模型	10
10.2 CryptoPP库中的密钥派生和基于口令的密码算法	10
10.3 使用CryptoPP库中的密钥派生和基于口令的密码算法	10
10.3.1 示例一：使用密钥派生函数HKDF	10
10.3.2 示例二：利用基于口令的密钥派生函数实现数据保护	10
10.4 小结	10
11 第11章公钥密码数学基础	11
11.1 C/C++系统预定义的整数范围	11
11.2 CryptoPP库中大整数的构造	11
11.3 使用CryptoPP库的大整数	11
11.4 CryptoPP库中的数论算法	11
11.4.1 素性检测	11
11.4.2 数论常用算法	11
11.4.3 其他算法	11
11.4.4 产生素数有关的类	11
11.4.5 算法综合使用示例及习题	11
11.5 CryptoPP库中的代数结构	11
11.5.1 群、环、域的定义	11
11.5.2 CryptoPP库中的代数结构	11
11.5.3 使用CryptoPP库中的代数结构	11
11.6 密码学中的困难问题	11

11.7 小结	11
12 第12章公钥加密 (Public Key Encryption)	12
12.1 基础知识	12
12.2 CryptoPP库中的公钥加密算法	12
12.3 使用CryptoPP库中的公钥加密算法	12
12.3.1 示例一：使用非集成公钥加密算法RSAES	12
12.3.2 示例二：使用集成公钥加密算法ECIES	12
12.4 小结	12
13 第13章数字签名 (Digital Signature)	13
13.1 基础知识	13
13.2 CryptoPP库中的数字签名算法	13
13.3 使用CryptoPP库的数字签名算法	13
13.3.1 示例一：使用RWSS数字签名算法	13
13.3.2 示例二：使用ECNR数字签名算法	13
13.4 小结	13
14 第14章密钥协商 (Key Agreement)	14
14.1 基础知识	14
14.1.1 Diffie-Hellman密钥协商算法	14
14.2 CryptoPP库中的密钥协商算法	14
14.3 使用CryptoPP库中的密钥协商算法	14
14.3.1 示例一：使用经典的DH密钥协商算法	14
14.3.2 示例二：使用具有认证功能的ECMQV密钥协商算法	14
14.4 小结	14
15 第15章建立安全信道	15
15.1 基础知识	15
15.2 产生共享信息	15
15.2.1 方案分析	15
15.2.2 算法和参数的选取	15
15.2.3 方案执行流程图	15
15.3 完成文件的加密和认证	15
15.3.1 方案分析	15
15.3.2 算法和参数的选取	15
15.3.3 方案执行流程图	15
15.4 示例代码	15
15.4.1 服务端示例代码	15
15.4.2 客户端示例代码	15
15.4.3 程序运行结果说明	15
15.5 方案总结	15
15.6 小结	15

16 附录	16
16.1 附录1-示例程序的GUI版	16
16.2 附录2-基于CryptoPP（Crypto++）的软件产品	16
16.3 附录3-CryptoPP库算法索引	16
16.4 附录4-PKCS标准	16
16.5 附录5-网络资源及书籍推荐	16

1 第1章CryptoPP库简介

1.1 CryptoPP库简介

1.2 CryptoPP库作者简介

1.2.1 Wei Dai 简介

1.2.2 Jeffrey Walton 简介

1.3 CryptoPP库内容简介

1.4 CryptoPP库的历史版本

1.5 其他的密码程序库

1.6 小结

2 第2章安装和配置CryptoPP库

2.1 下载CryptoPP库

2.2 在Windows系统下安装CryptoPP库

2.3 在Linux系统下安装CryptoPP库

2.4 小结

3 第3章程序设计基础

3.1 C/C++基础知识

- 3.1.1 面向对象程序设计的常用概念
- 3.1.2 类 (Class) 和对象 (Object)
- 3.1.3 类的数据成员 (Data Member) 和成员函数 (Member Function)
- 3.1.4 继承 (Inheritance)
- 3.1.5 类成员的访问属性 (Access Property)
- 3.1.6 重载 (Overloading)
- 3.1.7 构造函数 (Constructor) 和析构函数 (Destructor)
- 3.1.8 类型转换 (Type Cast)
- 3.1.9 多态性 (Polymorphism) 和虚函数 (Virtual Function)
- 3.1.10 纯虚函数 (Pure Virtual Function) 和抽象类 (Abstract Class)
- 3.1.11 传引用 (By Reference)、传值 (By Value) 和传指针 (By Pointer)
- 3.1.12 友元函数 (Friend Function) 和友元类 (Friend Class)
- 3.1.13 内存分配 (Allocate) 和释放 (Free)
- 3.1.14 模板 (Template)
- 3.1.15 异常处理 (Exception Handling)
- 3.1.16 命名空间 (Namespace)

3.2 数据结构和算法

- 3.2.1 使用SecByteBlock类

3.3 面向对象的程序设计原则和设计模式

- 3.3.1 创建型模式 (Creational Pattern)
- 3.3.2 结构型模式 (Structural Pattern)
- 3.3.3 行为型模式 (Behavioral Pattern)
- 3.3.4 其他模式 (Other Pattern)

3.4 小结

4 第4章初识CryptoPP库

4.1 使用帮助文档

4.2 CryptoPP库的源代码文件

4.3 数据编码

4.3.1 整数的b进制表示

4.3.2 Base编码

4.3.3 ASN.1编码

4.3.4 编码与加密的区别

4.4 Pipeling范式数据处理

4.4.1 Pipeling范式数据处理概念

4.4.2 Pipeling范式数据处理原理

4.4.3 使用Pipeling范式数据处理技术

4.4.4 以自动方式使用Pipeling范式技术

4.4.5 以手动方式使用Pipeling范式技术

4.4.6 以半手动或半自动方式使用Pipeling范式技术

4.4.7 一个特殊的BufferedTransformation类—ByteQueue

4.4.8 单链型到多链型Pipeling范式数据处理

4.5 计时器工具

4.6 秘密分割工具

4.7 Socket网络工具

4.8 压缩工具

4.9 小结

5 第5章随机数发生器 (Random Number Generator)

5.1 基础知识

5.2 CryptoPP库中的随机数发生器算法

5.3 使用CryptoPP库中的随机数发生器算法

5.3.1 示例一：使用LC_RNG算法

5.3.2 示例二：使用AutoSeededX917RNG算法

5.3.3 示例三：以Pipelining范式方式使用AutoSeededX917RNG算法

5.4 小结

6 第6章Hash函数 (Hash Function)

6.1 基础知识

6.2 CryptoPP库中的Hash函数算法

6.3 使用CryptoPP库中的Hash函数算法

6.3.1 示例一：计算字符串的Hash值

6.3.2 示例二：计算文件的Hash值

6.3.3 示例三：以Pipeling范式方式使用Hash函数

6.4 小结

7 第7章流密码 (Stream Cipher)

7.1 基础知识

7.2 CryptoPP库中的流密码算法

7.3 使用CryptoPP库中的流密码算法

7.3.1 示例一：使用XSalsa20算法加解密字符串

7.3.2 示例二：使用XSalsa20算法加解密文件

7.3.3 示例三：以Pipelining范式方式使用ChaCha12算法

7.4 小结

8 第8章分组密码 (Block Cipher)

8.1 基础知识

8.1.1 分组密码的运行模式

8.2 CryptoPP库中的分组密码算法和操作模式

8.3 使用CryptoPP库中的分组密码算法

8.3.1 示例一：以CBC模式运行分组密码Camellia

8.3.2 示例二：以EAX模式运行分组密码Camellia

8.4 小结

9 第9章消息认证码 (Message Authentication Code)

9.1 基础知识

9.1.1 消息认证码的构造

9.2 CryptoPP库中的消息认证码算法

9.3 使用CryptoPP中的消息认证码算法

9.3.1 示例一：使用HMAC算法

9.3.2 示例二：利用Hash函数自定义消息认证码算法

9.4 小结

10 第10章密钥派生和基于口令的密码 (Key Derivation and Password-based Cryptography)

10.1 基础知识

10.1.1 密钥派生函数的其他参数

10.1.2 利用派生函数实现数据保护的模型

10.2 CryptoPP库中的密钥派生和基于口令的密码算法

10.3 使用CryptoPP库中的密钥派生和基于口令的密码算法

10.3.1 示例一：使用密钥派生函数HKDF

10.3.2 示例二：利用基于口令的密钥派生函数实现数据保护

10.4 小结

11 第11章公钥密码数学基础

11.1 C/C++系统预定义的整数范围

11.2 CryptoPP库中大整数的构造

11.3 使用CryptoPP库的大整数

11.4 CryptoPP库中的数论算法

11.4.1 素性检测

11.4.2 数论常用算法

11.4.3 其他算法

11.4.4 产生素数有关的类

11.4.5 算法综合使用示例及习题

11.5 CryptoPP库中的代数结构

11.5.1 群、环、域的定义

11.5.2 CryptoPP库中的代数结构

11.5.3 使用CryptoPP库中的代数结构

11.6 密码学中的困难问题

11.7 小结

12 第12章公钥加密 (Public Key Encryption)

12.1 基础知识

12.2 CryptoPP库中的公钥加密算法

12.3 使用CryptoPP库中的公钥加密算法

12.3.1 示例一：使用非集成公钥加密算法RSAES

12.3.2 示例二：使用集成公钥加密算法ECIES

12.4 小结

13 第13章数字签名 (Digital Signature)

13.1 基础知识

13.2 CryptoPP库中的数字签名算法

13.3 使用CryptoPP库的数字签名算法

13.3.1 示例一：使用RWSS数字签名算法

13.3.2 示例二：使用ECNR数字签名算法

13.4 小结

14 第14章密钥协商 (Key Agreement)

14.1 基础知识

14.1.1 Diffie-Hellman密钥协商算法

14.2 CryptoPP库中的密钥协商算法

14.3 使用CryptoPP库中的密钥协商算法

14.3.1 示例一：使用经典的DH密钥协商算法

14.3.2 示例二：使用具有认证功能的ECMQV密钥协商算法

14.4 小结

15 第15章建立安全信道

15.1 基础知识

15.2 产生共享信息

15.2.1 方案分析

15.2.2 算法和参数的选取

15.2.3 方案执行流程图

15.3 完成文件的加密和认证

15.3.1 方案分析

15.3.2 算法和参数的选取

15.3.3 方案执行流程图

15.4 示例代码

15.4.1 服务端示例代码

15.4.2 客户端示例代码

15.4.3 程序运行结果说明

15.5 方案总结

15.6 小结

16 附录

16.1 附录1-示例程序的GUI版

16.2 附录2-基于CryptoPP（Crypto++）的软件产品

16.3 附录3-CryptoPP库算法索引

16.4 附录4-PKCS标准

16.5 附录5-网络资源及书籍推荐