Introduction

- Introduction of NFV means that VNFs deployed on shared infrastructure is quickly replacing dedicated physical equipment for many applications.

- But some use cases still include PNFs as part of the E2E service, typically at the network edge:
  - vCPE UC in ONAP R1
  - 5G RAN UC proposed for ONAP R2

- To achieve the full benefits of unified management and automation capabilities, native support for PNFs should be added in ONAP:
  - Some support already exists today in R1
  - This presentation proposes some principles and highlights areas that need further discussion.
R2 PNF requirements

• The following requirement candidates are derived from the 5G RAN UC
  - Support for PNF Onboarding
    • Design & Orchestration (1)
    • Plug and Play (2)
  - Support for PNF Configuration (3)
    • Needs PNF IP address from (2)
  - Support for PNF Data Collection (4)
    • Needs configuration support from (3)

• Some of the flows such as PnP may depend on additional network functions not shown in this figure
Use case service design pattern

- Two main classes of services
  - *Infrastructure services* provide resources for the UC supporting an entire network or area
  - *Customer services* are instantiated once per customer to allocate dedicated resources
    - These may include PNFs
- For the 5G RAN UC, all services are in principle infrastructure services
  - But gNBs share the same requirements as customer services for highly automated and large scale roll-out
Design-time and run-time entities

- Large scale roll-out requires template re-use
  - Design once, instantiate multiple times
- This principle must be supported also when service templates include PNFs
  - Corresponds to run-time binding of PNF nodes in the service topology to specific HW resource
Run-time binding using TOSCA

- There should be some flexibility when designing the service
  - Different parameters (serial number, MAC address etc) could be used depending on PNF type and the supported plug and play procedure

- Using an input parameter allows the service designer to declare what information must be provided when instantiating the service
  - But not fully in line with principle to avoid run-time attributes in the design-time model
    - Compare to VNF placement policies

```yaml
topology_template:
  inputs:
    serial_number:
      type: string
  node_templates:
    EdgeDevice:
      type: org.onap.resource.pnf.VendorAndModel
      properties:
        serial_number: { get_input: serial_number }
```

The service designer creates a service template where PNF serial number is an input parameter.

The network/service operator supplies the serial number when instantiating the service template.
PNF design model

• New PNF types can be created in SDC, but what are catalog entries supposed to capture in their definition?
  - Do they have an internal composition, and what resources would be applicable for PNFs?
    • VNFs (VFs) are built up from VDUs, VLs, CPs etc
  - Is there any information that should be possible to import by onboarding vendor supplied data and artifacts?
PNF Plug and Play requirements

• ONAP Plug and Play for PNFs should support
  - Providing the PNF with initial configuration and SW image if required
  - Updating the PNF instance status in the inventory (AAI) when the PNF is detected
    • Register the PNF management IP address in case it is dynamically allocated from pool
    • There are two different scenarios since the PNF instance record may or may not already exist in the inventory

• Security aspects must also be considered
PNF Plug and Play procedures

• On the following slides two examples are shown
  - vCPE UC from R1
  - Network scenario based on 3GPP TS 32.508 (Procedure flows for multi-vendor PnP)
    • While used here as example, the alignment of the 3GPP management architecture and ONAP is currently not settled

• Another solution is draft-ietf-netconf-zerotouch mentioned in the proposed SD-WAN UC

• As a platform, ONAP should be able to support several different PnP procedures!
Custom workflow has been added to SDNC to receive DMaaS event and update AAI with PNF information such as MAC address and WAN IP address.
Plug and Play – Derived from TS 32.508 (example)

- **DHCP**: Provides outer IP configuration, also FQDN/IP for CA/RA, SeGW and initial EM.
- **CA/RA**: Provides certificate to be used by node in the operator network.
- **DNS**: In this example the service template triggers configuration of instance-specific PNF data that PnP depends on in the initial EM.
- **Initial EM**: Provides initial configuration and SW for PNF.
- **SeGW**: As another option PNF could register itself with ONAP using information from initial configuration including credentials.

**Figures**: OSS/BSS, DCAE, SDNC, AAI, SO.
Example sequence

Custom workflow

SO

Controller(s)

Initial EM

eNB (PNF)

Create PNF node (serial#)

Prepare for PNF

PNF PnP event (serial#)

Configure PNF node

Configure PNF

Start PNF node

Start PNF

Configure PNF complete

Start PNF complete

Configure PNF complete
Example sequence issues

• What is the preferred work division between SO and controllers for the TOSCA standard lifecycle operations (create, configure, start)?
  - Execution of workflows
  - Update of inventory

• What controllers are involved in the different steps?
  - Orchestration
  - Handling PnP event from external system or node

• If orchestration is initiated before the PNF is available, the process will potentially block for a (very) long time in the PNF creation step
  - Will this be a problem?
Security aspects

• Both PnP examples send events to ONAP in order to update inventory with information about the PNF

• For vCPE, the event even triggers instantiation of the customer service

• For security reasons, it is important to ensure
  - The event source is authenticated
  - The inventory update is connected to policy deciding if operation is allowed for this event source
    • Prevents any VNF able to send events to ONAP from updating the inventory

• Other note: A standard VES event for PnP should perhaps be defined
Thank You