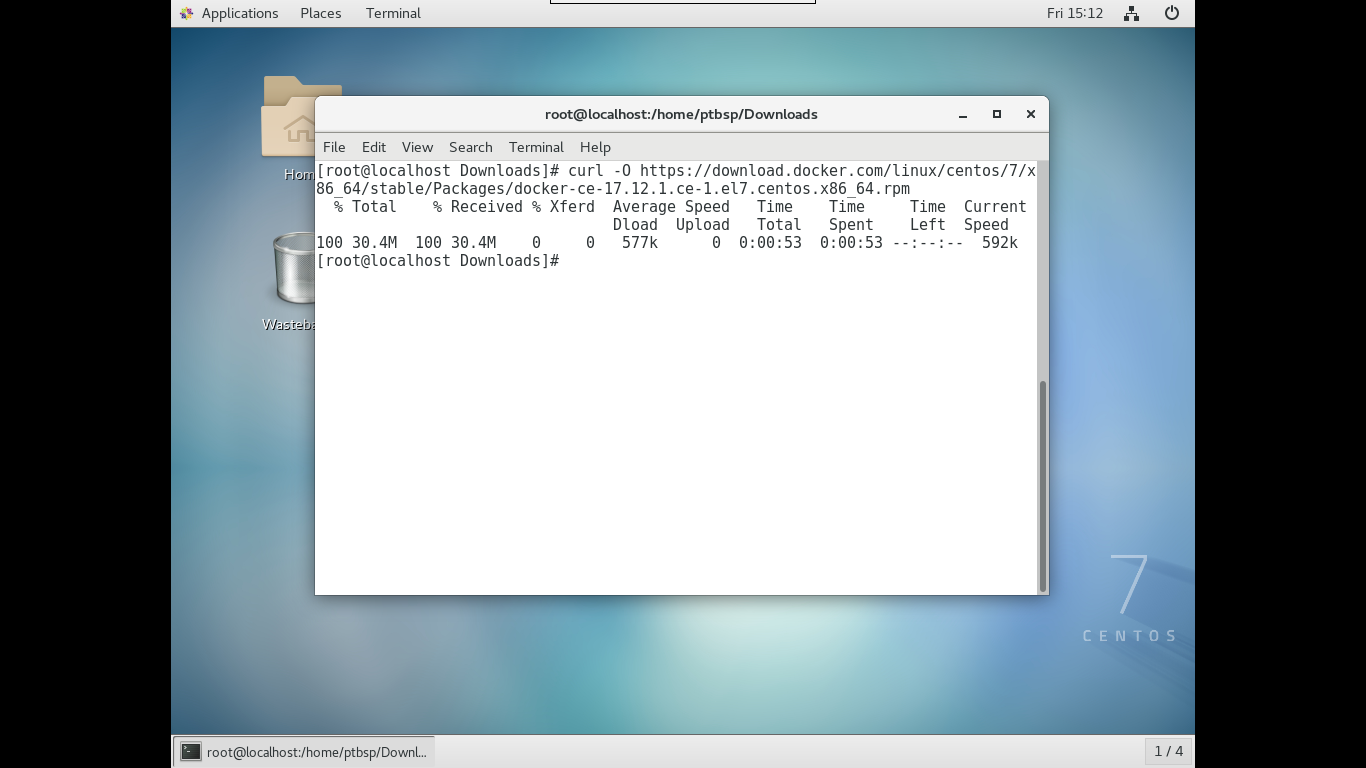
**Fission**

1. **Setup Docker**

Minimum requirement docker version 17.10, download docker :

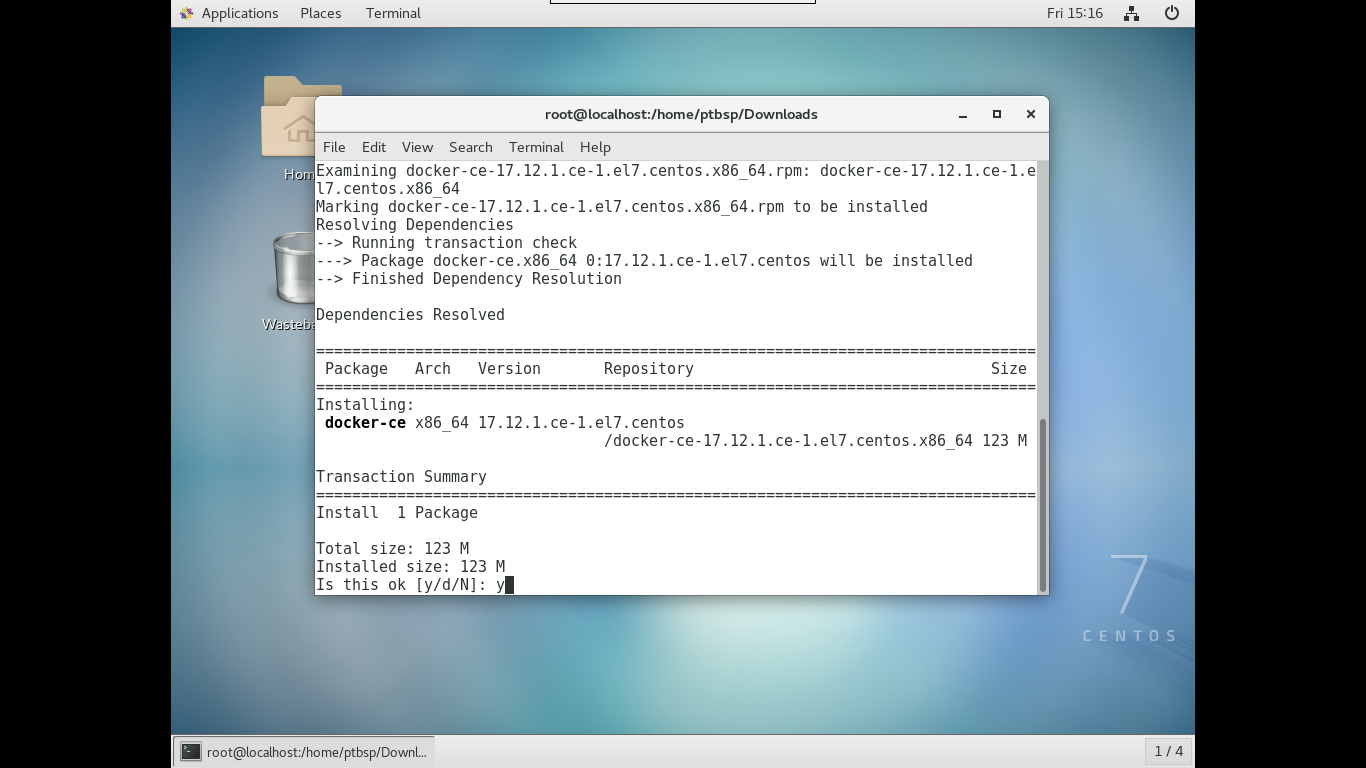
$ curl -O https://download.docker.com/linux/centos/7/x86\_64/stable/Packages/[docker-ce-17.12.1.ce-1.el7.centos.x86\_64.rpm](https://download.docker.com/linux/centos/7/x86_64/stable/Packages/docker-ce-17.12.1.ce-1.el7.centos.x86_64.rpm)

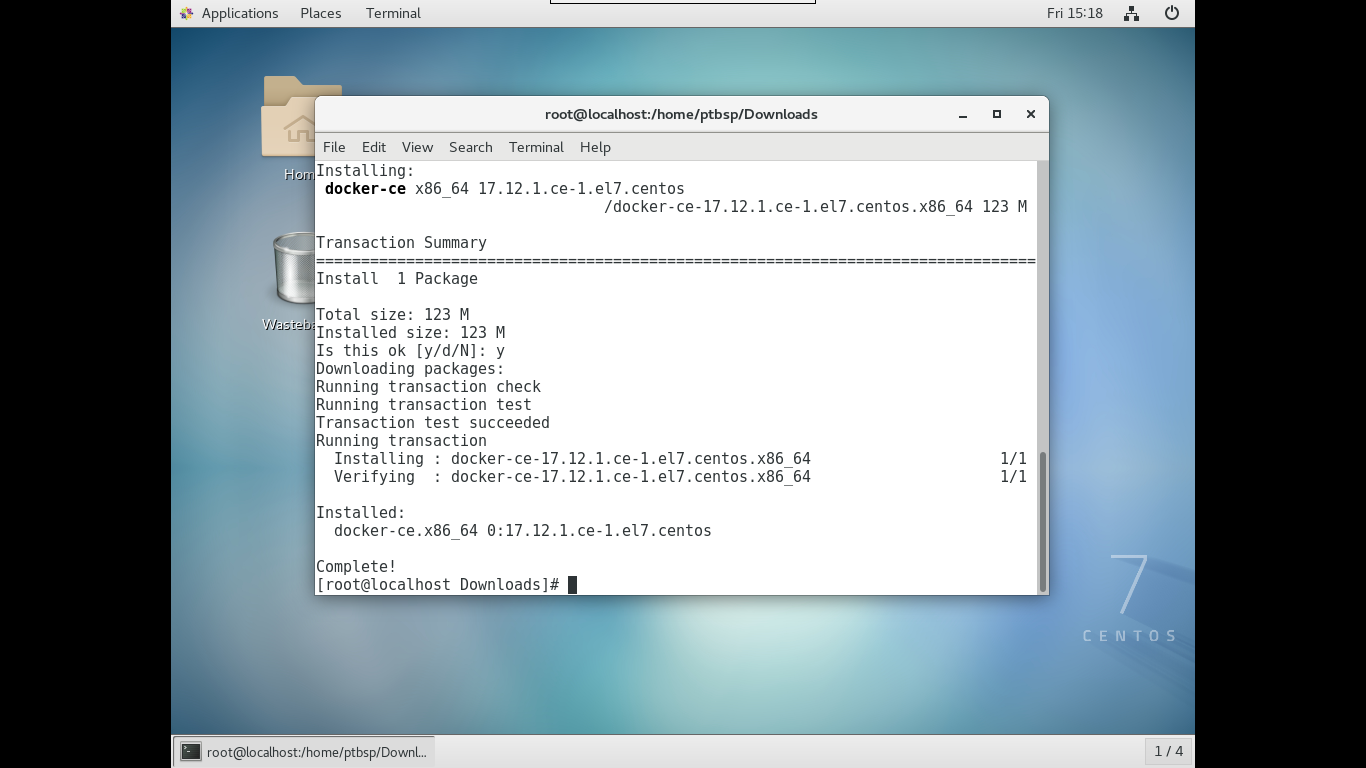


Install docker dengan menjalankan perintah :

$ chmod +x docker-ce-17.12.1.ce-1.el7.centos.x86\_64.rpm

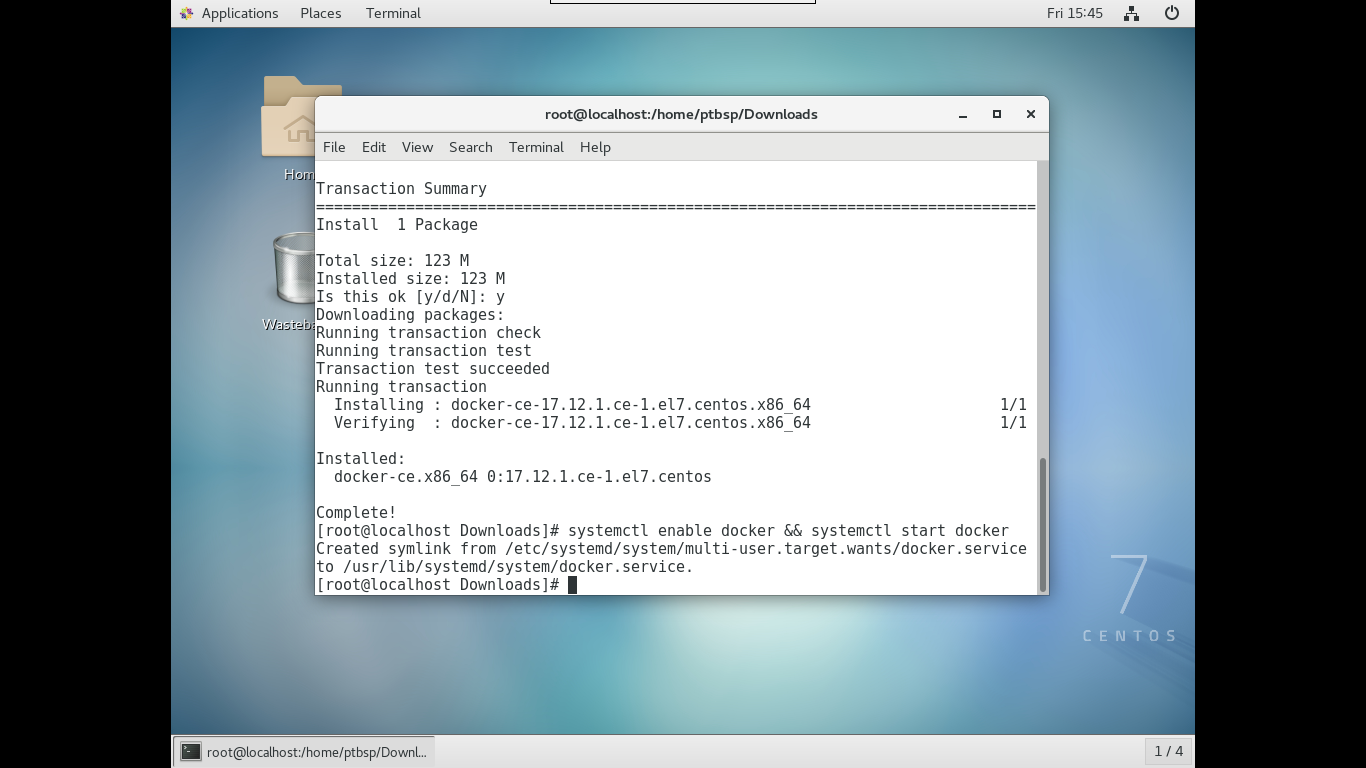
$ yum install docker-ce-17.12.1.ce-1.el7.centos.x86\_64.rpm





Enable dan Start docker

$ systemctl enable docker && systemctl start docker



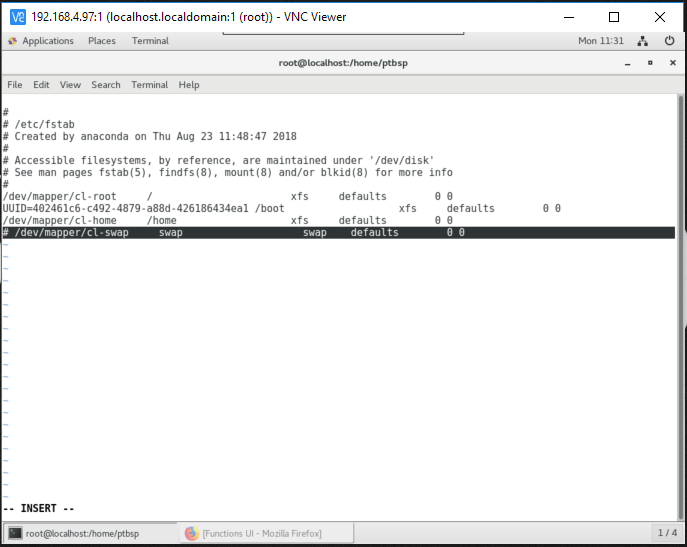
1. **Setup Kubernetes**

Disable konfigurasi swap memory menggunakan perintah

$ swapoff -a

Atau edit bagian swap pada file /etc/fstab menjadi komentar dengan menambahkan ‘#’

$ vi /etc/fstab



Membuka akses port-port kubernetes pada firewall :

$ firewall-cmd --permanent --add-port=6443/tcp

$ firewall-cmd --permanent --add-port=2379-2380/tcp

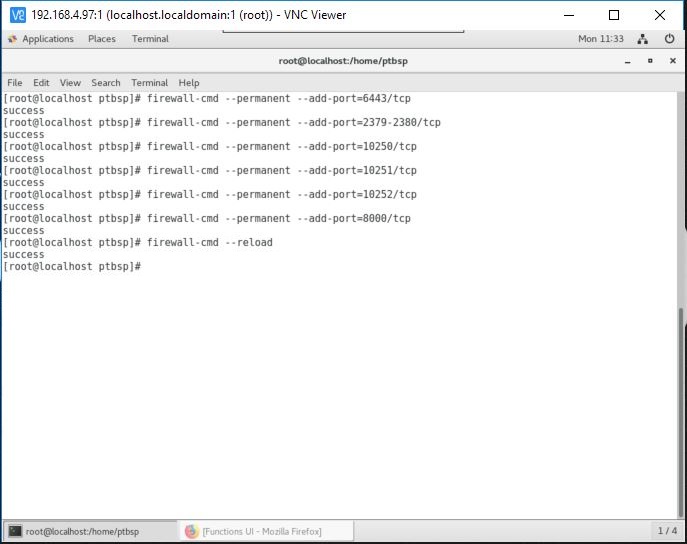
$ firewall-cmd --permanent --add-port=10250/tcp

$ firewall-cmd --permanent --add-port=10251/tcp

$ firewall-cmd --permanent --add-port=10252/tcp

$ firewall-cmd --permanent --add-port=8000/tcp

$ firewall-cmd --reload



Membuat file konfigurasi iptables

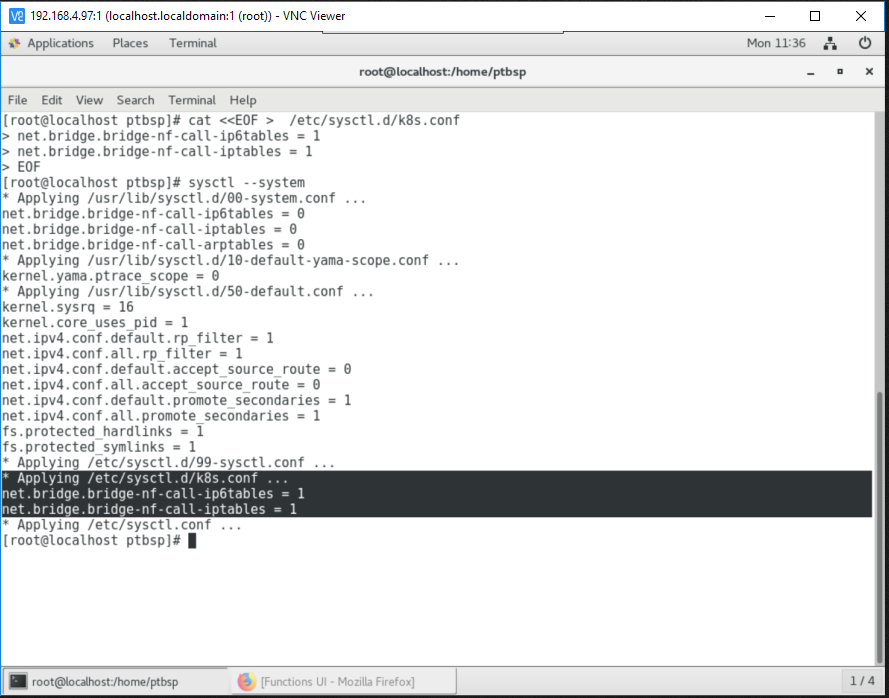
$ cat <<EOF > /etc/sysctl.d/k8s.conf

> net.bridge.bridge-nf-call-ip6tables = 1

> net.bridge.bridge-nf-call-iptables = 1

> EOF

$ sysctl –system



Membuat file konfiguraasi repository untuk installasi kubernetes

$ cat <<EOF > /etc/yum.repos.d/kubernetes.repo

> [kubernetes]

> name=Kubernetes

> baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-x86\_64

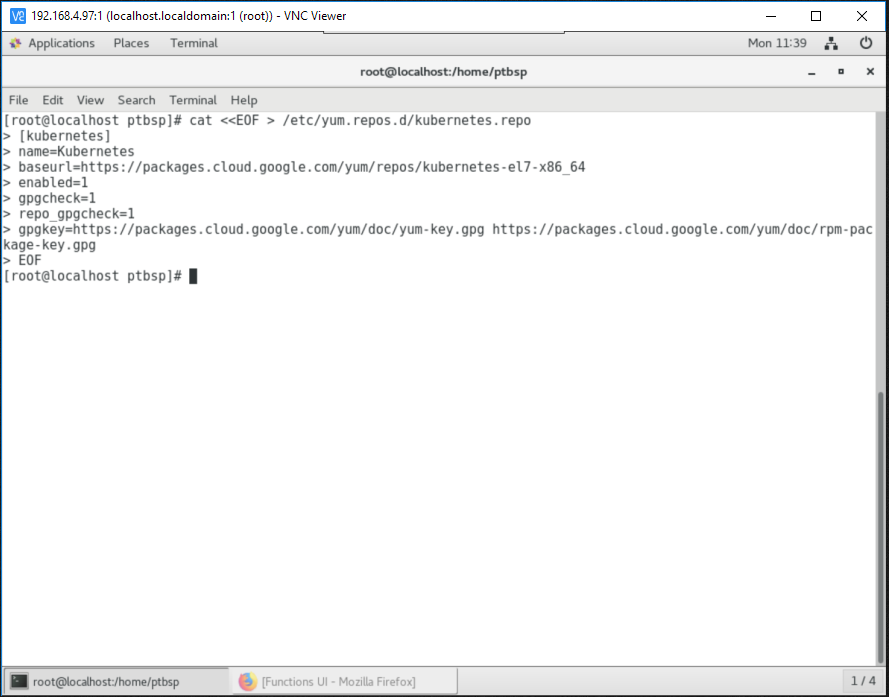
> enabled=1

> gpgcheck=1

> repo\_gpgcheck=1

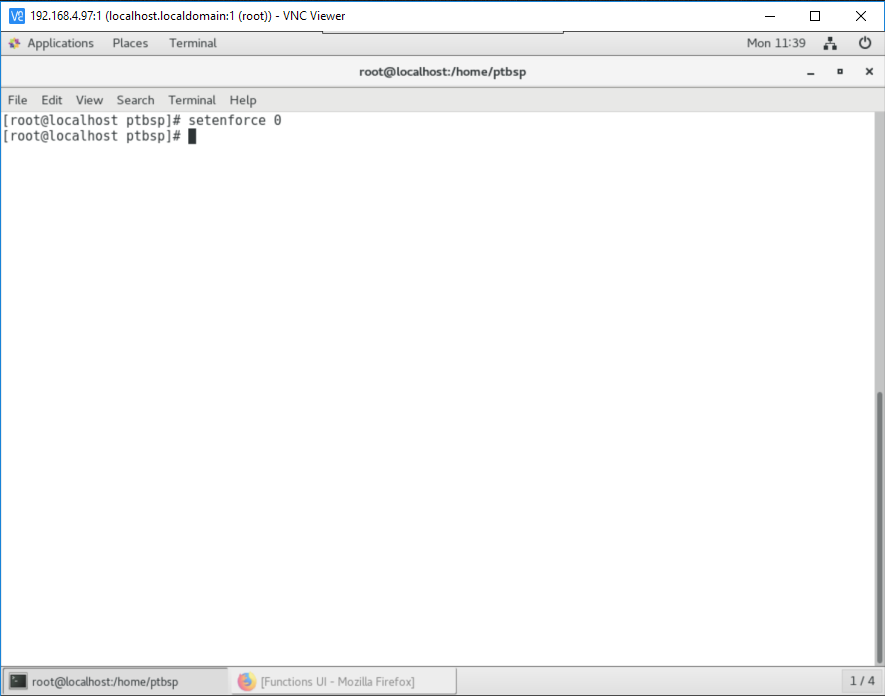
> gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg

> EOF



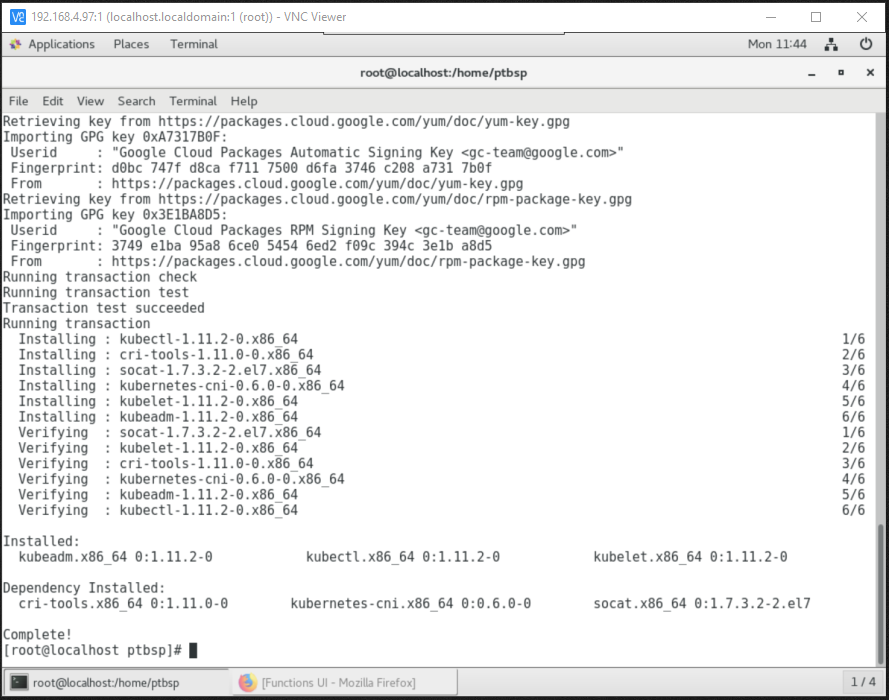
Disable setenforce dengan menjalankan perintah

$ setenforce 0



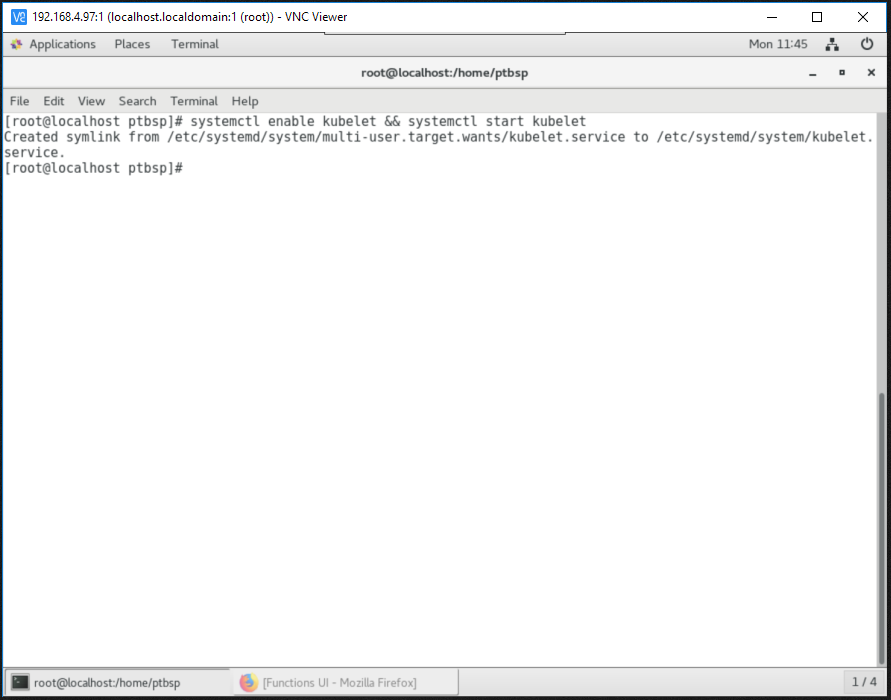
Install kubernetes package

$ yum install -y kubelet kubeadm kubectl



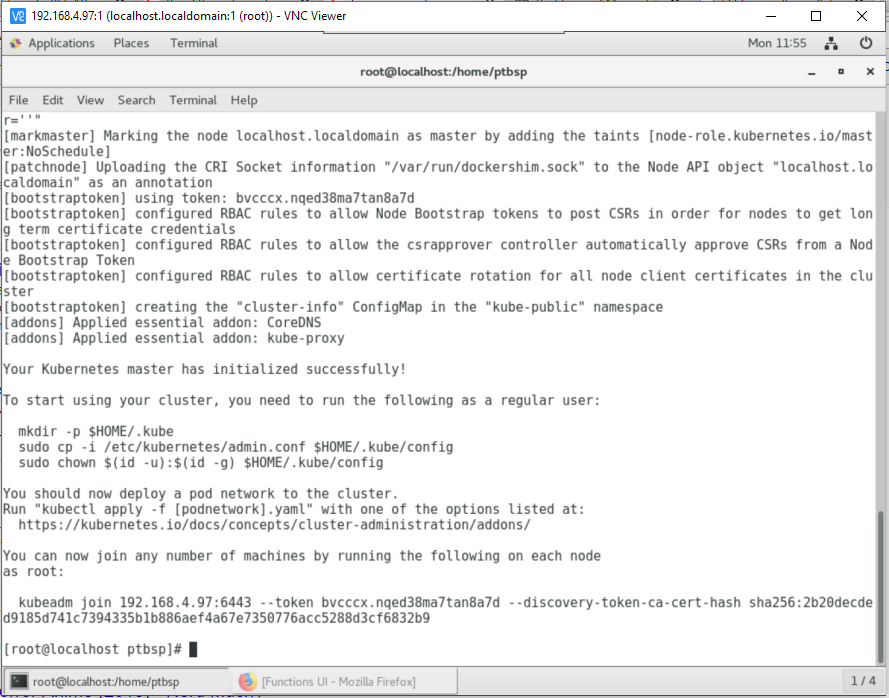
Start dan enable service kubelet

$ systemctl enable kubelet && systemctl start kubelet



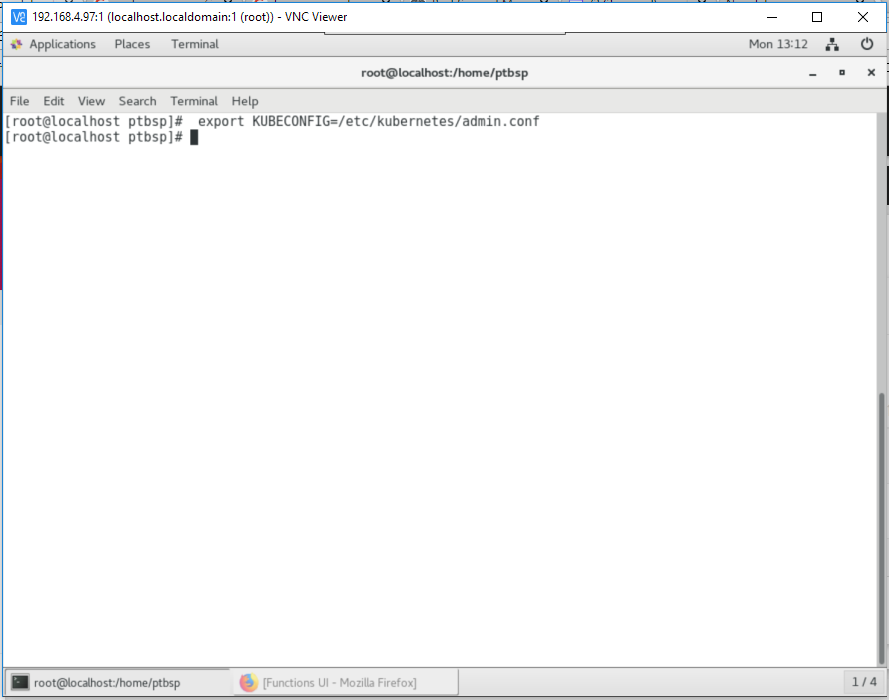
Inisialisasi kubernetes menggunakan cidr dan advertise ip

$ kubeadm init --pod-network-cidr=192.168.0.0/16 --apiserver-advertise-address=192.168.4.97



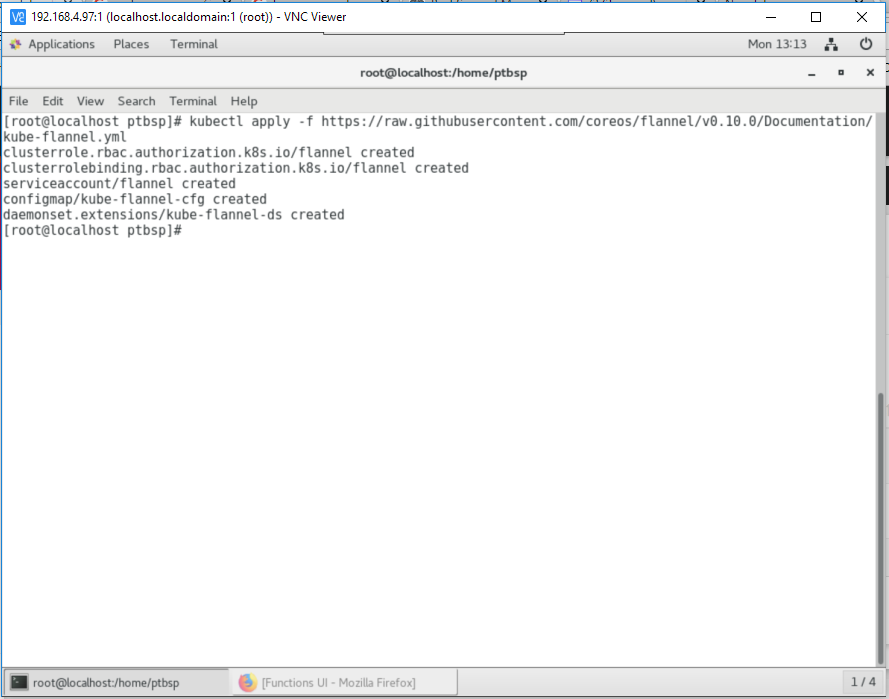
Buat system variable untuk settingan file konfigurasi kubernetes

$ export KUBECONFIG=/etc/kubernetes/admin.conf



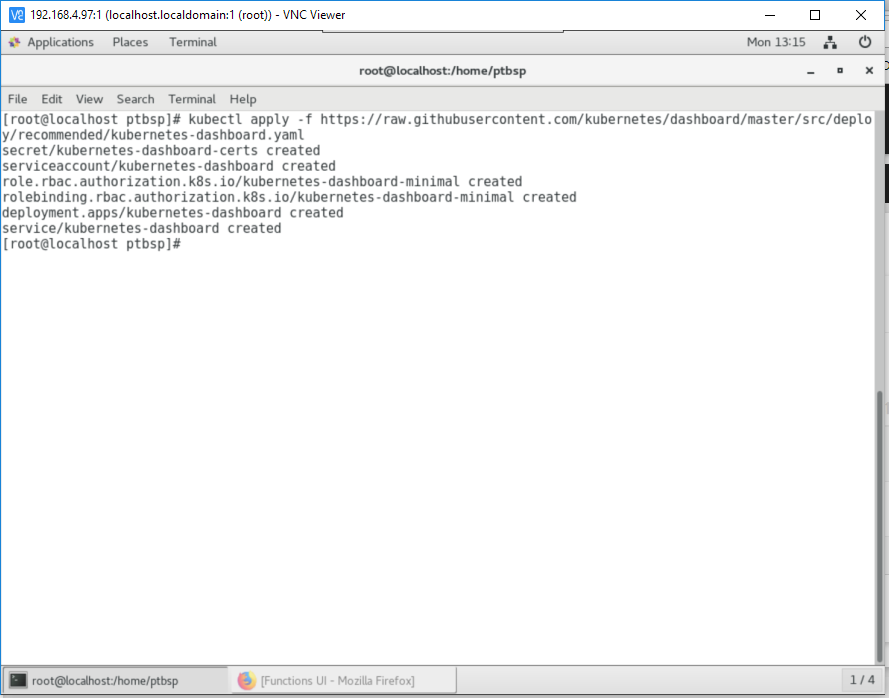
Install pod network add-on pada kubernetes (menggunakan flannel)

$ kubectl apply -f https://raw.githubusercontent.com/coreos/flannel/v0.10.0/Documentation/kube-flannel.yml



Install dashboard pada kubernetes

$ kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/master/src/deploy/recommended/kubernetes-dashboard.yaml



Membuat role admin pada kubernetes dashboard :

$ cat <<EOF > dashboard-admin.yaml

> apiVersion: rbac.authorization.k8s.io/v1beta1

> kind: ClusterRoleBinding

> metadata:

> name: kubernetes-dashboard

> labels:

> k8s-app: kubernetes-dashboard

> roleRef:

> apiGroup: rbac.authorization.k8s.io

> kind: ClusterRole

> name: cluster-admin

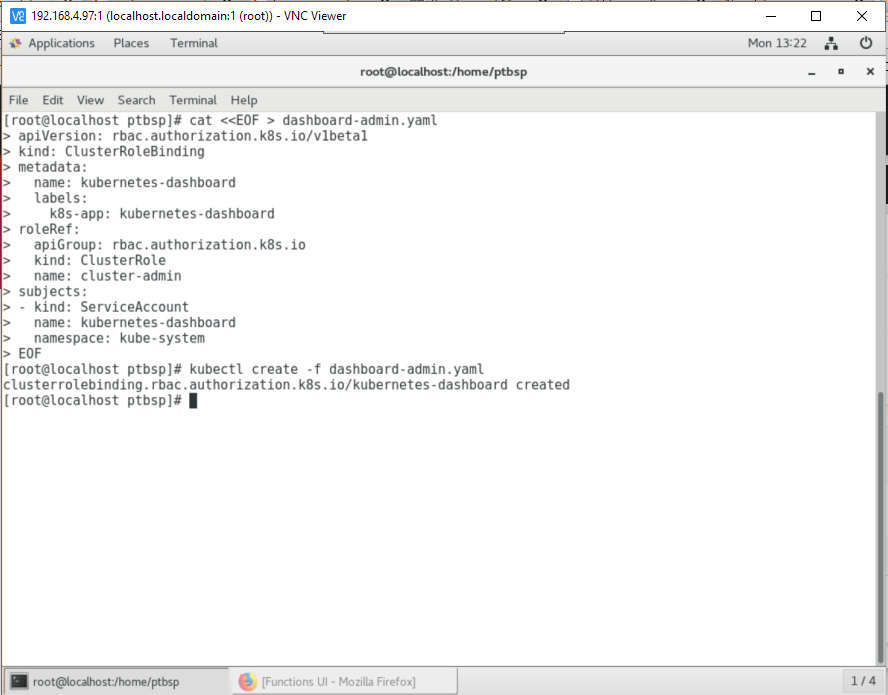
> subjects:

> - kind: ServiceAccount

> name: kubernetes-dashboard

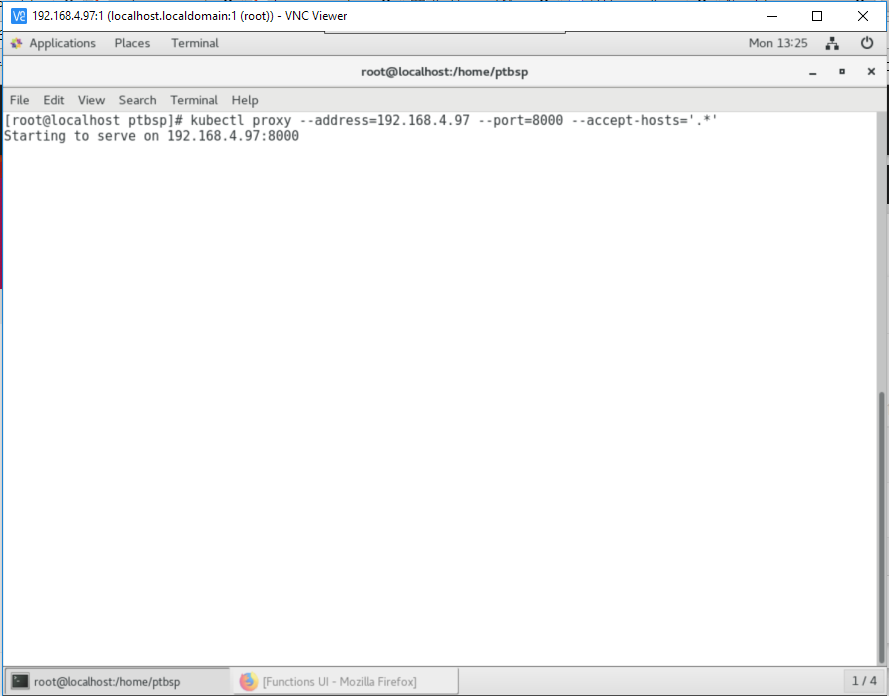
> namespace: kube-system

$ kubectl create -f dashboard-admin.yaml

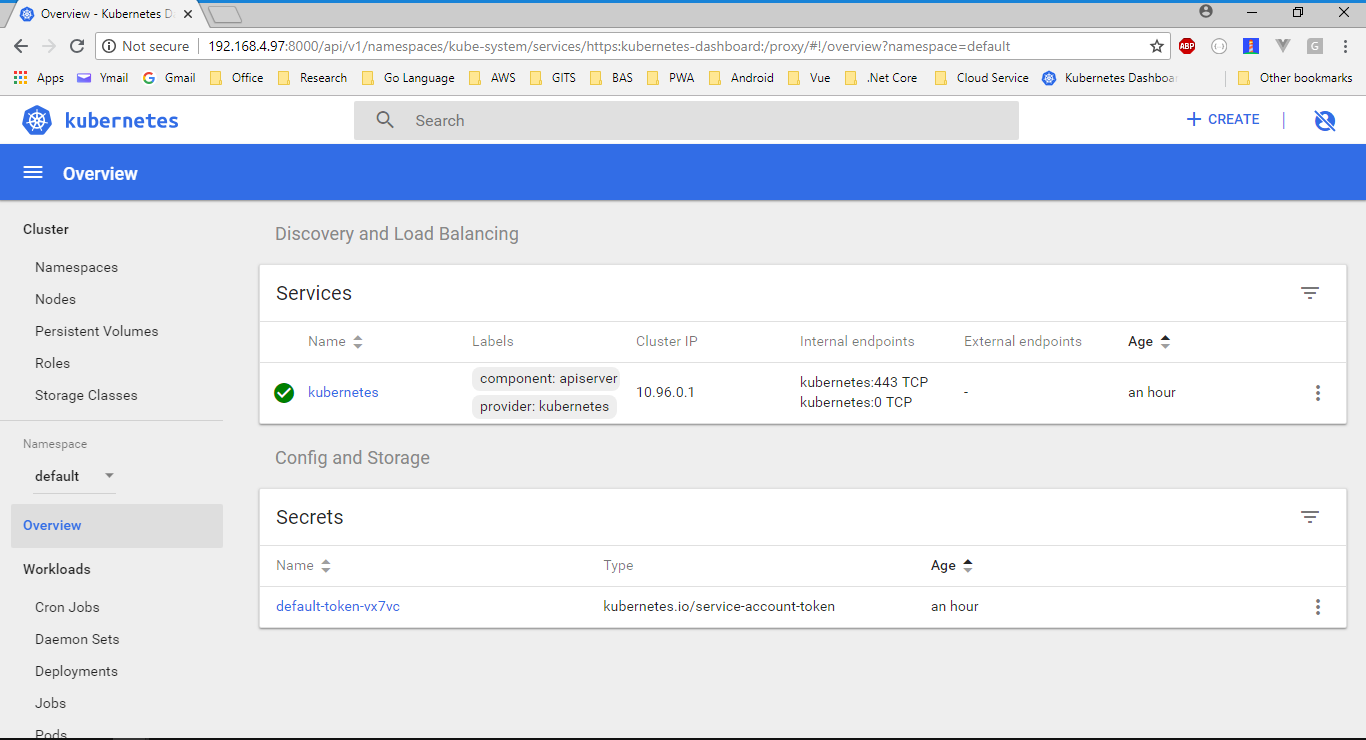


Membuka akses dashboard kubernetes

$ kubectl proxy --address=192.168.4.97 --port=8000 --accept-hosts='.\*'



http://192.168.4.97:8000/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/



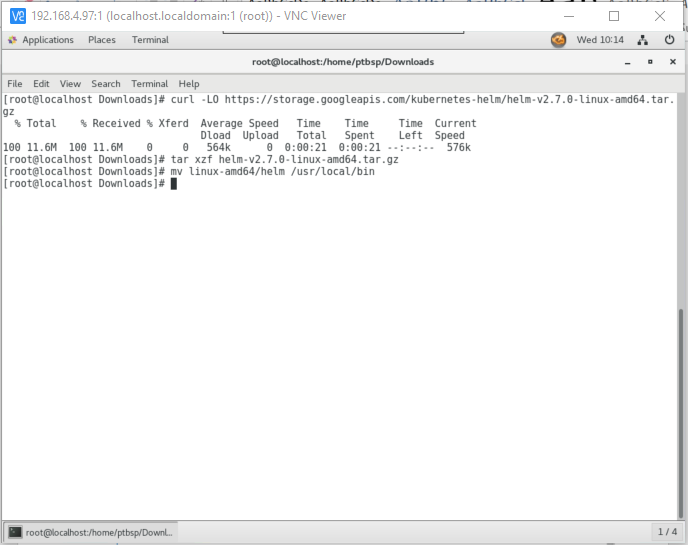
1. **Setup Helm**

Helm adalah installer untuk kubernetes, berikut perintah untuk men-download, meng-extract dan mendaftarkan helm kedalam binary directory :

$ curl -LO https://storage.googleapis.com/kubernetes-helm/helm-v2.7.0-linux-amd64.tar.gz

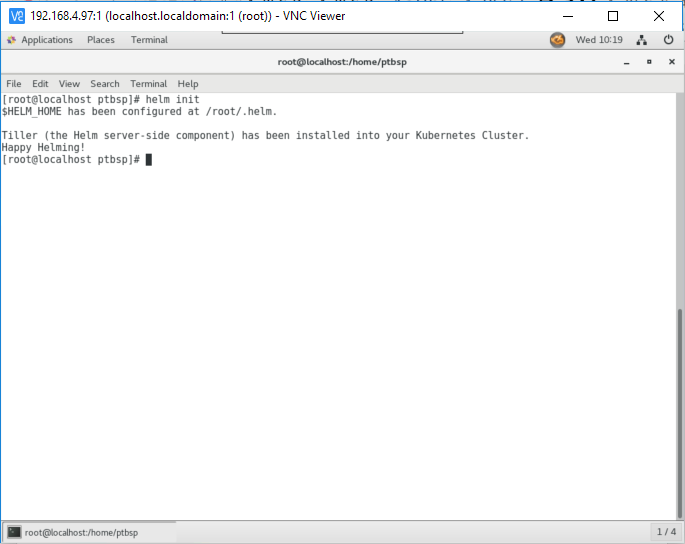
$ tar xzf helm-v2.7.0-linux-amd64.tar.gz

$ mv linux-amd64/helm /usr/local/bin



Inisiasi helm pada master kube kubernetes

$ helm init

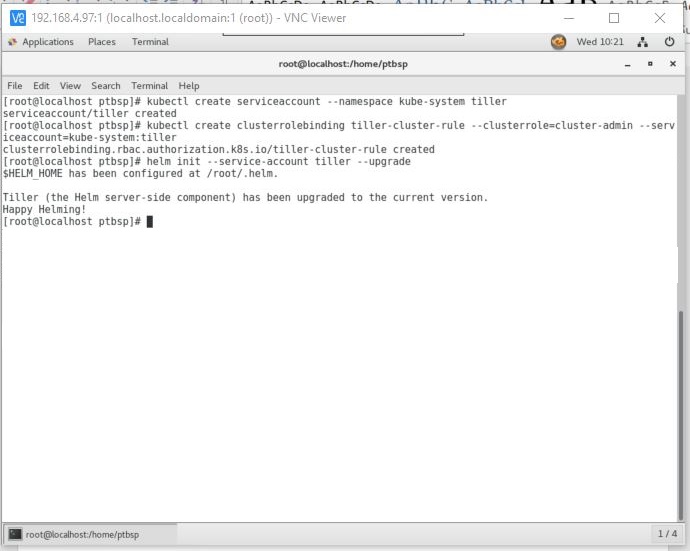


Konfigurasi RBAC untuk menggunakan helm pada kubernetes

$ kubectl create serviceaccount –-namespace kube-system tiller

$ kubectl create clusterrolebinding tiller-cluster-rule --clusterrole=cluster-admin --serviceaccount=kube-system:tiller

$ helm init –-service-account tiller --upgrade



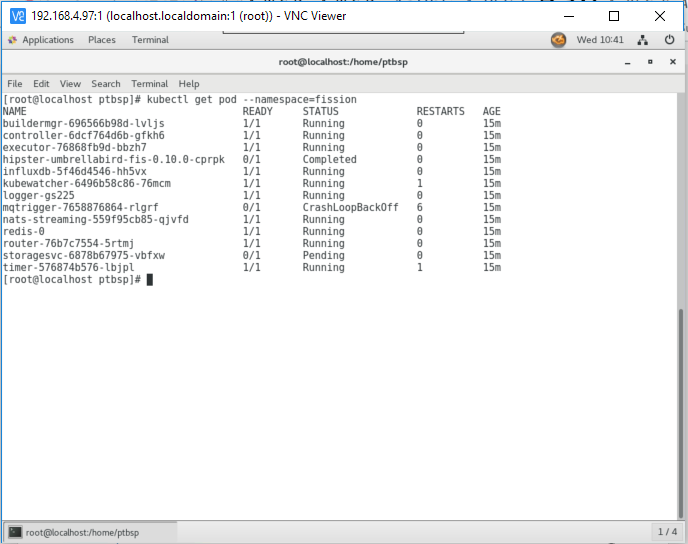
1. **Setup Fission**

Install fission menggunakan helm

$ helm install --namespace fission https://github.com/fission/fission/releases/download/0.10.0/fission-all-0.10.0.tgz

Check pod fission pada kubernetes menggunakan perintah

$ kubectl get pod –-namespace=fission



Namun masih ada pod yang bermasalah yaitu pod mqtrigger dengan status CrashingLoopBackOff

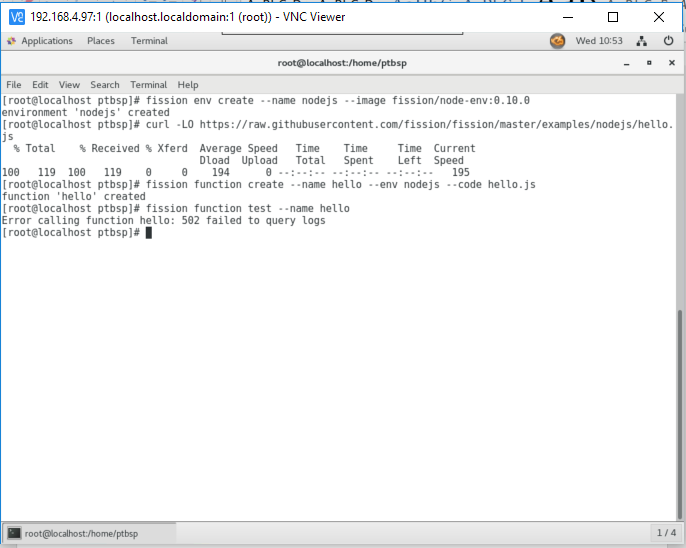
Membuat contoh funsi menggunakan fission

$ fission env create --name nodejs --image fission/node-env:0.10.0

$ curl -LO https://raw.githubusercontent.com/fission/fission/master/examples/nodejs/hello.js

$ fission function create --name hello --env nodejs --code hello.js

$ fission function test --name hello



Namun test fungsi masih error