Towards an Adaptive and Effective IDS Using OpenFlow

Sebastian Seeber, Gabi Dreo Rodosek

Towards an adaptive and effective IDS using OpenFlow

Sebastian Seeber and Gabi Dreo Rodosek

Universität der Bundeswehr München, Department of Computer Science
85577 Neubiberg, Germany
sebastian.seeber, gabi.dreo@unibw.de

Abstract. Processing huge amounts of traffic from core network components with respect to security remains a challenging task, since the amounts of data increase continuously. Therefore, new approaches need to be investigated to detect and handle attacks already in high-speed environments. In this PhD research, we will develop a new approach for detecting network attacks by processing data from core network components taking advantage of properties of OpenFlow in an SDN environment. Using this, we can collect metadata about forwarded traffic in an immediate and effective way. In addition, our solution will enable dynamic and adaptive redirection of traffic to various IDSs including cloud-based IDS solutions.

1 Introduction & Motivation

Security at network level states an important research area since consumers and companies push their data continuously into cloud environments [5]. A reason for this evolution is the growing popularity of cloud services as well as simplicity and fast dynamic expandability of resources on demand. In addition, distributed denial-of-service (DDoS) attacks increased dramatically during the last years [2]. A recent report from Akamai [1] shows 90 % growth of DDoS attacks in the last 12 month. The advent of software defined networking (SDN) promises a large variety of possibilities to improve network monitoring and traffic steering. Nevertheless, widespread deployments using SDN principles are still rare and mostly static (proactive) configured.

Network security solutions are primarily based on inspecting traffic. The most well-known approaches are on the one hand analyzing whole packets using deep packet inspection (DPI) and on the other hand using rather statistical data provided by NetFlow/IPFIX techniques [12]. The operational environment affects the chosen traffic inspection method. Environments carrying fewer amounts of data (average bandwidth from few Mb/s to 1 Gb/s) are suitable for DPI [8] whereas high-throughput (up to 100 Gb/s) environments like Internet Exchange Points (IXP) have only a chance inspecting traffic using NetFlow/IPFIX [17] and even sampled. A drawback of the latter approaches lays in their design, because statistics are calculated after a flow is terminated (by flow aging, TCP session termination or fixed interval). Therefore, these introduce a delay in a subsequent detection process.

In contrast, OpenFlow is able to raise an event or update a flow counter at arrival time of a packet depending on a match or mismatch with respect to an

existing or non-existing flow. Deployments of commercial existing intrusion detection systems are mostly implemented in a static manner. If multiple Intrusion Detection Systems (IDSs) exist, traffic redirection is mainly based on subnets or IP addresses.

Our proposed effective IDS uses OpenFlow as a key-enabler and is adaptive since it involves multiple IDSs on-demand by taking into account immediate detection results. Keeping these thoughts in mind, the following research questions arise and need to be investigated:

RQ1 *How to use OpenFlow to gather statistics to detect attacks/anomalies?*
RQ2 *How to find an optimal trade-off between performance and detection rate?*
RQ3 *How to classify attack and reason suitable IDS and their sequence?*
RQ4 *How to compare cloud security solutions and how to validate them?*
RQ5 *How to verify the chosen path (Service Chain Verification)?*

The remainder of the paper is organized as follows. Section 2 discusses related work. We introduce our approach in Section 3. Concluding remarks and future work are described in Section 4.

2 Related Work

The use of SDN concepts towards network security is not new. Kreutz et al. [14] argue for building dependable and secure SDN applications. Therefore, they identified and described current threat vectors in SDN environments that could be exploited and propose a general design to overcome these threats. Besides this Scott et al. [16] investigated possible new security issues introduced through SDN and identified affected layers. François et al. [11] reviewed SDN security approaches according to their scope, practicability and advantages.

Research also has been done focusing on more specific attacks and their mitigation. Using self organizing maps the authors of [9] propose a method to detect DDoS attacks based on flow analysis. Feamster et al. [10] and Schehlmann et al. [15] investigated possibilities to detect botnet traffic by using distributed monitoring approaches. Combining traditional network features (sFlow) and OpenFlow, Giotis et al. [13] proposed a mechanism to detect anomalies and mitigate attacks by modifying flow tables.

3 Approach

The research in this PhD presents a new approach in providing security on the network level. To overcome existing static security function deployments, our approach provides security functions in a dynamic and free composable manner. Traditional approaches follow a static sequence of different security functions based on parameters like state of protective system, anticipated attack probability or expected risk-level. Instead, the mechanism concatenates security functions (e.g., various specialized IDSs) based on immediate detection results. OpenFlow is a well-established protocol in SDN deployments and therefore the most suitable enabler for our approach.

Overall Architecture Our proposed overall architecture is depicted in Figure 1. Starting from left, the input traffic reaches a network core OpenFlow enabled switch (`OF-Switch`). It is important that this switch is as near as possible to an IXP to cope with large amounts of attack traffic. On top of this switch a light IDS (`OF-IDS`) is implemented using OpenFlow counters (*Meter-, Group-, Flow-Table* - OpenFlow 1.3) as input data. RQ1 and RQ2 are related to the development of an effective IDS based on monitoring data from an OpenFlow enabled switch.

Triggered by events of `OF-IDS` the connected SDN-Controller (`SDN-C`) processes the event provided by the IDS and information from the associated flow in order to decide about the subsequent detection step. RQ3 will answer this question by using traffic and attack characterization techniques to introduce adaptiveness. After a decision is taken by the `SDN-C`, OpenFlow rules are applied to the underlying `OF-Switch` meeting the decision requirements. The procedure between incoming traffic at the `OF-Switch` and `SDN-C` decision (including applying OpenFlow rules) will be called `detection cycle (DC)`. The described behavior repeats at every point an IDS (`IDS-A, IDS-B, IDS-C`) is present. Various DCs are conceivable, depending on the number of available IDSs and detection results. In cases where not only on-site IDSs, like Suricata [7], SNORT [6] or Bro [3] are considered, the architecture is flexible in including cloud-based IDS solutions (`Cloud-IDS`), e.g., Cloudflare [4]. For example, the red-dotted line shows a case where only on-site IDSs are involved, whereas the green path involves a cloud-based IDS. Therefore, RQ4 investigates techniques to evaluate cloud security solutions (cloud-based IDS) and comparison criteria. It's important to mention, that only attack traffic is redirected to cloud-based IDS solutions. Therefore, privacy aspects in using a cloud-based IDS are minimized compared to solutions that redirect the whole traffic to the cloud.

In the first step each `SDN-C` will decide autonomously, without including previous IDS results or decisions from earlier `SDN-C`s. Later a connection between them will be established to enhance detection capabilities and design the `SDN-C`s as a distributed controller.

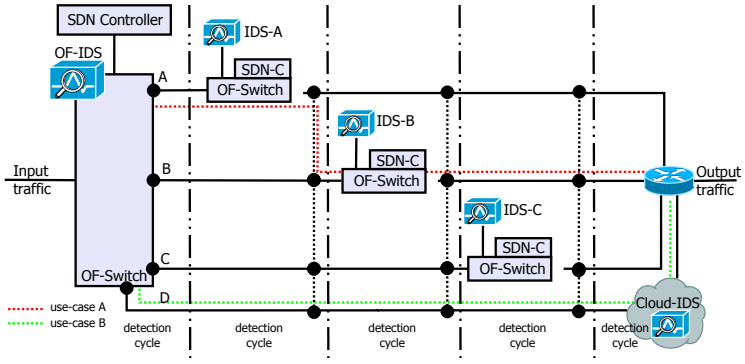

Fig. 1: Architecture Overview including Example Paths

Use-Cases Meeting requirements of various deployment scenarios enhances applicability of our proposed architecture and helps to improve the continuous development process. However, we will start implementing a very narrow-focused deployment and enhance it based on our findings and additional use-cases.

- **Software as a Service (SaaS)** products like web-hosting, e-mail or financial management provider are usually accessible from all over the world. In addition, these services seem to be easily configurable, also by people not familiar with security principles.
- **Platform as a Service (PaaS)** providers in most cases access to a preconfigured machine regardless of the underlying infrastructure (virtual/ physical). Users of PaaS require a rare understanding of configuring these systems and knowledge in securing them, but gain significant freedom in choosing applications running on top.
- **Company's Internet Access (CIA)** Companies providing services to customers and make Internet access available to employees are responsible for (amongst others) securing their network against attacks or attackers trying to gather customer data. Furthermore, they have to satisfy their customers by meeting SLAs as contracted (e.g., availability, response time).

Service Chaining Composing a sequence of successive security functions raises immediately concerns about (a) trustworthiness of each security function and (b) the right and entire processing of a composed chain. Therefore, our approach will utilize the benefits of SDN and distributed controllers to establish a verifiable chain of security functions. RQ5 is the core of this aspect and enables adaptability.

4 Concluding Remarks & Perspectives

Since the importance of security raises if more and more companies push their data and processing capabilities into cloud services, our approach tries to provide a solution to inspect traffic by various IDSs including cloud-based IDSs. Finally, it will redirect or block attack traffic at the first point of occurrence in the monitored network. Our proposed solution is also relevant since DDoS attacks are getting more and more popular.

As a next step, we need to investigate statistics derived from OpenFlow in order to detect attacks. At this point a trade-off between detection rate and performance of the OpenFlow device is essential to provide an effective solution. Furthermore, a mechanism needs to be investigated that decides about the next IDS in the detection process, to be adaptive. Later an evaluation needs to be done regarding the involvement of cloud-based IDSs. Therefore, we will investigate mechanisms to prove cloud-based IDSs performance and reliability. An interesting part of research is the verification of the path the traffic took. In that context, an investigation of service chaining and verification techniques needs to take place.

We envision, as extension of the work in this PhD, a thorough study of SLA restrictions and verification of legal requirements in introducing cloud-based IDS solutions needs to be investigated.

Acknowledgment

This work was partly funded by FLAMINGO, a Network of Excellence project (ICT-318488) supported by the European Commission under its Seventh Framework Programme.

References

1. Akamai - Q4 2014 State of the Internet – Security Report, http://www.stateoftheinternet.com/resources-web-security-2014-q4-internet-security-report.html, last accessed on 2015-01-28
2. Arbor Networks - Worldwide Infrastructure Security Report 2014, http://pages.arbornetworks.com/rs/arbor/images/WISR2014.pdf
3. Bro Network Security Monitor, http://www.bro.org/, last accessed on 2015-01-28
4. Cloudflare, Inc., https://www.cloudflare.com/, last accessed on 2015-01-28
5. Franklin Morris, Infographic: SMB Cloud Adoption Trends in 2014, http://www.pcworld.com/article/2685792/infographic-smb-cloud-adoption-trends-in-2014.html, last accessed on 2015-01-28
6. Snort, https://www.snort.org/, last accessed on 2015-01-28
7. Suricata IDS/IPS, http://www.http://suricata-ids.org//, last accessed on 2015-01-28
8. AbuHmed, T., Mohaisen, A., Nyang, D.: A survey on deep packet inspection for intrusion detection systems. arXiv preprint arXiv:0803.0037 (2008)
9. Braga, R., Mota, E., Passito, A.: Lightweight DDoS flooding attack detection using NOX/OpenFlow. In: Local Computer Networks (LCN), 2010 IEEE 35th Conference on. pp. 408–415. IEEE (2010)
10. Feamster, N.: Outsourcing home network security. In: Proceedings of the 2010 ACM SIGCOMM workshop on Home networks. pp. 37–42. ACM (2010)
11. François, J., Dolberg, L., Festor, O., Engel, T.: Network Security through Software Defined Networking: a Survey. In: IIT Real-Time Communications (RTC) Conference-Principles, Systems and Applications of IP Telecommunications (IPTComm). ACM
12. Fry, C., Nystrom, M.: Security Monitoring. " O'Reilly Media, Inc." (2009)
13. Giotis, K., Argyropoulos, C., Androulidakis, G., Kalogeras, D., Maglaris, V.: Combining OpenFlow and sFlow for an effective and scalable anomaly detection and mitigation mechanism on SDN environments. Computer Networks 62, 122–136 (2014)
14. Kreutz, D., Ramos, F., Verissimo, P.: Towards secure and dependable software-defined networks. In: Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking. pp. 55–60. ACM (2013)
15. Schehlmann, L., Baier, H.: COFFEE: a Concept based on OpenFlow to Filter and Erase Events of botnet activity at high-speed nodes. In: GI-Jahrestagung. pp. 2225–2239 (2013)
16. Scott-Hayward, S., O'Callaghan, G., Sezer, S.: SDN security: A survey. In: Future Networks and Services (SDN4FNS), 2013 IEEE SDN for. pp. 1–7. IEEE (2013)
17. Sperotto, A., Schaffrath, G., Sadre, R., Morariu, C., Pras, A., Stiller, B.: An Overview of IP Flow-Based Intrusion Detection. Communications Surveys Tutorials, IEEE 12(3), 343–356 (Third 2010)