Control scheduling with node taints

STANDARD (/KUBERNETES-ENGINE/DOCS/CONCEPTS/TYPES-OF-CLUSTERS)

This page provides an overview of node taints (https://kubernetes.io/docs/concepts/configuration/taint-and-toleration/) on Google Kubernetes Engine (GKE). When you schedule workloads to be deployed on your cluster, node taints help you control which nodes they are allowed to run on.

Overview

When you submit a workload to run in a cluster, the scheduler determines where to place the Pods associated with the workload. The scheduler is free to place a Pod on any node that satisfies the Pod's CPU, memory, and custom resource requirements.

If your cluster runs a variety of workloads, you might want to exercise some control over which workloads can run on a particular pool of nodes.

A **node taint** lets you mark a node so that the scheduler avoids or prevents using it for certain Pods. A complementary feature, *tolerations*, lets you designate Pods that can be used on "tainted" nodes.

Taints and tolerations work together to ensure that Pods are not scheduled onto inappropriate nodes.

Taints are *key-value pairs* associated with an *effect*. The following table lists the available effects:

Effect	Description
`NoSchedule`	Pods that do not tolerate this taint are not scheduled on the node; existing Pods are not evicted from the node.
`PreferNoSchedule`	Kubernetes avoids scheduling Pods that do not tolerate this taint onto the node.
`NoExecute`	The Pod is evicted from the node if it is already running on the node, and is not scheduled onto the node if it is not yet running on the node.

Note: Some system Pods (for example, **`kube-proxy`** and **`fluent-bit`**) tolerate all **`NoExecute`** and

NoSchedule taints, and will not be evicted.

Advantages of setting node taints in GKE

You can add node taints to clusters and nodes in GKE or by using the **kubectl taint** (https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#taint) command. Specifying node taints in GKE has several advantages over **kubectl**:

- Taints are preserved when a node is restarted or replaced.
- Taints are created automatically when a node is added to a node pool or cluster.
- Taints are created automatically during cluster autoscaling.

Before you begin

Before you start, make sure you have performed the following tasks:

- Enable the Google Kubernetes Engine API.

  Enable Google Kubernetes Engine API (https://console.cloud.google.com/flows/enableapi?apiid

- If you want to use the Google Cloud CLI for this task, install (/sdk/docs/install) and then initialize (/sdk/docs/initializing) the gcloud CLI.

★ **Note:** For existing gcloud CLI installations, make sure to set the **compute/region** and **compute/zone** properties (/sdk/docs/properties#setting_properties). By setting default locations, you can avoid errors in gcloud CLI like the following: **One of [--zone, --region] must be supplied: Please specify location**.

Create a cluster with node taints

When you create a cluster in GKE, you can assign node taints to the cluster. This assigns the taints to *all nodes created with the cluster*.

Note: If you create a node pool, the node pool does not inherit taints from the cluster. If you want taints on the node pool, you must use the `--node-taints` flag when you create the node pool.

If you create a Standard cluster with node taints that have the `NoSchedule` effect or the `NoExecute` effect, GKE can't schedule some GKE managed components, such as `kube-dns` or `metrics-server` on the default node pool that GKE creates when you create the cluster. GKE can't schedule these components because they don't have the corresponding tolerations for your node taints. You must add a new node pool that satisfies one of the following conditions:

- No taints
- A taint that has the `PreferNoSchedule` effect
- The `components.gke.io/gke-managed-components=:NoSchedule` taint

Any of these conditions allow GKE to schedule GKE managed components in the new node pool.

For instructions, refer to Isolate workloads on dedicated nodes (/kubernetes-engine/docs/how-to/isolate-workloads-dedicated-nodes).

gcloudConsole (#console)API (#api)
(#gcloud)

To create a cluster with node taints, run the following command:

```
gcloud container clusters create CLUSTER_NAME \
    --node-taints KEY=VALUE:EFFECT
```

Replace the following:

- *CLUSTER_NAME*: the name of the new cluster.
- *EFFECT*: one of the following effect values (#effects): `PreferNoSchedule`, `NoSchedule`, or `NoExecute`.
- *KEY*=*VALUE*: a key-value pair associated with the *EFFECT*.

For example, the following command applies a taint that has a key-value of `dedicated=experimental` with an effect of `PreferNoSchedule`:

```
gcloud container clusters create example-cluster \
    --node-taints dedicated=experimental:PreferNoSchedule
```

Create a node pool with node taints

When you apply a taint to a node, only Pods that tolerate the taint are allowed to run on the node. In a GKE cluster, you can apply a taint to a node pool, which applies the taint to all nodes in the pool.

To create a node pool with node taints, you can use the Google Cloud CLI, the Google Cloud console, or the GKE API.

gcloudConsole (#console)API (#api)
(#gcloud)

To create a node pool with node taints, run the following command:

```
gcloud container node-pools create POOL_NAME \
    --cluster CLUSTER_NAME \
    --node-taints KEY=VALUE:EFFECT
```

Replace the following:

- *POOL_NAME*: the name of the node pool to create.
- *CLUSTER_NAME*: the name of the cluster in which the node pool is created.
- *EFFECT*: one of the following effect values (#effects): **PreferNoSchedule**, **NoSchedule**, or **NoExecute**.
- *KEY*=*VALUE*: a key-value pair associated with the *EFFECT*.

For example, the following command creates a node pool on an existing cluster and applies a taint that has a key-value of **dedicated=experimental** with a **NoSchedule** effect:

```
gcloud container node-pools create example-pool --cluster example-cluste
    --node-taints dedicated=experimental:NoSchedule
```

This command creates a node pool and applies a taint that has key-value of **`special=gpu`** with a **`NoExecute`** effect:

```
gcloud container node-pools create example-pool-2 --cluster example-clus
    --node-taints special=gpu:NoExecute
```

Configure Pods to tolerate a taint

You can configure Pods to tolerate a taint by including the **`tolerations`** field in the Pods' specification. Here's a portion of a Pod specification (https://v1-21.docs.kubernetes.io/docs/reference/generated/kubernetes-api/v1.21/#pod-v1-core).

This Pod can be scheduled on a node that has the **`dedicated=experimental:NoSchedule`** taint:

```
tolerations:
- key: dedicated
  operator: Equal
  value: experimental
  effect: NoSchedule
```

Add a taint to an existing node

You can add taints to an existing node by using the **`kubectl taint`** (https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#taint) command:

```
kubectl taint nodes NODE_NAME KEY=VALUE:EFFECT
```

For example, the following command applies a taint that has a key-value of **`dedicated=experimental`** with a **`NoSchedule`** effect to the **`mynode`** node:

```
kubectl taint nodes mynode dedicated=experimental:NoSchedule
```

You can also add taints to nodes that have a specific label by using the `-l` selector along with the specified label and value:

```
kubectl taint nodes -l LABEL=LABEL_VALUE KEY=VALUE:EFFECT
```

Note: In GKE, all node pools have a label in the following format (where ***POOL_NAME*** is the name of the node pool):

```
cloud.google.com/gke-nodepool:POOL_NAME
```

For example, the following command adds a taint with key **`dedicated-pool`** to GKE nodes in the **`my_pool`** node pool:

```
kubectl taint nodes -l cloud.google.com/gke-nodepool=my_pool dedicated-pool=my
```

Inspect taints for a node

To see the taints for a node, use the **`kubectl`** command-line tool.

1. Get a list of all nodes in your cluster by running the following command:

   ```
   kubectl get nodes
   ```

2. Inspect a node by running the following command:

   ```
   kubectl describe node NODE_NAME
   ```

3. In the returned output, look for the `Taints` field. The output is similar to the following:

   ```
   Taints:  dedicated-pool=mypool:NoSchedule
   ```

Remove a taint from a node

You can use `kubectl taint` to remove taints. You can remove taints by key, key-value, or key-effect.

For example, the following command removes all the taints with the `dedicated` key from the `mynode` node:

```
kubectl taint nodes mynode dedicated-
```

Note: Starting in GKE version 1.22, cluster autoscaler combines existing node and node pool information to represent the whole node pool. Cluster autoscaler detects node pool updates and manual node changes to scale cluster up.

Remove all taints from a node pool

To remove all taints from a node pool, run the following command:

```
gcloud beta container node-pools update POOL_NAME \
--node-taints="" \
--cluster=CLUSTER_NAME
```

Replace the following:

- *POOL_NAME*: the name of the node pool to change.
- *CLUSTER_NAME*: the name of the cluster in which the node pool was created.

What's next

- Learn more about node pools (/kubernetes-engine/docs/concepts/node-pools).
- Read the Kubernetes documentation for taints and tolerations (https://kubernetes.io/docs/concepts/configuration/taint-and-toleration/).

- Read the `kubectl taint` documentation (https://kubernetes.io/docs/reference/generated/kubectl/kubectl-commands#taint).

Last updated 2023-01-24 UTC.