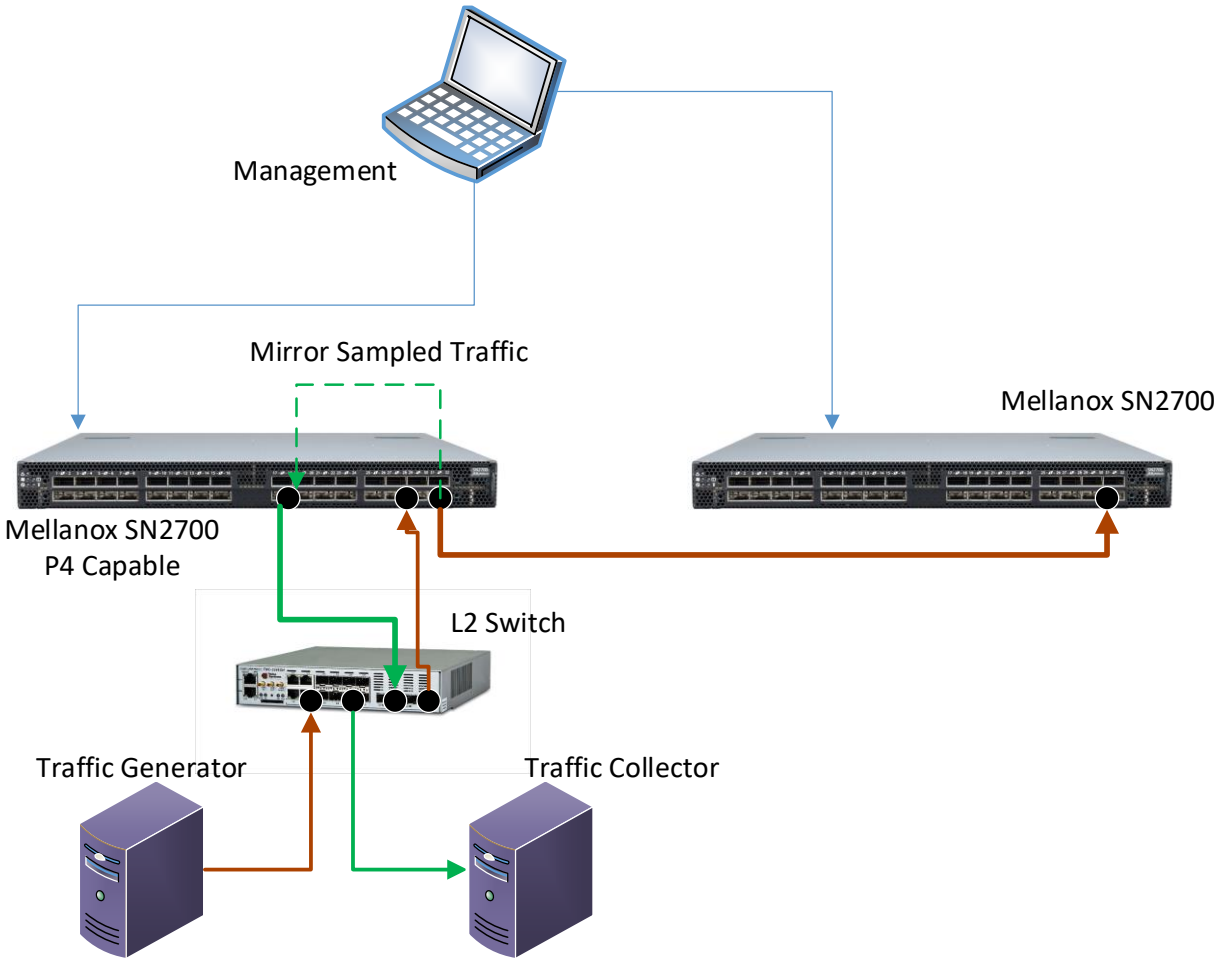
Traffic Sampling-on-Demand on Mellanox Switch Programmed by P4

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. Any P4 capable switch must support ability to perform mirror of sampled traffic.



Goals:

The project's objective is to learn P4 programming language and deploy it on Mellanox P4 Capable switch (SN27000), along with the new traffic sampling feature. The project will include the following phases:

- Learn the P4-16 language
  - Refer to http://p4.org/
  - Read the paper **The P416 Programming Language:** http://delivery.acm.org/10.1145/3140000/3139648/p5-budiu.pdf?ip=132.68.46.73&id=3139648&acc=ACTIVE%20SERVICE&key=0D17F1A88EABC760%2E73E0BAC401C36D1F%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&__acm__=1537165784_5ffb6ebf9b56f7f7a0069597c6ec9ab7
  - 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://github.com/p4lang/tutorials/tree/master/exercises/p4runtime

- Implement Traffic 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)
- Add to it traffic Sampling-on-Demand capability (using Programmable block 2)
- Configure the above table using Mellanox using p4runtime API
- Raise the above setup and demo:
  - Per port and per flow (5-Tuple) Sampling-on-Demand
  - Ability to perform VLAN encapsulation of egressed sampled traffic
  - Ability to perform packet truncate of egressed sampled traffic

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).

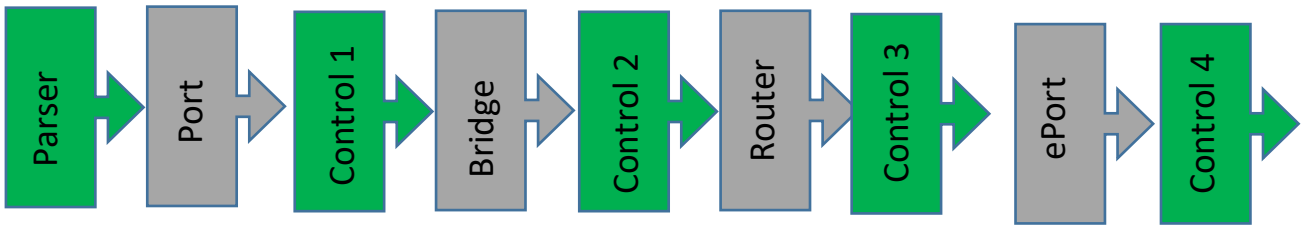
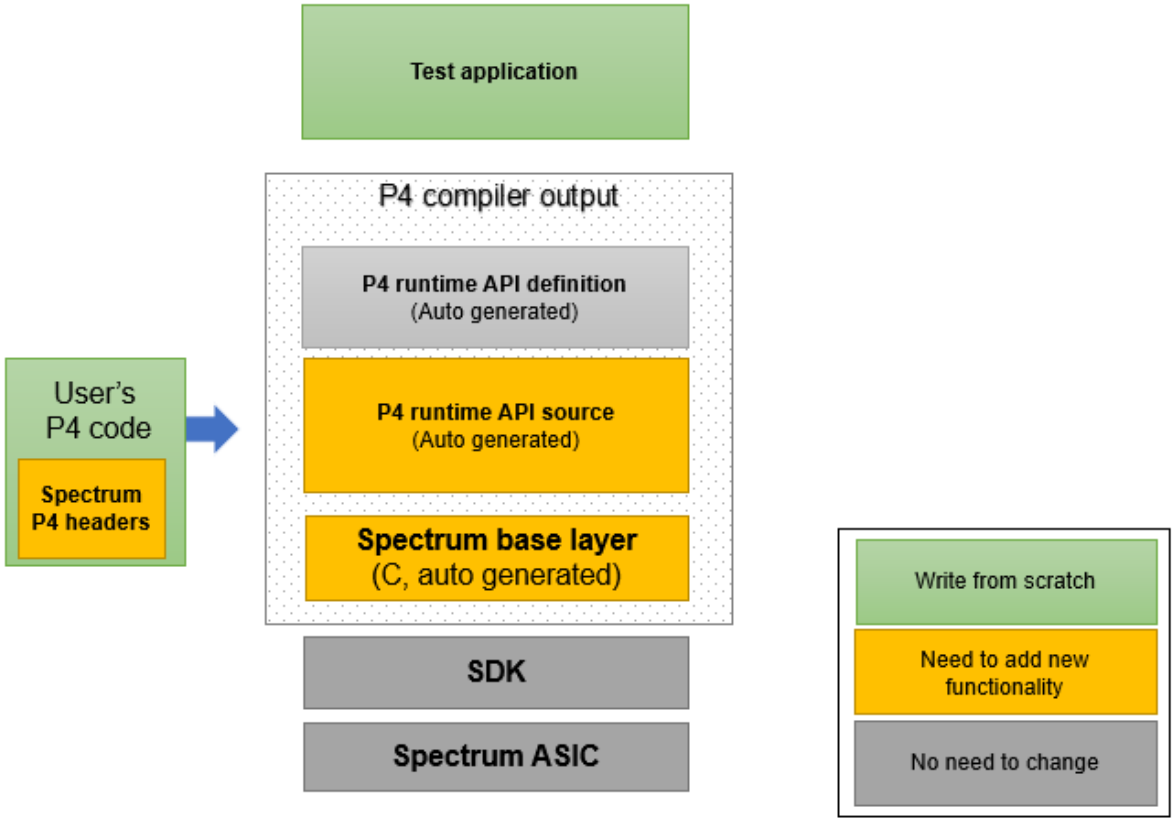


Figure 1. Target architecture**.**

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 & Omer Shabtai from

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.