Study on Segment Routing in Software Defined Networking

May Thu Zar Win, Aung Htein Maw, Khin Than Mya
University of Computer Studies, Yangon.
maythuzarwin@ucsy.edu.mm, ahmaw@ucsy.edu.mm, khinthanmya@ucsy.edu.mm

Abstract

Segment Routing (SR) is an emerging technology to enforce the effective routing strategies without relying on signaling protocols. SR only needs to define the path information through the network at the ingress device and this mechanism make Software Defined Networking (SDN) routing management more simple and efficient. In this paper, we intend to involve as a small part of SDN-SR evolution by studying the architecture of SR and use-cases of SR in SDN environment. Moreover, we highlight the advantages of SR by comparing SR and other route computing methods applied in some use cases.
Keywords *- Segment routing, Software Defined Networking, use cases.*

1. Introduction

SDN is an emerging architecture that is flexible and responsive to rapid changes. In general, SDN most commonly means that networks are controlled by software applications and SDN controllers rather than the traditional network management consoles and commands that required a lot of administrative overhead and could be tedious to manage on a large scale. An operator can manage the physical functioning of the data plane through software installed on the controller [1]. That data plane comprises a network of OpenFlow enabled switches. One popular SDN protocol that allows such methods. Finally, section 5 gives conclusion. communication between the controller and the network switches is OpenFlow and there are many controllers such as ONOS, OpenDaylight, Floodlight, POX, NOX, etc.

Several developments and advances have been made to introduce new functions and applications that can give more maturity and specialized operations to the system. Some developments include the adaptation of layer 3 (L3) protocols, such as IGP (Interior Gateway Protocol) and BGP (Border Gateway Protocol) to support routing methods into the SDN environment [2]. SR is one of the examples of L3 path establishment method in SDN.

SR is also a source routing based tunneling techniques that allow a host or an edge router to steer a packet through the network by using a lists of segments. Today, network operators expect the offering of an increasing variety of cloud based services with stringent service level agreement (SLA). And they also anticipate SR will be filling the gaps of IP networks by providing scalability and flexibility properties. In this paper, we study the SR architecture and describe some of the use cases of SR and then we point out the advantages of SR in SDN by comparing other route computing methods.

The remainder of this paper is structured as follows. Section 2 describes the architecture of SR and the example scenario of SR is followed by section 3. Section 4 illustrates some of the use cases of SR. Section 5 represents compare and contrast with SR and other route computing.

2. Segment Routing Architecture

Segment Routing allows the network operator to specify a path from ingress to egress using a forwarding path that is completely abstract from the Interior Gateway Protocol (IGP) shortest path [3]. In SR, a node steers a packet through a controlled set of instructions, called segments by prepending the packet with an SR header. A segment can represent any instruction, topological or service-based and it can have a local semantic to an SR node or global within an SR domain [4].

In the SR domain, nodes and links are assigned Segment Identifiers (SIDs), which are advertised into the domain by each SR router using extensions to Intermediate System-Intermediate System/Open Shortest Path First (IS-IS/OSPF). The main types of SID are:

- Node SID: A node SID allocate to a loopback that identifies a unique SID for each switch/router in the network. The forwarding semantic associated with Node SID is to forward the packet on the shortest path towards the node associated with that Segment ID [5].

▪ Adjacency SID: The forwarding semantic associated with that SID is to forward the packet over the corresponding adjacency. Each router will assign a locally significant segment ID for each of its IGP adjacencies.

▪ Service SID: A service segment refers to a service offered by a node (e.g. firewall, vpn, etc.) and it used to deliver the packet corresponding to the service provided by the node processing the packet.

These SIDs allow an ingress node to select a path through the network using either a single SID to represent the destination node or using a series of SIDs, called a segment list, which specifies a particular path through the network that an SR tunnel should traverse [1]. An SR- enabled node supports the following data-plane operations:

▪ CONTINUE: Forwarding action performed based on active segment.

▪ PUSH: Add a segment ahead of the SR header of the packet and set that segment as the active segment.

▪. NEXT: The active segment is completed; the pointer is moved to the next segment in the list.

At the entrance of the SR domain, the ingress SR edge router pushes the SR header on top of the packet. At the exit of the SR domain, the egress SR edge router removes the SR header [4]. A property of the architecture is that the entire source route of the packet, including the identity of the ingress and egress edge routers is always available with the packet. This allows for interesting accounting and service applications.

3. Example Scenarios of SR

In SR architecture, when a new traffic flow has been established, a request is sent to the con controller that running on SR application and the controller performs the path computation using the proper segment lists [6].The computed segment list is sent back to the requested node and it sent the packets through the specific traffic flow of the segment lists to the destination.

Figure 1 illustrates the example network for SR and the two numbers described on each link represent the metric cost and the available bandwidth unit, respectively. As an example, assume the target path of the incoming traffic flow is p1= {A, B, C, F}. Therefore, the segment list may be SL1= {F}, only require the SID (Segment ID) of destination node. If the target path is not the shortest path to the destination, a more complex segment list is required, for example, if the desired path is p_2= {A, B, C, F, H}, the segment list may be SL_2={C, H}.

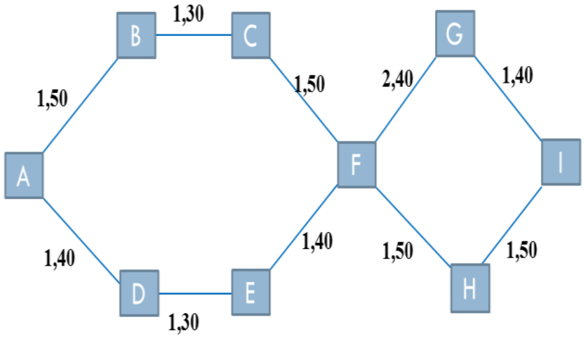

Figure 1. Example scenario for SR

Table1. Available paths and segment lists

Source – Destination	Available Paths	Segment Lists
A-F	{A,B,C,F} {A,D,E,F}	{C,F} {D,F}
A-I	{A,B,C,F,G,I} {A,B,C,F,H,I} {A,D,E,F,G,I} {A,D,E,F,H,I}	{B,G,I} {C,H.I} {D,G,I} {D,H,I}

Table 1 depicts the ECMPs (Equal Cost Multi Path) for the source to destination nodes (A to F and A to I) and segment lists for those paths. Moreover, SR computes shortest path, minimum hops count path and it also support load balancing among ECMP paths. To classify among ECMPs a more detailed segment list is required. SR can also compute paths based on the network requirement such as, bandwidth, latency and costs. If the traffic flow f_1 require 40 units of bandwidth for path p_1={C, F, H, I} and f_2 require 30 units of bandwidth from node E to I. In this case, f_2 can't be established due to the lack of bandwidth. So, SR re-route f_1 to the path p_2={C, F, G, I} and p_3= {E, F, H, I} for f_2. The controller that running on SR application can simply re-route f_1 from p_1 to p_2 by sending a new segment list SL_2= {G, I} to node C. So, SR can be easily rerouted by maintaining the route information in packet header as an order list of labels at the ingress node without degrading the performance of the network.

4. Use-Cases of SR in SDN

In this section, we described some of the use cases of SR in SDN environment.

4.1 Traffic Engineering Using SR

Traffic Engineering (TE) algorithms aims at to determine the packet routing paths in order to satisfy specific QoS requirements. SR can give control over TE paths without increasing control plane overhead at the transit nodes. These are some references papers that proved the SDN based SR-TE work well. Luca Davoli et al. [7] proposed SR-based TE in SDN paradigm and also implemented a simple heuristic approach for flow allocation by implementing with Ryu controller and Mininet and they also showed the better performance results for that work. Francesco et al. [8] proposed an efficient segment list encoding algorithms that accounts ECMP and also guarantee optimal path computation by reducing Segment Lists Depth (SLD) in SR based networks. They measured their performance in terms of SLD and described the better performance is directly proportional to the minimum number of labels included in the segment lists.

4.2 Application Aware Engineering (A2E) Using SR

Application aware engineering means the computation of path or route for each application data based on the application preferences. Different classes of applications need different paths characteristics. For instance: FTP (File Transfer Protocol) traffic will be forwarded through the path having higher bandwidth but SSH (Secure Socket Share) traffic which performs better in low latency so, it will be forwarded through the path having low latency. For the voice traffic, it forwards the low delay path because voice traffic is one of the delay sensitive traffic.

As an example scenario, figure 2 illustrates six hosts and six switches leaf and spine topology that show how application aware engineering work in SDN based SR. Suppose S_1, S_2, S_3, S_4, S_5 and S_6 switches have the node SID of 101, 102, 103, 104, 105 and 106 respectively. And also suppose that the yellow lines between switches mean the high bandwidth and low delay path. The green lines mean the high bandwidth and high delay path. The blue lines mean the low bandwidth and low delay path.

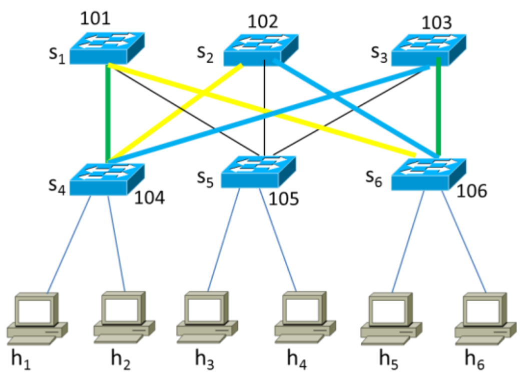

Figure 2. Example scenario for application aware engineering using SR.

The available SID paths between host h1 and h6 are 104-101-106, 104-102-106 and 104-103-106 respectively. For the voice traffic, S_1 would place the segment list {102,106} in the SR header and forward to S_2. For the case of large bandwidth application traffic, S_4 would push a SR header with segment list {101,106} and forward it to S_1. S_1 then forward to destination node.

4.3 Service Function Chaining (SFC) Using SR

Service provider networks offer a variety of services and implement them using a variety of appliances, independent from their routers. Due to the nature of these services, appliances must frequently be updated, migrated, or replaced. The burden of maintaining multiple types of systems together with the tight integration of appliances with network devices does not provide the flexibility needed to support such dynamic behaviors. Operators are hence suffering from large management overhead, which impacts their operational expenditures [5]. Network Function Virtualization (NFV), is gaining popularity and promises to solve the above problem and become a key player in future networks. But operators still lack a way to apply these functions into their network in a flexible and maintainable manner.

For this purpose, SFC was proposed and it is a type of traffic steering through the service function path, which is an ordered list of service functions in the packet header. Since SR already provides a similar framework, it becomes a suitable candidate to implement SFC on existing MPLS and IPv6 data-plane [5].

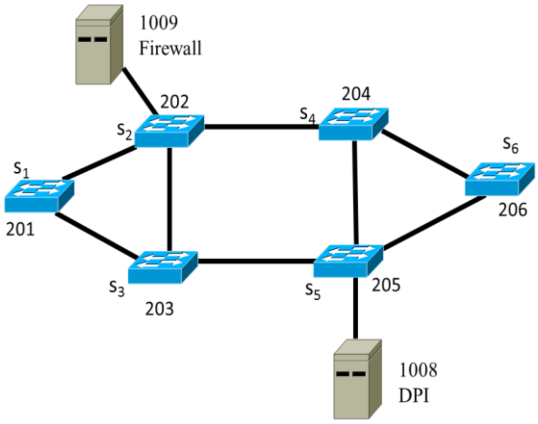

Figure 3. Example scenario for SFC

In our sample topology, the node SID and service segments are assign as follows: switches S1, S2, S3, S4, S5 and S6 assign the node SID 201, 202, 203, 204, 205 and 206 respectively. In addition, service SID 1009 will assign for firewall service and 1008 for Deep Packet Inspection (DPI) service. For example, the operator manages to apply a set of services, in a fixed order, for any traffic between S1 and S6 that must apply Firewall and DPI service.

Service segments have local significance to each device. Therefore, a service segment is typically combined with a node segment that ensures delivery to the service node. To apply the service chain, S_1 will push a SR header with segment list {202, 1009, 205, 1008, 206} and forward the packet. S_2 will receive the packet and use the next segment, 1009, to identify the service function for packet processing. After S_2 receives the packet back from the Firewall service, it will forward the packet to S_5, based on the top SID 205. S_5 will forward the packet to the DPI service and then forward the packet towards its final destination.

5. Comparison of Use cases for SR and other routing scheme

In this section, we delineate SR or other routing scheme that performed in three use cases mention in section 4. In table 2, the first three approaches performed SR based TE and the last one implement TE with Hedera flow scheduling. Although all approaches described in table 2 are able to perform traffic engineering effectively, SR-based approaches reduce the complexity of maintaining flow rules because SR only need to install the segment lists in ingress node. Moreover, SR can also be applied in multi-domain network.

Table2. Use case I - TE

Approach	Routing Scheme	Addressed solution
Label encoding [8]	Used SR with modified label encoding algorithm.	Minimum SLD based optimal path computation.
Efficient routing algorithm based SR [9].	Used SR with proposed bandwidth oriented heuristic algorithm.	Reduce network congestion. Build a bandwidth satisfied path between SR-enabled nodes.
Optimal path computation architecture for cloud network on SDN [10].	Use Route Computation element (RCE) that is extended to SR-PCE (Path Computation Element).	Real-time data transmission in multi-domain network.
Dynamic flow scheduling [11]	ECMP base Hedera flow scheduling.	Perform TE by detecting large flow in edge switches. Bandwidth overhead can occur at the switches.

According to table 3 and 4, SR show impressive outcome in these cases. SR can enforce a flow through any topological path and service chain while maintaining per-flow state only at the ingress node to the SR domain. That is very useful in TE and A2E because we can steer traffic by adding appropriate segment lists to ingress node. Thus, there is no need to consider the communications delay between controller and network devices.

Table3. Use case II – A2E

Approach	Routing Scheme	Addressed solution
Application aware path selection [12]	Used DPI to classify traffic. Based on buffer playtime.	Only applying to YouTube traffic.
Application aware routing [13]	Used LARAC Algorithm and considered delay variation.	Classify different classes of applications to different paths
A2E SR [3]	SR with Path Computation Element	Differentiating flows per application.

Table4. Use case III - SFC

Approach	Routing Scheme	Addressed solution
SFC placement in Data Center [14]	Used Multi-layer worst-fit and Multi-layer best fit	Solve multilayer bin packing problem.
IPV6 SR for SFC [15]	IPV6 SR. Use DPKG tools.	Designed chaining different services.

5. Conclusion

Segment routing is built for the SDN era. The combination of a SDN controller with SR enabled infrastructure is extremely powerful. In this paper, we discuss the overall architecture of SR and give some of the use cases of SR in SDN environment. Today's networks are forced to be application-aware, to improve user experience, provide service differentiation, and reduce operational costs. Therefore, for the future work, we will implement the application aware segment routing scheme in SDN by using ONOS controller and mininet emulator. The bandwidth and delay are defined as a QOS parameter of this application engineering (AE) task. To reduce the computational complexity of this complex AE task, applications traffics may be classified into two categories: non delay-sensitive (NDS) applications and delay-sensitive applications (DS). The propose AE will be consisted of three main tasks: (1) calculates end-to-end bandwidths and delays for each nodes of the links; (2) compute the best feasible path by using modified Maximum Delay-Weighted Capacity Routing Algorithm (MDWCRA) based on computed bandwidth and delay; and (3) extract SR's node segment ID from the computed path and then add SR tunnels and policies for each applications. Our AE approach will be improved the overall performance of the network by steering the best optimal routes for each application.

References

[1] Cisco, "Software Defined Networking for Dummies", [online] Available, https://www.cisco.com/c/dam/en/us/solutions/collateral/datacentervirtualization/applicationcentricinfrastructure/sdnfordummies.pdf.

[2] D.J.Q.Martina, C.Cervello Paster, "Influence of the path establishment method on a SDN controller time response", InActas XII Jornadas de Ingeniería Telemática, 2015, pp. 116-123.

[3] Alcatel Lucent, "Segment Routing and Path Computation Element," Technology White Paper, [online] Available, http://www.resources.alctaellucent.com/SegmentRoutingandPathComputationElement.pdf.

[4] C. Filsfils, S. Previdi (Eds.) et al, "IPV6 Segment Routing Header", IETF draft-ietf-spring-segment-routing-architecture, June 2016.

[5] C.Filsfils, N.K. Nainar, C.Pignataro, J.C. Cardona and P.Francois, "The Segment Routing Architecture", IEEE Global Communications Conference (GLOBE COM), 2015 Dec 6, pp. 1-6.

[6] A.Sgambelluri, F.Paolucci, A.Giorgetti, F. Cugini and P.Castoldi, "Experimental demonstration of segment routing", *Journal of Light wave Technology*, 2016 Jan, pp. 205-212.

[7] L.Davoli, L.Veltri, P.L.Ventre, G.Siracusano and S.Salsano, "Traffic Engineering with Segment Routing: SDN based Architectural Design and Open Source Implementation", *2015 Fourth European Workshop on*

Software Defined Networks, IEEE, 2015 December.

[8] F.Lazzeri, G.Bruno , J.Nijhof, A.Giorgetti , P.Castoldi, “Efficient label encoding in segment-routing enabled optical networks”, *IEEE Optical Network Design and Modeling (ONDM)*, International Conference, 2015 May 11, pp. 34-38.

[9] Min Chieh Lee, Jang Ping Sheu, “An efficient routing algorithm based on segment routing in software defined networking,” *Computer Networks*, 2016 July, pp 44-55.

[10] H.Cho, J.Park, J.M.Gil, Y.S.Jeong, J.H.Park “An Optimal Path Computation Architecture for the Cloud-Network on Software-Defined Networking”, Sustainability, 2015 May , pp. 5413-5430.

[11] M. Al-Fares, S. Radhakrishnan, B. Raghavan, N. Huang, A. Vahdat, “Hedera: dynamic flow scheduling for data center networks,” Proceedings of Networked Systems Design and Implementation Symposium, NSDI, vol. 10, 2010 Apirl.

[12] M. Jarschel, et aI., "SDN-based application-aware networking on the example of youtube video streaming", 2013 Second European Workshop on Software Defined Networks, IEEE, 2013 Oct, pp. 87-92.

[13] L.C.Cheng, K.Wang, Y.H.Hsu, “Application-aware Routing Scheme for SDN-based cloud datacenters”, 2015 Seventh International Conference on Ubiquitous and Future Networks, IEEE, 2015 Jul 7, pp. 820-825.

[14] C.H.Hsieh, J.W.Chang, C.Chen, S.H.Lu, "Network-aware service function chaining placement in a data center." *18th Asia-Pacific IEEE Network Operations and Management Symposium* (*APNOMS)*, 2016 Nov 10, pp. 1-6.

[15] Lebrun, David. "Leveraging IPv6 Segment Routing for Service Function Chaining" CoNEXT 2015 student workshop, 2015.