UDC 004.75

LOAD BALANCING EMULATION IN SOFTWARE-DEFINED NETWORKS USING MININET

Lopushen J. J., Postgraduate student
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine, Kyiv

The problem of testing a programmed behavior of the software-defined network controller has been examined in this article. Since testing the behavior of the controller is often impossible to test on the real network, the emulation of software-defined networks has been examined, as well as compatibility with a real SDN-controller. Possibilities of extending the functionality of SDN-controller has been analyzed. The ability of a POX SDN-controller to work with emulated networks has been utilized to set up load balancing in an emulated software-defined network with a star topology. An experiment with a strategy of load balancing, which was defined programmatically, has been conducted using Mininet emulator together with the Miniedit GUI. The strategy of load balancing in SDN has been programmed on the controller side without changing the network configuration. The results of the experiment have been presented. Conclusions on the future work, which will be dedicated to more complex algorithms, that can be tested using the suggested approach, has been made.

Key words: Software Defined Network, Load Balancing, SDN controller, POX controller, network emulation.

аспірант, Лопушен Є. Ю. Емуляція балансування навантаження в програмно-конфігурованих мережах з використанням Mininet/ Національний технічний університет України «Київський політехнічний інститут ім. Ігоря Сікорського», Україна, Київ

У статті проаналізовано проблему тестування запрограмованої поведінки контролера програмно-конфігурованої мережі. Так як тестування поведінки контролера часто є неможливим на реальній мережі, був проведений аналіз можливостей емуляції програмно-конфігурованої мережі, а також сумісності емульованої мережі з контролером SDN. Також, були розглянуті можливості розширення функціоналу контролера SDN. Можливості розширення функціоналу SDN контролера POX були використані для налаштування балансування навантаження в емульваній програмно-конфігурованій мережі з топологією типу "зірка". Був проведений експеримент з заданою програмно стратегією балансування навантаження на емуляторі Mininet з використанням Miniedit GUI. Стратегія балансування навантаження була налаштована на стороні контролера без зміни

конфігурації самої мережі. Результати експерименту представлені у вигляді графіка. Були зроблені висновки і поставлені задачі для подальшої роботи з тестуванням більш складних алгоритмів роботи з SDN.

Ключові слова: програмно-конфігурована мережа, балансування навантаження, SDN-контролер, контролер POX, емуляція мережі.

аспирант, Лопушен Е. Ю., Эмуляция балансировки нагрузки в программно-конфигурируемых сетях с использованием Mininet / Национальный технический университет Украины «Киевский политехнический институт им. Игоря Сикорского», Украина, Киев

В статье проанализирована проблема тестирования запрограммированного поведения контроллера программно-конфигурируемой сети. Так как тестирование поведения контроллера часто является невозможным на реальной сети, был проведен анализ возможностей эмуляции программно-конфигурируемых сетей, а также совместимости эмулируемой сети с контроллером SDN. Также, были рассмотрены возможности расширения функциональности контроллера SDN. Возможности расширения функциональности SDN контроллера POX были использованы для настройки балансировки нагрузки в эмулируемой программно-конфигурируемой сети с топологией типа "звезда". Был проведён эксперимент с программно заданной стратегией балансировки нагрузки на эмуляторе Mininet с использованием Miniedit GUI. Стратегия балансировки нагрузки была задана на стороне контроллера без изменения конфигурации самой сети. Результаты эксперимента представлены в виде графика. Были сделаны выводы и поставлены задачи для дальнейшей работы с тестированием более сложных алгоритмов работы с SDN.

Ключевые слова: программно-конфигурируемая сеть, балансировка нагрузки, SDN-контроллер, контроллер POX, эмуляция сети.

Introduction

Load balancing is an important task for modern distributed applications. A piece of software or hardware, that performs balancing the load between the nodes of a distributed application is called load balancer. Different types of load balancers are used for various tasks. This article suggests the implementation of software-based load balancer in a software-defined network.

Software-defined networks

Software-defined network (SDN) is a new concept of engineering a computer network [1], which suggests extracting the control layer of the network to the specialized piece of software, called controller. SDN controller makes it possible to perform centralized management and

monitoring of the network as well as exposes API for extending the set of features of the network. SDN controller interacts with the network hardware via specialized API called Northbound interface. Most of the switches used in SDN are so-called white-label [2], which means that they require no proprietary software to be installed and usually run a simple distribution of Linux operating system. Thus, the control over the network is fully delegated to the software, so flexible scaling [3] or hybrid cloud [4] can be implemented on the application level. In this paper, one of such controllers will be used, which retrieves the information of network hardware via the API and performs IP-based load balancing according to the programmatically-defined policy.

POX controller

Pox controller is one of the most widespread SDN controllers, written using Python programming language and has an API for managing the network from the outside [5]. The controller is fully open-source and is distributed together with Mininet within the same VirtualBox image. POX controller is convenient for simple topologies and supports Python scripts as behavior scenarios.

SDN emulator Mininet

The design of computer networks is impossible without the ability to check the potential result because experimenting on real hardware is often an expensive or impossible operation. In the case of software-defined networks, the behavior of the controller is required to be tested before deploying to the real server. Specialized emulators are used for testing the SDN controller on the chosen topology. One of the most widespread is an open source emulator Mininet [6]. The emulator is distributed in a bundle with Ubuntu Linux within a VirtualBox image.

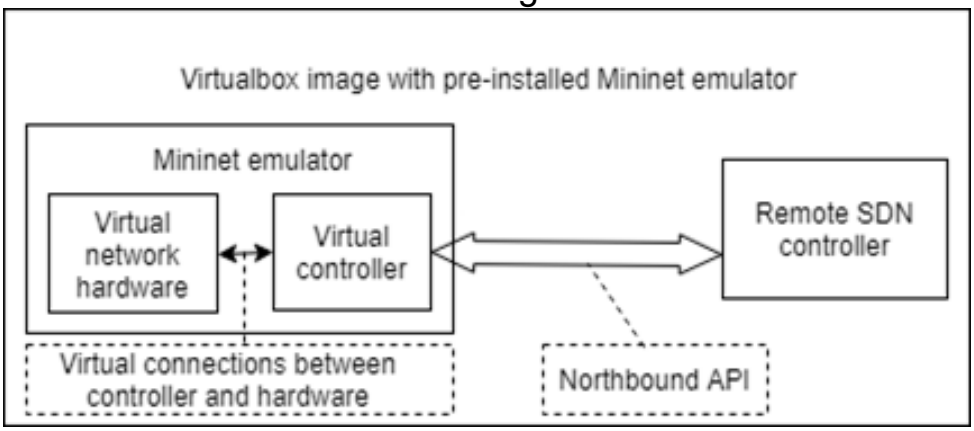

Fig. 1. Interaction between SDN and Mininet emulator

Graphical user interface Miniedit

Mininet emulator by default supports working with terminal only, however, there is a GUI available. The Miniedit tool lets network engineers build complex network topologies and conveniently manage the configuration of any piece of emulated hardware. Also, Miniedit provides

access to the terminal of any host of the network, which facilitates testing of emulation. For example, it is possible to ping to send requests via ping or curl utilities within the emulated network Such an advantage is used in this paper to illustrate the process of load-balancing in SDN.

Building a software-defined network in Mininet emulator

In this paper, a star topology is built in Miniedit to test the controller-based load balancer. In this configuration, a single switch is used, which is controlled by an SDN-controller. The switch is connected to 8 hosts, 6 of them act as clients and 2 have a simple test web-server running to receive requests from the clients. The curl utility is used to send HTTP requests.

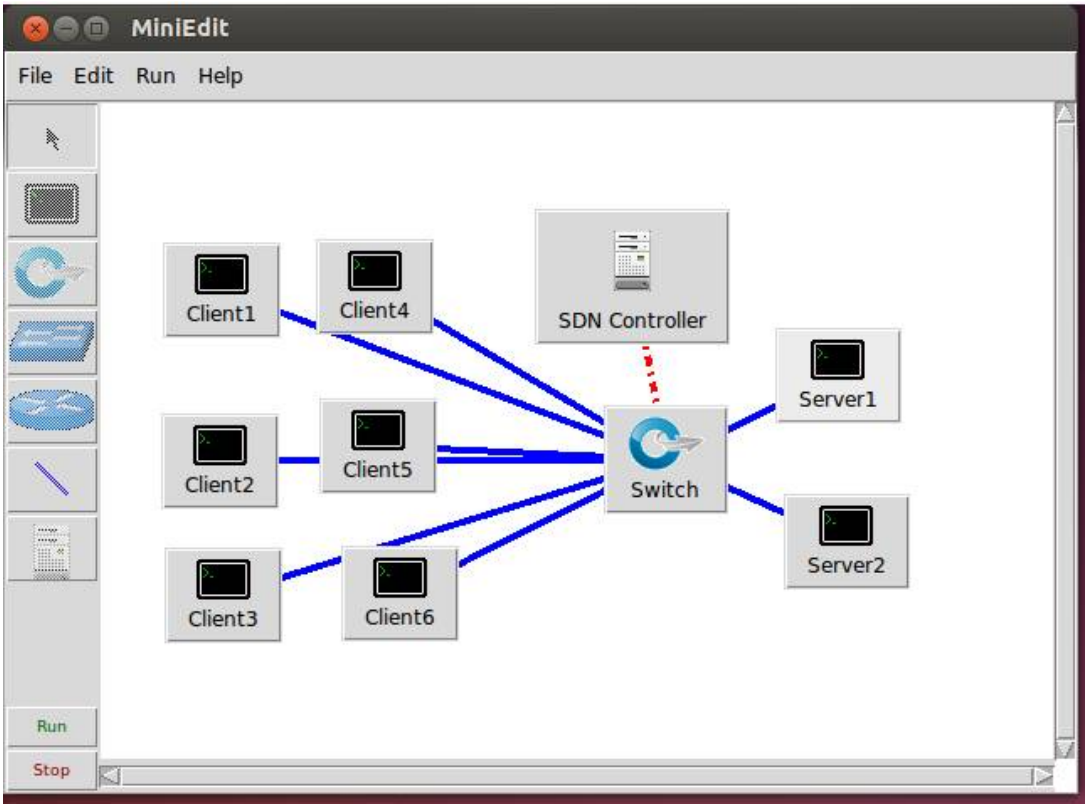

Fig. 2. the network topology for conducting the experiment

Miniedit automatically assigns the IP addresses to hosts and adds an initial flow table to the SDN controller. Default flow table assumes that all links are up, and all hosts can reach each other in case they are connected. The initial configuration is illustrated in the Fig.3.

Conducting the experiment of load balancing in Mininet

The experiment is conducted against the configured topology. The first server is assumed to be able to accept 2 times more requests than the second one. This behavior is programmed on the controller side. Overall 100 requests were made to the load balancer, configured on the controller from the clients in the network. Each request has been logged on the controller side. The count of the requests, that reached each server is illustrated on the Fig.4.

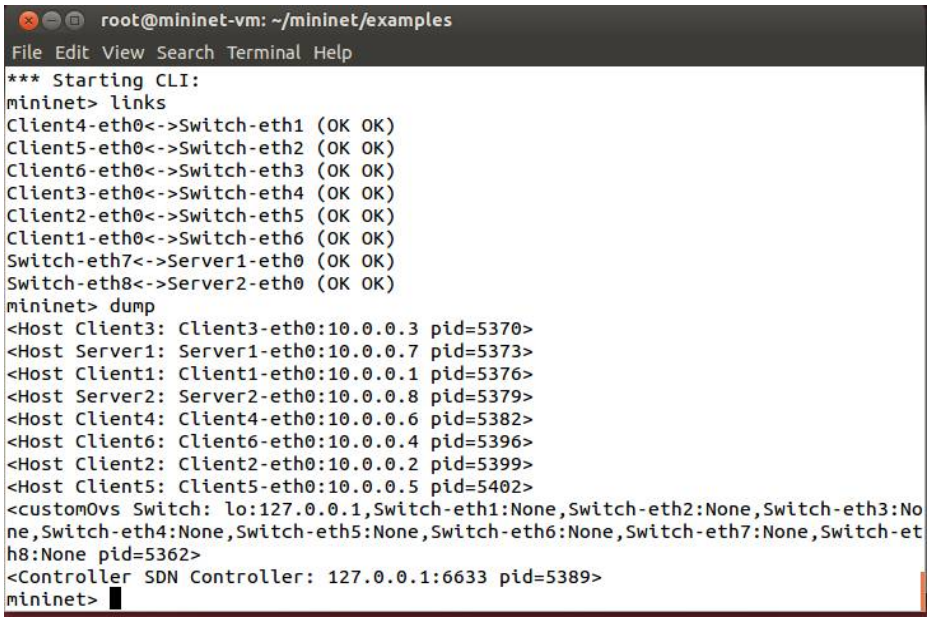

Fig. 3. The network configuration for conducting the experiment

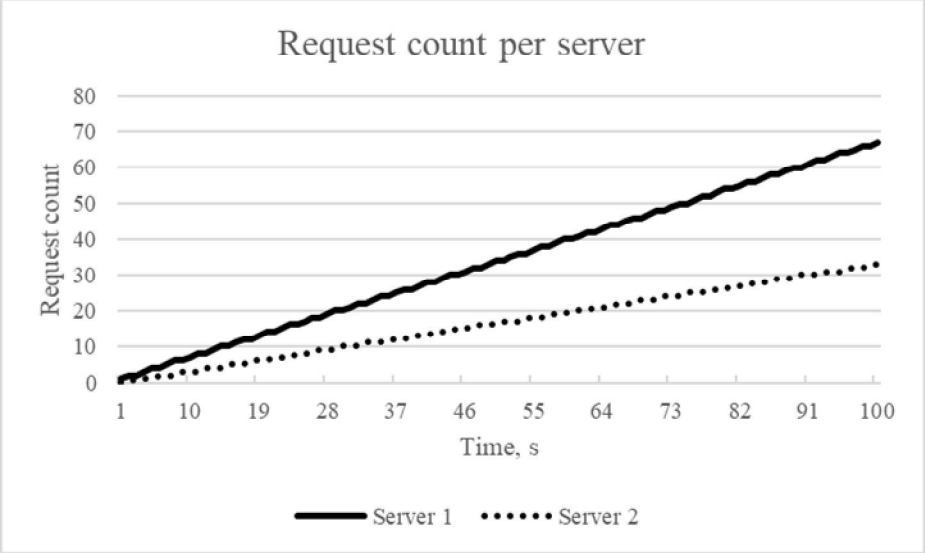

Fig. 4. Request count sent to servers

Conclusions and future work

This paper is focused on a way of emulation of a software-defined network in Mininet emulator in a bundle with Miniedit GUI. A software load balancer was run within the controller against the emulated network. The result of the experiment shows how the traffic can be redirected by a switch managed by an SDN controller. To conclude, the approach it is feasible to apply on different kinds of topologies. Future work will be dedicated to deploying more complex network topologies and programming more sophisticated behavior on the controller side.

Литература:

1. Monaco, M., Michel, O., & Keller, E. (2013, November). Applying operating system principles to SDN controller design. In Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks (p. 2). ACM.

2. Davoli, L., Veltri, L., Ventre, P. L., Siracusano, G., & Salsano, S. (2015, September). Traffic engineering with segment routing: SDN-based architectural design and open source implementation. In 2015 Fourth European Workshop on Software Defined Networks (pp. 111-112). IEEE.

3. Кулаков, Ю. А., & Лопушен, Е. Ю. (2017). Способ масштабирования распределённых приложений в программно-конфигурируемых сетях с использованием гибридного облака. Міжнародний науковий журнал" Науковий огляд", 8(40).

4. Кулаков, Ю. О., & Лопушен, Є. Ю. (2018). Способ динамической балансировки нагрузки в программно-конфигурируемых сетях. Адаптивні системи автоматичного управління, 1(32), 87-91.

5. Prete, L. R., Shinoda, A. A., Schweitzer, C. M., & de Oliveira, R. L. S. (2014, June). Simulation in an SDN network scenario using the POX Controller. In 2014 IEEE Colombian Conference on Communications and Computing (COLCOM) (pp. 1-6). IEEE.

6. Gupta, M., Sommers, J., & Barford, P. (2013, August). Fast, accurate simulation for SDN prototyping. In Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking (pp. 31-36). ACM.

References:

1. Monaco, M., Michel, O., & Keller, E. (2013, November). Applying operating system principles to SDN controller design. In Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks (p. 2). ACM.

2. Davoli, L., Veltri, L., Ventre, P. L., Siracusano, G., & Salsano, S. (2015, September). Traffic engineering with segment routing: SDN-based architectural design and open source implementation. In 2015 Fourth European Workshop on Software Defined Networks (pp. 111-112). IEEE.

3. Kulakov, Y. O., & Lopushen, J. J. (2017). The way of scaling distributed applications in software-defined networks using hybrid cloud. Mizhnarodnyi naukovyi zhurnal "Naukovyi ohliad", 8(40).

4. Kulakov, Y. O., & Lopushen, J. J. (2018). The way of dynamic load balancing in software-defined networks. Adaptyvni systemy avtomatychnoho upravlinnia, 1(32), 87-91.

5. Prete, L. R., Shinoda, A. A., Schweitzer, C. M., & de Oliveira, R. L. S. (2014, June). Simulation in an SDN network scenario using the POX Controller. In 2014 IEEE Colombian Conference on Communications and Computing (COLCOM) (pp. 1-6). IEEE.

6. Gupta, M., Sommers, J., & Barford, P. (2013, August). Fast, accurate simulation for SDN prototyping. In Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking (pp. 31-36). ACM.