Augmenting Monitoring Infrastructure For Dynamic Software-Defined Networks

1st Jaroslav Pešek
Dept. of Digital Design
FIT CTU in Prague
Prague, Czech Republic
jaroslav.pesek@fit.cvut.cz

2nd Richard Plný
Dept. of Digital Design
FIT CTU in Prague
Prague, Czech Republic
plnyrich@fit.cvut.cz

3rd Josef Koumar
Dept. of Digital Design
FIT CTU in Prague
Prague, Czech Republic
koumajos@fit.cvut.cz

4th Kamil Jeřábek
Dept. of Information Systems
FIT BUT
Brno, Czech Republic
ijerabek@fit.vutbr.cz

5th Tomáš Čejka
Dept. of Tools for Administration and Security
CESNET, a.l.e.
Prague, Czech Republic
cejkat@cesnet.cz

***Abstract*—Software-Defined Networking (SDN) and virtual environment rise new challenges for network monitoring tools. The dynamic and flexible nature of these network technologies require adaptation of monitoring infrastructure to overcome challenges of analysis and interpretability of the monitored network traffic. This paper describes a concept of automatic on-demand deployment of monitoring probes and correlation of network data with infrastructure state and configuration in time. Such approach to monitoring SDN & virtual networks is usable in several use cases such as IoT networks and anomaly detection, and it increases visibility into the complex and dynamic networks. Additionally, it can help with creation of well-annotated datasets that are essential for any further research.**

***Index Terms*—Network Monitoring, Software-Defined Networks, Internet of Things, Network Security**

I. Introduction

Network traffic monitoring has undergone a long evolution, and the technology to monitor network infrastructures is an essential building stone for every critical communication network. Many existing tools are optimized even for high-speed traffic processing, traditionally deployed on fixed core places in infrastructure, e.g., at uplink or perimeter.

However, modern networks are becoming more dynamic contrary to traditional physical infrastructures. With the rise of virtualization and containerized environments, the requirements of flexibility, configurability, and on-demand interconnections are accenting recent technologies like Software-Defined Networking (SDN), virtual switches, and software-based logical overlay networks.

Moreover, network infrastructures are getting more heterogeneous as they provide connectivity to various devices, such as mobile devices or the Internet of Things (IoT). For network security and operations reasons, it is essential to adapt monitoring tools to new trends of dynamic networks. However, several challenges prevent using of “out of the box” tools, such as:

1) devices migrations (roaming), NAT, limited life-cycle of containers and virtual machines can lead to IP address reuse, i.e., IP addresses cannot be used as a unique persistent identifiers without further information;
2) virtual network switches and virtual network infrastructures can be created on the fly and they can carry local traffic that is not visible at the perimeter (which is traditionally equipped with monitoring probes);
3) traditional way of monitoring system deployment is rather static and, usually, no additional monitoring probes are inserted into the running network;
4) based on the previous points, it is challenging to work with the collected data from monitoring system without any additional information about configuration and status of the virtual network in particular time window.

The listed challenges can be found in the existing works, such as paper [1] that discussed the limits of traditional monitoring tools in 5G and SDN environments. Also, many published works focus on SDN from a network traffic and security perspective, such as [2] (differences of DDoS attacks in SDN). However, we argue that the information from network traffic and virtual network configuration must be tightened and available together (when possible) during the analysis and interpretation. Therefore, this paper describes a concept that addresses these challenges by i) enhancing the open-source monitoring infrastructure by the information about the dynamic topology changes and network device logs in the virtual environment with virtualization and containers and ii) automation of on-demand deployment and releasing of monitoring probes.

This paper is organized as follows: Sec. II provides an overview of the most related works. Sec. III explains our approach to adapt the monitoring tools to SDN and virtual environments. Sec. IV discusses several domains benefiting from the proposed concept. Sec. V concludes the paper and lists potential future work.

II. Related Work

Software defined network (SDN) was described in [3] as “a structure for simplification and improvement of network management, which is highly flexible and scalable.” It is a network paradigm, which is meant to overcome problems and complexities of current network architectures [4]. The flexibility of this structure is achieved by splitting the control and data planes [3], [4]. Such plane separation makes policy enforcement and device configuration much easier. A central SDN controller controls state of the network and program devices (most commonly using OpenFlow protocol [5]).

The crucial part of network administrators’ jobs is to maintain smooth operation of their networks [6]. Network (security) monitoring tools are therefore necessary part of deployed network solutions. Anomaly detection has become crucial for detection of unknown attacks and intrusion attempts [7]–[9]. Unsupervised Machine Learning (ML) is typically used for this task, as described in [10].

Ghafir et al. [6] mention monitoring principles, such as Deep Packet Inspection (DPI) and flow-based monitoring. However, DPI requires unencrypted data and is not feasible in high-speed networks [11].

Flow-based network monitoring, as described in [12], utilizes so called flows. A flow is defined as “a set of IP packets passing an observation point in the network during a certain time interval, such that all packets belonging to a particular flow have a set of common properties” [12]. This type of monitoring typically processes only packet headers. This significantly lowers the needed amount of data processed. Moreover, packet content is not used so it can be applied on encrypted traffic as well. Such monitoring infrastructure better handles privacy and can be scaled more easily [12].

When using flow-based monitoring [12] in classic networks, the communicating devices are usually identified by IP addresses. However, virtual infrastructures may dynamically spawn virtual devices and destroy them in a short period of time. Therefore, it may not be clear which communication the captured flows represent. Network telemetry data, such as flows, is not always directly linked to the devices they originate from. This can lead to incorrect actions being taken when responding to a threat alert. Furthermore, virtualization may quickly change IP address assignments and routing which further deepens the problem.

Additionally, monitoring of IoT networks is also problematic. Santos et al. [13] state that IoT networks lack the interoperability and many of IoT devices are resource constrained and have low accessibility. This creates a vast heterogeneous landscape, which is very specific and therefore hard to monitor [13]. Therefore, more detailed monitoring approach might be needed for this scenario.

Microsoft proposed a solution for monitoring application and user activity of cloud networks in [14] — Network Watcher. This solution analyzes flow logs to visualize network activity, identify possible security threats and more [14]. However, Network Watcher is available for Azure virtual networks only. We aim to provide an open-source solution. Similarly, Amazon Virtual Private Cloud (VPC) provides monitoring based on flow logs [15], but only for the users of the VPC, which is not our interest. Furthermore, we aim to provide more detailed monitoring compared to papers [14], [15].

Ghazali et al. [16] proposed an enhancement of flow-based monitoring for VXLAN-based overlay networks. They extended this monitoring approach with an ID of the corresponding VXLAN. This differs from our main goal, which is correlation of IP flow records with logs of network devices. On the other hand, we can possibly apply work of Ghazali et al. [16] in our future work. VXLAN ID may be beneficial for network administrators, but it is currently out of the scope of this paper.

III. Proposed Solution

To address the challenges identified in previous section, we propose a solution that augments traditional network telemetry data with information about infrastructure status and modifications. Rather than solely capturing network telemetry data or flows, as addition we suggest capturing logs, configuration files, and data from orchestration and virtualization tools, such as Kubernetes and Proxmox, concerning virtual entities such as containers and virtual machines respectively. Stated approach will enable to conduct comprehensive analysis and associate them with the relevant devices.

A. Infrastructure

The proposed infrastructure incorporates a virtual edge router that is assigned to each virtual network within the hypervisor. This router serves as a gateway between the virtual network and outer networks, such as the Internet. Its primary responsibility is to manage the routing of incoming and outgoing traffic for the virtual network. In addition, the flow exporter is employed within a switch which interconnects router and machines. The schematic diagram of the designed infrastructure is illustrated in Fig. 1.

This type of IP flow data sources provides high-level information about communicating devices including overall volume of the transferred traffic. Furthermore, our approach uses more flexible open source tool *ipfixprobe*[1] that is capable of extending IP flow data with additional information and statistics. Thus, the collected flow data can be used in combination with advanced ML-based classifiers to process the traffic (as it is described in the following papers [17]–[19] and many more). This flow exporter is being automatically deployed based on system events in hypervisor and SDN infrastructure (system logs are being continuously inspected for this purpose).

B. Processing

The hypervisor produces logs and metadata, which are transmitted to a log collector offering valuable insight into the operations and performance of the virtual network, as well as

[1] https://www.github.com/CESNET/ipfixprobe

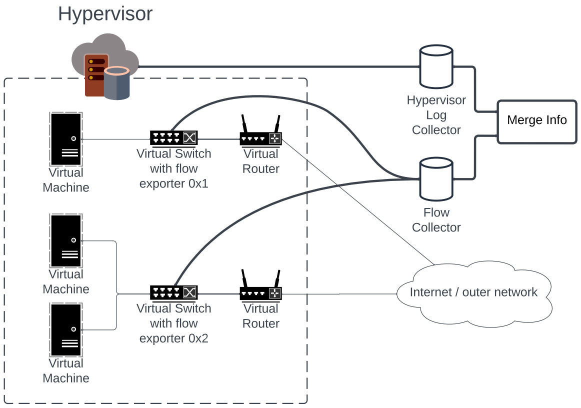

Fig. 1. SDN architecture with a hypervisor, containing a virtual network having an edge router connected to virtual machines via a switch. Exported flows are collected by a flow collector, while the hypervisor sends its logs and configuration changes to a log collector. These collectors are combined to allow holistic monitoring of the virtual network.

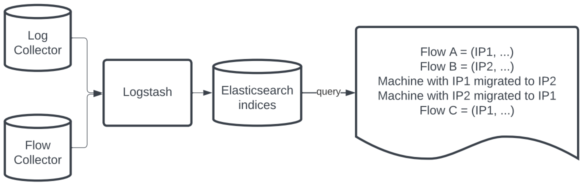

Fig. 2. Merged collectors through ELK stack. Migration of the servers would not break an awareness, even though addresses had changed. We comprehend that flow B and flow C track the same machine, even after their IP changes. Without log collector, there is no such information about current state of SDN configuration.

changes to its infrastructure. In parallel, the endpoint for flow exporters is flow collector.

The fusion of information from the log and flow collectors enables a comprehensive monitoring of the virtual network, providing a detailed overview of network operations and boosting the detection and explainability of issues with greater accuracy and efficacy.

For a proof of concept, we employed the ELK stack[2] which includes Elasticsearch for indexing collected data, Logstash as the input interface for processing and Kibana for search and visualisation. This integration allows for a stream processing. The ELK workflow diagram is illustrated in Fig. 2.

IV. Use Cases

Our approach is suitable for monitoring the network traffic of SDNs. However, we propose several other use cases since the proposed solution is general enough to apply in other areas as well.

A. Monitoring of IoT

As we mentioned in Sec. II, IoT networks need to be monitored in more detailed way. The architecture described in Sec. III can be applied to the IoT use case to provide a comprehensive solution for monitoring and managing IoT devices and networks. IoT devices, such as sensors, cameras, and actuators, can be connected to the virtual network via the virtual switch or so-called IoT gateway, providing network connectivity and enabling communication with other devices and the Internet.

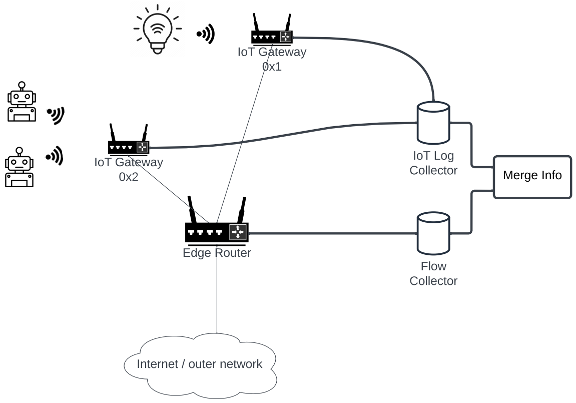

Fig. 3. Purposed schema is applicable to IoT network as well. Instead of hypervisor there is a IoT gateway which sends logs to collector. Flows are collected by edge router connected to outer network (usually internet).

The edge virtual router with flow exporter acts as the gateway for devices in network, routing traffic in and out of the network and sending flow records to the flow collector. IoT gateway then replaces hypervisor and provides logs and infrastructure metadata. The architecture for IoT use case is described in Fig. 3.

B. Anomaly Detection

As mentioned earlier, anomaly detection is a crucial part of detection of previously unknown network attacks. However, many methods utilize unsupervised ML, which usually leads to a high false-positive rate. Our monitoring infrastructure qualifies using hypervisor logs to estimate the behavior of the communication of the VM. Usability determines how the device should behave from the type of VM taken from hypervisor logs, i.e., database server, web server, etc. That means anomaly detection using unsupervised ML can be improved by the estimation of a cluster, which may decrease the false-positive rate.

C. Annotation of Network Traffic

The annotated datasets are critical for classifying network traffic and detecting security threats. Nevertheless, a suitable infrastructure is required. Our presented monitoring infrastructure for SDN can utilize hypervisor logs with a combination of VMs logs and annotate the network traffic of VMs. This approach allows for the creation of well-annotated network IDS datasets, which can, for example, help train better ML models for more reliable detection.

[2] www.elastic.co/elastic-stack/

D. Visualization

Although technological progress has significantly enhanced the ability to analyze a significant part of network traffic automatically, human-in-the-loop often remains an integral part of the analysis process. Well-designed visualization can give the analyst a more comprehensive and intuitive understanding of the analyzed data, enabling them to recognize potential patterns that may not be immediately apparent from raw data. The suggested concept can help to identify the underlying relationships and structures within the network, allowing the analyst to understand better what is happening and make more informed decisions based on their insights.

V. Conclusion

In this paper, we addressed the challenges of network traffic monitoring in dynamic networks and IoT networks and proposed a concept to adapt monitoring tools to SDN and virtualized environments. Our approach includes augmentation of monitoring infrastructure with dynamic topology transitions, network device logs and releasing monitoring probes.

By fusing information about the dynamic topology modifications and network device logs, our monitoring system can provide more accurate and timely data for analysis and interpretation. Furthermore, automating the deployment and release of monitoring probes can improve the efficiency of the monitoring system.

Our proposed solution can benefit many domains, including network security, IoT, and mobile networks. With the increasing use of SDN and virtualized environments, the need for flexible and adaptable monitoring tools will continue to grow. Our solution satisfies this need and enables more effective network monitoring and management.

Future Work

The described work can be enhanced in future in several possible ways as follows.

1) The proposed solution will be deployed in a real-world environment to obtain more realistic overview about efficiency and performance in dynamic networks. This evaluation includes various scenarios and use cases, with emphasis on scalability and robustness, in long-term period.
2) ML techniques for VM profiling and anomaly detection can be integrated with the prototype.
3) A more comprehensive evaluation can help to retrieve more detailed and accurate performance metrics for the presented augmented monitoring infrastructure and its capabilities.
4) With the increasing adoption of new network architectures such as cloud computing or fog computing, it is valuable to explore an adaptation of our solution to these new architectures deeply.

Acknowledgment

This research was partially funded by the Ministry of Interior of the Czech Republic in project "Flow-Based Encrypted Traffic Analysis" (VJ02010024), by the Ministry of Education, Youth and sports of the Czech Republic in project "e-Infrastructure CZ" (LM2023054), and by the Brno University of Technology grant in project "Smart information technology for a resilient society" (FIT-S-23-8209).

References

[1] P.-W. Tsai *et al.*, "Network monitoring in software-defined networking: A review," *IEEE Systems Journal*, vol. 12, no. 4, pp. 3958–3969, 2018.

[2] Y. Maleh, Y. Qasmaoui, K. El Gholami, Y. Sadqi, and S. Mounir, "A comprehensive survey on sdn security: threats, mitigations, and future directions," *Journal of Reliable Intelligent Environments*, pp. 1–39, 2022.

[3] R. Masoudi and A. Ghaffari, "Software defined networks: A survey," *Journal of Network and Computer Applications*, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1084804516300297

[4] D. Kreutz *et al.*, "Software-Defined Networking: A Comprehensive Survey," *Proceedings of the IEEE*, vol. 103, no. 1, 2015.

[5] N. McKeown *et al.*, "Openflow: Enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, 2008. [Online]. Available: https://doi.org/10.1145/1355734.1355746

[6] I. Ghafir *et al.*, "A Survey on Network Security Monitoring Systems," in *2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*, 2016.

[7] D. Liu *et al.*, "Network traffic anomaly detection using clustering techniques and performance comparison," in *CCECE*, 2013.

[8] M. Ahmed *et al.*, "A survey of network anomaly detection techniques," *Journal of Network and Computer Applications*, vol. 60, 2016.

[9] G. Pu *et al.*, "A hybrid unsupervised clustering-based anomaly detection method," *Tsinghua Science and Technology*, vol. 26, no. 2, 2021.

[10] I. Syarif, A. Prugel-Bennett, and G. Wills, "Unsupervised clustering approach for network anomaly detection," in *Networked Digital Technologies: 4th International Conference, NDT 2012, Dubai, UAE, April 24-26, 2012. Proceedings, Part I 4*. Springer, 2012.

[11] P. Piskac and J. Novotny, "Using of time characteristics in data flow for traffic classification," in *Managing the Dynamics of Networks and Services: 5th International Conference on Autonomous Infrastructure, Management, and Security, AIMS 2011, Nancy, France, June 13-17, 2011. Proceedings 5*. Springer, 2011.

[12] R. Hofstede *et al.*, "Flow monitoring explained: From packet capture to data analysis with netflow and ipfix," *IEEE Communications Surveys & Tutorials*, 2014.

[13] L. Santos *et al.*, "Flow Monitoring System for IoT Networks," in *New Knowledge in Information Systems and Technologies*. Cham: Springer International Publishing, 2019.

[14] Microsoft, "Azure network watcher traffic analytics," Feb. 2023. [Online]. Available: https://learn.microsoft.com/en-gb/azure/network-watcher/traffic-analytics

[15] Amazon Virtual Private Cloud User Guide, "Logging ip traffic using vpc flow logs," 2023. [Online]. Available: https://docs.aws.amazon.com/vpc/latest/userguide/flow-logs.html

[16] O. Ghazali and S. Khurram, "Enhanced IPFIX flow monitoring for VXLAN based cloud overlay networks," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, 12 2019.

[17] R. Plný *et al.*, "DeCrypto: Finding Cryptocurrency Miners on ISP Networks," in *Secure IT Systems: 27th Nordic Conference, NordSec 2022*. Springer, 2023.

[18] J. Luxemburk and T. Čejka, "Fine-grained TLS services classification with reject option," *Computer Networks*, vol. 220, 2023.

[19] Z. Tropková *et al.*, "Novel HTTPS classifier driven by packet bursts, flows, and machine learning," in *17th International Conference on Network and Service Management (CNSM)*. IEEE, 2021.