ICN Local Controller test cases

Bare-Metal deployment test case

This test case expected to test the ICN local controller for following scenario.

Some terms that are used in the test case

ESP (Edge Service Provider) : ESP owns the edge location. Set of users of ESP are allowed to perform the operations on the infra-local-controller. ESP procures the server hardware, switches and connects them together. Also,it is ESP responsibility to get the Internet connectivity.

ESP-Tenant : ESP-Tenant is the customer of ESP. Each ESP-tenant has its own users to maintain and manage their Edgelets.

Edgelet : Edgelets are Kubernetes Clusters. Each Edgelet belongs to one ESP-Tenant. Edgelets are created by ESPs. Once created Edgelets are managed by ESP-Tenant users.

Test case details:

- Two tenants - T1 and T2.
    - T1 needs two edgelets - E1, E2
    - T2 needs one edgelet - E3
- E1 requires two server K8S Cluster
- E2 requires one-server K8S Cluster
- E3 requires one-server K8S Cluster.
- E1:
    - OS : Ubuntu 18.04 on all servers
    - Use KUD version 2.0 for K8S.
    - Server1 is for K8S master and worker
    - Server2 is dedicated for K8S worker.
    - Need two provider networks - VLAN 100, VLAN 200
    - Need Server2 be labelled with "SSD", "Provder_network_1"
    - Need Server1 be labelled with "Provider_network_2"
- E2:
    - OS: RH 8.2 on all servers
    - Use KUD version 3.0 for K8S
    - Server is for both K8S master and worker,
    - Need three provider networks : VLAN 300, VLAN 301 and VLAN 302.
    - Need Server be labelled with "Provider_Network_1", "Provider_Network_2" and "Provider_Network_3".
- E3:
    - OS: Ubuntu 19.04 on all servers
    - Use RKE for Kuerbenetes installation
    - Server is for both K8S master and worker,
    - Need three provider networks : VLAN 400, VLAN 401 and VLAN 402.
    - Need Server be labelled with "Provider_Network_1", "Provider_Network_2" and "Provider_Network_3".

We understand that today, only one Linux version and only one KUD version is supported.

Steps:

- ESP creates T1
- ESP adds E1 for T1
- ESP adds E2 for T1
- ESP creates T2
- ESP adds E3 for T2
- ESP dedicates VLAN 100, 200, 300, 301 and 302 for T1.
- ESP dedicates VLAN 400, 401 and 402 for T2.
- T1 uploads Ubuntu 18.04 image
- T1 uploads KUD 2.0 image
- T1 uploads RH8.2 image
- T1 uploads KUD 3.0 image
- T1 on behalf of E1 adds description for servers.
- T1 on behalf of E1 enters node labels.
- T1 on behalf of E1 enters provider network information.
- T1 initiates the infrastructure installation for E1.
- T1 on behalf of E2 adds description for servers.
- T1 on behalf of E2 enters node labels.
- T1 on behalf of E2 enters provider network information.
- T1 initiates the infrastructure installation for E2.

- T2 uploads Ubuntu 19.04 image
- T2 uploads RKE image
- T2 on behalf of E3 adds description for servers.
- T2 on behalf of E3 enters node labels.
- T2 on behalf of E3 enters provider network information.
- T1 initiates the infrastructure installation for E3.

Proof:

- Ensure that there are three K8S clusters created.
- Ensure by running applications in these K8S clusters.
- Make sure that T2 cann't view T1's images and also ensure that they can't look into T1s cluster.

Kuralamudhan Ramakrishnan Itohan Ukponmwan Akhila Kishore Enyinna Ochulor Huifeng Le Tingjie Chen