



(Semi) Line Rate Traffic Classifier on Nvidia Switch

Abstract

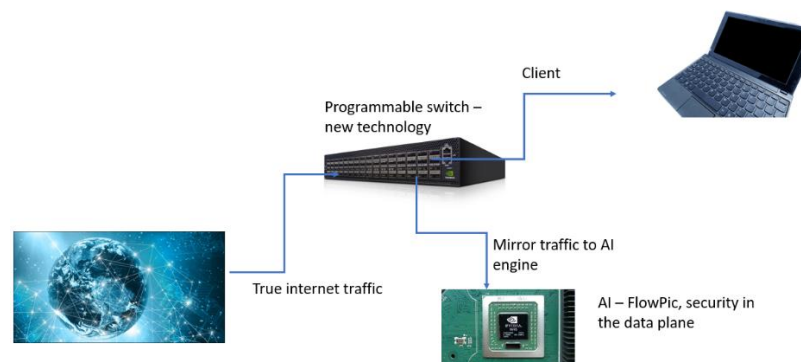
In this project we will classify network traffic at semi line rate with Nvidia programmable switch and Deep Learning methods. The students will use cutting-edge technology: data plane programming using P4. The main effort in this project will be to leverage NVIDIA technology and using an existing AI model to do semi line rate classification of encrypted traffic.

Traffic classification, the categorization of network traffic into appropriate classes, is important to many applications, such as quality of service (QoS) control, pricing, resource usage planning, malware detection, and intrusion detection. Because of its importance, many different approaches have been developed over years to accommodate the diverse and changing needs of different application scenarios. In particular, the growing trends of Internet traffic encryption and an increase Virtual Private Networks (VPNs) and The Onion Router (ToR) usage, raise additional challenges to network traffic classification.

Using Deep Learning (DL) methods, we were managed in earlier project to train a Convolutional Neural Network (CNN) model that can classify encrypted internet traffic. The model is trained with labeled FlowPic images build from the flow packets metadata (packet arrival time, packet length). This technology enables real-time traffic classification, once the model receives FlowPic image of a certain flow.

Programming Protocol-independent Packet Processor (P4) is a high-level language that can be deployed in the future into Software Defined Networks (SDN) and can actually serve as an alternative to OpenFlow that is currently used – due to its flexibility and ability program the data plane and support emerging new protocols.

In this project we will use Mellanox (Nvidia) SN3700 P4-capable Spectrum-2 based switch. The P4 programmed switch will send to the collector, for a certain flow, its packet metadata. The collector will build FlowPic image and send it to the trained model for a rapid flow classification.



Picture 1: Classifying Internet flows using P4 programmable switch and a FlowPic AI engine



objective

The purpose of this project is to implement a flow classifier at semi line rate using Nvidia switch and a Deep Learning classifier.

What will be done:

1. Ramp-up P4 using [1], [2], [3], [4].
2. Ramp-up on Mellanox architecture (see appendixes)
3. Install previous semester FlowPic project and verify the AI classifier correct operation (<https://gitlab.cs.technion.ac.il/lccn/s2020-deep-learning-based-traffic-classification-new>).
4. Install previous semester Mellanox P4 project and verify correct operation of the switch (<https://gitlab.cs.technion.ac.il/lccn/s2020-intelligent-traffic-debugger-using-p4>)
5. Implement the FlowPic classifier on pre-recorded data (PCAP) and live client-server Internet flows.

General requirements for all LCCN Projects are specified at the lab website:

<https://lccn.cs.technion.ac.il/lab-courses/>

Instructors:

NVIDIA: Mati Kadosh, Idan Barnea

Rafael: Aviel Glam

Technion: Eran Tavor

Prerequisites:

Introduction to Networking Course (236334)

Appendix A: Mellanox p4 target architecture



The current Mellanox p4 target architecture compress from 5 programmable blocks (1 parser block, and 4 control - match action).

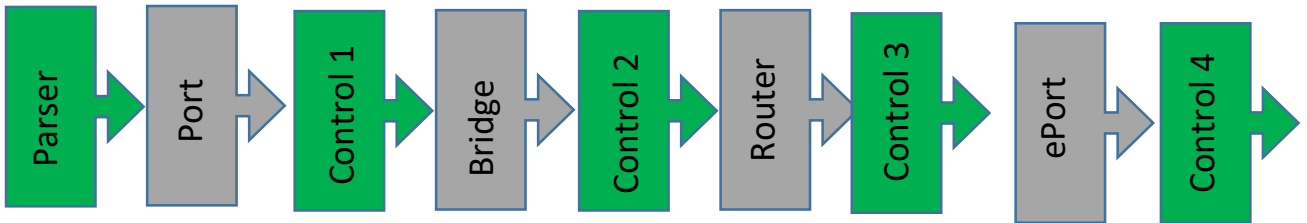


Figure 1. Target architecture.

Programmable block 1: parser

Mellanox provides parsing graph base line user will be able to add up to 4 new nodes to the packet-parsing graph.

Programmable block 2: ingress port

Ability to define chain of multiple match action tables supported actions – drop, forward to port , mirror, packet modification, routing(including ECMP) ,tunnels encap ,tunnel decp , set QoS, counters, meters ,go to table.

Programmable block 3: ingress router

Ability to define chain of multiple match action tables supported actions – drop, mirror, packet modification, routing(including ECMP) ,tunnels encap ,tunnel decp , set QoS, counters, meters ,go to table.

Programmable block 4: egress router

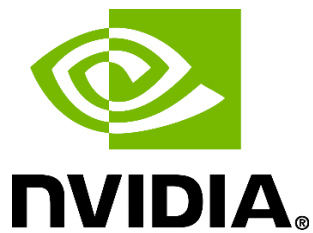
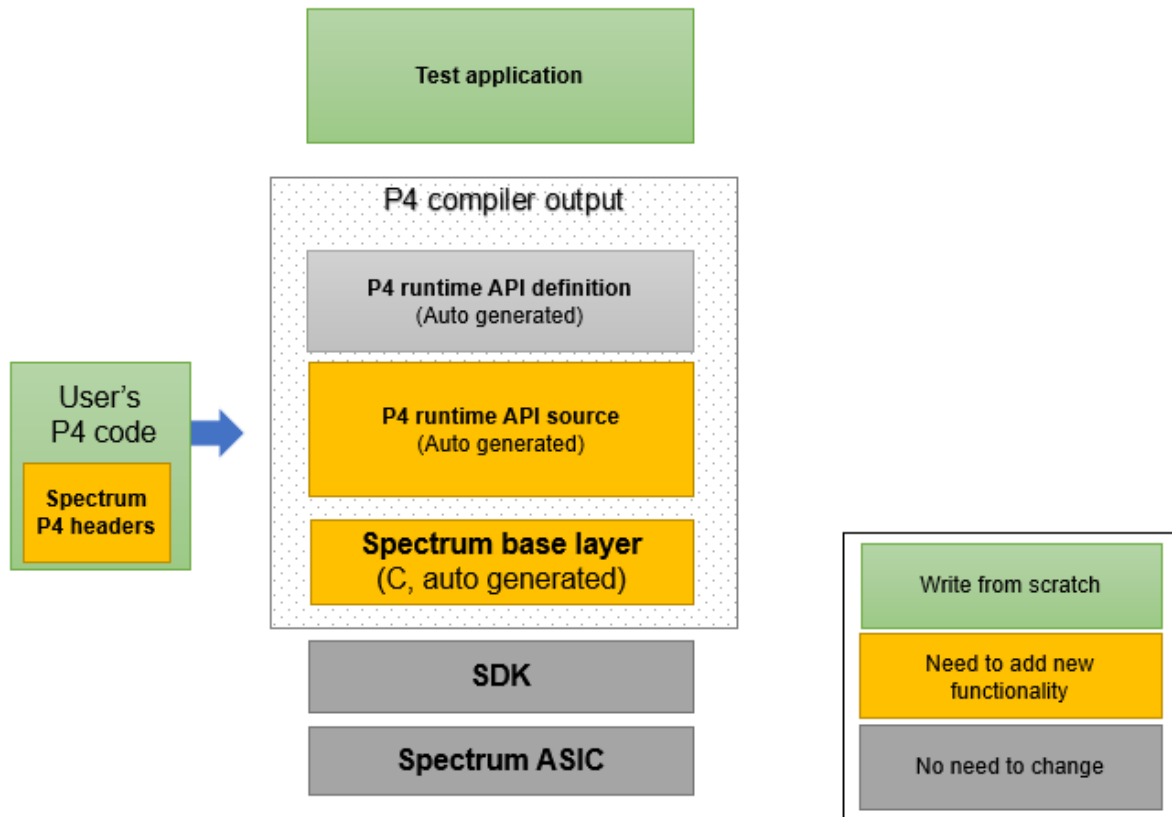
Ability to define chain of multiple match action tables supported actions – drop, mirror, packet ,forward to port , packet modification, set QoS, counters, meters ,go to table

Programmable block 5: egress port

Ability to define chain of multiple match action tables supported actions – drop, egress mirror, packet modification, set QoS, counters, meters ,go to table “



Appendix B: Architectural schema





References and Resources

- [1] <http://p4.org/>
- [2] The P416 Programming Language: <https://dl.acm.org/citation.cfm?id=3139648>
- [3] Basic P4 exercise on Mininet: <https://github.com/p4lang/tutorials/tree/master/exercises/basic>
- [4] Prototyping in P4: <https://p4.org/p4/prototyping-in-p4.html>
- [5] <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8845315&tag=1>

Additional resources

1. P4 tutorials on GitHub (see readme for install instructions):

<https://github.com/p4lang/tutorials>

2. P4 mailing list:

http://mail.p4.org/pipermail/p4-dev_p4.org/

3. P4 runtime:

<https://p4.org/p4-runtime/>

4. Mellanox SDK API:

http://www.mellanox.com/page/products_dyn?product_family=124&mtag=switchx_sdk

5. Mellanox P4 compiler:

Code repository will be shared with the students.