KPLABS Course

Certified Kubernetes Application Developer

State Persistence

ISSUED BY

Zeal Vora

REPRESENTATIVE

instructors@kplabs.in

Module 1: Docker Volumes

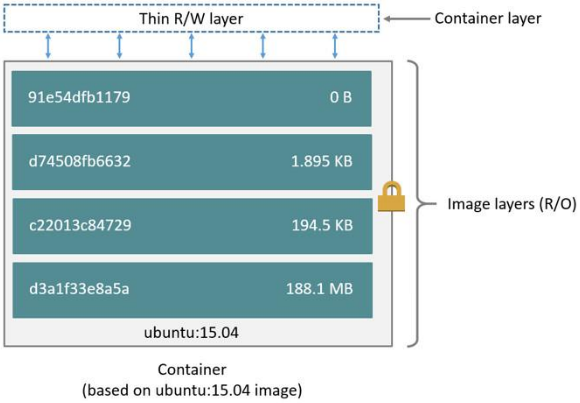
3.1 Challenges with files in Container Writable Layer

By default, all files created inside a container are stored on a writable container layer. This means that:

The data doesn't persist when that container no longer exists, and it can be difficult to get the data out of the container if another process needs it.

Writing into a container's writable layer requires a storage driver to manage the filesystem. The storage driver provides a union filesystem, using the Linux kernel.

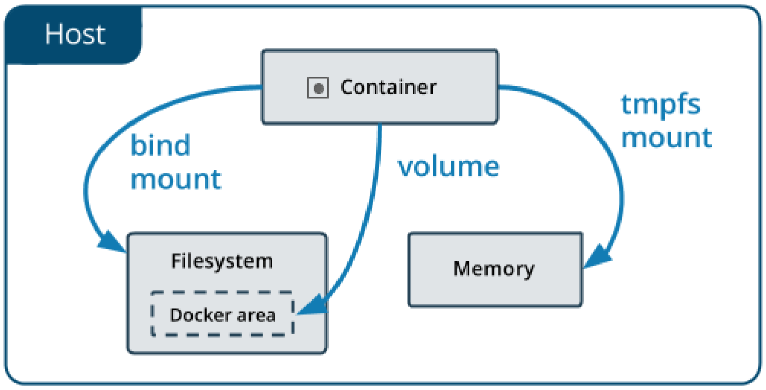
This extra abstraction reduces performance as compared to using data volumes, which write directly to the host filesystem.



3.2 Ideal Approach for Persistent Data

Docker has two options for containers to store files in the host machine, so that the files are persisted even after the container stops: volumes, and bind mounts.

If you're running Docker on Linux you can also use a tmpfs mount.



3.3 Important Pointers to Remember:

A given volume can be mounted into multiple containers simultaneously.

When no running container is using a volume, the volume is still available to Docker and is not removed automatically.

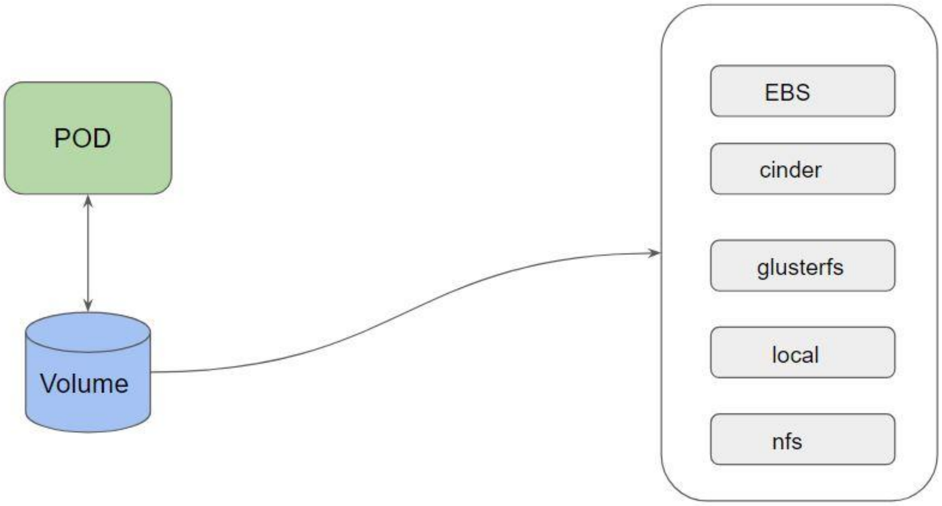
When you mount a volume, it may be named or anonymous. Anonymous volumes are not given an explicit name when they are first mounted into a container, so Docker gives them a random name that is guaranteed to be unique within a given Docker host.

Module 2: Volumes in Kubernetes

On-disk files in a Container are ephemeral.

When there are multiple containers who want to share the same data, it becomes a challenge.

One of the benefits of Kubernetes is that it supports multiple types of volumes.



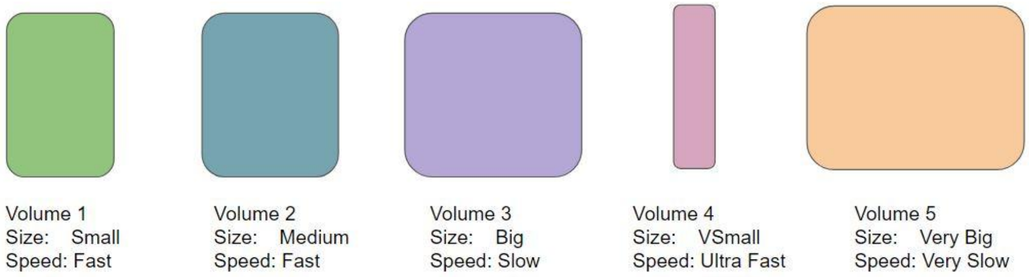
Module 3: PersistentVolume and PersistentVolumeClaim

3.1 PersistentVolume (PV)

A PersistentVolume (PV) is a piece of storage in the cluster that has been provisioned by an administrator or dynamically provisioned using Storage Classes

Every Volume which is created can be of different type.

This can be taken care by the Storage Administrator / Ops Team



3.2 PersistentVolumeClaim (PVC)

A PersistentVolumeClaim is a request for the storage by a user.

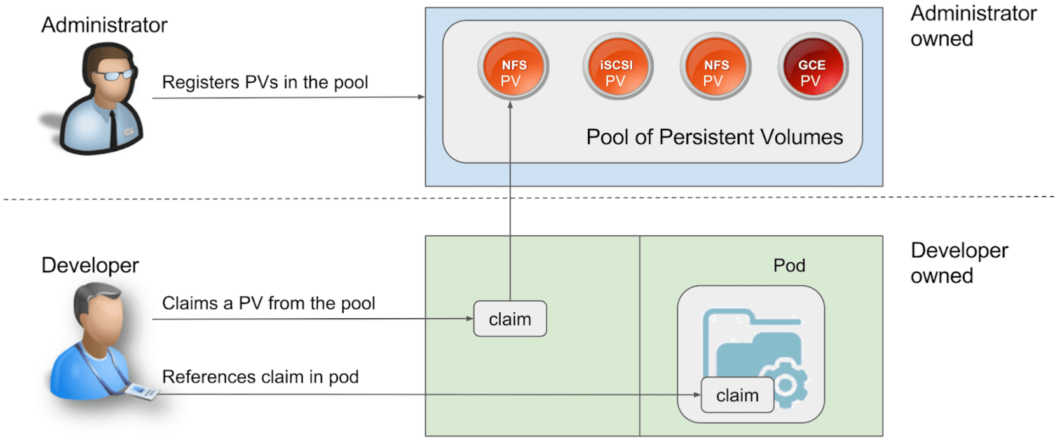
Within the claim, user need to specify the size of the volume along with access mode.

Developer:

I want a volume of size 10 GB which has speed of Fast for my pod.

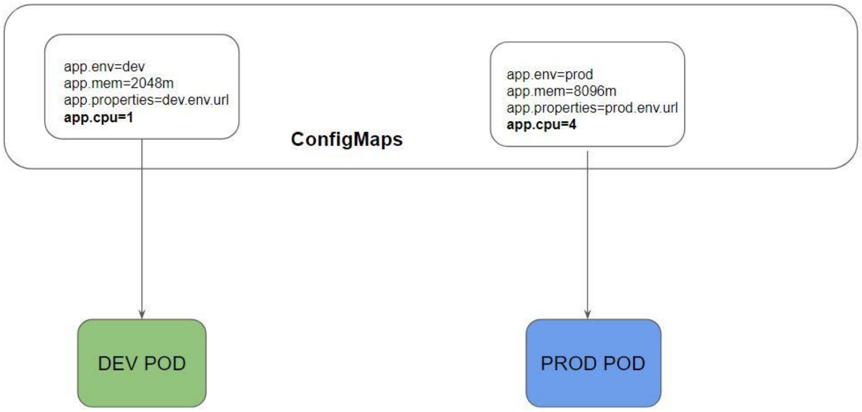
3.3 High-Level Working Steps:

- Storage Administrator takes care of creating PV.
- Developer can raise a “Claim” (I want a specific type of PV).
- Reference that claim within the PodSpec file.



Module 4: ConfigMaps

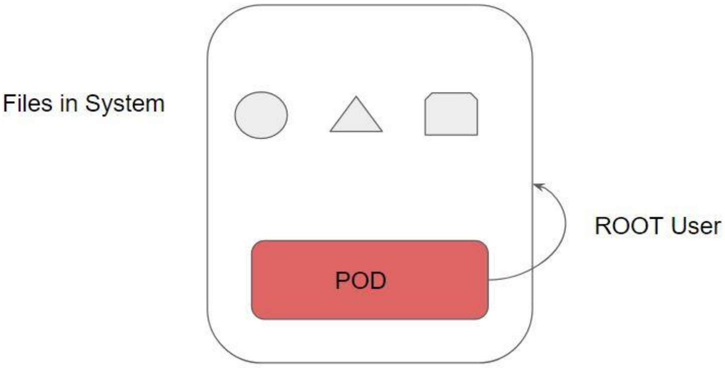
ConfigMaps allow you to decouple configuration artifacts from image content to keep containerized applications portable.



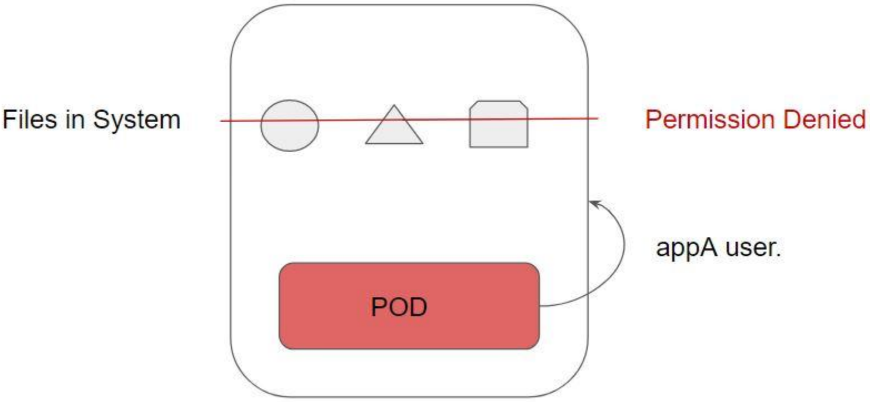
Module 5: Security Contexts

When you run a container, it runs with the UID 0 (Administrative Privilege)

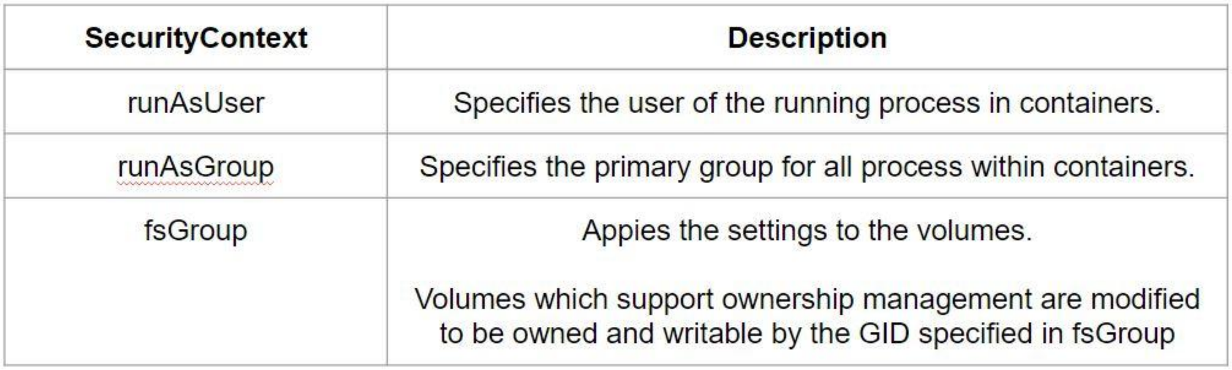
In-case of container breakouts, an attacker can get root privileges to your entire system.



We can run POD and container with limited privilege user instead of the ROOT user.



The following are the three important permissions:



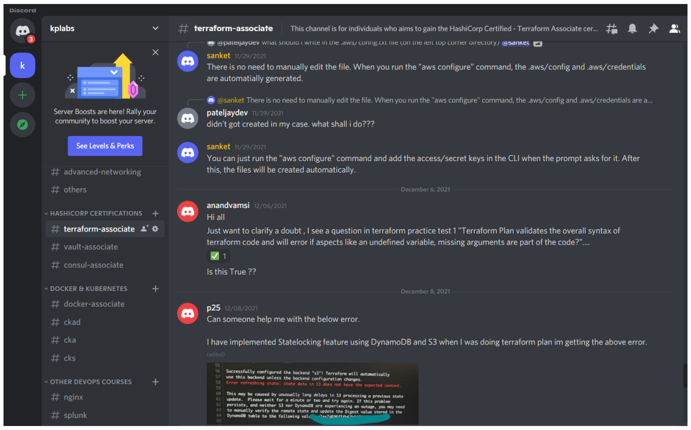
SecurityContext	Description
runAsUser	Specifies the user of the running process in containers.
runAsGroup	Specifies the primary group for all process within containers.
fsGroup	Appies the settings to the volumes. Volumes which support ownership management are modified to be owned and writable by the GID specified in fsGroup

Join Our Discord Community

We invite you to join our Discord community, where you can interact with our support team for any course-based technical queries and connect with other students who are doing the same course.

Joining URL:

http://kplabs.in/chat



This channel is for individuals who aims to gain the HashiCorp Certified - Terraform Associate cer...

@pateljaydev what should i write in the .aws/config.txt file (on the left top corner directory) @sanket

sanket 11/29/2021
There is no need to manually edit the file. When you run the "aws configure" command, the .aws/config and .aws/credentials are automatially generated.

@sanket There is no need to manually edit the file. When you run the "aws configure" command, the .aws/config and .aws/credentials are a...

pateljaydev 11/29/2021
didn't got created in my case. what shall i do???

sanket 11/29/2021
You can just run the "aws configure" command and add the access/secret keys in the CLI when the prompt asks for it. After this, the files will be created automatically.

December 6, 2021

anandvamsi 12/06/2021
Hi all

Just want to clarify a doubt , I see a question in terraform practice test 1 "Terraform Plan validates the overall syntax of terraform code and will error if aspects like an undefined variable, missing arguments are part of the code?"....

✅ 1

Is this True ??

December 8, 2021

p25 12/08/2021
Can someone help me with the below error.

I have implemented Statelocking feature using DynamoDB and S3 when I was doing terraform plan im getting the above error.

(edited)

```

Successfully configured the backend "s3"! Terraform will automatically
use this backend unless the backend configuration changes.
Error refreshing state: state data in S3 does not have the expected content.

This may be caused by unusually long delays in S3 processing a previous state
update.  Please wait for a minute or two and try again. If this problem
persists, and neither S3 nor DynamoDB are experiencing an outage, you may need
to manually verify the remote state and update the Digest value stored in the
DynamoDB table to the following valu[redacted]

```