Chapter 6
Toward Formal Verification of SDN Access-Control Misconfigurations

Amina Saadaoui
University of Carthage, Tunisia

ABSTRACT

Software-defined networking (SDN) allows centralizing and simplifying network management control. It brings a significant flexibility and visibility to networking, but at the same time creates new security challenges. The promise of SDN is the ability to allow networks to keep pace with the speed of change. It allows frequent modifications to the network configuration. However, these changes may introduce misconfigurations by writing inconsistent rules for single flow table or within a multiple open flow switches that need multiple FlowTables to be maintained at the same time. Misconfigurations can arise also between firewalls and FlowTables in OpenFlow-based networks. Problems arising from these misconfigurations are common and have dramatic consequences for networks operations. To avoid such scenarios, mechanisms to prevent these anomalies and inconsistencies are of paramount importance. To address these challenges, the authors present a new method that allows the automatic identification of inter and inter FlowTables anomalies. They also use the Firewall to bring out real misconfigurations.

INTRODUCTION

In SDN Network devices can be programmed via different communication protocols, such as OpenFlow. In fact, an openFlow network consists of a distributed collection
of switches managed by a program running on a logically-centralized controller. Each switch has a flow table that stores a list of rules for processing packets. Each rule consists of a pattern (matching on packet header fields) and actions (such as forwarding, dropping, modifying the packets, or sending them to the controller). The OpenFlow controller installs or uninstalls rules in the switches, reads traffic statistics, and responds to events. For each event, the controller program defines a handler, which may install rules or issue requests for traffic statistics. Therefore, OpenFlow and Software-Defined Networking (SDN) can simplify network management by offering programmers network-wide visibility and direct control over the underlying switches from a logically-centralized controller, but at the same time brings new security challenges by raising risks of software faults (or bugs), especially switches misconfigurations. Since companies rely only on the availability of their networks, such misconfigurations are costly. Due to the magnitude of this problem, our goal is to develop a method that allows to automatically identify configuration errors among the set of switches rules which should be well configured with respect to the firewall configuration. Finding the correct flow rules is challenging due to a number of reasons. First of all, an openflow switch generally comprises thousands of flow rules that are dependent and second flow rules do not always exactly match firewall rules.

In this paper, we propose a new approach to discover misconfigurations in real-case openFlow switches configurations already designed, our proposed method could be used also before updates occurred by the controller to verify if changes will induce further misconfigurations. This paper is organized as follows: Section 2 presents a summary of related work. Section 3 overviews the formal representation of firewall configurations and security policies and details FDD structure. In Section 4, we present our 65 method to discover and remove superfluous rules. In Section 5, we present our approach to discover simple and distributed firewalls misconfigurations. In Section 6, we articulate our approach to resolve simple firewall misconfigurations. In Section 7, we present first a study of the complexity of our inference systems, and then we address the implementation and evaluations of our tool. Finally, we present our conclusions and discuss our plans for future work.

RELATED WORK

Recently, there have been many verification tools proposed for SDN. Some tools debug controller softwares or applications, while others check the correctness of network policies.
XWRAPComposer: A Multi-Page Data Extraction Service
Ling Liu, Jianjun Zhang, Wei Han, Calton Pu, James Caverlee, Sungkeun Park, Terence Critchlow, David Buttler and Matthew Coleman (2008). Intelligent Information Technologies: Concepts, Methodologies, Tools, and Applications (pp. 1147-1173). www.igi-global.com/chapter/xwrapcomposer-multi-page-data-extraction/24336?camid=4v1a