What’s New in Kubernetes 1.17

CLOUD NATIVE
COMPUTING FOUNDATION

Presenters

Bob Killen

1.17 Enhancements Lead

@MrBobbyTables

Guinevere Saenger

1.17 Release Lead

@guincodes

David McKay

1.17 Communications Lead & Moderator

@rawkode

Kubernetes 1.17 - The Chillest Release

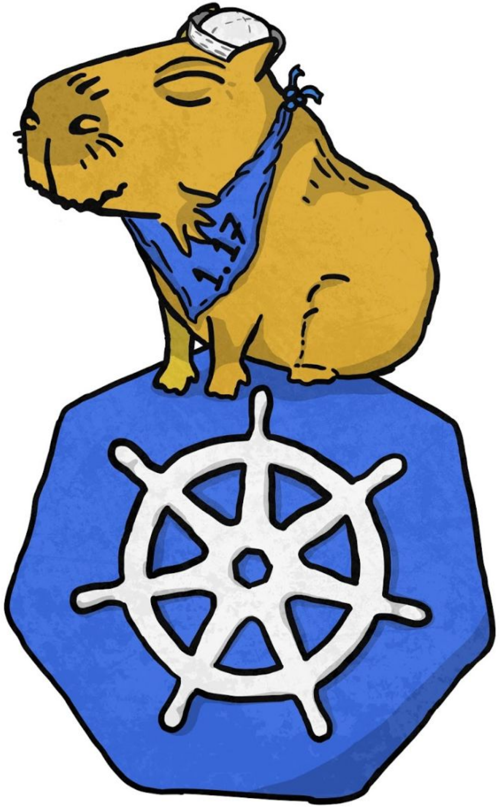

image credit: Alison Dowdney and Tyler Hale

Agenda

- ★ Stability
- ★ Snapshot/Restore Volume Support
- ★ Topology Aware Routing
- ★ SIG Updates
- ★ Q&A

1.17 Enhancements

CLOUD NATIVE
COMPUTING FOUNDATION

Overview

★ 22 total enhancements tracked in 1.17

  ○ 14 Stable Enhancements

  ○ 4 Graduating to Beta

  ○ 4 Introduced Alpha features

Highlights

CLOUD NATIVE
COMPUTING FOUNDATION

Stability

A stability release, really?

Stability (Part 1)

- ★ Taint Node by Condition
- ★ Configurable Pod Process Namespace Sharing
- ★ Schedule DaemonSet Pods by kube-scheduler
- ★ Dynamic Maximum Volume Count
- ★ Kubernetes CSI Topology Support
- ★ Provide Environment Variables Expansion in SubPath Mount

Stability (Part 2)

★ Defaulting of Custom Resources

★ Move Frequent Kubelet Heartbeats To Lease API

★ Break Apart The Kubernetes Test Tarball

★ Add Watch Bookmarks Support

★ Behavior-Driven Conformance Testing

★ Finalizer Protection For Service Load Balancers

★ Avoid Serializing The Same Object Independently For Every Watcher

Snapshot / Restore Volume Support

- **Beta** Feature
- Provides a standardized API design (CRDs) along with enabling base PV snapshot/restore support for CSI volume drivers.
- Underlying support is now enabled (**`VolumeSnapshotDataSource`** Feature Gate), but for most users it will require installation of the external-snapshotter to make use of it.

Snapshot / Restore Volume Support

- **`VolumeSnapshotClass`** - Defines the configuration for how snapshots may be done for a provided csi-driver.
- **`VolumeSnapshot`** - Defines a snapshot of a provided Persistent Volume Claim (PVC).
  - The PVC, PV, and **`VolumeSnapshotClass`** for a **`VolumeSnapshot`** object **MUST** use the same CSI Driver

Blog Post: Kubernetes 1.17 Feature: Kubernetes Volume Snapshot Moves to Beta

Topology Aware Routing

- **Alpha** feature
  - must be manually enabled
- Services can be configured with a set of **`topologyKeys`** that dictate service -> pod routing preferences.

```
apiVersion: v1
kind: service
metadata:
  name: my-awesome-service
spec:
  selector:
    app: my-awesome-app
  ports:
    - port: 443
      targetPort: 443
  topologyKeys:
    - kubernetes.io/hostname
    - kubernetes.io/zone
    - kubernetes.io/region
    - “*”
```

SIG Updates

API MACHINERY

Defaulting of Custom Resources

Allows CRD's to specify default values for properties in the OpenAPI Spec for the type.

MUST use this for defaulting:
`apiextensions.k8s.io/v1`

Usage: Defaulting

Status: Stable

Tracking Issue

Enhancement Proposal

Watch Bookmarks

Add support for watch bookmarks to reduce load on kube-apiserver.

Status: Stable

Tracking Issue

Enhancement Proposal

Less Object Serializations

Avoid serializing the same object independently for every watcher.

Performance Improvement Results:

- ~5% lower cpu-usage
- ~15% less memory usage

Status: Stable

[Tracking Issue]

[Enhancement Proposal]

ARCHITECTURE

Behavior-driven Conformance Testing

Kicks off effort to improve conformance testing.

Status: Stable

[Tracking Issue]

[Enhancement Proposal]

Remove project-wide usage of node role labels

Deprecate usage of unofficial node-role labels:
`node-role.kubernetes.io/*`

NOTE: First party kubernetes projects (kubeadm) **MAY** continue to use them, but nothing else within the project is allowed to **DEPEND** on them.

Status: Alpha

[Tracking Issue]

[Enhancement Proposal]

CLOUD PROVIDER

CLOUD NATIVE
COMPUTING FOUNDATION

Uniform labeling of cloud-providers node labels

Flags beta cloud-provider labels for deprecation.

- `beta.kubernetes.io/instance-type`
  `node.kubernetes.io/instance-type`
- `failure-domain.beta.kubernetes.io/zone`
  `topology.kubernetes.io/zone`
- `failure-domain.beta.kubernetes.io/region`
  `topology.kubernetes.io/region`

Status: Stable

[Tracking Issue]()

[Enhancement Proposal]()

CLUSTER LIFECYCLE

Structured Output for kubeadm

Enables support for kubeadm to output it’s logs in the form of json, yaml, go templates, or unbuffered text.

Status: Alpha

Tracking Issue

Enhancement Proposal

NETWORK

Topology aware routing of services

Services can route traffic based on a predefined list of preferences defined by the **`topologyKeys`** parameter.

Status: Stable

[Tracking Issue]

[Enhancement Proposal]

Add IPv4/IPv6 dual-stack support

Enables support for multiple IPv4/IPv6 addresses to be attached to pods and services.

Status: Alpha

[Tracking Issue]

[Enhancement Proposal]

New Endpoint API

Redesign of the current core/v1 endpoint API to mitigate current performance/scalability issues.

Status: Beta

Tracking Issue

Enhancement Proposal

Finalizer Protection for Service LoadBalancers

Ensures proper cleanup and deletion of service type LoadBalancer objects.

Status: Stable

Tracking Issue

Enhancement Proposal

NODE

CLOUD NATIVE
COMPUTING FOUNDATION

Configurable Pod Process Namespace Sharing

Enables containers within a pod to share the same Process ID(PID) Namespace.

Status: Stable

[Tracking Issue]

[Enhancement Proposal]

Move frequent Kubelet heartbeats to Lease API

Improves the performance and scalability of the node-lifecycle-controller by adding a lightweight ***"lease"*** object and reducing the frequency of node full heartbeat updates.

Status: Stable

Tracking Issue

Enhancement Proposal

SCHEDULING

CLOUD NATIVE
COMPUTING FOUNDATION

Taint Node by Condition

As of 1.12 - Kubernetes automatically applies the right scheduling restriction (e.g. `NoExecute`) based on the node's condition.

Status: Stable

[Tracking Issue]

[Enhancement Proposal]

Schedule DaemonSet Pods by kube-scheduler

As of 1.12 DaemonSets no longer bypass the normal scheduling mechanisms.

Status: Stable

[Tracking Issue]

[Enhancement Proposal]

STORAGE

CLOUD NATIVE
COMPUTING FOUNDATION

Snapshot / Restore Volume Support for Kubernetes

Provides a standardized API design (CRDs) along with enabling base PV snapshot/restore support for CSI volume drivers.

Usage: VolumeSnapshot
Usage: VolumeSnapshotClass

Status: Beta

Tracking Issue

Enhancement Proposal

Dynamic Maximum Volume Count

Moves the amount of maximum attachable volumes from a cluster wide setting, to a dynamic per-node/CSI value.

Status: Stable

Tracking Issue

Enhancement Proposal

CSI Topology Support

Enables topology preferences to be applied to storage-classes by means of the **`allowedTopologies`** parameter.

Usage: Storage Classes: Allowed Topologies

Status: Stable

Tracking Issue

Enhancement Proposal

Environment variables expansion in sub path mount

Provides a means to use environment variables in **`subPath`** mounts by means of a new directive: **`subPathExpr`**.

Usage: Using subPath with expanded environment variables

Status: Stable

Tracking Issue

Enhancement Proposal

In-tree storage plugin to CSI Driver Migration

Migrates the internals of the in-tree volume plugins to call out to their CSI equivalent while maintaining original API compatibility.

Status: Beta

[Tracking Issue]()

[Enhancement Proposal]()

TESTING

CLOUD NATIVE
COMPUTING FOUNDATION

Break apart the kubernetes test tarball

Breaks apart the monolithic "*mondo*" tarball containing all the test binaries needed for Kubernetes, for all platforms into individual artifacts.

Status: Stable

[Tracking Issue]()

[Enhancement Proposal]()

WINDOWS

RunAsUserName for Windows

Adds the Windows equivalent to the linux security context option: **RunAsUser**

Usage: Configure RunAsUserName for Windows Pods and containers

Status: Beta

Tracking Issue

Enhancement Proposal

Questions?

CLOUD NATIVE
COMPUTING FOUNDATION

Thank You

CLOUD NATIVE
COMPUTING FOUNDATION