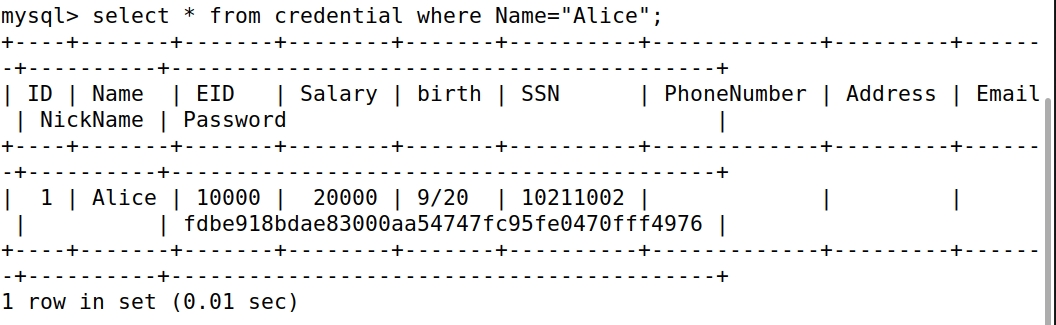
|  |
| --- |
| 哈尔滨工业大学(深圳) |
| **《网络与系统安全》 实验报告** |
|  |
| 实验二  SQL注入 实验  学 院: 计算机科学与技术学院   |  |  | | --- | --- | | 姓 名: | 覃煜淮 | | 学 号: | 220110803 | | 专 业: | 计算机类 | | 日 期: | 2025年4月 | |

# 一、实验过程

每个实验步骤（共9个小任务（任务1.1、1.2，任务2.1、2.2、2.3，任务3.1、3.2、3.3，任务4））要求有具体截图和分析说明。

\*\*任务1.1\*\* 运行上述命令后，需要使用SQL命令打印员工Alice的所有概要信息。请提供你的结果截图。

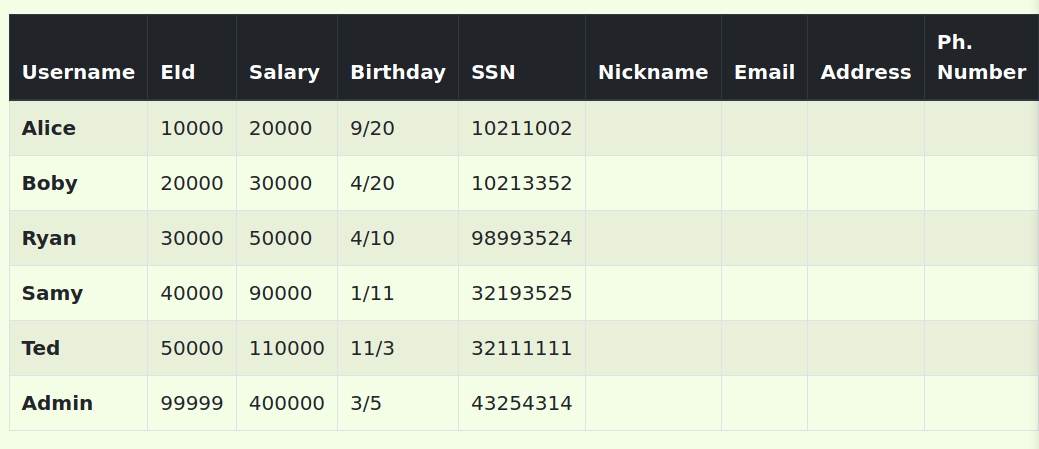
执行use 确定数据库名称，show tables展现数据库中的表名，执行下列sql得到结果



\*\*任务1.2\*\* [选做，需要安装sqlmap]运行上述命令后，可以将credential中的所有信息dump下来。请提供你的结果截图，并根据第一条命令找出可以注入的信息点。

\*\*任务2.1\*\*:网页SQL注入攻击。利用Admin账户，可以尝试使用其他账户。

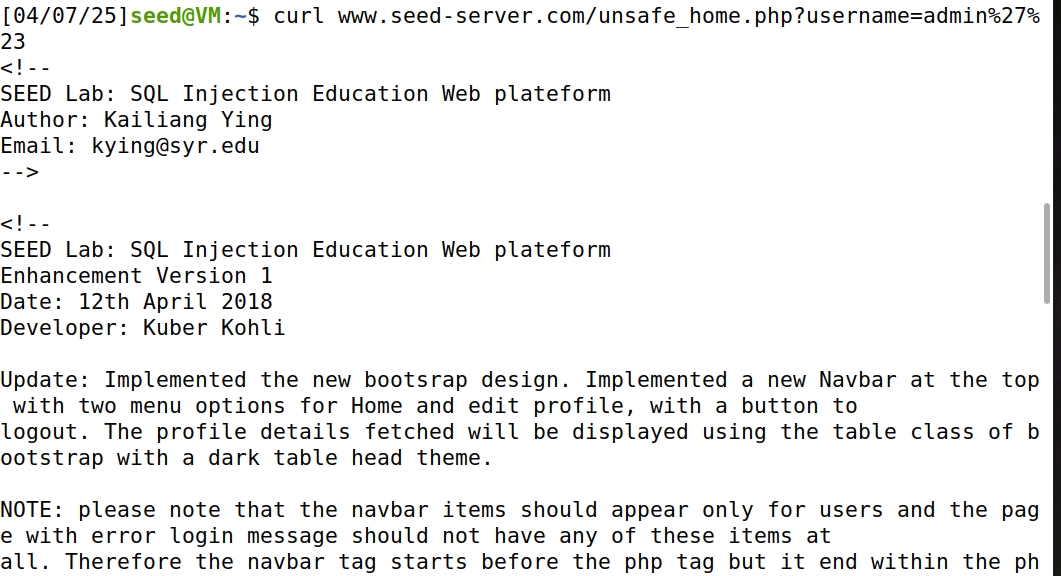
在用户名处输入admin’;#即可获得所有账户信息，此处的#号表注释掉了之后的where判断条件的内容（即password）

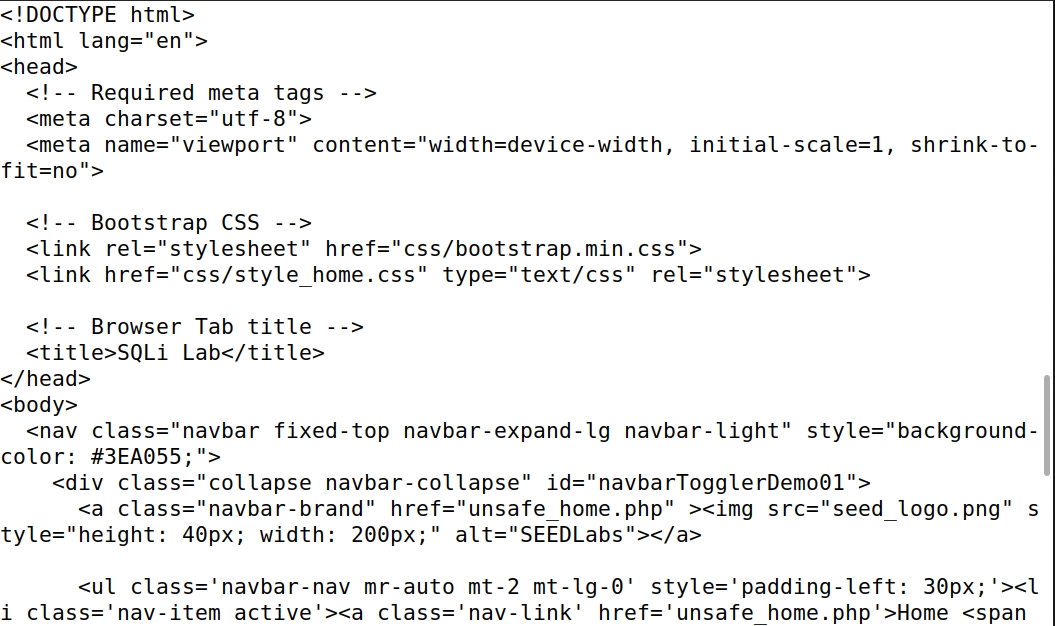


\*\*任务2.2\*\*:命令行SQL注入攻击。利用curl命令进行攻击

命令行攻击命令以及截图如图所示：

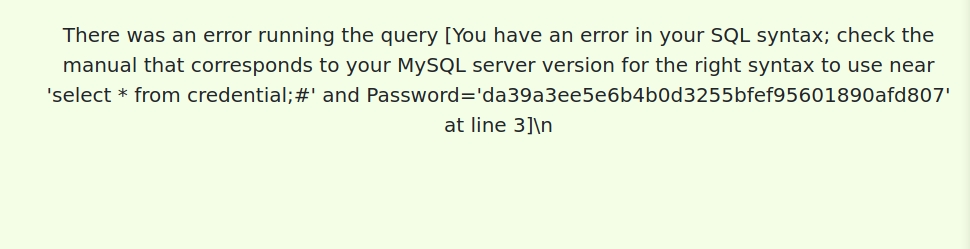
在URL中%27代表单引号，%23代表#





\*\*任务2.3\*\*:追加一条新的SQL语句。请尝试通过登录页面运行两条SQL语句,并说明是否能够获取到信息。

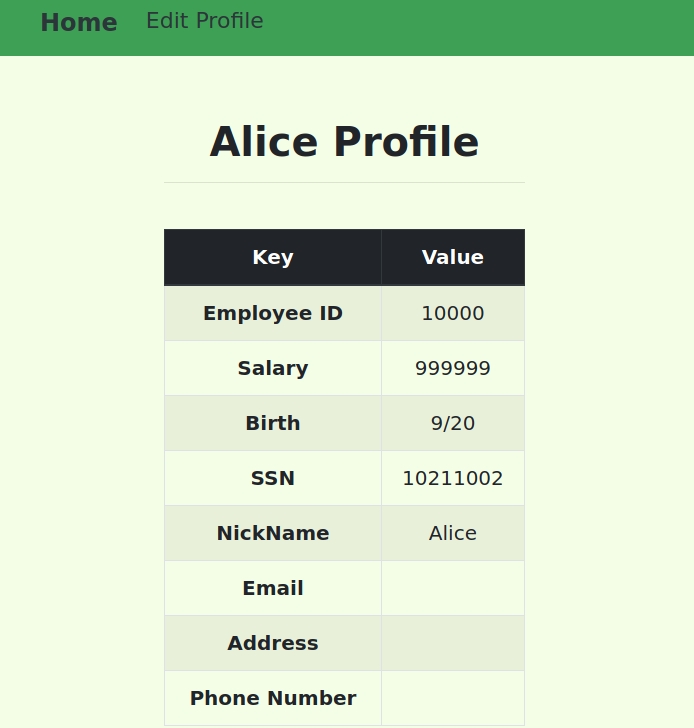
尝试追加一条sql语句在admin之后，出现错误如下，未获取到信息



未获取到信息的原因：

后端使用了预编译语句或参数化查询，防止了多条sql的注入

\*\*任务3.1\*\*:修改自己的工资。



在NickName处输入Alice’,Salary=999999 where Name=”Alice”;#即可

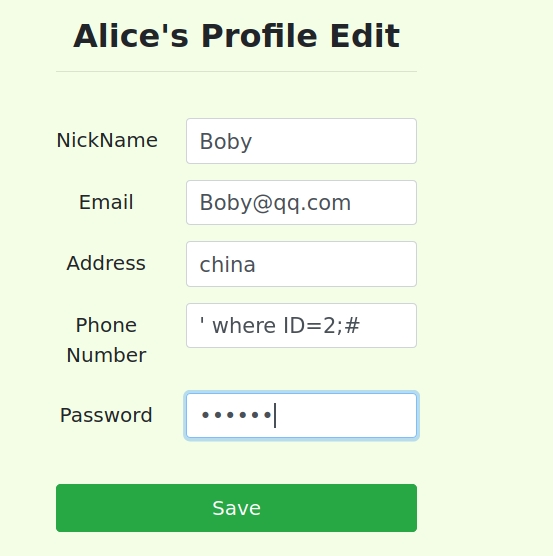
\*\*任务3.2\*\*:修改其他人的工资。



在Nickname处输入Boby’,Salary=1 where Name=”Boby”;#

\*\*任务3.3\*\*:修改他人密码。

修改操作如下：



在容器中检验发现更改密码前后的密码变化：





\*\*任务4\*\*：在这个任务中，我们将使用预处理语句机制来修复SQL注入漏洞。

如果你在任务4中还完善了其他代码，请一并说明。

// 准备SQL语句

$stmt = $conn->prepare("SELECT id, name, eid, salary, ssn

FROM credential

WHERE name= ? and Password= ?");

// 绑定参数

$stmt->bind\_param("ss", $input\_uname, $hashed\_pwd);

// 执行查询

$stmt->execute();

// 获取结果

$result = $stmt->get\_result();

if ($result->num\_rows > 0) {

// 只取第一行

$firstrow = $result->fetch\_assoc();

$id = $firstrow["id"];

$name = $firstrow["name"];

$eid = $firstrow["eid"];

$salary = $firstrow["salary"];

$ssn = $firstrow["ssn"];

}

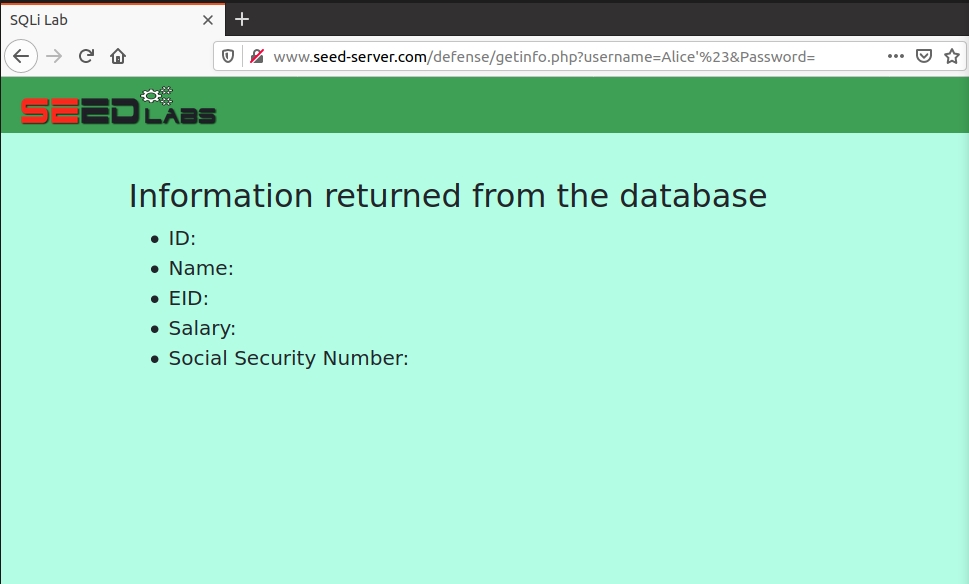
// 关闭语句和数据库连接

$stmt->close();

$conn->close();

如上代码所示，以上为更改替换的代码，使用了prepare预编译方法，将数据和代码分离，参考指导书的部分代码，注释如上

最终在网页测试sql注入情况如下：



成功防止用户通过不合理的sql注入获取信息

【选做】有余力的同学可以完善下unsafe\_home.php、unsafe\_edit\_frontend.php和unsafe\_edit\_backend.php代码文件，把相关的漏洞修复。

# 二、遇到问题及解决方法

问题：在修改其他用户的密码这一块耽误时间较久，当时还请教了老师但是每次在网站上依旧报错

解决方案：经过在mysql容器内测试最终发现其实用户密码已经更改，哈希值已经改变但是执行在网站确实会出现报错情况

# 三、对本次实验的建议

本次实验很好，感觉是特别有意思的一次实验，使用了多容器技术，也简单了实践了早有耳闻的sql注入技术，很有收获，希望之后指导书能够更新出我遇到问题的原因，让我进一步学习