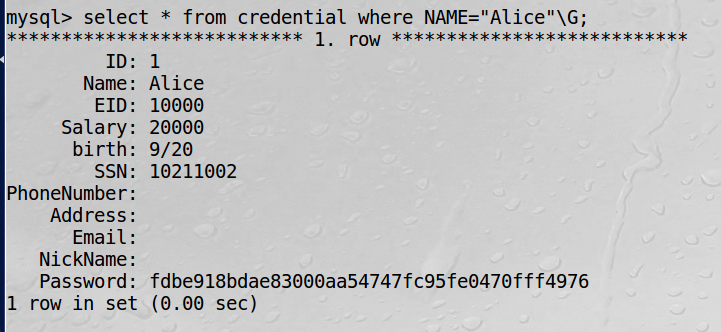
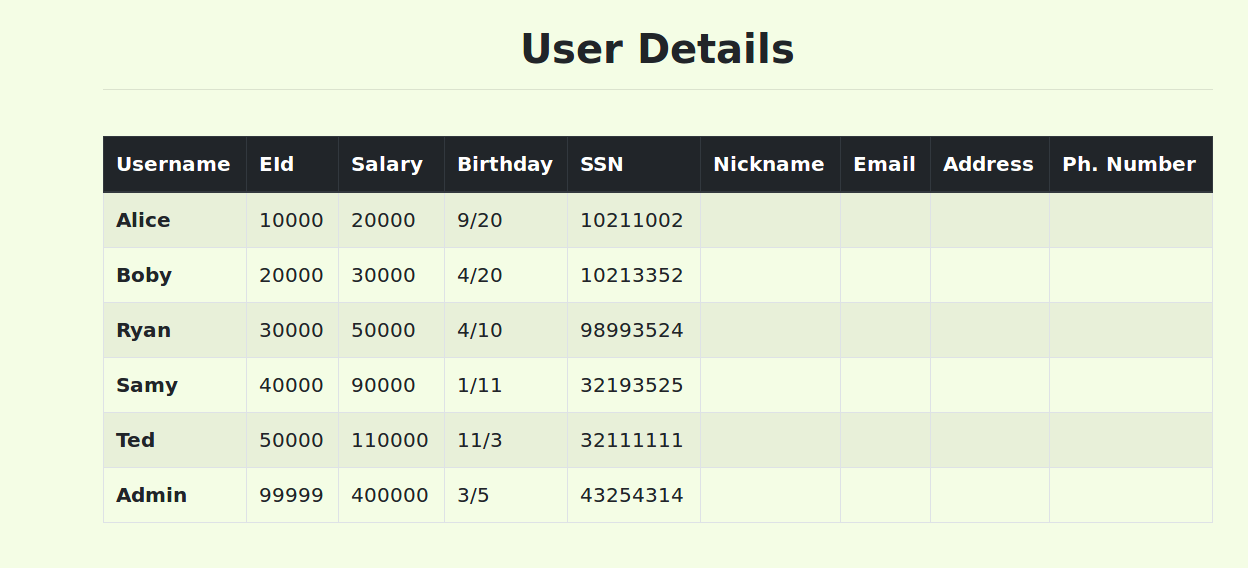
Task1：



Task2：

登录网址，在输入username一栏输入admin’#

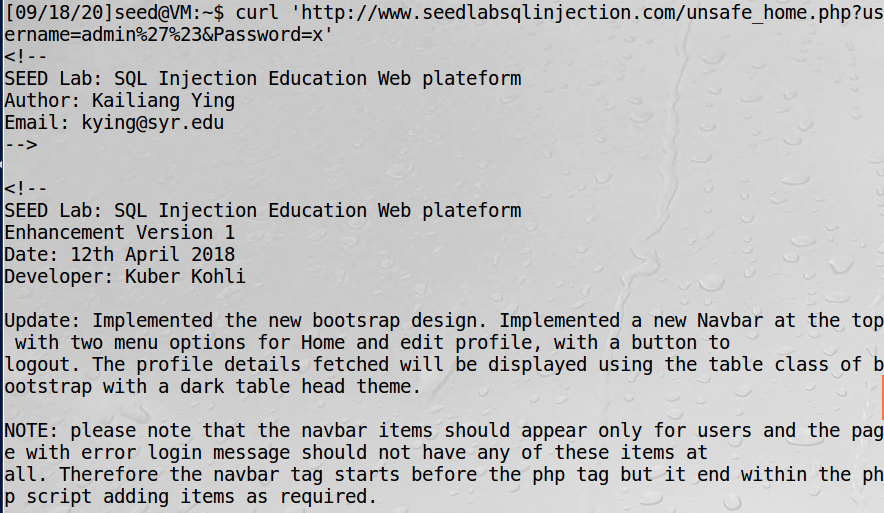
‘单引号是闭合标志，#号用于注释掉同行后面的内容，这样就相当于只根据用户名查询，登录admin用户，而php文件中定义登录admin后返回所用用户信息，如下



在terminal中输入

curl ‘http;//seedlabsqlinjection.com/unsafa\_home.php?username=admin%27%23Password=x’

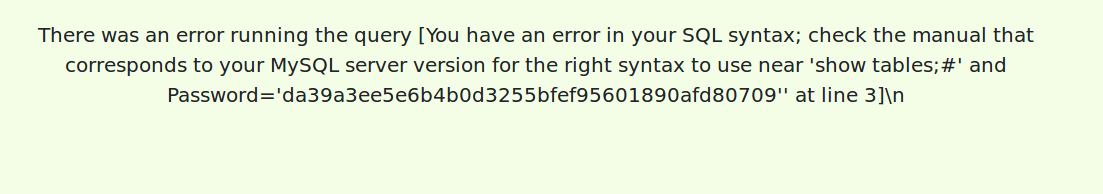
‘单引号和#号用转义符表示即可，得到网页内容如下





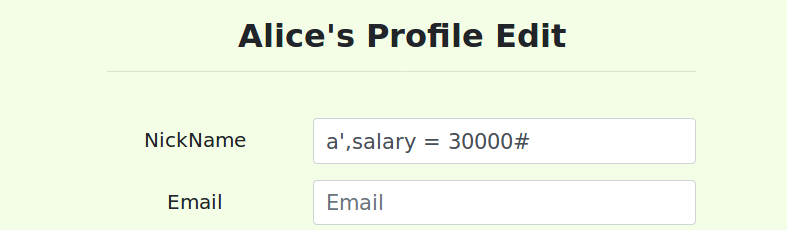
Task2.3:

构造select \* from credential where username=’admin’;show tables;#......攻击，发现报错如下，php query函数一次只允许传递一个sql语句。



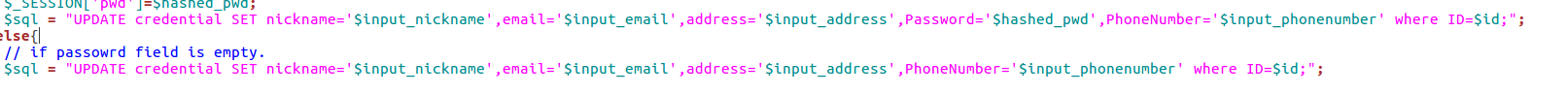
Task3

在随便一个空中填入即可修改，需要用#号去除同一行后面的’单引号



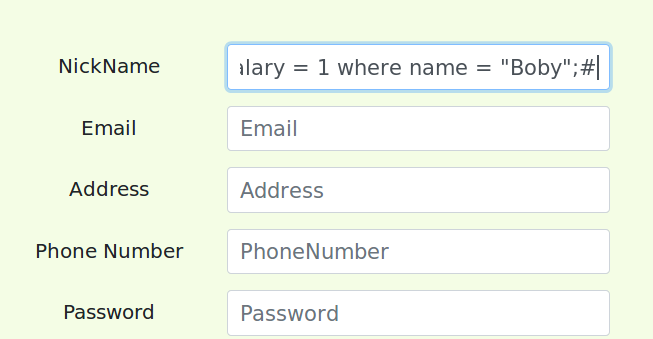


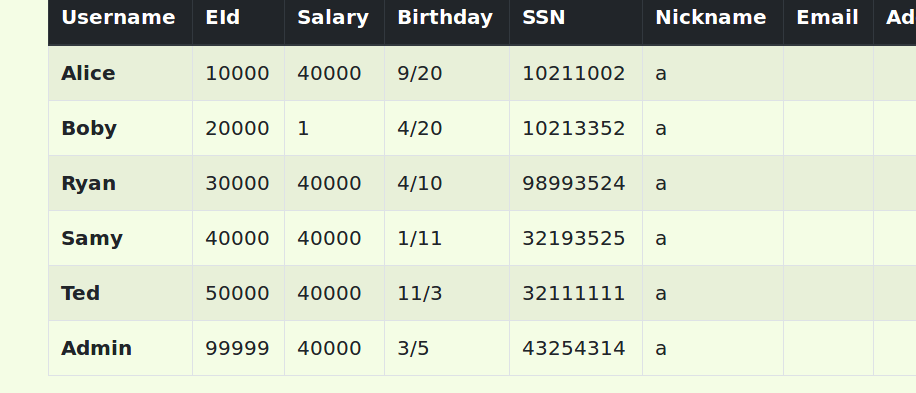
Task3.2:



确认php文件中update语句写在一行中

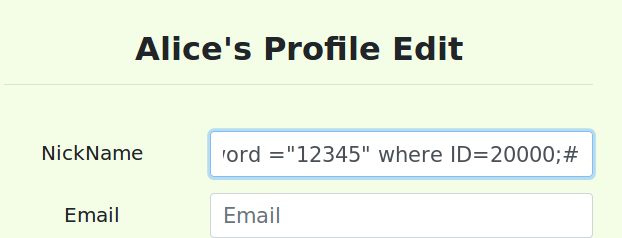
应用where字句指定name= “Boby”即可，最后要加上#注释掉后面的语句。





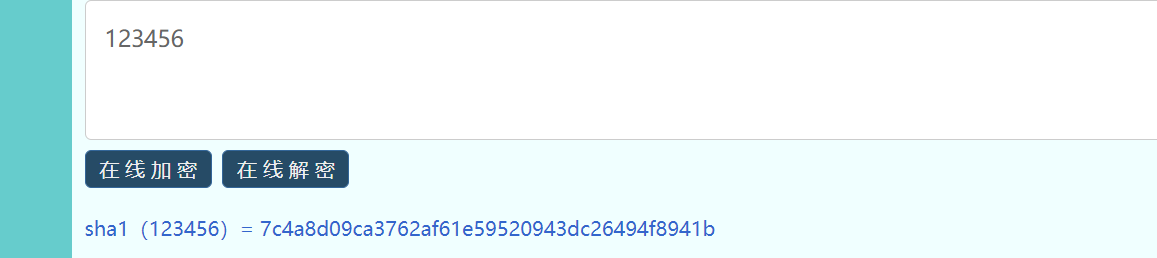
Task3.3：

第一次攻击，失败

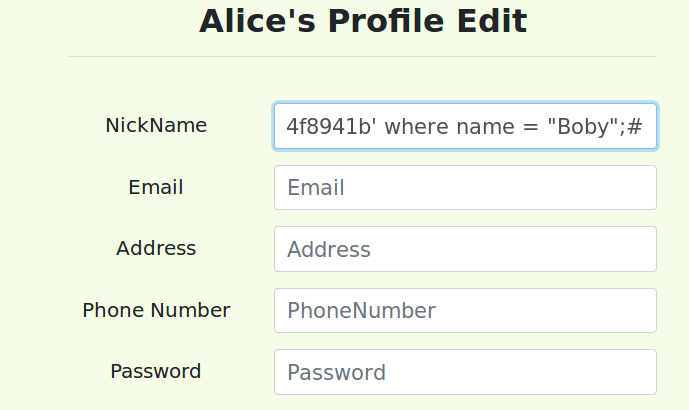


分析：原来修改密码，php文件调用sha1函数将密码转换为hash值，但登陆的时候会将明文转换为hash值验证，后来改sql语句的时候不能直接用明文改

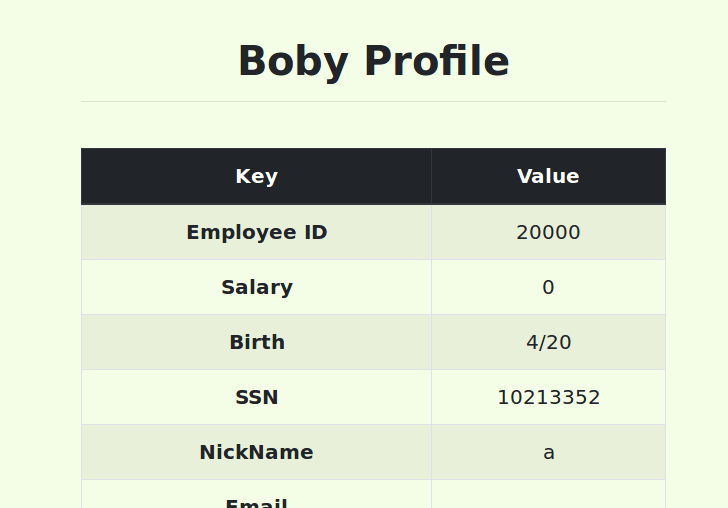
找到sha1转换器



实施攻击



然后登陆成功如下

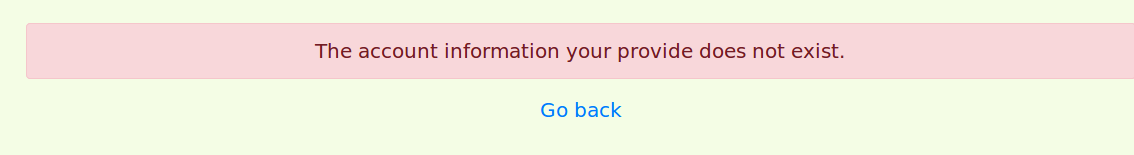


Task4：

将原来代码替换成safe\_home.php和safe\_edit backend.php

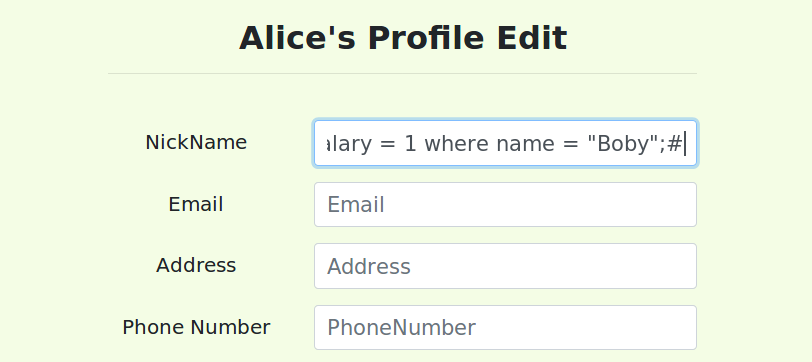
换成有预处理的sql语句，现将语句送过到服务器，再将数据说明数据类型送过去，保证了sql语句的安全

效果如下

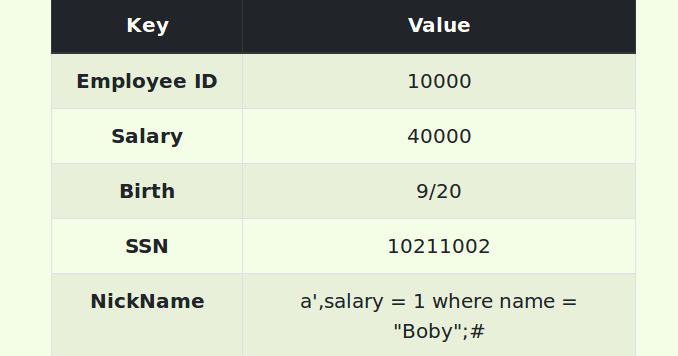


原来的闭合加注释的登陆方式失效

正常用密码登陆后



尝试发动攻击如上，效果如下



可以看到，输入并没有被当成代码执行，而是全部作为数据存储进nickname

证明了prepared statement 可以有效防止sql注入攻击。