HARDWARE ACCELERATION OF NETWORK SECURITY AND MONITORING APPLICATIONS

Lukáš Kekely
Master Degree Programme (2), FIT BUT
E-mail: xkekel00@stud.fit.vutbr.cz

Supervised by: Jan Kořenek
E-mail: korenek@fit.vutbr.cz

Abstract: This paper deals with the design of SW controlled acceleration system for high-speed networks. The main goal is to provide easy access to acceleration for various network security and monitoring applications. The proposed system is designed for 100 Gbps networks. It enables high-speed processing on an FPGA card together with flexible SW control. The combination of hardware speed and software flexibility allows creation of complex HW accelerated network applications.

Keywords: FPGA, 100 Gbps, networking, security, monitoring, Software Defined Networking

1 INTRODUCTION

As the Internet is constantly evolving, the speed of network links rises and network devices need more processing power to deal with network traffic. In network core, the traffic rate doubles approximately every 18 months which can lead to the need for faster than 100 Gbps technologies very soon. As the transfer speed of the infrastructure rises, the performance of network monitoring and security cannot fall behind.

We propose flexible and robust hardware acceleration system that aids various monitoring and security applications to satisfy performance needs of 100 Gbps networks. The system provides functions for accelerated capturing and basic processing of network traffic together with easy and flexible software control. Software applications can utilize proposed system and specify details of advanced processing and control mechanism (intelligence). Therefore, the system can be used for wide range of different applications. The proposed system is based on previous results of the Liberouter project [1]. The contribution of this paper is the overall software and firmware design of the system, including the design and implementation of its several modules (e.g. packet parser or Cuckoo Hash packet filter).

Following chapters briefly describe the design of system firmware and software together with possible solutions to selected challenging parts of system regarding the employment in 100 Gbps networks.

2 SYSTEM DESIGN

Standard model which is widely used in 10 Gbps networks relies on HW card performing packet capture, usually with packet distribution among CPU cores. Captured traffic is then sent over the host bus to the memory, where packets are processed by CPU cores. This model cannot be applied to 100 Gbps networks due to two major performance bottlenecks. First, the throughput of today's PCI Express busses is insufficient. The second bottleneck lies in limited computational power, which is insufficient for monitoring or security tasks. We propose new acceleration model which overcomes mentioned bottlenecks by well-defined hardware/software co-design. Proposed model is inspired by the concepts of new promising network architecture called Software Defined Networking [3]. The main idea is to give to the hardware ability to handle basic traffic processing and leave only granular control of HW and more advanced tasks to software.

The proposed system was designed with respect to basic characteristics of monitoring and security tasks. Only small portion of packets has important information for network monitoring. Therefore, it is possible to extract interesting data from packets in HW and send just them to SW in predefined format, which we call Unified Header (UH). Then only few bytes are transferred through the PCI Express bus and also CPU has lower load because packet parsing is done in hardware. Furthermore, flow data (NetFlow or IPFIX) can be aggregated to records (Flow UH) directly in HW, which brings even higher performance savings. On the other hand, security and advanced monitoring applications perform deep packet inspection on suspicious traffic and have to analyze whole packets. For example, detection of attacks over SSH needs only SSH packets, extraction of URL from HTTP requests needs only several first packets of HTTP flows. Therefore the proposed system provides control over HW packet preprocessing at the flow level granularity.

First packets of new unknown flows are processed by SW, which can decide to offload their further processing into HW. The performance of the system strongly depends on the duration time of the decision. It means time between arrival of the first packet of the flow and activation of SW-defined rule in HW. Observed timing of packets inside flows in real network is shown in Fig. 1. The figure shows already received portion of packets over time passed since the first packet of the flow. You can see in graphs that decision with expected duration in order of milliseconds does not have high negative impact on system performance. With expected decision time the system is able to offload processing of more than 90 % of packets, leaving only one tenth of traffic for SW processing. Another possible problem could arise from limited capacity of HW flow classifier and flow cache. Therefore, our system should offload processing of only the heaviest flows. Fortunately, our measurements show that network traffic has heavy-tailed distribution of flow sizes. Fig. 2 shows portion of packets carried by specified percentage of the heaviest flows on network. It can be seen that 0.1 % of the heaviest flows carries over 50 % of all packets and 1 % carries even around 85 %.

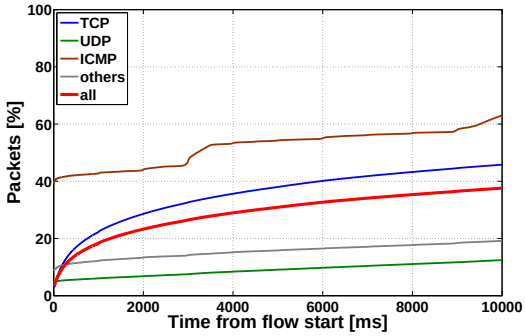

Figure 1: Already arrived portion of packets over time from flow start.

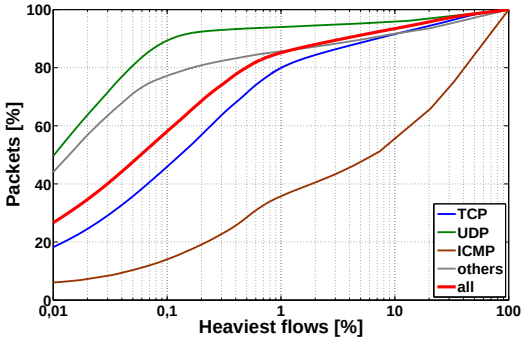

Figure 2: Portion of packets carried by the heaviest flows.

Top-level scheme of the proposed system is shown in Fig. 3. The processing of incoming packets starts with header parsing and extraction of interesting metadata. Metadata are passed to the classifier, which has to find a rule for processing of subsequent packets. The rule specifies the SW channel and how the packet is treated in firmware. It can be processed in a flow cache, trimmed, send to the host memory as a whole packet or as a unified header. Data in host memory are monitored by exporter which analyzes received data and export flow records to the collector. Advanced monitoring and security functionality can be added to system in form of plugins. These plugins can read data from selected channels and can also specify which types of traffic they want to inspect and which flows can be processed in hardware. For example, plugin for HTTP monitoring needs to inspect every packet in the HTTP flow until it acquires required information (e.g. URL). Specifications of interesting and uninteresting bulk traffic from plugins are passed to controller, which aggregates them into rules and configures firmware's behavior in order to achieve maximal HW acceleration.

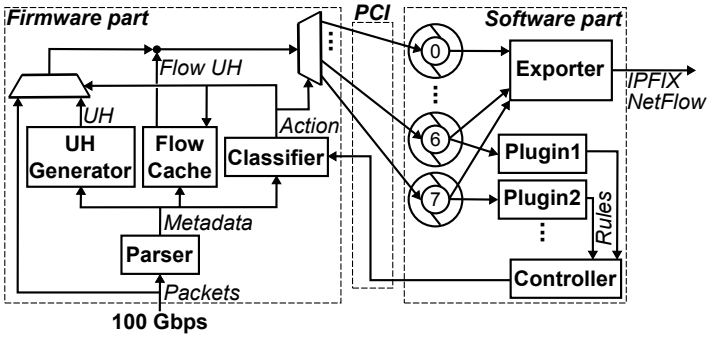

Figure 3: System architecture.

Although the top-level architecture of the system is straightforward and logical, we had to solve many challenging tasks to create a prototype of the system.

Hardware Platform. We are developing next version of hardware acceleration platform presented in [5]. The new platform is an FPGA card with NetCOPE firmware. The card has large Virtex-7 FPGA, PCI Express v3 interface, MoSys Bandwidth Engine and one 100 Gbps interface.

Packet Parsing. Our Low-Latency Modular Packet Header Parser [2] is able to achieve more than 400 Gbps throughput while maintaining processing latency under 40 ns. Parser design enables easy extension of protocol support and fine tuning of performance parameters (latency, area, throughput) to fit the needs of concrete application.

Packet Classification. We have designed three-step packet classification process. Classification starts with assignment of default class, continues through simple classification of IP prefixes and ports and is finished by precise flow classification. Simple classifier contains rather small constant rule set and is implemented using on-chip TCAM. Precise flow based classifier can be realized using hash table with Cuckoo Hashing [4]. Our VHDL implementation achieves mean hash table load factor of more than 60 % using 2 hash functions, over 80 % using 3 and around 95 % using 4.

3 CONCLUSION

The paper proposed universal and robust hardware acceleration support system for easy deployment of advanced monitoring and security applications in high-speed networks. Proposed system is also suitable for acceleration of application layer protocols analysis in 100 Gbps networks, which is challenging task even for 10 Gbps. The system is currently being implemented by the Liberouter team who has substantial experience with the design of network security and monitoring systems.

REFERENCES

[1] Cesnet TMC Group. Liberouter [online]. URL: https://www.liberouter.org

[2] Kekely, L., Puš, V., Kořenek, J. Low-Latency Modular Packet Header Parser for FPGA. In: ACM/IEEE Symposium on Architectures for Networking and Communications Systems. Austin, US: ACM, 2012. 77-78. ISBN 978-1-4503-1685-9.

[3] ONF Market Education Committee. Software-Defined Networking: The New Norm for Networks. ONF White Paper. Palo Alto, US: Open Networking Foundation, 2012.

[4] Pagh, R., Rodler, F.: Cuckoo Hashing. In: Algorithms – ESA. Heidelberg: Springer, 2001.

[5] Puš, V. Hardware Acceleration for Measurements in 100 Gb/s Networks. In: Proceedings of 6th International Conference on Autonomous Infrastructure, Management and Security (AIMS). Heidelberg: Springer, 2012. 46-49. ISBN 978-3-642-30632-7