DESIGN OF FLOW MONITORING PROBE

Martin Žádník
Master Degree Programme (2), FIT BUT
E-mail: xzadni00@stud.fit.vutbr.cz

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

ABSTRACT

Traffic monitoring based on IP flows provides essential information for variety of applications such as incident handling, attack detection, usage-based accounting, etc. This paper explores ways to implement the flow monitoring probe using ordinary PC with hardware acceleration card. To this end the paper discuss novel methods addressing shortcoming of current implementations as well as current standards of network measurements. The design of the probe for acceleration card is outlined. Estimated throughput and quality of the probe from the feature perspective is discussed in the conclusion.

1 INTRODUCTION

The Internet is based on IP protocol and is composed of many network domains which are more or less administrated by different entities. It is obvious that such an environment is not too reliable. Various types of attacks can cause denial of service, leakage of information or to increase interconnection latency. Moreover problems are also caused by complex network topologies where the router's precise configuration is of essence. Therefore there is a need for monitoring devices which are able to provide accurate data about spectrum of traffic mix, attacks, applications, etc. Such type of systems can help network operators to man- age current network or plan new network topology.
We can observe several complementary directions in network monitoring today:

- *Simple network monitoring* – is based on counters which provide data about the device utilization. These data are coarse-grained and fails to give greater details of the traffic mix.
- *Packet capturing* – is the opposite solution to the previously mentioned. Packets are sampled and captured for analysis in collector station.
- *Packet inspection* – is based on the parsing and interpretation of the packet content. As such it serves for virus and worms detection.
- *Flow-based monitoring* – aggregate information about flows (for definition of flow see [4]). It is able to provide crucial information not only about volume but also about spectrum of the traffic mix and about behavior of individual entities on the network.

Monitoring based on flows provides good abstraction and preserves a lot of valuable information at the same time. Therefore it is very popular and widely used in many forms (NetFlow, IPFIX). Nevertheless a lot of networks suffer from a lack of flow-capable devices. So far, flow data are mostly generated by IP routers. Unfortunately they are usually busy doing their own job (routing, switching, filtering, etc.). Therefore they often impose mandatory sampling to decrease the input bandwidth. The situation is even worse during attacks when they are unable to monitor at all. That significantly decreases value of statistical information they export.
An autonomous dedicated probe for flow monitoring has several advantages: no need to change flow-incapable routers, high speed of data processing, large flow cache, various enhancements to protect itself against malicious traffic (for example [1], [2]).
A robust solution for high-speed flow monitoring requires some kind of wire acceleration. As competitive platform was chosen an network acceleration card equipped with Field Programmable Gate Array (FPGA) and external memory. The card has a PCI interface which allows to plug it in the host PC.

2 DESIGN

The probe is divided into two parts – hardware and software. The hardware part is intended to process incoming packets at the wire speed. Whereas the software part post-processes the data tranfered from the card and exports them to the collector. The Figure 1 shows the simplified architecture of the whole probe together with estimated input transfer speeds. Following text describes individual units in greater detail.

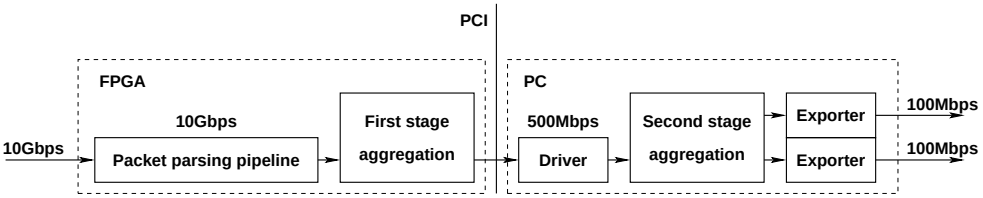

Figure 1: Block Structure

The first stage of probe in the FPGA decelerates incoming stream by aggregation packet information into flow-records. Before the flow-record is created or updated, incoming packet is processed at network layer L1, L2 and L3 to verify CRC, extract information about IP addresses, ports, protocol, length of the packet and other fields (complete list can be found in [4]). Fields that determines the flow (known as key-fields) are subject of the hash function. Its result is the address to the memory where is stored the context for this particular flow. Collisions caused by hash (two flows map to the same memory location) are handled in first aggregation stage where the old record is exported to the software and new record is created for the new packet. Simulations show that good hash function and sufficient memory capacity will keep the collision rate reasonably low.
Second stage implemented in software creates new or updates corresponding existing flow-record by the data acquired from the transfered flow-record from hardware. Again hash is used to address corresponing flow-record but this time collisions are handled by additional lookup in the list (see Fig. 2).
Finally when the flow meets one of the terminating criterion (see [4]) the flow-record is exported by standard flow protocol (NetFlow, IPFIX) to the collector where it is subject of further

analysis.

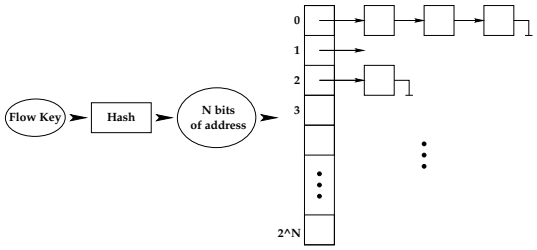

Figure 2: Lookup structure

3 CONCLUSION

In the beginning, the work provides a theoretical background about IP networks monitoring which is necessary for their management and analysis. The emphasis is given on flow-based monitoring and its potential applications.
The architecture will be implemented in VirtexIIPro FPGA utilizing the COMBO6X platform[6]. The acceleration card will be able to hold up to 256K of simultaneous flows in its memory. Whereas the secondary aggregation implemented in software is limited only by the capacity of host DRAM.
Supposing that the card will be able to aggregate incoming traffic in rate one to twenty (twenty packets belongs to one flow on average) then the bandwidth of the whole system will be suitable to monitor 10 Gbps interface.
Enhanced heuristics (adaptive sampling, sample and hold, filters, pre-aggregation) are already included in the detail design which is unfortunately beyond the scope of this paper.

REFERENCES

[1] Estan, C., Varghese, G. New Directions In Traffic Measurement: Focusing on the Elephants, Ignoring the Mice, San Diego, University of California. http://portal.acm.org/citation.cfm?doid=859716.859719, 2003

[2] Duffield, N., Lund, C.: Predicting resource usage and estimation accuracy in an ip flow measurement collection infrastructure. In *IMC '03: Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement*, pages 179–191. ACM Press, 2003.

[3] Kosnár, T.: Notes to Flow-Based Traffic Analysis System Design, CESNET, Prague. http://www.cesnet.cz/doc/techzpravy/2004/ftas-arch/, 2004

[4] Quittek, J., Zseby, T., Claise, B. and Zander, S.: Requirements for IP flow information export (IPFIX). RFC 3917, Internet Engineering Task Force, October 2004.

[5] Claise, B.: Cisco systems netflow services export version 9. RFC(Informational) 3954, Internet Engineering Task Force, 2004.

[6] Liberouter: Liberouter Project WWW Page. http://www.liberouter.org, 2006