Ministry of Housing and Urban Affairs
Government of India

State Infra prerequisites for implementation kick-off

Outline

1. Overview
2. Requirements
3. The basic need to provision Kubernetes Cluster
4. Check Network Adapters
5. Master Cluster Master Node(s)
6. Worker Node(s)& User Cluster Worker Nodes
7. Complete Infra Specifications

Overview

This page discusses the infrastructure requirements for UPYOG services. It also explains why UPYOG services are containerised and deployed on Kubernetes.

Requirements

UPYOG Infra is abstracted to **Kubernetes** which is an open-source containers orchestration platform that helps in abstracting a variety of infra types that are being available across each state, like Physical, VMs, on-premises clouds(**VMware, OpenStack, Nutanix**, etc.), commercial clouds (**Google, AWS, Azure, etc**.), SDC and NIC into a standard infra type. Essentially it unifies various infra types into a standard and single type of infrastructure and thus UPYOG becomes **multi-cloud supported, portable, extensible, high-performant and scalable** containerized workloads and services. This facilitates both declarative configuration and automation. Kubernetes services, eco-system, support and tools are widely available.

The basic need to provision Kubernetes Cluster

Kubernetes as such is a set of components that designated jobs of scheduling, controlling, monitoring

Master Cluster

- 3 or more machines running one of:
    - Ubuntu 16.04+
    - Debian 9
    - CentOS 7
    - RHEL 7
    - Container Linux (tested with 1576.4.0)
- 4 GB or more of RAM per machine (any less will leave little room for your apps)
- 2 CPUs or more

User Cluster

- 3 or more machines running one of:
    - Ubuntu 16.04+
    - Debian 9
    - CentOS 7

    - RHEL 7
    - Container Linux (tested with 1576.4.0)
- 2 GB or more of RAM per machine (any less will leave little room for your apps)
- 2 CPUs or more
- Full network connectivity between all machines in the cluster (public or private network is fine)
- Unique hostname, MAC address, and product_uuid for every node.
- Certain ports are open on your machines. See below for more details
- Swap disabled. You must disable swap in order for the Kubelet to work properly

Verify the MAC Address and `product_uuid` Are Unique for Every Node

- You can get the MAC address of the network interfaces using the command `ip link` or `ifconfig -a`
- The product_uuid can be checked by using the command `sudo cat /sys/class/dmi/id/product_uuid`

It is very likely that hardware devices will have unique addresses, although some virtual machines may have identical values. Kubernetes uses these values to uniquely identify the nodes in the cluster. If these values are not unique to each node, the installation process may fail.

Check Network Adapters

If you have more than one network adapter, and your Kubernetes components are not reachable on the default route, we recommend you add IP route(s) so Kubernetes cluster addresses go via the appropriate adapter.

Check Required Ports

Master Cluster Master Node(s)

Protocol	Direction	Port Range	Purpose
TCP	Inbound	6443*	Kubernetes API server
TCP	Inbound	2379-2380	etcd server client API
TCP	Inbound	10250	kubelet API
TCP	Inbound	10251	kube-scheduler
TCP	Inbound	10252	kube-controller-manager
TCP	Inbound	10255	Read-only kubelet API

Worker Node(s)& User Cluster Worker Nodes

Protocol	Direction	Port Range	Purpose
TCP	Inbound	10250	kubelet API
TCP	Inbound	10255	Read-only kubelet API
TCP	Inbound	30000-32767	NodePort Services**

** Default port range for Nodeport Services.
Any port numbers marked with * are overridable, so you will need to ensure any custom ports you provide are also open.

Complete Infra Specifications

Systems	Specification	Spec/Count	Comment
User Accounts/VPN	Dev, UAT and Prod Envs	3	
User Roles	Admin, Deploy, ReadOnly	3	
OS	Any Linux (preferably Ubuntu/RHEL)	All	
Kubernetes as a managed service or VMs to provision Kubernetes	Managed Kubernetes service with HA/DRS (Or) VMs with 2 vCore, 4 GB RAM, 20 GB Disk	If no managed k8s 3 VMs/env	Dev - 3 VMs UAT - 3VMs Prod - 3VMs
Kubernetes worker nodes or VMs to provision Kube worker nodes.	VMs with 4 vCore, 16 GB RAM, 20 GB Disk / per env	3-5 VMs/env	DEV - 3VMs UAT - 4VMs PROD - 5VMs
Storage (NFS/iSCSI)	Storage with backup, snapshot, dynamic inc/dec	1 TB/env	Dev - 1000 GB UAT - 800 GB PROD - 1.5 TB
VM Instance IOPS	Max throughput 1750 MB/s	1750 MS/s	
Storage IOPS	Max throughput 1000 MB/s	1000 MB/s	
Internet Speed	Min 100 MB - 1000MB/Sec (dedicated bandwidth)		
Public IP/NAT or LB	Internet-facing 1 public ip per env	3	3 Ips
Availability Region	VMs from the different region is preferable for the DRS/HA	at least 2 Regions	

Private vLan	Per env all VMs should within private vLan	3	
Gateways	NAT Gateway, Internet Gateway, Payment and SMS gateway, etc	1per env	
Firewall	Ability to configure Inbound, Outbound ports/rules		
Managed DataBase (or) VM Instance	Postgres 12 above Managed DB with backup, snapshot, logging. (Or) 1 VM with 4 vCore, 16 GB RAM, 100 GB Disk per env.	per env	DEV - 1VMs UAT - 1VMs PROD - 2VMs
CI/CD server self-hosted (or) Managed DevOps	Self Hosted Jenkins: Master, Slave (VM 4vCore, 8 GB each) (Or) Managed CI/CD: NIC DevOps or AWS CodeDeploy or Azure DevOps	2 VMs (Master, Slave)	
Nexus Repo	Self-hosted Artifactory Repo (Or) NIC Nexus Artifactory	1	
DockerRegistry	DockerHub (Or) SelfHosted private docker reg	1	
Git/SCM	GitHub (Or) Any Source Control tool	1	
DNS	main domain & ability to add more sub-domain	1	
SSL Certificate	NIC managed (Or) SDC managed SSL certificate per URL	2 URLs per env	