Piggybacking Network Functions on SDN with P4 support

Chang Liu*, Arun Raghuramu*, Chen-Nee Chuah*, and Balachander Krishnamurthy**

*University of California, Davis; **AT&T Labs-Research

Previous study [3] proposed the idea of piggybacking security checks on the first few packets sent to the controller via reactive routing. However, it can be challenging to implement it in practice due to scalability issues. Emerging switches that are programmable using languages like P4 now make it possible to identify initial flow packets in the data plane, at line rate. In this work, we propose to utilize the P4-enabled programmable switches and OVS to instantiate a SDN-based network security architecture, which leverages the benefit of inspecting first few packets of active flows, and SDN's dynamic control over the data plane for immediate mitigation. This architecture can also be extended to other applications, such as traffic classification and traffic dispersion graph generation.

1. INTRODUCTION

Software-defined networking (SDN) is widely considered as the networking architecture of choice for future networks. There are a lot of discussions and attempts to leverage SDN's appealing properties for security applications. In [3], we proposed the SDN-Defense framework, which piggybacks security checks on the first few packets sent to the controller. This feasibility study based on campus WiFi traffic showed that up to 73% of malicious flows can be detected by inspecting just the first three packets of a flow, and 90% of malicious flows from the first four packets.

In [3], SDN-Defense is based on reactive routing. It delays installation of forwarding rules until the Kth packets of each active flow is sent to the controller, where K is a design parameter tunable by the SDN controller. We note that, reactive routing is not widely deployed in the networks because 1) The controller becomes the bottleneck of the network under large traffic rate. 2) Additional end-to-end latency is introduced for packets going through the switch-controller-switch loop.

To resolve scalability issues, we propose to utilize a combination of P4 enabled programmable switches and OpenFlow switches to instantiate the SDN-Defense framework. This architecture leverages 1) the capabilities of P4-enabled switches to identify initial flow packets and send a copy of them to the controller, while forwarding packets on as normal; and 2) the OpenFlow interface between OF-enabled switches and the SDN controller for immediate attack mitigation. We note that the SDN-Defense framework is not limited to security applications, but also applicable to traffic classification and traffic dispersion graph (TDG) generation.

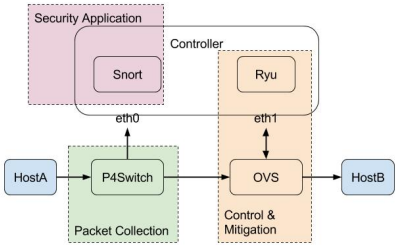

Figure 1: Architecture Overview.

2. DEMO ARCHITECTURE

Figure 1 depicts the architecture of the demo. HostA serves as an traffic generator and sends packets to HostB. When packets are processed in the P4Switch, they are forwarded on to the next hop as usual, and the first K packets of each flow are identified and a copy of them are sent to the controller (interface eth0). An instance of Snort sniffs packets on interface eth0 and sending alerts to the controller application via Unix Domain Socket[1]. The controller application is developed in Ryu [2]. When it receives an alert from Snort, it extracts the 5-tuple information about the malicious flow and installs a rule into OVS to drop this flow. Detailed information about the demo is available online [1].

3. REFERENCES

[1] Demo: `https://github.com/cchliu/SDN-Defense`.

[2] Ryu: `https://osrg.github.io/ryu/`.

[3] C. Liu, A. Raghuramu, C-N. Chuah, et al. Piggybacking network functions on sdn reactive routing: A feasibility study. In *ACM SOSR*, 2017.

[1]The security application can be co-located with the SDN controller or placed at a separate server. Its position is also a design parameter affecting the communication overhead with the controller application.