OpenQFlow: Scalable OpenFlow with Flow-based QoS

April 10, 2013
Nam-Seok Ko (nsko@etri.re.kr)

Net-Computing Convergence Research Section
Smart Network Research Department
Agenda

- SDN (Software Defined/Driven Networking)
  - What is SDN?
  - Problem Analysis

- OpenQFlow
  - Scalability and QoS Enhancement for SDN
  - Prototype Implementation

- Standardization Opportunities

- Q & A
What is SDN?

- **Software Defined (or Driven) Networking (SDN)**
  - An enabler of network programmability through
  - separation of control plane from data plane
  - open interfaces among control plane, data plane and application layers
Problem Analysis

- Scalability and Performance Issues

  - Need to setup a separate rule for each N uflow
  - Maximum N exceptions

- Scalability Issues in Supporting Fine-grained QoS
  - Forwarding and QoS rules are tightly coupled
  - Need to setup separate QoS rules for each microflow

- Performance Issues
  - Every packet in an microflow should search multiple rule tables
Related Works

- **DevoFlow (Devolved Flow)**
  - Minimize the interactions between OpenFlow switches and controlled
  - Keep flows in the data-plane as much as possible
    - Provision enough wild-card rules to data-plane
    - Rule-cloning: microflow-based exact match rules
    - Determine long-lived flow using statistics sampling or triggering
    - Controllers get involved in handling long-lived flows

- **DIFANE (Distributed Flow Architecture for Networked Enterprises)**
  - Distributing the rules across “authority switches”
OpenQflow

- Objectives
  - To support scalable and stateful SDN which provides microflow-based QoS

- Distinctive Features
  - Clear separation of QoS rules from forwarding rules
  - Flow learning at microflow level
    - Learn every information in the first packet processing of a microflow
    - Simplify forwarding for the subsequent packets in a flow
  - Fine granular flow management regardless of the granularities of forwarding and QoS rules
    - Coarse granular forwarding and QoS rules – aggregation of forwarding and QoS rules
    - QoS profile types of QoS rules
      - E.g., if (DSCP value = 10) then 10Mbps guaranteed bandwidth for each flow
OpenQflow (cont’d)

- Distinctive Features (cont’d)
  - Complex packet processing in edge node but simpler processing in core node – SDN header
    - Flow label – an unique identifier for each microflow in an SDN domain
      - Does not necessarily mean that each and every microflow has its own flow label; flow label is sharable among multiple best-effort flows
        e.g., best effort traffic share one single flow label to next hop node
      - Short-lived flows may not need to have a separate flow label as well
    - QoS information
      - QoS type, rate, delay, jitter, etc.
OpenQflow (cont’d)

- Separate Rules for Forwarding and QoS
  - Much fewer exceptions

- Flow Learning Table
  - N dynamic learning flows

- Forwarding
  - Exception is not required

- QoS
  - N uflows

- Flow Learning
  - Exception is not required

- Controller

- Fewer interaction b/w switch and controller
  - Separation of QoS rules from Forwarding rules
  - Multiple micro-flows could share one QoS profile

- Performance Enhancement
  - Only the first packet goes through all the complex packet processing and then learn the information into flow state table
  - All the subsequent packets are processed according to the flow state table
OpenQflow (cont’d)

- Complex processing in edge node & simpler processing in core node
  - Edge node
    - Lookup multiple flow tables and refer to SDN controller for undefined flows
    - encapsulate/decapsulate SDN header (flow label, QoS information, etc.)
  - Core Node
    - Lookup one table against the SDN header (mostly it will be in the format of label)
OpenQflow (cont’d)

- Flow Learning Table
- Forwarding Rule
- QoS Rule
- Scheduler

1st Packet

Flow creation and packet scheduling into calendar queue based on flow QoS

Core #1

Flow Processing

Subsequent packets

Core #X

Flow Processing

QoS Co-processing

Scheduling out packets from calendar queue

Data plane prototype on a commercial multicore processor (Cavium multicore CPU)

Throughput monitoring, fair bandwidth calculation
Standardization Activities

- **ONF**
  - OpenFlow Switch Specification - OF 1.4 (08/2012)
    - open communication protocol between control plane and data plane
  - OpenFlow Management and Configuration Protocol - OF-Config 1.1
    - remote configuration of openflow switch

- **IETF & IRTF**
  - ForCES
  - SDNP BoF, SDNRG

- **ITU-T**
  - Q.21 of SG13 Future Networks
    - Y.FNsdn - Framework of software-defined networking
    - Y.FNsdn-fm - Requirements of formal specification and verification methods for SDN

- **ETSI**
  - NFV ISG
Standardization Opportunities

- **Forwarding Architecture**
  - Separation of QoS rules from forwarding rules
  - Flow learning table

- **Scalable Stateful SDN – SDN header**
  - Flow label – make simpler packet processing in core node
    - Default flow label for short-term and best-effort flows,
    - or separate flow label per each flow for enhanced packet processing
  - QoS information – enhanced QoS processing
    - Label-inferred packet processing,
    - or separate encoding for explicit QoS treatment (QoS type, rate, delay jitter, etc.)

- **Where?**
  - Study feasibility in ITU-T and/or ETSI in framework level
  - Creation or modification of protocols should be done in ONF and/or IETF
Q & A

Thank you.