

“网络安全综合实验（I）”实验报告

**题目： 数据库安全基础实验**

院 系 网络空间安全学院

专业班级 xxx

姓 名 xxxx

学 号 xxxx

日 期 2021 年 5 月

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 评分表 | 数据库安全基础实验报告评分  （50分） | | | | 完成  （50分） | 成绩 | |
| 评分 项目 | 过程 | 分析与小结 | 撰写 | 创新 | 完成任务 | 合计 | 教师 |
| 分值 | 20 | 20 | 5 | 5 | 50 | 100 | 签名 |
| 评分 |  |  |  |  |  |  |  |

报告要求及评分规则

1.请按照模板给出的格式，包括行距、字体、段落格式等，完成报告；

2.报告封面保持一致(如实填写基本信息及完成日期)；

3.实验报告内容应包括：封面、评分规则、实验过程记录、实验问题分析与总结、参考文献及资料列表；

4提交：电子材料应包括本实验电子版（doc）、实验参考文献资料的电子资源文件（pdf）。

5.主要考察能力： 实验动手能力、问题分析与归纳能力、文档规范撰写能力、创新能力；

6.其他要求：可按各指导老师要求的时间和提交方式提交；每次课实验报告可以单独提交；如果需要最终纸质报告的，可去掉模板中说明文字（斜体）之后，双面打印；

7.总评分=课程每次实验分数之和/课程实验次数。

其中：

每次实验分=实验完成分（50分）+实验报告分（50分）-扣分+加分；

计算方法：

实验完成分=（完成任务数量/总任务数量）\*50；

实验报告分=以下1-4项合计；

1）过程（要求：实验过程完整、清晰）(满分20)（优秀：18+ 良好：16+ 一般：14+）；

2）问题分析与小结（要求：有条理、细致）（满分20）（优秀：18+ 良好：16+ 一般14+）；

3）撰写（要求：语句通畅、格式规范）（满分5）（优秀：4+ 良好：3+ 一般：2+）；

4）创新（要求：见解独到、有创意）（满分5）（优秀：4+ 良好：3+ 一般：2+）；

扣分=报告迟交天数\*2分（满分10分）  （发现雷同抄袭的内容，该次实验不得分）；

加分=搜集整理与实验相关的学习资料作为附件，资料能帮助同学更好掌握相关知识的；获得同组人互评优秀的；提交最终报告时间为班级前3名。（满分5分）（项数\*1分）；

同组评价：

本实验中你的同组人姓名： xxxx 学号： xxxxxx

你给同组人评价： 优 （优 良 中 及格）；

理由： 对于Linux的基本命令熟悉，能熟练使用命令行进行操作。勇于并善于探索问题、解决问题，善于使用搜索引擎搜索问题的解决方法，并加以实践进行验证。

# 

# 数据库安全基础实验

## 实验环境及要求

### 实验平台及说明

本实验同组成员： xxxx 、 xxxx ；

### 实验场景设置

2021年春天，你已经熟悉了“银河”公司的Linux服务器基本情况。公司指派你和Bob，负责数据管理员的工作。在了解几种类型数据操作的同时，寻找发现安全相关问题，并利用mysql的功能解决。

根据以上场景，完成以下操作，作为数据库安全基础实践通关考核。可组内讨论、查阅资料。

## 过程记录/实验任务（共8个关卡，20个小任务）

## 关卡1 磨刀不误砍柴功

环境准备：为了使你们尽快熟悉业务数据，会将原来备份的一个数据库文件交给你，请你负责导入系统中；为了进行职责分工，将建立两个专门负责Galaxy数据库的账户，给你自己和你的伙伴Bob使用。

1）确认数据库服务mysql已经开启

**解答：**

执行systemctl status mariadb.service命令，查看服务状态，显示已开启，如图1-1所示。

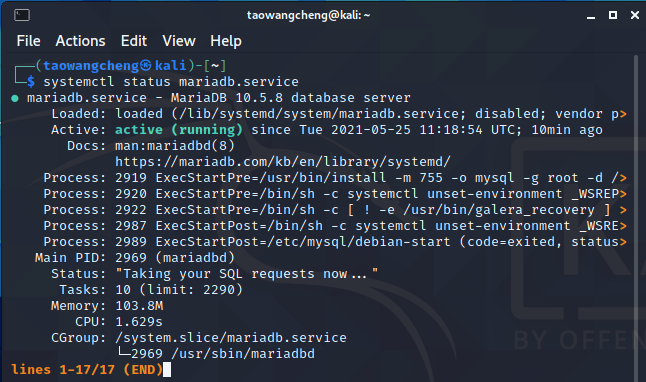


图1-1 mysql服务已开启的截图

2）确认web服务apache2已经开启

**解答：**

执行systemctl status apache2.service命令，查看服务状态，显示已开启，如图1-2所示。

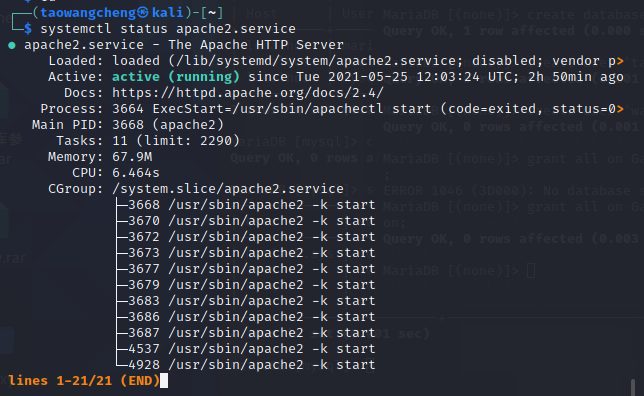


图1-2 apache2服务已开启的截图

3）如果没有开启，请查阅资料，开启mysql/mariadb服务及apache2服务。

**解答：**

图1-1、图1-2显示服务已经开启。

4) 控制台访问数据库

启动数据库控制台，查看已有数据库列表；

创建数据库G*alaxy;*

**解答：**

（a）先切换为root身份，执行命令mysql -u root -p登入数据库（密码为空），再执行命令show databases; 查看已有数据库列表，如图1-3所示。

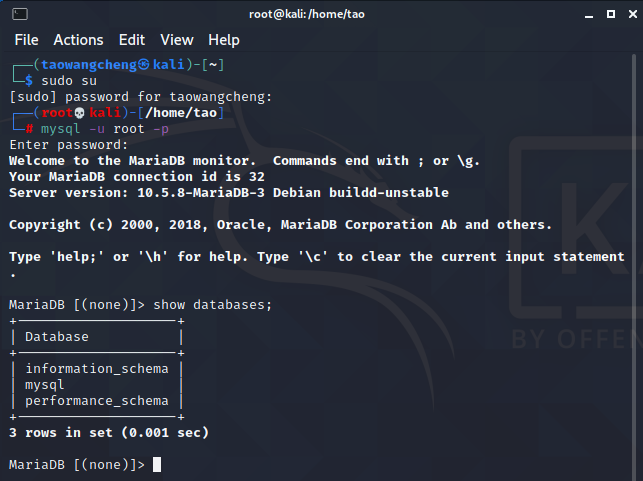


图1-3 查看已有数据库列表的截图

（b）执行命令create database Galaxy;创建数据库, 如图1-4所示。

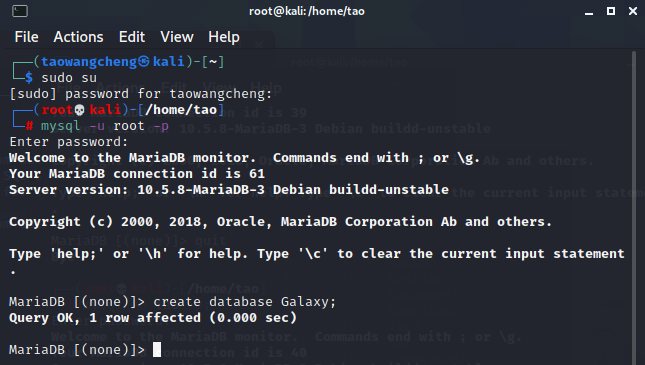


图1-4 创建数据库的截图

5）安装phpmyadmin管理工具

**解答：**

可以采用apt-get安装和压缩包解压安装两种方式，这里采用较为直接的压缩包解压安装方式：

1. 将压缩文件复制到新创建的phpmyadmin目录下 ，如图1-5所示。

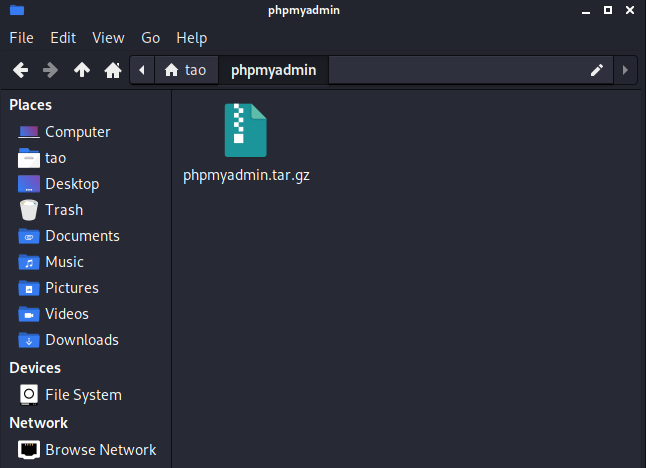


图1-5 复制压缩文件的截图

（b）执行命令tar -xvf phpmyadmin/phpmyadmin.tar.gz解压缩，如图1-6所示。



图1-6 解压缩文件的截图

（c）执行命令sudo mv phpMyAdmin-4.9.7-all-languages /var/www将解压后的文件夹移动到/var/www/php2目录下，如图1-7所示。



图1-7 移动解压后文件的截图

（d）通过浏览器访问数据库管理端，如图1-8所示。

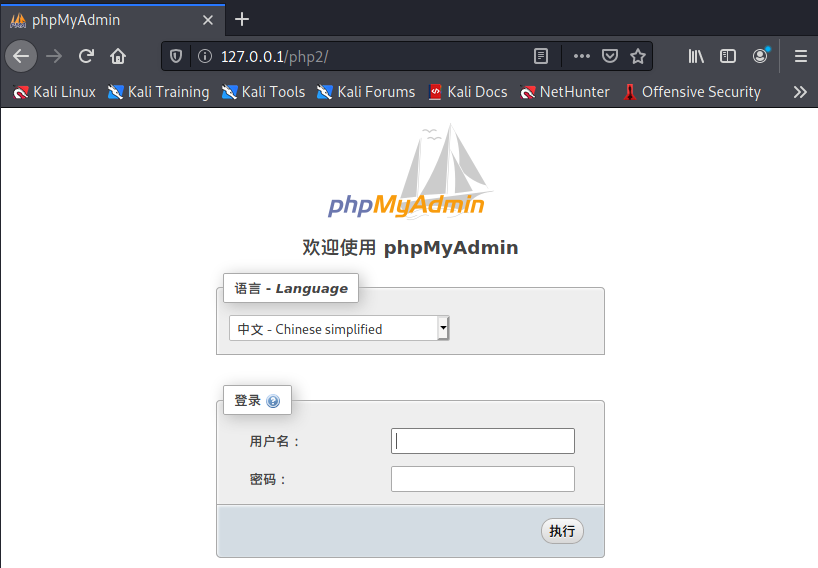


图1-8 访问数据库管理端的截图

root用户不能登录，因为空密码登录被禁止，如图1-9所示。

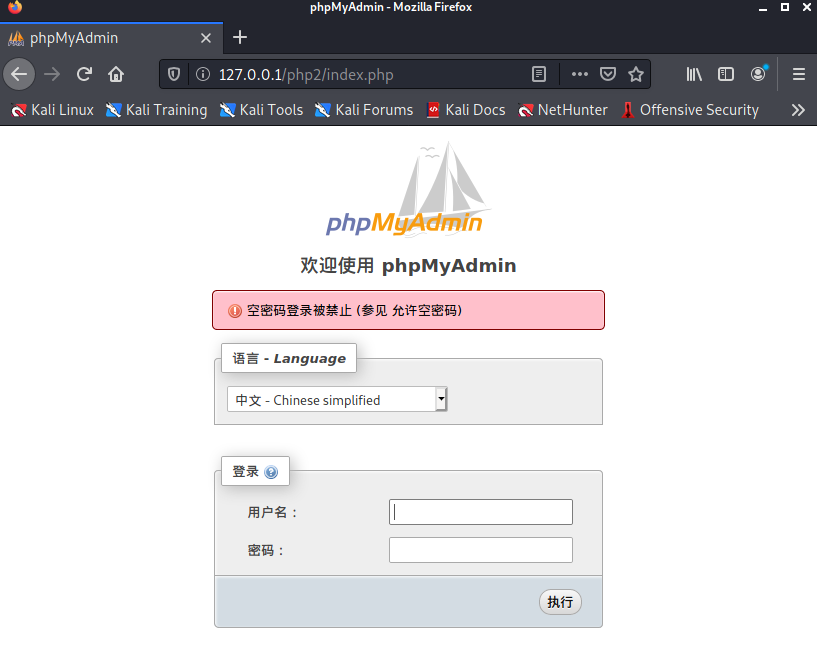


图1-9 root登录的截图

6）创建数据库用户Alice；

**解答：**

（a）登入数据库后，执行命令create user 'taowangcheng'@'%' identified by '123456';创建新账户，如图1-10所示。

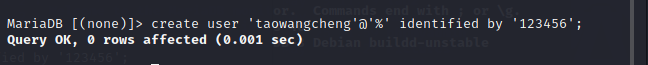


图1-10 创建新账户的截图

（b）执行命令use mysql；进入mysql数据库，再执行命令select

user,host,password from user;查看新账户，如图1-11所示。

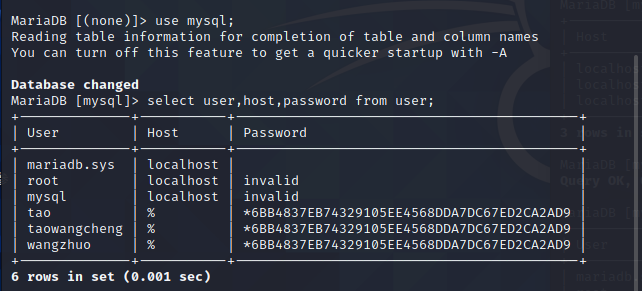
**

图1-11 查看新账户的截图

7）授权访问，允许Alice访问数据库Galaxy，并获得所有权限，可以转授。

**解答：**

执行命令grant all on Galaxy.\* to 'taowangcheng'@'%' with grant option;对新账户授权并允许转授，如图1-12所示。

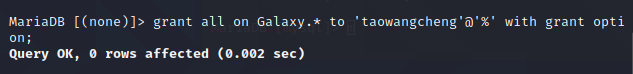


图1-12 通过控制台为新账户授权的截图

8）通过浏览器打开phpmyadmin，用Alice账户登录，对Galaxy进行操作；

**解答：**

浏览器打开phpmyadmin，用新账户登录，如图1-13所示。

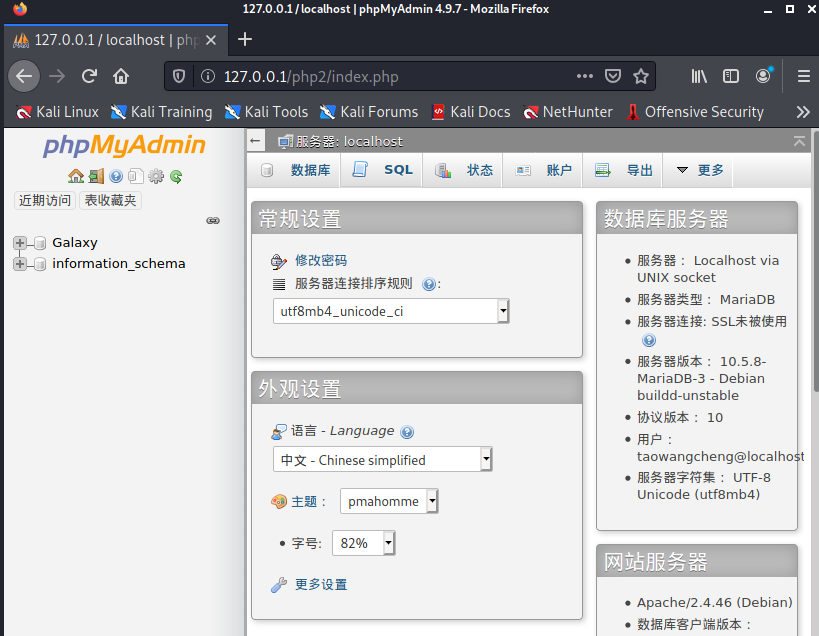


图1-13 新账户通过phpmyadmin访问新增数据库Galaxy的截图

## 关卡2 一个响指

9）数据恢复：从指导老师提供的文件中恢复数据库Galaxy

**解答：**

（a）选择左侧Galaxy数据库，从右边功能中，选择“导入”，选择备份的数据Galaxy.sql文件完成导入，如图2-1所示。



图2-1 通过phpmyadmin工具进行数据库导入的截图

（b）导入成功，如图2-2所示。



图2-2 数据库导入成功的截图

10）查看数据库

导入成功后，通过浏览器查看数据库内容，如图2-3所示，该数据库中目前有数据表：

AirLineInfo,AirTicketDistributor,TicketInfo;

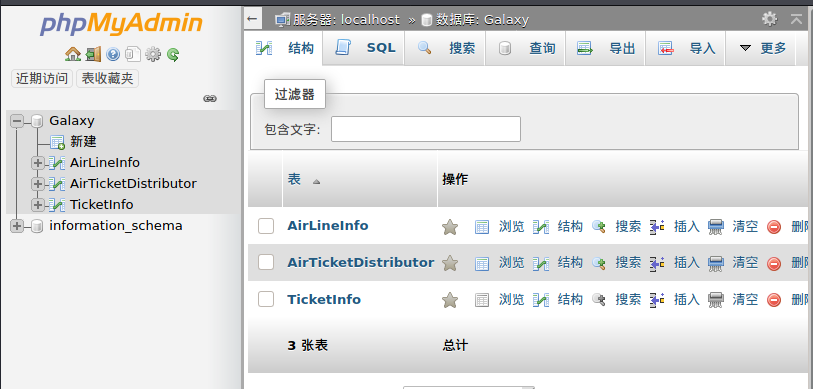


图2-3 查看数据库的截图

另外，查看各表内数据，其中有没有安全隐患？请列出来。

1.分销商密码是明文存放的，数据管理员可以获取分销商的明文口令。

2.分销商密码为123456，该口令过于简单。

3.分销商的电话信息也是明文的，一旦系统被越权访问，可以直接获得分

销商清单。

## 关卡3 24小时紧急任务

11）创建表

根据应急状态下的新需求，需要知道每位乘客的身份证号码和住址，便于在第一时间，联系到乘客，进行信息告知；请你为此需求，增加一张新表格。（提示，包括身份证号码，姓名，电话号码，国籍、住址、邮件地址、出生日期。身份证号码作为主关键字。另外，订票表格中，乘客姓名可以用身份证号码代替，或者额外加上身份证信息。）

**解答：**

通过phpmyadmin工具创建表并添加列，生成SQL语句，预览如图3-1所示。

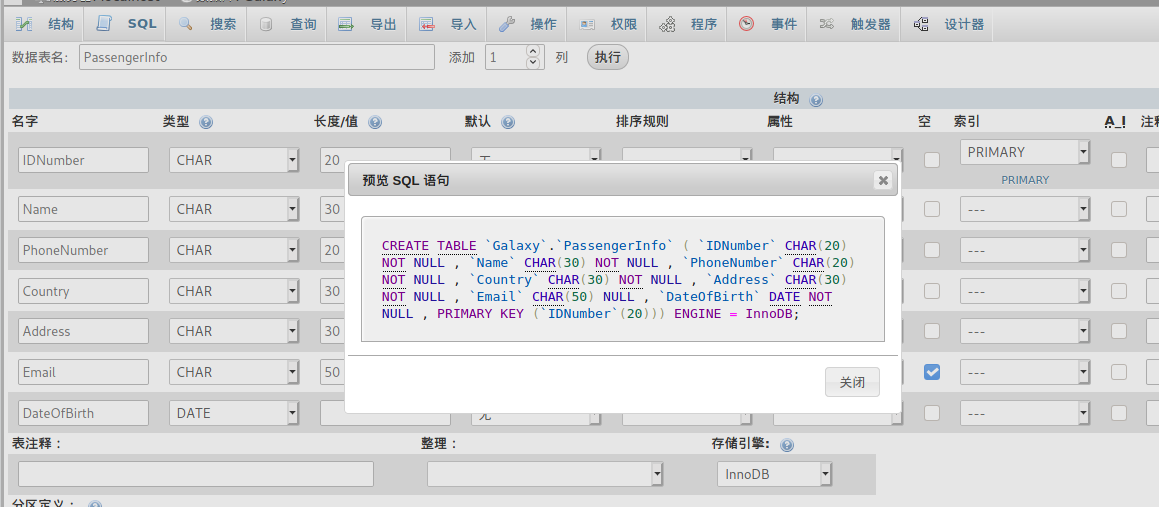


图3-1 SQL语句预览的截图

点击保存，成功创建表，如图3-2所示。



图3-2 创建表的截图

## 关卡4 仿真数据

12）增加表中数据(Insert)

为你的乘客信息表，增加模拟数据，模拟数据尽量保证随机性：身份证号码由数字组成，以你的学号后四位作为开头，姓名用8位字母随机组成，邮件地址以15位以内的小写字母加数字组合为前缀，后面接@，在接标准域名；生日构成，年龄不超过200岁。

**解答：**

（a）通过phpmyadmin工具获得SQL语句模板，如图4-1所示。

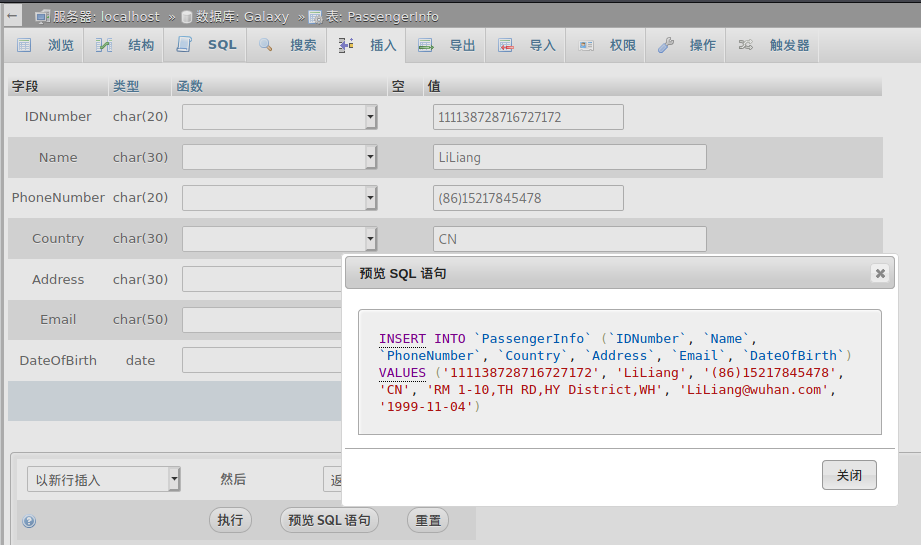


图4-1 SQL语句模板的截图

（b）编写python脚本执行批量数据插入操作（源代码见附件），如图4-2所示。

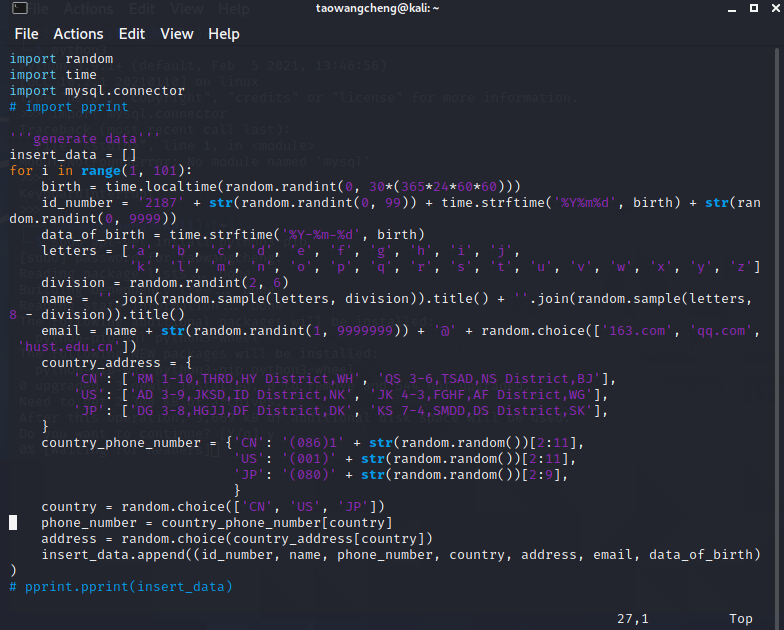


图4-2 编写python脚本的截图

（c）执行脚本，登录Galaxy数据库查看，数据插入成功，如图4-3所示。

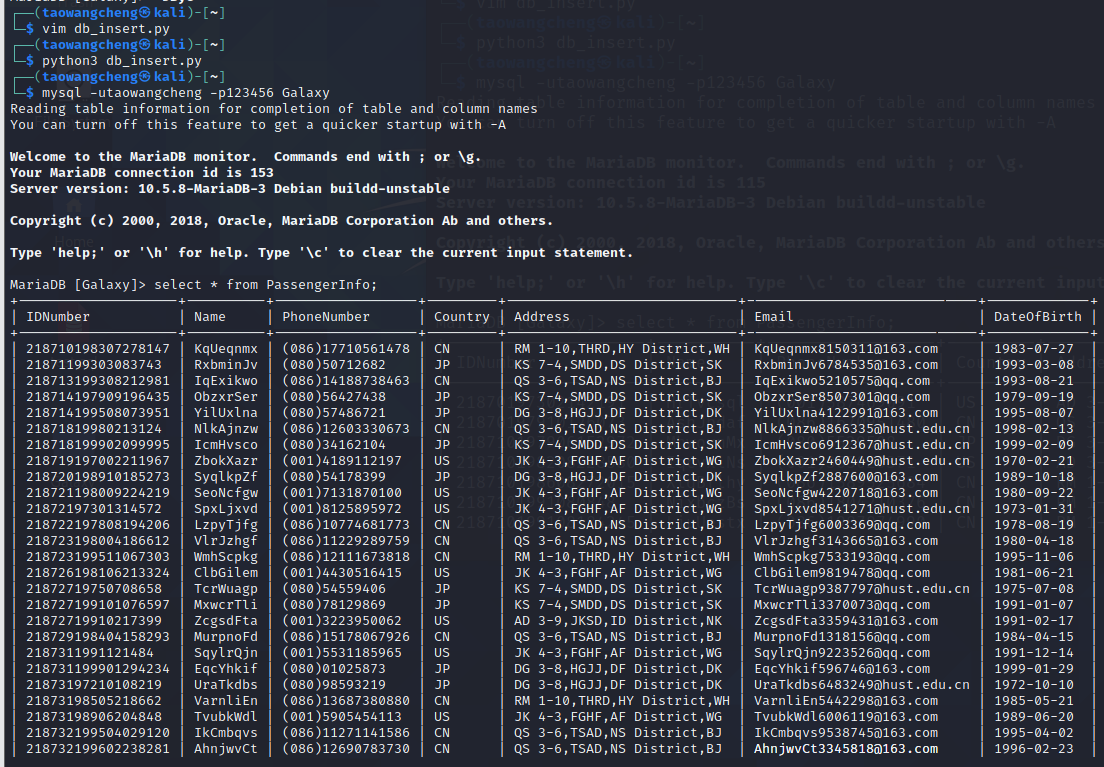


图4-3 成功插入数据的截图

## 关卡5 天气原因

13）修改数据(Update)

因为天气原因，2020年5月28日，飞往上海的航班JD5712推迟1小时，请按班次信息，修改数据库中相应的航班信息；

**解答：**

命令行登录数据库Galaxy后，执行命令Update AirLineInfo set DepartureTime=Date\_add(DepartureTime,INTERVAL 1 HOUR) where FlightNum ='JD5712'and DepartureTime LIKE "2020-05-28%";修改航班信息，如图5-1所示。

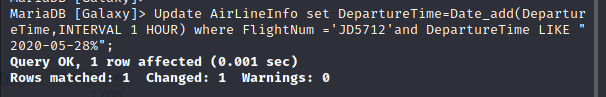


图5-1 更新表的截图

## 关卡6 人多力量大

14）授权他人访问(Grant)

Alice授权Bob对数据库galaxy的访问权限，便于Bob导入乘客信息数据。

**解答：**

（a）关卡一中已经对自己账户授权并允许转授，先执行命令create user 'wangzhuo'@'%' identified by '123456';为同伴创建账户，如图6-1所示；再执行命令grant all on Galaxy.\* to 'wangzhuo'@'%';为同伴账户授权，如图6-2所示。

**

图6-1 为同伴创建账户的截图



图6-2 为同伴账户授权的截图

（b）用同伴账户登录数据库，查看数据库，如图6-3所示。

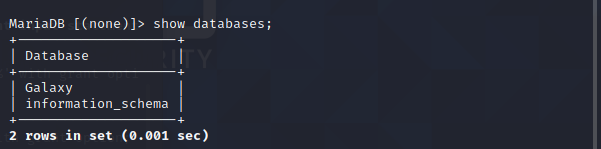


图6-3 同伴账户查看数据库的截图

## 关卡7 百密一疏

15）删除数据（Delete）

以Bob账户，登录数据库；导入了Bob负责产生的数据；但Bob使用中，查询数据后， Bob故意对分销商WangHu信息进行了删除！

**解答：**

（a）以同伴账户登录数据库Galaxy，如图7-1所示。

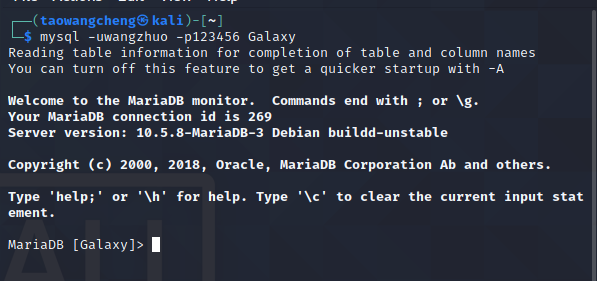


图7-1 同伴账户登录数据库的截图

（b）执行命令select \* from AirTicketDistributor;查看分销商信息，发现 WangHu 账号的信息，如图7-2所示。

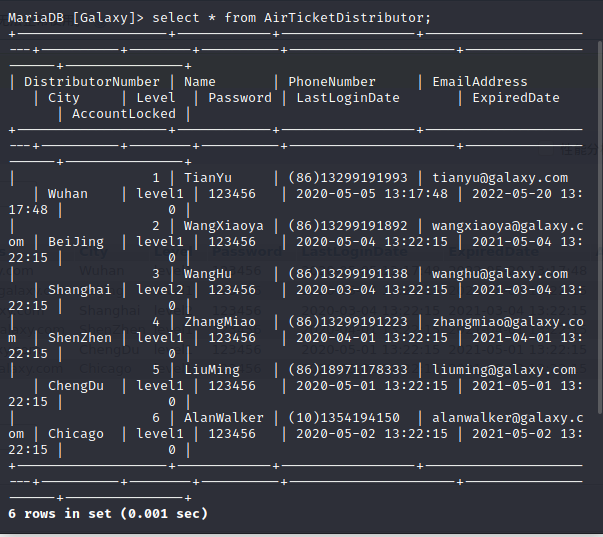


图7-2 查看分销商信息的截图

（c）执行命令Delete from AirTicketDistributor where name='WangHu';删除WangHu账号信息，如图7-3所示。



图7-3 删除分销商信息的截图

## 关卡8 亡羊补牢

16）撤销授权(Revoke)

Alice上线后，发现分销商信息缺失！首先，收回了对Bob的授权，防止损害扩大；

请测试Bob是否还能继续访问Galaxy数据库。

**解答：**

（a）以自己账户登录Galaxy数据库，执行命令select \* from AirTicketDistributor;查看分销商信息，发现 WangHu 账号被删除，如图8-1所示。

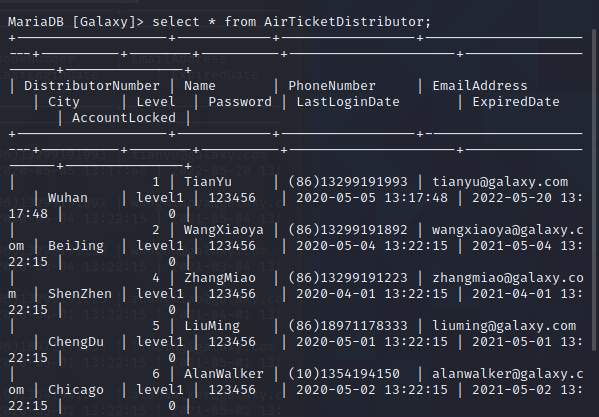


图8-1 分销商信息缺失的截图

（b）执行命令revoke all on Galaxy.\* from 'wangzhuo'@'%';收回对同伴账户的授权，如图8-2所示。



图8-2 收回授权的截图

（c）同伴账户无法再登录Galaxy数据库，如图8-3所示。

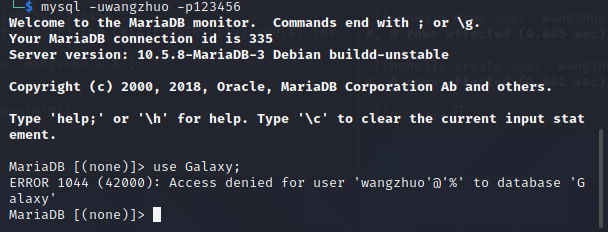


图8-3 验证授权收回成功的截图

17）口令保护（Update）

因为出现的安全问题，引起了你的注意，发现分销商密码是明文存放的，数据管理员可以获取分销商的明文口令。为此，你对该口令进行了简单保护，用hash函数处理密码，将口令的hash值存入数据库的口令字段，代替原来的明文密码。

**解答：**

（a）通过phpmyadmin工具，找到表格后，执行语句UPDATE `AirTicketDistributor` SET `Password`= MD5(Password) WHERE 1;对口令进行保护，如图8-4所示。

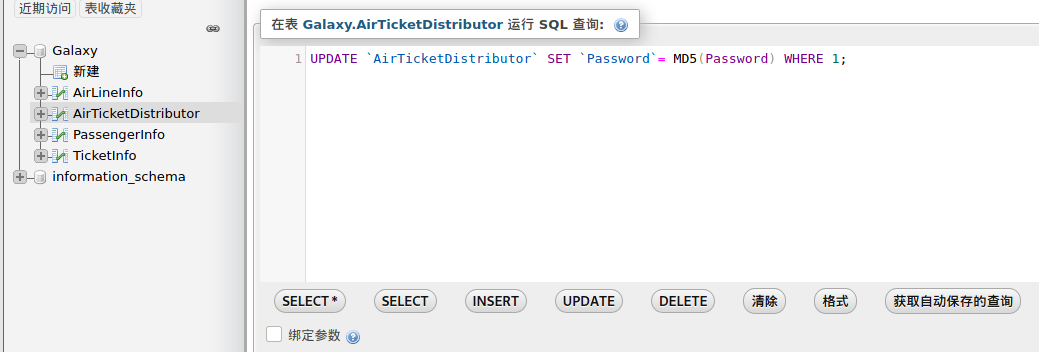


图8-4 加密口令的截图

（b）查看表格，口令保护成功，如图8-5所示。



图8-5 验证口令保护成功的截图

18）加密数据(Encrypt)

同时，因为分销商的电话信息也是明文的，一旦系统被越权访问，可以直接获得分销商清单，因此需要对该项进行安全防护，简单加密表项。

**解答：**

（a）通过phpmyadmin工具，找到表格后，通过结构功能将PhoneNumber的类型修改为char(100)，如图8-6所示。

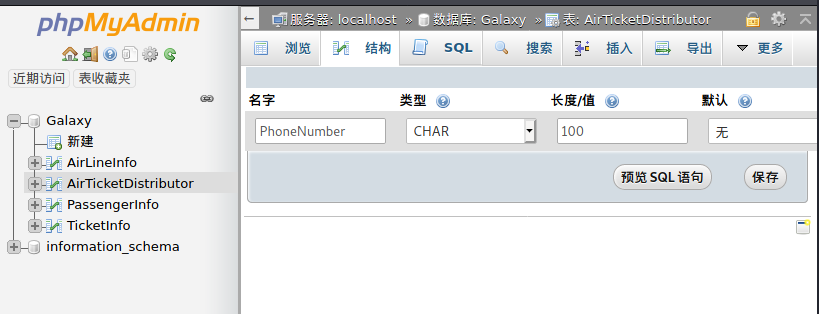


图8-6 修改表格结构的截图

（b）执行语句

update AirTicketDistributor set PhoneNumber=

HEX(AES\_ENCRYPT(PhoneNumber,'1234'))WHERE 1;

对电话信息进行防护，如图8-7所示。

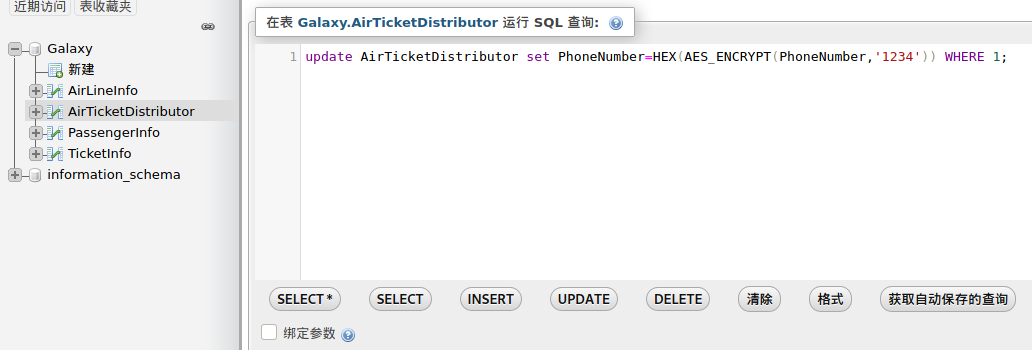


图8-7 电话信息防护的截图

（c）查看表格，电话信息防护成功，如图8-8所示。



图8-8 验证电话信息防护成功的截图

19）备份数据库(Backup)

完成这些记录后，请你对系统进行备份；

**解答：**

执行命令mysqldump -utaowangcheng -p123456 Galaxy >

/home/tao/Backup.sql对Galaxy数据库进行备份，如图8-9所示。

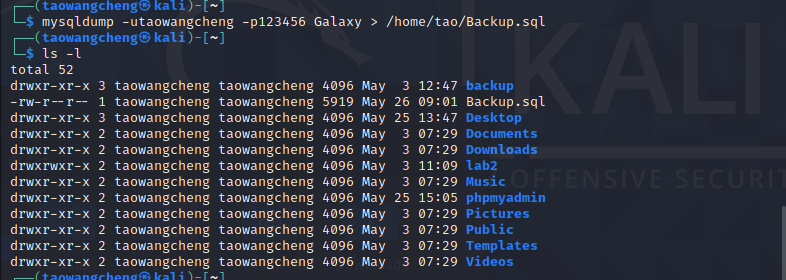


图8-9 备份数据库的截图

20）密文查询

在命令行模式下，查询口令是‘123456’的所有分销商，因为该口令过于简单，准备通知他们修改口令。

**解答：**

登录Galaxy数据库后，执行命令SELECT \* FROM `AirTicketDistributor` WHERE PASSWORD=MD5('123456') or Password='123456';查询，如图8-10所示。

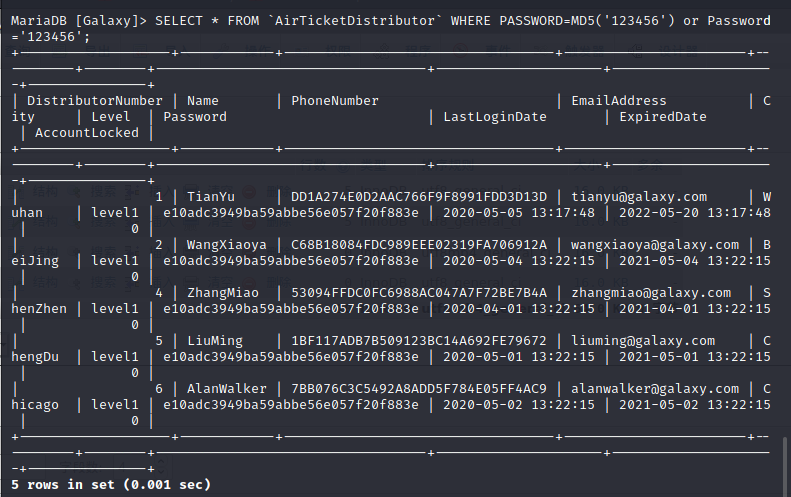


图8-10 查询密文的截图

## 课外阅读与扩展关卡

1）了解kali下mariadb服务是MySQL吗？查找mariadb、Mysql、Oracle相关资料了解MySQL的背景；我们国家有自己的数据库系统吗？

2）[阅读MySQL手册英文版](file:///C:\Users\liumi\AppData\Roaming\Microsoft\Word\3%20Mysql\refman-8.0-en.a4.pdf)中，重点查阅与安全有关的操作，比如Grant/Revoke；MD5；AES\_Encrypt/AES\_Decrypt的段落；

3）\*创新扩展实验：请你建一个仿真数据库，用来记录某城市的注射新冠疫苗的相关信息，假设有100000人；//要记录哪些信息（比如注射疫苗的时间、地点、型号、人的姓名、证件号、联系方式）？谁可以访问这些信息？如何保证信息真实有效？

## 实验问题分析与总结

## 关键分析

（1）数据库基本操作

在尚未学习数据库课程的情况下，学习数据库操作有一定难度，主要通过两种方式解决：一方面，搜索网络资料来学习某一项具体的操作，尤其阅读官方文档；另一方面，借助于phpMyAdmin图形化工具生成SQL语句，方便快捷。

（2）批量数据插入

这是本次实验的难点，主要体现在：

1）首先需要掌握一门编程语言的MySQL数据库API接口实现（通过其它方式也可以）;

2）生成随机数据的难点在于随机数据并不是完全随机的，例如电话号码的前缀、地址应与国家匹配，生日要与身份证号匹配等，还有更多的因素需要考虑，但鉴于实验的主要目的在于掌握批量数据插入操作，并未完全解决这些因素，例如人名要具有意义等。其中值得突出的是对于随机出生年月的处理：

如果采用随机年份、月份、日期的形式，比较复杂，因为不同月份的天数不同，年份对于月份的天数也有影响。这里采用了一个比较精妙的处理（具体可参考源代码），即随机生成秒数，再利用time模块中的localtime函数将其变为时间对象，通过strftime函数即可实现自动获取相应的年月日，而不必考虑具体的实现细节。

3）在执行python脚本的过程中，遇到提示没有模块的错误信息。上网搜索后并未解决问题，后来发现是因为kali中python命令代表python2.x，python3命令才代表python3.x。因此，网络上的解决办法只可作为参考，不能全盘相信。

（3）创新扩展实验可以参照关卡四来做，考虑时间因素，具体不再实践。

## 实验小结：

本次实验主要学习数据库的基本操作，具体包括创建数据库，创建表，插入、更新、删除数据，以及系统管理中的添加用户、授权操作等，并初步了解了数据库中的安全隐患和基础防护措施。通过这次实验，基本掌握了数据库的基本操作，为今后进一步学习数据库安全打下了基础。但要熟练掌握数据库的操作，还需要进一步的学习与实践。

## 参考文献及资料列表

[1] <https://mariadb.org/documentation/>

[2] <https://www.cnblogs.com/developer_chan/p/9229845.html>

[3] <https://blog.csdn.net/qq_40309183/article/details/88913961>