Section 11 Database Permissions

As we have mentioned before there are a several types of security to discuss associated with database access in web apps. We started with user authentication. Validating that the user is registered in the application and allowing them to access only content specific to the user.

At this point you should be aware that all users and all transactions happening in the application are passed to and from the database under one single application user. We even simplified it by creating the config.php file so we only need to enter and maintain the credentials in one single place.

A rule to live by for web apps (well mostly): Give the absolute minimum amount of rights to a database user in order to get the job done.

Almost all SQL environments allow for a variety of security measures. Oracle/SQL Server/DB2/MySql all allow for different database users to have specific permissions. As we explore and work with these concepts in the wild, you will find that your specific hosting environment may dictate what you can and cannot do (this is very important).

First let's explore c9.io, the we will discuss hosting:

Open MySQL on c9, Use the CRUD db;

select host, user, password from mysql.user;

```
mysql> select host, user, password from mysql.user;
+-------------+------------------+-------------------------------------------+
| host        | user             | password                                  |
+-------------+------------------+-------------------------------------------+
| localhost   | root             |                                           |
| e7128b67d9e7 | root            |                                           |
| 127.0.0.1   | root             |                                           |
| ::1         | root             |                                           |
| localhost   | debian-sys-maint | *A4E863B86086C01837C43D9735E66438BDA36F1D |
| %           | markrfoote       |                                           |
| %           | formcollectorApp |                                           |
| localhost   | formcollector    | *4F2FA325D269406FD00C17B50829856E3DB586CA |
+-------------+------------------+-------------------------------------------+
8 rows in set (0.00 sec)
```

Create a user for your CRUD Application:
CREATE USER 'CRUD_application'@'localhost' IDENTIFIED BY 'ReallyComplicatedPassword';

```
mysql> source query.sql
Query OK, 0 rows affected (0.00 sec)

mysql> select host, user, password from mysql.user;
+-------------+------------------+-------------------------------------------+
| host        | user             | password                                  |
+-------------+------------------+-------------------------------------------+
| localhost   | root             |                                           |
| e7128b67d9e7 | root            |                                           |
| 127.0.0.1   | root             |                                           |
| ::1         | root             |                                           |
| localhost   | debian-sys-maint | *A4E863B86086C01837C43D9735E66438BDA36F1D |
| %           | markrfoote       |                                           |
| %           | formcollectorApp |                                           |
| localhost   | formcollector    | *4F2FA325D269406FD00C17B50829856E3DB586CA |
| localhost   | CRUD_application | *F181326BE41435CC6B2D1D723AA27456CFF37341 |
+-------------+------------------+-------------------------------------------+
9 rows in set (0.00 sec)
```

At this point, the user exits but cannot do anything. They can't even log in. We need to grant permissions;

```
GRANT ALL PRIVILEGES ON * . * TO 'CRUD_application'@'localhost';
```

```
mysql> GRANT ALL PRIVILEGES ON * . * TO 'CRUD_application'@'localhost';
Query OK, 0 rows affected (0.00 sec)
```

This command gives full access to all databases and all tables for the user.
If we were going to only give access to the CRUD database, is should look like this:

```
mysql> GRANT ALL PRIVILEGES ON CRUD . * TO 'CRUD_application'@'localhost';
Query OK, 0 rows affected (0.00 sec)
```

After you are done make changes to permissions, run:

```
FLUSH PRIVILEGES;
```

This will refresh the permissions.

Now check that your new user worked:

```
QUIT
```

Then log in:

```
mysql -u CRUD_application -p
ReallyComplicatedPassword
```

```
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.02 sec)

mysql> quit
Bye
markrfoote:~/workspace $ mysql -u CRUD_application -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 49
Server version: 5.5.44-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2015, 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>
```

You are now logged in as the application user with full privileges to do anything.
So far, our development has always been under the root account without a password.

In production environments, the administrator account is typically referred to as the ‘SA’ account for “System Administrator” and only held by the database administrators.
We have been in a development environment so having passwords and security are less important to implement. In any production environment, having security fully implemented is critical. As a developer, typically you won’t have or need the root or SA credentials, but you will be given full access to your database which is all you need.

You can check the privileges of a user using Show Grants

SHOW GRANTS FOR CURRENT_USER;
SHOW GRANTS; *-same as above.*
SHOW GRANTS FOR CRUD_application@localhost;

```
mysql> SHOW GRANTS FOR CURRENT_USER;
+-----------------------------------------------------------------------------------------------------------------------------------+
| Grants for CRUD_application@localhost                                                                                             |
+-----------------------------------------------------------------------------------------------------------------------------------+
| GRANT ALL PRIVILEGES ON *.* TO 'CRUD_application'@'localhost' IDENTIFIED BY PASSWORD '*F181326BE41435CC6B2D1D723AA27456CFF37341' |
+-----------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)
```

```
mysql> show Grants for CRUD_application@localhost;
+-----------------------------------------------------------------------------------------------------------------------------------+
| Grants for CRUD_application@localhost                                                                                             |
+-----------------------------------------------------------------------------------------------------------------------------------+
| GRANT ALL PRIVILEGES ON *.* TO 'CRUD_application'@'localhost' IDENTIFIED BY PASSWORD '*F181326BE41435CC6B2D1D723AA27456CFF37341' |
+-----------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)
```

```
mysql> SHOW GRANTS;
+-----------------------------------------------------------------------------------------------------------------------------------------------+
| Grants for CRUD_application@localhost                                                                                                         |
+-----------------------------------------------------------------------------------------------------------------------------------------------+
| GRANT ALL PRIVILEGES ON *.* TO 'CRUD_application'@'localhost' IDENTIFIED BY PASSWORD '*F181326BE41435CC6B2D1D723AA27456CFF37341' |
+-----------------------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)
```

```
mysql> show grants for current_user;
+-----------------------------------------------------------------------+
| Grants for markrfoote@%                                               |
+-----------------------------------------------------------------------+
| GRANT ALL PRIVILEGES ON *.* TO 'markrfoote'@'%' WITH GRANT OPTION |
+-----------------------------------------------------------------------+
1 row in set (0.00 sec)
```

Add another user to test:
CREATE USER 'project1_app'@'localhost' IDENTIFIED BY 'ComplicatedPassword';
GRANT SELECT ON project1 . * TO 'project1_app'@'localhost';
SHOW GRANTS FOR project1_app@localhost;

```
mysql> SHOW GRANTS FOR project1_app@localhost;
+-----------------------------------------------------------------------------------------------------------------------------------------------+
| Grants for project1_app@localhost                                                                                                             |
+-----------------------------------------------------------------------------------------------------------------------------------------------+
| GRANT USAGE ON *.* TO 'project1_app'@'localhost' IDENTIFIED BY PASSWORD '*05C9D1567594310F5A49336F1E474C352168DBD5' |
| GRANT SELECT ON `project1`.* TO 'project1_app'@'localhost'                                                                                    |
+-----------------------------------------------------------------------------------------------------------------------------------------------+
2 rows in set (0.00 sec)
```

Not let's test these limited rights:
In a new terminal window, type:
`mysql -u project1_app -p`
Then "`ComplicatedPas sword`"

```
markrfoote://home/ubuntu/workspace $ mysql -u project1_app -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 56
Server version: 5.5.44-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2015, 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>
```

```
SHOW DATABASES;
```

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| project1           |
+--------------------+
2 rows in set (0.00 sec)
```

User can't see the database basis that they don't have rights to.

```
mysql> use project1
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
```

SHOW TABLES;

```
Database changed
mysql> show tables
    -> ;
+--------------------+
| Tables_in_project1 |
+--------------------+
| CustBal            |
| Friends            |
| Sec2               |
| bowlResults        |
| bowlers            |
| characters         |
| movies             |
| people             |
| person             |
+--------------------+
9 rows in set (0.00 sec)
```

SELECT * FROM Friends;
DELETE FROM Friends WHERE FriendID = 4;

```
mysql> SELECT * FROM Friends;
+----------+-----------+----------+------------+
| FriendID | FNAME     | LNAME    | dob        |
+----------+-----------+----------+------------+
|        1 | Mashrur   | Hossain  | 1981-12-25 |
|        2 | Matt      | Berstein | 1980-08-05 |
|        3 | Anastasia | Ivanov   | 1989-04-01 |
|        4 | Mark      | Futre    | 1989-07-04 |
+----------+-----------+----------+------------+
4 rows in set (0.00 sec)

mysql> DELETE FROM FRIENDS WHERE FriendID = 4;
ERROR 1142 (42000): DELETE command denied to user 'project1_app'@'localhost' for table 'FRIENDS'
mysql>
```

The DELETE action was denied because the user doesn't have rights to do anything other than select.

Generally your applications will need to SELECT, INSERT, UPDATE,and DELETE rows on a table. In most cases, the applications won't need to CREATE, ALTER, DROP tables and databases.

Here are a list of basic GRANT types.:

```
ALL PRIVILEGES - everything

CREATE - create tables and databases
DROP -  tables and databases

SELECT - table rows
INSERT - table rows
UPDATE - table rows
DELETE - table rows

GRANT OPTION - give and take away other users' rights
```

Lets update CRUD_application with appropriate rights.
Start by removing the GRANT ALL PRIVILEGES. Then add permissions for rows only.

Under root, do the following:

```
SHOW GRANTS FOR CRUD_application@localhost;
```

```
mysql> SHOW GRANTS FOR CRUD_application@localhost;
+-----------------------------------------------------------------------------------------------------------------------------------------------+
| Grants for CRUD_application@localhost                                                                                                         |
+-----------------------------------------------------------------------------------------------------------------------------------------------+
| GRANT ALL PRIVILEGES ON *.* TO 'CRUD_application'@'localhost' IDENTIFIED BY PASSWORD '*F181326BE41435CC6B2D1D723AA27456CFF37341' |
+-----------------------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)
```

We can giveth, and we can taketh away:

```
REVOKE ALL PRIVILEGES ON *.* FROM 'CRUD_application'@'localhost';
```

Revoke is the opposite of GRANT

```
mysql> REVOKE ALL PRIVILEGES ON *.* FROM 'CRUD_application'@'localhost';
Query OK, 0 rows affected (0.00 sec)
```

SHOW GRANTS FOR CRUD_application@localhost;

```
mysql> SHOW GRANTS FOR CRUD_application@localhost;
+-----------------------------------------------------------------------------------------------------------------------------------------------+
| Grants for CRUD_application@localhost                                                                                                         |
+-----------------------------------------------------------------------------------------------------------------------------------------------+
| GRANT USAGE ON *.* TO 'CRUD_application'@'localhost' IDENTIFIED BY PASSWORD '*F181326BE41435CC6B2D1D723AA27456CFF37341' |
+-----------------------------------------------------------------------------------------------------------------------------------------------+
1 row in set (0.00 sec)
```

Now let's add only what the application needs:

```
GRANT SELECT, INSERT, UPDATE, DELETE ON CRUD . * TO
'CRUD_application'@'localhost';
SHOW GRANTS FOR CRUD_application@localhost;
```

```
mysql> GRANT SELECT, INSERT, UPDATE, DELETE ON CRUD . * TO 'CRUD_application'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql> SHOW GRANTS FOR CRUD_application@localhost;
+-----------------------------------------------------------------------------------------------------------------------------------+
| Grants for CRUD_application@localhost                                                                                             |
+-----------------------------------------------------------------------------------------------------------------------------------+
| GRANT USAGE ON *.* TO 'CRUD_application'@'localhost' IDENTIFIED BY PASSWORD '*F181326BE41435CC6B2D1D723AA27456CFF37341' |
| GRANT SELECT, INSERT, UPDATE, DELETE ON `CRUD`.* TO 'CRUD_application'@'localhost'                                                |
+-----------------------------------------------------------------------------------------------------------------------------------+
2 rows in set (0.00 sec)
```

Test it out through the command line in a new window:

```
mysql -u CRUD_application -p
ReallyComplicatedPassword
```

```
markrfoote:~/workspace $ mysql -u CRUD_application -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 62
Server version: 5.5.44-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2015, 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>
```

Try to create a table:

```
CREATE TABLE people ( PersonID int
                     ,first_name varchar(100)
                     ,last_name  varchar(100)
);
```

```
mysql> CREATE TABLE people ( PersonID int
    ->                      ,first_name varchar(100)
    ->                      ,last_name  varchar(100)
    -> );
ERROR 1142 (42000): CREATE command denied to user 'CRUD_application'@'localhost' for table 'people'
```

Try to update a row:

```
SELECT * FROM ToDos;
```

```
mysql> SELECT * FROM ToDos;
+--------+---------+----------------------+----------------------------------------+----------
----------------+------------+
| ToDoID | User_ID | ToDoTitle            | ToDoDescription                        | Complete
ateTS           | CompleteTS |
+--------+---------+----------------------+----------------------------------------+----------
----------------+------------+
|     22 |    NULL | CRUD Security Chapter | Write security chapter for CRUD Project |     NULL
                | NULL       |
```

```
UPDATE ToDos
SET ToDoDescription = 'Test Update Permissions'
WHERE ToDoID = 22;
```

```
mysql> UPDATE ToDos
    -> SET ToDoDescription = 'Test Update Permissions'
    -> WHERE ToDoID = 22;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> SELECT * FROM ToDos;
+--------+---------+----------------------+------------------------+----------+-
-+------------+
| ToDoID | User_ID | ToDoTitle            | ToDoDescription        | Complete |
 | CompleteTS |
+--------+---------+----------------------+------------------------+----------+-
-+------------+
|     22 |    NULL | CRUD Security Chapter | Test Update Permissions |     NULL |
 | NULL       |
```

Now let's test the full application:
Update the CRUD application config.php

query.sql | config.php

```php
<?php
/*Configuration Settings*/

define('DB_HOST', 'localhost'); /*Database Server*/
define('DB_NAME', 'CRUD'); /*Database Name*/
define('DB_USER', 'root'); /*Database Username*/
define('DB_PWD',  ''); /*Database Password*/

?>
```

query.sql | config.php

```php
<?php
/*Configuration Settings*/

define('DB_HOST', 'localhost'); /*Database Server*/
define('DB_NAME', 'CRUD'); /*Database Name*/
define('DB_USER', 'CRUD_application'); /*Database Username*/
define('DB_PWD',  'ReallyComplicatedPassword'); /*Database Password*/

?>
```

Now test out your application.

To-do Main View

New To-do

Connected successfully

Title	Description	DueDate	Action
Test Link	testing new link help	04-03-2016	Update Delete
test title	Test description	06-16-2016	Update Delete
asdfasd	asdfsdaf	06-21-2016	Update Delete

You should be able to login, insert, update, delete like normal.
Congratulations! You know have an application specific login.

We can also Drop users all together:
DROP USER project1_app@localhost;

```
mysql> DROP USER 'CRUD_application'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql> select host, user, password from mysql.user;
+--------------+------------------+-------------------------------------------+
| host         | user             | password                                  |
+--------------+------------------+-------------------------------------------+
| localhost    | root             |                                           |
| e7128b67d9e7 | root             |                                           |
| 127.0.0.1    | root             |                                           |
| ::1          | root             |                                           |
| localhost    | debian-sys-maint | *A4E863B86086C01837C43D9735E66438BDA36F1D |
| %            | markrfoote       |                                           |
| %            | formcollectorApp |                                           |
| localhost    | formcollector    | *4F2FA325D269406FD00C17B50829856E3DB586CA |
+--------------+------------------+-------------------------------------------+
8 rows in set (0.00 sec)
```

Challenge:
Create an application specific login for your LeadCollectorApp. Only give it the minimum rights required.