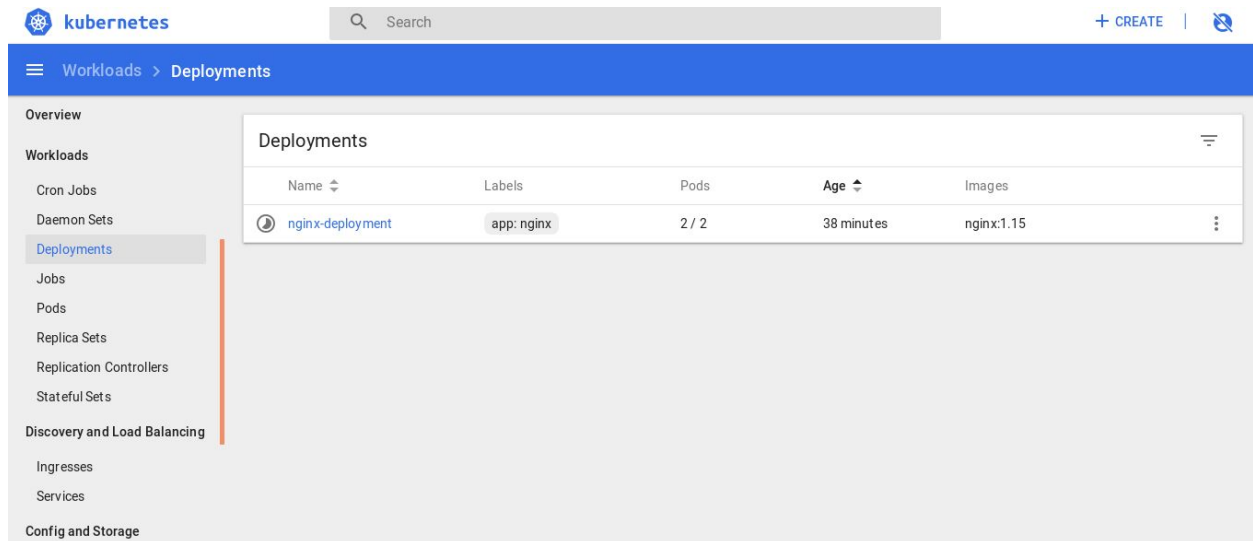


Rolling Updates from the UI

If there is a need to update the docker Image on the deployed applications. This can easily be achieved from the Dashboard/Command Line.

Login to the Kubernetes Dashboard from the Link provided in K8s Lab instructions

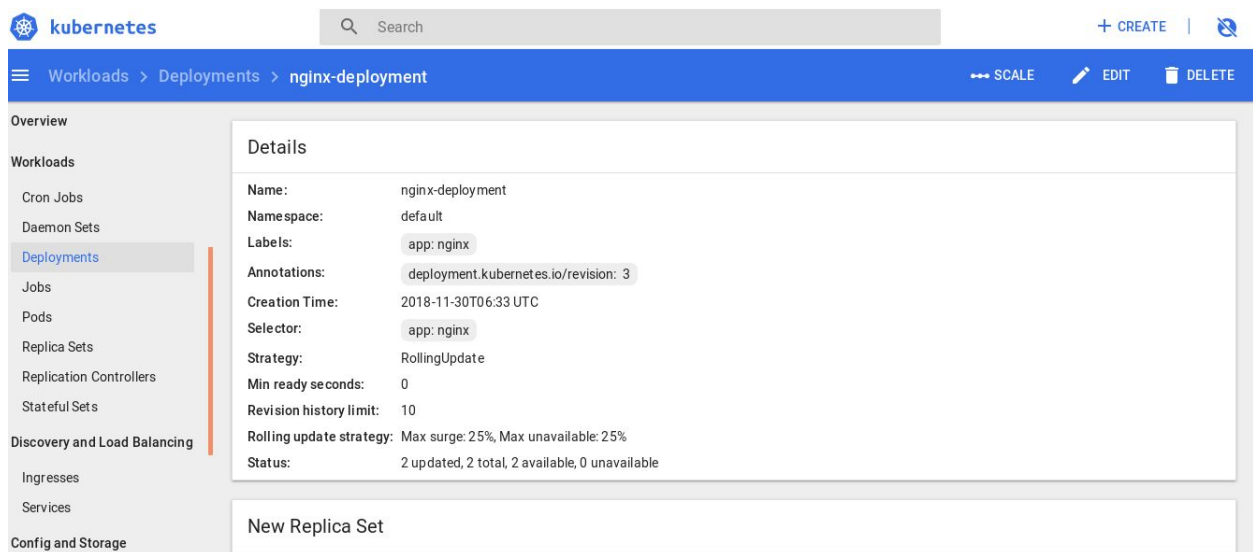
Goto the Deployments



The screenshot shows the Kubernetes Dashboard interface. The top navigation bar includes the 'kubernetes' logo, a search bar, and a '+ CREATE' button. The left sidebar contains a menu with categories: Overview, Workloads, Discovery and Load Balancing, and Config and Storage. Under 'Workloads', 'Deployments' is selected. The main content area displays a table of Deployments. The table has columns for Name, Labels, Pods, Age, and Images. One deployment is listed: 'nginx-deployment' with labels 'app: nginx', 2 / 2 pods, an age of 38 minutes, and image 'nginx:1.15'.

Name	Labels	Pods	Age	Images
nginx-deployment	app: nginx	2 / 2	38 minutes	nginx:1.15


Click on the Deployment that needs to be updates.



The screenshot shows the details page for the 'nginx-deployment'. The breadcrumb navigation is 'Workloads > Deployments > nginx-deployment'. The top right of the main area has buttons for 'SCALE', 'EDIT', and 'DELETE'. The left sidebar is the same as the previous screenshot. The main content area is titled 'Details' and lists various deployment attributes. Below the details is a section for 'New Replica Set'.

Attribute	Value
Name	nginx-deployment
Namespace	default
Labels	app: nginx
Annotations	deployment.kubernetes.io/revision: 3
Creation Time	2018-11-30T06:33 UTC
Selector	app: nginx
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

Click on the EDIT button



The screenshot shows the AWS Management Console interface for editing a Deployment. The search bar at the top contains the text: `apiVersion: apps/v1beta2kind: Deploymentmetadata:name: nginx-deploymentspec:selector:matchLabels:app: nginxreplic`. The left sidebar shows the navigation menu with 'Details' selected. The main content area displays the Deployment configuration in a code editor. The configuration is as follows:

```
26  "template": {
27    "metadata": {
28      "creationTimestamp": null,
29      "labels": {
30        "app": "nginx"
31      }
32    },
33    "spec": {
34      "containers": [
35        {
36          "name": "nginx",
37          "image": "nginx:1.15",
38          "ports": [
39            {
40              "containerPort": 80,
41              "protocol": "TCP"
42            }
43          ],
44          "resources": {},
45          "terminationMessagePath": "/dev/termination-log",
46          "terminationMessagePolicy": "File",
47          "imagePullPolicy": "IfNotPresent"
48        }
49      ],
50      "restartPolicy": "Always",
51      "terminationGracePeriodSeconds": 30,
52      "dnsPolicy": "ClusterFirst",
53      "securityContext": {},
54      "schedulerName": "default-scheduler"
55    }
56  }
```

At the bottom of the console, there are three buttons: **CANCEL**, **COPY**, and **UPDATE**.

And update the "image": nginx:(version to be updated to)
And then click on update.

This will update the nginx image on the fly.

Wait for few minutes and the application will be updated to the latest version.

Rollbacks

In the same way if the application needs to be downgraded to a previous docker version, this can be achieved in the same manner.

Dashboard>Deployments>EDIT>(image version)>Update