**OpenShift Fundamentals**

Container platforms like openshift provides orchestration and other services that containers need in order for users to take full advantage of containerisation.  
  
Container Platform:An application platform that uses containers to build,deploy,serve and orchestrate applications running inside it.

OpenShift uses two tools:

1.container runtime

2.orchestration engine

**Containers**:Containers hold everything that an application requires to function.

In OpenShift,The service that handles creation and management of containers is***Docker***.

Although Docker engine can facilitate the management of containers in a single OS on a single server.

Highly scalable and available applications needs the containers to be distributed over several servers.

To orchestrate docker containers across a number of servers, a container orchestration engine is required

In openShift orchestration engine used is ***Kubernetes***.

Kubernetes employs a master/node architecture.

Master server maintains the information about the cluster and manages the nodes

Nodes runs all the application containers.

OpenShift uses kubernetes Master/Node Architecture and provide some extra features.

In Kubernetes containers are created using components called pods.

**Routing Layer** : It provides easy DNS access and consistent endpoints to the applications.

**Builder Image:** It is a special container image that included applications and libraries for a given application.

**Integrated Container Registry** : Contains Application image and Builder Image.  
Application image creates application containers across various nodes and these nodes are connected to application service which is connected to routing layer.

**Virtualization**: In a single physical server,multiple virtual machines can run isolating different servers but each needed to have its own OS.Each VM has its own linux kernel.

\*SSH is a widely used remote access protocol

**\*Persistent storage** is any data **storage** device that retains data after power to that device is shut off. It is also sometimes referred to as non-volatile **storage**.

\*HostName:Label assigned to a computer connected to internet.

One of the key features of Ansible is that it uses YAML, a human-readable data-transport language, to describe all the work it does.

Ansible doesn’t use an agent on the systems it’s controlling; instead, it uses SSH as a transport mechanism and to execute remote commands.Therefore Ansible is only required in master.

**Fundamentals source:***OpenShift in Action- Jamie Duncan and John Osborne*

OpenShift-Installation :

**1.Install CentOS-7**

Centos can be installed on your machine.OpenShift was installed on CentOS in virtual machine

**2.Install Docker**

OpenShift works on container orchestration and provides other functionalities.Docker is required for OpenShift

To Instal on CentOSl:  **sudo yum install -y docker**

**3.Configure Docker Insecure registry**

This step is followed if a proper TLS (Transport Layer Security) certificate is not available.

On CentOS or RHEL, edit the file **/etc/docker/daemon.json** by adding these lines:

**{**

**"insecure-registries": ["172.30.0.0/16"]**

**}**

**4.Start Docker**

Before starting Docker, create a system group named **docker** and assign this group to your user so you can run Docker commands with your own user, without requiring root or sudo access. This allows you to create your OKD cluster using your own user.

For Example:

In my Case:

**$ sudo groupadd docker**

**$ sudo usermod -a -G docker sgoel**

If you are on root user then you need to logout and login through the above used User.

Run $id to ensure you are added in docker group.

**$ id**

**uid=1000(sgoel) gid=1000(sgoel) groups=1000(ricardo),10(wheel),1001(docker)**

**context=unconfined\_u:unconfined\_r:unconfined\_t:s0-s0:c0.c1023**

Verify Docker is Running $ docker version

Ensure That the insecure registries optionhas been enabled by running docker info

**$ docker info**

**... Skipping long output ...**

**Insecure Registries:**

**172.30.0.0/16**

**127.0.0.0/8**

**5.Open Firewall Ports**

Next, open firewall ports to ensure your OKD containers can communicate with the master API. By default, some distributions have the firewall enabled, which blocks required connectivity from the OKD containers to the master API. If your system has the firewall enabled, you need to add rules to allow communication on ports **8443/tcp** for the master API and **53/udp** for DNS resolution on the Docker bridge subnet.

**Obtain the subnet address by :**

**$ docker network inspect bridge | grep Subnet**

**"Subnet": "172.17.0.0/16",**

**Add a new zone by :**

**$ sudo firewall-cmd --permanent --new-zone okdlocal**

**Success**

**Include the Subnet address:**

**$ sudo firewall-cmd --permanent --zone okdlocal --add-source 172.17.0.0/16**

**Success**

**Add the rules to okdlocal zone:**

**$ sudo firewall-cmd --permanent --zone okdlocal --add-port 8443/tcp**

**success**

**$ sudo firewall-cmd --permanent --zone okdlocal --add-port 53/udp**

**success**

**$ sudo firewall-cmd --permanent --zone okdlocal --add-port 8053/udp**

**success**

**Reload the firewall to enable the rule:**

**$ sudo firewall-cmd --reload**

**success**

**Ensure that the new zone and rules are in place:**

**$ sudo firewall-cmd --zone okdlocal --list-sources**

**172.17.0.0/16**

**$ sudo firewall-cmd --zone okdlocal --list-ports**

**8443/tcp 53/udp 8053/udp**

**6.Download The okd tools:**

To deploy a local OKD cluster using **oc**, you need to download the OKD client tools package. For some distributions, like CentOS and Fedora, this package can be downloaded as an RPM from the official repositories. Please note that these packages may follow the distribution update cycle and usually are not the most recent version available.

For this tutorial, download the OKD client package directly from the official GitHub repository so you can get the most recent version available. At the time of writing, this was OKD v3.11.

Go to the [OKD downloads page](https://www.okd.io/download.html#oc-platforms) to get the link to the OKD tools for Linux, then download it with **wget:**

**$ cd ~/Downloads/**

**$ wget** [**https://github.com/openshift/origin/releases/download/v3.11.0/openshift-origin-client-tools-v3.11.0-0cbc58b-linux-64bit.tar.gz**](https://github.com/openshift/origin/releases/download/v3.11.0/openshift-origin-client-tools-v3.11.0-0cbc58b-linux-64bit.tar.gz)

**Uncompress the downloaded Package:**

**tar -xzvf openshift-origin-client-tools-v3.11.0-0cbc58b-linux-64bit.tar.gz**

**Make it easier to use oc system-wide:**

**$ sudo cp openshift-origin-client-tools-v3.11.0-0cbc58b-linux-64bit/oc /usr/local/bin/**

**Check oc command is running:**

**$ oc version**

**oc v3.11.0+0cbc58b**

**kubernetes v1.11.0+d4cacc0**

**features: Basic-Auth GSSAPI Kerberos SPNEGO**

**7.Start okd cluster**

**$ oc cluster up**

**... Skipping long output ...**

**OpenShift server started.**

**The server is accessible via web console at:**

**https://127.0.0.1:8443**

**You are logged in as:**

**User: developer**

**Password: <any value>**

**To login as administrator:**

**oc login -u system:admin**

**Access the cluster by :**

[**https://127.0.0.1:8443**](https://127.0.0.1:8443/)

**Check status of cluster by:**

**$ oc cluster status**

**Login as admin:**

**oc login -u system:admin**

**8.Errors Faced:**

**Error downloading packages:**

**wget-1.14-18.el7\_6.1.x86\_64: [Errno 256] No more mirrors to try.**

**Fix: sudo yum install wget**

**Error:TLS Handshake Error/Connection Timed out:**

**Fix: Run the command again.**

**Error:Unable to reach** [**https://127.0.0.1:8443**](https://127.0.0.1:8443/)

**Fix:**

**1.Enable port forwarding in virtual machine**

**2.Add the below firewall settings:**

**$firewall-cmd --zone=public --add-port=8443/tcp --permanent**

**$firewall-cmd --reload**

**Error:API Server Error:Get** [**https://127.0.0.1:8443/healthz?timeout=32s**](https://127.0.0.1:8443/healthz?timeout=32s)**:**

**Connect : connection refused ()**

**Probable Reason:**

**Lower Resources on PC (Weak VM):** [**https://github.com/openshift/origin/issues/20617#issuecomment-488709187**](https://github.com/openshift/origin/issues/20617#issuecomment-488709187)

**Error: Unable to access deployed app from host machine:**

**Connection refused.**