IETF Network Slice NBI YANG

draft-wd-teas-ietf-network-slice-nbi-yang-02

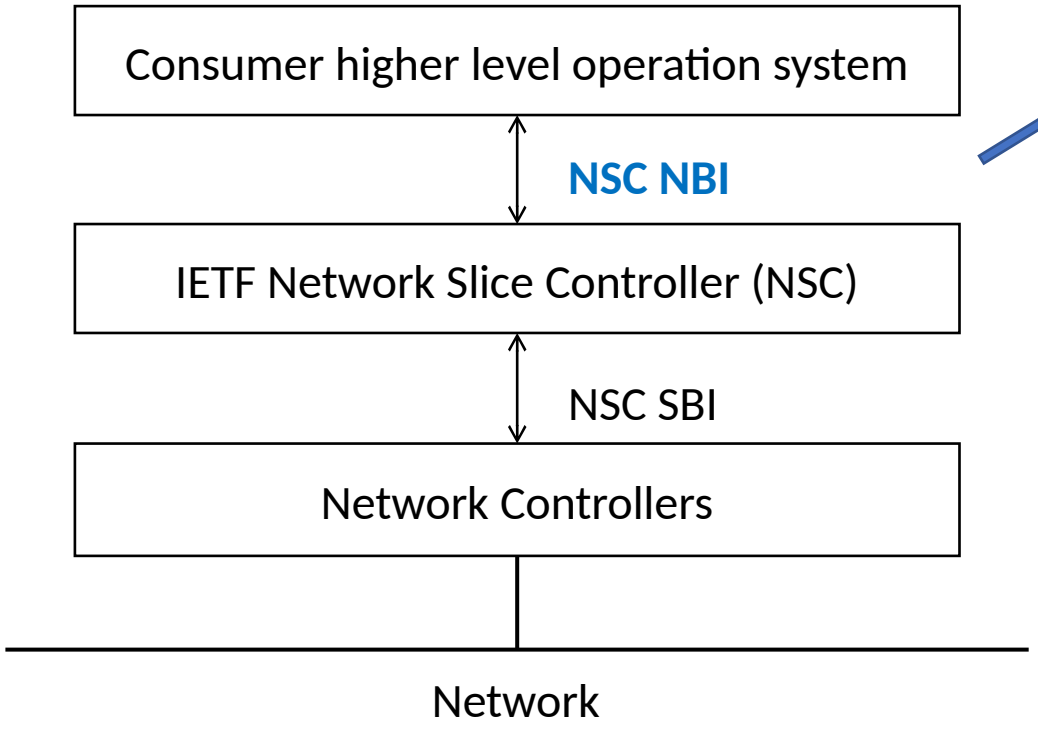
TEAS WG

Mar 2021

Bo Wu, Dhruv Dhody, Reza Rokui (presenting), Liuyan Han

IETF Network Slice NBI YANG Key components

- This update aims to keep this draft consistent with the IETF Network Slice definition draft.
- Modelling consideration: support IETF Network Slice configuration and monitoring



From IETF NS definition WG draft

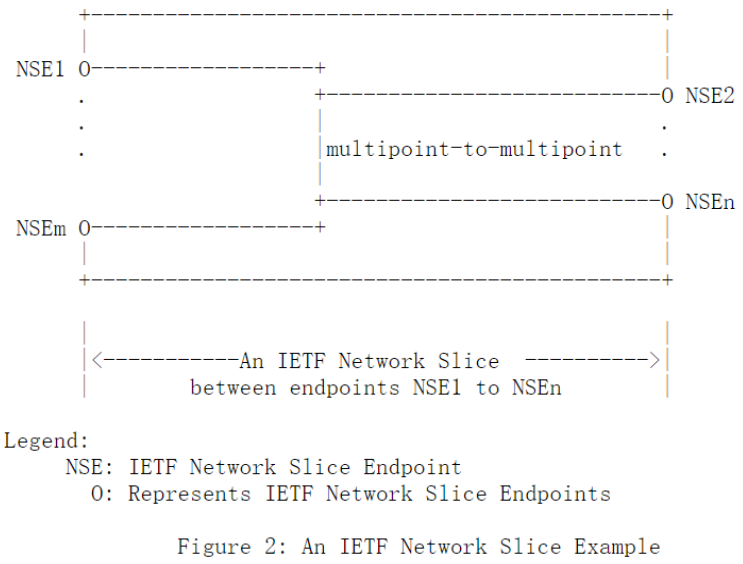
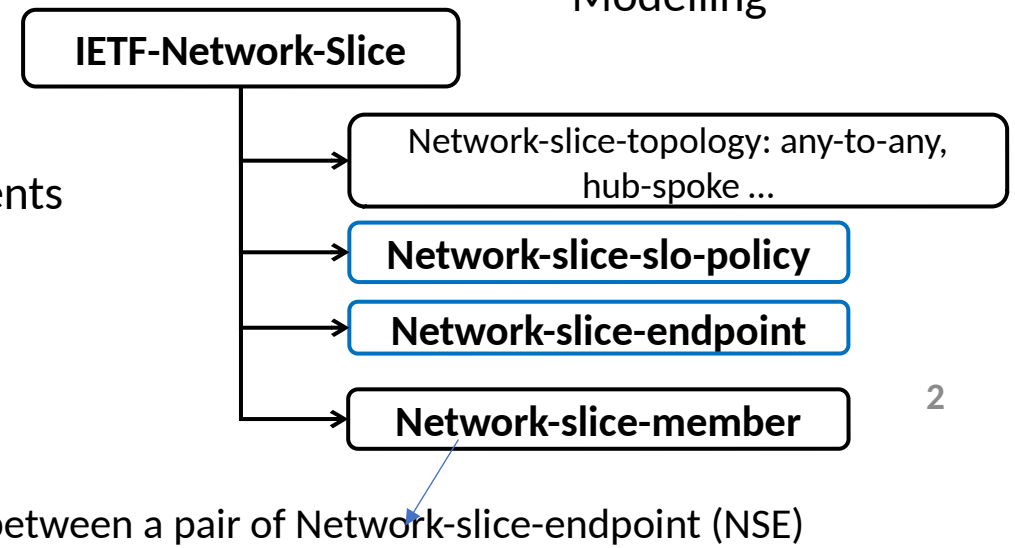


Figure 2: An IETF Network Slice Example

Modelling

YANG components



A connection between a pair of Network-slice-endpoint (NSE)

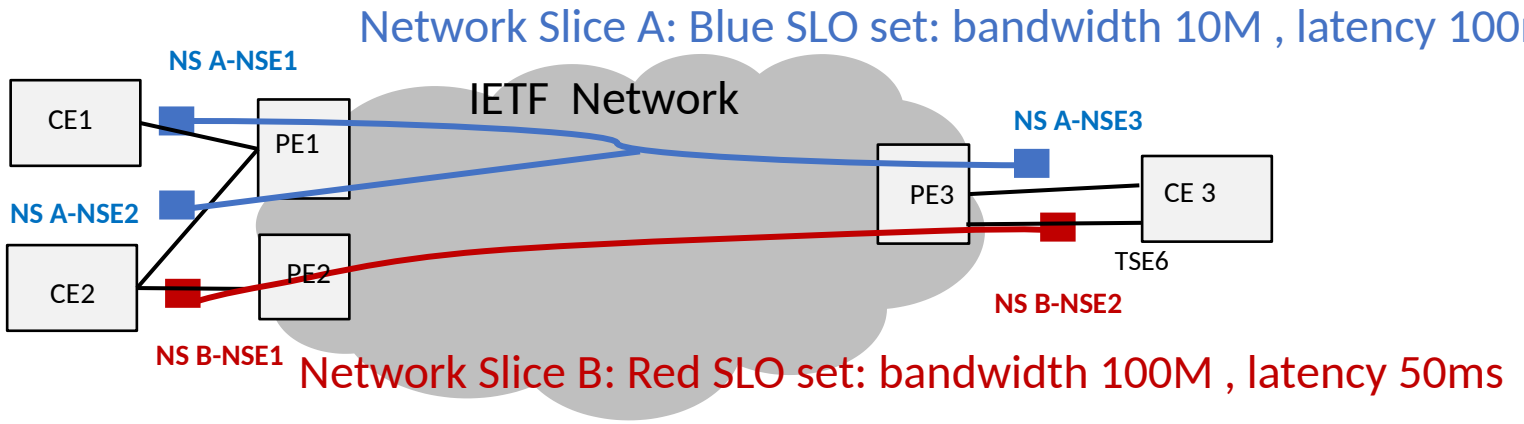
Major updates

- Replace multiple SLO sets per NS into one SLO set - remove modelling concept “connection-group”
- Add CE-facing or PE-facing mapping text to NSE modeling concepts
- Add JSON examples to clarify the usage of the YANG model, NS templates and NS SLO usage

Network Slice SLO Modelling

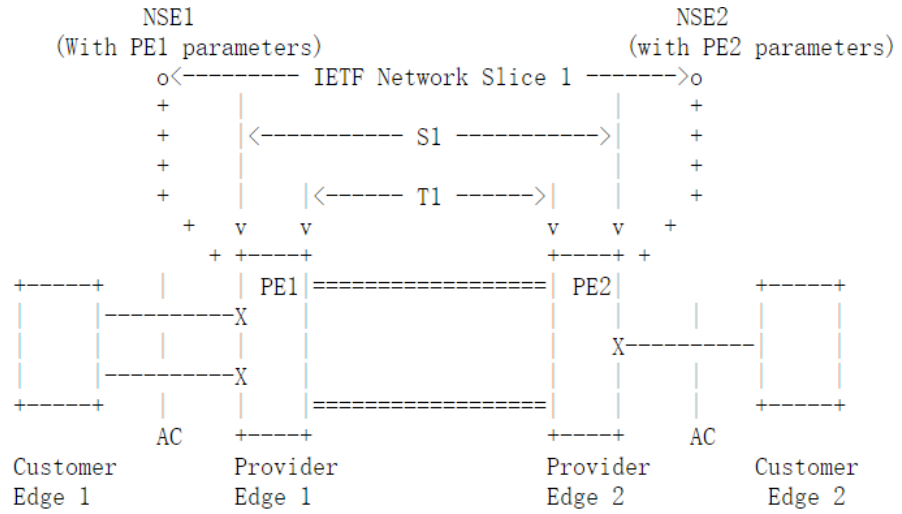
- An “IETF Network Slice” supports one global SLO policy set for a slice.
  - Support the Minimal set in draft-ietf-teas-ietf-network-slice-definition defines: Guaranteed Minimum Bandwidth, Guaranteed Maximum Latency, etc.
  - Be flexible to extend other SLO attributes in future
- When a customer has a service requirement with more than one SLO policy sets, it could create multiple slices using separate API calls, one for each slice with a specific SLO policy set.

Network Slices examples

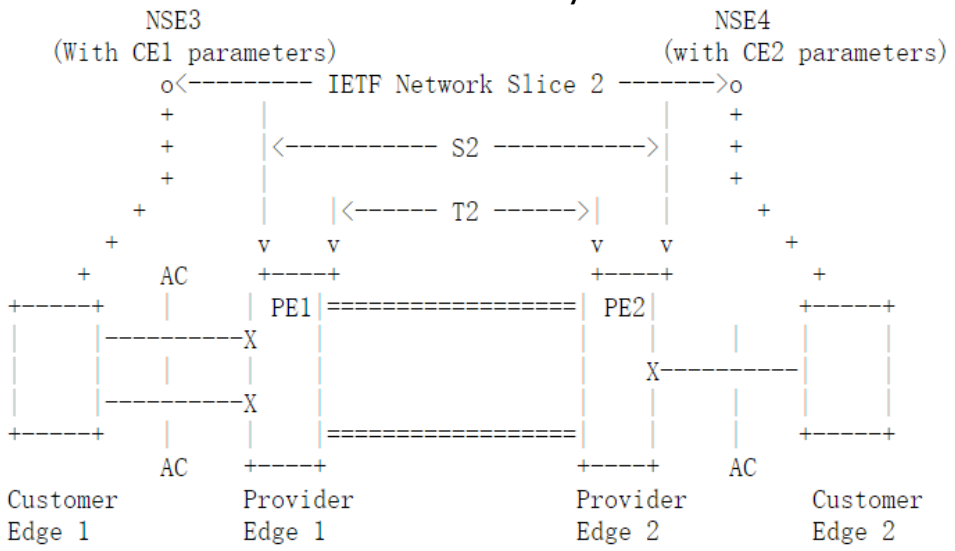


Network Slice Endpoint Modelling

- Modelling Consideration
  - An NSE should be uniquely identified.
  - An NSE is an abstract entity with attributes that can map to a network node, e.g. CE or PE.
  - An NSE can only belong to one single Network Slice.
  - Will be aligned to the final definition with WG consensus, the current definition allows for flexibility!



Legend:
O: Representation of the IETF network slice endpoints (NSE)
+: Mapping of NES to PE or CE nodes on IETF network
X: Physical interfaces used for realization of IETF network slice
S1: L0/L1/L2/L3 services used for realization of IETF network slice
T1: Tunnels used for realization of IETF network slice



Legend:
O: Representation of the IETF network slice endpoints (NSE)
+: Mapping of NES to PE or CE nodes on IETF network
X: Physical interfaces used for realization of IETF network slice
S2: L0/L1/L2/L3 services used for realization of IETF network slice
T2: Tunnels used for realization of IETF network slice

Open issue: Integrate Service Function Chains (SFC) as part of Network Slice

- A slice may require the invocation of service functions (firewall, for example) in a given order
- The relationship with NSE is not clear and draft-ietf-teas-ietf-network-slice-definition and draft-nsdt-teas-ns-framework does not give much detailed description on SF

Next Step

- Solicit comments and reviews from WG
- Solicit WG adoption

Backup

```
module: ietf-network-slice
   +--rw ietf-network-slices
      +--rw ns-templates
      |  +--rw slo-template* [id]
      |     +--rw id                        string
      |     +--rw template-description?   string
      +--rw ietf-network-slice* [ns-id]
         +--rw ns-id                 string
         +--rw ns-description?       string
         +--rw ns-tag*               string
         +--rw ns-topology?          identityref
         +--rw (ns-slo-policy)?
         |  +--:(standard)
         |  |  +--rw slo-template?   leafref
         |  +--:(custom)
         |     +--rw slo-policy
         |        +--rw policy-description?   string
         |        +--rw ns-metric-bounds
         |           +--rw ns-metric-bound* [metric-type]
         |              +--rw metric-type          identityref
         |              +--rw metric-unit          string
         |              +--rw value-description?   string
         |              +--rw boundary?            uint64
         +--rw status
         |  +--rw admin-enabled?   boolean
         |  +--ro oper-status?     operational-type
         +--rw ns-endpoint* [ep-id]
         |  +--rw ep-id                 string
         |  +--rw ep-description?       string
         |  +--rw ep-role?              identityref
         |  +--rw location
         |  |  +--rw altitude?    int64
         |  |  +--rw latitude?    decimal64
         |  |  +--rw longitude?   decimal64
         |  +--rw node-id?              string
         |  +--rw ep-ip?                inet:host
         |  +--rw ns-match-criteria
         |  |  +--rw ns-match-criteria* [match-type]
         |  |     +--rw match-type    identityref
         |  |     +--rw value?        string
```

```
|  +--rw ep-network-access* [network-access-id]
         |  |  +--rw network-access-id             string
         |  |  +--rw network-access-description?   string
         |  |  +--rw network-access-node-id?       string
         |  |  +--rw network-access-tp-id?         string
         |  |  +--rw network-access-tp-ip?         inet:host
         |  +--rw ep-rate-limit
         |  |  +--rw incoming-throughput
         |  |  |  +--rw maximum-throughput?   te-types:te-bandwidth
         |  |  +--rw outgoing-throughput
         |  |     +--rw maximum-throughput?   te-types:te-bandwidth
         |  +--rw ep-protocol
         |  +--rw status
         |  |  +--rw admin-enabled?   boolean
         |  |  +--ro oper-status?     operational-type
         |  +--ro ep-monitoring
         |     +--ro incoming-utilized-bandwidth?
         |     |       te-types:te-bandwidth
         |     +--ro incoming-bw-utilization        decimal64
         |     +--ro outgoing-utilized-bandwidth?
         |     |       te-types:te-bandwidth
         |     +--ro outgoing-bw-utilization        decimal64
         +--rw ns-member* [ns-member-id]
            +--rw ns-member-id              uint32
            +--rw ns-member-description?   string
            +--rw src
            |  +--rw src-ep-id?   leafref
            +--rw dest
            |  +--rw dest-ep-id?   leafref
            +--rw monitoring-type?         ns-monitoring-type
            +--ro ns-member-monitoring
               +--ro latency?      yang:gauge64
               +--ro jitter?       yang:gauge32
               +--ro loss-ratio?   decimal64
```