西 安 邮 电 大 学

毕 业 设 计（论 文）

题　　目：企业数据库加密技术的研究与实现

学　　院：通信与信息工程学院

专　　业：信息对抗技术

班　　级：对抗 1201 班

学生姓名：罗冲

导师姓名：闵祥参　职称：讲师

起止时间：2016 年 2 月 29 日至 2016 年 6 月 17 日

毕业设计（论文）承诺书

本人承诺：

本人所提交的毕业论文《企业数据库加密技术的研究与实现》是本人在指导教师指导下独立研究、写作的成果，论文中所引用他人的文献、数据、图件、资料均已明确标注；对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式注明并表示感谢。

本人深知本承诺书的法律责任，违规后果由本人承担。

学　　生（签字）：　　时间：　2016 年　6　月　2　日

指导教师（签字）：　　时间：　2016 年　6　月　3　日

西安邮电大学本科毕业设计(论文) 选题审批表

<table>
<tr><td>申　报　人</td><td>闵祥参(0604010)</td><td>职　　称</td><td>讲师</td><td>学　　院</td><td colspan="3">通信与信息工程学院</td></tr>
<tr><td>题 目 名 称</td><td colspan="7">企业数据库加密技术的研究与实现</td></tr>
<tr><td>题目来源</td><td>科研</td><td colspan="2"></td><td>教学</td><td></td><td>其它</td><td>✓</td></tr>
<tr><td>题目类型</td><td>硬件设计</td><td></td><td>软件设计</td><td>✓</td><td>论文</td><td></td><td>艺术作品</td><td></td></tr>
<tr><td>题目性质</td><td colspan="2">实际应用</td><td colspan="2"></td><td colspan="2">理论研究</td><td>✓</td></tr>
<tr><td>题目
简述</td><td colspan="7">（为什么申报该课题）
随着数据库系统的广泛应用，数据库系统的安全也显得越来越重要。密码技术成为保障数据库系统中信息安全的一个重要手段。数据库系统作为信息系统的核心部件，其安全性在一定程度上决定着整个系统的安全性。人们已经认识到这个问题的严重性，并开始着力于数据库安全性的提高。因此，深入分析现有加密算法的原理和设计安全的数据库加密应用系统便成为一个非常重要的问题。</td></tr>
<tr><td>对学
生知
识与
能力
要求</td><td colspan="7">1、具有良好的学习态度； 2、有一定的自学和创新思考的能力； 3、有一定的信息安全、数据库及加密技术等相关的知识； 4、有一定的编程能力。</td></tr>
<tr><td>预期
目标</td><td colspan="7">（本题目应完成的工作，题目预期目标和成果形式）
1、对数据库安全做详细介绍，分析加密技术在数据库安全中的作用。 2、介绍数据库加密技术的相关概念，并根据当今数据库应用的具体现状和我国具体实际情况，学习研究数据库加密的层次和关键技术，分析比较几种加密算法，选取合适的加密算法运用与数据库加密。 3、设计某企业数据库系统。 4、编程实现企业数据库加密系统。</td></tr>
<tr><td>时间
进度</td><td colspan="7">2016 年 2 月 29 日——3 月 12 日查找毕设论题相关资料，并进一步学习，结合课题申请书，总结并形成开题报告；2016 年 3 月 12 日——3 月 19 日，分析加密技术在数据库安全中的作用；2016 年 3 月 19 日——3 月 26、介绍数据库加密技术的相关概念，并根据当今数据库应用的具体现状和我国具体实际情况，学习研究数据库加密的层次和关键技术，分析比较几种加密算法，选取合适的加密算法运用与数据库加密。2016 年 3 月 27 日——4 月 14 日，设计某企业数据库系统。2016 年 4 月 15 日——5 月 14 日，编程实现企业数据库加密系统。2016 年 5 月 15 日——6 月 7 日，完成论文部分的书写工作。2016 年 6 月 8 日——6 月 17，准备 ppt，完成答辩。</td></tr>
<tr><td colspan="2">系（教研室）主任
签字</td><td colspan="2">2016 年 3 月 2 日</td><td>主管院长
签字</td><td colspan="3">2016 年 3　月 4 日</td></tr>
</table>

西安邮电大学本科毕业设计（论文）开题报告

学号	03126011	姓名	罗冲	导师	闵祥参
题目	企业数据库加密技术研究与实现				

选题目的（为什么选该课题）

随着互联网开放程度越来越高，网络信息安全问题已经日趋严峻，数据库作为信息的集结地，也逐渐吸引了黑客的目光，数据库被入侵事件也多有发生。随着数据库系统的广泛应用，数据库系统的安全也显得越来越重要。数据库系统作为信息系统的核心部件，其安全性在一定程度上决定着整个系统的安全性。而数据库加密技术作为数据库的最后一道保护措施，已经逐渐被企业数据库管理员重视起来，并开始着力于数据库安全性的提高。数据库加密技术通过加密数据库中数据，使得数据库中的重要数据在密文方式下工作，即使数据库中数据被窃取，也可以保证数据信息不会遭到泄露，确保了数据库数据的安全。因此，深入分析现有加密算法的原理和设计安全的数据库加密应用系统便成为了必要。

前期基础（已学课程、掌握的工具，资料积累、软硬件条件等）

在大学期间，我已经学过和间接了解到了与此选题相关的课程和资料，如《安全数据库》、《信息安全算法设计》、《网络安全技术》等，以及自己学习了 java 这门编程语言为此系统的实现打下了语言基础。在 java 学习过程中掌握了 Myeclipse 以及在 windows 环境下 mysql 数据库的使用，与此同时，我也在搜集大量数据库安全相关的书籍和资料如《目基于高级数据加密标准 AES 的数据库加密技术研究与实现_王劲东》，在学习中掌握的更多。

要解决的问题（做什么）

设计一套企业人事管理系统,选择加密算法实现对其数据库的加密.这个系统包含四个模块、分别为登录模块、管理员模块、员工管理、部门管理模块实现对其增删改查。数据库中要采用合适的加密算法实现对敏感数据的加密。当用户登录后，系统判断是否为授权用户，是则为之分配密钥，用来解密查询数据库中数据，要实现这个系统，必须研究现有加密算法以及加密的粒度，密钥的生成，以及密钥管理等任务。另一方面要实现这个系统必须掌握客户端与服务端传输数据的后台开发知识，开源数据库 Mysql 的使用，Java 后台的一些框架，以及采用 Java 编程语言如何实现与数据库的交互，从而完成这个系统加密的设计以及功能的实现。

工作思路和方案（怎么做）

问题思路分析：

为了实现对数据库的加密，得查阅相关资料了解到现有的加密算法，选择合适的算法作为本数据库的加密算法。在中国知网、图书馆、或者论坛里查找有关数据库加密粒度的书籍和资料，选择适合本数据库的加密粒度。

由于加密算法公开,应该对加密密钥更加安全的保管,可以将密钥单独设计为一张表,提高系统的安全性。为系统添加一位超级管理员，用来管理管理员模块，一般管理员不能直接操作管理员模块。同时这个超级管理员具有最高权限。

在 Java 与数据库连接模块使用 JDBC 技术，学习 Spring 和 Mybatis 作为后台逻辑处理框架.在用户登录成功后，授予用户密钥，使用 sql 语句从数据库查询判断字段是否加密，如果加密用密钥解密后显示，否则直接明文查询显示。

执行方案步骤：

首先进一步了解加密技术在数据库安全中的重要作用,了解数据库加密技术的相关概念，以及分析当前几种机密算法，选定一种合适的加密算法应用于本系统。

其次学习 Mysql 数据库的安装与使用，根据系统模块完成系统的数据库设计，深入学习 Java 连接数据库 JDBC 技术，完成与数据库的连接模块。

根据系统逻辑，设计管理员模块，包括管理员的增删改查。判断敏感数据调用加密模块对数据进行加密后利用 sql 命令存储。

编写加密引擎，实现密钥的生成以及密钥的保存，对传入数据的加密。

编写员工列表显示模块，包括增删改查，判断是否为敏感数据调用加密引擎实现加密后存储。

编写部门管理模块，判断是否为敏感数据调用加密引擎实现加密后存储。

最后整合系统，编写登录模块，实现权限管理，为合法用户授权，获取密钥解密数据实现对系统的访问。

指导教师意见:

签字：　　2016 年 3 月 25 日

西安邮电大学毕业设计（论文)成绩评定表

<table>
<tr><td colspan="2">学生姓名</td><td>罗冲</td><td>性别</td><td>男</td><td>学号</td><td>03126011</td><td>专业班级</td><td>信息对抗技术 1201</td></tr>
<tr><td colspan="2">课题名称</td><td colspan="7">企业数据库加密技术的研究与实现</td></tr>
<tr><td>指导教师意见</td><td colspan="8">评分（百分制）：_____ 指导教师(签字)：_____ _____年___月___日</td></tr>
<tr><td>评阅教师意见</td><td colspan="8">评分（百分制）：_____ 评阅教师(签字)：_____ _____年___月___日</td></tr>
<tr><td>验收小组意见</td><td colspan="8">评分（百分制）：_____验收教师(组长)(签字)：_____ _____年___月___日</td></tr>
<tr><td>答辩小组意</td><td colspan="8">评分（百分制）：_____ 答辩小组组长(签字)：_____ _____年___月___日</td></tr>
<tr><td colspan="2">评分比例</td><td colspan="7">指导教师评分 20(%) 评阅教师评分 30(%) 验收小组评分 30(%) 答辩小组评分 20(%)</td></tr>
<tr><td colspan="2">学生总评成绩</td><td>百分制成绩</td><td colspan="2"></td><td>等级制成绩</td><td colspan="3"></td></tr>
<tr><td>答辩委员会意见</td><td colspan="8">毕业论文(设计)最终成绩(等级)：_____
学院答辩委员会主任(签字)：_____ _____年___月___日</td></tr>
</table>

目录

摘要

作为信息系统数据集中存储与共享的平台，随着计算机网络的发展以及信息系统的普及，数据库安全越来越成为信息安全领域的一个严重问题。数据库的现有安全措施包括网络防火墙入侵检测、访问控制等，但是近年来数据库泄漏事件的原因却源于系统管理漏洞和企业内部人员因为某种原因将信息故意泄漏，数据库现有安全措施不能避免数据库管理者对数据的泄漏。

对数据库加密相关技术及加密算法进行深入分析，对比常用的 DES 加密算法和 AES 加密算法，基于安全考虑决定采用 AES 算法实现了一个企业数据库加密系统。系统功能主要分为管理员模块、员工模块以及部门模块并且对其管理做了一定的权限设定，各模块均实现敏感字段的加密存储。系统使用 Mysql 提供数据存储，使用 MyEclipse 开发工具，服务器使用 Tomcat，开发模式为 B/S 模式，由于这些知识现在已经很完善了，所以在技术上完全可行。

本系统从安全与效率角度出发，设计了二级密钥来维护系统的安全性，首先用户登录口令，采用 MD5 消息摘要加密算法加密,其次维护用户的工作密钥，即对应的工作密钥进行二次加密并维护，将二次密钥保存在本地文件中，这样的设计进一步加强了系统安全性，保证了系统的数据安全。

关键字：数据库加密；AES 算法；敏感字段；加密存储；MD5 消息摘要算法

ABSTRACT

As information systems data centralized storage and sharing platform, with the popularity of computer networks and information systems development, database security is increasingly becoming a serious problem in the field of information security. Existing security measures database includes a network firewall, intrusion detection, access control, etc, but in recent years, but the cause of the spill database from loopholes in management systems and internal staff for some reason will deliberately leaked information. Database existing security measures can't prevent the leakage of the data database managers.

Database encryption-related technology and encryption algorithms in-depth analysis, compared to the commonly used DES encryption algorithm and AES encryption algorithm, based on security considerations decided AES algorithm to encrypt a corporate database system. System administrator functions are divided into modules, staff module, as well as department and its management module to do a certain set of permissions, encryption modules are storing sensitive field. The system uses Mysql relational databases provide data storage, use Myeclispe development tools, server using Tomcat, development model of B / S mode, since this knowledge is now perfect, so technically feasible.

This system from the Angle of safety and efficiency, the two key design to maintain security of the system, the first user login password using MD5 message digest encryption algorithm, followed by the maintenance user work key that corresponds to the work of key secondary encryption and maintenance, the secondary key is stored in a local file, this design is further enhanced system security, ensure data security system.

Keywords: database encryption; AES algorithm; sensitive field; encrypted storage; MD5 message digest algorithm

引言

21 世纪是信息时代，互联网成为人们快速、大量获取信息的主要渠道，是人们日常生活必不可少的一部分。数据库技术是一个极其重要的领域，有了数据库技术后极大方便了人们的工作，学习和生活。然而，数据库作为信息的集结地，慢慢引起了不法分子的注意，不法入侵的事件也频繁发生。

在当下生活、工作、学习都依赖于在线应用进行的时代，企业数据库的安全性的尤显得额外重要，它的安全性代表着整个系统的安全性。随着攻击手段的层出不穷，现有的保护措施已经不能保证系统能够免遭破坏，保证其安全性。越来越多的企业已然了解到了此问题的严重性，致力于提升系统安全性的研究。

本文主要论述了企业人事管理系统中一些常见模块的实现以及对其的加密设计。该系统做了一定的权限设定，root 用户登陆后可以对普通权限用户进行管理。对登陆过程和一些用户信息的录入也做了一些校验，这样提高的管理员的工作效率，节省了用户的时间。本系统实现了所有模块的加密设计，界面友好，操作简单。

论文分为七个部分，第一部分主要讲解了该课题现状及意义，研究目标、内容和研究方法；第二部分主要讲解了数据库安全技术，着重描述了加密技术包括加密粒度的分析和加密算法的选择；第三部分是系统需求分析和框架设计，其中包含了系统设计的目标，用例图分析，对运行环境的总体规定，开发环境以及所用到的一些技术；第四部分是该系统的具体实现，包括如何实现数据库连接、加解密引擎等；第五部分系统测试。第六部分是整片论文的总结与展望，谢辞以及参考文献。

1 绪论

1.1 课题背景及意义

数据库技术开始发展于上世纪 60 年代，随着互联网时代的来临，该技术已广泛应用于各行各业，据相关报道，现代互联网应用中 80%以上都在使用该技术。现今各大企业都依赖数据库进行海量信息的存储，其中存储的数据也越来越重要，无论是在本地环境还是互联网环境中，Database 面临着严重的安全问题。信息存取的安全性、敏感信息的保密性和完整性逐渐吸引了众多从事安全方面者的目光。

1.2 国内外研究现状

数据库的安全性强调了保护 DBMS 和其保存的数据不被恶意攻击和不当的操作被窃取、扩散。国外大型 Database 厂商，如 Oracle、MYSQL、DB2 等关系型 DBMS 大都采用了对应的登录检查、权限管理、安全审计等保护措施，很大程度的防止了重要信息被非法盗用和修改。初期数据库很少有对应的信息加密技术。现在各大主流数据库厂商的产品大都提供相应的加密功能。

数据库加密函数，加密过程对用户是透明的，只需调用加密函数即可，密文在解密之后通过视图呈现给用户。IBM 公司是在其数据库 DB2 7.2 版本中首次引入加密技术，通过加解密函数，用户可以直接在 SQL 命令语句中执行加解密功能,该加密技术着重考虑安全，设置了安全字典，限制管理的权限。SQL Sever 数据库从其 2005 版本中引入加密功能，提供多层密钥和多种加密算法。由于厂商提供的对应加密功能比较有局限性，所以其应用并不普及。

现有数据库加密方法分为三种：

(1)软件加密

软件加密包括库内加密和库外加密。库内加密指在内部完成操作，操作的过程对使用者可见。在对 Database 进行存取之前实现加解密，加密的密钥一般保管在系统能够访问到的数据库表中。库外加密是在数据库管理系统内核层完成加密保存将加密系统设计为一个外层工具。操作表时，通过该外层工具实现表加密设置。

(2)硬件加密

与软件加密相对的就是硬件加密，是指在磁盘存储空间与 DBMS 之间添加一层用作加密处理。不过因为中间添加一层，可能导致系统变得更加复杂，而且数据读取时也非常难以控制，因此，硬件加密并没有被成功的推广。

(3)基于操作系统层次的保护方案

操作系统是所有软件运行的前提，所有数据的保存，最后都会体现在操作系统层次上的文件中。数据存储在磁盘上，文件系统可以直观、方便的管理这些数据，它是实现数据存储的核心，探索在文件系统中实现加密储存不失为一个非常明智的选择。

1.3 研究目标及内容

1.3.1 研究目标

（1）实现企业管理人事信息的数据库系统。

（2）实现各模块中信息的增删改查。

（3）完成各模块信息的加密处理。

（4）对密钥完成安全存储。

（5）实现系统的权限设定。

1.3.2 研究内容

企业数据库的现有安全措施包括网络防火墙、入侵检测、访问控制等安全措施不能避免数据库管数据泄露问题，而数据库加密可以有效的解决这些问题。本文对企业人事管理系统各个模块的功能研究，完成了登录检查、加解密引擎与密钥管理等模块的基本设计原理和实现过程，并使用 Java 中 JCA 和 JCE 密码包对系统的加密过程进行实现。

1.4 研究方法

在系统的设计实现中，主要的问题是对数据库的加密设计，在了解数据库相关知识的提前下通过查阅相关文献和书籍学习常见加密算法并掌握 Java 语言如何封装加密操作进而实现数据库加密。在老师的指导，和同学的帮助下完成了企业数据库加密系统的研究与实现。

2 数据库安全技术

2.1 数据库安全概述

数据库安全是指保护其中数据存储的安全性，以预防非法操作导致数据被窃取、修改、中断的手段。目前数据库的安全技术主要有：用户权限控制、数据库数据加密、审计日志和强制存取控制。

在以上数据库安全方法中，用户权限控制、审计日志、以及强制存取控制并不能从根本上避免攻击问题，只是降低了数据库被非法访问的风险。数据库加密处理则从根本上对数据进行密文转换，即使信息有可能被窃取，也不能直观的获得。因此，数据库加密是目前企事业单位主流的加密技术。

2.2 数据库加密的基本条件

2.2.1 加密粒度的选择

数据库的加密粒度一般被分为表级、记录级以及数据项三种。MySQL 这 3 种加密的特性可大致归纳如下：

（1）表级加密：耗费时间少，速度快，不会出现死锁，加密对象是整个表数据，锁定的粒度大。

（2）记录级加密：是指将表中每一行数据看成是一个整体，将数据库表中的每一行看作是目标对象，然后在一起做处理，这种方式加密会确保数据更加安全，灵活性更高，但其涉及多个密钥间的解密，会降低系统的使用效率。

（3）数据项加密：指直接对数据库表中的每个敏感数据单独执行加密操作，将会产生数量繁多的密钥，这样加密的方式数据的保密性极高，灵活性也是机器高的，不过其运行性能较差，关联到所有数据的处理操作，大大降低了系统的使用效率。

这三种方式的加密，数据项加密这种方法的效率较高，而且灵活性高，但是会产生数量庞大的密钥，操作性能低下。记录级加密虽然密钥数量较少，不过必然会对整条记录进行加解密，然后对整条记录中的某一个数据进行提取，这样的操作将严重对系统数据库的操作能力造成影响，会造成一些不必要数据项的解密操作。表级加密的方式只需少量的密钥，加解密速度快工作量少，同时也降低了密钥的维护难度。这样的方式将大大增强整个系统的操作速度。

虽然数据库的加密可以极大保护数据的完整性以及安全性，但是这样的操作本质上是将原本明文进行繁琐的加密操作，以实现数据变成无法直接阅读的密文。不过在

一般应用使用数据库时都要完成对数据库中存储数据的管理和使用，在有些条件查询中，必须识别特定数据项，完成某些功能操作。所以，在加密数据项时不能对所有数据项都进行加密，下面几种情况是不能作为加密数据项的：

（1）数据库的索引字段：为了实现对数据库的快速查询，通常需要建立索引，而数据库管理系统要求它们的建立一定要是明文的，不然将无法快速遍历索引字段值。

（2）在条件查询时关系运算符操作的比较字段：在系统应用中很多情况下需要按条件进行筛选，而按条件进行挑选时，DBMS 需要将数据库中的数据项进行比较，而如果是密文是无法完成这样的操作的。

（3）表之间的关联关系字段：在现实应用中，很多情况下都需要进行表之间的关联操作，然而如果对表之间的关联字段进行加密，就无法正常读取完成表之间的条件连接运算。

2.2.2 密钥管理

在现实生活中的即时网络通讯中，每次会话的密钥会动态生成，在结束通讯时进行销毁，密钥是可以随时清除的。然而有些必要数据是要进行永久保存的，如果密钥频繁更换，保存的信息就会失效，因为它们是以密文的形式保存的，没有密钥就还原不出密文，因此数据库中数据的密钥不能随时清除，需要进行长期的保存。在进行密钥保存的同时，也面临着一个新的问题，就是如何做到在数据使用期间密钥不会被泄露，所以密钥的管理应该更严格。在现实应用中一般一个需求对个多个数据表，而为了安全通常使用多个密钥进行加密，加密数据所使用的密钥也面临被窃取的风险，因此，密钥数量多且不容易安全的进行存储，需要组织设计安全的密钥体系。

2.2.3 时间上的开销

在加密数据库中信息时，有一个非常棘手的问题，就是加密后会影响系统的使用效率问题。在对数据库进行加密后，每次的检索以及录入都需要对数据先进行加密或者解密操作，降低了数据库的操作速度，增大了时间上的开销，进而影响了整个系统的响应速度。如果设计不合理还有可能让整个系统变得一塌糊涂，慢的让人难以接受。那么加密也就变得毫无意义可言。其次，进行加密后，产生的密文的长度会扩展，提高了存储空间的成本。

为了解决上述问题经过深入研究，决定采用以下方法，协调两者之间的冲突：

（1）由于表级加密的方式开销小、速度快、密钥管理方便，因此采用表级加密的方式；

（2）对比常用的加密算法，由于 DES 加密算法存在一定的安全漏洞，决定采用 AES 加密算法对数据项进行加密；

（3）数据表的密钥利用 JDK 中提供的密码包生成；

2.3 加密算法

DES 加密算法因为密钥的长度较小(56 位)，已经不适应当前互联网时代对信息进行加密的需求，而 AES 作为 DES 的替代者，集成了安全性高、性能强、方便使用等优点。

AES 有三个密钥长度：128，192，256 位，相对而言，AES 的 128 密钥比 DES 的 56 密钥强 1021 倍。AES 算法主要包括三个方面：轮变化、圈数和密钥扩展。下面的表格 2-1 展示了 AES 加密算法的几种模式以及 Java 对其的实现。

表格 2-1

密钥长度	默认	工作模式	填充方式	实现方式
128、192、256	128	ECB、CBC、PCBC、	NoPadding、PKCS5Padding、ISO10126Padding	JDK（256 位密钥需要获得无政策限制权限文件）
128、192、256	128	ECB、CBC、PCBC、CTR、CTS、CFB	PKCS7Padding、ZeroBytePadding	Bouncy Castle

AES 实现数据加密如图 2-1 所示：

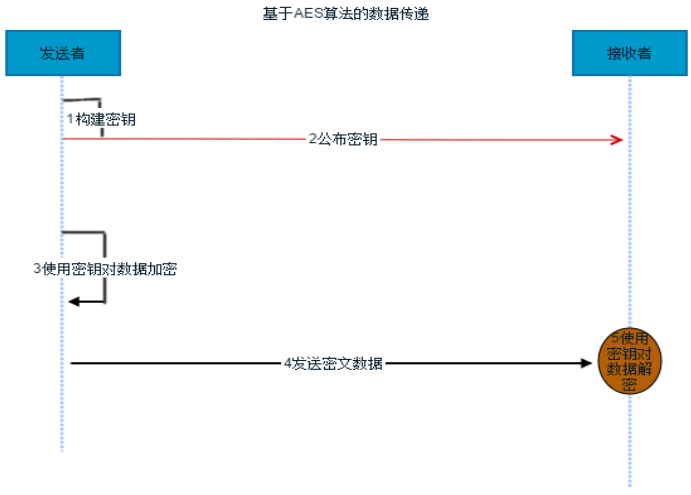

图 2-1 AES 加密流程

3 企业数据库加密系统设计

3.1 系统总体设计

该系统主要是为企业的人事管理提供了便捷的操作，下图 3-1 展示了该系统的使用流程：

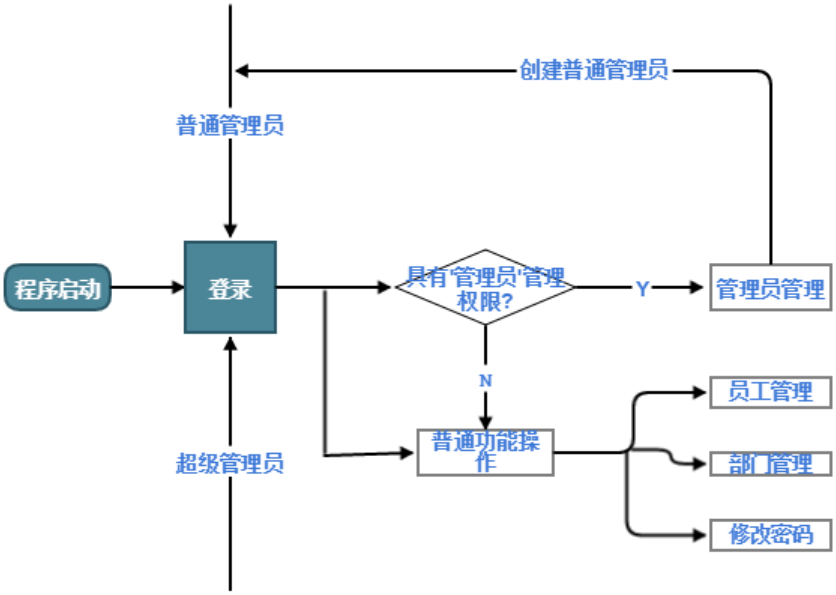

图 3-1 系统功能操作流程图

简单叙述下该系统的操作流程图：首先在用户登录系统时，检查用户登录是否为 root 用户，如果是则拥有管理“管理员”模块的权限，可以创建其它普通登录用户，否则就只能操作系统的员工管理模块，部门管理模块，以及修改当前用户密码操作。图 3-1 直观的显示了超级用户的权限与普通用户的区别就在于拥有“管理员”模块的功能。

3.2 系统功能设计

本系统主要涉及到页面显示以及后端逻辑处理两个部分：

（1）管理员登录：在用户输入用户名，密码以及验证码后，后台采用 MD5 消息摘要算法对密码进行完整性检测，判断正确后，提取当前用户的权限，执行不同的权限操作。

（2）管理员管理：在第一步完成后，如果当前的用户具有管理员管理权限，则可以操作这个模块，完成对普通管理员的增加，删除以及修改，相应的在执行对数据库操作之前，实现数据的加密。

（3）员工管理：在用户登录后，可以使用这个模块进行员工信息的查看，以及增加员工，修改员工信息，删除员工等操作，在对数据库执行操作前对数据进行加解密操作。

（4）部门管理：对所有部门的信息进行查看以及修改，显示时对数据库中的数据执行解密操作，修改时首先进行加密在进行数据库操作。

（5）修改密码：登陆后，在密码出现任何问题时，可以使用这个模块对登录口令进行修改，并加密更新数据库中的信息。

3.3 加解密模块设计

此模块应当处于应用程序与后台服务器之间，在获取信息后，利用该模块对数据进行处理，在将其与数据库发生交互。因为此系统要频繁的与数据库进行数据交互，所以考虑到使用的方便性以及减小工作量，同时为了降低这部分代码的耦合度，将加解密模块设计封装为一个单独的模块，在每次需要使用时进行注入。在与数据库发生数据交换时的调用关系如下图 3-2 所示：

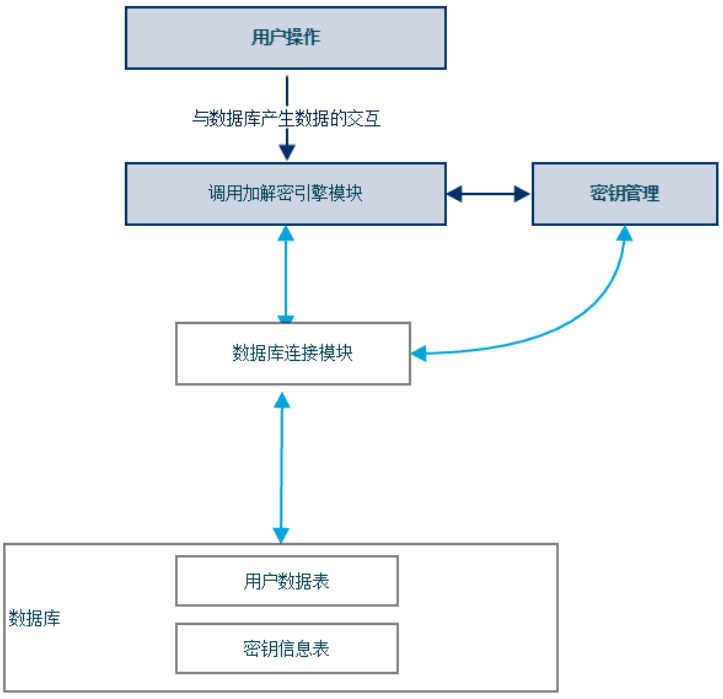

图 3-2 加解密引擎关系图

此模块主要的工作分为在查询数据库中信息时被调用，对密文信息进行解密，在需要对数据库执行插入和修改操作时，使用加密模块对信息加密后进行存储。一个数据库加密系统的安全与效率主要在于算法的选择。非对称加密算法由于不同用户需要不同的公钥和私钥，这样会产生数量巨大的密钥，在密钥管理上造成了很大的瓶颈，因此这里我们选用对称加密算法，而且对称加密算法的运算速度要远远高于非对称加密算法，由于 DES 对称加密算法层多次暴露出安全问题，因此本文采用其替代算法 AES 对称加密算法。

在需要查询数据时，调用解密引擎模块，这个过程仅仅是加密过程的相反运算，和加密模块的功能相反。加解密模块的处理方式如图 3-3 所示：

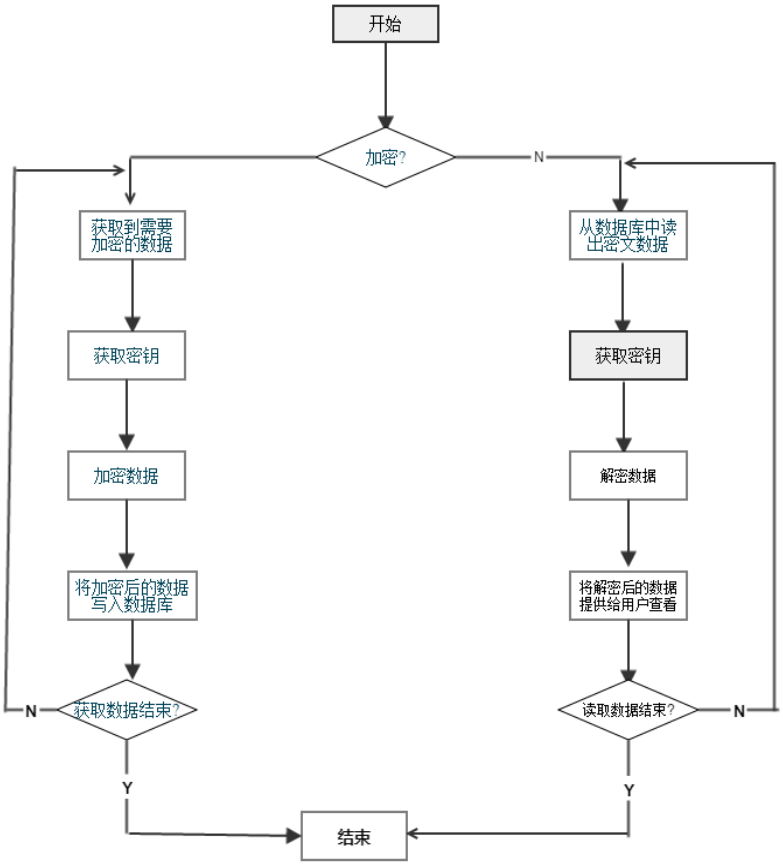

图 3-3 加解密引擎模块工作流程

3.4 密钥存储设计

在整个企业数据库加密系统中，除了数据库使用的基本安全措施外，最核心的点就在于采用的对称加密算法的密钥的存储安全。因为如果密钥信息泄露，加密后的数据将变得透明，因为加密算法是向外公开的，所以整个加密系统的关键就在于密钥的安全保护。如果所有的数据都采用同一个密钥，那么整个系统的健壮性将很低，如果

这个密钥一旦被窃取，整个系统的安全性就不复存在。然而基于数据项级的加密，也就是每一个需要加密的数据都采用一个不同的密钥，这样将会产生巨大数量的密钥。这么多的密钥在管理上会存在很大的问题，大复杂的密钥管理会使查询密钥花费大量的时间，严重的降低了系统的使用效率。所以经过衡量后，本文采用了一种类型数据一个密钥的加密方式，也就是基于记录的加密粒度，这样可以减少密钥的数量，提高系统在时间上的开销。但是还存在另外一个问题，就是直接将加密密钥保存在数据库系统中，如果数据库遭到入侵，那么密钥的安全将无法保证，所以本文还设计了二层密钥的管理方法。二层加密的思想如图 3-4 所示：

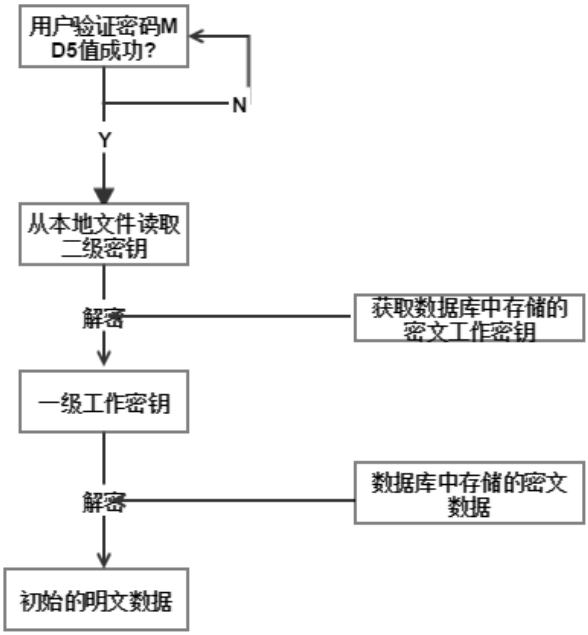

图 3-4 密钥管理流程

一级密钥为工作密钥，即为系统核心密钥，采用 AES 加密算法来加密数据，为了保护其安全所以设计了二级密钥，同样采用 AES 加密算法，对一级工作密钥进行加密，将产生的密钥密文保存在数据库中，而将二级密钥保存在本地文件中，在对其进行加密。这样即使数据库遭到非法入侵，入侵者也无法获取系统核心密钥，因为拿到的密文，即不能分辨其加密的密钥，也无法得知其采用了哪种加密算法进行了二次加密，二级密钥和一级工作密钥保存在不同的系统上，大大降低了系统的安全风险。

为了方便频繁的密钥查询，将一级工作密钥保存在数据库的密钥表中，包括表名，字段名，以及对应的 KEY，密钥表设计如表 3-1：

表 3-1 一级工作密钥表

字段名称	类型	备注	字段描述
TNAME	VARCHAR(50)	NOT NULL	表名
COLNAME	VARCHAR(50)	NOT NULL	字段名
SKEY	VARCHAR(200)	NOT NULL	密钥

本系统中所有密钥的产生都是按照 AES 对称加密算法的，密钥采用的是默认 128 位的，因为 Java 中 JDK 提供了 KeyGenerate 类用于生成对称加密算法需要的密钥。通过两层密钥加密的方式，使系统的安全性提升了一个新的层次。

密钥的存储设计，关系到密钥管理的安全性以及合理性，密钥管理模块将会按照密钥存储设计获取到一级工作密钥提供给加解密引擎模块使用。

3.5 加密程序设计

加密程序设计主要是将需要加密的数据信息，调用密钥管理程序获取到密钥，然后调用加解密引擎完成数据的处理，进行保存。

用户在使用本系统时，按照相应的操作会获取到对应的 SQL 语句，获取到对应的密钥，调用加解密程序实现加解密操作，最终实现用户所进行的操作。

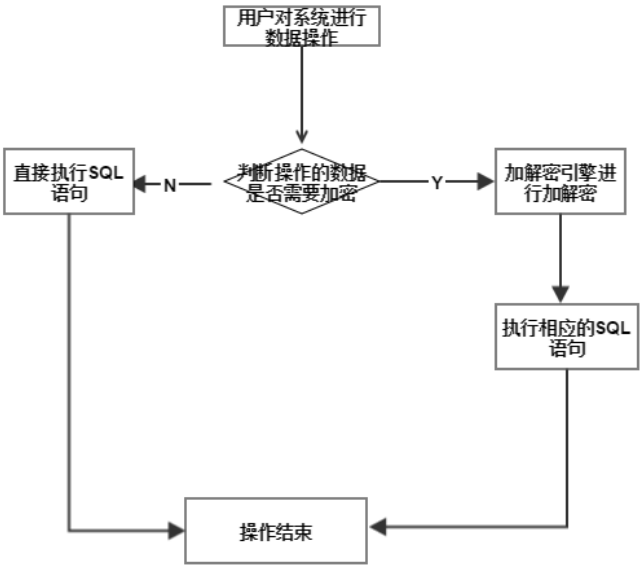

图 3-5 加解密数据库操作流程图

加解密程序设计需要根据不同的操作执行不同的 SQL，操作不同数据，最终完成数据的加解密操作。

系统在判断用户的操作后，只需调用加解密引擎模块完成数据的加解密操作即可，而对应的密钥获取模块则封装在加解密引擎中，这样更有利于系统的解耦。

在访问数据库时，将执行下图 3-6 所示流程，获取到密文后，执行解密操作后，返回结果，至此，完成一次数据库查询操作。

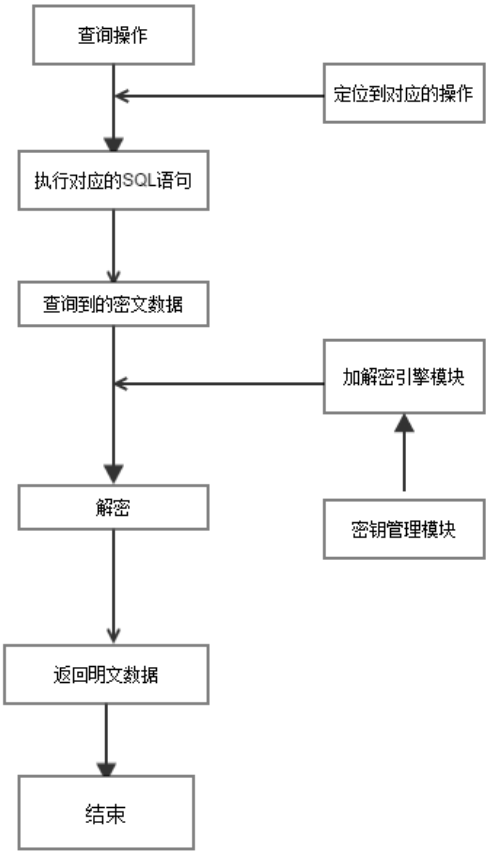

图 3-6 查询数据库流程图

4 企业数据库加密系统实现

4.1 系统总体需求分析

系统总体需求分析，是极其重要的，因为它是完成一个系统开发的初始点。需求分析根据系统的实际要求，进行分析设计，完成系统使用的要求和实用性。

根据分析结果，企业人事管理系统的加密实现，系统主要分为了三个模块，管理员模块，员工管理模块，以及部门管理模块，在加上系统的基础功能，包括用户的登录验证以及修改当前用户密码的操作，在每个模块的实现中加入的加解密引擎模块，这一系列的操作构成整个系统的完整性，健壮性和安全性。数据库中总共包含管理员表（admininfo），员工信息表（emp），部门信息表（dept）以及工作密钥的密文存储表（KDC）。

4.2 系统功能实现

4.2.1 数据库设计实现

(1)管理员信息表 E-R 图（图 4-1）

管理员信息表包括管理员的 ID，管理员用于登录的用户名，登录口令的 MD5 散列值，登录口令的 AES 算法加密值，管理员姓名，身份证号码，性别，联系电话，电子邮箱，入职时间。

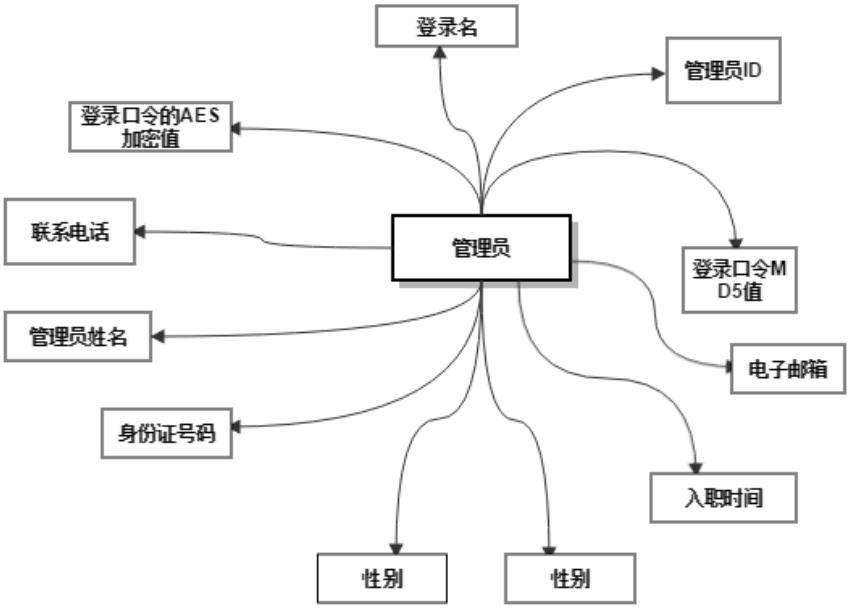

图 4-1 管理员信息表 E-R 图

（2）员工信息表 E-R 图（图 4-2）

员工信息表中包含员工 ID，员工姓名，年龄，联系电话，电子邮件，身份证号码，入职时间，性别，薪资，员工所属部门编号。

图 4-2 员工信息表 E-R 图

（3）部门信息表 E-R 图（图 4-3）

部门信息表中包含了部门编号，部门名称，部门所在地，部门经理。

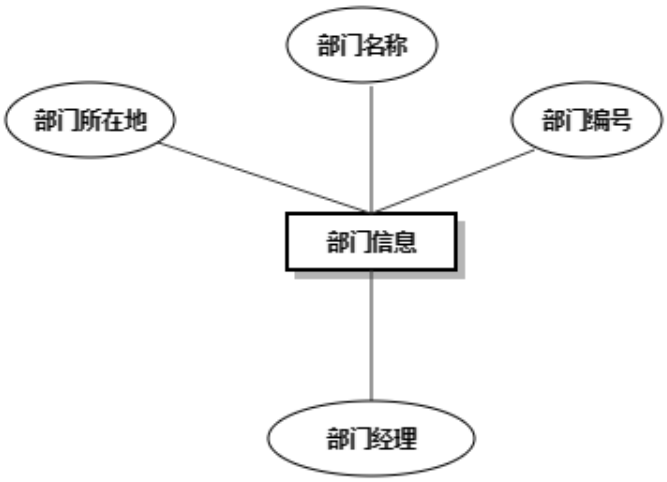

图 4-3 部门信息表 E-R 图

（4）工作密钥存储表 E-R 图（图 4-4）

工作密钥存储表中包含了密钥 ID，表名称，字段名称，以及工作密钥的密文。

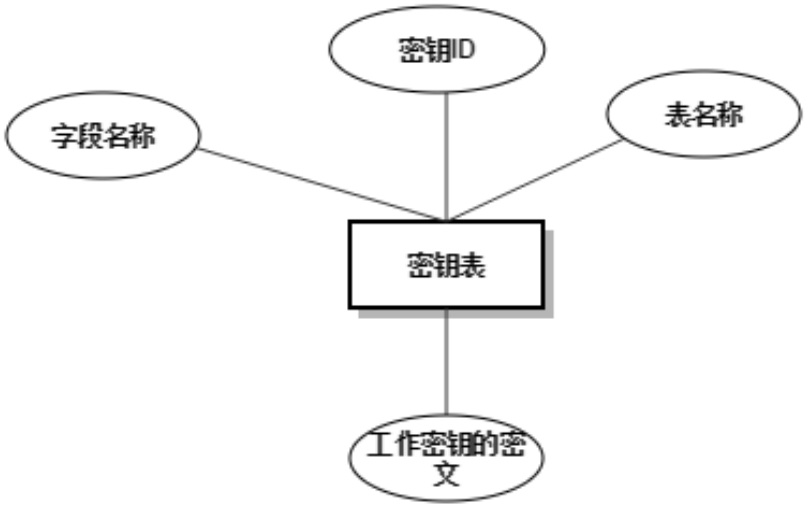

图 4-4 工作密钥信息表 E-R 图

4.2.2 实现各模块的基本增删改查

由于每个模块的基本操作原理都是一样的，这里选择管理员管理模块进行演示说明。在登录用户具有超级管理员权限时，可以操作这个模块，可以对所有的管理员进行修改，删除，以及查询和添加操作。在这里附上管理员模块操作时的具体的实现代码。

（1）超级管理员进行增加管理员操作时执行的核心代码如下：

```
/*将用户填写数据加密*/
adminInfo.setPassword_md5(tool.encryptMD5(adminInfo.getPassword()));
adminInfo.setPassword(tool.encryptAES("admininfo", adminInfo.getPassword()));
adminInfo.setIdcard_no(tool.encryptAES("admininfo", adminInfo.getIdcard_no()));
adminInfo.setPhone(tool.encryptAES("admininfo", adminInfo.getPhone()));
adminInfo.setEmail(tool.encryptAES("admininfo", adminInfo.getEmail()));
//调用数据库访问层保存管理员信息
dao.saveAdminInfo(adminInfo);
```

系统在提交添加的信息后，前端请求根据注解定位到对应控制器，首先调用加解密引擎完成数据的加密操作，然后调用 DAO 完成数据库的插入动作。

（2）删除管理员：

在使用删除操作时，不需要加解密操作，只需将被删除用户的id传给后台控制器，即可完成删除操作，具体代码如下：

```
@RequestMapping(value="/{id}",method=RequestMethod.DELETE)
    public boolean delete(@PathVariable("id") Integer id){
    if(id!=null){
        dao.deleteAdminInfo(id);
    }
    return true;
    }
```

（3）修改管理员信息：

```
/**修改密码后需要将数据库中密文也随之更新计算相应密文存入数据库*/
adminInfo.setIdcard_no(tool.encryptAES("admininfo", adminInfo.getIdcard_no()));
adminInfo.setPhone(tool.encryptAES("admininfo", adminInfo.getPhone()));
adminInfo.setEmail(tool.encryptAES("admininfo", adminInfo.getEmail()));
adminInfo.setPassword_md5(tool.encryptMD5(adminInfo.getPassword()));
adminInfo.setPassword(tool.encryptAES("admininfo", adminInfo.getPassword()));
dao.updateAdminInfo(adminInfo);
return "redirect:/admin/admin_list/1";
}
```

这个模块的实现时，采用了现在世面上流行的 Spring+MyBatis 框架来完成，使用 SpringMVC 实现了 MVC 结构，达到了系统降低耦合度的目的。

处理的流程为：用户请求给 DispatcherServlet 主控制器，主控制器调用 HandlerMapping 根据请求找相应的 Controller 处理执行 Controller 约定方法，可以调用其他业务组件，例如 DAO 等，将结果数据放入 Model 对象，处理完返回一个视图名，主控制器调用 ViewResolver 根据 Controller 返回的视图名去调用相应的视图组件 JSP 生成响应信息。

MyBatis 封装了 JDBC 操作，将 SQL 查询结果映射为对象，将对象属性值映射到 SQL。

具体实现可以分为以下过程:

首先解析 SqlMapConfig.xml 和 SqlMap.xml(定义 SQL)，将解析出的 SQL 生成 MappedStatement(预编译 Statement)，将传入的 SQL 参数给 Statement 绑定，参数类型可以为 java 对象、Map 类型、int、String 类型，执行 mappedStatement,返回 SQL 结果，将返回的 SQL 结果转化为 java 对象、Map、Int、String 类型返回。

4.3 加解密引擎的实现

4.3.1 加解密引擎

由于 Java 语言的简单易用性，而且 JDK1.7 对加密体系提供了完善的支持。本系统采用的 AES 加密算法加密，虽然 JDK 本身也提供了实现方式，但是其过程比较繁琐，BouncyCastle 封装了更加简单便捷的操作。提供了 JCE1.2.1 的实现，因为它是轻量级的密码术算法包，因此在 JDK1.4 后，都是可以运行的。

生成密钥 Key 的过程：

```
KeyGenerator keyGenerator =KeyGenerator.getInstance("AES");

keyGenerator.init(new SecureRandom());

SecretKey secretKey = keyGenerator.generateKey();

byte[] keyBytes = secretKey.getEncoded();
```

Java 中提供了 KeyGenerate 用于生成 Key，只需要提供加密算法即可。BouncyCastle 提供加解密引擎包，只需要提供加密算法、填充模式以及加密模式和密钥长度即可。加密过程中的主要实现如下：

```
Key converseKey = new SecretKeySpec(keyBytes, "AES");

Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");

cipher.init(Cipher.ENCRYPT_MODE, converseKey);

byte [] result = cipher.doFinal(data.getBytes());

data= Base64.encodeBase64String(result);
```

进行加密时首先进行密钥的转换，Java 中对于密码的操作封装在 javax.crypto 包下,在获取密码工具类 Cipher 时需要指明加密算法、工作模式以及填充方式。

解密的操作与加密相反，同样是获取密钥后，转换密钥，获取实例的方式也不会改变，即在初始化时将声明为解密模式 Cipher.DECRYPT_MODE，解密的核心代码如下：

```
Key converseKey = new SecretKeySpec(keyBytes, "AES");

Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");

cipher.init(Cipher.ENCRYPT_MODE, converseKey);

byte [] result = cipher.doFinal(data.getBytes());
```

4.3.2 数据库连接模块

由于 Mysql 的免费和开源，提供多语言的支持，并且相比于 Oracle 和 SqlServer 它是轻量的，因此本系统采用 Mysql 数据库来存储数据。

数据库连接模块的任务是把频繁重复的数据库连接动作封装在一个工具类中，采用 MyBatis 框架，将 JDBC 操作进行封装，数据库的连接只需要在 Spring 框架的配置文件中配置一次即可。配置代码如下：

```
<bean id="mydatasource" class="org.apache.commons.dbcp.BasicDataSource">
        <property name="driverClassName" value="com.mysql.jdbc.Driver"/>
        <property name="username" value="root"/>
        <property name="password" value="mysql"/>
        <property name="url" value="jdbc:mysql://localhost:3306/enterprise"/>
    </bean>
```

4.4 密钥信息模块

在密钥的信息模块实现的是密钥的获取和存储，其获取从直观上看是在直接检索数据库的密钥表。密钥表中保存的是一级工作密钥加密后的密文数据，如图 4-5 所示：

id	tname	skey
1	admininfo	GkLMqJl514jto2i99Bvy10v1/XAmrbsNvd82DjbbpKo=
2	dept	mDDTkB2UxJsJpmUTAeiAGpQ9jVFK4Bvq9r9huYuPdSo=
3	emp	enHw0IC/3zucOCqA2mj2I/I/dRmu1+HmD0yV6iarFNU=

图 4-5 密钥信息表中数据

为安全起见，对工作密钥加解密的二级密钥保存在本地的文件系统中，可以保存在记事本里，通过 Java 提供的输入流来获取保存在本地的二级密钥完成对一级工作密钥的解密。二级密钥在本地的存储形式如图 4-6 所示：

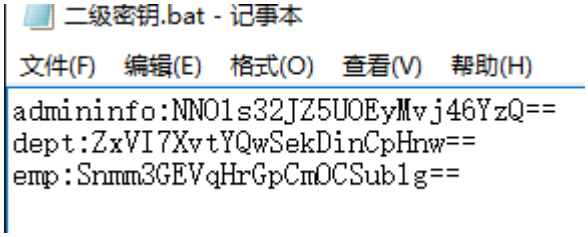

图 4-6 本地二级密钥的存储

5 系统功能与测试

5.1 系统功能

5.1.1 企业数据库加密系统登录

在使用企业人事管理系统之前，首先需要将本项目部署到服务器上，本地采用tomcat7.0.2 作为应用服务器。然后启动 tomcat 服务器。为了测试服务器是否已经启动成功，在浏览器地址栏输入：localhost：8080，结果如图 5-1 所示，则启动成功。

图 5-1 tomcat 启动成功

服务启动后，在浏览器地址栏输入本系统的登录路径：http://localhost:8080/Enterprise-Security/login/toLogin 跳转到系统登录页面如下：

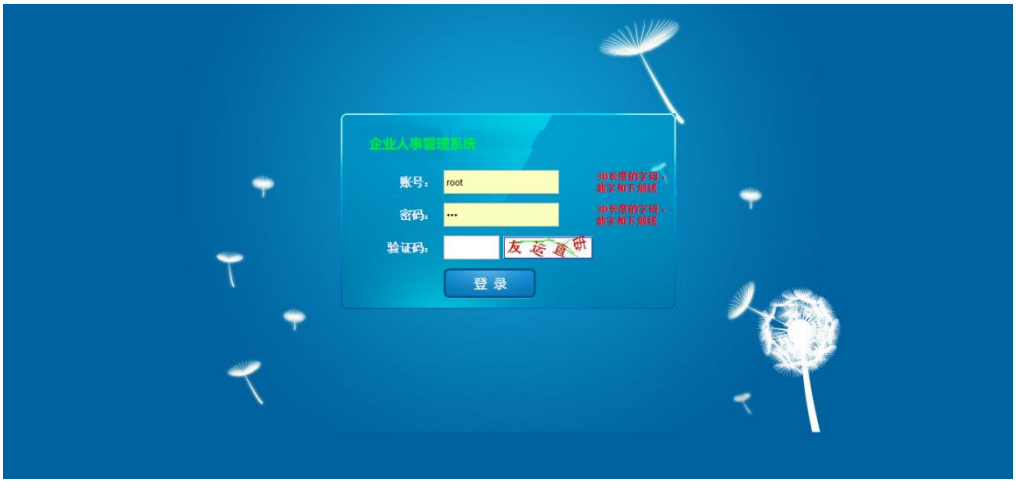

图 5-2 企业人事管理系统登录页面

5.1.2 首页

在登录页面输入用户名，密码，以及验证码后如果验证成功，会进入到首页页面，失败则返回错误提示信息，重定向到系统的登录页面，首页效果如下图 5-3 所示：

图 5-3 系统首页展示

在首页可以直观的看到系统的各个模块，然后就可以根据需求成功的操作本系统。

5.1.3 管理员管理

在首页上直接点击管理员图标，如果拥有超级管理员权限，则可以直接进入到管理员管理列表，显示了所有本系统的管理员信息，可以进行增删改查操作。如果仅仅为一般管理员则跳转到无权限出错页面，效果如图 5-4 所示：

图 5-4 管理员管理列表

管理员列表采用了分页技术，方便用户的查看，且每条数据后面可以进行相应的修改和删除。

图 5-5 无权限操作提示页面

如果当前管理员没有超级管理员权限则不能操作管理员管理模块，如图 5-5 所示。

5.1.4 员工管理

这个模块的功能，所有本系统的管理员都可以对其进行操作，当首次点击会进入到员工信息列表，如下图所示：

图 5-6 员工管理列表

修改员工信息操作如图 5-7 所示：

图 5-7 修改员工信息

系统给出了相关的操作提示,避免用户在修改信息时出错,提高了系统使用的便捷性。

在列表页面，可以直接进行删除操作，如图 5-8 所示：

图 5-8 删除员工信息

增加员工：

图 5-9 增加员工信息

5.1.5 部门管理

部门管理模块，这要是为企业提供 部门信息的管理，用户同样可以进行相应的增加，删除，修改等操作，需要提示的是，用户最好根据提示信息录入数据信息，这样可以提高工作的效率，也减少了录入数据的出错率。

图 5-10 部门信息列表

修改部门信息时，如果出错会有数据出错提示，根据提示操作即可，如图 5-11 所示：

图 5-11 录入信息时的错误提示

5.1.6 修改密码

修改密码这个模块，主要的是为了当前用户的密码出现任何问题时想要改改密码提供便捷操作，首先用户需要输入当前的密码，然后输入两次要修改的新的密码，在当前密码验证通过后，还要进行两次输入密码的相同性比较，如果输入无误，才可以操作成功否则会有错误提示信息，重新进行输入，效果如图 5-12 所示：

图 5-12 修改密码

5.2 功能模块实现数据库加解密

系统所有模块，在进行数据库存取时都实现了加解密操作，在查询时，首先从数据库中获取到密文数据，完成数据的解析，最终显示到页面。系统设计的二级密钥保护，实现过程封装在加解密引擎中。

在每次操作时，调用引擎模块，完成对数据的加解密处理，数据库中为密文存储，页面显示明文数据。如图 5-13 所示：

id	login_name	password_md5	password	real_name	idcard_no
2	lisi	202cb962ac59075b964b07152d234b70	qUZWI+9aZEyW5WXpv0TTyw==	李四	Gp4PhYYaJdXJa/Cc9ZqT
3	root	202cb962ac59075b964b07152d234b70	qUZWI+9aZEyW5WXpv0TTyw==	罗冲	gu5xZANnN1FXRKdrlolcm
4	ye_w	96e79218965eb72c92a549dd5a330112	mS9yaPBtID1t03PauaT+Ig==	叶伟	dQj1pCeZAyVbQl8VdH9e
5	wangwu	81dc9bdb52d04dc20036dbd8313ed055	tB8IVmEnCaivZa2vT09jWQ==	王五	N5nIzQZnL/RbjwLZWORI
6	lyh	b59c67bf196a4758191e42f76670ceba	2PcpnxgZwUbp90oNu92uSQ==	刘雨恒	KeVoXAw0bX7vXlBeBEW/

图 5-13 数据库中的密文存储

用户操作时显示明文数据：

图 5-14 操作页面的明文数据

5.3 系统测试

系统测试渗透整个开发过程，在开发完一个模块都会进行测试，预先估计出正确的显示结果，然后设计测试用例去完成测试，对比结果，检测是否满足预期结果。

（1） 登录检测

表 5-1 系统登录检测方案

测试用例	管理员登录
测试类型	通用性测试
测试对象	管理员超级管理员
输入数据	用户名，登录口令，验证码
执行步骤	普通管理员以及超级管理员 1、进入登录页面填写用户名和口令以及验证码 2、点击登录，进入主页
预期结果	能够登录到主界面输入用户名、口令、验证码，输入错误，有提示信息显示。
实际结果	完成满足
测试结论	PASS

系统实际效果图如下：

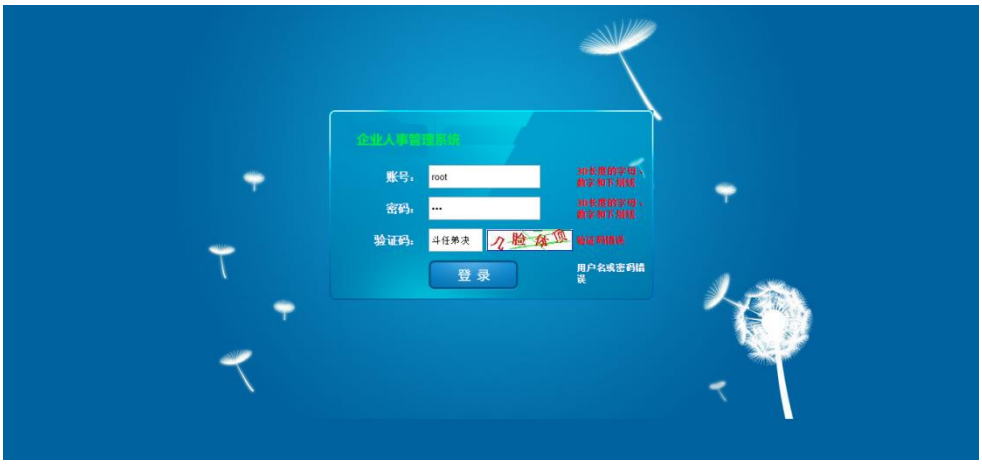

图 5-15 登录时错误提示

（2） 数据加解密验证

为了验证系统已经实现了用户数据的加密，而且能够在管理员查看时还原为明文，所以在用户进行每一步操作时，可以在后台数据库直接查看经过处理后的密文，方案设计如表 5-2 所示：

表 5-2 管理员模块检测方案

测试案例	管理员模块
测试类型	后台密文数据是否修改
测试对象	超级管理员
输入数据	在管理员模块进行增加操作
执行步骤	1、点击进入管理员模块，执行修改数据项 2、修改完成后点击提交，在后台数据库查看密文数据的修改，以及前端明文的变化
预期结果	数据库中密文数据发生了改动，浏览器中明文也做出了相应的变动。
实际结果	密文数据做出了相应的修改
测试结论	PASS

测试结果如 5-16-5-18 所示：

图 5-16 管理员模块修改数据

	real_name	idcard_no	gender	phone	email
tEyW5WXpv0TTyw==	李四	Gp4PhYYaJdXJa/Cc9ZqTZw==	0	uHSTByXOpq0Yb+dx/I1KKg==	s0uMc
tEyW5WXpv0TTyw==	罗冲	gu5xZANnN1FXRKdrlolcmNRpSogs4nRpkAIM9Oi...	0	9JKBYQb1K3F5fYgoIuUrFA==	s0uMc
ivZa2vT09jWQ==	叶伟最帅	dQj1pCeZAyVbQl8VdH9eahiKw9FINwm7ED54W...	0	si5WXTbhrjQGQVvTmX5yFQ==	dhyxE
ivZa2vT09jWQ==	王五	N5nIzQZnL/RbjwLZWORPaufbZbI3qXQdptVL3H...	0	lf1ea/mM57Ws2pfCuwh23w==	I1bnP
Jbp90oNu92uSQ==	刘雨恒	KeVoXAw0bX7vXlBeBEW/Ww==	0	/6I4slgKbUT/Mm88OqP1BQ==	CAAVF

图 5-17 数据库中之前保存的密文数据

rd	real_name	idcard_no	gender	phone	
9aZEyW5WXpv0TTyw==	李四	Gp4PhYYaJdXJa/Cc9ZqTZw==	0	uHSTByXOpq0Yb+dx/I1KKg==	s
9aZEyW5WXpv0TTyw==	罗冲	gu5xZANnN1FXRKdrlolcmNRpSogs4nRpkAIM9Oi...	0	9JKBYQb1K3F5fYgoIuUrFA==	s
3CbsQPnhZzL5jlg==	叶伟最帅	dQj1pCeZAyVbQl8VdH9eahiKw9FINwm7ED54W...	0	2eMDy6mDSotAESdA+Acmw==	d
nCaivZa2vT09jWQ==	王五	N5nIzQZnL/RbjwLZWORPaufbZbI3qXQdptVL3H...	0	lf1ea/mM57Ws2pfCuwh23w==	I
ZwUbp90oNu92uSQ==	刘雨恒	KeVoXAw0bX7vXlBeBEW/Ww==	0	/6I4slgKbUT/Mm88OqP1BQ==	C

图 5-18 数据库在修改操作执行后的密文数据

6 总结与展望

6.1 总结

基于 Java 的企业人事管理系统的设计与实现，让我学习了到了以前没有接触的数据加密，密码学方面的知识，包括各种加密算法如 DES、AES、MD5 消息摘要算法等，让我对于安全领域有了新的认识和好奇。同时了解到了 Java 语言对各种加密算法的整合与实现。在系统的实现过程中当然还需要一定的前端知识。为了使整个系统分层降低耦合度，研究了 MVC 设计结构，采用了 SpringMVC 来实现 MVC 架构。同时系统使用了 MyBatis 持久层框架。在这些技术整合后，我实现了我预期的效果，为企业人事管理系统写出了满意的页面，实现了整个系统的数据加密存储与查询，并且对加密密钥实现了二级加密，保护密钥的安全性。

在老师和同学的指导下，自己亲手完成了这个系统的设计与开发，我更加了解一个系统开发需要经历的步骤，以及对 Java 语言掌握的更加牢固，以前忽略的点，在开发过程中暴露，通过自己的调试改正，对以前的学习误区有了新的认识，对流行的 B/S 开发模式有了更加深入的了解，基本掌握了使用 Spring+MyBatis 框架进行项目的开发。在开发过程中，也遇到了很多问题，如数据加密后存储，前端查询时显示密文，解密出错，经过耐心的排查也都一一解决了。通过这次毕业设计很大程度的锻炼了我的技术以及克服了我之前对 Bug 的恐惧，提高了自己的自信心。

6.2 展望

由于以前没有做过系统的开发，接触的都是一些 Demo 的实践，缺乏一定的理论知识与开发实践，难免系统设计的不合理和完善。本系统的主要缺陷有以下几点：

（1）由于对数据备份方面的知识欠缺，无法实现已删除信息的恢复。

（2）对前端知识学习较少，系统的界面比较单调，有待进一步提高。

（3）数据库的设计不是很合理，所有管理员信息都存储在同一张表中，在判断超级管理员时只能将代码写死，在修改数据后，可能会引起系统错误，在接下来的工作中会进行这部分的修复。

（4）在加密时，只考虑了数据本身是否为敏感信息，对其进行加密后，会影响某些功能操作，如对薪资加密后，如果执行对薪资的相关条件查询将无法进行接下来也会对此进行完善。

在做毕设的过程中本身就是一个学习与发现自身不足的阶段，通过动手来做，不断的提高与完善，并积累一定的开发经验，学会去发现问题、解决问题，对自己以后的工作和学习也打下了夯实的基础。

致　谢

在半学期的学习与努力中毕业设计就这样的完成了。对系统软件的开发有了更深的了解，也当是自己即将迈入职场前的最后一次考验吧。非常感谢我的毕设老师闵祥参老师，她为我提供了很多优秀的参考文献，与设计思路，帮助我完成开题报告，中期汇报检查，以及为我指导系统中存在的漏洞，提出建议让我的系统变得更加的完善。另外，我也要感谢我的那些好朋友，在我完成毕设期间，帮我解决技术问题，分享给我技术资料，使我的毕业设计能顺利完成。

参考文献

[1]Jim Arlow . UML2.0 and the Unified Process[M]. 机械工业出版社, 2006 年:2-56.

[2]王保罗. Java 面向对象程序设计[M]. 清华大学出版社, 2003 年:89-103.

[3]刘京华. Java Web 整合开发王者归来[M]. 清华大学出版社, 2010 年:45-56.

[4]杜波依斯. MySQL 技术内幕[M]. 机械工业出版社, 2011 年:45-82.

[5]张银鹤. JSP+Ajax 网站开发典型实例[M].电子工业出版社, 2009 年:78-115.

[6] Metsker S J. Java 设计模式[M].电子工业出版社, 2012 年:45-92.

[7]孙卫琴. Tomcat 与 Java Web 开发技术详解[M]. 电子工业出版社, 2009 年:96-123.

[8]吴烈,唐伟. 考勤工资管理系统的设计与实现[D].辽宁工程技术大学.

[9]贺松平. .基于 MVC 模式的 B/S 架构的研究及应用[M]. 华中科技大学, 2006 年 4 月:1-12.

[10]朱红,司光亚. JAVA Web 编程指南[M].电子工业出版社, 2001 年:75-85.

[11]罗时飞. 精通 Spring—深入 Java EE 开发核心技术[M]. 电子工业出版社, 2008 年:12-45.

[12] (美)哈罗普 (美)马可赛克. Spring 专业开发指南[M].电子工业出版社, 2006 年:45-85.

[13]Arlow,Ila Neustadt.UML 2 and the unified process[M]. practical object-oriented analysis and design,人民邮电出版社,2006 年:12-14.

[14]Bruce Eckel. Thinking in Java 4[M]. 机械工业出版社. 2007 年:32-56.

[15] 李茹. 《数据库系统概论》的教学实践与探索[J]. 山西教育学院学报,1999,02:119-120.

[16] 李茹. 《数据库系统概论》的教学实践与探索[J]. 山西教育学院学报,1999,02:119-120.

[17]赵晓峰,叶震. 几种数据库加密方法的研究与比较[J]. 计算机技术与发展,2007,02:219-222.

[18]李春堡,曾平. 数据库原理与应用[M].清华大学出版社.2008(4):48-56.

[19]朱勤,骆轶姝,乐嘉锦. 数据库加密与密文数据查询技术综述[J]. 东华大学学报(自然科学版),2007,04:543-548.

[20]李刚彪. 数据库加密技术的研究与实现[D].太原理工大学,2010.