A Tool for OpenFlow Network Managment

Callebe T. Gomes, Diogo Menezes F. Mattos, Natalia Castro Fernandes
and Otto Carlos M. B. Duarte
Grupo de Teleinformática e Automação - PEE/COPPE - DEL/POLI
Universidade Federal do Rio de Janeiro - Rio de Janeiro, Brazil
Email:{callebe,menezes,natalia,otto}@gta.ufrj.br

Abstract—**With the increasing use of virtual networks and the emergence of new technologies for creating and managing these networks, there is the need to create a tool that makes managing the virtual networks easy. This paper discusses the implementation of a tool for this purpose, based on OpenFlow and NOX technology to create virtual networks, and the features that this tool provides.**

I. Introduction

The OpenFlow platform [2] provides a new paradigm of network control and management. NOX [1], which is a centralized network controller for OpenFlow, creates network applications to control the packet flows, but, this applications provides a hard interface for humans. Hence, we developed an efficient management tool for OpenFlow networks based on NOX interface, called "Virtual Network Management System". Our tool provides the main features needed for controlling a virtual network plataform.

II. Implementation

The implemented the "Virtual Network management System" as a web application, because we wanted an application that can be easly accessed of any computer with an internet connection without the need to install any software. For an easier deployment, we implement our client interface based on HTML, CSS, JavaScript and SVG. Also, the Web Server was implemented with Python, to keep compatibility with Nox. Our tool works as follows. There is an application running in NOX that implements a web server. This web server provides an interface to the other developed applications in Nox over the HTTP protocol. Hence, the client of the Virtual Network Management System tool sends a request to the NOX web server, forwarding the query or command requested by the user to be executed on the network. The Virtual Network Management System is composed of three layers: Data, Presentation and Data Processing. The Data Layer is composed of the NOX applications that are responsible to execute the commands and get the network information. The Presentation Layer is composed by the HTML, CSS, SVG and JavaScript files that are responsible for providing information to the browser in a easy way to use. The Data Processing Layer is implemented between the Data and Presentation Layer. It is responsible for creating the messages to be sent to the Data Layer, for receiving the answers of the Data Layer, and for sending this answer for the Presentation Layer. Some parts of the Presentation Layer are written in Scalable Vector Graphics (SVG). It is a language that describes two-dimensional graphics and graphical applications in XML. SVG provides animation by the JavaScript manipulation.

III. Features

Virtual Network Management System provides features for the users such as information searching and network control/monitoring. We divided these features into three parts: Switches Statistics, Topology and Flow Switches. The Switch Statistics has two features. The first one is to collect and show all switch information such as: Switch descriptions, switch statistics, table statistics and flow information. The other feature of the Switch Statistics is to create flows for a particular switch. In this case, the web interface provides a formulary for the user to search a specific switch in the network. The Switches Topology shows the Network Topology and Spanning Tree Topology. This tool uses a software called Graphviz to make the topology graphs. In Topology Graph we can see some information such as: switches IP address, switches MAC address and the switches ports that are connected between which other. The Switches Flows tool is responsible for showing all the different flows on the network. For each different flow, the tool shows the switches that forward that flow and the corresponding input port in each switch. Our tool provides a search formulary to make easier the search for a particular flow. The search result is presented as a logical topology that shows how the switches forward the flow. In view of the logical topology there are four possible commands: create flow, delete flow, migrate flow of one path to another path, and move flows. In the flow migration, the flow is moved from a path and placed in another.

Acknowledgment

This work was supported by CNPq, CAPES, FINEP, FUNTTEL, and FAPERJ.

References

[1] N. Gude, T. Koponen, J. Pettit, B. Pfaff, M. Casado, N. McKeown, and S. Shenker. NOX: Towards an operating system for networks. *ACM SIGCOMM Computer Communication Review*, 38(3):105–110, July 2008.

[2] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S., and J. Turner. OpenFlow: Enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, 38(2):69–74, April 2008.