Pinpointing Delay and Forwarding Anomalies Using Large-Scale Traceroute Measurements

Romain Fontugne\textsuperscript{1}, Emile Aben\textsuperscript{2}, Cristel Pelsser\textsuperscript{3}, Randy Bush\textsuperscript{1}
November 1, 2017

\textsuperscript{1}IIJ Research Lab, \textsuperscript{2}RIPE NCC, \textsuperscript{3}University of Strasbourg / CNRS
Understanding Internet health?
Understanding Internet health?
Understanding Internet health?
Understanding Internet health? (Problems)

Manual observations

- Traceroute / Ping / Operators' group mailing lists
- Slow process
- Small visibility
Understanding Internet health? (Problems)

Manual observations

- Traceroute / Ping / Operators' group mailing lists
- Slow process
- Small visibility

→ Our goal: Systematically pinpoint network disruptions

- Delay changes
- Forwarding anomalies (not covered here, see the paper)
Silly solution: frequent traceroutes to the whole Internet!

→ Doesn’t scale
→ Overload the network
Better solution: mine results from deployed platforms

→ Cooperative and distributed approach
→ Using existing data, no added burden to the network
Actively measures Internet connectivity

- Multiple types of measurement: ping, traceroute, DNS, SSL, NTP and HTTP
- 10 000 active probes!
- Data for numerous measurements is made publicly available
Two repetitive large-scale measurements

- **Builtin**: traceroute every 30 minutes to all DNS root servers ($\approx 500$ server instances)
- **Anchoring**: traceroute every 15 minutes to 189 collaborative servers

Analyzed dataset

- May to December 2015
- 2.8 billion IPv4 traceroutes
- 1.2 billion IPv6 traceroutes
Monitor delays with traceroute?

Challenges:

- Noisy data
- Traffic asymmetry
- Packet loss
Monitor delays with traceroute?

Traceroute to “www.target.com”

```
~$ traceroute www.target.com
traceroute to target, 30 hops max, 60 byte packets
1  A  0.775 ms  0.779 ms  0.874 ms
2  B  0.351 ms  0.365 ms  0.364 ms
3  C  2.833 ms  3.201 ms  3.546 ms
4 Target  3.447 ms  3.863 ms  3.872 ms
```
What is the RTT between B and C?

Differential RTT: $\Delta_{CB} = RTT_C - RTT_B \equiv RTT_{CB}$
What is the RTT between B and C?

\[ RTT_C - RTT_B = RTT_{CB} ? \]

- No!
- Traffic is asymmetric
- \( RTT_B \) and \( RTT_C \) take different return paths!
What is the RTT between B and C?

- No!
- Traffic is asymmetric
- $RTT_B$ and $RTT_C$ take different return paths!
- Differential RTT: $\Delta_{CB} = RTT_C - RTT_B = d_{BC} + e_p$
Problem with differential RTT

Monitoring $\Delta_{CB}$ over time:

→ Delay change on BC? CD? DA? BA???
Proposed Approach: Use probes with different return paths

Differential RTT: $\Delta_{CB} = x_0$
Proposed Approach: Use probes with different return paths

Differential RTT: \( \Delta_{CB} = \{x_0, x_1\} \)
Proposed Approach: Use probes with different return paths

Differential RTT: \( \Delta_{CB} = \{x_0, x_1, x_2, x_3, x_4\} \)
Proposed Approach: Use probes with different return paths

Differential RTT: \( \Delta_{CB} = \{x_0, x_1, x_2, x_3, x_4\} \)

Median \( \Delta_{CB} \):
- Stable if a few return paths delay change
- Fluctuate if delay on BC changes
Median Diff. RTT: Example

Tier1 link, 2 weeks of data, 95 probes:

- Stable despite noisy RTTs
- Normally distributed
- Conf. interval: Wilson score
- Normal ref.: exp. smooth.
Detecting Delay Changes

Significant RTT changes:
Confidence interval not overlapping with the normal reference
Results

Analyzed dataset

- Atlas *builtin/anchoring* measurements
- From May to Dec. 2015
- Observed 262k IPv4 and 42k IPv6 links

We found a lot of delay changes!
Let’s see only two prominent examples
Case study: DDoS on DNS root servers

Two attacks:

- Nov. 30th 2015
- Dec. 1st 2015

Almost all server are anycast

- Congestion at the 531 sites?
- Found 129 instances altered by the attacks
Observed delay changes

- Certain servers are affected only by one attack
- Continuous attack in Russia
Unaffected root servers

Very stable delay during the attacks

- Thanks to anycast!
- Far from the attackers
Congested links for servers F, I, and K

→ Concentration of malicious traffic at IXPs
Case study: Telekom Malaysia BGP leak

Australia's internet hit hard by massive Malaysian route leak

Telekom Malaysia apologises for BGP bungle.

Global Collateral Damage of TMnet leak

Earlier today a massive route leak initiated by Telekom Malaysia (AS4788) caused significant network problems for the global routing system. Primarily affected was Level3 (AS3549 - formerly known as Global Crossing) and their customers. Below are some of the details as we know them now.

Starting at 08:43 UTC today June 12th, AS4788 Telekom Malaysia started to announce about 119,000 of prefixes to Level3 (AS3549, the Global crossing AS), whom in turn accepted.
Case study: Telekom Malaysia BGP leak
Case study: Telekom Malaysia BGP leak

Diagram:
- Google
- TM
- Level3
- Level3 (GBLX)
- Seattle
- London
Case study: Telekom Malaysia BGP leak
Case study: Telekom Malaysia BGP leak

Not only with Google... but about 170k prefixes!
Congestion in Level3

Rerouted traffic has congested Level3 (120 reported links)

- Example: 229ms increase between two routers in London!
Congestion in Level3

Reported links in London:

→ Traffic staying within UK/Europe may also be altered
Summary

Detect and locate delay and forwarding anomalies in billions of traceroutes

- Non-parametric and robust statistics
- Diverse root causes: remote attacks, routing anomalies, etc...
- Give a lot of new insights on reported events

Online detection for network operators

- http://ihr.iiijlab.net/