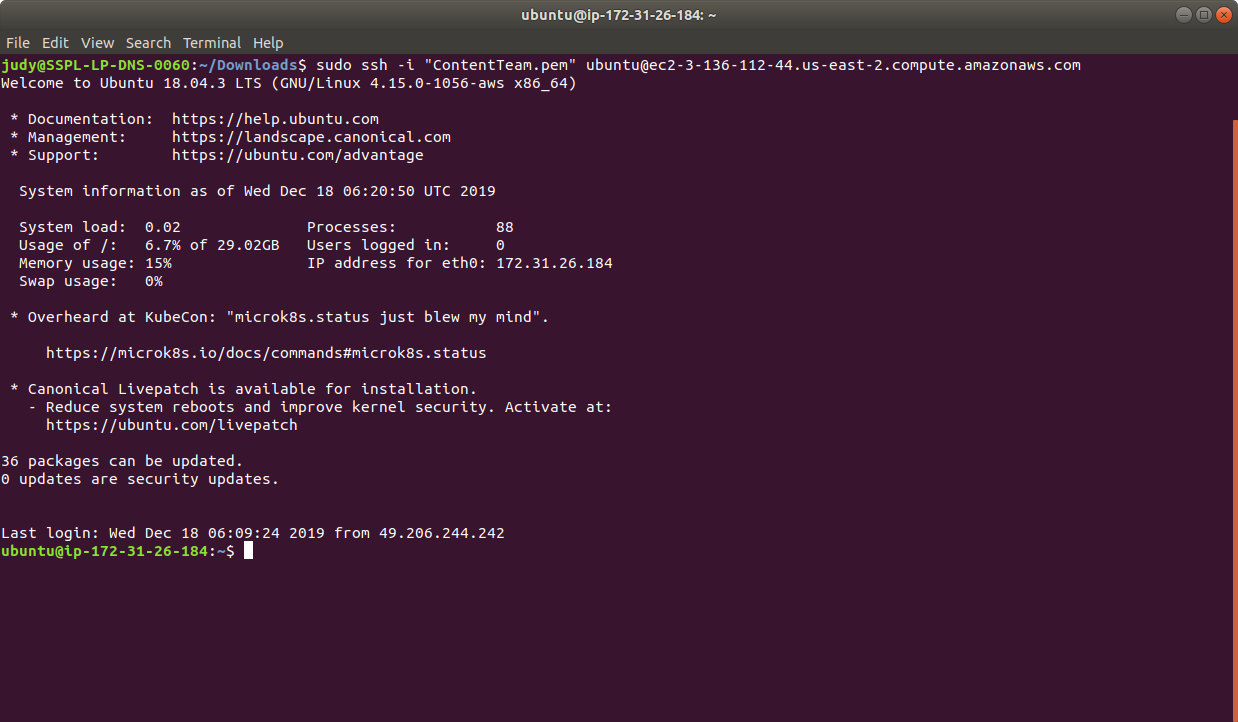
4.1 Deployment Rolling Update by Editing the Deployment File

In this demo, we will show you how to update deployment rolling by editing the deployment file.

* Login to your aws console
* Restart your ec2 instance and your EKS cluster nodes
* Open your terminal and SSH to the ec2 instance



* If you don’t have an existing EKS cluster, create one with the command, **eksctl create cluster --name=myeks-cluster --nodes=2 --region=us-east-2**
* A rolling update applies changes to the configuration of the pods being managed by a replication controller. The changes can be passed as a new replication controller configuration file; or, if only updating the image, a new container image can be specified directly. Let’s take an example of applying update to deployment rolling using editing configuration file.
* Let’s say you are running version 1.7.9 of nginx:

**cat > replication-nginx-1.7.9.yaml**

apiVersion: v1

kind: ReplicationController

metadata:

name: my-nginx

spec:

replicas: 5

template:

metadata:

labels:

app: nginx

spec:

containers:

- name: nginx

image: nginx:1.7.9

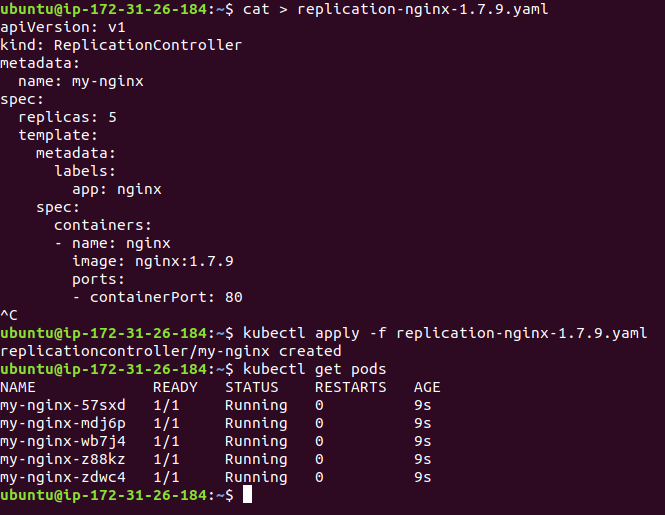
ports:

- containerPort: 80

* Next, submit the ReplicationController to the Kubernetes cluster using kubectl command as shown below:

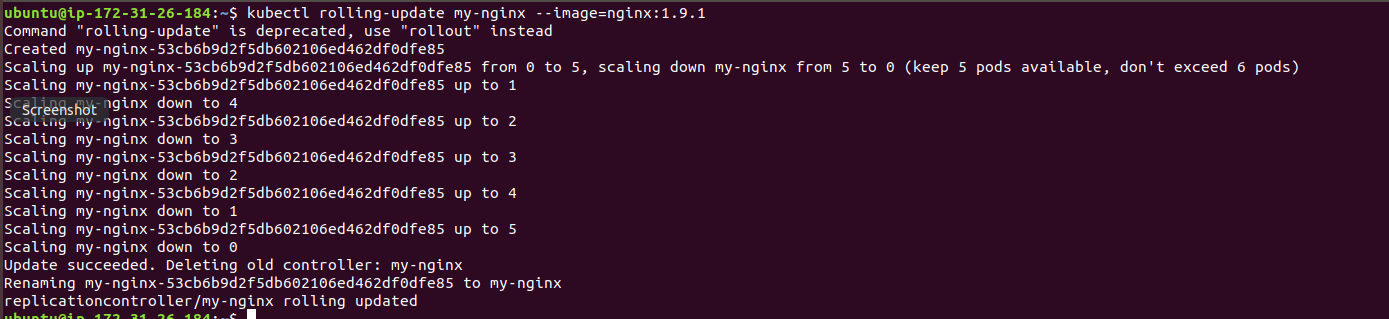
**kubectl apply -f replication-nginx-1.7.9.yaml**

* Run **kubectl get pods** to verify nginx is running



* To update to version 1.9.1, you can use kubectl rolling-update –image to specify the new image as shown below:

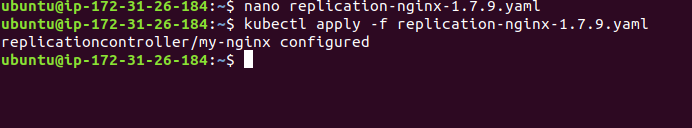
**kubectl rolling-update my-nginx --image=nginx:1.9.1**

****

* The same thing can be achieved by editing the yaml file as shown below and updating the version to 1.9.1. Next, submit the file in the cluster.



* Once edited, you can submit the updated yaml to the Kubernetes cluster using **kubectl apply -f replication-nginx-1.7.9** command as shown below:



* You can see the rollout status as shown below after applying below changes:

**kubectl describe replicationcontrollers/my-nginx**

