OFRewind
Enabling Record & Replay Troubleshooting for Networks
Andreas Wundsam (ICSI / UC Berkeley) · Dan Levin (TU-Berlin / T-Labs) · Srini Seetharaman (Deutsche Telekom Inc.) · Anja Feldmann (TU Berlin / T-Labs)

Motivation
- Troubleshooting networks is hard
  - Large scale, wide spread, distributed state
  - Black box components (not instrumentable)
- Current toolset:
  - Aggregated statistics (SNMP)
  - Sampled date (NetFlow)
  - Local traces (tcpdump, Endace et.al) Single Vantage Point
- Replay debugging to the rescue?

Key Insights
- Not all traffic is equal (control plane: <1% traffic, >90% bugs)
- Can enable network-wide, consistent Replay Debugging by
  - Selective recording
  - Separation of control and data plane recording
- Localize bugs through
  - Device Mapping
  - Time dilation
  - Trace bisection

Our Approach: OFRewind
- Enabled by Split Forwarding Architectures, such as OpenFlow.
- Select traffic to be recorded
  - Always-On record of OpenFlow Control-Plane
  - Dynamic, flexible partial recording of Data-Plane (e.g., L3 routing updates)
- After fault occurrence, re-inject events into the network
- Sub-select recorded Control- and Data-Plane traffic for fault localization

System Architecture
- Two Primary System Components: OFRewind + DataStore
  - OFRewind: proxy between controller and switches for control-plane recording/replay
  - Orchestrates multiple Datastores for data-plane recording/replay
  - Maintains global ordering of all flows observed in network
  - Allows precise time-control over replay pace, ensuring flow ordering during replay is preserved

Case Studies
Switch Bug (CPU inflation)
- CPU inflation reproduced through replay
- and localized by bisection of traffic

Quagga RIP Bug
- Race condition Quagga RIP daemon reproduced and localized by coordinated recording and replay.

Performance Evaluation
Controller Performance
- Simple Controller
- OFRewind, NOX, Flowvisor...

Impact on Switch Performance