加解密实验

黎锦灏

上海交通大学　网络空间安全学院

2021 年 5 月

目录

1. PGP 收发加密及签名邮件
2. OpenSSL 加解密

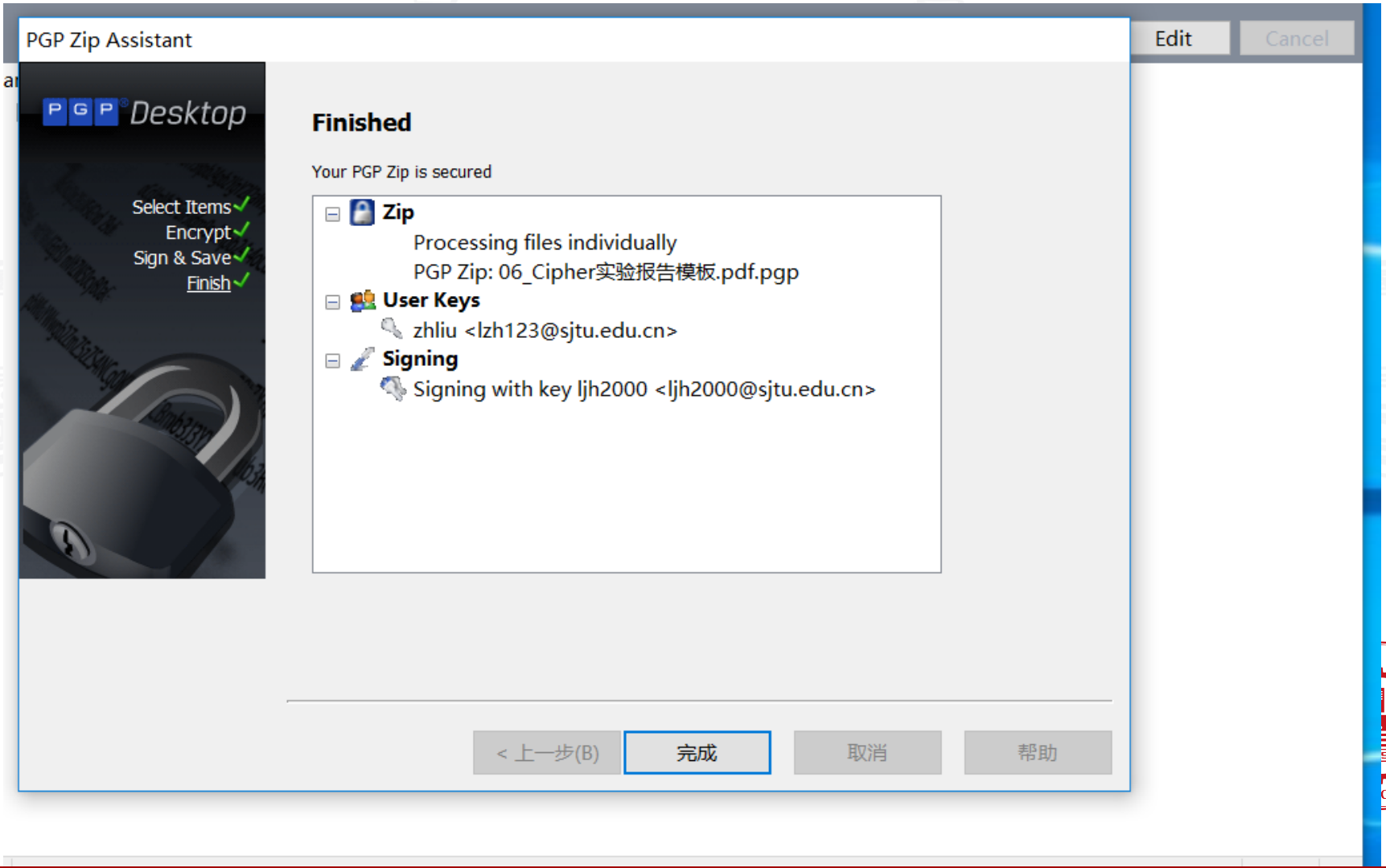
实验场景

- 操作系统：Windows 10 虚拟机
- 加密软件：PGP
- 发送者邮箱（本人）：ljh2000@sjtu.edu.cn
- 接收者邮箱（合作者）：lzh123@sjtu.edu.cn

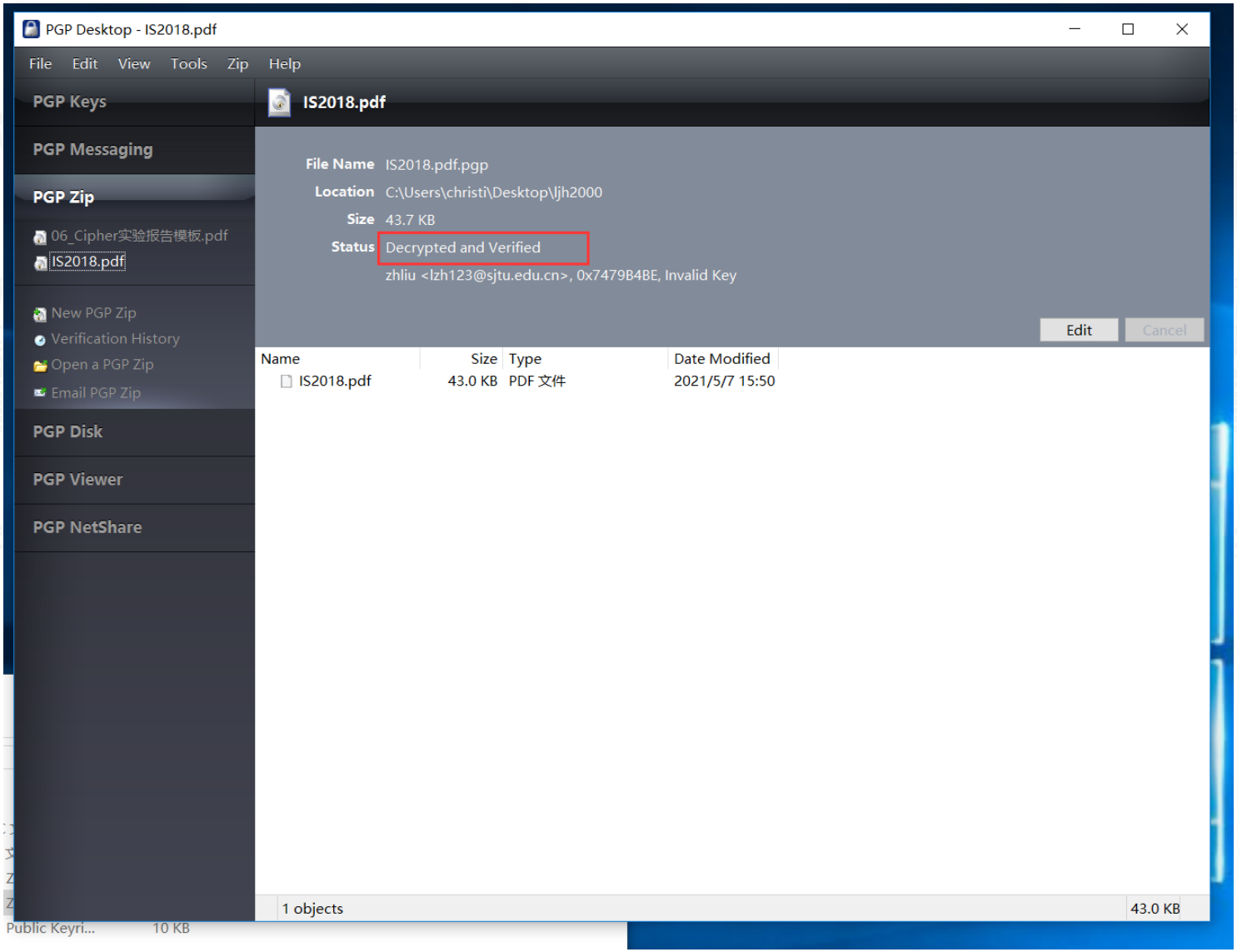
实验过程

1、首先，A 与 B 分别生成公钥并交换信息。
2、对于 A 来说，需要传输文件时，先用 B 的公钥加密，再用 A 自己的私钥签名，将加密并签名过的文件发送给 B。
3、B 在接收到邮件之后，可以用 A 的公钥进行签名验证，并使用自己的私钥解密。
4、对比 B 解密后的文件和 A 发送的源文件是否一致，可以检验 PGP 中邮件加解密和签名验证的实现结果。

实验过程


PGP Zip Assistant
Edit
Cancel
PGP Desktop
Select Items
Encrypt
Sign & Save
Finish
Finished
Your PGP Zip is secured
Zip
Processing files individually
PGP Zip: 06_Cipher实验报告模板.pdf.pgp
User Keys
zhliu <lzh123@sjtu.edu.cn>
Signing
Signing with key ljh2000 <ljh2000@sjtu.edu.cn>
< 上一步(B)
完成
取消
帮助

实验过程


PGP Desktop - IS2018.pdf
File Edit View Tools Zip Help
PGP Keys
PGP Messaging
PGP Zip
06_Cipher实验报告模板.pdf
IS2018.pdf
New PGP Zip
Verification History
Open a PGP Zip
Email PGP Zip
PGP Disk
PGP Viewer
PGP NetShare
IS2018.pdf
File Name IS2018.pdf.pgp
Location C:\Users\christi\Desktop\ljh2000
Size 43.7 KB
Status Decrypted and Verified
zhliu <lzh123@sjtu.edu.cn>, 0x7479B4BE, Invalid Key
Edit
Cancel
Name Size Type Date Modified
IS2018.pdf 43.0 KB PDF 文件 2021/5/7 15:50
1 objects
43.0 KB
Public Keyri... 10 KB

思考题

公私钥在签名、加密过程中的作用

- 双方都需要保存自己的公私钥，并通过交换得到对方的公钥
- 接收方公钥：用于发送方对发送文件进行加密；
- 发送方私钥：用于发送方对发送文件进行签名；
- 发送方公钥：用于接收方对接受文件的签名验证；
- 接收方私钥：用于接收方对加密文件进行解密。

思考题

公私钥面临的安全威胁：
1、公钥本身是公开的，需要注意分发与交换的权限分配。
2、私钥是通过 PGP 软件直接生成的，如果 PGP 软件被攻击，攻击者取得了软件权限或直接得到了生成的私钥，可能导致私钥的泄露。用户对密钥管理的疏忽或在不安全的环境中使用私钥，也可能会造成私钥被窃取。

思考题

公私钥的保护措施：
1、最小化权限原则。
2、使用风险低、安全性好的公私钥管理软件（例如：PGP）。
3、传输私钥时防止窃听，尽量不要随意备份密钥。
4、可以使用信任的第三方证书管理机构来协助管理密钥。

OpenSSL 指令

文件对称加解密：

1. **加解密算法名称：** des3
2. **命令：**

加密：

- openssl enc -des3 -in test.txt -out encrypt.txt -pass pass:123456

解密：

- openssl enc -des3 -d -in encrypt.txt -out decrypt.txt -pass pass:123456

OpenSSL 指令

计算文件摘要

1. 摘要算法名称：sha-1
2. 命令：
   - openssl sha1 -out digest1 test.txt

OpenSSL 指令

OpenSSL 证书管理签发 CA 根证书（命令）:

- openssl req -config openssl.cnf -new -x509 -days 3650 -key ca.key -out ca.crt

签发客户证书（命令）:

- openssl req -config openssl.cnf -new -key client.key -out client.csr
- openssl ca -config openssl.cnf -keyfile ca.key -cert ca.crt -in client.csr -out client.pem -days 730

加解密结果

diff 命令对比加解密后的结果，检查一致性。
文件摘要也可用 diff 命令对比修改前后的差异。

三种加密算法对比

从 OpenSSL speed 的测试中可以看出：
随着明文长度的增加，三种算法的速度都几乎不变。
对于相同大小的明文来说，3DES 比 SHA-1 略慢，且两种算法都比 RSA 快很多。

三种加密算法对比

3DES：对称加密算法

主要用于对数据的直接加密，保证信息的机密性。

密钥长度较短，加解密速度快，处理量大。

但是密钥需要定期更换，大型网络需要保存的密钥量大，管理难度大，且安全性偏低，密钥需要保密。

三种加密算法对比

SHA-1：消息摘要算法

常用的消息摘要算法，提供不可抵赖性，将以变长的消息压缩成一个定长的鉴别码。

不需要密钥，具有不可逆性，能保证消息的完整性，在输入的消息改变时，输出的摘要也会随之变化。

计算速度较快（比需要使用密钥的 MAC 要显著更快），哈希碰撞率较低。

三种加密算法对比

RSA：公钥加密算法

优点：安全性很高，公开公钥，只需要对私钥保密；密钥生命周期较长；可以用于数字签名和密钥交换。

缺点：加密速度慢，数据处理量小，且密钥长度较长。

应用：RSA 算法可以在无密钥传输的情况下实现保密通信，常用于数据加解密、密钥协商交换、数字签名，RSA 加密可以保证信息的不可抵赖性、完整性、机密性。(一般公钥用于加密对称加密中的密钥，私钥用于解密和进行数字签名。)

证书管理的安全威胁

- 可信 SSL 证书的中间人攻击
- OCSP 应答器不可信任
- 对证书撤销列表 CRL 可能存在的重放攻击 (nextUpdate)

证书管理的防范建议

- 证书认证机构需鉴别证书申请者提交的身份信息的真伪;
- 防止私钥泄露和被破解，加强安全管理措施;
- 确保提供服务的 OCSP 应答器是可以信赖的;