



CLOUD
NATIVE
CON
Seattle 2016



KubeCon
A CNCF EVENT

Automating Infrastructure Deployment for Kubernetes

Alena Prokharchyk, *Rancher Labs*,
@lemonjet

Brian Scott, *The Walt Disney Company*,
@brainscott



Kubernetes is awesome

- It automates your application deployment
- Provides DNS/Service Discovery
- Manages application health, and recovers from failures
- Handles application upgrades

Kubernetes Cluster management should be handled the same way

Tools and sources



kubeadm



ANSIBLE



kubernetes / kops

kubernetes-incubator / bootkube



Terraform

Challenges

1. Cloud infrastructure management

- Storage
- Networking
- Dns/ServiceDiscovery

2. Health monitoring of Kubernetes cluster. Recovery.

3. Kubernetes cluster upgrade

Kubernetes is the most popular orchestration platform among Rancher users



And we wanted to make Kubernetes deployments and upgrades easy

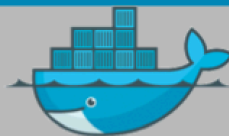
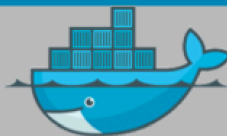
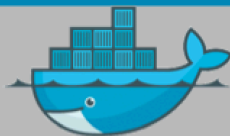
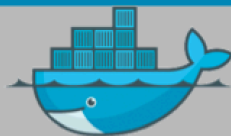
Lay the foundation

- Setup networking
- Get internal DNS in place
- Choose storage driver solution

All of the above works across clouds



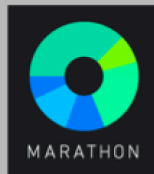
Applications
(Containerized)



Application
Orchestration
(Containerized)



kubernetes



And more to come...

Infrastructure
Services

Monitoring

Reliable
Storage

Load
Balancing

Network
(SDN)

DNS

Database
RDBMS, K/V

And
More...



Raw Hardware
Physical/Virtual

Compute
CPU x86/ARM

Storage
SSD

Network
Fast Fabric

Build your own deployment template

Add Template

Name

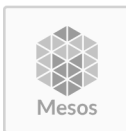
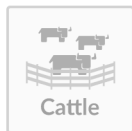
Description

Sharing

☒ Private: Only available to me





☐ Public: Available to all users (admin only)

Orchestration




Edit Config

Add all the framework services

 Healthcheck Service in Library Rancher healthcheck service Required	 Internal DNS in Library Required	 Metadata in Library Required	 Network Manager in Library Required
---	---	--	--

... one of the CNI drivers

Networking




PROJECT
CALICO

Calico
in Library

A Pure Layer 3 Approach to Virtual
Networking for Highly Scalable Data
Centers


Enable



**RANCHER
IPSEC**

Rancher IPsec
in Library

Edit Config Disable




**RANCHER
VXLAN**

Rancher VXLAN
in Library

Rancher Networking plugin using
VXLAN overlay.

Enable




weaveworks

Weave
in Library

Networking plugin from Weaveworks.

Enable


... and a storage driver

**RANCHER
EFS**

Rancher EFS
in Library

Docker volume and Flex Vol plugin for
EFS


Enable

**REX-Ray**
Openly serious about storage

REX-Ray
in Library

Docker volume plugin for various
providers

Enable

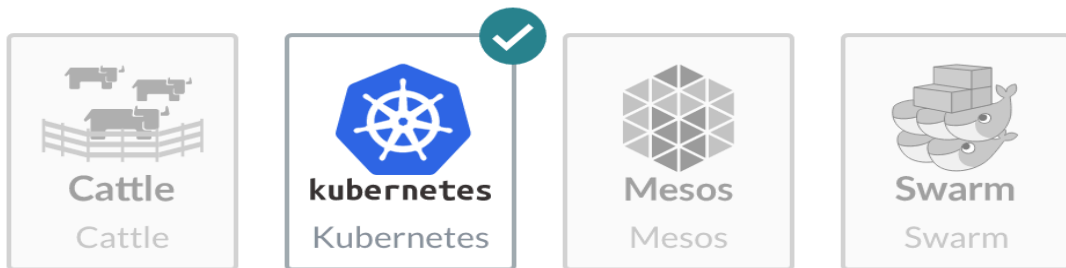
**EMC²**
SCALEIO

ScaleIO NAS/DAS Store
in Library

ScaleIO NAS/DAS Store

Enable

Then deploy the template with 1 click



Orchestration: Kubernetes

Framework: Internal DNS, Metadata, Network Manager, Healthcheck Service

Networking: Rancher IPsec

Scheduling: Scheduler

Cluster deployment orchestration choices

- **Standalone**

No resilience to host failure, mostly used for development and training


- **Resilient Overlapping Data and Control Planes**

Resilient to hosts failures. Potential performance issues with etcd/kubernetes components

- **Resilient Separated-Planes**

High-performance, production-ready environment

Controller plane health monitoring



—

kubernetes

<div><div>⚠ Degraded</div></div>	controller-manager ⓘ					
<div><div>📦 Active</div></div>	etcd + 1 Sidekick ⓘ					
<div><div>📦 Active</div></div>	kubectld ⓘ					
<div><div>📦 Active</div></div>	kubelet ⓘ	09:12:05 PM	<div>INFO</div>	service.update.info	Service reconciled: Requested: 1, Created: 1, Unhealthy: 0, Bad: 0, Incomplete: 0	<div><div>🔧</div><div>⋮</div></div>
<div><div>⚠ Unhealthy</div></div>	kubernetes + 1 Sidekick ⓘ	09:12:03 PM	<div>INFO</div>	service.instance.create	Creating extra service instance	<div><div>📦</div><div>⋮</div></div>
<div><div>📦 Active</div></div>	proxy ⓘ	09:12:03 PM	<div>INFO</div>	service.update.wait (2 sec)	Waiting for instances to start	<div><div>🔧</div><div>⋮</div></div>
		09:12:02 PM	<div>INFO</div>	service.instance.delete	Removing unhealthy service instance	<div><div>📦</div><div>⋮</div></div>

Health Checks in Rancher

- Health check agent runs on every host
- Up to 3 agents check on every health check enabled container
- Once considered unhealthy by all 3 agents, container gets reported and replaced

Various health failure recover strategies to suit different types of services

Recreate on quorum, used by etcd cluster

Regular aka “always recreate”

```
kubernetes:
  scale: 1
  retain_ip: true
  health_check:
    port: 80
    interval: 2000
    unhealthy_threshold: 3
    strategy: recreate
    response_timeout: 2000
    request_line: GET /healthz HTTP/1.0
    healthy_threshold: 2
```

```
etcd:
  scale: 1
  retain_ip: true
  scale_policy: &id001
    increment: 1
    max: 3
    min: 1
  health_check:
    port: 2378
    interval: 5000
    recreate_on_quorum_strategy_config:
      quorum: 2
    unhealthy_threshold: 3
    strategy: recreateOnQuorum
    response_timeout: 3000
    request_line: GET /health HTTP/1.0
    healthy_threshold: 2
```

Reliable etcd cluster

1. Configuring remote backups

1. Failure Recovery

1. Disaster Recovery

1. Restoring Backups

Automated upgrades

- In Rancher we follow Kubernetes release process and test every major and almost every minor version
- Once tested, Rancher Infrastructure Catalog gets a new published Kubernetes template
- Current Kubernetes clusters get a notification and an option to upgrade to the latest

Upgrade UI

Stack:

kubernetes ▼

Add Service ▼

Upgrade available

Upgrade: kubernetes



kubernetes

Catalog: Infra

Category: Orchestration

Upgrade Kubernetes Stack

Rancher Kubernetes service

Template Version

v1.4.0-rancher1



Select a version of the template to upgrade to

DEMO

In works

- Cluster Federation support (Ubernetes)
- Pluggable DNS providers (Rancher DNS, SkyDNS as an option)
- Improvements to cluster deployment



Thank you

rancher.com