Software Defined Internet Exchanges
A feasibility evaluation at the AMS-IX

Siem Hermans, Jeroen Schutrup
Research Question

What is the feasibility of transitioning the AMS-IX to an Industrial Scale Software Defined Internet Exchange Point?
The Amsterdam Internet Exchange*

*Not only situated in Amsterdam

- Providing peering services
  - Saves costs
  - Resilience
- Common shared Layer 2 Ethernet platform
- Built on top of MPLS/VPLS
Technical concepts

SDX basics

• BGP Traffic delivery
  • Routing on prefix
  • No end-to-end policies
  • Indirect policies

• SDX leverages OpenFlow
  • Fabric is perceived as a single entity

• Use Cases
  • Application specific peering
  • (D)DoS mitigation
  • Et cetera.

• Primarily helpful for inbound traffic engineering

\[ dIP \in \{P1, P2, P3, P4\} \land dPort=443 \rightarrow fwd(C) \]
Sounds familiar...
Sounds familiar...

RFC 5575 - Dissemination of Flow Specification Rules
Sounds familiar...

RFC 5575 - *Dissemination of Flow Specification Rules*

......so why not FlowSpec?
- Not transparent to the participant
- Adoption is limited due to ossification of the Internet
- Scalability issues at large scale
  - TCAM allocation for ACL / PBR rules is limited
Related Work

Sources

• Feamster et al. SDX: A Software Defined Internet Exchange". In: Open Networking Summit (2013)


(a) Number of Forwarding Table Entries.
Growth pattern

- Original paper tests up to 500 participants
- AMS-IX is continuously growing
- Growing closer towards 800 unique participants
- Scalability is an important factor for feasibility
Technical concepts

iSDX controller

- Traditional route server
- Every participant calculates its own forwarding entries
- Configuration conflicts are resolved by Refmon
Methodology
Controller enhancements

- **Enhancements**
  - Bypass the route server
  - Fixing program breaking bugs
  - Addition of Redis

- **Data set:** AMS-IX RIB dump
  - IPv4 ~150k unique prefixes
  - IPv6 ~17k unique prefixes
Methodology
Controller enhancements

• Enhancements
  - Bypass the route server
  - Fixing program breaking bugs (3)
  - Addition of Redis (4)

• Data set: AMS-IX RIB dump
  • IPv4 ~150k unique prefixes
  • IPv6 ~17k unique prefixes

• Limitations
  • iSDX requires multiple tables
  • Switch platform (MLXe)
    • OpenFlow (OF) 1.0 switch
    • NetIron 5.9, OF 1.3 compliant
  • No support for Virtual Chassis
  • Future: Brocade SLX
  • Fallback: Open vSwitch
Methodology
Test scenarios

Scenario #1 - Validation
• Up to four outbound policies for 10% of the total participants. Up to 800 peers.

Scenario #2 - Policy expansion
• Up to sixteen outbound policies for 10, 30 or 50% of the total participants. Up to 800 peers.

Scenario #3 - Granular policies
• Up to four prefix based outbound policies for 10% of the total prefixes. Up to 800 peers.
Results
Scenario #1 – Validation

- **Reproduction of results**
  - Matches original iSDX scalability findings
  - Linear growth pattern perceived as participants increase

- Maximum supported flows heavily dependent on switch platform
  - Brocade MLXe supports 128,000 flows per chassis

- New Brocade SLX platform
  - More capable Merchant Silicon (Broadcom Tomahawk, Jericho)
Results
Scenario #2 – Policy Validation

- Growth pattern
  - Similar growth pattern perceived as in Scenario #1
  - Amount of flows exceeds current hardware platform

*Scalability is heavily tied to constraints set by the IXP (Tolerated amount of policies, port ranges, et cetera.)*
Results
Scenario #3 – Granular policies

Impact

• Defining policies on destination prefix heavily impacts scalability

• Aggregation is possible but not performed by iSDX

• Total amount of policies for AMS-IX scale exceeds 140 million flow entries

• Exceeds capabilities of any current hardware platform
Technical concepts

iSDX Fabric

IXP Fabric (Virtual Chassis)

VMAC Translation

Main (in)
- M In Port = #1
- A SRC → ::01
- M In Port = #2
- A SRC → ::02

Main (out)
- M DST = ::01
- A DST → A
- M DST = ::02
- A DST → B

Outbound
- M SRC = ::01, TP_DST 80
- A DST → ::02

Inbound
- M SRC = ::01, TP_DST 22
- A → Drop
- M -
- A -
Results
MAC compression

iSDX on the fabric
- Abstracts ASes from ports
- Scales up to 28 ASes in one MAC
- Embeds Next-Hop ASes in MAC address
  - Overriding BGP behavior
  - iSDX design choice
Infrastructural impact

- **iSDX was designed for virtual chassis infrastructures** *(Brocade VCS, Cisco VSS/VPC, Juniper VC)*
- **AMS-IX has MPLS/VPLS multi-hop infrastructure**
  - Implementation is still feasible
  - OpenFlow pipeline on the edges
  - Normal MPLS traffic forwarding
  - MAC learning via VPLS infrastructure
Conclusion

• **Scalability**
  - Compression of flows has limitations
  - Defining fine-grained policies is still limited by hardware at this kind of scale
  - iSDX as a concept is feasible
  - Scalability is feasible if the AMS-IX heavily constrains boundaries
    • Affects neutrality of the IXP

• **Deployment impact**
  - Allows for gradual transition to iSDX design
  - iSDX can be deployed alongside current MPLS/VPLS infrastructure
Future work

Moving forward

- Rewrite controller software
  - Improve robustness
  - Include support for IPv6
  - Include multi-threading in the Fabric Manager (Refmon)

- More efficient policy distribution over PE switches
  - Allow for extended scalability in multi-hop configurations

- Include MPLS state in iSDX controller
  - Omit the need for a second lookup

- Include support for defining policies per port
  - Work in process:
    ENDEAVOUR project at the University of Louvain (prof. M. Canini)
Research Project 2
System and Network Engineering

Thank you
siem.hermans@os3.nl
jeroen.schutrup@os3.nl
github.com/jeroen92/sdx-ixp