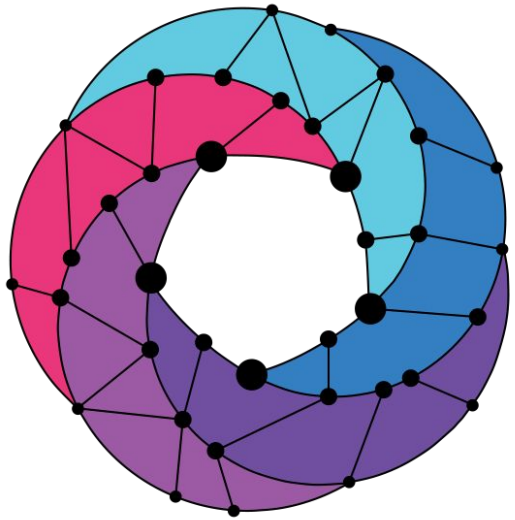


Open Source
MANO

OSM #13 Hackfest

Deployment Walkthrough

ETSI

Agenda

1. Prepare the environment
2. Deployment overview
3. Magma Orchestrator
4. Magma Access GW
5. SRS LTE

Open Source
MANO

Prepare the environment

ETSI

VIM Creation

```
$ osm vim-create --name aws-site --account_type aws \
  --auth_url https://aws.amazon.com --user <Access Key ID> \
  --password <Secret Access Key> --tenant <EKS VPC> \
  --description "AWS site, with your user" \
  --config '{key_pair: <team key>, region_name: <region>, flavor_info: \
    "{t2.nano: {cpus: 1, disk: 100, ram: 512}, \
      t2.micro: {cpus: 1, disk: 100, ram: 1024}, \
      t2.small: {cpus: 1, disk: 100, ram: 2048}, \
      t2.medium: {cpus: 2, disk: 100, ram: 4096}, \
      t2.large: {cpus: 2, disk: 100, ram: 8192}}"}'
```

K8s cluster addition

Copy `kube_config` file in case is not in the VM

```
$ osm k8scluster-add --creds ./kube_config \
      --version 1 \
      --vim aws-site \
      --k8s-nets "{k8s_net1: <EKS VPC>}" \
      --description "Base K8scluster" eks-cluster
```

Check your environment

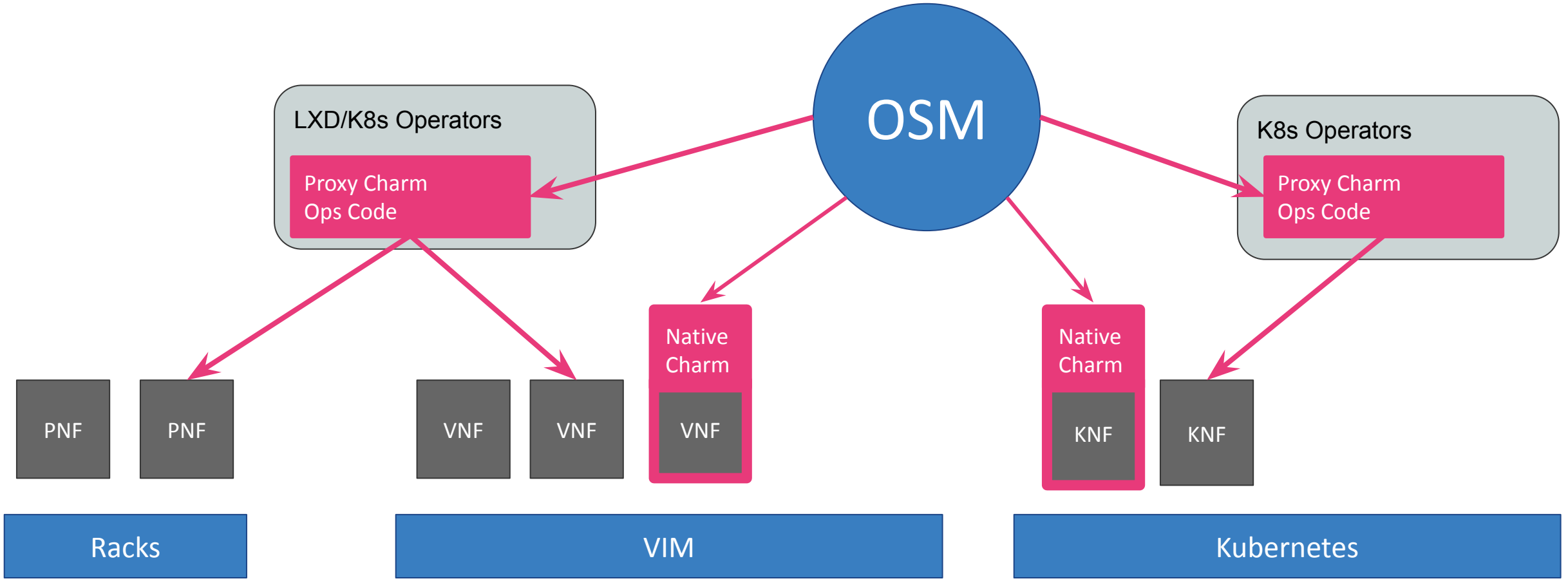
```
$ osm vim-list
$ osm vim-how <VIM ID>
$ osm k8scluster-list
$ osm k8scluster-show <K8s cluster ID>
```

Open Source
MANO

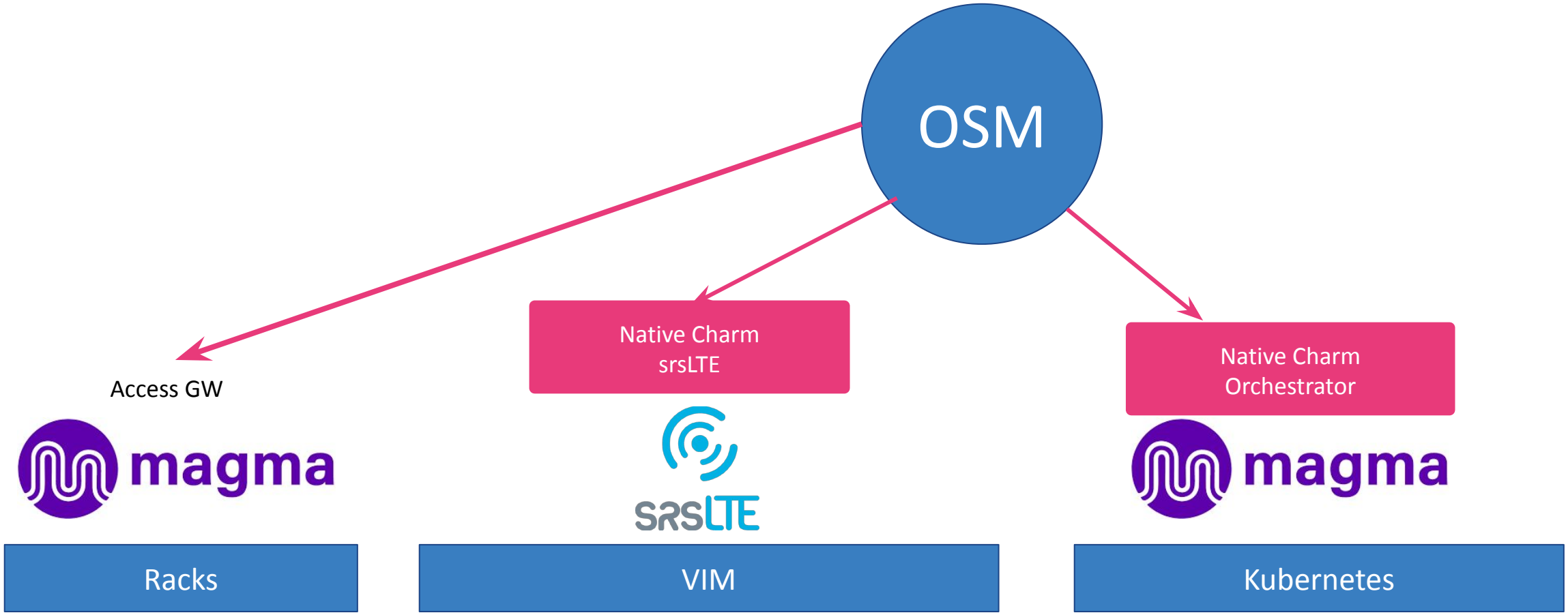
Deployment overview

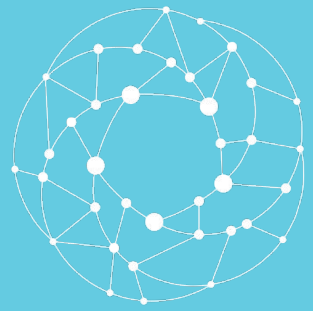
ETSI

Reality is messy and mixed



Reality is messy and mixed



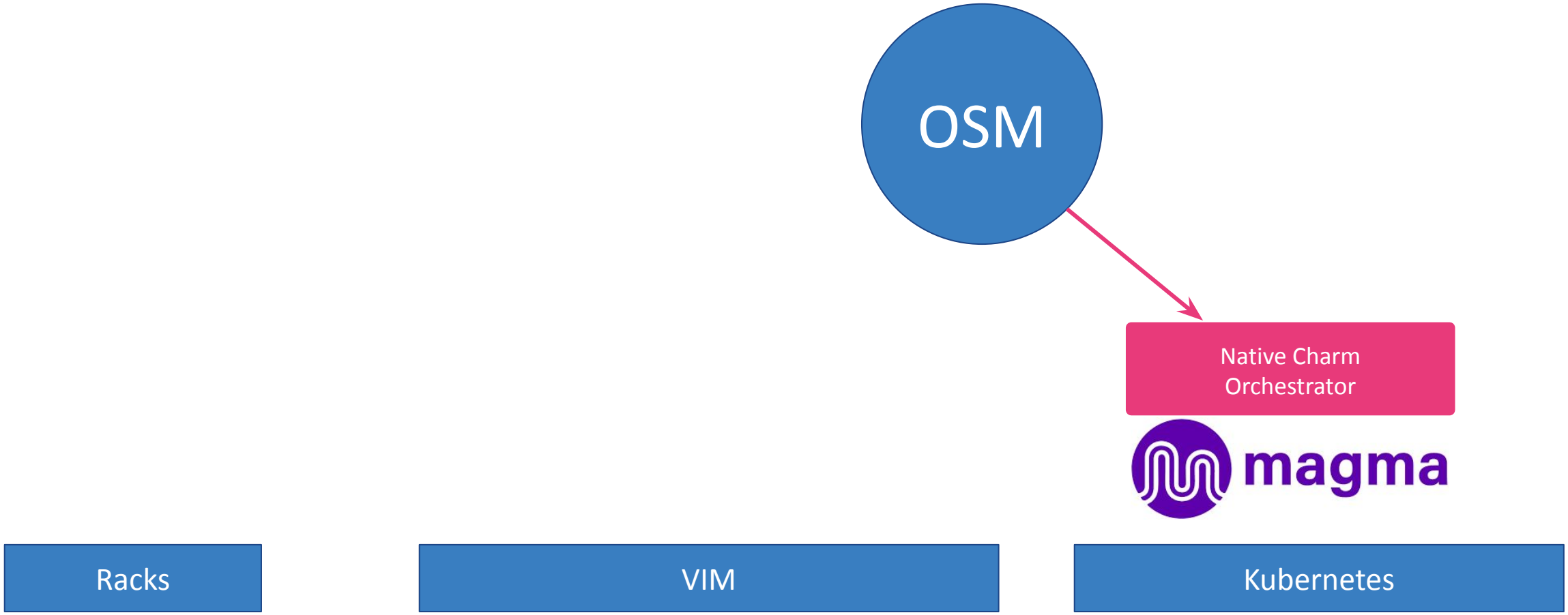


Open Source
MANO

Magma Orchestrator

ETSI

Magma Orchestrator



What are juju bundles?

Open Source MANO

- Bundles are collections of charms.
- They represent an entire model, rather than a single application.
- From a technical point of view, a bundle is a YAML file.

Charms + Config + Relations ⟶

```
bundle: kubernetes
applications:
  mariadb-k8s:
    charm: cs:~juju/mariadb-k8s-2
    scale: 1
  mediawiki-k8s:
    charm: cs:~juju/mediawiki-k8s-3
    scale: 1
    options:
      debug: true
relations:
- - mariadb-k8s:server
  - mediawiki-k8s:db
```

Referencing the juju-bundle (VNFd)


```
vnfd:
  [...]
  kdu:
  - name: magma-orc-kdu
    juju-bundle: bundle.yaml
```

juju-bundles/bundle.yaml (VNFd)

Open Source MANO

```
bundle: kubernetes
applications:
  nms-magmalte:
    charm: magma-nms-magmalte
  # ...
  orc8r-certifier:
    charm: magma-orc8r-certifier
    channel: edge
    scale: 1
    trust: true
    options:
      domain: team1.osmhackfest.com
  # ...
relations:
- - nms-magmalte:magma-orc8r-certifier
  - orc8r-certifier:magma-orc8r-certifier
```

Day-1 operations (VNFd)

```
vnfd:
  description: K8s container deployment of Magma Orchestrator
  df:
  - id: default-df
    lcm-operations-configuration:
      operate-vnf-op-config:
        day1-2:
          - id: magma-orc-kdu
            initial-config-primitive:
              - name: create-orchestrator-admin-user
              parameter:
                    - seq: 0
                      name: application-name
                      data-type: STRING
                      default-value: orc8r-orchestrator
                    - name: url
                      data-type: STRING
                      default-value: ''
                  [...]
```

Adding day-2 operations (VNFd)


```
vnfd:
  description: K8s container deployment of Magma Orchestrator
  df:
  - id: default-df
    lcm-operations-configuration:
      operate-vnf-op-config:
        day1-2:
          - id: magma-orc-kdu
            config-primitive:
              - name: get-admin-credentials
               parameter:
                    - name: application-name
                      data-type: STRING
                      default-value: nms-magmalte
                [...]
```

Onboarding to OSM

```
$ osm nfpkg-create magma_orc_cnf.tar.gz
$ osm nspkg-create magma_orc_ns.tar.gz
```

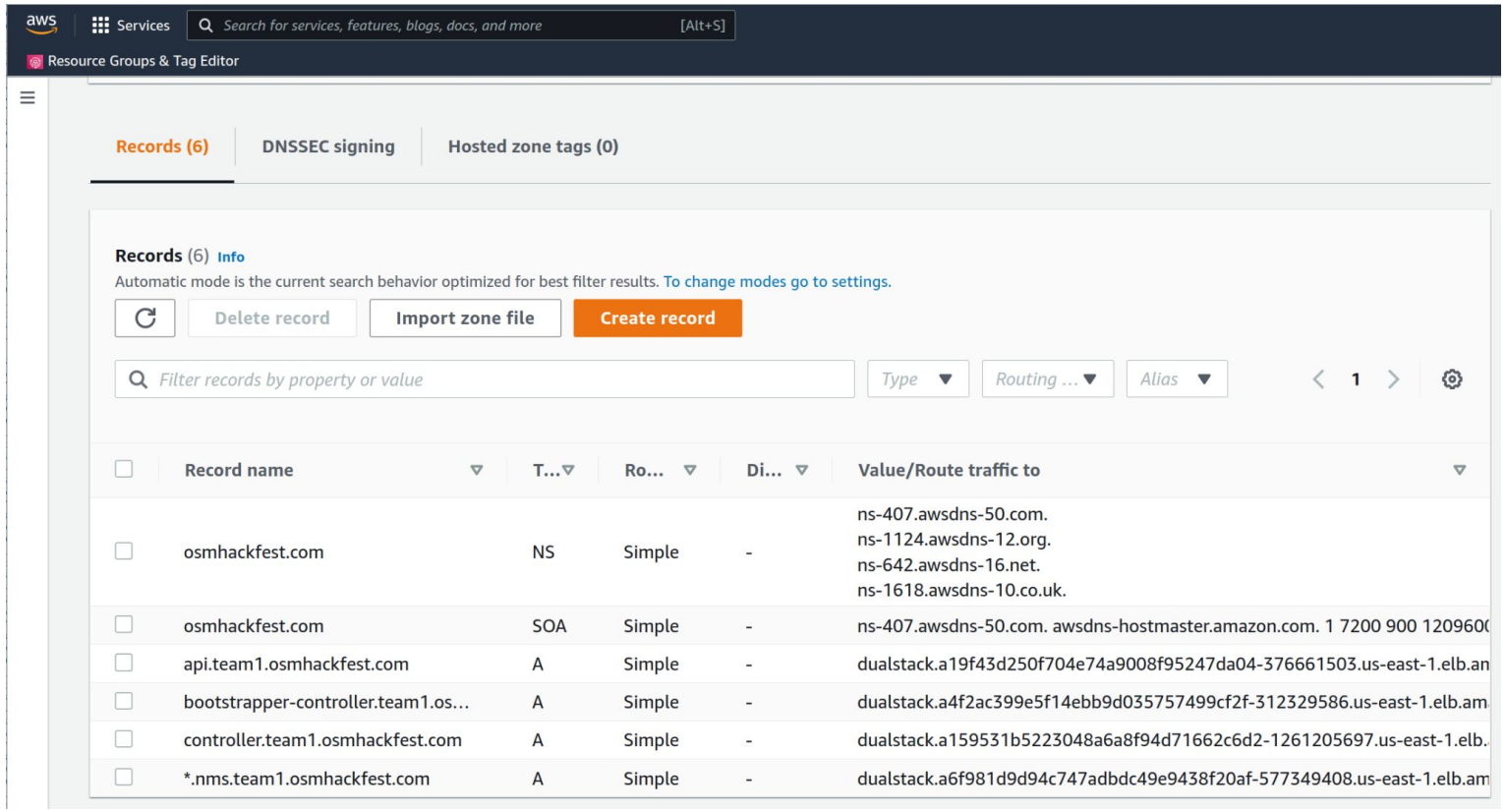
Deploy Magma Orchestrator

```
$ osm ns-create --ns_name magma_orc_cnf_ns \
                --nsd_name magma_orc_cnf_ns \
                --vim_account aws-site
```

Progress with osm ns-show

```
| admin-status          | "ENABLED"                                                                          |
| deploymentStatus      | null                                                                               |
| configurationStatus   | [                                                                                  |
|                       |     "status": "INSTALLING SW"                                                      |
| vcaStatus             | {                                                                                  |
|                       |         "meter_statuses": {},                                                      |
|                       |         "status": {                                                                |
|                       |           "status": "maintenance",                                                 |
|                       |             "agent_status": {                                                      |
|                       |               "status": "executing",                                               |
|                       |             "workload_status": {                                                   |
|                       |               "status": "maintenance",                                             |
|                       |         "agent_status": {                                                          |
|                       |           "status": "started",                                                     |
|                       |         "instance_status": {                                                       |
|                       |           "status": "running",                                                     |
|                       |         "modification_status": {                                                   |
|                       |           "status": "idle",                                                        |
|                       |         "meter-status": {                                                          |
|                       |         "model-status": {                                                          |
|                       |           "status": "available",                                                   |
| operational-status    | "init"                                                                             |
| config-status         | "init"                                                                             |
| detailed-status       | "Stage 2/5: deployment of KDUs, VMs and execution environments. 0/2. Deployed at VIM" |
|                       |         "vim_status": "DONE"                                                       |
|                       |         "vim_status": "DONE"                                                       |
|                       | revision_number: 175, 'router:external': true, segmentation_id: null,\n  shared: true, status: |
|                       |         "vim_status": "ACTIVE"                                                     |
|                       | 'router:external': false, segmentation_id: null, shared: false,\n  status: ACTIVE, subnets: |
|                       |         "vim_status": "ACTIVE"                                                     |
|                       |       "operational-status": "running"                                              |
|                       |         "operational-status": "init",                                              |
|                       |         "detailed-status": "10% Preparing libblockdev-part2 (amd64)",              |
|                       |         "VCA-status": "maintenance",                                               |
|                       |         "status": "running",                                                       |
|                       |         "status-time": "1614889418.2858396",                                       |
```

Configure route53



[How to configure route53]

Download certificates

Install *admin_operator.pfx* in your browser

```
$ juju scp -m $model-name --container="magma-orc8r-certifier" \
    orc8r-certifier/0:/var/opt/magma/certs/..data/admin_operator.pfx \
    admin_operator.pfx
Password: password123
```

Copy *rootCA.pem* in Magma AGW machine

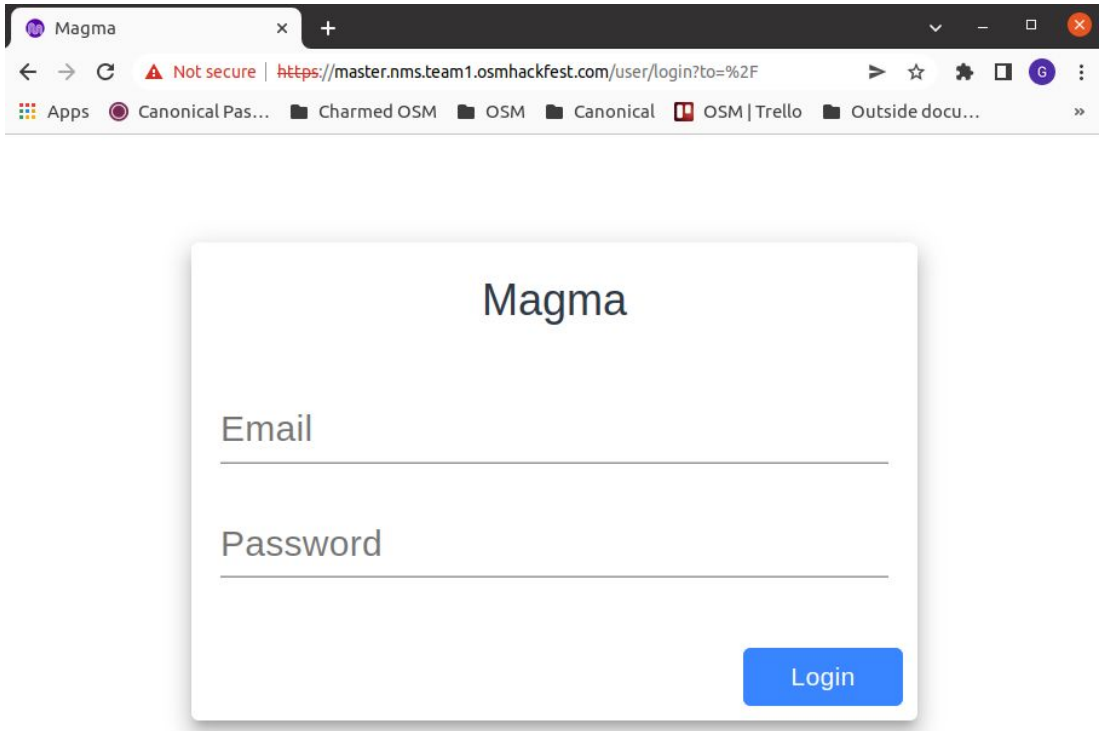
```
    $ juju scp -m $model-name --container="magma-orc8r-certifier" \
       orc8r-certifier/0:/var/opt/magma/certs/..data/rootCA.pem \
       rootCA.pem
```

Magma Orchestrator GUI

Get the GUI credentials

```
$ osm ns-action magma_orc_cnf_ns \
        --vnf_name magma_orc_cnf \
        --kdu_name magma-orc-kdu \
        --action_name get-admin-credentials
$ osm ns-op-show <Op-id>
```

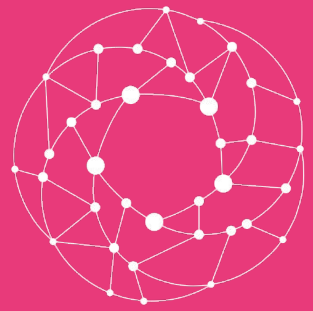
[Magma GUI]



Configure Magma Orchestrator

Open Source MANO

1. Create new tenant in the NMS (optional)
2. Create NMS user for tenant
3. Log in to the NMS (https://<TENANT>.nms.<YOUR DOMAIN>)
4. Create an LTE network in tenant's NMS
5. Configure PLMN (MCC: 772, MNC: 17)
6. Create APN (name: default)
7. Import subscribers to the NMS (csv file available on each team's Google Drive)

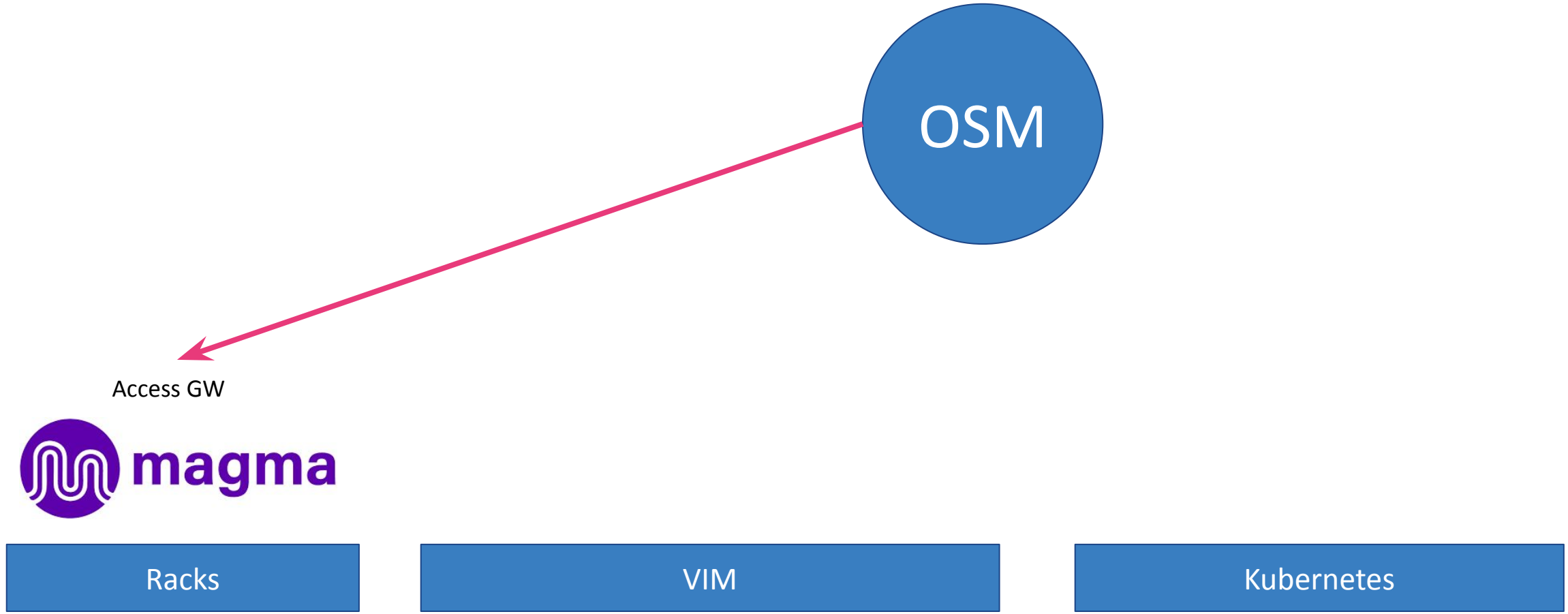


Open Source
MANO

Magma Access GW

ETSI

Magma Access GW



PNF vs VNF

Open Source MANO

PNF

Higher unit cost
Bulky
Fragile
Needs care

VNF

Low unit cost
Compact
Recyclable
Needs no care

https://www.linkedin.com/pulse/technology-analogy-physical-virtual-network-functions-milind-kulkarni/

PNF - Bare Metal?

Open Source MANO

Physical Network Function

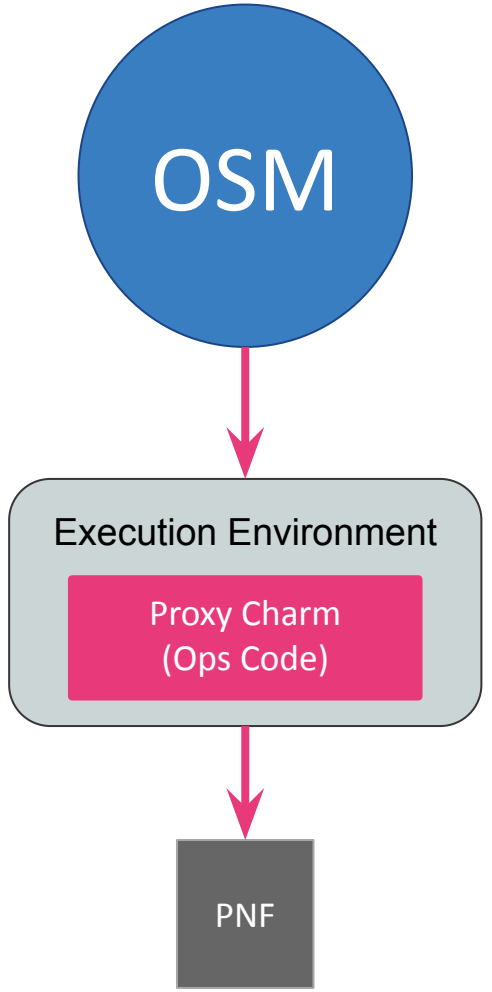
Implementation of a NF via a tightly coupled software and hardware system

- PNF refers to a function that is fixed
  - Purpose built to provide a specific function - hardware appliance
- PNF does not always mean bare metal
  - Can be unmanaged software in VM
- VNF does not always mean running in VM or Container
  - OpenStack Ironic allows for management of bare metal like a VM

https://www.etsi.org/deliver/etsi_gs/nfv/001_099/003/01.04.01_60/gs_nfv003v010401p.pdf

How do we Manage a PNF

- OSM must be given information about the PNF
  - Register a PNF as a logical entity with IP and other info
- Use in standard network function package descriptors
  - Network service and virtual network function descriptors
  - Templates that tell OSM about the PNF
- On Network Service deployment
  - OSM does not launch any VM or container
  - OSM creates an execution environment for the PNF
  - All actions execute in this environment



Breaking it down: Creating Packages

```
$ osm nfpkg-create magma_agw_pnf.tar.gz
$ osm nspkg-create magma_agw_ns.tar.gz
```

Telling OSM About the PDU

- Need to tell OSM some information
  - Name
  - Type
  - Interfaces with IP addresses

```
VIMID=`osm vim-list | grep osm_ |\
       awk '{ print $4 }'`

osm pdu-create --descriptor_file \
               pdu.yaml
```

```
name: MagmaAGW
description: Magma Access GW
type: gateway
vim_accounts: [ <VIM_ID> ]
shared: false
interfaces:
 -   name: gateway_public
     ip-address: <Magma AGW SGi IP>
     mgmt: true
 -   name: vnf_internal
     ip-address: <Magma AGW S1 IP>
     mgmt: false
```

Onboard the PNF in OSM

```
$ osm ns-create --ns_name magma_agw_ns \
                --nsd_name magma_agw_ns \
                --vim_account aws-site
```

Configure Magma AGW

Copy `rootCA.pem` from Magma Orchestrator to Magma AGW machine

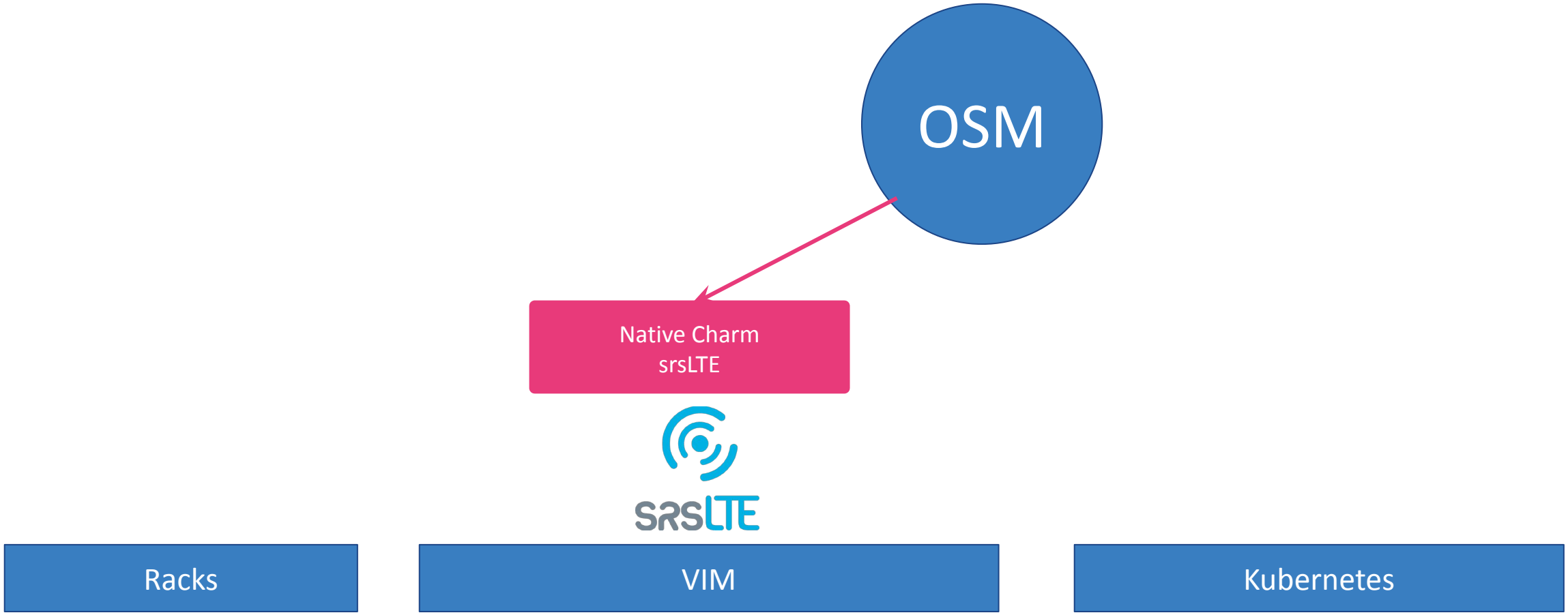
```
$ sudo su
# magma-access-gateway.configure \
        --domain team1.osmhackfest.com \
        --root-ca-pem-path ./rootCA.pem
```

Open Source
MANO

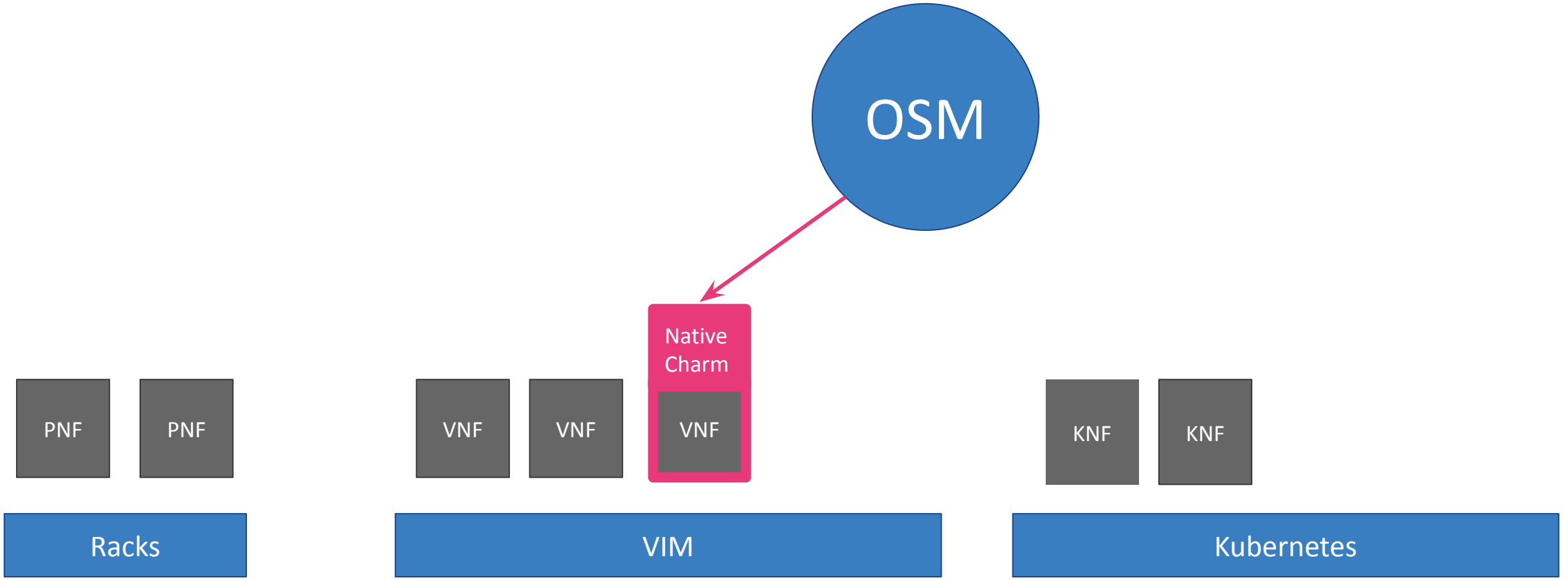
SRS LTE

ETSI

srsLTE EnodeB + UE emulator



srsLTE EnodeB + UE emulator



eNodeB and UE Emulator VNF

- Image: Ubuntu20.04 cloud image
  - Just Enough Operating System (JEOS)
  - Base OS with cloud init and ssh, very few packages installed
- All software is installed at deployment time

eNodeB and UE Emulator Operations

- Native charm:
  - Name: srs-enb-ue
  - Actions:
    - Attach UE
    - Detach UE

Day-1 operations (VNFd)

```
vnfd:
  description: srsLTE VDU
  df:
  - id: default-df
    lcm-operations-configuration:
      operate-vnf-op-config:
        day1-2:
        - id: srsLTE-vdu
          initial-config-primitive:
          - name: config
            execution-environment-ref: srs-enb-ue-ee
            parameter:
            - name: bind_address_subnet
              value: <bind_address_subnet>
            - name: mme_addr
              value: <mme_addr>
              [...]
```

Day-2 operations (VNFd)

```
vnfd:
  description: K8s container deployment of Magma Orchestrator
  df:
  - id: default-df
    lcm-operations-configuration:
      operate-vnf-op-config:
        day1-2:
        - id: magma-orc-kdu
          config-primitive:
          - name: attach-ue
            execution-environment-ref: srs-enb-ue-ee
            parameter:
            - data-type: STRING
              name: usim-imsi
            - data-type: STRING
              name: usim-k
            - data-type: STRING
              name: usim-opc
          - name: detach-ue
            execution-environment-ref: srs-enb-ue-ee
              [...]
```

Onboarding to OSM

```
$ osm nfpkg-create srs-lte-enb_vnfd.tar.gz
$ osm nspkg-create srs-lte-enb_nsd.tar.gz
```

Deploy srs LTE

```
$ osm ns-create
  --ns_name enb --nsd_name srs-lte-enb_nsd --vim_account aws-site \
  --config "{vld: [
              {name: mgmt, vim-network-name: <EKS public subnet> }
            ],
            additionalParamsForVnf: [
              {member-vnf-index: 'srsLTE',
               additionalParams: {
                 bind_address_subnet: '<subnet CIDR>',
                 mme_addr: '<AGW eth1 IP>',
                 enb_mcc: '722',
                 enb_mnc: '71'}}]}"
```

Attach UE to Magma

Open Source MANO

```
$ osm ns-action enb --vnf_name "srsLTE"
        --vdu_id srsLTE-vdu \
        --action_name attach-ue \
        --params '{
            usim-imsi: "722170000000008",
            usim-k: "c8eba87c1074edd06885cb0486718341",
            usim-opc: "17b6c0157895bcaa1efc1cef55033f5f"
          }'
```