
北交大 15 级学期实训方案

C++项目-动态口令认证系统



目 录

一、实训目标	- 1 -
二、项目说明	- 1 -
2.1 项目介绍	- 1 -
2.2 开发环境	- 2 -
2.3 模块设计	- 2 -
2.3.1 服务器端模块设计	- 2 -
2.3.2 软令牌模块设计	- 2 -
三、任务计划	- 3 -
四、开发团队	- 4 -
五、过程管理	- 5 -
六、考核标准	- 5 -
七、教辅资料	- 6 -

一、实训目标

本次实训结合软件工程专业前两年的主要课程：项目管理，数据库系统，面向对象编程与设计，内容研发以提高学生使用 C++ 开发软件项目的实际动手能力为目标，涉及 C++ 开发的多个知识点，通过完成本项目使学生在知识、能力和素质等方面应达到的如下目标：

1. 提高软件系统的设计能力，如需求分析、界面设计、数据库设计和功能设计。了解相应的文档标准并完成文档的编写。
2. 熟悉软件开发、测试、构建环境的工具，熟练掌握 Visual Studio、g++ 等 C++ 开发环境的使用。
3. 具有独立上机调试程序的能力，从而具有一定的解决实际工程问题的分析、设计和实现能力。
4. 形成良好的编码习惯，培养团队开发和协同工作的意识，提高沟通能力和自我表达能力。
5. 具备以下 C++ 开发能力：
 - 能够编写 Windows 图形界面应用程序，以及 Linux 守护进程；
 - 能够开发多线程应用系统，并利用互斥锁和信号量解决线程间共享资源的竞争问题；
 - 能够编写 socket 通讯程序，实现客户端与服务器之间稳定可靠的 socket 通讯；
 - 对密码算法有初步了解，能够根据算法文档实现密码算法，并能利用密码算法实现对系统关键信息进行保护；
 - 能够实现对 MySQL 数据库的访问。
6. 对动态口令技术有初步了解，并理解动态口令的工作原理。

二、项目说明

2.1 项目介绍

动态口令（One Time Password）是一种安全性较高的身份认证技术，它是根据专门的

算法产生的随机口令序列，每个口令只能使用一次并且不可预测，目前已经广泛应用于网上银行、手机银行、网络游戏、电子商务、移动办公等领域。动态口令的客户端形式多种多样，包括刮刮卡、智能卡、令牌、短信、软件令牌（手机、PC）等，根据工作机制的不同，动态口令可以分为时间机制动态口令、事件机制动态口令、挑战/应答动态口令。

完整的动态口令认证系统功能复杂，涉及到加密、密钥保存，网络协议等多项技术，客户端实现 PC 端的时间机制软令牌，服务器端实现动态口令的认证、同步、增加令牌、注销令牌四个接口，以及相关的基本功能。

2.2 开发环境

- Windows、Visual Studio
- Linux、g++、Oracle

2.3 模块设计

2.3.1 服务器端模块设计

服务器端	初始化模块	
	通讯模块	
	报文解析模块	
	交易处理模块	动态口令认证模块
		动态口令同步模块
		增加令牌模块
		注销令牌模块
	数据库操作模块	
	日志模块	

2.3.2 软令牌模块设计

软令牌	配置文件读写模块	
	联机初始化模块	
	用户登录模块	
	动态口令生成模块	
	用户删除模块	

三、任务计划

阶段	进程/天	学时	任务描述	知识点
立项	0.5	4	分组， 确定项目角色	组织机构与决策 项目计划管理
			项目启动会议,确定项目目标及计划	
需求分析	1	8	需求分析讨论	需求业务模型
			需求分析评审	需求文档,用例图, 需求规格说明
项目设计	2	16	概要设计	原型设计
			概要设计评审	评审过程管理
			详细设计	数据库设计
			详细设计评审	Powerdesinger 使用
			后端和前端整合	接口设计
开发阶段	12	96	开发环境搭建	MFC 编程 Socket 通讯 文件读写 加解密算法 动态口令生成算法
			软令牌-配置文件读写模块	
			软令牌-联机初始化模块	
			软令牌-用户登录模块	
			软令牌-动态口令生成模块	
			软令牌-注销令牌模块	
			开发环境搭建	g++编程 (Makefile 的编写) 守护进程的编写 文件读写 多线程的编写 Socket 通讯
			服务器端-初始化模块	
			服务器端-通讯模块	
			服务器端-报文解析模块	
			块	

			服务器端-动态口令认证模块	互斥锁和信息量的使用 MySQL 数据库的访问
			服务器端-动态口令同步模块	加解密算法 动态口令生成算法
			服务器端-增加令牌模块	
			服务器端-注销令牌模块	
			服务器端-数据库操作模块	
			服务器端-日志模块	
			接口测试工具	
			压力测试工具	
测试及完善阶段	2	16	功能测试	测试目标及测试原则 缺陷管理
			压力测试	
			根据测试结果进行功能完善	
上线阶段	1	8	生成安装程序	项目的发布与部署 Shell 脚本的编写
答辩	1.5	8	演示及答辩	
项目总结		4	项目总结会议	
总计	20	160		

四、 开发团队

学生分组完成本次实训，每组 4-5 人，可根据以下岗位设置进行分工：



五、过程管理

配置管理：Git

项目管理：QST-TES 平台

- 1、按照软件工程标准流程进行项目过程控制：对立项阶段、需求分析阶段、设计阶段、编码阶段、测试阶段、交付阶段和维护阶段等各个阶段进行标准开发流程的管理和监控。
- 2、通过 QST-TES 平台下发每项任务，学生按时完成任务并上传交付物，平台记录在线时间，上传时间，跟踪项目进度。
- 3、质量控制：平台支持小组交叉评审、教师初审、阶段评审，列出缺陷报告，评审报告
- 4、平台上学员每天提交日清。
- 5、老师可以设置甲乙双方角色，甲方提出需求变更，双方通过协商、讨论等方式最终达成一致意见，公开开会讨论的过程，由老师评审。
- 6、由小组组长和带班老师共同组成的评审小组，对结果进行评审，分为个人表现和项目验收成绩，过程成绩由平台提供。

六、考核标准

1、项目验收指标

- 代码规范
- 功能完整、无异常
- 界面设计合理，美观
- 技术掌握程度（加分项：应用最新技术，技术难度高）
- 是否有创新
- 演示效果（讲解清晰、演示流畅、重点突出）

- 文档完备性，规范性
- 可靠性（出错率低、容错性强、易于恢复）
- 可维护性（可适应不同场景进行灵活设置、易于识别错误和改正等）

2、个人表现主要考核指标如下：

- 团队配合能力，组长评分和组员互评
- 积极主动性，组长评分和组员互评
- 分工的工作量，难易度，答辩时自述，结合平台查看过程的执行情况
- 解决问题能力，答辩时通过提问判断
- 压力承受能力，临时任务完成情况
- 日志/周报是否及时提交，内容是否详尽，结合平台进行评审
- 考勤情况，平台自动评判
- 实训前后进步程度，组长评分和组员互评

根据项目得分并结合在开发过程中的贡献，由老师和组长对组内成员给出个人成绩。

总成绩=过程成绩+个人成绩+答辩成绩，建议权重 3:3:4。

七、 教辅资料

在 TES 教学平台中，对老师提供以下教学辅导资料：

- 项目说明
- 项目实施
- 项目文档
- 实施方案
- 考核标准