Customizing Data-plane Processing in Edge Routers

Fulvio Risso, Politecnico di Torino

http://fulvio.frisso.net
My kids
Francesco creates a Christmas card
How do my kids connect to the Internet?
How the edge of a NSP looks like?

- Edge network router
- WAN accelerator
- CDN web cache
- IDS/Firewall
- Network Monitor
- QoS
- Internet

- Processing linecards
- High-speed interconnect
- Interface linecards
A network with programmable edge routers

Both domestic and NSP routers can be part of the vision

Network Node Virtualization Framework

Applications operating on the network slice associated to User 1

Applications operating on the network slice associated to User N

Switching path (hardware)

Network traffic

Network Service Provider

Management server

User authentication, applications, permissions...

Network traffic
A possible use case

- Parental control
- Personal firewall
- Web advertis. killer
- Network monitor
- QoS
- VPN
- WAN accelerator
- IDS
- GTalk contact filter
- Personal firewall
- Video streaming optimizer
- Web cache
- Network monitor
- Traffic in
- Lawful interception
- Traffic out

Network edge node
We propose the creation of **user-driven data plane applications** that operate on a **network slice** associated to a given **actor**, enabling the customization of the **processing** of the traffic inside the **network edge router**.
General architecture of the prototype

Motherboard Mini-ITX, 4 GbE ports (Intel) + 1 GbE (on mainboard)
WiFi USB Dongle
CPU i5-3450S
RAM 4 GB DDR3

User Apps in VMs
OpenvSwitch

Flowvisor

Network gateway

Internet

MySQL

Management server (DB + apps on disk)
The current edge node

Controller
- User 1
  - App3
  - App2
  - App1

Controller
- User 2
  - App5
  - App4
  - App1

Controller of this node
- CNode
- App

Default Controller
- Default App

Web Node Manager
- Tomcat on port 80

Network Hypervisor

Controller plane

Switching plane

SoftSwitch

OPEN VSWITCH
An Open Virtual Switch

ARP → *: send to all ports
* → ARP: send to all ports
DHCP → *: send to all ports
* → DHCP: send to all ports
mcast, bcast: send to all ports
Default: send to Controller
Managing user applications

Create the custom application flow

Applications can be selected from the list of available apps and copied to the list of the installed apps, started/stopped, and stacked in a different calling order.

Install/Uninstall applications

Users can install their own application by simply uploading their file. Application that are under the ownership of the selected user can also be uninstalled. Applications are stored in the management server and downloaded in the network node when a new user is recognized.

http://config.ctrl
Validation environment

Debian 7, 32 bits
Vanilla Open vSwitch
Modified FlowVisor

2 User controllers, 3 applications each
- DNSFilter
- Gsafe
- NetMon

1 NSP controller
- Default bridging

OpenFlow Controller

OpenvSwitch

Network gateway

L2 network

Two large file transfers (350 Mbps each)
### Memory and CPU load

<table>
<thead>
<tr>
<th>Component</th>
<th>Memory occupancy</th>
<th>CPU consumption (% on a single core)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Controller User 1</td>
<td>165 Mbytes</td>
<td>35%</td>
</tr>
<tr>
<td>Controller User 2</td>
<td>165 Mbytes</td>
<td>35%</td>
</tr>
<tr>
<td>Controller Edge Router</td>
<td>60 Mbytes</td>
<td>0%</td>
</tr>
<tr>
<td>Default Controller</td>
<td>46 Mbytes</td>
<td>0%</td>
</tr>
<tr>
<td>NSP Controller (transparent bridging)</td>
<td>48 Mbytes</td>
<td>11%</td>
</tr>
<tr>
<td>FlowVisor</td>
<td>150 Mbytes</td>
<td>81%</td>
</tr>
<tr>
<td>Open vSwitch</td>
<td>8 Mbytes</td>
<td>100%</td>
</tr>
<tr>
<td>Tomcat Web Server</td>
<td>156 Mbytes</td>
<td>0%</td>
</tr>
<tr>
<td>Operating System</td>
<td>274 Mbytes</td>
<td>2%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>1072 MBytes</strong></td>
<td><strong>264%</strong></td>
</tr>
</tbody>
</table>
Openflow is great

We were able to complete this prototype only thanks to OpenFlow

but....
Thanks to...


- The anonymous reviewer of the EWSDN paper
Team

Fulvio Risso, 41, Project Leader

Marco Pramotton, 24, OpenFlow prototype

Ivano Cerrato, 25, General architect

Marco Cita, 25, User interface, management server
To transport,
or to trasform,
that is the question.