Firewall-as-a-Service for Campus Networks Based on P4-SFC

Marco Häberle[1], **Benjamin Steinert**[2], **Michael Menth**[3]

[1]marco.haeberle@uni-tuebingen.de
[2]benjamin.steinert@uni-tuebingen.de
[3]menth@uni-tuebingen.de
University of Tuebingen, Chair of Communication Networks, Tuebingen, Germany *

Abstract: Taking care of security is a crucial task for every operator of a campus network. One of the most fundamental security-related network functions that can be found in most networks for this purpose are stateful firewalls. However, deploying firewalls in large campus networks, e.g., at a university, can be challenging. Hardware appliances that can cope with today's high data rates at the border of a campus network are not cost-effective enough for most deployments. Shifting the responsibility to run firewalls to single departments at a university is not feasible because the expertise to manage these devices is not available there. For this reason, we propose a cloud-like infrastructure based on service function chaining (SFC) and network function virtualization (NFV) that allows users to deploy network functions like firewalls at a central place while hiding most technical details from the users.

Keywords: Service Function Chaining, Software Defined Networking, Firewall

1 Introduction

In recent years, network function virtualization (NFV) gained traction among Internet service providers and operators of campus networks and data centers. NFV aims to reduce cost and improve flexibility by replacing hardware-based network functions with virtual equivalents, virtual network functions (VNF), that are run on commercial off-the-shelf servers. NFV is worked on actively by both academia and standardization bodies [YWL+18].

Network traffic is often steered through VNFs with the help of service function chaining (SFC). The SFC architecture standardized by the IETF in RFC 7665 consists of classifiers, service functions, and service function forwarders. Service functions are single network functions, e.g., firewall or NAT appliances. These service functions may be VNFs. The classifier is the entry point to an SFC-enabled domain and assigns every packet to a service function path consisting of service functions that the packet needs to traverse. This path is encoded into the packet, e.g., by using the network service header or by pushing an MPLS label stack. Similar to NFV, SFC is worked on by academia and standardization bodies [BJSE16].

In this work, we propose to use SFC and NFV to secure campus networks of universities in a flexible, yet simple to manage way. An easy-to-use self-service portal enables individual departments to define service function chains and configure service functions without the need

* This work was supported by the bwNET2020+ project which is funded by the Ministry of Science, Research and the Arts Baden-Württemberg (MWK). The authors alone are responsible for the content of this paper.

to operate expensive and hard to configure network appliances. Instead, service functions are deployed as NFVs in a local SFC cloud that is run in a central data center in the campus network. A prototype of the self-service portal and the sfc infrastructure is available at GitHub[1].

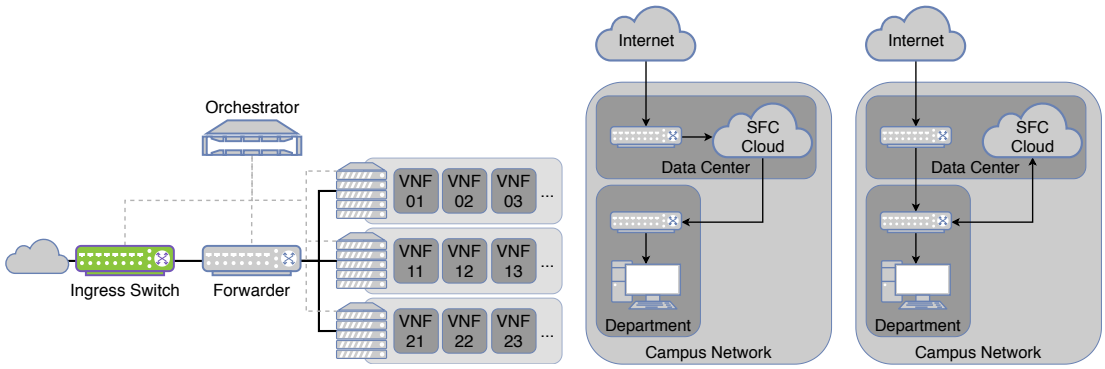
In the following section, we give an overview of the local SFC cloud. In Section 3, we describe the self-service portal. In Section 4, we conclude our work.

2 Local SFC Cloud

In the following, we give an overview of the setup of the local SFC cloud and explain how it can be integrated in an existing campus network with minimal changes.

2.1 SFC Cloud Infrastructure

The infrastructure is largely based on P4-SFC [SHHM20]. Service function chaining is realized using MPLS segment routing similar to [CXF+20]. The components are shown in Figure 1(a).



(a) SFC cloud components. (b) Integration: routing (c) Integration: tunnel

Figure 1: The setup of the SFC cloud and options to integrate it into a campus network.

Incoming traffic is classified by an *ingress switch* implemented in P4. Classification is done according to the service function chain configured in the self-service portal. The classified traffic is then sent to the VNF hosts by *service function forwarders*. These forwarders may be any traditional off-the-shelf switches with support for MPLS segment routing.

Several *VNF hosts* run the VNFs that have been configured in the self-service portal. The VNFs themselves are executed either in a dedicated LXC container or in a dedicated KVM virtual machine. The MPLS router module of the Linux kernel of the VMs or containers serves as an SFC proxy for SFC-unaware VNF applications.

A central *orchestrator* transforms the settings taken in the self-service portal into an appropriate network configuration and deploys the requested VNFs on the VNF hosts. Hereby, MPLS

[1] https://github.com/uni-tue-kn/p4-sfc-faas

labels are assigned to the VNFs and the label stack for each service function chain is configured on the ingress switch for classification.

2.2 Traffic Classification in P4

P4 provides the ability to execute a single data plane definition on different software and hardware platforms. So far, P4-SFC supports the Intel Tofino switching ASIC and the software-based bmv2 switch for traffic classification.

The Intel Tofino supports up to 64 100G Ethernet ports, its successor up to 32 400G Ethernet ports. On this platform, traffic classification happens in line speed. While this high performance is highly future-proof, a more cost-effective solution with less performance may be desirable for some use-cases.

The bmv2 is an open-source software switch that provides an easy way to run P4 programs. However, it is not production-ready and is intended for development of P4 programs only. In a virtual machine with 8 virtual CPUs and 15 GiB of RAM, it achieves a throughput of only around 1 Gb/s when forwarding packets between two hosts [bmv].

2.3 Integration into Campus Networks

The SFC Cloud is deployed in a data center inside the campus network. Depending on the structure of the campus network and the requirements of the departments that use the SFC cloud, several options exist to integrate it into existing infrastructure.

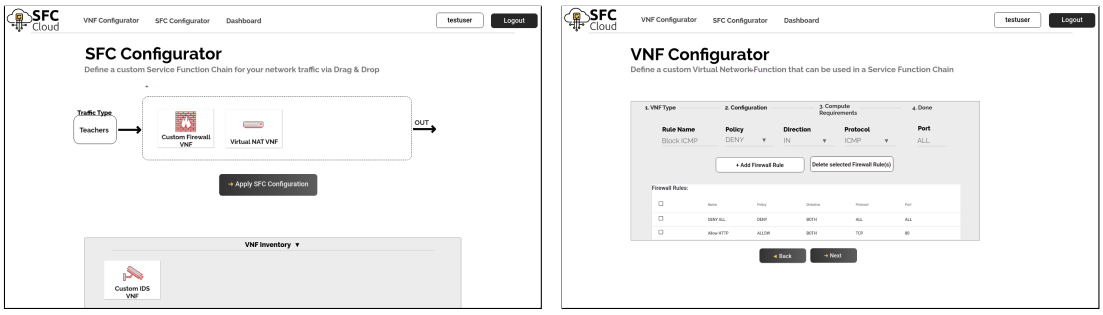
The most straight forward option is to modify the routing of all traffic going to departments that use the SFC cloud. This option is shown in Figure 1(b). While this results in minimal technical overhead, the administrative overhead may be cumbersome when deploying the system.

Alternatively, departments that want to use the SFC cloud may use the approach shown in Figure 1(c). The traffic that enters and leaves the network of the department is tunneled to the SFC cloud. This approach needs modifications of the network configuration at the department only without modifying any other parts of the campus network. However, even lightweight tunneling protocols like GRE have a protocol overhead that should not be ignored. As a consequence, this approach is suitable for departments with a low traffic volume only.

3 Self-Service Portal

The self-service portal is the central point to configure service function chains and VNFs. It features a user and permission model that allows to delegate responsibility for specific traffic classes, e.g., traffic to and from an individual department. Defining traffic classes and template VNFs, as well as managing user rights is done by administrators of the SFC cloud system.

Users with delegated rights for specific traffic classes are able to define service function chains easily. A sample chain configuration is shown in Figure 2(a). When configuring a service function chain, users can simply drag-and-drop VNFs from a VNF inventory. The VNF inventory contains VNFs that are predefined by administrators, as well as VNFs that have been configured by the user himself. For this purpose, the self-service portal provides configuration wizards for selected VNF types. An example, configuring a firewall VNF, is shown in Figure 2(b).



(a) SFC configuration with drag and drop.

(b) Configuration of Firewall VNF.

Figure 2: Self-Service Portal.

4 Conclusion

Securing campus networks, e.g. at universities, becomes more and more difficult, complex, and expensive with increasing network traffic. Service function chaining in combination with network function virtualization can help in solving these issues. In this work, we presented a cloud-like infrastructure that enables users in a campus network, e.g. single departments at a university, to deploy virtual network functions like firewalls at a central point in the network. A self-service portal enables users to easily configure these functions.

References

[BJSE16] D. Bhamare, R. Jain, M. Samaka, A. Erbad. A survey on service function chaining. *Journal of Network and Computer Applications* 75:138 – 155, 2016.

[bmv] Performance of bmv2. https://github.com/p4lang/behavioral-model/blob/master/docs/performance.md. accessed 26-10-2020.

[CXF+20] F. Clad, X. Xu, C. Filsfils, D. Bernier, C. Li, B. Decraene, S. Ma, C. Yadlapalli, W. Henderickx, S. Salsano. Service Programming with Segment Routing. Internet-draft draft-ietf-spring-sr-service-programming-03, Internet Engineering Task Force, Sept. 2020. Work in Progress.

[SHHM20] A. Stockmayer, S. Hinselmann, M. Häberle, M. Menth. Service Function Chaining Based on Segment Routing Using P4 and SR-IOV (P4-SFC). In *ISC High Performance International Workshops*. Pp. 297–309. 2020.

[YWL+18] B. Yi, X. Wang, K. Li, S. k. Das, M. Huang. A comprehensive survey of Network Function Virtualization. *Computer Networks* 133:212 – 262, 2018.