Multi-layer Onion-ring Visualization of Distributed Clusters for SmartX MultiView Visibility and Security

Jun-Sik Shin* Muhammad Usman† JongWon Kim‡

School of Electrical Engineering and Computer Science

Gwangju Institute of Science and Technology, Republic of Korea

ABSTRACT

To smoothly operate a cloud testbed-style playground of SDN-enabled distributed clusters, we need a promising solution to collectively monitor distributed clusters in a unified manner that can embrace multiple layers of physical and virtualized resources and inter-connecting flows. Also, it is required to detect various security attacks that exploit the widened attack surface due to virtualization and containerization. Thus, by leveraging multi-layer visibility framework denoted as SmartX MultiView Visibility Framework (MVF) [1], we are currently establishing a security extension of SmartX MVF. However, current tree-style visualization associated with SmartX MVF is not intuitive enough to accommodate the complicated dependencies for multi-level security extension. Therefore, in this paper, an on-going realization of multi-layer onion-ring visualization is introduced that can support multi-level visibility points for collecting visibility data from multiple layers.

Keywords: Unified monitoring, multi-layer visibility, measurement and tracing, and security analysis with visibility data.

1 INTRODUCTION

With OF@TEIN Playground [2], we have been operating several testbed-style playgrounds that allow operators and developers to conduct customized testing environment for SDN and cloud research. To smoothly operate a cloud testbed-style playground of SDN-enabled distributed clusters, we need a solution to collectively monitor distributed clusters in a unified manner that can embrace multiple layers of physical and virtualized resources and inter-connecting flows. This kind of unified monitoring (e.g., multi-layer visibility) is promising in reducing service outages and operating costs. With it, we can facilitate the performance and availability of heterogeneous physical and virtual resources, inter-connected by legacy and SDN-enabled networks. Also, it is required to detect various security attacks that exploit the widened attack surface due to virtualization and containerization. Thus, in this paper, by leveraging multi-layer visibility framework denoted as SmartX MultiView Visibility Framework (MVF) [1], we are currently establishing a security extension of SmartX MVF.

By supporting multiple security levels over SmartX MVF multi-layered visibility effort [1], we propose multi-level security extension to ultimately detect hidden vulnerable points. Current tree-style visualization associated with SmartX MVF is not intuitive enough to accommodate the complicated dependencies for multi-level security extension. It is inefficient in showing both multi-layer and multi-level visibility together on a single-view visualization. Therefore, in this paper, an on-going realization of multi-layer onion-ring visualization is introduced that can support multi-level visibility points for collecting visibility data from multiple layers.

2 CONCEPT DESIGN

2.1 Security Extension of SmartX MVF

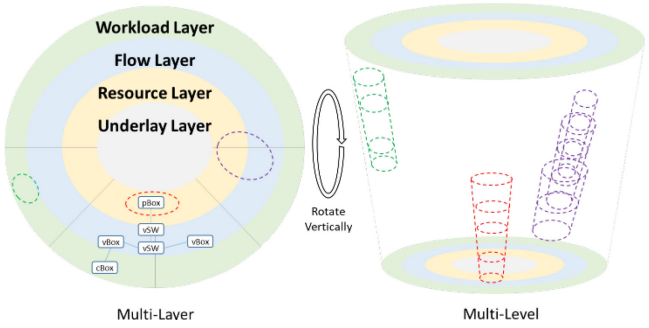

Figure 1: SmartX MVF onion-ring visualization with multi-level visibility points.

Distributed clusters typically consist of clustered SmartX Boxes, which basically include multiple levels of hardware and software components (mostly due to hierarchical interaction among computing, storage, and networking components). So, for improved visibility and security operation, it would be nice to have multi-level-aware visibility points. With this, later, we can associate the security level to the multi-level visibility points. To enable this kind of multi-level visibility points systematically, as explained earlier, we leverage SmartX MVF by aligning multi-level visibility points with multiple SmartX MVF layers for underlay, resource (physical and virtualized), flow, and workload.

2.2 Onion-ring Visualization

Currently, SmartX MVF provides an interactive visualization dashboard for playground topology, resource status, and limited flow information. The topology visualization is in a tree-style view focusing on

* e-mail: jsshin@nm.gist.ac.kr
† e-mail: usman@nm.gist.ac.kr
‡ e-mail: jongwon@gist.ac.kr, corresponding author

the inter-connections among distributed physical and virtualized resources. For example, the vertices of the tree represent physical/virtualized boxes (denoted as pBox and vBox) and virtual switches. And, distributed branches from tree root show network topology inside a box, respectively. Thus, we can easily understand the inter-connection among distributed clusters.

However, visualizing multi-layer and multi-level in a flat tree style is quite difficult. Moreover, the increased adoption of virtualization and containerization makes the playground much more complicated than before. For example, containerized entities (e.g., cBoxes for machine container instances) may be placed inside vBoxes, which are again placed inside pBoxes. Also, there exist several overlay-oriented inter-connection networking among them, even across multiple sites. Furthermore, to-be-visualized multi-levels visibility points for visibility and security operation will be an additional difficulty.

To address the above difficulties, we focus on onion-ring-style visualization as shown in Figure 1. Tiered rings on the surface partially cover the multiple layer visualization requirements of SmartX MVF. Also, the color-coded areas, separated with lines correspond to distributed sites (i.e., clusters) that host physical, virtualized, or containerized entities (e.g., boxes, switches, and microservices functions). Dashed circles on the surface may highlight inspection areas where multi-level security is being applied. If we rotate the onion-ring, the viewpoint of the dashboard will change to multi-level focused visualization.

Finally, SmartX MVF defines two types of visibility points. Passive measurement points cover resource statistics and packet collection. Active tracing points are used to assist the persistent and secured operation by even generating intentional probing packets. Thus, it is important to realize practical and light-weight tools to support the targeted visibility points for SmartX MVF.

3 Implementation Progress

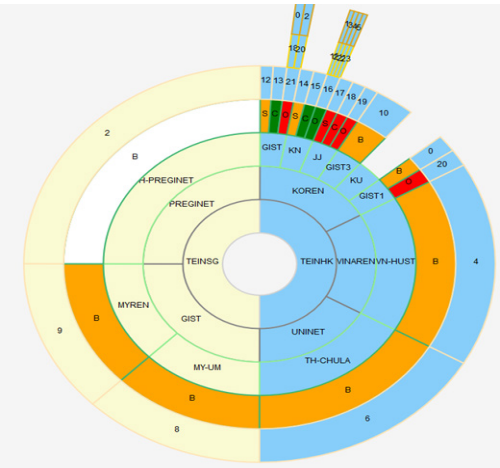

Figure 2: Onion-ring visualization for OF@TEIN+ Playground.

In this section, we explain the on-going implementation of onion-ring visualization with multi-level visibility points, by targeting OF@TEIN+ Playground (an international multi-site SDN-Cloud testbed [2]).

Figure 2 shows current implementation of onion-ring visualization. The dashboard is effective in showing underlay-layer, resource-layer, and flow-layer visualization together in a single unified view. At the underlay layer, from the center to outside, we can see international network PoPs (e.g., TEINSG, TEINHK), national PoPs (e.g., VINAREN, UNINET, …) and sites (e.g., VN-HUST, TH-CHULA, …). On top of distributed sites, SmartX (cluster) boxes are shown with their type (C, S, B) as well as their IDs. Some physical boxes have a number of virtual boxes within them, respectively. For example, SmartX Type-O box in KN site has five vBoxes and SmartX Type-O box in GIST site have 3 vBoxes. With the associated box IDs, we can easily visualize resource usage graphs in the boxes. Thus, with the support of visualization dashboard, the operators and developers of testbed playground can easily grasp the overlay-based playground topology and check its relationship with the underlay WAN networks that belong to separate network operators.

We implemented four types of measurement points and one type of tracing point. Measurement points can collect the visibility data for IP packets, VXLAN packets, pBox computing resource status, Libvirtd-based vBox computing resource status. Especially, the measurement-style visibility point allows us to capture all packets with desired protocol matches in a light-weight manner by leveraging eBPF (extended Berkeley Packet Filter) [3]. Also, PerfSONAR monitoring package is adopted for tracing by placing it as a Docker container into the target box. With the PerfSonar-based tracing point, we can check end-to-end networking bandwidth and latency.

4 Future Works

The proposed visualization is still facing several challenges to solve. First, it is required to diversify the types of visibility points to satisfy the requirements of multi-level visibility points. Also, the implementation for 3D-enhanced onion-ring visualization is needed by addressing the multi-layer and multi-level visibility implications via iterative refinements, starting from a single site to distributed multiple sites.

Acknowledgments

This work was supported by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2017-0-00421, Cyber Security Defense Cycle Mechanism for New Security Threats and No. 2015-0-00575, Global SDN/NFV Open-Source Software Core Module/Function Development).

References

[1] M. Usman, A. C. Risdianto, J. Han, M. Kang, and J. Kim, "SmartX MultiView visibility framework leveraging open-source software for SDN-Cloud playground," *in Proc. NetSoft 2017*, IEEE, July 2017.

[2] A. C. Risdianto, et al, "OF@TEIN: A community effort towards open/shared SDN-Cloud virtual playground," *in Proc. APAN-NRW 2015*, Asia Pacific Advanced Network, Aug. 2015.

[3] Jay Schulist, Daniel Borkmann, and Alexei Starovoitov, "Linux Socket Filtering aka Berkeley Packet Filter (BPF)" [online] available at https://www.kernel.org/doc/Documentation/networking/filter.txt