

# IBM Cloud Private Installation



# Things to consider (1 of 3)

Type of environment:

- dev, test, staging, prod

Types of workloads:

- middleware
- database requirements

Virtualization platform options:

- VMware
- IBM Cloud
- AWS



Type of environment drives decisions such as:

- Deploy an HA topology or not
- Components to include, such as Vulnerability Advisor or an LDAP directory configuration

Workload requirements drive decisions around:

- Storage considerations
- Middleware to be prepared to install

Virtualization platform influences availability of deployment automation

# Things to consider (2 of 3)

## Architecture/Operating system:

- x86 Linux (RHEL, Ubuntu)
- ppc64le Power Linux (RHEL)
- zLinux (RHEL)



## Network topology:

- "Native" network used for the VMs
- Cluster network (Network CIDR)
- Service network (Service Cluster IP Range)

## Operating system drives:

- Install archive
- Topology:
  - zLinux supports worker nodes only
  - Docker on RHEL has special storage system requirements (devicemapper, direct-lvm)
  - Homogenous cluster is easier to work with than heterogenous cluster

## Network topology drives:

- Choice of subnet and range (default 10.1.0.0/16)
- Choice of service subnet and range (default 10.0.0.1/24)

# Things to consider (3 of 3)

## Shared storage options:

- GlusterFS
- NFS

## PKI certificates:

- signed
- self-signed



## GlusterFS requires at least three additional VMs:

- Heketi admin server
- Heketi admin server resiliency concerns

## NFS considerations:

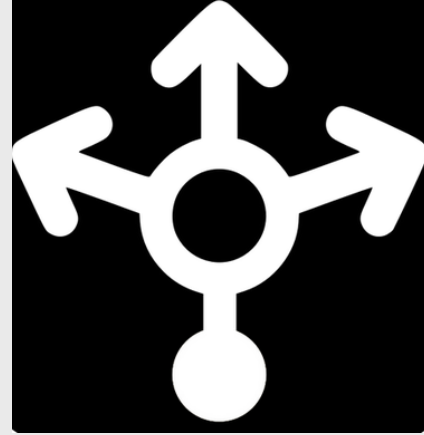
- High availability concerns
- Scalability concerns
- Dynamic storage provisioner in beta using CSI (Container Storage Interface)

# Load balancers and proxies

Master node load balancer

Proxy node load balancer

Ingress server



Installing ICP with an Internet proxy in front of it has special considerations

# “Air gap” installation considerations

Boot node serves as a bridge/jump server

- Boot node has access to Internet
- Boot node has access to all ICP cluster machines

ICP private docker registry

ICP private Helm repository

Will build pipeline have access to public dockerhub.com?

Will ICP have access to the IBM Helm chart repository?



# Pre-installation overview (1 of 4)

## Create the VMs

- Allocate the disk to the file systems
- Disk for Docker (devicemapper, direct-lvm or overlay2?)
- Most VMs are created by cloning the first one

Configure network on each VM with static IP

Set the hostname

Configure DNS with cluster host names or create `/etc/hosts` on boot-master and copy to all nodes



# Pre-installation overview (2 of 4)

Configure passwordless SSH for root from boot-master to all nodes, including boot-master

Install Ansible on boot-master

Configure a non-root Ansible user for all nodes (**optional**)

- Ansible user needs passwordless `sudo` on all nodes including boot-master.
- Configure passwordless SSH for the non-root Ansible user including the boot-master.
- Configuring a non-root Ansible user is commonly required due to restrictions on who has root.





# Pre-installation overview (3 of 4)

Configure yum repos or RHS (preferred)

Update to latest RHEL RPMs (e.g., 7.5) Reboot all nodes to pick up kernel updates.

Install NTP on all nodes

- Start and enable ntpd service
- Set vm.max\_map\_count on all nodes
- Set immediately and for reboot in `/etc/sysctl.conf`
- **Ansible** `lineinfile path=/etc/sysctl.conf line='vm.max_map_count=262144' insertafter=EOF state=present`



# Pre-installation overview (4 of 4)

## Install Docker on all nodes

- Start and enable Docker service
- **Ansible** `lineinfile path='/lib/systemd/system/docker.service' line='MountFlags=shared' insertafter='StartLimitInterval=*' state=present`
- Restart docker on all nodes



# Installation

## Options:

- Get them right, often the only (clean) way to change them is to do a reinstall.
- Walk through the configuration options in config.yaml



# Installation (continued)

## Overview:

- Load docker images from ICP install tar ball on the boot-master node.
- Get the initial ICP install artifacts
- Configure ICP hosts file; copy root ssh id\_rsa to ssh\_key; edit config.yaml
- Stop firewalld on all nodes
- Move ICP install tar ball to images directory in `<ICP_HOME>/cluster`
- Start the install



# Post installation

## Various simple “smoke tests”

- Log into the console and browse around
- Check that all pods are “ready”
- Inspect the installation log

Install kubectl on all of the master nodes (at a minimum on the boot-master node)

- ``docker run --net=host -v /usr/local/bin:/data ibmcom/kubernetes:v1.9.1-ee cp /kubectl /data``

Install ICP CLI on boot master

Install Helm on the boot master



# Post installation (continued)

Install a sample application

- BlueCompute – relatively comprehensive microservices application
- Kube Toy – Fun “hello, world” app
- Or another sample application



