**Docker Volume**

A Docker volume is a managed storage mechanism provided by Docker to persist data used by Docker containers. It allows data to be stored separately from the container's filesystem and lifecycle, enabling data persistence even when containers are stopped, removed, or updated. This is important for storing configuration files, logs, databases, and other data that need to persist across multiple container instances.

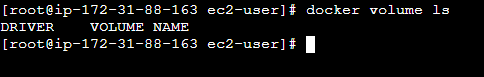
**Features of Docker Volumes:**

1. **Persistence:** Volumes allow data to persist beyond the lifecycle of a container.
2. **Isolation**: Volumes are isolated from the container's filesystem, providing a separate space for data storage.
3. **Portability**: Volumes can be moved between containers and even between hosts in a Docker Swarm environment.
4. **Manageability**: Docker provides commands to create, inspect, and manage volumes easily.

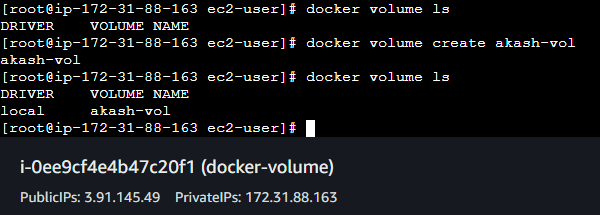
**Types of Docker Volumes:**

1. **Named Volumes:** These are explicitly created and managed using Docker CLI commands. Named volumes are given a specific name and can be reused across multiple containers.
2. **Anonymous Volumes**: These are automatically created by Docker when you use the -v flag without specifying a volume name. They are typically used for short-lived data.
3. **Bind Mounts:** Unlike volumes, bind mounts map a host directory or file to a container. They are useful for development, debugging, and other scenarios where you want to share a specific directory or file from the host with the container.

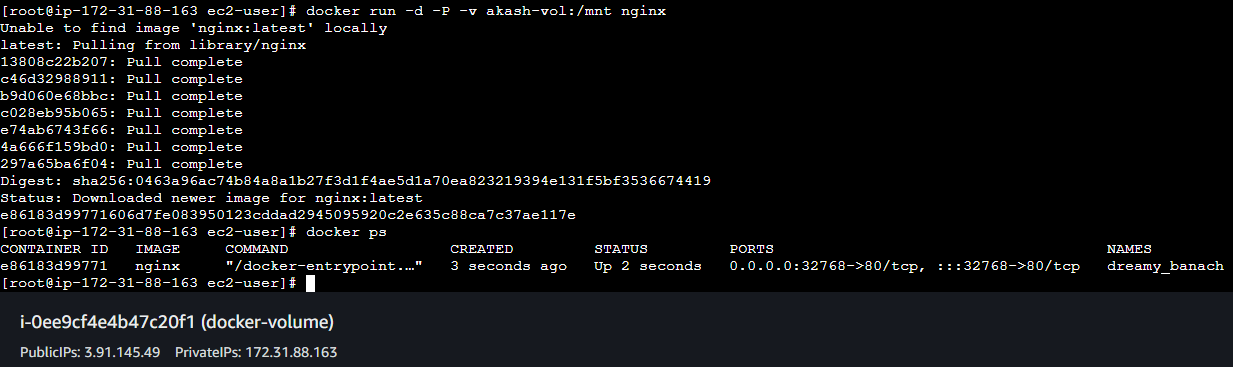
* Hit command “docker volume ls” to list all the volumes.



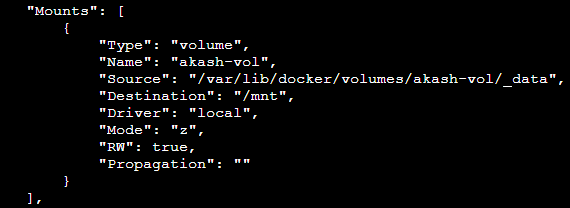
* We don’t have any volume present.
* We will have to create a volume.
* Hit command “docker volume create volume\_name”.
* After creating a volume hit command “docker volume ls” to list all the created volumes.



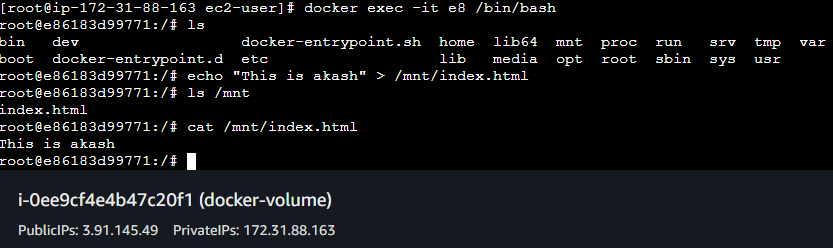
* Now the volume is created on the location /var/lib/docker/volume.
* Now we need to attach the volume to the container.
* Hit command “docker run -d -P -v akash-vol:/mnt nginx”.
* This command will pull the nginx image and create a container by running the image in detach mode and will attach the volume akash-vol to the container.



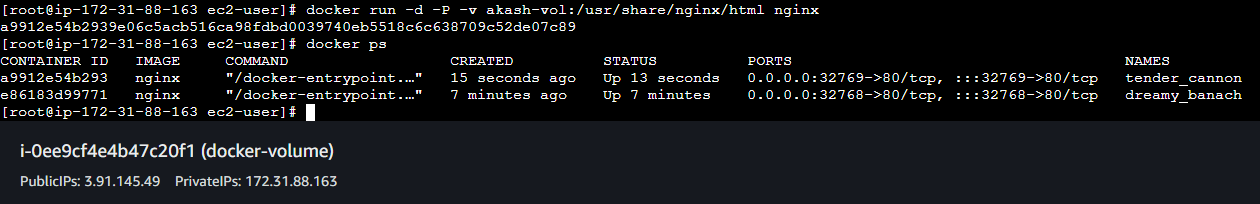
* The volume is attached to the container.
* We can verify it by “docker insepct container\_id”.



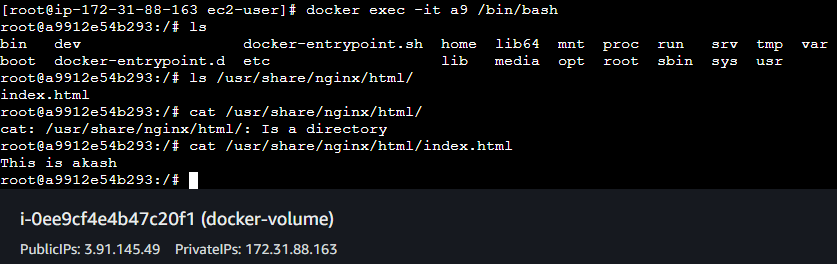
* Now we will enter the container and create a file in the volume called index.html.
* Hit command “docker exec -it container\_id /bin/bash”.

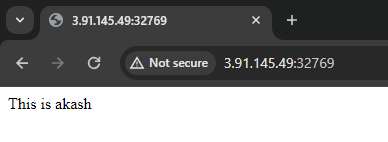


* Exit from the container.
* Now we will attach the same volume to another container.
* Hit command “docker run -d -P -v akash-vol:/usr/share/nginx/html nginx”.

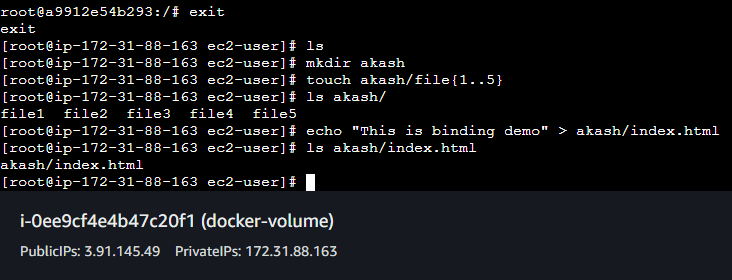


* Enter the container to check if the file is present or not which we have created.





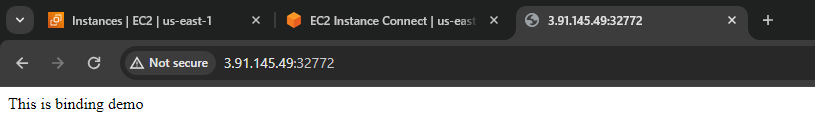
* Our file is successfully added to the html.
* That is our html directory is now attached to the volume we have created.
* Any changes in the html directory will reflect in the volume.
* Now exit from the container.
* Create a directory.
* Add some files in the directory.



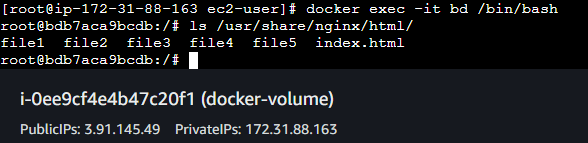
* Now we have to bind the directory named akash to the container.
* Hit command “docker run -d -P -v akash:/usr/share/nginx/html nginx”.



* Now hit the IP of the instance with the port number.



* Enter the container and check the html directory all the files should be present there.



* Try removing the files.
* Hit command “rm -rf /usr/share/nginx/html/file1”.
* Exit from the container.
* Check the bind directory.
* The file should be removed from here also.