Zero-Trust Cybersecurity Approach for Dynamic 5G Network Slicing with Network Service Mesh and Segment-Routing over IPv6

Bruno Dzogovic[1], Bernardo Santos[1], Ismail Hassan[1], Boning Feng[1]
Department of Computer Science[1]
Oslo Metropolitan University[1]
Oslo, Norway
{bruno.dzogovic, bersan, ismail, boningf}
@oslomet.no

Van Thuan Do[2], Niels Jacot[2]
Wolffia AS[2]
Oslo, Norway / Oulu, Finland
{vt.do, n.jacot}
@wolffia.net

Thanh Van Do[3,1]
Telenor Research[3]
Telenor ASA[3] / Oslo Metropolitan University[1]
Oslo, Norway
thanh-van.do@telenor.no

***Abstract*—As the 5G mobile networks become widely adopted across various industries and verticals, additional requirements for strengthening their security emerge. Traditional security approaches have been successful in preventing adversarial activities across generic networks and datacenters, but the complexity and extent of the 5G communication systems renders these insufficient. Therein the need for a stringent tactic to ensure the reduction of the attack surface within the 5G core networks. This paper examines the potential threat of Distributed Denial of Service (DDoS) and specifically, flooding attacks that can wreak havoc on the 5G mobile infrastructure as well as design a solution according to the zero-trust security model to ensure the continuity of the service in corresponding disaster scenarios.**

Keywords— Zero-Trust; 5G; Network Service Mesh; Network Slice Selection Function; Cyber Warfare; Botnets; DDoS; SRv6.

I. INTRODUCTION

The complexity of 5G core networks can vary in contrast to the deployments different Telco operators apply. 5G supports a variety of verticals, such as industrial Internet of Things (IoT), automotive and transport or healthcare. This suggests that the number of connected and managed devices is by orders of magnitude higher than it was the case with 4G Long-Term Evolution (LTE), which increases the cyber threat surface. Furthermore, 5G introduces the concept of network slicing to logically divide virtual networks, which obscures potential vulnerabilities. Adversaries can launch attacks within a single or across network slices, or from one Public Landline Mobile Network (PLMN) to another one between core networks. Some of the means utilized in flooding attacks can be various devices connected to the 5G network, such as IoT, sensors, cameras, computers, etc. The Layer-7 flooding attacks can be exceptionally sophisticated and by avoiding usage of malformed packets, significantly difficult to detect. According to the A10 Networks 2020 State of DDoS Weapons Report for 2020, by forming a botnet, the adversaries can generate massive attacks on large scales and execute DDoS on any desirable endpoint using the preceding means. Consequently, one of the most efficiently used protocols for distributing botnets across IoT backends is the Simple Service Discovery Protocol (SSDP), which serves as a discovery protocol for connected devices and represents a simple mechanism for utilizing DDoS attacks as cyber warfare and cyber terrorism [1].

The Network Service Mesh (NSM) connectivity paradigm will allow containerized 5G virtual network functions to be connected in a zero-trust service model where restrictive policy applies to the communication inside the core network, between remote locations in virtual or physical environments alike; this shall maintain integration with the underlying transport network fabrics to delegate appropriate Quality of Service (QoS) and Quality of Experience (QoE) to the end users.

This research utilizes the design science paradigm to contrive a methodology for automated container orchestration of 5G core networks and applying the NSM zero-trust design to safeguard virtualized 5G mobile core networks in containerized hybrid cloud environments. Consequently, a fundamental solution for mitigating flooding attacks is proposed.

II. RESEARCH BACKGROUND

A. Zero-Trust Networks

The concept of zero-trust network architecture originates from the premise that no architecture is sufficiently secure at any point in time. Therefore, every deployment is primarily assumed as insecure, based on which the further steps of implementing security measures are being taken into consideration. The zero-trust approach can be applied on three levels in a deployment: user, application, and infrastructure level [2]. A traditional security architecture follows a layered approach in enabling security, whereas the zero-trust model separates the infrastructure fabrics into control and user plane,

978-1-6654-8162-5/22/$31.00 ©2022 IEEE

while assigning the services and hosts into classified categories based on least-access policies and rule sets. One of the key steps in achieving a zero-trust policy is to identify the most critical assets in the network as well as their value. As depicted in Fig. 1, the zero-trust model eliminates the network locality element and corroborates the security of the user, disregarding his location.

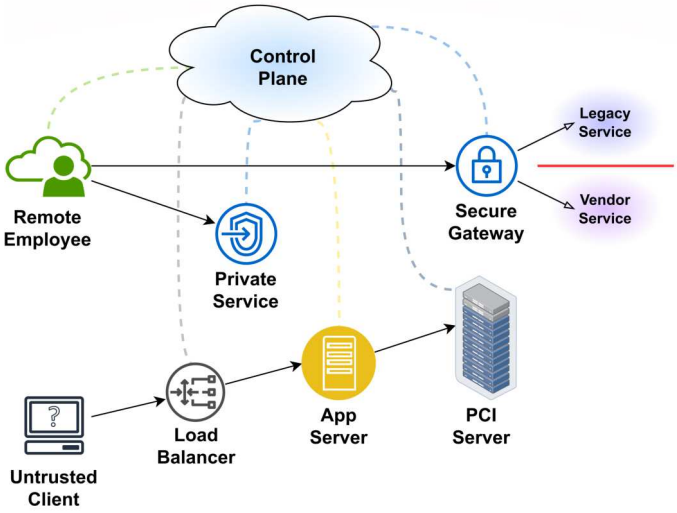

Fig. 1. The Zero-Trust network security deployment model

This way, the zero-trust model can reduce the overall operational complexity and strengthen the security compared to the traditional models, which are based on zones and firewalls, or also known as the "perimeter model" [3]. In the perimeter model, the networks are typically secured using techniques such as Network Address Translation (NAT). For the modern-day requirements, certain zones within the perimeter model can be abstemiously insecure, like for example placing webservers into exclusion zones, or Demilitarized Zones (DMZ), where the traffic is being monitored and controlled. The modern cyberattack landscape has rendered this approach obsolete due to numerous disadvantages, such as lack of intra-zone traffic inspection, lack of elasticity in host positioning, individual points of failure, etc. The network locality is another construct that sets limitations for defining security requirements that are mostly governed by Virtual Private Networks (VPN). A VPN allows for tunnelling to remote locations, where the traffic is decapsulated and routed, and can be an ideal backdoor that always becomes neglected. By removing the network locality requirement, VPNs become superseded. In this case, the management of security is repositioned from the core networks to the edge [3].

B. The 5GC Core Network

The mobile core networks are comprised of components that have specific roles, such as gateways, databases, various processing units, etc. (see Fig. 2). The 5G core reinvents these hardware modules into a virtualized and service-based network function architecture, which can be deployed as a software and scaled as much as the underlying infrastructure allows. The virtual functions in 5G can be customized to suit the requirements of the Telco service provider and accommodate various users and devices. As denoted, the 5G core network consists of a control plane and a user plane. The user plane harbors the Radio Access Network (RAN), which is a separate functionality that defines the radio access of wireless devices and where the user data and traffic flows. The wireless devices can be various User Equipment (UE) that either attaches to the RAN using a 5G frontend and SIM authentication, or a non-3GPP access (like Wi-Fi or WiMAX) and directed to the Access and Mobility Function (AMF) and the User Plane

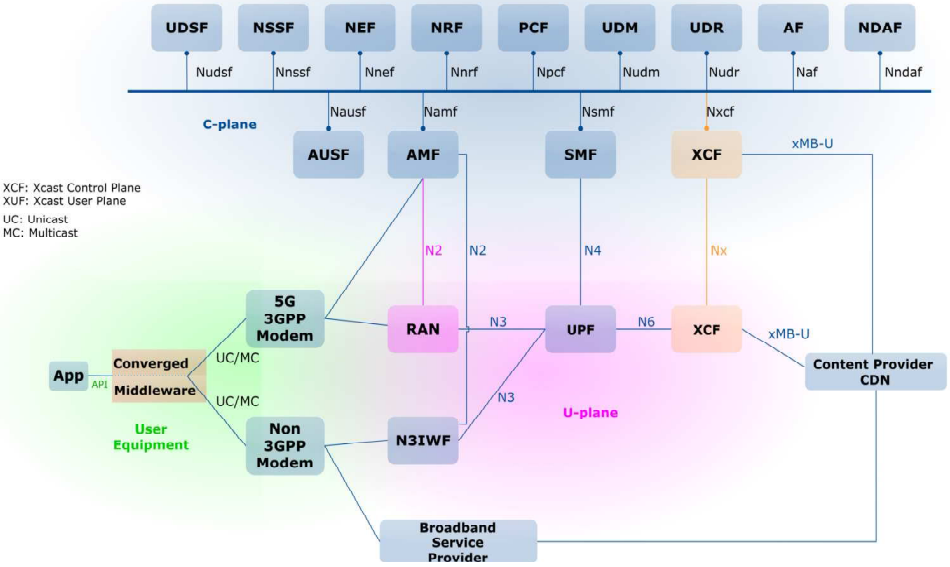

Fig. 2. 5GC mobile core network architecture

Function (UPF). During a defined attachment procedure, the UE devices exchange control plane messages to reach the core [4].

In both access scenarios, the UE exchanges communication with the AMF either through the RAN or the N3IWF function, via the N1/N2 interface. During a 5G Standalone UE authentication through the RAN and the wireless random-access procedures, the next Generation Node-B (gNB) sends a Non-Access Stratum (NAS) initial registration message request with the 5G UE's Global Unique Temporary Identifier (GUTI) code and retrieves the UE context, which is a combination of the Subscription Permanent Identifier (SUPI) code and MM context from the AMF. Consequently, the AMF sends back the UE context data in the form of HTTP 200 code. The request message includes the "RAN UE NGAP ID" and the "RRC Establishment Cause", Network Slice Selection Assistance Information (NSSAI) as well as additional information like the UE capabilities, which is indication from the gNB that the UE can establish a stable radio link and can be attached to a specific network slice provided that its credentials from the SIM card are stored in the database within the 5G core network. In case of an inter-cellular handover, a new AMF needs to contact the old AMF and retrieve the context of the UE for the same to be able to attach to a new gNB base stations in case of mobility [4].

The message exchange between the Software-Defined RAN controller and the AMF is encoded within the NGAP protocol, which carries the NAS registration requests/responses. A device can have multiple states, depending on whether it successfully authenticates in the network. Typically, a non-authorized UE will receive a REJECT message, while during successful authentication there will be states such as INITIATED or REGISTERED. Consequently, the AMF requests from Authentication Server Function (AUSF) to start an authentication procedure, which in turn proceeds to reaching the User Data Management function (UDM) for searching through the database for the requested user information [4]. The Unified Data Repository (UDR) database is queried and in case a valid user with legitimate credentials from the SIM is present, UDR responds with the subscription data in the form of HTTP POST message to UDM, including a defined policy for connectivity that is governed by Policy Control Function (PCF). After a successful registration, the UE is assigned an IP address by Session Management Function (SMF) and QoS parameters, while the UPF ensures the connectivity of the UE to the internet altogether with routing and Domain Name System (DNS) access [4].

C. Network Slice Selection Function in 5G

The AMF uses the Network Slice Selection Function (NSSF) to retrieve information related to a network slice. Users can select a network slice with defined QoS parameters, network optimization and different support features from a serving PLMN depending on the subscription. Different slices can have different Single Network Slicing Selection Assistance Information (S-NSSAI) identifiers that stand for slices with various service types. NSSF handles delegating the allowed, configured, or restricted NSSAIs to AMF for Protocol Data Unit (PDU) session registration. To prevent cross-slice lateral movements of attackers in compromised NSSF endpoints, solutions like OAuth 2.0, HTTPS or rate limiting can help in circumventing potential unauthorized access. These include the usage of Public Key Infrastructure (PKI) and validation tokens, which will ensure that the legitimate AMF instances can communicate with the adjacent NSSF [5].

D. Threats to the Mobile Core Networks

The 5G networks can suffer from the same attacks as any other network or critical infrastructure, whether it is in the form of social engineering, ransomware, or various forms of DDoS attacks [6]. The denial of service has proven a reliable tactical warfare when executed at a massive scale. Typically, DDoS attacks can be categorized in two major groups: volumetric (or flooding/link saturation) and low volume (or slow rate) attacks. Both can be conducted using three methods: spoofing, reflection, and amplification. Spoofing attacks are direct and compromised nodes attack the victim directly, while reflection and amplification attacks use a reflector or a proxy node that sends an IP datagram based on a previously received one. The latter obfuscates the attacker's location, because the reflector nodes point to a spoofed IP address of the victim. Therefore, mitigation of such attacks can be exceptionally complicated to apply and prevention of the same requires advanced strategies. Normally, any device or tool such as services, network devices, Internet of Things (IoT) devices or botnets of the same, can be used to target organizations. In this research, we focus on the flooding/link saturation attacks, which are difficult to differentiate from the normal traffic because they use standard URL-based requests and do not always require exploiting of a known system or network vulnerability. Therefore, the devices that are compromised are usually authenticated and confirmed in the network. Some standard ways of preventing and detecting flooding attacks are deep packet inspection, detection of abnormal traffic activity, traffic profiling, introducing IP blacklisting, etc.

These solutions have functioned in the traditional networking model and monolithic infrastructures, but since the highly scalable deployments like 5G require by orders of magnitude more resources and dynamic endpoints, the procedures become deficient and cannot scale accordingly [7]. Previous research has indicated success in preventing a denial of service by adapting a SDN controller in the network to detect and mitigate SYN flood attacks that are based on sending customized User Datagram Protocol (UDP) datagrams [8], but little is known about the use-case scenario concerning 5G infrastructure in terms of using simpler forms of SSDP amplification attacks. These can be executed also through multifaceted IoT botnets and can be exceptionally difficult to detect and counteract [9][10].

A DDoS attack can execute simply in two stages from the Command-and-Control Center (C&C). The first phase is a generated probe request that serves as a scan to obtain amplification parameters, and the second is the crafting of a packet datagram for the message generated with adjacent protocol port. During the reconnaissance phase, the attackers can also use sophisticated machine learning and AI methods to find the most vulnerable endpoints. The flooding can occur

through reflected nodes and across subnets, in which case a carpet-bombing DDoS attack can disperse the traffic through a range of IP subnets that makes it even more difficult to detect and mitigate. Another obfuscation method is a pulse-wave attack that performs series of short and high-intensity synchronized pulses that are exceptionally adept at circumventing hybrid mitigation (on premise and in cloud) and can target the mitigation mechanism itself in addition to the victim [10]. Therefore. the adversaries can time the execution of the attack to target endpoints based on a cause, for example a rush hour in a city center. In this case, there would be vehicles with infected IoT sensors or modules that can be triggered to execute the SSDP discovery procedure and flood the 5G vehicular network slice, causing disruptions of unprecedented levels.

III. The Network Service Mesh Model

Mitigation of DDoS attacks is different than other attacks like ransomware and may include adjustment of load balancers to identify and respond to DDoS patterns. Since DDoS attacks can stem from vulnerable Memcached services [11], a mitigation strategy can thus begin from the service layer. This is where the Network Service Mesh (NSM) plays a crucial role. As previously explained, the intra-cluster communication of 5G virtual functions in containers can be a L7 HTTP payload and there is a potential of ethernet headers, IP headers and the TCP connection to be stripped away and replaced. The NSM ensures that the payload being transported within the mesh is indeed a L7 HTTP traffic, regardless of the location of the network functions.

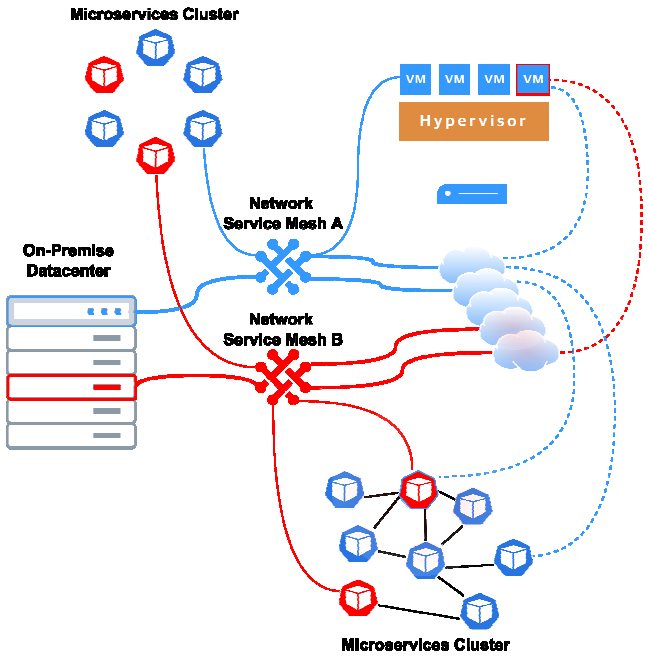

Fig. 3. Hybrid connectivity across clouds, virtualized and physical infrastructure with Network Service Mesh

Fig. 3 describes possibility of workloads being connected to small highly granular Network Services that only involve their immediate collaborators for a particular purpose (as in the case with database replication). Because Network Service Mesh authentication uses the same Spiffe-ID [12] that the workloads themselves use to communicate at L7, the auditability of the system based on a cryptographic identity, therefore, extends from L3 to L7 [13]. NSM facilitates advanced networking for services that require refined communication control and integration with existing multifaceted network structures governed by SDNs (Software-Defined Networks), like in the case with Kubernetes [14]. In other words, the SDN controller orchestrates the network resources on L1/L2/L3, while NSM enables scalable and secure microservices connectivity in hybrid infrastructures over separately configured transport networks. NSM constructs the zero-trust concept by decoupling services irrespective of their running location if they are members of the same namespace with defined rule sets [13]. The standalone CNI (Container Networking Interface) service layer forms an overlay that allocates a separate addressing space to split the communication plane from the Kubernetes control plane and is usually based on complex encapsulation methods like VxLAN, which introduces performance overheads and scaling constraints. Other CNIs such as Calico, expand the connectivity palette and offer Border Gateway Protocol (BGP) integration with the underlying network infrastructure and incorporation of the running microservice cluster within BGP autonomous systems (AS), including connectivity policies [15]. Another major disadvantage of the default CNI in Kubernetes is that there are no means for handling Layer-2 resources and managing of virtual network functions in the lower layers. Containers can communicate within a cluster sufficiently using an overlay network, but when workloads need to be fragmented across various locations and environments, a separate standalone mechanism is required that can also scale in parallel, as well as provide appropriate security for the virtualized remote workloads. NSM addresses this implementation issue by adopting the NSMgr (NSM manager) on each node in the cluster. The managers communicate between each other to respond to network service requests from clients and create a virtual wire (vWire) between the clients and the Network Service Endpoint, which can utilize any lower layer transport network mechanism [13].

A. *Enabling Network Service Mesh through Service Function Chaining*

To better understand how NSM can deliver Network Function Virtualization (NFV) into microservices environment, it is necessary to examine the concept of Service Function Chaining (SFC). Enabling NFV entails introducing on-demand or dynamic management of hardware network resources. The concept of network virtualization has been known for a long time and is applied to traditional datacenter workloads, but since Telco operators began adopting the cloud paradigm for the mobile communications use-case, the dynamic delivery of virtual network resources has become a stringent requirement. 5G follows a service-based architecture and therefore diverse services need to be dynamically operated. Each service can have different requirements and policies and should be available on user demand. It should also have high-availability and fault-tolerance, as well as employ optimization strategies to comply with certain QoS/QoE parameters specified in Service

Level Agreements (SLA). To satisfy these requirements, services are being deployed in the edge in the form of micro-clouds [16]. SFC enables automation for provisioning network resources that can be clustered together to form chains of multiple services that work together like an assembly. The implementation of SFC is governed by the Management and Orchestration (MANO) NFV model, standardized by ETSI [17].

In a SFC model, the virtual functions are deployed in a manner that respects a clearly defined order, where each function supports a different service and thus forms an end-to-end communication chain using a Virtual Network Function Forwarding Graph (VNFFG). A service function forwarder acts like an intermediary between nodes to determine the communication route and enable IP encapsulation, domain forwarding, network overlay transport, etc. (see Fig. 4). The services involved in the deployment need an automated service discovery method because of the possibility to deploy dynamic clusters at a large scale and for providing rolling upgrades of actively running services, which is a major challenge [16].

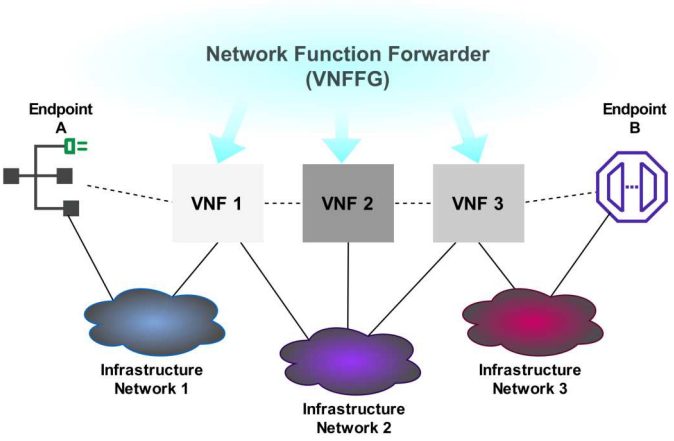

Fig. 4. Service Function Chaining architecture with VNF forwarding graphs

Furthermore, the SFC architecture involves other considerable challenges in terms of implementation and performance. Optimizations can be applied to augment the QoS and QoE. They may include network latency minimization, resource utilization arrangements, cost reduction, power and energy saving, SLA-based adjustments and other approaches [16]. The QoS is one of the biggest challenges within SFC, especially concerning the dynamic traffic steering, dynamic resource allocation, the overall end-to-end service delivery, and advanced security requirements. When deployed in distributed hybrid environments, services typically communicate across different regions, clouds, vendors, or physical infrastructure. In container clusters, services are connected in abstract compositions through Layer-7 network services that live on Layer-2 and Layer-3 resources. Kubernetes groups the endpoints and services into pods that can be further categorized in namespaces to be horizontally scalable and enable chains to be formed based on connectivity policies and virtualization pass-through from the underlying hardware [14].

B. Customizing 5G Network Slices using Service Function Chaining

The network slicing concept in 5G has emerged as a result of the requirement for providing different users with different QoS and therein, QoE, which are complex topics. The QoS is governed in the transport Multiprotocol Label Switching (MPLS) networks and Software-Defined Wide Area Networks (SD-WAN), and involve the combination of parameters like bandwidth, latency, jitter, and packet loss [18]. To be able to provide granular QoS control within the SDN controllers, a service layer should be available for selection of network slicing parameters within the 5G NSSF function. Each service function chain node needs to implement an apparatus that focuses on contraction and expansion of a VNF, while pertaining optimization to satisfy the end-to-end QoS parameters of the service function chain. Therefore, we introduce the concepts of Segment-Routing over IPv6 (SRv6) for the BGP-MPLS transport network layer to support the SFC and enable traffic engineering policies for the NSM endpoints (see Fig. 5) [19].

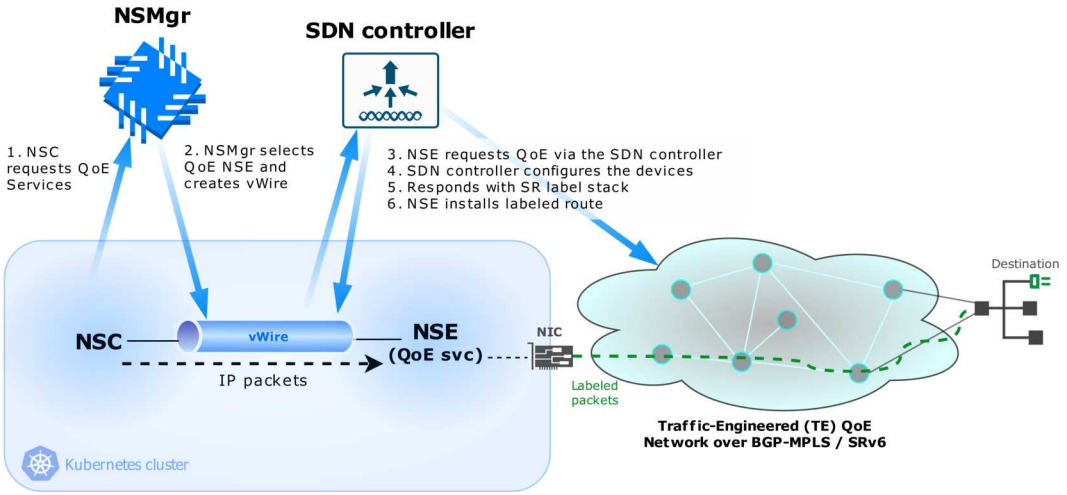

Fig. 5. SDN-orchestrated QoE Slice as a Kubernetes NSM Service (QSaaNS) over a traffic-engineered BGP-MPLS/SRv6 transport network

When implemented within the service function chain, the NSSF function should satisfy certain criteria [20]:

- Selection of slice delegated to the AMF function,
- Information about the slice and PDU session QoS parameters,
- Usage of HTTP standard messaging for NSSAI information exchange,
- Negotiation of slicing features.

SFC is an important prerequisite to 5G network slicing. The support of SFC within NSM can be based on traffic steering in context of assigning separate network flows with adjacent policies. The container-based microservice load balancing in cloud-native environments is segregated within stages that include the following [21]:

- Defining each chain as a sequence of cloud-native functions,
- Satisfying the flow requirements of each function,
- Assigning a policy for flow requests to originate and pass through all designated network service endpoints.

The simplification of the traffic engineering can be achieved via grouping nodes within segments and defining which packets can traverse which segments. Enabling the SPRING model (Source Packet Routing in Networking) that defines the segment-routing SRv6 is achieved through the Fast Reroute (FRR) framework in the current experimental testbed [22].

IV. Implementation

To experiment with the Network Service Mesh and its role in providing unified security for 5G network slices within the NSSF function, the zero-trust model is implemented through the OpenDaylight SDN controller and NSM for Kubernetes clusters [13][23]. The 5G core network is based on the open-source OpenAirInterface5G software [24], whose 5G core functions are adapted for orchestration using Kubernetes [14].

As described in the previous chapter, Kubernetes imposes limits on connectivity in terms of number of interfaces within a pod as well as the possibility to pass through virtual functions from the physical network interfaces into containers. To address this drawback and enable additional interfaces in a container to split the functionality of the network functions into user and control plane, the Multus plugin is used for complementing Kubernetes pods with additional virtual interfaces [25]. The host machines on which the containerized 5G core networks are running have a SR-IOV virtualization endpoint and bypassed Linux kernel networking with a Vector Packet Processing mechanism (VPP). The VPP enhances the scaling of virtual functions in contrast to the Linux kernel networking and retains performance in case of high-speed connectivity with low latency, which was shown in our previous research [26]. This is instituted to minimize operating-system related inconsistencies and minimize packet loss and error rate. Furthermore, VPP itself represents a viable method for alleviating pressure from flooding attacks on the core network.

For establishing the hybrid infrastructure model, the radio frontend networks are deployed in a Cloud-Radio Access model, which includes OpenStack, AWS, and Azure clouds [27][28][29], while the 5G core networks are situated on premises in a datacenter. The communication between the datacenter and the cloud providers needs to be routed using BGP-MPLS, and for that purpose, the FRR framework is implemented in spine/leaf configuration using Virtual Route Forwarding (VRF) [30]. The computation of a path between the cloud regions and the core network on-premises is achieved by SRv6 segment routing that is realized with PathmanSR [31]. PathmanSR calculates the best route in terms of QoS/QoE between OpenDaylight routers carrying SR-path segment stacks. PathmanSR does not manage the physical network infrastructure, but only the SR-paths. The default Kubernetes overlay network is replaced with Calico BGP that supports Vector Packet Processing [32] and the networking is appointed as a NSM service to be able to fix policies for routing outside the cluster and between different core network AMF instances (see Fig. 6).

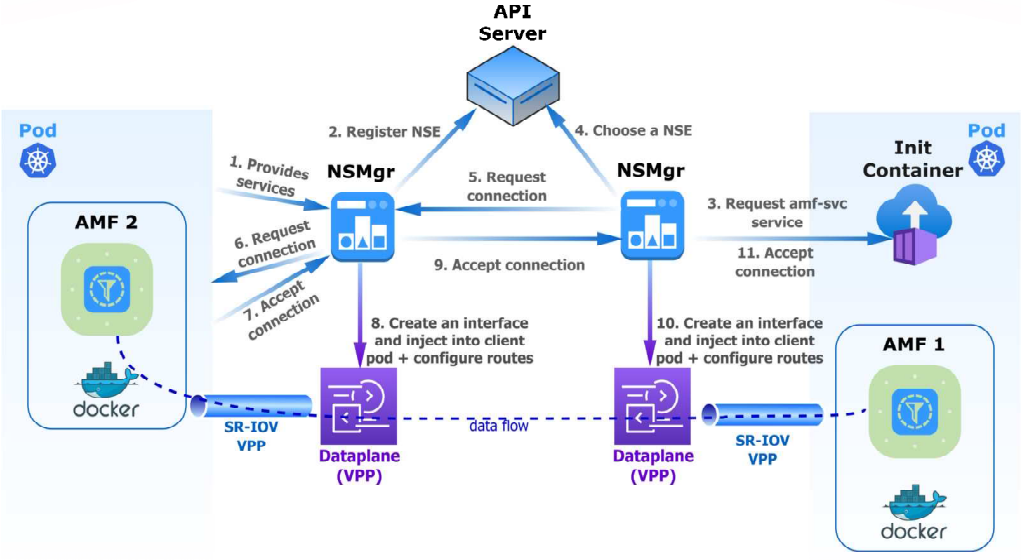

Fig. 6. NSM connection establishment between different remote 5G AMF endpoints in a hybrid infrastructure model

Consequently, the extra-cluster communication requires a load balancer to be able to connect to external networks such as the BGP fabrics, which is accomplished by installing the MetalLB load balancer for Kubernetes bare metal deployments [33] and does not exist as an option for Kubernetes by default. MetalLB can work in Layer-2 and Layer-3 modes. In Layer-3 mode, since it utilizes BGP, MetalLB triggers a conflict in the communication with the cluster networks of the Calico-VPP CNI as the MetalLB load-balancer will not have a route to the ToR (Top-of-the-Rack) router because this is already established with the cluster networks. This compatibility impediment is resolved by allowing the FRR BGP router to serve as a spine router and utilize VRF (Virtual Route Forwarding), where the MetalLB load-balancing endpoints are considered as leaves with their own autonomous system and instantiated as BGP speakers that route towards a VRF-2 virtual router. The Calico-VPP establishes another route to the VRF-1 virtual router as the main cluster route propagates towards the exterior networks. The packets of each are labeled differently and as the ToR router is therefore split in two VRF segments with two virtual routing maps; consequently, by utilizing judicious inter-VRF route leaking, the two routing table maps are then re-imported after the routes from VRF-2 propagate into the VRF-1 routing tables, upon which the merged ToR routing table of two SRv6 segments is assembled. The spine router will then manage the routing between different autonomous systems using the labelled segments so they can be reached from external SDNs and other NSM microservice clusters. This subsequently enables the Calico-VPP cluster network to work uninterruptedly and independently from the MetalLB Layer-3 load-balancer, while the Kubernetes service can be exposed to a public IP address and can communicate to other clusters in other external networks over BGP-MPLS in remote clouds and BGP communities.

Nevertheless, this will not fulfill the requirements of SRv6 as MetalLB will need to route IPv6 networks with custom prefixes through external providers, which can be accomplished by enabling the FRR mode of MetalLB and therein Bidirectional Forwarding Detection (BFD) [34] support for BGP sessions to further split the routed addresses in two segments with two different prefixes and reduce transport network convergence periods. By default, MetalLB advertises only the IP address of the configured peers without any supplementary attributes. Therefore, MetalLB presents each IP as a /32 prefix, which can be rejected by a transit provider as routes with prefix values higher than /24 are typically refused. At this point, it is required that the prefix with value /24 is advertised to the transit provider and simultaneously retain the capability to route between peers internally (in this case, the cluster networks of Calico-VPP). This is achieved by splitting the IP address space into two routable segments and the one with /32 prefix is not being advertised to the peer routers by assigning a "no-advertise" community. The peers will then propagate the addresses with prefix /24 to the transit providers and the initial /32 network address space will be used to forward and load-balance traffic into the cluster.

A. Network Service Mesh as a Service

Cloud users and tenants should be able to utilize the NSM as an end-to-end service to be able to disperse workloads across hybrid infrastructures or regions in a secure way, or in case advanced connectivity features are required. The zero-trust approach within NSM will assume that no connection of any User Equipment is secure enough to the corresponding core network, while attaining connectivity policy and allowing users to deploy the NSM containers as a cloud service. This on-demand approach requires the NSM to be tightly integrated within the cloud service layer to interwork with the underlying cloud networking module, which in OpenStack is Neutron [35]. Containers in OpenStack can communicate to the Neutron fabrics using additional plugins, but NSM can bypass this requirement due to the implementation of MetalLB and the possibility to pass through the VNFs from the NIC card to containers at the physical compute nodes by using SR-IOV (Single Root Input/Output Virtualization). This will further reduce the complexity of implementation.

V. EVALUATION

To assess the implemented design and the preliminary efficiency of this setup, the solution is realized in a controlled cloud environment and probed using the Momentum botnet with IoC (Indicator of Compromise) "Trojan.Linux.MIRAI.SMMR1" against a generic firewall. At this stage, we employ an infected IoT sensor device in the 5G network that attaches using SIM authentication without accounting for vulnerability exploits. The Momentum botnet creates a backdoor to establish a connection to IRC channel and receive remote commands. Furthermore, Momentum utilizes the fast flux technique to render the C&C network resilient and employs multiple IP addresses with a domain name to obfuscate the attack and mislead mitigation. The reflector unit will then scan for available devices in the network and initiate service discovery to probe for other devices that can be used as bots and to further propagate the attack on port 1900 SSDP. Addressing flooding attacks and DDoS amplification adversarial activity, as described previously, is feasible when the attacker knows the IP addresses of the targeted endpoints, which in this case is the 5G AMF container function. For experimental purposes, it is assumed that an attacker has gained access to the cloud infrastructure level over a vulnerable container and launches an attack on the core network using a proxy container. To measure the efficiency of the NSM in preventing the flooding and DDoS attacks, it is thus necessary to compare a case when an attack is launched through a compromised container on the core network in a flat network overlay mode, opposed to the situation when NSM is fully implemented and integrated with the transport network over BGP-MPLS and SRv6. For this purpose, a threshold of detection is adjusted that defines a probability of impending DDoS attack that is based on segmented regression analysis. The goal is to ascertain whether the generic firewall can detect the attack before the AMF function stops operating and compare this to a deployment with the zero-trust model using NSM. As a control, we execute the experimental payload to a 5G core without establishing additional security layers and by relying to the SIM authentication and default load-balancing mechanisms into

place. In addition to that, the NSM can change the FQDN name of the AMF endpoint and migrate it to another cloud, while Kubernetes resets the adjacent container and initializes a new one. The new container will obtain information from the service discovery component Etcd in Kubernetes, and resume with the last working state to avoid downtime. Consequently, we also measure the downtime of the AMF function in both cases to ascertain the service recovery speed or whether it can recover in the first place.

To determine the threshold of registering new devices in the AMF, which will indicate that the DDoS is disrupting the function, we develop a model based on segmented regression analysis to find at what time the service will begin collapsing and whether it can recover. Although the scale of a SSDP based DDoS attack can be massive, in this experiment we adjust the packet ratio based on the 1Gbps network available bandwidth and constraints on hardware resources mapped into the containers. For evaluation, we define the segmented regression equation as:

$$\hat{y} = b_0 + b_1 x_1 + b_2 (x_1 - x^k) x_k \quad (1)$$

Where:

x_1 is the value of the independent variable,

$x^{(k)}$ is the break point (knot), or time when the AMF becomes saturated, based on the payload (packet rate to bandwidth ratio), and starts dropping new UE registration request messages without returning a response due to the inability to manage further requests. This should drop suddenly or gradually after surpassing the break point threshold and thereby we select this value manually,

x_k is the break point dummy variable, which takes a binary value related to the independent variable value and whether it surpasses the defined threshold. The knot dummy variable is defined as follows:

$$X_k = \begin{cases} 0 \; if \; x_1 \leq x^{(k)} \\ 1 \; if \; x_1 > x^{(k)} \end{cases} \quad (2)$$

The break point is the time where the service becomes unavailable due to the attack, which can serve as an indicator also if the attack can compromise the core function and at which moment in time. The segmented regression model is implemented in R with a simple dataset defining a relationship between a payload measured in Mbps and time of the duration of the attack (in minutes). The AMF function is considered disabled if no further UE devices can perform the attach procedure, whereas the existing 200 UE devices will lose any connection to the Internet.

A. Results

In the case of a flat network architecture, the attack incapacitates the AMF function completely after 9 minutes of constant payload that reaches the maximum link capacity of the network supported by the Vector Packet Processor and 3 replicas of the same container. The AMF function does not recover as there is no mechanism in place to provide regenerative support. The segmented regression analysis shows that the break point occurs at approximately 9 minutes which is the link saturation threshold, after which the service gradually becomes unavailable and after 15 minutes, the attached devices drop the connection to the AMF being unable to attach (see Fig. 7).

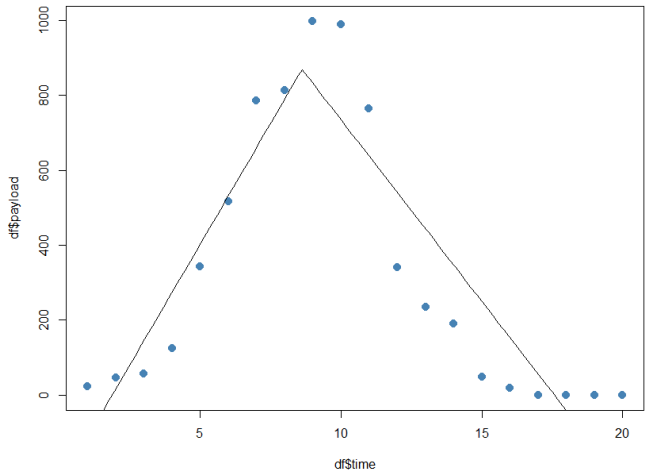

Fig. 7. Segmented regression analysis of the tests on the AMF function in case of a DDoS attack in a flat network model

Kubernetes attempts to revitalize the failed AMF function containers; however, the flat Ethernet transport layer of the network disables the connection between the 5G gNB base station to the core, which had to be restarted manually. This is the control experiment, which serves as a comparison basis to the consequent tests.

Subsequently, we measure the same scenario with a firewall and load-balancer that utilizes round-robin algorithm, configured to address traffic bursts (Fig. 8).

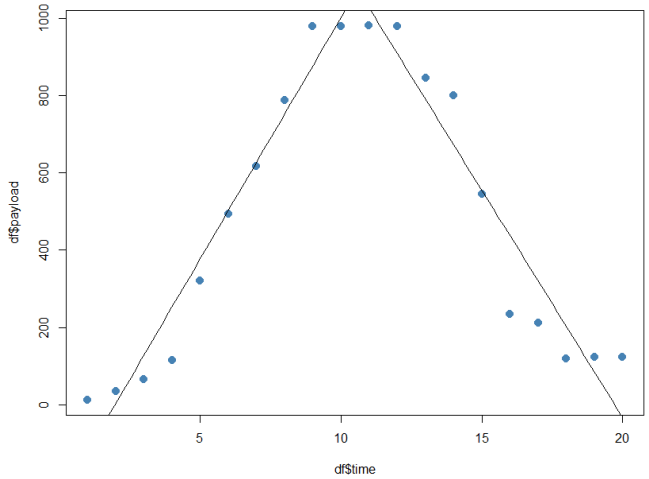

Fig. 8. Segmented regression analysis of the tests on the AMF function in case of a DDoS attack in a network supported by firewall and a load-balancer

Other algorithms are not assessed, and in those cases, results may vary. However, the load-balancer plays a key role in enabling some level of fault tolerance to the network functions, which is observed from the results in Fig. 8 where the link saturation occurs from approximately 7 minutes and lasts for 5 minutes continuously. After that, the Quality of Service deteriorates considerably, and this approach becomes deficient. The load is distributed between 3 replicas of the same network function and the operation is maintained with minimal access bandwidth available (between 118 and 123 Mbps). In this case, the devices do not exhibit a connection disruption, but a severe reduction in the Quality of Service. This remaining available bandwidth is reallocated between 200 UE devices and therein the available service is practically unusable. The orchestration layer does not attempt any regeneration of the network function, as the activity is not being flagged as malicious or disruptive for the container.

The last test refers to the Network Service Mesh model of the 5G core network, which also utilizes a load-balancer to advertise the routes of the cluster network to an external endpoint in an AWS cloud location. In this case, the AMF runs in three replicas (one in an OpenStack cloud, the other in AWS and the third on-premises in a datacenter). Fig. 9 shows the segmented regression analysis and how the NSM redirects the traffic towards a new AMF container that is being generated at new cloud locations after a certain threshold of the attack is reached, at approximately 7 minutes. The service availability marginally decreases for the UE devices, but it continues to provide a satisfactory QoS, with another minor divergence occurring at approximately 13 minutes during the attack. Nevertheless, the QoS continues to be uninterrupted and as predicted for all the 200 attached devices, the attack digresses and is mitigated by the shift of IP addresses of the AMF endpoints in the NSM namespace.

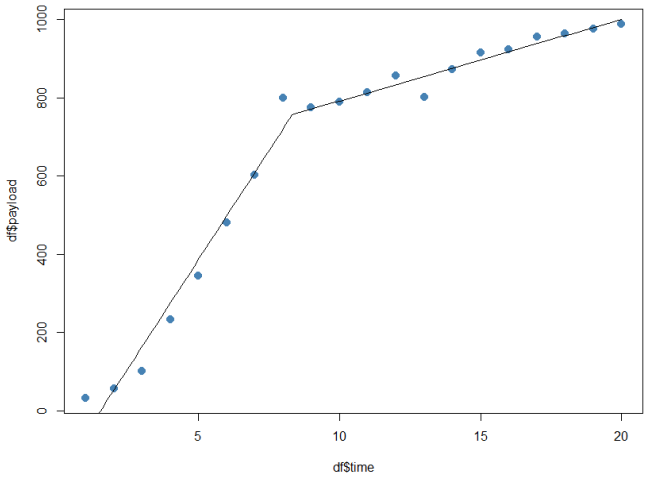

Fig. 9. Segmented regression analysis of the tests on the AMF function in case of a DDoS attack over a traffic-engineered SRv6 BGP-MPLS network with Network Service Mesh in the Kubernetes cluster

This suggests that the threshold at which we specify the initiation of new containers during a traffic burst or in case of a flooding attack, can be set to satisfy both scenarios and adopt a hybrid infrastructure model to provide resiliency, fault tolerance and retain uninterrupted Quality of Service and Quality of Experience, as well as security for the end users. The same threshold can be learned by a machine-learning model and adjusted dynamically to correspond to a heterogeneous and mutable environment.

VI. Conclusion

Adversaries can exploit a secondary vulnerability in a cloud to gain access to a mobile core network and therefore execute further attacks. In this case, the SIM authentication in 5G and a generic firewall will not be sufficient to protect the network from flooding attacks because the reflectors and bots are previously authenticated. The Network Service Mesh can enhance the privacy of data and information traversing international networks and during roaming in 5G, while providing a regenerative mechanism in disruptive scenarios. The protocol and routing technology agnostic approach allows for the establishment of a zero-trust model that addresses unknown vulnerabilities of core networks variety of attacks. However, the complexity of deployment can vary according to the scale and structure of the transport networks, and this may require a certain degree of autonomy in the implementation of the NSM and corresponding policies because of the dynamic nature of containerized environments and the large scale of the mobile core networks. As a result, in the current experimental scenario there is no assumption of an automated mechanism to steer the malicious traffic to a honeypot or a dummy network for offloading the DDoS amplification stream away from the actual 5G core network functions. With a similar functionality, the system would then be capable of preventing malicious traffic payloads to inundate the AMF and disrupt the user connectivity to the mobile core network and therein the Internet. Consequently, if combined with advanced replication techniques in Kubernetes, it is possible to further alleviate DDoS pressure on the AMF function in which case NSM will steer the traffic to new AMF replicas from the one that is a subject to DDoS and re-authenticate connected UEs without losing connectivity, while regenerating the AMF function to the desired number of replicas. Nonetheless, the zero-trust model diminishes the attack surface and protects containerized 5G cloud core networks without vulnerability assessment of container images and allows for extensibility of the solution to adapt to the requirements of various Telco architectures.

Acknowledgment

This paper is a result of the H2020 Concordia project (https://www.concordia-h2020.eu) which has received funding from the EU H2020 programme under grant agreement No 830927. The CONCORDIA consortium includes 23 partners from industry and other organizations such as Telenor, Telefonica, Telecom Italia, Ericsson, Siemens, Airbus, etc. and 23 partners from academia such as CODE, university of Twente, OsloMet, etc.

References

[1] A10-Networks, “DDoS Attack Mitigation: A Threat Intelligence Report – The Global State of DDoS Weapons”. 2022. [Online]. Available: https://www.a10networks.com/resources/reports/ddos-attack-mitigation-a-threat-intelligence-report/

[2] Palo Alto Networks, “What is a Zero Trust Architecture”. 2022. [Online]. Available: https://www.paloaltonetworks.com/cyberpedia/what-is-a-zero-trust-architecture

[3] E. Gilman and D. Barth, “Zero Trust Networks,” O'Reilly Media, Inc. 2017. ISBN: 9781491962190.

[4] European Telecommunications Standards Institute, Technical Specification, “ETSI-TS-129-518: 5G; 5G System; Access and Mobility Management Services; Stage 3 (3GPP TS 29.518 version 16.6.0 Release 16),” 2021 ETSI [Online]. Available: https://www.techstreet.com/standards/etsi-ts-129-518?product_id=2208730

[5] Oracle Inc., Cloud-Native Core, “Network Slice Selection Function (NSSF) Cloud Native User’s Guide,” 2022. [Online]. Available: https://docs.oracle.com/en/industries/communications/cloud-native-core/2.2.1/nssf/introduction1.html#GUID-32FF1229-9937-4CD7-A465-E56CF6459DF5

[6] J. Cáceres-Hidalgo and D. Avila-Pesantez, "Cybersecurity Study in 5G Network Slicing Technology: A Systematic Mapping Review," 2021 IEEE Fifth Ecuador Technical Chapters Meeting (ETCM), 2021, pp. 1-6, doi: 10.1109/ETCM53643.2021.9590742.

[7] CompTIA, DDoS response and mitigation guide, “What is a DDoS attack and how does it work?” 2022. [Online]. Available: https://www.comptia.org/content/guides/what-is-a-ddos-attack-how-it-works

[8] M. K. Forland, K. Kralevska, M. Garau and D. Gligoroski, "Preventing DDoS with SDN in 5G," 2019 IEEE Globecom Workshops (GC Wkshps), 2019, pp. 1-7, doi: 10.1109/GCWkshps45667.2019.9024497.

[9] Yong-joon Lee, Hwa-sung Chae & Keun-wang Lee (2021) Countermeasures against large-scale reflection DDoS attacks using exploit IoT devices, Automatika, 62:1, 127-136, doi: 10.1080/00051144.2021.1885587.

[10] João J.C. Gondim, Robson de Oliveira Albuquerque, Ana Lucila Sandoval Orozco, “Mirror saturation in amplified reflection Distributed Denial of Service: A case of study using SNMP, SSDP, NTP and DNS protocols,” in Future Generation Computer Systems, Volume 108, 2020, Pages 68-81, ISSN 0167-739X, https://doi.org/10.1016/j.future.2020.01.024.

[11] CloudFlare, Inc. (2022, April 1). Memcached DDoS Attack [Online]. Available: https://www.cloudflare.com/learning/ddos/memcached-ddos-attack/

[12] SPIFFE, Cloud-Native Foundation Project. (2022, April 1). A universal identity control plane for distributed systems [Online]. Available: https://spiffe.io/docs/latest/spiffe-about/spiffe-concepts/

[13] Network Service Mesh, Cloud-Native Foundation Project. (2022, April 1). Enterprise Concepts Technical Documentation [Online]. Available: https://networkservicemesh.io/docs/concepts/enterprise_users/

[14] Kubernetes, Linux Foundation Project. (2022, April 1). Production grade container orchestration [Online]. Available: https://kubernetes.io/

[15] Tigera, Inc. (2022, April 1). Project Calico [Online]. Available: https://www.tigera.io/project-calico/

[16] D. Bhamare, R. Jain, M. Samaka and A. Erbad, “A survey on service function chaining,” in Journal of Network and Computer Applications, Volume 75, pp. 138-155, ISSN 1084-8045, 2016 https://doi.org/10.1016/j.jnca.2016.09.001.

[17] European Telecommunications Standards Institute (ETSI), “Open-Source Management and Orchestration”, 2022. [Online]. Available: https://www.etsi.org/technologies/open-source-mano

[18] M. Varela, L. Skorin-Kapov and T. Ebrahimi, “Quality of Service Versus Quality of Experience,” in Möller S., Raake A. (eds) Quality of Experience. T-Labs Series in Telecommunication Services. Springer, Cham, 2014. doi: https://doi.org/10.1007/978-3-319-02681-7_6.

[19] Cisco Systems, Inc. (2022, April 1). Segment-Routing Configuration Guide for Cisco ASR 9000 Series Routers (IOS XR Release 6.6.x) [Online]. Available: https://www.cisco.com/c/en/us/td/docs/routers/asr9000/software/asr9k-r6-6/segment-routing/configuration/guide/b-segment-routing-cg-asr9000-66x/b-segment-routing-cg-asr9000-66x_preface_00.html

[20] European Telecommunications Standards Institute, Technical Specification, “ETSI TS 129 531: 5G System; Network Slice Selection Services, Stage 3 (Version 15.6.0 Release 15),” 2021 ETSI. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/129500_129599/129531/15.06.00_60/ts_129531v150600p.pdf

[21] B. Dab, I. Fajjari, M. Rohon, C. Auboin and A. Diquélou, "Cloud-native Service Function Chaining for 5G based on Network Service Mesh," in 2020 IEEE International Conference on Communications (ICC 2020), 2020, pp. 1-7, doi: 10.1109/ICC40277.2020.9149045.

[22]] Internet Engineering Task Force (IETF), “Source Packet Routing in Networking (SPRING) Problem Statement and Requirements”, 2016. [Online], Available: https://datatracker.ietf.org/doc/html/rfc7855

[23] OpenDaylight Project. (2022, April 1). A modular open platform for customizing and automating networks of any size and scale [Online]. Available: https://www.opendaylight.org/

[24] OpenAirInterface, 5G software alliance for democratising wireless innovation. 2022. [Online]. Available: https://openairinterface.org/

[25]] D. Smith, “Multus takes a leading role in container networking”, RedHat Blog. December 4, 2020. [Online]. Available: https://www.redhat.com/en/blog/multus-takes-leading-role-container-networking

[26] B. Dzogovic, B. Santos, B. Feng, V. T. Do, N. Jacot, V. D. Thanh, “Optimizing 5G VPN+ Transport Networks with Vector Packet Processing and FPGA Cryptographic Offloading” in Bentahar J., Awan I., Younas M., Grønli TM. (eds) Mobile Web and Intelligent Information Systems (MobiWIS 2021). Lecture Notes in Computer Science, vol 12814, 2021. Springer, Cham, doi: https://doi.org/10.1007/978-3-030-83164-6_7.

[27] OpenStack Project. (2022, April 1). Open-Source Cloud Software, 2022. [Online]. Available: https://www.openstack.org/

[28] Amazon, Inc. (2022, April 1). Amazon Web Services (AWS) [Online]. Available: https://aws.amazon.com/

[29] Microsoft Corporation. (2022, April 1). Microsoft Azure Cloud. [Online]. Available: https://azure.microsoft.com/

[30] FRRouting Project. (2022, April 1). A free and open-source Internet routing protocol suite for Linux and Unix Platforms [Online]. Available: https://frrouting.org/

[31] Segment Routing Project. (2022, April 1). A source-routing architecture that seeks the right balance between distributed intelligence and centralized optimization [Online]. Available: https://www.segment-routing.net/

[32] Calico-VPP Kubernetes dataplane. (2022, April 1). Implementation guide [Online]. Available: https://projectcalico.docs.tigera.io/getting-started/kubernetes/vpp/getting-started

[33] MetalLB, A Kubernetes load-balancer for bare metal clusters, 2022. [Online]. Available: https://metallb.universe.tf/

[34] Cisco systems, Inc., Bidirectional Forwarding Detection (BFD). (2022, April 8). Technical documentation [Online]. Available: https://www.cisco.com/c/en/us/td/docs/ios/12_0s/feature/guide/fs_bfd.html

[35] OpenStack Neutron documentation, 2022. [Online]. Available: https://docs.openstack.org/neutron/latest/