Michalopoulos, D. S., et al. Network slicing via function decomposition and flexible network Design, in *28th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'17),* 8-13 October 2017, Montreal, QC, Canada

DOI: https://doi.org/10.1109/PIMRC.2017.8292661

Network Slicing via Function Decomposition and Flexible Network Design

Diomidis S. Michalopoulos*, Mark Doll*, Vincenzo Sciancalepore†,
Dario Bega‡§, Peter Schneider*, and Peter Rost*
*Nokia Bell Labs Germany, †NEC Europe Ltd., ‡University Carlos III Madrid, §IMDEA Networks Institute

***Abstract*—We argue for flexible network design as an architecture prototype for next generation networks. Such flexible design is developed by capitalizing on the concept of network function decomposition in conjunction with with its relation to network slicing. A detailed view of the proposed functional architecture is put forward, where the role of specific network function blocks for forming network slices with given requirements is underlined. We further highlight the impact of common architecture over multiple tenants and elaborate on the emerging business models associated with multi-tenancy together with the resulting implications on security.**

I. Introduction

The next generation of the 3GPP mobile access system is supposed to support not only a single new radio access technology (RAT) but a multitude of RATs, e. g., 3GPP Long Term Evolution (LTE), 5G "New Radio" as well as non-3GPP RATs such as IEEE 802.11 or satellite communication [1]. Furthermore, a diversity of services with partly contradicting requirements mandate for a highly flexible mobile network architecture that supports coexistence of multiple RATs in a single system architecture.

A. Network slicing in 5G

One of the key enablers for a flexible architecture is *network slicing*. The concept of network slicing was originally proposed by Next Generation Mobile Networks (NGMN) alliance [2]. NGMN defines a *Network Slice Instance*, in the following simply *slice*, as "a set of network functions, and resources to run these network functions, forming a complete instantiated logical network to meet certain network characteristics required by the Service Instance(s)." A *Service Instance*, in the following short *service*, denotes "an instance of an end-user service or a business service that is realized within or by a Network Slice" [2]. Network slicing allows for implementing and running different services independently from each other in different slices and with a distinct set of resources. Hence, it is an enabler to support highly diverse services on a single infrastructure while fulfilling quality of service (QoS) guarantees for each service.

B. Flexible network architecture

One possibility to exploit network slicing is to provide customized network operation for each network slice, which may be offered through parameterization or flexible network function composition. In the case of the latter option, a logical network would be implemented as a set of individual function blocks, each optimized for the particular usage. A slice then becomes a composition of these function blocks into a chain or more generally into a network of function blocks. Decomposition into function blocks enables sharing of network functions among slices for reuse and consistency among slices, or where common resources must be shared. A slice may be partly composed of a set of common function blocks to be shared across slices and a set of dedicated function blocks that implement customized and optimized functionality of a slice. Furthermore, decomposition enables the function blocks of a slice to be placed according to its service needs and the concrete deployment scenario, i. e., the available execution environments such as distributed (edge) or centralized resources.

C. State of the Art

In the conventional 3GPP Evolved Packet System (EPS), functions are grouped a-priori running as independent entities, e. g., enhanced Node B (eNB), Serving Gateway (S-GW), or Mobility Management Entity (MME). Therefore, a specific function placement is already designed into the system, namely co-locating all functions assigned to such an entity. This results in static function assignments preventing rapid network re-configurations or on-demand service deployments. This has driven several academic institutions and industrial partners towards full flexible solutions, which aim at *(i)* identifying and decomposing network elements into multiple fine-grained basic network functions [3] and *(ii)* optimally placing and inter-connecting these basic network functions. The former objective introduces several challenges, as tightly coupled network functions may require additional (non-standardized) interfaces to properly run in different network entities. The latter accounts for routing and optimal placement solutions to efficiently handle diverse service requirements. An example is provided in [4], wherein the network function orchestration (and placement) problem with the objective of optimizing operational costs and utilization, without violating service-level agreements (SLAs), is described with an Integer Linear Programming (ILP) formulation, further solved through heuristic solutions. In [5], a game theoretic approach is formulated to simplify (and control) the interaction between the service tenant, asking for a specific set of network functions with given requirements, and the network provider, optimally

This work has been performed in the framework of the H2020-ICT-2014-2 project 5G NORMA. The authors would like to acknowledge the contributions of their colleagues. This information reflects the consortiums view, but the consortium is not liable for any use that may be made of any of the information contained therein.

deploying such functions to fulfil service SLAs. Routing problems between network functions are faced in [6], where prediction schemes are used to cope with delay issues. In addition, [7] proposes a dynamic routing function deployment model for satisfying demands of different applications by means of the software-defined networking (SDN) paradigm. Nevertheless, none of the above works directly addresses the decomposition and placement problem leveraging on the network slice concept.

II. Proposed Functional Architecture

A. Overview and main Benefits

In the proposed architecture, the above introduced concepts of network slicing and function decomposition are embraced and combined with the notion of software-defined mobile network control (SDMC). In fact, SDMC extends the SDN paradigm beyond mere packet forwarding to put the (almost) full set of mobile network functions under a centralized control.

The main benefit of this approach is a complete network programmability, in the sense that network functions can be easily adapted to the requirements of network slices in a dynamic fashion. This is contrary to existing "static" architectures, where network functions are pre-configured both in terms of their operation and in terms of their physical location. Such network programmability is achieved via proper interfacing to the decomposed functions, thereby allowing for composing the required end-to-end functionality when and where is needed.

B. Flexible design via NF decomposition

Fig. 1 depicts the proposed control and user plane functional architecture, where radio access network (RAN) slicing is applied based on a common Medium Access Control (MAC) layer approach [11]. Further slicing options will be described in Section IV. Functions are categorized based on their control plane or user plane features. They are further classified into distributed, common, and dedicated functions per slice or control application.

Distributed control functions are implemented as virtual network functions (VNFs) throughout the network. The tight coupling to the user plane functions under their control poses stringent latency requirements and implies massive multiplicity of state, which renders them non-eligible to the SDMC concept [15]. Common and dedicated control functions, in contrast, run as applications on top of the north-bound interface of SDMC shared controller (SDM-X) and slice-specific SDMC dedicated controller (SDM-C), respectively. SDM-X applications solely control functions shared by multiple slices whereas SDM-C applications control functions dedicated to individual slices. A negotiation process between those entities is required to ensure that possibly conflicting decisions from different slices are resolved.

The depicted user plane functions resemble the standard 3GPP LTE user plane up to layer 2. Non-access stratum (NAS) functions have been summarized into a single NAS function block. The user plane has been decomposed into a set of function blocks, mostly following LTE protocol layers. Function blocks are interconnected by SDN-enabled transport where different function placement for the individual function blocks is foreseen. As explained in Section III in detail, functional decomposition allows for composing slice-individual function chains that implement services provided by each slice. At the same time, it allows for reusing and possibly sharing with other slices as many function blocks as desired, as well as optimizing the use of an individual protocol layer. In this context, a specific PDCP flavor can be optimized to support new compression and ciphering schemes, e. g., a stateless packet compression with block cipher specifically for sensor devices. The implementation of a future version can omit support for previous versions, thereby avoiding the overhead and complexity that typically comes with maintaining backwards compatibility.

C. Slicing from different standpoints: SDM-C and SDM-X

While the functional split is fundamental for composing, instantiating, and dynamically optimizing network chains to specific service requirements, SDMC controllers act as mediators between service requirements and network facilities. In our approach, the concept of SDMC for controlling network functions belonging to a specific slice is introduced. A single SDM-C is instantiated per network slice, able to analyze service requirements and to properly issue management operations (e. g., function placements, scaling activities) through southbound interfaces, which directly interact with VNFs or physical network functions (PNFs). The SDM-C enforces QoS constraints by monitoring continuously performance figures within the slice, e. g., latency and throughput. In the case of SLA violations, the SDM-C may issue alert messages towards the management entity, promptly performing a function chain re-orchestration of the given slice. This introduces flexibility in the network design because the network operator can easily create a new network slice optimized for a particular network service while fulfilling tenant requirements through SDM-C applications running on top of the network controller.

The access to common network functions and resources is controlled by the SDM-X. Examples are radio resources (spectrum), analog and mixed signal processing from antenna arrays to analog-digital conversion (lower PHY) and baseband processing. Due to the scarcity of radio resources, advanced resource management solutions are needed on top of the SDM-X. Regular interactions between SDM-X and SDM-C result in resource allocation and management policies, which aim at maximizing the process efficiency. In particular, SDM-C conveys a resource demand request to the SDM-X based on performance metrics such as latency, throughput, resilience, or reliability. These messages may result in different network function placement, resource allocation, or computational resource usage. The SDM-X may accommodate such requests while avoiding service degradation. Alternatively, it can decline the request and propose to update shared service

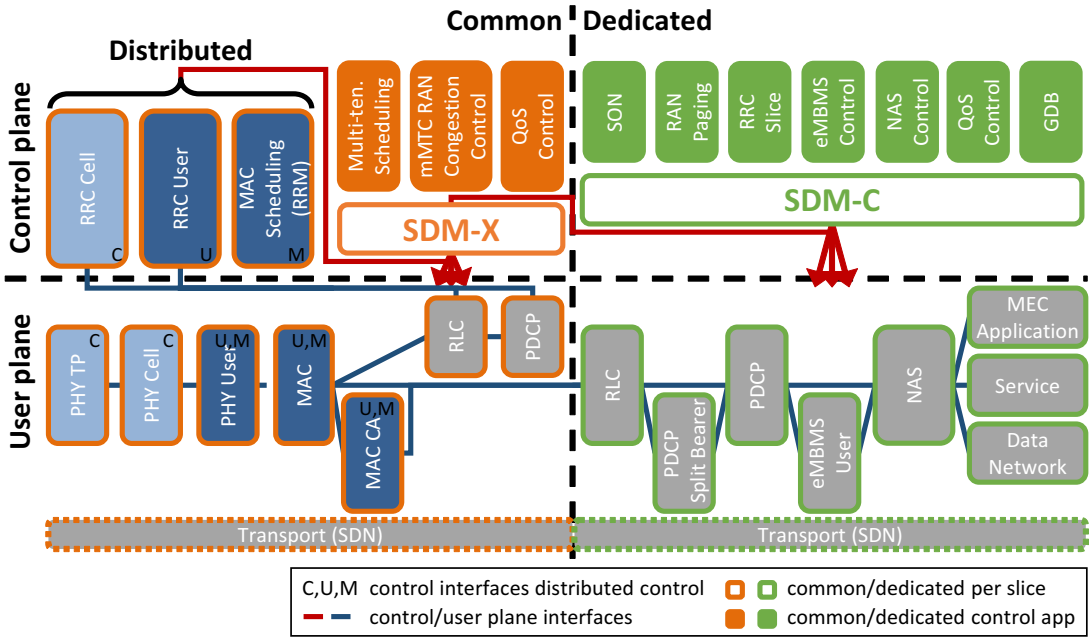

Fig. 1. The proposed control and user plane functional architecture.

requirements based on the current resource availability. While the details of this interaction are left out of the scope of this paper, three examples of network slicing operations and placements are discussed in the ensuing section.

III. Network Slicing Leveraging Flexible Network Design

A fundamental feature of network slicing is the ability to offer independent network instances which support services with distinct requirements. In this regard, the flexible network design discussed in Section II can facilitate the realization of the network slicing concept in the RAN. In particular, the decomposition of network functions into smaller elements allows them to be effectively used on demand, thereby assigned to network instances with distinct characteristics.

In the following, we elaborate on the conceptual connection between flexible architecture and network slicing. With emphasis on the data layer, we distinguish two levels of flexible function allocation: *(i) Function selection*, reflecting which functions are included per slice, and *(ii) function placement*, associated to where functions are located. We illustrate these two levels of flexible function allocation by mirroring them onto three major slice types, namely enhanced mobile broadband (eMBB), low latency, and mission critical slice. A detailed view is provided in Fig. 2.

A. eMBB slice

1) Function selection: The eMBB slice serves applications which are associated with high data rate transmissions. As a result, it involves network functions that facilitate increasing the throughput. In the context of function groups presented in Section II, the eMBB slice would involve the PDCP split-bearer and MAC CA function blocks from the user plane domain. Combinations of the different transmission legs are possible, in the sense that the MAC carrier aggregation function can be applied to one or more components of the bearer split at PDCP. An exemplary realization of such flexible function allocation for the eMBB slice is shown at the left part of Fig. 2.

2) Function placement: Fig. 2 also depicts an exemplary architecture scenario in terms of function placement. In particular, the PDCP and PDCP split-bearer blocks are located at the edge cloud, facilitating thus the implementation of multi-connectivity and minimizing mobility signaling to the core network [8]. Contrarily, RLC and lower layers of the protocol stack need to be co-sited at the radio access node, since their synchronous interaction requires inter-layer communication with very low latency.

B. Low latency slice

1) Function selection: In contrast to eMBB, applications with low latency requirements are in principle not demanding in data rates[1]. As a result, multi-connectivity is not foreseen for such applications and is therefore excluded (cf. Fig. 2). It is important to note that for ultra low latency requirements, functions such as the outer automatic repeat request (ARQ) in RLC can be excluded, hence the RLC would operate in the unacknowledged mode.

2) Function placement: The tight latency requirements for this slice push towards its implementation as close to the radio access as possible. To this end, the use of the edge cloud is not foreseen, implying that even the PDCP functions need to

[1]With the exception of virtual reality services, low latency applications are usually associated with machine-type packets of bursty nature, whose size is considerably smaller than the typical MBB packet size.

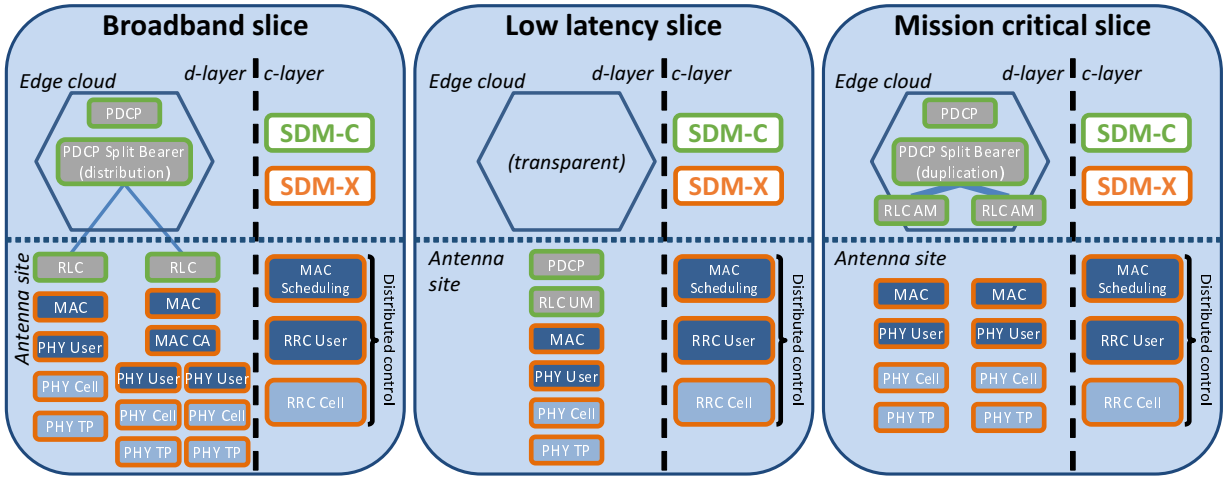

Fig. 2. Flexible function allocation illustrated in the context of the considered network slices.

be executed at the radio access as well. This results in the edge cloud being "unused" to the low latency slice.

C. Mission critical slice

1) Function selection: The basic differentiation of this service type with respect to the previously discussed is the demand for achieving ultra high reliability, necessary for mission critical services. Such reliability levels are in principle hard to attain with existing standards, implying that new access techniques should be employed. In this regard, the data duplication method has been proposed as a special case of multi-connectivity, which transmits multiple replicas of the message via independent links [8], [9]. This involves modifying the PDCP functionality such that the data stream is not split into multiple streams (as is the case in the eMBB slice). Instead, data streams are duplicated and coordinated in the sense that correctly transmitted messages are omitted from further duplication. Note that the mission critical slice entails the RLC acknowledged mode (RLC AM) function block, contrary to unacknowledged mode used for the low latency slice.

2) Function placement: With reference to Fig. 2, the execution of the PDCP and RLC functionalities is envisioned to take place at the edge cloud. The main reason for such consideration is the centralized coordination capabilities offered by such consideration. Specifically, since the duplicated streams need to be jointly coordinated, a distributed implementation of the RLC AM functionality would result in large signaling overhead between the involved nodes.

D. Control plane considerations

Control plane-related functions do not require optimal placement schemes. However, a proper engagement facilitates basic functionalities in order to run independently each slice while fulfilling given requirements. In Fig. 2, we have highlighted the connections between the control plane functions and each of the above-mentioned function blocks, categorized per slice type. These examples describe the fundamental connections between control-plane and data-plane functions while running isolated and shared slices. Nonetheless, they can be easily extended for advanced slice settings.

Each of the control function blocks comprises a set of control functionalities. The MAC scheduling function block is in charge of managing radio resources, with the radio resource control (RRC) Cell and RRC User block handling all per cell and per user state, respectively. This includes system broadcast and RRC signaling messages, as radio resource management (RRM) adapts the radio interface based on current load and radio channel conditions to improve throughput, latency, reliability or energy efficiency of the radio interface. To provide more flexibility, we consider using SDMC controllers described in Section II, namely SDM-X and SDM-C, as a means to run applications controlling common and dedicated functions, respectively. An example is represented by self-organizing network (SON) applications, e. g., distributed-SON functions, which can be used to easily perform configuration and optimization operations in low-latency slices from the SDM-C perspective. Multi-tenancy scheduling and control is another interesting application, which is investigated in the next section.

IV. Multi tenancy: Common architecture over multiple tenants

Besides enabling flexibility, network slicing realizes the notion of *multi tenancy networks*. This notion corresponds to dynamic infrastructure sharing which allows mobile operators to reduce the deployment and operational costs (CAPEX and OPEX) involved in initial roll-out of their networks, as well as to increase the network utilization.

A. Multi-tenancy network

The enhanced capabilities of dynamic sharing opens new business models which involve new players in the mobile network ecosystem. The following players are recognized as main entities in a multi tenancy network:

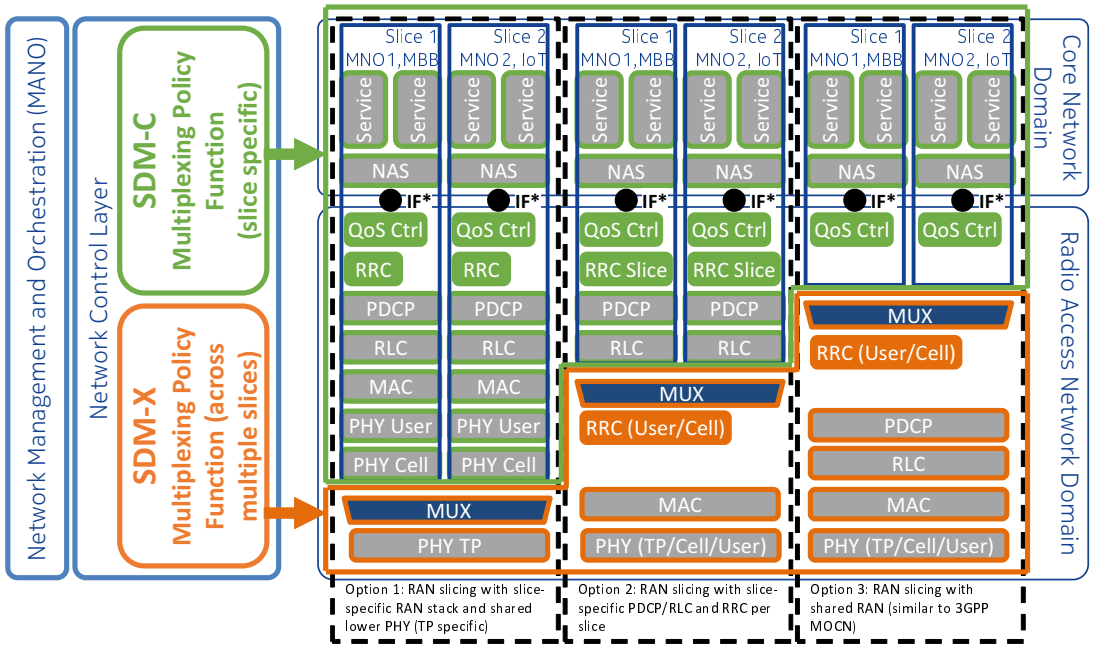

Fig. 3. Network slicing visualized via different levels of network sharing across multiple tenants

- **Mobile Network Operator (MNO)** who operates and owns the infrastructure of the mobile network providing Internet connectivity and telecommunication services to subscribers,
- **Network Slice Tenant** who can be either Mobile Virtual Network Operators (MNOs) as Wi-Fi first operators, or an enterprise, e. g. from a vertical industry. The network slice tenant acquires a network slice from the MNO and leveraging on the proposed flexible architecture to deliver a specific service to its customers.

A network that provides multi-tenancy has to support the capacity to dynamically share the network's resources among tenants, in order to optimize and maximize the resource utilization according to SLA requirements. This calls for new criteria that allow the MNO *(i)* to decide whether to accept or reject the tenant request, so as to optimize the resource utilisation [12] or maximize the overall profits [13], and *(ii)* to control resource sharing either centralized or in a distributed way. Such criterion not only allocates resources to tenants in a fair manner, but also shares the resources of each tenant fairly among its users taking into account the different pricing levels according to the tenants' needs and the numbers and locations of each tenant's users [14].

In the proposed network prototype, these new algorithms are implemented on top of SDM-X by means of the Multi-tenancy Scheduling application, as depicted in Fig. 1. This application is responsible for controlling and managing the resources acquired by each tenant and the interactions among them, through its interfaces with SDM-X and SDM-C.

The network slices owned by different tenants may be instantiated with a different number of common function blocks. Fig. 3 presents three different levels of network sharing: Option 1, where the two tenants share only the lower physical layer (PHY); Option 2, where also the MAC layer is shared; and Option 3 where all the RAN blocks (except QoS Scheduling) are shared. Apparently, this leads to new security aspects which have to be taken into account. A detailed analysis of the security aspects related to a multi-tenancy network is provided below.

B. Security aspects

1) Slice isolation: When a network supports multiple tenants by creating tenant specific network slice instances, it is necessary to isolate these slice instances in a way that one tenant is not aware of the other tenants and has no means to access or even modify information in the other tenants' slices. In Network Function Virtualization (NFV) environments, this type of isolation is a basic feature that also includes the capability to limit the resource usage of each tenant slice instance in a well-defined way. This prevents a tenant from using so many resources that other tenants cannot get resources anymore and thus experience a Denial of Service (DoS).

Tenant isolation in NFV environments is endangered by vulnerabilities in the NFV software, for instance in hypervisors. However, assuming that the relevant NFV software is designed, implemented, configured and operated with highest care to minimize the number of errors and thus the vulnerability, tenant isolation can be achieved in the edge cloud and in NFV environments at access points.

Multi-tenancy is not restricted to infrastructure that provides an NFV platform, but also affects non-virtualized "bare metal" RAN equipment. Depending on the nature of the equipment, it may or may not be aware of the different tenants. In the former case, equipment specific mechanisms need to facilitate multi-tenancy and provide proper isolation.

For example, a radio scheduler implemented on bare metal equipment may be configurable to ensure certain amounts of radio resources for each of several different slices. Naturally, the radio scheduler will not mix up data belonging to different radio bearers, so that it maintains isolation between those radio bearers and consequently between the different slice instances.

2) Slice specific security policies: Isolated network slices facilitate the implementation of individual security policies. In the RAN, this mainly affects access stratum (AS) security policies. As an example, different network slices may offer a different choice of crypto algorithms, or different preferences regarding which algorithm to choose. As another example, some slices may enforce encryption and maybe even integrity protection of the user plane, while others may allow an unprotected user plane. Such an individual security setup per slice can be leveraged by making the PDCP handling a slice specific function, rather than a common function.

3) Trust relationships: As mentioned above, a likely scenario leveraged by network slicing is that an MNO provides individual network slice instances for verticals. The MNO can provide isolation between the tenant slices as described above. However, a vertical as a tenant typically has no means to verify the effectiveness of the isolation, but must trust the MNO to ensure it. Moreover, if the MNO controls the infrastructure, the MNO is able to access everything that is processed on this infrastructure. Consequently, the MNO must be *trusted* to not illegally access a tenant's traffic. Trust in the MNO is also required concerning the correct resource assignment, since a tenant has no practical means to monitor the correct assignment of edge cloud infrastructure resources or radio interface resources to the tenant's slice.

4) Protecting common functions: Common functions that are operated by an MNO and used by several tenants must be protected. Any internal interfaces of such functions must not be accessible for tenant functions. Only dedicated, carefully secured interfaces must be available to tenants. Such interfaces may need to be subject to access control, which includes authenticating the slices accessing the common functions, as well as authorizing their requests. An example could be a tenant requesting certain QoS parameters for a radio bearer. In this case, a common function may check whether the request is covered by the tenant's SLA. Moreover, tenants may inadvertently misuse or even deliberately try to abuse interfaces exposed by common functions. To mitigate this threat, such interfaces must be designed and implemented with high care to minimize their vulnerability.

In this context, it is important to take into account the way the split between common functions and slice-specific functions is implemented. If, for example, the Packet Data Convergence Protocol (PDCP) layer is a common function, then it would have access to the AS keys of all slices. This means that any successful attack against the common PDCP function could also affect the security of all slices. In contrast, if AS keys are maintained in individual PDCP instances within the slices and the common function operates below PDCP, then a compromise of this common function does not reveal the cleartext data for all radio bearers that use encryption, thereby providing a higher level of security.

V. Conclusion

Flexible network design offers the possibility to cope with a diverse set of requirements, as well as supporting multiple RATs in a unified architecture. This work unveiled the potential of flexible network design when this is employed in conjunction with network slicing. Emphasis was put on network function decomposition and its applicability on deployments involving network slicing. Relevant topics arising from this unified architecture paradigm, such as multi-tenancy and security implications on the network slicing architecture, were discussed.

References

[1] 3GPP TR 23.799 *Study on Architecture for Next Generation System (Release 14)*, December 2016.

[2] NGMN Alliance, *Description of Network Slicing Concept*, Jan 2016.

[3] M. R. Sama, X. An, Q. Wei, S. Beker, *"Reshaping the mobile core network via function decomposition and network slicing for the 5G Era,"* in *Proc. of IEEE WCNC*, Apr 2016.

[4] M. F. Bari, S. R. Chowdhury, R. Ahmed, and R. Boutaba, *"On orchestrating virtual network functions,"* in *Proc. of 11th IEEE International Conference on Network and Service Management (CNSM)*, Nov 2015.

[5] S. D'Oro, L. Galluccio, S. Palazzo, G. Schembra, *"A Game Theoretic Approach for Distributed Resource Allocation and Orchestration of Softwarized Networks,"* in *IEEE Journal on Selected Areas in Communications (JSAC)*, Jan 2017.

[6] K. Kawashima, T. Otoshi, Y. Ohsita, M. Murata, *"Dynamic placement of virtual network functions based on model predictive control,"* in *Proc. of IEEE/IFIP NOMS*, Apr 2016.

[7] C. Bu, X. Wang, M. Huang, K. Li, *"SDNFV-based Dynamic Network Function Deployment: Model and Mechanism,"* in *IEEE Communications Letters*, Jan 2017.

[8] A. Ravanshid et al., *Multi-connectivity functional architectures in 5G*, IEEE International Conference on Communications Workshops (ICC), Kuala Lumpur, Malaysia, 2016.

[9] D. S. Michalopoulos, I. Viering, and L. Du, *User plane aspects of multi-connectivity in 5G*, IEEE International Conference on Telecommunications (ICT), Thessaloniki, Greece, 2016.

[10] EU H2020 5G NORMA, Deliverable D4.1, *RAN architecture components intermediate report*, Available: https://5gnorma.5g-ppp.eu/wp-content/uploads/2016/12/5g_norma_d4-1.pdf, Nov 2016.

[11] P. Rost et al., *Network Slicing to Enable Scalability and Flexibility in 5G Mobile Networks*, IEEE Communications Magazine, May 2017.

[12] V. Sciancalepore, K. Samdanis, X. Costa-Perez et al., *"Mobile Traffic Forecasting for Maximizing 5G Network Slicing Resource Utilization,"* in *Proc. of IEEE INFOCOM*, May 2017.

[13] D. Bega, M. Gramaglia, A. Banchs et al., *"Optimising 5G infrastructure markets: The Business of Network Slicing,"* in *Proc. of IEEE INFOCOM*, May 2017.

[14] P. Caballero, A. Banchs, G. de Veciana et al., *"Network Slicing Games: Enabling Customization in Multi-Tenant Networks,"* in *Proc. of IEEE INFOCOM*, May 2017.

[15] P. Rost, M. Puente, A. Banchs, I. Berberana, M. Breitbach, M. Doll, H. Droste, C. Mannweiler, K. Samdanis, and B. Sayadi, *"Mobile network evolution towards 5G," IEEE Communications Magazine, vol. 54, no. 5*, May 2016.