Taints and tolerations

Toader Sebastian

BANZAICLOUD

Concepts

- ➔ K8s allows to mark ("taint") a node so that no pods can be scheduled to it unless a pod explicitly tolerates the taint
- ➔ Taints allow a node to repel a set of pods
- ➔ Tolerations are applied to pods, and allow (but do not require) the pods to be scheduled onto nodes with matching taints

Taints

- Add a taint to a node

```
kubectl taint nodes nodeA key1=value1:NoSchedule
```

- Taint has a **key**, **value and effect**
  - NoSchedule
  - PreferNoSchedule
  - NoExecute

- Remove taint from node

```
kubectl taint nodes nodeA key1:NoSchedule-
```

Tolerations

- ➔ Specify toleration for a pod in the PodSpec

```yaml
tolerations:
- key: "key1"
  operator: "Equal"
  value: "value1"
  effect: "NoSchedule"
```

Tolerations

- ➔ A toleration “matches” a taint if the keys are the same and the effects are the same, and **operator** is:
  - “**Exists**” (in this case no **value** should be specified), or
  - “**Equal**” and the **value**s are equal

```yaml
tolerations:
- key: "key1"
  operator: "Exists"
  effect: "NoSchedule"
```

Tolerations

- A toleration “matches” a taint if the keys are the same and the effects are the same, and **operator** is:
  - “**Exists**” (in this case no **value** should be specified), or
  - “**Equal**” and the **value**s are equal

```
tolerations:
- key: "key1"
  operator: "Equal"
  value: "value1"
  effect: "NoSchedule"
```

Tolerations

- “Operator” defaults to “Equal” if not specified
- An empty key with operator **Exists** matches all keys, values and effects which means this will tolerate everything

```yaml
tolerations:
- operator: "Exists"
```

Tolerations

- An empty effect matches all effects with key

```
tolerations:
- key: "key1"
  operator: "Equal"
  value: "value1"
```

BANZAICLOUD

Effect

- **NoSchedule**
  - no pods can be scheduled to the node unless a pod explicitly tolerates the taint
- **PreferNoSchedule**
  - try to avoid placing a pod that does not tolerate the taint on the node, but it is not required
- **NoExecute**
  - Pods running on the node that do not tolerate the taint are evicted
  - **tolerationSeconds** - delayed pod eviction

Multiple taints and toleration

- Multiple taints can be applied to a node

```
kubectl taint nodes nodeA key1=value1:NoSchedule
kubectl taint nodes nodeA key1=value1:NoExecute
kubectl taint nodes nodeA key2=value2:NoSchedule
```

- Multiple tolerations can be applied to a pod

```
tolerations:
- key: "key1"
  operator: "Equal"
  value: "value1"
  effect: "NoSchedule"
- key: "key1"
  operator: "Equal"
  value: "value1"
  effect: "NoExecute"
```

Taint nodes by condition

- Kubernetes 1.6 introduced alpha support for representing node problems through taints
- Enabled through **TaintNodesByCondition**
  - disabled by default
- Promoted to beta in Kubernetes 1.12 and
  - enabled by default
- Node lifecycle controller automatically creates taints corresponding to Node conditions (e.g.: `node.kubernetes.io/network-unavailable,` `node.kubernetes.io/not-ready`)
- Only taints nodes with **NoSchedule** effect

Taint based evictions

- **NoExecute** taint effect impacts pods already running on a node
  - pods that do not tolerate the taint are evicted
  - pods that tolerate the taint without specifying tolerationSeconds in their toleration specification remain bound forever
  - pods that tolerate the taint with a specified tolerationSeconds remain bound for the specified amount of time
- The **TaintBasedEvictions** feature gate introduced in Kubernetes 1.6 as alpha feature automatically taints nodes with NoExecute effect if certain condition is true
  - disabled by default

DaemonSets

- DaemonSet pods are created with **NoExecute** tolerations for the following taints with no **tolerationSeconds**:
  - node.alpha.kubernetes.io/unreachable
  - node.kubernetes.io/not-ready
- DaemonSet pods are never evicted due to these problems

DaemonSets

- DaemonSet controller automatically adds the following **NoSchedule** tolerations to all daemons, to prevent DaemonSets from breaking.
  - node.kubernetes.io/memory-pressure
  - node.kubernetes.io/disk-pressure
  - node.kubernetes.io/out-of-disk
  - node.kubernetes.io/unschedulable
  - node.kubernetes.io/network-unavailable

Demo

Do you have any questions?

We are hiring slide

➔ Are you interested in

- containers, clouds and Kubernetes
- work in/learn Golang
- actively contribute to high-profile open source projects
- troubleshooting complex issues in distributed systems
- defining and shaping the future of application deployments

https://banzaicloud.com/careers/

BANZAI**CLOUD**