Design and Implementation of the Service-Aware Traffic Engineering (SATE) in the LISP Software-Defined Wireless Network (LISP-SDWN)

Sarang Wi
Dept. of Electrical and Computer Eng.
Sungkyunkwan University
Suwon, Korea
dnl9795@skku.edu

Eunil Seo
Dept. of Electrical and Computer Eng.
Sungkyunkwan University
Suwon, Korea
seoei2@skku.edu

Tai-Myoung Chung
Dept. of college of software
Sungkyunkwan University
Suwon, Korea
tmchung@skku.edu

***Abstract*— Software Defined Wireless Networks (SDWN) have been considered to have a feasible architecture that enables the fast deployment of new services and solutions in response to the explosion in the number of users and network traffic. Currently, the telecommunications sector is ensuring flexibility in network management and configuration. However, fluctuations in traffic are still beyond the control of SDWN providers. This paper suggests ways to manage fluctuations of traffic with service type. We propose the design of a service-aware network management service that achieves the maximum network utilization among heterogeneous Radio Access Networks (RANs) as a form of Traffic Engineering (TE). In this paper, we implement and test the Service-Aware Traffic Engineering (SATE) that distributes the network traffic to RANs according to the service type of traffic in the network layer. The traffic shift latency (e.g., from an LTE RAN to a Wi-Fi RAN) is considered as a performance metric that does not affect the end-to-end latency of some network applications (e.g., VoIP), and it is 3.51ms from our testbed. Therefore, it might not affect the end-to-end latency of the VoIP application in the telecommunications. SATE is implemented using OpenDaylight (ODL) and Ingress/Egress Tunneling Routers (xTRs) running on Vector Packet Processing (VPP).**

***Index Terms*— SDN, SDWN, LISP, Traffic Engineering, Service-Aware, OpenDaylight, VPP**

I. Introduction

For most use cases, the approaches to deploy SDNs have been limited to wired networks (e.g., Google data centers [1]), and to meet the needs of a Wireless Mobile Network (WMN), Software-Defined Wireless Networks (SDWN) have been proposed as a form of programmable Wireless Mobile Network (WMN) that decouples the data plane from the control plane, enabling direct programmability of the network controls and the abstraction of the underlying infrastructure for wireless applications [2]- [11].

However, the performance maturity of the SDWN has not yet been proven in carrier-grade networks requiring fast-failover, reliability, scalability and 50ms of the minimum required recovery time for the VoIP. We declaim that one of the performance degradation factors is the current IP address scheme. The current Internet architecture has suffered control overhead for Mobility Management (MN) by shapely increasing the number of mobile devices and traffic because it was not designed to easily accommodate mobility. Mobility requires an extra operational cost of the location management system that maintains the mapping between the identity and the location of the mobile client in this current Internet.

To cope with this limitation, the IP address scheme was replaced with LISP [12]- [15] in the LISP-deployed Software-Defined Wireless Network (LISP-SDWN) [16]. LISP-SDWN improved mobility management in a SDWN. LISP-SDWN adopted the Locator/ID Separation protocol (LISP) as a substitution of the current IP address scheme requiring the operation cost to maintain the location management system.

In LISP-SDWN, we propose a detailed design for the Service-Aware Traffic Engineering (SATE) and test the traffic shift latency from the Radio Access Network (RAN) to the other RAN to evaluate how much the latency affects time-critical, application such as VoIP, etc.

Motivation and Challenges: Our first research question was how to make the network management system aware of the service from the network traffic. To achieve this service awareness, we investigated the relationship between the End-ID (EID), which is the routable address for LISP, and the service. Moreover, we notice that the EID is supposed to not change at all. Finally, we came up with the idea of binding the EID and service in the network management system, such as a cloud service and cloud server EID. In this paper, we demonstrate how the service awareness is realized in LISP-SDWN and the test performance is carried out as well.

Our Contributions:

- As far as we know, we are the first frontier realizing service-aware traffic shifting in the network layer, not in the application layer.
- We invent a new approach though which we use binding information of the EID and service to manage the network traffic.
- We implement Service-Aware Traffic Engineering (SATE) into LISP-SDWN.
- We test and obtain the traffic shift latency to show the feasibility of SATE using our draft testbed.

IRJECE

This is an open access article under the CC BY-SA 4.0 license (https://creativecommons.org/licenses/by-sa/4.0/)

Section II described the background information for the LISP-SDWN and its related works. Section III proposes the SATE architecture and testbed. Section IV shows the test results and the performance analysis. The conclusion and future plans are provided in Section V.

II. Related Works and Background of LISP-SDWN

The Identifier/Locator (ID/LOC) scheme protocols are shown. Moreover, the LISP operation and background information for LISP-SDWN are described in this section. The tunnelling-based Identifier/Locator (ID/LOC) split protocols are LISP [12], HAIR [17], APT [18] and TRRP [19], which uses encapsulation included in a new header that has a source and destination locators. The address rewriting-based ID/LOC split protocols are ILNP [20], Six/One Router [21], RANGI [22], GLI-Split [23], IRON [24], HIP [25], Shimp6 [26] and Name-Based Socket [27], which replaces the existing source and destination addresses with the locators.

The Mapping System (MS) is linked with an EID-to-RLOC Map Database (afterword Map database), that is the key network infrastructure for internetworking with LISP sites in Fig. 1

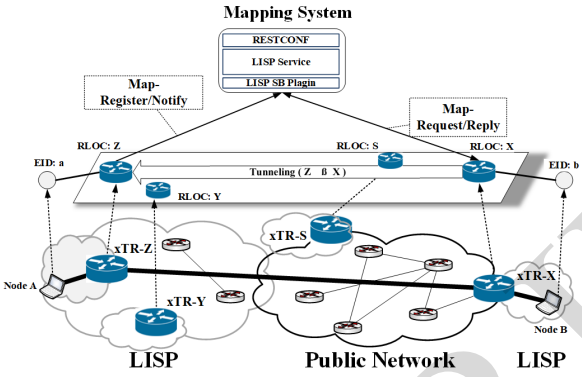


Fig. 1. System Architecture of LISP

Each Ingress/Egress Tunnelling Router(xTR) typically contains a small piece of a Map database: the EID-to-RLOC mapping for the EID-Prefixes behind the xTR. The EID-prefixes Map to one of the xTR's own globally visible IP addresses. The EID-to-RLOC Map entries must be configured on all xTR s for a given LISP site. A Map database is distributed in several Mapping Servers of the MS, and it maps EIDs to RLOCs as a location management system. From a LISP architecture perspective, MS enables LISP to split the location from the identity, which avoids the session disconnection by a host mobility. Once the mobile node with EID *a* initiate the session connection with a node with EID *b*, it will not change its session initiated address, EID *a* during the roaming in Fig. 1.

TABLE I. BASIC NOTATION

Notation	Name
MS	Mapping System
ETR	Egress Tunnelling Router
ITR	Ingress Tunnelling Router
GTR	Gateway Tunnelling Router
xTR	Ingress/Egress Tunnelling Router
RAN	Radio Access Network
Map Cache	EID-to-RLOC Map Cache
Map Database	EID-to-RLOC Map Database

With the Identifier/Location (ID/LOC) split scheme, the Mobile Nodes (MNs) can be seamlessly supported with multiple wireless interfaces, and they can handover and offload from an interface to another interface for user preference and engineering traffic purposes. In this sense, the proposed Service-Aware Traffic Engineering (SATE) is feasible in LISP-SDWN.

The LISP-deployed Software-Defined Wireless Network (LISP-SDWN) [16], a double separation architecture of the Control/Data plane decomposition and the Identifier/Locator split scheme, was proposed and designed as our previous work in Fig. 2.

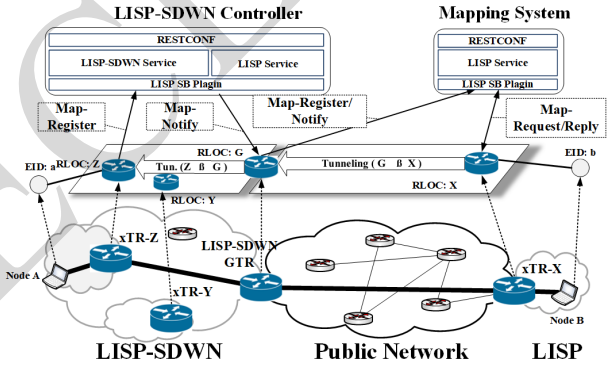


Fig. 2. System Architecture of LISP-SDWN

Map Database Entry of Mapping System(MS)

EID: M		
RLOC	Priority/Weight	State
g0	1/100	Up

SDWN *Map Database* Entry of LISP-SDWN Controller

EID: M			
RLOC	Priority/Weight	State	RAN
a0	1/100	Up	LTE
b0	2/50	Up	WiFi

OpenFlow Table

Rule	Sour. IP	Dest. IP	Ingress	Action
1	*	g0	2	Set "VLAN LTE" Actions: new_dstination:a0, Output to port 3
2	*	g0	2	Set "VLAN Wi-Fi" Actions: new_destination:b0, Output to port 4

Fig. 3. EID-to-RLOC Map Database Entry in LISP-SDWN GTR

This is an open access article under the CC BY-SA 4.0 license (https://creativecommons.org/licenses/by-sa/4.0/)

The LISP-SDWN Controller has a SDWN Map database, and it also manages both the EID-to-RLOC Map cache of the Ingress/Egress Tunnelling Routers (xTRs) and a flow table of the OpenFlow routers (See Fig. 2). The LISP-SDWN Controller is enhanced by both the information of an OpenFlow table and an SDWN Map database. A SDWN Map database is composed of the Map database of xTRs, and it is a locally accessible central database in the SDWN while a standard Map database of MS in Fig. 3 is a publicly-accessible distributed database.

Through the connecting the SDWN Map database of LISP-SDWN Controller and the Map database of the MS, LISP-SDWN achieves routing scalability. All RLOCs is assigned in a SDWN Map database and mapped to the RLOC of LISP-SDWN GTR (e.g., *G* in Fig. 2). Also, *G* is registered in MS. Instead of <*M, b0*>, Map database entry <*M, g0*> is installed in MS in Fig. 3 and it will attract packets destined to EID a to *G* from the outside of a SDWN. The LISP-SDWN Gateway Tunnelling Router (LISP-SDWN GTR) represents all xTR residing in a SDWN behind GTR. The LISP-SDWN controller can make its entire SDWN to be one LISP site to communicate with each other LISP sites operated by other providers.

The proposed LISP-SDWN achieves routing scalability, that is, seamless mobility among heterogeneous RANs without any LISP function modification during LISP deployment into a SDWN. It is powered up with EID-to-RLOC mapping information as well as with the flow information of OpenFlow tables in Fig. 2 and 3.

The SDWN Map database entry has one more extra field for the type of Radio Access Network, such as LTE or Wi-Fi, in Fig. 3. This means the MN with EID *a* has two connections with LTE and Wi-Fi, and it is possible to shift some traffic from the LTE connection to a Wi-Fi connection and vice-versa, and a further description is provided in the next section.

III. Service-Aware Traffic Engineering in LISP-SDWN

This section shows the architecture and data flow of Service-aware Traffic Engineering (SATE), as shown in Fig. 4 and 5, and the test scenario is provided. SATE is running on the Vector Packet Processing (VPP) platform, which was contributed by CISCO and implemented on OpenDaylight (ODL), as shown in Fig. 6.

TABLE II. Mapping Table Between Server and its EID

Name	EID
Web Server	6.0.1.0/24
DNS Server	6.0.3.0/24
Cloud Server	6.0.4.1/24
…	…

A. Assumption

In a LISP environment, the Routing Locator (RLOC) is separated from the End-Identifier (EID). Moreover, it is possible that each node does not change its EID during roaming. We assume that a server node has its own EID, and that the EID is known to the users who get a service from the server, such as a web server, cloud server, etc. In this sense, we propose a following Service-to-EID mapping table in Table II.

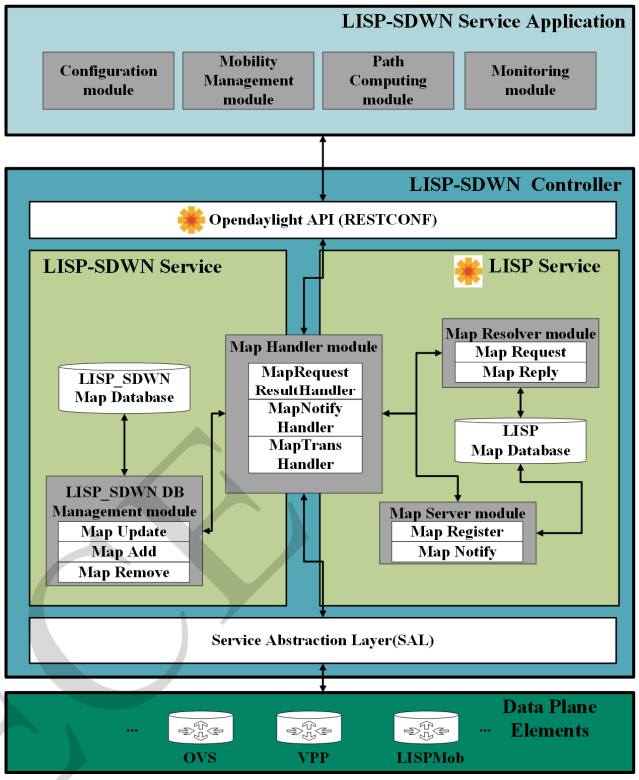


Fig. 4. LISP SDWN SATE Service Architecture

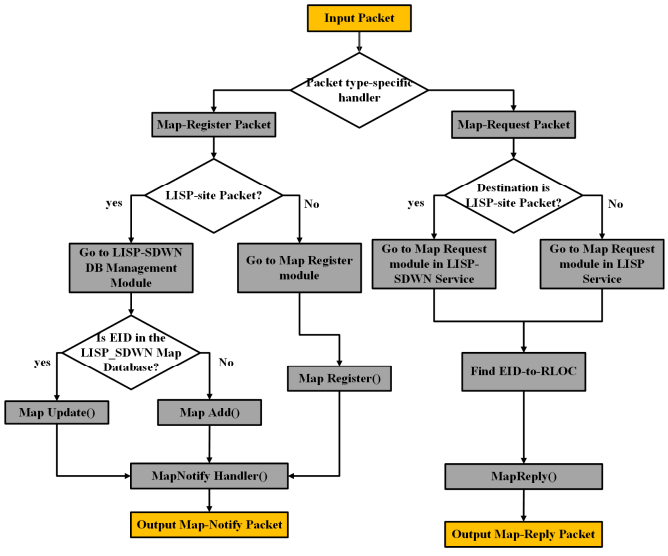


Fig. 5. LISP SDWN SATE Data Flow

IRJECE

This is an open access article under the CC BY-SA 4.0 license (https://creativecommons.org/licenses/by-sa/4.0/)

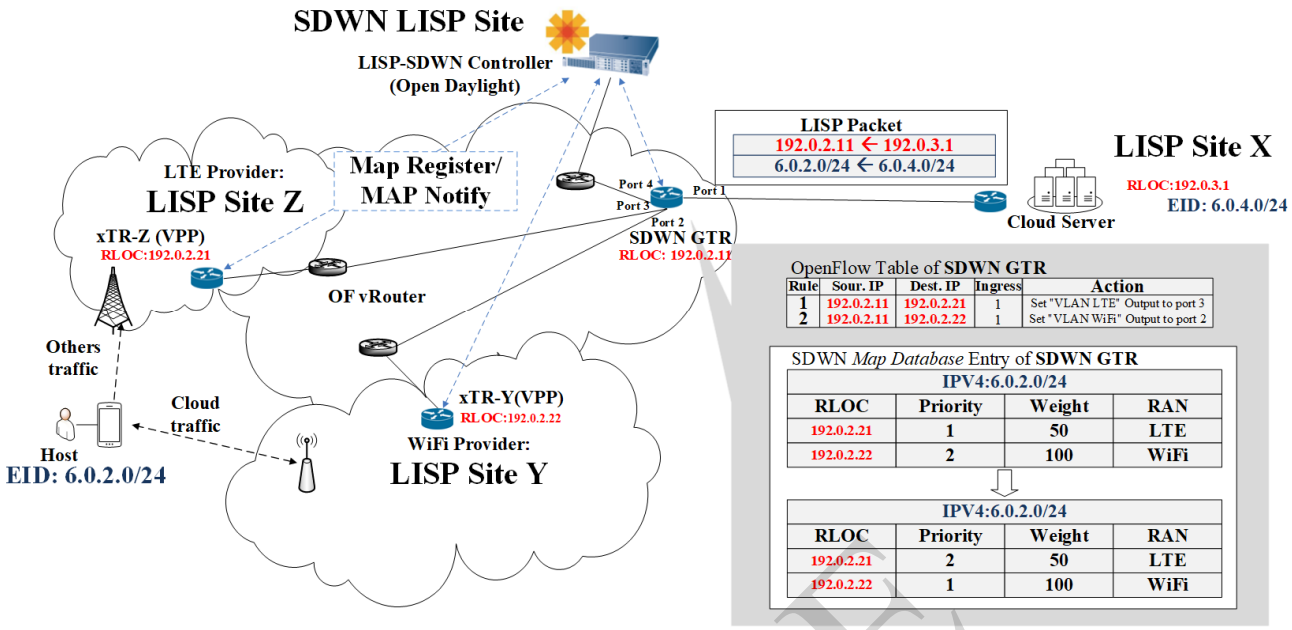


Fig. 6. Traffic Engineering Simulation with SATA

B. LISP-SDWN SATE Architecture

Our proposed SATE is based on LISP-SDWN, a double separation architecture of the Control/Data plane decomposition and Identifier/Locator (ID/LOC) split [16]. We design the Service-Aware Traffic Engineering (SATE) based on LISP-SDWN in detail and implement it to show the feasibility of SATE. The data flow and the system architecture for SATE are illustrated in Fig 4 and 5.

The Description of component are following:

- **LISP-SDWN DB Management Module:** This module manages and maintains map database entries consisting of EID of a node and RLOC of a xTR for tunneling purpose in the LISP-SDWN site. When a LISP-SDWN Controller receives a map registration, a map update or a map notification, this module updates a LISP-SDWN Map Database according to the type of request.
- **LISP-SDWN Map Database (Internal EID/RLOC)**: It is an internal map database composed by map information of the inside mobile nodes and stationary hosts and is managed by the LISP-SDWN Controller.
- **LISP Map Database (External EID/RLOC)**: It is a public map database managed by a standard Mapping System and has RLOC of the LISP-SDWN GTR attracting packets destined for internal mobile nodes.
- **Map Server Module**: This module is for the mapping service. The following two methods can be selected according to the location of the sender: 1) When a sender is within a LISP-SDWN, it sends the add or update request to the Map Translation Management Module and Map Handler Module. 2) When a sender is outside the LISP-SDWN, it sends the add or update request message to the Map Handler Module.
- **Map Resolver Module**: This module is for the mapping service on behalf of Map Sever Module, which processes the mapping search query and provides mapping information to the requester.
- **Map Handler Module**: When a register notification is received from the Map Server, this module provides notice regarding with an update of the Map database to the data plane element. When the map rule is received from the Map Resolver Module, this module delivers a reply to suit the needs of each request. When packets are transmitted from the outside of the LISP-SDWN to the inside of LISP-SDWN, this module operates as a LISP-SDWN service application.

C. Test Scenario

This section describes the Traffic Engineering (TE) scenario based on SATE. In Table II, the known servers are mapped with EID. The Traffic with EID from the cloud server is considered with the cloud service traffic, which does not require an expensive cellular connection because the cloud service traffic is defined as bandwidth-hungry and delay-tolerant.

Let's assume that a Host has two interfaces from a Wi-Fi RAN and a LTE RAN in a LISP-SDWN and it uses an LTE interface to upload/download data. However, the majority of the data is uploaded into the cloud server. From the SDWN

This is an open access article under the CC BY-SA 4.0 license (https://creativecommons.org/licenses/by-sa/4.0/)

provider's perspective, an expensive cellular connection is wasted, and this wasted cellular connection could be used to serve the user more.

To overcome this inefficient network utilization, the specific traffic can be classified in the LISP-SDWN Controller, and traffic shifting can be performed by SATE.

When a node gets data service, the data communication is set up and carried out between the node EID 6.0.2.0/24 and the other node EID 6.0.4.0/24 in Fig. 5. When LISP-SDWN GTR receives the LISP packets with a source EID 6.0.4.0/24 and destination EID 6.0.2.0/24, it notices the following information:

1) The Mobile Node with EID 6.0.2.0/24 has two interfaces, which is RLOC 192.0.2.21 from a LTE connection and RLOC 192.0.2.22 from a Wi-Fi connection through the SDWN Map database.

2) The source host with EID 6.0.4.0/24 is a cloud server for cloud service traffic.

Once LISP-SDWN GTR decides to forward the packets classified as a cloud traffic to the Wi-Fi network, it sets up the tunnelling between the LISP-SDWN GTR and xTR-B, and it capsulate the LISP packets with a source RLOC 192.0.2.11 and destination RLOC 192.0.2.22. The capsulated packets are forwarded to an OF vRouter via port 2 of the SDWN GTR according to the flow table. All OF-Routers in a SDWN have the corresponding flow entry issued by the LISP-SDWN Controller.

We demonstrate that the cloud traffic shifts from a cellular network to a Wi-Fi network by the LISP-SDWN Controller in the network layer. In Fig. 6, LISP-SDWN GTR has an SDWN Map database and a flow table.

IV. Performance Evaluation

We consider the following metrics to evaluate the performance of Service-Aware Traffic Engineering (SATE) in a LISP-SDWN. LISP-SDWN was designed for dynamic mobility in the Wireless Mobile Network (WMN), so the latency of the traffic shift performed by SATE should be short and should not affect the end-to-end communication latency.

Delay of Location Update is a time-delay to update the location information of the Mobile Node in the LISP-SDWN GTR by the LISP-SDWN Controller.

Delay of Traffic Shift is a time-delay to shift traffic from an interface to another interface (e.g., from an LTE Radio Access Network to a Wi-Fi Radio Access Network.)

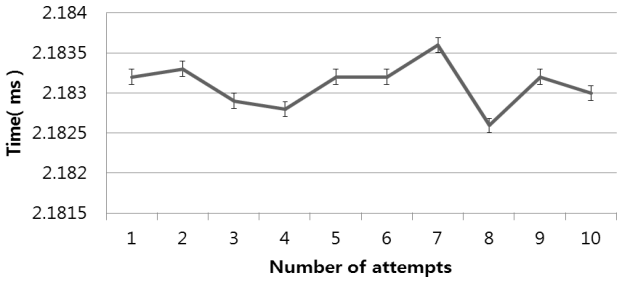


Fig. 7. Latency of EID-to-RLOC Map Change Notification

A. EID-to-RLOC Map Update

To show the location update delay, we simulate 10 times and average the delays. When the EID-to-RLOC Map entry from xTR-Y is updated, xTR-Y informs SDWN GTR of the updated Map entry by Solicit-Map-Request (SMR). The latency of the map entry change notification is shown in Fig. 7.

After the Map change notification by xTR-Y, LISP-SDWN GTR updates the EID-to-RLOC Map entry by sending a Map Request to the LISP-SDWN Controller, and it will receive a Map reply from a LISP-SDWN Controller. LISP-SDWN GTR sets up data tunnelling between the source RLOC 192.0.2.11 and the destination RLOC 192.0.2.22, and it forwards the traffic to xTR-Y. The latency of the Map entry update is shown in Fig. 8.

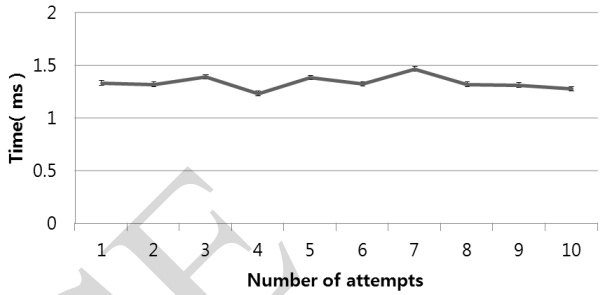


Fig. 8. Second step of EID-to-RLOC Map Update

B. Traffic Shift

Time-delay to shift the cloud service traffic from an LTE RAN to a Wi-Fi RAN measured through SATE. At this time, the sender's packet rate is 10 packets per second. Fig. 9 shows the latency to shift the cloud service traffic from xTR-Z to xTR-Y.

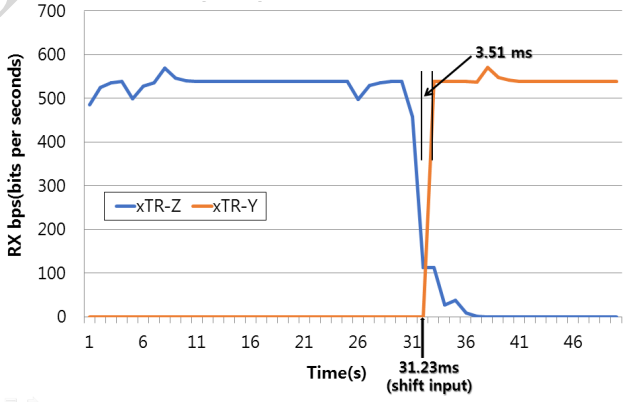


Fig. 9. Cloud Service Traffic is shifted frim xTR-Z to xTR-Y

V. Conclusion and Future Plan

From the perspective of a SDWN provider, some traffic (e.g., cloud service traffic) requires transmission on Wi-Fi or a small cell network to maximize the network utilization among heterogeneous Radio Access Networks (RAN) because the cloud service traffic does not require an expensive cellular

IRJECE

This is an open access article under the CC BY-SA 4.0 license (https://creativecommons.org/licenses/by-sa/4.0/)

network. The proposed Service-Aware Traffic Engineering (SATE) shows the feasibility with a low latency (3.51ms), which does not affect the minimum required recovery time for VoIP (e.g., 50ms). It is possible to achieve such a short delay because it is carried out in the network layer. As a future plan, we will research how to recognize the service type for traffic by implementing a Machine Learning algorithm.

ACKNOWLEDGMENT

This work was supported by the Institute for Information & Communications Technology Promotion (IITP) with a grant funded by the Korean government (MSIP) No B0184-15-1003 and No R-20160222-002755, and the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (NRF-2010-0020210).

This research was supported by the MIST (Ministry of Science and ICT), Korea, under the National Program for Excellence in SW supervised by the IITP (Institute for Information & communications Technology Promotion)(2015-0-00914).

REFERENCES

[1] O. N. Foundation. (2012) Software-defined networking: The new norm for networks [internet]. [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdnresources/white-papers/wp-sdn-newnorm.pdf

[2] X. Jin, L. Li, L. Vanbever, and J. Rexford, "Softcell: Scalable and flexible cellular core network architecture," Proc. 9th Int. Conf. Emerging Netw. Exp. Technol, pp. 163–174, 2013.

[3] C. Bernardos et al., "An architecture for software defined wireless networking," IEEE Wireless Commun, vol. 21, no. 3, pp. 52–61, 2014.

[4] A. Gudipati, D. Perry, L. E. Li, and S. Katti, "Softran: Software defined radio access network," ACM SIGCOMM HotSDN Workshop, Aug 2013.

[5] R. Riggio, M. Marina, J. Schulz-Zander, S. Kuklinski, and T. Rasheed, "Programming abstractions for software-defined wireless networks," Network and Service Management IEEE Transactions on, vol. 12, no. 2, pp. 146–162, Juen 2015.

[6] R. Riggio, K. Gomez, T. Rasheed, J. Schulz-Zander, S. Kuklinski, and M. Marina, "Programming abstractions for software-defined wireless networks," Network and Service Management (CNSM) 2014 10th International Conference on.

[7] B. Nunes, M. Mendonca, X. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past present and future of programmable networks," IEEE Commun. Surv. Tutor., vol. 16, no. 3, pp. 1617–1634, Aug 2014.

[8] N. McKeown et al., "Openflow: Enabling innovation in campus networks," SIGCOMM Comput, vol. 38, no. 2, p. 6974, Mar 2008.

[9] R. Atkinso and S. Bhatti, "Identifier-locator network protocol (ilnp) architectural description," RFC 6740, 2012.

[10] G. Sun, G. Liu, H. Zhang, , and W. Tan, "Architecture on mobility management in openflow-based radio access networks," in Proceedings of 2013 IEEE Global High Tech Congress on Electronics (GHTCE). IEEE, pp. 88–92, 2013.

[11] M. HadZialic, B. Dosenovic, M. Dzaferagic, and J. Musovic, "Cloudran: Innovative radio access network architecture," In E LMAR, 2013 55th International Symposium, pp. 115–120, Sept 2013.

[12] D. Farinacci, V. Fuller, D. Meyer, , and D. Lewis, "Locator/id separation protocol (lisp)," Internet Engineering Task Force, vol. RFC 6830, 2013.

[13] V. Fuller, D. Farinacci, D. Meyer, and D. Lewis, "Lisp alternate topology (lisp+alt)," Internet Engineering Task Force, vol. RFC 6836, 2013.

[14] D. F. D. Meyer, D. Lewis, "Lisp mobile node," Internet Engineering Task Force, vol. draft-meyer-lisp-mn-15, Jan 2017.

[15] V. Fuller and D. Farinacci, "Locator/id separation protocol (lisp) mapserver interface," Internet Engineering Task Force, vol. RFC 6833, 2013.

[16] V. Z. Eunil Seo and T.-M. Chung, "Leveraging software-defined wireless network (sdwn) by locator/identifier split scheme for mobility management," Journal of Communications, vol. 12, no. 8, pp. 464–474, 2017.

[17] A. Feldmann, L. Cittadini, W. Muhlbauer, R. Bush, and O. Maennel, "Hair: Hierarchical architecture for internet routing," in Proceedings of the 2009 Workshop on Re-Architecting the Internet. ACM, p. 4348, 2009.

[18] D. Jen, M. Meisel, D. Massey, L. Wang, B. Zhang, , and L. Zhang, "Apt: A practical transit mapping service," Internet Engineering Task Force, vol. draft-jen-apt-01, Nov 2007.

[19] W. Herrin, "Tunneling route reduction protocol (trrp)," Internet Engineering Task Force, 2008.

[20] R. Atkinson, S. Bhatti, , and S. Hailes, "Ilnp: Mobility, multi-homing, localised addressing and security through naming," Telecommunication Systems, vol. 42, no. 34, p. 273291, Dec 2009.

[21] C. Vogt, "Six/one router: A scalable and backwards compatible solution for provider-independent addressing," in Proceedings of the 3rd International Workshop on Mobility in the Evolving Internet Architecture. ACM, p. 1318, 2008.

[22] X. Xu, "Routing architecture for the next generation internet (rangi)," Internet Engineering Task Force, vol. raft-xu-rangi-04, August 2010.

[23] M. Menth, M. Hartmann, , and D. Klein, "Global locator, local locator, and identifier split (gli-split)," University of Wrzburg, vol. Technical Report, no. 470, Apr 2010.

[24] F. Templin, "The internet routing overlay network (iron)," Internet Engineering Task Force, vol. RFC6179, no. 470, Mar 2011.

[25] R. Moskowitz, "Host identity protocol (hip) architecture," Internet Engineering Task Force, vol. RFC 4423, 2006.

[26] E. Nordmark, "Shim6: Level 3 multihoming shim protocol for ipv6," Internet Engineering Task Force, vol. RFC 5533, 2009.

[27] J. Ubillos, M. Xu, Z. Ming, and C. Vogt, "Name-based sockets architecture," Internet Engineering Task Force, vol. draft-ubillos-name-based-sockets-03, September 2010.

IRJECE

This is an open access article under the CC BY-SA 4.0 license (https://creativecommons.org/licenses/by-sa/4.0/)