Challenges and solutions to consistent data plane update in software defined networks

Sharvari Komajwar, Turgay Korkmaz

PII: S0140-3664(18)30357-8
DOI: https://doi.org/10.1016/j.comcom.2018.08.008
Reference: COMCOM 5764

To appear in: *Computer Communications*

Received date : 13 April 2018; Revised date : 7 July 2018; Accepted date : 15 August 2018

Challenges and Solutions to Consistent Data Plane Update in Software Defined Networks

Sharvari Komajwar
Department of Computer Science
The University of Texas at San Antonio
San Antonio, Texas-78249
Email: sharvari.komajwar@utsa.edu

Turgay Korkmaz
Department of Computer Science
The University of Texas at San Antonio
San Antonio, Texas-78249
Email: turgay.korkmaz@utsa.edu

Abstract— **Software Defined Networking (SDN) has emerged as a promising paradigm to make network management easier while supporting various applications requiring different guarantees in terms of performance, availability and correctness. In essence, SDN decouples the control and data planes such that a logically centralized SDN controller finds the paths based on given requirements, and then accordingly installs the necessary forwarding rules on the distributed SDN switches, which are simple forwarding devices. Due to various reasons such as congestions, failures or policy changes in the network, the SDN controller needs to find new paths and shift the traffic from the original paths to the new paths. However, during this transition, numerous time consuming steps need to be performed such as updating nodes on the new paths. If not carefully coordinated, these updating steps may cause loops, black-holes, and performance degradations. In this paper, we provide a detailed survey of the existing techniques that try to consistently update the new state of the data plane. We also discuss open issues and future challenges.**

Index Terms— **SDN; Routing; Network state consistency**

I. INTRODUCTION

Software Defined Networking (SDN) has been receiving significant attention due to its promises in simplifying the network management and supporting emerging applications, especially real-time cloud and big-data applications. Unlike the traditional networking paradigm, SDN has separated network management and control from forwarding functions. This decoupling of control plane and data plane enables SDN to be more flexible in performing many common network services such as routing [1]–[3], multicast [4], [5], security [6], bandwidth management, quality of service (QoS) [7]–[9] and storage optimization etc.

In general, SDN has a logically centralized controller (software) which makes network management and control decisions (e.g., finding paths based on current topology and given traffic requirements). The controller then installs the necessary forwarding rules on the simple forwarding switches in the data plane through a standard open protocol (e.g., OpenFlow [10]). The behavior of each switch in the data plane is determined by the installed rules. Upon receiving a packet, each switch looks up the rules installed by the controller. If there is a match, the packet is forwarded to the next switch based on the matched rule. Otherwise, the switch informs the controller about this packet and expects to get the necessary forwarding rules from the controller.

SDN has several advantages as a result of decoupling the control plane and the data plane. However, it also introduces new challenges such as how to quickly and correctly handle congestion and node/link failure. Many techniques have been proposed to deal with congestion [11], [12] and link/node failure [13]–[18]. In general, the SDN controller can find new paths for the affected flows using various path selection algorithms proposed in the literature [14]–[16], [19]–[21], and switch