**Project2 RSA算法实现**

林士翰 15307130120

**一、概述**

用C/C++实现了128bits的RSA算法，包括素数和密钥对生成。使用RSA算法加密DES密钥，用DES算法对传输内容进行加解密。能够在Linux下编译后执行。

**二、文件描述**

keygen.cpp：素数和密钥对生成源码

encrypt.cpp：使用RSA算法和公钥加密DES密钥，用DES算法对传输内容加密

decrypt.cpp：使用RSA算法和私钥解密DES密钥，用DES算法对传输内容解密

DES.cpp, DES.h：DES算法源码

Integer.h, Prime.h, RSA.h, stdcpp.h, Util.h：算法其他细节的源码

public.key：公钥文件

private.key：私钥文件

DESKey.rsa：用RSA算法和公钥加密后的DES密钥

make.sh：用于编译的shell脚本

**三、使用说明和实验**

**1. 编译**

在Linux或者macOS下运行make.sh即可编译，生成keygen, encrypt, decrypt三个可执行文件。



**2. 密钥对生成**

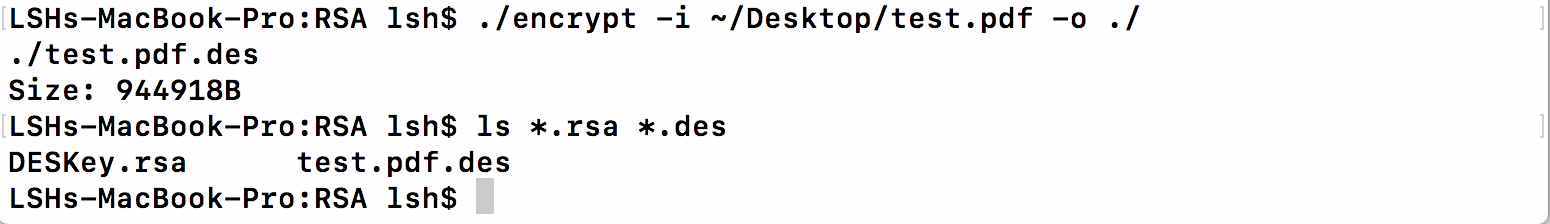
直接运行keygen可以在当前文件夹下生成密钥对文件，public.key 和private.key。



**3. 加密文件**

运行encrypt，并用-i和-o分别指明输入输出文件路径(-o非必选，默认为当前文件夹)。运行后生成被加密后的文件，以及用公钥加密后的DES密钥文件DESKey.rsa。

如下图所示，在~/Desktop/下有一个test.pdf文件，现在对其加密，并将加密后的文件放置在当前文件夹下。

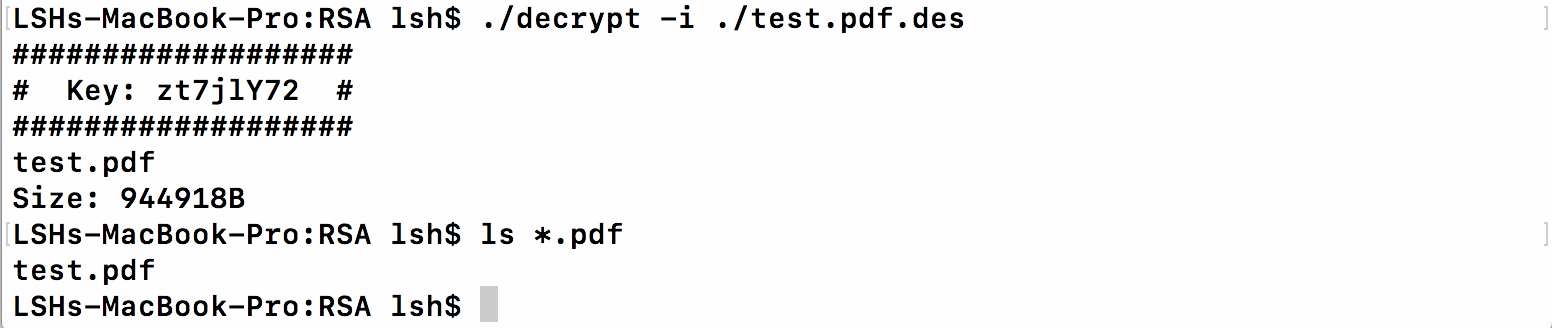


当前文件夹下出现了加密后的文件test.pdf.des和被加密后的DES密钥文件DESKey.rsa。

**4. 解密文件**

运行decrypt，并用-i和-o分别指明输入输出文件路径(-o非必选，默认为当前文件夹)。运行后在终端显示DES密钥，并生成解密后的文件。

如下图所示，将之前加密的文件test.pdf.des解密，并将解密后的文件放置在当前文件夹下。



终端显示了用私钥解密后得到的DES密钥，当前文件夹下出现了原文件test.pdf。