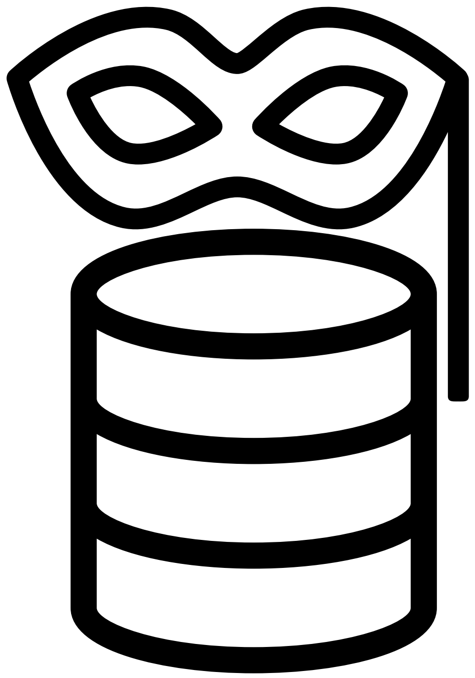
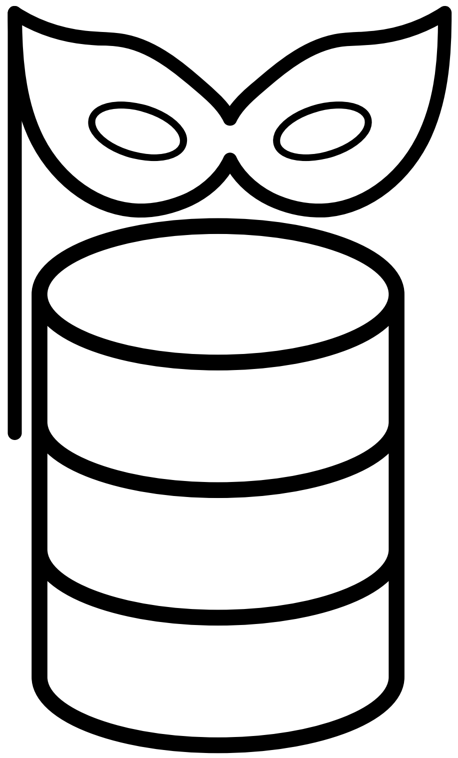
A quick tour of MySQL 8.0 roles





Giuseppe Maxia

Software explorer

#fosdem
#mysqldevroom

About me

Who's this guy?

- Giuseppe Maxia, a.k.a. "The Data Charmer"
- QA Architect at VMware
- Several decades development and DB experience
- Long timer MySQL community member.
- Blog: http://datacharmer.blogspot.com
- Twitter: @datacharmer

#fosdem
#mysqldevroom

Roles overview

A long coveted feature finally arrives

- Available since MySQL 8.0.0
- Created like an user
- Granted like privileges
- Need to be activated (with tricks)

Before roles

Up until the current GA (MySQL 5.7) there were no roles

In short: a lot of work,
with many chances to make mistakes

- **CREATE USER**
- **GRANT, GRANT, and more granular GRANT**
- **CREATE USER**
- **GRANT, GRANT again, and then GRANT**
- **CREATE USER**
- **GRANT, GRANT, GRANT, GRANT, oops!**

Advantages of roles

Why bother with this new feature?

- **Faster user administration**
  - define a role once
  - assign it many times
- **Centralised grants handling**
  - grant and revoke privileges to roles
  - No need to edit all users profiles
- **Easy to understand grants statistics**

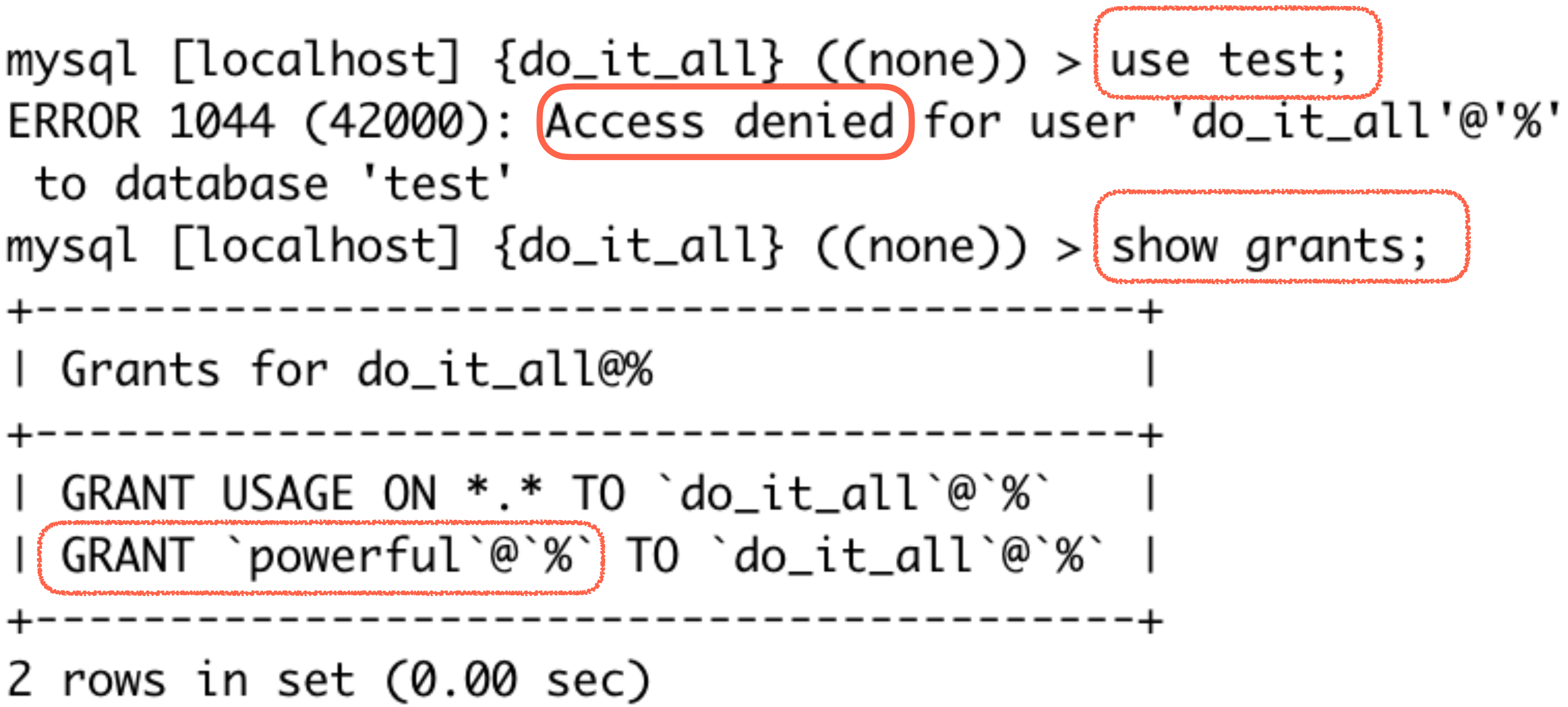
A BAD example. (1)

```
mysql> create role powerful;
Query OK, 0 rows affected (0.00 sec)

mysql> grant all on *.* to powerful;
Query OK, 0 rows affected (0.00 sec)
```

So far, so good

```
mysql> create user do_it_all
    -> identified by 'msandbox';
Query OK, 0 rows affected (0.00 sec)

mysql> grant powerful to do_it_all;
Query OK, 0 rows affected (0.00 sec)
```

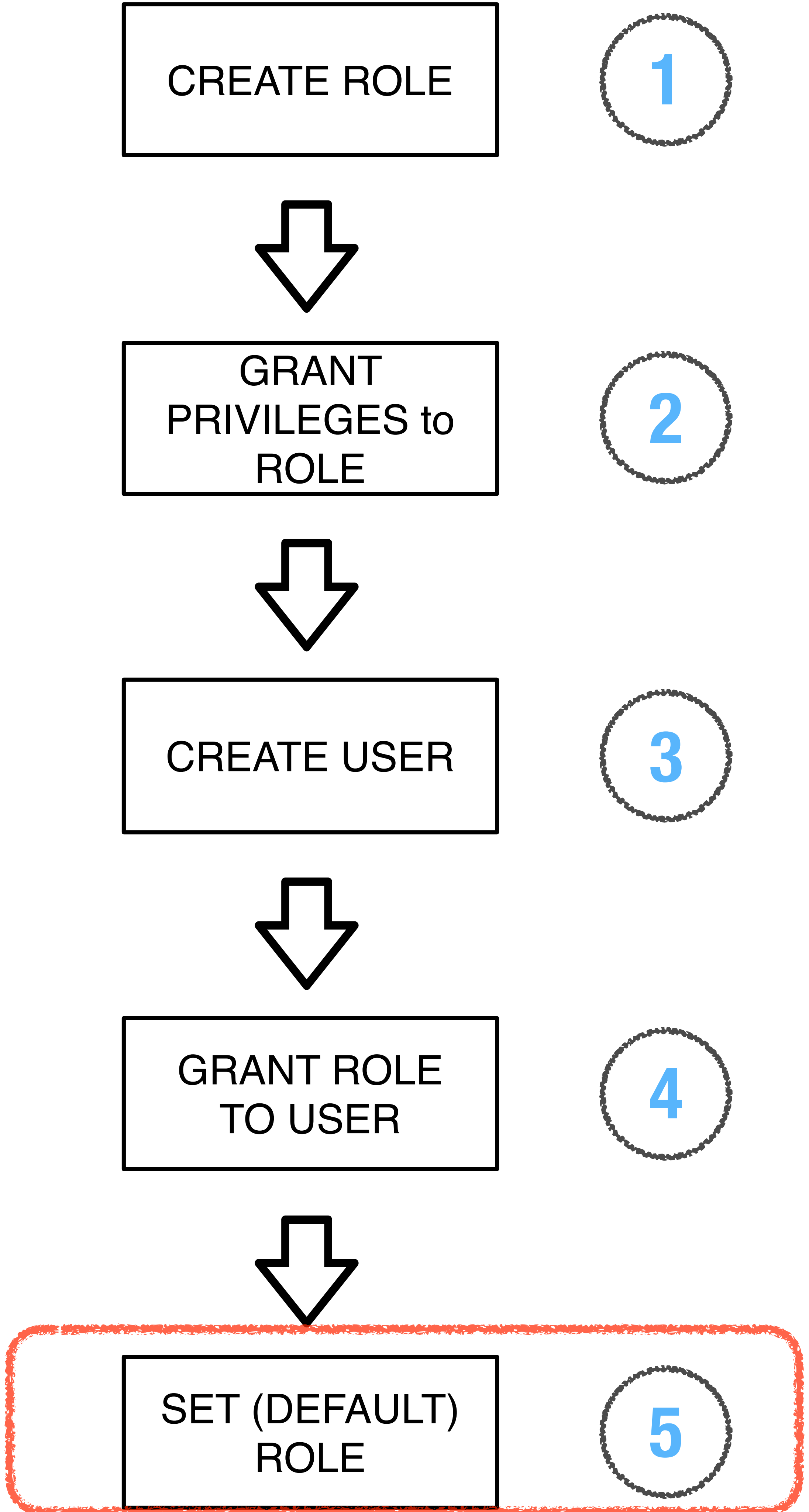
A BAD example. (2)



```
mysql [localhost] {do_it_all} ((none)) > use test;
ERROR 1044 (42000): Access denied for user 'do_it_all'@'%'
 to database 'test'
mysql [localhost] {do_it_all} ((none)) > show grants;
+--------------------------------------+
| Grants for do_it_all@%               |
+--------------------------------------+
| GRANT USAGE ON *.* TO `do_it_all`@`%`   |
| GRANT `powerful`@`%` TO `do_it_all`@`%` |
+--------------------------------------+
2 rows in set (0.00 sec)
```

WHAT DID JUST HAPPEN ? STAY TUNED TO FIND OUT

Roles usage


CREATE ROLE
1
GRANT PRIVILEGES to ROLE
2
CREATE USER
3
GRANT ROLE TO USER
4
SET (DEFAULT) ROLE
5

Create role

Like creating a user

```
CREATE ROLE r_lotr_dev;
```

```
## NOTE: there is no "IDENTIFIED" clause
```

grant privileges to role

Same as we do it with users

```
GRANT ALL ON lotr.* TO r_lotr_dev;
```

Create user

This one is already known

```
CREATE USER aragorn IDENTIFIED BY
'lotrpwd';
```

Grant role to user

We grant a role in a way similar to granting a privilege

```
GRANT r_lotr_dev TO aragorn;

## NOTE: there is not an "ON" clause
## in the GRANT statement.
```

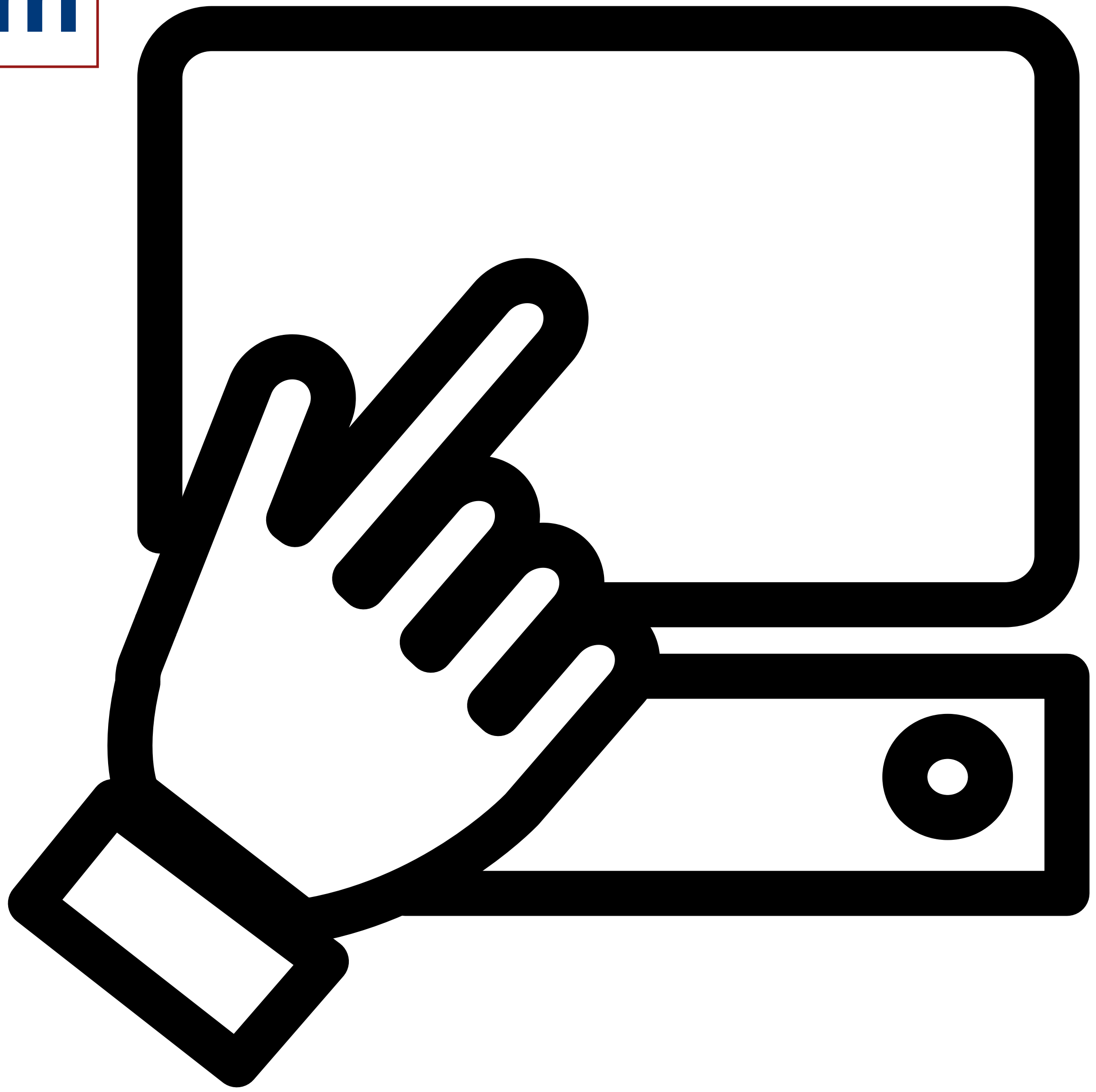
Set [default] role

The role needs to be activate

```
ALTER USER aragorn DEFAULT ROLE
r_lotr_dba;

## OR

SET DEFAULT ROLE r_lotr_dba
TO aragorn;
```

There is more than one way to do it
Unfortunately

#fosdem
#mysqldevroom



Some important points

A user can be granted more roles

Grants are the total of all roles privileges

- User can have many roles
- The default role can be a list of roles

Roles are saved in 'user' table

This may cause some confusion

- Roles are users without login (= account locked and expired password)

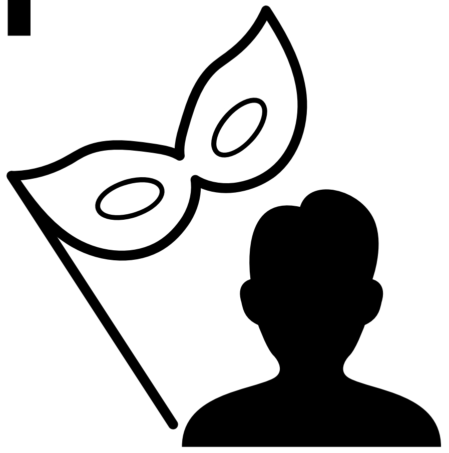
Granting a role to a user is not enough

It is there, but you can't see it

- When we grant a privilege, the user can use it immediately.
- When we grant a role, we first need to set the default.

We can grant a user to a user

This may look tricky, but it is really simple



- Roles are users without login
- But roles with login (i.e. users) can be granted.
- Privileges are assigned regardless of the host of the user.

```
GRANT root@'localhost' to someuser;
```

- user someuser@'%' has all privileges of root from any host

SET ROLE anyone?

You can lose track easily here

- **SET ROLE role_name** is a session assignment of a role
- **SET DEFAULT ROLE role_name** is a permanent assignment of a role for a user
- **SET ROLE DEFAULT** means assign the default role to this user for the session.

Telling roles from users

Sadly, it's up to the DBA's ingenuity

- Roles are users with expired password and locked account.
- A good workaround is using a naming convention to tell roles apart (e.g. "r_something")

There is a feature request about this matter, but I haven’t seen any progress on it.

Tables for roles

We have two new tables in 'mysql' DB dedicated to roles

- **role_edges** reports which roles are assigned to which users.
- **default_roles** takes track of the current default roles assigned to users.

#fosdem
#mysqldevroom

Roles in action

Create roles

```
CREATE ROLE r_lotr_observer;
CREATE ROLE r_lotr_tester;
CREATE ROLE r_lotr_dev;
CREATE ROLE r_lotr_dba;

GRANT SELECT on lotr.* TO r_lotr_observer;
GRANT SELECT, INSERT, UPDATE, DELETE on lotr.* TO r_lotr_tester;
GRANT ALL on lotr.* TO r_lotr_dev;
GRANT ALL on *.* TO r_lotr_dba;
```

Create users and apply roles

```
CREATE USER bilbo     IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER frodo     IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER sam       IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER pippin    IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER merry     IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER boromir   IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER gimli     IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER aragorn   IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER legolas   IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER gandalf   IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER galadriel IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;
CREATE USER gollum    IDENTIFIED BY 'msandbox' PASSWORD EXPIRE;

GRANT r_lotr_observer TO pippin, merry, boromir, gollum;
SET DEFAULT ROLE r_lotr_observer to pippin, merry, boromir, gollum;
GRANT r_lotr_tester TO sam, bilbo, gimli;
SET DEFAULT ROLE r_lotr_tester to sam, bilbo, gimli;
GRANT r_lotr_dev to frodo, aragorn, legolas;
SET DEFAULT ROLE r_lotr_dev to frodo, aragorn, legolas;
GRANT r_lotr_dba TO gandalf, galadriel;
SET DEFAULT ROLE r_lotr_dba to gandalf, galadriel;
```

Users and roles

```
mysql> select host, user, authentication_string from mysql.user
+-----------+------------------+-------------------------------------------+
| host      | user             | authentication_string                     |
+-----------+------------------+-------------------------------------------+
| %         | aragorn          | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | bilbo            | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | boromir          | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | frodo            | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | galadriel        | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | gandalf          | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | gimli            | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | gollum           | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | legolas          | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | merry            | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | pippin           | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| %         | r_lotr_dba       |                                           |
| %         | r_lotr_dev       |                                           |
| %         | r_lotr_observer  |                                           |
| %         | r_lotr_tester    |                                           |
| %         | sam              | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
| localhost | mysql.sys        | *THISISNOTAVALIDPASSWORDTHATCANBEUSEDHERE |
| localhost | root             | *6C387FC3893DBA1E3BA155E74754DA6682D04747 |
+-----------+------------------+-------------------------------------------+
```

Finding roles empirically

```
mysql> select host, user from mysql.user where password_expired='y'
 and account_locked='y';
+------+-----------------+
| host | user            |
+------+-----------------+
| %    | r_lotr_dba      |
| %    | r_lotr_dev      |
| %    | r_lotr_observer |
| %    | r_lotr_tester   |
+------+-----------------+
4 rows in set (0.00 sec)

mysql> select host, user from mysql.user where user like 'r\_%';
+------+-----------------+
| host | user            |
+------+-----------------+
| %    | r_lotr_dba      |
| %    | r_lotr_dev      |
| %    | r_lotr_observer |
| %    | r_lotr_tester   |
+------+-----------------+
4 rows in set (0.00 sec)
```

role_edge table

(Which roles were assigned)

```
mysql> select * from role_edges;
+-----------+-----------------+---------+------------+-------------------+
| FROM_HOST | FROM_USER       | TO_HOST | TO_USER    | WITH_ADMIN_OPTION |
+-----------+-----------------+---------+------------+-------------------+
| %         | r_lotr_dba      | %       | galadriel  | N                 |
| %         | r_lotr_dba      | %       | gandalf    | N                 |
| %         | r_lotr_dev      | %       | aragorn    | N                 |
| %         | r_lotr_dev      | %       | frodo      | N                 |
| %         | r_lotr_dev      | %       | legolas    | N                 |
| %         | r_lotr_observer | %       | boromir    | N                 |
| %         | r_lotr_observer | %       | gollum     | N                 |
| %         | r_lotr_observer | %       | merry      | N                 |
| %         | r_lotr_observer | %       | pippin     | N                 |
| %         | r_lotr_tester   | %       | bilbo      | N                 |
| %         | r_lotr_tester   | %       | gimli      | N                 |
| %         | r_lotr_tester   | %       | sam        | N                 |
+-----------+-----------------+---------+------------+-------------------+
```

default_roles table

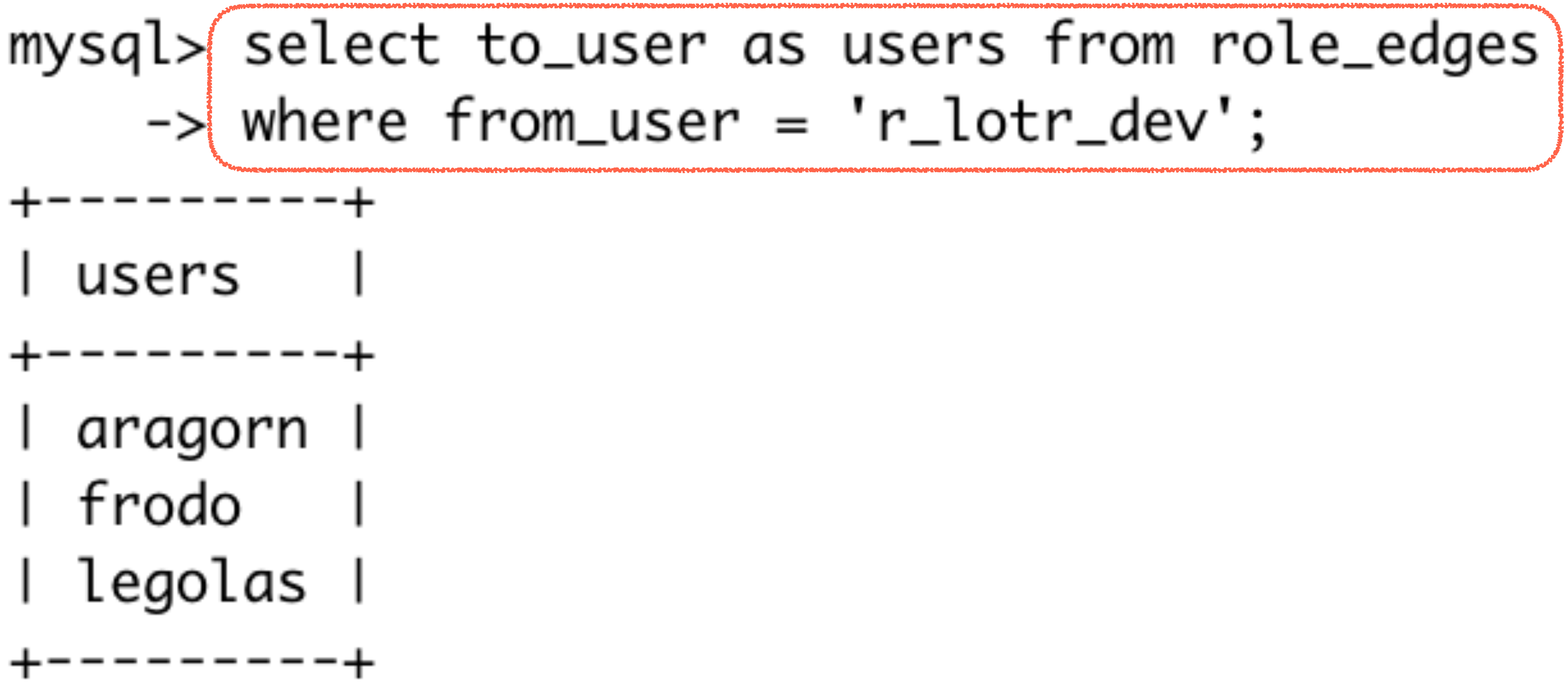
(Which roles are set as default)

```
mysql> select * from  default_roles;
+------+-----------+-------------------+-------------------+
| HOST | USER      | DEFAULT_ROLE_HOST | DEFAULT_ROLE_USER |
+------+-----------+-------------------+-------------------+
| %    | aragorn   | %                 | r_lotr_dev        |
| %    | bilbo     | %                 | r_lotr_tester     |
| %    | boromir   | %                 | r_lotr_observer   |
| %    | frodo     | %                 | r_lotr_dev        |
| %    | galadriel | %                 | r_lotr_dba        |
| %    | gandalf   | %                 | r_lotr_dba        |
| %    | gimli     | %                 | r_lotr_tester     |
| %    | gollum    | %                 | r_lotr_observer   |
| %    | legolas   | %                 | r_lotr_dev        |
| %    | merry     | %                 | r_lotr_observer   |
| %    | pippin    | %                 | r_lotr_observer   |
| %    | sam       | %                 | r_lotr_tester     |
+------+-----------+-------------------+-------------------+
```

Who are the DBAs?

```
mysql> select to_user as users from role_edges
    -> where from_user = 'r_lotr_dba';
+-----------+
| users     |
+-----------+
| galadriel |
| gandalf   |
+-----------+
```

Who are the developers?



```
mysql> select to_user as users from role_edges
    -> where from_user = 'r_lotr_dev';
+---------+
| users   |
+---------+
| aragorn |
| frodo   |
| legolas |
+---------+
```

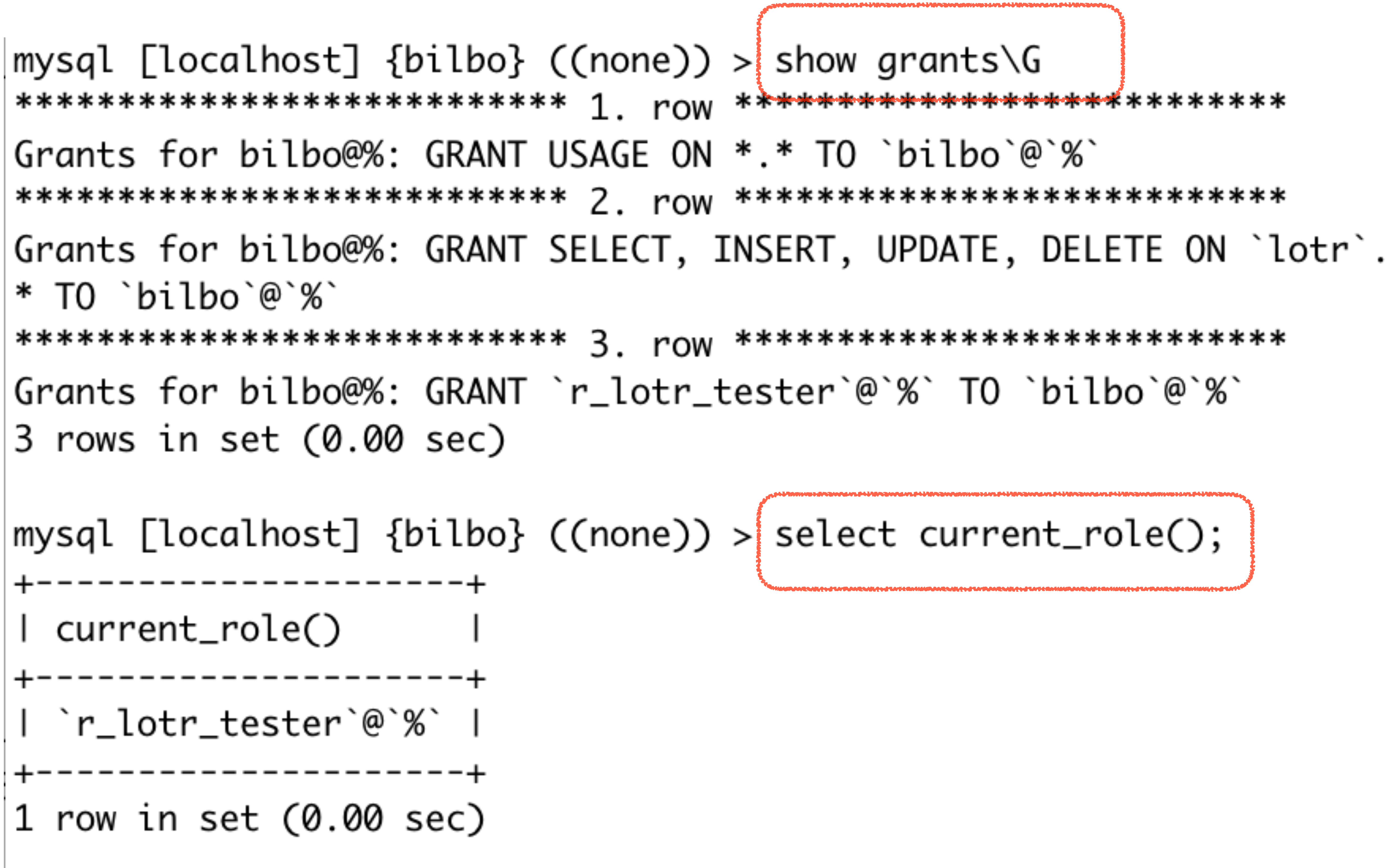
Roles summary

```
mysql> select from_user as role, count(*) as count,
    -> group_concat(to_user) as users
    -> from role_edges group by role;
+-----------------+-------+----------------------------+
| role            | count | users                      |
+-----------------+-------+----------------------------+
| r_lotr_dba      |     2 | galadriel,gandalf          |
| r_lotr_dev      |     3 | aragorn,frodo,legolas      |
| r_lotr_observer |     4 | boromir,gollum,merry,pippin |
| r_lotr_tester   |     3 | bilbo,gimli,sam            |
+-----------------+-------+----------------------------+
```

Default roles summary

```
mysql> select default_role_user as default_role,
    -> group_concat(user) as users
    -> from default_roles group by default_role;
+------------------+-----------------------------+
| default_role     | users                       |
+------------------+-----------------------------+
| r_lotr_dba       | galadriel,gandalf           |
| r_lotr_dev       | aragorn,frodo,legolas       |
| r_lotr_observer  | boromir,gollum,merry,pippin |
| r_lotr_tester    | bilbo,gimli,sam             |
+------------------+-----------------------------+
```

user with default role

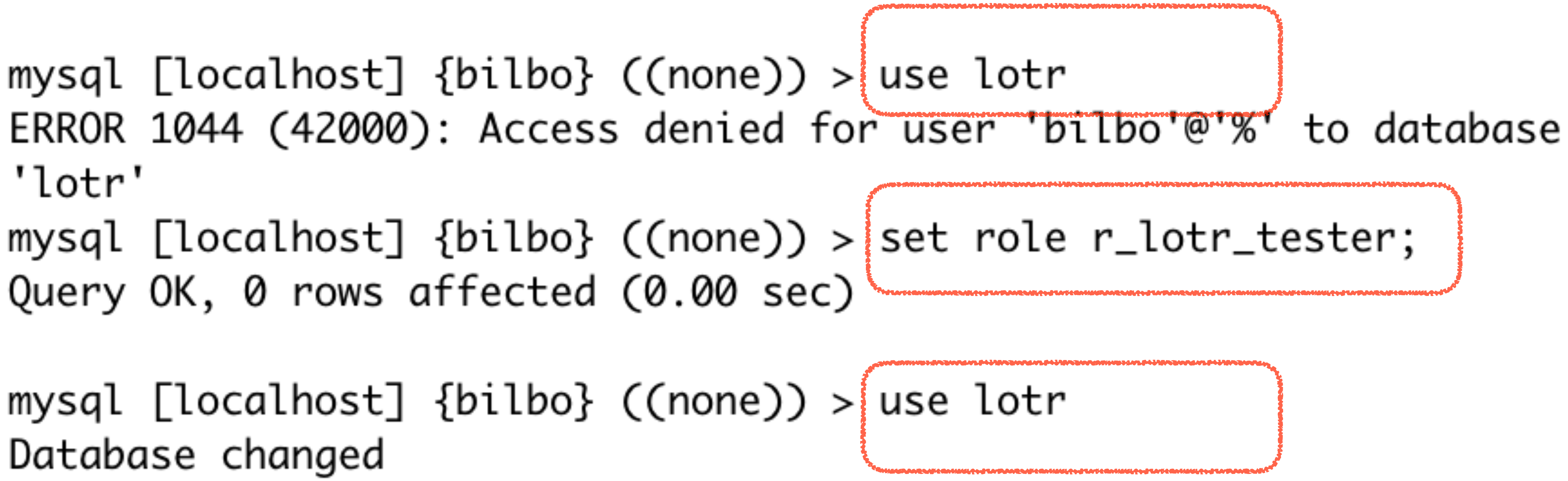


```
mysql [localhost] {bilbo} ((none)) > show grants\G
*************************** 1. row ***************************
Grants for bilbo@%: GRANT USAGE ON *.* TO `bilbo`@`%`
*************************** 2. row ***************************
Grants for bilbo@%: GRANT SELECT, INSERT, UPDATE, DELETE ON `lotr`.
* TO `bilbo`@`%`
*************************** 3. row ***************************
Grants for bilbo@%: GRANT `r_lotr_tester`@`%` TO `bilbo`@`%`
3 rows in set (0.00 sec)

mysql [localhost] {bilbo} ((none)) > select current_role();
+----------------------+
| current_role()       |
+----------------------+
| `r_lotr_tester`@`%`  |
+----------------------+
1 row in set (0.00 sec)
```

User without default role

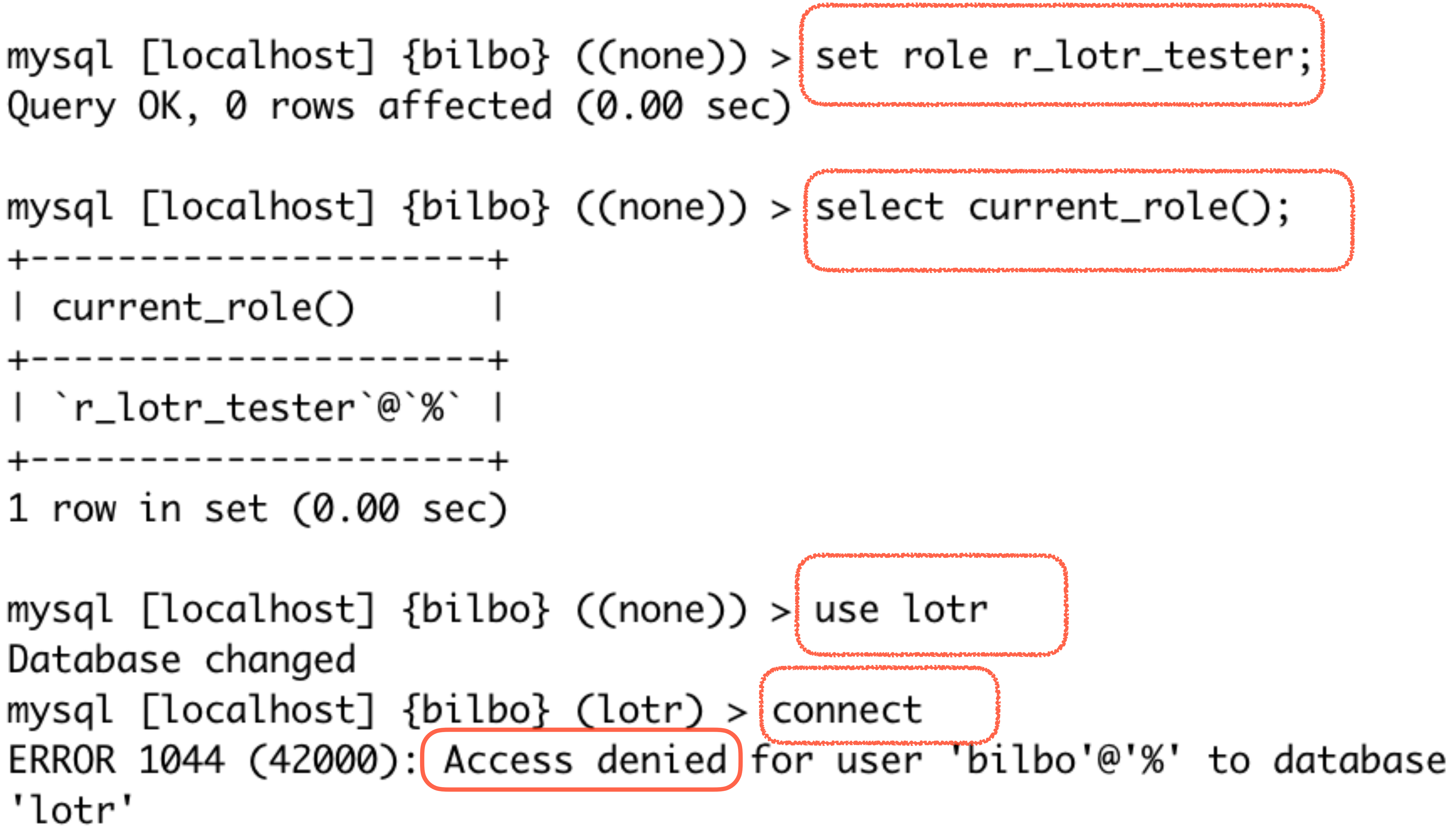
```
mysql [localhost] {bilbo} ((none)) > show grants\G
*************************** 1. row ***************************
Grants for bilbo@%: GRANT USAGE ON *.* TO `bilbo`@`%`
*************************** 2. row ***************************
Grants for bilbo@%: GRANT `r_lotr_tester`@`%` TO `bilbo`@`%`
2 rows in set (0.00 sec)

mysql [localhost] {bilbo} ((none)) > show grants for bilbo using r_
lotr_tester\G
*************************** 1. row ***************************
Grants for bilbo@%: GRANT USAGE ON *.* TO `bilbo`@`%`
*************************** 2. row ***************************
Grants for bilbo@%: GRANT SELECT, INSERT, UPDATE, DELETE ON `lotr`.
* TO `bilbo`@`%`
*************************** 3. row ***************************
Grants for bilbo@%: GRANT `r_lotr_tester`@`%` TO `bilbo`@`%`
3 rows in set (0.00 sec)
```

User without default role



```
mysql [localhost] {bilbo} ((none)) > use lotr
ERROR 1044 (42000): Access denied for user 'bilbo'@'%' to database
'lotr'
mysql [localhost] {bilbo} ((none)) > set role r_lotr_tester;
Query OK, 0 rows affected (0.00 sec)

mysql [localhost] {bilbo} ((none)) > use lotr
Database changed
```

SET ROLE is not permanent



```
mysql [localhost] {bilbo} ((none)) > set role r_lotr_tester;
Query OK, 0 rows affected (0.00 sec)

mysql [localhost] {bilbo} ((none)) > select current_role();
+---------------------+
| current_role()      |
+---------------------+
| `r_lotr_tester`@`%` |
+---------------------+
1 row in set (0.00 sec)

mysql [localhost] {bilbo} ((none)) > use lotr
Database changed
mysql [localhost] {bilbo} (lotr) > connect
ERROR 1044 (42000): Access denied for user 'bilbo'@'%' to database
'lotr'
```

Back to the BAD example. (2)

```
mysql [localhost] {do_it_all} ((none)) > use test;
ERROR 1044 (42000): Access denied for user 'do_it_all'@'%'
 to database 'test'
mysql [localhost] {do_it_all} ((none)) > show grants;
+--------------------------------------------+
| Grants for do_it_all@%                     |
+--------------------------------------------+
| GRANT USAGE ON *.* TO `do_it_all`@`%`      |
| GRANT `powerful`@`%` TO `do_it_all`@`%`    |
+--------------------------------------------+
2 rows in set (0.00 sec)

mysql [localhost] {do_it_all} ((none)) > _
```

Back to the BAD example. (3)

```
mysql [localhost] {do_it_all} ((none)) > set role powerful;
Query OK, 0 rows affected (0.00 sec)

mysql [localhost] {do_it_all} ((none)) > show grants\G
*************************** 1. row ***************************
Grants for do_it_all@%: GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DR
OP, RELOAD, SHUTDOWN, PROCESS, FILE, REFERENCES, INDEX, ALTER, SHOW DATA
BASES, SUPER, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION
 SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTE
R ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE, CREATE ROLE,
DROP ROLE ON *.* TO `do_it_all`@`%`
*************************** 2. row ***************************
Grants for do_it_all@%: GRANT BINLOG_ADMIN,CONNECTION_ADMIN,ENCRYPTION_K
EY_ADMIN,GROUP_REPLICATION_ADMIN,REPLICATION_SLAVE_ADMIN,ROLE_ADMIN,SET_
USER_ID,SYSTEM_VARIABLES_ADMIN ON *.* TO `do_it_all`@`%`
*************************** 3. row ***************************
Grants for do_it_all@%: GRANT `powerful`@`%` TO `do_it_all`@`%`
3 rows in set (0.00 sec)

mysql [localhost] {do_it_all} ((none)) > use test
Database changed
```

Roles can be active by default

It’s a all-or-nothing option

- Starting in 8.0.2
- You can use option **activate_all_roles_on_login**
  - When enabled, all roles become active by default
- And **mandatory_roles**
  - When set, all users will get the role(s) defined

Example with mandatory roles (1)

```
mysql> create schema lotr;
Query OK, 1 row affected (0.00 sec)

mysql> grant select on lotr.* to r_lotr_reader;
Query OK, 0 rows affected (0.00 sec)

mysql> set global mandatory_roles='r_lotr_reader';
Query OK, 0 rows affected (0.00 sec)

mysql> create user dummy identified by 'msandbox';
Query OK, 0 rows affected (0.00 sec)
```

Example with mandatory roles (2)

```
$ mysql lotr -u dummy -p
ERROR 1044 (42000): Access denied for user 'dummy'@'%'
to database 'lotr'
# ====== as root ========
mysql> set global activate_all_roles_on_login=1;

$ mysql lotr -u dummy -p
mysql> show grants;
+-------------------------------------------+
| Grants for dummy@%                        |
+-------------------------------------------+
| GRANT USAGE ON *.* TO `dummy`@`%`         |
| GRANT SELECT ON `lotr`.* TO `dummy`@`%`   |
| GRANT `r_lotr_reader`@`%` TO `dummy`@`%`  |
+-------------------------------------------+
3 rows in set (0.00 sec)
```

Example with mandatory roles (3)

```
$ mysql lotr -u root -p
mysql> show grants\G
*** 1. row ***
Grants for root@localhost: GRANT SELECT, INSERT, UPDATE, DELETE,
CREATE, DROP, RELOAD, SHUTDOWN, PROCESS, FILE, REFERENCES, INDEX, ALTER,
SHOW DATABASES, SUPER, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE,
REPLICATION SLAVE, REPLICATION CLIENT, CREATE VIEW, SHOW VIEW, CREATE
ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE TABLESPACE,
CREATE ROLE, DROP ROLE ON *.* TO `root`@`localhost` WITH GRANT OPTION
*** 2. row ***
Grants for root@localhost: GRANT
BACKUP_ADMIN,BINLOG_ADMIN,CONNECTION_ADMIN,ENCRYPTION_KEY_ADMIN,GROUP_REPL
ICATION_ADMIN,PERSIST_RO_VARIABLES_ADMIN,REPLICATION_SLAVE_ADMIN,RESOURCE_
GROUP_ADMIN,RESOURCE_GROUP_USER,ROLE_ADMIN,SET_USER_ID,SYSTEM_VARIABLES_AD
MIN,XA_RECOVER_ADMIN ON *.* TO `root`@`localhost` WITH GRANT OPTION
*** 3. row ***
Grants for root@localhost: GRANT SELECT ON `lotr`.* TO
`root`@`localhost`
```

Latest addition

- In 8.0.3+
- You can set the **default role** within **CREATE USER**.
- Sounds good
- Until you notice that you are activating a role that has not been assigned yet.

😧

Q & A

@datacharmer

#fosdem
#mysqldevroom