OpenShift at Sea

Deploying OpenShift on Norwegian Cruise's Bliss

Clark Hale  Eric Salom  Brian Barreto
Openshift Architect  Sr Linux Sys Admin  Server and Virtualization

May 9th, 2019
M/S Norwegian Bliss
The Problem

If run in a traditional VM deployment, our "Mustering" application, a critical safety system, required resources well beyond what was available in the limited Data Center on a ship.

We needed a way to maximize the resource density in a highly available way that could be automatically deployed to the ship and maintained remotely.
First Application

- The first application on the ship is a "Mustering" application
- A safety system that ensures everyone on the ship is accounted for during an emergency
- Absolutely critical for safe ship operation!
Enter OpenShift
Why OpenShift?

● IT staff is limited on ships requiring automation
● Physical space and power are limited on ships
● VMware provides solid, known base
● Containers provide higher density
● OpenShift provides application self-healing
● Ease of management and enforcing deployment automation
● IT Staff relying on shore side for troubleshooting
Our Challenges
Communication

- Satellite connection only
- OK Bandwidth - terrible latency
- All software must be staged locally on the ship before deployment
- Connectivity can be worse when ship is in some areas (e.g. Straits of Alaska).
- Not just data, but PHONE as well.
Challenges

● Images from Ship capsule
  ○ Not well documented
  ○ Lots of images
  ○ Lots of Ansible config
● Gluster/OCS Brick Multiplexing
● F5 Stickiness
● Upgrade path!
Our Architecture
M/S Norwegian Bliss
Architecture
Components

● On Shore:
  ○ Satellite
  ○ GitHub Enterprise
  ○ Active Directory

● On Ship:
  ○ Capsule
  ○ Local Git Copy
  ○ Active Directory (Unique to each Ship)

● For each OCP
  ○ Ansible Tower
  ○ F5 VIPs
Process
Synchronize Local Data

Shore Side

Satellite

Ship Side

Capsule
Build VMs & Config F5, etc.
Populate Ansible Tower
Execute Install

Install OCP Cluster - Bliss

Job Template
Environments for code deployment

- Dev
  - Used to test upgrade and other tweaks just for OCP
- Qa-shore
  - Deployment and validation of vendors code
- Qa-ship
  - Code staging and UAT before migrating to the ships
- Prod
  - Shoreside production environment
- Dr
  - Shoreside disaster recovery environment
- Bliss
  - Ship production environment
- Encore
  - Ship production environment
Code Deployment Flow
Future Plans
Future

- More Ships with OpenShift
  - M/S Encore, M/S Joy in progress
- More automation (F5, VMWare)
- Changing the Ship's image repository to Nexus.
  - Nexus to be used for holding container images instead of Satellite
- NCLH developers really want Istio!
  - Istio is a Service Mesh
- OpenShift in the Cloud!
THANK YOU

linkedin.com/company/Red-Hat
youtube.com/user/RedHatVideos
facebook.com/RedHatInc
twitter.com/RedHat