4/5/2022

RBI

Assignment

Abalfazl Meshki

Contents

Introduction:

This document is to describe the method that has been used to provide a method to satisfy the assignment.

The assignment consists of several parts which demand different method to satisfied.

Assignment Description:

In this scenario we are going to setup a flow to make the flow of deploying a web application automated and easy. In other words, using the Microservice concept and automation tools, users be able to install, configure and deploy all the requirements components of web application automatically.

Technology Stack:

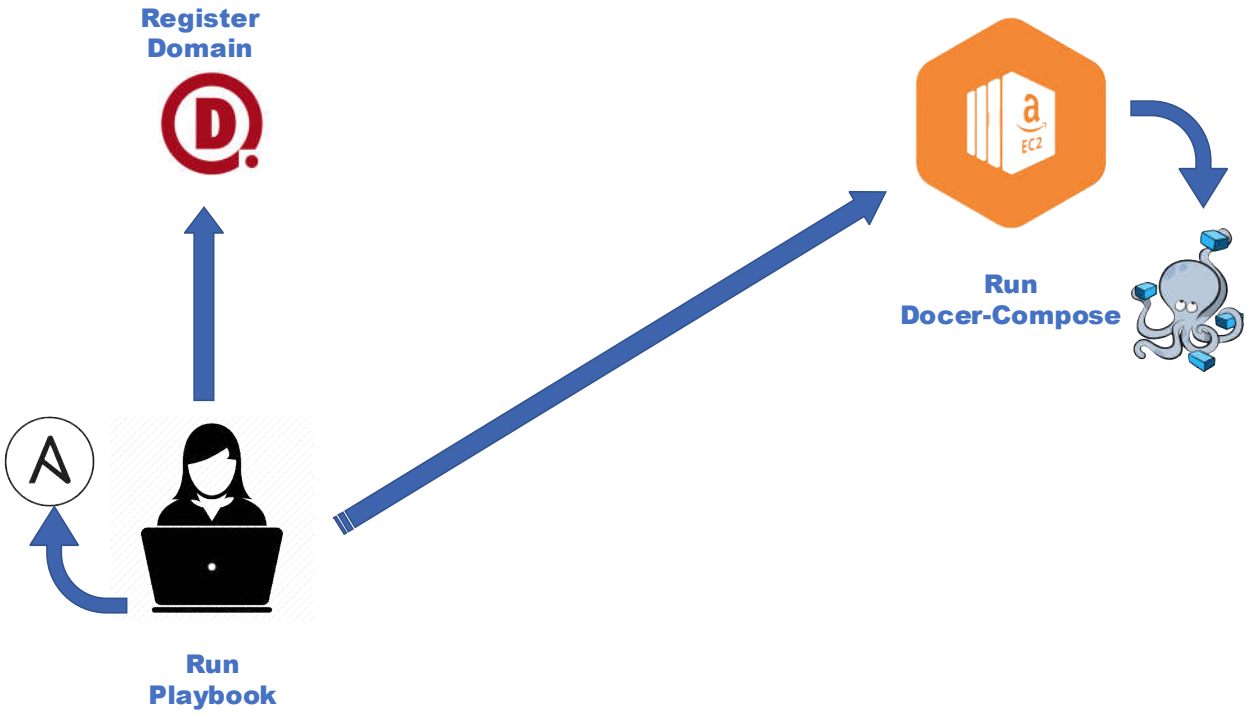
The technologies and tools which has been used for the assignment have been depicted below.

The work fellow consists of two section which are Microservice side and deployment side.

Docker-compose and docker are the tools which have been used for brining up the web applications. Two microservices have been deployed which one is responsible to web application and the other is responsible for providing reverse proxy service.

Ansible on the other hand, has been chosen for the sake of automation as it is responsible to install the prerequisites and then start the website.

- ✓ As the assignment demand to have a valid certificate signed by “Let’s encrypt” a valid domain and a server with public IP address has been demanded so “meshki.me” has been bought and the destination server is EC2 on AWS.



As you can see on the picture, a separated workstation has been chosen as Ansible Controller Node, a repo in GitHub provides all the manifests and an EC2 server works as Ansible Managed Node.

Docker-Compose:

As you can see on the docker compose, two Microservices has been deployed by building containers based on the latest version of NGINX containers.

Web Microservice:

This microservice would bring up the contents of a specific repository which has been mentioned on the Task document(https://github.com/dockersamples/linux_tweet_app). It listens on only port 80 and apart from installing some packages for the sake of demo and test, no more configuration has been done for it.

Proxy Microservice:

This microservice used the NINGX as the base image. It has “default.conf” file which configures the Nginx to works as the reverse proxy server and redirects incoming connections (www.meshki.me) to the “web microservice. It has a script for staring the microservice which would used the “certbot” to generate the valid certificate and the copy it to the destination which has been used by Nginx to offload the SSL connection.

Ansible:

This playbook consists of a host inventory and playbook. Inside the inventory file, IP address of the destination server should provide the service. For Ansible to connect to this server, required information such as the username to connect or the SSH key to use has been depicted.

The playbook YAML file consists of several task which first, update the operation system and then install the required packages such as git docker-ce and grab the docker-compose. Then it would checkout manifests. At the end it will build the images for docker-compose and bring up them.

MOP:

1) Lunch and instance on EC2 using a subnet which has Public-IP assignment available. The OS needs to be AWS Linux2. Let the instance to get ready and record the IP address which is going to be used on playbook.
2) Update the record of “www.meshki.me” or DNS server (mine is domain.com). and let the record be populated on DNS servers. (Using the dig or nslookup command, you can query DNS server.)
3) Make sure that ansible has been installed on the workstation and checkout the codes out of the GIT repo which has been mentioned previously.
4) Inside the “hosts” file, change the IP address in “meshki” group to the IP address of the created EC2 instance and make sure there is no connection restriction from the workstation to the Instance.
5) Using the following command (ansible-playbook -i hosts playbook.yml), start the playbook and check the following task to bring up the docker-compose. At the end, if all goes well, a website with valid certificate would be available on www.meshki.me.