文章编号:1674-7070(2011)02-0168-05

OpenPGP 协议在 JavaMail 中的研究与实现

傅德胜[1]　吴宜谦[1]

摘要

研究了 OpenPGP 协议的加密原理及其使用算法,提出一种基于 OpenPGP 协议的 JavaMail 客户端的设计方法,弥补了 JavaMail 安全性上的一些缺陷. 试验结果表明:在 JavaMail 中加入 OpenPGP 协议可有效地提高 E-mail 的保密性.

关键词

OpenPGP;JavaMail;加密算法;安全协议

中图分类号 TP311

文献标志码 A

0　引言

电子邮件是 Internet 提供的最广泛的服务之一. 据统计,在 100 个 Internet 用户中,大约有 80 个人的主要目的就是收发电子邮件,但是,电子邮件的内容均以明文形式在网络中传递,并没有任何文件内容完整性验证机制,攻击者可在邮件传输中截获数据. 解决电子邮件的安全问题主要是采用密码技术.

OpenPGP 是使用公开密钥加密算法加密邮件的一个非私有协议,该协议得到 Qualconn、IBM 等公司的支持,也很快将成为 IETF 标准. OpenPGP 源于 PhilAimmermann 在 1991 年发布的 PGP (Pretty Good Privacy),它基于早期 PGP 的二进制信息通信格式和身份认证格式,是一种近年来得到广泛使用、成型的端到端的安全邮件标准.

JavaMail 是 Java 对电子邮件处理的延伸,它提供和通信协议无关的 Java 解决方案,可以处理各种 E-mail 格式,包括 IMAP、POP、SMTP,以及 MIME 和其他与 Internet 相关的通信协议.

本文主要研究在 JavaMail 中加入 OpenPGP 协议的运用,以完善 JavaMail 规范本身在安全性方面的缺憾.

1　OpenPGP 协议

1.1　OpenPGP 原理

作为一种使用公钥密码系统加密电子邮件的不受专利限制的协议(RFC2440),OpenPGP 定义了加密消息、签名、私钥、公钥证书的标准格式,通过数字签名、数字加密、数据压缩、Radix-64 编码变换等技术提供对消息和数据文件的完整性与机密性服务.

OpenPGP 使用公钥传递每次会话中使用的对称密钥,实际传输的消息内容使用对称密钥进行加密,而每个对称密钥是在会话开始前随机生成并且只使用一次,会话密钥必须和消息同时发送. 接收方收到消息后通过私钥对对称密钥解密,在解密后使用对称密钥对内容进行解密. 具体的流程如图 1 所示.

1.2　OpenPGP 信任模型

OpenPGP 的信任模型与 PGP 信任模型相同,均为网络信任模型. 该模型包含了直接信任模型和等级信任模型,另外增加了第三方的监督者的概念[1]. 在 OpenPGP 中,存在 2 种密钥环:私钥环和公钥环. OpenPGP 协议允许任何人使用任何其他人的公钥,这也是网络信任

收稿日期 2010-09-07

作者简介

傅德胜,男,教授,博导,研究方向为信息安全. 000501@ nuist. edu. cn

1　南京信息工程大学　计算机与软件学院,南京,210044

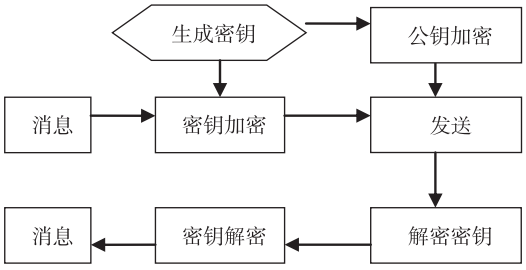

图 1 OpenPGP 工作流程

Fig. 1 OpenPGP flow chart

的实现方式. 如图 2 所示:实线连接的 2 个人可以见面,完全信任对方并且交换公钥. 现在,既然 Alice 信任 Charlie,而 Charlie 又给 Bob 数字签名过,所以 Alice 拥有 Bob 的公钥;David 同样拥有和 Alice 一样的公钥表单.

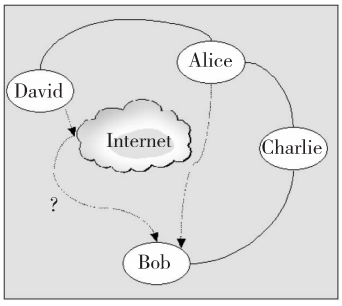

图 2 OpenPGP 网络信任模型

Fig. 2 OpenPGP web trust model

表 1 OpenPGP 的数据包结构

Table 1 OpenPGP packet structure

OpenPGP 的公钥数据包		OpenPGP 的私钥数据包	
字节	内容	字节	内容
1	版本号	不定	公钥数包
4	密钥创建时间	1	S2K 标志
2	密钥的有效性	1	对称加密算法(可选)
1	公钥算法	1	S2K 描述符(可选)
不定	公钥各参数 MPI	8	初始向量(可选)
		不定	加密的私钥各参数 MPI
		2	私钥各参数的校验和

1.3 OpenPGP 数据包

在 OpenPGP 中,本地使用的公钥保存在后缀为. pkr 的公钥环文件中,私钥保存在后缀为. skr 的私钥环文件中. 从公钥环导出的单个公钥保存在 asc 文件中,即 OpenPGP 定义的公钥证书. 密钥中的整数在存储时采用 Big Endian 的字节顺序,大整数以多精度整数 MPI(Multi-Precision Integer)的形式存储. MPI 的前 2 个字节是大整数的长度信息,从第 3 个字节起是按 Big Endian 顺序存放的大整数. OpenPGP 密钥数据以数据包为单位进行组织,一个数据包由数据包头(PacketHeader)和数据包体(PacketBody)2 部分组成,数据包头描述了数据包长度、类型等信息,包体存储相关的密钥信息. 表 1 描述了有关 OpenPGP 中公钥与私钥数据包的结构信息[2].

2 OpenPGP 算法

OpenPGP 源于 PGP,在继承 PGP 众多优点的同时,极大地扩展了 PGP 所支持的加密算法的数目. 2 种算法的比较如表 2 所示.

表 2 OpenPGP 算法与 PGP 算法比较

Table 2 Comparison of OpenPGP algorithm with PGP algorithm

	OpenPGP	PGP
公钥加密算法	RSA,Diffie-Hellman,DSA,Elgamal,Elliptic Curve,ECDSA	RSA,DSA,ECDSA
对称加密算法	AES,Triple-DES,IDEA,CAST5,Blowfish,SAFER-SK128,DES/SK	AES,IDEA,CAST5,DES/SK
压缩算法	ZLIB,ZIP	ZIP
散列算法	MD5,SHA-1,RIPE-MD/160,Double-Width SHA,MD2,TIGER,HAVAL	MD5,SHA-1,MD2

OpenPGP 安全机制的核心是公钥加密算法,主要包括 RSA、DSA、Diffie-Hellman、Elliptic Curve 算法等. 其中,最具代表性的算法是 RSA 和 ECC 椭圆曲线算法.

2.1 RSA 算法过程

1) 找出 2 个素数 p,q,计算 $n = p \times q, \phi = (p-1) \times (q-1)$(对素数 p 有欧拉函数值 ϕ 为 $p-1$).

2) 选择一个数 e,使得 $\mathrm{Gcd}(e,\phi) = 1$ 并且 $e < \phi$,这里 $\mathrm{Gcd}(x,y)$ 为 x,y 的最大公约数.

3) 求出一个逆元数 d,满足 $e \times d = 1 \bmod \phi$(采用扩展的欧几里德算法计算). 到此,就可以确定 public key 为(e,n),private key 为(d,n).

加解密过程:对任一数字 m,有密文 $c = m^e \bmod n$,对密文 c 能还原出明文 $c' = c^d \bmod n = m$.

因为由 $m < n$,Euler 定理和 Fermat 定理有:$c^d \bmod n = m^{e \times d} \bmod n = m^{k \times \phi} + 1 \bmod n = m \bmod n = m$

2.2 Elliptic Curve 算法过程

1) 选取基域 Fq,Fq 的椭圆曲线具体给定为确定的形式. 在 $Eq(a,b)$ 中选一个阶数很大的点 $P(x,$

y),它的阶数为一个大素数 n. 用户执行如下计算生成密钥:在区间[1,$n-1$]中随机选取一个整数 d,计算点 $Q=dP$(d 个 P 相加). 在椭圆曲线密码系统中,公开的信息是 Fq,椭圆曲线的参数为 P,Q,n. 用户的私钥是整数 d,且是保密的.

2) 加密. 将明文分块并数字化,每个数字化明文块的长度不大于[$\log_2 P$],然后对每个明文块一次进行下面的加密变换:选择 $k \in \mathbf{Z}_n$,计算点$(x_1,y_1)=kP$,计算点$(x_2,y_2)=kQ$,如果 $x_2=0$ 则重新选择 $k \in \mathbf{Z}_n$. 计算 $c=m \times x_2$. 密文为 $C=(x_1,y_1,c)$.

3) 解密. 使用私钥 d 对密文(x_1,y_1,c) 计算:点$(x_2,y_2)=d(x_1,y_1)$,再计算 x_2^{-1},于是明文 $m=c \times x_2^{-1}$.

3 OpenPGP 在 JavaMail 中的实现

3.1 发送邮件加密类伪代码

```
public static byte[] encrypt (byte[] 明文,KeyBundle 公钥)
{
List < KeyBundle > list = new ArrayList < KeyBundle > ();
list. add(公钥);
return encrypt(明文,list);
}
```

由于 JavaMail 在传输邮件中用的是 SMTP(Simple Mail Transfer Protocol)协议,不提供加密服务,所以将封装好的加密类代码运用到 JavaMail 提供的方法中.

使用加密邮件发送步骤如下.

1) 获取 Session

① 实行 Authentication 类的子类中的 public PasswordAuthentication getPasswordAuthentication()方法.

② New 一个上面类的实例,设置用户名和密码.

③ New 一个 Properties 对象,设置 mail. smtp. host and mail. smtp. auth 属性.

④ 通过 Session,获取一个 Session 实例.

2) 生成 Message

① 没有附件的邮件

第 1 步. New 一个 MimeMessage 实例(根据 Session).

第 2 步. 给 Message 实例设置 subject、text 属性.

第 3 步. 用封装好的加密方法加密邮件内容.

② 有附件的邮件

第 1 步. 根据 Session new 一个 MimeMessage 实例(Message).

第 2 步. 设置 Message subject 属性.

第 3 步. 用封装好的加密方法加密邮件内容.

第 4 步. New 一个 MimeBodyPart 实例和 Mulipart(MimeMulipart)实例.

第 5 步. 给 MimeBodyPart 实例设置邮件文本内容.

第 6 步. 将 MimeBodyPart 实例,添加到 Mulipart 实例.

第 7 步. 根据附件数循环:

New MimeBodyPart 实例;

获取 FileDatasource;

将 FileDatasource 设置到 MimeBodyPart;

设置 MimeBodyPart 的文件名;

将 MimeBodyPart 添加到 Mulipart.

第 8 步. 将 Mulipart 设置成 MimeMessage 的内容.

3) 发送邮件

① 设置 Message 的 fromAddress, toAddress, ccAddress, bccAddress.

② Transport 发送邮件.

3.2 接收方解密类伪代码

```
public static byte[] decrypt(byte[]加密过的数据,KeyBundle 私钥,String 私钥密码)
{
MessageFactory mf = null;
mf = MessageFactory. getInstance("OpenPGP");
Collection msgs = mf. generateMessages(new ByteArrayInputStream(加密过的数据));
//得到集合中的 EncryptedMessage 对象
Message message = (Message) msgs. iterator(). next();
if (! (message instanceof EncryptedMessage)) {
throw new MessageException ("Not a encrypted message.");
}
EncryptedMessage em = (EncryptedMessage) message;
Message msg = emdecrypt(私钥,私钥密码. toCharArray());
return 解密后的明文;
}
```

JavaMail 在接收邮件时用的是 POP3(Post Office Protocol)协议,需要把封装好的解密方法运用到 JavaMail 提供的 POP3 的接口中.

接受邮件并解密步骤如下:

1) New Properties 实例,设置 mail. pop3. host

的值;

2)获取 Session 实例;

3)根据 Session,获取 Store 实例;

4)连接 Store;

5)获取 Index 文件夹;

6)打开文件夹;

7)获取文件夹里面所有 Message;

8)用封装好的解密方法解密邮件内容;

9)判断 Message 的 MimeType 类型如果是 text/*类型,直接可以从 Message 从获取邮件 from 地址、标题和内容,否则执行下面的步骤;

10)从 Message 中获取 Multipart;

11)遍历 Multipart 中的 BodyPart;

12)判断 BodyPart 的 Disposition 是否是 Part.ATTACHMENT;

13)如果不是,直接获取 BodyPart 里面的 content;

14)否则获取 BodyPart 的文件名和文件流(Inputstream),将流写入本地文件,实现附件的下载.

4 应用试验

图 3 是一个测试邮件加密后的内容展示. 由图 3 可以看出,通过加密后的邮件内容,如果不通过系统的解密,内容很难破译,即使在邮件发送中被不法分子截获,也不会泄露邮件中的信息.

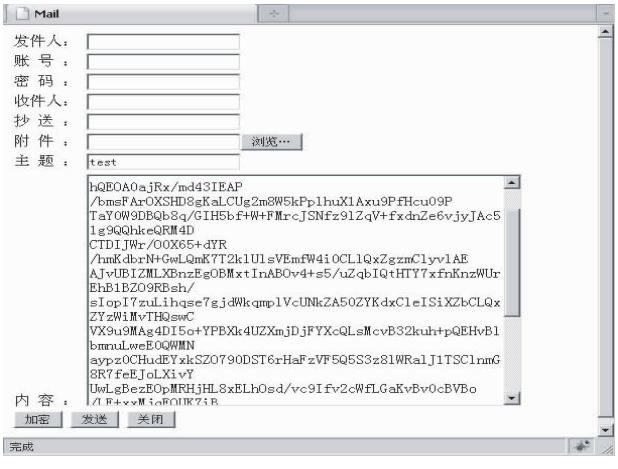

图 3 加密后的内容

Fig. 3 Encrypted content

5 结束语

OpenPGP 作为当前最先进的加密技术已经成为一种安全协议标准. 使用 OpenPGP 加密协议标准,特别是对于电子邮件,可以有效地保护用户的信息安全,从而维护用户的利益. 把 OpenPGP 协议和 Sun 提供标准的邮件服务 JavaMail 结合起来是在邮件安全上的一次探索,将对其加以完善和发展,以在实际中推广应用.

参考文献

References

[1] Abdul-Rahman A. The PGP trust model EDI-Forum[EB/OL]. [2010-08-20]. http://www.cs.ucl.ac.uk/staff/F.AbdulRahman/docs/. ,1997

[2] 龙勤,李斌,潘爱民. 一种实用的前向安全 OpenPGP 扩展[J]. 小型微型计算机系统,2005,26(8):1301-1305
LONG Qin,LI Bin,PAN Aimin. Practical OpenPGP extension with forward security[J]. Mini-Micro Systems,2005,26(8):1301-1305

[3] 王安国,胡铭曾. OpenPGP 协议在电子邮件系统中的应用[J]. 哈尔滨师范大学:自然科学学报,2005,21(4):51-53
WANG Anguo,HU Mingceng. The application of OpenPGPin E-mail system[J]. Natural Science Journal of HarBin Normal University,2005,21(4):51-53

[4] 陈良臣,廖碧成,林碧英,等. 基于 OpenPGP 的安全邮件系统的研究与设计[J]. 计算机应用研究,2007,24(3):157-159
CHEN Liangchen,LIAO Bicheng,LIN Biying,et al. Research and design of secure E-mail system based on OpenPGP[J]. Application Research of Computers,2007,24(3):157-159

[5] The International PGP. PGP user's guide [EB/OL]. [2010-08-20]. http://www.pgpi.org/doc/guide/

[6] Schneie B. 应用密码学:协议、算法与 C 源程序[M]. 吴世忠译. 北京:机械工业出版社,2001
Bruce Schneier. Applied cryptography: Protocols algorithms and source code in C[M]. New York:John Wiley & Sons,Inc,2000

[7] Sun Microsystems,Inc. JavaMail API design specification (Version1.4)[EB/OL]. (2005-04-17)[2010-08-20]. http://java.sun.com/products/javamail/JavaMail-1.2.pdf,2005

[8] 钟珞,刘玲,夏红霞. 基于 JavaMail API 的 Web 邮件系统开发[J]. 武汉理工大学学报,2006,28(6):84-86
ZHONG Luo,LIU Ling,XIA Hongxia. Development research of web mail system based on JavaMail API[J]. Journal of Wuhan University of Technology,2006,28(6):84-86

[9] 孙鹏. JavaMail 规范的研究和实现[D]. 成都:四川大学计算机学院,2005
SUN Peng. The research and implementation of JavaMail specification[D]. Chengdu: College of Computer Science,Sichuan University,2005

Research and implementation of OpenPGP protocol in JavaMail

FU Desheng[1] WU Yiqian[1]

1 School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing 210044

Abstract This paper studies into the encryption principle and several main algorithms in OpenPGP protocol, and presentes the design of JavaMail client based on the OpenPGP protocol, which remedies the security defects of JavaMail. Result shows that the add of OpenPGP protocol into JavaMail can effectively improve the E-mail confidentiality.

Key words OpenPGP; JavaMail; encryption algorithm; security protocol