kubernetes

Kubernetes Vagrant CI / CD

GitHub Project Repo: kubernetesVagrant

Sample of Work of fully automate build, deploy, validate containers locally. Deploy on a k8s cluster and validations.

September 28, 2021

Outline

Introduction

- Container Runtime Interface(s)
  - Docker
  - Podman
  - CRI-O

CI / CD

- Project smooth CI / CD
- How to accomplish requirements
- The correct tool of choice

Bibliography

Outline

Introduction

- Container Runtime Interface(s)
  - Docker
  - Podman
  - CRI-O

CI / CD

- Project smooth CI / CD
- How to accomplish requirements
- The correct tool of choice

Bibliography

Outline

Introduction

- Container Runtime Interface(s)
  - Docker
  - Podman
  - CRI-O

CI / CD

- Project smooth CI / CD
- How to accomplish requirements
- The correct tool of choice

Bibliography

High level description

- VM Vs Container.
- What is a Pod (pea pod)?
- Container Runtime Interface(s) (CRI).
  - Docker.
  - Podman.
  - CRI-O.
- Docker Vs Podman.
- What is actually k8s?

Virtual Machine Vs Container

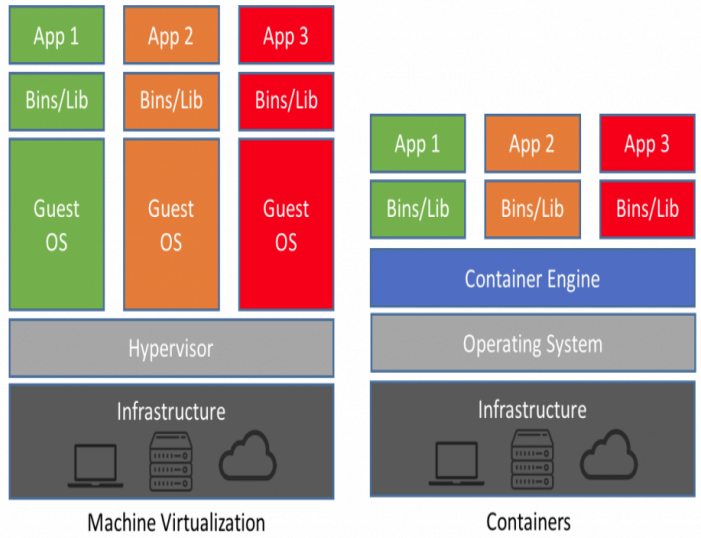

Figure 1: k8s Overview

High level description

- VM Vs Container.
- What is a Pod (pea pod)?
- Container Runtime Interface(s) (CRI).
  - Docker
  - Podman
  - CRI-O
- Docker Vs Podman.
- What is actually k8s?

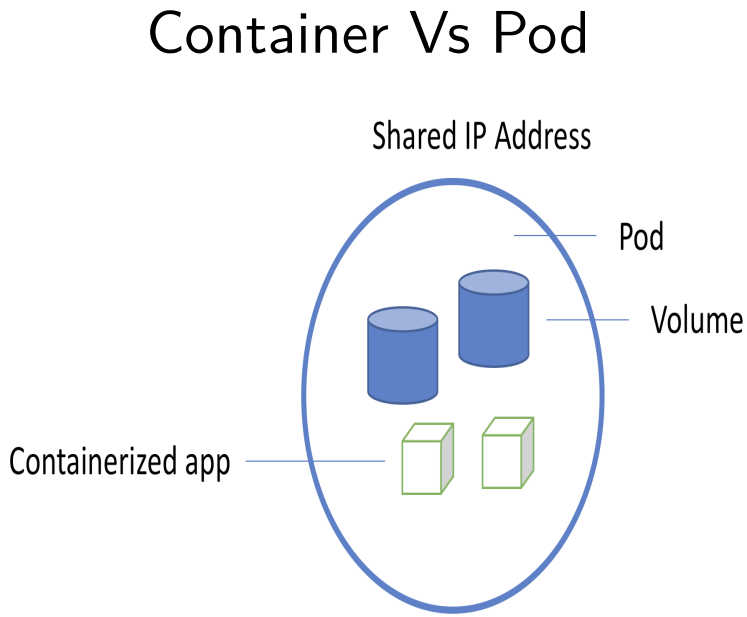

Figure 1: k8s Overview

High level description

- VM Vs Container.
- What is a Pod (pea pod)?
- Container Runtime Interface(s) (CRI).
  - Docker
  - Podman
  - CRI-O
- Docker Vs Podman.
- What is actually k8s?

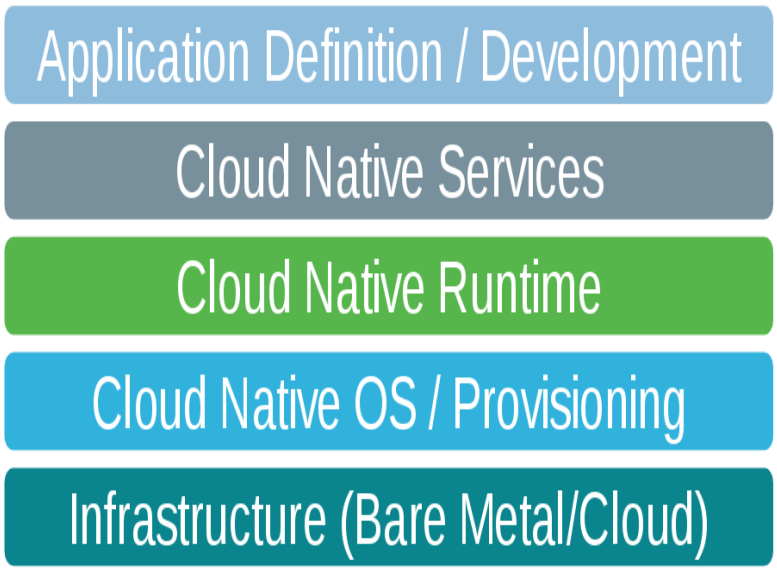

Figure 1: k8s Overview

High level description

- VM Vs Container.
- What is a Pod (pea pod)?
- Container Runtime Interface(s) (CRI).
  - Docker
  - Podman
  - CRI-O
- Docker Vs Podman.
- What is actually k8s?

Most known (insecure) socket

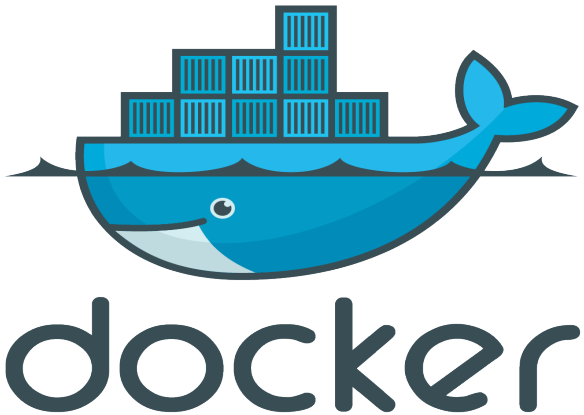

Figure 1: k8s Overview

High level description

- VM Vs Container.
- What is a Pod (pea pod)?
- Container Runtime Interface(s) (CRI).
  - Docker
  - Podman
  - CRI-O
- Docker Vs Podman.
- What is actually k8s?

Most unknown (secure) socket

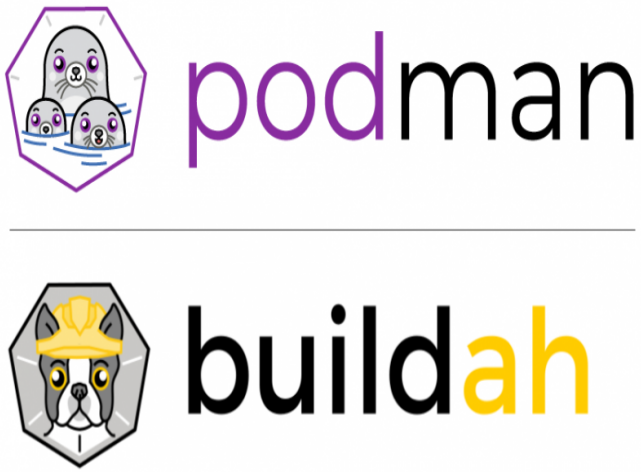

Figure 1: k8s Overview

High level description

- VM Vs Container.
- What is a Pod (pea pod)?
- Container Runtime Interface(s) (CRI).
  - Docker
  - Podman
  - CRI-O
- Docker Vs Podman.
- What is actually k8s?

Lightest socket currently

Figure 1: k8s Overview

High level description

- VM Vs Container.
- What is a Pod (pea pod)?
- Container Runtime Interface(s) (CRI).
  - Docker
  - Podman
  - CRI-O
- Docker Vs Podman.
- What is actually k8s?

Highly recommended read about it

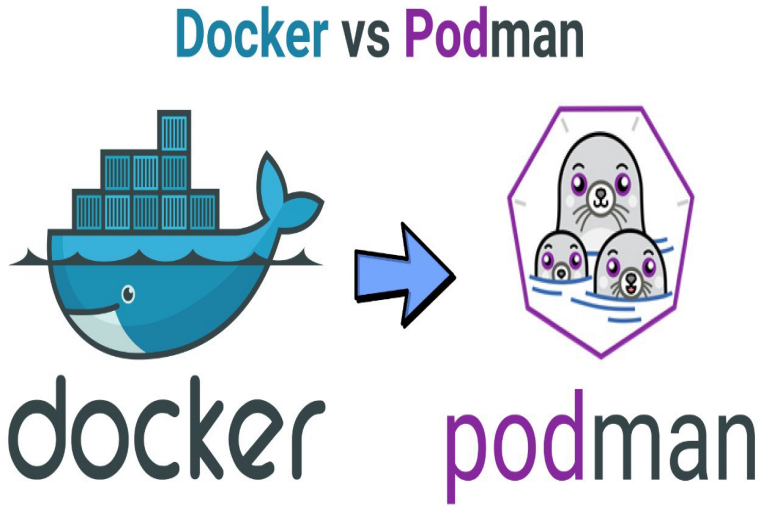

Figure 1: k8s Overview

High level description

- VM Vs Container.
- What is a Pod (pea pod)?
- Container Runtime Interface(s) (CRI).
  - Docker
  - Podman
  - CRI-O
- Docker Vs Podman.
- What is actually k8s?

Is a puzzle of elements

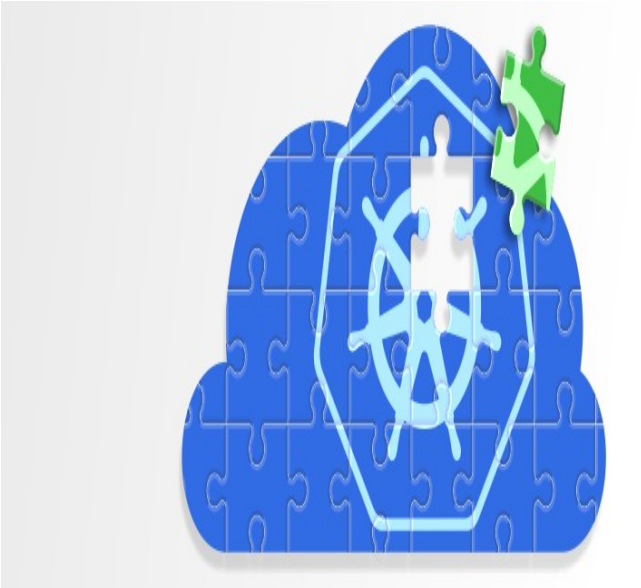

Figure 1: k8s Overview

Problems / Desires / Solutions on CI / CD

Assessment Requirements

- CI / CD (Locally / Remotely).
- Everything As a Code
- Validation Locally!!!
- OS / Inf. dependencies.

Justification of Requirements

- Works on my PC, why not on Cloud?
- Human error (manual).
- Test (automatically).
- GitHub Actions, Jenkins

Problems / Desires / Solutions on CI / CD

Assessment Requirements

- CI / CD (Locally / Remotely).
- Everything As a Code
- Validation Locally!!!
- OS / Inf. dependencies.

Justification of Requirements

- Works on my PC, why not on Cloud?
- Human error (manual).
- Test (automatically).
- GitHub Actions, Jenkins

Problems / Desires / Solutions on CI / CD

Assessment Requirements

- CI / CD (Locally / Remotely).
- Everything As a Code
- Validation Locally!!!
- OS / Inf. dependencies.

Justification of Requirements

- Works on my PC, why not on Cloud?
- Human error (manual).
- Test (automatically).
- GitHub Actions, Jenkins

Problems / Desires / Solutions on CI / CD

Assessment Requirements

- CI / CD (Locally / Remotely).
- Everything As a Code
- Validation Locally!!!
- OS / Inf. dependencies.

Justification of Requirements

- Works on my PC, why not on Cloud?
- Human error (manual).
- Test (automatically).
- GitHub Actions, Jenkins

Problems / Desires / Solutions on CI / CD

Assessment Requirements

- CI / CD (Locally / Remotely).
- Everything As a Code
- Validation Locally!!!
- OS / Inf. dependencies.

Justification of Requirements

- Works on my PC, why not on Cloud?
- Human error (manual).
- Test (automatically).
- GitHub Actions, Jenkins

Solutions to Requirements

- Solution has to be reproducible locally. Exactly as cloud.
- Minimal human interaction. Auto error handling (Rollback)!
- Powerful PCs! (8 CPUs / 35 GB RAM). Browsing (tabs)?
- No Vendor binding (Azzure DevOps, Jenkins, Bamboo etc).

Problems / Desires / Solutions on CI / CD

Assessment Requirements

- CI / CD (Locally / Remotely).
- Everything As a Code
- Validation Locally!!!
- OS / Inf. dependencies.

Justification of Requirements

- Works on my PC, why not on Cloud?
- Human error (manual).
- Test (automatically).
- GitHub Actions, Jenkins

Solutions to Requirements

- Solution has to be reproducible locally. Exactly as cloud.
- Minimal human interaction. Auto error handling (Rollback)!
- Powerful PCs! (8 CPUs / 35 GB RAM). Browsing (tabs)?
- No Vendor binding (Azzure DevOps, Jenkins, Bamboo etc).

Problems / Desires / Solutions on CI / CD

Assessment Requirements

- CI / CD (Locally / Remotely).
- Everything As a Code
- Validation Locally!!!
- OS / Inf. dependencies.

Justification of Requirements

- Works on my PC, why not on Cloud?
- Human error (manual).
- Test (automatically).
- GitHub Actions, Jenkins

Solutions to Requirements

- Solution has to be reproducible locally. Exactly as cloud.
- Minimal human interaction. Auto error handling (Rollback)!
- Powerful PCs! (8 CPUs / 35 GB RAM). Browsing (tabs)?
- No Vendor binding (Azzure DevOps, Jenkins, Bamboo etc).

Problems / Desires / Solutions on CI / CD

Assessment Requirements

- CI / CD (Locally / Remotely).
- Everything As a Code
- Validation Locally!!!
- OS / Inf. dependencies.

Justification of Requirements

- Works on my PC, why not on Cloud?
- Human error (manual).
- Test (automatically).
- GitHub Actions, Jenkins

Solutions to Requirements

- Solution has to be reproducible locally. Exactly as cloud.
- Minimal human interaction. Auto error handling (Rollback)!
- Powerful PCs! (8 CPUs / 35 GB RAM). Browsing (tabs)?
- No Vendor binding (Azzure DevOps, Jenkins, Bamboo etc).

Solution to problem

Problems

- Solution has to be reproducible locally. Exactly as cloud.
- Minimal human interaction. Auto error handling (Rollback)!
- Powerful PCs! (8 CPUs / 35 GB RAM). Browsing (tabs)?
- No Vendor binding (Azzure DevOps, Jenkins, Bamboo etc).

Solutions

- Containers. Build / deploy / validate locally (controlled env)!
- Fully automated procedure on every step!
- Launch a k8s cluster locally and run all tests locally!
- High Level Programming Language with error handling!

Solution to problem

Problems

- Solution has to be reproducible locally. Exactly as cloud.
- Minimal human interaction. Auto error handling (Rollback)!
- Powerful PCs! (8 CPUs / 35 GB RAM). Browsing (tabs)?
- No Vendor binding (Azzure DevOps, Jenkins, Bamboo etc).

Solutions

- Containers. Build / deploy / validate locally (controlled env)!
- Fully automated procedure on every step!
- Launch a k8s cluster locally and run all tests locally!
- High Level Programming Language with error handling!

Solution to problem

Problems

- Solution has to be reproducible locally. Exactly as cloud.
- Minimal human interaction. Auto error handling (Rollback)!
- Powerful PCs! (8 CPUs / 35 GB RAM). Browsing (tabs)?
- No Vendor binding (Azzure DevOps, Jenkins, Bamboo etc).

Solutions

- Containers. Build / deploy / validate locally (controlled env)!
- Fully automated procedure on every step!
- Launch a k8s cluster locally and run all tests locally!
- High Level Programming Language with error handling!

Solution to problem

Problems

- Solution has to be reproducible locally. Exactly as cloud.
- Minimal human interaction. Auto error handling (Rollback)!
- Powerful PCs! (8 CPUs / 35 GB RAM). Browsing (tabs)?
- No Vendor binding (Azzure DevOps, Jenkins, Bamboo etc).

Solutions

- Containers. Build / deploy / validate locally (controlled env)!
- Fully automated procedure on every step!
- Launch a k8s cluster locally and run all tests locally!
- High Level Programming Language with error handling!

Ansible

Possible Questions

- Why Ansible?
- Are there any benefits of this tool?
- Ansible works on ssh how it will work locally?
- How it can interact with Containers, k8s, Cloud, tests?

Possible Answers

- Written in Python 2/3. Developed and maintained by RedHat.
- Woks perfectly without extra configurations on all OS.
- It can be configured to run on localhost without ssh session.
- It has infinite amount of packages for OS, Containers, Cloud.

Ansible

Possible Questions

- Why Ansible?
- Are there any benefits of this tool?
- Ansible works on ssh how it will work locally?
- How it can interact with Containers, k8s, Cloud, tests?

Possible Answers

- Written in Python 2/3. Developed and maintained by RedHat.
- Woks perfectly without extra configurations on all OS.
- It can be configured to run on localhost without ssh session.
- It has infinite amount of packages for OS, Containers, Cloud.

Ansible

Possible Questions

- Why Ansible?
- Are there any benefits of this tool?
- Ansible works on ssh how it will work locally?
- How it can interact with Containers, k8s, Cloud, tests?

Possible Answers

- Written in Python 2/3. Developed and maintained by RedHat.
- Woks perfectly without extra configurations on all OS.
- It can be configured to run on localhost without ssh session.
- It has infinite amount of packages for OS, Containers, Cloud.

Ansible

Possible Questions

- Why Ansible?
- Are there any benefits of this tool?
- Ansible works on ssh how it will work locally?
- How it can interact with Containers, k8s, Cloud, tests?

Possible Answers

- Written in Python 2/3. Developed and maintained by RedHat.
- Woks perfectly without extra configurations on all OS.
- It can be configured to run on localhost without ssh session.
- It has infinite amount of packages for OS, Containers, Cloud.

References I

GNU LESSER GENERAL PUBLIC LICENSE
GNU Operating System
available at https://www.gnu.org/licenses/lgpl.html.