rtgwg
Internet-Draft
Intended status: Informational
Expires: September 9, 2020

S. Homma
T. Nakamura
NTT
X. de Foy
InterDigital Inc.
A. Galis
University College London
LM. Contreras
Telefonica
R. Rokui
Nokia
P. Martinez-Julia
NICT

March 8, 2020

Gateway Function for Network Slicing
draft-homma-rtgwg-slice-gateway-02

Abstract

This document describes the roles and requirements for a slice gateway that is a function or function group for handling data plane traffic, such as connecting/disconnecting and compose/decompose network slice subnet instances and providing network slices from end to end. The interworking between management and control elements at the management and control planes with the gateway function for controlling and orchestrating end-to-end network slices are also presented in this document.

Table of Contents

1. Introduction

Network slicing is an approach to create separate virtual networks in support of service depending on several requirements on the same physical resources, and it enables networks to adapt to requirements, which is diverse more, inexpensively and flexibly. The overview is introduced in [Slicing_Tutrial] and [NECOS].

It's also expected to enhance usability of infrastructural networks for tenants and create new business opportunities. For example, by using network slices lent from infrastructure operators, other industrial companies can provide communication services including ensurance of network transport without having physical infrastructure.

From a business point of view, a slice includes a combination of all the relevant network resources, functions, and assets required to fulfill a specific business case or service, including OSS, BSS and DevOps processes.

From the network infrastructure point of view, network slice requires the partitioning and assignment of a set of resources that can be used in an isolated, disjunctive or non- disjunctive manner for that slice.

From the tenant point of view, network slice provides different capabilities, specifically in terms of their management and control capabilities, and how much of them the network service provider hands over to the slice tenant. As such there are two kinds of slices: (A) Inner slices, understood as the partitions used for internal services of the provider, retaining full control and management of them. (B) Outer slices, being those partitions hosting customer services, appearing to the customer as dedicated networks.

Network slices are established with combination of various technologies, such as software defined network (SDN), network function virtualization (NFV), or traffic engineering, and managed/ operated with automation technologies such as orchestrator.

Assumed use cases of network slices include establishment of virtual networks whose qualities are guaranteed from end to end under the supervision of multi-domain orchestrators. In such cases, a network slice subnet is created on each domain, such as access network and core network, and an end-to-end network slices is composed of connected subnets.

Network slice subnets are built based on specifications of the underlay network, and thus the used technologies might vary. Therefore, a gateway function, which enables to connect subnets while adapting the differentiations and forward data packets to/from the appropriate next subnet, is required.

This document describes the gateway function for network slicing, called slice gateway or SLG, and its role and requirements. Note that defining a new data plane technology is not a goal of this draft. In addition, this draft aims to specify management-related requirements for an SLG, which may be implemented using existing data plane technologies.

2. Definition of Terms

This section describes definitions and terminologies related to network slicing, especially gateway function and interconnection network slices established in each domain. Other complementary definitions are described in [I-D.homma-slice-provision-models].

Network Slicing: Network slicing is a methodology to create separate logical networks in support of services, depending on several requirements, on the same physical resources. This is possible by combinations of several network technologies.

Network Slice (NS): An NS is a logical separate network that provides specific network capabilities and characteristics. It is

composed of set of resources including virtual/physical forwarding functions, links, and network functions. There are several types of NS depending on network types where the NS is deployed. Transport slice is one of them and the detailed definition is provided in [I-D.rokui-teas-transport-slice-definition]. (Note that 3GPP and other SDOs use definitions "Network Slice Instance/ NSI" and "Network Slice Subnet Instance/NSSI", but they are expressed as NS in this document.)

Network Slice Instance (NSI): An NSI is a logical network instance composed with required infrastructure resources, including networking (WAN), computing (NFVI) resources, and some include additional network service functions such as firewall or load-balancer. It is composed of one or more Network Slice Subnet Instances. Note that, the word "instance" is not used in the definition of transport slice discussed in the NS-DT (Network Slice Design Team).

Network Slice Subnet: An NS subnet is a representation of a set of resources structuring a part of NSI within a single domain. Network slice subnet concept is proposed in [TS.28.530-3GPP]. Note that, in the definition of transport slice discussed in the NS-DT, Network Slice and Network Slice Subnet are not distinguished, and all types of network slices in transport network are called transport slice.

Network Slice Subnet Instance (NSSI): An NSSI is a partial logical network instance represented as a network slice instance. It is a minimal unit managed or provided as a network slice. One or more NSSI structure an NSI or an E2E-NSI.

End-to-End Network Slice Instance (E2E-NSI): An E2E-NSI is an NS providing connectivity among end points. An E2E-NS is used for emphasizing end to end connectivity provided by an NS.

Network Slice as a Service (NSaaS): An NSaaS is a service delivery model in which a third-party provider (e.g., vertical customer) hosts NSs and makes them available to customers. In this model, there are mainly two roles: NS provider and NS tenant.

Network Slice Provider (NS Provider): An NS provider is a person or group that designs and instantiates one or more NSIs/NSSIs, and provides them to NS tenants. In some cases, an NS provider is an infrastructure operator simultaneously. This includes NSI, NSSI, and E2E-NSI providers.

Network Slice Tenant (NS Tenant): An NS tenant is a person or group that rents and occupies NSs from NS providers.

Domain: A domain is a group of a network and devices administrated as a unit with common rules and procedures.

Administrative Domain: An administrative domain is a group of networks and devices managed by an administrator.

Resource: A resource is element used to create virtual networks. There are several types of resources, i.e., connectivity, computing and storage.

Network Function Virtualization (NFV): NFV is the concept or technologies to provide dedicated network appliances as software.

Software Defined Network (SDN): SDN is the concept or technologies to separate network control plane from data plane, and control network devices dynamically and flexibly.

Virtual Network: A virtual network is a network running a number of virtual network functions. The detailed definition is provided in [RFC8453].

Virtual Network Function (VNF): A virtual network function (VNF) is a network function whose functional software is decoupled from hardware. One or more virtual machines running different software and processes on top of industry-standard high-volume servers, switches and storage, or cloud computing infrastructure, and capable of implementing network functions traditionally implemented via custom hardware appliances and middleboxes (e.g., router, NAT, firewall, load balancer, etc.)

Slice Gateway Function (SLG): An SLG is a function or a group of functions to connect/disconnect NSSIs. The roles are described in the following sections.

Business Support System and Operation Support System (BSS/OSS): BSS/OSS are systems to support service providing and operation of network devices.

Orchestrator: Orchestrator is an entity to operate network components automatically. There are several types of orchestrators including NFV Orchestrator (NFVO) or service orchestrator defined by ETSI NFV and Open Source MANO (OSM) ([NFV-Architectural-Framework] and [OSM-White-Paper]).

SLG Controller (SLG-Ctrl): An SLG-Ctrl is an entity that controls SLGs. An SLG-Ctrl is controlled by upper-level operation systems such as OSS/BSS or orchestrator.

3. Motivations and Roles of SLG

One of the main roles of SLG is the enablement of interworkings between data plane with management and control elements for controlling and orchestrating end-to-end slices.

Use cases of network slices are discussed in several Standard Developing Organizations (SDOs). Some examples are described in use cases document ([I-D.netslices-usecases]).

In some proposed use cases, an NS is structured across multiple network domains. The capability of NSSIs might be different because the components are domain-specific. In particular, the differentiation in capability between different administrative domains is large.

Moreover, several variations can be considered on NS provisioning in NSaaS (ref. [I-D.homma-slice-provision-models]), and some cases need abstraction of underlay infrastructure to NS tenants. SLG solution provides controllability of network functions for manipulation of NSs intensively, and it can be expected to emphasize the manageability of NSIs in such cases.

For connecting some different NSSIs and providing a NS that guarantees the prescribed quality from end to end, SLGs are required to connect such NSSIs. SLGs enable to provide E2E-NSIs independently of specifications of underlay networks by hiding the differentiations and connecting between NSSIs. An overview of this concept is shown in Figure 1. SLGs glue NSSIs established on each domain and provide an E2E-NSI.

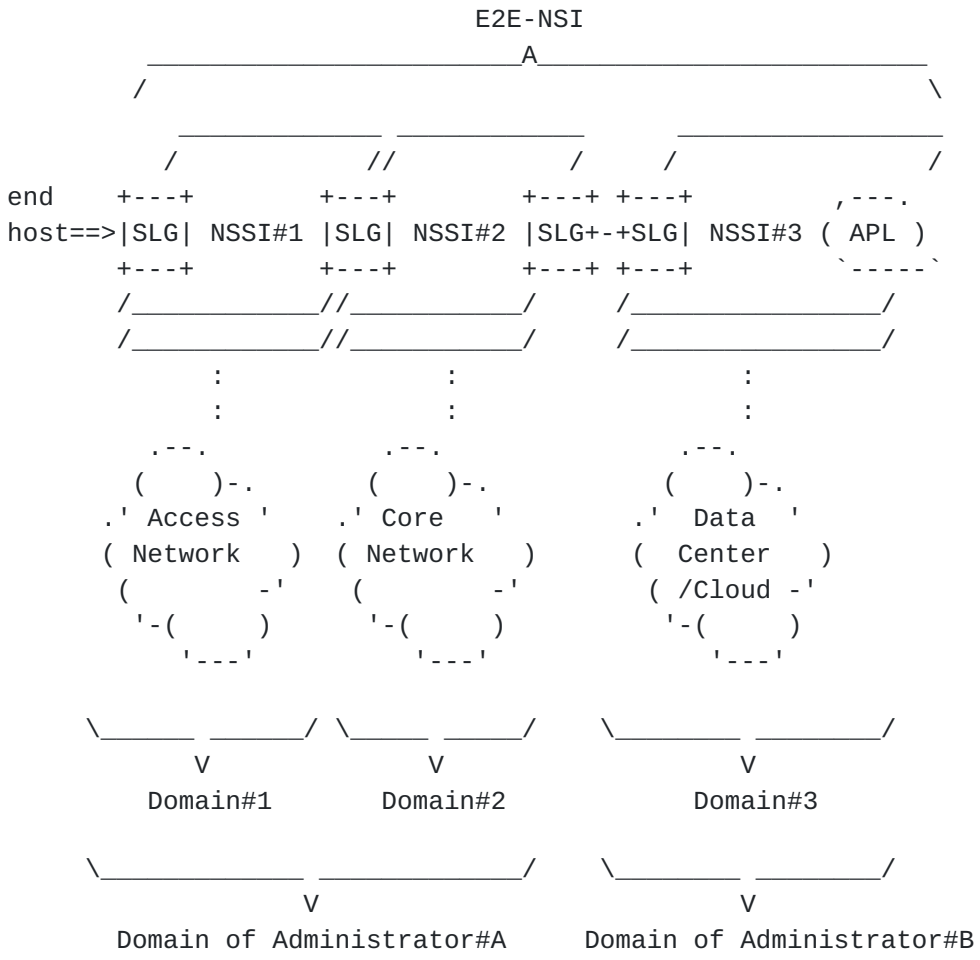

Figure 1: E2E-NSI composed of multiple NSSIs

Moreover, identification of user service traffic and their allocation/disallocation to the appropriate NSSI are required at the edges of E2E-NSIs, as shown in Figure 2, and SLGs might take on these roles.

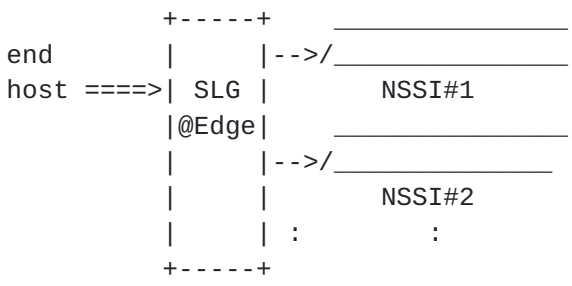

Figure 2: NSSI selection of SLG

Note that, this model has the assumption that transitions of data packets from one NSSI to another are executed at only SLGs. Also, an SLG is not necessarily implemented as a single device or virtual machine (VM).

4. Architecture of Network Slicing System

NSs are composed of several (virtual) network functions and links, and the characteristics of each NS are based on the assumed service. Also, some of NSs are deployed across multiple administrative domains. For deploying the appropriate NSs based on each service requirements, a management system, which enables to control network resources totally within a domain, and interaction between such management systems are required.

An SLG is a network function, and SLGs are installed at edge of NSSIs. NSs are dynamically created, deleted, and moved depending on requests from network operator or NS tenants. Therefore, some SLGs would be required to be VNF for flexible deployment.

This section describes overview of NS management system architecture (Section 4.1) . It refers [NFV-Architectural-Framework] and [OSM-White-Paper] for management of whole of NS and VNFs, and [RFC8453] and [I-D.ejj-teas-ns-framework] for transport network/slice manipulation.

4.1. Network Slice Management System Architecture

The architecture overview of NS management system is shown in Figure 3.

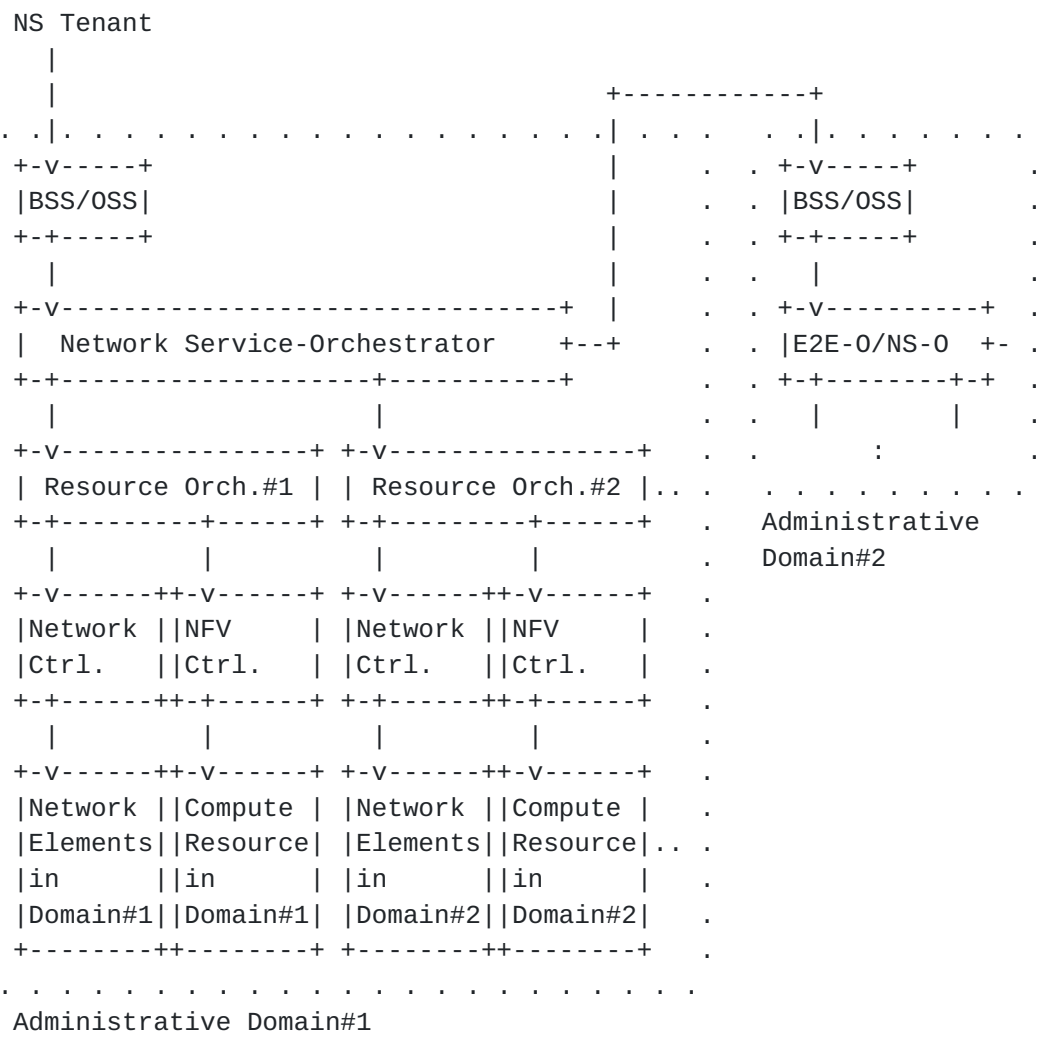

Figure 3: Overview of NS Management Architecture

Orchestrators manage whole resources including network elements and compute resources (i.e., routing, bandwidth, network functions). In this figure, the resources are managed by resource orchestrators installed in each domain, and the network service orchestrator operates resource orchestrators. A resource orchestrator includes modules for handling each resource type, such as NFVO (ref. [OSM-White-Paper]) and Transport Slice Controller/TSC (ref. [I-D.ejj-teas-ns-framework]). These orchestrators have recursive structure, and a network service orchestrator sometimes control other network service orchestrators in other administrative domains.

NSs are requested from NS tenants via BSS/OSS and the order to create an NS is given to orchestrators. Orchestrators manipulate network elements and compute resouces via controllers. When an NS across multiple administrative domains are requested, the network service

orchestrator request orchestrators in other administrative domains to create NSSIs.

SLGs are also controlled via orchestrators. An SLG can be implemented in a network element, or may be hosted on a compute resource if it runs as a VNF.

5. Requirements for SLG

An SLG is basically a component in the data plane and has the roles of data packet processing. Moreover, it is required to have functions for control/management processes such as connecting to underlay networks or managing NSs.

Furthermore, an SLG might be required to support handling services provided on NSs in addition to controlling of NS because an SLG is an edge node on an E2E-NSI.

In this section, we describe the requirements for an SLG in terms of the following aspects and their interworkings.

1. Data plane for NSs as infrastructure

2. Control/management plane for NSs as infrastructure

3. Data plane for services on NSs

4. Control/management plane for services on NSs

5.1. Management of NS as Infrastructure

5.1.1. Data Plane Aspect

5.1.1.1. Identification/Classification

SLGs at the edge of E2E-NSs MUST have the capability to identify and classify data packets, and assign them to the appropriate E2E-NS. This requirement varies depending on the location.

Fixed Access: An SLG MUST identify and classify data packet with access point, including CPE or WiFi-AP, or subscriber ID such as VLAN-ID. Moreover, in some services, an SLG should identify and classify data packets based on user device or application used in the communication.

Mobile Access: An SLG MUST identify and classify data packet with subscriber-ID such as IMSI, radio-wave bandwidth, or identifier of tunnels. Moreover, in some services, an SLG should identify and

classify data packets based on application used in the communication or location of the user equipment (UE).

Connection Point between NSSI: An SLG MUST identify and classify data packet based on the tunnel-ID or virtual routing and forwarding (VRF) that received the packets. If specific slice identifier such as a value mapped in the metadata field of the IP header is used; an SLG should identify and classify data packets with the ID.

5.1.1.2. Transporting/Forwarding

SLGs MUST provide functions for transport data packets depending on the specifications of the underlay networks.

Encapsulation/Decapsulation/Tagging: In network slicing, duplication of IP addresses of user packets between NSs MUST be accepted, thus, using techniques that enable separation of a network logically is preferred. In short, some tunnel protocols or tagging approaches should be used as transport of NSs. For this reason, SLG MUST support encapsulation or tagging of data packets based on the specification of the underlay network. Also, SLG MUST support the packets' decapsulation or untagging. Examples of tunnel protocols and tags that can be used for creating NSs on L2/L3 segments are described below.

L2 Segment: VLAN, MPLS, Segment Routing MPLS (SR-MPLS), PPPoE, etc.

L3 Segment: GRE, L2TP, GTP-U, VxLAN, IPv6 Segment Routing (SRv6), etc.

VxLAN, SR-MPLS, and SRv6 are described in their specification documents ([RFC7348], [I-D.ietf-spring-segment-routing-mpls], and [I-D.ietf-6man-segment-routing-header]).

Translation of Encapsulation/Tagging Form: SLG MUST support to translate tunnel header or tag of received packets to the appropriate tunnel header or tag when it forwards data packets to the next NSSI that has different transport capability.

Distribution of Traffic: Some NSs have multiple route between the same end points within the same NSSI because of traffic engineering, switching to a redundant path, or other reasons, and SLG MAY forward data packets with the appropriate route based on some trigger information. An example of the overview of this

requirement is shown in Figure 4. In this figure, there are two routes, main and sub, between SLGs, and an SLG switches forwarding route depending on the network situation such as congestion occurrence on the current route.

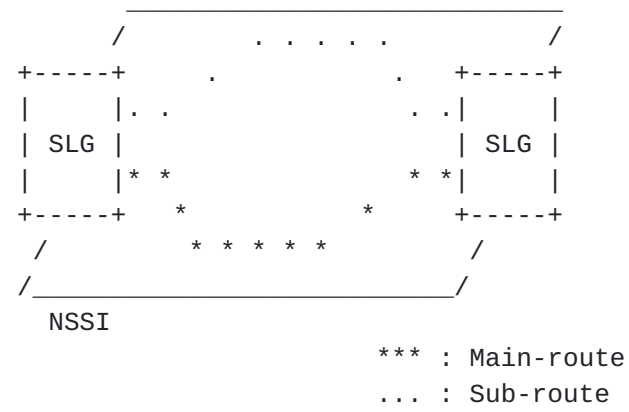

Figure 4: An example of traffic distribution by SLG

5.1.1.3. Isolation among NSs

In NSaaS, isolation control is required for avoiding an NS being affect by other NSs. Traffic engineering or QoS control is ones of the most fundamental approaches to prevent disturbances among NSs.

Traffic Shaping/Policing: An SLG MUST execute traffic shaping and policing at its egress and ingress ports to avoid an NS using excessive traffic bandwidth.

Quality of service (QoS) Control: If there is an order of priority between NSs on the same underlay infrastructure, an SLG should remark the appropriate QoS parameter of the outer-most header of each packet following the preconfigured setting and provide packet scheduling based on the QoS parameter for providing priority control. The field that SLG refers may vary depending on the specification of the underlay network. For example, COS value is remarked in L2 segments; on the other hand, DSCP value is remarked in L3 segments. In mobility networks standardized by 3GPP, QoS class is managed by using QoS Class Identifier (QCI) or 5G QoS Identifier (5QI) conveyed in extention header of user plane protocol, and they mapped to DSCP ([I-D.henry-tsvwg-diffserv-to-qci]).

5.1.1.4. Service Chaining as Infrastructural Mechanism(*Optional)

If an SLG is composed of a combination of several components, a service chaining mechanism is required to make them work together and achieve SLG functionality.

Moreover, some NSs may traverse NFVs such as firewalls or cache servers for providing value-added services to their users. In such cases, SLG might be required to support service chaining mechanisms, such as handling of network service header (NSH) defined in [RFC8300]. If an NS includes the service chaining architecture defined in [RFC7665], some SLG would be required to support following functions; classifier(CF), service function forwarder (SFF), and inter boundary node(IBN). (Details of CF, SFF and IBN are described in SFC documents; [RFC7665], [RFC8459].)

5.1.2. Control/Management Planes Aspects

5.1.2.1. Interfaces to Controllers or Operation Systems

SLG MUST have interface to its controller or operation systems for set parameters related to the data plane functions described in Section 5.1.1. In addition, an SLG at the edges of E2E-NSs MUST have interfaces to authentication servers.

5.1.2.2. Address Resolution/Routing

An SLG MUST support address resolution or routing mechanisms to connect to underlay network elements including routers or L2 switches.

5.1.2.3. Authentication Authorization Accounting (AAA)

For preventing entry of irregular traffic to NSs, an SLG at the edge of E2E-NS MUST support AAA mechanism for incoming traffic. Also, when an SLG connects to another SLG in other administrative domain, SLGs should have a mechanism to confirm that the connection is established with the regular processes. For example, an SLG is required to support authentication of the opponent SLG with key information indicated from higher-level operation systems.

5.1.2.4. Operation Administration and Maintenance(OAM)

In management of NSs, OAM mechanisms for both underlay and overlay networks is required for SLGs. For an underlay network, an SLG MUST have OAM functions to confirm connectivity to interconnect equipment. For an overlay network as an NS, an SLG MUST have OAM functions to confirm connectivity to the nodes on the same NS.

5.1.2.5. Traffic Monitoring

An SLG shall support monitoring of traffic amount and latency as a mechanism for checking whether each of the accomodated NSs is satisfying its SLO. When an NS can't fulfill its SLO, the SLG MUST send a notification to any listening system. A use case where the traffic monitoring of SLGs is used for such closed loop schem is introduced in [I-D.pedro-nmrg-intelligent-reasoning]

5.2. Management of Services on NS (*Optional)

5.2.1. Data Plane Aspect

5.2.1.1. Identification/Classification

In NSaaS, some NS tenants may need delivery of an individual service to each user, device, or application on the same NS. For such service deliveries, an SLG might be required to identify and classify user traffic based on some information such as subscriber ID or payload of data packets. Also, an SLG should be controllable from the NS tenant.

5.2.1.2. QoS Control

An NS accommodates several communication devices and SLGs might be required to have fair queueing mechanisms for maintaining service quality of each user. Also, different types of service traffic that have different priorities might coexist on an NS. For example, some NS providers might provide telephone and internet access services to their users with an NS. In such cases, SLG might be required to provide QoS control mechanisms for enforcing priority control based on service priorities.

These QoS controls are executed depending on the information of inner packets and are independent of isolation mechanisms as infrastructure. An SLG might be required to have a hierarchical QoS control mechanism in case that both QoS controls for services over NSs and isolation between NSs are required.

5.2.1.3. Steering/Service Chaining(Cooperation with VNFs)

SLG might be required to support steering or service chaining function for conveying data packets to the appropriate network functions deployed on an NS based on the classification result and user's contract information.

5.2.2. Control/Management Planes Aspects

5.2.2.1. Interfaces to Service Management Systems

An SLG might have interfaces to controllers for managing user policies on each NS. Some controllers might be deployed on the same NS. If some controllers are located at external networks, they might require SLGs to have APIs.

5.2.2.2. Collection of Telemetry information

In an NSaaS, collection of telemetry information of each NS might be required for understanding traffic usage. Thus, an SLG might be required to support to collect and report telemetry information of connected NSs.

6. Structure of SLG

SLG is composed of data plan entities and controller. SLG Data plane entity (SLG-D) has functions to manipulate NSSIs and to handle traffic on slices. In some cases, an SLG-D is composed of several physical devices and/or virtual instances. Function types supported by SLG-D are listed Section 5. SLG controller (SLG-C) accomodates multiple SLG Data plane entities via its southbound interface, and sets configurations into each SLG-D. SLG-C also has a northbound interface(NBI) and it provides accesses to two endpoints: higher operation systems such as orchestrator, and to external customers and tenants which own their NSSIs. Then, functionality and controllability exposed to customers/tenants ashould be limited, and the access must be secure (i.e., authentication and admission control should be supported). The overview of SLG structure is shown in Figure 5.

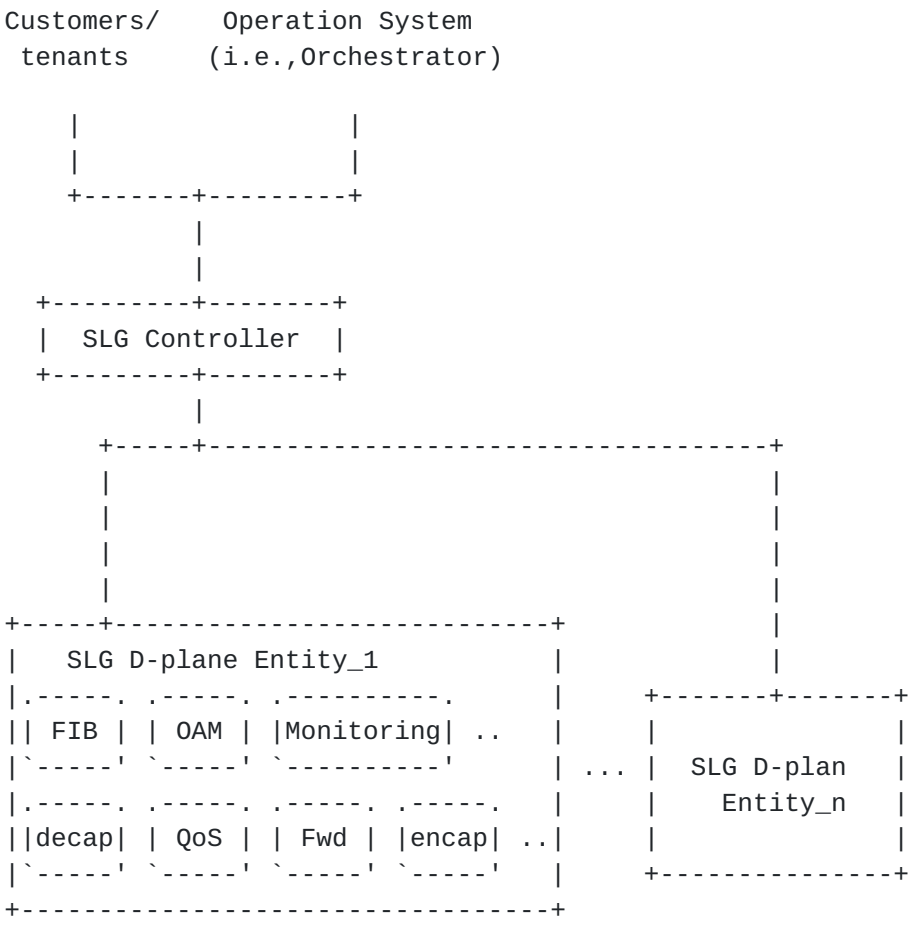

Figure 5: Overview of SLG Structure

An example of data model for customers/tenants is shown in Appendix B

7. Deployment of SLG

This section describes considerations related with deployment of SLGs.

7.1. Examples of Components Required to Maintain SLG Functions

For providing E2E-NSIs on existing network infrastructures, some components located at boundaries of domains are required to have the same set of functionality as an SLG. Examples of such components in each domain type are described below.

Fixed Network: CPE/HGW, Service Edge, Gateway Router, etc.

Mobile Network: User Equipment, Radio-AP, eNodeB, S/P-GW ([TS.36.300-3GPP]), etc.

Data Center: Gateway Router, L2 switch, ToR switch, Server, etc.

7.2. SLG Types Depending on Locations on NS

There are mainly three types of SLG for creating E2E-NSI across multiple administrative domains. The requirements of each SLG type are listed in Appendix A.

7.2.1. Edge SLG(E-SLG)

E-SLG is located at an edge of an E2E-NSI, and supports identification, classification and authentication of user traffic in addition to fundamental SLG functions, such as transport and isolation. Also, it might be required to have capabilities for services delivered on an NS.

7.2.2. Inter-Subnet SLG(IS-SLG)

IS-SLG is located between NSSIs within a single administrative domain and has only fundamental functions such as QoS control or tranlation of headers.

This type of SLG enables to separate an NSI into some NSSIs. It will provide modularities of NSSIs, and simplify the management of NSIs. However, it is not necessarily required if a common transport mechanism in all domains is used.

7.2.3. Inter-Domain SLG(ID-SLG)

ID-SLG is located between NSSIs established on different domains. It supports authentication for connecting to the opponent SLG in addition to fundamental functions.

7.3. Horizontal Connection

The connection form of an SLG varies depending on which type it is. Examples of horizontal connection forms of each SLG type are described below.

E-SLG: An E-SLG accommodates several hosts and NSSIs. This has a forwarding table of end hosts and insert their packets to the appropriate NSSI. An overview of this connection is shown in Figure 6.

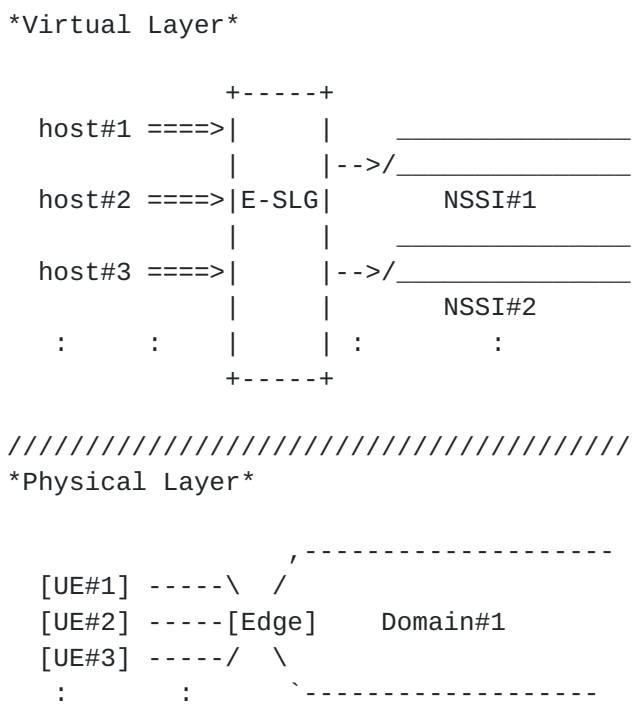

Figure 6: Overview of horizontal connection of E-SLG

IS-SLG: An IS-SLG has the role of mediator between NSSIs and passes packets received from an NSSI to the next one. If transport methods used in each domain are different, the IS-SLG translate packet form to the appropriate one. An overview of this connection is shown in Figure 7.

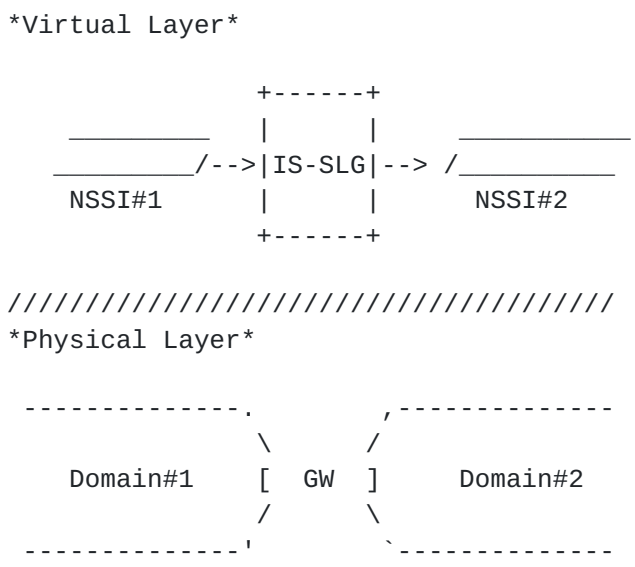

Figure 7: Overview of horizontal connection of IS-SLG

ID-SLG: An ID-SLG passes data packets to another ID-SLG located on a different administrative domain. Some tunnel established between them in advance may be used for the passing of packets. An overview of this connection is shown in Figure 8.

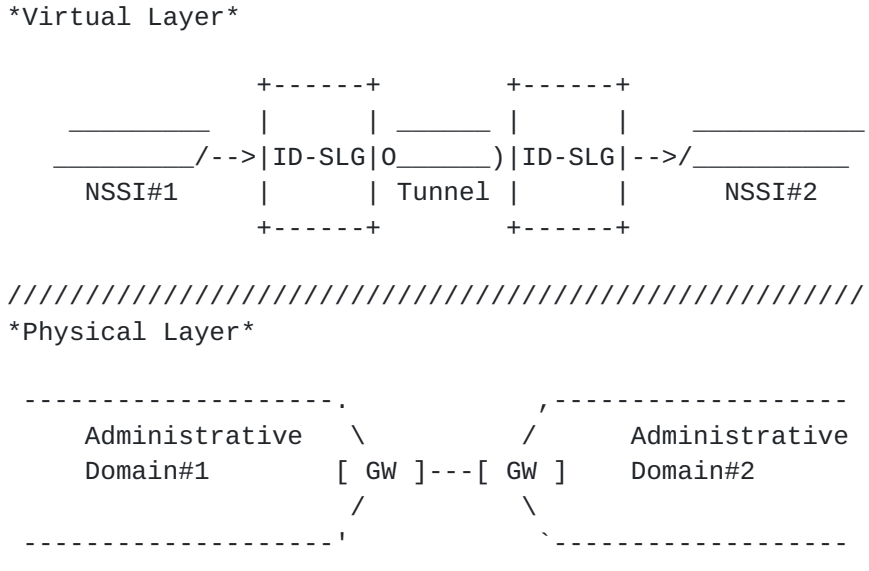

Figure 8: Overview of horizontal connection of ID-SLG

7.4. Vertical Connection

There are two patterns of vertical connection of SLGs in the middle of E2E-NSs. The first pattern is that the SLGs accommodate only a set of NSSIs, which are composition of the same E2E-NS. In this pattern, such SLGs are not required to support NSSI selection, however, establishment of a new SLG is required when a new E2E-NS is created. This might causes extra overheads because of deploying many SLGs.

The other pattern is that such SLGs are acceptable to accommodate multiple NSSIs from each domain. In this pattern, SLGs support NSSI selection. On the other hand, this pattern can restrain the number of SLGs. Also, it is easy to provide transit of data packets from an NSSI to another NSSI on the same domain.

The overviews of these patterns are shown in Figure 9 and Figure 10.

Figure 9: Overview of vertical connection of SLG: Separated Pattern

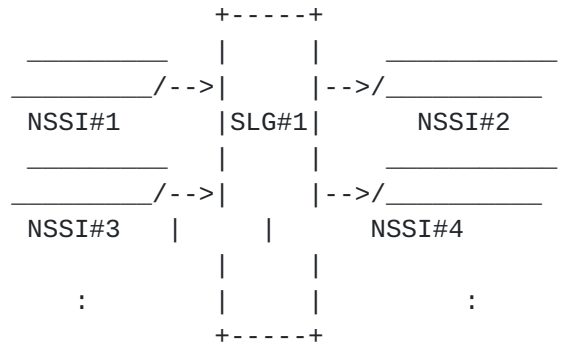

Figure 10: Overview of vertical connection of SLG: Shared Pattern

8. Interconnection between NSSIs

SLG provides interconnectivity between NSSIs. The concept and fundamental framework including the related NS information model are described in NSSIs interconnection document ([I-D.defoy-coms-subnet-interconnection]).

This section is focused on interconnection between NSSIs established on different administrative domains, and describes considerations related to this condition.

8.1. Pre-arrangement of transport protocols

For interconnection between different administrative NSSIs, pre-arrangement of the transport protocol, which is used to connect between SLGs is required. Orchestration systems indicate the protocol and configuration to each SLG.

8.2. Quality Assurance between SLGs

In addition to establishing connection, quality control of communication is important. SLGs of egress side should execute traffic shaping to prevent some NSs from excessively occupying the link between SLGs. Moreover, some SLGs are connected to several other SLGs that are deployed on the different locations. Therefore SLGs of the ingress side should execute traffic policing to avoid excessive inflow of traffic into some NSs. The parameters for these controls are pre-configured by orchestration systems.

The above approaches are ones of the simplest ways to provide quality assurance of inter-administrative subnets. If there is stricter isolation request, more considerations would be required.

8.3. Secure Interconnection

For connecting networks of different administrators, secure interconnection schemes are required. Especially, in an NSaaS, networks might be connected to several networks, and schemes for ensuring secure connectivity would be more important.

SLGs confirm whether the opponent SLG is regular when it requests to connect, and reject the request if the SLG is not regular. In some cases, SLGs might be confirm whether the inner packets received from the other SLGs are sent from regular users.

9. Interfaces of SLG Controller

9.1. Southbound Interface

SLG-C supports protocols to communicate with SLG-Ds. Information and prameters exchanged between SLG-D and SLG-C are TBD.

9.2. Northbound Interface for Higher Operation Systems

TBD

9.3. Northbound Interface for Customers/Tenants

SLG-C can provide some capabilities, such as NS allocation and obtaining of NS status, for customers/tenants to handle their own NSs. An example of data models for this interface is shown in Appendix B.

10. Security Considerations

Requirements and considerations for SLG related to security are described in Section 5 and Section 8.

11. IANA Considerations

This memo includes no request to IANA.

12. Acknowledgement

The authors would like to thank Li Qiang for her kind review and valuable feedback.

13. Informative References

[I-D.defoy-coms-subnet-interconnection]
Foy, X., Rahman, A., Galis, A., kiran.makhijani@huawei.com, k., and L. Qiang, "Interconnecting (or Stitching) Network Slice Subnets", draft-defoy-coms-subnet-interconnection-01 (work in progress), October 2017.

[I-D.henry-tsvwg-diffserv-to-qci]
Henry, J., Szigeti, T., and L. Contreras, "Diffserv to QCI Mapping", draft-henry-tsvwg-diffserv-to-qci-03 (work in progress), February 2020.

[I-D.homma-slice-provision-models]
Homma, S., Nishihara, H., Miyasaka, T., Galis, A., OV, V., Lopez, D., Contreras, L., Ordonez-Lucena, J., Martinez-Julia, P., Qiang, L., Rokui, R., Ciavaglia, L., and X. Foy, "Network Slice Provision Models", draft-homma-slice-provision-models-00 (work in progress), February 2019.

[I-D.ietf-6man-segment-routing-header]
Previdi, S., Filsfils, C., Raza, K., Dukes, D., Leddy, J., Field, B., daniel.voyer@bell.ca, d., daniel.bernier@bell.ca, d., Matsushima, S., Leung, I., Linkova, J., Aries, E., Kosugi, T., Vyncke, E., Lebrun, D., Steinberg, D., and R. Raszuk, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-08 (work in progress), January 2018.

[I-D.ietf-spring-segment-routing-mpls]
Filsfils, C., Previdi, S., Bashandy, A., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing with MPLS data plane", draft-ietf-spring-segment-routing-mpls-11 (work in progress), October 2017.

[I-D.netslices-usecases]
kiran.makhijani@huawei.com, k., Qin, J., Ravindran, R., Geng, L., Qiang, L., Peng, S., Foy, X., Rahman, A., Galis, A., and G. Fioccola, "Network Slicing Use Cases: Network Customization and Differentiated Services", draft-netslices-usecases-02 (work in progress), October 2017.

[I-D.pedro-nmrg-intelligent-reasoning]
Martinez-Julia, P. and S. Homma, "Intelligent Reasoning on External Events for Network Management", draft-pedro-nmrg-intelligent-reasoning-01 (work in progress), March 2020.

[I-D.rokui-5g-transport-slice]
Rokui, R., Homma, S., Lopez, D., Foy, X., Contreras, L., Ordonez-Lucena, J., Martinez-Julia, P., Boucadair, M., Eardley, P., Makhijani, K., and H. Flinck, "5G Transport Slice Connectivity Interface", draft-rokui-5g-transport-slice-00 (work in progress), July 2019.

[NECOS] NECOS, "Novel Enablers for Cloud Slicing", <hhttp://www.h2020-necos.eu>.

[NFV-Architectural-Framework]
Network Functions Virtualisation (NFV) ETSI Industry Specification Group (ISG), "Network Functions Virtualisation (NFV); Architectural Framework", December 2014, <http://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.02.01_60/gs_NFV002v010201p.pdf>.

[OSM-White-Paper]
ETSI, "OSM White Paper", October 2016, <https://osm.etsi.org/images/OSM-Whitepaper-TechContent-ReleaseONE-FINAL.pdf>.

[RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014, <https://www.rfc-editor.org/info/rfc7348>.

[RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", RFC 7665, DOI 10.17487/RFC7665, October 2015, <https://www.rfc-editor.org/info/rfc7665>.

[RFC8300] Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

[RFC8453] Ceccarelli, D., Ed. and Y. Lee, Ed., "Framework for Abstraction and Control of TE Networks (ACTN)", RFC 8453, DOI 10.17487/RFC8453, August 2018, <https://www.rfc-editor.org/info/rfc8453>.

[RFC8459] Dolson, D., Homma, S., Lopez, D., and M. Boucadair, "Hierarchical Service Function Chaining (hSFC)", RFC 8459, DOI 10.17487/RFC8459, September 2018, <https://www.rfc-editor.org/info/rfc8459>.

[Slicing_Tutrial]
IEEE NetSoft2018, "Network Slicing Landscape Tutorial", June 2018, <http://netsoft2018.ieee-netsoft.org/program/tutorials/; http://discovery.ucl.ac.uk/10051374/>.

[TS.23.501-3GPP]
3rd Generation Partnership Project (3GPP), "3GPP TS 23.501 (V16.0.0): System Architecture for 5G System; Stage 2", September 2018, <http://www.3gpp.org/ftp//Specs/archive/23_series/23.501/23501-g00.zip>.

[TS.28.530-3GPP]
3rd Generation Partnership Project (3GPP), "3GPP TS 28.530 (V1.0.0): Management and orchestration of networks and network slicing; Concepts, use cases and requirements (work in progress)", June 2018, <http://ftp.3gpp.org//Specs/archive/28_series/28.530/28530-100.zip>.

[TS.36.300-3GPP]
3rd Generation Partnership Project (3GPP), "Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2", December 2007, <http://www.qtc.jp/3GPP/Specs/36300-830.pdf>.

Appendix A. Requirements for each SLG Type

The requirements for each SLG type are listed in Figure 11.

	E-SLG	IS-SLG	ID-SLG	Reference
*Data-Plane of NS as Infrastructure				
Identification/Classification	M	O	O	Section 5.1.1.1.
Transport/Forwarding	M	O	M	Section 5.1.1.2.
Isolation	M	M	M	Section 5.1.1.3.
Service Chain	O	O	O	Section 5.1.1.4.
*Control/Management-Plane of NS as Infrastructure				
IF to Ctrl/OpS	M	M	M	Section 5.1.2.1.
Addr Resolution/Routing	M	M	M	Section 5.1.2.2.
AAA	M	-	M	Section 5.1.2.3.

OAM		M	M	M	Section 5.1.2.4.
Monitoring		M	M	M	Section 5.1.2.5.
*Data-Plane for Service on NS					
Identification/ Classification		O	-	O	Section 5.2.1.1.
QoS Control		O	O	O	Section 5.2.1.2.
Steering/ Service Chain		O	-	O	Section 5.2.1.3.
*Control/Management-Plane for Service on NS					
IF to Service Manager		O	O	O	Section 5.2.2.1.
Telemetry		O	O	O	Section 5.2.2.2.

M: Mandatry, O: Optional, - : Not Required

Figure 11: List of Requirements for each SLG

Appendix B. Example of Data Model for Service Management

An SLG provides customers/tenants some capabilities: NS allocation, change of forwarding policy for user traffic on the target NS, reporting telemetry information of the target NS, etc. Examples of tree diagram of data model for service management is shown in Figure 12.

```
module: slg-svc-man
 +--rw ns-allocation
    +--rw current-ns-id?
    |  +--rw ns-id? string
    |     +--rw nssi-id*        string
    +--rw flow-id
    |  +--rw ipv4?      boolean
    |  |  +--rw src-ipv4-address?       inet:ipv4-address
    |  |  +--rw dst-ipv4-address?       inet:ipv4-address
    |  +--rw ipv6?      boolean
    |  |  +--rw src-ipv6-prefix?        inet:ipv6-prefix
    |  |  +--rw dst-ipv6-prefix?        inet:ipv6-prefix
```

```
   |  +--rw src-port?          inet:port-number
   |  +--rw dst-port?          inet:port-number
   |  +--rw protocol?          uint8
   |  +--rw option-tag?
   |     +--rw tag-id* string
   +--rw new-ns-id?    string
         +--rw ns-id?
            +--rw nssi-id*     string


+--rw uflow-policy
   +--rw ns-id?   string
   |  +--nssi-id*     string
   +--rw flow-id?
   |  +--rw ipv4?     boolean
   |  |  +--rw src-ipv4-address?      inet:ipv4-address
   |  |  +--rw dst-ipv4-address?      inet:ipv4-address
   |  +--rw ipv6?     boolean
   |  |  +--rw src-ipv6-prefix?       inet:ipv6-prefix
   |  |  +--rw dst-ipv6-prefix?       inet:ipv6-prefix
   |  +--rw src-port?         inet:port-number
   |  +--rw dst-port?         inet:port-number
   |  +--rw protocol?         uint8
   |  +--rw option-tag?
   |     +--rw tag-id*        string
   +--rw forwarding-policy
      +--rw traffic-class             inet8
      +--rw guaranteed-rate-value?    uint64
      +--rw permissible-jitter-value? uint64
      +--rw service-function-path-id? inet24

+--ro telemery-report
   +--ro ns-id?   string
      +--ro nssi-id*  string
         +--ro flow-id*
            +--ro packet-received?    uint64
            +--ro packet-sent?        uint64
            +--ro bytes-received?     uint64
            +--ro bytes-sent?         uint64
            +--ro latency?
               +--ro endpoint-id*     string
```

Figure 12: Tree Diagram of Data Model for Service Management

Appendix C. Complementation of Network Slicing in 3GPP

The 3GPP 5GS is natively support network slicing (ref. [TS.23.501-3GPP], and UPF provides some functions as SLG, such as NS selection, QoS control, traffic steering, etc. 3GPP is responsible for standardizing user plane manipulation for mobility management, and interworking with transport on underlay network and external networks of 5GS such as DNs is out of scope in 3GPP.

SLG concept will provide complemental definitions of functions and interfaces for providing E2E-NSI including 5GS. A way of interworking between transport slice and RAN/UPF is described in [I-D.rokui-5g-transport-slice]. In this case, RAN/UPF and transport slice endpoints connected to them behave as ID-SLGs.

Another way for interworking is RAN/UPF support functionalities of SLG for transport slice. However, this is not supported in 3GPP standards, and may inhibit multi-vendor implementation.

Authors' Addresses

Shunsuke Homma
NTT
Japan

Email: shunsuke.homma.fp@hco.ntt.co.jp

Takayuki Nakamura
NTT
Japan

Email: takayuki.nakamura.gy@hco.ntt.co.jp

Xavier de Foy
InterDigital Inc.
Canada

Email: Xavier.Defoy@InterDigital.com

Alex Galis
University College London
United Kingdom

Email: a.galis@ucl.ac.uk

Luis M. Contreras
Telefonica
Ronda de la Comunicacion, s/n
Sur-3 building, 3rd floor
Madrid 28050
Spain

Email: luismiguel.contrerasmurillo@telefonica.com
URI: http://lmcontreras.com/

Reza Rokui
Nokia
Canada

Email: reza.rokui@nokia.com

Pedro Martinez-Julia
NICT
Japan

Email: pedro@nict.go.jp