Persistent storage for Containers

Anil Degwekar
What are we talking about?

Containers have become popular – replacing many Physical / Virtual Machine use cases

But: The persistent storage problem for containers is still not fully solved
  – Considered by many to be the #1 challenge for containers adoption in the Enterprise
The problem

• As Containers move from one server to another, the associated storage also needs to move

• Easy to accomplish, if
  – Using cloud storage
  – Networked storage (NFS)

• Not so easy for block storage
The Container mobility problem

Database

Volume export

Storage Array

Node 1

Container
Docker volume

Node 2

Container
Docker volume

volume

volume
Docker volume plugins

- Docker has a large collection of volume plugins (drivers)

- Limitations
  - Plugin required on every node
  - Plugin options can vary quite a lot
  - Container movement across node is not seamless
  - Volume spec is somewhat preliminary
  - Volumes can get orphaned
  - No data management features (snapshots, etc.)
Kubernetes persistent volumes

• Kubernetes also has a large collection of volume plugins (drivers)

• Persistent Volume plugins have limitations similar to Docker
  – But the volume spec is somewhat more advanced compared to Docker

• Spec differs considerably from Docker volume plugin spec
Past attempts to solve these problems

- ClusterHQ (Flocker)
- Portworx
- Rex-Ray
- CSI
ClusterHQ (Flocker) Architecture

Compute node1
- Docker
- Flocker-Plugin
- Container 1
  - Flocker volume
  - Flocker-Agent
  - Array - plugin

user
- Flocker-Control-Service

Compute node2
- Docker
- Flocker-Plugin
- Container 2
  - Flocker volume
  - Flocker-Agent
  - Array - plugin

Storage Array
Portworx architecture

- Allows Container volumes to span arrays
- A single array volume can be split into multiple container volumes
- Supports additional services
  - HA
  - Snapshots
  - Encryption
  - Etc.
Rex-Ray overview

- Common framework for all Container orchestrators
- Runs as a container in Docker
- Open source
- Multiple deployment modes
  - Standalone
  - Agent and Controller
Container Storage Interface (CSI)

• Interface between Orchestrators and Storage Plugins

• Promise
  – Write a plugin once, and use it with any Container Orchestrator

• Managed by CNCF

• But: spec is at a preliminary stage
Timeline

- First release of Docker
- First release of Kubernetes
- First release of CSI
- First release of Flocker
- First release of Rex-Ray
- First release of Portworx
- ClusterHQ closed
What more is needed

- Advanced data management features (snapshots, clones)
- Data reduction features (de-duplication, compression)
- Encryption

Many stateful applications need these services to migrate to Containers
What are we doing in this space

- EMC had a partnership with ClusterHQ
- Rex-Ray project was open source - part of {code} sponsored by Dell EMC
- Dell EMC is a major contributor to CSI
- Volume plugin available for ScaleIO
- CoprHD has a container solution – open source
Call for action

- **Storage vendors**
  - Keep the container story in mind when developing your solutions
  - Participate in CNCF and CSI

- **Standards bodies**
  - Need to come up with some common standards in this space

- **Application developers**
  - Be aware of this issue
  - If your application uses Block storage
  - And you want to migrate it to Containers