CROSSENT
ross the bridge to the future

Openstack KUBO Deploying Guide

Abhilash S

㈜크로센트
2018. 07

This report is solely for the use of Crossent. No part of it may be circulated, quoted, or reproduced for distribution outside Crossent organization without prior written approval from Crossent.

Agenda

1. Openstack Infrastructure Setting
2. Deploying Bosh Director on Openstack
3. Deploying KUBO on Openstack
4. KUBO Deployment Architecture on Openstack

Note: This deployment Guide is targeted only Ubuntu 16.04 users.

1. Openstack Infrastructure Setting (1/4)

- Install Pre-Requirements (1/4)

Note: This installation Guide does not provide Deep Dive into Openstack Infrastructure Setting.

Note: This installation Guide requires Openstack Load Balancer (HaProxy Provider).

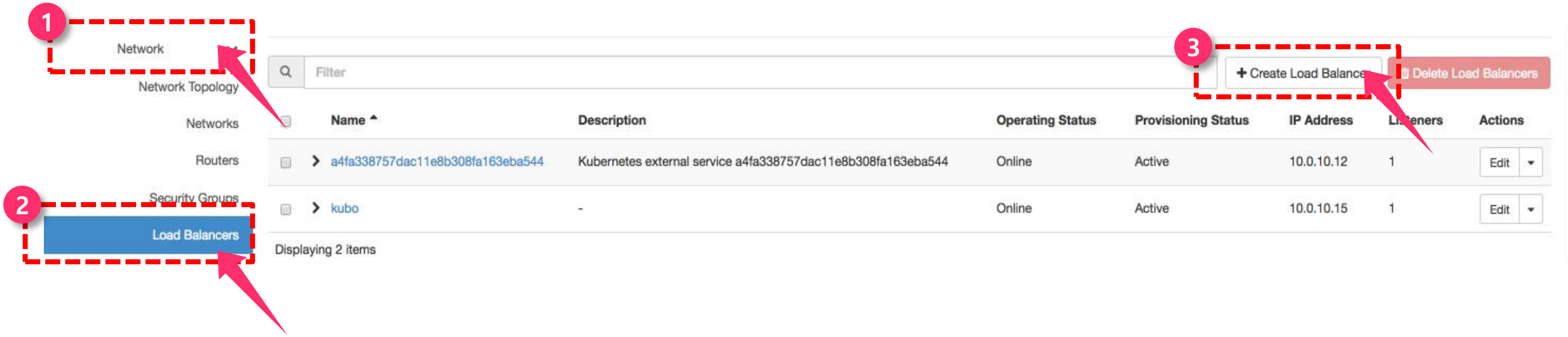
Note: This installation Guide is carried on Openstack OCATA version.

Note: This installation Guide assumes your Openstack Tenant has preconfigured Private and Public Networks .

- ❖ Follow Below link to create Openstack Infrastructure.

https://bosh.io/docs/init-openstack/

- ❖ Create HaProxy Load Balancer in Openstack.



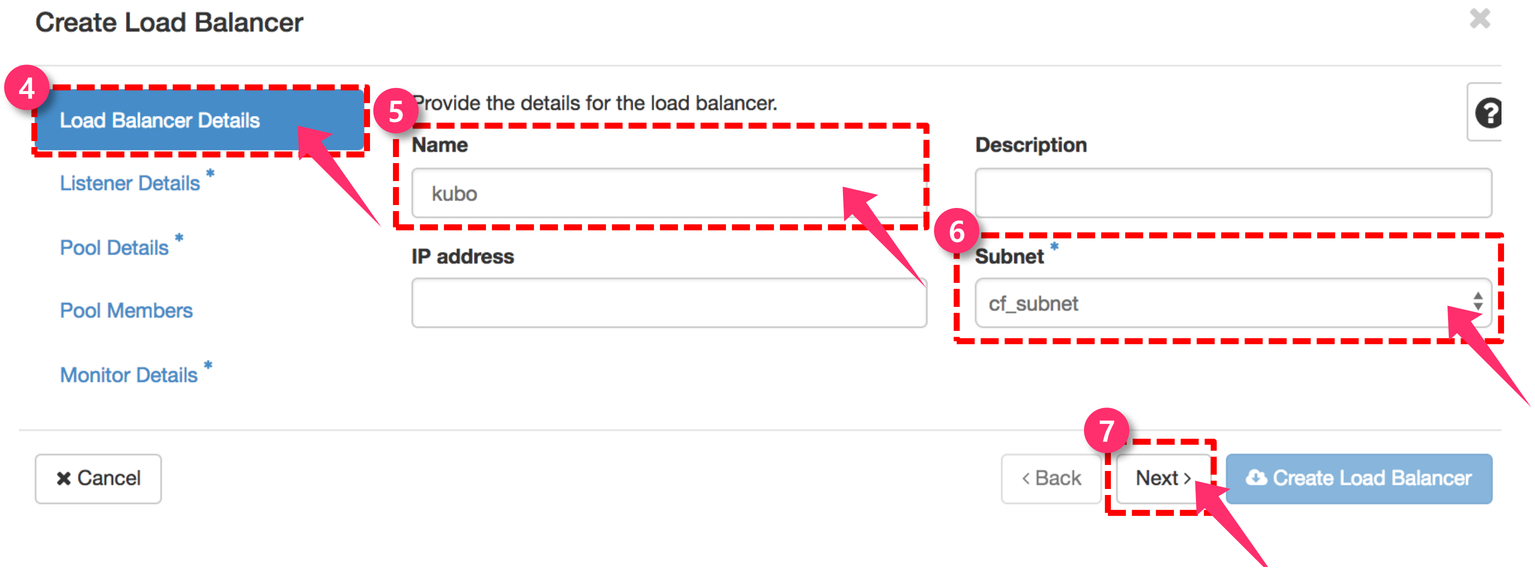
	Name	Description	Operating Status	Provisioning Status	IP Address	Listeners	Actions
□	a4fa338757dac11e8b308fa163eba544	Kubernetes external service a4fa338757dac11e8b308fa163eba544	Online	Active	10.0.10.12	1	Edit
□	kubo	-	Online	Active	10.0.10.15	1	Edit

Displaying 2 items

1. Openstack Infrastructure Setting (2/4)

- Install Pre-Requirements (2/4)

❖ Create HaProxy Load Balancer in Openstack (**Continuing**) .



Create Load Balancer

4 Load Balancer Details

5 Provide the details for the load balancer.

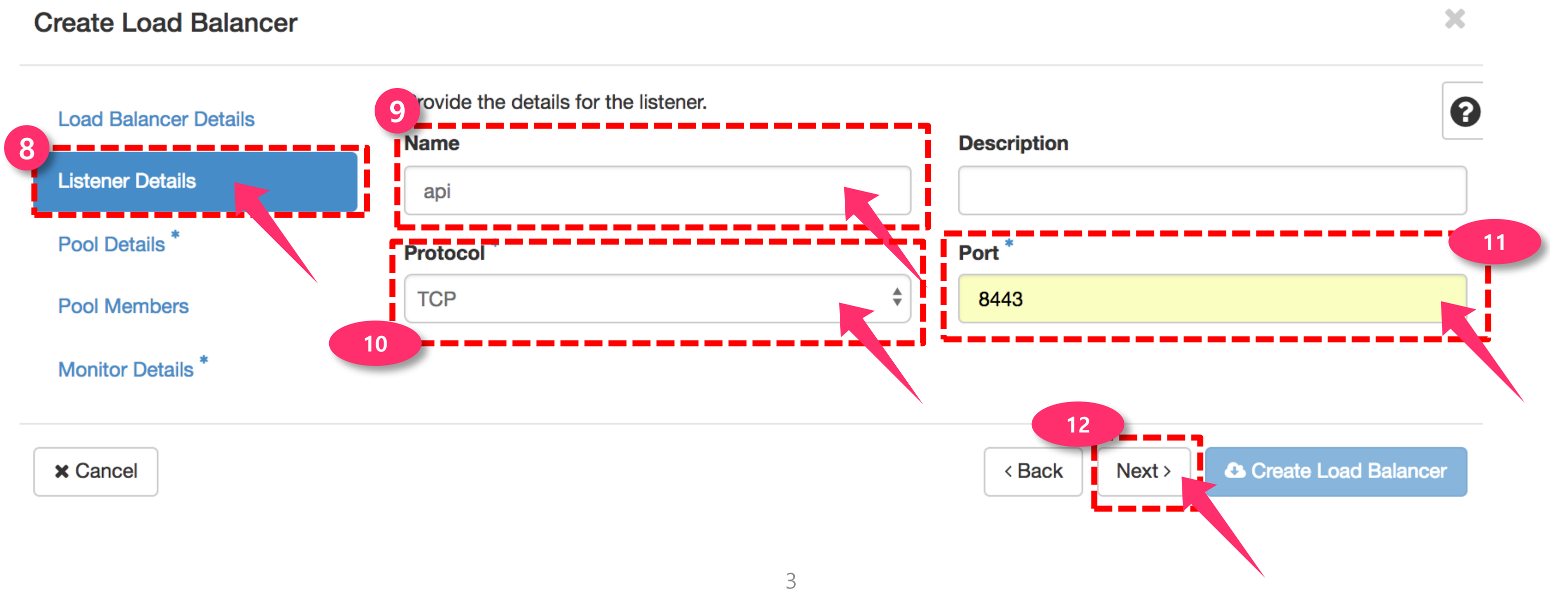
Name: kubo

Description:

IP address:

6 Subnet *: cf_subnet

7 Next ›



Create Load Balancer

8 Listener Details

9 Provide the details for the listener.

Name: api

Description:

10 Protocol: TCP

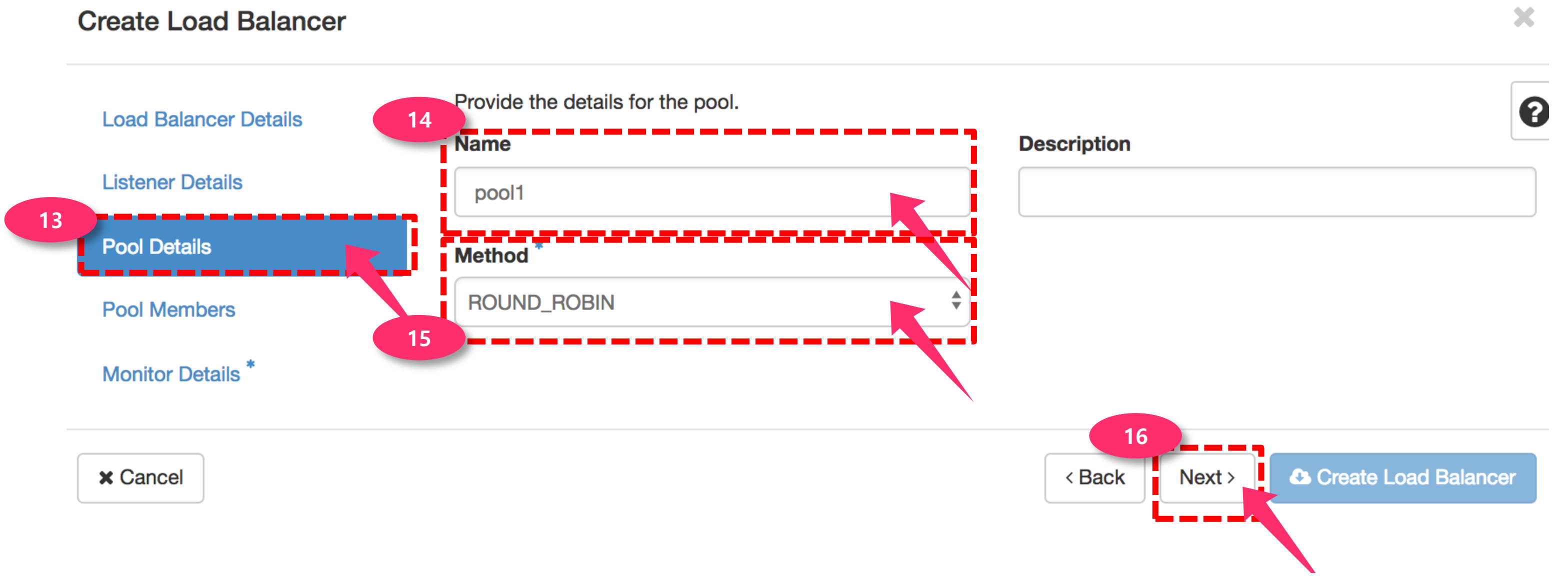
11 Port *: 8443

12 Next ›

1. Openstack Infrastructure Setting (3/4)

- Install Pre-Requirements (3/4)

❖ Create HaProxy Load Balancer in Openstack (**Continuing**) .

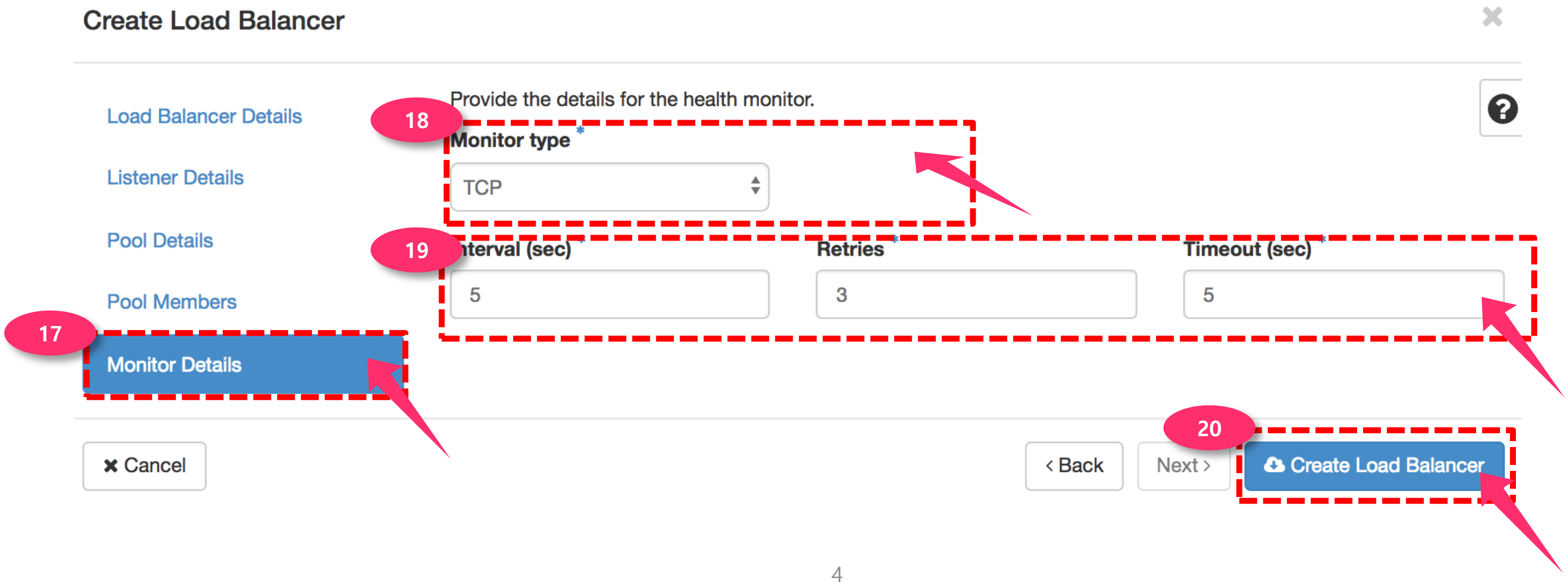


Provide the details for the pool.

Name: pool1

Description:

Method *: ROUND_ROBIN



Provide the details for the health monitor.

Monitor type *: TCP

Interval (sec) *: 5

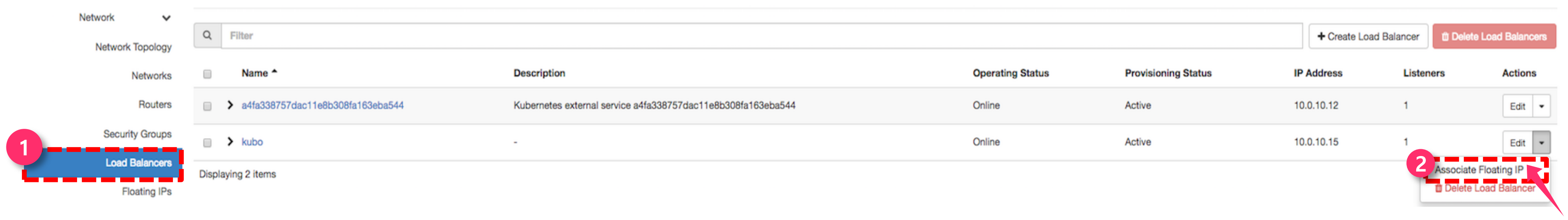
Retries *: 3

Timeout (sec) *: 5

1. Openstack Infrastructure Setting (4/4)

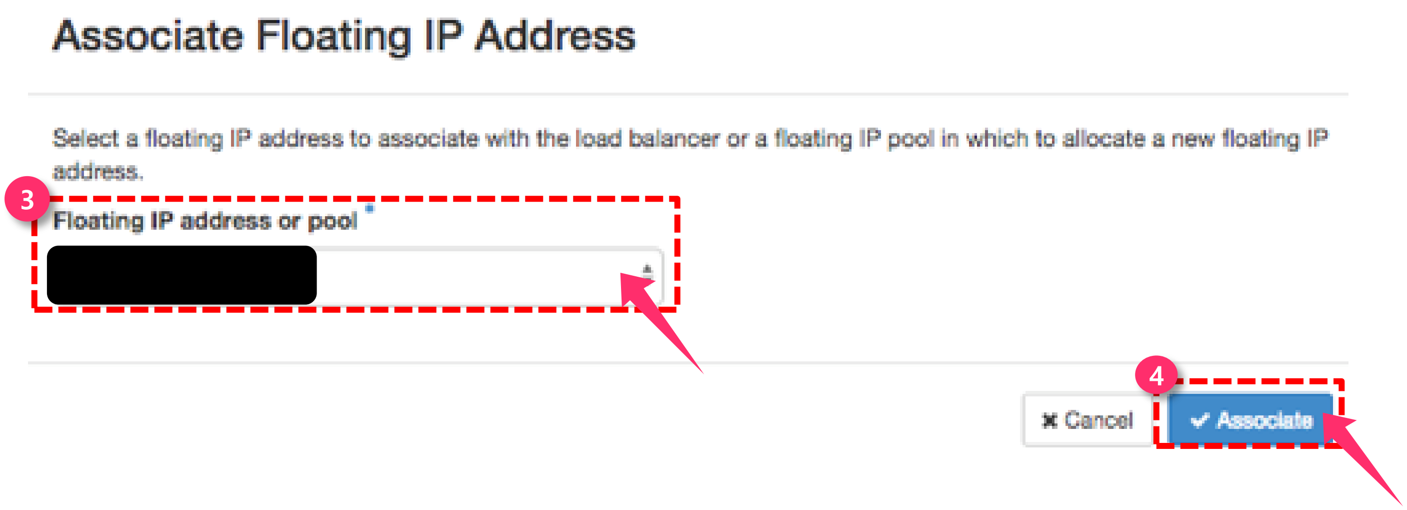
- Install Pre-Requirements (4/4)

❖ Assign Floating IP (Public IP) HaProxy Load Balancer in Openstack.



□	Name ▲	Description	Operating Status	Provisioning Status	IP Address	Listeners	Actions
□	a4fa338757dac11e8b308fa163eba544	Kubernetes external service a4fa338757dac11e8b308fa163eba544	Online	Active	10.0.10.12	1	Edit
□	kubo	-	Online	Active	10.0.10.15	1	Edit

Displaying 2 items



Associate Floating IP Address

Select a floating IP address to associate with the load balancer or a floating IP pool in which to allocate a new floating IP address.

Floating IP address or pool *: [redacted]

2. Deploying Bosh Director on Openstack (1/6)

- Deploying Bosh-Director (1/6)

Note: Perfrom below steps in your Inception.

- ssh into inception

```
$ ssh -i your-key ubuntu@yours-instance-public-ip
$ mkdir ~/workspace && cd ~/workspace
```

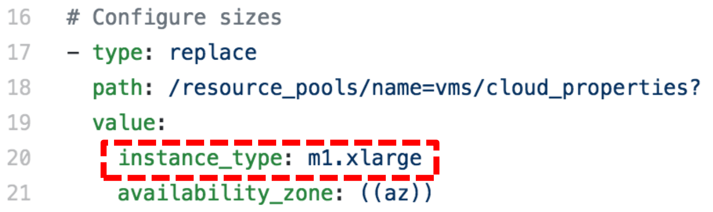
- Next clone the bosh-deployment and kubo-deployment repositories

```
$ git clone https://github.com/cloudfoundry/bosh-deployment
$ git clone https://github.com/cloudfoundry-incubator/kubo-deployment
$ cd bosh-deployment
```

- Edit Openstack cpi.yml file to configure Nat VM.

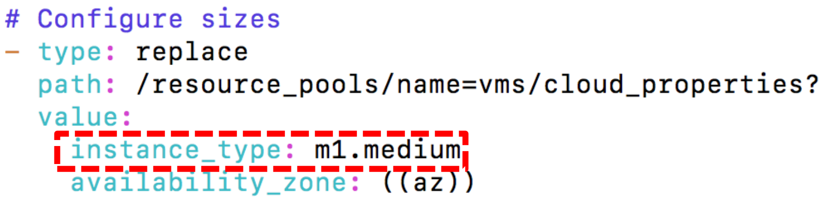
```
$ vi ~/workspace/bosh-deployment/openstack/cpi.yml
```

Before Editing



```
# Configure sizes
- type: replace
  path: /resource_pools/name=vms/cloud_properties?
  value:
    instance_type: m1.xlarge
    availability_zone: ((az))
```

After Editing



```
# Configure sizes
- type: replace
  path: /resource_pools/name=vms/cloud_properties?
  value:
    instance_type: m1.medium
    availability_zone: ((az))
```

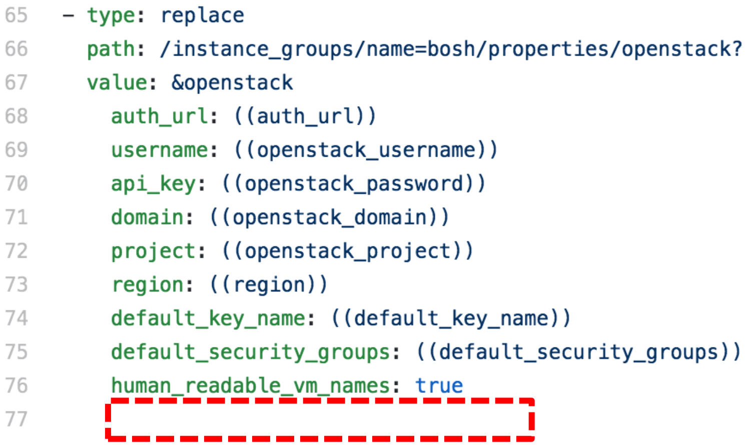
2. Deploying Bosh Director on Openstack (2/6)

- Deploying Bosh-Director (2/6)

❖ Edit Openstack cpi.yml file to configure Cinder Availability Zone .

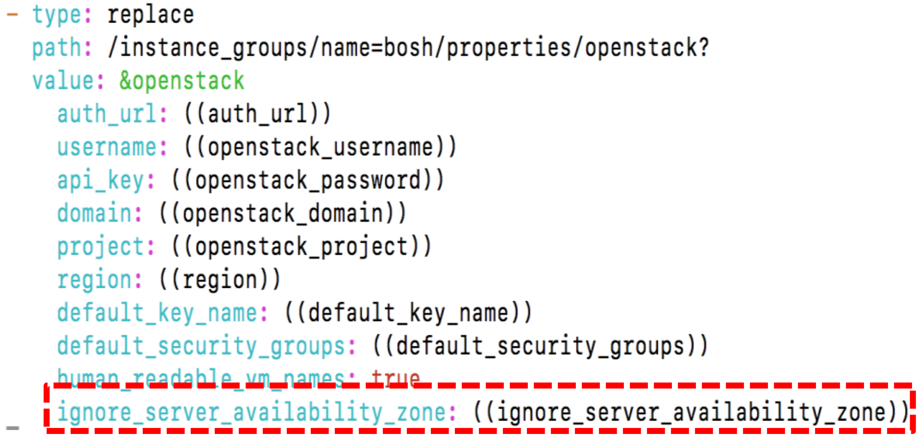
```
$ vi ~/workspace/bosh-deployment/openstack/cpi.yml
```

Before Editing



```
- type: replace
  path: /instance_groups/name=bosh/properties/openstack?
  value: &openstack
    auth_url: ((auth_url))
    username: ((openstack_username))
    api_key: ((openstack_password))
    domain: ((openstack_domain))
    project: ((openstack_project))
    region: ((region))
    default_key_name: ((default_key_name))
    default_security_groups: ((default_security_groups))
    human_readable_vm_names: true

```

After Editing



```
- type: replace
  path: /instance_groups/name=bosh/properties/openstack?
  value: &openstack
    auth_url: ((auth_url))
    username: ((openstack_username))
    api_key: ((openstack_password))
    domain: ((openstack_domain))
    project: ((openstack_project))
    region: ((region))
    default_key_name: ((default_key_name))
    default_security_groups: ((default_security_groups))
    human_readable_vm_names: true
    ignore_server_availability_zone: ((ignore_server_availability_zone))
```

Note: enable “**ignore_server_availability_zone: true**” only when your openstack compute nodes are in multizone. If not set “**ignore_server_availability_zone: false**”.

Note: Our openstack compute nodes are in singlezone. So we set “**ignore_server_availability_zone: false**”.

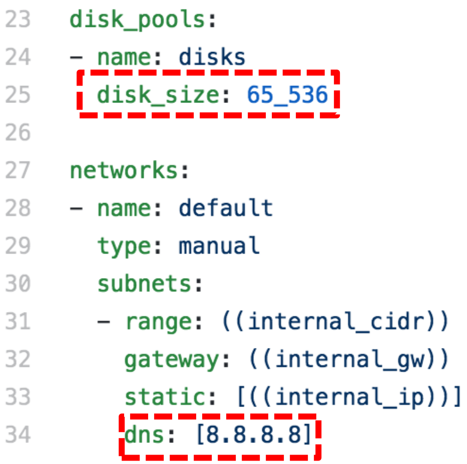
2. Deploying Bosh Director on Openstack (3/6)

- Deploying Bosh-Director (3/6)

❖ Edit Bosh bosh.yml file to configure Gateway.

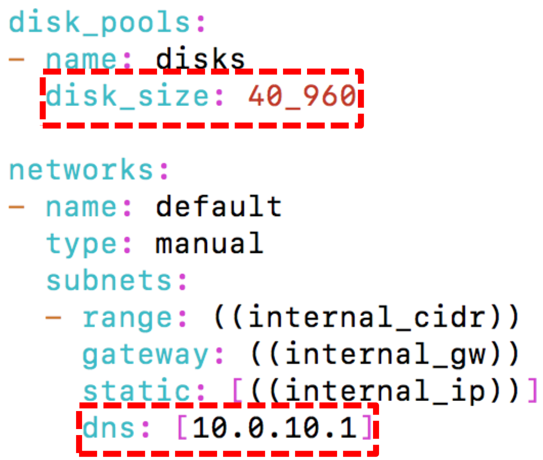
```
$ vi ~/workspace/bosh-deployment/bosh.yml
```

Before Editing



```
disk_pools:
- name: disks
  disk_size: 65_536

networks:
- name: default
  type: manual
  subnets:
  - range: ((internal_cidr))
    gateway: ((internal_gw))
    static: [((internal_ip))]
    dns: [8.8.8.8]
```

After Editing



```
disk_pools:
- name: disks
  disk_size: 40_960

networks:
- name: default
  type: manual
  subnets:
  - range: ((internal_cidr))
    gateway: ((internal_gw))
    static: [((internal_ip))]
    dns: [10.0.10.1]
```

Your Private Subnet Gateway IP

❖ Create directory to store director deployment state and credentials.

```
$ cd ~/workspace/bosh-deployment/ && mkdir kubo && cd kubo
```

2. Deploying Bosh Director on Openstack (4/6)

- Deploying Bosh-Director (4/6)

- Below command create bosh-init vm on virtualbox by means of bosh.yml as base manifest.

- Yaml files with –o command set variables director_name, internal_ip, internal_gw, internal_cidr, outbound_network_name using –v command.

- It also create state.json to record running state and creds.yml **(for certs & credentails)** in your ~/deployments/vbox directory

- Following Command Creates Director's vm and install it's jobs

```
$ bosh create-env ~/workspace/bosh-deployment/bosh.yml
  --state=~/workspace/bosh-deployment/kubo/state.json
  --vars-store=~/workspace/bosh-deployment/kubo/creds.yml
  -o ~/workspace/bosh-deployment/openstack/cpi.yml
  -o ~/workspace/bosh-deployment/jumpbox-user.yml
  -o ~/workspace/bosh-deployment/uaa.yml
  -o ~/workspace/bosh-deployment/credhub.yml
  -o ~/workspace/bosh-deployment/openstack/disable-readable-vm-names.yml
  -o ~/workspace/bosh-deployment/local-dns.yml
  -o ~/workspace/kubo-deployment/configurations/generic/dns-addresses.yml
  -v director_name="kubo"
  -v internal_ip=10.0.10.252
  -v internal_gw=10.0.10.1
  -v internal_cidr=10.0.10.0/24
  -v auth_url=http://your-openstack-identity-public-ip:5000/v3
  -v az=nova
  -v default_key_name=paasta
  -v default_security_groups=[bosh]
  -v net_id=bc8559ef-b82d-4f03-b4f5-650e6d84add8
  -v openstack_password=xxxxx  # Your Openstack Tenant Username
  -v openstack_username=xxxxx  # Your Openstack Tenant Password
  -v openstack_domain=default
  -v openstack_project=crossent
  -v private_key=~/workspace/paasta.pem
  -v region=RegionOne
  -v ignore_server_availability_zone=false # Enable "ignore_server_availability_zone=true" if your openstack compute nodes in multizones
```

2. Deploying Bosh Director on Openstack (5/6)

- Deploying Bosh-Director (5/6)

❖ Log into director with your alias name.

```
$ bosh alias-env kubo -e 10.0.10.252 --ca-cert <(bosh int  ~/workspace/bosh-deployment/kubo/cred
s.yml --path /director_ssl/ca)

$ export BOSH_CLIENT=admin

$ export BOSH_CLIENT_SECRET=`bosh int ~/workspace/bosh-deployment/kubo/creds.yml --path /ad
min_password`
```

❖ Create directory to download releases and stemcells

```
$ mkdir ~/workspace/releases && cd ~/workspace/releases
```

❖ Download Upload Openstack Stemcell

```
$ wget https://s3.amazonaws.com/bosh-core-stemcells/openstack/bosh-stemcell-3586.16-openstack-
kvm-ubuntu-trusty-go_agent.tgz
$ bosh -e kubo upload-stemcell bosh-stemcell-3586.16-openstack-kvm-ubuntu-trusty-go_agent.tgz
```

2. Deploying Bosh Director on Openstack (6/6)

- Deploying Bosh-Director (6/6)

❖ Target & Log into director credhub

```
$ export CREDHUB_CLIENT=credhub-admin
$ export CREDHUB_SECRET=$(bosh int --path /credhub_admin_client_secret ~/workspace/bosh-deplo
yment/kubo/creds.yml)
$ export CREDHUB_CA_CERT=$(bosh int --path /credhub_tls/ca ~/workspace/bosh-deployment/kubo
/creds.yml)
$ credhub login -s https://10.0.1.252:8844 --skip-tls-validation
```

❖ List out Certificates and Passwords in credhub

```
$ credhub find
```

❖ To Delete Certificates and Passwords in credhub

```
$ credhub delete -n /director_name/deployment-name/certificate-name
```

4. Deploying KUBO on Openstack (1/9)

- Deploying Kubo (1/6)

- Download kubo-release.

```
$ cd ~/workspace && mkdir releases && cd ~/workspace/releases
$ wget https://github.com/cloudfoundry-incubator/kubo-release/releases/download/v0.21.0/kubo-release-0.21.0.tgz
```

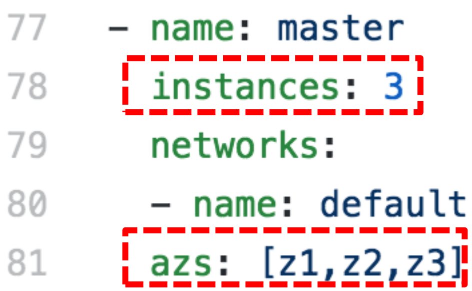
- Upload kubo-release

```
$ bosh -e kubo upload-release ~/workspace/releases/kubo-release-0.21.0.tgz
```

- Edit kubo-deployment cfcr.yml file for deploying kubernetes master on Openstack

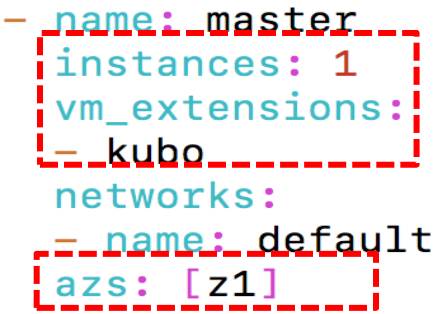
```
$ cd ~/workspace/kubo-deployment
$ vi ~/workspace/kubo-deployment/manifests/cfcr.yml
```

Before Editing



```
- name: master
  instances: 3
  networks:
  - name: default
  azs: [z1,z2,z3]
```

After Editing



```
- name: master
  instances: 1
  vm_extensions:
  - kubo
  networks:
  - name: default
  azs: [z1]
```

Your LB Extension Name in Cloud-config

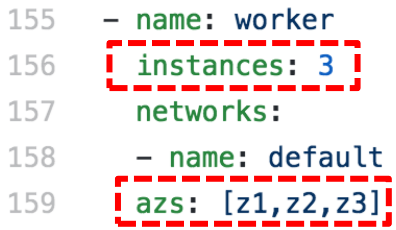
4. Deploying KUBO on Openstack (2/9)

- Deploying Kubo (2/6)

❖ Edit kubo-deployment cfcr.yml file for deploying Kubernetes worker-nodes Openstack

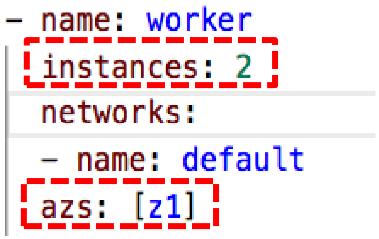
```
$ vi ~/workspace/kubo-deployment/manifests/cfcr.yml
```

Before Editing



```
- name: worker
  instances: 3
  networks:
  - name: default
  azs: [z1,z2,z3]
```

After Editing

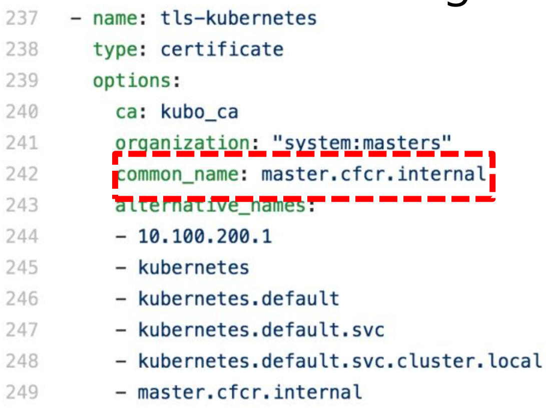


```
- name: worker
  instances: 2
  networks:
  - name: default
  azs: [z1]
```

❖ Edit kubo-deployment cfcr.yml file for deploying kubernetes master certificates.

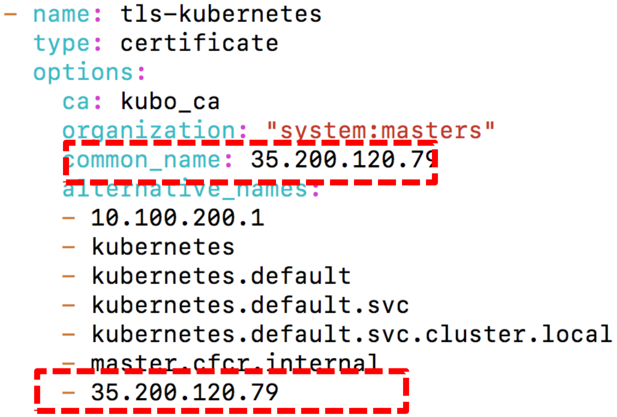
```
$ vi ~/workspace/kubo-deployment/manifests/cfcr.yml
```

Before Editing



```
- name: tls-kubernetes
  type: certificate
  options:
    ca: kubo_ca
    organization: "system:masters"
    common_name: master.cfcr.internal
    alternative_names:
    - 10.100.200.1
    - kubernetes
    - kubernetes.default
    - kubernetes.default.svc
    - kubernetes.default.svc.cluster.local
    - master.cfcr.internal
```

After Editing



```
- name: tls-kubernetes
  type: certificate
  options:
    ca: kubo_ca
    organization: "system:masters"
    common_name: 35.200.120.79
    alternative_names:
    - 10.100.200.1
    - kubernetes
    - kubernetes.default
    - kubernetes.default.svc
    - kubernetes.default.svc.cluster.local
    - master.cfcr.internal
    - 35.200.120.79
```

Note: Change 35.200.120.79 IP according to your LB Floating IP.

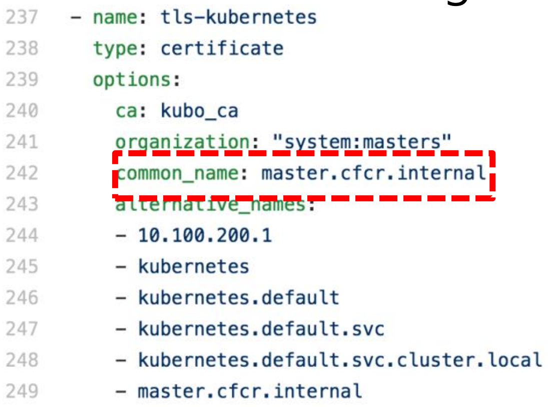
4. Deploying KUBO on Openstack (2/9)

- Deploying Kubo (2/6)

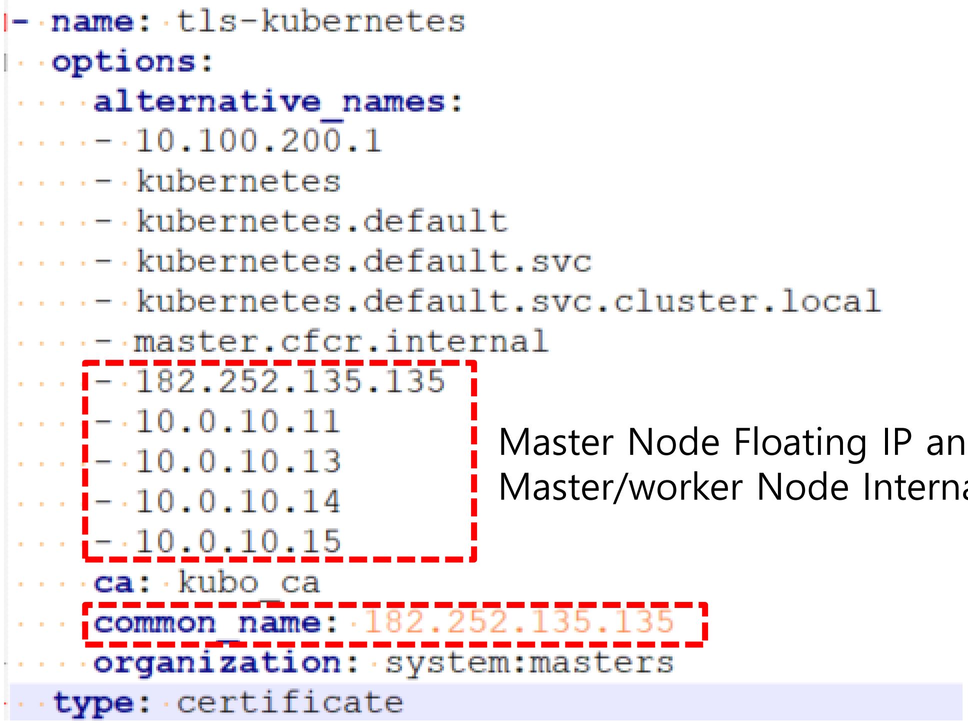
❖ If you Use prometheus monitoring like blow

```
$ vi ~/workspace/kubo-deployment/manifests/cfcr.yml
```

Before Editing



```
- name: tls-kubernetes
  type: certificate
  options:
    ca: kubo_ca
    organization: "system:masters"
    common_name: master.cfcr.internal
    alternative_names:
    - 10.100.200.1
    - kubernetes
    - kubernetes.default
    - kubernetes.default.svc
    - kubernetes.default.svc.cluster.local
    - master.cfcr.internal
```



```
- name: tls-kubernetes
  options:
    alternative_names:
    - 10.100.200.1
    - kubernetes
    - kubernetes.default
    - kubernetes.default.svc
    - kubernetes.default.svc.cluster.local
    - master.cfcr.internal
    - 182.252.135.135
    - 10.0.10.11
    - 10.0.10.13
    - 10.0.10.14
    - 10.0.10.15
    ca: kubo_ca
    common_name: 182.252.135.135
    organization: system:masters
  type: certificate
```

Master Node Floating IP and
Master/worker Node Internal IP ADD

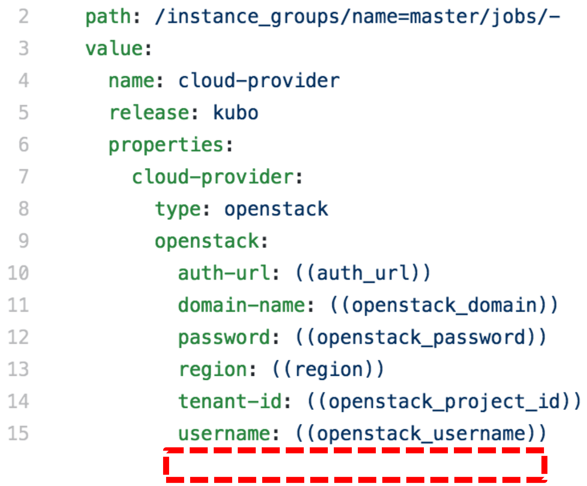
4. Deploying KUBO on Openstack (3/9)

- Deploying Kubo (3/6)

❖ Edit kubo-deployment cloud-provider.yml if your openstack compute nodes are in multizone.

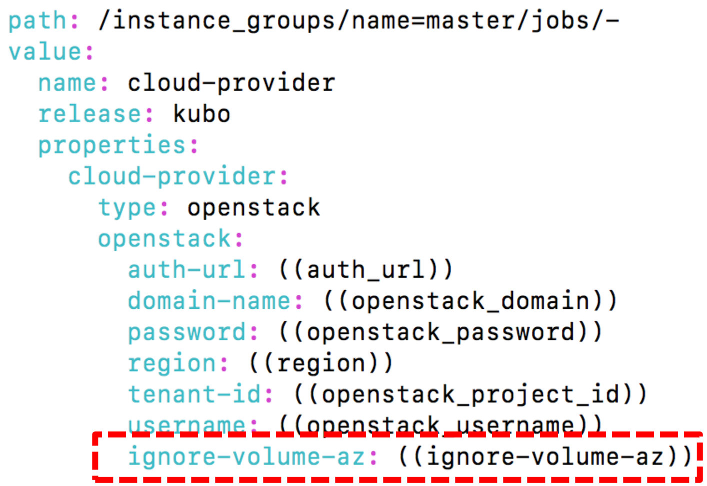
```
$ vi ~/ workspace/kubo-deployment/manifests/ops-files/iaas/openstack/cloud-provider.yml
```

Before Editing



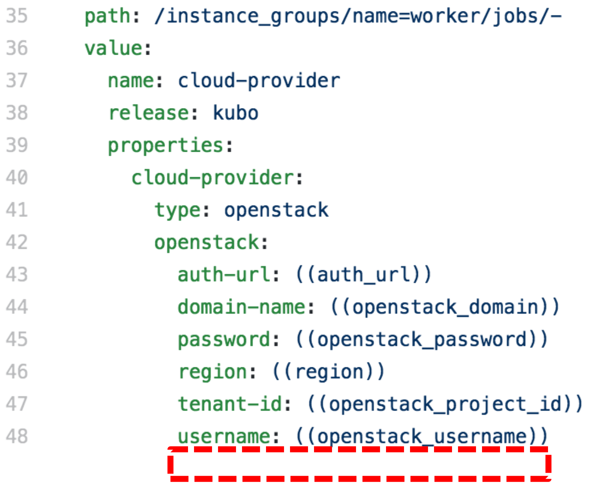
```
path: /instance_groups/name=master/jobs/-
value:
  name: cloud-provider
  release: kubo
  properties:
    cloud-provider:
      type: openstack
      openstack:
        auth-url: ((auth_url))
        domain-name: ((openstack_domain))
        password: ((openstack_password))
        region: ((region))
        tenant-id: ((openstack_project_id))
        username: ((openstack_username))
```

After Editing



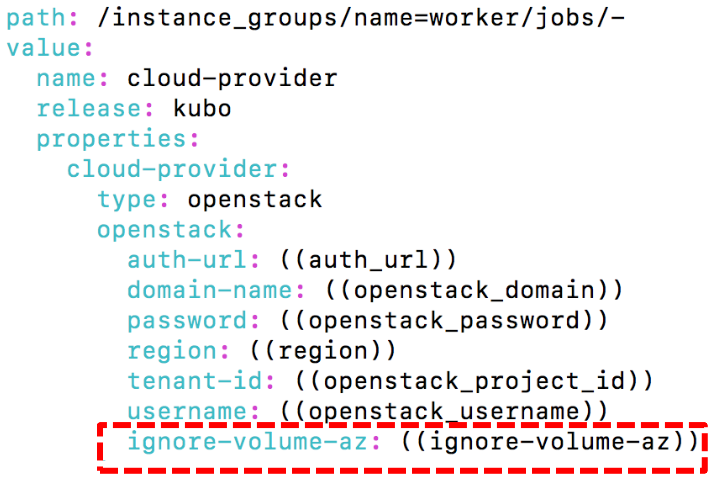
```
path: /instance_groups/name=master/jobs/-
value:
  name: cloud-provider
  release: kubo
  properties:
    cloud-provider:
      type: openstack
      openstack:
        auth-url: ((auth_url))
        domain-name: ((openstack_domain))
        password: ((openstack_password))
        region: ((region))
        tenant-id: ((openstack_project_id))
        username: ((openstack_username))
        ignore-volume-az: ((ignore-volume-az))
```

Before Editing



```
path: /instance_groups/name=worker/jobs/-
value:
  name: cloud-provider
  release: kubo
  properties:
    cloud-provider:
      type: openstack
      openstack:
        auth-url: ((auth_url))
        domain-name: ((openstack_domain))
        password: ((openstack_password))
        region: ((region))
        tenant-id: ((openstack_project_id))
        username: ((openstack_username))
```

After Editing



```
path: /instance_groups/name=worker/jobs/-
value:
  name: cloud-provider
  release: kubo
  properties:
    cloud-provider:
      type: openstack
      openstack:
        auth-url: ((auth_url))
        domain-name: ((openstack_domain))
        password: ((openstack_password))
        region: ((region))
        tenant-id: ((openstack_project_id))
        username: ((openstack_username))
        ignore-volume-az: ((ignore-volume-az))
```

3. Deploying KUBO on Openstack (4/9)

- Deploying Kubo (4/6)

❖ Create and Save bosh director cloud-config

```
$ vi ~/workspace/kubo-deployment/manifests/cloud-config.yml
```

```
azs:
- cloud_properties:
    availability_zone: nova
  name: z1
compilation:
  az: z1
  network: default
  reuse_compilation_vms: true
  vm_type: minimal
  workers: 4

networks:
- name: default
  subnets:
  - azs:
    - z1
    range: 10.0.10.0/24
    reserved: [10.0.10.2-10.0.10.10]
    static: [10.0.10.11-10.0.10.30]
    gateway: 10.0.10.1
    cloud_properties:
      net_id: bc8559ef-b82d-4f03-b4f5-650e6d84add8
      security_groups: [bosh]
    dns:
    - 10.0.10.1
- name: vip
  type: vip

vm_types:
- cloud_properties:
    instance_type: m1.medium
    root_disk_size_gb: 20
  name: minimal
- cloud_properties:
    instance_type: m1.medium
    root_disk_size_gb: 20
  name: small
- cloud_properties:
    instance_type: m1.large
    root_disk_size_gb: 100
  name: small-highmem

disk_types:
- name: 5120
  disk_size: 5120
- name: 10240
  disk_size: 10240

vm_extensions:
- name: kubo
  cloud_properties:
    security_groups: [bosh]
    loadbalancer_pools:
      - name: pool1
        port: 8443
```

Openstack LB Setting

3. Deploying KUBO on Openstack (5/9)

- Deploying Kubo (5/6)

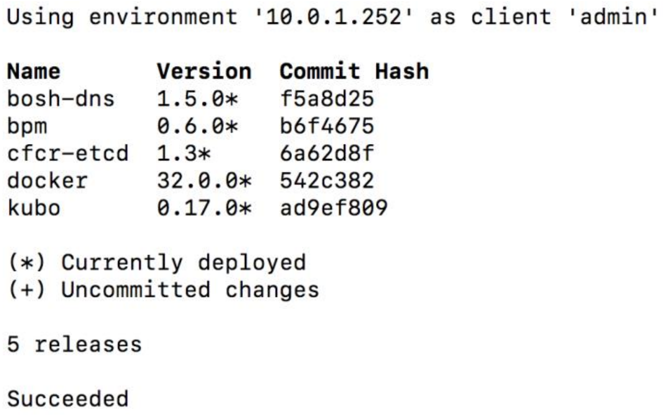
- ❖ Update bosh director cloud-config

```
$ bosh -e kubo update-cloud-config ~/workspace/kubo-deployment/manifests/cloud-config.yml
```

- ❖ Check the desired releases and stemcell to deploy kubo on Openstack bosh

```
$ bosh -e kubo releases
```

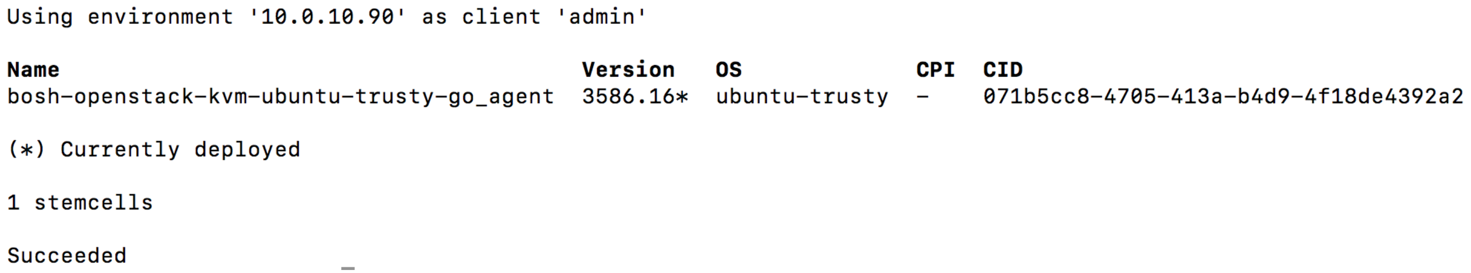
Output



```
Using environment '10.0.1.252' as client 'admin'

Name        Version  Commit Hash
bosh-dns    1.5.0*   f5a8d25
bpm         0.6.0*   b6f4675
cfcr-etcd   1.3*     6a62d8f
docker      32.0.0*  542c382
kubo        0.17.0*  ad9ef809

(*) Currently deployed
(+) Uncommitted changes

5 releases

Succeeded
```

```
$ bosh -e kubo stemcells
```

Output



```
Using environment '10.0.10.90' as client 'admin'

Name                                     Version   OS             CPI  CID
bosh-openstack-kvm-ubuntu-trusty-go_agent 3586.16*  ubuntu-trusty  -    071b5cc8-4705-413a-b4d9-4f18de4392a2

(*) Currently deployed

1 stemcells

Succeeded
```

3. Deploying KUBO on Openstack (6/9)

- Deploying Kubo (6/6)

❖ Deploy Kubo on Openstack

```
$ bosh -e kubo -d cfcr deploy /home/abhisr/workspace/kubo-deployment/manifests/cfcr.yml
          -o /home/abhisr/workspace/kubo-deployment/manifests/ops-files/iaas/openstack/cloud-provider.yml ₩
         -v auth_url=http://your-openstack-identity-public-ip:5000/v3 ₩
          -v openstack_domain=default -v openstack_password= Openstack-Tenant-Password ₩
          -v region=RegionOne -v openstack_project_id=07435feff3104e61990545767f7b7471 ₩
          -v openstack_username=Openstack-Tenant-Username -v ignore-volume-az=false
```

Note: Our openstack compute nodes are in singlezone. So we set "**ignore-volume-az=false**".

Note: If You openstack compute nodes are in MultiZone. So we set "**ignore-volume-az=true**".

Note: If You use prometheus Monitoring you must use blow option

```
-o /home/ubuntu/workspace/v0.21.0/kubo-deployment/manifests/ops-files/allow-privileged-containers.yml ₩
```

❖ Deploy Kubernetes add-ons

```
$ bosh -e kubo -d cfcr run-errand apply-specs
```

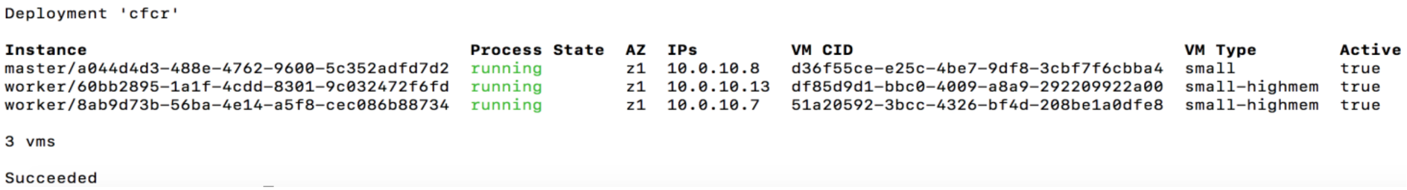
3. Deploying KUBO on Openstack (6/9)

- Deploying Kubo (6/6)

❖ Check Kubernetes Deployment

```
$ bosh -e kubo -d cfcr vms
```

Output



```
Deployment 'cfcr'

Instance                                     Process State  AZ  IPs         VM CID                                   VM Type        Active
master/a044d4d3-488e-4762-9600-5c352adfd7d2  running        z1  10.0.10.8   d36f55ce-e25c-4be7-9df8-3cbf7f6cbba4  small          true
worker/60bb2895-1a1f-4cdd-8301-9c032472f6fd  running        z1  10.0.10.13  df85d9d1-bbc0-4009-a8a9-292209922a00  small-highmem  true
worker/8ab9d73b-56ba-4e14-a5f8-cec086b88734  running        z1  10.0.10.7   51a20592-3bcc-4326-bf4d-208be1a0dfe8  small-highmem  true

3 vms

Succeeded
```

❖ ssh into master and work vms

```
$ bosh -e kubo -d cfcr ssh master
```

```
$ bosh -e kubo -d cfcr ssh worker/61b51092-6a00-4c74-8c10-e3e6b479487e
```

3. Deploying KUBO on Openstack (7/9)

- Accessing Kubernetes (1/3)

❖ Download tls-kubernetes certificate from credhub

```
$ bosh -e kubo -d cfcr int <(credhub get -n "/kubo/cfcr/tls-kubernetes" --output-json) --path=/value
/ca > ~/workspace/kubo-deployment/kubo/kubernetes.crt
```

❖ Download kubo-admin-password password from credhub

```
$ bosh -e kubo -d cfcr int <(credhub get -n "/kubo/cfcr/kubo-admin-password" --output-json) --pat
h=/value >  ~/workspace/kubo-deployment/kubo/kubernetes_pwd.crt
```

❖ Configure Kubernetes Cluster by using above Certificate (kubernetes.crt)

```
$ kubectl config set-cluster "dev" --server https://you-lb-floating-ip:8443 --embed-certs=true --certif
icate-authority=~/workspace/kubo-deployment/kubo/kubernetes.crt
```

❖ Configure Kubernetes User and Context (Note: Use Kubernets User Password from above kubernetes_pwd.crt)

```
$ kubectl config set-credentials "dev" --token=your-kubernetes_pwd.crt-password
```

```
$ kubectl config set-context "dev" --cluster="dev" --user="dev
```

```
$ kubectl config use-context "dev"
```

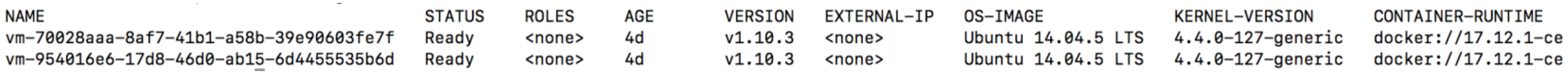
3. Deploying KUBO on Openstack (8/9)

- Accessing Kubernetes (2/3)

❖ Obtain kubernetes node Information

```
$ kubectl get node -o wide
```

Output

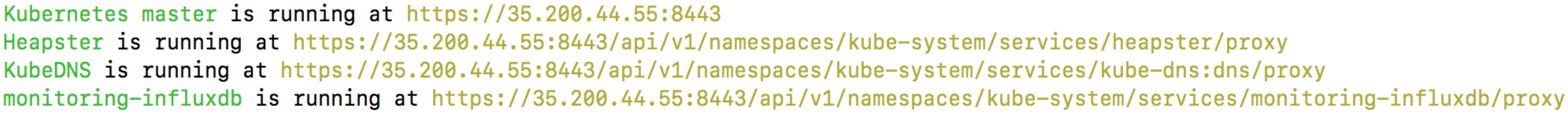


```
NAME                                     STATUS   ROLES    AGE       VERSION   EXTERNAL-IP   OS-IMAGE             KERNEL-VERSION      CONTAINER-RUNTIME
vm-70028aaa-8af7-41b1-a58b-39e90603fe7f  Ready    <none>   4d        v1.10.3   <none>        Ubuntu 14.04.5 LTS   4.4.0-127-generic   docker://17.12.1-ce
vm-954016e6-17d8-46d0-ab15-6d4455535b6d  Ready    <none>   4d        v1.10.3   <none>        Ubuntu 14.04.5 LTS   4.4.0-127-generic   docker://17.12.1-ce
```

❖ Obtain kubernetes Cluster Information

```
$ kubectl cluster-info
```

Output



```
Kubernetes master is running at https://35.200.44.55:8443
Heapster is running at https://35.200.44.55:8443/api/v1/namespaces/kube-system/services/heapster/proxy
KubeDNS is running at https://35.200.44.55:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy
monitoring-influxdb is running at https://35.200.44.55:8443/api/v1/namespaces/kube-system/services/monitoring-influxdb/proxy
```

❖ Get all information regarding your Namespace

```
$ kubectl get pods --namespace=kube-system
$ kubectl get all -n kube-system
```

❖ Check 8001 port is active on your notebook, if active kill the process

```
$ sudo lsof -PiTCP -sTCP:LISTEN
$ sudo kill -9 your-pid-number
```

❖ Accessing to Kubernetes Dashboard

```
$ kubectl proxy
  Starting to serve on 127.0.0.1:8001
$ http://localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/#!/login
```

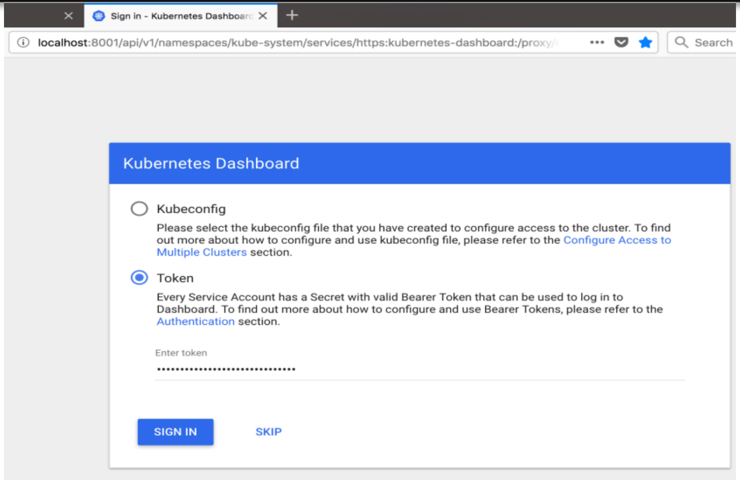
3. Deploying KUBO on Openstack (9/9)

- Accessing Kubernetes (3/3)

❖ Accessing Kubernetes Dashboard in your Browser

Select token option and paste above kubernetes_pwd.crt password to login into your kubernets cluster

Output


Sign in - Kubernetes Dashboard
localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/
Search

Kubernetes Dashboard

○ Kubeconfig

Please select the kubeconfig file that you have created to configure access to the cluster. To find out more about how to configure and use kubeconfig file, please refer to the Configure Access to Multiple Clusters section.

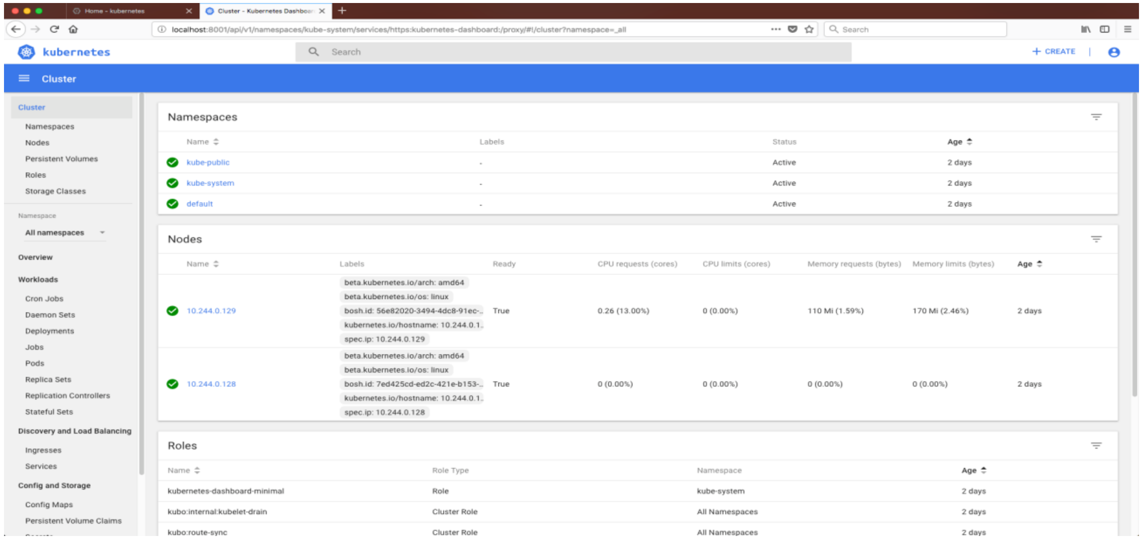
● Token

Every Service Account has a Secret with valid Bearer Token that can be used to log in to Dashboard. To find out more about how to configure and use Bearer Tokens, please refer to the Authentication section.

Enter token: ••••••••••••••••••••••••••••••••

❖ Kubernetes Home Dashboard

Output



Namespaces

Name	Labels	Status	Age
✓ kube-public	-	Active	2 days
✓ kube-system	-	Active	2 days
✓ default	-	Active	2 days

Nodes

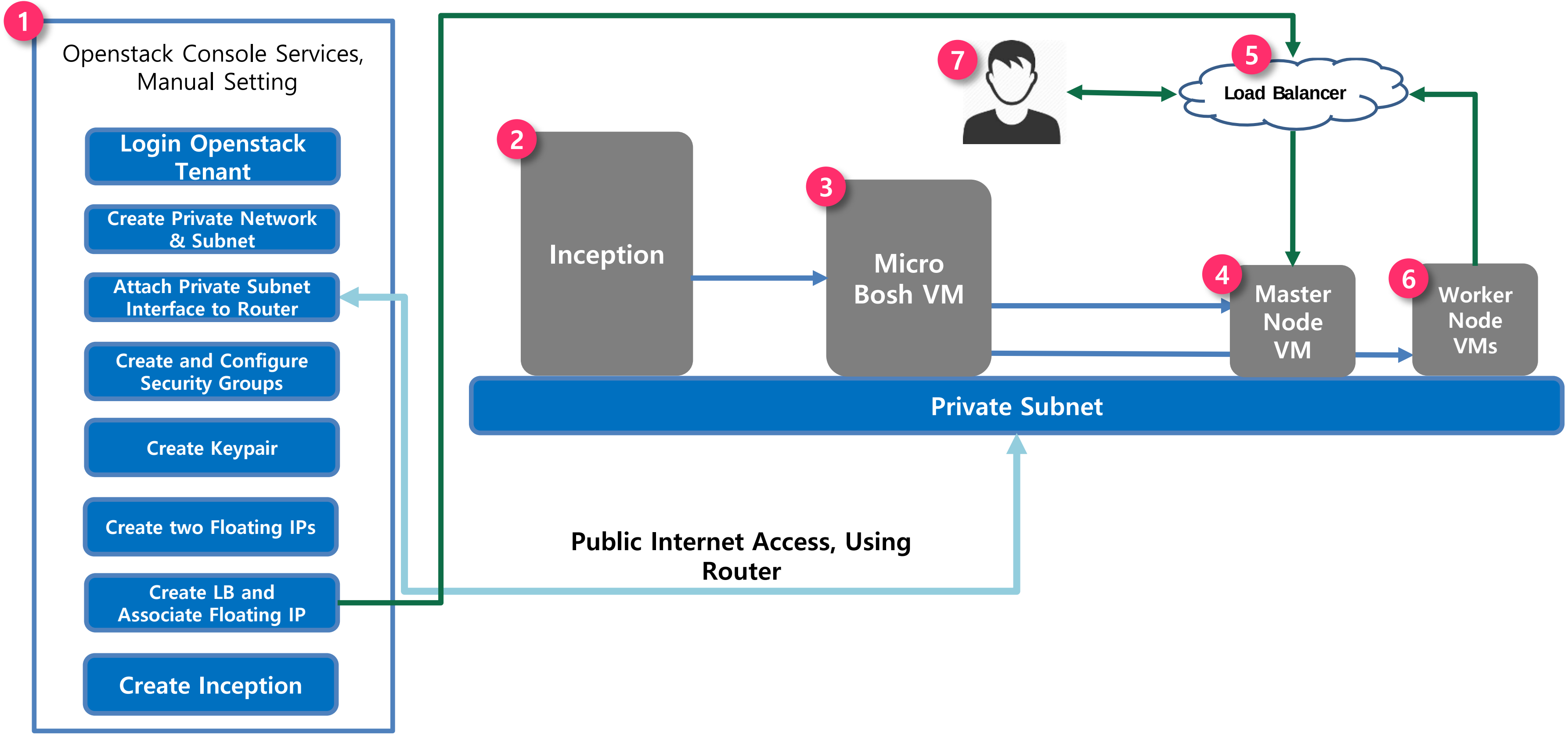
Name	Labels	Ready	CPU requests (cores)	CPU limits (cores)	Memory requests (bytes)	Memory limits (bytes)	Age
✓ 10.244.0.129	beta.kubernetes.io/arch: amd64 beta.kubernetes.io/os: linux bosh.id: 56e82020-3494-4dc8-91ec-.. kubernetes.io/hostname: 10.244.0.1.. spec.ip: 10.244.0.129	True	0.26 (13.00%)	0 (0.00%)	110 Mi (1.59%)	170 Mi (2.46%)	2 days
✓ 10.244.0.128	beta.kubernetes.io/arch: amd64 beta.kubernetes.io/os: linux bosh.id: 7ed425cd-ed2c-421e-b153-.. kubernetes.io/hostname: 10.244.0.1.. spec.ip: 10.244.0.128	True	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	2 days

Roles

Name	Role Type	Namespace	Age
kubernetes-dashboard-minimal	Role	kube-system	2 days
kubo:internal:kubelet-drain	Cluster Role	All Namespaces	2 days
kubo:route-sync	Cluster Role	All Namespaces	2 days

4. KUBO Deployment Architecture on Openstack

- Deployment Architecture


1
Openstack Console Services, Manual Setting
Login Openstack Tenant
Create Private Network & Subnet
Attach Private Subnet Interface to Router
Create and Configure Security Groups
Create Keypair
Create two Floating IPs
Create LB and Associate Floating IP
Create Inception
7
5
Load Balancer
2
Inception
3
Micro Bosh VM
4
Master Node VM
6
Worker Node VMs
Private Subnet
Public Internet Access, Using Router

THANK YOU