Service Chaining for Providing Network Security as a Service for Remote Enterprise Users

Jong-Geun Park, Jung-Tae Kim and Jong-Hyun Kim
Intelligent Security Research Group
Electronics & Telecommunications Research Institute
Daejeon, Republic of Korea, 34129
Email: {queue; jungtae_kim; jhk}@etri.re.kr

***Abstract*—With the introduction of Software-Defined Networking and Network Functions Virtualization and the rapid spread of cloud computing technology, network security as a service is attracting attention increasingly. It can provide customized network security service according to the needs of users. It can reduce the capital and operating expenditures of the enterprise and also avoid vendor lock-in problem. To provide cloud-based on-demand network security services, end-to-end service chaining should be considered so that user traffic is delivered to the Internet via predefined virtual network security functions without any modification or manipulation of the user's traffic, and vice versa. In this paper, we present an integrated service chaining mechanism to provide network security as a service for remote enterprise customers.**

Keywords–Service Chaining; Network Security as a Service; SECaaS

I. INTRODUCTION

Software-Defined Networking (SDN) and Network Functions Virtualization (NFV) have evolved with driving the next generation network transformation. SDN decouples the logically centralized controller plane from the data plane, and enables programmable and abstract network infrastructure [1]. NFV aims to implement various network functions in software that can run on a range of industry standard server hardware, and that can be moved to, or instantiated in, various locations in the network as required, without the need for installation of new equipment [2]. Service chaining is attracting attention as a key technology in SDN/NFV that can steer a dynamic traffic route. A Service Function Chain (SFC) defines an ordered set of abstract service functions and ordering constraints that must be applied to packets and/or frames and/or flows selected as a result of classification. The term "service chain" is often used as shorthand for service function chain [3].

These emerging technologies allow service providers not only to deploy various virtualized network security functions, but also to offer them to their customers as on-demand network security as a service. It can reduce the capital expenditures (CapEx) and operating expenditures (OpEx) of the enterprise and also avoid vendor lock-in problem. But, in order to support the service, end-to-end service chaining should be provided so that user traffic is delivered to the Internet via predefined virtual network security functions without any modification or manipulation of the user's traffic, and vice versa.

The remainder of this paper is organized as follows. In Section 2, we describe service chaining mechanism where endpoints are not in the cloud. In Section 3, a network control architecture for the service chaining is briefly presented. Finally, in Section 4, we summarize our work.

II. SERVICE CHAINING FOR NETWORK SECURITY AS A SERVICE

In this section, we present how to provide network security as a service where two endpoints are not in the cloud. If the user terminal or Internet service nodes are in the cloud, service chaining for the network security service can easily be configured by the cloud network controller. For example, the networking-sfc project [4] that is a subproject of Openstack Neutron provides service chaining with port-chaining technology in an OpenStack [5] environment. A user terminal or a service node created as a virtual instance in the OpenStack can get a network security service through ordered virtual network security functions applied by port-chaining technology.

However, if both endpoints are not in the cloud, more complex control is required. Traffic between them does never go through virtual security functions located in the cloud unless service chaining control is applied. As mentioned earlier, only service chaining through the cloud network controller can not provide network security as a service. Therefore, there is a need for an integrated service chaining control architecture that controls all the different network domains between the two terminals.

We show it in more detail in Figure 1. If the service chaining is not set at all, the traffic from the user terminal to the Internet service is delivered directly without passing through the cloud data center as indicated by the red dotted line in Figure 1. However, when a user's traffic needs to be served cloud based network security service, it is required to be delivered to the cloud data center in the following manner.

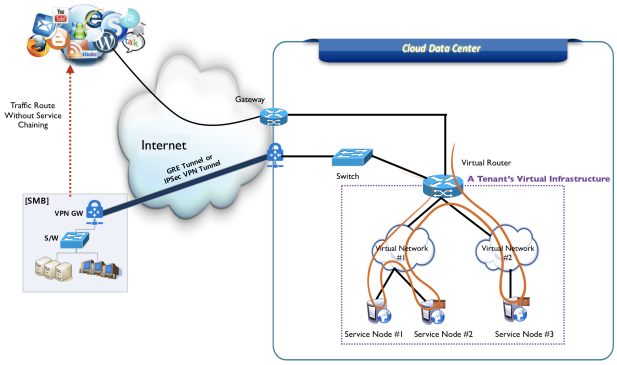

Figure 1. Service chaining for providing network security as a service for remote enterprises

 ISBN: 978-1-61208-674-3

A. Between Enterprises and the Cloud Data Center

At first, in order to forward traffic to the cloud data center, some or all of customer's traffic is delivered to the network device providing the tunneling method, such as Generic Routing Encapsulation (GRE) router or Virtual Private Network (VPN) gateway. Then, traffic can be delivered through a GRE or VPN tunnel without manipulating the original packet header.

In case that if the security-as-a-service provider is also Internet service provider, or if enterprises are geographically close to the cloud data center, the traffic can be delivered on a Layer-2 basis instead of a tunnel.

B. Within the Cloud Data Center

Users' traffic arriving at the cloud data center through the tunnel should be routed to the predefined virtual network security functions in an ordered sequence. This can be enabled by service chaining which steers packet forwarding path from when packets are arrived until they exit from the cloud.

In order to control service chaining, first of all, a tenant information need to be identified. It can be usually classified by the source address or subnet of the traffic. Then, the traffic is transmitted to the virtual router which is a gateway connected to tenant's virtual infrastructure. It can be forwarded on a physical network switch with Policy Based Routing (PBR) that is a technique used to make routing decisions based on administrator's policies. For example, a packet is forwarded based on the source address, not the destination address in it.

To the next step, the traffic arriving at the virtual router should be transmitted to virtual network security functions. However, it is directly routed to the Internet via the cloud gateway without network security services, since the virtual router decides the next hop depends on the destination address. Eventually, an additional chaining policy is required to route traffic from virtual router to virtual network security functions.

In our implementation, we adopt PBR in the virtual router. For example, we may setup a PBR rule on a virtual router of an OpenStack cloud as in Figure 2. If the namespace of the virtual router is *qrouter-1234*, at first, a routing rule for traffic with the 172.16.1.0/24 subnet as the source address is added to the PBR-A table. Then, as a new routing rule in the PBR-A table, forward it as the default routing rule to the qr-abc interface (20.1.0.1) that is an interface of the virtual router, via 10.21.0.5 (the IP address of the first virtual network security function or a service function classifier).

```
#> ip netns exec qrouter-1234 ip rule add from 172.16.1.0/24 table PBR-A
#> ip netns exec qrouter-1234 ip route add default via 20.1.0.5 dev qr-abc table PBR-A
#> ip netns exec qrouter-1234 ip route flush cache
```

Figure 2. An example of PBR setup on a virtual router of Neutron

Finally, the traffic is forwarded along the local service chaining path (the ordered sequence of virtual security functions) using the port-chaining mechanism of Neutron networking-sfc project, and then transmitted to the Internet service through the Internet gateway of the cloud data center.

III. Architecture of Integrated Service Chaining Controller

The service chaining controller can be a part of the network orchestrator. For example, Neutron is a OpenStack project which is responsible for networking services and networking-sfc project [4] provides APIs and implementations to support service function chaining in Neutron. Therefore, it is only responsible for the virtual infrastructure in the cloud with the networking-sfc service plugin and the virtual switch control agent.

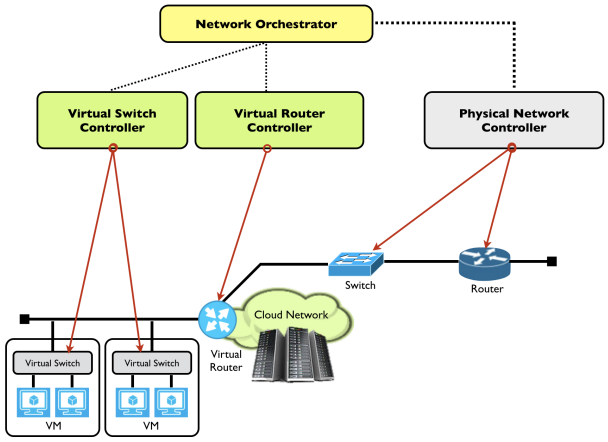

Figure 3. Network control architecture for E2E service chaining

However, as mentioned above, the integrated service chaining control for traffic passing through the cloud requires control of service chaining for not only the virtual infrastructure, but also the physical network devices and the virtual router. Therefore, as shown in Figure 3, the integrated service chaining controller includes virtual switch controller as well as virtual router controller and physical network controller.

IV. Conclusion

When we provide network security as a service for remote enterprise customers, the service chaining should be controlled to pass through the cloud without any manipulation of the original user's traffic. In this work, we have additionally applied the policy based routing technology to virtual router and physical network equipment. It can support seamless service chaining for providing network security as a service for remote enterprise users.

Acknowledgment

This work was supported by Institute for Information and Communications Technology Promotion (IITP) grant funded by Korea government (MSIT). (No.2016-0-00078, Cloud based Security Intelligence Technology Development for the Customized Security Service Provisioning)

References

[1] ONF, "SDN architecture," Open Networking Foundation, Tech. Rep., June 2014.

[2] NFV White Paper, "Network Functions Virtualisation: An Introduction, Benefits, Enablers, Challenges & Call for Action. Issue 1," Oct. 2012.

[3] J. Halpern and C. Pignataro, "Service function chaining (sfc) architecture," Internet Requests for Comments, RFC 7665, Oct. 2015.

[4] "OpenStack Networking-sfc Project," URL: https://wiki.openstack.org/wiki/Neutron/ServiceInsertionAndChaining [retrieved: Aug. 2018].

[5] "OpenStack," URL: http://www.openstack.org/ [retrieved: Aug. 2018].

 ISBN: 978-1-61208-674-3