Leveraging Dataplane Disaggregation to Improve Testing Pipeline

Mark Cannon, President
Rich Renner, CTO
One Source Integrations, LLC
Acknowledgements

Blackbox Testing of Stratum-Enabled Switches

ONF Connect 2018
Dec 5, 2018
Alireza Ghaffarkhah, Tomek Madejski, Waqar Mohsin, Konstantin Weitz, Rob Shakir, and more
Google Inc

https://vimeo.com/307167637
Problem Statement

Cost efficient, automated testing continues to be a major challenge for Network Operators.

- Vendor-specific testing is not scalable; even for single box testing.
- Integration / Solution testing compounds the problems exponentially.
Goals

1. Advancing vendor-agnostic, “black box” testing

1. Topology-agnostic, “multi-box” testing

1. Automated Testing

1. Virtual Testing for cost-efficiency
Test Vectors are a set of Test Cases.
Each Test Case is defined as...
  ○ Set of Operations
  ○ Set of External Stimuli
  ○ Set of Expected Behaviors
Our Approach

Vendor-agnostic, “black box” testing with Test Vectors

- Leveraging Test Vectors
- Injecting network traffic into Test Cases for realistic External Stimuli use cases
- Initial and iterative rounds of testing performed in an orchestrated, virtual network environment
“Black Box” Workflow - Vendor Agnostic

1. Test Vector feeds
   a. Topology orchestration
   b. Traffic profile generation
   c. Ordered operations

1. Any additional external stimuli are applied

1. Exhibited behavior is compared to Expected Behavior
Demo 1 - “Black Box”
Adding Traffic to the Testing Pipeline

- Black box - Loopback TC
  Test case with expected packet that goes in port 1 and exits port 2
  - Show Topology
    - Traffic endpoint
    - Test Framework node
    - Stratum BMv2 container (black box)
  - Visual here is showing p4 code example
  - Compile / Execute
  - Visual state verification (before and after)
    - Live subscribe
    - traffic/grafana
Introduction - Test Matrices

- Test Matrices are a set of Test Cases designed to validate a multi-box topology.

- Each Test Case is defined as...
  - Set of Operations
  - Set of External Stimuli
  - **A Set of Expected Behaviors across multiple devices**

- Set of Expected Behaviors is fed into a database and saved for comparison and validation
Our Approach

Vendor-agnostic, “multi-box” testing with Test Matrices

- Leveraging Test Matrices
- Injecting network traffic into Test Cases for realistic External Stimuli use cases
- Initial and iterative rounds of testing performed in an orchestrated, virtual network environment
“Multi-Box” Workflow - Topology Agnostic

Initial State Validation
Ex; BGP Process Restart

1. Test Matrix feeds “Initial State”
   a. Multi-box topology orchestration
   b. Traffic profile generation
   c. Ordered operations
   d. “Initial State recorded”

2. Any additional external stimuli are applied

3. “Exhibited State” is compared to “Initial State”

4. Test Case can be re-run on any topology.
Multi-box - Forwarding TC

Test case with expected packet that enters switch 1 port 1, exits switch 1 port 2 and then enters switch 2 port 1 and exits switch 2 port 2
- Show Topology
  - Traffic endpoint
  - Test Framework node
  - Stratum BMv2 container (switch1)
  - Stratum BMv2 container (switch2)
- Visual here is showing p4 code example
- Compile / Execute
- Visual state verification (before and after)
  - Live subscribe
  - traffic/grafana
“Multi-Box” Workflow - Topology Agnostic

**Desired State Validation**
Ex: Link Failure, Fast Re-Route Validation

1. Test Matrix feeds “Initial State”
   a. Multi-box topology orchestration
   b. Traffic profile generation
   c. Ordered operations
2. Any additional external stimuli are applied
3. Snapshot of “Desired State”
4. Exhibited State is compared to “Desired State”
5. Test Matrix can be re-run on any topology (requires new State Captures)
Recap

1. Demonstrated adding traffic and virtualization to accelerate and reduce cost of “black box” testing.

1. Demonstrated methodologies for automating topology-agnostic, “multi-box” solution testing.
Thank You

Any Questions?

For further information please contact us at info@osi.io
Backup
The End-to-End Test Framework

- Vendor provided TVs
- Network Model
- Test framework
- gRPC
- SUT (single blackbox switch)
- Traffic generators and validators (IXIA, commodity server, etc)

P4 specs & P4Info
- Topology info
- SDN Controller Trace
- Handcrafted TVs
- TVS
- Test Vector Generator (TVG)
- Repo shared with vendors

Cloud or dedicated machines