KubeOne

Kubernetes Cluster Lifecycle Management Tool

Who are we?

Marko Mudrinić

Software Developer @ Loodse

@xmudrii

Artiom Diomin

Software Developer @ Loodse

@kron4eg

Agenda

- Introduction to KubeOne
- Core concepts and architecture
- Demo: Manage Kubernetes HA cluster on AWS
- Demo: Configure the cluster and explore KubeOne features

Introduction

What is KubeOne?

- A tool for **managing** Kubernetes **cluster lifecycle**
  - Installs and provisions Kubernetes, upgrades, un-provisions the cluster
- **Open source** and **vendor neutral**
- Works on the most popular **cloud providers**, on **on-prem** and on **bare metal**
- Supports **1.13+ Highly-Available** clusters

Why we built KubeOne?

- Kubernetes brought us a new way for managing our workload...
- but managing Kubernetes clusters is still a hard task.
- We want to apply lessons learned managing workload to clusters.

In a search for a **feature-complete** solution, we decided to build **KubeOne**

Why KubeOne?

- Uses the **latest technologies** to bring **many features** in an **easy to consume** manner
- Brings **declarative** cluster representation
- Provides **ready to use** cluster
- Optionally configures various features on the provisioning time:
  - PodSecurityPolicy, DynamicAuditLog, metrics-server and more
- Ability to integrate KubeOne with **infrastructure provisioning tools**

Supported providers

- KubeOne is supposed to work on **any** provider, including on-prem and bare metal
- Officially supported providers enjoy **additional features** such as:
  - Support for managing worker nodes using Kubermatic machine-controller
  - Automatically deploy cloud provider specific features like external CCM
  - Use Terraform integration to pick up information about infrastructure from the Terraform state
- Officially supported providers include AWS, GCE, DigitalOcean, Hetzner, Packet, OpenStack and VMware vSphere
- Support for Microsoft Azure is coming up soon

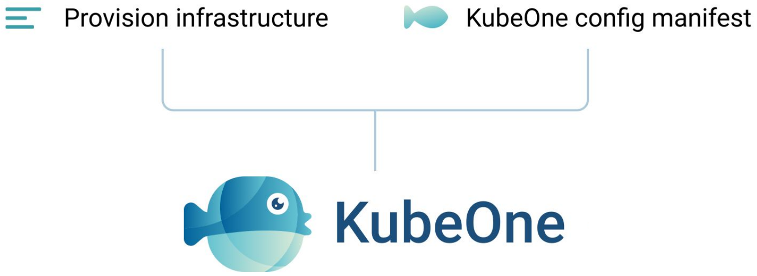
Architecture

Architecture

- KubeOne uses many tools/solutions as **building blocks**
  - kubeadm is used to provision and join **control plane nodes** and handle **cluster upgrades**
  - Kubermatic machine-controller based on Cluster-API is used to manage **worker nodes**
- The environment is prepared over SSH
  - Including installing and upgrading binaries, configuring components and running kubeadm
- client-go is used for deploying various cluster features such as CNI

Installation process

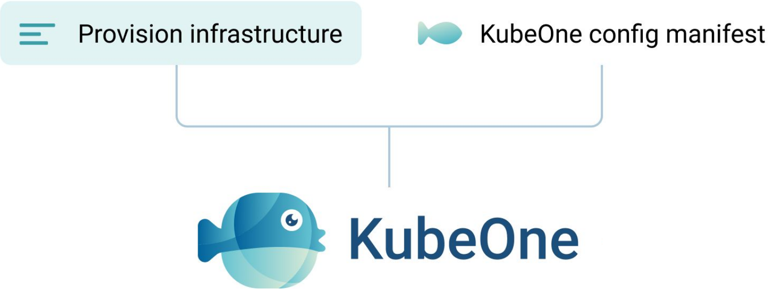
Installation process



Provision infrastructure

KubeOne config manifest

Installation process

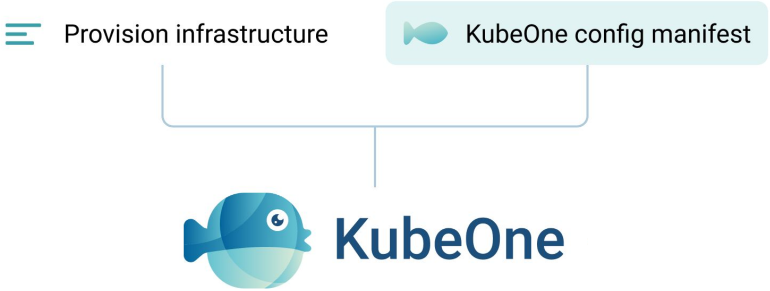


Provision infrastructure

KubeOne config manifest

KubeOne

Installation process

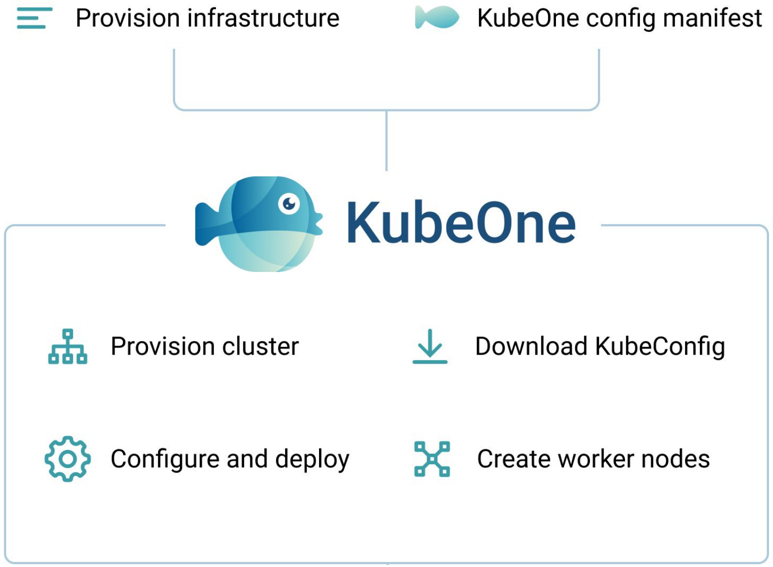


Provision infrastructure

KubeOne config manifest

KubeOne

Installation process



Provision infrastructure

KubeOne config manifest

KubeOne

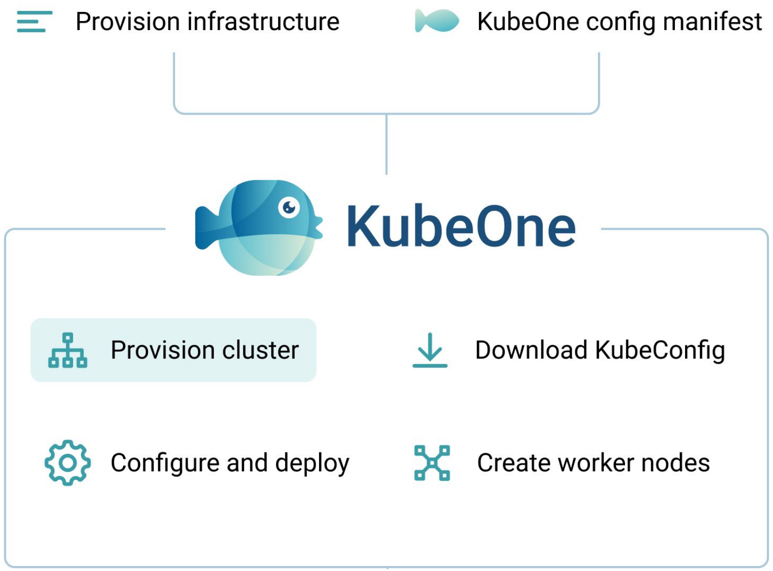
Provision cluster

Download KubeConfig

Configure and deploy

Create worker nodes

Installation process



Provision infrastructure

KubeOne config manifest

KubeOne

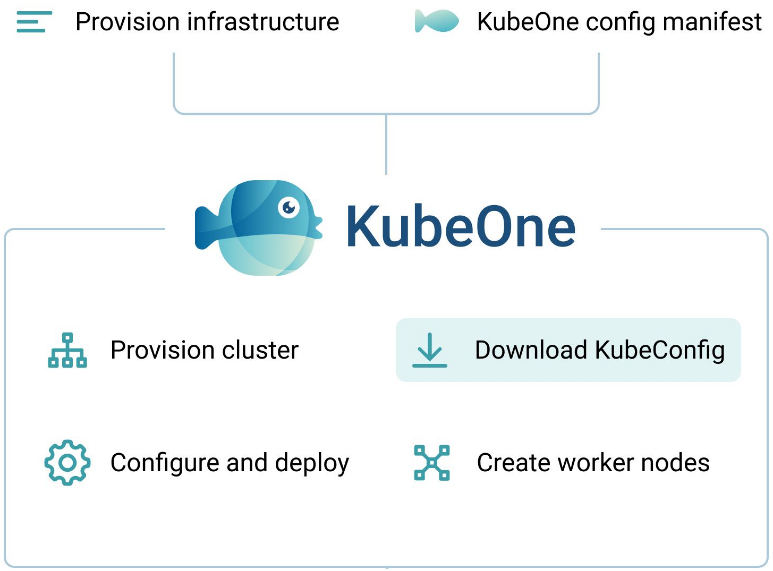
Provision cluster

Download KubeConfig

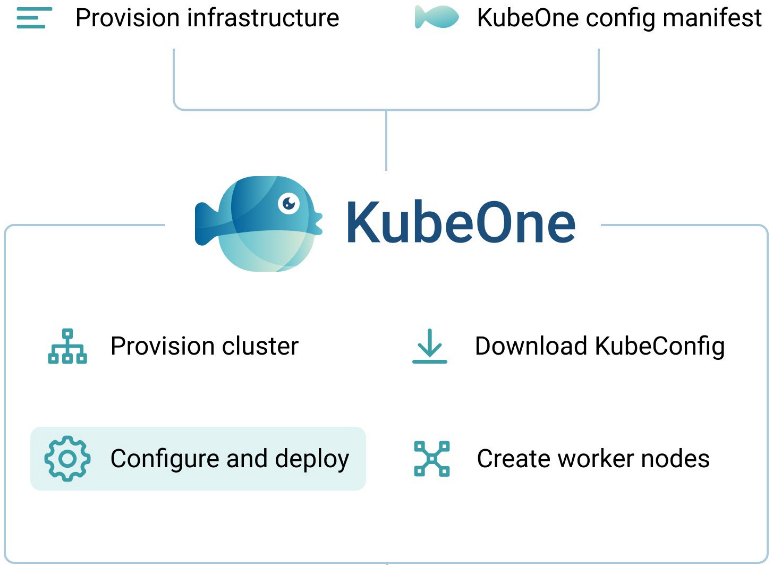
Configure and deploy

Create worker nodes

Installation process



Installation process



Provision infrastructure

KubeOne config manifest

KubeOne

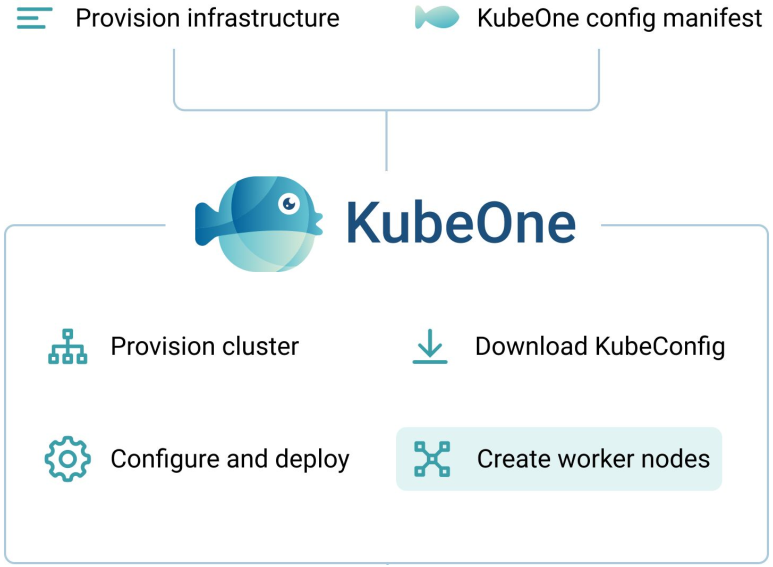
Provision cluster

Download KubeConfig

Configure and deploy

Create worker nodes

Installation process



Provision infrastructure

KubeOne config manifest

KubeOne

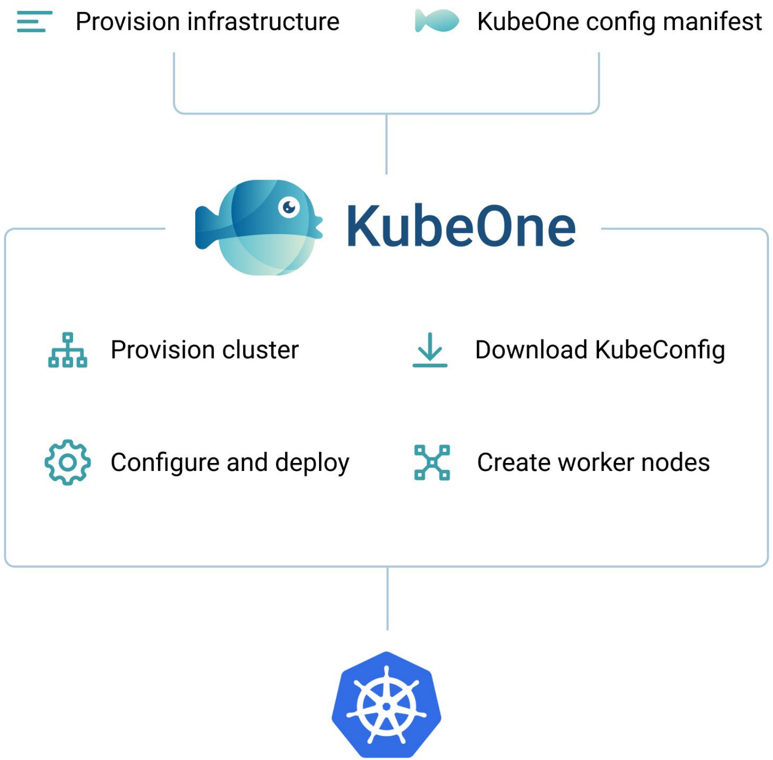
Provision cluster

Download KubeConfig

Configure and deploy

Create worker nodes

Installation process



Provision infrastructure

KubeOne config manifest

KubeOne

Provision cluster

Download KubeConfig

Configure and deploy

Create worker nodes

Demo time!

Create cluster on AWS

Create cluster on AWS

- Step 1: Create instances and infrastructure to be used by Kubernetes
  - KubeOne comes with example Terraform scripts that can be used to get started
- Step 2: Build KubeOne configuration manifest
  - Defines what Kubernetes version will be installed, what machines will be used, how the cluster will be provisioned…
- Step 3: Run `` `kubeone install` `` command
- Step 4: Enjoy!

Building KubeOne Cluster manifest

```yaml
apiVersion: kubeone.io/v1alpha1
kind: KubeOneCluster
versions:
  kubernetes: 1.14.2
cloudProvider:
  name: aws
```

Building KubeOne Cluster manifest

```yaml
apiVersion: kubeone.io/v1alpha1
kind: KubeOneCluster
versions:
  kubernetes: 1.14.2
cloudProvider:
  name: aws
```

Building KubeOne Cluster manifest

```yaml
apiVersion: kubeone.io/v1alpha1
kind: KubeOneCluster
versions:
  kubernetes: 1.14.2
cloudProvider:
  name: aws
```

Building KubeOne Cluster manifest

```
apiVersion: kubeone.io/v1alpha1
kind: KubeOneCluster
versions:
  kubernetes: 1.14.2
cloudProvider:
  name: aws
```

Upgrade process

Upgrade process

- The control plane nodes are upgraded **in-place**
- Upgrading control plane nodes include upgrading:
  - Kubernetes binaries
  - core Kubernetes components
  - all components deployed by KubeOne
- Worker nodes are upgraded by **rolling out** MachineDeployment

Conclusion

Conclusion

KubeOne is a tool for **managing** Kubernetes **cluster lifecycle**

- Find KubeOne on GitHub: **https://github.com/kubermatic/kubeone**
- Follow us on Twitter: @Loodse, @xmudrii, @kron4eg
- Check out Loodse blog: https://loodse.com/blog
- Join `#kubeone` on Kubermatic Slack: http://slack.kubermatic.io

Thank you for your time!