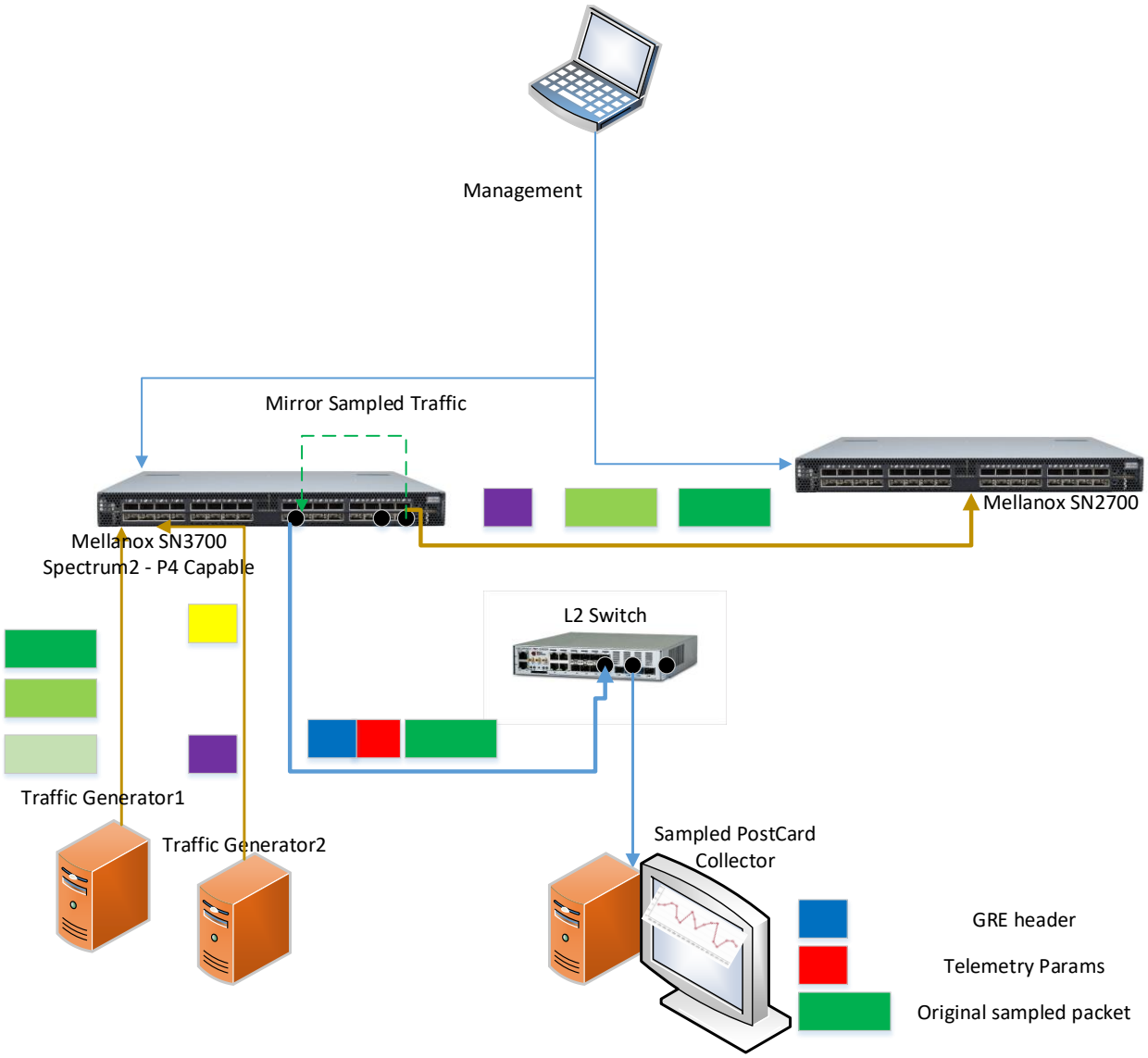
Enhanced Postcard on Mellanox Switch Programmed by P4

Abstract:

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.

Network monitoring plays a significant role in network management. It is used for a variety of applications such as QOS, billing, traffic engineering, security and anomaly detection. While some of these applications require only flow statistics, many require more specific packet-level information as well as networking hardware resource operational status. The new Mellanox SN3700 P4-capable Spectrum-2 based, supports the ability to perform mirror of sampled traffic, encapsulate it to GRE and add to it additional telemetry meta data like: Switch ID, time stamp, ingress/egress queues occupancy and congestion status, egress port link utilization and more . Unlike similar techniques like In-band Network Telemetry (INT) and in-situ Operation Administration and Maintenance (OAM), the Postcard-Based Telemetry (PBT) does not require inserting telemetry data into sampled user packets, but directly exports the telemetry data to a collector through separated OAM packets called postcards.


Management
Mirror Sampled Traffic
Mellanox SN2700
Mellanox SN3700
Spectrum2 - P4 Capable
L2 Switch
Traffic Generator1
Traffic Generator2
Sampled PostCard
Collector
GRE header
Telemetry Params
Original sampled packet

Goals:

The project's objective is to learn P4 programming language and deploy it on Mellanox P4 Capable switch (SN37000). The project will include the following phases:

- Learn the P4-16 language
  - Refer to http://p4.org/
  - Read the paper **The P416 Programming Language:**

    https://dl.acm.org/citation.cfm?id=3139648
  - Perform basic P4 exercise on Mininet - https://github.com/p4lang/tutorials/tree/master/exercises/basic
  - Learn about P4RunTime and perform P4 exercise on Mininet –

    https://p4.org/api/announcing-p4runtime-a-contribution-by-the-p4-api-working-group.html

    https://github.com/p4lang/tutorials/tree/master/exercises/p4runtime

- Implement Traffic postcard sampling-on-Demand using P4 and P4RunTime on Mininet
- Learn the Mellanox p4 target architecture (See Appendix A)
- Learn the Mellanox p4 Architecture Schema (See Appendix B)
- Refer to previous student project - https://gitlab.cs.technion.ac.il/lccn/w2018-mirror-sampling-mellanox-p4 and implement mirror sampling.
- Implement on the ingress sampled traffic GRE tunnel encapsulation with the additional telemetry params: switch ID, time stamp, **ingress/egress queues occupancy and congestion status, Switch latency.**
- Inject CAIDA traffic from Traffic Generator 1 along with **random bursts generated by Traffic Generator 2 – in order to congest the** egress queue.
- Implement on the collector side a GUI dash board that will show:
  - queue status and occupancy changes over time. Refer to previous student project https://gitlab.cs.technion.ac.il/lccn/s2019-postcard-p4
  - List the 1st level and 2nd level congested flows
- (nice-to-have) Configure the above using Mellanox p4runtime API

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).

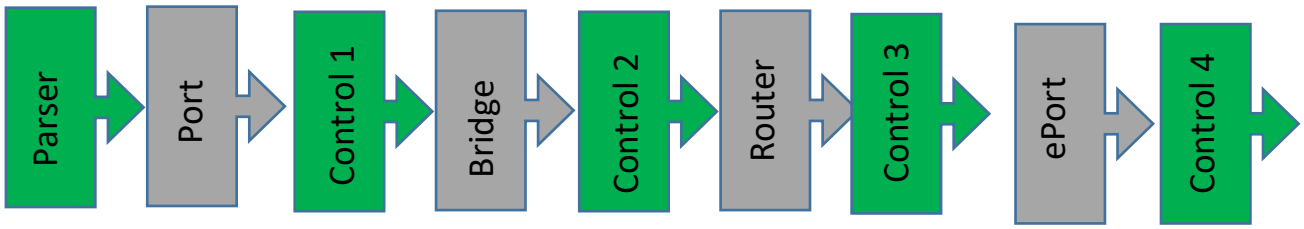
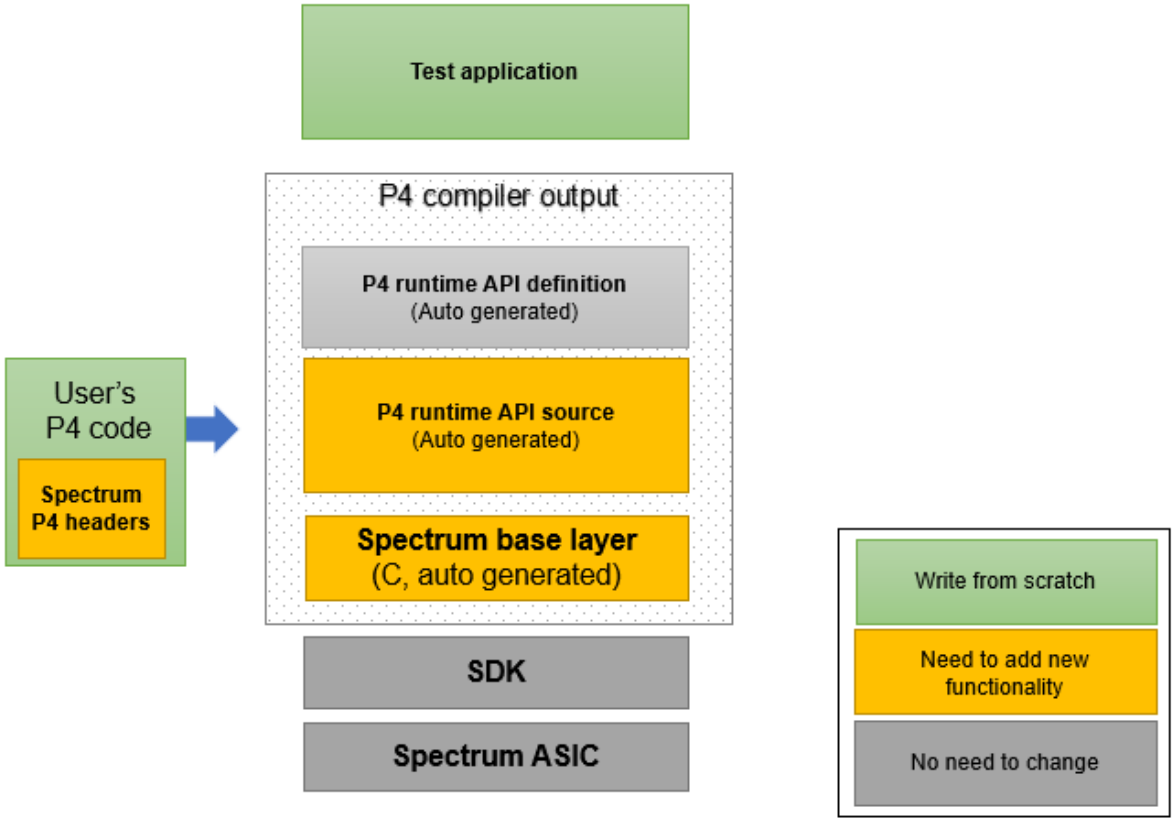


Figure 1. Target architecture**.**

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



Requirements:

Introduction to Networking Course (236334)

Guided by:

Matty Kadosh & Alan Lo

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.