注入攻击

注入攻击的本质是将用户输入的内容当做代码执行

两个关键点

1. 用户能够控制输入
2. 原本程序要执行的代码，拼接了用户的输入

错误回显会给攻击者极大的便利，比如攻击者在参数中输入一个单引号，引起查询语句的语法错误，服务器紧接着返回了错误信息，通过错误信息可以判断服务器所用数据库的类型，进而构造相应的攻击手段。

当web服务器关闭了错误回显，我们称之为盲注（Blind Injection）

那么攻击者需要找到一个方法验证SQL注入语句是否成功。

最简单的方法就是构造一个简单的条件语句，根据页面是否发生变化来判断是否存在漏洞

Example：

http:// [www.newspaper.com/items.php?](http://www.newspaper.com/items.php?) id = 2

SELECT title ， description FROM table WHERE id = 2

攻击者可以构造 id = 2 and 1=2;

因为1=2为矛盾式，所以该语句不会被执行，将会返回一个空的页面，或者什么都不返回；

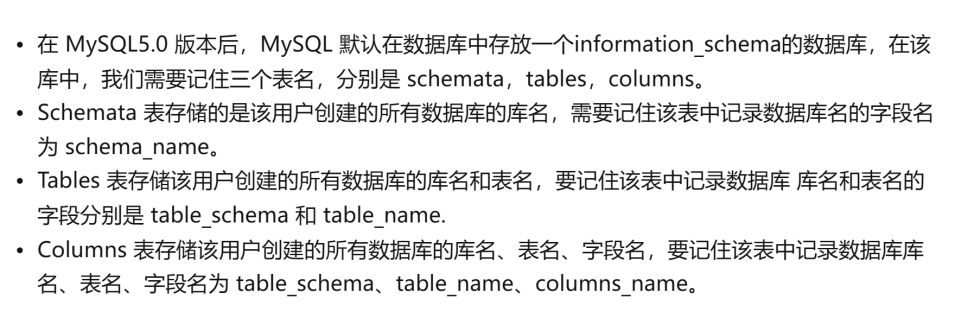
当构造id=2 and 1=1 时，成功返回信息，说明SQL注入成功。

另一种盲注的技巧

Timing attack

在MYSQL中有一个BENCHMARK（cout，expr）函数，它可以将expr函数执行cout次，返回执行的时间，这个函数是用来测试函数性能的。

攻击者通过时间长短的变化，可以就可以知道SQL语句是否注入成功。



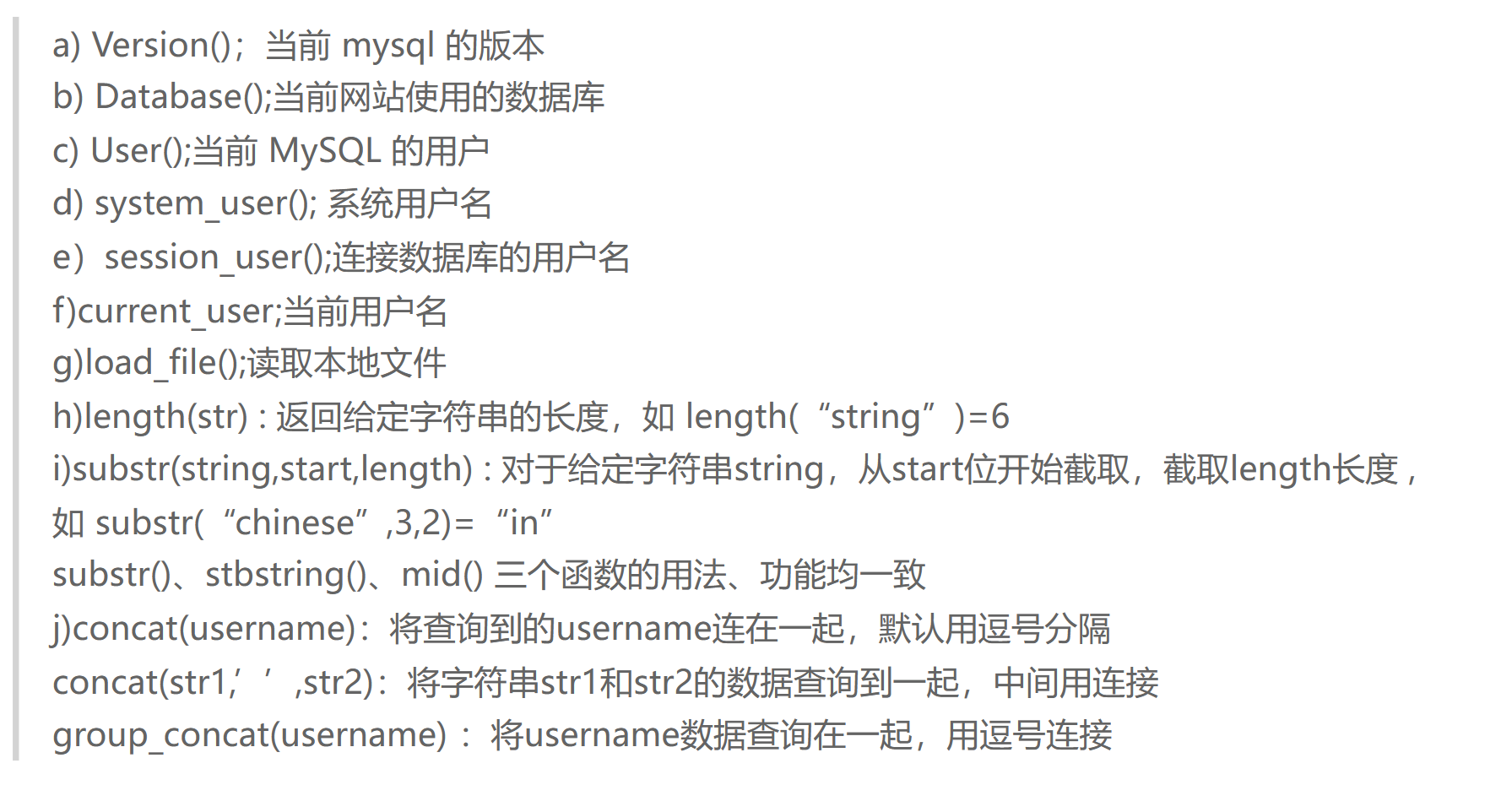
MYSQL 5.0版本后，默认在数据库中存放着一个information\_schema 的库

里面存放着有关数据库的信息，其中三个表名 schemata tables,columns

Schemata 表存放该用户创建的所有数据库的库名，字段名schema\_name

Table 表存放用户创建的所有数据库的库名和表名 字段名table\_schema table\_name

Columns 中存放着所有数据库的表名和表名和字段名 字段名 columns\_name



注释符号：

1. #
2. --+
3. /\*\*/

**SQL注入漏洞攻击流程 注入点探测--信息获取--权限获取**

SQL注入的类型介绍

按照注入点类型分类

1. 数字型注入点
2. 字符型注入点 ：我们需要将烦人的引号去掉
3. 搜索型注入点 ：select \* from 表名 where 字段 like ‘%关键字%

按照数据提交的方式来分类

1. GET 注入
2. POST 注入
3. Cookie 注入
4. HTTP 头部注入

按照执行效果来分类

1. 基于布尔的盲注 ：即可以根据返回页面判断条件真假的注入
2. 基于时间的注入 ：用条件查询语句检查时间延迟是否执行
3. 基于报错注入 ： 即页面会返回错误信息，错误信息中会包含一些隐私信息
4. 联合查询注入 ：可以使用union的情况下的注入
5. 堆查询注入：可以同时执行多条语句的执行时的注入
6. 宽字节注入：

可能有注入的点：

所有可能与后端进行交互的点都可能会产生注入漏洞

所有与后端交互的点都可以尝试