# Taints and Tolerations in Kubernetes

Taints and tolerations are Kubernetes features that allow you to control which nodes are eligible to schedule certain pods. They work together to ensure that pods are not scheduled onto inappropriate nodes.

## Taints

A taint is applied to a node and can repel a set of pods unless those pods have a matching toleration. Taints consist of a key, value, and effect, and can have one of the following effects:

- NoSchedule: Pods will not be scheduled on the node unless they have a toleration matching the taint.

- PreferNoSchedule: The system will try to avoid placing a pod on the node but it is not guaranteed.

- NoExecute: Pods will be evicted from the node if they do not have a toleration matching the taint.

Command to add a taint to a node:

kubectl taint nodes [NODE\_NAME] [KEY]=[VALUE]:[EFFECT]

## Tolerations

A toleration is applied to a pod and allows (but does not require) the pods to schedule onto nodes with matching taints. Tolerations are specified in the pod specification and consist of keys, values, and effects, and optionally a duration that the pod can tolerate the taint.

Example of a toleration in a pod spec:

tolerations:  
- key: "key"  
 operator: "Equal"  
 value: "value"  
 effect: "NoSchedule"

## Use Cases

Taints and tolerations are useful in several scenarios, such as ensuring that special-purpose or high-resource nodes are reserved for appropriate pods, separating nodes with sensitive data or hardware from general workload pods, and managing nodes with software or hardware that is incompatible with certain pods.