第5章 杂凑算法

5.1 SHA-256

【实验目的】

掌握目前普遍使用的SHA算法的基本原理，了解其主要应用方法。

【原理简介】

SHA（Secure Hash Algorithm）算法由美国NIST开发，作为数字签名标准中使用的Hash算法，并在1993年作为联邦信息处理标准公布。NIST在1995年公布了其改进版本SHA-1，在2001年发布了三个额外的SHA变体，这三个函数都将消息对应到更长的消息摘要，以它们的摘要长度（以位计算）加在原名后面来命名：SHA-256，SHA-384和SHA-512。SHA-256将不定长的输入变换为256b定长输出，作为输入数据的摘要（又称为数据指纹），反映了数据的特征。设摘要长度为n，则对于给定输入数据，找到另一不同数据但具有相同摘要的概率为2^{-n}，根据生日攻击的原理，寻找到两个不同数据具有相同摘要的概率为$2^{-\frac{n}{2}}$。Hash算法被广泛用于数据完整性保护、身份认证和数字签名当中。

关于SHA-256基本原理及数学基础的介绍，请参阅相关参考书。

【实验环境】

安装Windows操作系统的PC一台，其上安装Visual C++ 6.0以上编译器。

【实验步骤】

（1）从http://cryptopp.sourceforge.net/docs/ref521/sha_8cpp-source.html网页上得到算法的源代码。

（2）构造一个长度为1KB左右的文本文件，以SHA-256算法对文件计算Hash值。

（3）在上述文本文件中修改一个字母或汉字，再次计算Hash值，与步骤（2）中Hash值进行比较，看看有多少比特发生改变。

（4）测试SHA-256算法的速度。

【实验报告】

（1）简述SHA算法流程。

（2）写出步骤（2）和步骤（3）中的文本文件和Hash值。

（3）写出所使用机器的硬件配置以及SHA-256的测试速度。

【思考题】

考虑 Hash 算法如何用于数据完整性校验，与常用的 CRC 校验方法有何不同？

5.2 Whirlpool

【实验目的】

通过本实验，掌握 Whirlpool 算法的基本原理，了解其主要应用方法。

【原理简介】

2000 年，Vincent Rijmen 和 Paulo S.L.M.Barreto 设计了 Whirlpool，它是目前 NESSIE（New European Schemes for Signature, Integrity, and Encryption）唯一推荐使用的 Hash 函数，同时它也被国际标准组织 ISO 和国际电子技术协会 IEC 采用作为 ISO/IEC 10118-3 国际标准。

Whirlpool 是在分组密码 Square 的基础上设计的，算法的输入长度不超过 2^{256}b，产生 512b 的 Hash 值。最初的版本中，S 盒是随机生成的，具有良好的密码学特性；2001 年的版本中，对它进行了改进，使其密码学特性更好，而且更便于用硬件实现；2003 年的版本中，进一步修改了扩散阵列（Diffusion Matrix）。Whirlpool 是个很新的算法，实现方面经验很少，拥有与 AES 相似的性能和空间特性，与 SHA-512 相比，Whirlpool 需要更多硬件资源，但性能更好。

关于 Whirlpool 基本原理及数学基础的介绍，请参阅相关参考书。

【实验环境】

安装 Windows 操作系统的 PC 一台，其上安装 Visual C++ 6.0 以上编译器。

【实验步骤】

（1）从 http://cryptopp.sourceforge.net/docs/ref521/whrlpool_8cpp-source.html 网页上得到算法的源代码。

（2）构造一个长度为 1KB 左右的文本文件，以 Whirlpool 算法对文件计算 Hash 值。

（3）在上述文本文件中修改一个字母或汉字，再次计算 Hash 值，与步骤（2）中 Hash 值进行比较，看看有多少比特发生改变。

（4）测试 Whirlpool 算法的速度。

【实验报告】

（1）简述 Whirlpool 算法流程。

（2）写出步骤（2）和步骤（3）中的文本文件和 Hash 值。

（3）写出所使用机器的硬件配置以及 Whirlpool 的测试速度。

【思考题】

思考 Whirlpool 与本章前面介绍的 Hash 算法的区别。

5.3 HMAC

【实验目的】

掌握目前普遍使用的 HAMC 算法的基本原理，了解其主要应用方法。

【原理简介】

HMAC 是密钥相关的哈希运算消息认证码（keyed-Hash Message Authentication Code），HMAC 运算利用哈希算法，以一个密钥和一个消息为输入，生成一个消息摘要作为输出。

关于 HMAC 基本原理及数学基础的介绍，请参阅相关参考书。

【实验环境】

安装 Windows 操作系统的 PC 一台，其上安装 Visual C++ 6.0 以上编译器。

【实验步骤】

（1）从 http://cryptopp.sourceforge.net/docs/ref521/hmac_8cpp-source.html 网页上得到算法的源代码。

（2）构造一个长度为 1KB 左右的文本文件，以 HMAC 算法对文件计算 Hash 值。

（3）在上述文本文件中修改一个字母或汉字，再次计算 Hash 值，与步骤（2）中 Hash 值进行比较，看看有多少比特发生改变。

（4）测试 HMAC 算法的速度。

【实验报告】

（1）简述 HMAC 算法流程。

（2）写出步骤（2）和步骤（3）中的文本文件和 Hash 值。

（3）写出所使用机器的硬件配置以及 HMAC 的测试速度。

【思考题】

思考 HMAC 与本章前面介绍的 Hash 算法的区别。