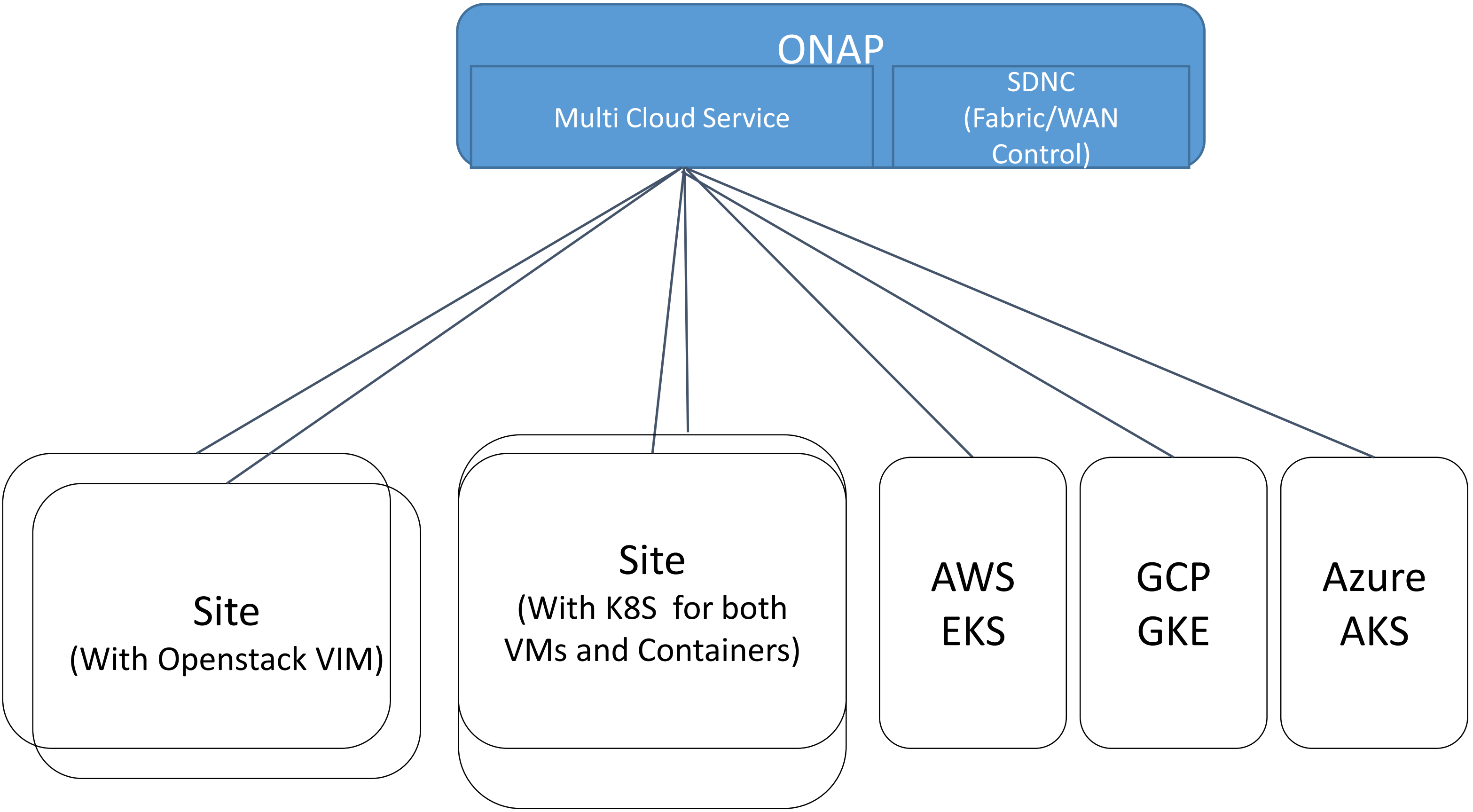
ONAP
OPEN NETWORK AUTOMATION PLATFORM

Kubernetes Support for VM & Container based VNFs R3 Update and Potential R4 items

Contact : Srini Addepalli or Victor (Intel) for questions/clarifications or to provide feedback

ONAP – Support for K8S based Sites

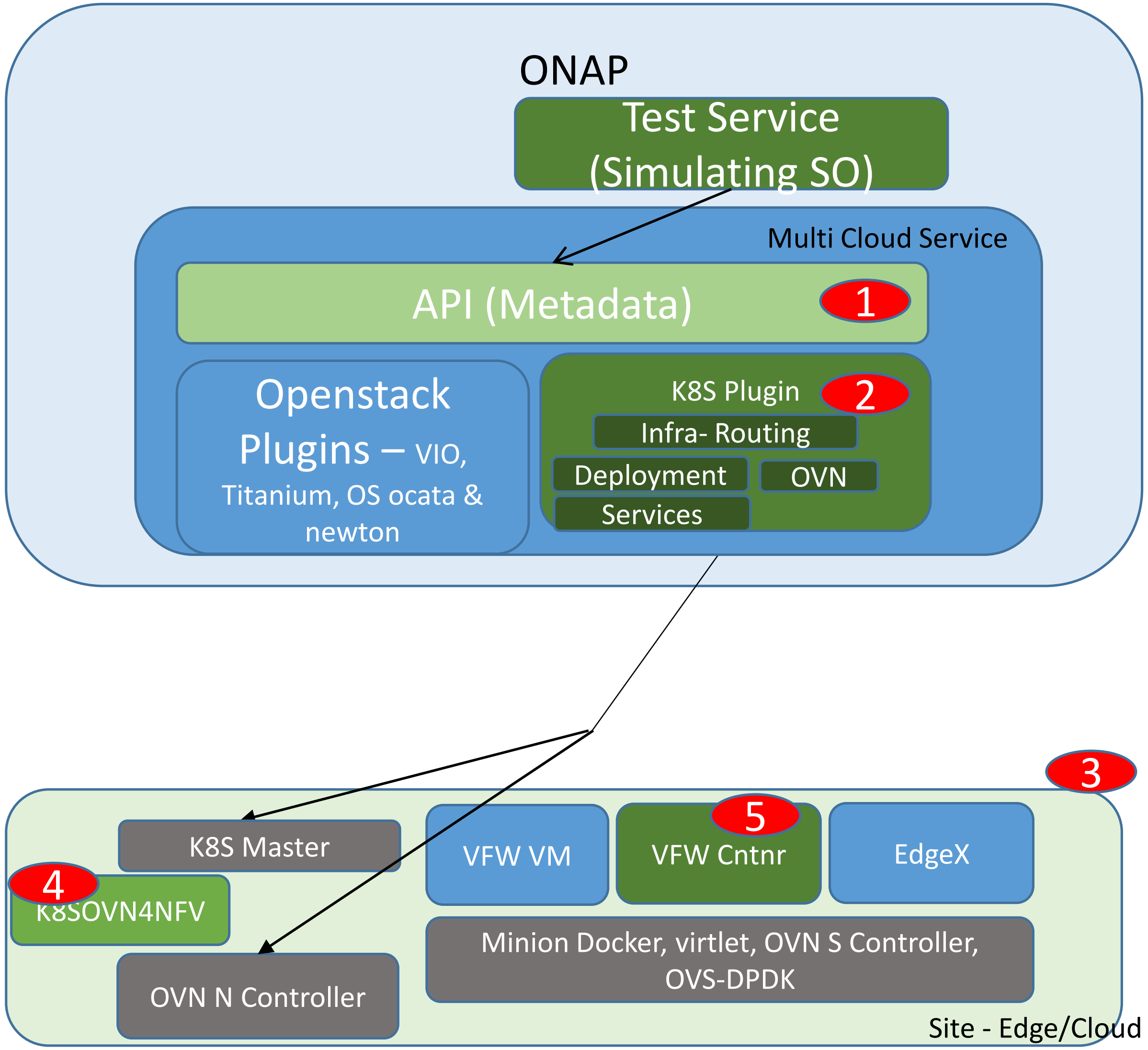


ONAP
Multi Cloud Service
SDNC (Fabric/WAN Control)
Site (With Openstack VIM)
Site (With K8S for both VMs and Containers)
AWS EKS
GCP GKE
Azure AKS

- Current support as in R2: Openstack based remote Clouds, Support multiple Openstack variations – Windriver Titanium, VMWare VIO, Native Newton, Ocata. Only VM based VNFs.

- Goals for R3 and R4
  - Support containerized workloads
  - Support containerized VNFs
  - Support both VMs and containers on same compute nodes. (Bare-metal deployment)
  - Support for multiple virtual networks
  - Support for dynamic creation of Virtual networks
  - Support public cloud CaaS such as AWS EKS, GCP GKE and Azure AKS (Only containers, not VMs)

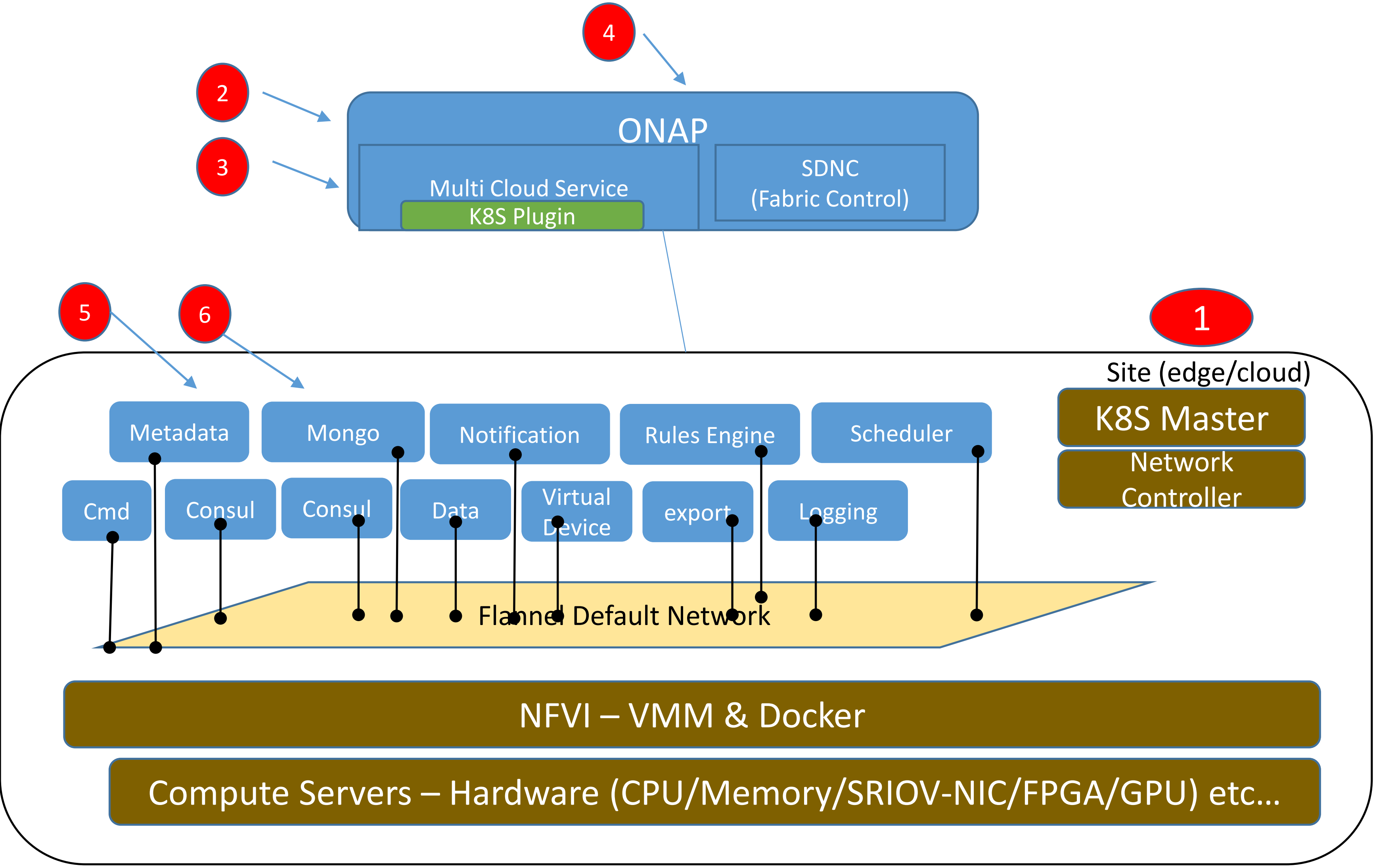
R3 Work Items



R3 work items are limited to Multi-Cloud Service and below.

1. Uniform API across cloud technologies (HEAT, K8S, Azure etc..)
2. K8S Multi-Cloud Service plugin
   - Support for deployment and services.
   - K8S yaml artifacts
   - Networking – OVN, flannel and Multus (Create/Delete VNs, Distributed Router, Gateways, SNAT in Gateway)
3. Kubernetes Reference Deployment
   - Installation of software & configuration to make K8S based sites.
   - Additional of virtlet, Multus, OVN and flannel.
4. K8S-OVN4NFV (OPNFV project, visualized as part of ONAP work)
   - Support for multiple virtual networks
   - Support for dynamic creation/deletion of virtual networks
5. VFW as containers
   - Firewall, generator and sink in containerized format

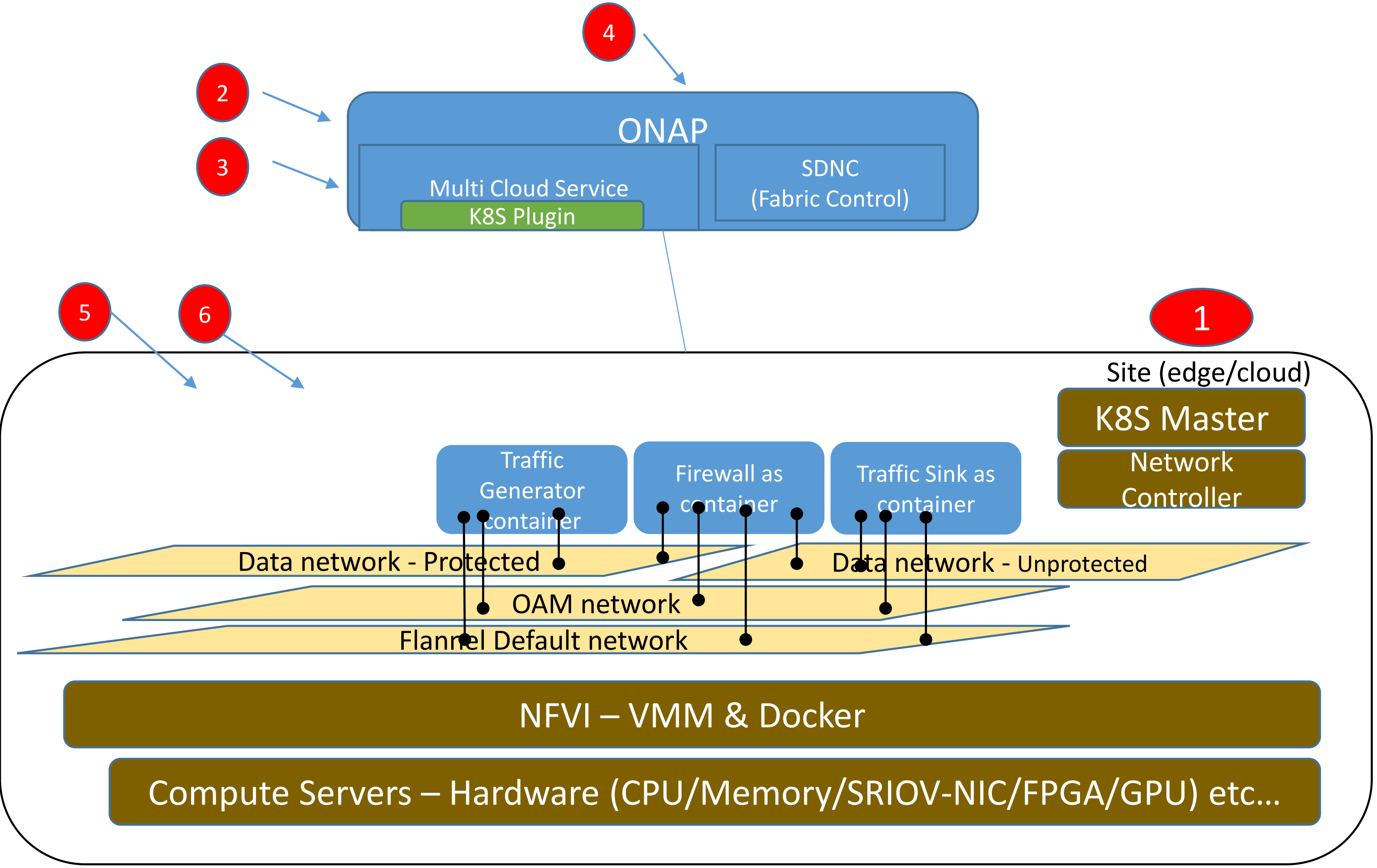
R3 Scenarios – EdgeX deployment



1. One time: Prepare K8S based site using KRD (if it does not exist)
2. One time: Register the K8S Site in ONAP by adding Kubeconfig file in ONAP (if the site is not added earilier)
3. EdgeX onboarding: Copy EdgeX deployment and service yaml files in Multi-Cloud
4. Instantiate EdgeX (by calling Multi-Cloud Service API) via postman or via script
5. Check if all EdgeX containers are successful brought up on the site (using K8S utilities on the site)
6. Basic EdgeX testing to ensure that functionality also works
   - Use consul dashboard to check the services and their status

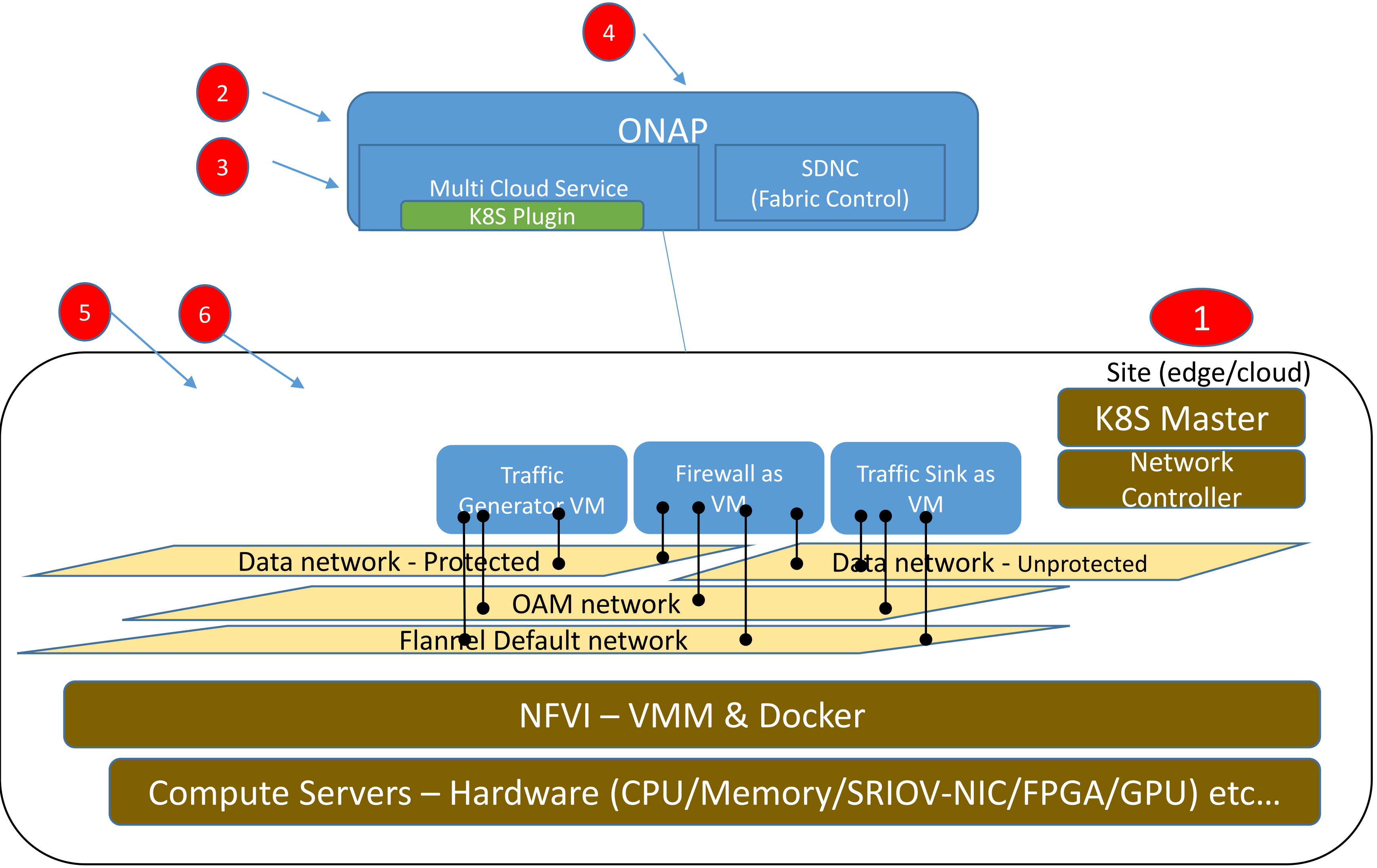
Repeat step 4 to 6 by bringing second instance of EdgeX on a different namespace.

R3 Scenarios – vFirewall scenario



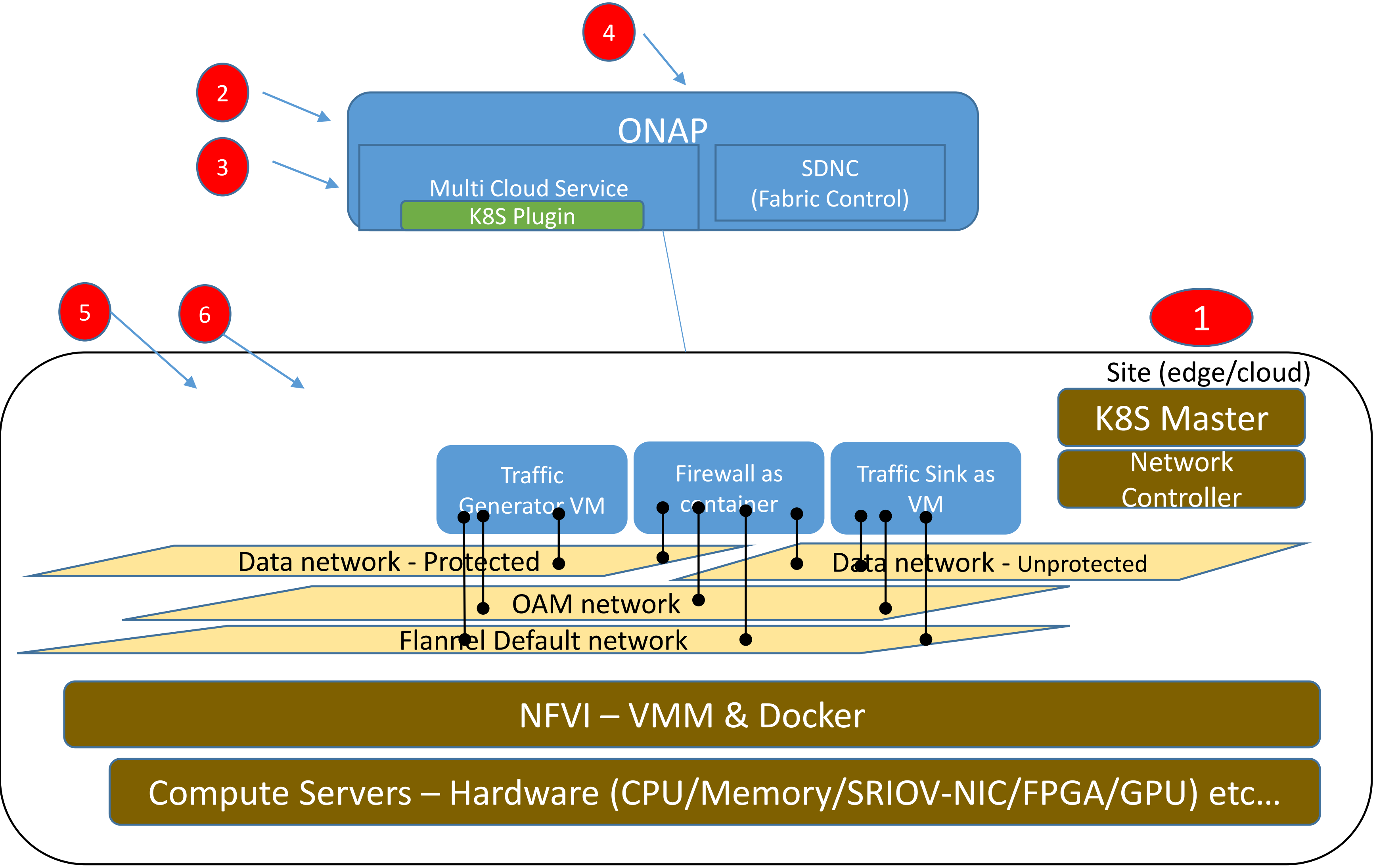
1. One time: Prepare K8S based site using KRD (if it does not exist)
2. One time: Register the K8S Site in ONAP by adding Kubeconfig file in ONAP (if the site is not added earilier)
3. vFirewall onboarding: Create deployment and service yaml file and put them in location expected by K8S plugin
4. Instantiate vFirewall (by calling Multi-Cloud Service API) via postman or via script
5. Check if all firewall containers are successful brought up on the site (using tools) and also ensure that three additional virtual networks are created. Also ensure that firewall belongs in all data networks. Ensure that generator and sink belong to different data networks.
6. Basic firewall testing to ensure that functionality also works
   - Check the sink dashboard to ensure that right packet streams are received by sink.

R3 Scenarios – vFirewall scenario (But as VMs)



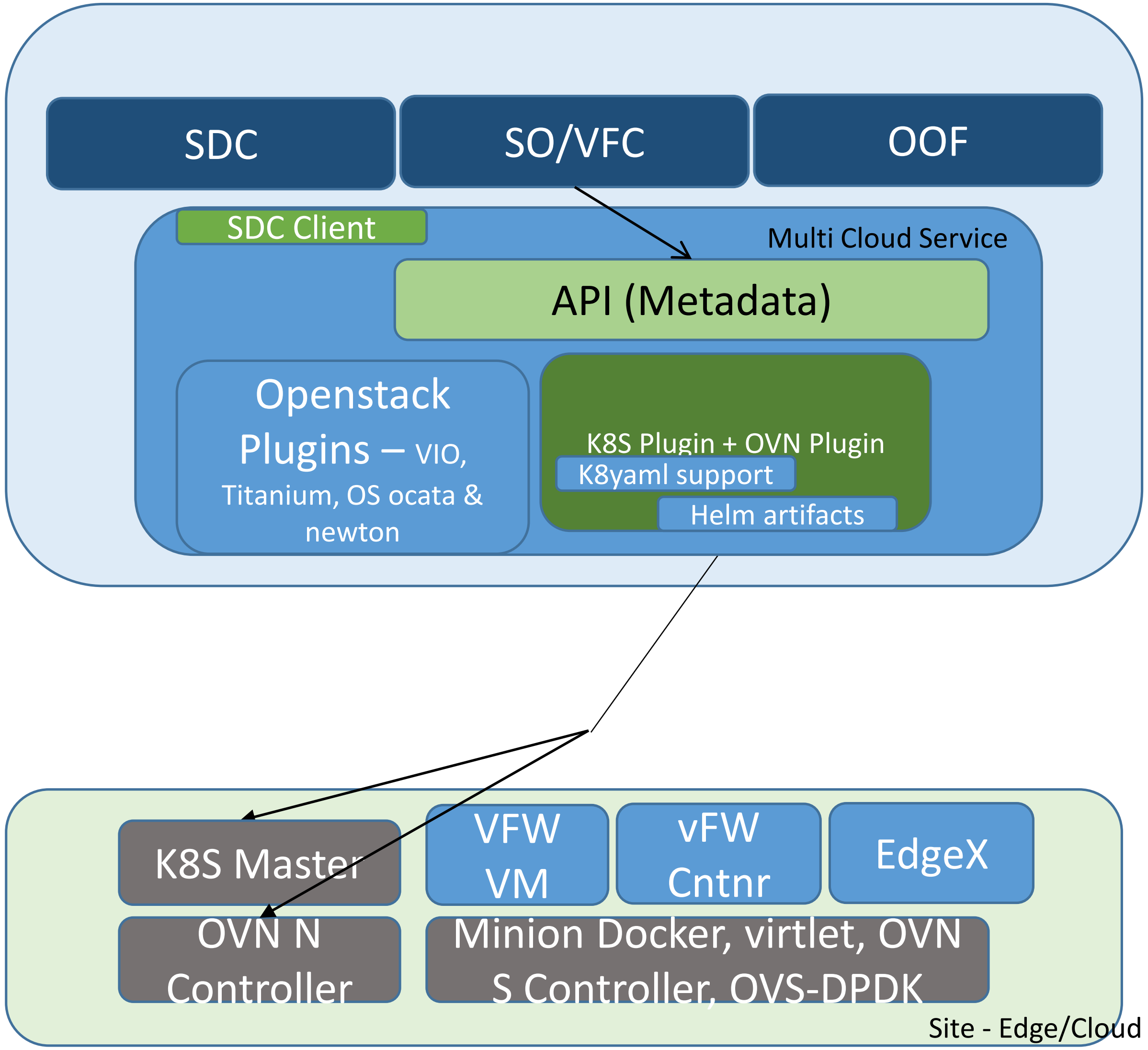
1. One time: Prepare K8S based site using KRD (if it does not exist)
2. One time: Register the K8S Site in ONAP by adding Kubeconfig file in ONAP (if the site is not added earilier)
3. vFirewall onboarding: Create deployment and service yaml files and put them in location expected by K8S plugin
4. Instantiate vFirewall (by calling Multi-Cloud Service API) via postman or via script
5. Check if all firewall containers are successful brought up on the site (using tools) and also ensure that three additional virtual networks are created. Also ensure that firewall belongs in all data networks. Ensure that generator and sink belong to different data networks.
6. Basic firewall testing to ensure that functionality also works
   - Check the sink dashboard to ensure that right packet streams are received by sink.

R3 Scenarios – vFirewall scenario (Hybrid – VMs and containers)



1. One time: Prepare K8S based site using KRD (if it does not exist)
2. One time: Register the K8S Site in ONAP by adding Kubeconfig file in ONAP (if the site is not added earilier)
3. vFirewall onboarding: Create deployment and service yaml files and put them in location expected by K8S plugin
4. Instantiate vFirewall (by calling Multi-Cloud Service API) via postman or via script
5. Check if all firewall containers are successful brought up on the site (using tools) and also ensure that three additional virtual networks are created. Also ensure that firewall belongs in all data networks. Ensure that generator and sink belong to different data networks.
6. Basic firewall testing to ensure that functionality also works
   - Check the sink dashboard to ensure that right packet streams are received by sink.

Phase 2 Scope - Focus on SDC+SO and Multi Cloud



- Phase2 (Beyond Casablanca)
  - Ensure that K8S Plugin and Azure plugin expose same API.
  - Integrate with SO/VFC – Create generic workflows (based on HEAT flows, but extend this generically to include any cloud technology specific artifacts) – Already being done in R3 itself
  - Support for Helm based artifacts
  - Any carryovers from Phase 1
  - Leveraging load balancers in the site
  - Endpoint creation for data interface IP addreses
  - Support CaaS (IBM, GCP, AWS EKS etc…)
  - Support for OpenShift, VMWare Kubernetes Service (Mostly testing?)
  - HPA functionality & Create compute flavors
  - OOF based placement decisions
  - cAdvisor in the Minion nodes (Kubelet)
  - Node exporter in edge clouds
  - Support for all ‘kinds’ including service mesh based application support – Persistent volumes, Stateful sets, ISTIO mesh etc..
  - SRIOV-NIC support and any other networking (TF?)
- End-to-End functionality:
  - SDC : Ability to add K8S artifacts in CSAR.
  - SO : Make the SO independent of Cloud technologies & also HEAT independent.
  - Multi-Cloud getting hold of artifacts from SDC
  - How to create E2E Service (TBD) – VNF SDK & BPMN?

ONAP

OPEN NETWORK AUTOMATION PLATFORM