# Debug Docker Errors

**Seriously have you tried just rebooting your machine?**

For general container logs you can  use the standard docker logs command:

```
docker logs -f --until=120s laravel
```

**SSLCertificateFile: file ‘/config_items/certs/impressto.pem’ does not exist or is empty**

If the folder ~/Sites/impressto-docker/config/certs exists but is empty you will need to run this terminal command:

```
cd ~/Sites/impressto-docker
./createSSLCert.sh
```

**Ngserve is not Running**

If you are unable to load the webapp on ngserve.impressto.localhost, it is likely caused by a missing dependency in the ~/Sites/impressto-webapp folder. Most likely it is a missing environment.local.ts file.

You can test ngserve by logging into the docker container with “impressto” and running the following:

```
cd /var/www/impressto-webapp;
ng serve ---configuration local --base-href /app/  --ssl true
```

Once you have fixed the issue you can run the “impressto” command again and wait a few minutes for ng-serve to rebuild the files.

**Cannot create container from docker-compose due to short volume name**

You may have forgotten to edit the file ~/Sites/impressto-docker/.env.example.  Make changes as needed and save the file as .env. You can also run:

```
cd ~/Sites/impressto-docker;
./prepareDockerConfigs.sh;
```

**Containers fail to load or shut down randomly on your machine but not others**

If you see this happening it is likely RAM related. Either you are running out of memory of you have bad RAM.

First try importing a docker container. If the imported container works you likely do not have hardware issues. If you are still having crashed containers after importing a container image, you need t start testing your system hardware. A common symptom of bad RAM is random computer crashes and intermittent freezing interfaces. If you need to reboot your machine several times a day, your hardware is probably baked.

**Try memtester in Ubuntu 20**

```
sudo apt-get install memtester
sudo memtester 1024 5
```

Another option is GTK Stress Tester but that will not find memory faults.

**Composer running out of memory**

Composer defaults to a maximum of 1.5G of memory usage. Sometimes this is not enough for a composer update. If you notice that builds are not completing correctly for this reason, a work-around is the following command:

```
COMPOSER_MEMORY_LIMIT=-1 composer update
```

**Error: Cannot find module ‘nan’**

This is  more of an angular issue on mac but you may run into it while setting up your local webapp. Fix is to go into the ~/Sites/impressto-webpp folder and enter this command:

```
npm i -g nan
```

**Can’t connect to docker daemon. Is ‘docker -d’ running**.

This error may also appear as:

```
ERROR: Couldn't connect to Docker daemon at http+docker://localhost - is it running?
```

This error is most commonly seen if you have not added your user to the docker group. Rum the following commands:

```
sudo groupadd docker;


sudo usermod -aG docker $USER;
```

After that simply reboot your machine and the problem should go away.

**create .: volume name is too short, names should be at least two alphanumeric characters**

Did you remember to rename the docker root folderfile .env.example to env ?

Also this can happen if the formatting in the docker-compose.yml is not correct (bad indenting).

**Cannot use port 80**

If you have ngonx,  apache or skype installed on the host system that will block the use of port 80. To determine what is running on port  80 use this command:

```
sudo lsof -i tcp:80
```

This should display something like this

```
sudo lsof -i tcp:80

COMMAND  PID     USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
nginx   1858     root    6u  IPv4   5043      0t0  TCP ruir.mxxx.com:http (LISTEN)
nginx   1867 www-data    6u  IPv4   5043      0t0  TCP ruir.mxxx.com:http (LISTEN)
nginx   1868 www-data    6u  IPv4   5043      0t0  TCP ruir.mxxx.com:http (LISTEN)
nginx   1869 www-data    6u  IPv4   5043      0t0  TCP ruir.mxxx.com:http (LISTEN)
nginx   1871 www-data    6u  IPv4   5043      0t0  TCP ruir.mxxx.com:http (LISTEN)
```

identify the PID of the process using port 80 and kill it using a command like this

```
sudo lsof -t -i tcp:80 -s tcp:listen | sudo xargs kill
```

You can also permanently turn off apache on the host with:

```
sudo service apache2 stop;
sudo service mysql stop;

# also apache and mysqlfrom starting as a service on bootup
sudo systemctl disable apache2 mysql;
```

in some cases it is easiest to just completely remove apache2 from the host system

```
sudo apt-get --purge remove apache2;
sudo apt-get remove apache2-common;
```

---

**NodeJS – FATAL ERROR**: Ineffective mark-compacts near heap limit Allocation failed – JavaScript heap out of memory

This is not actually a docker error but may occur if you are runniing webpack builds inside docker (not recommended). If you are getting this error on our host system try the following command which is what we have used on the feature and builder servers:

```
 #increase node memory to 2gb
export NODE_OPTIONS=--max-old-space-size=2048
```

**Performance issue with Mac:**

Follow the official instructions for installing Docker on Mac. In a nutshell you will need to download Docker for Mac, and install it as you would any other Mac app. **IMPORTANT: make sure you have the latest version of docker for Mac**. Once installed you will need to allocate enough memory for docker to run the containers. Recommended size is 8GB. Not setting the memory limit may cause the elastic search container to exit with a 137 error code (docker container out of memory). Linux does not require this config as it allocates memory directly from the host system.

---

Revision #1
Created 2024-06-14 00:56:19 UTC by peterd
Updated 2024-06-14 00:57:18 UTC by peterd