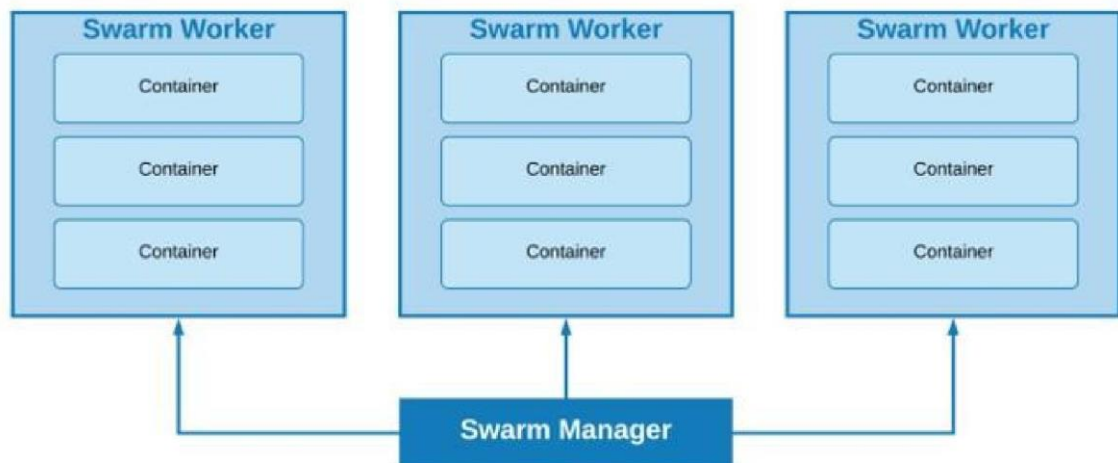


DOCKER SWARM

What is Docker Swarm

1. Docker swarm is an orchestration service it is used to manage multiple containers at the same time
2. It is implemented by cluster
3. Docker Engine helps to create docker swarm
4. Docker swarm is having two nodes
 - Manager Node
 - Worker Node

Docker Swarm Architecture



Step 1: Launch one instance and connect with the created instance.

- Switch to the root user by using the command "**sudo su -**".

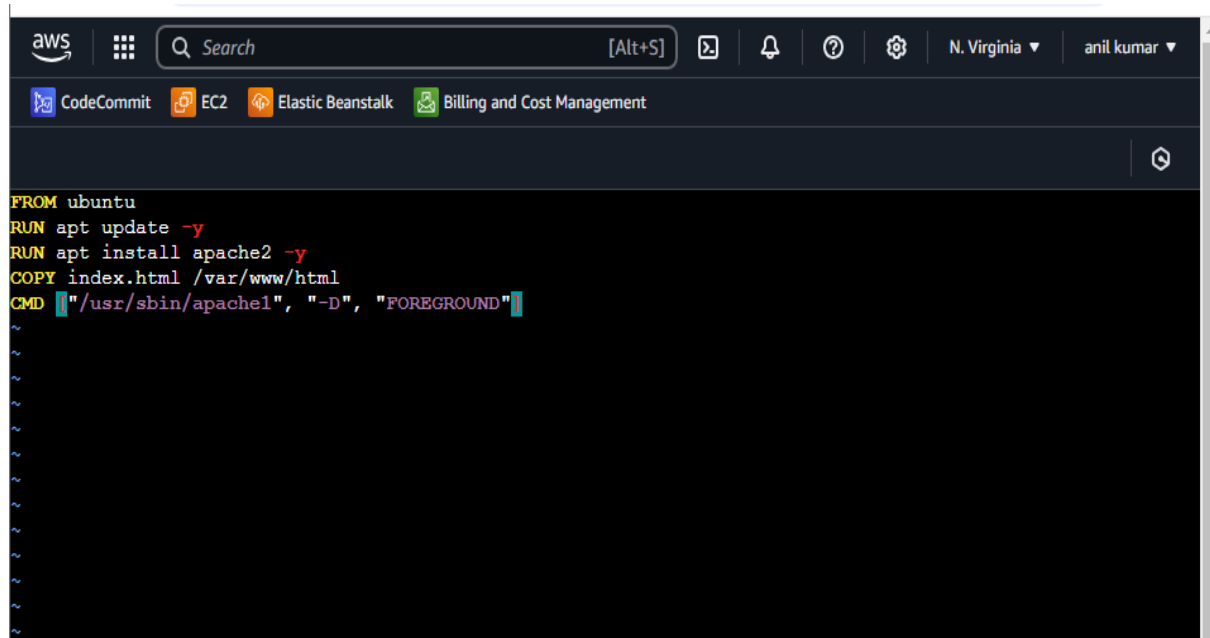
Step 2: Install the docker in a terminal and start the docker and check the status by using the following commands.

- **yum install docker** - Install docker
- **systemctl start docker** -To start docker

Step 4: Perform the command called “**vi index.html**”. Here insert the html code in that as shown in the below figure.

- **systemctl status docker** -To check the status of docker

Step 3:Create the docker file by using the command “**vi Dockerfile**” in that you may perform the following instruction as shown in the below figure.

A screenshot of an AWS CloudShell terminal window. The top bar shows the AWS logo, a search bar, and navigation icons. Below the bar, there are links to CodeCommit, EC2, Elastic Beanstalk, and Billing and Cost Management. The terminal itself has a dark background with yellow and red text. It shows the following commands: FROM ubuntu, RUN apt update -y, RUN apt install apache2 -y, COPY index.html /var/www/html, and CMD ["/usr/sbin/apache1", "-D", "FOREGROUND"]. The cursor is at the end of the CMD line.

```
FROM ubuntu
RUN apt update -y
RUN apt install apache2 -y
COPY index.html /var/www/html
CMD ["/usr/sbin/apache1", "-D", "FOREGROUND"]
```

Step 4 : Perform the command called “**vi index.html**”. Here insert the html code in that as shown in the below figure.

Step 6: Switch to the docker hub and create on repository.

- Switch to the terminal perform the command called “**docker tag image1 anilvasa/repo**”

Step 7: Perform the command “**docker login**”.

- Here have to provide the username and password of a dockerhub account, Example.
- Username : *****
- Password :*****

Step 8 : Push the image into your docker hub repository by using the following command.

- **docker tag image1 anilvasa/repo**

Step 9 : Initalize the swarm docker by using the following command.

- “**docker swarm init**” - Used to create the Manager node and it generates the token as shown in the below figure.

```
[root@ip-172-31-4-167 ~]# docker swarm init
Swarm initialized: current node (kw364phlklih0y5q5va3e66) is now a manager.

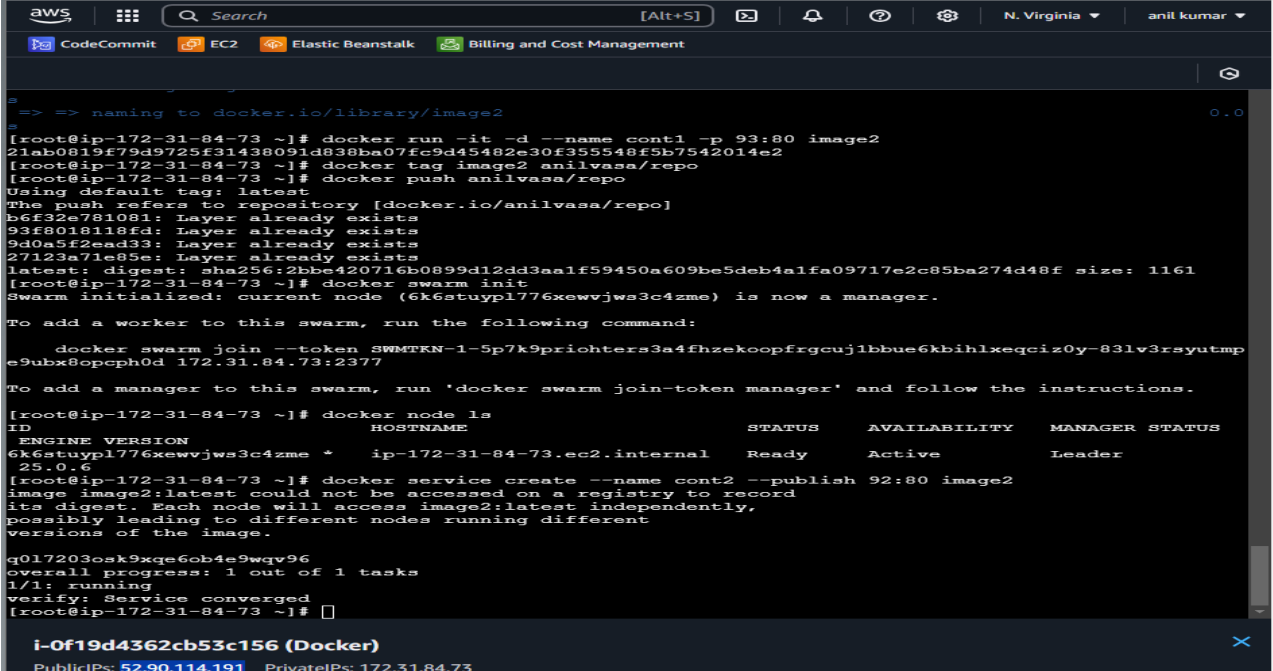
To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTEN-1-1n3h4tjnxdsfw3qzeo9nla9ze6bszi24poe737z17rj6yqsxlw-0ykcwbha11hgqoa50x114ve4q 172.31.4.167:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.
```

Step 10: Create the service by using the following command.

- “docker service create --name cont2 --publish 92:80 image2”.
- Here the service is created as well as the container also created as shown in the given figure.



```
[root@ip-172-31-84-73 ~]# docker run -it -d --name cont1 -p 93:80 image2
21ab0819f79d9725f31438091d838ba07fc9d45482e30f355548f5b7542014e2
[root@ip-172-31-84-73 ~]# docker tag image2 anilvasa/repo
[root@ip-172-31-84-73 ~]# docker push anilvasa/repo
Using default tag: latest
The push refers to repository [docker.io/anilvasa/repo]
b6f32e781081: Layer already exists
93f8018118fd: Layer already exists
9d0a5f2ead33: Layer already exists
27123a71e85e: Layer already exists
latest: digest: sha256:2bbe420716b0899d12dd3aa1f59450a609be5deb4a1fa09717e2c85ba274d48f size: 1161
[root@ip-172-31-84-73 ~]# docker swarm init
Swarm initialized: current node (6k6stuypl776xewvjs3c4zme) is now a manager.

To add a worker to this swarm, run the following command:

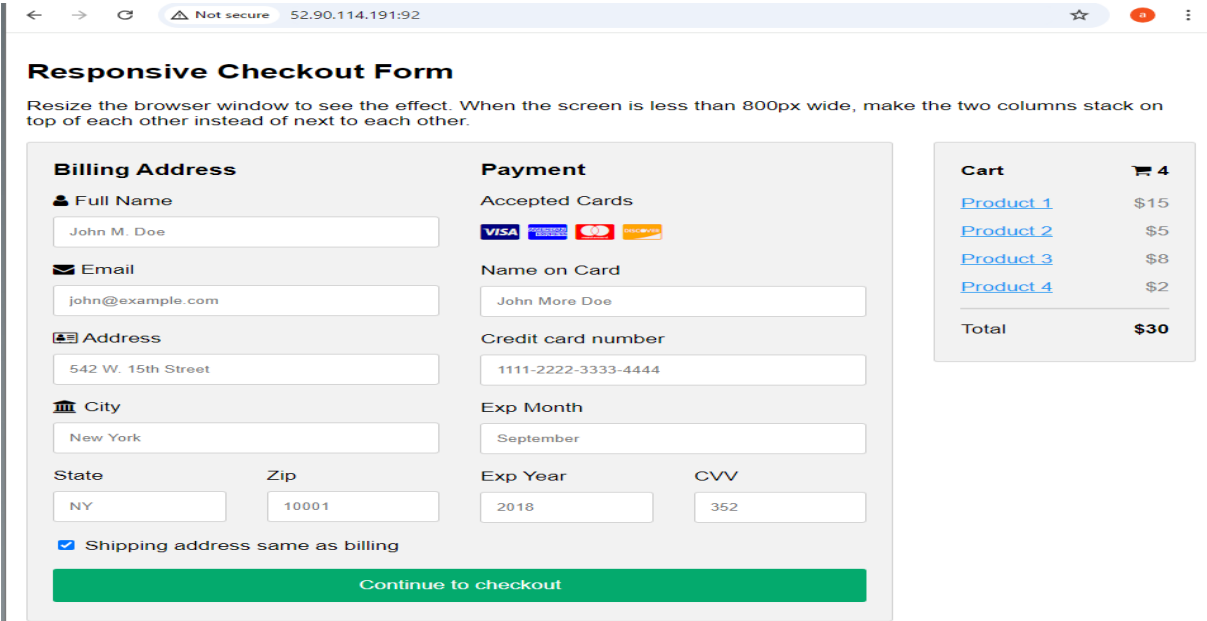
    docker swarm join --token SWMTKN-1-5p7k9prioshters3a4fhzekoopfrgcujlbbue6kbihlxeqciz0y-83lv3rsyutmp
e9ubx8opcph0d 172.31.84.73:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

[root@ip-172-31-84-73 ~]# docker node ls
ID                HOSTNAME                STATUS      AVAILABILITY    MANAGER STATUS
ENGINE VERSION
6k6stuypl776xewvjs3c4zme *   ip-172-31-84-73.ec2.internal   Ready      Active           Leader
25.0.6
[root@ip-172-31-84-73 ~]# docker service create --name cont2 --publish 92:80 image2
image image2:latest could not be accessed on a registry to record
its digest. Each node will access image2:latest independently,
possibly leading to different nodes running different
versions of the image.
q017203osk9xqe6ob4e9wqv96
overall progress: 1 out of 1 tasks
1/1: running
verify: Service converged
[root@ip-172-31-84-73 ~]#
```

i-Of19d4362cb53c156 (Docker)
PublicIPs: 52.90.114.191 PrivateIPs: 172.31.84.73

- Copy the public IP and enter in a google with the port number you can access output as shown in the figure.



← → ↺ ⚠ Not secure 52.90.114.191:92 ☆ 🔴 ⋮

Responsive Checkout Form

Resize the browser window to see the effect. When the screen is less than 800px wide, make the two columns stack on top of each other instead of next to each other.

Billing Address

Full Name

John M. Doe

Email

john@example.com

Address

542 W. 15th Street

City

New York

State

NY

Zip

10001

Shipping address same as billing

☒

Continue to checkout

Payment

Accepted Cards

VISA MASTERCARD AMEX DISCOVER

Name on Card

John More Doe

Credit card number

1111-2222-3333-4444

Exp Month

September

Exp Year

2018

CVV

352

Cart

4

Product 1

\$15

Product 2

\$5

Product 3

\$8

Product 4

\$2

Total

\$30

