Journal of Network and Computer Applications 87 (2017) 32–45

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Review

Multi-domain Software Defined Networking: Research status and challenges

Franciscus X.A. Wibowo*, Mark A. Gregory, Khandakar Ahmed, Karina M. Gomez

RMIT University Melbourne, Australia

ARTICLE INFO

Keywords:
Software Defined Networking
Networking
OpenFlow
Controller
East-West
Multi-domain

ABSTRACT

A key focus of the transition to next generation computer networking is to improve management of network services thereby enhancing traffic control and flows while simplifying higher-level functionality. Software-defined networking (SDN) is an approach that is being developed to facilitate next generation computer networking by decoupling the traffic control system from the underlying traffic transmission system. SDN offers programmability in network services by separating the control plane from the data plane within network devices and providing programmability for network services. Enhanced connectivity services across the global digital network require a multi-domain capability. This paper presents a review of the current research status in SDN and multi-domain SDN, focusing on OpenFlow protocol, and its future related challenges.

1. Introduction

Software Defined Networking (SDN) concepts are moving from the data center to the enterprise networks and Wide Area Networks (WAN) presenting a number of challenges. Telecommunication network growth and complexity continues unabated and people and machines can communicate with each other through various network types utilizing a range of technologies. Global IP Traffic is predicted to increase from 59.9 Exabytes per month in 2014 to 168.4 Exabytes per month in 2019 (Cisco, 2015). Smartphone and tablet use accounted for 40% of the global IP traffic in 2014 and is expected to rise to 67% in 2019. This phenomenon has led to the need for a high capacity, reliable and yet cost-effective network that can carry increasing traffic volume, with dynamic and distinct applications for each entity.

The global digital networks are currently struggling to meet the increasing traffic volumes, and the shift from traditional enterprise networks to distributed Cloud Computing exacerbates this trend. The current network architectures were based on a vertically integrated approach with discrete semi-autonomous devices, which limit the potential for flexible flow management and network service innovation.

Service and network providers face challenges in operating networks, which consist of a vast number of network devices, for example, switches, routers, and gateways. IT departments must configure thousands of devices in order to implement network-wide services, which will result in difficulty to maintain Quality of Service (QoS), security and other policies. Current networks are built based on the monolithic or vertical approach, which integrates the control and forwarding functionality in a box, with vendor system standardization being a secondary consideration. The different vendor designs and construction approaches limit interoperability and have a detrimental effect on flexible network service innovation.

Global digital networks are evolving to cope with increasing traffic volumes and connected devices. There is a need for next generation network management and control systems that provide flexibility and device programmability, to facilitate dynamic updates and the introduction of new network services without hardware replacement. SDN offers one approach in providing "programmable networks" and vendors have generally adopted SDN as the next evolution of computer networking. SDN decouples the control plane from the forwarding plane in network devices and carries out traffic management utilizing a hierarchy of systems known as controllers (Foundation, 2012). Controllers connect to network switching and routing devices using open interfaces and protocols, e.g. OpenFlow (McKeown et al., 2008).

Scalability of the network has been one of the active and contentious topics in SDN. There are two common approaches in SDN controller implementation to improve the scalability, which includes centralised and distributed approaches. A centralised model is the simplest one, and it relies on the increase performance of standard controller. However, it introduces a single point of failure (SPOF) to the network. Distributed controller model eliminates SPOF and improves the scalability of the network, but it needs a method to coordinate all the controllers which could be in different domain.

A domain in SDN can be referred to an SDN administrative domain. Multi-domain SDN requires interconnection of controllers in different domains to exchange information across domain. Multi-domain SDN will enable the interconnection of global SDN domains, introduce

* Corresponding author.

http://dx.doi.org/10.1016/j.jnca.2017.03.004
Received 13 October 2016; Received in revised form 2 February 2017; Accepted 5 March 2017
Available online 07 March 2017
1084-8045/

interoperability between domains, and provide better provisioning of cross domain services. Currently, some ongoing researches are being done in multi-domain SDN, such as its architectures (Helebrandt and Kotuliak, 2014), distributed multi-domain controller architecture (Phemius et al., 2014), inter-domain communication platform (Lin et al., 2015) and application (Jahan et al., 2014), and routing mechanism (Kotronis et al., 2015).

Several surveys have studied SDN from different points of view. Jarraya et al. (2014) had compiled a survey on SDN providing the first taxonomy to classify SDN research works. Nunes et al. (2014) studied the state of the art of programmable network with the emphasis on SDN, along with its implementation alternatives, and its promising research directions. Another survey by Farhady et al. (2015) successfully present deep understanding of all three SDN layers. Recently, survey paper by Kreutz et al. (2015) presented a comprehensive survey on SDN which covered almost all aspect of SDN, starting from its definition, architecture and applications, until the current ongoing research efforts and challenges. However, those surveys do not present or only mentioned at very high level the multi-domain implementation aspects of SDN. Therefore, this article focuses on the multi-domain aspect of SDN.

In this paper, our aim is to provide an overview of the recent developments in multi-domain SDN (using academic and industry sources), and analyse the main research issues and approaches for future multi-domain SDN developments. The key contributions of this paper are:

- a tutorial on SDN and OpenFlow that includes a discussion of their origins, architecture and principal components.
- a review of controller implementation, both open-source and commercial, and a table with a comparison of the controller features.
- a review of current research into multi-domain SDN and the major challenges to be addressed by future research.

The rest of this paper is organized as follows. In Section 2, we present a brief overview of SDN, including its architecture and the OpenFlow protocol. Section 3 discuss the multi-domain implementation of SDN. in Section 4, we present multi-domain SDN challenges and identify the future research in multi-domain SDN. Section 5 concludes the paper.

2. Overview Software Defined Networking

Existing networks were generally built with proprietary hardware and systems from a single vendor. This would lead to a vendor “lock in”, where it was difficult to shift to another vendor or to adopt multi-vendor solutions. The use of vendor specific network devices and systems often led to the organization's systems becoming tailored to match the intricacies of the vendor equipment and systems as illustrated in Fig. 1. Programmable networks, that permitted the separation of the control and data planes, were seen to be the solution to the vendor “lock in” problem and the introduction of low-cost white label SDN-enabled networking devices provided a more flexible approach that organizations are now beginning to exploit. Although programmable networking was first introduced in the late 1990s, SDN has revolutionized the shift to programmable networking and SDN has become the focus of next generation networks.

2.1. SDN background

SDN has evolved over the past decade to provide a more flexible and dynamic networking architecture that incorporates improved support for management and network services. The SDN approach is for the management of traffic flows to be decoupled from the underlying infrastructure and systems that forward traffic. A standardized and open protocol was introduced to facilitate the separation of control and data planes. This protocol, known as OpenFlow, was developed to facilitate control traffic transfer between management systems, known as controllers, and the network devices, such as a switch, that forward data traffic.

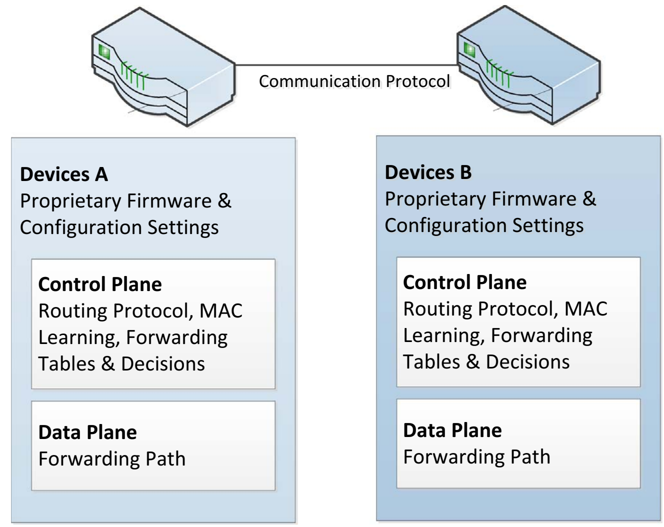

Fig. 1. Proprietary Network Element Configuration.

The development of SDN originated from the early work on programmable networking and the separation of control logic from the data transfer mechanism. There are two schools of thought regarding the concept of programmable networking including active networks and open signalling. However, the idea of decoupling the control logic from the data transfer mechanism emerged later, as a new architecture, to reduce the complexity of the distributed computations.

2.1.1. *Programmable Networks: Active Network and OPENSIG*

The active network concept was introduced in the mid-1990s in an endeavor to control a network in real-time. Active networking introduced a method that permits packets flowing through the network to carry instructions to be executed at network nodes. The code carried within the packets alters the network operation either temporarily for an individual packet or for a stream of packets. In this approach, the network devices become a dynamically programmable environment that can be dynamically altered using the code carried by the packets, which differs from the rigidity of traditional networking (Farhady et al., 2015; Xia et al., 2015).

Implementations of active networks include SwitchWare (Alexander et al., 1998) and conventional computer routing suites such as Click, XORP, Quagga, and BIRD (Xia et al., 2015). With the active networking implementations, the operations and behavior of the network can be modified dynamically. Although the active networking approach offered a new paradigm by providing a more dynamic environment, there was only minor development of the control plane. The active networking approach placed the intelligence at the end points (which can be inferred to be computers and servers acting as smart devices) whilst utilizing enhanced switches and routers to execute and carry out limited tasks based on the instructions carried within packets traversing the network. Thus, in active networking, packets are entities that can determine or control how nodes manage packets and streams.

In addition to the active networking approach introduced by the IP network community, another method known as Open Signalling (OPENSIG) was proposed by the telecommunication network community (Campbell et al., 1999). The OPENSIG suggested to provide an access to network hardware by means of open and programmable network interfaces. This idea was motivated by the need to separate the