Taint and tolerations (1 and 2 session)

Scheduler puts container/pod , where it gets suitable for pods which doesn't define where it has to run in which node. In this case, nodeselector doesn't work , that's why taint and toleration came into the light.

If node got tainted, which has toleration token only will able to run pod.

```
3) Taint and tolerations Based Scheduling:

        1) it is giving dedicate tency.
```

Why pods don't go on master nodes?

```
root@master01:~# kubectl describe  node master01 |  grep  -i taint
Taints:             node-role.kubernetes.io/control-plane:NoSchedule
root@master01:~#
```

```
root@master01:~#
root@master01:~# kubectl taint node  worker01   payment:NoSchedule
```

now, it is not normal node

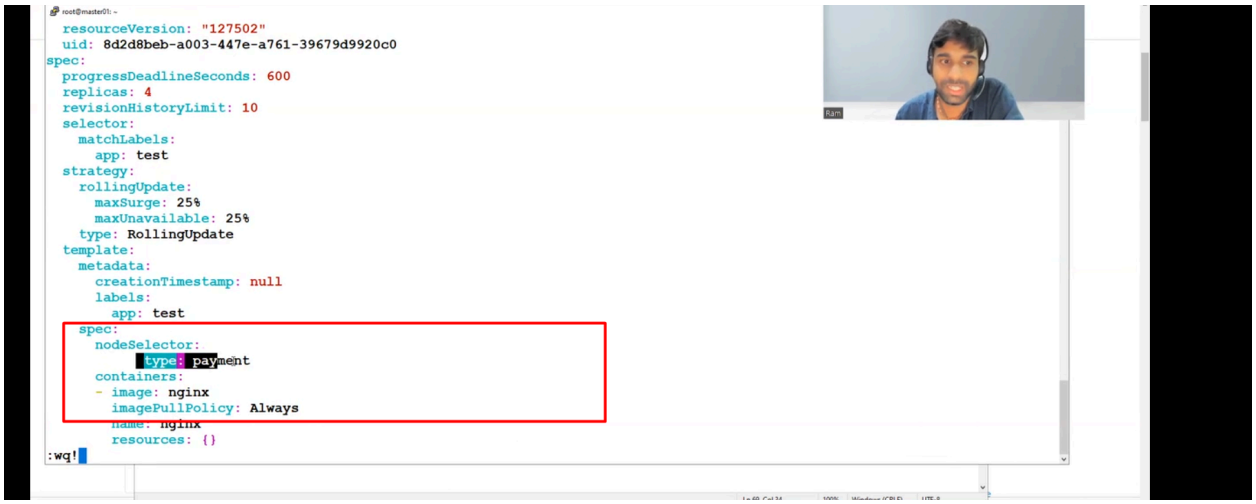
Join Stack Overflow to find the best answer to your technical question, help others answer theirs.
Sign up with email | Sign up with Google | Sign up with GitHub | Sign up with Facebook

Toleration

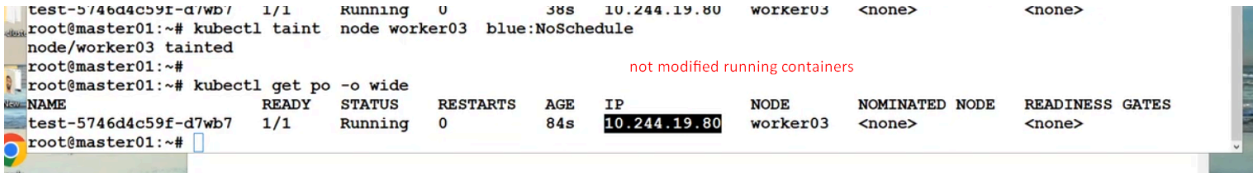
```
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: krnetwork
  name: krnetwork
spec:
  replicas: 1
  selector:
    matchLabels:
      app: krnetwork
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: krnetwork
    spec:
      tolerations:
        - key: payment
          effect: NoSchedule
          operator: Exists
      containers:
      - image: nginx
        name: nginx
        resources: {}
status: {}
:wq
```

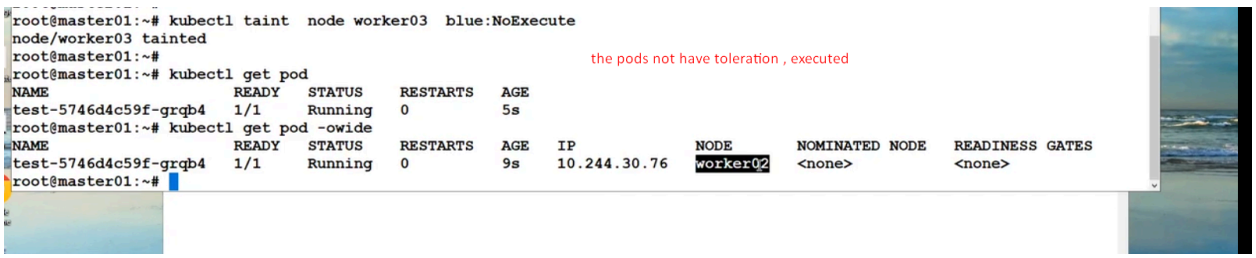
Join Stack Overflow to find the best answer to your technical question, help
Sign up with email | Sign up with Google | Sign up with GitHub | Sign up with Facebook

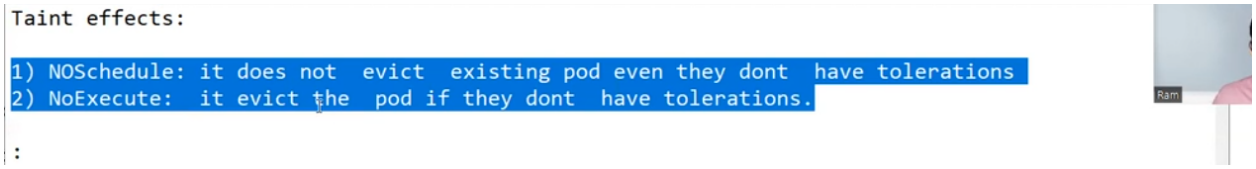
If you replicate the pods , pods will run any nodes. It isn't mandatory to pods run on tainted node , then you have to add nodeselector parameter in manifest.



2nd session







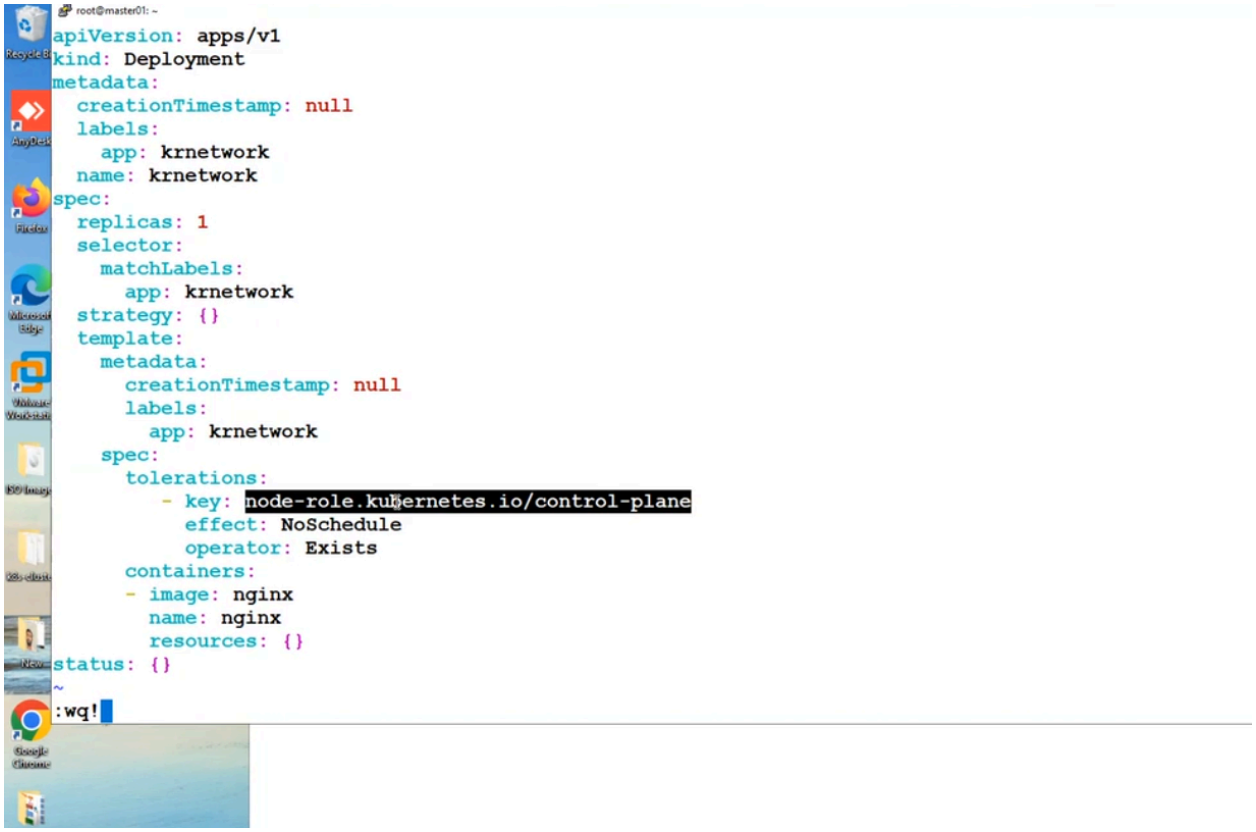
```
root@master01:~# kubectl taint  node worker03  blue:NoExecute-
node/worker03 untainted
```

untaint

```
      app: krnetwork
  spec:
    tolerations:
      - key: tier
        value: db
        effect: NoSchedule
        operator: Equal
    containers:
    - image: nginx
      name: nginx
      resources: {}
status: {}
```

```
tolerations:


Exists: Key/effect  match
Equal= key=value match
```


root@master01: ~
apiVersion: apps/v1
kind: Deployment
metadata:
creationTimestamp: null
labels:
app: krnetwork
name: krnetwork
spec:
replicas: 1
selector:
matchLabels:
app: krnetwork
strategy: {}
template:
metadata:
creationTimestamp: null
labels:
app: krnetwork
spec:
tolerations:
- key: node-role.kubernetes.io/control-plane
effect: NoSchedule
operator: Exists
containers:
- image: nginx
name: nginx
resources: {}
status: {}
~
:wq!