MODEL BASED TESTING
TECHNIQUES FOR SOFTWARE DEFINED NETWORKS

ASMA BERRIRI

UNDER THE SUPERVISION OF:
PR. DJAMAL ZEGHLACHE & DR. NATALIA KUSHIK

UNIVERSITÉ PARIS-SACLAY
TÉLÉCOM SUDPARIS
SAMOVAR, CNRS

INSTITUT MINES-TÉLÉCOM
1. Motivation & Research Questions

2. Contributions
   • 2.1 Model Based Testing for SDN Architectures: A Graph / Path Enumeration based Approach
   • 2.2 Test Derivation for SDN-enabled Switches: A Logic Circuit based Approach
   • 2.3 Test Generation for OpenFlow Switches: An Extended Finite State Machine based Approach
   • 2.4 Test Derivation for a Controller Application: An Adaptation of the Logic Circuit based Approach

3. Conclusion and Future Work
Motivation & Research Questions
We focus on

- Model Based Testing for SDN architectures and their components
- Test Generation

**Research Question 1:** How to assure the correct behaviour of architectures?

**Research Question 2:** How to assure the correct forwarding behaviour of an SDN-enabled switch in the data plane?

**Research Question 3:** How to assure the correct behaviour of an SDN switch in its interaction and interfacing with a controller?

**Research Question 4:** How to assure the correct behaviour of the module of a controller responsible for translating, for a given switch $S$, requests into corresponding ports of $S$?
Estimate the fault coverage of a TS?

Mutation Analysis

Fault model $FM = <S, @, FD>$
- $S$: specification
- $@$: conformance relation
- $FD$: fault domain

Goal: w.r.t. $FM$, a $TS$ that is
- **Exhaustive**: $\forall I \in FD, I @ S$ is detected by $TS$
- **Complete**: $\forall I_1, I_2 \in FD, I_1 @ S$ passes $TS$ while $I_2 @ S$ fails $TS$
Contribution-I
Model Based Testing for SDN Architectures
A Graph / Path Enumeration based Approach

Contribution-II
Test Derivation for SDN-enabled Switches
A Logic Circuit based Approach

Contribution-III
Test Generation for OpenFlow Switches
An Extended Finite State Machine based Approach

Contribution-IV
Test Derivation for a Controller Application
An Adaptation of the Logic Circuit based Approach
Test the entire SDN architecture to check implemented network topologies conform to the requested ones.

- Proposed $FM = \langle=, FD\rangle$ where $FD$ containing the network topologies and the conformance relation is the "equality"

- Proposed a novel test generation approach based on path enumeration

- Deriving complete test suites w.r.t. $FM$, for detecting inconsistencies

Derive **test sequences** that guarantee the desired **fault coverage** by formally modelling the switch via **Logic Circuit (LC)**

Test the **forwarding functionality** of the switch as a “**stateless system**”

A specification $S$ as a **set of rules** installed in the switch

- Proposed $FM = <S, =, FD>$

- Proposed an algorithm for obtaining a **LC** simulating $S$

- Proposed **novel test generation approaches**
  - Active
    - Deriving **exhaustive/complete TSs** w.r.t. $FM$ for detecting inconsistencies
  - Passive (run-time verification)

Test Generation for OpenFlow Switches: An Extended Finite State Machine Based Approach

Derive test sequences that guarantee the desired fault coverage by formally modelling the switch via Extended Finite State Machine (EFSM)

Test the switch in its interaction with the controller as a “stateful system”

- Proposed an EFSM model (S) derived from the OpenFlow requirements
- Proposed FM = <S, ≃, FD>
- Proposed a novel test generation approach
- Deriving exhaustive TSs w.r.t. FM for detecting inconsistencies

Asma Berriri et al. (2019). “Extended Finite State Machine based Test Generation for an OpenFlow Switch”. working paper or preprint. url: https://hal.archives-ouvertes.fr/hal-02262841
Derive **test sequences** that guarantee the desired **fault coverage** by adapting the **LC** based approach (Contribution-II)

**Test a module of the controller:** the **Link_Translator**

A specification $S$ as a set of mappings w.r.t. a switch $s$

- Proposed an algorithm for obtaining a **LC** simulating $S$
- Proposed the adaptation of **LC based test generation approach**
- Proposed an algorithm for test execution

- Given a requested “topology” w.r.t. a given switch $s$, the problem is to check that the **Link_Translator** translates it into the correct pair of ports of $s$
- **Link_Translator** has prior knowledge about the data plane
EVALUATION

Experimental Set Up

- Experiments have been carried out in virtualized environments using
  - Open vSwitch (different versions)
  - Different SDN controllers

- A number of software tools have been developed

The experimental results have shown

<table>
<thead>
<tr>
<th>Contribution</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contribution I</td>
<td>✓ The effectiveness of the proposed approaches in detecting implementation bugs in the SDN architecture under test &lt;br&gt; ✓ The efficiency/’power’ of the derived TSs</td>
</tr>
<tr>
<td>Contribution II</td>
<td>✓ The high fault coverage of the TSs, derived based on LC fault models, for SDN-enabled switch faults &lt;br&gt; ✓ The effectiveness of the run-time verification approach for switches</td>
</tr>
<tr>
<td>Contribution III</td>
<td>✓ The effectiveness of the proposed EFSM based approach comparing it to a random generation approach as a baseline &lt;br&gt; ✓ Its effectiveness in detecting switch-to-controller interaction errors/bugs</td>
</tr>
</tbody>
</table>
CONCLUSION AND FUTURE WORK

Thesis Contributions: Novel Model Based Testing Approaches for SDN Architectures and their Components

► Advance the state of the art in the field
► Have been shown to be efficient in detecting errors
► Allow to provide guarantees about the quality of tests

Future Work & Perspectives

► Extending fault models
► Optimizing test generation and test execution
► Estimating the complexity of the proposed algorithms
► Enhancing experiments
Thank you for your attention!

Questions?..