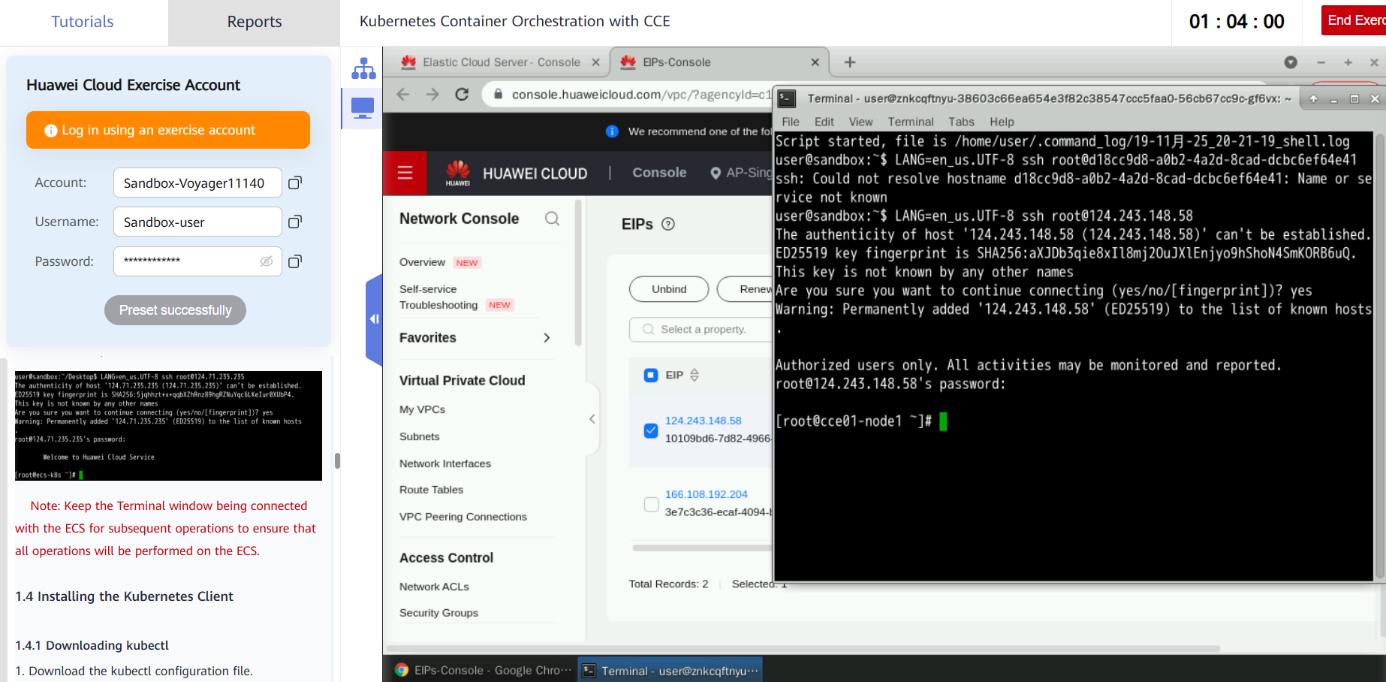
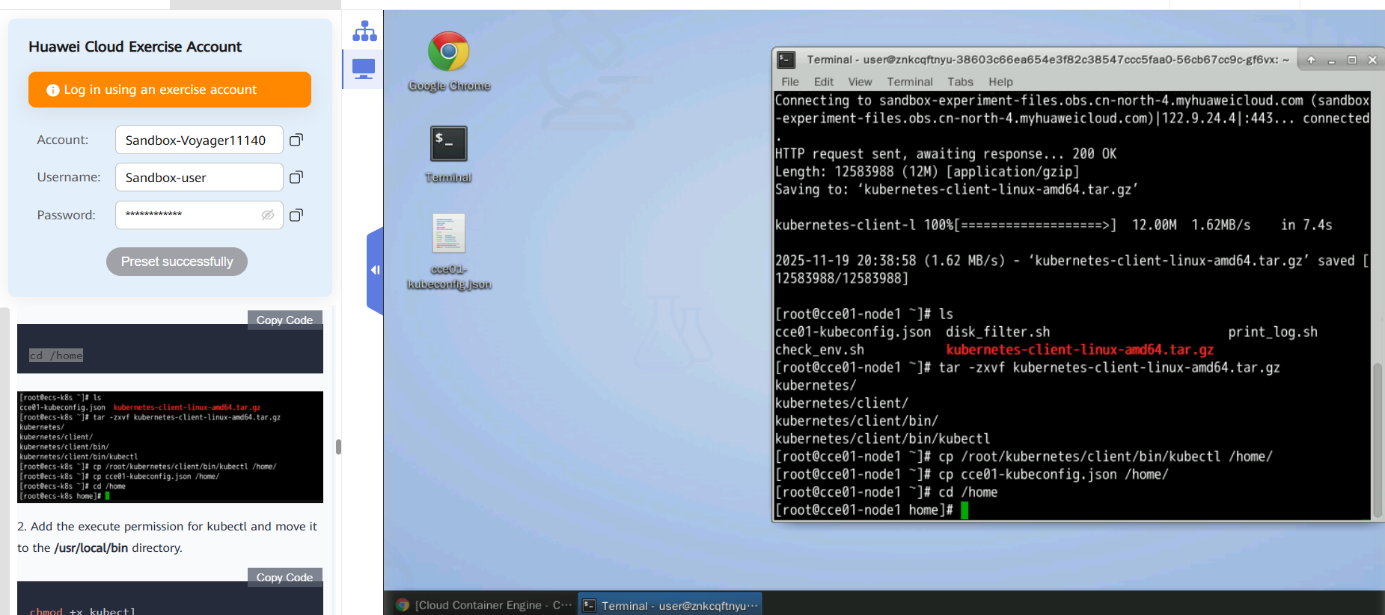
## Huawie Kubernetes denetleyici Deployment ve DaemonSet

Koolab’de verilen Tutorials talimatlarına uyarak öncelikle koolab ortamına giriş yapıyoruz. Giriş yapıldıktan sonra tutorials ayarlamalarına göre koolab ortamındaki siteye giriş yaparak cce01 adında CCE (Cloud Container Engine) oluşturuyoruz. Burd oluşturduğumuz cce01 running olduğu zaman cce01-node isimli node oluşturuyoruz. Node conteinerları çalıştıran fiziksel sunucudur.Daha sons tutorials doğrultusunda ecs-k8s isimli ECS oluşturuyoruz .Bu ecs-k8s ‘in EIP’sine ulaşarak EIP değerine ssh ile uzaktan bağlantı yapıyoruz.



Daha sonra CCE cluster ‘ına girip kubectl configure diyerek kubectl download yapıyoruz. Masaüstüne alıyoruz dosyayı.Daha sonra kubectl configure dosyasını ecs-k8s ‘e yüklüyoruz.EIP yerine kendi EIP’mizi giriyoruz.Bu kod ile: scp /home/user/Desktop/cce01-kubeconfig.json root@EIP:/root



Kubectl için yürütme iznini ekleyip ve /usr/local/bin dizinine taşıyoruz.

chmod +x kubectl

mv kubectl /usr/local/bin

Yapılandırma dosyasını $HOME/.kube/config dosyasına kopyalayıp kubectl erişim modunu internal olarak değiştiriyoruz (bu, VPC içindeki erişim anlamına gelir).

mkdir $HOME/.kube

cp cce01-kubeconfig.json $HOME/.kube/config

kubectl config use-context internal

A screenshot of a computer

AI-generated content may be incorrect.

Yukarıda kubectl ile egzersiz yapılmış ve kubectl nodu gösterilmiştir.Deployment yapılandırması yaparken kubectl create deployment mydep --image=nginx kodu ile mydep adlı deployment oluşturuyoruz.

A screen shot of a computer

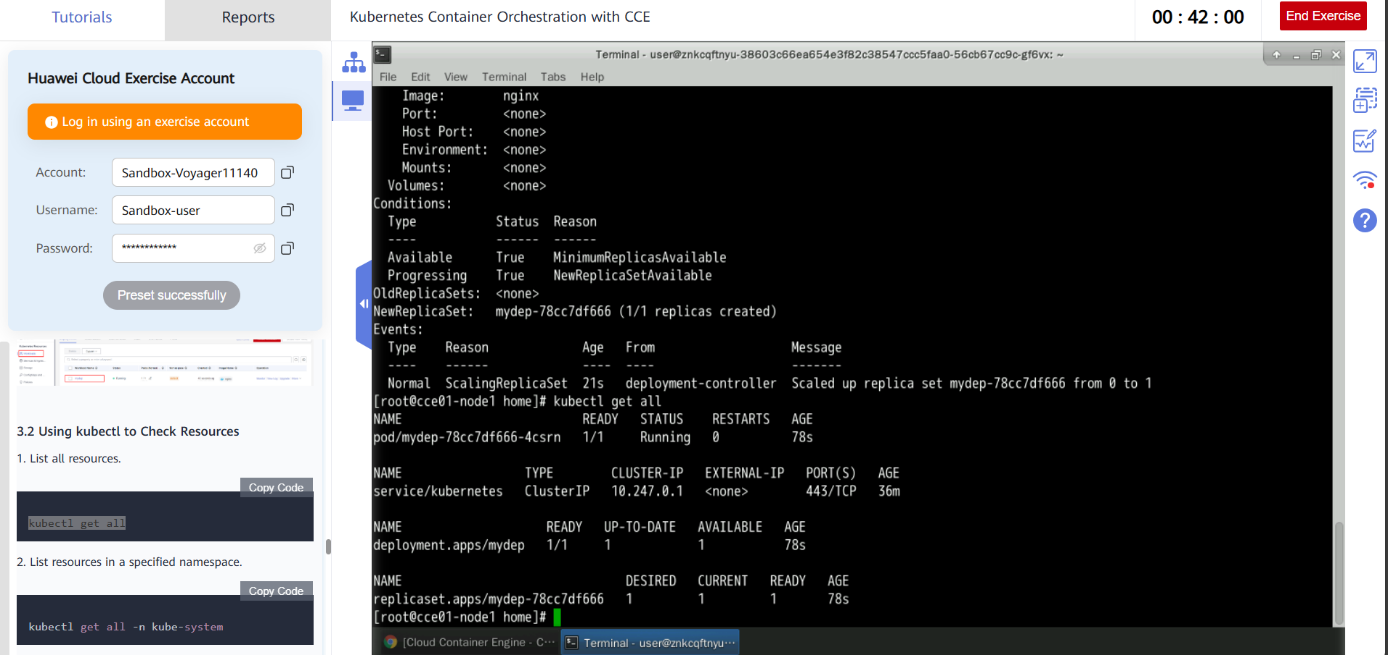
AI-generated content may be incorrect.

kubectl describe deployment mydep kodu ile mydep hakkında detaylı bilgilere erişiyoruz.

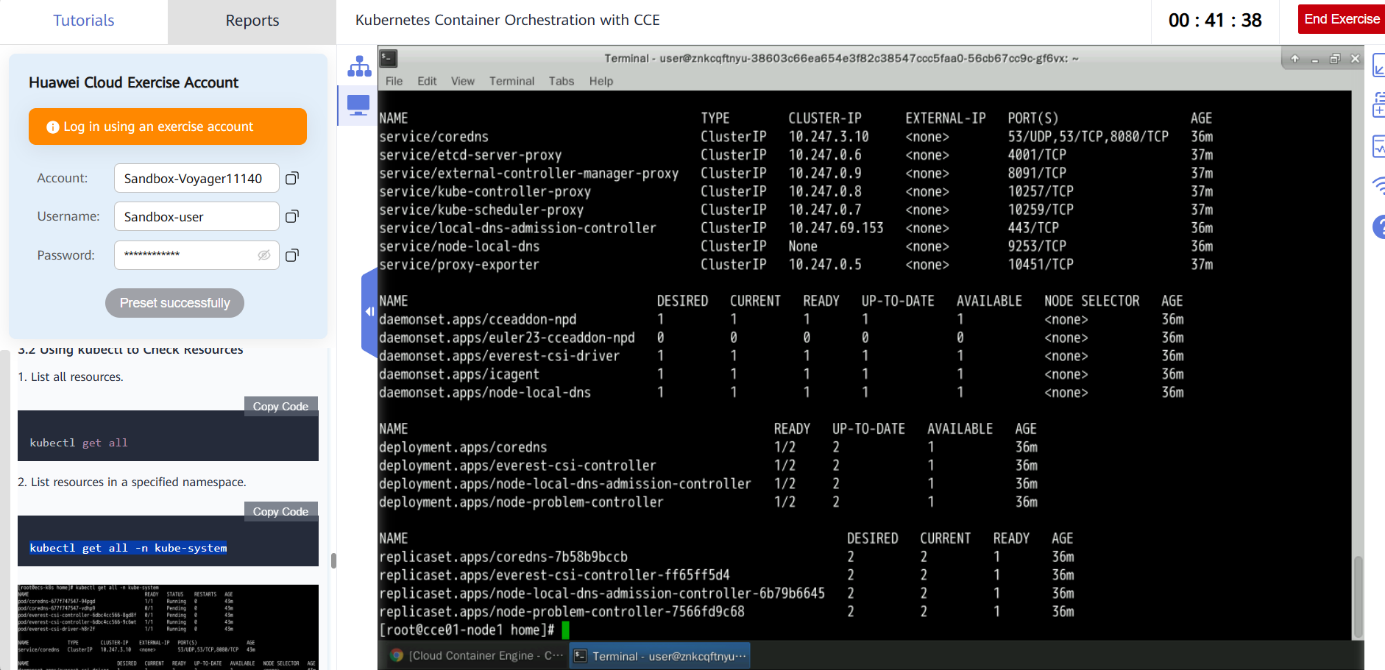
A computer screen shot of a black screen

AI-generated content may be incorrect.

Kubectl get all kodu ile tüm kaynakları getirir.

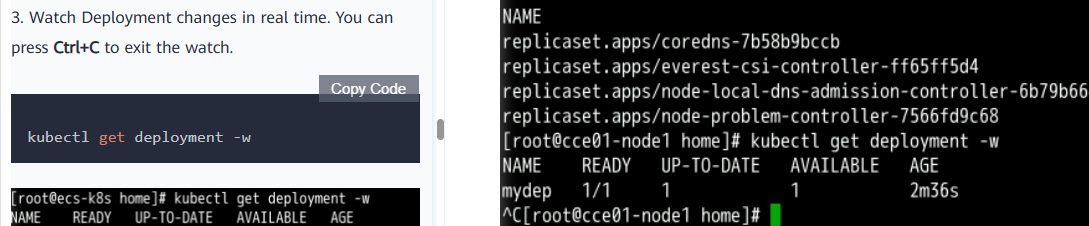
****

**Kubectl get all -n kube-system kodu ile aynı namespace’deki kaynakları listeliyoruz.**

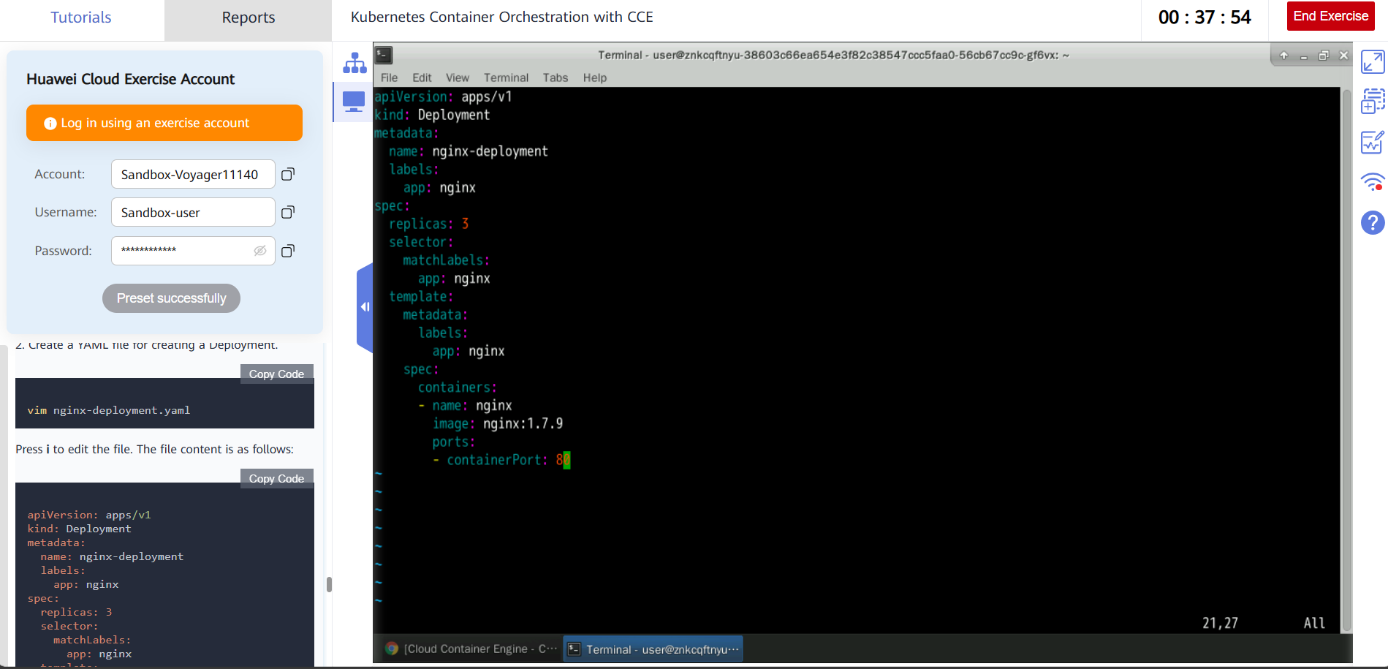
****

**Dağıtım değişikliklerini gerçek zamanlı olarak izleyip Ctrl+C ile çıkış yapıyoruz.**

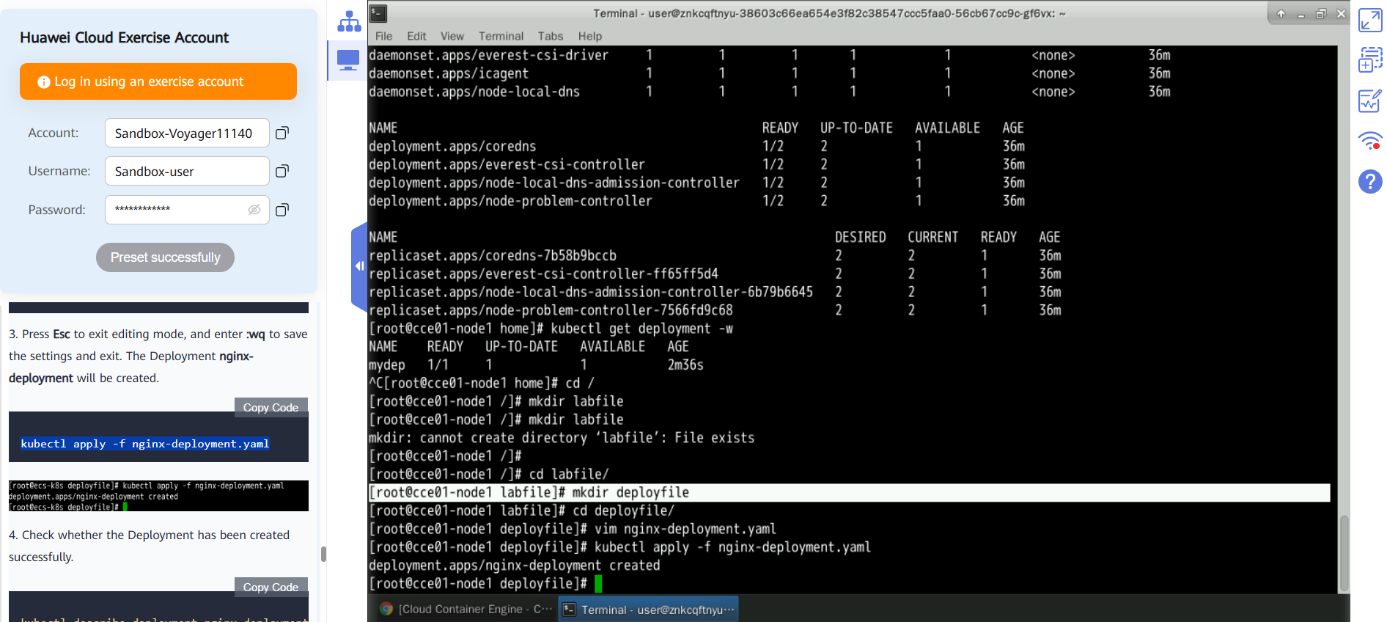
**kubectl get deployment -w**

****

**Deployment yapmak için yaml dosyası oluşturuyoruz. vim nginx-deployment.yaml**

****

**kubectl apply -f nginx-deployment.yaml komutu ile deployment.yaml dosyası oluşturulur.**

****

**Oluşturduktan sonra kubectl get deployment kodu ile listeliyoruz. Daha sonra kubectl get pod ile podları listeliyoruz. Verilen talimatlar doğrultusunda ngnix-deployment’ı siliyoruz. Daemonset alıştırması: DaemonSet, yönettiği pod'ların tüm küme düğümlerinde çalışmasını sağlayan bir denetleyicidir. Daemonset ve iş dosyaları için klasör oluşturuyoruz. Yaml dosyası yazıp Daemonset oluşturuyoruz. Daha sonra aynı namespace’de olanları listeliyoruz.**

**A computer screen shot of a black screen

AI-generated content may be incorrect.**

**Daemonset.yaml dosyası**

**A screen shot of a computer

AI-generated content may be incorrect.**

**DaemonSet tarafından yönetilen pod'ları listeleyerek ve pod ayrıntılarını ekliyoruz.**

**kubectl get pod --namespace=kube-system -o wide kodu ile**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**