

Paper Evaluation, OpenNF: Enabling Innovation in Network Function Control

Jiani Jiang <jianij@kth.se>

1. Paper summary

To realize packet processing that can be redistributed across a series of NF instances, a framework that is efficient and provides state control is required, to reach the three goals of short SLAs, traffic monitor and operation, and low operating cost. There is a trade-off between NF accuracy and performance during the reallocation of flows across NF instances. Therefore, this paper presented OpenNF that can satisfy all the requirements and still keep efficiency. The key idea is to enable control applications to directly manage the operation and performance of NFs. Also, it defined a southbound API to request events or the export or import of internal NF state and a northbound API for the control applications interactions. When the control application issues the state operation, the controller will translate that into the southbound API calls to do the state transfer, and then communicate with forwarding module to update the forwarding state in the switch and steer the traffic. In addition, two new schemes, an event abstraction for monitoring and a two phase forwarding state update scheme, are designed for the controller to guarantee the loss-freedom, order-preservation, and consistency for state and state operations.

2. Top 3 contributions

- a. Except dynamic reallocation of packet processing that enables new services, it realizing the ability to quickly move copy or share network function state in a safe control module.
- b. Make it possible to realize the goals of short SLAs, low operating cost and accuracy at the same time.
- c. Raised a new southbound API that can allow a controller to request the export or import of NF state to accommodate a variety of NFs with the increase code size by only 9.8% and acceptable increased packet processing time.

3. Problems

- a. I think it is deal with flow-based traffic. But the evaluation doesn't show the cases like a burst happens or there is mix traffic with large flows.
- b. What if the traffic needs to be processed through multiple NFs? There may be a chain of NFs, can it cooperates with Metron (paper 15th) to deal with that?
- c. Also I think it needs buffer to support state transfer. How to decide the trade-off between the buffer and the mount of transferred states?