DOCKER VOLUME

What is Docker Volume

1. When we create a container then by default volume will be

created

2. Volume is a directory inside of our container

3. If we stop the container then also, we can access the volume

4. Volume can be created in one container

5. While creating the container only we can attach the created

volume to that container

How many ways to map the volumes

There are two ways to map the volumes

1. Container ---Container

2. Host ---Container

- There are two ways to create volumes
  1. Automation
  2. Manual

Automation process

1. Install the docker

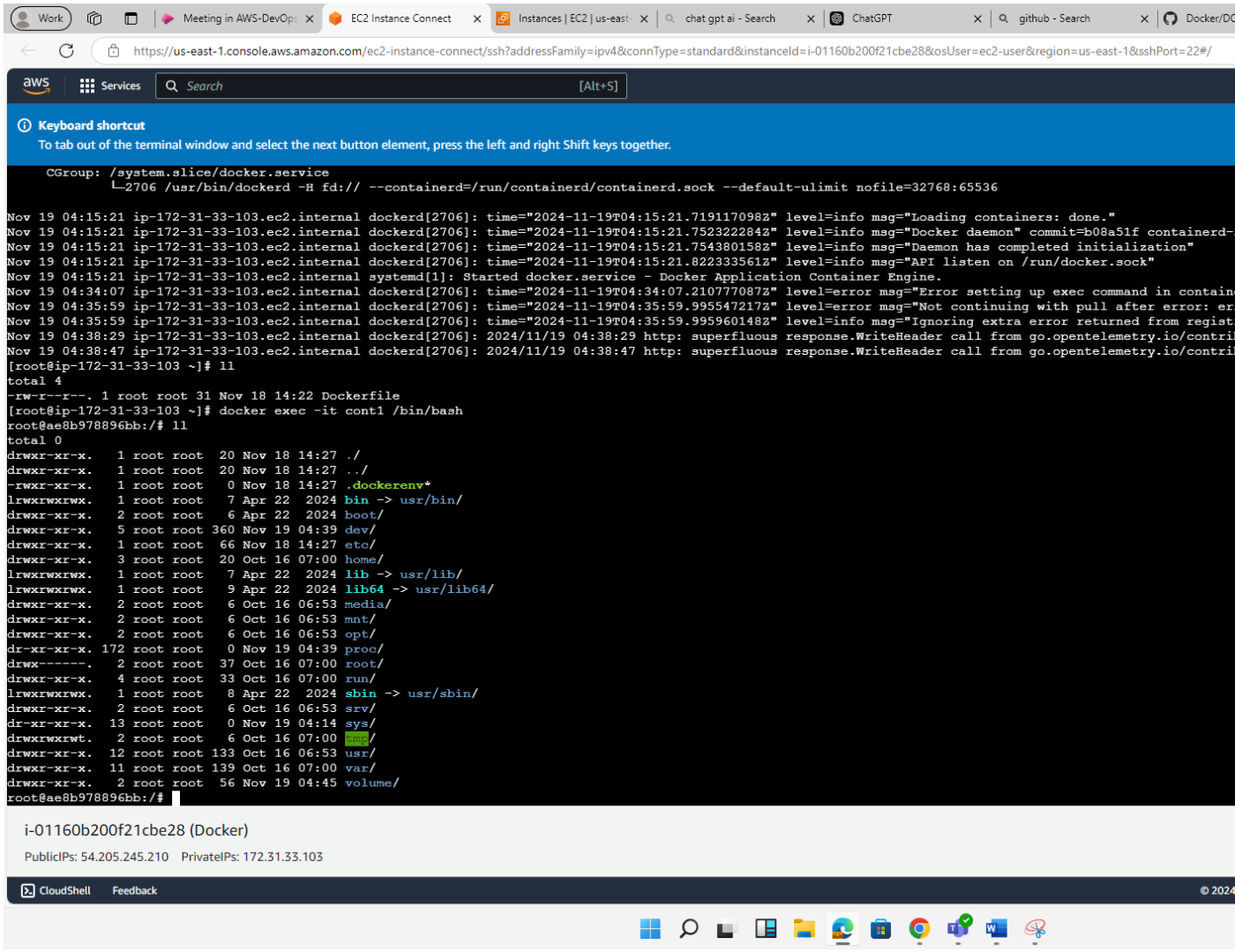
- yum install docker -y ---To install docker packages
- systemctl start docker ----To start docker service
- systemctl status docker---To check status of docker

2. Create Dockerfile
   vi Dockerfile

   # FROM ubuntu
   #VOLUME [“/volume”]
   :wq!

3. Build the image by command **Docker build -t image1** It is used to build the image
4. Create the container to the image1
   Docker run -it --name cont-1
   image1
   It can create the container
   with the volume

Volume is created in container as shown in below figure



➢ To share that created volume for container to container we are having the command

#docker run -it --name cont2 --privileged=true --volumes-from cont1 ubuntu

```
lrwxrwxrwx.   1 root root    7 Apr 22  2024 bin -> usr/bin/
drwxr-xr-x.   2 root root    6 Apr 22  2024 boot/
drwxr-xr-x.  11 root root 2820 Nov 19 04:49 dev/
drwxr-xr-x.   1 root root   66 Nov 19 04:49 etc/
drwxr-xr-x.   3 root root   20 Oct 16 07:00 home/
lrwxrwxrwx.   1 root root    7 Apr 22  2024 lib -> usr/lib/
lrwxrwxrwx.   1 root root    9 Apr 22  2024 lib64 -> usr/lib64/
drwxr-xr-x.   2 root root    6 Oct 16 06:53 media/
drwxr-xr-x.   2 root root    6 Oct 16 06:53 mnt/
drwxr-xr-x.   2 root root    6 Oct 16 06:53 opt/
dr-xr-xr-x. 174 root root    0 Nov 19 04:49 proc/
drwx------.   2 root root   37 Oct 16 07:00 root/
drwxr-xr-x.   4 root root   33 Oct 16 07:00 run/
lrwxrwxrwx.   1 root root    8 Apr 22  2024 sbin -> usr/sbin/
drwxr-xr-x.   2 root root    6 Oct 16 06:53 srv/
dr-xr-xr-x.  13 root root    0 Nov 19 04:14 sys/
drwxrwxrwt.   2 root root    6 Oct 16 07:00 tmp/
drwxr-xr-x.  12 root root  133 Oct 16 06:53 usr/
drwxr-xr-x.  11 root root  139 Oct 16 07:00 var/
drwxr-xr-x.   2 root root   56 Nov 19 04:45 volume/
root@eda44741500d:/# cd volume
root@eda44741500d:/volume# ll
total 0
drwxr-xr-x. 2 root root 56 Nov 19 04:45 ./
drwxr-xr-x. 1 root root 20 Nov 19 04:49 ../
-rw-r--r--. 1 root root  0 Nov 19 04:45 a1
-rw-r--r--. 1 root root  0 Nov 19 04:45 a2
-rw-r--r--. 1 root root  0 Nov 19 04:45 a3
-rw-r--r--. 1 root root  0 Nov 19 04:45 a4
-rw-r--r--. 1 root root  0 Nov 19 04:45 a5
root@eda44741500d:/volume#
```

i-01160b200f21cbe28 (Docker)

PublicIPs: 54.205.245.210 PrivateIPs: 172.31.33.103

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie prefere

```
  Main PID: 2706 (dockerd)
     Tasks: 10
    Memory: 117.3M
       CPU: 2.787s
    CGroup: /system.slice/docker.service
            └─2706 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock --default-ulimit nofil

Nov 19 04:15:21 ip-172-31-33-103.ec2.internal dockerd[2706]: time="2024-11-19T04:15:21.719117098Z" level=info msg
Nov 19 04:15:21 ip-172-31-33-103.ec2.internal dockerd[2706]: time="2024-11-19T04:15:21.752322284Z" level=info msg
Nov 19 04:15:21 ip-172-31-33-103.ec2.internal dockerd[2706]: time="2024-11-19T04:15:21.754380158Z" level=info msg
Nov 19 04:15:21 ip-172-31-33-103.ec2.internal dockerd[2706]: time="2024-11-19T04:15:21.822333561Z" level=info msg
Nov 19 04:15:21 ip-172-31-33-103.ec2.internal systemd[1]: Started docker.service - Docker Application Container E
Nov 19 04:34:07 ip-172-31-33-103.ec2.internal dockerd[2706]: time="2024-11-19T04:34:07.210777087Z" level=error ms
Nov 19 04:35:59 ip-172-31-33-103.ec2.internal dockerd[2706]: time="2024-11-19T04:35:59.995547217Z" level=error ms
Nov 19 04:35:59 ip-172-31-33-103.ec2.internal dockerd[2706]: time="2024-11-19T04:35:59.995960148Z" level=info msg
Nov 19 04:38:29 ip-172-31-33-103.ec2.internal dockerd[2706]: 2024/11/19 04:38:29 http: superfluous response.Write
Nov 19 04:38:47 ip-172-31-33-103.ec2.internal dockerd[2706]: 2024/11/19 04:38:47 http: superfluous response.Write
[root@ip-172-31-33-103 ~]# docker ps
CONTAINER ID   IMAGE         COMMAND       CREATED         STATUS         PORTS     NAMES
eda44741500d   ubuntu        "/bin/bash"   3 hours ago     Up 3 hours               cont2
e8b978896bb    imagevolume   "/bin/bash"   18 hours ago    Up 3 hours               cont1
[root@ip-172-31-33-103 ~]#
```

i-01160b200f21cbe28 (Docker)

PublicIPs: 54.205.245.210 PrivateIPs: 172.31.33.103

Manual process

- To create volume manually for the container by using command

#docker run -it --name new cont name -v/volume name image name

Host to container

create the volume in host by using command

#docker volume create volume name

. If we want to add files in host volume we need to inspect the volume, there you see the path

var/lib/docker/volume/_data This is path to add files

To share the host volume to the container the command is

docker run -it --name new cont name --mount source=volume name,destination=new volume name image name