How to fix the MySQL error message: Access denied for user 'qube_readonly'@'localhost'

The MySQL access error on the supervisor is due to the mysql **qube_readonly** user having no permission to read when connecting from the hostname **localhost**, which is shorthand for "this machine".

All SQL commands following in **BLUE** are run with the `mysql` client utility, which can be found at the following locations:

OS X: `/usr/local/mysql/bin/mysql`

Linux: `/usr/bin/mysql`

Windows: `mysql` (usually in the SEARCH path, courtesy of the Qube installation)

For the remainder of this page, if you see **mysql** in the command line, add the proper path to `mysql` for your particular operating system.

> (i) **Performance charts**
> This will also enable WranglerView to view performance charts when run from the supervisor.

How to fix the problem when logged into the supervisor

It's necessary to **explictly grant** the **qube_readonly** user **read-only** access from localhost by running the SQL statement **on the supervisor machine**:

```
mysql -u root -e "GRANT SELECT ON *.* TO 'qube_readonly'@'localhost';"
```

Then, restart the QubeGUI, and the "access denied" messages should no longer occur.

Why it occurs

The MySQL default permissions are refusing the qube_readonly user access to the databases. We'll take a quick look at why this occurs, and then how to fix it.

Find out who's granted some sort of permission from where:

```
mysql -u root -e "SELECT user,host from mysql.user ORDER BY user;"

+---------------+----------------+
| user          | host           |
+---------------+----------------+
|               | localhost      |
| jburk         | localhost      |
| jburk         | 10.0.1.150     |
| pfx_dw        | 127.0.0.1      |
| pfx_dw        | localhost      |
| qube_readonly | %              |
| root          | 192.168.60.165 |
| root          | 10.0.1.101     |
| root          | 127.0.0.1      |
| root          | jburk-17-mbpro |
| root          | 10.10.10.1     |
| root          | localhost      |
+---------------+----------------+
```

The one's we're interested in here are:

- the first one, with the "blank" user value: this is for any user not explicitly listed from the host localhost, which is another way to refer to the local machine the supervisor is on in my test configuration.
- the one for the user **qube_readonly** at host %, which is the "wildcard" meaning "all hosts".

So, despite the **qube_readonly** user being granted MySQL `SELECT` (read-only) permissions by the

```
mysql -u root -e "SHOW GRANTS FOR 'qube_readonly'@'%';"
+------------------------------------------+
| Grants for qube_readonly@%               |
+------------------------------------------+
```

```
| GRANT SELECT ON *.* TO 'qube_readonly'@'%' |
+------------------------------------------+
```

statement, the default explicit host definition in the next **USAGE** statement overrides the '**%**' wildcard in the previous statement, **effectively denying access** to any user **not specifically granted access** from the local machine:

```
mysql -u root -e "SHOW GRANTS FOR ''@'localhost';"
+----------------------------------------+
| Grants for @localhost                  |
+----------------------------------------+
| GRANT USAGE ON *.* TO ''@'localhost'   |
+----------------------------------------+
```

The **USAGE** permission is effectively an empty GRANT statement, which means "no read permission", and overrides the previous permissions grant for **qube_readonly** from **all machines.**