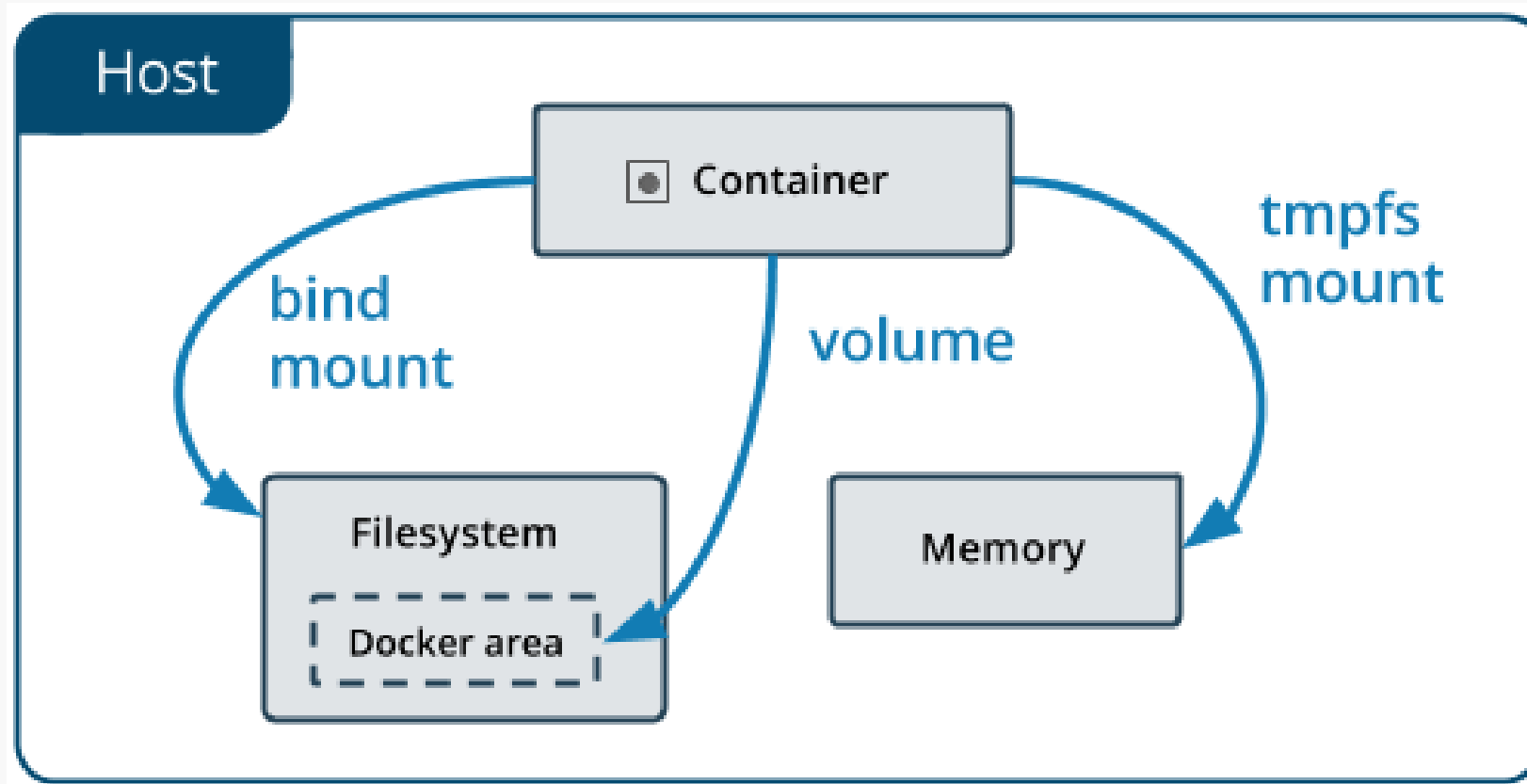# Docker Volumes Features

VISHWANATH M S

VISHWACLOUDLAB.COM

Docker Storage mount points

- **Volumes**
- **Bind mounts**
- **tmpfs mounts**

Docker Storage mount points

Docker Storage -- Volumes

- Volumes are easier to back up or migrate than bind mounts.

- You can manage volumes using Docker CLI commands or the Docker API.

- Volumes work on both Linux and Windows containers.

- Volumes can be more safely shared among multiple containers.

- Volume drivers let you store volumes on remote hosts or cloud providers, to encrypt the contents of volumes, or to add other functionality.

- New volumes can have their content pre-populated by a container.

Docker Storage – Volumes - Command

- Syntax → "-v"

- Create volume: $ **docker volume create** my-vol

- List Volumes: → $ **docker volume ls**

- Remove Volumes: → $ **docker volume rm** my-vol

Docker Storage – Volumes - Command

- Start a container with a volume

- $ docker run –d --name devtest **-v my-vol:/app** tomcat:latest

- Use a read-only volume

- $ docker run –d --name devtest **-v my-vol:/app:ro** tomcat:latest

Docker Storage – Bind mounts

- Bind mounts have limited functionality compared to Volumes.

- The file or directory is referenced by its full or relative path on the host machine.

- The file or directory does not need to exist on the Docker host already.

- You can't use Docker CLI commands to directly manage bind mounts.

Docker Storage – bind mounts - Command

- Start a container with a bind mounts

- $ docker run –d --name devtest **-v "$(pwd)"/target:/app** tomcat:latest

- Use a read-only bind mounts

- $ docker run –d --name devtest **-v "$(pwd)"/target:/app:ro** tomcat:latest

Docker Storage – tmpfs mounts

- The container can create files outside the container's writable layer.

- *tmpfs* mount is temporary and only persisted in the host memory.

- When the container stops, the *tmpfs* mount is removed, and files written there **won't** be persisted.