

# Assignment 1: Docker

## IN720 Virtualisation

### Introduction

In this assignment you will create a set of Docker containers to operate a highly-available web site. You will then deploy those containers to a CoreOS cluster running on AWS.

This assignment is worth 25% of your mark in this paper.

### 1 Create a data container

Your first container's role is to hold your web site code. Create a container that mounts a *volume* at `/var/www/html`. This directory just needs to hold a simple web page in a file named `index.html`. This container image will use a Docker Hub automated build so that we can easily get a new image any time we commit a change to its GitHub repository.

### 2 Create a web server container

Your second container will run your web server. Create a container from Ubuntu 14.04 with Apache installed on it. This container should mount the volume provided by your first one and will expose port 80.

### 3 Deployment

You will deploy your containers to a cluster of CoreOS machines running on AWS. Write Fleet service files that allow you to deploy data and web server containers on two machines in the cluster. Note that you will need to ensure that the data and web server containers are deployed together on the same containers.

You will deploy another container that runs a load balancing server on the third system in your cluster. This container image will be supplied by the lecturer, but you will need to write the Fleet service file to deploy it.

Since your load balancer will need to find your web servers, you will need to set up *sidekick services* that announce the presence of the web servers to etcd.

Also, note that you will need to control the order in which Fleet deploys your containers. The data containers need to deploy first and the web servers should deploy next. It doesn't actually matter when the load balancer deploys. It will watch etcd and pick up the web servers as they appear.

## 4 Assignment submission and marking

Some of the assignment deliverables will be uploaded to your GitHub account. Place all of these into a repository named `virtualisation_assn1`.

Specifically, your submission will include the3 following parts:

**Data container (25 marks)** You will commit your build context files to the GitHub repository mentioned above (You need to do this for the autobuild anyway.) You will also have a corresponding container image on your Docker Hub.

**Web server container (15 marks)** This container image will also be placed on your Docker Hub.

**Fleet service files (25 marks)** Your Fleet service files will be committed into a directory in your GitHub repository named `fleet`.

**Deployment (35 marks)** When you are ready to deploy, contact the lecturer to receive details for your CoreOS cluster. You will then deploy your containers onto the cluster and demonstrate that they run.

NB:

- Be sure that your images have a proper maintainer.
- Be sure that your images' `From` attributes specify a specific version.
- Be sure that your images apply updates when they are built. Note also that you need to use something like a “last modified” attribute to be sure this happens<sup>1</sup> (See Turnbull)

This assignment is due at 9:00 AM on Monday, 8 September. Tag both your GitHub repository and your Docker Hub images with the tag “assignment1”.

---

<sup>1</sup>Do you know why? Don't you think you should?