A Resource Delegation Framework for Software-Defined Networks

Ilya Baldin ibaldin@renci.org
Shu Huang shuang@renci.org
Rajesh Gopidi rajesh@renci.org
The Problem

• Offering network services in a *multi-domain* environment
• In SDN environment primarily boils down to
  – Controller coordination for provisioning (e.g. DISCO)
  – ‘Big-switch’ abstractions
• What we would like
  – Offer heterogeneous network services (QoS, resiliency, virtual networks) across many providers
  – Allow providers to trust ‘alien’ controllers
  – Support nested virtualization
  – Separate resource management from provisioning
    • Split off coordination of resource allocation on long term scale from short-term provisioning

This work is funded by DOE ASCR award DE-SC0007425 and NSF grant CNS-1111256
The Overview

- Multi-domain environment where customers use ‘services’ (connections or virtual networks)
  - Different QoS requirements (latency, bandwidth, resiliency, isolation)
  - E.g. the Internet can be one of the overlays
The Problem Formulation

• **Label**
  – Labels can be translated in some points in the network.

• **Technology level**
  – Binds to a label offset or type.
  – Technology levels have a partial order where A < B indicates B can enclose A and carry its traffic.
  – E.g. 802.3 > IP (not necessarily OSI-compliant).

• **Label extents within technology levels** are defined per port, per network element along with other consumable resources.
  – Outgoing bandwidth, Buffer space, Flow table space.

• **Represent multi-dimensional spaces** that can be tested for inclusion/intersection.

• **A portion of such volume represents a delegation of resources**
  – Can be used by a controller to form forwarding rules managing traffic.
  – Can be further subdivided to create nested delegations.
Example

- Constraints are needed to describe delegations
  - Volume inclusion
  - Path/label continuity
  - Label/bandwidth accounting
- ILP formulation in the paper

Customer View

Provider View
The Architecture

• Form and describe delegations
  – Resource manager
  – Coordinated by customer, directly or via broker
• Inform controllers
  – Controllers know the constraints
• Enforce delegations on behalf of providers at switch level
  – PDP vs. PEP
  – Flow space manager
• Pervasive Authz
The Prototype

• **Delegation framework**
  – GUI tool to form reservations and describe using GraphDB
  – Floodlight module to accept these descriptions

• **Sample multi-domain application**
  – Virtual transport provider built out of 3 other providers (virtual or physical)
  – Provides transport path-based services using a portion of L2 MAC address field delegated to it for path identification

• **Tested in a GENI slice**
The Experiment

Transport Provider D
The Experiment (GENI Slice)
The Prototype Modules in Floodlight

• **Topology delegation**
  – Accepts constrained delegation descriptions in GraphDB

• **Topology verification**
  – Uses stochastic probing across delegated label space to verify connectivity
  – LLDP not suitable for this purpose

• **ARP Resolution**
  – Listens for client ARP requests
  – Performs substitution of MAC address with Path IDs

• **Circuit computation**
  – Computes paths and assigns path IDs
Outcomes

• Direct control over provider equipment with verifiable constraints
• Explicit communication of constraints to controllers
• Nested virtualization
• Efficient use of label spaces
• Dynamic resource allocation
• Multiple approaches in one architecture
• Support for an economy