

# Network verification: Lessons learned and outlook

Ratul Mahajan



**June 15, 2020 T-Mobile  
Network Outage Report**

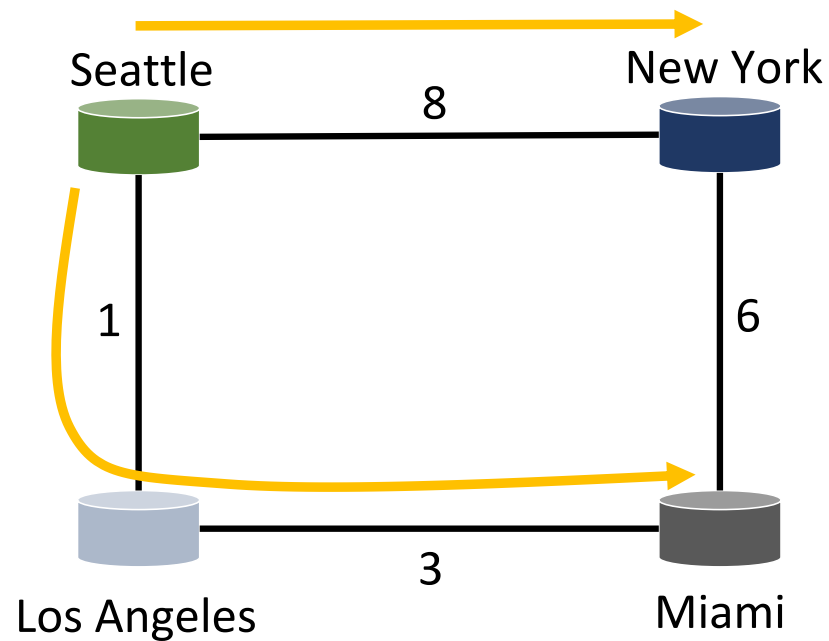
**PS Docket No. 20-183**

A Report of the Public Safety and Homeland Security Bureau  
Federal Communications Commission  
October 22, 2020

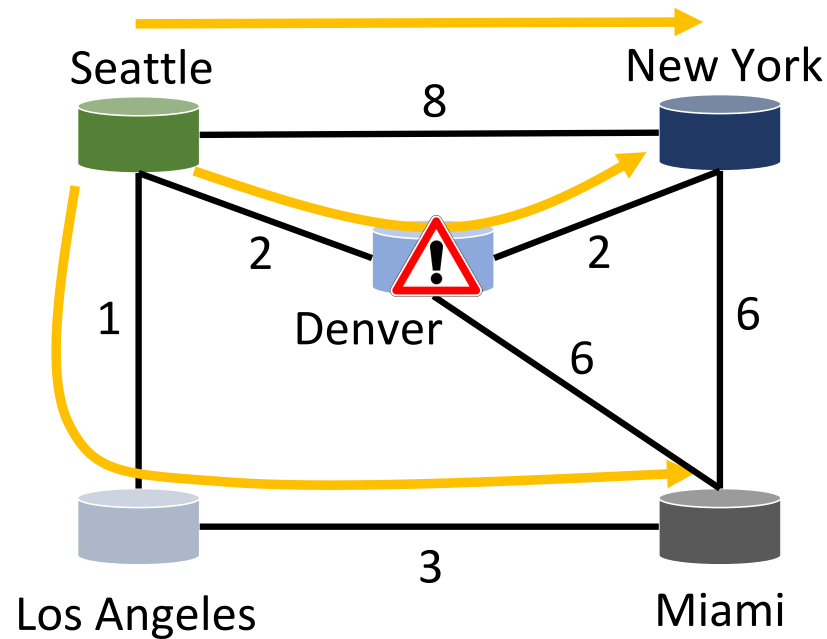
“At least 41% of all calls that attempted to use T-Mobile’s network during the outage failed, including at least 23,621 failed calls to 911.”

“[An old woman] who has dementia, could not reach [her son] after her car would not start and her roadside-assistance provider could not call her to clarify her location; she was stranded for seven hours”

# Anatomy of the outage (illustration)

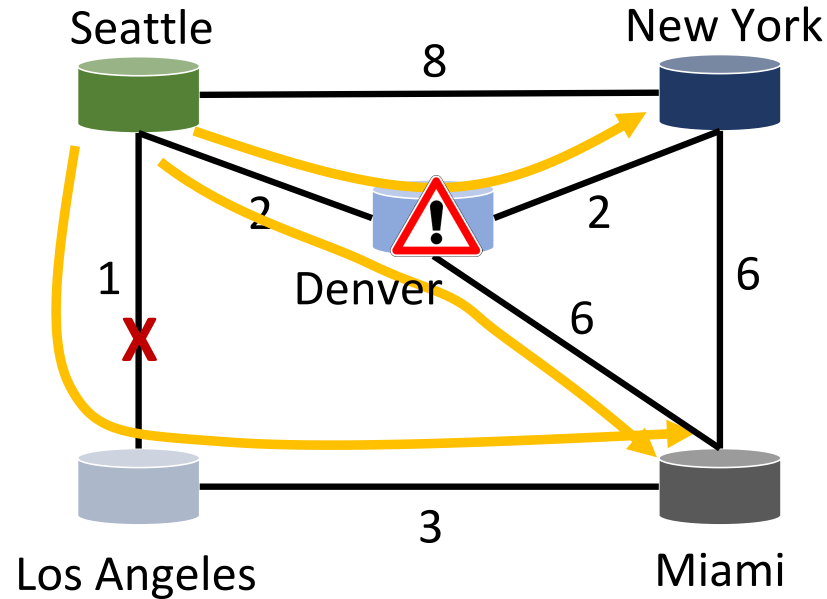


# Anatomy of the outage (illustration)



# Anatomy of the outage (illustration)

What if T-Mobile could guarantee that no traffic will transit Denver?



What if T-Mobile could predict the impact of link failure?

## Microsoft Says Config. Change Caused Azure Outage

Standard protocol for applying changes was not followed

## Microsoft: Misconfigured Network Device Caused Outage

Microsoft suffers intermittent Azure outage over DNS resolution issues

Microsoft 365 and Teams, Dynamics, SharePoint Online, OneDrive and Xbox Live among those affected

May 03, 2019 By: Sebastian Moss

## Google cloud is down, affecting numerous applications and services



Chad Fullerton  
@chad\_fullerton

Google Cloud outage appears to be outside of North America too, according to [DownDetector.com](https://downdetector.com) - reports in UK, France, Austria,

## With Confidence In AWS Shaken, Who Could Benefit?

Amazon.com, Inc. (NASDAQ: AMZN) faced a setback Tuesday due to an outage at its cloud computing platform — Amazon Web Services, or AWS....

[benzinga.com](https://benzinga.com)

## Google details 'catastrophic' cloud outage events: Promises to do better next time

Data-center automation software was behind what Google describes as a 'catastrophic failure' last Sunday.

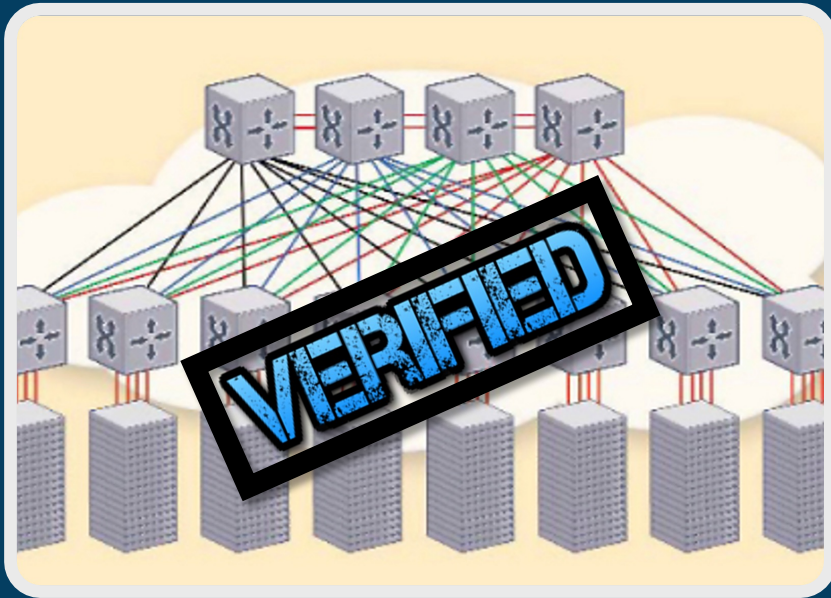
By Liam Tung | June 7, 2019 -- 12:39 GMT (05:39 PDT) | Topic: Cloud

## Amazon's massive AWS outage was caused by human error

One incorrect command and the whole internet suffers.

By Jason Del Rey | @DelRey | Mar 2, 2017, 2:20pm EST

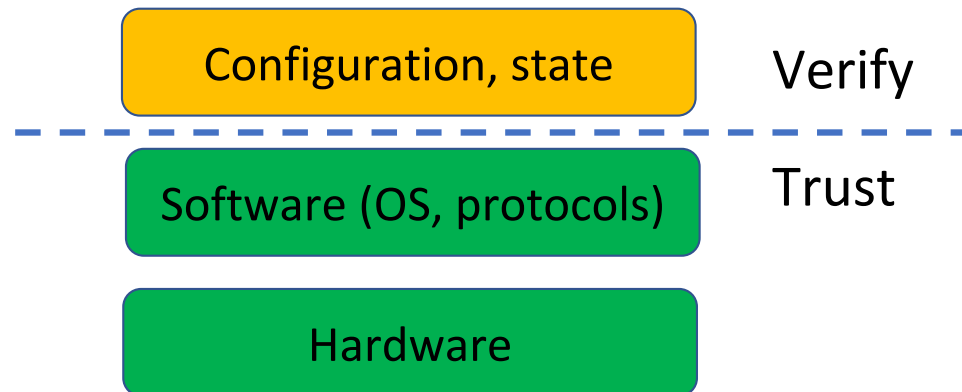
# Network verification to the rescue



Guarantee network behavior<sup>\*</sup>

<sup>\*</sup> Some behaviors under some assumptions

# How network verification slices the problem



# The “haystack” of network behaviors is HUGE

## **Large scale**

---

$O(10^3)$  devices  
 $O(10^6)$  routes  
 $O(10^9)$  packets

## **Complex interactions**

---

Distributed routing  
Protocol redistribution  
Rich route filtering

# Batfish: A production-grade network verifier



2012



2015



2017



2022



Op  
Us  
The  
Fo

(NSDI 2015)

## **A General Approach to Network Configuration Analysis**

*Ari Fogel Stanley Fung Luis Pedrosa Meg Walraed-Sullivan*

*Ramesh Govindan Ratul Mahajan Todd Millstein*

University of California, Los Angeles University of Southern California Microsoft Research

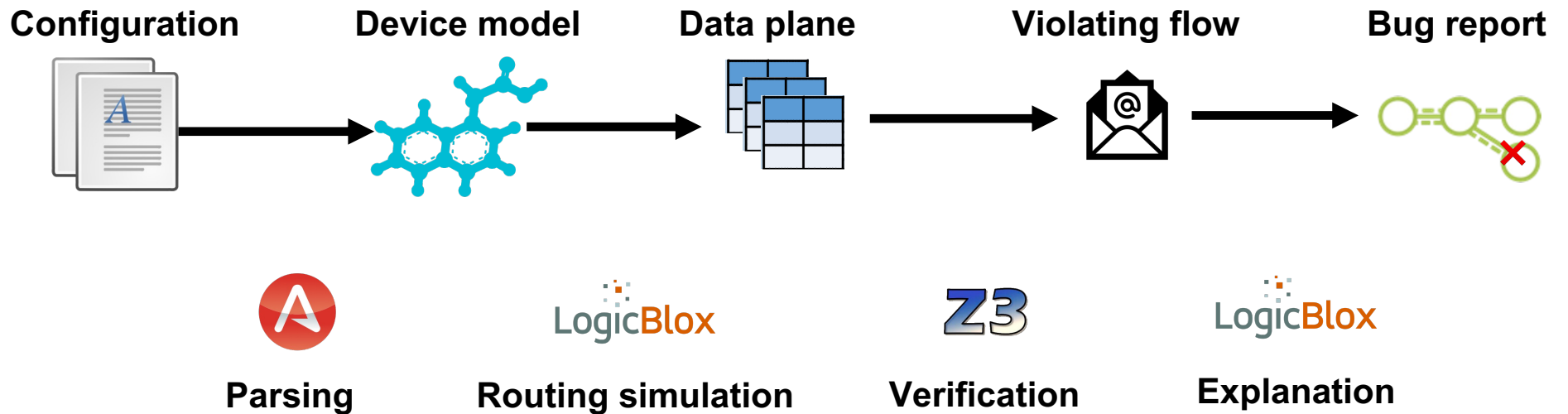
<https://github.com/batfish/batfish>

# Batfish does proactive network verification

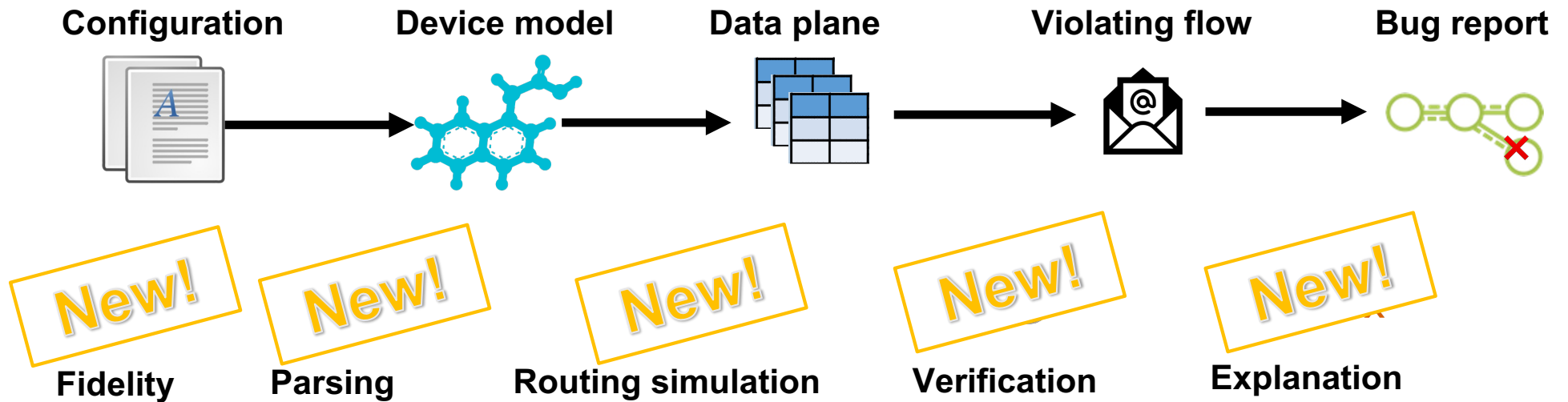


Verify configuration changes ***before*** they affect the network

# Batfish's original 4-stage pipeline



# Batfish's **2023** 4-stage pipeline



**1500x faster, 400x larger networks**



(SIGCOMM 2023)

## **Lessons from the evolution of the Batfish configuration analysis tool**

**Matt Brown**  
Intentionet

**Ari Fogel**  
Intentionet

**Daniel Halperin**  
Intentionet

**Victor Heorhiadi**  
Intentionet

**Ratul Mahajan**  
Intentionet  
University of Washington

**Todd Millstein**  
Intentionet  
UCLA

# Lesson 1: Datalog was great for prototyping, but not for production use

Three key challenges:

1. **Expressiveness**
2. **Performance**
3. **Deterministic convergence**

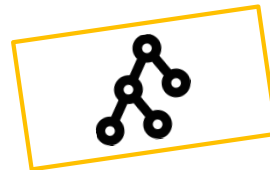
**Solution:** replace Datalog with imperative code



Parsing



Routing simulation



Verification



Explanation

# Lesson 2: Model fidelity is hard, but not why you think

**Concern:** “Every software version will have different semantics!”

**Reality:** The real challenge is **undocumented semantics**

**Solution:** New stage to benchmark Batfish against an emulator



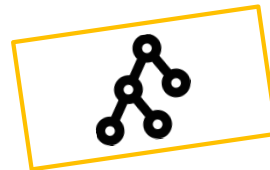
Fidelity



Parsing



Routing simulation



Verification



Explanation

## Lesson 3: Usability is hard for reasons you think, and then some

**Ambiguity:** “Hosts **A** can reach hosts **B**”

- ALL **applications** can reach SOME **DNS server** (e.g., in the same AZ)
- SOME **SNMP collector** can reach **infrastructure elements**
- ALL **service frontends** can reach ALL **backend VIPs**

**Solution:** custom assertions for each use case.

# What's next for network verification?

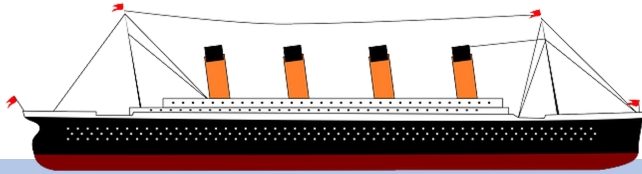
Make it an *effective* and *universal* practice

## Key hurdles

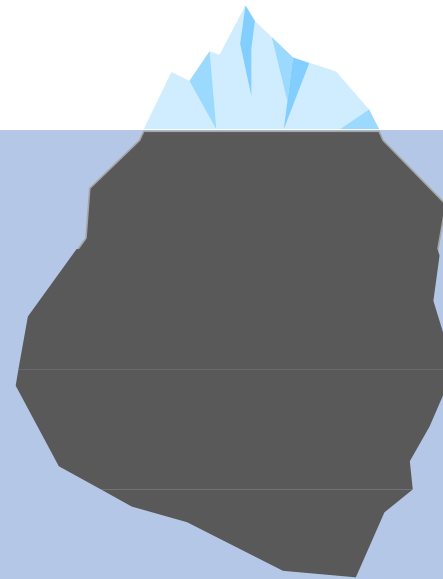
- Lack of network automation
- Lack of expertise
- Lack of precise specifications

# Network verification is only as good as its invariants

Network change



invariants



untested  
network  
behaviors

# Network verification is only as good as its invariants

This article was published on: 10/4/21


[Home](#) / [Featured](#) / Facebook outage triggered by BGP configuration issue as services fail for 6 billion

Featured [Read This](#)

## Facebook outage triggered by BGP configuration issue as services fail for 6 billion

## WAN router IP address change blamed for global Microsoft 365 outage


Command line not vetted using full qualification process, says Redmond. We think it involved chewing gum somewhere

 [Paul Kunert](#)

Mon 30 Jan 2023 / 13:35 UTC

# Inspiration from code coverage

Merged

 progwriter

73.03%

< 81.81% >

(-0.01%)

64.88%

(-0.02%)

Overview

Diff

Coverage Changes3

Files

Commits

/ projects

Files					Complexity	Coverage
<div>allinone/src/main/java/org/batfish/allinone</div>					<div>52.38%</div>	<div>62.91%</div>
<div>batfish-client/src/main/java/org/batfish/client</div>					<div>61.10%</div>	<div>64.57%</div>
<div>batfish-common-protocol/src/main/java/org/batfish</div>	+17	+8	+12	-3	<div>+12.00%70.02%</div>	<div>-0.02%78.02%</div>
<div>batfish/src/main/java/org/batfish</div>	+6	+3	+2	+1	<div>+4.00%62.26%</div>	<div>-0.01%70.63%</div>
<div>coordinator/src/main/java/org/batfish/coordinator</div>					<div>63.00%</div>	<div>65.27%</div>
<div>minesweeper/src/main/java/org/batfish/minesweeper</div>					<div>61.71%</div>	<div>72.96%</div>
<div>question/src/main/java/org/batfish/question</div>					<div>71.06%</div>	<div>81.47%</div>

# NetCov: Coverage for network configurations

Current view: [top level - config](#)

Test: [internet2.init](#)

Date: 2022-09-20 14:00

Filename

- [atla.conf](#)
- [chic.conf](#)
- [clev.conf](#)
- [hous.conf](#)
- [kans.conf](#)
- [losa.conf](#)
- [newy32aoa.conf](#)
- [salt.conf](#)
- [seat.conf](#)
- [wash.conf](#)

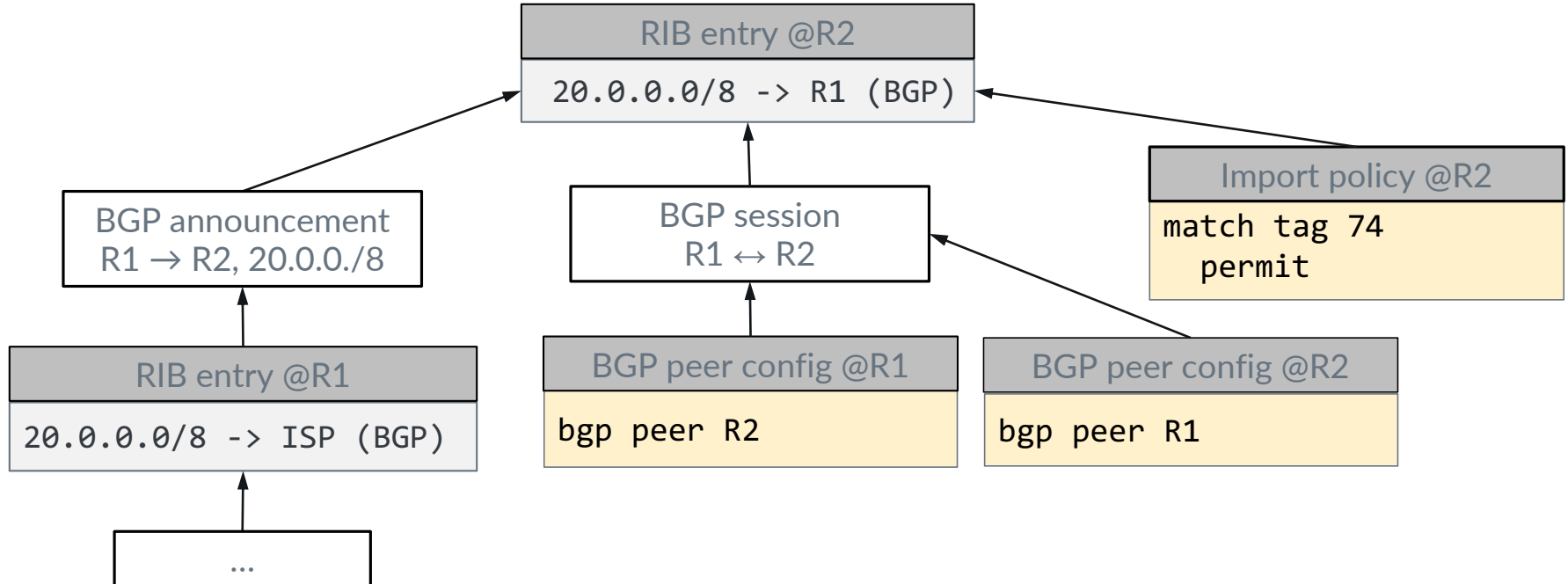
```
12105 /* reject routes we should never accept */
12106 policy-statement SANITY-IN {
12107     /* Reject any BGP prefix if a private AS is in the path */
12108     term block-private-asn {
12109         from as-path PRIVATE;
12110         then reject;
12111     }
12112     /* Reject any BGP NLRI=Unicast prefix if a commercial ISP's AS is in the path */
12113     term block-commercial-asn {
12114         from as-path COMMERCIAL;
12115         to rib inet.0;
12116         then reject;
12117     }
12118     term block-nlr-transit {
12119         from as-path NLR;
12120         then reject;
12121     }
12122     /* Reject BGP prefixes that should never appear in the routing table */
12123     term block-martians {
12124         from {
12125             /* default */
12126             route-filter 0.0.0.0/0 exact;
12127             /* rfc 1918 */
12128             route-filter 10.0.0.0/8 orlonger;
12129             /* rfc 3330 - loopback */
12130             route-filter 127.0.0.0/8 orlonger;
12131             /* rfc 3330 - link-local */
12132             route-filter 169.254.0.0/16 orlonger;
12133             /* rfc 1918 */
12134             route-filter 172.16.0.0/12 orlonger;
12135             /* iana reserved */
12136             route-filter 192.0.2.0/24 orlonger;
12137             /* 6to4 relay */
12138             route-filter 192.88.99.1/32 exact;
12139             /* rfc 1918 */
12140             route-filter 192.168.0.0/16 orlonger;
12141             /* rfc 2544 - network device benchmarking */
12142             route-filter 198.18.0.0/15 orlonger;
12143             /* rfc 3171 - multicast group addresses */
12144             route-filter 224.0.0.0/4 orlonger;
12145             /* rfc 3330 */
12146             route-filter 240.0.0.0/4 orlonger;
12147         }
12148         then reject;
12149     }
12150     /* Reject BGP prefixes which Abilene originates */
12151     term block-abilene {
12152         from as-path ABILENE;
12153         then reject;
12154     }
12155 }
```

Total	Coverage
2	64886
26.1 %	

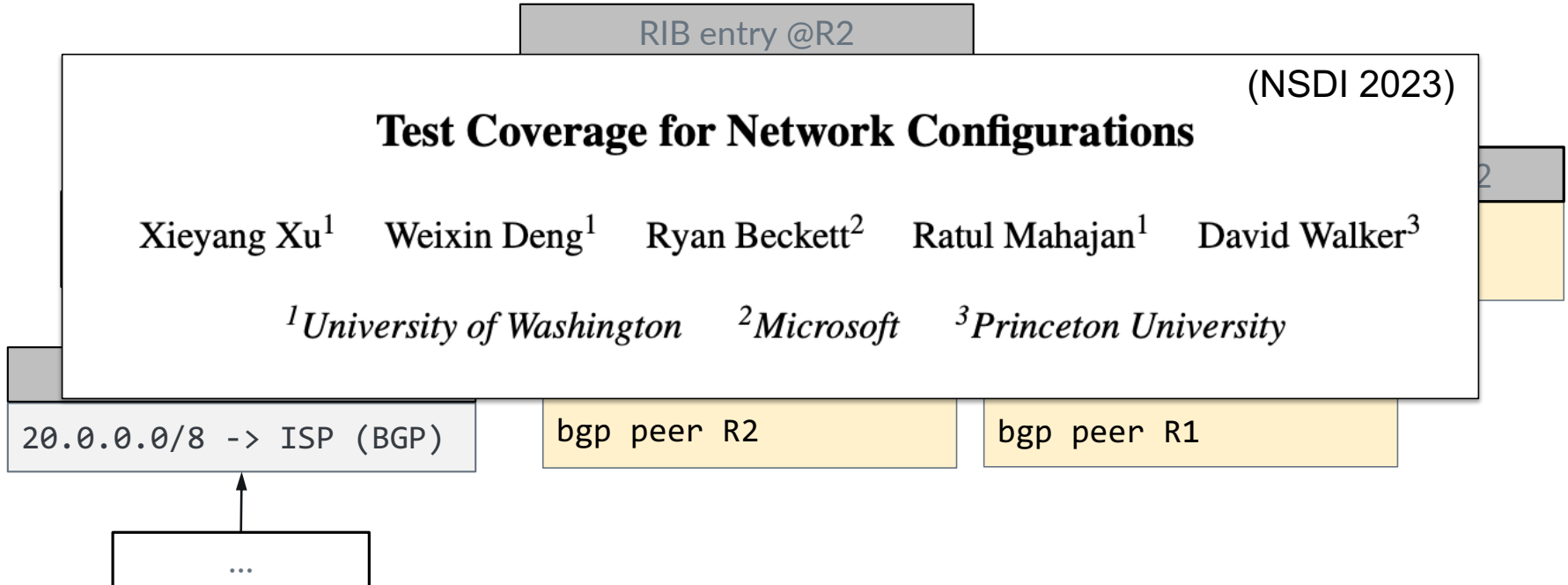
1211 / 5019
4376 / 10800
1156 / 3512
1196 / 4801
1235 / 6178
1832 / 8960
770 / 6545
568 / 3063
1845 / 6030
2723 / 9978

<https://github.com/UWNetworksLab/netcov>

## NetCov maps tested data plane state to covered config lines



NetCov maps tested data plane state to covered config lines



# Toward “specification-less” verification

**Insight:** Network’s spec may not be known but a change’s intent is

- The change should have no impact on reachability
- The change should make the new subnet reachable from here
- The change should make traffic on *path1* take *path2*

**Solution:** Differential network verification

- A relational language for network changes
- A evaluation procedure based on finite state transducers

w/ Xieyang Xu, Zak Kincaid, Arvind Krishnamurthy, David Walker, and Yifei Yuan

# Summary

Network verification is key to high network availability

First generation of tools have taught us a lot about what (does not) work

Must focus now on making network verification an effective and universal practice