Teaching SDN Security Using Hands-on Labs in CloudLab

Xiaohong Yuan
Computer Science Department
North Carolina A&T State University
Greensboro, USA
xhyuan@ncat.edu

Zhipeng Liu
Computer Science Department
North Carolina A&T State University
Greensboro, USA
zliu2@aggies.ncat.edu

Younghee Park
Computer Science Department
San Jose State University
San Jose, USA
younghee.park@sjsu.edu

Hongxin Hu
Computer Science Department
Clemson University
Clemson, USA
hongxih@clemson.edu

Hongda Li
Computer Science Department
Clemson University
Clemson, USA
hongdali@g.clemson.edu

***Abstract*—Software-Defined Networking (SDN) represents a major transition from traditional hardware-based networks to programmable software-based networks. While SDN brings visibility, elasticity, flexibility, and scalability, it also presents security challenges. This paper describes some of the hands-on labs we developed for teaching SDN security using the CloudLab platform. The hands-on labs have been used in a graduate level course on SDN/NFV related technologies. Our teaching experience of the hands-on labs is discussed. The hands-on labs can be adopted by other instructors to teach SDN security.**

Keywords—Software-defined networking (SDN), Network function virtualization (NFV), CloudLab, Security, Teaching

I. Introduction

Software Defined Network (SDN) offers a centralized, programmable and visible network that can dynamically evolve to the needs of businesses [1]. In comparison to a traditional network, the distinctive characteristics of SDN include the separation of control plane and data plane, a centralized view embodied in a simplified device acting as a controller, virtualization of all functions within the network, and the openness to change [2]. According to Google's report, the company has fully utilized its wide-area networks with SDN-based network management [3]. SDN shares close affiliations with Network Function Virtualization (NFV). NFV offers abstractions of hardware as key network functionalities, such as firewall, network connections, and load balancing [4]. Overwhelming management complexity, high costs, lack of scalability and slow market deployment rate are just a few notable drawbacks hardware-based network functions present [5]. The concept of server-less, built on the basics of NFV, is an emerging new paradigm in virtualization and has already significantly changed the economics of offloading computations to the cloud [6].

Significant granularity, visibility, flexibility, and elasticity are definite advantages that SDN and NFV bring to networking, but new security challenges are identified as well [5]. Several key security challenges in SDN have been identified and addressed, such as scanning attack prevention ([7] [8]), distributed denial-of-service (DDoS) attack detection [9], saturation attack mitigation ([10] [11]), topology poisoning attack prevention ([12] [13]), and Man-in-the-Middle (MITM) attacks ([14] [15]).

The cloud platform has been an effective delivery approach for cybersecurity education. However, commercial cloud platforms, such as Amazon Web Services (AWS), are expensive and restrictive to certain security labs. To meet the high demands of cybersecurity educators, an open laboratory platform named CloudLab has been proposed to create hands-on labs in.

CloudLab is sponsored by the NSF for academic researchers to develop and experiment on new cloud architectures and new cloud computing applications [5]. CloudLab provides easy-to-setup experimental environments created on the cloud for scientific research purposes on cloud computing. CloudLab is built upon a distributed infrastructure with clusters at three sites: Clemson University, University of Utah, and University of Wisconsin-Madison. CloudLab combines an estimate of 5,000 cores and 500 Terabytes of storage in the latest virtualization technology. For every connected node, CloudLab supports SDN technology such as 2x10 Gbps network interfaces. A 100 Gbps full-mesh SDN interconnect lets researchers instantiate a wide range of in-cluster experimental topologies.

CloudLab supports OpenFlow standard, which is an open standard protocol that organizes and monitors flows. CloudLab can be easily used in a two-step process: step 1 - create a user profile to encapsulate every resource component needed for the experiment (hardware, storage, network resources and software artifacts); and step 2 - instantiate the created profile to setup a virtualized experiment environment within a few minutes, when contrasted with traditional methods, this reduces request and wait times, as well as

979-8-6425-4896-7/20/$26.00 ©2020 CISSE

redeployment time. A profile can also be shared to make it accessible to a broader group of people.

A distinct gap exists between explanations of emerging SDN and NFV technologies and university course curricula across the nation [5]. This course introduces SDN and NFV, the attacks to the three main layers of SDN, and defense techniques shown in the current research. Students will complete hands-on labs that demonstrate the security issues of SDN/NFV and defense techniques.

The rest of the paper is organized as follows. Section II describes the labs used for this course. Section III describes how the course was managed and introduces our teaching methods. Section IV describes the teaching experience of this course. Section V concludes the paper.

II. SDN Labs

The SDN security labs in CloudLab used in the course consist of ten lab exercises. They are:

- Lab 1 Starting with CloudLab
- Lab 2 Software Defined Networking
- Lab 3 Local Host Hijacking
- Lab 4 Flooding Attacks to the SDN Data Plane
- Lab 5 Man-in-the-middle Attacks in the SDN Data Plane
- Lab 6 API Misuse Attacks to the SDN Controller
- Lab 7 MITM Attack with Flow Rule Manipulation
- Lab 8 FlowVisor
- Lab 9 Resolve Conflicting Flows

Labs 1, 2, 3, 4, 5, 6 were introduced and explicitly described in previous paper [5]. Lab manuals can be found online [30]. Labs 7, 8, and 9 are described below:

A. Lab 7 MITM Attack with Flow Rule Manipulation

- Lab Description: The controller is responsible for flow settings in switches so that all flow processing in the data-path is based on instructions from the controller. The controller then sets the flow rules in switch flow tables to either forward the flow packets to a particular port or drop packets coming from that particular source. The flow rules change depending on different network topologies, various user requests, and network protocols. This lab demonstrates how an user/attacker can modify flow rules using static flow pusher. Fig. 1 displays sequence diagram of this lab. Transparency in the figure refers to Host 3 (attacker) responding to Host 1 instead of Host 2 without Host 1 noticing the change.

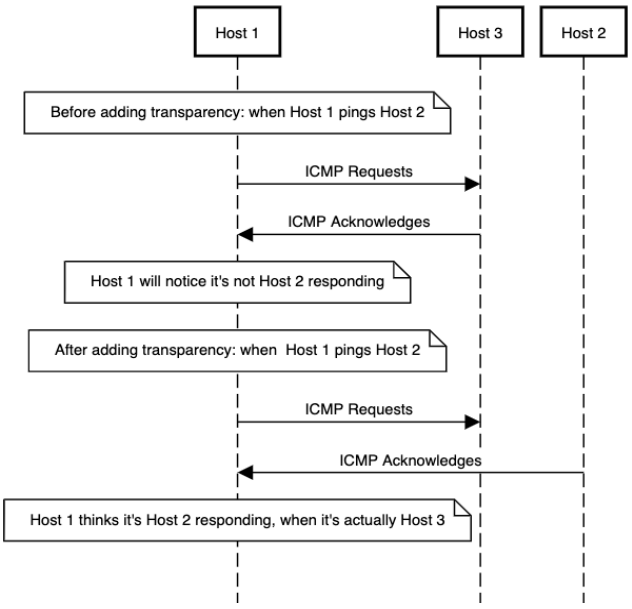

Fig. 1. Sequence diagram for Lab 7

- Learning Objectives: The students will be able to conduct a MITM attack through simple flow rule manipulation through the Representational State Transfer (REST) API.

B. Lab 8 FlowVisor

Lab Description: FlowVisor is a special purpose OpenFlow controller that acts as a transparent proxy between OpenFlow switches and multiple OpenFlow controllers. FlowVisor creates rich slices of network resources and delegates control of each slice to a different controller. Slices can be defined by any combination of switch ports (layer 1), src/ dst ethernet address or type (layer 2), src/dst IP address or type (layer 3), and src/dst TCP/ UDP port or ICMP code/type (layer 4). FlowVisor enforces isolation between each slice, i.e., one slice cannot control another's traffic. A slice is also a flow rule.

- Learning Outcomes: Students will be able to write flow rules to slice an OpenFlow network and have each slice be controlled by a separate controller.

C. Lab 9 Resolve Conflicting Flows

- Lab Description: The switch acts as the first line of filter for flows (a series of packets behaving the same way) in the data plane of SDN before the flows are allowed to be forwarded to the controller in the control plane. However, if conflicting flows occur frequently and switch is unable to respond, the flows are forwarded to controller and remain idle for the duration of the connection, this may lead to potential serious DoS attacks. This lab demonstrates how to resolve such conflicts with priority approach. Fig. 2 illustrates the sequence diagram of this lab. From the diagram it's obvious that our first rule conflicts with

our second rule, since the packets from Host 1 cannot go to two different hosts at the same time.

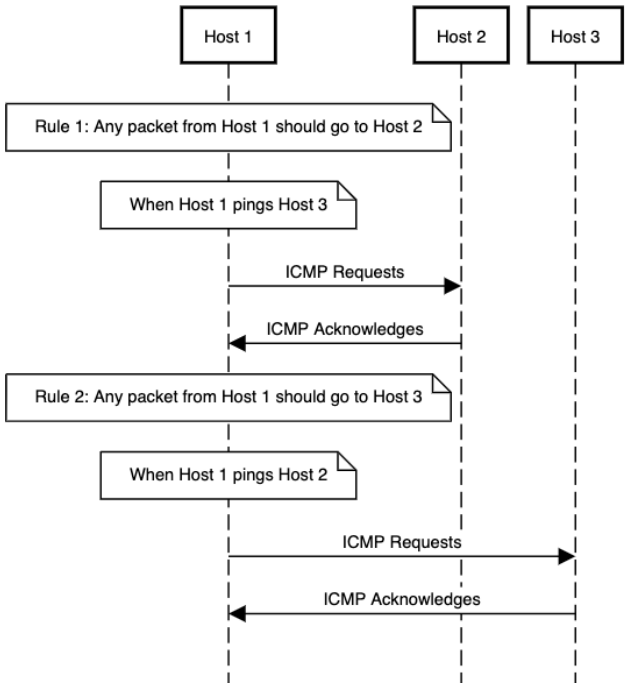

Fig. 2. Sequence Diagram for Lab 9

- Learning Outcomes: Students will be able to identify conflicting rules and segregate flows to ensure data packets are received by the intended users. Students will apply the rule priority approach to resolve flow conflicts and applying the common OpenFlow parameters.

III. Course on SDN Security

This course was taught as a special topic graduate level course entitled "Advanced Security for Emerging Networks" at North Carolina A&T State University in the Spring 2019 semester. This course held face-to-face meetings twice a week. Fifteen students enrolled in the class.

Upon completion of this course, we expect students to be able to:

- Explain the key components of SDN/NFV architecture and concepts
- Explain the major security issues in different layers of SDN/NFV
- Identify defense techniques for attacks to SDN/NFV
- Conduct research, and give presentations/tutorials on their research
- Conduct implementation-oriented hands-on labs related to SDN/NFV security

Almost every week of the semester, the students were asked to complete one of the nine listed labs. The students were then graded on completion of the lab. Each student submitted their work in the form of either screenshots or video recordings. Each student selected a SDN/NFV-related topic and developed a lab exercise that pertains to the topic. Each student was to present their final project in an approximate 10-minute span. The final projects were assessed based on the students' presentation skills, knowledge base, critical thinking and overall impressions. Then students were given a lab survey and a course survey. The results of labs and surveys are discussed in the next section.

This course was designed in seminar style, executed through guided inquiry collaborative learning ([20] [21]). Each student was assigned to prepare materials to either teach one or several chapters of the selected textbook [22] or teach and demonstrate a lab. This style requires students to study, prepare and have adequate knowledge of the subject, as a result the students enhance their teaching skills while stimulating their fellow students to actively participate in discussions ([16] [17]). The students demonstrated creativity and utilized many teaching methods and tools, including gamification tools such as Plickers, Kahoot, and multimedia tools such as YouTube videos and PowerPoint slides. Past research has indicated that the use of gamification tools significantly adds to project-based learning [18]. One student taught the class using a method similar to POGIL teaching. The student created the teaching material as handouts. Students first had to read material to build up knowledge, then discussed in groups before finally answering the assessment questions from the handouts. The mixing of these teaching methods increased learner's motivation, enhanced understanding of technical content and brought an upbeat atmosphere. Previous research shows that the use of gamification tools allow faculty to clearly identify whether the students have successfully mastered the concepts and allow instructors to further structure peer-to-peer active learning more effectively in class [19].

IV. Teaching Experience

An anonymous student survey and a questionnaire on student reflection was conducted on the course module. This section presents some details from the results.

The survey results are shown in Table I.

A total of twelve students participated in the survey. Students' self-ranking on knowledge attained in learning objectives for the labs showed that eighty-three-point four percent (83.4%) strongly agreed or agreed that the learning objectives of the labs are met. The majority of labs requires working in terminal on Linux, which 100% of students responded that they have strong familiarity.

Even though eighty-three percent (83%) of students believe labs are somewhat difficult, seventy-five percent (75%) of students believe that they are more interested in computer security after taking this course. Seventy-five percent (75%) students expressed having either high or very high interests in the labs. Almost 70% of class was very motivated to learn the labs. Almost 100% of the students

found that the hands-on learning aspect of the class was the most valuable to their learning. Majority of students also commented they wish to apply the knowledge learned in this course to their own research areas. One hundred percent (100%) of students recognized SDN and NFV as easy to deploy and advantageous to any other methods they've experienced using.

TABLE I. SURVEY RESULTS ON SDN SECURITY LABS

Survey Question	Agree	Strongly Agree
Have strong familiarity with Linux.	25%	75%
My preparation and ability were sufficient for me to successfully understand the labs.	66.67%	25%
The lab instructions were clear.	41.67%	25%
The labs are somewhat difficult.	33.33	41.67%
I clearly understand the objectives of the labs.	33.3%	66.7%
The labs were a valuable part of this course.	25%	75%
Approximately, I spent more than an average of 5 hours on each lab.	50%	16.66%
The most time-consuming part of the labs is instantiating and prerequisite installations.	83.37%	
A result of the labs, I am more interested in computer security	83.37%	

When answering the question "What changes could be made to the labs to enhance your learning," some students said that they expected to learn more mitigation methods for the problems posed by the labs, while some wished to have more demo videos available when they attempted the labs. When answering to the question "The most important thing learned from the lab experience," the class reached a consensus by identifying "learning SDN, NFV and many methods and tools to simulate attack and explore" as the most important learning outcome.

The class created several interesting labs, the following is a listing of several topics.

- Project 1 SDN controller NOX/POX Lab – introduce the basic steps in developing net apps using OpenFlow framework on NOX controller.
- Project 3 Lab on Managing a Virtual Network Function (NFV): Load Balancing using Round Robin Control with Ryu Controller – This lab provides hands-on experience with Ryu controller and load balancing.
- Project 5 Mitigating Host Location Hijacking Attacks Lab - This lab demonstrates how TopoGuard is used to mitigate a topology poisoning attack.
- Project 10: Lab on Open-Source Routing and Network Simulation Using the OpenDayLight SDN Controller with the Mininet Network Emulator, and with MiniEdit Mininet Graphical User interface - This lab demonstrates some features of OpenDaylight and how to capture OpenFlow messages get exchanged between the controller and the emulated switches.
- Project 12 Introduction Lab to OpenFlow Tutorial (OVS) with Ryu Controller

More specifically on Project 3:

Project 3 Lab on Managing a Virtual Network Function

- Lab Description: The goal of this lab is to give the students a hands-on experience with OpenFlow, and how it can be used for NFV deployment. Using a basic topology, which contains two sources, a destination, two virtual network functions (IDS), an OVS and a controller, we will show how different OpenFlow rules can be used for NFV management. This is a lab exercise that is split into two parts.
- Part 1 Description: In first half of the lab, the student will explore Round Robin load balancer for a VNF Snort application. Snort will be running as IDS on VNF1 and VNF2 and student will try to balance the load across the two VNF instances by directing each new flow request to one of the two VNF instances in a round robin fashion. Student will use the Netcat application to generate traffic between a source and destination.
- Part 2 Description: In second part of the lab, student will use a Proportional Integral approach to do load balancing. An overview of the system is shown in Fig. 3. In this approach, the load on VNF1 and VNF2 is monitored, and flow-forwarding decisions are made based on the load information retrieved from the hosting VMs. Student will run a RINA distributed application to get the state (load) of the VNFs to the controller VM. Once the Ryu controller has the IDS load information, it will use the Proportional Integral (PI) load balancer to balance the load across the VNF instances based on the load information. This load balancing information is then provided to the Ryu controller, which updates the OpenFlow rules on the OVS switch to balance the load.

979-8-6425-4896-7/20/$26.00 ©2020 CISSE

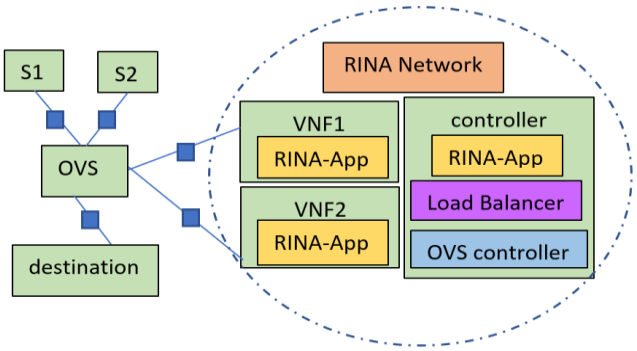

Fig. 3. Overview of a proportional integral approach

- Learning Outcomes: Students will identify and learn two load-balancing approaches while utilizing NFV. Students will also be able to slice network using RYU controller and apply various API packages such as SNORT, LUAJIT, GHTTP2, DAQ and RINA.

When asked to "Describe how your experience with teaching a topic in this class helped you identify and develop professional qualities and skills" in the student reflection questionnaire, one student answered this way:

> *"I learned how to prepare a lecture and the work involved in sifting through information and resources to present an educational session to a class. It is one thing to read papers and to gain an understanding. That's a skill I have developed over the years as a student. It is another to take what you have learned and organize and synthesize the information in order to present it as a lecture. It's quite difficult at this point, and I learned a lot from that exercise. Specifically, how to pick out what is important for understanding the concept, what is important for a person to know generally about the concept, and what forms a useful knowledge of the concept."*

V. Conclusion

This paper describes a course designed to teach students about SDN security knowledge through hands-on labs in CloudLab, and how the SDN related security vulnerabilities can be exploited. The course consists of nine hands-on lab exercises simulating various attacks as well as delivering core foundation knowledge. Students learn and apply the concepts to NFV and the three main layers of SDN. Students were also asked to apply the acquired knowledge and create new labs. Student were required to present course topics under the supervision of the course instructor.

The course was taught in the Spring 2019 semester. Our teaching experience showed that students were highly interested in the labs and ended the course with more interest in computer security. The hands-on labs on SDN security taught in this course may be adopted by instructors teaching network security, web security, and network functions.

Since students from the current course designed new labs for the subject, these labs may be included as part of the course in the future. More sophisticated labs can also be introduced for the course. Potential subjects that can be taught in the course are serverless [6], lightweight virtualization [28], and IoT management [29].

Acknowledgments

This work was partially supported by grants from National Science Foundation (NSF-DGE-1723663, NSF-DGE-1723804, and NSF-DGE-1723725). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF.

References

[1] D. Kreutz, F.M. Ramos, P. Verissimo, C.E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14-76, 2015.

[2] P. Goransson, C. Black, and T. Culver, *Software defined networks: a comprehensive approach*. Morgan Kaufmann, 2016.

[3] A. Vahdat, D. Clark, and J. Rexford, "A purpose-built global network: Google's move to SDN," *Queue*, vol. 13, no. 8, pp. 100, 2015.

[4] European Telecommunications Standards Institute, "Network Function Virtualisation - Introductory White Paper," European Telecommunications Standards Institute, 2012. [Online]. Available: https://portal.etsi.org/nfv/nfv_white_paper.pdf. [Accessed: Mar. 12, 2007].

[5] Y. Park, H. Hu, X. Yuan, and H. Li, "Enhancing Security Education Through Designing SDN Security Labs in CloudLab," In Proc. of the 49th ACM Technical Symposium on Computer Science Education, Feb. 2018, pp. 185-190.

[6] P. Aditya, I.E. Akkus, A. Beck, R. Chen, V. Hilt, I. Rimac, ... and M. Stein, "Will Serverless Computing Revolutionize NFV?" Proc. of the IEEE, 2019.

[7] K. Giotis, C. Argyropoulos, G. Androulidakis, D. Kalogeras, and V. Maglaris, "Combining OpenFlow and sFlow for an effective and scalable anomaly detection and mitigation mechanism on SDN environments," *Computer Networks*, vol. 62, pp. 122-136, 2014.

[8] S. A. Mehdi, J. Khalid, and S. A. Khayam, "Revisiting traffic anomaly detection using software defined networking," In Proceedings of the 14th international conference on Recent Advances in Intrusion Detection (RAID'11), Springer-Verlag, 2011, pp. 161–180.

[9] B. Braga, M. Mota, and P. Passito, "Lightweight ddos flooding attack detection using nox/openflow," In Proceedings of the 2010 IEEE 35th Conference on Local Computer Networks (LCN'10). IEEE, 2010, pp. 408–415.

[10] S. Lim, S. Yang, Y. Kim, S. Yang, and H. Kim, "Controller scheduling for continued SDN operation under DDoS attacks," *Electronics Letters*, vol. 51, no. 16, pp. 1259-1261, 2015.

[11] R. Mohammadi, R. Javidan, and M. Conti, "Slicots: An sdn-based lightweight countermeasure for tcp syn flooding attacks," *IEEE Transactions on Network and Service Management*, vol. 14, no. 2, pp. 487-497, 2017.

[12] M. Dhawan, R. Poddar, K. Mahajan, and V. Mann, "Sphinx: Detecting security attacks in software-defined networks," in Proceedings of the 22nd Annual Network and Distributed System Security Symposium (NDSS'15), February 2015.

[13] S. Hong, L. Xu, H. Wang, and G. Gu, "Poisoning network visibility in software-defined networks: New attacks and countermeasures," in Proceedings of the 22nd Annual Network and Distributed System Security Symposium (NDSS'15), February 2015.

[14] J. Yao, Z. Han, M. Sohail, and L. Wang, "A Robust Security Architecture for SDN-Based 5G Networks," *Future Internet*, vol. 11, no. 4, pp. 85, 2019.

[15] J.P. Mabel, K.A. Vani, and K.R.M. Babu, "SDN Security: Challenges and Solutions," In Emerging Research in Electronics, Computer Science and Technology, Springer, Singapore, 2019, pp. 837-848.

[16] A. Spruijt, A.D.C Jaarsma, H.A.P. Wolfhagen, P. van Beukelen, and A.J.J.A. Scherpbier, "Students' perceptions of aspects affecting seminar learning." *Medical teacher*, vol. 34, no. 2, pp. 129-135, 2012.

[17] Spruijt, A., Wolfhagen, I., Bok, H., Schuurmans, E., Scherpbier, A., Van Beukelen, P., & Jaarsma, D. (2013). Teachers' perceptions of aspects affecting seminar learning: a qualitative study. BMC medical education, 13(1), 22.

[18] Khan, A., Ahmad, F. H., & Malik, M. M. (2017). Use of digital game based learning and gamification in secondary school science: The effect on student engagement, learning and gender difference. Education and Information Technologies, 22(6), 2767-2804.

[19] Leung, E., & Pluskwik, E. (2018). Effectiveness of Gamification Activities in a Project-based Learning Classroom.

[20] Hanson, D. M. (2006). Instructor's guide to process-oriented guided-inquiry learning. Lisle, IL: Pacific Crest.

[21] Shih, J. L., Chuang, C. W., & Hwang, G. J. (2010). An inquiry-based mobile learning approach to enhancing social science learning effectiveness. Journal of Educational Technology & Society, 13(4), 50-62.

[22] Goransson, P., Black, C., & Culver, T. (2016). Software defined networks: a comprehensive approach. Morgan Kaufmann.

[23] H.Wang, L. Xu, and G. Gu, "Floodguard: A dos attack prevention extension in software-defined networks," in Proceedings of the 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'15) , June 2015.

[24] J. Fruhlinger, "The Mirai botnet explained: How teen scammers and CCTV cameras almost brought down the Internet," *CSOonline*, vol. 9 March, 2018.

[25] B.H. Oh, S. Vural, N. Wang, and R. Tafazolli, "Priority-Based Flow Control for Dynamic and Reliable Flow Management in SDN," *IEEE Transactions on Network and Service Management*, vol. 15, no. 4, pp. 1720-1732, 2018.

[26] M. Zhang, J. Bi, J. Bai, Z. Dong, Y. Li, and Z. Li, "FTGuard: A Priority-Aware Strategy Against the Flow Table Overflow Attack in SDN," In SIGCOMM Posters and Demos, pp. 141-143, August 2017.

[27] R. Izard, "Project Floodlight," *atlassian.net*, Apr. 29, 2019. [Online]. Available:https://floodlight.atlassian.net/wiki/spaces/floodlightcontroller/pages/1343518/Static+Entry+Pusher+API. [Accessed Apr. 29, 2019].

[28] R. Morabito, "Lightweight Virtualization in Edge Computing for Internet of Things," 2019.

[29] A.M. Zarca, J.B. Bernabe, R. Trapero, D. Rivera, J. Villalobos, A. Skarmeta, and P. Gouvas, "Security Management Architecture for NFV/SDN-aware IoT Systems," IEEE Internet of Things Journal, 2019.

[30] Y. Park, H. Hu, X. Yuan, "Security Hands-on Labs in SDN/NFV," June, 2019. [Online]. Available: https://irislab.me/index-lab.html [Accessed Jun. 10, 2019]

979-8-6425-4896-7/20/$26.00 ©2020 CISSE