

Title: java.net.UnknownHostException: host.docker.internal: Name or service not known on AWS EC2

Post Body:

I ran into this "java.net.UnknownHostException: host.docker.internal: Name or service not known" problem when deploying a dockerized spring boot application on an AWS EC2 T2.micro instance. The spring boot application failed to start because of this error.

But the weird part is, I did not use the variable "host.docker.internal" anywhere in my application: not in the code, not in the yaml file, not in the .env file:

```
$ sudo grep -Rl "host.docker.internal" ~ /home/ec2-user/.bash_history
```

And when I run the following command it shows nothing but previous command to search for it:

```
$ cat /home/ec2-user/.bash_history | grep "host.docker.internal"
```

Locally I am using Windows 10 for development, and I can successfully bring up the stack with docker-compose.

Here is the EC2 instance OS version info:

```
$ cat /etc/*release NAME="Amazon Linux" VERSION="2" ID="amzn" ID_LIKE="centos rhel fedora" VERSION_ID="2" PRETTY_NAME="Amazon Linux 2" CPE_NAME="cpe:2.3:amazon:linux:2:latest:us-east-1:aws:ec2" RESPONSE="2017-09-30" _source="https://aws.amazon.com/amazon-linux-2/"
```

And here is the docker-compose file that I used on the EC2 instance:

```
version: '2' services: backend: container_name: backend image: 'dockerhubuser/backend:0.0.4' ports: - '8080:8080'
```

And here is my .env file on the EC2 instance:

```
SERVER_PORT=8080 KEY_STORE=/etc/ssh/backend/keystore.p12 KEY_STORE_PASSWORD=keystorepass REDIS_HOST=redis REDIS_PORT=6379 DB_HOST=redis
```

I am pretty sure that this .env file is being used when bringing up the stack with "docker-compose up" because I can see the SERVER_PORT in the log matches this file when I change it.

```
2021-01-02 20:55:44.870 [main] INFO o.s.b.w.e.tomcat.TomcatWebServer - Tomcat initialized with port(s): 8080 (https)
```

But I keep getting the error complaining about "host.docker.internal".

Here are things that I have tried but not working:

1. Hard-code the db host in property spring.datasource.url in application.yml
2. Add the following entry to /etc/hosts file (see <https://stackoverflow.com/a/48547074/1852496>)

```
172.17.0.1 host.docker.internal
```

3. Add the following entry to /etc/hosts file, where "ip-172-31-33-56.us-east-2.compute.internal" is what I got when running command "echo \$HOSTNAME"

```
ip-172-31-33-56.us-east-2.compute.internal host.docker.internal
```

4. Terminate the instance and created another T2.micro instance, but got same result.
5. Edit inbound rules to allow TCP:3306 from anywhere.

Can someone take a look? Any help appreciated.

Accepted Answer:

It works on Ubuntu 20.04 after adding "172.17.0.1 host.docker.internal" to /etc/hosts file. Make sure the docker engine version is 20.10-beta1 or newer.

Highest Rated Answer: None