中国科学技术大学计算机学院

《信息安全导论》实验报告

实验题目：密码学及其应用

学生姓名：郭耸霄

学生学号：PB20111712

完成日期：2022 年 3 月 7 日

计算机实验教学中心制

2020 年 09 月

实 验 报 告

11 系 20 级 3 班　　郭耸霄 PB20111712　　2022 年 3 月 7 日

1 实验题目

密码学及其应用。

2 实验目的

1、掌握 OpenSSL 的命令；
2、掌握在 C 程序中使用 OpenSSL 的方法；
3、掌握 PGP 的使用。

3 实验内容

1、使用 OpenSSL 的常用命令；
2、利用 OpenSSL 编程实现 AES 加密、解密；
3、用 PGP 实现加密和解密。

4 实验步骤

4.1~4.4 部分主要描述实验中遇到的问题，4.5 部分主要描述实验的过程及方法。

4.1 准备阶段

问题 0　课程网站上没有 openssl 及 Win32Openssl 的下载链接。

解决方案 0　在互联网上搜索 openssl 的下载链接，最后找到其官网，但其官网没有 1.0.1 版本，故先下载了与之版本较近的 1.1.1。

问题 1　Windows2003 虚拟机的 Internet Explorer 无法访问 https 协议的网站，导致无法下载各种资源。

解决方案 1　使用我们学校使用 http 协议的 ftp 服务器进行文件传输。对我而言，可以利用学校给师生分配的 ftp 空间中 http://home.ustc.edu.cn/~logname 的部分。

4.2 使用 OpenSSL 的常用命令

问题 2　在使用 openssl 指令时,操作系统提示“无法找到组件:没有找到 VCRUNTIME140.dll,因此这个应用程序没有启动。重新安装应用程序可能会修复此问题。”

解决方案 2　1、我先尝试用一台 Windows11 操作系统的 32 位 vcruntime.dll 程序复制到 Windows2003 虚拟机上，但是提示该程序不可用。2、之后我在互联网上搜索解决办法，根据 Microsoft 公司的官方指引，下载了一个 vc_redist.iso 镜像文件进行安装，解决了这个问题。

4.3 利用 OpenSSL 编程实现 AES 的加密、解密

问题 3 编译 cryptoDemo.cpp 时报出连接错误：“LINK : fatal error LNK1104: cannot open file 'libeay32.lib'”。

解决方案 3 我在课程群中提出了这个问题，得到了费尧（PB19051104）同学的帮助，他提示我将 cryptoDemo.cpp 文件中需要的库 “libeay32.lib” 改为 “libcrypto.lib”，便解决了这个问题。究其原因，是 openssl1.0.1 与 1.1.1 版本编译出的库名称不同。

问题 4 运行 cryptoDemo.exe 时，操作系统提示 “损坏的图像：应用程序或 DLL C:\WINDOWS\system32 \VCRUNTIME140.dll 为无效的 Windows 映像。请再检测一遍你的安装盘。”。

解决方案 4 1、由于前面刚刚遇到关于这个文件的问题,我便尝试直接再次运行 vc_redist.iso，但是没有解决这个问题。

2、然后我先删除了 VCRUNTIME140.dll，再启动 vc_dist.iso，选择 “修复”，成功解决了这个问题。

4.4 用 PGP 实现加密和解密

这一部分没有遇到问题。

4.5 修改例程 cryptoDemo.cpp 为 encfile.cpp：从命令行接受 3 个字符串类型的参数：参数 1，参数 2，参数 3。参数 1=enc 表示加密，参数 1=dec 表示解密；参数 2 为待加密、解密的文件名；参数 3 为密码。以文件 cryptoDemo.cpp 为测试文件，以你的学号为密码，验证你的程序 encfile.cpp 的正确性

问题 5 运行解密程序时抛出异常。

解决方案 5 经过仔细调试，发现 flag 的值始终为 false。其原因是我使用了 “=” 对两个字符串进行比较。将其改为库函数 strcmp 便解决了问题。

问题 6 解密后的文件只是原文件的前一部分，而后一部分丢失。

解决方案 6 问题出在向文件的输出。我开始使用 fputs 输出，并用 strlen 计算字符串长度，二者均是遇到字符 “'\0'” 停止。将输出方法改为 fwrites，计算字符串长度使用直接赋给文件字符数（采用 fseek 文件结束符），解决了这个问题。由于本实验主要在密码学部分，故程序的健壮性有待提高，比如 fopen 没有成功打开文件时应该抛出异常等等。我的 encfile.cpp 如下：

```
// encfile.cpp : Defines the entry point for the console application.
// Windows: cl encfile.cpp
// Linux: gcc -o encfile encfile.cpp -lcrypto

#include <memory.h>
```

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

#include "openssl\aes.h"

#pragma comment(lib,"libcrypto.lib")

void testAes(char inString[], int inLen, char passwd[], int pwdLen,bool flag)
{
  int i,j, len, nLoop, nRes;
  char enString[1024];
  char deString[1024];

  unsigned char buf[16];
  unsigned char buf2[16];
  unsigned char aes_keybuf[32];
  AES_KEY aeskey;

  // 准备32字节(256位)的AES密码字节
  memset(aes_keybuf,0x90,32);
  if(pwdLen<32){ len=pwdLen; } else { len=32;}
  for(i=0;i<len;i++) aes_keybuf[i]=passwd[i];
  // 输入字节串分组成16字节的块
  nLoop=inLen/16; nRes = inLen%16;
  if(!flag){
  // 加密输入的字节串
  AES_set_encrypt_key(aes_keybuf,256,&aeskey);
  for(i=0;i<nLoop;i++){
    memset(buf,0,16);
    for(j=0;j<16;j++) buf[j]=inString[i*16+j];
    AES_encrypt(buf,buf2,&aeskey);
    for(j=0;j<16;j++) enString[i*16+j]=buf2[j];
  }
  if(nRes>0){
    memset(buf,0,16);
    for(j=0;j<nRes;j++) buf[j]=inString[i*16+j];
    AES_encrypt(buf,buf2,&aeskey);
    for(j=0;j<16;j++) enString[i*16+j]=buf2[j];
    //puts("encrypt");
  }
  enString[i*16+j]=0;
  FILE*fp=fopen("encrypto.txt","wb");
```

```
    fwrite(enString,inLen,1,fp);
    fclose(fp);
    free(fp);
    }else{
    // 密文串的解密
    AES_set_decrypt_key(aes_keybuf,256,&aeskey);
    for(i=0;i<nLoop;i++){
      memset(buf,0,16);
      for(j=0;j<16;j++) buf[j]=inString[i*16+j];
      AES_decrypt(buf,buf2,&aeskey);
      for(j=0;j<16;j++) deString[i*16+j]=buf2[j];
    }
    if(nRes>0){
      memset(buf,0,16);
      for(j=0;j<16;j++) buf[j]=inString[i*16+j];
      AES_decrypt(buf,buf2,&aeskey);
      for(j=0;j<16;j++) deString[i*16+j]=buf2[j];
      //puts("decrypt");
    }
    deString[i*16+nRes]=0;
    FILE*fp=fopen("decrypto.txt","wb");
    fwrite(deString,inLen,1,fp);
    fclose(fp);
    free(fp)
    }
    //比较解密后的串是否与输入的原始串相同
    //if(memcmp(inString,deString,strlen(inString))==0)
    //{ printf("test success\r\n");} else { printf("test fail\r\n");}
      //printf("The original string is:\n %s ", inString);
      //printf("The encrypted string is:\n %s ", enString);
      //printf("The decrypted string is:\n %s ", deString);
}

int main(int argc, char* argv[])
{

    //char inString[] = "This is a sample. I am a programer.\n";
    //char passwd[] = "0123456789ABCDEFGHIJK";
    bool flag=strcmp("enc",argv[1]);
    FILE*fp=fopen(argv[2],"rb");
    fseek(fp,0,SEEK_END);
    int file_size=ftell(fp);
    char*inString=(char*)malloc(file_size*sizeof(char));
```

```
    fread(inString,file_size,1,fp);
    fclose(fp);
    free(fp);
    testAes(inString, file_size, argv[3], strlen(argv[3]),flag);

    return 0;
}
```

以下是实验过程截图：

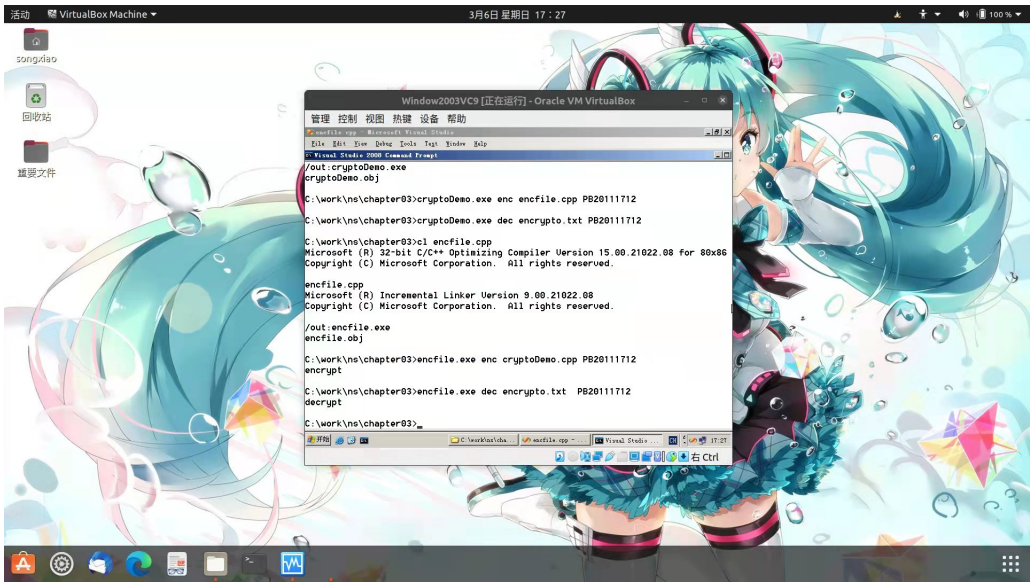

图 1: encfile.cpp 的编译运行

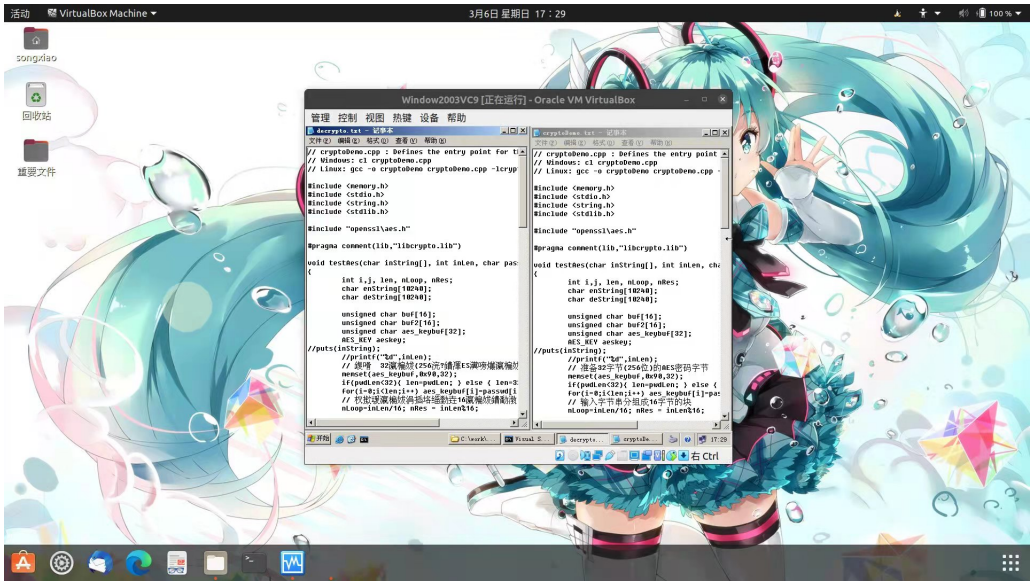

图 2: 加密又解密的文件与原文件对照

两文件中的汉字并不相同，原因在于编码方式不同。但是可以说明二者的二进制文本相同。

5 实验总结

互助学习 这次实验过程中同学们在课程群中展开了激烈的讨论，并解决了很多共有的问题。互助学习是我们值得学习的学习手段。

资料查询 在向同学询问解决方案不成功的同时，也需要在互联网上主动搜索资料。如何在海量信息中获取有效信息成了解决问题的关键。

基础知识 前文所述的问题 5、6 占用了我很多时间。它们本应该在程序设计课程中解决。这体现了学习是逐步推进的。如果基础不牢，就会地动山摇。

6 意见建议

实验文档 应该及时跟进时代，要解决好 https 无法访问的问题。这一部分不属于本课程内容，并对实验进度造成不小的影响。所幸我们学校的 ftp 空间使用的是 http 协议，否则将造成更多困扰。