

<b>Name: Bose. Vincent Bryan</b>	<b>Date Performed: 11/13/2024</b>
<b>Course/Section: CPE31S2</b>	<b>Date Submitted: 11/13/2024</b>
<b>Instructor: Engr. Robin Valenzuela</b>	<b>Semester and SY: 2024-2025</b>
<b>Activity 11: Containerization</b>	
<b>1. Objectives</b>	
Create a Dockerfile and form a workflow using Ansible as Infrastructure as Code (IaC) to enable Continuous Delivery process	
<b>2. Discussion</b>	
<p>Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications. By taking advantage of Docker's methodologies for shipping, testing, and deploying code quickly, you can significantly reduce the delay between writing code and running it in production.</p> <p>Source: <a href="https://docs.docker.com/get-started/overview/">https://docs.docker.com/get-started/overview/</a></p> <p>You may also check the difference between containers and virtual machines. Click the link given below.</p> <p>Source: <a href="https://docs.microsoft.com/en-us/virtualization/windowscontainers/about/containers-vs-vm">https://docs.microsoft.com/en-us/virtualization/windowscontainers/about/containers-vs-vm</a></p>	
<b>3. Tasks</b>	
<ol style="list-style-type: none"> <li>1. Create a new repository for this activity.</li> <li>2. Install Docker and enable the docker socket.</li> <li>3. Add to Docker group to your current user.</li> <li>4. Create a Dockerfile to install web and DB server.</li> <li>5. Install and build the Dockerfile using Ansible.</li> <li>6. Add, commit and push it to your repository.</li> </ol>	
<b>4. Output (screenshots and explanations)</b>	
<ol style="list-style-type: none"> <li>1. Create a repository for the activity.</li> </ol>	

Activities Firefox Web Browser Wed 08:06

Restore Session x BOSE-13/HOA-11.1-BOSE x

https://github.com/BOSE-13/HOA-11.1-BOSE

BOSE-13 / HOA-11.1-BOSE

Type to search

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

HOA-11.1-BOSE Public

Pin Unwatch 1 Fork 0 Star 0

main 1 Branch 0 Tags

Go to file + <> Code

BOSE-13 Initial commit fca9d86 · 15 minutes ago 1 Commit

README.md Initial commit 15 minutes ago

README

HOA-11.1-BOSE

About

No description, website, or topics provided.

Readme Activity 0 stars 1 watching 0 forks

Releases

No releases published [Create a new release](#)

Packages

No packages published

```
vbbosc@workstation:~$ git clone git@github.com:BOSE-13/HOA-11.1-BOSE.git
Cloning into 'HOA-11.1-BOSE'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.
```

2. Installed docker

```

vbbose@workstation:~$ sudo apt install -y docker.io
[sudo] password for vbbose:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  ieee-data liblvm7 python-certifi python-chardet python-jmespath
  python-kerberos python-libcloud python-lockfile python-netaddr
  python-openssl python-requests python-selinux python-simplejson
  python-urllib3 python-xlrd
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  bridge-utils containerd pigz runc ubuntu-fan
Suggested packages:
  aufs-tools btrfs-progs cgroupfs-mount | cgroup-lite debootstrap docker-doc
  rinse zfs-fuse | zfsutils
The following NEW packages will be installed:
  bridge-utils containerd docker.io pigz runc ubuntu-fan
0 upgraded, 6 newly installed, 0 to remove and 0 not upgraded.
Need to get 65.7 MB of archives.
After this operation, 292 MB of additional disk space will be used.
Get:1 http://ph.archive.ubuntu.com/ubuntu bionic/universe amd64 pigz amd64 2.4-1 [57.4 kB]
Get:2 http://ph.archive.ubuntu.com/ubuntu bionic/main amd64 bridge-utils amd64 1.5-15ubuntu1 [30.1 kB]
Get:3 http://ph.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 runc amd64 1.1.4-0ubuntu1-18.04.2 [3,822 kB]
Get:4 http://ph.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 containerd amd64 1.6.12-0ubuntu1-18.04.1 [31.5 MB]
Get:5 http://ph.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 docker.io amd64 20.10.21-0ubuntu1-18.04.3 [30.3 MB]
Get:6 http://ph.archive.ubuntu.com/ubuntu bionic/main amd64 ubuntu-fan all 0.12.10 [34.7 kB]
Fetched 65.7 MB in 40s (1,636 kB/s)
Preconfiguring packages ...
Selecting previously unselected package pigz.
(Reading database ... 180887 files and directories currently installed.)
Preparing to unpack .../0-pigz_2.4-1_amd64.deb ...
Unpacking pigz (2.4-1) ...
Selecting previously unselected package bridge-utils.
Preparing to unpack .../1-bridge-utils_1.5-15ubuntu1_amd64.deb ...
Unpacking bridge-utils (1.5-15ubuntu1) ...
Selecting previously unselected package runc.
Preparing to unpack .../2-runc_1.1.4-0ubuntu1-18.04.2_amd64.deb ...
Unpacking runc (1.1.4-0ubuntu1-18.04.2) ...
Selecting previously unselected package containerd.
Preparing to unpack .../3-containerd_1.6.12-0ubuntu1-18.04.1_amd64.deb ...
Unpacking containerd (1.6.12-0ubuntu1-18.04.1) ...
Selecting previously unselected package docker.io.
Preparing to unpack .../4-docker.io_20.10.21-0ubuntu1-18.04.3_amd64.deb ...
Unpacking docker.io (20.10.21-0ubuntu1-18.04.3) ...
Selecting previously unselected package ubuntu-fan.
Preparing to unpack .../5-ubuntu-fan_0.12.10_all.deb ...
Unpacking ubuntu-fan (0.12.10) ...
Setting up runc (1.1.4-0ubuntu1-18.04.2) ...
Setting up containerd (1.6.12-0ubuntu1-18.04.1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /lib/systemd/system/containerd.service.
Setting up bridge-utils (1.5-15ubuntu1) ...


```

```

vbbose@workstation:~$ systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: e
   Active: active (running) since Wed 2024-11-13 08:00:28 +08; 21s ago
     Docs: https://docs.docker.com
    Main PID: 4105 (dockerd)
      Tasks: 9
     CGroup: /system.slice/docker.service
             └─4105 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/contain
lines 1-8/8 (END)

```

### 3. Add to Docker group to your current user

```
Open  *Dockerfile
~/HOA-11.1-BOSE/Ubuntu_Docker

FROM ubuntu:latest
MAINTAINER BOSE <qvbbbose@tip.edu.ph>

ARG DEBIAN_FRONTEND=noninteractive

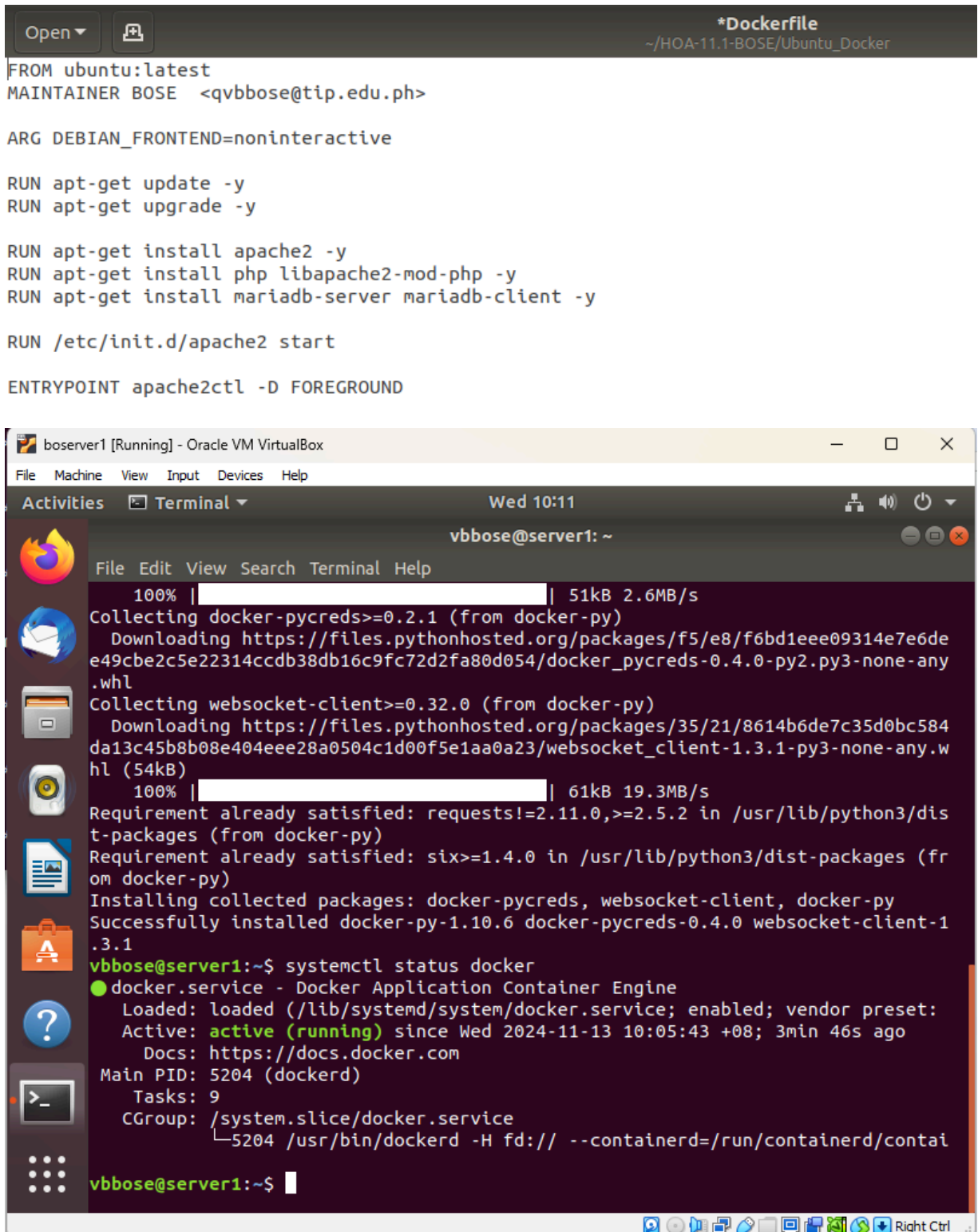
RUN apt-get update -y
RUN apt-get upgrade -y

RUN apt-get install apache2 -y
RUN apt-get install php libapache2-mod-php -y
RUN apt-get install mariadb-server mariadb-client -y

RUN /etc/init.d/apache2 start

ENTRYPOINT apache2ctl -D FOREGROUND
```

4.



The image shows two parts: a Dockerfile at the top and a terminal window at the bottom. The Dockerfile is for an Ubuntu-based container with Apache2 and MariaDB installed. The terminal window shows the installation of Docker on a server named 'server1'.

```
*Dockerfile
~/HOA-11.1-BOSE/Ubuntu_Docker

FROM ubuntu:latest
MAINTAINER BOSE <qvbbbose@tip.edu.ph>

ARG DEBIAN_FRONTEND=noninteractive

RUN apt-get update -y
RUN apt-get upgrade -y

RUN apt-get install apache2 -y
RUN apt-get install php libapache2-mod-php -y
RUN apt-get install mariadb-server mariadb-client -y

RUN /etc/init.d/apache2 start

ENTRYPOINT apache2ctl -D FOREGROUND
```


Terminal output:

```
boserver1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal Wed 10:11
vbbose@server1: ~

100% | 51kB 2.6MB/s
Collecting docker-pycreds>=0.2.1 (from docker-py)
  Downloading https://files.pythonhosted.org/packages/f5/e8/f6bd1eee09314e7e6de
e49cbe2c5e22314ccdb38db16c9fc72d2fa80d054/docker_pycreds-0.4.0-py2.py3-none-any
.whl
Collecting websocket-client>=0.32.0 (from docker-py)
  Downloading https://files.pythonhosted.org/packages/35/21/8614b6de7c35d0bc584
da13c45b8b08e404eee28a0504c1d00f5e1aa0a23/websocket_client-1.3.1-py3-none-any.w
hl (54kB)
100% | 61kB 19.3MB/s
Requirement already satisfied: requests!=2.11.0,>=2.5.2 in /usr/lib/python3/dist-
packages (from docker-py)
Requirement already satisfied: six>=1.4.0 in /usr/lib/python3/dist-packages (fr
om docker-py)
Installing collected packages: docker-pycreds, websocket-client, docker-py
Successfully installed docker-py-1.10.6 docker-pycreds-0.4.0 websocket-client-1
.3.1
vbbose@server1:~$ systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset:
   Active: active (running) since Wed 2024-11-13 10:05:43 +08; 3min 46s ago
     Docs: https://docs.docker.com
   Main PID: 5204 (dockerd)
     Tasks: 9
    CGroup: /system.slice/docker.service
            └─5204 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/contai

vbbose@server1:~$
```

5. This is the yml to run the playbook.

Open ▾

docker.yml  
~/HOA-11.1-BOSE

```
---  
- hosts: all  
  become: true  
  pre_tasks:  
    - name: Install Updates (Ubuntu)  
      tags: always  
      apt:  
        update_cache: yes  
        changed_when: false  
        when: ansible_distribution == "Ubuntu"  
  
- hosts: Ubuntu  
  become: true  
  roles:  
    - Ubuntu
```

OUTPUTS:

```

vbose@workstation:~/HOA-11.1-BOSE$ ansible-playbook --ask-become-pass docker.yml
BECOME password:

PLAY [all] *****

TASK [Gathering Facts] *****
[DEPRECATION WARNING]: Distribution Ubuntu 18.04 on host 192.168.56.106 should
use /usr/bin/python3, but is using /usr/bin/python for backward compatibility
with prior Ansible releases. A future Ansible release will default to using the
discovered platform python for this host. See https://docs.ansible.com/ansible
/2.9/reference_appendices/interpreter_discovery.html for more information. This
feature will be removed in version 2.12. Deprecation warnings can be disabled
by setting deprecation_warnings=False in ansible.cfg.
ok: [192.168.56.106]

TASK [Install Updates (Ubuntu)] *****
ok: [192.168.56.106]

PLAY [Ubuntu] *****

TASK [Gathering Facts] *****
ok: [192.168.56.106]

TASK [Ubuntu : Install Docker] *****
ok: [192.168.56.106]

TASK [Start the Docker Service in Ubuntu] *****
ok: [192.168.56.106]

TASK [Ubuntu : Ensure group docker exists] *****
ok: [192.168.56.106]

TASK [Ubuntu : Adding the current user to the docker group] *****
ok: [192.168.56.106]

TASK [Ubuntu : Create a docker directory] *****
ok: [192.168.56.106]

TASK [Copy Dockerfile to Ubuntu] *****
ok: [192.168.56.106]

TASK [Ubuntu : Build Docker Image] *****
[WARNING]: Please specify build.path instead of path. The path option has been
renamed and will be removed in Ansible 2.12.
[WARNING]: The value of the "source" option was determined to be "build".
Please set the "source" option explicitly. Autodetection will be removed in
Ansible 2.12.
[DEPRECATION WARNING]: Param 'path' is deprecated. See the module docs for more
information. This feature will be removed in version 2.12. Deprecation
warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
changed: [192.168.56.106]

```

Docker is running in server1

```

vbbose@server1:~$ docker images
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
apache-mariadb-image latest      79cc28f4a62f  2 minutes ago  580MB
ubuntu              latest      59ab366372d5  4 weeks ago   78.1MB

```

6. “git push” successful.

```

vbbose@workstation:~/HOA-11.1-BOSE$ git add *
vbbose@workstation:~/HOA-11.1-BOSE$ git commit -m "DONE"
[main 5ba2baf] DONE
 5 files changed, 93 insertions(+)
 create mode 100644 Ubuntu_Docker/Dockerfile
 create mode 100644 ansible.cfg
 create mode 100644 docker.yml
 create mode 100644 inventory
 create mode 100644 roles/Ubuntu/tasks/main.yml
vbbose@workstation:~/HOA-11.1-BOSE$ git push
Counting objects: 11, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (11/11), 1.43 KiB | 1.43 MiB/s, done.
Total 11 (delta 0), reused 0 (delta 0)
To github.com:BOSE-13/HOA-11.1-BOSE.git
   fca9d86..5ba2baf  main -> main

```

### Reflections:

Answer the following:

1. What are the benefits of implementing containerizations?
  - Implementing containerization in a Linux environment like Ubuntu offers significant advantages for developers, system administrators, and organizations. Key benefits include process isolation, which enhances security and prevents application conflicts, as containers share the same host kernel while remaining separate. Portability is another major advantage, as containers encapsulate applications and their dependencies into a single image, ensuring consistent performance across different environments and infrastructures. Additionally, container orchestration platforms such as Kubernetes facilitate easy scalability, allowing applications to be adjusted quickly in response to varying demands. Overall, containerization improves resource efficiency, promotes rapid deployment, simplifies dependency management, and fosters collaboration among teams, making it a valuable approach in modern software development.

### Conclusions:



Integrating Docker and Ansible into a Continuous Delivery (CD) strategy signifies a major improvement in software deployment. Docker facilitates the creation of portable containers that ensure consistency across environments, while Ansible provides an Infrastructure as Code framework for automating infrastructure management. This combination streamlines workflows, reduces deployment times, and promotes collaboration within development and operations teams, aligning with DevOps principles. Ultimately, using Docker and Ansible enhances organizations' ability to deliver reliable and efficient software, improving operational resilience, product quality, and customer satisfaction in a competitive landscape.

**GITHUB LINK: <https://github.com/BOSE-13/HOA-11.1-BOSE>**