

Docker Swarm Ingress

TRAINING MATERIALS - MODULE HANDOUT

Contacts

team.qac.all.trainers@qa.com

www.consulting.qa.com

Contents

Overview	2
Configuring NGINX for HTTP Ingress	2
Tasks	2

Overview

Ingress in Swarm allows us to load balance traffic over multiple instances of our applications. Documentation for Swarm indicates that exposing the services to outside of the network can be done using a load balancer such as HAProxy or NGINX: <https://docs.docker.com/engine/swarm/images/ingress-lb.png>.

Configuring NGINX for HTTP Ingress

We will consider NGINX as a tool for ingress here utilising upstreams so that traffic will be load-balanced. As per the Swarm documentation we can configure NGINX to load balance over all nodes in the Swarm. As a potential solution, NGINX can be run in a Docker container on a separate machine.

Below is an NGINX configuration for load balancing across two nodes using a python server as an example. If you were to use this you would replace the IP addresses with those of your Swarm nodes.

```
events{}
http {
    upstream python-http-server {
        server 10.0.0.1:9000;
        server 10.0.0.2:9000;
    }
    server {
        location / {
            proxy_pass http://python-http-server;
        }
    }
}
```

To get the container up and running, the following docker command can be used:

```
docker run -d -p 80:80 --name nginx-swarm-ingress --mount type=bind,source=$(pwd)/nginx.conf,target=/etc/nginx/nginx.conf nginx:latest
```

Tasks

Configure a 2 node Swarm (1 manager, 1 node) using the [bobcrutchley/python-http-server](#) image. Create a 3rd VM to use for the HTTP ingress, setting up NGINX in a Docker container with the correct configurations. Confirm that the application is accessible from the internet and that it is being load balanced (refresh the page).