Advanced Science and Technology Letters
Vol.51 (CES-CUBE 2014), pp.102-105
http://dx.doi.org/10.14257/astl.2014.51.24

Study on Search Algorithm for Finding an Efficient Service-Chaining Path

Hyeonseok Oh[1], Daeun Yu[1], Yoon-Ho Choi[2], Namgi Kim[1]

[1] Department of Computer Science, Kyonggi University, Iui-dong, Yeongtong-gu, Suwon-si, Gyeonggi-do, Korea
{lims, deyoo, ngkim}@kyonggi.ac.kr

[2] Department of Convergence Security, Kyonggi University, Iui-dong, Yeongtong-gu, Suwon-si, Gyeonggi-do, Korea
ychoi@kyonggi.ac.kr

Abstract. Much attention has been paid to network virtualization since it can better utilize network resources and provide easier management than conventional networks. Virtual machines created by virtualization provide services, and a network flow passes through a set of virtual machines that provides the required services. Service chaining is to ensure provisions of the required services. Due to the service-chaining concept, studies on which virtual machines are selected and how to connect them in an efficient manner have been highlighted ever more. This paper proposes a new algorithm to search for an efficient service-chaining path.

Keywords: Virtual Network, Software-Defined Network, Network Function Virtualization, Service Chaining, Finding Compatible Virtual Machine

1 Introduction

With advancements in network technologies, services and service network users are increasing. However, it is difficult to expand networks or create a new network to meet demands quickly. To relieve this problem, the concept of host virtualization, in which one physical host has multiple logical hosts, has been proposed. The concept of virtualization has been extended to network areas so that the physical equipment of network components, such as switches and links, can be utilized as logical equipment. Network virtualization facilitates convenient network management as well as the maximum utilization of idle resources. In addition, it provides divided logical sets of equipment, thereby easily providing an independent network environment to each user according to their requirements.

However, optimum path searches for network flows and traffic management have been more difficult than before. To solve this problem, the concept of a Software-Defined Network (SDN) has been introduced. In an SDN, it is possible to achieve a specific flow to pass through a desired path. Thus, service-chaining technology, by which a flow passes through specific, required hosts, has been proposed. In addition to the conventional concept of the optimal path, which simply indicates a path from source and destination points, service chaining can add nodes that must be passed

ISSN: 2287-1233 ASTL

through. This paper proposes an algorithm that searches the optimal service chain path in consideration of the services that each virtual machine provides as well as network performance to solve the above problem.

2 Related studies

A method proposed in this paper is applied to logical network environments, which is called network virtualization. Physical equipment in a real network becomes logical equipment to create a virtual network [1]. Such virtual nodes and links over the virtualized network are mapped to physical devices in the network, a process called Virtual Network Embedding (VNE). During the VNE creation process, it is important to meet the constraints of the host process capability over the virtual network while creating no excessive loads on the network. This is called an NP-Hard problem. To solve the NP-hard problem, there are two approaches: heuristic programming and integer programming. Both approaches aim to reduce network load and produce optimum performance.

To determine a path according to the network flow characteristics, a technology, which identifies network circumstances and orders, commands the switches according to the circumstances, as necessary. The SDN used for this purpose, in contrast with traditional switches that serve packet transfers, is divided into two planes: a data plane that is responsible for data transmission and a control plane that indicates transmission direction. A controller collects information regarding overall network topology to manage the control plane in a centralized manner. Using the shortest path-finding algorithm, a flow is produced over the shortest path in consideration of the capacity of each link, thereby preventing network performance degradation.

Since most traditional networks consist of proprietary hardware-based devices, it is difficult to remove existing devices or add new functions. To solve this problem, Network Function Virtualization (NFV) has been proposed [2]. NFV consists of high-performance network components and functions required for services, which are implemented by software that adds them to each component accordingly. As a result, a network structure that can respond to changes flexibly and does not depend on specific proprietary equipment vendors can be made.

Service chaining, a process where devices providing only required services are passed, according to flow requirements, using SDN technology, has been proposed[3]. This technology is proposed to improve security performance via service chains that correspond to an attack type of a specific flow, once flows have been classified according to attack types in the security sector.

The concept of service chaining was previously studied in web-related sectors. For example, a study on web service composition that provides new services by combining existing web services was conducted. Web service composition determines whether two web services are required to be linked using the definition of a user's requirements as well as the input, the output, the previous state, and the modified state of the results, as received after each service is provided. To represent the correlation between web services, a graph or a tree is employed. A study on such correlations is

underway to search for services to be linked using search algorithms that are compatible for each data structure used in the representation of the correlation.

3 Proposed Algorithm

In this section, we explain an algorithm for finding a service-chain path.

3.1 Network structure

In this paper, the environment only considers a network structure consisting of high-performance devices required in NFV and an SDN-applied network by adding controllers. In each server, an internal network is composed of a number of virtual machines, and a virtual machine serves one of the functions of the network components, such as firewalls or detection systems. All switches in the network are connected to a controller, while a controller receives traffic flow information from switches to identify flow characteristics. The characteristics of every virtual machine are also provided to a controller according to a specific event or a certain interval, and this information is employed later in the proposed algorithm. Once the traffic flow is identified, a virtual network path, reconfigured logically according to the characteristics, can be produced. A network manager can input the virtual network topology, which is optimized for a specific flow inside the controller.

3.2 Detailed explanation of the algorithm

The algorithm proposed in this paper is divided into two phases. The first phase is to search virtual machines compatible with the required service, and the second phase is to determine the optimal path for the virtual machine found in the previous phase. A vector-space model is used to search for virtual machines compatible with the required service by employing various pieces of information about virtual machines, while the shortest distance-search algorithm is used to find the optimal path between virtual machines.

The virtual network topology collected in a controller is used to find compatible virtual machines needed to meet the user requirements or flow characteristics. Such information can be collected by requesting it from every host at the time of system initialization. The information used includes the number of currently connected sessions, the processing capability according to packet protocols, the average size of processed packets, and the service number provided by a virtual machine. Each element is converted into an integer format and then represented by a 4D vector. The values of each element can be updated through periodic communication with devices as well as through controller operations. A single 4D vector produced using the above operation can be arranged in a 3D vector space using all the used information, other than the service number provided by the virtual machines.

The first phase of the algorithm proposed in this paper is employed based on the Euclidian distance, along with the use of the above-mentioned information. The virtual machine with the shortest distance among every virtual machine vector and

requested vector in the 3D vector model is selected as the optimal virtual machine. Upon the initial information request, as mentioned earlier, distances between virtual machines, with respect to the requested vector, are sorted out in an ascending order. The smallest value is returned when the same request is received, unless the values in the network topology are changed. Through the above process, the most compatible virtual machines for the requested vector can be obtained.

In the next phase, the optimal path for the virtual machines is calculated using a set of virtual machines obtained through the previous process. First, the shortest distance between the selected machines is searched. Generally, shortest path first (SPF) algorithms can be used to search for the shortest distance. A set of virtual machines in each step of the service chain is searched, and the shortest path between virtual machines is found, thereby creating a final path connecting the shortest path. When the first and second virtual machines are searched, many virtual machine paths shall be considered. In this case, a virtual machine, with the shortest distance value among the results obtained using the shortest path algorithm, is selected.

4 Conclusion

This paper proposed a service-chaining method based on finding compatible virtual machines according to user requirements in NFV- and SDN-based network environments in which a single controller manages a number of virtual machines. To select compatible virtual machines, required elements are proposed, and these elements are converted into an N-dimensional space vector where virtual machines are arranged to determine similarities based on distances to the requested vector, thereby acquiring a virtual machine set that meets the requirements. To create an efficient service-chaining path, the proposed method searched virtual machines in each step of the chain that is set according to the flow characteristics. The shortest distance between virtual machines was based on SPF algorithm.

A future study is scheduled to select optimal virtual machines using more elements that can reflect network administrator or user requirements better than the current options, as well as develop algorithms that can construct an efficient service-chain path.

Acknowledgement. This paper is supported by the Industrial Core Technology Development Program (Next-Generation Communication Network) in the Ministry of Science, ICT & Future Planning [10047541, Development of the auto-defense and auto-scalable SDN smart security networking system using NFV-based service chaining].

References

1. VMware, “VMware Virtual Networz`1king Concepts,” White Paper, 2007
2. ETSI, “Network Function Virtualization (NFV); Prof of Concepts, Framework,” 2013
3. Cisco, “Enabling Service Chaining on Cisco Nexus 1000V Series,” White Paper, 2013

