VeriFlow: Verifying Network-Wide Invariants in Real Time

Ahmed Khurshid, Wenxuan Zhou, Matthew Caesar, P. Brighten Godfrey
University of Illinois at Urbana-Champaign
Modern networks are complex.
Motivation

Modern networks are complex.
Modern networks are complex.
Modern networks are complex.

E.g., Loops, Black holes, Security Violations, ...
Motivation

Modern networks are complex.

E.g., Loops, Black holes, Security Violations, ...

Monday, August 13, 2012
Motivation

Modern networks are complex.

Serious consequences!

E.g., Loops, Black holes, Security Violations, ...

Monday, August 13, 2012
Modern networks are complex.

Serious consequences!

E.g., Loops, Black holes, Security Violations, ...

Debugging the data plane

- Diagnose problems as close as possible to actual network behavior
- Data plane is a “narrower waist” than configuration
What if we can detect bugs in real time? (~1 ms)

- Provide immediate warning
- Block dangerous changes
Motivation

What if we can detect bugs in real time? (~1 ms)

- Provide immediate warning
- Block dangerous changes

Is it possible to check network-wide invariants in real time as the network evolves?
Motivation

Challenge 1: Obtaining real time view of network

• Solution: interpose between SDN controller and devices
Motivation

Challenge 1: Obtaining real time view of network

• Solution: interpose between SDN controller and devices

Challenge 2: Verification speed

• Solution: Formal methods?
Challenge 1: Obtaining real time view of network

- Solution: interpose between SDN controller and devices

Challenge 2: Verification speed

- Solution: Formal methods? No, too slow!

Anteater, Mai, Khurshid, Agarwal, Caesar, Godfrey, and King. (SIGCOMM'11)
ConfigChecker, Al-Shaer, Marrero, El-Atawy, and ElBadawi. (ICNP 09)
HSA, Kazemian, Varghese, and McKeown. (NSDI'12)
Outline

• Motivation
• Design
• Evaluation
• Conclusion
Our Approach: VeriFlow

SDN Controller

VeriFlow

Diagram showing the interaction between SDN Controller and VeriFlow.
Our Approach: VeriFlow

- SDN Controller
  - Monitor all updates!

VeriFlow

Diagram showing the connection between SDN Controller and VeriFlow, indicating the monitoring of updates.
Our Approach: VeriFlow

Monitor all updates!
Our Approach: VeriFlow

Monitor all updates!
Overview

VeriFlow

Generate Equivalence Classes

Updates
Overview

VeriFlow

Generate Equivalence Classes

Generate Forwarding Graphs

Updates
Overview

VeriFlow

- Generate Equivalence Classes
- Generate Forwarding Graphs

Updates

Monday, August 13, 2012
Overview

VeriFlow

Generate Equivalence Classes

Generate Forwarding Graphs

Run Queries

Updates
1. Limit the Search Space

Updates

VeriFlow
1. Limit the Search Space

VeriFlow

Generate Equivalence Classes

Updates
1. Limit the Search Space

VeriFlow

Updates

Generate Equivalence Classes

Equivalence class: Packets experiencing the same forwarding actions throughout the network.
1. Limit the Search Space

VeriFlow

Generate Equivalence Classes

Equivalent class: Packets experiencing the same forwarding actions throughout the network.

Updates

Fwd’ing rules
1. Limit the Search Space

VeriFlow

*Generate Equivalence Classes*

*Equivalence class:* Packets experiencing the same forwarding actions throughout the network.

Fwd’ing rules: 0.0.0.0/1

Updates
1. Limit the Search Space

VeriFlow

Generate Equivalence Classes

Equivalence class: Packets experiencing the same forwarding actions throughout the network.

Updates

Fwd’ing rules

0.0.0.0/1  64.0.0.0/3
1. Limit the Search Space

**VeriFlow**

**Generate Equivalence Classes**

*Equivalence class:* Packets experiencing the same forwarding actions throughout the network.

- **Updates**
- **Fwd’ing rules**
- **Equiv classes**

- 0.0.0.0/1
- 64.0.0.0/3

Monday, August 13, 2012
1. Limit the Search Space

**VeriFlow**

**Generate Equivalence Classes**

**Equivalence class**: Packets experiencing the same forwarding actions throughout the network.

Updates

Fwd’ing rules
Equiv classes

0.0.0.0/1
64.0.0.0/3

Find only equivalence classes affected by the update using a trie-based data structure

Monday, August 13, 2012
2. Represent Forwarding Behavior

VeriFlow

Generate Equivalence Classes
2. Represent Forwarding Behavior

VeriFlow

Generate Equivalence Classes

Updates
2. Represent Forwarding Behavior

VeriFlow

Generate Forwarding Graphs

Generate Equivalence Classes

Updates
2. Represent Forwarding Behavior

VeriFlow

Generate Equivalence Classes

Generate Forwarding Graphs

Updates

Forwarding graphs:
2. Represent Forwarding Behavior

VeriFlow

Generate Equivalence Classes

Generate Forwarding Graphs

Updates

Forwarding graphs:

All the info to answer queries!
3. Run Graph Alg. to Check Invariants

VeriFlow

Generate Equivalence Classes

Generate Forwarding Graphs

Updates
3. Run Graph Alg. to Check Invariants

VeriFlow

- Generate Equivalence Classes
- Generate Forwarding Graphs

Updates
3. Run Graph Alg. to Check Invariants

VeriFlow

Generate
Equivalence
Classes

Generate
Forwarding
Graphs

Run
Queries

Updates
Reachability Queries:
- Black holes,
- Routing loops,
- Isolation of multiple VLANs,
- Access control policies,
...

General Queries
3. Run Graph Alg. to Check Invariants

- Updates
  - Generate Equivalence Classes
  - Generate Forwarding Graphs
  - Run Queries

VeriFlow
3. Run Graph Alg. to Check Invariants

VeriFlow

Generate
Equivalence
Classes

Generate Forwarding
Graphs

Run Queries

Updates

Diagram showing the flow of processes:
- Updates
- Generate Equivalence Classes
- Generate Forwarding Graphs
- Run Queries
3. Run Graph Alg. to Check Invariants

VeriFlow

Generate Equivalence Classes

Generate Forwarding Graphs

Run Queries

Updates

Good rules

Monday, August 13, 2012
3. Run Graph Alg. to Check Invariants

VeriFlow

Generate Equivalence Classes

Generate Forwarding Graphs

Run Queries

Updates

Good rules

Bad rules

Monday, August 13, 2012
3. Run Graph Alg. to Check Invariants

VeriFlow

- Generate Equivalence Classes
- Generate Forwarding Graphs
- Run Queries

Diagram:
- Updates
- Good rules
- Bad rules

Diagnosis report:
- Type of invariant violation
- Affected set of packets
Outline

• **Motivation**
• **Design**
• **Evaluation**
• **Conclusion**
Evaluation Setup

A new experiment not in the paper [with Kelvin Zou]

- Mininet OpenFlow network
- 172 switches, 172 hosts
- NOX controller, learning switch app
- TCP connections between random pairs of hosts
A new experiment not in the paper [with Kelvin Zou]

- Mininet OpenFlow network
- 172 switches, 172 hosts
- NOX controller, learning switch app
- TCP connections between random pairs of hosts
Microbenchmark Runtime

CDF

Graph cache update
Equivalence class search
Graph build
Query
Total verification

Microseconds
Microbenchmark Runtime

CDF

Graph cache update
Equivalence class search
Graph build
Query
Total verification

Microseconds

Monday, August 13, 2012
99% of updates verified within 200 μs
Conclusion

VeriFlow achieves real-time verification

- A layer between SDN controller & network devices
- Rigorous checking within hundreds of μs