Answering Why-Not Queries in SDNs with Negative Provenance

Yang Wu†‡, Andreas Haeberlen†, Wenchao Zhou§, Boon Thau Loo†
†University of Pennsylvania, §Georgetown University, ‡Student, ¶Presenter

1 What is the problem?
It is hard to debug the absence of events with existing “network debuggers”.

SDN can have bugs
- Examples: race conditions, logic errors
- Distributed execution brings more subtleties.

Existing debuggers produce backtraces
- This is called ‘positive provenance’.
- Examples: ndb (HotSDN ’12), SNP (SOSP ’11)

↑ SNP outputs a positive provenance tree.

Problem: A positive symptom is required as a starting point for backtrace.

Internet DNS and HTTP requests
DNS Server
HTTP Server
Admin
SDN Controller
↓ Faulty flow entry
§ Strings
... Why does the HTTP server not get any requests?

2 Our approach

We can ask Why-Not queries. The answer is generated using Negative Provenance.

How do Why-Not queries help?
- Asking “ Why did something happen? ” does not help when explaining absence of events.
- Asking “ Why didn’t something happen? ” directly explains absence of events.
- Positive provenance → negative provenance

↑ The negative provenance answer for the scenario on the left.

Key challenges
- There are infinitely many negative events!
- Even finding a consistent definition is hard.
- Naïve explanation is too verbose to be useful (see the huge tree on the right).

3 How does it work?
We can use counterfactual reasoning to generate negative provenance.

Generate negative provenance
- Counter-factual reasoning. Find all the ways in which a missing event could have occurred, and show why each of them did not happen.

Simplify explanations as much as possible
- Program analysis. Most explanations are inconsistent with the controller program. Basically, only a few relevant rules could be produced under very specific conditions.

↑ Without simplification and summarization, the tree on the left actually looks like this.

Summarize naïve answer
- Heuristics to summarize naïve answer: e.g. hide transient messages from admins.

Status
- We have a running prototype that can answer concrete Why-Not queries in SDNs.
- Supports programs written in Pyreic (NSDI ’13).

Please ask me for details!
To appear at HotNets-XII.

This work is supported in part by NSF grants CNS-0845552, CNS-1040672, CNS-1054229, and CNS-1065130.