Cryptographic Path Validation for SCION in P4

Lars-Christian Schulz, Robin Wehner, David Hausheer

Otto-von-Guericke-University Magdeburg
Faculty of Computer Science
Networks and Distributed Systems (NetSys) Lab

Euro P4 2023 – Paris, France
December 8, 2023

Agenda

The SCION Internet Architecture

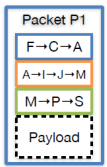
Path-based Network Architecture

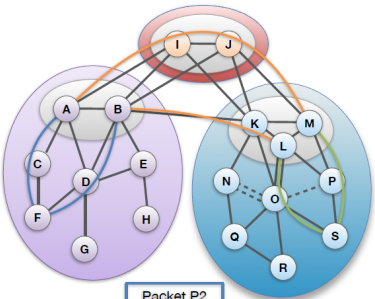
Control Plane - Routing

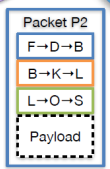
- Constructs and Disseminates Path Segments

Data Plane - Packet forwarding

- Combine Path Segments to Path
- Packets contain Path
- Routers forward packets based on Path
  - Simple routers, stateless operation







SCION

Motivation

- SCION encodes path in packet header as hop fields
  - Hop field MAC must be validated by routers
- Current SCION networks rely on software routers
  - Open-source reference router (Go): $< 5\,\text{Gbit/s}$
  - Proprietary router by Anapaya Systems (DPDK)
  - In development: eBPF/XDP router
- Future applications of SCION will require performance and reliability of hardware
  - SCION Internet Exchanges (IXPs)
  - Support growth of the SCION network

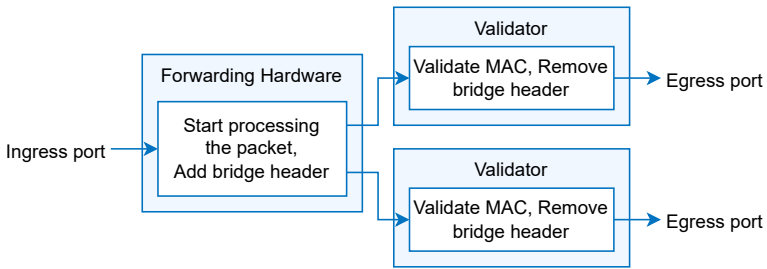
SCION Border Router in P4 for Tofino 2

- SCION is difficult to support on Tofino, because
  - Routers do not accelerate AES-CMAC
  - Headers are large and cumbersome to parse
- SCION in P4 pioneered by de Ruiter and Schutijser[1]
  - Implementation for Tofino 1 architecture (100 Gbit/s per port)
  - Static lookup table for hop field validation
- We needed support for SCION extensions (EPIC, DRKey, INT, etc.)
- Make use of newer Tofino 2 hardware (400 Gbit/s per port)
- Our contributions:
  - Modularize border router architecture to split forwarding and validation part
  - E□cient implementation of AES on Tofino 2 for SCION validation in P4

[1] de Ruiter and Schutijser: Next-generation internet at terabit speed: SCION in P4, CoNEXT '21

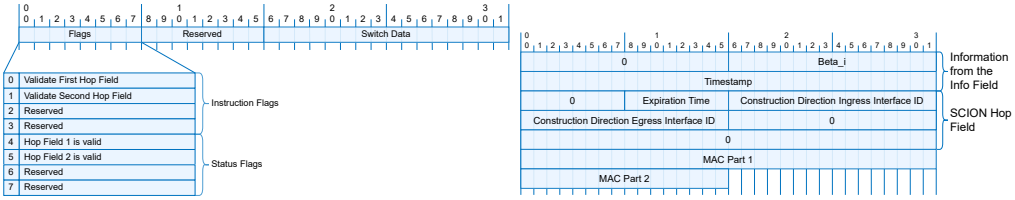
Modularizing the Border Router

- Parsing and forwarding SCION packets fits P4 well
- Cryptographic validation of HFs does not
- Idea:
  - Combine Tofino and FPGA (currently ongoing work)
  - Use multiple independent P4-programmable pipelines (this work)



Bridge Header

- Reparsing the header in the cryptographic validator is costly
- Store validation request in **bridge header** prepended to packet



- Original packet is sent as payload as switch cannot buffer during validation

Efficient AES in P4

- AES-128 Basics
  - 128-bit block cipher, 10 rounds
  - Each round used a different key derived from cipher key (key schedule)
- Chen[2] implemented AES on first-gen Tofino
  - Using "scrambled lookup tables" combining SubBytes and AddRoundKey with key expansion in control plane
  - Up to 2 rounds per pipeline pass → 5 passes to complete

```
AES Cipher
  AddRoundKey(state, key[0])
  for round := 1 to 10
    SubBytes(state)
    ShiftRows(state)
    MixColumns(state)
    AddRoundKey(state, key[i])
  end for
  SubBytes(state)
  ShiftRows(state)
  AddRoundKey(state, key[11])
```

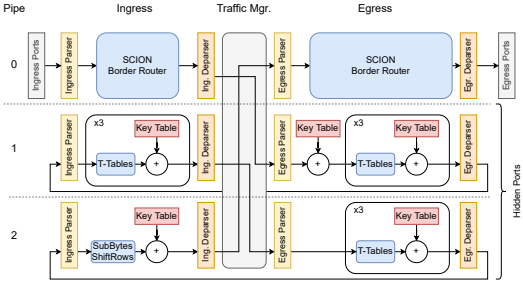
[2] Chen: Implementing AES Encryption on Programmable Switches via Scrambled Lookup Tables, SPIN '20

Efficient AES in P4

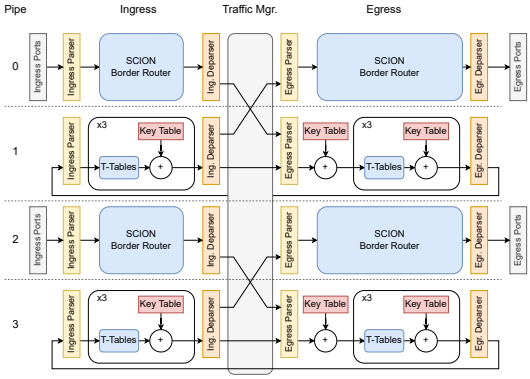
- Our approach:
  - Use T-table optimization proposed by Daemen and Rijmen directly → **dynamic key expansion possible if needed**
  - Make use of higher pipeline capacity in Tofino 2 → **3 instead of 2 rounds**
  - Use of both ingress and egress pipelines → **6 rounds per pass**
  - Work on two blocks in parallel → **double the throughput**
  - Dedicate whole pipeline to recirculation → 3.2 Tbit/s recirculation BW per pipe
- Result: Can calculate two 16 byte AES-CMACs per packet at line rate

Folded Pipelines on Tofino

1 BR Pipe + 2 AES Pipes



2 BR Pipes + 2 AES Pipes



Pipeline Layout Variants

- Overall router performance depends on pipeline layout

Layout	Pipes	Ports	Bandwidth
BR w/o AES	4/4	32/32	12.8 Tbit/s
BR + 2 AES Pipes	3/4	8/32	3.2 Tbit/s
2 BRs + 2 AESs Pipes	4/4	16/32	3.2 Tbit/s
BR + 1 AES Pipe	2/4	8/32	1.6 Tbit/s

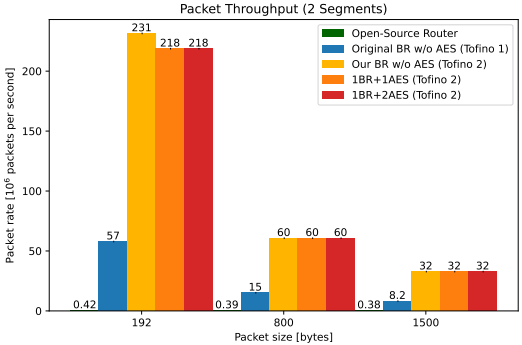
- Different layouts are optimal depending on Tofino 2 variants and physical port configuration of the switch

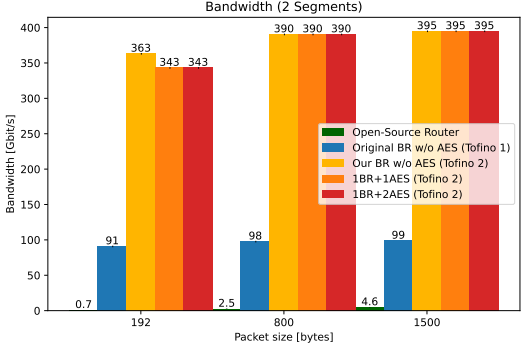
Evaluation

- Tofino 2 Hardware Utilization
  - Very high resource utilization in SCION parser
    - Can parse up to 8 hop fields
    - Trade-o between feature support and max. path length
  - Number of tables lookups (in SRAM) are bottleneck in AES pipes
- Evaluation Setup
  - 1BR+1AES Pipe and 1BR+2AES Pipe configuration
  - Use remaining pipe as traffic generator (connected via external cable)
  - Reference open-source and original Tofino 1 border router for comparison
  - Tofino 2: Edgecore DCS810; Tofino 1: UfiSpace S9180-32X
  - Reference router: AMD EPYC 7543P (32 cores), 128 GB RAM, 2x 100G Ethernet

Evaluation

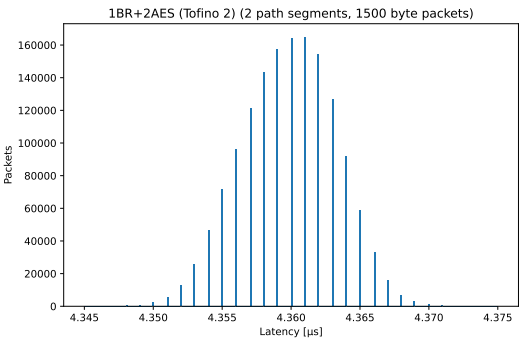
Throughput per Switch Port



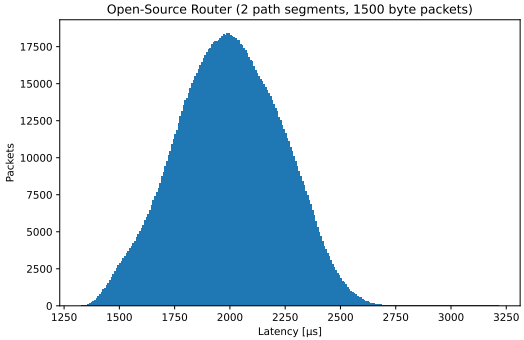


Evaluation

Latency



Mean = 4.36 µs; StdDev = 0.003 µs
(1BR+2AES and 1BR+1AES)



Mean = 1996.6 µs; StdDev = 235.65 µs

Tofino 2 without AES: Mean = 2.79 µs; StdDev = 0.003 µs

Conclusions

- Achieved 400G line-rate SCION forwarding with cryptographic HF validation
- Our router is flexible: HF validation can be performed by
  - 1 dedicated switch pipe (1.6 Tbit/s)
  - 2 dedicated switch pipes (3.2 Tbit/s)
  - An external device (e.g., FPGA) (12.8 Tbit/s)
- Future work:
  - FPGA-based validation solution is not implemented yet
  - Support for SCION extensions, e.g., in-band telemetry, bandwidth reservations
- Poster: *High-Speed Per-Packet Checksums on the Intel Tofino*

Lars-Christian Schulz
lschulz@ovgu.de

Robin Wehner
robin.wehner@ovgu.de

David Hausheer
hausheer@ovgu.de

Source Code and Evaluation Artifacts

https://github.com/netsys-lab/scion-p4

SCION Education, Research, and Academic Network

https://sciera.readthedocs.io