Implementing Carrier Services in OpenFlow- OpenFlow Switch as MPLS PE-Router

Authors:
Konstantin Agouros  
Xantaro Deutschland GmbH  
kagouros@xantaro.net

Omkar Deshpande  
Technische Hochschule Cologne  
omkar.deshpande21@gmail.com

Keywords: SDN, OpenFlow, MPLS, BGP, OpenDaylight

The aim of this project is to build an OpenFlow underlay network and deploy value-added carrier services like L2VPNs, L3VPNs. As many projects in this area rely on the functional level only and implement the functionality in a virtual environment this research also includes testing the setup with real hardware in the form of a physical OpenFlow switch and physical routers that build the MPLS core network. Performance comparisons and scalability envelopes of such deployments with traditional IP/MPLS-based networks will also be presented.

Software Defined Networking (SDN) using OpenFlow allows the control of traffic in an SDN controlled network based on header fields of the packets or even the content if packets are sent to the SDN controller for analysis. In an ideal world networks can be built from scratch and in such an environment SDN technology can be used to control all of the networks. However, in real life existing infrastructures and SDN networks need to co-exist and interwork.

In order to integrate with the classical infrastructure, the OpenDaylight (ODL) controller is used to learn the topology of the network and based on that push the flow definitions to OpenFlow switch so that packets can enter the Multi Protocol Label Switching (MPLS) network with the right labels attached to them.

MPLS is a well-established technology in carrier networks to create label switched paths (LSP) across various routers while keeping the forwarding tables much less complex allowing for faster forwarding of the packets. The management of these paths can be manual or using routing protocols like Label Distribution Protocol (LDP), Resource Reservation Protocol-Traffic Engineering (RSVP-TE). These paths are computed using Path Computation element (PCE) within a domain or across multiple domains. When a node in the network also called as Path Computation Client (PCC) requires an LSP, it sends request to PCE using Path Computation Element Protocol (PCEP). PCE then calculates the path with the help of Constraint Shortest
Path First (CSPF) algorithm fulfilling constraints like Bandwidth guaranteed, end-to-end delay, maximum links traversed. PCE contributes in separating path calculation from network topology creation which traditionally has been accomplished by ingress router that receives the path request. PCE can also change or request to tear down the LSP when no longer required.

The entry point into an MPLS network is called Provider Edge (PE) router and it is responsible for pushing the right label on a packet when it enters the MPLS network, and remove the label and forward it using classical forwarding based on Layer 2 or Layer 3 forwarding tables at the exit of the MPLS network. Since a PE router is also responsible for calculating the path, the key concern here is that it has restricted visibility of the network. Hence a PE router cannot calculate MPLS-TE end-to-end path across multiple areas.

By shifting the role of head-end router to a centralized entity, in our case ODL controller which provides a global view of the topology, helps in improving the performance and adds more flexibility to the network. In order to calculate LSPs at the centralized controller, it needs to have a database termed as Traffic Engineering Database (TED) which stores topology information used for computing path computation. There are a number of approaches to collect this topology information. One of the routine procedure is to get all the topology resources and information from an Interior Gateway Protocol (IGP) by peering it with the controller. But IGP has limitations when a network is divided across multiple domains. It could be a challenging task to decide the location of the central entity in such scenario.

Considering some of the drawbacks of the existing method, an alternative approach is to use an extension of Border Gateway Protocol (BGP) called as BGP-LS (Border Gateway protocol Link-State) to distribute the link-state information to the ODL controller. This way link-state and traffic engineering information can be extracted and shared with the ODL controller. If controller uses IGP peering to learn link-state information, it needs to peer with each area to obtain entire visibility of the network. When comparing it with the IGP peering, extending peering with BGP appears to be a lot more feasible. The main reason for using BGP-LS is fact that it supports multi-hop sessions whereas IGPs require direct sessions. BGP-LS also supports extensive filtering which allows you to not accept any BGP information from the controller. But with IGP in place, any issue on the controller can affect your core network due to the lack of filtering capabilities in IGPs.

This work examines the feasibility of using the combination of an OpenFlow Switch and an OpenFlow controller to fulfil the functionality of a PE router. Starting with static paths and then integrating with the routing protocols for the network to create flows that manipulate the labels like a PE-Router would be based on the information learnt from the classic network. This deployment is one way to achieve integration with traditional networks as it provides benefits of inter-domain routing, adaptive path computation and overall giving a better way to utilize the current network resources.