Assignment 3: Virtualization using Docker

Assignment 3.1: Container Virtualization

Objectives:
Learn how to use Docker technology to package an application with all of its dependencies into a standardized unit for software development. Learn how to achieve resource isolation and allocation using Dockers. Dockers are similar to VM used in assignment 2 but much more lightweight

Background:
Docker[1] containers helps to wrap up a piece of software in a complete file system that contains everything it needs to run: code, runtime, system tools, system libraries – anything you can install on a server. This guarantees that it will always run the same, regardless of the environment it is running in. Containers have similar resource isolation and allocation benefits as virtual machines but a different architectural approach allows them to be much more portable and efficient.

Docker runs on a variety of Linux distributions, Docker doesn't run natively on OS X and Microsoft Windows. Installing Docker on OS X and Microsoft Windows platforms is done via a virtual machine.

TODO:

1. **Prepare** the Lab by following the tutorials[2]. We advise you to follow the tutorials and download and configure all the software needed for the lab.

2. **Contextualize a Docker container** to deploy the URL Shortner Web service (developed for Assignment 1.1). The service should be available (visible) from other containers first on the **same** host or from **remote** host. To enable remote containers to communicate use the p2p vpn (zerotier) system[3] .
   a. Test available bandwidth between docker containers using iperf[4].
   b. Test broadcasting messages between docker containers using socat [5], netcat[6]
   c. Deploy URL Shortner Web service (developed for Assignment 1.1).

3. **How to get your assignment GRADED:**

---

[1] Docker web site https://www.docker.com/what-docker
[2] Introduction to Docker Version: 0.0.3
http://cdn.oreillystatic.com/en/assets/1/event/115/Introduction%20to%20Docker_%20Containerization%20is%20the%20New%20Virtualization%20Presentation.pdf
[3] https://www.zerotier.com
[4] https://en.wikipedia.org/wiki/Iperf
[5] http://www.buntschu.ch/blog/?p=242
[6] http://scitechconnect.elsevier.com/wp-content/uploads/2013/09/Introduction-to-Netcat.pdf

1. Show a working prototype to the Lab assistant
2. Submit via Blackboard a tar file containing
   a. Upload Doker file /the image to file <url-Will-be-provided during lab>
   b. A report describing your experience with Docker
   c. The tar file should be named: <groupnumber>_Docker_3.1.tar

Assignment 3.2: Container Orchestrations

Objectives:

Learn how to use tools (Kubernetes) to orchestrate multiple containers.

Background:

Kubernetes[7] is an open-source system for automating deployment, scaling, and management of containerized applications. It groups containers that make up an application into logical units for easy management and discovery. Kubernetes builds upon 15 years of experience of running production workloads at Google, combined with best-of-breed ideas and practices from the community.

TODO:

1. **Prepare** at least 3 VMs to create a Kubernetes cluster.

2. **Install Kubernetes** in your cluster [8]. Ensure that the relevant Kubernetes daemons are running, and the overlay network[9] is working properly.

3. Create a **Deployment[10]** in the kubenetes to deploy URL Shortner Web service (developed for Assignment 1.1). There should be at least 3 replicas of your services (pods in kubernetes) in the deployment.

4. Create a **Service[11]** in the kubenetes to make your application service (in this case URL Shortner) accessible from outside. Try to understand Kubernetes load balancing during the experiment, and how to achieve it in the platform.

How to get your assignment GRADED:
1. Show a working prototype to the Lab assistant
2. **Submit via Blackboard a** tar file containing

---

[7] https://kubernetes.io/

[8] https://kubernetes.io/docs/setup/independent/install-kubeadm/

[9] https://medium.com/google-cloud/understanding-kubernetes-networking-pods-7117dd28727

[10] https://kubernetes.io/docs/concepts/workloads/controllers/deployment/

[11] https://kubernetes.io/docs/concepts/services-networking/service/

a. Upload ***Deployment yaml file/Service yaml file*** to file <url-Will-be-provided during lab>
b. A short report describing your experience with Kubernetes to deployment your web services
c. The tar file should be named: <groupnumber>_Docker _3.2.tar