

# 《计算机系统安全》 实验指导书



班级： 1504201

学号： 150120526

姓名： 殷悦

**哈尔滨工业大学（威海）**

## 实验报告一

同组人:

指导教师: 董开坤

日期:

实验题目	基于 S/Key 协议的身份认证系统设计与实现
实验目的	通过设计和实现一个简单的身份认证系统，加深对信息系统安全认证的概念和基本原理的理解，培养学生基本的安全信息系统设计与实现能力。
仪器设备	Python 3, php, mysql
<p><b>实验原理</b></p> <ol style="list-style-type: none"><li>1. S/Key 协议的设计与实现</li><li>2. 界面的设计</li><li>3. 身份认证的基本原理和过程</li><li>4. 客户端与服务器网络编程实现</li><li>5. 本地文件读取</li><li>6. 数据库的增改查</li><li>7. 日志系统的设计</li></ol>	

## 实验内容及步骤

### 1、身份认证系统设计

设计身份认证系统的功能、主要界面、主要软件模块，以及采用的认证技术路线和方法。

### 2、编程实现所设计的身份认证系统

在 Python, PHP, MySql 程序设计环境下，编程实现基于 S/Key 协议的身份认证系统。实现的身份认证系统具备以下功能：

- (1) 支持用户名/口令/验证码机制的身份认证；
- (2) 满足动态口令的技术要求；
- (3) 当前口令序列使用完毕后能够继续协商；
- (4) 记录用户登录日志，支持日志查看。

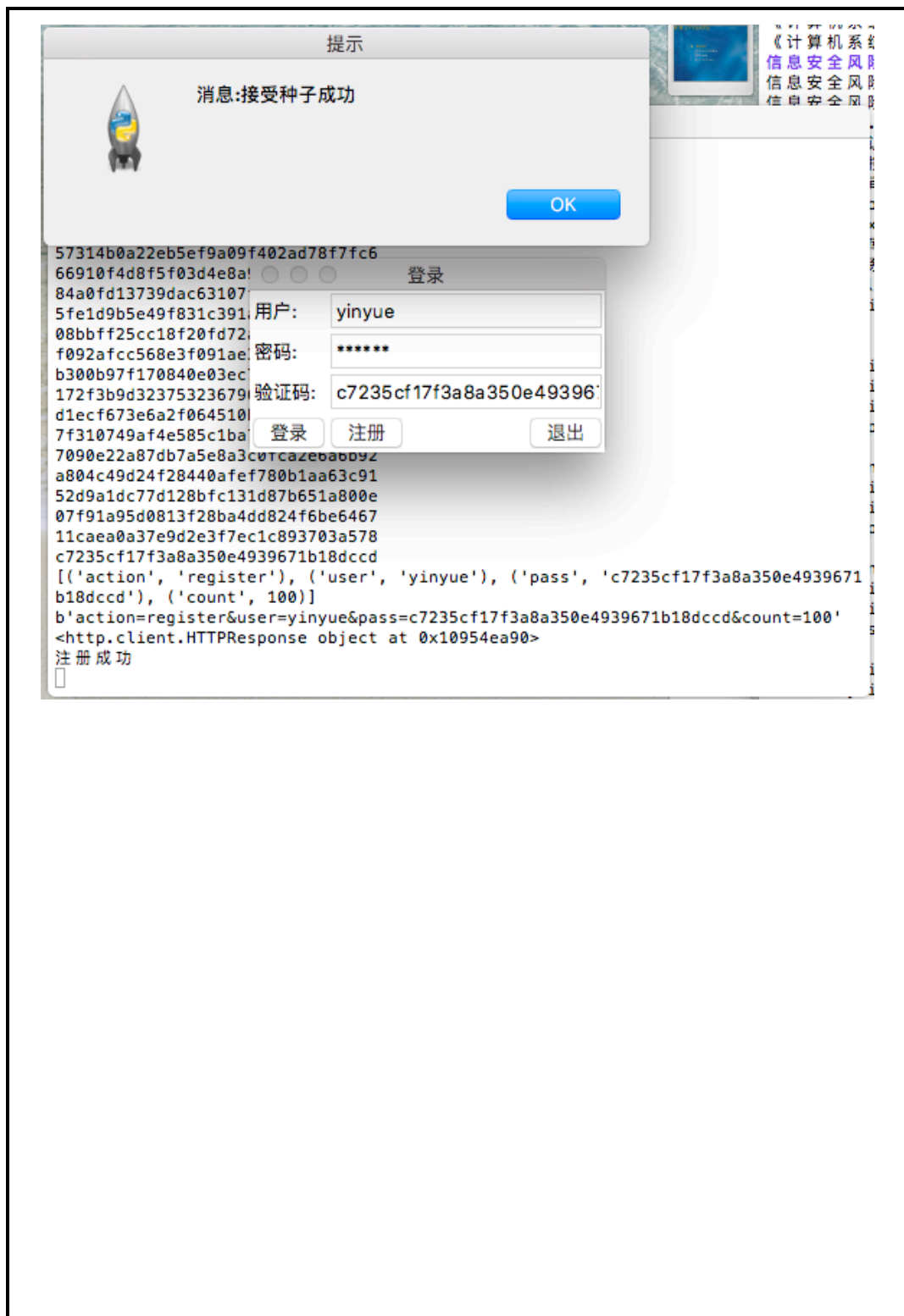
### 3、软件测试及优化。

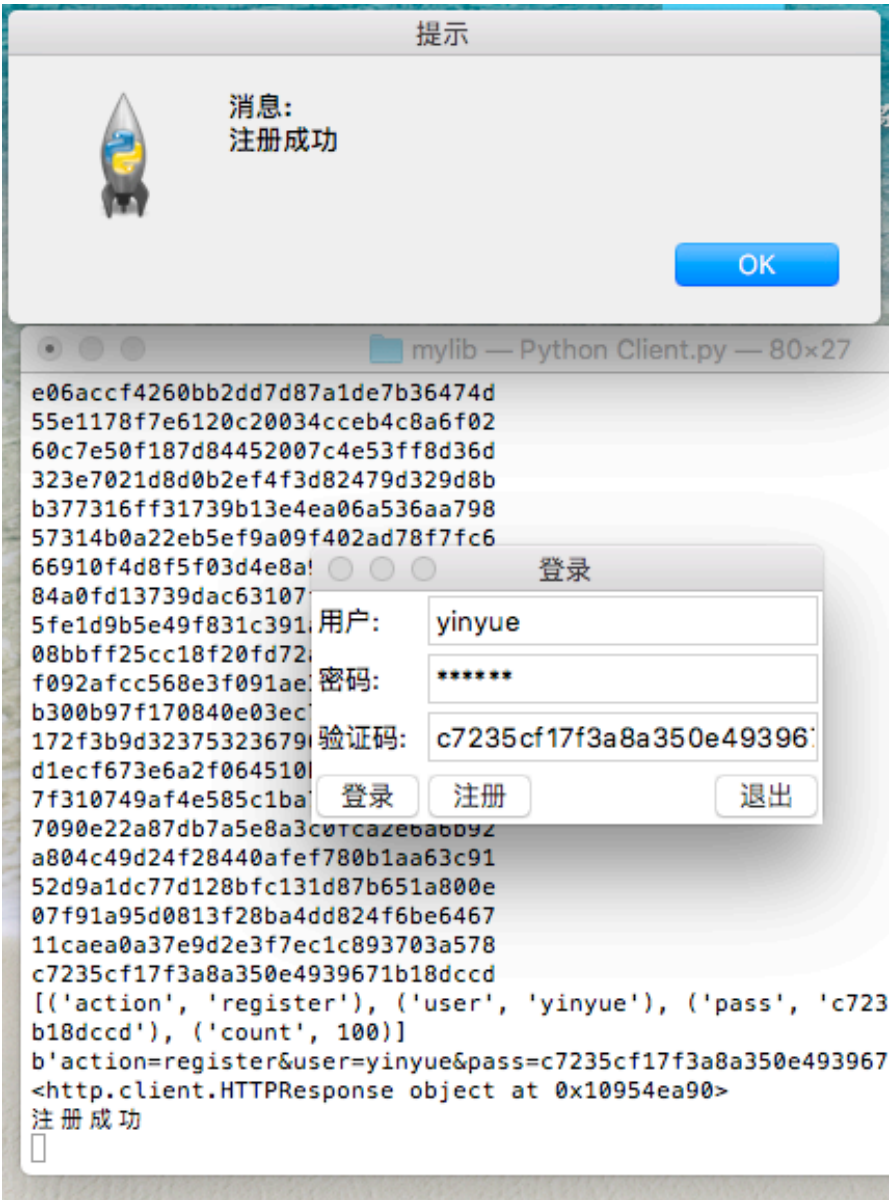
在实验室所在的局域网内，对设计实现的身份认证系统进行简单的测试，对发现的问题和软件存在的缺陷进行改进和优化。

1.程序:客户端用Python3编写，服务端用PHP编写，数据库使用MySQL，界面使用了Python原生的tkinter库，网络使用Python的urllib库发送POST请求和服务端通信，哈希部分采用hashlib



2.注册账号:客户端从服务器请求种子，获取N的值，得到种子后，保存到本地，对密码和种子进行异或操作，然后对异或后的密码进行N次哈希操作，将用户名和密码发送至服务器，然后PHP将用户名和密码保存到数据库，如果数据库抛出失败信息为用户名重复，则提示注册失败，否则提示注册成功





The screenshot displays a Python Client.py window with a list of hexadecimal strings. A '提示' (Message) dialog box shows '消息: 注册成功' (Message: Registration successful) with an 'OK' button. A '登录' (Login) dialog box is also present, with fields for '用户:' (User) containing 'yinyue', '密码:' (Password) containing '\*\*\*\*\*', and '验证码:' (Verification code) containing 'c7235cf17f3a8a350e493967'. The '登录' (Login) button is highlighted.

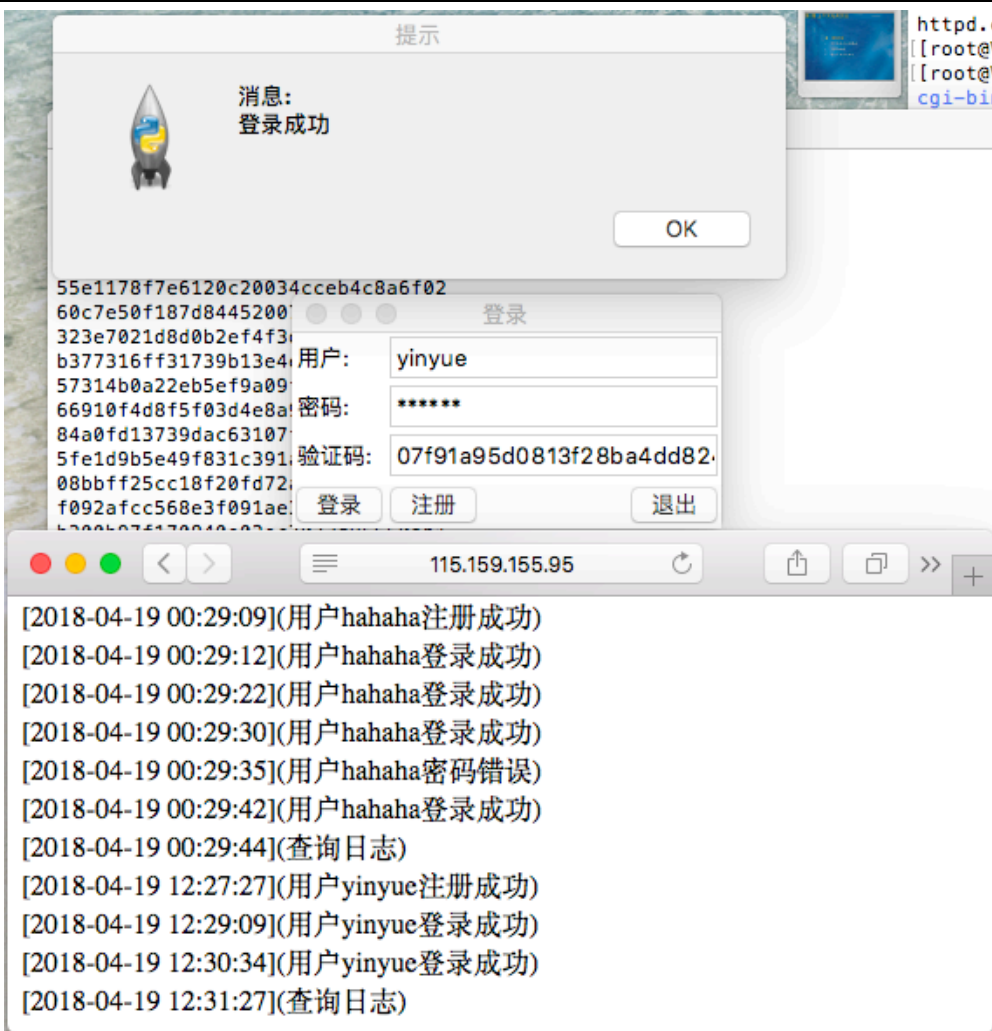
3.登录:程序从本地文件读出seed和n的值,将密码和seed进行异或操作,然后对操作后的结果执行n-1次哈希操作,将结果发送服务器进行验证,服务器对得到结果进行1次哈希操作,如果操作后的值 and 数据库中保存的值相同,则认证成功,将密码变为客户端新发的值,对N进行自减一操作,客户端收到消息后,判断如果成功,则对n进行自减一操作,然后写入本地文件



The screenshot displays a web application interface. At the top, a modal dialog box titled "提示" (Notice) shows a rocket icon and the message "消息: 登录成功" (Message: Login successful), with an "OK" button. Below this, a "登录" (Login) form is visible, containing fields for "用户:" (User) with the value "yinyue", "密码:" (Password) with masked characters "\*\*\*\*\*", and "验证码:" (Captcha) with the value "07f91a95d0813f28ba4dd82". The form has buttons for "登录" (Login), "注册" (Register), and "退出" (Logout). In the background, a large block of text displays a long hexadecimal string and a JSON-like object representing a successful login action.

```
55e1178f7e6120c20034cceb4c8a6f02
60c7e50f187d8445200
323e7021d8d0b2ef4f3
b377316ff31739b13e4
57314b0a22eb5ef9a09
66910f4d8f5f03d4e8a
84a0fd13739dac63107
5fe1d9b5e49f831c391
08bbff25cc18f20fd72
f092afcc568e3f091ae
b300b97f170840e03ec7072095b50ab1
172f3b9d323753236796804128b865cc
d1ecf673e6a2f064510b50a16f7bf2ec
7f310749af4e585c1ba716f7a10f332b
7090e22a87db7a5e8a3c0fca2e6a6b92
a804c49d24f28440afef780b1aa63c91
52d9a1dc77d128bfc131d87b651a800e
07f91a95d0813f28ba4dd824f6be6467
[('action', 'login'), ('user', 'yinyue'), ('pass', '07f91a95d0
6467')]
b'action=login&user=yinyue&pass=07f91a95d0813f28ba4dd824f6be64
```

4.日志:系统每进行一次注册登录或者查询日志操作,不论操作成功与失败,都在数据库中写入一条记录,日志中记录了操作时间,操作类型,操作对象,操作结果



The screenshot shows a web application interface. At the top, a message box says "提示: 登录成功" (Notice: Login successful) with an OK button. Below it is a login form with fields for "用户:" (User) containing "yinyue", "密码:" (Password) containing "\*\*\*\*\*", and "验证码:" (Captcha) containing "07f91a95d0813f28ba4dd82". There are buttons for "登录" (Login), "注册" (Register), and "退出" (Logout). Below the login form is a log display showing various system events with timestamps and user names.

5.数据库:用户表中存放了用户名,处理后密码,N的值,日志表中存放了日志时间和日志内容

Column	Datatype	PK	NN	UQ
user	VARCHAR(20)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
pass	VARCHAR(64)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
count	INT(11)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Column	Datatype	PK	NN	UQ
time	TIMESTAMP	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
record	VARCHAR(45)	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Limit to 1000 rows

1 • `SELECT * FROM test.user;`

100% 25:1

Result Grid Filter Rows: Search Edit:

user	pass	count
test	098f6bcd4621d373cade4e832627b4f6	87
123	123	100
234	36e1a1a6f63d5b53ac5cf1957f9950f7	100
345	f8c73906dea9e69125eaec39f2e24468	100
456		95
yinyue	07f91a95d0813f28ba4dd824f6be6467	98
nihao	ff3c40cf73677347d5bab555d31a05d0	82
hahaha	125c89ee770422a14df3428d019f9cc9	96
NULL	NULL	NULL

1 • `SELECT * FROM test.log;`

100% 1:1

Result Grid Filter Rows: Search

time	record
2018-04-19 00:24:09	用户nihao登录成功
2018-04-19 00:24:21	用户nihao登录成功
2018-04-19 00:24:28	用户nihao登录成功
2018-04-19 00:28:43	用户nihao登录成功
2018-04-19 00:28:47	用户nihao密码错误
2018-04-19 00:28:50	用户nihao密码错误
2018-04-19 00:28:52	查询日志
2018-04-19 00:29:09	用户hahaha注册成功
2018-04-19 00:29:12	用户hahaha登录成功
2018-04-19 00:29:22	用户hahaha登录成功
2018-04-19 00:29:30	用户hahaha登录成功
2018-04-19 00:29:35	用户hahaha密码错误
2018-04-19 00:29:42	用户hahaha登录成功
2018-04-19 00:29:44	查询日志
2018-04-19 12:27:27	用户yinyue注册成功
2018-04-19 12:29:09	用户yinyue登录成功
2018-04-19 12:30:34	用户yinyue登录成功
2018-04-19 12:31:27	查询日志
NULL	NULL



```

mylib — root@VM_21_212_centos:/var/mydocker/www/html — ssh root@wifewifi.com — 104x51

?php
$host="localhost";
$db_user="root";
$db_pass="";
$db_name="test";
$conn=mysql_connect($host,$db_user,$db_pass) or die("error connecting");
mysql_select_db($db_name);
mysql_query("set names UTF8");
header("Content-Type: text/html; charset=utf-8");
if($_POST['action']=='login'){
    $user=$_POST['user'];
    $pass=$_POST['pass'];
    $passmd5=md5($pass);
    $res=mysql_query("select * from user where user='".$user.'" and pass='".$passmd5.'"");
    if(mysql_fetch_array($res)){
        //echo $user.':'.$pass.':'.$passmd5;
        if(mysql_query("update user set pass='".$pass."',count=count-1 where user='".$user.'"")){
            echo '登录成功';
            mysql_query("insert into log(record) value ('用户".$_POST['user'].".登录成功')");
        }else{
            echo '密码正确,写入数据库出错';
            mysql_query("insert into log(record) value ('用户".$_POST['user'].".密码正确,写数据库失败')");
        }
    }else{
        echo '密码错误';
        mysql_query("insert into log(record) value ('用户".$_POST['user'].".密码错误')");
    }
}else if($_POST['action']=='register'){
    if(mysql_query("insert into user(user,pass,count) values('".$_POST['user']. "','"._POST['pass']. "','"._POST['count']. "')")){
        echo '注册成功';
        mysql_query("insert into log(record) value ('用户".$_POST['user'].".注册成功')");
    }else{
        echo '注册失败';
        mysql_query("insert into log(record) value ('用户".$_POST['user'].".注册失败')");
    }
}else if($_POST['action']=='seed'){
    echo "seedseed";
    mysql_query("insert into log(record) value ('用户获取seed')");
}else{
    mysql_query("insert into log(record) value ('查询日志')");
    $res=mysql_query("select * from log");
    while($row=mysql_fetch_array($res)){
        echo '['.$row['time'].']['.$row['record'].']<br>';
    }
}

1,1      All

```

```
mylib — Client.py (~/Documents/lear...三/信息系统安全/mylib) - V...
1  # -*- coding: utf-8 -*-
2  import os
3  import socket
4  import hashlib
5  import tkinter.messagebox
6  from tkinter import *
7  from urllib import request,parse,error
8
9  class Login(Frame):
10     url='http://115.159.155.95/skey.php'
11     N=0;
12     def __init__(self, master):
13         # abc
14         frame=Frame(master)
15         frame.pack()
16         self.lab1=Label(frame,text = "用户:")
17         self.lab1.grid(row=0,column=0,sticky=W)
18         self.lab2=Label(frame,text="密码:")
19         self.lab2.grid(row=1,column=0,sticky=W)
20         self.lab2=Label(frame,text="验证码:")
21         self.lab2.grid(row=2,column=0,sticky=W)
22         self.ent1=Entry(frame)
23         self.ent1.grid(row=0,column=1,sticky=W)
24         self.ent2=Entry(frame,show="*")
25         self.ent2.grid(row=1,column=1,sticky=W)
26         self.ent3=Entry(frame)
27         self.ent3.grid(row=2,column=1,sticky=W)
28         self.button1=Button(frame,text="登录",command=self.Log)
29         self.button1.grid(row=3,column=0,sticky=W)
30         self.button2=Button(frame,text="注册",command=self.Reg)
31         self.button2.grid(row=3,column=1,sticky=W)
32         self.button3=Button(frame,text="退出",command=frame.quit)
33         self.button3.grid(row=3,column=1,sticky=E)
34     def md5n(self,pas,n):
35         while n>0:
36             n=n-1
37             hash=hashlib.md5()
38             hash.update(pas.encode("utf8"))
39             pas=hash.hexdigest()
40             print(pas)
41         return pas
42
43     def Log(self):
44         s1=self.ent1.get()
45         s2=self.ent2.get()
46         s3=self.ent3.get()
47         info=[]
48         info.append(('action','login'))
```

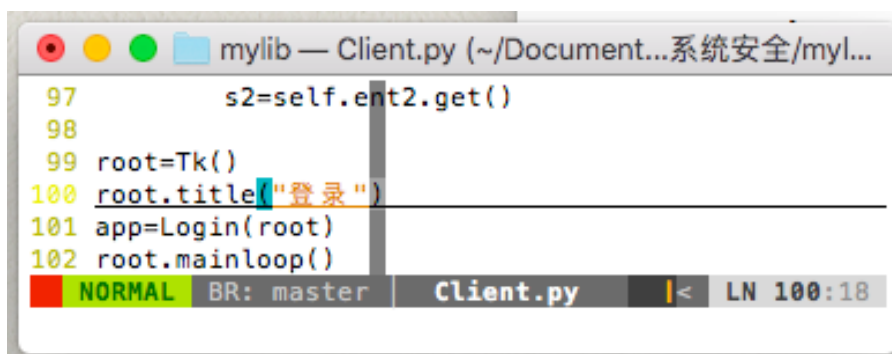
**NORMAL** BR: master **Client.py** |<8 python 31% LN 32:65

```

mylib — Client.py (~/Documents/learn...三/信息系统安全/mylib) - VI...
49     info.append(('user', s1))
50     tkinter.messagebox.showinfo('提示', "消息:接受种子成功")
51     seed='seedseed'
52     with open(s1 + '.txt', 'r') as f:
53         N=eval(f.read())
54         f.close()
55         md5npas = self.md5n(s2, N - 1)
56         self.ent3.delete(0, len(s3))
57         self.ent3.insert(0, md5npas)
58         info.append(('pass', md5npas))
59         print(info)
60         data = bytes(parse.urlencode(info), encoding='utf8')
61         print(data)
62     with request.urlopen(self.url, data=data) as rsp:
63         msg = rsp.read().decode('utf8')
64         if msg.find('成功')!=-1:
65             with open(s1 + '.txt', 'w') as f:
66                 f.write(str(N-1))
67                 f.close()
68         tkinter.messagebox.showinfo('提示', "消息:\n%s" % msg)
69
70     def Reg(self):
71         s1 = self.ent1.get()
72         s2 = self.ent2.get()
73         s3 = self.ent3.get()
74         N=100
75         info = []
76         info.append(('action', 'register'))
77         info.append(('user', s1))
78         md5npas = self.md5n(s2, N)
79         self.ent3.delete(0, len(s3))
80         self.ent3.insert(0, md5npas)
81         info.append(('pass', md5npas))
82         info.append(('count', N))
83         print(info)
84         data = bytes(parse.urlencode(info), encoding='utf8')
85         print(data)
86         with request.urlopen(self.url, data=data) as rsp:
87             msg = rsp.read().decode('utf8')
88             print(rsp)
89             print(msg)
90             if msg.find('成功')!=-1:
91                 with open(s1 + '.txt', 'w') as f:
92                     f.write(str(N))
93                     f.close()
94
95         tkinter.messagebox.showinfo('提示', "消息:\n%s" % msg)
96         s1=self.ent1.get()

```

NORMAL BR: master Client.py |<f-8 python 66% LN 68:66



```
97         s2=self.ent2.get()
98
99     root=Tk()
100     root.title("登录")
101     app=Login(root)
102     root.mainloop()
```

NORMAL BR: master Client.py |< LN 100:18

## 实验结论及分析

### 4、实验分析和总结。

当客户端使用计算一次性口令时，客户端的密码可以是任意长度，但长度若不超过 8 个字符，就难以保证安全性。

服务器上存有客户信息的数据库，用于存储上次成功登录收到的一次性口令和  $n$  的值，当进行认证时，会把接收到的口令信息( $\text{hash}(\text{seed}^{\wedge}\text{pass})$ )进行  $(N-n-1)$  次进行哈希，然后和上次存储的口令哈希( $\text{hash}(\text{seed}^{\wedge}\text{pass})$ )进行  $(N-n)$  次进行比较，如果匹配，则认证成功。认证成功后，将  $n$  的值减一，然后将新哈希存入库，等待下次认证，若  $n$  为 1，则需要重新初始化  $\text{seed}$  和  $n$  的值

客户端注册时获取  $\text{seed}$  和  $n$  值，对  $\text{seed}$  和  $\text{pass}$  进行异或操作，然后进行  $N$  次哈希操作，然后将用户名和哈希后的口令发往服务器，若注册成功，则将  $\text{seed}$  和  $n$  存入本地。下次登录时，从本地读取  $\text{seed}$  和  $n$ ，然后将密码和  $\text{seed}$  进行异或操作，然后进行  $n-1$  次哈希操作，将结果发向  $n$ ，若当  $n$  为 1，则需要重新初始化  $n$  和  $\text{seed}$  的值。

## 实验报告二

同组人:

指导教师:

日期:

实验题目	信息系统安全风险评估
实验目的	1、掌握信息安全风险评估软件 RiskAssess 的安装方法和主要功能； 2、利用信息安全风险评估软件 RiskAssess 完成一个信息安全风险评估实例。
仪器设备	风险评估软件 RiskAssess
<p>实验原理</p> <ol style="list-style-type: none"><li>1. 学习信息安全风险评估基本流程，信息安全风险评估的每个过程阶段工作</li><li>2. 学习矩阵法和相乘法两种常用安全风险计算方法</li></ol>	

## 实验内容及步骤

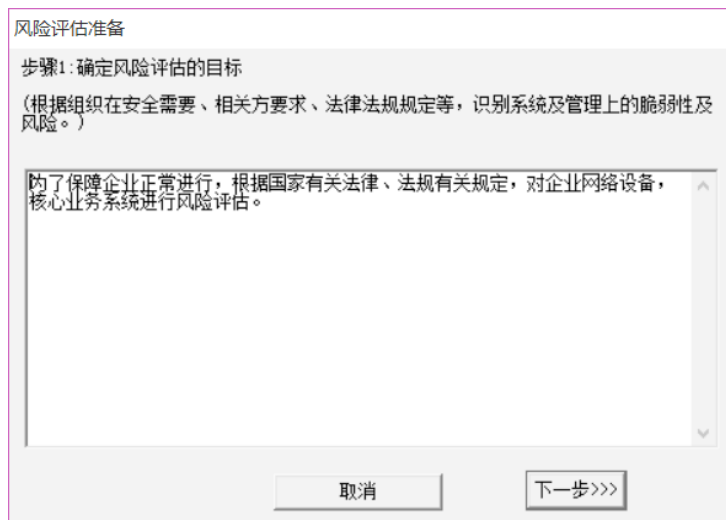
### (1) 安装风险评估软件

### (2) 使用 RiskAssess 创建工程

#### (1)新建评估工程



导航栏选择 风险评估准备->风险评估向导



风险评估准备

步骤2: 确定风险评估的范围  
(组织信息及与相关的各类资产、管理机构, 或某个信息系统、关键业务流程或部门等。)

数字兰曦系统

<<<上一步      取消      下一步>>>

风险评估准备

步骤3: 组建适当的评估管理与实施团队  
(成立由管理层、相关业务骨干、IT技术人员等组成的风险评估小组。)

殷悦

<<<上一步      取消      下一步>>>

风险评估准备

步骤4: 进行系统调研  
(为风险评估依据和方法的选择、评估内容的实施奠定基础。)

1. 调查企业服务器机房设备  
2. 调查网络基本结构  
3. 调查系统基本情况  
4. 调查系统应用架构  
5. 调查工作流程  
6. 调查人员状况

<<<上一步      取消      下一步>>>

## 风险评估准备

## 步骤5: 确定评估依据和方法

(根据调研结果, 确定评估依据和评估方法。)

信息安全风险评估规范

&lt;&lt;&lt;上一步

取消

下一步&gt;&gt;&gt;

## (3) 填写资产记录、脆弱性记录、威胁识别

修改资产记录

资产编号: 1 资产名称: 企业领导小组成员

机密性等级: 很高

完整性等级: 很高

可用性等级: 很高

选择加权方法

☐ 最高等级

☒ 权重方法

机密性: 0.4 完整性: 0.3 可用性: 0.3

资产价值: 5 是否重要资产: ☒

资产描述:

修改记录 取消

资产表

编号	资产名	机密性等级	完整性等级	可用性等级	权重方法	资产价值	重要资产	资产描述
1	企业领导小...	很高	很高	很高	权重法	5	是	
2	核心路由器...	高	很高	很高	权重法	4.6	是	
3	服务器群	高	很高	高	权重法	4.3	是	



添加脆弱性记录

脆弱性编号: 1 脆弱性名称: 硬件设备故障

脆弱性等级: 中 如果被威胁利用, 将对资产造成一般损害。

重要资产关联

资产编号	资产名称
<input type="checkbox"/> 1	企业领导小组成员
<input checked="" type="checkbox"/> 2	核心路由器交换机
<input checked="" type="checkbox"/> 3	服务器群

脆弱性描述

添加记录 取消

脆弱性表

编号	脆弱性	脆弱性等级	资产关联	脆弱性描述
1	硬件设备故障	中	{2, 3}	
2	人员失误操作	高	{1, 2, 3}	
3	木马病毒攻击	很高	{3}	
4	未授权非法访问	很高	{2, 3}	
5	网络洪水攻击	很高	{3}	

添加威胁记录

威胁编号: 1 威胁名称: 意外断电

威胁等级: 中 出现的频率中等 (或 > 1 次/半年); 或在某种情况下可能会发生; 或被证实曾经发生过。

威胁描述

添加记录 取消

威胁表

添加

修改

编号	威胁	威胁等级	威胁描述
1	意外断电	中	
2	系统宕机	高	
3	系统过负载	很高	

## (4)选择风险评估方法，此处选择矩阵法

第一步:威胁脆弱性关联

第二部:计算安全事件可能发生的可能性

矩阵法——安全事件可能性表

编号	威胁编号	威胁	威胁等级	脆弱性编号	脆弱性	脆弱性等级	安全事件可能性	可能性等级
1	1	意外断电	中	1	硬件设备故障	中	12	3
2	1	意外断电	中	2	人员失误操作	高	16	3
3	1	意外断电	中	4	未经授权非法访问	很高	20	4
4	2	系统宕机	高	2	人员失误操作	高	18	4
5	2	系统宕机	高	3	木马病毒攻击	很高	22	5
6	2	系统宕机	高	5	网络洪水攻击	很高	22	5
7	3	系统过负载	很高	1	硬件设备故障	中	17	4
8	3	系统过负载	很高	3	木马病毒攻击	很高	25	5
9	3	系统过负载	很高	5	网络洪水攻击	很高	25	5

## 第三步:计算安全事件的损失

矩阵法——安全事件损失表

编号	资产编号	资产	资产价值	脆弱性编号	脆弱性	脆弱性等级	安全事件损失	损失等级
1	2	核心路由器交换机	高	1	硬件设备故障	中	14	3
2	3	服务器群	很高	2	硬件设备故障	高	14	3
3	1	企业领导小组成员	很高	2	人员失误操作	高	21	4
4	2	核心路由器交换机	高	2	人员失误操作	高	19	4
5	3	服务器群	高	2	人员失误操作	高	19	4
6	3	服务器群	高	3	人员失误操作	高	19	4
7	2	核心路由器交换机	高	3	木马病毒攻击	很高	22	5
8	3	服务器群	高	4	未经授权非法访问	很高	22	5
9	3	服务器群	高	4	未经授权非法访问	很高	22	5
				5	网络洪水攻击	很高	22	5

## 第四步:计算风险值

矩阵法——资产风险表

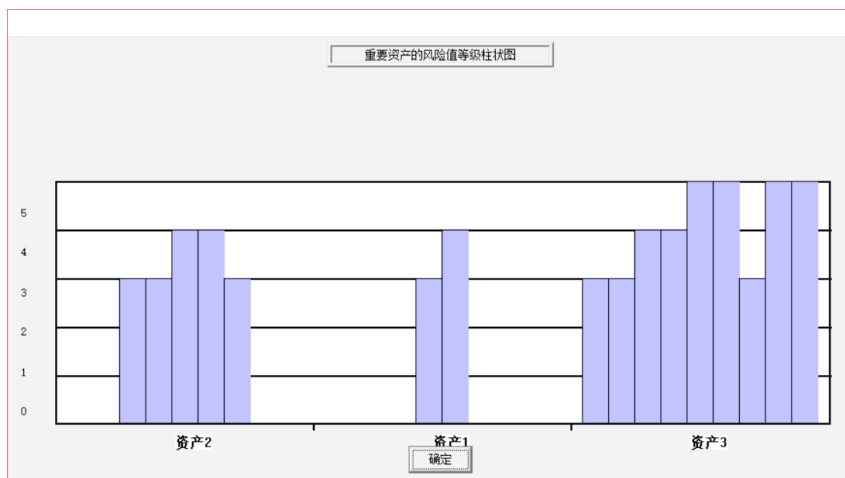
编号	资产	资产价值	威胁	威胁等级	脆弱性	脆弱性等级	安全事件可能性	安全事件损失	风险值
1	核心路由器...	5	意外断电	3	硬件设备故障	3	3	3	13
2	服务器群	4	意外断电	3	硬件设备故障	3	3	3	13
3	企业领导小...	5	意外断电	3	人员失误操作	4	3	4	16
4	核心路由器...	5	意外断电	3	人员失误操作	4	3	4	16
5	服务器群	4	意外断电	3	人员失误操作	4	3	4	16
6	核心路由器...	5	意外断电	3	未经授权非法访问	5	4	5	23
7	服务器群	4	意外断电	3	未经授权非法访问	5	4	5	23
8	企业领导小...	5	系统宕机	4	人员失误操作	4	4	4	20
9	核心路由器...	5	系统宕机	4	人员失误操作	4	4	4	20
10	服务器群	4	系统宕机	4	人员失误操作	4	4	4	20
11	服务器群	4	系统宕机	4	木马病毒攻击	5	5	5	25
12	服务器群	4	系统宕机	4	网络洪水攻击	5	5	5	25
13	核心路由器...	5	系统过负载	5	硬件设备故障	3	4	3	17
14	服务器群	4	系统过负载	5	硬件设备故障	3	4	3	17
15	服务器群	4	系统过负载	5	木马病毒攻击	5	5	5	25
16	服务器群	4	系统过负载	5	网络洪水攻击	5	5	5	25

## 第五步:确定风险等级

矩阵法——确定风险等级

编号	资产	资产价值	威胁	威胁等级	脆弱性	脆弱性等级	安全事件可能性	安全事件损失	风险值	风险等级
1	核心路由器	5	意外断电	3	硬件设备故障	3	3	3	13	3
2	服务器群	4	意外断电	3	硬件设备故障	3	3	3	13	3
3	企业领导小...	5	意外断电	3	人员失误操作	4	3	4	16	3
4	核心路由器	5	意外断电	3	人员失误操作	4	3	4	16	3
5	服务器群	4	意外断电	3	人员失误操作	4	3	4	16	3
6	核心路由器	5	意外断电	3	未授权非法访问	5	4	5	23	4
7	服务器群	4	意外断电	3	未授权非法访问	5	4	5	23	4
8	企业领导小...	5	系统宕机	4	人员失误操作	4	4	4	20	4
9	核心路由器	5	系统宕机	4	人员失误操作	4	4	4	20	4
10	服务器群	4	系统宕机	4	人员失误操作	4	4	4	20	4
11	服务器群	4	系统宕机	4	木马病毒攻击	5	5	5	25	5
12	服务器群	4	系统宕机	4	网络洪水攻击	5	5	5	25	5
13	核心路由器	5	系统过载	5	硬件设备故障	3	4	3	17	3
14	服务器群	4	系统过载	5	硬件设备故障	3	4	3	17	3
15	服务器群	4	系统过载	5	木马病毒攻击	5	5	5	25	5
16	服务器群	4	系统过载	5	网络洪水攻击	5	5	5	25	5

## 第六步:图形分析



## 第七步:生成风险分析报告

## (5)添加安全措施

添加安全措施

×

安全措施编号

2

安全措施名称

防火墙

关联的脆弱性

脆弱性序号	脆弱性名称
<input type="checkbox"/> 1	硬件设备故障
<input checked="" type="checkbox"/> 2	人员失误操作
<input type="checkbox"/> 3	木马病毒攻击
<input type="checkbox"/> 4	未授权非法访问
<input checked="" type="checkbox"/> 5	网络洪水攻击

描述

添加记录

取消

已有安全措施表		添加	脆弱性关联	删除
编号	安全措施	脆弱性关联	状态	安全措施描述
1	设备冗余	{1, 5}		
2	防火墙	{2, 5}		
3	杀毒软件	{2, 3}		
4	授权访问规则	{2, 4}		

(8)生成文档

	殷悦威胁列表.doc	Today at 10:30 AM	12 KB
	殷悦已有安全措施确认表.doc	Today at 10:25 AM	10 KB
	殷悦脆弱性列表.doc	Today at 10:31 AM	14 KB
	殷悦资产识别清单.doc	Today at 10:29 AM	14 KB
	殷悦重要资产清单.doc	Today at 10:30 AM	13 KB
	殷悦风险处理计划.doc	Today at 10:04 AM	10 KB
	殷悦风险评估报告.doc	Today at 10:03 AM	23 KB
	殷悦风险评估方案.doc	Today at 9:21 AM	12 KB

## 实验结论及分析

### 1、风险评估软件 RiskAssess 的安装

### 2、学习操作 RiskAssess 的主要功能

#### (1) 新建评估工程

#### (2) 风险评估准备

#### (3) 识别评估要素:资产记录、脆弱性记录、威胁识别

#### (4) 学习矩阵法、相乘法

#### (5) 完成安全措施确认, 生成文档

### 3、RiskAssess 的实际应用

## 殷悦风险评估报告（矩阵法）

对整个风险评估过程和结果进行总结, 详细说明被评估对象、风险评估方法、资产、威胁、脆弱性的识别结果、风险分析、风险统计和结论等内容。

资产	威胁	脆弱性	风险	风险等级
核心路由器交换机	意外断电	硬件设备故障	13	3
服务器群	意外断电	硬件设备故障	13	3
企业领导小组成员	意外断电	人员失误操作	16	3
核心路由器交换机 核心系统宕机 人员失误操作 20 由 4 器交换机	服务器群意系统 宕机外人员失误 操作断 20 电 4	服务器群人系统 宕机员木马病毒 攻击失 25 误 5 操作	服务器群 1 系统 宕机 6 网络洪水 攻击	2535
核心路由器交换机 机服系统过负载 务硬件设备故障 器 17 群 3	服务器群意系统 过负载外硬件设 备故障断 17 电 3	服务器群人系统 过负载员木马病 毒攻击失 25 误 5 操作	服务器群 1 系统 过负载 6 网络洪 水攻击	2535
核心路由器交换机	意外断电	未授权非法访问	23	4
服务器群	意外断电	未授权非法访问	23	4
企业领导小组成员	系统宕机	人员失误操作	20	4