IETF Network Slice Service YANG Model
draft-ietf-teas-ietf-network-slice-nbi-yang-02

TEAS WG
July 2022 (IETF 114 Philadelphia)

• Bo Wu (Huawei Technologies)
• Dhruv Dhody (Huawei Technologies)
• **Reza Rokui (Ciena)** *(presenting on behalf of authors)*
• Tarek Saad (Juniper Networks)
• Liuyan Han (China Mobile)

• Contributors: Luis M. Contreras (Telefonica), and John Mullooly (Cisco)
NS Service model Status Summary

Rev-02 summary:

- Align the “sdp”, “connectivity-construct”, “attachment-circuit” with terms of the NS framework
- Reuse the grouping definition of RFC9181, including “vpn-common:service-status”, “vpn-common:vpn-topology”
- Used a choice statement in the YANG model for connectivity-construct to model P2P, P2MP and A2A types
- Add more examples to illustrate the use of P2P, P2MP, any-to-any connectivity constructs, and the also use of 'service-match-criteria'
- Add more 'service-match-criteria' match types to support a slice service with multiple connectivity-constructs, e.g. match-ip-address, DSCP-match, and "match-any" for default forwarding behavior
- Remove Appendix C, which describes SDP mapping in 5G End-to-end Network
  - See new draft-gcdrb-teas-5g-network-slice-application IETF Network Slice Application in 5G End-to-End Network Slice

Issues status

- Open issues, 5 total open issues, 11 closed:
  - https://github.com/lana-wu/ietf-ns-nbi/issues
NS service “connectivity-construct”

- NS framework draft defines three basic connectivity constructs:
  - P2P, P2MP, A2A (any2any)
- An NS service can have one or multiple connectivity constructs.

---

```
+--rw connectivity-construct* [cc-id]
   |   +--rw cc-id uint32
   |   +--rw (connectivity-construct-type)? ++-(p2p)
   |       |   |   +--rw p2p-sender-sdp? -> ../../sdps/sdp/sdp-id
   |       |   |   +--rw p2p-receiver-sdp? -> ../../sdps/sdp/sdp-id
   |   |   ++-(p2mp)
   |       |   |   +--rw p2mp-sender-sdp? -> ../../sdps/sdp/sdp-id
   |       |   |   +--rw p2mp-receiver-sdp* -> ../../sdps/sdp/sdp-id
   |   +--rw (a2a)
   |       +--rw a2a-sdp* [sdp-id] +--rw sdp-id leafref
   |       +--rw (slo-sle-policy)? ... 
   +--rw (slo-sle-policy)? ...
```

---

```
+--rw connection-groups
   +--rw connection-group* [connection-group-id] string
       +--rw connection-group-id identityref
       +--rw (slo-sle-policy)? ...
   +--rw connectivity-construct* [cc-id]
       |   +--rw cc-id uint32
```

---

TEAS meeting @ IETF 114 Philadelphia
Open issue #1 slo-sle-policy

- https://github.com/lana-wu/ietf-ns-nbi/issues/15,24,26

Three issues

1) SLO/SLE can be configured at 3 levels - slice, connection group and CC. It is required to state if the policy at the lower level inherits and updates what is specified or do not inherit anything and just go by what is specified at the lower level.

2) Currently we have option for a well-known template or custom only. Sometimes what is needed is only a slight change in the otherwise well-known template. Can a mechanism be provided for it?

3) There is no mechanism to configure a template. Is it a good idea for a service model to allow such a template configuration?

Status: SLO/SLE policy inheritance and optimization mechanisms in discussion

---

For example:
slice service: slo-sle='high-BW-slo-sle'
At connectivity-construct id=cc-id1, can override the bound
"metric-bounds": {
 "metric-bound": [
  {"metric-type": "service-slo-one-way-bandwidth", "metric-unit": "mbps", "bound": "1000"
  }
],
Open issue#2 - When “match-type” is “service-flow-match”, how to support complex combinations of IP packet headers?

• [https://github.com/lana-wu/ietf-ns-nbi/issues/6](https://github.com/lana-wu/ietf-ns-nbi/issues/6)

• Options:
  • Add a leafref of ACL name to complement the current <match-type,value>  
  • Explicitly define various combinations, like L3SM definition

• **Proposal**: Add ACL name
**Open issue#3 Technology agnostic nature may cause interoperability difficult**


**Issue**: To maintain the technology agnostic nature of the YANG model we use

1. A pair of (identity, value (string)) to support many attributes;
   - a common practice, for instance different metric-types use a common leaf metric-value of uint64 (RFC8776).
2. Support for “opaque” attributes that can be configured without an identity
   - This allow new attributes to added with updating YANG model

**Concern**: Using a string instead of an explicit type in YANG there could be interoperability issues. The use of opaque attributes that are not defined in YANG adds to this issue.

**Proposal**: Ask feedback from the WG and Yang-Doctors on how to balance the need for a technology-agnostic model v/s the issues pointed out above.

```yang
--rw sdp-peering
    +++w protocol* [protocol-type]
    |  +++w protocol-type identityref
    |  +++w attribute* [attribute-type]
    |     +++w attribute-type identityref
    |     +++w value* string
    +++w opaque* [attribute-name]
    ++-rw attribute-name string
    ++-rw value* string
```

**Identity peering-protocol-type:**
- peering-protocol-bgp
- peering-static-routing

**Identity peering-attribute-type:**
- remote-as
- neighbor
- local-as

---

**RFC8776:**
- ++-ro path-metric* [metric-type]
- ++-ro metric-type identityref
- ++-ro accumulative-value? uint64
- ro path-affinities-values
- ++-ro path-affinities-value* [usage]
  - ++-ro usage identityref
  - ++-ro value? admin-groups

---

**TEAS meeting @ IETF 114 Philadelphia**
Open issue#4 Handling of custom topology as input in the NS service model


• Proposal to provide more topology input to NSC during the realization
  • Similar to type 2 VN service defined in ACTN framework (RFC 8453)
    • draft-liu-teas-transport-network-slice-yang proposes a brand new model
    • draft-ietf-teas-actn-vn-yang refers to abstract TE topology

• Proposal:
  • Is this needed?
Next step

• Resolve the open issues
• Request further review and address comments