Deploying applications to Heterogeneous Hardware using Rancher and Docker

Bruno Grazioli Research Assistant, ZHAW

14th Docker Switzerland User Group Meetup 31st May 2017

Outline

- Brief introduction to Rancher
- Modifications in rancher-agent to support ARM (aarch64 systems)
- How to schedule applications to specific hosts
 - $\circ \quad \ \ \, \text{In Rancher and Swarm}$
- Multi-arch Docker images with the manifest tool

Rancher 101

- Container management platform focused on delivering containers on any infrastructure
 - Created in 2014
 - Packet, Orange (...) are using it
- Takes care of creating/managing your infrastructure
 - Cloud Providers (AWS, DO, Rackspace, Exoscale...)
 - Custom Hosts
 - Linux machines with Docker engine installed
- Multiple types of environments available
 - Cattle, Swarm, k8s, Mesos
 - Cattle is an orchestrator created by Rancher





Rancher 101

- Comprises of rancher-server and rancher-agent
- rancher-server runs in a standalone container
 - Manages environments/infrastructure
 - Provides user friendly UI...
- rancher-agent runs in a container in each host:
 - Is deployed when docker-engine initialized
 - Or must be run explicitly for addition of custom hosts
 - Runs in privileged mode
 - Controls network configuration, health monitoring and can support deployment of applications



What about ARM?

- ARM processors are present in many devices...
 - smartphones, tablets, chromebooks, embedded devices
- Interesting to consider contexts where components can be deployed to heterogeneous infrastructure
 - x86 in cloud and a Raspberry Pi, but could also be ARM server or larger ARM system at the edge
- Rancher has excellent support for docker-engine running on x86 infrastructure...
 - ...<u>support for other infrastructures is somewhat limited</u>
- Partial support for ARM existed in previous versions of rancher-agent
- <u>We updated ARM (aarch64) support to newer</u> <u>rancher-agent versions</u>



Building rancher-agent for ARM

- We chose to focus on aarch64 environment
 - Used Suse Leap on Raspberry Pi 3
 - Raspbian and other distros use armhf
 - Tooling and applications for aarch64 are not problematic
 - docker-engine version 1.12 available
 - Build tools available golang, gcc
- Instructions for building containers provided in rancher Dockerfiles
 - Mix of apt-get commands plus building specific binaries
 - Binaries include healthcheck, network-manager, scheduler





Experience building rancher-agent for ARM

- The build process itself was somewhat complex
 - Containers on rancher-agent host are written primarily in golang
- There were some specific tricks
 - Some binaries had to be installed into volumes manually
 - rancher-agent installs x86 binaries obtained from rancher-server; had to circumvent this
 - The network configuration took longer to come up on the Raspberry Pi...
 - ...resulting in issues for the healthcheck and scheduler
 - Docker swarm orchestration exhibited problems with Raspberry Pi joining the swarm
 - This was solved by removing a pruning phase in the rancher agent setup



Application Deployment

- Significant constraint exists relating to application deployment
 - Containers deployed on Raspberry Pi must have been built for this architecture
 - Partial support exists for this in docker registries
- Scheduling is based on labels to differentiate between hosts
 - For both Swarm and Cattle
 - Specific labels required according to the environment used



Examples

Cattle

version: '2' services: wordpress: image: aarch64/wordpress depends on: - mariadb labels: io.rancher.scheduler.affinity:host_label: type=raspberry ports: - 8080:80 environment: WORDPRESS DB PASSWORD: ... mysql: image: mariadb labels: io.rancher.scheduler.affinity:host label: type=VM environment: MYSQL ROOT PASSWORD: ... MYSQL DATABASE: wordpress

Swarm

- node.labels.type == VM

(...)





Creating multi-arch images for Docker registry



- Docker registry has support for multi-arch images
 - Currently this is limited to pulling images in the CLI
 - Getting multi-arch images into the registry is a bit more tricky
 - Although there is a third party tool solving this issue



image: icclab/ubuntu-multi-arch:16.04
manifests:

- image: ubuntu:16.04
 - platform:
 - architecture: amd64
 - os: linux
- image: aarch64/ubuntu:16.04
 - platform:
 - architecture: arm64
 os: linux



Questions?



