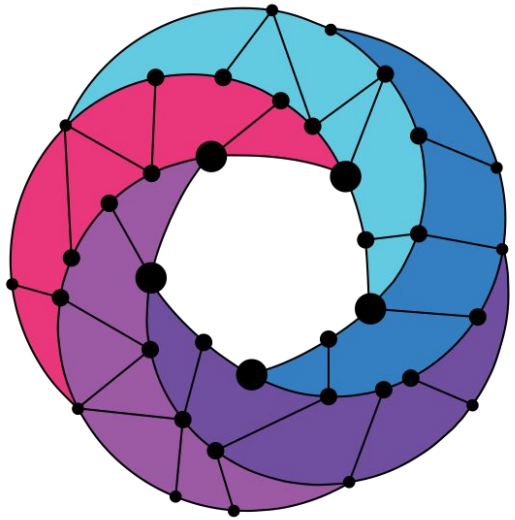


Open Source MANO

# OSM Hackfest – Session 4.3
# PNFs (Physical Network Functions)

Tomás Villaseca (Whitestack)



ETSI

# PNF / HNF - Definitions

- **PDU:** Physical deployment unit. It refers to the server itself.
- **PNF:** Physical network function. It refers to a HW box that provides a networking function. Routers, firewalls, load balancers…
- **HNF:** Hybrid network function: Network function composed of both physical and virtual elements.

In OSM, there are no fundamental differences between a VNF, a PNF or a Hybrid Network Function (HNF)

In those cases where we want to define NS packages consisting of PNF packages or HNF packages, **OSM needs to be instructed about the available PDUs.**

# Scenario description

Hybrid NS that can be launched and operated from OSM.

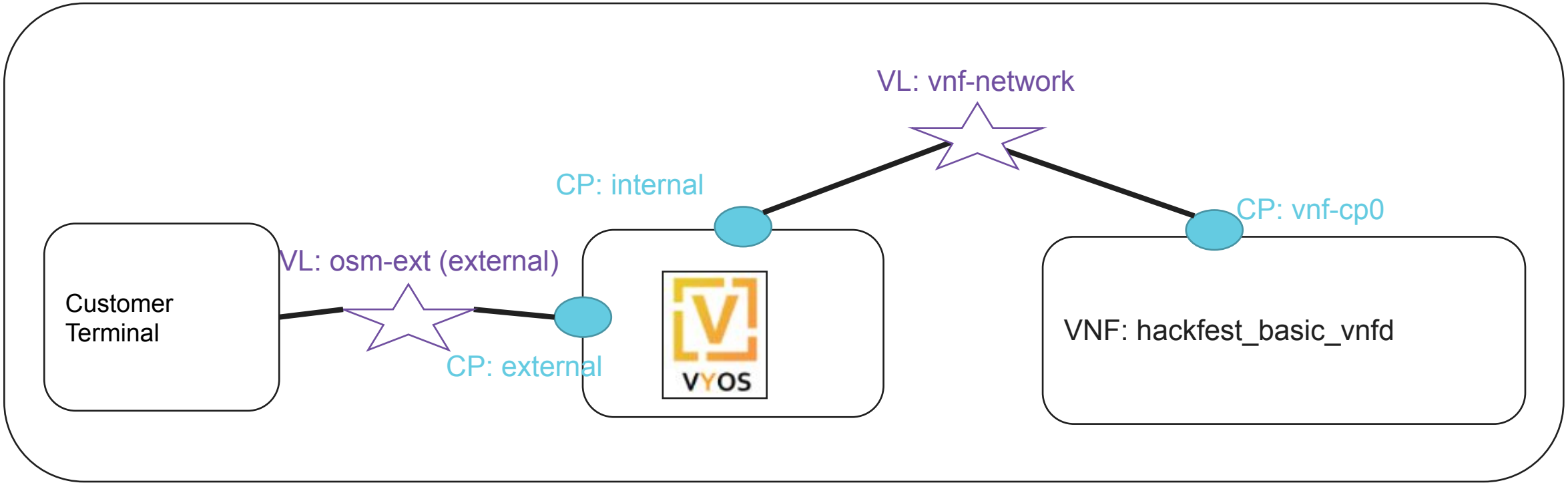
The elements are:

- A client VM and a VyOS router (our PDU)
- An Hybrid Network Service containing:
  - A PNF which will represent the existing VyOS router (PDU)
  - A VNF which implements the hackfest basic scenario that we have previously used

A 'day-2' proxy charm is used to create a new file in the VyOS router when we run the action.

So once the Network Service is launched, end customers we will be able to “configure” our PNF (in our case to create a new file)

# Scenario description




- External network: it should be available already in your VIM (our mgmt network)
- Internal network: it should be available already in your VIM

# Step 1 - PDU creation

a) Download the PDU template from here and make sure you update the VyOS IP addresses and the VIM account id accordingly:

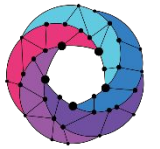
https://osm-download.etsi.org/ftp/osm-6.0-six/8th-hackfest/packages/PDU_router.yaml

```
name:           router01
description:    router
type:           gateway
vim_accounts:   [ 0a3a0a79-a86c-4812-9409-7509ff78d778 ]
shared:         false
interfaces:
 -  name:       eth0
    ip-address: [external IP address with no brackets]
    vim-network-name: PUBLIC
    mgmt:       true
 -  name:       eth1
    ip-address: [internal IP address with no brackets]
    mgmt:       false
```

b) Create the PDU in OSM:

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

# Step 2 - Onboard the VNF and NS descriptors




At the following location:

https://osm-download.etsi.org/ftp/osm-6.0-six/8th-hackfest/packages

You will find the following descriptors:

a) VNF descriptor (cirros_vnf.tar.gz) : contains the hackfest cirros_vnf scenario with one VDU

b) PNF descriptor (gateway_pnf.tar.gz): contains the PNF, requesting a PDU with type 'gateway' and containing a proxy-charm that will “configure” the router with a day-2 charm.

c) Hybrid NS descriptor (vyos_hnf_nsd.tar.gz) : contains both VNF and PNF.

# PNF and VNF descriptor

```
vdu:
  - description: gateway_pdu
    id: gateway_pdu
    interface:
    - external-connection-point-ref: gateway_public
      name: eth0
      type: EXTERNAL
    pdu-type: gateway
  vnf-configuration:
    initial-config-primitive:
    - name: config
      parameter:
      - name: ssh-username
        value: osm
      - name: ssh-password
        value: osm2019
      - name: ssh-hostname
        value: <rw_mgmt_ip>
      seq: '1'
    config-primitive:
    - name: configure-remote
      parameter:
      - name: filename
        data-type: STRING
        default-value: '/home/osm/touched'
    juju:
      charm: vyos-config
```

```
vnfd:vnfd-catalog:
  vnfd:
  - id: cirros_vnfd
    name: cirros_vnf
    short-name: cirros_vnf
    description: Simple VNF example with a cirros
    vendor: OSM
    version: '1.0'
    vdu:
    - id: cirros_vnfd-VM
      name: cirros_vnfd-VM
      description: cirros_vnfd-VM
      count: 1
    connection-point:
      - name: eth0
        type: VPORT
```

# NS descriptor

```
nsd:nsd-catalog:
  nsd:
  - constituent-vnfd:
    - member-vnf-index: 1
      vnfd-id-ref: vyos_pnf_charmed
    - member-vnf-index: 2
      vnfd-id-ref: cirros_vnfd
    description: vyos_pnf_nsd descriptor
    id: vyos_hnf_nsd
    name: vyos_hnf_nsd
    short-name: vyos_hnf_nsd
    vendor: OSM Composer
    version: '1.0'
```

```
vld:
- id: public_vld
  mgmt-network: 'true'
  name: public_vld
  type: ELAN
  vim-network-name: vnf-network
  vnfd-connection-point-ref:
  - member-vnf-index-ref: '2'
    vnfd-connection-point-ref: eth0
    vnfd-id-ref: cirros_vnfd
  - member-vnf-index-ref: '1'
    vnfd-connection-point-ref: gateway_public
    vnfd-id-ref: vyos_pnf_charmed
```

# Step 3 - Launch the Hybrid Network Service


```
osm ns-create --ns_name [name-of-your-ns] --nsd_name  vyos_hnf_nsd --vim_account
 [name-of-your-VIM]
```

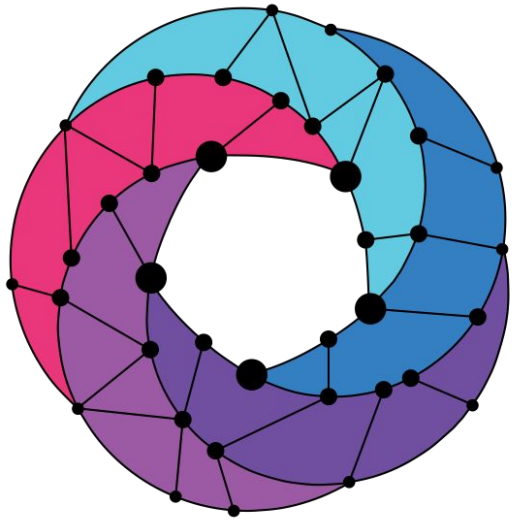
Then execute the following action “configure-remote” to “configure” our router:

```
osm ns-action --vnf_name 1 --action_name configure-remote --params '{filename:
 /home/osm/test}' [name-of-your-ns]
```

# Step 4 - Post checkings

The NS has been launched, the proxy charm should have “configured” the VyOS router with the new file (test) . You can check it by accessing the VyOS router using SSH (osm/osm2019)

From the router, check that you have connectivity to the VM created. SSH with cirros credentials (cirros/cubswin:))



# Open Source MANO

Find us at:

osm.etsi.org

osm.etsi.org/wikipub



ETSI