Verifiable Network Blocks
Towards Trusted Network Applications

Authors: Arseniy Zaostrovnykh†‡, Katerina Argyraki†, George Candea‡, Vova Kuznetsov†

Context
- Software network functions are great!
- They offer much needed flexibility and rapid updates.

Problem
- Flexibility + rapid development => bugs
- Yet the network is often a critical part of the system.

Goal
- Keep the flexibility, get rid of the bugs.
- Preserve most of the development ease.
- Design network-function development framework with built-in automated verification.

Approach
We run symbolic execution to automatically check a property. We substitute stateful components with light models, valid w.r.t. a specification of the property and the given use case.

Tools
- Based on DPDK
- VeriFast for formal verification
- Klee for automated verification

Legend
- Logical specification
- Code block
- Interface function declaration

API
- put
- get
- erase

Verified Implementation

Symbolic Model

"put" mock
"get" mock
"erase" mock

← Loop Invariant

Network Application

start
put(x)
get()
erase()