conditions the hardware switch starts using the software table. This behavior is unexpected since we chose the number of active connections below the hardware table capacity. The switch becomes significantly slower and frequently does not react to requests from the controller when using the software table. When the rerouting flows are not installed or incorrectly installed, traffic could either be sent permanently via the IDS (reduced performance) or directly to the service host (no attack detection). The results for the Selective Filtering show that the bypassing with simple rules can yield an increased performance compared to inline mode.

In summary, the bypassing algorithms show promising results. They can improve performance up to an extent where the introduction of the IDS has no impact on performance. Attack detection is within the margin of error to 100% under typical load for both algorithms. The algorithms profit from the use of the software switch.

Limitations: The main limitation of our measurement approach is that it counts only the number of detected attacks. Therefore, it is not yet possible to assert which attacks are detected and account for false-positives and false negatives. The approach can be extended to collect this type of information as well.

Also, our framework at the moment allows no direct tracking of what happens at the hardware switch when it becomes unresponsive in Scenario 2. Any assertion which flows get redirected and those that do not will require this functionality.

7 CONCLUSION

We introduced the problem that the performance of security devices is crucial to the success of cloud systems. After looking at related work and the technical background, we presented three algorithms - two dynamic ones, (1) Adaptive Blacklisting and (2) Adaptive Whitelisting, and a static one, (3) Selective Filtering, to improve the performance of network intrusion detection systems by selectively bypassing them. We evaluated these approaches using four scenarios realized in a testbed environment. The results show that our approch improves the performance using bypassing while upholding a high level of detection accuracy. The dynamic algorithms have severe problems when using a hardware switch. Performance, as well as detection accuracy, drops. The static approach behaves similarly for software and hardware switches and gives a decent increase in performance. Thus, this work confirms the potential of bypassing algorithms to improve intrusion detection performance.

In future work, we will extend our testbed environment to support 1:1 accounting for the attacks carried out. This addition will allow the automatic detection of false positives and false negatives, as well as duplicate detection. Furthermore, we plan to track all network traffic. With this additional information, we will analyze the effect of the hardware switch in detail. Additional data also allows improving our algorithms further to ensure better detection ratios. Also we will investigate further IDS including IDS capable of parallel processing as well as various attack/signature combinations.

8 ACKNOWLEDGMENTS

This work was funded by the German Research Foundation (DFG) under grant No. (KO 3445/16-1).

REFERENCES

[1] 2018. Global Hybrid Cloud Market 2014-2021 | Statistic. (Oct. 2018). https://www.statista.com/statistics/609581/worldwide-hybrid-cloud-market-size [Online; accessed 30. Oct. 2018].

[2] Adeeb Alhomoud, Rashid Munir, Jules Pagna Disso, Irfan Awan, and A. Al-Dhelaan. 2011. Performance Evaluation Study of Intrusion Detection Systems. *Procedia Computer Science* 5 (2011), 173–180. https://doi.org/10.1016/j.procs.2011.07.024

[3] Firas B. Alomari and Daniel A. Menascé. 2013. Self-protecting and Self-optimizing Database Systems. In *Proceedings of the 2013 ACM Cloud and Autonomic Computing Conference on - CAC '13*. ACM Press. https://doi.org/10.1145/2494621.2494631

[4] Ayushi Chahal and Ritu Nagpal. 2016. Performance of Snort on Darpa Dataset and Different False Alert Reduction Techniques. In *3rd International Conference on Electrical, Electronics, Engineering Trends, Communication, Optimization and Sciences (EEECOS)*. https://pdfs.semanticscholar.org/9634/2f678949bcae35eabda3cfafeb0d0abe1d32.pdf

[5] Margaret Chiosi, Don Clarke, Peter Willis, Andy Reid, James Feger, Michael Bugenhagen, Waqar Khan, Michael Fargano, Dr. Chunfeng Cui, Dr. Hui Deng, Javier Benitez, Uwe Micheel, Herbert Damker, Kenichi Ogaki, Tetsuro Matsuzaki, Masaki Fukui, Katsuhiro Shimano, Dominique Delisle, Quentin Loudier, Christos Kolias, Ivano Guardini, Elena Demaria, Roberto Minerva, Antonio Manzalini, Diego Lopez, Francisco Javier Ramon Salguero, Frank Ruhl, and Prodip Sen. 2012. Network Functions Virtualization (NFV), An Introduction, Benefits, Enablers, Challenges & Call for Action. SDN and OpenFlow World Congress, Darmstadt, Germany. (2012). http://portal.etsi.org/NFV/NFV_White_Paper.pdf

[6] David Day and Benjamin Burns. 2011. A Performance Analysis of Snort and Suricata Network Intrusion Detection and Prevention Engines. https://www.thinkmind.org/download.php?articleid=icds_2011_7_40_90007

[7] Michael Jarschel, Thomas Zinner, Tobias Hossfeld, Phuoc Tran-Gia, and Wolfgang Kellerer. 2014. Interfaces, Attributes, and Use Cases: A Compass for SDN. *IEEE Communications Magazine* 52, 6 (June 2014), 210–217. https://doi.org/10.1109/mcom.2014.6829966

[8] Joseph McKendrick. 2015. 2015 IOUG Data Integration For Cloud Survey. (May 2015). http://www.oracle.com/us/products/middleware/data-integration/ioug-di-for-cloud-survey-2596248.pdf Produced by Unisphere Research, a Division of Information Today, Inc.

[9] Weizhi Meng, Wenjuan Li, and Lam-For Kwok. 2014. Efm: Enhancing the Performance of Signature-based Network Intrusion Detection Systems Using Enhanced Filter Mechanism. *computers & security* 43 (2014), 189–204. https://doi.org/10.1016/j.cose.2014.02.006

[10] Aleksandar Milenkoski, Bernd Jaeger, Kapil Raina, Mason Harris, Saif Chaudhry, Sivadon Chasiri, Veronica David, and Wenmao Liu. 2016. Security Position Paper: Network Function Virtualization. (March 2016). https://cloudsecurityalliance.org/download/security-position-paper-network-function-virtualization/ Published by Cloud Security Alliance (CSA) - Virtualization Working Group.

[11] Open Networking Foundation. 2016. Impact of SDN and NFV on OSS/BSS - ONF Solution Brief. (1 March 2016). https://www.opennetworking.org/images/stories/downloads/sdn-resources/solution-briefs/sb-OSS-BSS.pdf

[12] Piotr Rygielski. 2017. *Flexible Modeling of Data Center Networks for Capacity Management*. Ph.D. Dissertation. University of Würzburg, Germany. https://opus.bibliothek.uni-wuerzburg.de/frontdoor/index/index/docId/14623

[13] Karen Scarfone and Peter Mell. 2007. *Guide to Intrusion Detection and Prevention Systems (IDPS)*. Technical Report. https://doi.org/10.6028/nist.sp.800-94 NIST Special Publication 900-94.

[14] Lambert Schaelicke, Thomas Slabach, Branden Moore, and Curt Freeland. 2003. Characterizing the Performance of Network Intrusion Detection Sensors. In *International Workshop on Recent Advances in Intrusion Detection*. Springer, 155–172. https://doi.org/10.1007/978-3-540-45248-5_9

[15] Holger Schulze. 2015. Cloud Security Spotlight Report. (2015). https://goo.gl/rMGh3x Presented by Information Security, LinkedIn Group Partner.

[16] Soumya Sen. 2006. Performance Characterization & Improvement of Snort As an IDS. *Bell Labs Report* (2006). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.720.2007&rep=rep1&type=pdf

[17] Wired Staff. 2018. The Average Webpage Is Now the Size of the Original Doom. *WIRED* (March 2018). https://www.wired.com/2016/04/average-webpage-now-size-original-doom

[18] Gina C Tjhai, Maria Papadaki, SM Furnell, and Nathan L Clarke. 2008. Investigating the Problem of Ids False Alarms: An Experimental Study Using Snort. In *IFIP International Information Security Conference*. Springer, 253–267. https://link.springer.com/content/pdf/10.1007%2F978-0-387-09699-5_17.pdf

[19] Giovanni Vigna, William Robertson, and Davide Balzarotti. 2004. Testing Network-based Intrusion Detection Signatures Using Mutant Exploits. In *Proceedings of the 11th ACM conference on Computer and communications security - CCS '04*. ACM, ACM Press, 21–30. https://doi.org/10.1145/1030083.1030088