

1.1 Kubernetes Dashboard

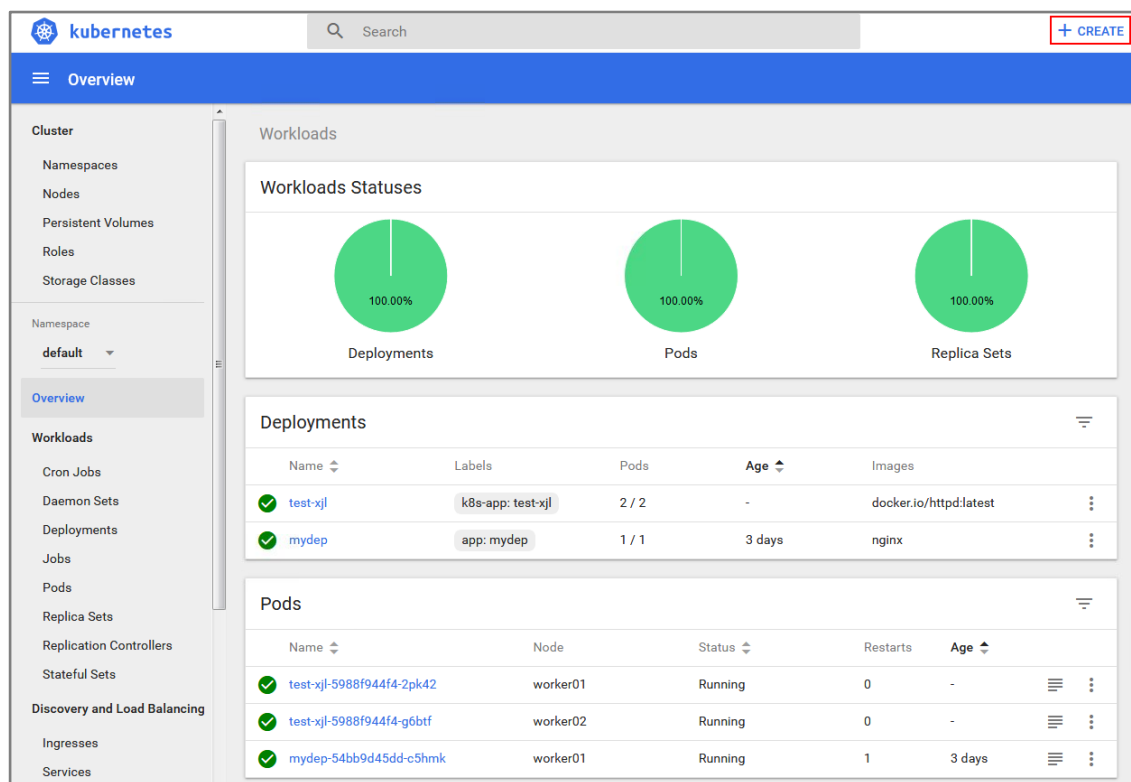
1.1.1 登录 Dashboard

在本实验中，使用以下格式登录浏览器，输入 URL：“k8s-master 的 IP 地址:30090”，这里是 192.168.137.11:30090

具体可回顾搭建指南中的 dashboard.yaml（使用 Nodeport 方式暴露 URL）

1.1.2 “CREATE AN APP” 方式部署 Deployment

步骤 1 进入 Kubernetes Dashboard 界面，点击右上角 “+CREATE” 按钮，进入 “Resource creation” 界面。



步骤 2 选择 “CREATE AN APP”，基本信息填写如下规划参数：

- App name: 填写 “k8s-dashboard-app”。
- Container image: 填写 “docker.io/nginx:latest”。
- Number of pods: 填写 “2”。
- Service: 选择 “External”。

- Port: 填写 “8080” 。
- Target port: 填写 “80” 。
- Protocol: 选择 “TCP” 。

CREATE FROM TEXT INPUT
CREATE FROM FILE
CREATE AN APP

App name *
k8s-dashboard-app
17 / 24
An 'app' label with this value will be added to the Deployment and Service that get deployed. [Learn more](#)

Container image *
docker.io/nginx:latest
Enter the URL of a public image on any registry, or a private image hosted on Docker Hub or Google Container Registry. [Learn more](#)

Number of pods *
2
A Deployment will be created to maintain the desired number of pods across your cluster. [Learn more](#)

Service *
External
Optionally, an internal or external Service can be defined to map an incoming Port to a target Port seen by the container. The internal DNS name for this Service will be: k8s-dashboard-app. [Learn more](#)

Port *
8080
Target port *
80
Protocol *
TCP

Port
Target port
Protocol *
TCP

SHOW ADVANCED OPTIONS

DEPLOY
CANCEL

步骤 3 点击 “SHOW ADVANCED OPTIONS” ，高级选项填写如下规划参数：

- Description: 填写 “k8s-dashboard-hcip” 。
- Labels: Key 填写 “release” ， Value 填写 “1.0” 。

Description
k8s-dashboard-hcip

Labels

Key	Value
k8s-app	k8s-dashboard-app
release	1.0

- Namespace: 点击右边 “ ” ，在下拉列表中，点击 “Create a new namespace” ，弹出窗口中，填写 “k8s-dashboard-app” 后，点击 “CREATE” 。

Namespace *
default

k8s-dashboard
kube-node-lease
kube-public
kube-system
Create a new namespace...

Create a new namespace

The new namespace will be added to the cluster.

Namespace name *

k8s-dashboard-app 17 / 63

A namespace with the specified name will be added to the cluster. [Learn more](#)

CREATE CANCEL

Namespace *
k8s-dashboard-app

- Image Pull Secret: 只有私有镜像才需要填写此项，本实验 pull 的公有镜像，无需填写。
- 其他参数无需填写，也可以根据自己的需要进行填写。

步骤 4 参数填写完毕后，点击“DEPLOY”。

Environment variables

Name	Value
HIDE ADVANCED OPTIONS	

DEPLOY CANCEL

步骤 5 点击创建的应用的名称，进入对应的详情页。

Cluster

Namespaces

Nodes

Persistent Volumes

Roles

Storage Classes

Namespace

k8s-dashboard-app

Overview

Workloads

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Discovery and Load Balancing

Workloads

Workloads Statuses

100.00%

Deployments

100.00%

Pods

100.00%

Replica Sets

Deployments

Name	Labels	Pods	Age	Images
k8s-dashboard-app	k8s-app: k8s-dashboard-app release: 1.0	2 / 2	-	docker.io/nginx:latest

Pods

Name	Node	Status	Restarts	Age
k8s-dashboard-app-697468b99-rlmm5	worker01	Running	0	-
k8s-dashboard-app-697468b99-v7vj6	worker02	Running	0	-

ments > k8s-dashboard-app

SCALE EDIT DELETE

Details

Name: k8s-dashboard-app

Namespace: k8s-dashboard-app

Labels: k8s-app: k8s-dashboard-app release: 1.0

Annotations: deployment.kubernetes.io/revision: 1 description: k8s-dashboard-hcip

Creation Time: 2019-07-01T08:29 UTC

Selector: k8s-app: k8s-dashboard-app release: 1.0

Strategy: RollingUpdate

Min ready seconds: 0

Revision history limit: 10

Rolling update strategy: Max surge: 25%, Max unavailable: 25%

Status: 2 updated, 2 total, 2 available, 0 unavailable

New Replica Set

Name	Labels	Pods	Age	Images
k8s-dashboard-app-69746	k8s-app: k8s-dashboard-app pod-template-hash: 697... release: 1.0	2 / 2	-	docker.io/nginx:latest

Sets > k8s-dashboard-app-697468b99
LOGS
SCALE
EDIT
DELETE

Details

Name: k8s-dashboard-app-697468b99

Namespace: k8s-dashboard-app

Labels: k8s-app: k8s-dashboard-app pod-template-hash: 697468b99 release: 1.0

Annotations: deployment.kubernetes.io/desired-replicas: 2 deployment.kubernetes.io/max-replicas: 3 deployment.kubernetes.io/revision: 1 description: k8s-dashboard-hcip

Creation Time: 2019-07-01T08:29 UTC

Selector: k8s-app: k8s-dashboard-app pod-template-hash: 697468b99 release: 1.0

Images: docker.io/nginx:latest

Status

Pods: 2 running

Pods

Name	Node	Status	Restarts	Age
k8s-dashboard-app-697468b99-rlmm5	worker01	Running	0	-
k8s-dashboard-app-697468b99-v7vj6	worker02	Running	0	-

k8s-dashboard-app-697468b99-rlmm5
EXEC
LOGS
EDIT
DELETE

Details

Name: k8s-dashboard-app-697468b99-rlmm5

Namespace: k8s-dashboard-app

Labels: k8s-app: k8s-dashboard-app pod-template-hash: 697468b99 release: 1.0

Annotations: description: k8s-dashboard-hcip

Creation Time: 2019-07-01T08:29 UTC

Status: Running

QoS Class: BestEffort

Network

Node: worker01

IP: 10.244.1.6

Containers

k8s-dashboard-app

Image: docker.io/nginx:latest

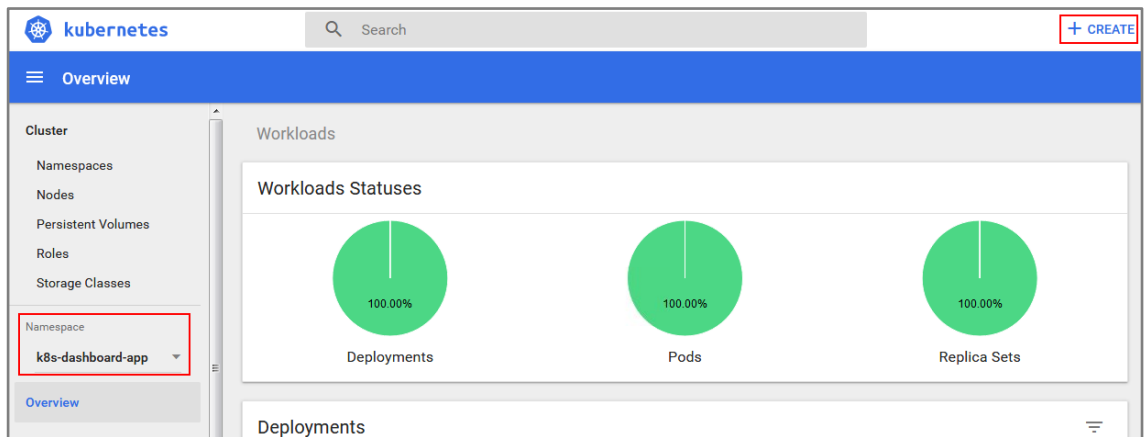
Environment variables: -

Commands: -

Args: -

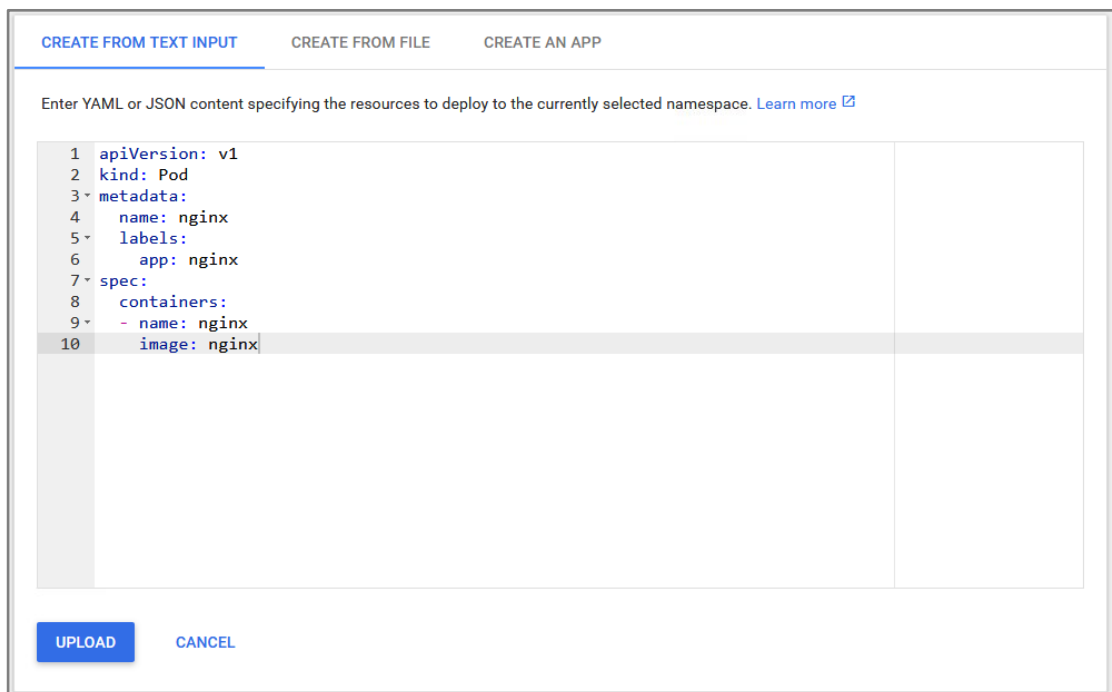
1.1.3 “CREATE FROM TEXT INPUT” 方式部署 Pod

- 步骤 1** 进入 Kubernetes Dashboard 界面，Namespace 选择 Dashboard 实验第一小节创建的 “k8s-dashboard-app”，选择后点击，右上角 “+CREATE” 按钮。



步骤 2 点击 “CREATE FROM TEXT INPUT” ， 输入如下文本信息：

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx
  labels:
    app: nginx
spec:
  containers:
  - name: nginx
    image: nginx
```



步骤 3 输入完成后，点击 “UPLOAD” ， 等待 Pod 创建成功。

UPLOAD

CANCEL

Cluster

- Namespaces
- Nodes
- Persistent Volumes
- Roles
- Storage Classes

Namespace

k8s-dashboard-app

Overview

Workloads

- Cron Jobs
- Daemon Sets
- Deployments
- Jobs
- Pods
- Replica Sets
- Replication Controllers
- Stateful Sets
- Discovery and Load Balancing
- Ingresses

Workloads

Workloads Statuses

Deployments

Pods

Replica Sets

Deployments

Name	Labels	Pods	Age	Images
✓ k8s-dashboard-app	k8s-app: k8s-dashboard- release: 1.0	2 / 2	-	docker.io/nginx:latest

Pods

Name	Node	Status	Restarts	Age
🔍 nginx	worker02	Waiting: ContainerCreating	0	-
✓ k8s-dashboard-app-697468b99-rlmm5	worker01	Running	0	-
✓ k8s-dashboard-app-697468b99-v7vj6	worker02	Running	0	-

Cluster

- Namespaces
- Nodes
- Persistent Volumes
- Roles
- Storage Classes

Namespace

k8s-dashboard-app

Overview

Workloads

- Cron Jobs
- Daemon Sets
- Deployments
- Jobs
- Pods
- Replica Sets
- Replication Controllers
- Stateful Sets
- Discovery and Load Balancing
- Ingresses

Workloads

Workloads Statuses

Deployments

Pods

Replica Sets

Deployments

Name	Labels	Pods	Age	Images
✓ k8s-dashboard-app	k8s-app: k8s-dashboard- release: 1.0	2 / 2	-	docker.io/nginx:latest

Pods

Name	Node	Status	Restarts	Age
✓ nginx	worker02	Running	0	-
✓ k8s-dashboard-app-697468b99-rlmm5	worker01	Running	0	-
✓ k8s-dashboard-app-697468b99-v7vj6	worker02	Running	0	-

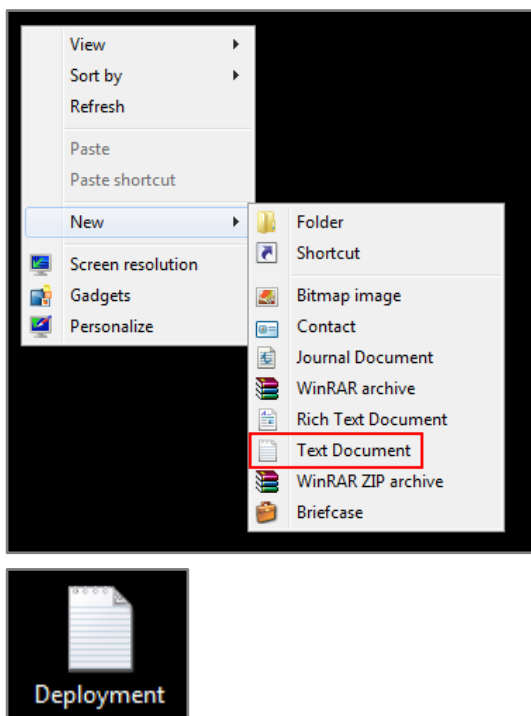
步骤 4 点击刚刚创建的 Pod 名称，可以看到 Pod 的详细信息。

Pods				
Name	Node	Status	Restarts	Age
✓ nginx	worker02	Running	0	-

Details	
Name: nginx	Network
Namespace: k8s-dashboard-app	Node: worker02
Labels: app: nginx	IP: 10.244.2.6
Creation Time: 2019-07-01T08:49 UTC	
Status: Running	
QoS Class: BestEffort	
Containers	
nginx	
Image: nginx	
Environment variables: -	
Commands: -	
Args: -	

1.1.4 “CREATE FROM FILE” 方式部署 Deployment

步骤 1 在客户端（以 Windows 7 PC 机为例）本地，右击新建一个名称为 “Deployment” 的 txt 文件。



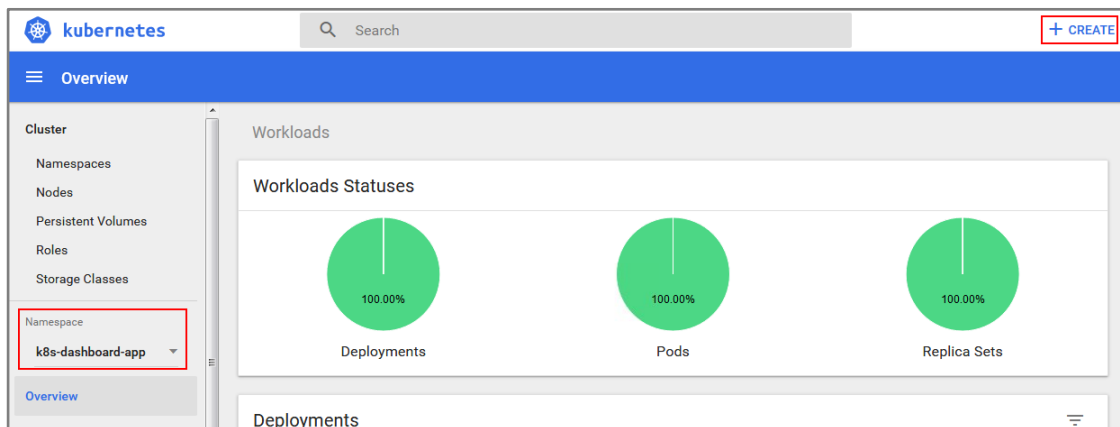
步骤 2 双击打开 txt 文件，并在文件中添加如下图所示文本，并保存。

```
apiVersion: apps/v1
kind: Deployment
metadata:
```

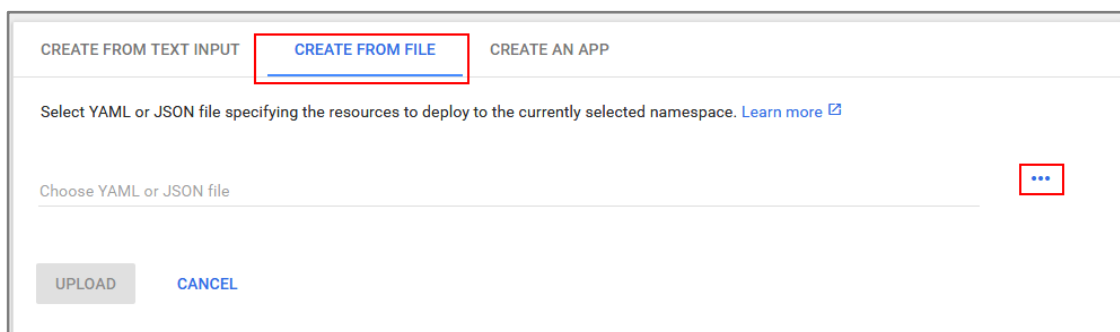


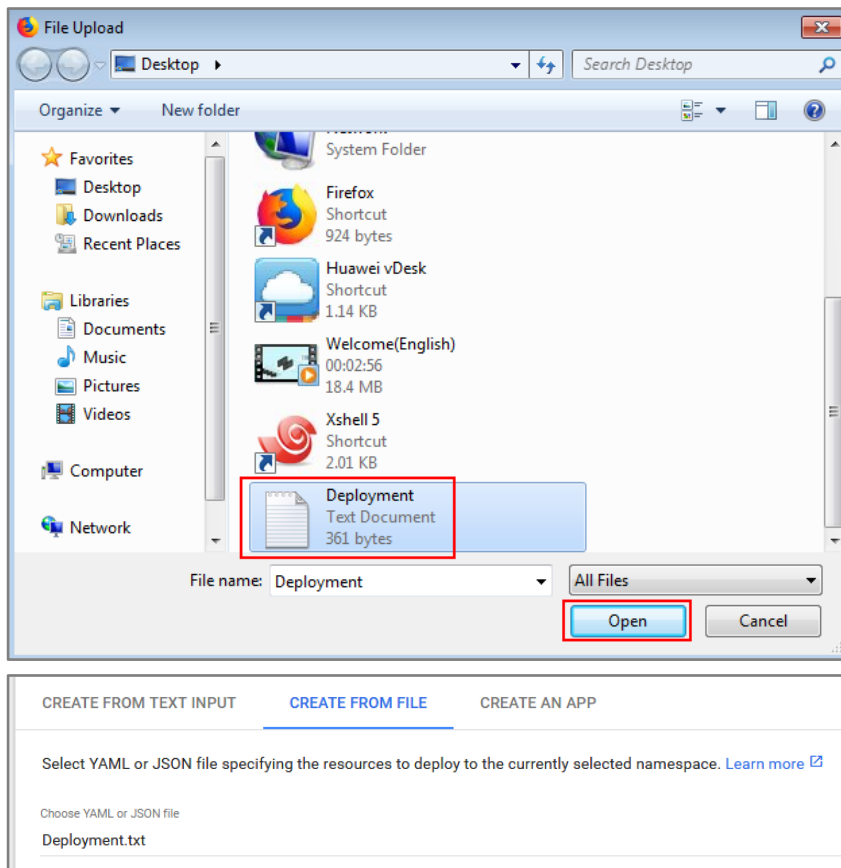
```
name: nginx-deployment
labels:
  app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx
          ports:
            - containerPort: 80
```

步骤 3 进入 Kubernetes Dashboard 界面, Namespace 选择 Dashboard 实验第一小节创建的 “k8s-dashboard-app”, 选择后点击, 右上角 “+CREATE” 按钮。

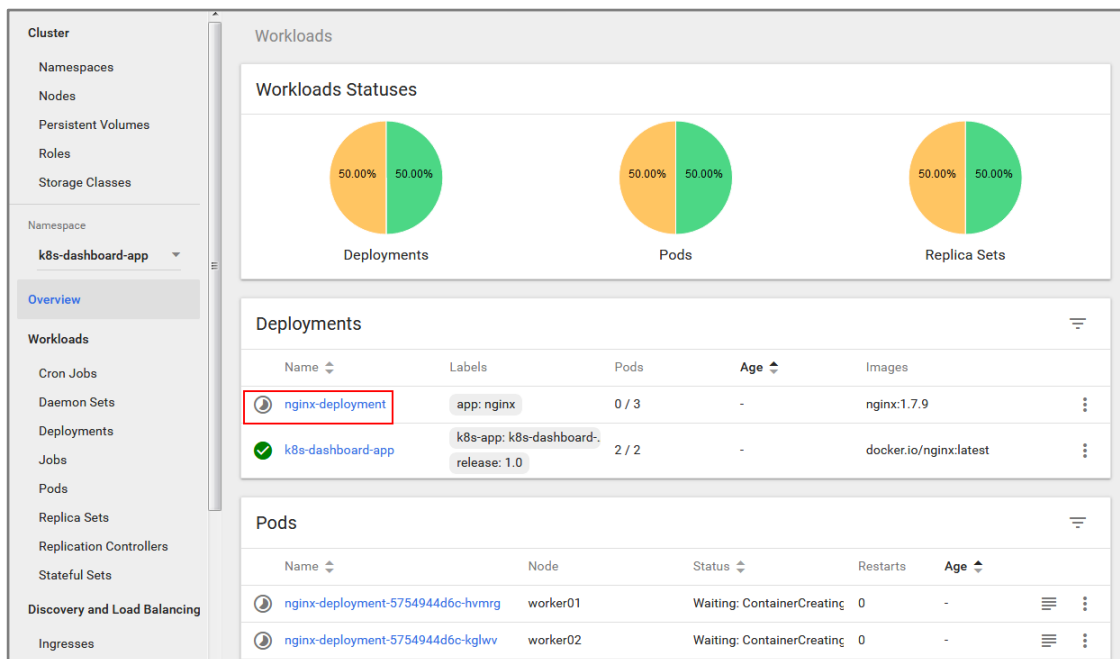


步骤 4 选择 “CREATE FROM FILE”, 点击 “...” , 选中刚刚创建保存的 txt 文件。





步骤 5 点击“UPLOAD”，等待 Deployment 创建成功。



步骤 6 点击刚刚创建的 Deployment 名称，可以查看 Deployment 详细信息。

Cluster
Namespaces
Nodes
Persistent Volumes
Roles
Storage Classes
Namespace
k8s-dashboard-app
Overview
Workloads
Cron Jobs
Daemon Sets
Deployments
Jobs
Pods
Replica Sets
Replication Controllers
Stateful Sets
Discovery and Load Balancing
Ingresses
Services

Workloads

Workloads Statuses

100.00%

Deployments

100.00%

Pods

100.00%

Replica Sets

Deployments

Name	Labels	Pods	Age	Images
nginx-deployment	app: nginx	3 / 3	-	nginx:1.7.9
k8s-dashboard-app	k8s-app: k8s-dashboard- release: 1.0	2 / 2	-	docker.io/nginx:latest

Pods

Name	Node	Status	Restarts	Age
nginx-deployment-5754944d6c-hvmrg	worker01	Running	0	-
nginx-deployment-5754944d6c-kglwv	worker02	Running	0	-
nginx-deployment-5754944d6c-vmzcx	worker02	Running	0	-