Automated Attack Discovery in Data Plane Systems

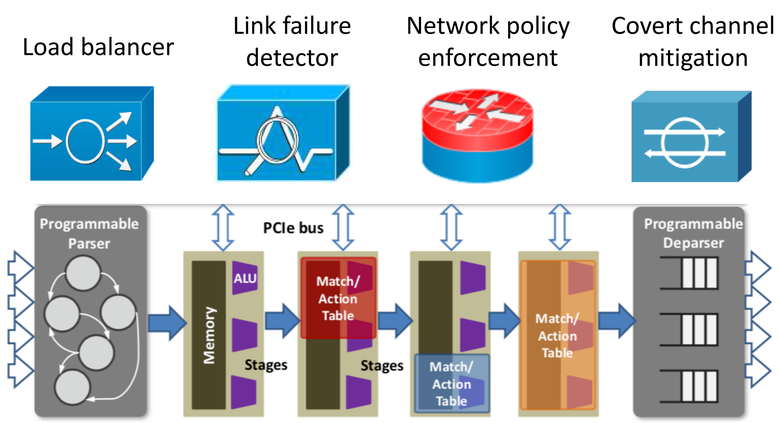
Qiao Kang, Jiarong Xing and Ang Chen
Rice University

Email: qiaokang@rice.edu

This work also appeared at CSET'19 workshop

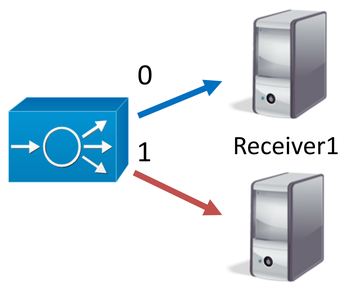
1. Problem

- ***Background***: Data plane systems are emerging
  - Enabled by programmable switches
  - Switch pipeline is programmable using P4
  - Fast reaction to dynamic network events



- ***Problem***: Data plane systems can be attacked
  - Example: load_balancer.p4

```
If (TCP.sport % 2)
   forward(0)
Else
   forward(1)
```



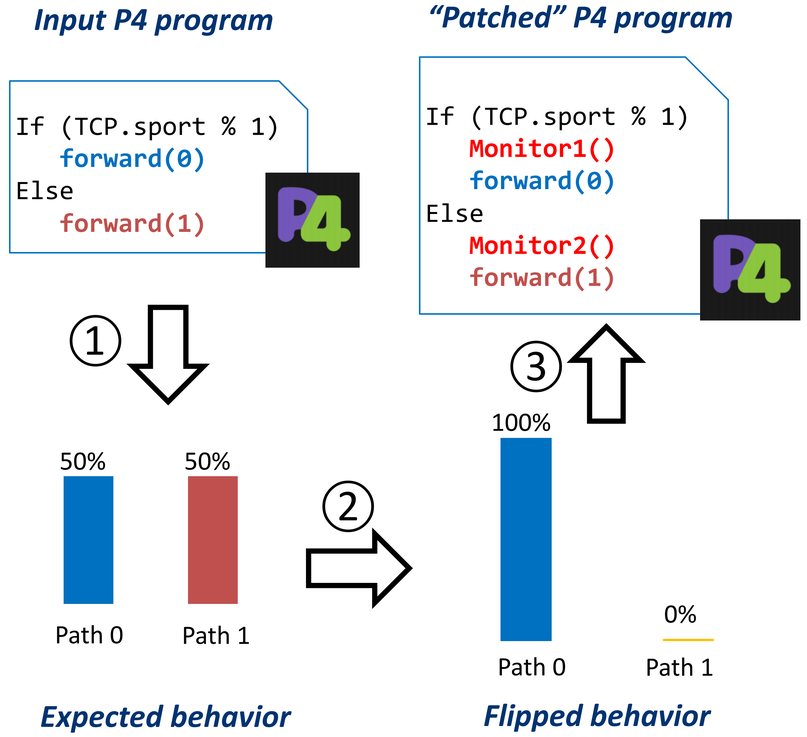
- ***New attacks:*** "Flipping" the expected behavior
  - Expected behavior: Evenly splitting traffic
  - Malicious traffic pattern: TCP.sport = 1, 3, 5, 7, …
  - "Flipped" behavior: All packets go to link 0

- ***A general class of attacks***
  - Applies to many data plane systems
  - Different systems are vulnerable to different patterns

2. Approach

- ***Our goal***: Given a data plane system, discover all malicious traffic patterns and synthesize defenses in an **automated manner**.

- ***Our system: 3-step automated attack discovery***
  - *Step* ①: Establish expected behaviors
  - *Step* ②: Flip the expected behaviors
  - *Step* ③: Synthesize runtime monitors

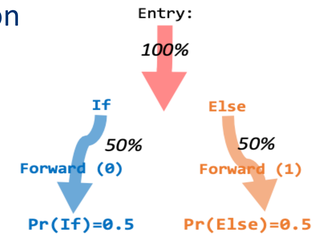


Automated

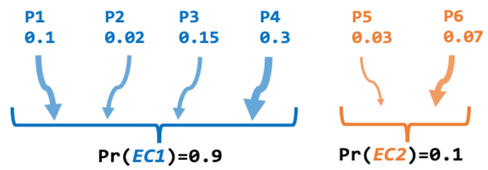
This work is partially supported by the National Science Foundation through grant CNS-1801884.

3. Challenges

- ***Challenge #1***: Quantifying expected behaviors
  - Probabilistic symbolic execution
  - Enabled by model counting
  - Study per-path probabilities



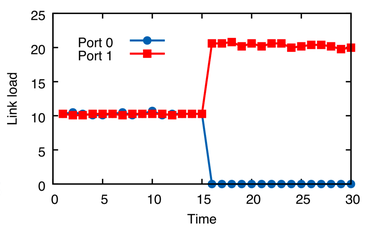
- ***Challenge #2***: Identifying Equivalence Classes (ECs)
  - Path# can be very large
  - *Group "equivalent" paths to ECs.*



- ***Challenge #3***: Handling stateful programs
  - Exploring N packets: state explosion
  - Use directed symbolic execution

4. Ongoing work

- ***Initial results***
  - Attack load_balancer.p4
  - t < 15s: Expected behavior
  - t = 15s: Attack starts
  - Attack detected by monitors



- ***Open questions***

1. How to group paths to ECs?
   - Too fined-grained: too many ECs
   - Too coarse-grained: lose useful information

2. How to deal with switch resource constraints?
   - P4 switches have limited memory and ALUs
   - Compress monitors using sketches