

1.3 数据库

1.3.0前言

在阅读本章之前，请各位先打开电脑，然后跟随文章的内容进行操作。在web的世界中，数据库可以说是web世界的核心。数据库存储着web世界中最核心的数据。现实生活中，许多hacker攻击网站的最终目的就是获取目标网站的数据库。一旦hacker攻击网站成功获取数据库，此时便是大家常说的“脱裤”。因本节讲解数据库的基础操作，不涉及漏洞。故CTF平台赛题的数目不会太多，还希望大家跟着教程动手练习。

1.3.1简介

1.3.1.1什么是数据库

数据库可以理解为用户的文件柜，用来存储用户的文件。在下图中我们可以看到数据库中存储的用户账户以及密码等信息。

user_id	first_name	last_name	user	password	avatar	last_login	failed_login
1	admin	admin	admin	5f4dcc3b5aa765d61d8327deb882cf99	/hackable/users/admin.jpg	2019-01-15 12:38:32	
2	Gordon	Brown	gordonb	e99a18c428cb38d5f260853678922e03	/hackable/users/gordonb.jpg	2019-01-15 12:38:24	
3	Hack	Me	1337	8d3533d75ae2c3966d7e0d4fcc69216b	/hackable/users/1337.jpg	2019-01-15 12:38:24	
4	Pablo	Picasso	pablo	0d107d09f5bbe40cade3de5c71e9e9b7	/hackable/users/pablo.jpg	2019-01-15 12:38:24	
5	Bob	Smith	smithy	5f4dcc3b5aa765d61d8327deb882cf99	/hackable/users/smithy.jpg	2019-01-15 12:38:24	

1.3.1.2为什么需要数据库

一个应用程序一般会有大量的用户信息，比如一个网站的注册信息。应用程序需要将这些信息存储起来，以方便下次继续编辑或者使用。简单的来说，可以把数据保存到一个excel表格里。但是随着网站功能的复杂，注册人数越来越多。此时在通过文件读取数据，不仅无法快速查询数据，而且需要大量的重复的代码。因此，数据库作为一个专门的数据库管理软件而出现。应用程序只需要调用数据库软件提供的接口来读写数据，至于数据的存储则完全有数据库软件负责。

1.3.1.3主流的数据库

在进行sql注入时，因每种数据库都有自己独特的特性。所以注入的方法略有不同。而我们则需要对这些数据库的特性都有所了解。在此，我们仅做简单的介绍，详细部分日后在详谈。

- 1.MYSQL:开源的。一般个人应用比较多。
- 2.SQL server:由微软开发的数据库管理系统，是Web上最流行的用于存储数据的数据库，它已广泛用于电子商务、银行、保险、电力等与数据库有关的行业。
- 3.oracle:Oracle产品覆盖了大、中、小型机等几十种机型，Oracle数据库成为世界上使用最广泛的关系数据系统之一。

1.3.2安装mysql

- window平台安装

分享一个链接：<https://blog.csdn.net/wyxeainn/article/details/75865434>

- ubuntu安装

ubuntu平台安装相对简单只需要输入几条命令即可。

- 1.sudo apt-get install mysql-server
- 2.配置mysql: sudo mysql_secure_installation
- 3.配置选项较多，除选择密码难度等级和设置密码的地方，其他默认回车即可。

```
There are three levels of password validation policy:
LOW      Length >= 8
MEDIUM  Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and digits
Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 0
```

```
Please set the password for root here.
New password:
Re-enter new password:
Estimated strength of the password: 50
Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No) : y
```

- 4.设置完成以后我们在终端输入 mysql -u root -p 然后回车输入刚刚设置的密码，然后再次回车。即可进入数据库。

```

Bye
root@xu-virtual-machine:/home/xu# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 5.7.25-0ubuntu0.18.04.2 (Ubuntu)

Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>

```

1.3.3基本概念

数据库有四层组成。大致关系如下：

- 1.最外层是数据库软件比如mysql.
- 2.一个数据库软件可以创建很多个数据库来供不同的应用程序使用。

```

-> ;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+

```

- 3.一个库中有很多的表。一般一个表便对应着一组有关系的数据。

```

mysql> show tables;
+-----+
| Tables_in_dwva |
+-----+
| guestbook |
| users |
+-----+
2 rows in set (0.00 sec)

```

- 4.一个表中有若干的字段，字段对应着数据库中存储的内容。

```

mysql> select * from users;
+-----+-----+-----+-----+-----+-----+-----+-----+
| user_id | first_name | last_name | user | password | avatar | last_login | failed_login |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | admin | admin | admin | 5f4dcc3b5aa765d61d8327deb882cf99 | /hackable/users/admin.jpg | 2019-01-15 12:38:32 | 0 |
| 2 | Gordon | Brown | gordonb | e99a18c428cb38d5f260853678922e03 | /hackable/users/gordonb.jpg | 2019-01-15 12:38:24 | 0 |
| 3 | Hack | Me | 1337 | 8d3533d75ae2c3966d7e0d4fcc69216b | /hackable/users/1337.jpg | 2019-01-15 12:38:24 | 0 |
| 4 | Pablo | Picasso | pablo | 0d107d09f5bbe40cade3de5c71e9e9b7 | /hackable/users/pablo.jpg | 2019-01-15 12:38:24 | 0 |
| 5 | Bob | Smith | smithy | 5f4dcc3b5aa765d61d8327deb882cf99 | /hackable/users/smithy.jpg | 2019-01-15 12:38:24 | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

1处是字段，2处是存储的数据。

1.3.4数据库的基本操作

- 1.查看所有的数据库

show databases;

```

mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| challenges |
| dvwa |
| mysql |
| performance_schema |
| security |
+-----+

```

- 2.创建一个新的数据库

create database <数据库名称>;

```

mysql> create database test;
Query OK, 1 row affected (0.00 sec)

```

3.删除数据库

```
drop database <数据库名称>;

mysql> drop database test; 执行以上删除操作
Query OK, 0 rows affected (0.00 sec)
```

4 . 进入一个数据库

```
use <数据库名称> ;

mysql> create database test;
Query OK, 1 row affected (0.00 sec)

mysql> use test
Database changed
```

1.3.5 对表的操作

1.查看某个数据库中的所有表

```
show tables;

mysql> show tables;
+-----+
| Tables_in_dwva |
+-----+
| guestbook      |
| users          |
+-----+
```

2.创建一个表

创建一个表准备需要有三个东西

- 1.表名
- 2 . 表字段名
- 3 . 定义字段的类型

比如要创建一个存储姓名和年龄的表

```
create table student (name char(100),age int(10));

其中student 是表名，name,age是字段名。char是定义name为字符型，int定义age为整型。
```

```
mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| student        |
+-----+
```

3 . 查询表的全部内容

语句为：select * from table_name

```
Database Changed
mysql> show tables;
+-----+
| Tables_in_dwva |
+-----+
| guestbook      |
| users          |
+-----+
2 rows in set (0.00 sec)

mysql> select * from users;
+-----+-----+-----+-----+-----+-----+-----+-----+
| user_id | first_name | last_name | user      | password                                     | avatar                                     | last_login      | failed_login |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | admin    | admin    | admin    | 5f4dcc3b5aa765d61d8327deb882cf99          | /hackable/users/admin.jpg               | 2019-01-15 12:38:32 | 0 |
| 2 | Gordon   | Brown    | gordonb   | e99a18c428cb38d5f260853678922e03          | /hackable/users/gordonb.jpg             | 2019-01-15 12:38:24 | 0 |
| 3 | Hack     | Me       | 1337     | 8d3533d75ae2c3966d7e0d4fcc69216b          | /hackable/users/1337.jpg                | 2019-01-15 12:38:24 | 0 |
| 4 | Pablo    | Picasso  | pablo     | 0d107d09f5bbe40cade3de5c71e9e9b7          | /hackable/users/pablo.jpg               | 2019-01-15 12:38:24 | 0 |
| 5 | Bob      | Smith    | smithy    | 5f4dcc3b5aa765d61d8327deb882cf99          | /hackable/users/smithy.jpg              | 2019-01-15 12:38:24 | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

4.删除一个表

语句为：drop table table_name;

比如我们删除我们刚刚创建的student表。

```
drop table student;

Database changed
mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| student        |
+-----+
1 row in set (0.00 sec)

mysql> drop table student ;
Query OK, 0 rows affected (0.02 sec)

mysql> show tables;
Empty set (0.00 sec)
```

1.3.5增删改查

对于数据的操作，无非就是四种。增加、修改、删除、查询。

1.3.5.1 增加数据

通用语法为：insert into tables (字段一，字段二) values (值1，值2)

```
mysql> insert into student (name,age) values ("zhang",111);
Query OK, 1 row affected (0.01 sec)

mysql> select * from student ;
+-----+-----+
| name | age |
+-----+-----+
| zhang | 111 |
+-----+-----+
1 row in set (0.00 sec)
```

1.3.5.2 修改数据

UPDATE table_name SET 字段名 = 新值 where 筛选条件。

where 语句用于在字段中筛选内容，比如我们刚刚插入的数据中年龄输入错误，我们要想修改年龄，就必须先定位到哪一行，然后直接修改年龄字段。如何定位到哪一行就是where语句的工作。

比如我们要修改我们刚刚插入的数据。

修改语句如下：update student set age=12 where name = "zhang";

```
mysql> update student set age=12 where name= "zhang";
Query OK, 1 row affected (0.00 sec)

Rows matched: 1  Changed: 1  Warnings: 0 (0.02 sec)

mysql> select * from student ;
+-----+-----+
| name | age |
+-----+-----+
| zhang | 12 |
+-----+-----+
1 row in set (0.00 sec)
```

1.3.5.3 查询数据

select 字段名 1、字段名 2 from 表名 (where 筛选条件) (limit 数量)；

注意事项:

- 可以使用 * 来代替字段名此时查询数据库的全部字段。
- 可以使用where来筛选 (也可用不用)
- limit属性用来限制返回的结果的数量,可以不用。

查询年龄为12岁的学生的姓名：

select name from student where age=12；

```
mysql> select * from student ;
+-----+-----+
| name | age |
+-----+-----+
| zhang | 12 |
+-----+-----+
1 row in set (0.00 sec)

mysql> select name from student where age=12 ;
+-----+
| name |
+-----+
| zhang |
+-----+
1 row in set (0.00 sec)
```

1.3.5.4 删除数据

语句：delete from table_name where 筛选条件

此时注意：如果没有指定where语句，那么将会删除表中的所有记录。

我们删除刚刚修改的那条数据。

```
mysql> select * from student ;
+-----+-----+
| name | age |
+-----+-----+
| zhang | 12 |
+-----+-----+
1 row in set (0.00 sec)

mysql> delect from student where age=12;
ERROR 1064 (42000): You have an error in your SQL syntax
to use near 'delect from student where age=12' at line 1
mysql> delete from student where age=12;
Query OK, 1 row affected (0.01 sec)

mysql> select * from student ;
Empty set (0.00 sec)

mysql> |
```

1.3.6 连接数据库

进入一个数据库，或者远程连接一个数据库（前提是数据库允许远程连接）。命令如下：

- mysql -h ip地址 -P 端口 -u 用户名 -p （回车输入密码即可）
- 默认端口是 3306,如果是默认，可以加 - P 参数。

1.3.7 CTF实战

当flag的位置在数据库里面时，一般会藏在用户名密码的表中，或者有的表名或者库名会有提示字样。



1.3.8 课后习题

请于CTF平台完成本章节习题，并与星球内提交WP.

- 1 . [mysql基础](#)

小结

CTF（web和内网渗透系列教程）的清单请在“<https://github.com/xuanhun/HackingResource>” 查看，定时更新最新章节链接

答疑、辅导请加入玄魂工作室--安全圈，一起成长探讨更私密内容。微信扫码了解详情：



星主：程序员－玄魂

星球：玄魂工作室-安全圈



知识星球

长按扫码预览社群内容
和星主关系更进一步

及时获取更多消息，请关注微信订阅号



玄魂工作室