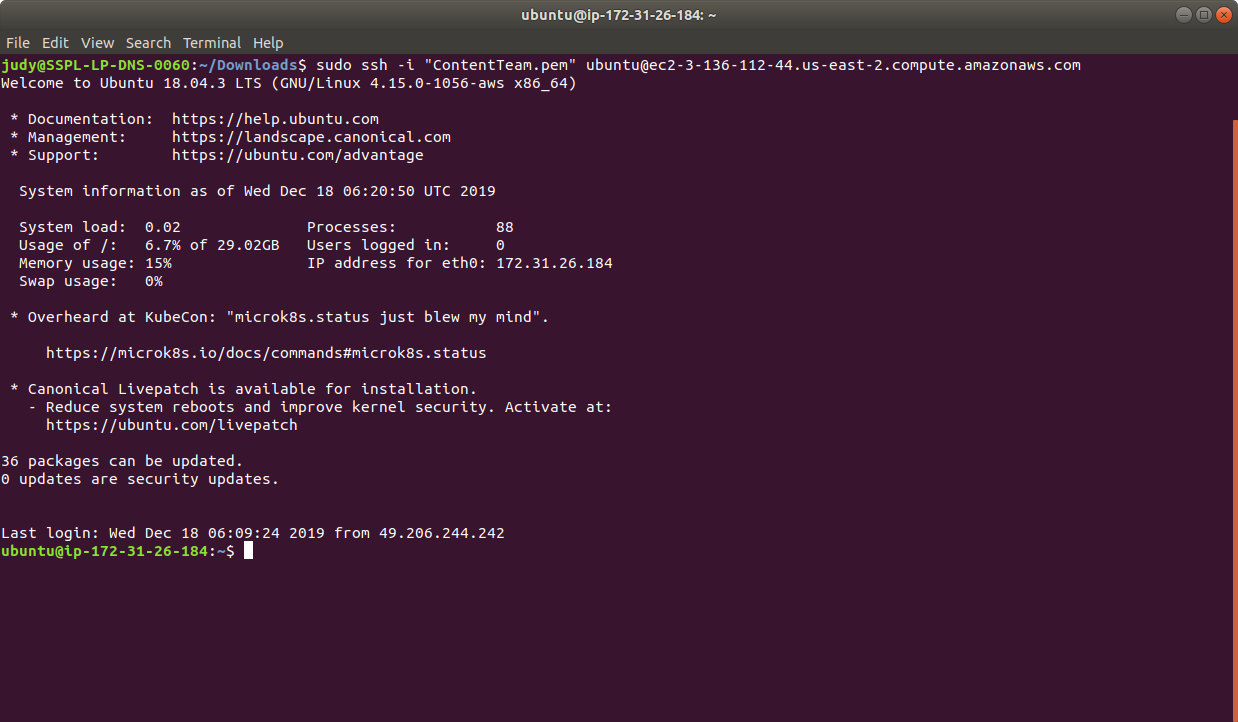
4.2 Deployment Rolling Update with kubectl Rollout (Command)

In this demo, we will show you how to update deployment rolling with kubectl rollout (command).

* 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**
* As the first step, let’s create a deployment named nginx-deployment, indicated by. metadata.namefield. The yaml file will look like the one shown below:

**cat > nginx-deployment.yaml**

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:

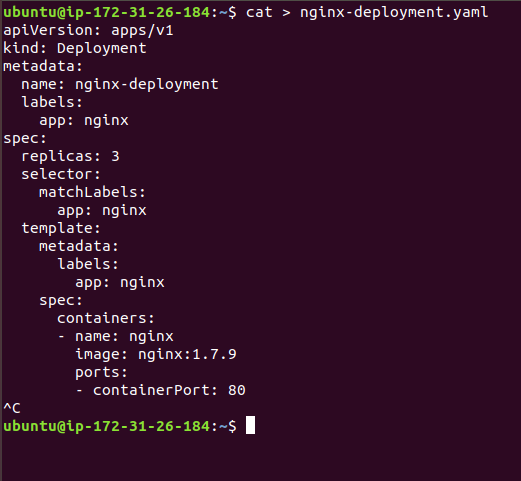
containers:

- name: nginx

image: nginx:1.7.9

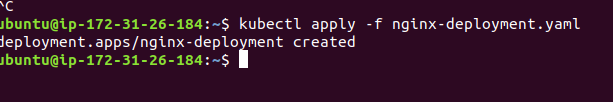
ports:

- containerPort: 80

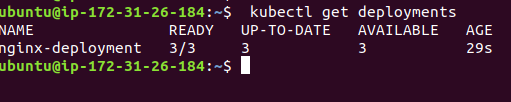


* Next, create the deployment by running the following kubectl command:

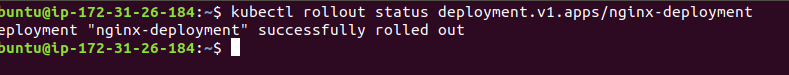
**kubectl apply -f nginx-deployment.yaml**

****

* Run the ***kubectl get deployments*** command to check if the deployment was created. If the deployment is still being created, the output is like the one shown below:

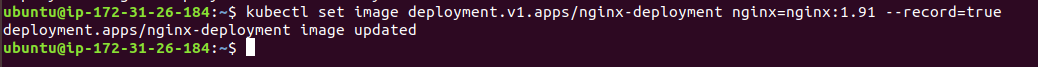


* To see the deployment rollout status, run **kubectl rollout status deployment.v1.apps/nginx-deployment**. The output is as shown below:



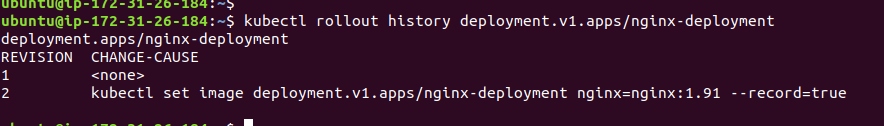
* Sometimes, you may want to rollback a deployment. For example, when the deployment is not stable, such as crash looping. Suppose that you made a typo while updating the deployment, by putting the image name as nginx:1.91 instead of nginx:1.9.1 the output will be shown similar to that shown below:

**kubectl set image deployment.v1.apps/nginx-deployment nginx=nginx:1.91 --record=true**



* You can now check the rollout history as shown below:

**kubectl rollout history** **deployment.v1.apps/nginx-deployment**



* This is how we can update the deployment rollout using kubectl commands.