EXPRESS: Implementing an SDN infrastructure over a federation of testbeds

(experiment within the OpenLab project)

Stefano Salsano
and the EXPRESS team: Giuseppe Siracusano, Federico Griscioli, Pier Luigi Ventre, Claudio Pisa, Andrea Detti, Nicola Blefari-Melazzi...

CNIT / University of Rome «Tor Vergata»

Pre-FIA workshop on Testbed Interoperability - Athens, March 17th 2014
EXPRESS objectives

• Design an innovative, resilient SDN system to extend the SDN applicability domain from fixed networks to intermittently connected network, like wireless mesh networks.

• Evaluate our solutions over a federation of three OpenLab testbeds (PlanetLab, NITOS and W-iLab.t), where PlanetLab plays the role of the core fixed network and NITOS and W-iLab.t play the role of the wireless mesh networks.
Outline

• **wmSDN**: wireless mesh Software Defined Network
  – SDN in a highly dynamic networking environment with network partitions /merging
  – Wireless Mesh Router (WMR) architecture
  – Controller selection aspects

• Testbed interoperability aspects
wmSDN
wireless mesh Software Defined Network

Controller

Internet

Client

Hosts

Controller

GW to Internet

Wireless Mesh Router (WMR)
wmSDN
control plane connections with controllers
wmSDN
control plane connections with controllers
wmSDN
“in-band” control plane

Controller

Internet

GW to Internet

Controller

Wireless Mesh Router (WMR)

Client

Hosts

17/03/2014
wmSDN
network partitions
wmSDN network partitions
Control and data plane

• The OLSR routing protocol for Mobile Adhoc Networks is used to establish a basic IP connectivity in the Wireless Mesh Net

• The OpenFlow/SDN control plane (switch-to-controller communication) goes “in-band” over the basic IP connectivity

• Data plane uses the IP connectivity or an “SDN based connectivity” in a flexible way

• When using SDN based connectivity, the routing of packet flows is decided by the SDN controller
Control and data plane

- IP based data-plane
- OpenFlow based data-plane

- IP based control-plane
- OpenFlow Controller (Home, Mesh, built-in)
wmSDN: architecture

- OpenFlow controllers
- OLSR Daemon
- OpenFlow switch (Open vSwitch)
- EFCM - External Flow table and Controller Manager
Details of WMR node architecture

- Each physical interface is associated with a virtual interface

- Traffic can be forwarded in two ways
  - IP forwarding which operates on the virtual interfaces (routing is set by OLSR)
  - with SDN rules in the OF switch
Master controller **election** in SDN fixed nets

- Communications among controllers are relatively reliable -> the controllers can run a master election procedure to take control of each switch

- (Implementation issue) Open vSwitch can only connect to a preconfigured set of controllers
Master controller election in SDN fixed nets

• In a traditional OF environment communications among controllers are relatively reliable -> the controllers can run a master election procedure to take control of each switch

• (Implementation issue) Open vSwitch can connect to a preconfigured set of controllers (must be known to the switch in advance)
Master controller election in SDN fixed nets

- In a traditional OF environment communications among controllers are relatively reliable -> the controllers can run a master election procedure to take control of each switch.
- (Implementation issue) Open vSwitch can connect to a preconfigured set of controllers (must be known to the switch in advance).
From Master Election to Controller Selection

• WMRs select the more appropriate controller given the connectivity status of the network (Controller Selection)

• The Controller Selection procedure is performed by the EFCM

• Simple strategy, based on a “Hierarchy of Controllers”: select the connected controller with the highest level in the hierarchy
From Master Election to Controller Selection
Experimenting in OpenLab testbeds

- Wireless testbeds:
  - W.iLab-T
  - NITOS
- Fixed testbed
  - PlanetLab
- Interconnection of Wireless testbeds with fixed “backbone”
- Ethernet over UDP tunnels across Planet Lab Europe to interconnect the testbeds
Testbeds interconnection issues

- Planet Lab nodes acting as L2 switches
- Ethernet in UDP tunnels
- External hosts with public IP address
- Management host
- Private IP addresses
- W.ilab-t
- NITOS

External host with public IP address

17/03/2014
Solutions for testbeds interconnection

Planet Lab nodes acting as L2 switches

External host with public IP address

Management host

Private IP address

External host with public IP address

NAT

NITOS

Ethernet in UDP tunnels

Creates the L2 overlay
Solutions for testbeds interconnection

Planet Lab

Ethernet in UDP tunnels

External host with public IP address

Planet Lab nodes acting as L2 switches

External host with public IP address

Private IP address

W.ilab-t

Management host

PF Port Forwarder

Private IP address

NITOS
Testbeds interconnection issues

- Planet Lab nodes acting as L2 switches
- Creates the L2 overlay, taking care of the connections coming from NAT boxes

- Ethernet in UDP tunnels
- Management host

- W.ilab-t
  - Private IP address

- NAT
  - Private IP address

17/03/2014
Solutions for testbed interconnection

- (1) “Port Forwarder” running in the node that interconnects a testbed with the Internet, it can run at kernel level or application level

- (2) Regular NAT translation of an outgoing UDP flow from a node in the testbed and terminating in the tunnel endpoint node in Planet Lab
Controller selection over federated testbeds

- The Network is divided in 3 partitions: blue, green and gray
- WMRs in different partition are connected to the best available controller (or using only IP routing if no controller is available)
Controller selection over federated testbeds

- Network partition are joined together
- WMRs are connected to the “best” available controller
http://netgroup.uniroma2.it/wmSDN/

Thank you for your attention

Questions?

UNIVERSITY OF ROME “TOR VERGATA”
Department of Electronics Engineering
Via del Politecnico, 1 - 00133 Rome - Italy

Stefano Salsano, Ph. D.
Assistant professor

Phone: +39 06 7259 7770  e-mail: stefano.salsano@uniroma2.it
Fax:  +39 06 7259 7435  http://netgroup.uniroma2.it/Stefano_Salsano