ILLINOIS TECH

College of Computing

Try CREASE out on FABRIC!

A Network Debugger for FABRIC Experiments

Alexander Wolosewicz — Illinois Tech

Nik Sultana — Illinois Tech

Ashish Gehani — SRI

Vinod Yegneswaran — SRI

http://crease.cs.iit.edu/

Introduction

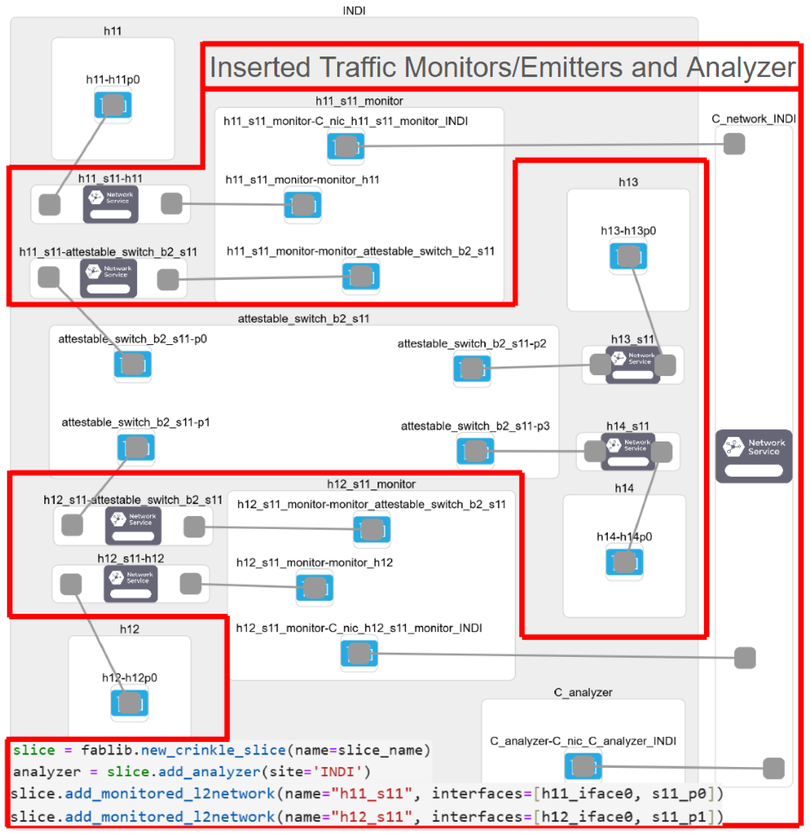
- Current state-of-the-art for network debugging primarily centered on industrial-scale networks
- Heavily reliant on specific hardware
- FABRIC, as a testbed, afford significant flexibility and relaxed constraints compared to industrial networks
- This flexibility allows for new techniques that are much more general, supporting any components, topology, or protocols

Approach

- Fablib extended to support adding VMs as “smart” bump-in-the-wire monitors
- Monitors communicate with an analyzer out-of-band, and insert unique IDs to the trailer of in-band packets
- Analyzer collects information on packet headers, ID, and location into database
- Database contains flow-level data, position, and packet-specific ID

Results

- Packet tracing can be done from a central point or the Jupyter instance, with significantly fewer commands than spawning and analyzing tcpdump jobs
- Devices which do not support capturing their interfaces can be monitored as easily as any node
- Lays a foundation for more active debugging by enabling anywhere probing and on-the-fly editing of header field values



Motivation

- Debugging a significant barrier to conducting experiments on FABRIC
- A research-friendly debugger should not be reliant on hardware or protocol support, since that is often the focus of the research
- Data collection could pave the way for automated assistance in tracking down and solving bugs

Packets across the network are recorded into a central database. The database can be queried to reduce the graph to one relevant to the issue.