RP1 research paper

Technical feasibility of Segment Routing Traffic Engineering to steer traffic through VNFs

February 9, 2019

Supervisors:

SURFnet: Marijke Kaat and Eyle Brinkhuis

Ronald van der Gaag
Security and Network Engineering
University of Amsterdam
ronald.vandergaag@os3.nl

Mike Slotboom
Security and Network Engineering
University of Amsterdam
mike.slotboom@os3.nl

***Abstract*—Steering traffic through Virtual Network Functions (VNFs) in a network could deliver tailored services to end users [4], such as firewalling and traffic inspection, as well as load balancing. Using Segment Routing (SR), which leverages the source routing paradigm [11], traffic steering can be implemented in a network.**

In this research, the implications and maturity of using SR over the MPLS data plane (SR-MPLS) in order to steer traffic through VNFs is examined. These VNFs can either be SR-aware or SR-unaware, which resulted in two scenarios. To verify the maturity of these scenarios, a Proof of Concept (PoC) was built in a virtual testbed, with one SR-aware VNF and one SR-unaware, which was made SR-aware using a router as proxy.

A firewall function was used as VNF in this testbed, but a fully SR-aware firewall appliance was not found. To test the fundamentals of using SR-MPLS to steer traffic through the VNF, the proxy functioned as firewall in this scenario. In both scenarios, SR-MPLS was configured including SR policies and the traffic was steered through the VNF accordingly.

This PoC indicated the maturity of this solution: the concept to steer traffic with SR-MPLS through the VNFs works, but SR-MPLS aware VNFs are not yet available.

***Keywords*— Segment Routing, Virtual Network Functions, Multi-protocol Label Switching, Proxy**

I. Introduction

Steering traffic through Virtual Network Functions (VNFs) in a network could deliver tailored services to end users [4], such as firewalling and traffic inspection, as well as load balancing. In this project we investigated the technical feasibility of using Segment Routing [11] to steer and deliver network traffic to VNFs as a use case for an Internet Service Provider (ISP). Segment Routing is a source routing paradigm and is used to steer a packet through a network using a predefined list of instructions (segments) [11].

SURFnet currently has a pilot where network traffic has to flow through predefined VNFs. In order to do so, several GRE tunnels are used per institute. This research examined if Segment Routing is able to steer traffic to VNFs and what the implications are in order to fully replace the current tunneling solution for SURFnet pilot.

II. Problem Description

During the pilot at SURFnet, regarding the implementation of VNFs (e.g. Firewall as a Service [8]) using Juniper Contrail [14], it became clear that the used technique (GRE tunnels) to steer traffic through these VNFs, was not sufficient for a deployment on a bigger scale.

In this pilot, network traffic is tunneled with GRE encapsulation through a predefined set of VNFs, before the packets reach their destination. For every customer, a set of new GRE tunnels has to be configured and maintained per core router. The pilot had to make use of these tunneling techniques due to the lack of tunneling functionality on OSI Layer 2 in the used version of Juniper Contrail. This limited SURFnet in the use of tunnelling techniques (i.e. MPLS-over-GRE/UDP or VXLAN) [14].

Maintaining a set of tunnels per institute adds configuration complexity, is non-scalable and results in unwanted overhead and administration. As a result of the added complexity,

testing and troubleshooting this solution is hard, which leads to unreliability. Moreover, the limitations of hardware results in a limited capacity when using tunneling [1].

Segment Routing could be a method to simplify the steering of traffic instead of the current manually set up GRE tunnels, which simplifies the possibilities for traffic engineering. With SURFnet already deploying Segment Routing in their new network, it would be a great opportunity to use this for steering traffic through the desired VNFs as well.

III. Background

A. Segment Routing Architecture

The Segment Routing (SR) Architecture consists of one overarching SR domain, which consists of nodes participating in the source-based routing model [11]. The SR domain can have a centralized, distributed or hybrid structure, where a SR controller [5] can be in place to enforce SR policies and assign SIDs [11]. Within this domain, the network traffic is sent to a destination using segments. The segments that the packet has to traverse are stored in the headers. This makes that the whole state is stored in the packet itself.

According to RFC 8402, *"a segment can represent any instruction, topological or service based in a SR domain"*[11], which means that a segment ID (SID) can be assigned to a node, network prefix (i.e. Prefix-SID and Anycast-SID), a link between nodes (i.e. Adjacency SID) or a specific service (i.e. Service SID) [11]. This makes it possible to make the network packets traverse specific nodes (with a node SID), network prefixes (with an Anycast-SID), a specific link (with an Adjacency SID) and also a service (with a Service SID)[9]. These SIDs are redistributed by the IGP throughout the SR domain.

Two types of data planes for SR are available: Segment Routing over Multi-Protocol Label Switching (SR-MPLS) and Segment Routing over IPv6, which is called Segment Routing version 6 (SRv6) [11]. In SR-MPLS, the MPLS labels are exploited to store the SIDs. The MPLS label stacking [17] can be used by SR to push multiple SIDs on a packet. There are no additional requirements to the MPLS data plane to use SR-MPLS [11]. SRv6 prepends the packet with a Segment Routing Header (SRH) and uses IPv6 addresses as SIDs [11].

In order to participate in a SR-MPLS domain, a node has to have a SR Global Block (SRGB) configured. This SRGB is used to reserve a set of local node labels for global segments, which are available within the SR domain and advertised by the Internal Gateway Protocol (IGP) (e.g. ISIS) [11]. A Binding SID is a local or global SID used to enforce a SR policy based on one SID.

Furthermore, a network function, whether it is virtualized or not, can also have a SID assigned as a node in the network [9] if it is supported.

The SR policy uses segment lists to determine where the packet has to be sent through based on the destination and this can be instantiated by a SR controller or computed at the ingress node of the SR domain [11]. This policy uses Label Switched Paths (LSPs) to configure the route the traffic will follow when using MPLS as dataplane. Using the SRGBs, the routers are able to define a table including the segments and shortest paths to these segments. This is depicted as an example in Figure 1, where a packet is sent from R1 to R6 using SR.

In this figure, there are two SR-MPLS instructions visualized. The first being the SIDs pushed by R1. With these SIDs, R1 determines which nodes the packet has to traverse. Because the route from R1 to R6 via R2 and R5 has a lower cost, this is the preferred route and only SID 16006 is pushed to the packet (i.e. Node SID of destination R6). However, if a label is added with the Node SID of an intermediate router, the traffic has to flow through this first (e.g. R4 in Figure 1).

The second SR-MPLS instruction is the 'popping' of the SIDs, which means that a SID is stripped from the packet if this node (i.e. the penultimate node) is directly attached to the node with the specific SID [11]. In Figure 1 R3, R4 and R5 are popping the first labels before they send it to the next router.

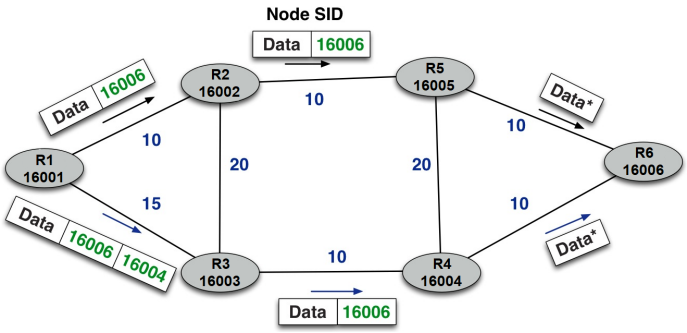


Fig. 1. Example of Segment Routing [16]

IV. Research Questions

With the problem description and background, we investigated whether SR-MPLS could replace the current GRE tunneling solution, regarding steering traffic through VNFs. The practical implications and the maturity of this solution need to be known, in order to state whether Segment Routing could replace the current situation. With this in mind, the main research question is stated as follows:

What are the practical implications and the maturity of steering network traffic through VNFs using Segment Routing over MPLS instead of the current GRE tunneling solution for SURFnet?

A. Sub research questions

There are two sub research questions to assist the main research question. These questions are:

1) **What are the implications for the SURFnet network using SR instead of the current tunneling solution?** The first question gives insight what to expect when the

[1]This was concluded from a meeting with Eyle Brinkhuis

current GRE tunneling solution at SURFnet is replaced or complemented by a SR-MPLS solution. Moreover, the operational implications when using SR-MPLS are also taken into consideration.

2) **What is the technical feasibility of steering traffic through VNFs in SURFnet's network using SR-MPLS?** This sub question builds upon the first question and provides an answer if it is possible to steer traffic through VNFs using SR-MPLS in SURFnet's production environment, regarding the current maturity of this technology. SRv6 is out of scope in this sub question, because SR-MPLS is used in SURFnet's network.

V. Related work

An experiment from Filsfils et. al in 2015 [10] poses several use cases to implement SRv6, such as "Traffic Engineering using Segment Routing Tunnels" and "Service Function Chaining" (SFC) [10]. Although this research only covered SRv6, the fundamentals of these use cases also apply to SR-MPLS and are used in this research.

A paper about VNF function chaining from Abdelsalam et. al [2] gives a clear overview of the SR components. Abdelsalam et. al derives the components in two main classes: *SR-aware VNFs* and *SR-unaware VNFs* [2]. The characteristics of these SR-aware and SR-unaware VNFs are used as basis for the scenarios, which are worked out in the results in Section VII-A3.

VI. Methodology

A. Approach

1) Alternatives based on existing related work: Based on the related work, the implications of using SR in combination with VNF were investigated, resulting in two scenarios. Moreover, the related work was examined to gain an indication of the maturity of SR-MPLS regarding the traffic steering through VNFs.

2) Proof of Concept: With a Proof of Concept (PoC) we determined whether SR-MPLS is feasible to steer traffic through the desired VNFs using a test environment. In order to do so, the virtual setup consisted of:

- three SR-aware Juniper routers (JunOS vMX 18.2R1.9);
- one SR-aware Juniper "proxy" (JunOS vMX 18.2R1.9);
- one firewall appliance (SR-unaware VNF) using Iptables (Ubuntu Server 18.04);
- two virtual machines for simulating a web server with serving two web pages using Docker and a client to request the pages (Ubuntu Server 18.04).

This setup was built in the available testbed at SURFnet using OpenStack and Juniper WiStar and is depicted in Figure 2.

It is important to prove that the network traffic was sent through the VNF. We proved this with two methods. The first is using a different firewall policy for different clients. With the different policies, we can see if the network traffic behaves as defined in the policies. If this is the case (i.e. port 8181 is blocked for one of the clients), then the traffic is successfully blocked by the firewall.

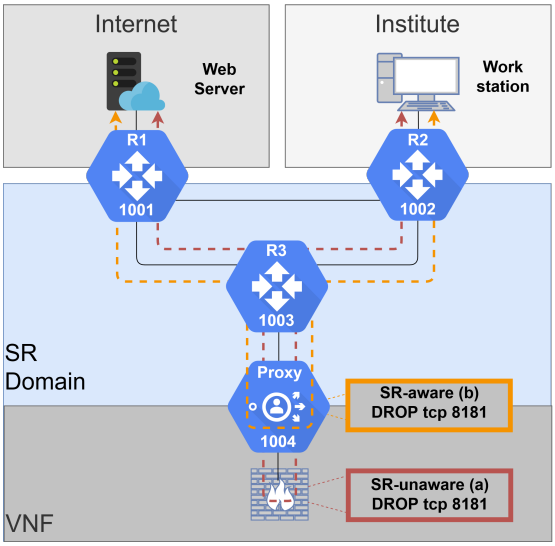


Fig. 2. Overview of the testbed with two scenarios

The second being with a TCPdump at the VNF side of the proxy. A TCPdump prints out the contents of the packets on a network interface [13]. With a TCPdump, we can see which packets flowed through the firewall. If the network packets between the Internet and Institute are visible in this TCPdump, we can conclude that the packets were successfully steered through the VNFs.

The tested scenarios in the PoC were successful if the network traffic was steered through the firewall and was filtered on the basis of the used firewall rules.

B. Scope

The scope of this research was limited to the examination of the applicability of SR-MPLS in combination with VNFs in the context of SURFnet. With this scope, we examined the alternatives when using SR-MPLS with VNFs and the maturity of the technique when this is used. The PoC was delivered to test the maturity of the alternatives.

To scope the VNFs further down, only the firewall function was considered. The firewall function is used in the current pilot at SURFnet and the approach for this VNF will be used for other network functions, because these functions share the same network characteristics.

VII. Results

This section describes the results of our research. In this section both sub questions are answered.

A. Implications of using Segment Routing instead of current tunneling solution

1) Current solution: In the pilot solution, every institute is assigned a set of two GRE tunnel per core router: The first GRE tunnel is used for the traffic originated from the internet and destined for the institute, where the second GRE tunnel is configured in reverse to enable symmetrical traffic. Because of the fact that for each institute new tunnels need

to be made and maintained, this solution is complex and not scalable. A detailed description and a visual overview of the current situation is given in Appendix A.

2) Proposed environment: In the proposed environment SR-MPLS is used to steer traffic through VNFs instead of the current tunneling solution. In order to use SR-MPLS in the network, an IGP will be used to exchange SID labels throughout the network.

a) Operational implications of proposed environment: When SR-MPLS is used (e.g. in combination with VNFs), the network is a trusted domain. Because the security implications are the same as implementing MPLS in a network, RFC 4381 regarding the security of BGP/MPLS IP Virtual Private Networks (VPNs) applies in this situation [11]. This means that filters at the boundaries of this domain need to be in place to prevent tempering of the SR network traffic. Moreover, extensions of protocols using SR are available (e.g. ISIS-SR-Ext [15]) [11] to redistribute the SIDs and to leverage additional security mechanisms, which are part of these protocols (e.g. encryption and authentication).

If a VNF (e.g. a firewall function) is assigned a SID, the traffic which is sent towards this segment has to flow through the function. In order to check if the traffic does flow this way and to gain insight in congestion and performance, the monitoring of the traffic has to be in place, which is stated in RFC 8403 and RFC 8287 [11]. Using a Path Monitoring System (PMS), the paths through the SR domain can be monitored with a traceroute using the configured LSP. This method is called 'LSP Data-Plane Monitoring'[12]. PMS can also be applied to check if the traffic flows through the VNFs based on the corresponding Segment SID. When the network traffic in the virtual function itself is also monitored, it is possible to verify if the function is able to process the data[12].

b) SR-awareness of VNFs: Assuming that a virtual firewall is placed in the reference network as depicted in Figure 3 and the traffic has to flow through this VNF. This means that the firewall function has to be (made) SR-aware to be reached in the SR domain. A proxy can be implemented to enable this awareness if the function itself is not SR-aware [2] [9]. To connect the proxy with the VNFs, multiple types of interfaces are possible, either a physical interface or sub-interfaces such as VLANs [9].

When a proxy is used, the SR information is stripped before the packet is sent to the SR-unaware functions, which results in a normal IP packet the function is able to process [9]. On top of that, when the traffic is sent back to the SR-domain, the proxy pushes the SR information and forwards the packet based on the applied policy [9]. It is also possible to use the proxy as an intermediate in an SFC, when chaining several services together [1].

Four types of SR-proxies (i.e. Static, Dynamic, Shared Memory and Masquerading proxies) are available with different characteristics and use cases[9].

1) A static proxy is used to remove the SR information of a packet and send the stripped packet to the intended SR-unaware function [9]. This proxy is static, because it can only be used in one service policy (i.e. segment list) at a time [9]. In order to make this proxy process symmetrical traffic, another proxy has to be configured [9].
2) The dynamic proxy is an improved version of the static proxy, which learns the SR information before stripping it from the packet [9]. The SR information is stored in the cache of the proxy and is added to the packet, if this comes back from the service.
3) The third proxy uses shared memory to hide the SR information from the SR-unaware VNF, if the VNF runs at the same compute node as the proxy. An application of this proxy is a SR-aware vRouter, which runs at a container host and forwards the traffic from and to the containers [9].
4) The masquerade proxy is only used in SRv6 and substitute the destination address of a packet with the last SID in the packet, resulting in a normal IPv6 packet [9].

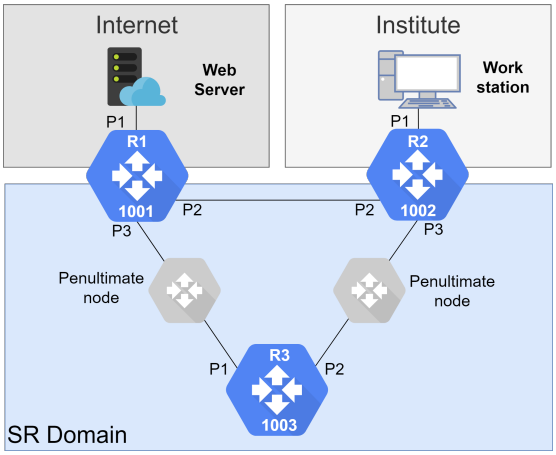
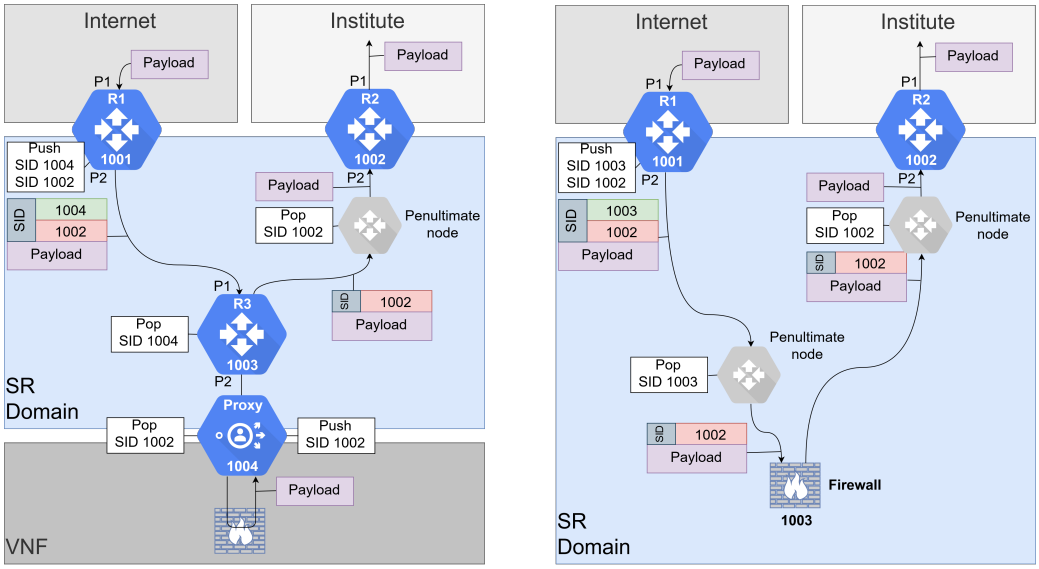


Fig. 3. Basic reference network

3) Scenarios of proposed environment: Using SR in production can have different approaches. A simplified network setup of SURFnet8, shown in Figure 3, is used as a reference network, where a virtual firewall will be placed to explain the concept. This reference network consists of a SR domain connecting the internet with the institutes, using three SR edge routers. It is possible that additional nodes (i.e. the grey nodes in Figure 3) are placed to connect these routers. In this case, two nodes are placed, which are the 'penultimate nodes' for the network traffic. This means that the traffic will flow from router R1 to R3 with an extra hop in between and the SID in a packet is 'popped' by the penultimate node, if this SID is the next SR node [11].

Two scenarios were defined when using either SR-aware or SR-unaware VNFs at SURFnet, as depicted in Figure 4.

a) Effects of using scenario (a): In scenario (a), the VNF is SR-unaware and a SR-aware proxy is used to 'pop' and 'push' the needed SR-labels for the unaware VNF. This proxy can be a SR-MPLS capable network device (e.g. router) or a dedicated proxy, which main focus is the proxy function.



(a) SR-unaware VNF, Dedicated SR-proxy

(b) SR-aware VNF as part of SR-domain

Fig. 4. The two identified scenarios when using Segment Routing to steer traffic in SURFnet8

The Proxy SID is distributed throughout the SR domain and if the proxy is a virtual appliance, the VNF cluster is able to migrate to another host. Using Topology Independent Loop-free Alternate Fast Re-route (TI-LFA), which is available in SR domains, the traffic can be rerouted to the new destination[3].

When a packet from the Internet is destined for the Institute, the first router (i.e. ingress router R1) in the SR-domain will apply a configured SR policy pushing two SIDs on the packet. First being the SR-proxy (i.e. SID 1004) and second being the SR node facing the Institute (i.e. 1002). This makes the network traffic flow towards the VNF before it goes to the Institute. The penultimate node (e.g. an intermediate router in the SR domain) to the proxy will pop the proxy label (i.e. SID 1004). The proxy will then strip the SR information and sends the packet to the VNF.

If the packet returns the label of the SR-endpoint for the Institute is added to the packet. However, when the packet had more than this label before it was stripped by the proxy, these labels also need to be pushed back onto the packets. This makes the proxy statefull in this scenario. The packet will continue its path towards the Institute (i.e. SID 1002).

Please note, that when the penultimate node is R3, the Institute SID will be 'popped' here after the packet being sent by the proxy. This is because R3 has a direct link to the Institute. For steering the network traffic bidirectional, the SR-endpoint which is facing the Institute has to insert the same label for the proxy (i.e. 1004) and the SR-endpoint facing the Internet (i.e. 1001).

b) Effects of using scenario (b): Next to SR-unaware VNFs, it is possible to use SR-aware functions [2]. In scenario (b), the firewall is part of the SR domain as a SR node, which sends the traffic further to the next nodes after processing, without stripping the SR information from the packets like the proxy does in scenario (a). This way, additional measures (e.g. MPLS-capable and maintaining both SR policies as well as adjacencies with other nodes) have to be in place in order to work. When these requirements are met, this solution is the most flexible. Besides the flexibility, none of the components is stateful due to the fact that the whole state is stored in the stacked SIDs. Furthermore, some SR-aware functions are able to internally process the SR headers based on policies (e.g. SR-aware Firewall).

In order to steer the traffic bidirectional with the same path, the labels which are pushed by R2 have a different destination. This means that R2 has to insert the same label for the VNF (i.e. 1003) and the SR-endpoint facing the Internet (i.e. 1001).

c) Scenarios advantages and disadvantages: The two scenarios are slightly different from each other, which results in advantages as well as disadvantages per scenario. In Table I, the identified situations are compared. The criteria for these scenarios were scalability and maintainability, regarding the additional effort when the scenario becomes reality in the network topology. The dynamic or static configuration of the SR policies are also taken into consideration.

B. Technical feasibility of using Segment Routing in combination with VNFs at SURFnet

Whether it is possible to implement SR-MPLS in such a way that the traffic will flow through the predefined set of VNFs (i.e. services), depends on configuration of the hardware and software used in the network [9]. To verify the technical feasibility of SR-MPLS to steer traffic through the VNF, the maturity of this solution was analyzed. This consisted of three parts: Examination of the maturity of SR-MPLS with VNFs and proxies and testing the possible solutions in a PoC.

1) Maturity of SR-MPLS-aware VNFs: The development of SR-MPLS-aware VNFs was not covered in the available

TABLE I
ADVANTAGES AND DISADVANTAGES OF THE FOUR SCENARIOS

Scen.	Advantages	Disadvantages
(a)	- Every VNF can be used in this scenario due to the Proxy	- At least one additional node is needed, which could become a bottleneck - Proxy needs to maintain state if, a static or dynamic proxy is used
(b)	- No state have to be stored at the nodes, because the state is kept in the packet - Most dynamic, reachability based on distribution of own SID - Proxies are not needed to strip SR information from the packet	- All VNFs need to be SR-aware and MPLS-capable

research. The research that was available only examined the application of SRv6 (e.g. SRv6 support for Snort, Iptables and NFtables [9]) instead of SR-MPLS, and therefore a fully SR-MPLS-aware firewall solution was not found during this research.

2) Maturity of SR-MPLS-aware proxies: When a proxy is used to enable the SR-awareness of the VNF, this proxy has to be SR-aware as well as MPLS-capable. The choice in implementations of open source SR-MPLS-proxies is small; only two solutions were found:

1) The Fast Data Project (i.e. FD.io), a Linux Foundation Project, leverages an open source SR-MPLS and SRv6 implementation in its Vector Packet Processing (VPP) software router [6]. According to Clad et. al, only the static SR-proxy as explained in Section VII-A2b is available for SR-MPLS in VPP at the moment [2], whereas two other types of proxies (i.e. dynamic and shared memory) are still in development for SR-MPLS.
2) Cumulus Linux was found to be capable of SR-MPLS as an early access feature in a specific type of physical switches (i.e. Mellanox switches) [7]. Cumulus Linux is therefore still in development is not applicable in the situation of a virtualized VNF cluster.

The proxy function can also be fulfilled by a (virtual) router. This way the proxy is able to be part of the SR domain, including routing protocols (e.g. ISIS and BGP). When a VNF is directly connected to this device, the SR information is stripped by the router and the packet is sent to the VNF.

3) PoC approach: In order to verify the workings of the scenarios of SR-MPLS for SURFnet8, the two scenarios were tested on feasibility: one with a SR-unaware VNF (i.e. Scenario (a)) and one with a SR-aware VNF (i.e. Scenario (b)), with the use of a dedicated proxy. The configuration of this PoC can be found on GitHub [18].

Scenario (b) was the only scenario with SR-aware VNFs, which makes it possible to compare with the proxy and SR-unaware VNFs. The basics of the testbed consist of routers and virtual machines, stated in Section VI-A2.

4) Configuration of scenario (c): After the network was configured, the Juniper routers were assigned a Node SID corresponding with the SIDs in Figure 2. These SIDs were advertised in the SR domain using the extended version of ISIS. However, the networks of the Internet and Institute were advertised using BGP to enable the steering of the traffic to these groups using LSPs and segment lists. These segment lists were configured on the ingress nodes of the network (i.e. R1 and R2) and contained the Node SIDs of the routers the traffic had to flow to.

To generate traffic, two web pages were served by the Internet virtual machine using Docker. One page was hosted on TCP port 8080 and the other on TCP port 8181. The firewall function was built using ipv4-forwarding and the filter FORWARDING chain of Iptables to block traffic. In this PoC the traffic destined for the Internet with TCP port 8181 had to be blocked to verify the working of the firewall.

In this scenario, a SR-unaware firewall was connected to the SR domain through the proxy. When the packets originated from the Internet or Institute, the packets got only the label 1004, which means that the proxy was the destination these packets for R1 and R3. Moreover, the incoming traffic on the proxy from the SR domain, was forwarded to the firewall using a static forwarding policy. Because the proxy was part of the SR domain and the packets to this proxy were sent by R3, the label was 'popped' by R3, which resulted in an IP packet instead of an MPLS encapsulated packet.

On the proxy, two segment lists to R1 and R2 were configured to be able to steer the traffic to the intended destination. This was 1002 for the edge router towards the Institute and 1001 for the edge router towards the Internet.

a) Results from scenario (c) in test environment: Scenario (a) was implemented in the PoC, where the traffic was sent through the firewall using the MPLS encapsulation with the intended SIDs.

```
root@R1> traceroute institute
traceroute to Institute (10.0.20.10)
 1  R3-R1 (10.0.13.2)
     MPLS Label=1004 CoS=0 TTL=1 S=1
 2  Proxy-R3 (10.0.34.2)
 3  Firewall (10.0.40.11)
 4  Proxy-Firewall (10.0.40.1)
 5  R3-Proxy (10.0.34.1)
     MPLS Label=1002 CoS=0 TTL=1 S=1
 6  R2-R3 (10.0.23.1)
 7  Institute (10.0.20.10)
```

Fig. 5. Scenario A: Traceroute results from R1 to Institute

As is depicted in Figure 5 the route is shown that the packet travels from router R1 to the virtual machine of the Institute. A network is passed per hop, reaching the firewall as the third hop. Only between R1 and R3 as well as Proxy and R3 an MPLS label is shown. This label was the result of the configured SR segment lists, which shows the working of this situation.

[2]The support for SRv6 proxies is more extensive, due to the availability of a proxy function in VPP and a Linux kernel implementation of SRv6 called 'Srext', which can function as proxy [2]

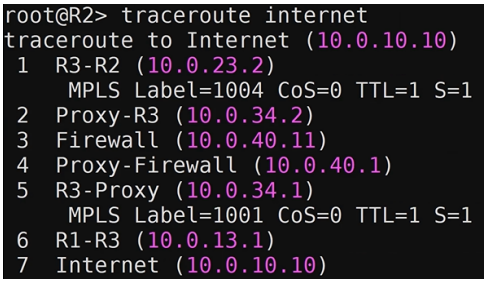


Fig. 6. Scenario A: Traceroute results from R2 to Internet

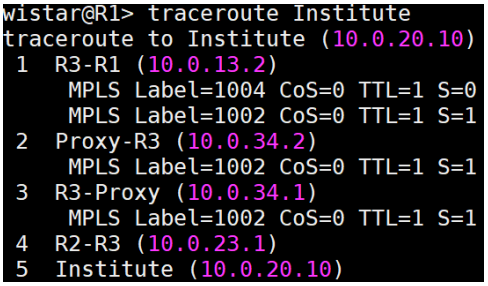


Fig. 7. Scenario B: Traceroute results from R1 to Institute

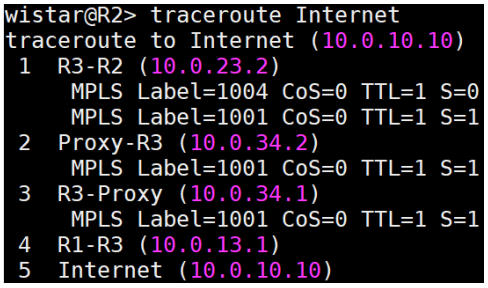


Fig. 8. Scenario B: Traceroute results from R2 to Internet

The traceroute in the from Institute to Internet are shown in Figure 6. In this screenshot, the traffic flowed from router R2 trough the firewall and to the virtual machine Internet, which served the web pages. The difference in this figure is the starting point and destination, which results in a different MPLS label in step 5 of the traceroute.

This was verified by checking the output of the TCPdump on the firewall, which showed that the traffic went through the firewall and blocked the traffic on port 8181. This confirmed the working firewall in combination with the proxy.

5) Configuration of scenario (d): The PoC setup of Scenario (b) used the same base as Scenario (a), with one alteration: the role of the SR-unaware firewall was fullfilled by the router called 'Proxy', which was configured as SR-aware firewall, due to the lack of a fully functioning SR-MPLS aware VNF.

The labels are in a similar manner added to the packets using the segment lists as Scenario (a) in the test environment, but also contained the destination SIDs. A packet from the Internet to the Institute therefore had two SIDs in the MPLS header: 1004 (i.e. firewall) and 1002 (i.e. R2). The packet originated from the Institute and headed to the Internet also received label 1004, but the destination was 1001 (i.e. R1).

Because the 'Proxy' functioned as firewall, this Juniper router had to filter the MPLS traffic. This could not be done in the test environment due to restrictions of the MPLS filtering on this device, resulting in the forwarding of all the packets with TCP ports 8080 and 8181, instead of only TCP port 8080. This was therefore not a firewall, but to verify the working of this scenario, this router was still used.

a) Results from scenario (d) in test environment: Scenario (b) was successfully tested, where the traffic was steered through the proxy, which functioned as firewall. In this case two labels where pushed in the packet. The traffic flow of the packet was verified in this scenario using traceroute as well.

In Figure 7, the traceroute is shown from R1 to the Institute. In this scenario, the label of the destination (i.e. 1002) was pushed onto the packet on R1 and was not popped before reaching the firewall (i.e. the second hop in the traceroute). This way, the packet continued its way to the Institute after leaving the firewall.

The route from R2 to the Internet showed a similar result, as is shown in Figure 8. The only difference is the destination label, which was 1001. This shows the bidirectional support of this scenario.

VIII. Discussion

Based on the results, the packets in the proposed environment are assigned labels at the ingress of the SR domain using a configured segment list in a SR policy. This way, the services a service provider (e.g. SURFnet) offers, can be used by institutes using the policies. When a service is enabled for a customer, this has to be set in the SR policy. When a SR controller is used in the network, the administration of these SR policies can be performed by a SR controller.

In order to implement this concept, two scenarios were identified, when using SR-MPLS in combination with VNFs in production. Although this research is based on the use case of SURFnet, the results of the concept are applicable to any organization using Segment Routing to steer traffic through VNFs.

The application of SR-MPLS with SR-unaware VNFs requires a proxy. In the PoC this proxy was demonstrated using a Juniper vMX router in Scenario (a). Based on the results, this scenario was successfully implemented and steered the traffic through the firewall. The maturity of this static proxy is therefore on a basic level and can be further tested on performance in a pilot environment. Please note, that only a static proxy is used in this PoC in Scenario (a), where dynamic and shared-memory proxies were still in development for SR-MPLS, as is stated in Section VII-B2.

Using a proxy makes the integration of SR-unaware VNFs possible in a SR domain, but when the VNF is SR-aware, this proxy is not necessary. Therefore, when the VNFs in a network are all SR-aware, Scenario (b) will be most suitable. However,

the maturity of this SR-aware scenario is still insufficient to be used in production based on results of the PoC. The concept is shown working in the PoC, but the lack of SR-MPLS-aware VNFs causes that this solution is still in development.

IX. Conclusion

In this research, the following research question is answered: *What are the practical implications and the maturity of steering network traffic through VNFs using Segment Routing over MPLS instead of the current GRE tunneling solution for SURFnet?* In order to answer this question, two sub questions are used, which are stated in Section IV-A, regarding the implications and the maturity.

When using labelling with SR-MPLS instead of static GRE tunneling, delivering the network traffic to a VNF will be simplified. If the traffic has to flow through a specific service, this can be implemented by pushing one SID to a packet as intermediate hop. In order to examine this solution, two scenarios were identified, with their own implications, regarding scalability, maintainability and support for symmetrical traffic.

The state-of-the-art of this technique, regarding the maturity of the SR-aware VNFs and the maturity of proxies, was examined as part of the second sub question. Two scenarios (Scenario (a) and Scenario (b)) were tested using SR-MPLS in a Proof of Concept (PoC).

The implementation of SR-MPLS to steer traffic through SR-aware VNFs is not mature enough to use in production. In the PoC, the functionality of SR-aware firewall was not fully tested in Scenario (b), due to the limitations of the MPLS filtering on the used Juniper router. This router was used, due to the absence of a fully SR-MPLS-aware firewall. This scenario was therefore not successful. However, the steering of the traffic was in place, where the packets followed the path of the predefined segment lists.

The use of a proxy is also still in development for SR-MPLS and only the static proxy is available in SR-MPLS using VPP or a router. This static proxy is verified in the PoC by using a Juniper vMX router. This scenario was called a success, because the traffic was steered through the firewall and the network traffic was filtered based on the firewall rules.

X. Future work

This research only focused on the consequences and the possibility of using Segment Routing (i.e. SR-MPLS) to steer traffic through VNFs. SURFnet's network SURFnet8 was used as a reference to apply the architecture and solutions on.

The operational implications were considered during this research, but a monitoring system was not implemented in the PoC. In future work, this research could be continued by practically implementing SR-MPLS in a pilot including the PMS to test the performance and the proposed scenarios.

Furthermore, developing a proxy or feature to make VNFs capable of handling SR-MPLS is also valuable future research.

References

[1] A. Abdelsalam. Chaining of segment routing aware and unaware service functions. https://networking.ifip.org/2018/images/2018-IFIP-Networking/D1-Abdelsalam.pdf, 2018.

[2] A. Abdelsalam, F. Clad, C. Filsfils, S. Salsano, G. Siracusano, and L. Veltri. Implementation of virtual network function chaining through segment routing in a linux-based nfv infrastructure. In *2017 IEEE Conference on Network Softwarization (NetSoft)*, pages 1–5, July 2017.

[3] A. Bashandy, C. Filsfils, B. Decraene, S. Litkowski, P. Francois, D. Voyer, F. Clad, and P. Camarillo. Topology Independent Fast Reroute using Segment Routing. Internet-Draft draft-ietf-rtgwg-segment-routing-ti-lfa-00, Internet Engineering Task Force, December 2018. Work in Progress.

[4] J. Blaser. On service chaining and segment routing. http://www.scriptiesonline.uba.uva.nl/document/657875, June 2018. Accessed on November 9th 2018.

[5] R. Bonica. A segment routing (sr) tutorial. https://www.nanog.org/sites/default/files/1_Bonica_Tutorial_Segment_Routing.pdf, 2016. Accessed on November 9th 2018.

[6] P. Bratach and A. Rolland. Segment routing - cumulus linux 3.7 - cumulus networks. https://docs.fd.io/vpp/19.04/srmpls_doc.html. Accessed on January 22th 2019.

[7] P. Bratach and A. Rolland. Segment routing - cumulus linux 3.7 - cumulus networks. https://docs.cumulusnetworks.com/display/DOCS/Segment+Routing. Accessed on January 22th 2019.

[8] E. Brinkhuis. Firewall as a service: flexibele firewall op een nfv-platform. https://blog.surf.nl/firewall-as-a-service-flexibele-firewall-op-een-nfv-platform/. Accessed on January 9th 2019.

[9] F. Clad, X. Xu, C. Filsfils, D. Bernier, C. Li, B. Decraene, S. Ma, C. Yadlapalli, W. Hendrickx, and S. Salsano. Service Programming with Segment Routing. Internet-Draft draft-xuclad-spring-sr-service-programming-01.pdf, SPRING, October 2018.

[10] C. Filsfils, N. K. Nainar, C. Pignataro, J. C. Cardona, and P. Francois. The segment routing architecture. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, Dec 2015.

[11] C. Filsfils and S. Previdi. Segment Routing Architecture. RFC 8402, RFC Editor, July 2018.

[12] R. Geib and C. Pignataro. A Scalable and Topology-Aware MPLS Data-Plane Monitoring System. RFC 8403, RFC Editor, July 2018.

[13] The Tcpdump Group. Manpage of tcpdump. https://www.tcpdump.org/manpages/tcpdump.1.html. Accessed on January 23th 2019.

[14] Inc. Juniper Networks. Contrail networking - juniper networks. https://www.juniper.net/us/en/products-services/sdn/contrail/contrail-networking/. Accessed on January 9th 2019.

[15] S. Previdi, L. Ginsberg, C. Filsfils, A. Bashandy, H. Gredler, and B. Decraene. IS-IS Extensions for Segment Routing. Internet-Draft draft-ietf-isis-segment-routing-extensions-22, Internet Engineering Task Force, December 2018. Work in Progress.

[16] D. Singh. Yet another blog about segment routing-part 1. https://packetpushers.net/yet-another-blog-about-segment-routing-part-1/. Accessed on January 15th 2019.

[17] D. Tappan, Y. Rekhter, A. Conta, G. Fedorkow, E. C. Rosen, D. Farinacci, and T. Li. MPLS Label Stack Encoding. RFC 3032, January 2001.

[18] R. van der Gaag and S. Slotboom. Sr-mpls proof of concept. https://github.com/RonaldvdG/SR-MPLS-PoC, 2019.

APPENDIX

A. Detailed overview of the current situation

In the current situation, institutes who participate in the 'Firewall-as-a-service' (FaaS) Pilot of SURFnet, have to travel through a set of two GRE tunnels (red and yellow in Figure 9) for each of the two core routers. In order to do so, SURFnet has a virtual firewall which advertises (via BGP) a more specific route to the institute than the original advertisement (also via BGP) towards the customer. With BGP, the most specific advertisement will be preferred above the less specific advertisement. This makes that the network traffic to go through the virtual firewall. But in case of a network failure towards the firewall, the network traffic can still reach the institute via the original static route. There are two GRE tunnels per core router, per institute needed. One of the GRE tunnels will be used for network traffic from the internet towards the institute and the other GRE tunnel for the network traffic from the institute towards the internet. An overview of this setup can be seen in Figure 9.

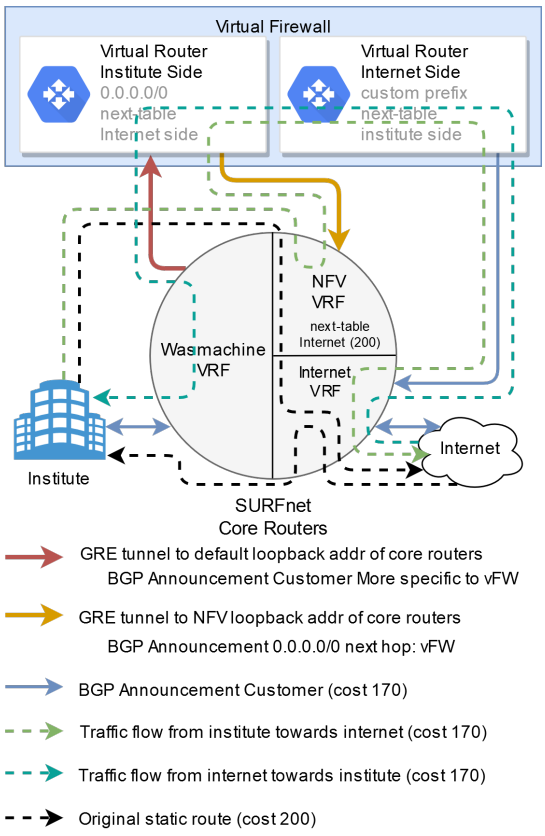


Fig. 9. Current tunneling solution with GRE tunnels

1) Network flow from internet towards institute: When network traffic is destined for the institute and originated from the internet, the traffic first arrive at the Internet Virtual Routing and Forwarding (VRF) interface of one of the core routers. Then the network traffic will go towards the unfiltered side of the virtual firewall, this is because of the more specific announcement. The traffic will be filtered and send to the filtered side of the virtual firewall. The next hop will make the network traffic go via a GRE tunnel (red arrow) towards the 'Wasmachine VRF'. The 'Wasmachine VRF' routes the network traffic further with the original announcements towards the institute and will thus deliver the network traffic for the institute.

2) Network flow from institute towards internet: When network traffic is destined for the internet and originated from the institute, the traffic will first arrive at the Network Function Virtualization (NFV) VRF. The virtual firewall advertises a default route (`0.0.0.0/0`) for the NFV VRF with a lower metric (170) than the original default route (with metric 200). This default route towards the virtual firewall will make the network traffic travel through a GRE tunnel (yellow arrow). Because of the lower metric, the network traffic will go via the virtual firewall as next hop. The virtual firewall will filter the traffic and send it towards the Internet VRF. The Internet VRF will send the network traffic further for it to reach its destination.

3) The GRE tunnels: As mentioned above, there are two GRE tunnels per core router. The first being between the virtual firewall (Institute Side) and the 'Wasmachine VRF'. The 'Wasmachine VRF' advertises the more specific route through this GRE tunnel. The second GRE tunnel is between the NFV VRF and the virtual firewall (Institute Side). The virtual firewall advertises a default route, with a lower metric (170) than the original default route (with metric 200), through this GRE tunnel.