DetNet Configuration YANG Model Update

draft-ietf-detnet-yang-08

Xuesong Geng (gengxuesong@huawei.com)
Don Fedyk(dfedyk@labn.net)
Mach Chen (mach.chen@huawei.com)
Yeoncheol Ryoo (dbduscjf@etri.re.kr)
Reshad Rahman(rrahman@cisco.com)
Zhenqiang Li (lizhengqiang@chinamobile.com)
Status

• Merged YANG models working on Yanglint data instances for various configuration scenarios.
• Need to check terminology for consistency and clarity.
• Need to confirm data plane match data plane drafts.
• Need Updated reference pointers in the YANG model.
History

- Version 00: accepted as a WG document after IETF 102
- Version 01: `ietf-detnet-topology-yang` is defined independently
- Version 02: updated following the feedback from IETF103
  - Add 'Sequence Number Generation'
    - OAM considerations
  - Add 'DetNet Service Decapsulation'
  - Add 'DetNet Transport Tunnel Decapsulation'
- Version 03: DetNet Configuration Structure Update in IETF104 and IETF105
- Version 04 :
  - Modify the scope of DetNet YANG Model
- Version 05/06:
  - Two YANG Models Discussion -> Comparison
- Version 07
  - Merging Models. Some terminology alignment.
- Version 08
  - Aggregation and Instance Models
Scenarios Covered by DetNet YANG Model
(w/o Aggregation)

Corresponding Data Plane drafts:
draft-ietf-detnet-ip-06
draft-ietf-detnet-ip-over-mpls-06
draft-ietf-detnet-mpls-07
draft-ietf-detnet-mpls-over-udp-ip-06 (Partial)
draft-ietf-detnet-tns-vpn-over-mpls-03 (Partial)
draft-ietf-detnet-mpls-over-tns-03 (Not yet)
draft-ietf-detnet-ip-over-tns-03 (Not yet)

Not shown Ethernet or other Tunnels as be underlay
**Flow Model Attributes Supported by YANG**

**App-flow, DetNet flow and DetNet service**

### App-flow

**Characteristics**
- **FlowId**: unique (manag.) ID
- **FlowType**: Eth, MPLS, IP
- **DataFlowSpecification**: src/dst-addr, label, VLAN, etc.
- **TrafficSpecification**: interval, pkt-size, max-packet
- **FlowEndpoints**: Src, Dst(s)
- **FlowRank**
- **FlowStatus**

**Requirements**
- **FlowRequirements**: MinBW, PD, PDV, Loss, etc.
- **FlowBiDir**

Service Requirements similar to e.g., 802.1Qcc Attributes like UserToNetworkRequirements

### DetNet flow

**Characteristics**
- **DnFlowId**: unique (manag.) ID
- **DnPayloadType**: Eth, MPLS, IP
- **DnFlowFormat**: MPLS, IP
- **DnTrafficSpecification**: interval, pkt-size, max-packet
- **DnFlowEndpoints**: Ingress, Egress(s)
- **DnFlowRank**
- **DnFlowStatus**

**Requirements**
- **DnFlowRequirements**: MinBW, MaxLatency, MaxLatencyVariation, MaxLoss, MaxConsecutiveLossTolerance, MaxMisordering
- **DnFlowBiDir**

### DN Service

- **DnServiceId**: unique (manag.) ID
- **DnServiceDeliveryType**: Eth, MPLS, IP
- **DnServiceConnectivity**: p2p, p2mp
- **DnServiceRank**
- **DnServiceDeliveryProfile**: MinBW, MaxLatency, MaxLatencyVariation, MaxLoss, MaxConsecutiveLossTolerance, MaxMisordering
- **DnServiceBiDir**
- **DnServiceStatus**

---

A DetNet flow contains one or more App-flows (N:1 mapping).

A DetNet service supports one or more DetNet-flows (M:1 mapping).

Service Requirements similar to e.g., 802.1Qcc Attributes like UserToNetworkRequirements
YANG Model: Traffic Profile

```yang
++-rw traffic-profile* [profile-number]
   +--rw profile-number               uint16
   +--rw traffic-requirements
      |   +--rw min-bandwidth?            uint64
      |   +--rw max-latency?              uint32
      |   +--rw max-latency-variation?    uint32
      |   +--rw max-loss?                 uint8
      |   +--rw max-consecutive-loss-tolerance? uint32
      |   +--rw max-misordering?          uint32
   +--rw traffic-specification
      |   +--rw interval?                 uint32
      |   +--rw max-packets-per-interval? uint32
      |   +--rw max-payload-size?         uint32
      |   +--rw average-packets-per-interval? uint32
      |   +--rw average-payload-size?     uint32
      +--ro member-applications*         app-flow-ref
      +--ro member-services*             service-sub-layer-ref
      +--ro member-groups*               aggregation-grp-ref
      +--ro member-forwarding-sublayers* forwarding-sub-layer-ref
```
YANG Model: App-flow

```yang
+--rw app-flows
    +--rw app-flow* [name]
        +--rw name                        string
        +--rw app-flow-bidir-congruent?   boolean
        +--ro outgoing-service?           service-sub-layer-ref
        +--ro incoming-service?           service-sub-layer-ref
        +--rw traffic-profile?            traffic-profile-ref

+--rw ingress
    +--rw name?               string
    +--ro app-flow-status?    identityref
    +--rw interface?          if:interface-ref
    +--:(data-flow-type)?
        +--:(ip-app-flow)
        +--:(mpls-app-flow)

+--rw egress
    +--rw name?                 string
    +--:(application-type)?
        +--:(ethernet)
            +--rw ethernet
                +--rw ethernet-place-holder?   string
                +--:(ip-mpls)
                    +--rw ip-mpls
                        +--rw (next-hop-options)
                            +--:(simple-next-hop)
                                +--rw outgoing-interface?    if:interface-ref
                                +--:(next-hop-list)
                                    +--rw next-hop-list
                                        +--rw next-hop* [hop-index]
                                            +--rw hop-index             uint8
                                            +--rw outgoing-interface?    if:interface-ref
```
YANG Model: App-flow

```
+--rw app-flows
  +--rw app-flow* [name]
    +--rw name
    +--rw app-flow-bidir-congruent? boolean
    -rw outgoing-service? service-sub-layer-ref
    -ro incoming-service? service-sub-layer-ref
    +--rw traffic-profile? traffic-profile-ref
    +--rw ingress
    |  +--rw name? string
    |  +--ro app-flow-status? identityref
    |  +--rw interface? if:interface-ref
    |  +--(data-flow-type)?
    |     +--:(ip-app-flow)
    |     |  +--:(mpls-app-flow)
    |     +--rw egress
    |        +--rw name? string
    |        +--(application-type)?
    |           +--:(ethernet)
    |              +--rw ethernet
    |              |     +--rw ethernet-place-holder? string
    |              +--:(ip-mpls)
    |                 +--rw ip-mpls
    |                 +--(next-hop-options)
    |                    +--:(simple-next-hop)
    |                       +--rw outgoing-interface? if:interface-ref
    |                       +--(next-hop-list)
    |                          +--rw next-hop* [hop-index]
    |                             +--rw hop-index uint8
    |                               +--rw outgoing-interface? if:interface-ref
```

Leafref - Service reference

Application identification value

Leafref - Interface reference
YANG Model: Aggregation

```
+--rw service-aggregation-group* [group-name]
  +--rw group-name aggregation-group
  +--rw outgoing
    +--rw traffic-profile? traffic-profile-ref
      +--rw service-protection
        +--rw service-protection-type? service-protection-type
        +--rw sequence-number-length? sequence-number-field
        +--rw aggregation-header
          +--rw mpls-label-stack
            +--rw entry* [id]
              +--rw id uint8
              +--rw label? rt-types:mpls-label
              +--rw ttl? uint8
              +--rw traffic-class? uint8
            +--ro services* service-sub-layer-ref
        +--rw aggregation-header
          +--rw mpls-label-stack
            +--rw entry* [id]
              +--rw id uint8
              +--rw label? rt-types:mpls-label
              +--rw ttl? uint8
              +--rw traffic-class? uint8
            +--ro services* service-sub-layer-ref
```

YANG Model: Service Sub-layer

Grouped or non-grouped

Leafref - Application references ( from App / Aggregate )

Service identification value ( from Service | Forwarding / Disaggregate )

Leafref - Service references ( from Service / Aggregate )

Leafref - Forwarding references ( from Forwarding / Aggregate )

Leafref - Forwarding references ( to Forwarding / Aggregate )

Leafref - Service reference ( to Service / Aggregate )

Leafref - Service references ( to Service / Disaggregate )

Leafref - Forwarding references ( to Forwarding / Disaggregate )
Case A: Aggregation of multiple App-flows at an ingress node

- Case A-1: Multiple App-flows are aggregated into a service sub-layer of DetNet flow
Individual DetNet flow 1 for App-flow 0
Individual DetNet flow 1 for App-flow 1
Case A-1: Ingress node 1 aggregates App flows 0 and 1 into a service sub-layer of DetNet flow 1
Case A1 Instance data

```json
> data -t data -f json case-a-1.xml
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }]
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      }
    ],
    "ietf-detnet-config:detnet": {
      "app-flows": {
        "app-flow": [
          {
            "name": "app-0",
            "app-flow-bidir-congruent": false,
            "outgoing-service": "ssl-1",
            "traffic-profile": 1,
            "traffic-class": 6
          },
          {
            "name": "app-1",
            "app-flow-bidir-congruent": false,
            "outgoing-service": "ssl-1",
            "traffic-profile": 1,
            "traffic-class": 7
          }
        }
      }
    }
  }
}
```
Case A1 Instance data

```
], "traffic-profile": [
{"profile-number": 1,
  "traffic-requirements": {
    "min-bandwidth": "100000000",
    "max-latency": 100000000,
    "max-latency-variation": 200000000,
    "max-loss": 2,
    "max-consecutive-loss-tolerance": 5,
    "max-misordering": 0
  },
  "member-applications": [
    "app-0",
    "app-1"
  ]
},
{"profile-number": 2,
  "traffic-requirements": {
    "min-bandwidth": "200000000",
    "max-latency": 100000000,
    "max-latency-variation": 200000000,
    "max-loss": 2,
    "max-consecutive-loss-tolerance": 5,
    "max-misordering": 0
  },
  "member-services": [
    "ssl-1"
  ]
},
{"profile-number": 3,
  "traffic-specification": {
    "interval": 5,
    "max-packets-per-interval": 10,
    "max-payload-size": 1500
  },
  "member-forwarding-sublayers": [
    "fsl-1"
  ]
],
"service-sub-layer": {
  "service-sub-layer-list": [
    {
      "name": "ssl-1",
      "service-rank": 10,
      "non-grouped": {
        "traffic-profile": 2,
        "service-operation-type": "service-initiation",
        "service-protection": {
          "service-protection-type": "none",
          "sequence-number-length": "long-sn"
        },
        "incoming": {
          "app-flow": [
            "app-0",
            "app-1"
          ]
        },
        "outgoing": {
          "service-outgoing-list": [
            {
              "service-outgoing-index": 0,
              "mpls-label-stack": {
                "entry": [
                  {
                    "id": 0,
                    "label": 100
                  }
                ]
              },
              "next-layer": [
                {
                  "index": 0,
                  "forwarding-sub-layer": "fsl-1"
                }
              ]
            }
          ]
        }
      }
    }
  ]
}
```

Case A1 Instance data

"forwarding-sub-layer": {
  "forwarding-sub-layer-list": [
    {
      "name": "fsl-1",
      "traffic-profile": 3,
      "forwarding-operation-type": "impose-and-forward",
      "incoming": {
        "service-sub-layer": [
          "ssl-1"
        ]
      },
      "outgoing": {
        "outgoing-interface": "eth2",
        "mpls-label-stack": {
          "entry": [
            {
              "id": 0,
              "label": 10000
            }
          ]
        }
      }
    }
  ]
}
Case B: Aggregation of multiple DetNet flows at an ingress node

• Case B-1: Multiple service sub-layers of DetNet flows are aggregated into a forwarding sub-layer

• Case B-2: Multiple service sub-layers of DetNet flows are aggregated into a service sub-layer of a new aggregated DetNet flow
Individual DetNet flow 1
Individual DetNet flow 2

Source 1: 1.1.1.1
Ingress 1: 2.2.2.2
Relay 1: 3.3.3.3
Relay 2: 5.5.5.5
Transit 1: 4.4.4.4
Transit 2: 5.5.5.5
Egress 1: 7.7.7.7
Destination 1: 8.8.8.8

Source 2: 1.1.1.2
Ingress 2: 2.2.2.2
Relay 1: 3.3.3.3
Relay 2: 6.6.6.6
Transit 1: 4.4.4.4
Transit 2: 5.5.5.5
Egress 2: 7.7.7.7
Destination 2: 8.8.8.9
Case B-1: The **service sub-layers** of DetNet flows 1 and 2 are aggregated into a **forwarding sub-layer**.
Case B-2: The service sub-layers of DetNet flows 1 and 2 are aggregated into a service sub-layer of Aggregated DetNet flow 1.

Notes:
- S and A labels in this diagram include d-CWs of their own.
Instance Data Notes

• The merged model currently has two ways to achieve aggregation labels

• The first one does it all in the service sublayer branch and creates a service aggregation label.

• The second one has an aggregation layer for grouping the service sub layers.

• We need to get to one way to do it.

• Instance data for both are shown.
Case B2 Instance data

> data -t data -f json case-b-2.xml

```json
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth3",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      },
      {
        "name": "eth4",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-10-02T23:59:00Z"
        }
      }
    ],
    "ietf-detnet-config:detnet": {
      "app-flows": [
        {
          "name": "app-1",
          "app-flow-bidir-congruent": false,
          "outgoing-service": "ssl-1",
          "traffic-profile": 1,
          "ingress": {
            "interface": "eth0",
            "src-ip-prefix": "1.1.1.1/32",
            "dest-ip-prefix": "8.8.8.8/32"
          }
        },
        {
          "name": "app-2",
          "app-flow-bidir-congruent": false,
          "outgoing-service": "ssl-2",
          "traffic-profile": 1,
          "ingress": {
            "interface": "eth1",
            "src-ip-prefix": "1.1.1.2/32",
            "dest-ip-prefix": "8.8.8.9/32"
          }
        }
      ]
    }
  }
}
```
"traffic-profile": [
    {
        "profile-number": 1,
        "traffic-requirements": {
            "min-bandwidth": "100000000",
            "max-latency": "100000000",
            "max-latency-variation": "200000000",
            "max-loss": "2",
            "max-consecutive-loss-tolerance": "5",
            "max-misordering": "0"
        },
        "member-applications": [
            "app-1",
            "app-2"
        ],
        "member-services": [
            "ssl-1",
            "ssl-2"
        ]
    },
    {
        "profile-number": 2,
        "traffic-requirements": {
            "min-bandwidth": "200000000",
            "max-latency": "100000000",
            "max-latency-variation": "200000000",
            "max-loss": "2",
            "max-consecutive-loss-tolerance": "5",
            "max-misordering": "0"
        },
        "member-services": [
            "ssl-1"
        ]
    },
    {
        "profile-number": 3,
        "traffic-specification": {
            "interval": "5",
            "max-packets-per-interval": "10",
            "max-payload-size": "1500"
        },
        "member-forwarding-sublayers": [
            "afl-1"
        ]
    }
],
"service-sub-layer": {
    "service-sub-layer-list": [
        {
            "name": "ssl-1",
            "service-rank": "10",
            "traffic-profile": 1,
            "service-operation-type": "service-initiation",
            "service-protection": {
                "service-protection-type": "none",
                "sequence-number-length": "long-sn"
            },
            "incoming": {
                "app-flow": ["app-1"
            ]
        },
        {
            "outgoing": {
                "aggregation-service-sub-layer": "asl-1",
                "service-label": {
                    "mpls-label-stack": {
                        "entry": [
                            {
                                "id": "0",
                                "label": "162"
                            }
                        ]
                    }
                }
            }
        }
    ]
}
Case B2 Instance data part 3

```json
[
  {
    "name": "ssl-1",
    "service-rank": 10,
    "traffic-profile": 2,
    "service-operation-type": "service-initiation",
    "service-protection": {
      "service-protection-type": "none",
      "sequence-number-length": "long-sn"
    },
    "incoming": {
      "app-flow": [
        "app-2"
      ]
    },
    "outgoing": {
      "aggregation-service-sub-layer": "asl-1",
      "service-label": {
        "mpls-label-stack": {
          "entry": [
            {
              "id": 0,
              "label": 105
            }
          ]
        }
      }
    }
  },
  {
    "name": "ssl-2",
    "service-rank": 10,
    "traffic-profile": 1,
    "service-operation-type": "service-initiation",
    "service-protection": {
      "service-protection-type": "none",
      "sequence-number-length": "long-sn"
    },
    "incoming": {
      "app-flow": [
        "app-2"
      ]
    },
    "outgoing": {
      "aggregation-service-sub-layer": "asl-1",
      "service-label": {
        "mpls-label-stack": {
          "entry": [
            {
              "id": 0,
              "label": 105
            }
          ]
        }
      },
      "forwarding-sub-layer": "afl-1"
    }
  }
]
```
Case B2 Instance data part 4

```
}, "forwarding-sub-layer": {
  "forwarding-sub-layer-list": [
    {
      "name": "afl-1",
      "traffic-profile": 3,
      "forwarding-operation-type": "impose-and-forward",
      "incoming": {
        "service-sub-layer": [nsl-1]
      },
      "outgoing": {
        "outgoing-interface": "eth2",
        "mpls-label-stack": {
          "entry": [
            {
              "id": 0,
              "label": 20000
            }
          ]
        }
      }
    }
  ]
}
```

Case B2 (Group Model) Instance data

```
> data -t data -f json case-b2-grouped.xml
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth0",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-09-02T23:59:00Z"
        }
      },
      {
        "name": "eth1",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-09-02T23:59:00Z"
        }
      },
      {
        "name": "eth2",
        "type": "iana-if-type:ethernetCsmacd",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2020-09-02T23:59:00Z"
        }
      }
    ],
  },
  "ietf-detnet-config:detnet": {
    "app-flows": {
      "app-flow": [
        {
          "name": "app-1",
          "app-flow-bidir-congruent": true,
          "outgoing-service": "ssl-1",
          "traffic-profile": 1,
          "ingress": {
            "name": "port1",
            "app-flow-status": "ready",
            "interface": "eth0",
            "src-ip-prefix": "1.1.1.1/32",
            "dest-ip-prefix": "8.8.8.8/32",
            "traffic-class": 40
          },
          "egress": {
            "name": "port1",
            "ip-mpls": {
              "outgoing-interface": "eth0"
            }
          }
        },
        {
          "name": "app-2",
          "app-flow-bidir-congruent": true,
          "outgoing-service": "ssl-1",
          "traffic-profile": 1,
          "ingress": {
            "name": "port2",
            "app-flow-status": "ready",
            "interface": "eth1",
            "src-ip-prefix": "1.1.1.1/32",
            "dest-ip-prefix": "8.8.8.9/32",
            "traffic-class": 40
          },
          "egress": {
            "name": "port2",
            "ip-mpls": {
              "outgoing-interface": "eth1"
            }
          }
        }
      ]
    }
  }
}
```
Case B2 (Group Model) Instance data part 2

"traffic-profile": [  
  {  
    "profile-number": 1,  
    "traffic-requirements": {  
      "min-bandwidth": "100000000",  
      "max-latency": 100000000,  
      "max-latency-variation": 200000000,  
      "max-loss": 2,  
      "max-consecutive-loss-tolerance": 5,  
      "max-misordering": 0  
    },  
    "member-applications": [  
      "app-1",  
      "app-2"  
    ]  
  },  
  {  
    "profile-number": 2,  
    "traffic-requirements": {  
      "min-bandwidth": "100000000",  
      "max-latency": 100000000,  
      "max-latency-variation": 200000000,  
      "max-loss": 2,  
      "max-consecutive-loss-tolerance": 5,  
      "max-misordering": 0  
    },  
    "member-groups": [  
      "ag1"  
    ]  
  },  
  {  
    "profile-number": 3,  
    "traffic-specification": {  
      "interval": 5,  
      "max-packets-per-interval": 10,  
      "max-payload-size": 1500  
    },  
    "member-forwarding-sublayers": [  
      "fsl-1"  
    ]  
  }],

"service-aggregation-group": [  
  {  
    "group-name": "ag1",  
    "outgoing": {  
      "traffic-profile": 2,  
      "service-protection": {  
        "sequence-number-length": "long-sn"  
      },  
      "aggregation-header": {  
        "mpls-label-stack": [  
          {  
            "id": 0,  
            "label": 1000  
          }  
        ]  
      }  
    },  
    "services": [  
      "ssl-1",  
      "ssl-3"  
    ]  
  }],
Case B2 (Group Model) Instance data part 3

```
"service-sub-layer": {
  "service-sub-layer-list": [
    {
      "name": "ssl-1",
      "service-rank": 10,
      "grouped": {
        "group-ref": "ag1"
      },
      "service-protection": {
        "service-protection-type": "none",
        "sequence-number-length": "long-sn"
      },
      "service-operation-type": "service-initiation",
      "incoming": {
        "app-flow": ["app-1"]
      },
      "outgoing": {
        "service-outgoing-list": [
          {
            "service-outgoing-index": 0,
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 100
                }
              ]
            },
            "next-layer": [
              {
                "index": 0,
                "forwarding-sub-layer": "fsl-1"
              }
            ]
          }
        ]
      }
    },
    {
      "name": "ssl-3",
      "service-rank": 10,
      "grouped": {
        "group-ref": "ag1"
      },
      "service-protection": {
        "service-protection-type": "none",
        "sequence-number-length": "long-sn"
      },
      "service-operation-type": "service-initiation",
      "incoming": {
        "app-flow": ["app-2"]
      },
      "outgoing": {
        "service-outgoing-list": [
          {
            "service-outgoing-index": 0,
            "mpls-label-stack": {
              "entry": [
                {
                  "id": 0,
                  "label": 101
                }
              ]
            },
            "next-layer": [
              {
                "index": 0,
                "forwarding-sub-layer": "fsl-1"
              }
            ]
          }
        ]
      }
    }
  ]
}
```
"forwarding-sub-layer": {
  "forwarding-sub-layer-list": [
    {
      "name": "fsl-1",
      "traffic-profile": 3,
      "forwarding-operation-type": "impose-and-forward",
      "incoming": {
        "service-sub-layer": [
          "ssl-1",
          "ssl-3"
        ]
      },
      "outgoing": {
        "outgoing-interface": "eth2",
        "mpls-label-stack": {
          "entry": [
            {
              "id": 0,
              "label": 10000
            }
          ]
        }
      }
    }
  ]
}
}
Case C: Aggregation of multiple DetNet flows at a relay node

• Case C-1: Multiple forwarding sub-layers of DetNet flows are aggregated into a forwarding sub-layer
• Case C-2: Multiple service sub-layers of DetNet flows are aggregated into a forwarding sub-layer
• Case C-3: Multiple service sub-layers of DetNet flows are aggregated into a service sub-layer of a new aggregated DetNet flow
• Case C-4: Multiple forwarding sub-layers of DetNet flows are aggregated into a service sub-layer of a new aggregated DetNet flow
Individual DetNet flow 1
Individual DetNet flow 2

**Diagram Description:**

- **Source 1**: 1.1.1.1
- **Ingress 1**: 2.2.2.2
- **Relay 1**: 3.3.3.3
- **Transit 1**: 4.4.4.4
- **Relay 2**: 5.5.5.5
- **Transit 2**: 6.6.6.6
- **Egress 1**: 7.7.7.7
- **Destination 1**: 8.8.8.8

**Data Flow:**
- **DATA 2**
  - **IP**: 1.1.1.2
  - **MPLS S-label**: 104
  - **MPLS F-label**: 10007

**MPLS Attributes:**
- **S-label**: 104
- **F-label**: 10007

**Routing:**
- **App-2**
- **DN-1**
- **DN-2**

**Additional Details:**
- **Source 2**: 1.1.1.2
- **Ingress 2**: 2.2.2.3
- **Relay 2**: 3.3.3.3
- **Transit 2**: 5.5.5.5
- **Egress 2**: 6.6.6.6
- **Destination 2**: 8.8.8.9

**Additional MPLS Details:**
- **S-label**: 103
- **F-label**: 10011
- **S-label**: 105

**Routing:**
- **App-2**
- **DN-2**

**Additional Data:**
- **DATA 2**
  - **IP**: 1.1.1.2
  - **MPLS S-label**: 103
  - **MPLS F-label**: 10006

**MPLS Attributes:**
- **S-label**: 103
- **F-label**: 10006

**Routing:**
- **App-2**
- **DN-2**
Case C-1: Relay node 1 aggregates the forwarding sub-layers of DetNet flows 1 and 2 into a forwarding sub-layer

Note: S-label in this diagram includes d-CW.
Case C-2: Relay node 1 aggregates the service sub-layers of DetNet flows 1 and 2 into a forwarding sub-layer.
Case C-3: Relay node 1 aggregates the service sub-layers of DetNet flows 1 and 2 into a service sub-layer of Aggregated DetNet flow 1.

Note: S and A labels in this diagram include d-CWs of their own.
Case C-4: Relay node 1 aggregates the forwarding sub-layers of DetNet flow 1 and 2 into a service sub-layer of Aggregated DetNet flow 1.

Note: S and A labels in this diagram include d-CWs of their own.
Case D: Aggregation of multiple DetNet-flows at a transit node

• Case D-1: Multiple forwarding sub-layer of DetNet flows are aggregated a forwarding sub-layer
Individual DetNet flow 1
Individual DetNet flow 2
Case D-1: Transit node 1 aggregates the **forwarding sub-layers** of DetNet flow 1 and 2 into a **forwarding sub-layer**.
Plan

• Comments and reviewing the current Model.
  • Terminology consistency
  • Double check Flow Model consistency
  • Removing unnecessary cases.

• General clean up
  • Add references
  • Include Yanglint sample configuration
Thanks