Challenges in Network Debugging

- Modern enterprise networks are large and complex
- Complex interactions between different devices increase chance of misconfiguration
  - Routers, firewalls, switches, NAT boxes, etc.
- Bugs in software implementation that only get triggered under certain conditions
  - May cause large black outs if not detected while testing the network
- Difficult to test the entire state space
  - $2^{32}$ IP addresses, $2^{16}$ port addresses, etc.

Limitations of Configuration Analysis

- Prediction is difficult
  - Various configuration languages
  - Dynamic distributed protocols
- Prediction misses implementation bugs in control plane

Our Approach: Data Plane Verification

- Closer to actual network behavior
- Less prediction
- Unified analysis for multiple control-plane protocols
- Can catch control-plane implementation bugs

Real-Time Verification: VeriFlow

- Our new tool, VeriFlow*, checks network-wide invariants in real time using data-plane state
  - Absence of routing loops and black holes, access control violations, etc.
- It functions by
  - Constructing a formal model of the network’s behavior
  - Using custom algorithms to automatically derive whether the network contains errors

VeriFlow Operation

- Our first tool, Anteater*, used data plane verification technique to debug network operations
- We evaluated Anteater with UIUC campus network
  - 178 routers
  - 1,627 FIB entries per router (mean)
- It revealed 23 real bugs with 3 invariants in 2 hours

VeriFlow’s Core Algorithmic Process

- Simulation Setup
  - 172 node Rocketfuel topology
  - Route Views BGP traces
  - 5 million RIB entries
  - 90K BGP updates
- 97.8% of the updates are verified within 1 millisecond