Self–Adaptive Routing

Arthur van Kleef
Marvin Rambhadjan

System and Network Engineering

June 30, 2010
Network management involves complex tasks.

<table>
<thead>
<tr>
<th>Congestion Control</th>
<th>Load Balancing, Rerouting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quality of Service</td>
<td>i.e. Voice and Video</td>
</tr>
<tr>
<td>Provisioning</td>
<td>Resource Reservation</td>
</tr>
</tbody>
</table>
Existing protocols

Review of existing protocols based on adaptivity

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Layer</th>
<th>Capabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF</td>
<td>3</td>
<td>Path Cost</td>
</tr>
<tr>
<td>BGP</td>
<td>3</td>
<td>Local Pref, MED, Next-Hop, AS-Path</td>
</tr>
<tr>
<td>MPLS(-TE)</td>
<td>2.5</td>
<td>Explicit LSP’s</td>
</tr>
<tr>
<td>PBB(-TE)</td>
<td>2</td>
<td>Service and Trunk coupling</td>
</tr>
<tr>
<td>STP</td>
<td>2</td>
<td>Path Cost, Priority</td>
</tr>
</tbody>
</table>

The control plane becomes more complex by adding new protocols and protocol extensions.
Control Plane

The control plane performs the following functions:

- Control Connections
- Disseminate connectivity information
- Calculate optimal path

Software programs are good at handling complex tasks.

Separate forwarding plane from control plane (ForCES).
Research Questions

- What is the architecture of a network that supports a software control plane?

- If the control plane becomes software, what is the general pattern of the programs that implement routing and network management?
OpenFlow Operation

1. Packet sent to switch
2. Switch passes packet to controller
3. Control decides action
4. Install flow entry
5. Packet forwarded to the host
OpenFlow Tuple

Flows describing traffic.

<table>
<thead>
<tr>
<th>Ingress Port</th>
<th>Ethernet src</th>
<th>Ethernet dst</th>
<th>Ethernet type</th>
<th>VLAN id</th>
<th>VLAN priority</th>
<th>IP src</th>
<th>IP dst</th>
<th>IP proto</th>
<th>IP ToS</th>
<th>TCP UDP src port</th>
<th>TCP UDP dst port</th>
</tr>
</thead>
</table>

- Actions
  - Send to controller
  - Forward
  - Flood
  - Drop

If switch does not have a flow-entry that matches a certain packet header then it forwards a packet to the controller.
Network Control Program

Reference  Desired behavior
Controller  Controller responsible for taking forwarding decisions
System  Programmable network infrastructure
Sensor  Monitoring for switch statistics
Resources were limited, so can we virtualize OpenFlow networks?

- **Hosts** User Mode Linux
- **OpenFlow Switches** User Mode Linux + Open vSwitch
- **Connections** Virtual Distributed Ethernet (VDE)
- **Controller** NOX OpenFlow Controller

Large topologies, runs on moderate hardware, flexible
Control Program components

Control Program consists of different components that:

- Maintain network topology in a graph
- Track the location of end hosts
- Monitors traffic and link utilization
- Calculates paths between source and destination
  - Shortest Path
  - All possible paths
- Installs flow entries on OpenFlow switches
NCP Operation

1. Receive packet in event
2. Learn ethertype
3. Locate destination node in the topology
4. Add or consider active policy
5. Calculate path to destination
6. Return flow entry
Example: Case Network

- OF\(N\) Open vSwitches
- Host\(N\) network hosts
- Virtual Distributed Switches
Example: Single Traffic Flow

One traffic flow
Example: Extra high priority traffic flow

Another traffic flow with a higher priority
Example: Link Failure

Two traffic flows with a link down
FlowVisor is a transparent proxy between OpenFlow switches and controller(s).

Multiple FlowVisor controllers can be added to balance the load between the controllers.

The network can be “sliced” and the control is delegated to a controller and are based on the following criteria:

<table>
<thead>
<tr>
<th>Layer</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>src / dst TCP or UDP, ICMP code</td>
</tr>
<tr>
<td>3</td>
<td>src / dst IP address, IP Protocol, IP TOS</td>
</tr>
<tr>
<td>2</td>
<td>src / dst Ethernet address, VLAN</td>
</tr>
<tr>
<td>1</td>
<td>Physical switch port</td>
</tr>
</tbody>
</table>
OpenFlow provides an architecture that support separation of the control plane and the forwarding plane.

The general pattern of the software control plane is a feedback control loop.

The test environment presented is very useful for OpenFlow experiments.
Questions