Inter-domain SDN Data Plane Validation: Next Steps at AmLight

Jeronimo Bezerra
Florida International University
<jab@amlight.net>

Marcos Schwarz
Rede Nacional de Ensino e Pesquisa
<marcos.schwarz@rnp.br>
Outline

• Introducing AmLight
• Troubleshooting production SDN networks
• Troubleshooting the Data Plane
• Inter-Domain Tracing Protocol
• Future Work
• Introducing the AmLight SDNTrace app
• Inter-Domain screenshots
**AmLight is a Distributed Academic Exchange Point**

- **Production** SDN Infrastructure since Aug 2014
- Responsible for the South America academic connectivity
- Carries Academic and Non-Academic/Commercial traffic
  - L2VPN, IPv4, IPv6, Multicast
- Supports Network Programmability/Slicing
  - OpenFlow 1.0
  - Flow Space Firewall for Network Programmability/Slicing
  - OGF Network Service Interface (NSI) enabled
- Currently, operating with more than 1k flows (production and experimentation)
- Web site: [www.sdn.amlight.net](http://www.sdn.amlight.net)

*With the SDN deployment, everything changed...*
Troubleshooting a production SDN network

- Troubleshooting a production environment has different requirements
  - It needs to be agile and least disruptive as possible
  - It might need historical information and understanding of traffic going through the network
  - Tools have to be handy!

- Legacy troubleshooting tools are partially useful or completely useless
  - OAM (Operation, Administration and Maintenance) is not supported by OpenFlow (yet)
  - Ping, traceroute, SNMP, Wireshark/Tcpdump are not made for OpenFlow networks

- Deep knowledge of the hardware and software platform is required:
  - Usage of the ”hidden” commands becomes part of your routine
Troubleshooting *Data Plane*?

- In some cases, everything looks fine, but traffic is not flowing
- Examples of data plane black holes:
  - A specific line card or interface discarding all traffic
    - Due to an interface memory issue, flows are installed but traffic is discarded
  - Interface down in one side but up in the remote and the SDN app doesn’t understand that
    - For instance: 10G LAN-PHY, Ethernet circuits and 100G long haul circuits
    - In this case, depending of the side, the SDN app installs the circuits pointing to the affected link, discarding all traffic
  - A specific installed flow entry crashed
    - Due to an interface memory issue, one specific flow is affected and traffic is discarded
    - Depending of the number of OpenFlow switches and flow entries, finding the problem might be extremely time-consuming
- In these cases, in-band tests are required:
  - Just a very few SDN apps test in-band per link
  - No SDN apps test in-band per flow
Data Plane Monitoring

• Monitoring individual flows is important but extremely expensive
  – Being proactive with all flows is desired but the interval between tests and number of flows need to be taken into consideration
  – Using a reactive approach is the best suggestion
    • Users won’t be happy, but your switches won’t crash

• Approaches to validate users’ flows are being proposed:
  “SDN traceroute: Tracing SDN Forwarding without Changing Network Behavior”
  “Multi-protocol Network Troubleshooting with Pathtrace protocol”

• AmLight’s developed a solution to trace users’ flows: SDNTrace
But, wait a minute! What about circuits that spans multiple domains?
Inter-domain Data Plane Troubleshooting

• Multi-domain virtual circuits are subject to problems in each domain they transverse
• Issues on links peering two domains are even more difficult to detect and troubleshoot:
  – Multiple NOCs, configuration inconsistencies, devices malfunction, unpredicted topology changes
• The legacy way: manually add an IP to each switch in the path, ping each switch until you isolate the issue
  – What about SDN? You don’t easily add an IP to an OpenFlow switch!
  – Lack of inter-domain tools, current efforts mainly on intra-domain

• Recent experience:
  – Two users (Brazil and UK)
  – Five domains in the path, including two OpenFlow-based (AmLight and Internet2)
  – 22 days & 45 e-mails to restore a single VLAN!

• Then we decided to work on this problem...
An inter-domain SDN data plane troubleshooting solution was created with the following initial requirements:

- User should not need to know the network topology or understand OpenFlow, just like a traditional traceroute
- Each domain in the path should be able to have different privacy policies
- The solution should not require topology or technology changes, just a few OpenFlow entries

Optional:
- The inter-domain trace protocol should be flexible enough to support different solutions of path trace
How does it work? (1) – Simple Version

- Step 1: Contract established between neighbor domains with the "color" of the peering switches.

<table>
<thead>
<tr>
<th>IP address</th>
<th>type</th>
<th>remote</th>
<th>label/color</th>
<th>interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>a.b.c.d:443</td>
<td>tracepath</td>
<td>domainA:switchA2</td>
<td>dl_src:111</td>
<td>p2</td>
</tr>
<tr>
<td>w.x.y.z:443</td>
<td>tracepath</td>
<td>domainB:switchB1</td>
<td>dl_src:110</td>
<td>p1</td>
</tr>
</tbody>
</table>

- Step 2: Each Controller Pushes the "colored" flows to the peering switches.
After a trace request:
Step 3: "Controller A" uses the contract and sends a probe matching the neighbor’s switch "color"
Step 4: Switch "A2" matches the probe with the inter-domain flow entry and forward it to "B1"
Step 5: Switch "B1" matches the probe with the "colored" flow and sends the probe to "Controller B"
Step 7: Controller B reports <"domain B", "switch B1", "port 1"> to Controller A
Step 8: Controller A forwards the report to the user

Step N: Domain B continues the intra-domain trace till the end of its domain. If there is a "Domain C" for such user circuit, process continues in the next domain.
How does it work? (3) – Full Version
AmLight SDNTrace

- Does not change user flow entries AT ALL
- Lightweight (2-4 flow entries needed per sw)
- Works with OpenFlow 1.0 and 1.3
- Based on Ryu SDN framework
- Trace flows from different SDN applications
- Trace User Flows based on Layer 2, Layer 3 or mix of layers
- Supports Inter-domain Tracing
- Beta code:
  - http://github.com/amlight/SDNTrace
SDNTrace running – Demo
Screenshots – Trace from RNP (left) to CLARA (right)
Screenshots – Trace from CLARA (right) to RNP (left)
Future

• Expand the solution to use the Network Service Interface (NSI) protocol
  • NSI provides models for describing network services and enables the use of shared resources through secure and reliable sessions for communication between domains
  • Used by GLIF AutoGOLE community for inter-domain L2 circuits provisioning
  • Supports authentication and encryption
  • NSI can be used to enforce the ”peering contract” and transport communication between controllers

• Deploy at AMPATH, AmLight, SouthernLight, ANSP and RNP in 2017

• Evolve to a more complex solution with alarms/triggers, etc.
AmLight’s Development Team:
• ANSP – Academic Network of Sao Paulo:
  • Antonio Francisco
  • Jorge Marcos
  • Rogerio Motitsuki
• RNP – Rede Nacional de Ensino e Pesquisa
  • Marcos Schwarz
• FIU – Florida International University
  • Jeronimo Bezerra

http://groups.geni.net/geni/wiki/GEC24Agenda/EveningDemoSession#Multi-protocolNetworkTroubleshootingwithPathtraceprotocol