Declarative Network Path Queries

Srinivas Narayana
Princeton University
Management = Measure + Control

Network Operators

Measure

Control

Software-Defined Networking (SDN)
Example: Where’s the Packet Loss?

Suspect: Faulty network device(s) along the way.
Example: Where’s the Packet Loss?

Idea: “Follow” the path of packets through the network.

ACL counters

100 pkts

Packet rewrite

25 pkts 😞
Example: Where’s the Packet loss?

Complex, Inaccurate Join Procedure with multiple datasets: traffic, forwarding, topology

High Overhead of collecting (unnecessary) data to answer a given question
Example: Where’s the Packet loss?

Complex, Inaccurate Join Procedure with multiple datasets: traffic, forwarding, topology

High Overhead of collecting (unnecessary) data to answer a given question
Lots of Measurement Use-Cases

- Traffic matrix
- Uneven load balancing
- DDoS source identification
- Port-level traffic matrix
- Congested link diagnosis
- Slice isolation
- Loop detection
- Middlebox traversal order
- Incorrect NAT rewrite
- Firewall evasion
- ...

Resource management
Policy enforcement
Problem diagnosis
Solution Approach

Path Query System

Declarative Query Specification
- Forwarding Independence
- Co-Measurement Independence
- Hardware Independence

Query-Driven Measurement
- Accurate Answers
- Pay Exactly For What You Query
- Commodity ("Match-Action") Hardware

Path Query Language

Query Run-Time System
Solution Approach

1. Path Query Language

Query Expressions → Forwarding → Statistics

2. Query Run-Time System

SDN controller

Payloads → Statistics
Path Query Language

• Single location tests:
  srcip=10.0.0.1
  inport=3

• Regular expression operators to combine tests:
  sw=1 ^ sw=4
  srcip=A ^ true* ^ dstip=B'
  ingress() ^ ~(sw=firewall)* ^ egress()

• SQL-like grouping operators for aggregation
  in_group(ingress(), [sw])
  ^ true*
  ^ out_group(egress(), [sw])
Query Run-Time System

- srcip=a ^ dstip=B’

- Each packet carries its own DFA state
  - Packet satisfies query iff it reaches accepting states
Evaluation

• Prototype on Pyretic + NetKAT + OpenVSwitch

• Queries: traffic matrix, DDoS detection, per-hop packet loss, firewall evasion, slice isolation, congested link

• Metrics and results on Stanford (all queries together):
  • Compile time: 5 seconds
  • # Rules: ∼ 650
  • # State bytes: 2 bytes
Queries? 😊

http://www.cs.princeton.edu/~narayana/pathqueries
Are you the Seminar organizer asking a question because no one else is and the awkward silence is making everyone uncomfortable?

Yes

Thank God. Please ask the question and let's get out of here!

No

Ok, you have a legitimate question. Do you actually care about the answer?

Yes!

Not really, I just want to show off.

No

FINE, ASK YOUR QUESTION.