Title: Unable to link mysql docker container with spring boot application - Communications link failure

Post Body:

I am new to Docker. I am using Spring boot micro service. It's running well On my local machine. Now, I need to create a docker image for my application. It has a dependency on Mysql server. I am using docker-compose to create my containers. I am getting a communications link failure error while running my custom image (spring boot application). The Mysql image is running well independently.

My yml file :

```
version: '2'  services:    mysql-dev:     image: mysql:5.7     container_name: mysql-dev     environment:       MYSQL_ROOT_PAS
```

application properties file :

```
db.driver: com.mysql.jdbc.Driver  spring.datasource.url = jdbc:mysql://mysql-dev:3308/onlinetutorialspoint?useSSL=false  sprin
```

Full Error Message :

> spring_boot_db_service_1 | 2019-01-28 13:34:06.955 INFO 1 --- [
> main] org.hibernate.cfg.Environment : HHH000206: hibernate.properties not found spring_boot_db_service_1 | 2019-01-28 13:34:07.000 INFO 1 --- [main] o.hibernate.annotations.common.Version : HCANN000001: Hibernate Commons Annotations {5.0.1.Final} spring_boot_db_service_1 | 2019-01-28 13:34:08.430 WARN 1 --- [**main] o.h.e.j.e.i.JdbcEnvironmentInitiator : HHH000342: Could not obtain connection to query metadata : Communications link failure** spring_boot_db_service_1 | spring_boot_db_service_1 | The last packet sent successfully to the server was 0 milliseconds ago. The driver has not received any packets from the server. spring_boot_db_service_1 | 2019-01-28 13:34:08.443 INFO 1 --- [
> main] org.hibernate.dialect.Dialect : HHH000400: Using dialect: org.hibernate.dialect.MySQL5Dialect spring_boot_db_service_1 | 2019-01-28 13:34:08.459 INFO 1 --- [main] o.h.e.j.e.i.LobCreatorBuilderImpl : HHH000422: Disabling contextual LOB creation as connection was null spring_boot_db_service_1 | 2019-01-28 13:34:08.921 WARN 1 --- [
> main] ConfigServletWebServerApplicationContext : Exception encountered during context initialization - cancelling refresh attempt: org.springframework.beans.factory.UnsatisfiedDependencyException: Error creating bean with name 'dbServiceImpl': Unsatisfied dependency expressed through field 'dbServiceDao'; nested exception is org.springframework.beans.factory.UnsatisfiedDependencyException: **Error creating bean with name 'dbServiceDaoImpl': Unsatisfied dependency expressed through field 'sessionFactory'; nested exception is org.springframework.beans.factory.BeanCreationException: Error creating bean with name 'sessionFactory'** defined in class path resource [com/htc/dbservice/configuration/DBConfiguration.class]: Invocation of init method failed; nested exception is org.hibernate.MappingException: Could not get constructor for org.hibernate.persister.entity.SingleTableEntityPersister spring_boot_db_service_1 | 2019-01-28 13:34:08.923 WARN 1 --- [
> main] o.s.b.f.support.DisposableBeanAdapter : Invocation of destroy method 'close' failed on bean with name 'eurekaRegistration': org.springframework.beans.factory.BeanCreationNotAllowedException: Error creating bean with name 'org.springframework.cloud.netflix.eureka.EurekaClientAutoConfiguration$RefreshableEurekaClientConfiguration': Singleton bean creation not allowed while singletons of this factory are in destruction (Do not request a bean from a BeanFactory in a destroy method implementation!) spring_boot_db_service_1 | 2019-01-28 13:34:08.926 INFO 1 --- [main] o.apache.catalina.core.StandardService : Stopping service [Tomcat] docker_spring_boot_db_service_1 exited with code 1

Accepted Answer: None

Highest Rated Answer:

Did you change the default mysql port? if not, you are pointing to wrong port number. When you map the port to pubilsh the service externally, the order is

```
<host-port>:<container-port>
```

```
services:       mysql-dev:        image: mysql:5.7        container_name: mysql-dev        environment:          MYSQL_ROOT_PA
```

And using docker-compose and connecting both services to the same network you can connect directly to the container without pubishing the mysql port to the external network. Try to change your spring app pointing to mysql-dev:3306, or try to connect to the container **spring_boot_db_service** using

```
docker-compose exec spring_boot_db_service bash
```

and make a connectivity test to the database container.