Lab solutions for 03-09

1

Made new Harbor account on `10.212.136.160`:

```
Username: TnT
Email: trondhth@stud.ntnu.no
First and last name: Tango November Tango
Password: dynamittH4rry
```

Made a new private project called `docker`

2

On both docker VM and manager VM:

```
curl http://10.212.136.140/harbor.crt > /usr/local/share/ca-
    certificates/harbor.crt
update-ca-certificates
service docker restart
```

Logged in to Harbor via shell with `docker login 10.212.136.160`

bookfaceimage can now be pulled down to manager VM.

Uploaded bookfaceimage to harbor with `docker tag bookfaceimage 10.212.136.160/docker/bookfaceimage:latest` `docker push 10.212.136.160/docker/bookfaceimage:latest` It works!

3

Installed docker on www1, www2, www3. On manager: Initiated docker master by typing `docker swarm init` Return token: `docker swarm join --token SWMTKN-1-5d0xxilhf03y8ds2zb1gdfrtj9duqm71-2096wfbcx0xm0ssjk34hi6941 10.10.0.70:2377` Added the token to all thre www servers. On manager: Checked if all nodes are connected by typing `docker node ls`. It works!

This process can be automated by for example create a script that stores the token in a file, and then uses `scp` and `ssh` to get it to run locally on the worker.

4

Started bookface webservers as a service with `docker service create -d ---replicas=3 --name=bookface --with-registry-auth -p 3000:80 10.212.136.160/docker/bookfaceimage:latest` Checked status on all three replicas by typing `docker service ls`. Showing 3/3!

Went to balancer and haproxy.cfg and added `server dockerMaster 10.10.0.70:3000 check` Also changed www1, www2 and www3 to listen to port 3000.

The server that is the docker master is manager, so haproxy points to this. Also haproxy points to www1. www2 and www3 as these are the docker containers that runs the whole infrastructure.

If one of the swarm- servers are taken down, the users of bookface won't notice anything significantly, because there are other servers in the swarm that routes the traffic, functioning as a fail-over.

5

TBA