

网络安全实验报告

题 目 基于口令的认证过程实现

专 业 信息安全 .

指 导 教 师 王彦 .

# 一、实验目的

掌握随机函数的使用，掌握散列函数，加解密函数的使用。包的格式与发包的顺序，可以等同于协议的三要素。掌握程序与数据库的链接。

**二、实验内容**

1.客户端输入用户名，口令，随机产生认证码，使用散列函数计算用户名与口令的散列值1，使用散列值1与认证码计算散列值2，将用户名，散列值2，认证码明文传送到服务器端。

2.服务器端以数据库（如access）保存用户名和散列值1的对应关系。收到客户端信息后，以同样的方法计算散列值2’。 如散列值2’=散列值2，则认证成功，成功后用散列值1加密认证码发送给客户端。客户端解密后写到指定文件。

# 三、实验过程

**基于口令的认证过程实现实验基本信息：**

实验环境：Ubuntu16.04 x64

编程语言：C 语言

## 1. 需求分析

实验需要编写服务端和客户端程序，模拟完成基于口令的认证服务过程实现。实现过程中的重点有：

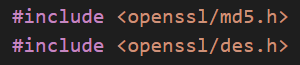
1. 散列函数和对称加密函数:客户端需要散列计算用户名与口令、散列值1与认证码，服务端需要散列计算散列值1与认证码；二者都需要以散列值1为对称密钥加解密认证码。
2. 高级语言与数据库的链接和对数据的存储和获取：实验需要我们通过编程，使用高级语言操纵数据库，进行数据的查询等操作。
3. 模拟协议的三要素进行数据交互：模拟协议，设计数据的语法、语义和时序，保证通讯双方能得到期望得到的数据，而不会因为格式、语序等过程导致一方得到失序的或错误的数据。
4. 随机函数

## 2. 程序结构

##### 2.1 解决重要问题

1.散列函数和对称加密函数:

使用openssl的MD5库作为散列函数以及DES库作为对称加密函数。



2.高级语言与数据库的链接和对数据的存储和获取：

在虚拟机上使用MYSQL数据库进行数据存储。连接数据库头文件”mysql.h”连接数据库。

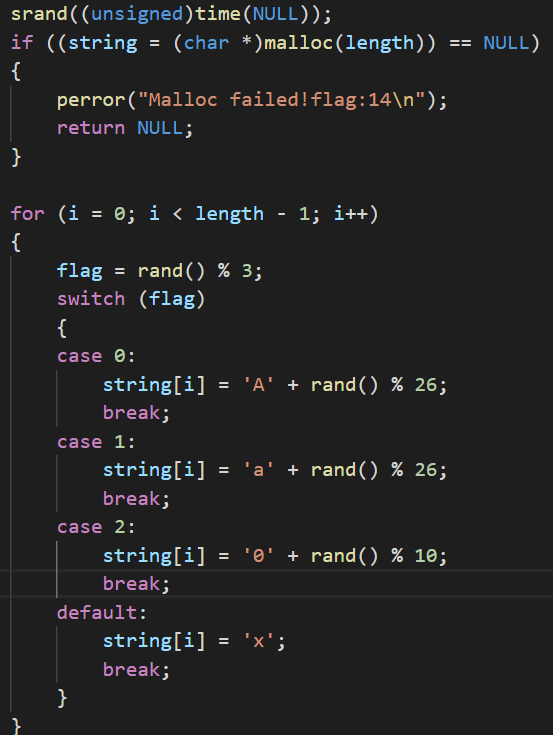
IMG_256

3.模拟协议的三要素进行数据交互：

服务端和客户端进行三次数据交互：

|  |  |  |
| --- | --- | --- |
| 发送方 | 接受方 | 数据内容 |
| 客户端 | 服务端 | 用户名 |
| 服务端 | 客户端 | “GET SEQ1” |
| 客户端 | 服务端 | 认证码 |
| 服务端 | 客户端 | “GET SEQ2” |
| 客户端 | 服务端 | 散列值2 |
| 服务端 | 客户端 | “GET SEQ3” |
| 服务端 | 客户端 | DEC散列值1（认证码） |

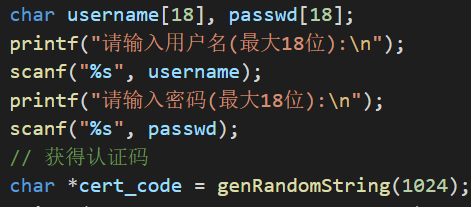
4.随机函数：输入期望得到随机串长度，malloc创造足够空间；以时间为随机种子，将随机数伪平均分配到所有字符。

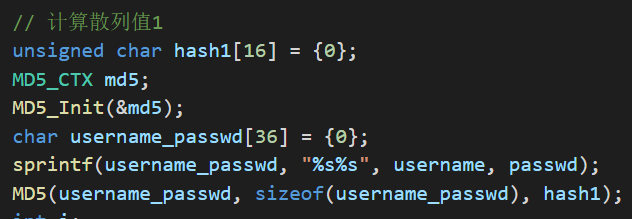


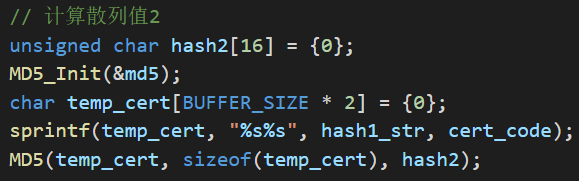
##### 2.1 服务端和客户端实现

1.客户端

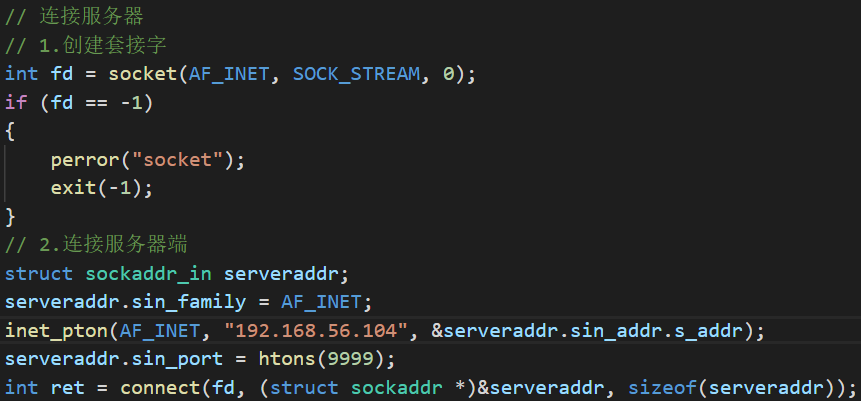
1）客户端用户输入用户名和密码（最长18位），使用随机函数获得1024长度随机串作为认证码。



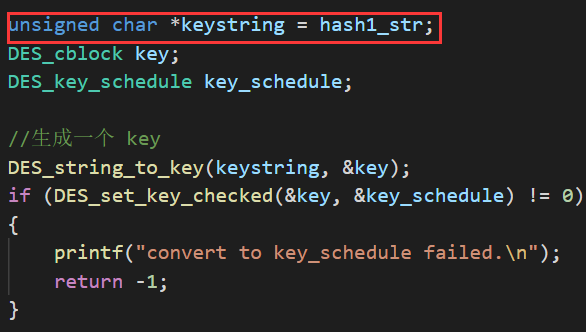
1. 连接用户名和密码的字符串，使用MD5函数进行散列得到散列值1。 
2. 连接散列值1和认证码，得到散列值2

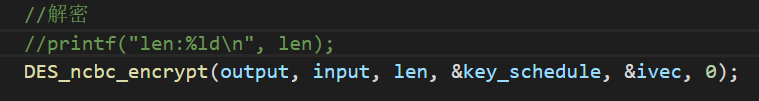


1. 连接服务器，按照模拟协议传输数据。

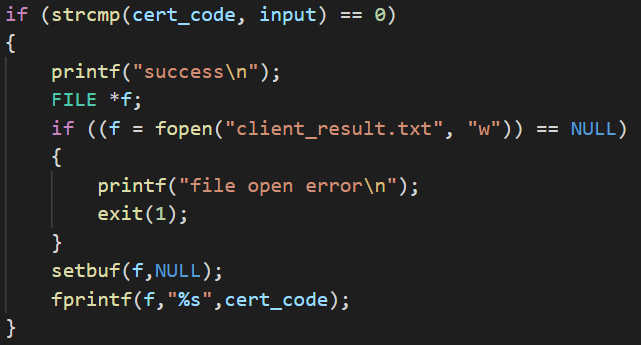


1. 三次数据传输完毕后，解密服务器传输来的加密后的数据。使用散列值1生成对称密钥key，解密传来数据。



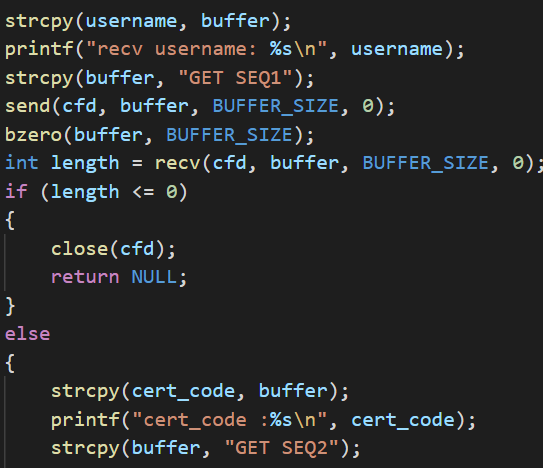


1. 将解密后的数据与认证码比较，若相同服务器认证成功，将认证码写入指定文件。

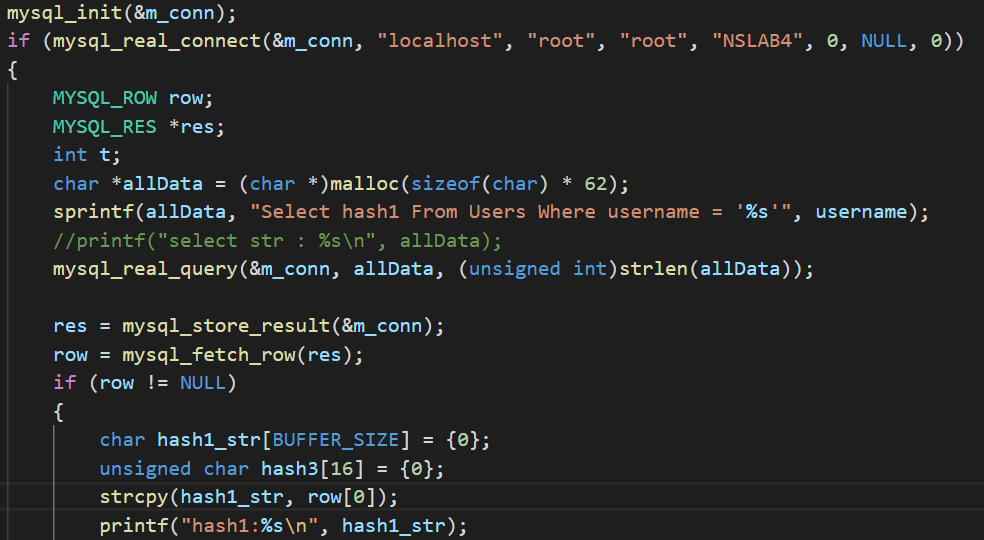


2.服务端

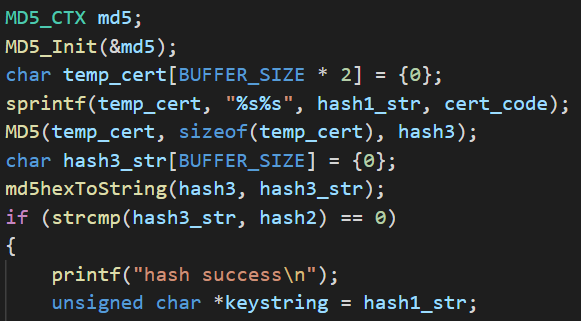
1）监听9999端口，等待客户端连接。若有连接，创建子线程，转到子函数多线程处理。处理函数中，等待客户端传来的数据，并分别回应GET SEQn（n=1,2,3）。



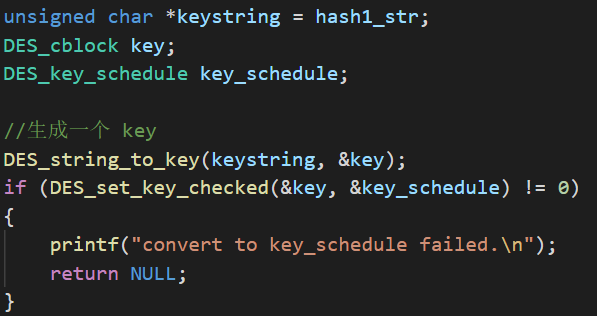
1. 连接数据库，使用客户端传来的用户名创建查询语句，找到服务器数据库中存储的与该用户名对应的散列值1。

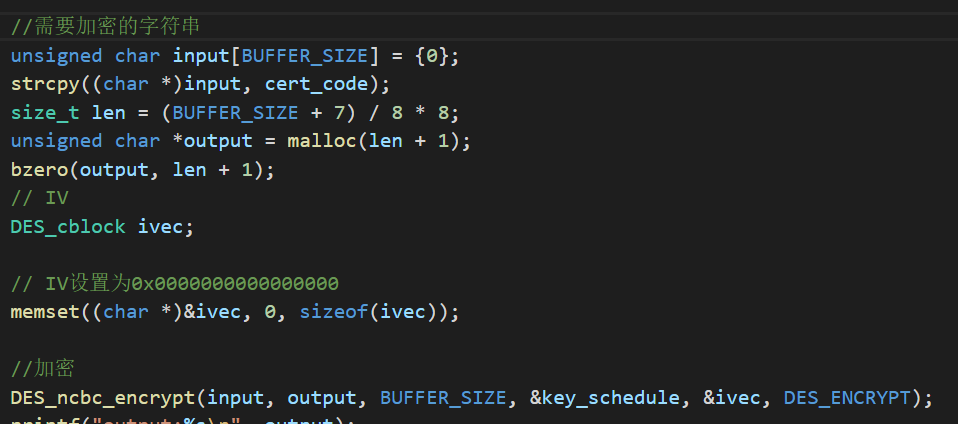


1. 使用与客户端相同的算法计算散列值2，并与客户端第三步传来的散列值2进行比较，若相同则用户名和密码对应，用户认证成功。



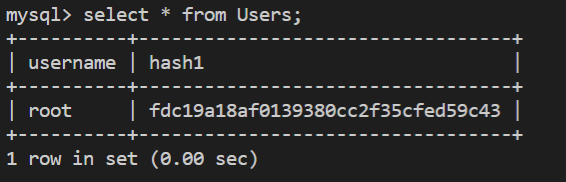
1. 使用散列值1生成对称密钥key，加密认证码。加密完成后发送给客户端。





# 四、实验结果

服务端192.168.56.103，数据库MYSQL，存储用户root，密码root。

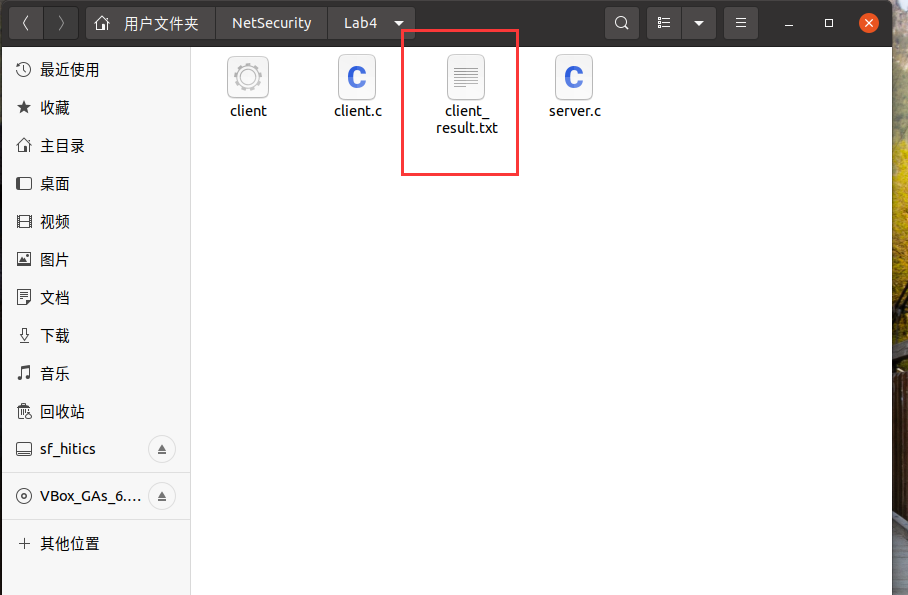


在服务端启动程序server.c

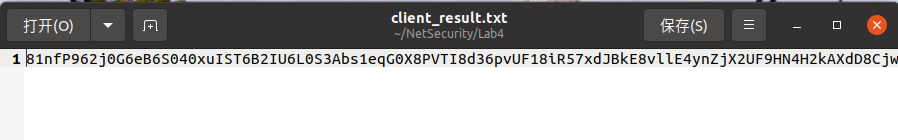
客户端192.168.56.104，启动服务端，输入用户名和密码。



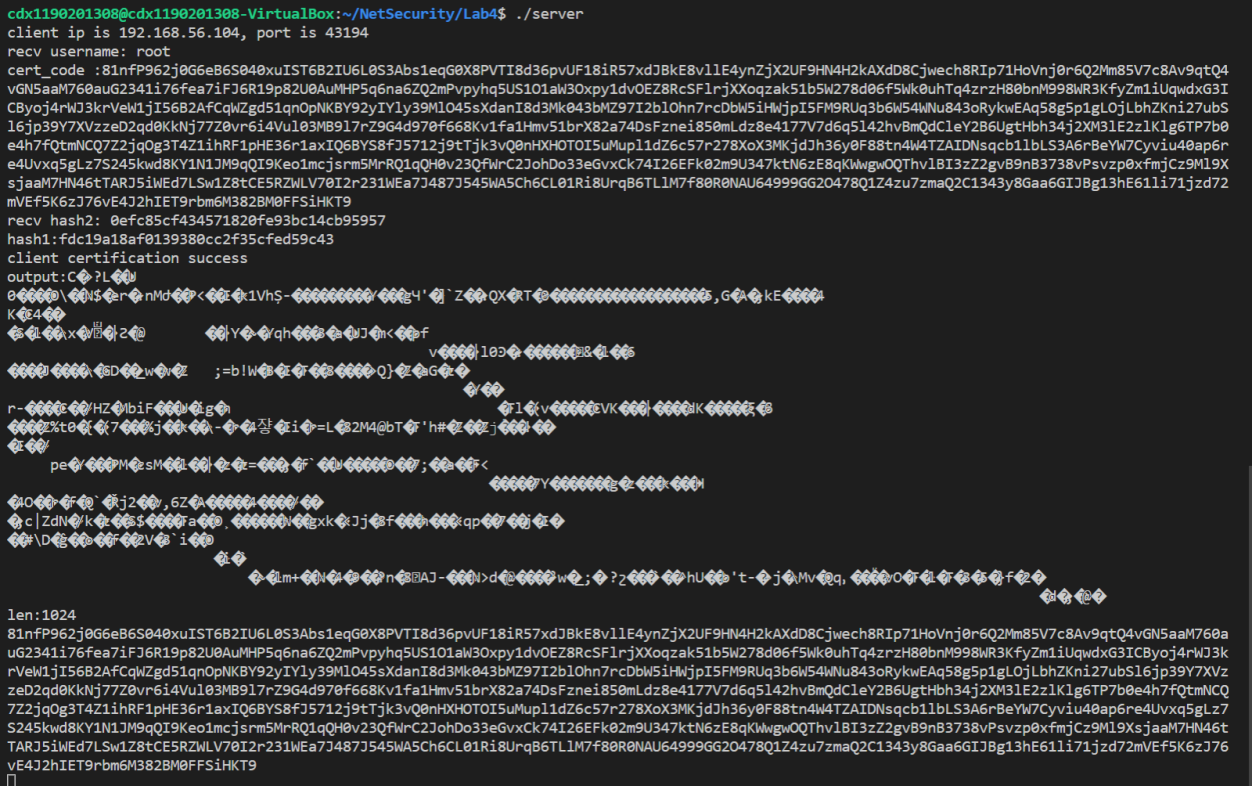
验证成功，客户端在同路径下得到输出文件client\_result.c



内含解密的认证码



服务端终端结果：



注：程序中的多余printf为为了方便查看通信双方数据是否一致所设置，可以注释掉。

# 五、心得体会 （出现问题分析）

学习了随机函数的编写，了解了散列函数、对称加密函数的使用方并了解Openssl库函数。通过定义包的格式与发包的顺序，进行服务端和客户端多端通讯。掌握了程序与数据库的链接。

问题：散列函数结果为16位unigned char，即16\*8位=128位，结果不能作为字符串输出，需要通过相关函数转为字符串，否则会出现乱码。