Ethernet OAM enabled
OpenFlow Controller

Ronald van der Pol
rvdp@sara.nl
SARA

(this work was funded by SURFnet GigaPort 3)
Outline

- Introduction
- What is OpenFlow
- What is IEEE 802.1ag Ethernet OAM
- Open source implementation of IEEE 802.1ag
- Integration of 802.1ag with NOX OpenFlow controller
- Future plans
Introduction

- Example of how end-users can easily add network (management & routing) protocols to OpenFlow switches
- NOX OpenFlow controller is extensible with 3rd party components
- We implemented IEEE 802.1ag as NOX component
- Demo: monitor link status by running IEEE 802.1ag protocol between OpenFlow controllers via the datapath of OpenFlow switches (but it also interoperates with switches that support 802.1ag, e.g. Cisco switch in our demo)
What is OpenFlow?

- API to program OpenFlow switches (insert flow entries)
- Separate control plane and data plane
  - Control plane: OpenFlow controller
  - Data plane: OpenFlow switch
- Flow entry: match $\rightarrow$ action
- Match on Ethernet src/dst address, ethertype, IP src/dst address, TCP/UDP port, etc
- Action:
  - Output to port(s)
  - Drop
  - FLOOD
  - Modify (e.g. VLAN tag rewrite)
  - Send to controller
IEEE 802.1ag

- Protocol for Ethernet OAM (Operations, Administration and Maintenance)
- Ethernet frames with type 0x8902
- Three message types:
  - Loopback Messages (L2 ping)
  - Link Trace Messages (L2 traceroute)
  - Continuity Check Messages (periodic hello messages)
- Sent from Ethernet ports by Maintenance Points (MP)
- Maintenance Association consists of several MPs
- MPs act on specific Maintenance Level (0 .. 7)
  - Packets with higher level are forwarded
  - Packets with same level are processed
  - Packets with lower level are dropped
IEEE 802.1ag Type Details

- **Continuity Check (CC)**
  - Detect loss of connectivity
  - Periodic hello messages from MEPs
  - Processed by MEPs
  - CC frames sent to multicast group, no replies are sent

- **Loopback Message/Reply (LBM/LBR)**
  - Check for reachability
  - Sent manually from MEPs via CLI
  - Processed by MIPs/MEPs
  - Unicast request, unicast reply

- **Link Trace Message/Reply (LTM/LTR)**
  - Path information
  - Sent manually from MEPs via CLI
  - Processed by MIPs/MEPs in path
  - Multicast request including TTL, unicast replies
802.1ag MEPs and MIPs

Maintenance Domain Levels

- **router**
- **Ethernet switch**

- **MIP**
- **MEP**

MD level 7
MD level 5
MD level 0

SC11 SCI.net Research Sandbox Experiment Results,
17 Nov 2011, Seattle, USA
rvdp@sara.nl
Open Source Implementation

- Dot1ag-utils ([http://nrg.sara.nl/dot1ag-utils](http://nrg.sara.nl/dot1ag-utils))
- New BSD License
- Supported on Arista, FreeBSD, Linux and MacOSX
- Integrated with NOX OpenFlow controller
- Work In Progress

- Powerful debugging tool for Ethernet based lightpaths, VPNs, etc.
- Ping to Ethernet MAC addresses of routers and switches
- You only need a server and install the software on it
Booth #642 demo

- Dutch Research Consortium booth
- Link status (UP/DOWN) monitoring of multi-domain (Canada, Netherlands, USA) Ethernet network
- Ethernet OAM (IEEE 802.1ag) used to monitor links
- Link status recorded in RRD database
- PerfSONAR used to publish current link status
- Collector retrieves current perfSONAR data and presents status of complete network on website
OpenFlow Demonstrations
by CRC (Ottawa), iCAIR (Chicago), NCHC (Tainan), SARA (Amsterdam)

SCInet Research Sandbox
Flow Entry:
dl_type=0x8902, actions=CONTROLLER

IEEE 802.1ag PDU

Pronto OpenFlow switch

IEEE 802.1ag PDU

IEEE 802.1ag PDU

OpenFlow Protocol

OpenFlow Protocol

dot1ag-utils component

NOX OpenFlow Controller

IEEE 802.1ag PDU

IEEE 802.1ag PDU

IEEE 802.1ag PDU
Monitoring of Ethernet OAM

Initiator

Ethernet VLAN

CCM PDU

OpenFlow

CCM PDU

OpenFlow

Receiver

Webbrowser

NOX

OpenFlow controller

dot1ag-utils

continuity check daemon

Unix server

RRD file

perfSONAR-PS

Measurement Point

Other Measurement Points

SOAP/XML

Apache

Webserver

CGI script

 perfSONAR client

Monitor

CCM PDU

Ethernet VLAN

CCM PDU

Ethernet VLAN

CCM PDU

Ethernet VLAN

CCM PDU
SC11 SCInet Research Sandbox Experiment Results,
17 Nov 2011, Seattle, USA

rvdp@sara.nl
Future Work

- Implement complete 802.1ag state machine

**Current limitations**
- No forwarding of 802.1ag frames
- Partly implemented Maintenance End Point functionality
- No SNMP agent implemented yet

- Implement ITU Y.1731
  - Frame loss measurements
  - Frame delay measurements
More Information

- Mail to rvdp@sara.nl or nrg@sara.nl
- http://nrg.sara.nl/dot1ag-utils
- http://www.openflow.org/
- Please visit us at Dutch Research Consortium booth $642

Thanks to our partners