



# Kubernetes based Cloud-region support in ONAP to bring up VM and container based VNFs/workloads

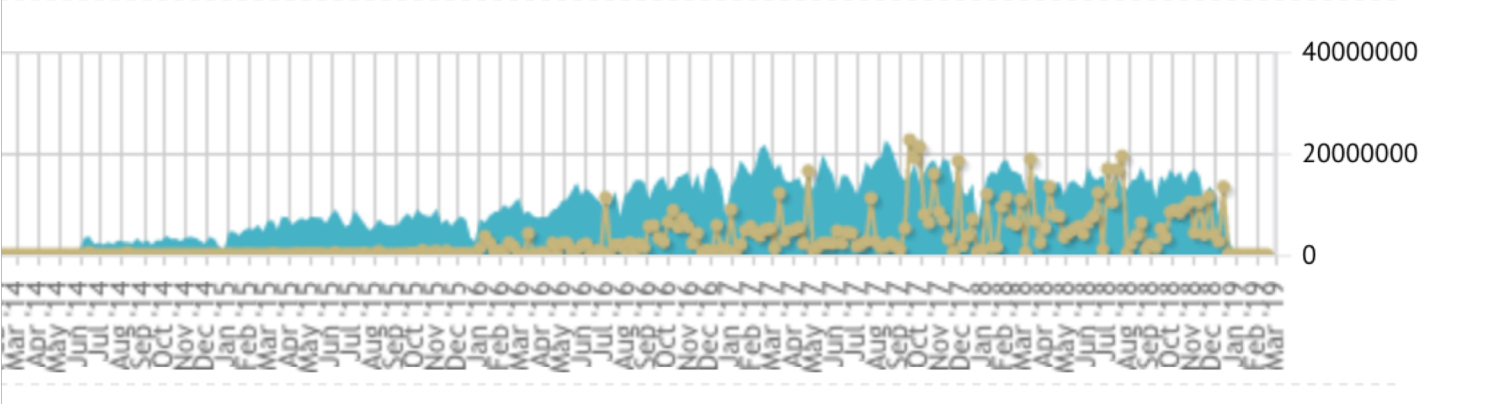
Srinivasa Addepalli  
Victor Morales

# Agenda

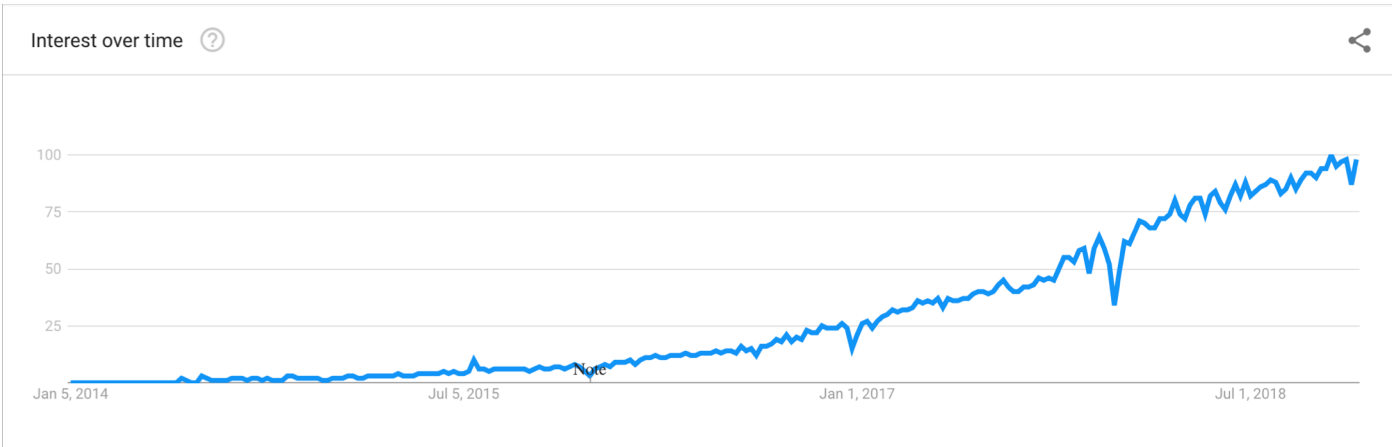
- Background
- Dublin features
- Details
  - K8S plugin
  - KRD
  - vFirewall use case

# Kubernetes by numbers

Kubernetes is becoming a site orchestrator choice to deploy network functions and application workloads in cloud-regions.

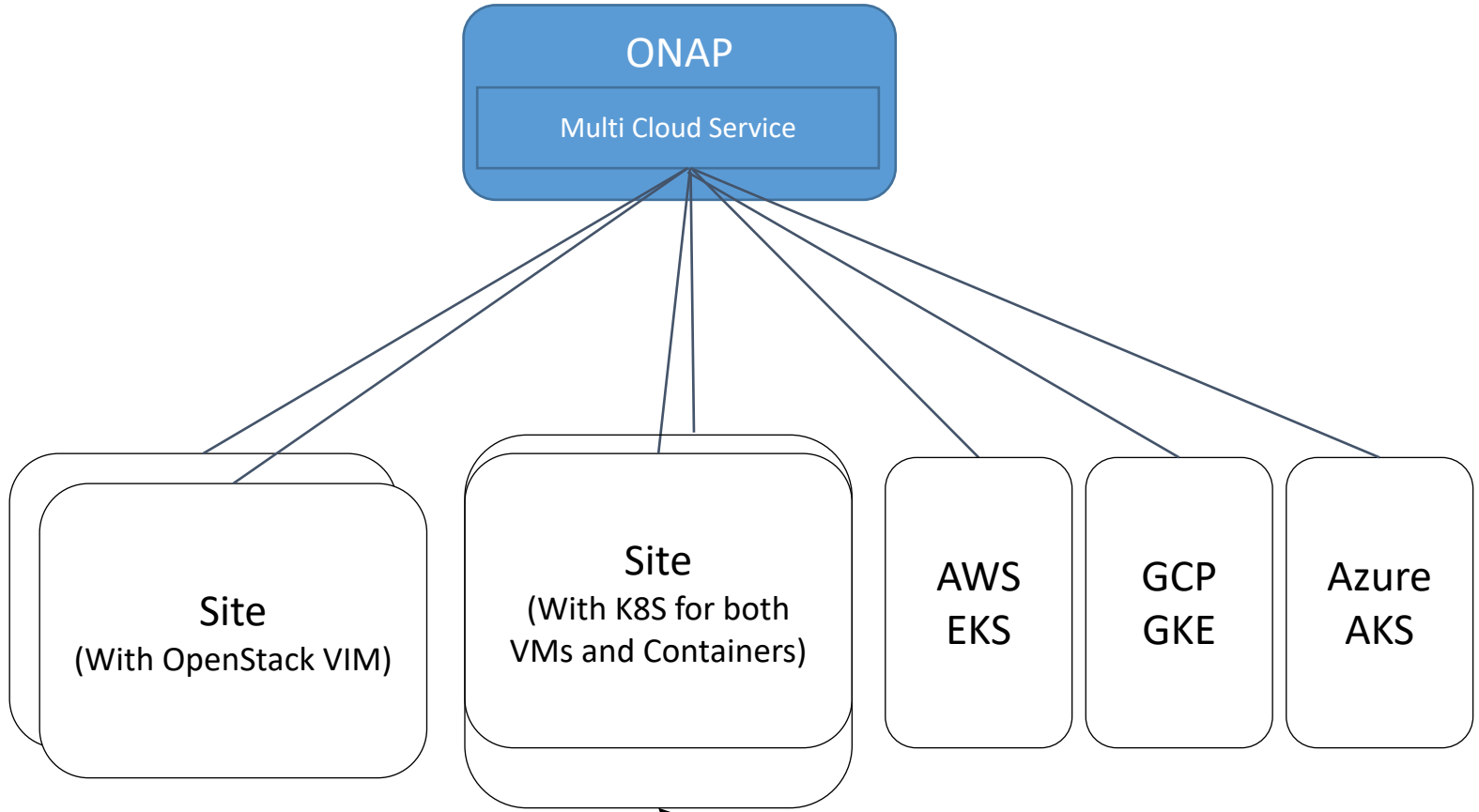


[http://stackalytics.com/?project\\_type=kubernetes-group&release=all&metric=commits](http://stackalytics.com/?project_type=kubernetes-group&release=all&metric=commits)



<https://trends.google.com/trends/explore?date=2014-01-01%202018-12-01&q=%2Fg%2F11b7lxp79d>

# Goal - ONAP support for K8S based Cloud regions



Kubernetes Reference Deployment (KRD)  
<https://git.onap.org/multicloud/k8s/tree/vagrant/>

# Network function requirements

Support for both containerized Network functions & VM based network functions

Multiple virtual networks – Ability to bring up workloads on multiple networks

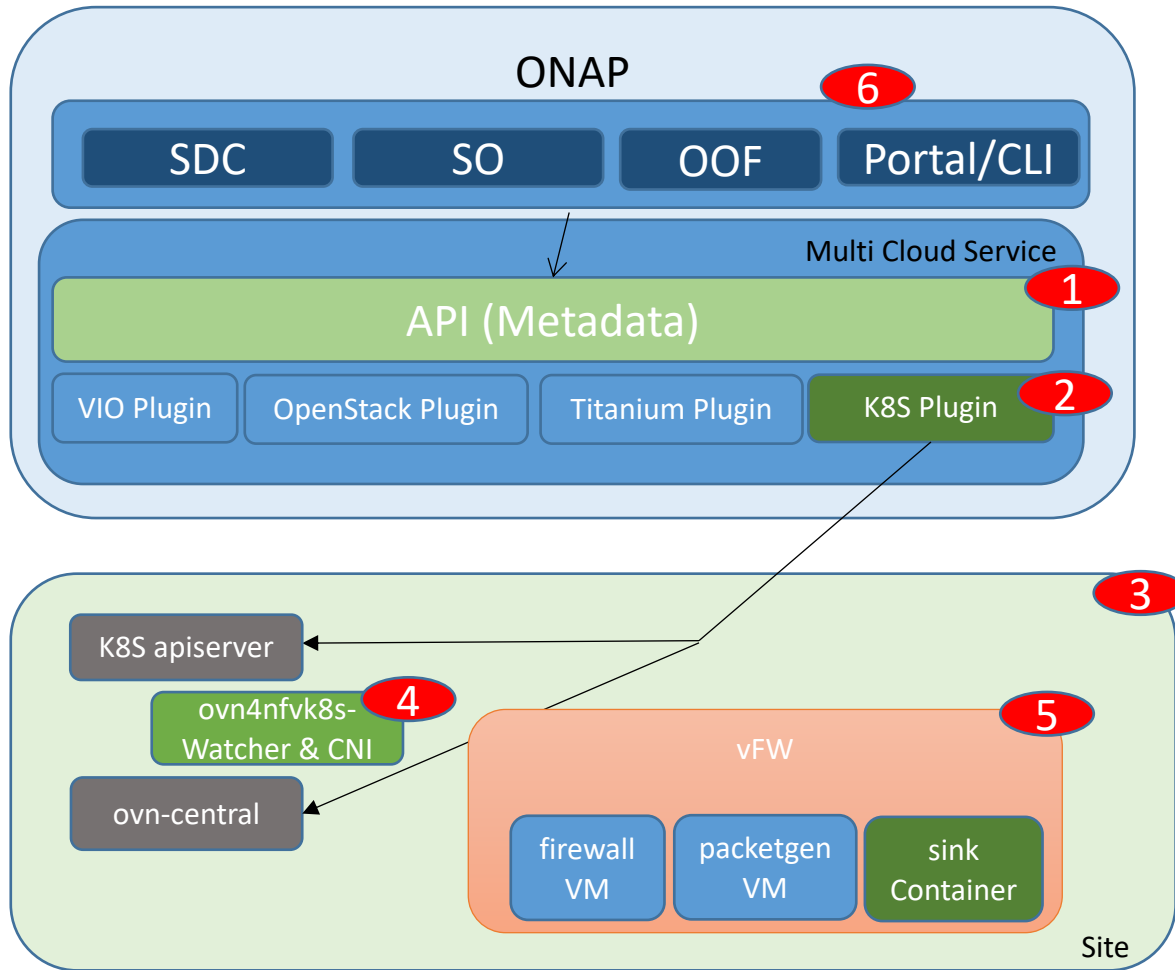
Dynamic creation and deletion of virtual networks

Programmable virtual networks

Provide networks – Ability to place workloads on both virtual and provider networks

High throughput, Low Latency and Jitter workloads

# Work items categories



1. **Uniform API** across cloud technologies (HEAT, K8S, Azure etc..)
2. **K8S Multi-Cloud Service plugin**  
(<https://github.com/onap/multicloud-k8s/tree/master/src/k8splugin>)
  - Support for deployment and services(K8S Helm charts).
  - Networking – OVN, flannel and Multus (Create/Delete VNs, Distributed Router, Gateways, SNAT in Gateway)
3. **Kubernetes Reference Deployment**  
(<https://github.com/onap/multicloud-k8s/tree/master/vagrant>)
  - Installation of software & configuration to make K8S based sites.
  - Additional of Virtlet, Multus, OVN, SRIOV and flannel.
4. **OVN4NFVK8S** (<https://github.com/opnfv/ovn4nfv-k8s-plugin>)
  - Support for multiple virtual networks
  - Support for dynamic creation/deletion of virtual networks
5. **Hybrid vFW ONAP use case** ([https://github.com/onap/multicloud-k8s/blob/master/vagrant/tests/integration\\_vcFW.sh](https://github.com/onap/multicloud-k8s/blob/master/vagrant/tests/integration_vcFW.sh))
  - Firewall and packet generator as Virtual Machines and Sink services containerized.
6. **Integration with SO, OOF and SDC**

# Multi-Cloud Enhancements

Feature	Description
Onboard agent for SDC	<ul style="list-style-type: none"><li>• Gets notified whenever VNF is onboarded</li><li>• Gets the helm charts from the CSAR (ksonnet based templates are for future)</li><li>• Stores them in DB (MongoDB)</li></ul>
K8S Service (continuation of R3 work)	<ul style="list-style-type: none"><li>• Day0 configuration profile support<ul style="list-style-type: none"><li>• Multiple configuration profiles can be created for each on-boarded VNF</li></ul></li><li>• VNF Management<ul style="list-style-type: none"><li>• Instantiate (Fill the parameters using config profile, dynamic information passed from SO, fills config-map data from the profile and make K8S calls to remote cloud-region API server).</li><li>• Terminate VNF, Query (to get IP address assigned and populating A&amp;AI vServer records)</li></ul></li><li>• Support for multiple kinds of resources (Deployment, POD, Config-map, statefulset, services, RBAC, ISTIO, CRDs)</li><li>• Multus, OVN and flannel networking support (SRIOV-NIC and others in future)</li><li>• Deployment of VM based workloads via virtlet (Kubevirt in future releases)</li><li>• Support for VNF replication across multiple cloud-regions that are part of oof-directives(Stretch)</li><li>• Support for dependency management across micro-services using DAG (Stretch)</li><li>• Support one meta ID for the entire VNF to satisfy SO requirements</li></ul>

# Other ONAP projects

Project	Enhancement description
SDC	<ul style="list-style-type: none"><li>• Support for K8S artifacts</li><li>• Support for multiple Cloud formats in a VNF CSAR</li></ul>
SO	<ul style="list-style-type: none"><li>• Make SO independent of Openstack</li><li>• SO to leverage MC always</li><li>• SO-MC API enhancements (to pass userParams to MC)</li></ul> <p>Any scale-out and change management related changes beyond R4</p>
OOF (Stretch)	<ul style="list-style-type: none"><li>• Support for VNF replication across cloud-regions as a policy.</li><li>• Pass multiple cloud-regions as part of oof-directives</li></ul>
Portal & CLI	<ul style="list-style-type: none"><li>• Manage Day0 configuration profiles</li></ul>
A&AI AND ESR (Mainly usage and at the most schema changes)	<ul style="list-style-type: none"><li>• Ensure that ESR GUI is enhanced to take kubeconfig and other reachability information</li></ul>
Integration	<ul style="list-style-type: none"><li>• vFirewall use cases using containers/VMs.</li><li>• EdgeXFoundry use case (non NF)</li></ul>
VNF SDK	Future releases



# K8S Multi-Cloud Service plugin

ONAP Multi-Cloud plugin written in Go lang which offers an API for interacting with Cloud regions supporting Kubernetes.

## Requirements:

- Go 1.11
- Docker
- docker-compose



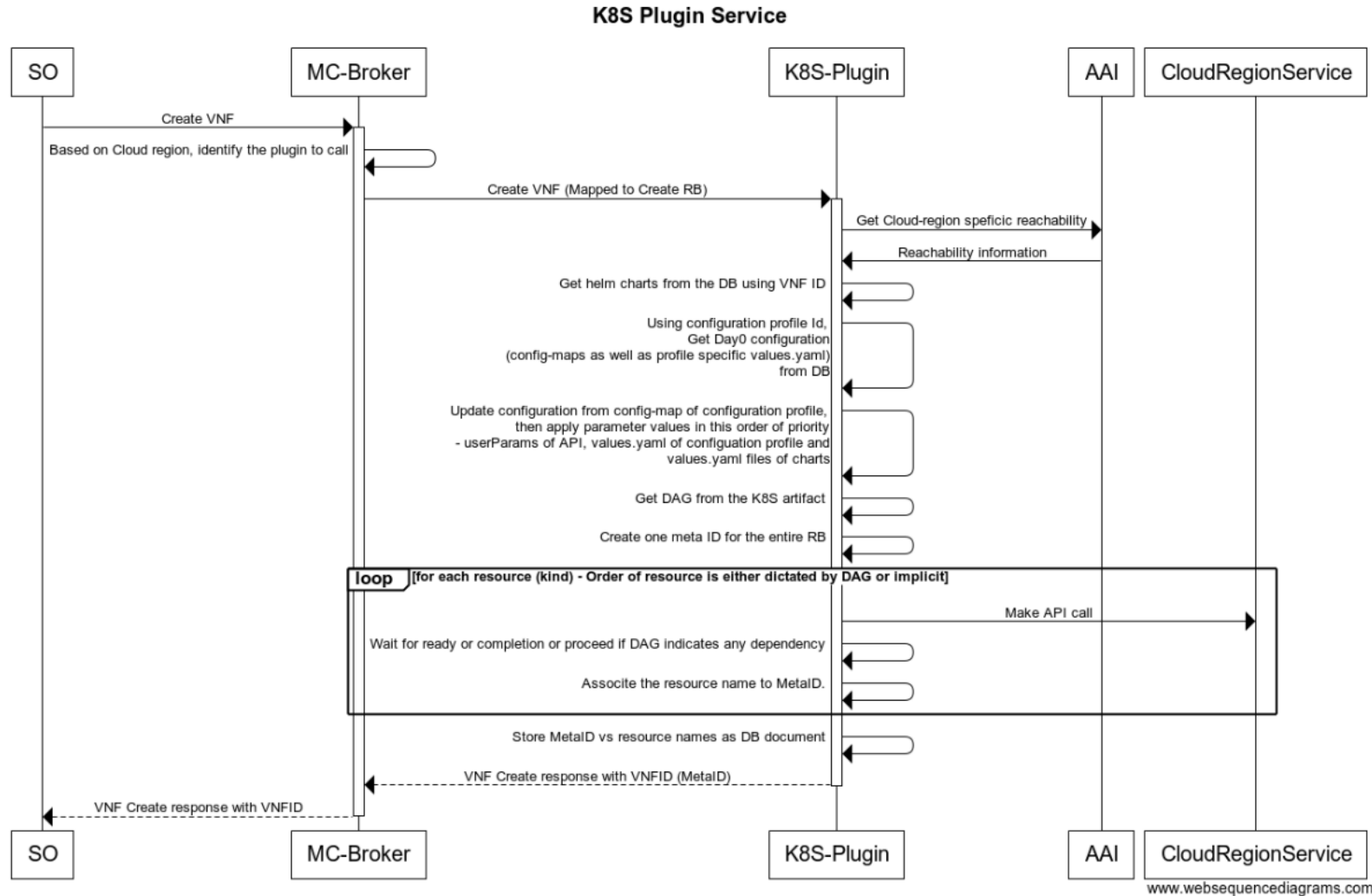
## Installation:

```
$ mkdir -p /opt/{kubeconfig,consul/config}
$ cp $HOME/.kube/config /opt/kubeconfig/krd
$ export KUBE_CONFIG_DIR=/opt/kubeconfig
$ git clone https://git.onap.org/multicloud/k8s/
$ cd deployments
$ ./build.sh
$ docker-compose up -d
```

## API

```
POST - /v1/vnf_instances/
GET - /v1/vnf_instances/{cloudRegionID}/{namespace}
DELETE - /v1/vnf_instances/{cloudRegionID}/{namespace}/{externalVNFDID}
GET - /v1/vnf_instances/{cloudRegionID}/{namespace}/{externalVNFDID}
```

# Create VNF



# Kubernetes Reference Deployment (KRD)

Internal tool that provisions a Kubernetes cluster with components satisfies the requirements of MultiCloud/k8s plugin.

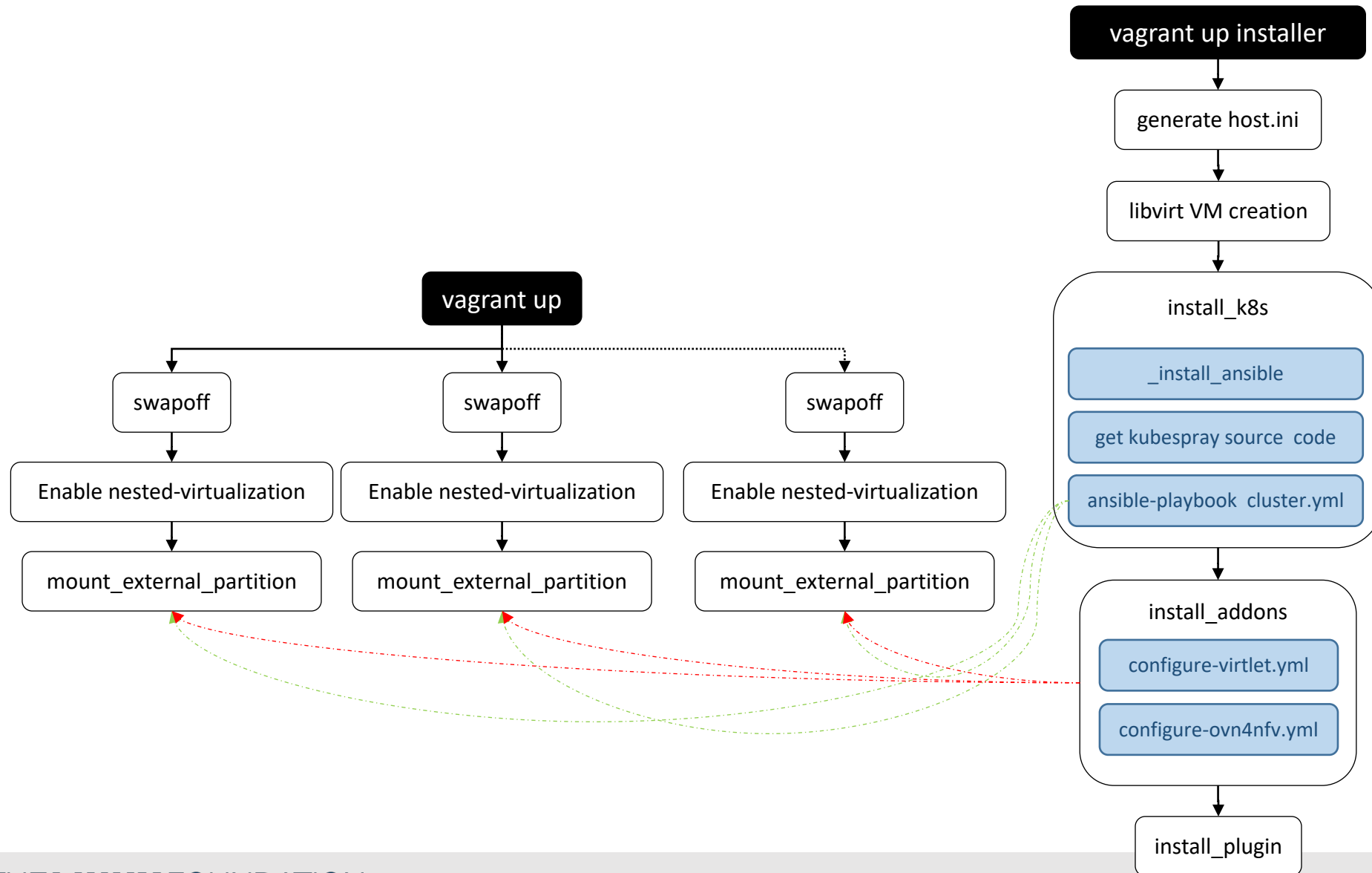


## Components

Name	Description	Source	Status
Kubernetes	Base Kubernetes deployment	<a href="#">kubespray</a>	Done
ovn-kubernetes	Integrates Opensource Virtual Networking	<a href="#">configure-ovn-kubernetes.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

<https://git.onap.org/multicloud/k8s/tree/vagrant/>

# KRD provisioning process(VMs)



# KRD provisioning process(Bare-metal)

The *vagrant/aio.sh* script contains the bash instructions for provisioning an All-in-One Kubernetes deployment in a Bare-Metal server

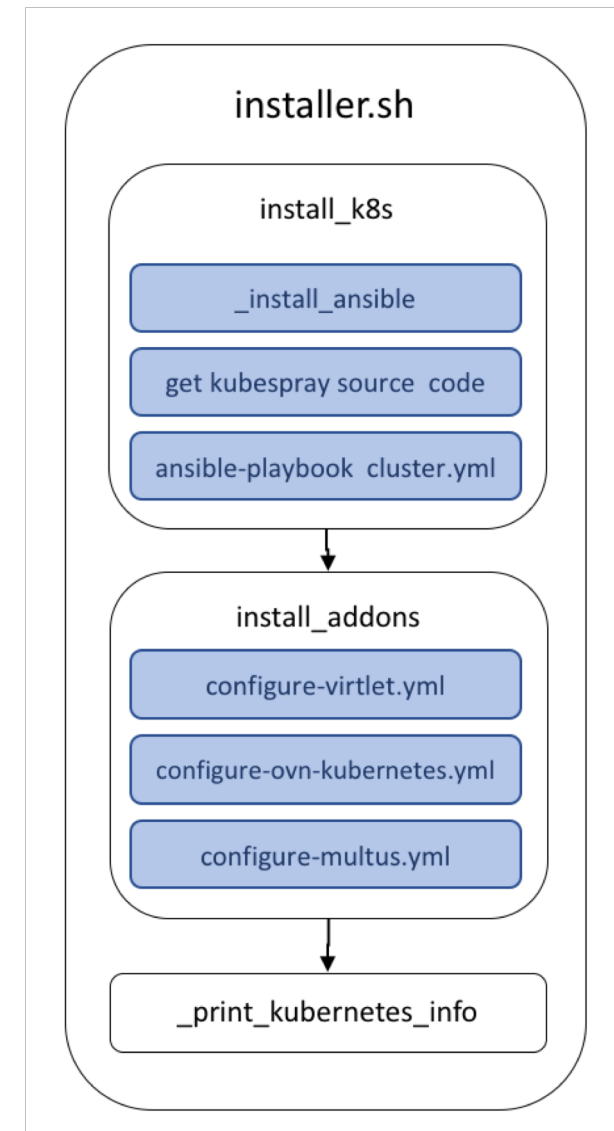
## Hardware Requirements

Concept	Amount
CPUs	8
Memory	32GB
Hard Disk	150GB

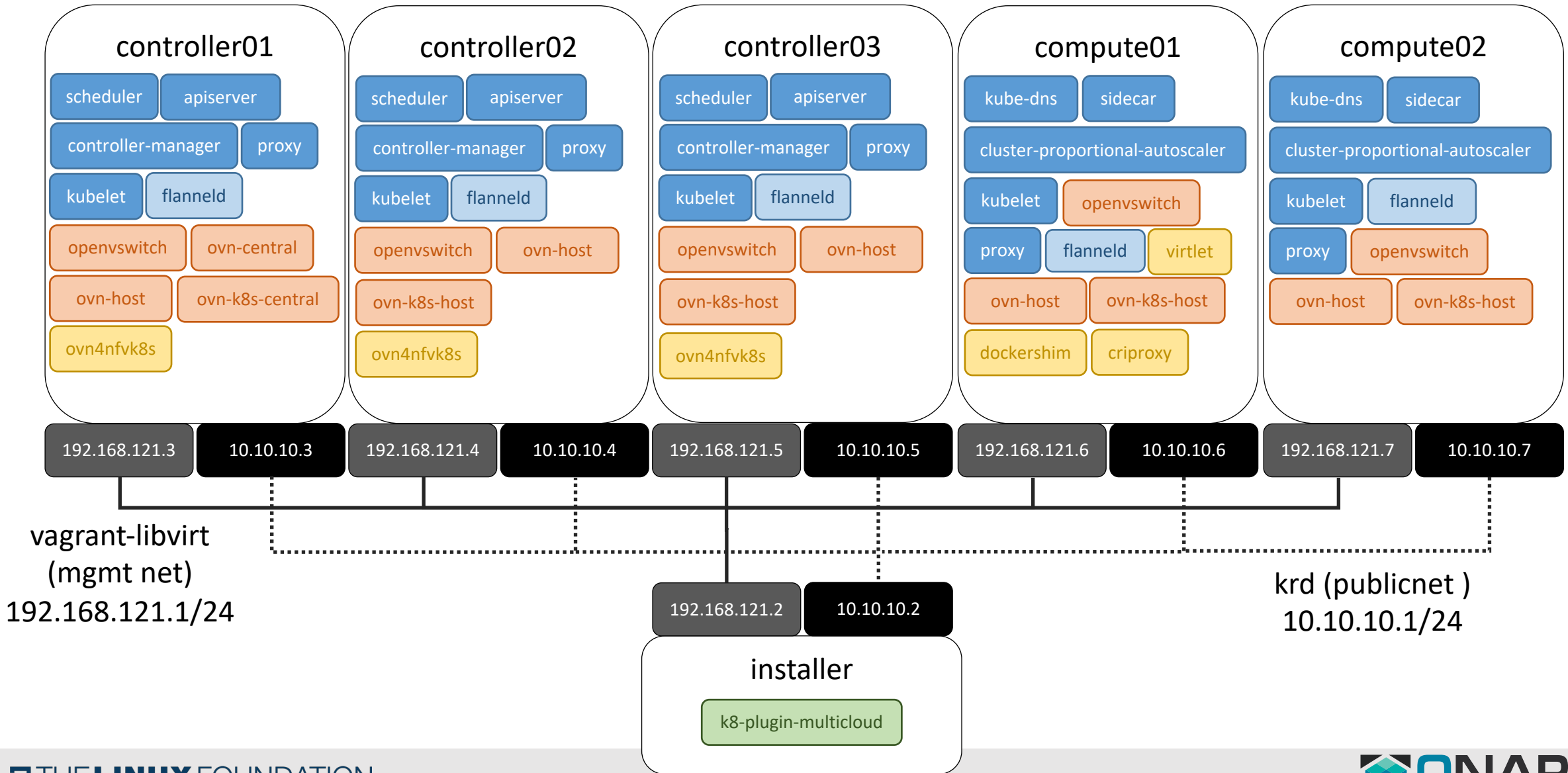
## Software Requirements

- Ubuntu Server 16.04 LTS

[https://docs.onap.org/en/latest/submodules/multicloud/k8s.git/docs/bare\\_metal\\_provisioning.html](https://docs.onap.org/en/latest/submodules/multicloud/k8s.git/docs/bare_metal_provisioning.html)



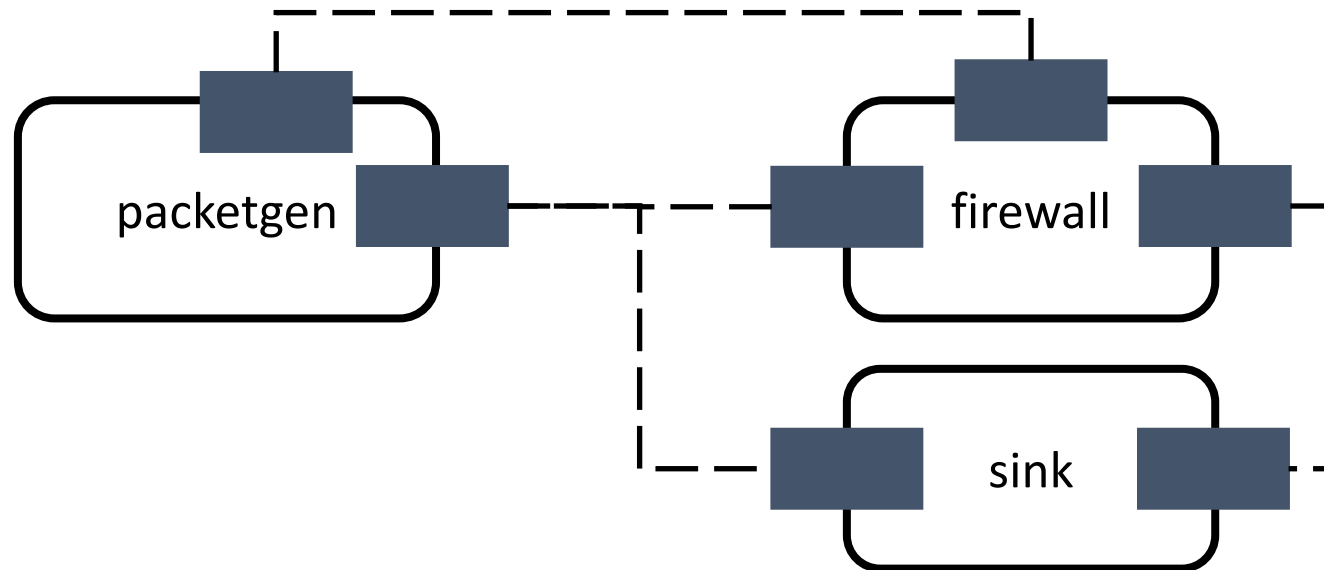
# config/default.yml (Pod Description File)



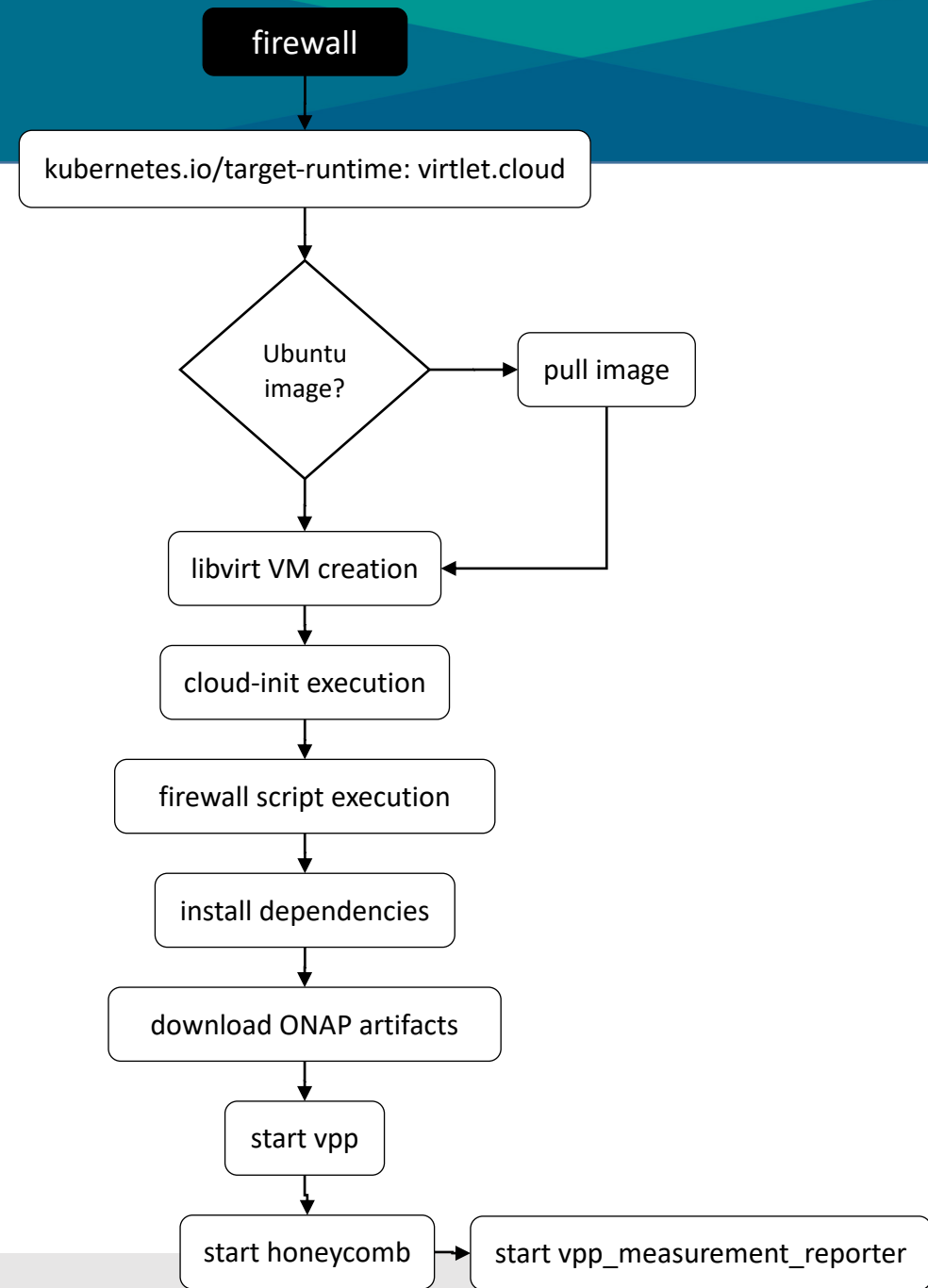
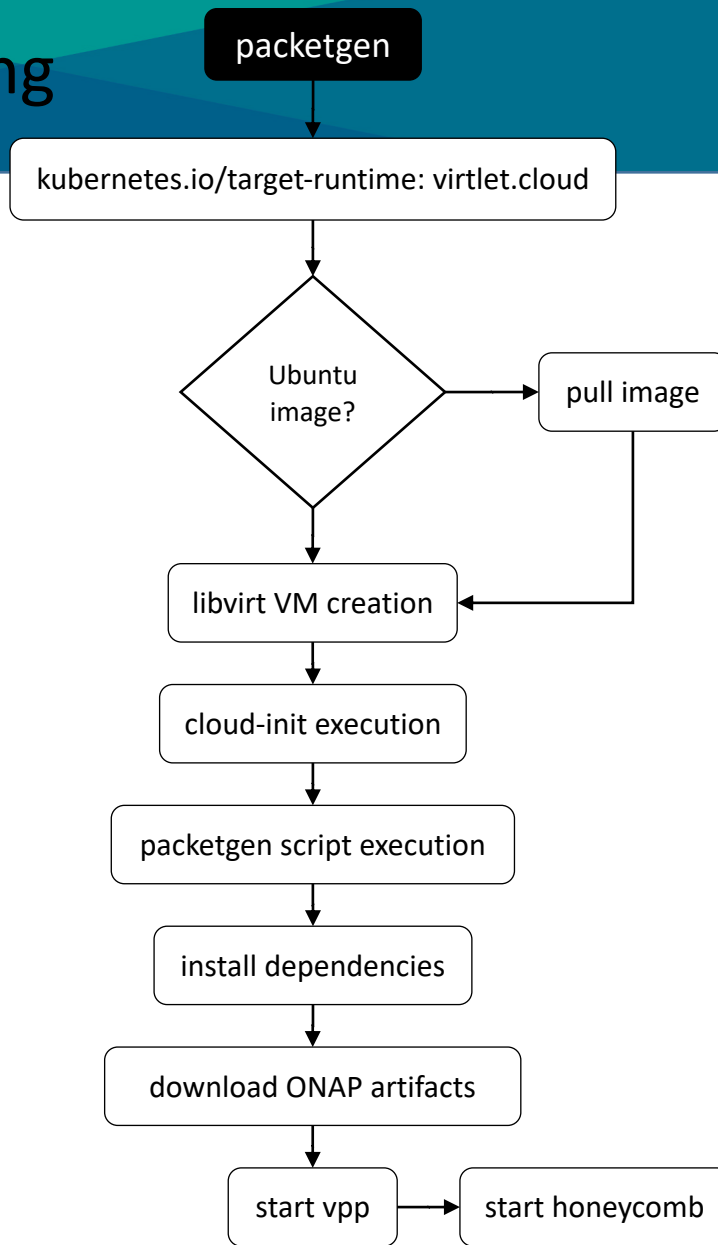
# Hybrid vFW ONAP use case

It is composed of three virtual functions (VFs):

- **Packet generator:** Sends packets to the traffic sink through the firewall. This includes a script that periodically generates different volumes of traffic.
- **Firewall:** Reports the volume of traffic passing through to the ONAP DCAE collector.
- **Traffic sink:** Displays the traffic volume that lands at the sink using the link <http://192.168.20.250:667> through your browser and enable automatic page refresh by clicking the "Off" button. You can see the traffic volume in the charts.



# VMs Provisioning

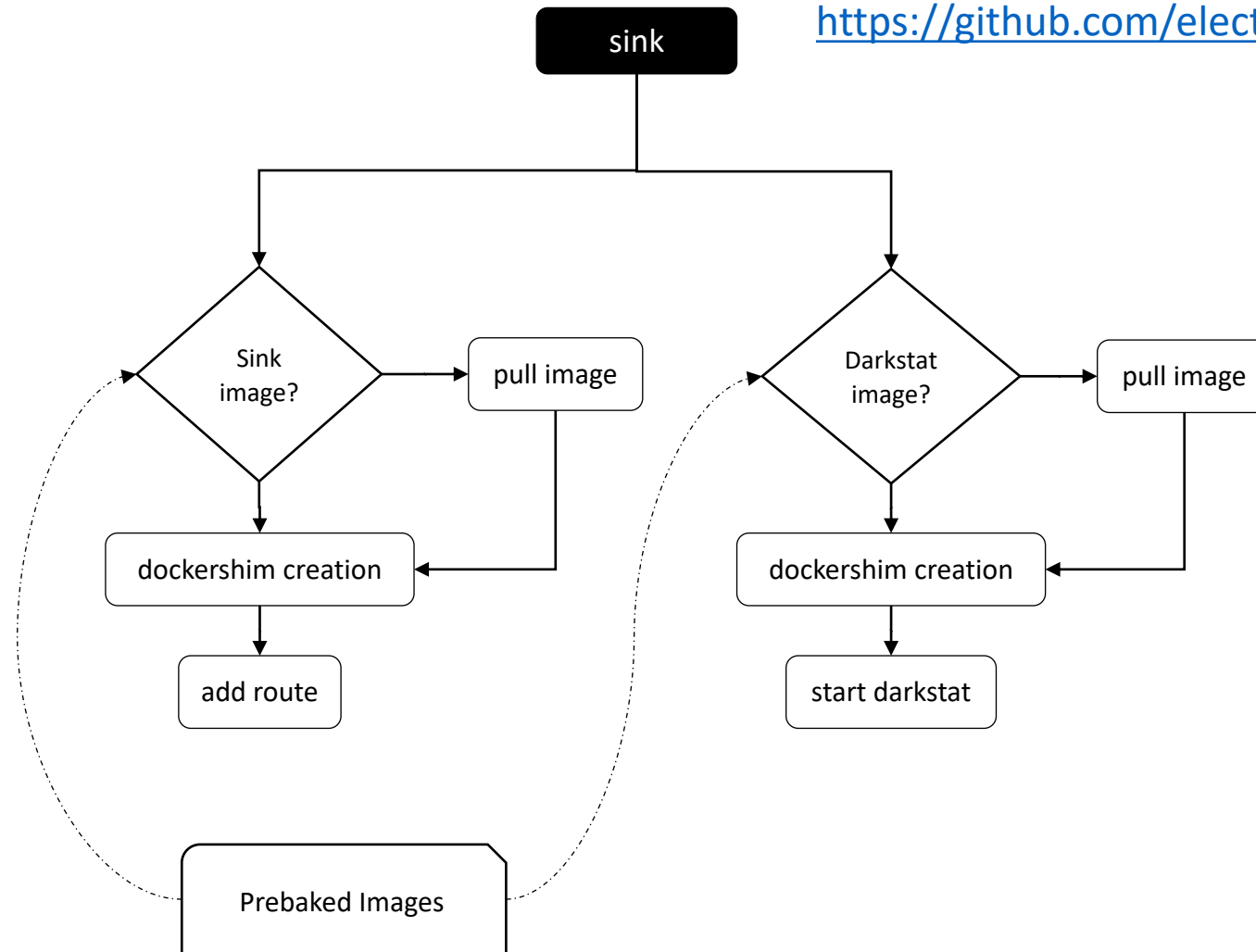


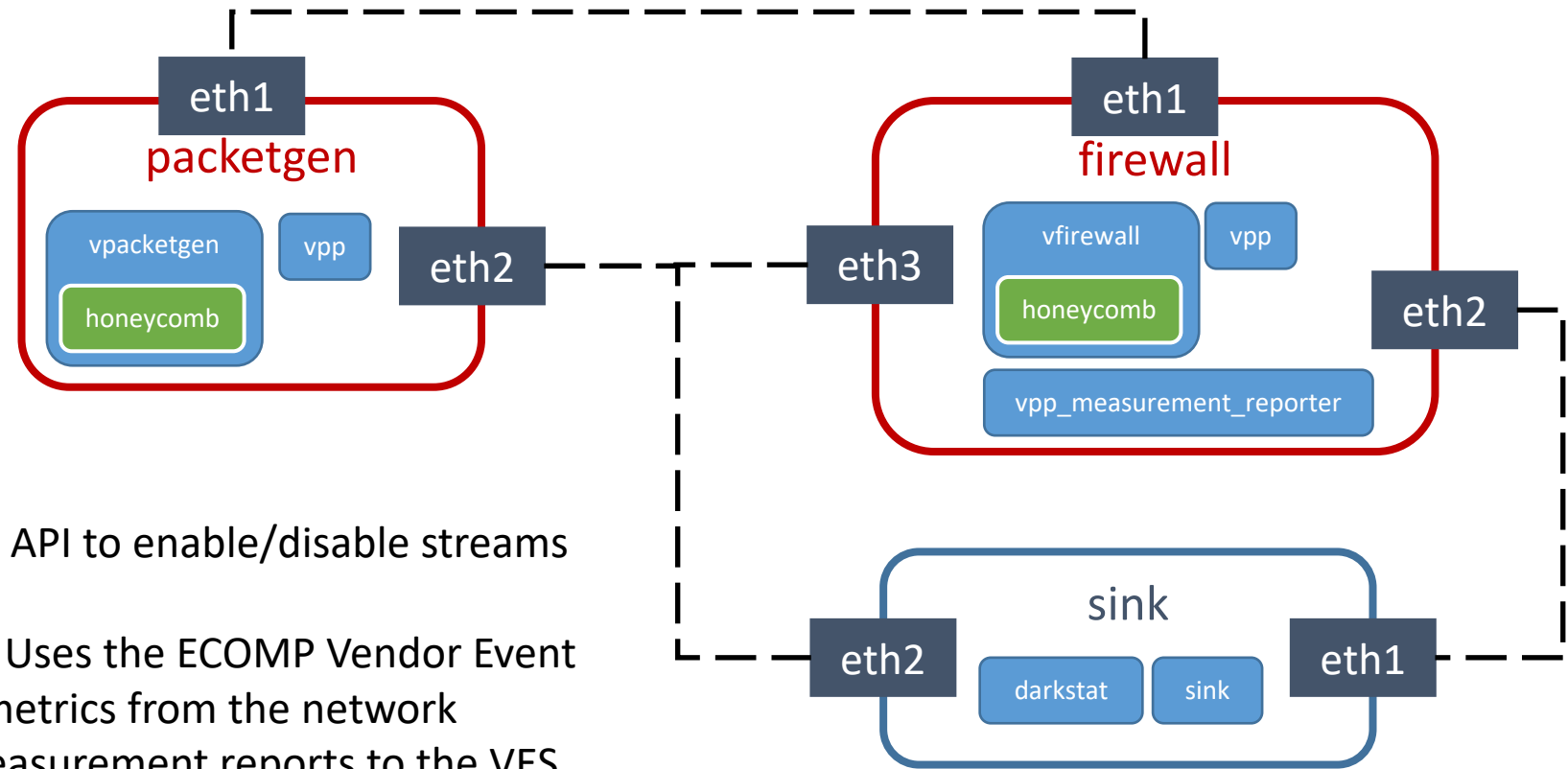
<https://github.com/electrocucaracha/vFW-demo>



# Containers Provisioning

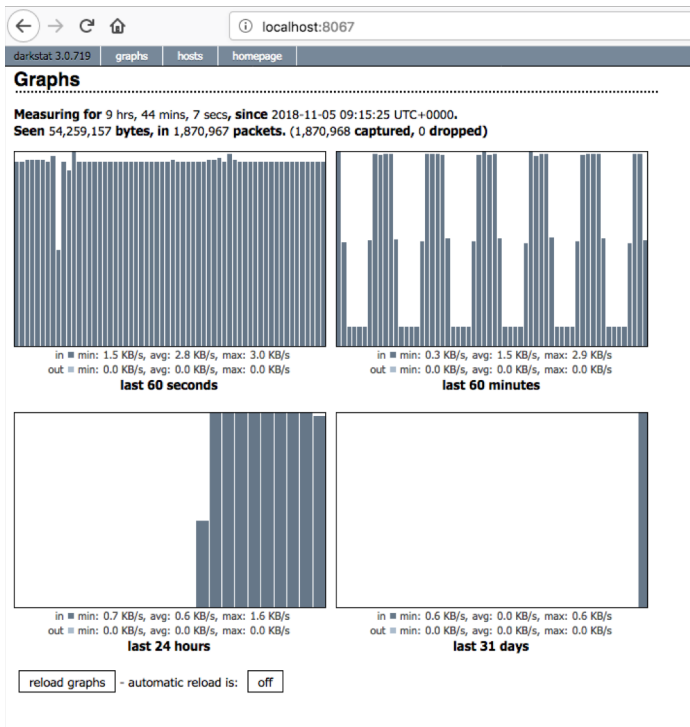
<https://github.com/electrocucaracha/cFW-demo>





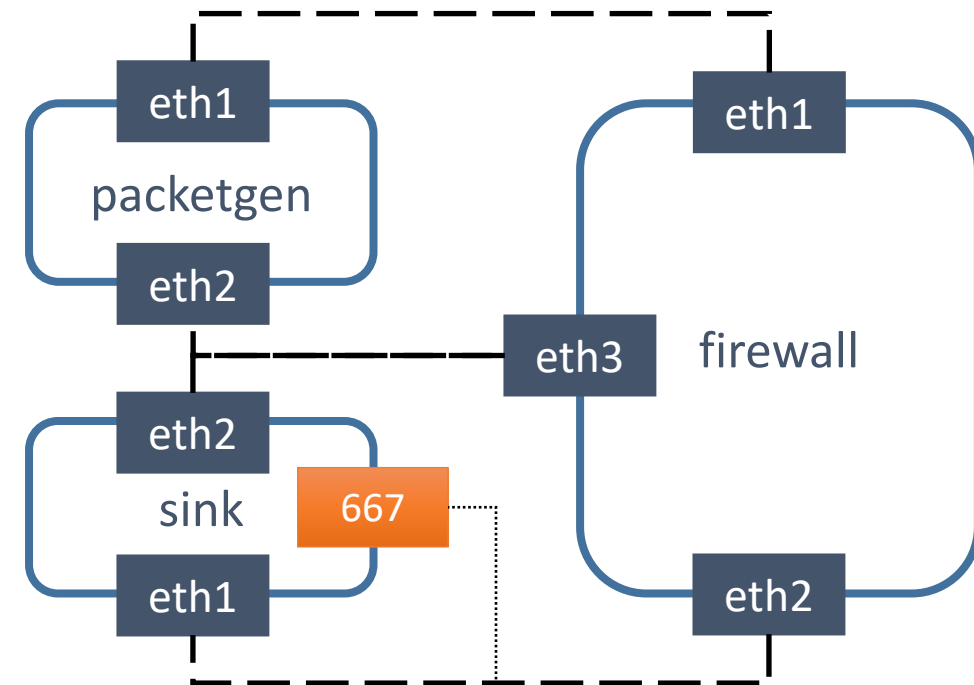
- `honeycomb` – Provides RESTful API to enable/disable streams on the VPP's packet generator.
- `vpp_measurement_reporter` – Uses the ECOMP Vendor Event Listener Library (VES) to read metrics from the network interface and send periodic measurement reports to the VES collector in DCAE.
- `darkstat` – Captures network traffic, calculates statistics about usage, and serves report over HTTP.

## localhost



X.X.X.X

## kubernetes



10.10.10.3

30667

172.25.103.9

8067

# Goals continue to be

- No changes in application containers.
- Minimal or no changes to Helm charts provided by vendors (Almost every micro service project in open source are providing Helm charts, but many assume that the solution is brought up manually)
  - Some observations & Guidelines:
    - Configuration files are hardcoded. Since, we expect to support multiple deployments using same Helm templates, configuration file names need to be part of the values file.
    - In some cases, configuration data is embedded in Helm charts as part of configMap. Need to make it as a separate configuration file and refer the content by file name macro.
    - In some cases, generic system configuration and application configuration combined in a configuration file. It is good to separate them out as application configuration may get changed (non-immutable containers).
- Support for
  - Immutable containers (Day2 configuration requires new container bring up)
  - Containers that get Day2 configuration from management systems.

# Feature contributors

## Contributing Companies

- AT&T, Intel, Ericsson, Jabil, VMWare, Nokia, Ericsson, AMDOCS (as part of Azure), Infosys (as part of AWS plugin)



# ONAP

OPEN NETWORK AUTOMATION PLATFORM