Project Overview

- GN3plus Open Call Project (CoCo)
- October 2013 – March 2015 (18 months)
- Partners: SURFnet (NL) & TNO (NL)
- Budget EUR 216K (50/50 split)
- 16.4 person months (50/50 split)
- Five work packages:
  - WP1: use cases & market demand
  - WP2: architecture, design & development
  - WP3: experimental validation
  - WP4: dissemination
  - WP5: project management
Community Connection (CoCo) Service

- Goal of CoCo service:
  - On-demand virtual private multi-domain, multipoint L2/L3 network instances
  - Interconnect laptops, VMs, storage, instruments, eScience resources
  - Each eScience community group can easily setup their own private CoCo instance via web portal
- Based on OpenFlow programmable network infrastructure
Use Cases Workshop

- Workshop for Dutch eScience researchers
- Held in Utrecht on 21 January 2014
- 15 participants
- Goal was twofold:
  - Get input for CoCoCo requirements by defining use cases
  - Get in contact with potential test users
Use Cases Workshop Results

Results from workshop:

- Desired service characteristics (some out-of-scope)
- Applications of CoCo service

Next use case steps:

- Refine the two high-level use cases in cooperation with “use case owners”
- In refinement (and next use case workshop) specific attention on:
  - feasibility and effort needed by network administrators to install CoCo agent
  - authentication and confidentiality requirements
CoCo Multi-domain Architecture

customer c1

data plane

domain d1

domain d2

domain d3

domain d4

control plane

CoCo agent a1

CoCo agent a2

CoCo agent a3

CoCo agent a4

web portal

web portal

web portal

web portal

customer c2

customer c3

customer c1
CoCo Control Plane

- Control plane consists of federated CoCo agents
- Each domain runs its own CoCo agent, based on OpenDaylight
- CoCo agents exchange information East-West about:
  - CoCo end nodes (used in web portal for CoCo candidates list)
  - CoCo instances identifiers (associated MPLS labels, etc)
  - Addresses used at end nodes (e.g. IP prefixes)
  - User and group authentication and policy parameters
- CoCo agent configures forwarding entries in OpenFlow switches via Southbound OpenFlow protocol
CoCo Data Plane

- MPLS based forwarding in the core
  - Outer MPLS label used to forward to destination PE switch.
  - Inner MPLS label identifies CoCo instance.
- Shortest Path Forwarding between two PEs (primary & backup)
- MPLS encapsulation and decapsulation done at PE
- At PE the customer traffic is aggregated onto MPLS paths
- All traffic between two PE switches aggregated
CoCo Forwarding Plane
Customer Connection Models

- OpenVPN based connection
  - Target users: laptops
  - User installs OpenVPN client on laptop
  - User connects with CoCo OpenVPN server

- OpenFlow based connection
  - Target users: servers, instruments, etc.
  - Campus network administrator installs OpenFlow switch at eScience group
  - eScience resources (servers, instruments, etc) connect to the OpenFlow switch
  - Campus network administrator configures 1 dedicated VLAN to carry CoCo traffic between OpenFlow switch and Customer Edge (CE) switch
  - Campus network administrator installs CoCo stub agent and sets up CoCo agent control plane peering relation with NREN
CoCo Customer Connection

stub CoCo agent

OF

CE

CoCo agent

PE

P

PE

customer

provider
Edge Encapsulation & Decapsulation

- Each PE has L2/L3 addresses behind it
- On ingress encapsulate traffic destined for those L2/L3 addresses with MPLS label that gets forwarded to that PE
- On egress pop the MPLS label and forward to CE
- L3 addresses are IPv4/IPv6 prefixes (aggregation and scalable)
- L2 addresses (MAC addresses) are a flat address space
  - Special attention needed for scalability
L3 VPN Service

- Each site has its own IPv4/IPv6 prefixes
- Each site runs its own address assignment mechanism (DHCP, SLAAC, etc)
- This has proven scalability over multiple domains (internet)
- CoCo infrastructure is based on OpenFlow switches
  - No next hop MAC address rewrite at each hop
  - Need a way to forward to the correct destination MAC address at final hop
  - Either use fake router MAC address at ingress and rewrite destination MAC address at egress
  - Or use destination MAC address of final IP hop already at ingress
L2 VPN Service

- MAC address are a flat user space
  - No aggregation, special needs for scalability
- Three challenges:
  - MAC learning
  - Address assignment
  - Broadcast, Unknown unicast & Multicast (BUM) traffic handling
L2 VPN Addressing Challenges

- MAC learning:
  - Need to learn L2 addresses used at all sites
    - ESADI
    - draft-ietf-trill-directory-assist-mechanisms-00
    - EVPN MP-BGP
  - Option: insert forwarding entries for active L2 addresses only

- Address assignment
  - MAC and IP addresses must be unique within multi-domain CoCo instance (VMs usually get generated MAC address)
  - Either centralised database or inter-domain negotiation

- BUM handling
  - Implement multicast (e.g. full mesh like VPLS). *Too much traffic?*
  - Forward BUM traffic to controller and handle in controller (e.g. proxy ARP). *For all multicast protocols?*
Summary

- The CoCo objectives are welcomed by eScience workshop participants
- CoCo only satisfies part of the requests
- Scalability by MPLS encap/decap and MPLS forwarding in the core
- BGP peering model to exchange information and policy between domains
- Scalability easy for L3 addressing
- Scalability harder to do for L2 addressing
Next Steps

- Implement single domain CoCo prototype (Q3 2014)
  - Using SURFnet OpenFlow testbed
  - Core network services based on OpenDaylight
- Test plan and testing/verification (Q2/Q3 2014)
- Involve end users and campus network managers
  - Followup on use cases workshop (Q3 2014)
- Enhance prototype to multiple domains (Q3/Q4 2014)
Related Work

- VPLS (used in IP exchange points)
- EVPN (being standardized in IETF L2VPN)
- BGP/MPLS VPN (used in GÉANT MDVPN)
Thank you!