**[ubuntu 설치]**

**Ansible Initial Setting**

$ sudo apt-get update

$ sudo apt-get install ssh git python2.7 python-pip

$ sudo pip install –U pip

$ sudo pip install –U jinja2

$ sudo apt-get install software-properties-common

$ sudo apt-add-repository ppa:ansible/ansible

$ sudo apt-get update

$ sudo apt-get install ansible python-netaddr

$ ssh-keygen -f "/home/stack/.ssh/known\_hosts" -R localhost

$ ssh-keygen

$ cp ~/.ssh/id\_rsa.pub ~/.ssh/authorized\_keys

$ vi /etc/sudoers에서 stack ALL=(ALL:ALL) NOPASSWD: ALL 추가

$ ssh localhost시 패스워드 없이 되는지

$ sudo fdisk –l 시 패스워드 없이 되는지

**Install Kubernetes by Kubespray**

$ git clone https://github.com/kubernetes-incubator/kubespray.git

$ cd kubespray

$ vi inventory/inventory.cfg

localhost ansible\_port=22 ansible\_host=localhost ip=<YOUR\_VM\_IP> iface=ens3

[etcd]

localhost

[kube-master]

localhost

[kube-node]

localhost

[k8s-cluster:children]

kube-node

kube-master

EOF

$ cp ~/kube-controller-manager.manifest.j2 ~/kubespray/roles/kubernetes/master/templates/manifests/

$ vi inventory/group\_vars/k8s-cluster.yml

26 kube\_version: v1.6.6

72 kube\_network\_plugin: flannel

79 kube\_service\_addresses: 10.96.0.0/16

84 kube\_pods\_subnet: 192.168.0.0/16

108 skydns\_server: "{{ kube\_service\_addresses|ipaddr('net')|ipaddr('10')|ipaddr('address') }}"

$ vi roles/download/defaults/main.yml

29 flannel\_version: v0.7.0

103 kube\_controller\_manager\_repo: "seungkyua/kube-controller-manager"

$ vi roles/kubernetes/master/defaults/main.yml

3 kube\_hostpath\_dynamic\_provisioner: "true"

$ ansible-playbook -u stack -b -i inventory/inventory.cfg cluster.yml

**Kubernetes Config**

// Context config

$ kubectl config set-cluster local-kubernetes --server=http://localhost:8080 --insecure-skip-tls-verify=true

$ kubectl config set-context local --cluster=local-kubernetes --user=root --namespace=kube-system

$ kubectl config use-context local

// Node labelling

$ kubectl label nodes openstack-control-plane=enabled --all --namespace=openstack --overwrite

$ kubectl label nodes openstack-compute-node=enabled --all --namespace=openstack --overwrite

$ kubectl label nodes openvswitch=enabled --all --namespace=openstack --overwrite

// Make general storage

$ cat > ~/general-storage-class.yaml <<EOF

apiVersion: storage.k8s.io/v1beta1

kind: StorageClass

metadata:

namespace: kube-system

name: general

annotations:

storageclass.beta.kubernetes.io/is-default-class: "true"

provisioner: kubernetes.io/host-path

EOF

$ kubectl create -f ~/general-storage-class.yaml

**Install Helm**

$ TMP\_DIR=$(mktemp -d)

$ curl -sSL https://storage.googleapis.com/kubernetes-helm/helm-v2.5.0-linux-amd64.tar.gz | tar -zxv --strip-components=1 -C ${TMP\_DIR}

$ sudo mv ${TMP\_DIR}/helm /usr/local/bin/helm

$ rm -rf ${TMP\_DIR}

$ helm init

$ helm serve & > /dev/null

$ helm repo add local http://localhost:8879/charts

**Install Openstack-Helm**

// Clone Openstack-Helm

$ cd;git clone https://github.com/openstack/openstack-helm.git

$ export WORK\_DIR=$HOME/openstack-helm

$ cd openstack-helm

$ make

//Openstack Environment Servie

$ helm install --namespace=openstack ${WORK\_DIR}/ingress --name=ingress

$ helm install --namespace=openstack ${WORK\_DIR}/mariadb --name=mariadb --set replicas=1 --set volume.class\_name=general

$ helm install --namespace=openstack ${WORK\_DIR}/memcached --name=memcached

$ helm install --namespace=openstack ${WORK\_DIR}/etcd --name=etcd-rabbitmq

$ helm install --namespace=openstack ${WORK\_DIR}/rabbitmq --name=rabbitmq

//Openstack Install

$ helm install --namespace=openstack ${WORK\_DIR}/keystone --name=keystone

$ helm install --namespace=openstack ${WORK\_DIR}/glance --name=glance --values=${WORK\_DIR}/tools/overrides/mvp/glance.yaml

$ helm install --namespace=openstack ${WORK\_DIR}/nova --name=nova --values=${WORK\_DIR}/tools/overrides/mvp/nova.yaml --set network.metadata.ip=10.96.120.234

$ helm install --namespace=openstack ${WORK\_DIR}/neutron --name=neutron --values=${WORK\_DIR}/tools/overrides/mvp/neutron.yaml --set conf.metadata\_agent.default.neutron.metadata.agent.nova\_metadata\_ip=10.96.120.234

$ helm install --namespace=openstack ${WORK\_DIR}/horizon --name=horizon --set=network.enable\_node\_port=true

**Review**

//Horizon 접속

$ kubectl get svc -n openstack| grep horizon-int

http://<your\_vm\_ip>:<horizon\_node\_port>

ex) http://115.68.227.218:31000

**확인**

stack@ubuntu:~/openstack-helm$ kubectl get pods

NAME READY STATUS RESTARTS AGE

flannel-ubuntu 1/1 Running 1 6h

kube-apiserver-ubuntu 1/1 Running 0 6h

kube-controller-manager-ubuntu 1/1 Running 0 6h

kube-dns-3841192733-9x2m1 3/3 Running 0 6h

kube-dns-3841192733-fmwlm 3/3 Running 0 6h

kube-proxy-ubuntu 1/1 Running 1 6h

kube-scheduler-ubuntu 1/1 Running 0 6h

kubedns-autoscaler-1833630871-7nrzm 1/1 Running 0 6h

tiller-deploy-3703072393-j6t54 1/1 Running 0 4h

stack@ubuntu:~/openstack-helm$ kubectl get pods -n openstack

NAME READY STATUS RESTARTS AGE

etcd-2616180944-3xcnc 1/1 Running 0 4h

glance-api-2457567718-16xp1 1/1 Running 0 4h

glance-bootstrap-g9zf9 0/1 CrashLoopBackOff 41 4h

glance-registry-2200911478-6xclm 1/1 Running 0 4h

horizon-3472765371-vd99w 1/1 Running 0 4h

ingress-api-3539170327-grvp3 0/1 CrashLoopBackOff 95 4h

ingress-error-pages-311805704-wrnwk 1/1 Running 0 4h

keystone-api-2317659808-hh89x 1/1 Running 0 4h

mariadb-0 1/1 Running 0 4h

mariadb-1 1/1 Running 0 4h

mariadb-2 1/1 Running 0 4h

memcached-2584335917-njxhd 1/1 Running 0 4h

neutron-dhcp-agent-08q3n 1/1 Running 0 4h

neutron-l3-agent-r4dg2 1/1 Running 0 4h

neutron-metadata-agent-0n2mv 1/1 Running 0 4h

neutron-server-517446919-qkm6c 1/1 Running 0 4h

nova-api-metadata-3489971480-k6gq9 1/1 Running 1 4h

nova-api-osapi-2421960727-j0rrl 1/1 Running 0 4h

nova-bootstrap-mslhr 0/1 CrashLoopBackOff 46 4h

nova-compute-bgdfn 1/1 Running 1 4h

nova-conductor-3606653488-6p387 1/1 Running 0 4h

nova-consoleauth-317403132-n9db7 1/1 Running 0 4h

nova-libvirt-zlwws 1/1 Running 0 4h

nova-novncproxy-479456006-nkbks 1/1 Running 0 4h

nova-scheduler-4259981746-1wgwr 1/1 Running 0 4h

ovs-agent-hkz95 1/1 Running 0 4h

ovs-db-q6bdt 1/1 Running 0 4h

ovs-vswitchd-7763g 1/1 Running 2 4h

rabbitmq-2221043465-lml3r 1/1 Running 0 4h

rabbitmq-2221043465-v8r34 1/1 Running 1 4h

rabbitmq-2221043465-xrrlp 0/1 CrashLoopBackOff 50 4h

stack@ubuntu:~/openstack-helm$ kubectl get svc

NAME CLUSTER-IP EXTERNAL-IP PORT(S) AGE

kube-dns 10.96.0.10 <none> 53/UDP,53/TCP 6h

tiller-deploy 10.96.69.8 <none> 44134/TCP 4h

stack@ubuntu:~/openstack-helm$ kubectl get svc -n openstack

NAME CLUSTER-IP EXTERNAL-IP PORT(S) AGE

etcd 10.96.39.173 <none> 2379/TCP 4h

glance 10.96.88.36 <none> 80/TCP 4h

glance-api 10.96.209.87 <none> 9292/TCP 4h

glance-reg 10.96.80.156 <none> 80/TCP 4h

glance-registry 10.96.209.73 <none> 9191/TCP 4h

horizon 10.96.160.12 <none> 80/TCP 4h

horizon-int 10.96.83.243 <nodes> 80:31000/TCP 4h

ingress-error-pages None <none> 80/TCP 4h

keystone 10.96.106.237 <none> 80/TCP 4h

keystone-api 10.96.217.117 <none> 80/TCP,35357/TCP 4h

mariadb 10.96.5.4 <none> 3306/TCP 4h

mariadb-discovery None <none> 3306/TCP 4h

memcached 10.96.165.77 <none> 11211/TCP 4h

metadata 10.96.120.234 <none> 80/TCP 4h

neutron 10.96.234.43 <none> 80/TCP 4h

neutron-server 10.96.82.2 <none> 9696/TCP 4h

nova 10.96.110.47 <none> 80/TCP 4h

nova-api 10.96.184.184 <none> 8774/TCP 4h

nova-metadata 10.96.230.103 <none> 8775/TCP 4h

nova-novncproxy 10.96.20.197 <none> 6080/TCP 4h

openstack 10.96.217.22 <none> 80/TCP,443/TCP 4h

rabbitmq 10.96.221.247 <none> 5672/TCP 4h