The cost of Security in the SDN control Plane

Raphael Durner
Chair of Communication Networks
Technische Universität München
r.durner@tum.de

Wolfgang Kellerer
Chair of Communication Networks
Technische Universität München
wolfgang.kellerer@tum.de

ABSTRACT

In OpenFlow enabled Software Defined Networks (SDNs) network control is carried out remotely via a control connection. In order to deploy OpenFlow in production networks, security of the control connection is crucial. For OpenFlow connections TLS encryption is recommended by the specification. In this work, we analyze the TLS support in the OpenFlow eco-system. In particular, we implemented a performance measurement tool for encrypted OpenFlow connections, as there is non available. Our first results show that security comes at an extra cost and hence further work is needed to design efficient mechanisms taking the security-delay trade-off into account.

CCS Concepts

•Networks → Network performance analysis;

1. INTRODUCTION

SDN and OpenFlow enable a huge number of new attack mitigation and reaction methods, as is depicted in reviews [17, 25, 24]. However one drawback regarding security is the centralized control plane as it introduces new attack vectors [18, 14, 26]. One example is the misuse of network hypervisors [27] to create black hole networks that can lead to data leakage without the network administrators noticing [19]. In order to protect the control plane connections against such kind of attacks, the OpenFlow protocol recommends the use of Transport Layer Security (TLS) for the control plane connection. TLS provides both encrypted and authenticated communication and can therefore prohibit a number of attacks at the control plane level.

CoNEXT Student Workshop'15, December 01 2015, Heidelberg, Germany

ACM 978-1-4503-4066-3/15/12. . . $15.00
DOI: http://dx.doi.org/10.1145/2842665.2843563

In this work, we explore the current adoption rate of TLS in the OpenFlow eco-system and we present measurements that show the cost of TLS encryption. Although there are some works, which study control plane performance of OpenFlow switches [15, 22, 16, 20, 21], to the best of our knowledge no one has inquired encryption yet. Therefore, we want to study delay aspects of the encryption in the control plane.

2. SUPPORT OF TLS IN THE OPENFLOW ECO-SYSTEM

Controllers		Switches	
Ryu [13]	✓	Open vSwitch [3]	✓
OpenDaylight [9]	✓	HP [6]	✓
libfluid [7]	✓	PICA8 [12]	✓
Floodlight [5]	✓	NEC [11]	✓
Onos [8]	✗	Cisco [2]	✓
OpenIRIS [10]	✗	Dell [4]	✗

Table 1: TLS support of OpenFlow Controllers and Switches

Table 1 shows the state of TLS support for common SDN controller platforms and SDN switch vendors in August 2015. Apart from Dell, every investigated switch vendor supports TLS with OpenFlow. For the switches, TLS is implemented using the vendor specific switch OS i.e., TLS support is not model dependent thus applies to all switches of a vendor in general. Some vendors also enable the installation of third party OS, which could make TLS possible in this case. The controller side support is worse: only four of six controllers support TLS.

For the deployment of OpenFlow, not only operation tools and equipment but also testing tools are important. *Cbench* [1] and *OFlops* [23] were developed for doing performance tests with OpenFlow switches and controllers. Although encryption influences the performance of the switches and controllers, no TLS support was added yet. As there is no tool available, that enables TLS, we implemented a test controller based on libfluid [7] to study the influence of encryption to the control plane performance.

3. COST OF SECURITY IN SDN: DELAY

We are verifying the effects of encryption to the OpenFlow performance with an experiment using different hardware and software switches: An NEC PF5240, a Pica 8 P3290, a Pica 8 P3297 and the software switch Open vSwitch.

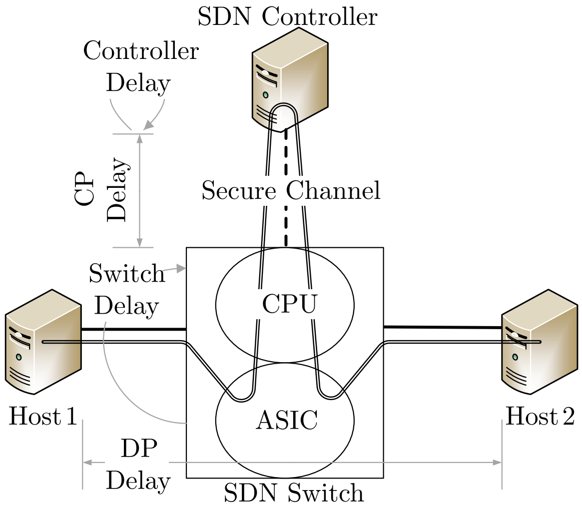

Figure 1: Packet path with relaying controller

We investigated the packet-in delay that occurs for the first packet of a flow in reactively managed networks. The measurement setup is shown in Figure 1. First we measured the delay of packets with matching flows (DP) and the delay from switch to controller (CP) separately. In the measurements our controller acts as a relay. On a packet-in the controller replies with an appropriate packet-out message but no forwarding rule is inserted. The controller delay was directly measured at the controller machine NIC. At Host1 we measured the round trip-time of packets using this setup. The additional switch processing delay for the first packet of a flow is then determined out of the DP, CP and controller delays subtracted from the end to end delay.

We did independent measurements for TLS and TCP for the different switches, the results are shown in Figure 2. We conducted 1000 runs for each measurement to get meaningful results. This leads to confidence intervals <0.05 ms of all measured latencies. As can be seen the switch adds by far the dominant part to the complete latency. Both PICA8 Switches run PICOS, however the P3297 has a more powerful CPU than the P3290, therefore latencies are smaller in general. Specifically the TLS overhead for the P3297 is much smaller as it has hardware acceleration for encryption of the P3297's CPU. The results of Open vSwitch supports this observation, as also low latencies and low overhead were measured and the Intel CPU is more powerful than the ones of the switches. In contrast to that, the packet-in delays of the NEC and the P3290, without hardware acceleration, differ significantly if encryption is used or not. The delays of the controller differs, as for some switches the payload of the respective packet is not sent to the controller along with the packet-in message.

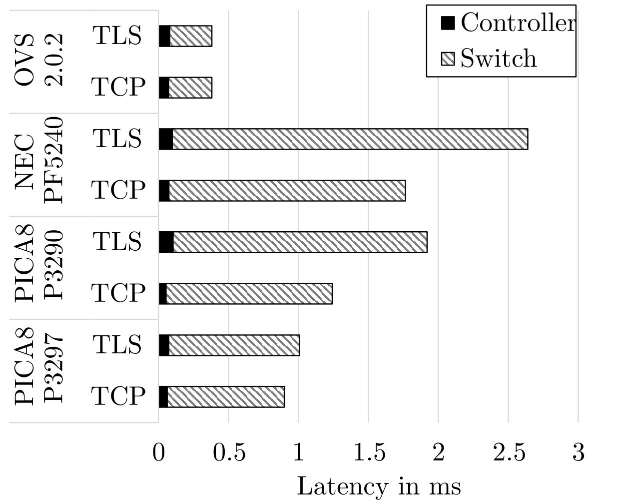

Figure 2: Latency added to the first packet by switches and controller using TCP and TLS

4. CONCLUSIONS AND FUTURE WORK

In this work we have had a look on the current state of TLS support in OpenFlow. In particular, we have shown the impact of TLS encryption to the packet-in delays of SDN switches. For a deployment of SDN in productive networks encryption is inevitable, but currently especially the control plane and testing software is lacking broad support. The evaluation of the delay measurements indicate the importance of CPU power for a good OpenFlow control plane performance of a switch. We have shown that the software solution Open vSwitch adds the lowest packet-in delay, this could give implications to future network architecture designs. Additionally it is shown that hardware acceleration support for encryption should be added to future OpenFlow switches. Currently the OpenFlow testing tools do not support TLS, although our results show that encryption may have a noticeable impact on control plane performance. In the future, we plan to further investigate performance metrics of OpenFlow implementations on different switches with and without TLS. To mention are for example flow-setup rate and flow-mod delay. Hence the development of OpenFlow tools using TLS is necessary. Additionally, the performance overhead regarding TLS for control plane software such as controllers is planned to be investigated in the future.

5. ACKNOWLEDGMENTS

This work was partially funded by the Federal Ministry of Education and Research Germany (BMBF) under grant number 16KIS0260. The authors alone are responsible for the content of the paper.

6. REFERENCES

[1] cbench - Controller Benchmarker. https://floodlight.atlassian.net/wiki/display/floodlightcontroller/Cbench.

[2] Cisco Plug-in for OpenFlow Commands. http://www.cisco.com/c/en/us/td/docs/switches/datacenter/sdn/command/openflow-cr-bookmap/cr-openflow.html.

[3] Configuring Open vSwitch for SSL. http://git.openvswitch.org/cgi-bin/gitweb.cgi?p=openvswitch;a=blob_plain;f=INSTALL.SSL;hb=HEAD.

[4] Dell OpenFlow Deployment and User Guide. http://topics-cdn.dell.com/pdf/force10-sw-defined-ntw_DeploymentGuide3_en-us.pdf.

[5] Floodlight. http://www.projectfloodlight.org/.

[6] HP OpenFlow Protocol Overview. http://h17007.www1.hp.com/docs/networking/solutions/sdn/devcenter/03_-_HP_OpenFlow_Technical_Overview_TSG_v1_2013-10-01.pdf.

[7] libfluid. http://opennetworkingfoundation.github.io/libfluid/.

[8] Onos. http://onosproject.org/.

[9] OpenDaylight. https://www.opendaylight.org/.

[10] OpenIris. http://openiris.etri.re.kr/.

[11] PF5240 – ProgrammableFlow Switch. http://www.nec.com/en/global/prod/pflow/images_documents/ProgrammableFlow_Switch_PF5240.pdf.

[12] Pica8 configuration guide. http://www.pica8.com/document/v2.4/pdf/ovs-configuration-guide.pdf.

[13] Ryu SDN control Framework. http://osrg.github.io/ryu/.

[14] J. J. M. Dover. A denial of service attack against the Open Floodlight SDN controller. (December 2013), 2013.

[15] K. He, J. Khalid, S. Das, A. Gember-Jacobson, C. Prakash, A. Akella, L. E. Li, and M. Thottan. Latency in Software Defined Networks. In *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems - SIGMETRICS '15*, pages 435–436, New York, New York, USA, 2015. ACM Press.

[16] K. He, J. Khalid, A. Gember-Jacobson, S. Das, C. Prakash, A. Akella, L. E. Li, and M. Thottan. Measuring control plane latency in SDN-enabled switches. In *Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research - SOSR '15*, pages 1–6, New York, New York, USA, 2015. ACM Press.

[17] Y. Jarraya, T. Madi, and M. Debbabi. A Survey and a Layered Taxonomy of Software-Defined Networking. *IEEE Communications Surveys & Tutorials*, 16(1):1955–1980, 2014.

[18] R. Kandoi and M. Antikainen. Denial-of-service attacks in OpenFlow SDN networks. In *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 1322–1326. IEEE, May 2015.

[19] D. Kreutz, F. M. Ramos, and P. Verissimo. Towards secure and dependable software-defined networks. *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking - HotSDN '13*, pages 55–60, 2013.

[20] M. Kuźniar, M. Canini, and D. Kostić. OFTEN testing OpenFlow networks. *Proceedings - European Workshop on Software Defined Networks, EWSDN 2012*, pages 54–60, 2012.

[21] M. Kuźniar, P. Perešíni, and D. Kostić. What You Need to Know About SDN Flow Tables. In *Passive and Active Measurement Conference*, pages 347–359, New York, New York, USA, 2015.

[22] A. Lazaris, D. Tahara, X. Huang, E. Li, A. Voellmy, Y. R. Yang, and M. Yu. Tango. In *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies - CoNEXT '14*, pages 199–212, New York, New York, USA, 2014. ACM Press.

[23] C. Rotsos, N. Sarrar, S. Uhlig, R. Sherwood, and A. W. Moore. OFLOPS: An open framework for OpenFlow switch evaluation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7192 LNCS:85–95, 2012.

[24] S. Scott-Hayward, S. Natarajan, and S. Sezer. A Survey of Security in Software Defined Networks. *IEEE Communications Surveys & Tutorials*, (c):1–1, 2015.

[25] S. Scott-Hayward, G. O'Callaghan, and S. Sezer. Sdn Security: A Survey. In *2013 IEEE SDN for Future Networks and Services (SDN4FNS)*, pages 1–7. IEEE, Nov. 2013.

[26] A. Shalimov, D. Zuikov, D. Zimarina, V. Pashkov, and R. Smeliansky. Advanced study of SDN/OpenFlow controllers. *Proceedings of the 9th Central & Eastern European Software Engineering Conference in Russia on - CEE-SECR '13*, (OCTOBER), 2013.

[27] R. Sherwood, G. Gibb, K.-k. Yap, G. Appenzeller, M. Casado, N. Mckeown, and G. Parulkar. FlowVisor : A Network Virtualization Layer FlowVisor : A Network Virtualization Layer. 2009.