Digital Signature System Based on RSA

Min PENG

Department of Information Engineering, Hunan Urban Construction College, Xiangtan, China

Email: chenlin_240886393@qq.com

Abstract: This article introduced some basic concepts and the process of realization in theory of RSA digital signature. It introduced the RSA algorithms fully, including the application condition and principle in RSA algorithm, the emergence of large prime numbers, the emergence of key pairs, encryption algorithm to the express and the decryption operations to cipher texts, thus built a theoretical foundation for realization. Basic principles in MD5 algorithm is also introduced in this text. It described the design and implementation of RSA digital signature detailedly, the realization of main module included the emergence of RSA key (a public key and private key), the realization of RSA encryption algorithm and decryption algorithm, the generation of MD5 message digest and the use of verification in RSA algorithm for implementing digital signature.

Keywords: digital signature; RSA; MD5; network information security

基于 RSA 的数字签名系统

彭　敏

湖南城建职业技术学院信息工程系，湘潭，中国，411104

Email: chenlin_240886393@qq.com

摘　要：介绍了 RSA 数字签名的一些基本概念和数字签名的理论实现过程；对 RSA 算法进行了全面系统的介绍，包括 RSA 算法的应用现状和原理、大素数的产生、密钥对的产生、对明文的加密运算和密文的解密运算，为具体实现打下了理论基础；还对 MD5 算法基本原理的介绍；详述了 RSA 数字签名的设计与实现，主要实现的模块包括 RSA 密钥的产生（一对公钥和私钥），RSA 加密算法和解密算法的实现，消息摘要 MD5 的生成以及利用 RSA 算法实现数字签名和签名的验证。

关键词：数字签名；RSA；MD5；网络信息安全

1 引言

数字签名技术是信息化建设中的关键安全机制之一，广泛应用于电子政务、电子商务、电子银行、电子证券等系统，甚至在日常的电子邮件中也有应用。数字签名提出的目的就是在网络环境下模拟日常的手工签名或印章，它可以抵御冒充、篡改、伪造、抵赖问题。数字签名原理比较简单，在具体实施数字签名时，发送方 A 对信息 M 实施数学变换 E，得到的信息记为 E(M)，与原信息唯一地对应，这是签名的过程；在接收方进行逆变换 D(E(M))，得到原来的消息，即为验证。如果数学变换性能优良，则 E(M)在传递过程中难以被破译、篡改，安全性极高。E、D 分别称为签名算法、认证算法。算法(签名、认证)是实现数字过程的核心。长期以来，人们围绕数字签名算法的设计作了大量的研究，取得了丰硕的成果，这些成果中很多已经被广泛应用于实际生产生活中。现今，普遍使用的签名算法绝大多数是基于以下的三个数字签名[1-4]。

大整数因子分解问题，由 kmuth 提出，代表算法 RSA。离散对数问题，有 J.Gill 提出，代表算法 ELGamal。椭圆曲线上的离散对数问题，代表算法 ECDSA。

本文主要研究的是大整数因子分解问题中的 RSA 算法实现的数字签名。

RSA 加密算法是一种非对称加密算法。在公钥加密标准和电子商业中 RSA 被广泛使用。RSA 是 1977 年由罗纳德•李维斯特（Ron Rivest）、阿迪•萨莫尔（Adi Shamir）和伦纳德•阿德曼（Leonard Adleman）一起提出的。当时他们三人都在麻省理工学院工作。RSA 就是他们三人姓氏开头字母拼在一起组成的。1973 年，

项目支持：湖南省教育厅资助科研项目(08A009)和(08B015)、湖南省重点学科建设项目资助、湖南省普通高校教学改革资助项目：计算机基础课程导向型自主学习的教学策略研究与实践，(湘教通[2007] 230 号，序号 119) 湖南工程学院教学改革研究项目院教字[2007]27 号

在英国政府通讯总部工作的数学家克利福德•柯克斯（Clifford Cocks）在一个内部文件中提出了一个相应的算法，但他的发现被列入机密，一直到 1997 年才被发表。

数字签名技术发展到今天，技术上已经成熟，各种标准也已经在最近几年中建立了起来，RSA 以其算法的简单性和高度的抗攻击性在实际通信中得到了广泛的应用。在许多操作平台如 Windos、Sun、Novell 等，都应用了 RSA 算法。另外，几乎所有的网络安全通信协议如 SSL,IPsec 等也都应用了 RSA 算法。ISO 几乎已制定 RSA 用作数字签名的标准。在 ISO9796 中，RSA 已成为其信息附件。法国银行界和澳大利亚银行界已使 RSA 标准化，ANSI 银行标准的草案也利用了 RSA。许多公司都采用了 RSA 安全公司的 PKCS。

2 数字签名的过程

简单地说，数字签名过程就是附加在数据单元上的一些数据,或是对数据单元所作的密码变换。这种数据或变换允许数据单元的接收者用以确认数据单元的来源和数据单元的完整性并保护数据,防止被人(例如接收者)进行伪造。它是对电子形式的消息进行签名的一种方法,一个签名消息能在一个通信网络中传输。基于公钥密码体制和私钥密码体制都可以获得数字签名,目前主要是基于公钥密码体制的数字签名。

公钥加密允许所有信息在网上传输，它能保证即使每个信息都被截获时，加密信息仍能处于安全状态。公钥加密的特点是加密和解密使用不同的密钥，即公开密钥和秘密密钥，一般地说，公钥用作加密密钥，私钥保密用作解密密钥。加密算法和解密算法也都是公开的。虽然公钥与私钥成对出现，但却不能根据公钥计算出私钥。公钥密码加密其实现的主要步骤如下[5-8]：

(1)每一用户产生一对密钥，分别用来加密和解密；

(2)每一用户将其中一个密钥存于公开的寄存器或其他可访问的文件中，该密钥称为公钥，另一个则为私钥；

(3)若 A 要给 B 发消息，则 A 用 B 的公钥对消息进行加密；

(4)B 收到消息后，用其私钥对消息进行解密 A。

数字签名是通过密码技术对电子文档进行的电子形式的签名，并非是书面签名的数字图像化。一般我们要对一些重要的文件进行签名，已确定它的有效性。伪造传统的签名并不困难，但是伪造数字签名是几乎不可能的，这是因为如果没有产生签名的私钥，要伪造由数字签名方案所产生的签名是不可行的。实际上人们可以否认曾经对一个文件签过名，但是要否认一个数字签名却困难得多，这是由于数字签名的生成需要使用私钥，它对应的公钥则用以验证签名，因而数字签名的一个重要性质就是非否认性。目前已经有一些方案，如数字证书，它是把一个实体(个人，组织或系统)的身份同一个私钥和公钥对“绑定”在一起，这使得一个人很难否认数字签名。综上所述，数字签名应满足以下要求：(1)接收方能确认或证实发送方的签字，但是不能伪造；(2)发送方把签字的消息发给接收方后，就不能否认所签名的消息。同样数字签名还应满足另外一个重要的方面：一旦收发双方就消息内容和来源发生争执时，应能给仲裁着提供发送方对所发消息签了字的证据。就其实质而言，数字签名是接收方能够向第三方证明接收到的消息及发送源的真实性而采取的一种安全措施，它的使用可以解决由于发送方不诚实而产生的纠纷，它可以保证发送方不能否认和伪造信息。数字签名的主要方式是：报文的发送方从报文文本中生成一个散列值(或报文摘要)。发送方用自己的专用密钥对这个散列值进行加密来形成发送方的数字签名。然后，这个数字签名将作为报文的附件和报文一起发送给报文的接收方。报文的接收方首先从接收到的原始报文中计算出散列值(或报文摘要)，接着再用发送方的公开密钥来对报文附加的数字签名进行解密。如果两个散列值相同，那么接收方就能确认该数字签名是发送方的。

3 RSA 算法数字签名

1978 年发表的 RSA 是公钥密码体制中最具代表性的算法，到目前为止它仍然是安全的。该算法的安全性基于“大整数的因数分解困难性问题”。RSA 签名算法描述如下：

(1)参数构造。

① 选取两个大素数 p,q；

② 计算出公开的模数：$n=q*p$；

③ 计算：$\varphi(n)=(p-1)(q-1)$，其中 φ 为 Euler 函数。

④ 随机选取 e，满足 $1<e<\varphi(n)$ $\gcd(e,\varphi(n))=1$，得到验证密钥 (e,n)；

⑤ 生成签名私钥 (d,n)，d 满 $ed=1\bmod(\varphi(n))$，销毁 q、p、$\varphi(n)$。

978-1-935068-10-5 © 2010 SciRes.

(2)签名和认证过程。

① 签名: $s=(H(m))^d \bmod n$ 其中 s 为签名，私钥为 (d,n)；

② 发送方 A 发送附加了签名的消息给 $B:A\to B:M//s$；

③ 认证：B 收到消息后，计算 $H=s^e \bmod(n)$，其中 (e,n) 为发送方 A 的公钥，计算 $H_1=H(m)$ 比较，如果 $H_1=H$ 则表示签名有效，否则无效。

4 MD5 算法

Ron Rivest 于 1990 年提出 MD4 单向散列函数，MD 表示消息摘要(Message Digest)，对输入消息，算法产生 128 位散列值。该算法首次公布之后，Bert den Boer 和 Antoon Bosselaers 对算法三轮中的后两轮进行了成功的密码分析。在一个不相关的分析结果中，Ralph MerKle 成功地攻击了前两轮。尽管这些攻击都没有扩展到整个算法，但 Rivest 还是改进了其算法，结果就是 MD5 算法。MD5 算法是 MD4 的改进算法，它比 MD4 更复杂，但设计思想相似，输入的消息可任意长，输出结果也仍为 128 位，特别适用于高速软件实现，是基于 32-位操作数的一些简单的位操作。

MD5 的全称是 Message-Digest Algorithm 5（信息-摘要算法），它将整个文件当作一个大文本信息，通过其不可逆的字符串变换算法，产生了这个唯一的 MD5 信息摘要。如果在以后传播这个文件的过程中，无论文件的内容发生了任何形式的改变（包括人为修改或者下载过程中线路不稳定引起的传输错误等），只要你对这个文件重新计算 MD5 时就会发现信息摘要不相同，由此可以确定你得到的只是一个不正确的文件，如果再有一个第三方的认证机构，用 MD5 还可以防止文件作者的“抵赖”，这就是它在数字签名中的应用。

5 VC++的实现

先随机产生一个大的数，然后用 miller-rabin 素数测试方法判断这个数是不是素数，利用费马小定理,对于给定的整数 n,可以设计一个素数判定算法.通过计算 d=2 n-1 mod n 来判定整数 n 的素性.当 d≠1 时，n 肯定不是素数。

```
long is_probable_prime( const vlong &p )
{// 为 1000 位的数字可以用相当长的一段时间
    const rep = 4;   const DWORD any[rep] =
{ 2,3,5,7   };/*,11,13,17,19,23,29,31,37..*/
    for ( DWORD i=0; i<rep; i+=1 )
    {           if ( modexp( any[i], p-1, p ) != 1 )
                return 0;   }
    return 1;}
    prime_factory::prime_factory( DWORD MP )
    {    np = 0;    // 初始化 pl
    char * b = new char[MP+1]; // 超出停止搜索
    for (DWORD i=0;i<=MP;i+=1) b[i] = 1;
    DWORD p = 2;
    while (1)
    {           // 跳过复合
        while ( b[p] == 0 ) p += 1;
        if ( p == MP ) break;
        np += 1;  // 划掉的倍数
        DWORD c = p*2;
        while ( c < MP )
        {    b[c] = 0;
             c += p;    }
        p += 1;    }
    pl = new DWORD[np];
    np = 0;
    for (p=2;p<MP;p+=1)
    {           if ( b[p] )
        {    pl[np] = p;
             np += 1;        }    }
    delete [] b;}
```

6 结束语

RSA 公钥体制是第一个将安全性基于分解因数的系统。它如同用任何加密方法的情形一样，许多潜在问题都对 RSA 方法的安全性起负面影响，在公开密钥(e，n)中，若 n 能被分解因数，则 p 和 q 被泄露，解密密钥 d 也就不再是秘密，进而整个 RSA 系统不安全。因此，在使用 RSA 系统时，对于 n 的选择是很重要的，必须使得公开 n 后，任何人无法从 n 得到 p 和 q。因此，提高 n 的位数无疑将大大提高 RSA 密码的安全性，这样，RSA 公钥体制才能在数字签名算法中保持良好的保密性能。因此在本设计中花了很多时间去了解和认识各种提高 RSA 加密安全的方法，特别是在 p, q, n 等参数的选择上。只有在 p, q, n 参数都足够大的情况下

才能保证加密的安全性，让 RSA 数字签名可以用于现在的各种商业活动中，保证用户的财产和隐私安全，因此现在应用基本上都要求 p, q 的位数在 512 位以上。

致　谢

感谢湖南工程学院田娟秀老师和谌新年教授对本文提供的帮助和宝贵意见，也感谢参考文献中所列的每一位作者。

References (参考文献)

[1] Eid, Mohamad; Alamri, Atif. A reference model for dynamic web service composition systems. International Journal of Web and Grid Services, v4, n2, Jun, 2008.

[2] Erickson, John; Siau, Keng. Web services, service-oriented computing, and service-oriented architecture: Separating hype from reality. Journal of Database Management, v19, n3.

[3] Cui, Lizhen; Yu, Haixu. Method for web services classification. Journal of Computational Information Systems. V4, n1, Feb, 2008.

[4] Paolini, Christopher P.; Bhattacharjee, Subrata. A web service infrastructure for thermochemical data, Journal of Chemical Information and Modeling, v48, n7, July, 2008.

[5] Badard, Thierry; Bedard, Yvan. Web services-oriented architectures for mobile SOLAP applications, International Journal of Web Engineering and Technology, v4, n4, Aug, 2008.

[6] Treiber, Martin; Dustdar, Schahram. Active web service registries, IEEE Internet Computing, v11, n5, Sept., 2007.

[7] Xilong Qu, Wenfang Sun, Jian Feng. Three-layered Resource Information Integration and Management Model Based on Web Service in Regional Networked Manufacturing System. Journal of Computational Information Systems.

[8] Zhongxiao Hao, Xilong Qu "The Design and realization of Digital Signature Based on Digital Stamp," Proceeding of IASP2009, pp. 529–551, April 2009.