

# ICN KUD (Kubernetes Deployment)

- Motivation
- Architecture blocks
- Setup information
  - Online setup
  - Offline setup ( WIP )
  - Proposed Workflow diagram
    - Offline setup flow - tested
- Challenges

## Motivation

This project offers a means for deploying a Kubernetes cluster that satisfies the requirements of [ONAP multcloud/k8s plugin](#).

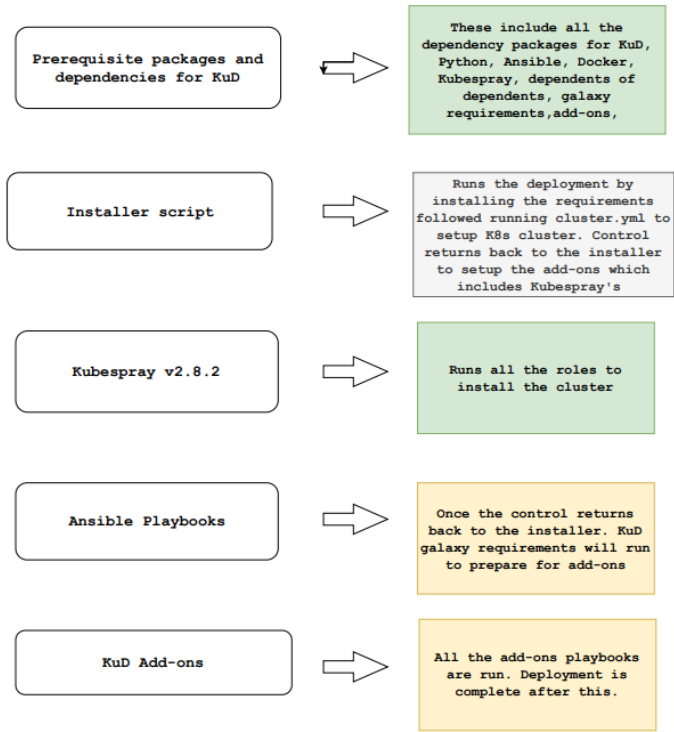
Its ansible playbooks allow provisioning a deployment on Virtual Machines and on Baremetal.

KuD facilitates virtual deployment using Vagrant and Baremetal deployment using the All-in-one script.

[blocked URL](#)

Name	Description	Source	Status
Kubernetes	Base Kubernetes deployment	<a href="#">kubespray</a>	Done
ovn4nfv	Integrates Opensource Virtual Networking	<a href="#">configure-ovn4nfv.yml</a>	Tested
Virtlet	Allows to run VMs	<a href="#">configure-virtlet.yml</a>	Tested
Multus	Provides Multiple Network support in a pod	<a href="#">configure-multus.yml</a>	Tested
NFD	Node feature discovery	<a href="#">configure-nfd.yml</a>	Tested
Istio	Service Mesh platform	<a href="#">configure-istio.yml</a>	Tested

## Architecture blocks



## Setup information

## Online setup

To deploy KuD independent of ICN please refer to the documents/instructions [here](#).

## Offline setup.

### Offline setup flow - tested

1. Get all the dependency packages and resolve the dependency in the right order
2. Install basic components for KuD
3. Run the installer script
4. Docker
5. Ansible
6. Get the kubespray prescribed version listed in KuD
7. Get the correct version of Kubeadm, etcd, hypercube, cni
8. Get the docker images used by kubespray
  - a. load them making sure the right versions are available.
9. We had the override the following defaults in Kubespray:
  - a. Download\_run\_once: true
  - b. Download\_localhost: true
  - c. Skip\_downloads: true
  - d. Strict\_dns\_check: False
  - e. Update\_cache: False
  - f. Helm\_client\_refersh: False
10. Get galaxy-requirements
11. Get galaxy requirements dependents
12. Run the roles
13. Get add-ons
14. Modify the ansible script to not pull from the web and instead use from release dir and run the playbook. (Tested for Multus)
15. Run all the add-on playbooks
16. KuD offline done. Run the test cases to verify

## Challenges

The values listed above are the changes that could be overridden in K8s-cluster.yml. However, some of the changes did not have this option. There are cases where we had to manually change some defaults in Kubespray code.

One of the places is when it checks for docker\_version in ubuntu\_bionic.yml. The existing version is 18.06.0 expected is 18.06.1. Tried to change this by supplying the correct/requested version by Kubespray. Regardless it fails unless we change the hardcoded version in the <https://github.com/kubernetes-sigs/kubespray/blob/release-2.8/roles/container-engine/docker/vars/ubuntu-bionic.yml#L6>

Another case where we encountered issues is when ansible runs a name: ensure docker packages are installed

```
update_cache: "{{ omit if ansible_distribution == 'Fedora' else False }}"
```

<https://github.com/kubernetes-sigs/kubespray/blob/release-2.8/roles/container-engine/docker/tasks/main.yml#L134>

Right now we have the K8s cluster setup in offline mode on the client-server machine replicated in the lab. However, we see that support for the offline version is poor for v2.8.2 currently used by KUD. Setting up the galaxy requirement has been a blocker. We have developed an approach to it to get the add-ons going.

The current system in KuD has a lot of Ansible code that will be rewritten

## Proposed workflow

1. Have KuD live to run for August 15 release.
2. September 15 Priorities
  - a. Update the version of Kubespray from 2.8.2 to 2.11 to leverage the features of Kubespray caching available in Master.
  - b. Test it on KuD-live version
  - c. Have the new daemon sets integrated with the online version since, some of them require a higher version of Kubeadm and kubectl version which is automatically updated once, kubespray is updated.
  - d. Have the existing addons like virtlet, nfd, cmk, rook etc be converted into daemon sets and tested in live KuD for the addons converted into Daemonsets. This depends we should also have non-daemon sets to test out our infra.
  - e. Provide OVN installation package information and OVN daemonset.yaml - Ritu?
  - f. Have docker registry to have come container images to be pulled from the provisioned servers.
3. October 15 Priorities
  - a. Host on Http-server put packages for addons so that the add-on scripts are not manipulated (If any)

- b. Have all the docker images used by Kubespray to pull from here from deployment.
- c. Also, have ansible roles created which will help to maintain a single version of KuD for both online and offline deployment.