EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

FACULTY OF
SCIENCE
Communication Networks

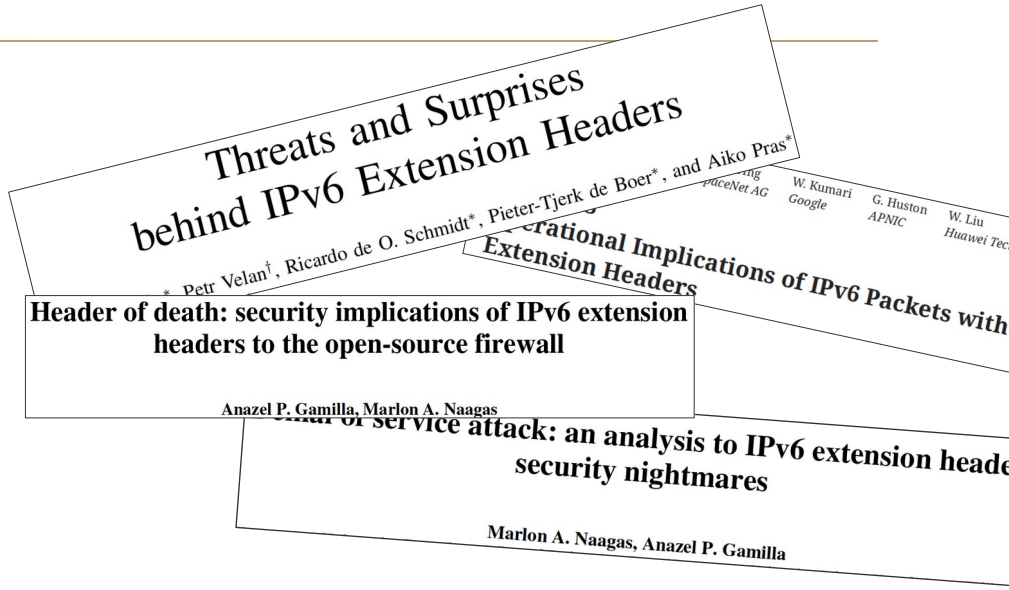
Implementation Update on P4-MNA using ISD

Fabian Ihle <fabian.ihle@uni-tuebingen.de>, Michael Menth <michael.menth@uni-tuebingen.de>

http://kn.inf.uni-tuebingen.de

Motivation

- For MNA to be successful, it must be efficiently implementable in hardware
  - Similar case: IPv6 EH



- Intel Tofino™ 2 as reference architecture for a hardware switching ASIC
  - Limited resources and operations with 400 Gb/s per port

- If MNA works on P4 hardware MNA likely works on vendor ASICs
  - We are doing research and do not build commercial MNA hardware

- P4-MNA
  - Basic MPLS functionality: pop, swap, forward
  - Process **ISD** MNA network actions in-scope at LSR (select, HBH)
  - □ How does the implementation work?
  - □ What are the capabilities of P4-MNA?

ISD Processing

- All relevant LSEs must be parsed before processing

- Network actions are executed through application of match+action tables (MATs)
  - 1 MAT per LSE in a NAS 16 MATs for 1 NAS
  - LSE is matched according to opcode and NAL field in MAT
  - More details in paper

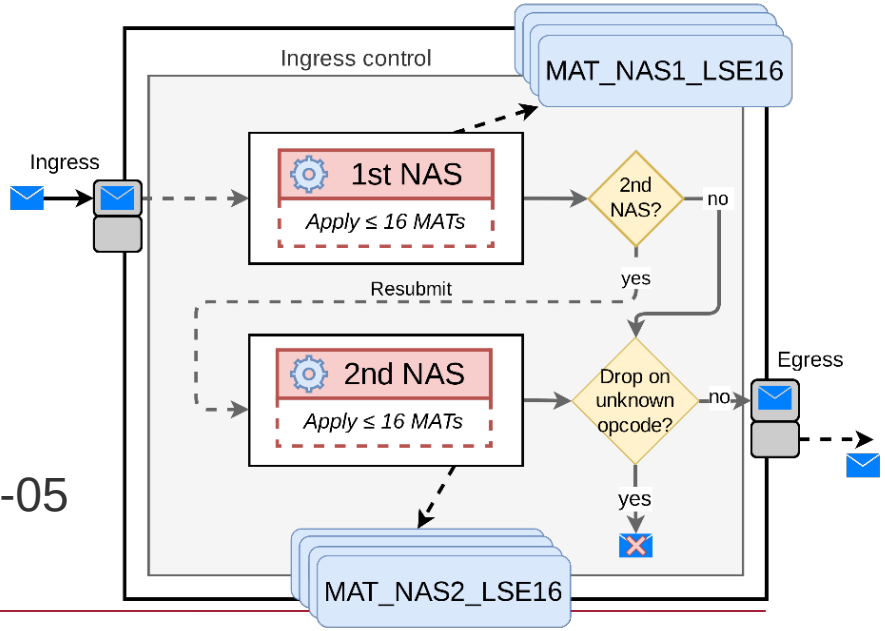
- 16 MATs fit into one pipeline iteration
  - 2nd NAS present resubmit process 2nd NAS

- Implementation applies network actions at line rate
  1. Performance evaluation: verify if MNA implementation applies network actions at line rate
  2. Use case evaluation: implemented draft-cx-mpls-mna-inband-pm-05

MAT_NAS1_LSE_0

Match keys:
hdr.net_action_lse[0].opcode &
hdr.net_action_lse[0].NAL &
meta.already_processed_lse[0]

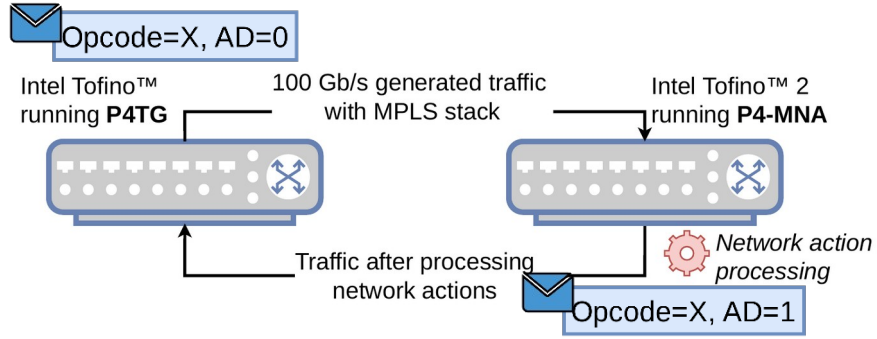
Match fields			Action	Parameters
Opcode	*NAL*	*already_ processed_lse[0]*		...
1	0	0	*lse_0_opcode_1_with_0_ad*	
...	...	0	...	
1	7	0	*lse_0_opcode_1_with_7_ad*	
...	...	0	...	
127	0	0	*lse_0_opcode_127_with_0_ad*	
...	...	0	...	
127	7	0	*lse_0_opcode_127_with_7_ad*	



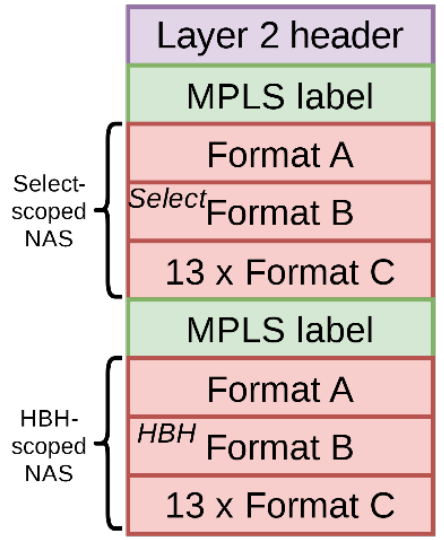
Performance Evaluation

- Goal: Verify if P4-MNA can process two NAS in line speed

- Define dummy network action
  - „Write arbitrary values into AD fields of Format B/C/D LSEs“
  - ⇒ Verifies that those network actions were applied



- Experiment setup
  - Use P4TG to send traffic to P4-MNA
  - MPLS stack with 32 LSEs generated at 100 Gb/s
    - Two NAS (select scope and HBH scope)
    - A NAS contains 14 network actions

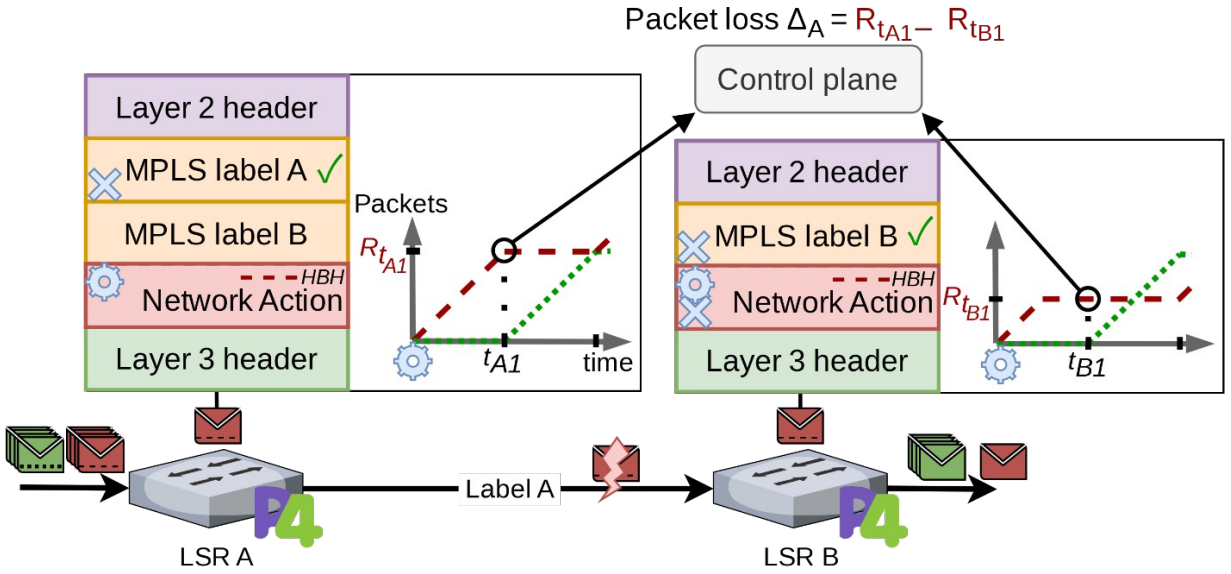


- Result
  - Measured received traffic rate with P4TG is equal to line rate ✓
  - Verified content of AD in network actions using wireshark dissector ✓
  - ~~P4-MNA is capable of 400 Gb/s and 34 LSEs for NAS, our traffic generator is not (yet)~~ ⇒

Use Case Evaluation

- Performance measurement (packet loss) with alternate-marking method[1]

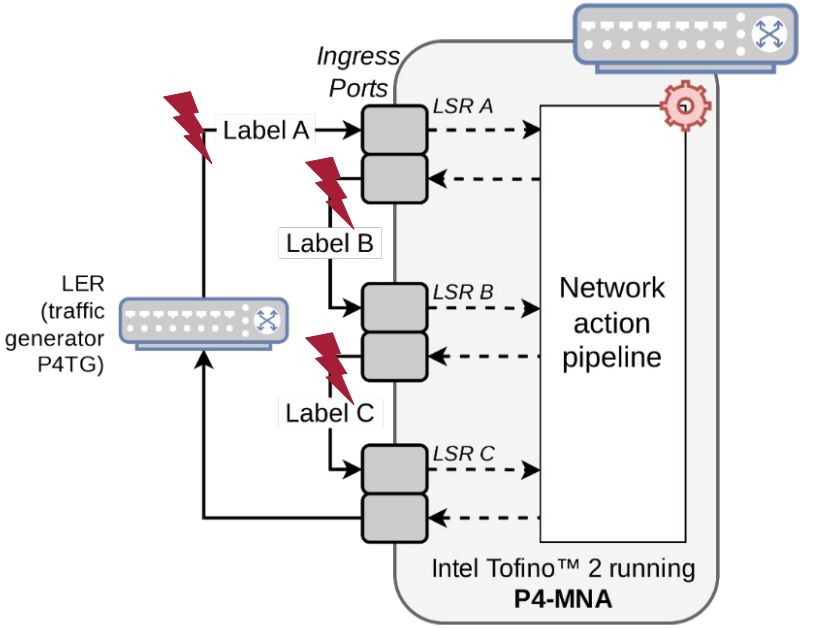
- Concept
  1. Send alternating colored batches of packets
     - Color contained in HBH-scoped network action
  2. LSRs count packets per flow per color
  3. If color alternates send a message to the control plane
     - Packet count of current color
     - Flow ID
     - Timestamp
  4. Control plane correlates counters according to timestamp
     - Gets messages from all LSRs
     - Calculates packet loss between two LSRs



[1]: draft-cx-mpls-mna-inband-pm-05

Use Case Evaluation – Testbed

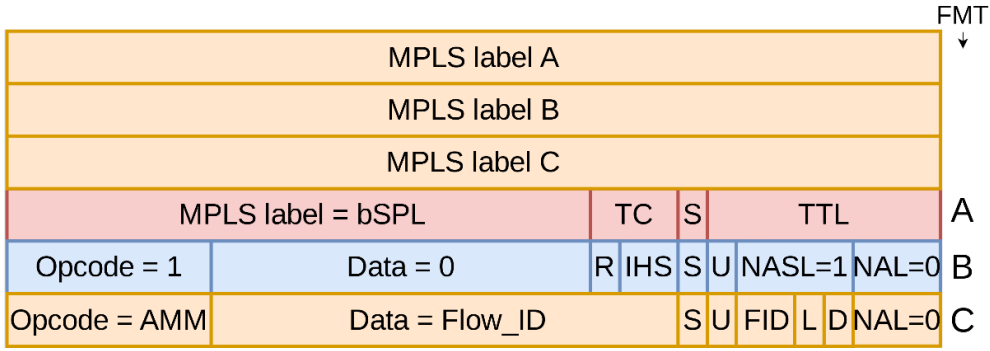
- Each ingress port of Tofino switch represents one LSR
  - LSR A / B / C drop 10% / 20% / 30% of traffic to simulate packet loss on each link



- SR-MPLS stack
  - 3 forwarding labels A, B, C address different ingress ports („LSRs“)
    - Popped by each „LSR“
  - AMM network action placed at bottom-of-stack (HBH scope)

- 100 Gb/s traffic generation containing network action with AMM for packet loss measurement [1]

- Evaluation
  - Measure packet loss using the AMM network action
  - Compare configured packet loss to measured packet loss



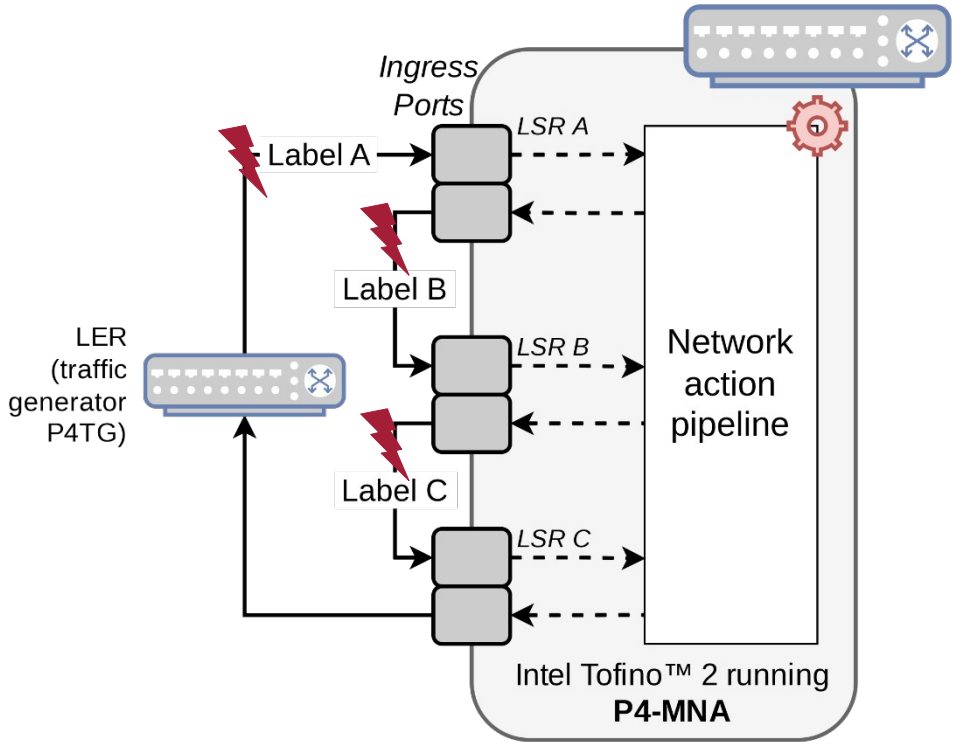
[1]: draft-cx-mpls-mna-inband-pm-05

Use Case Evaluation – Results

- P4-MNA is capable of implementing custom network actions

TABLE 2. **Calculated packet loss with the AMM network action.**

	Generated packets	LSR A	LSR B	LSR C
Color A	99428430	89480744	71578591	50097501
Color B	102344240	92104327	73683350	51580316
Total	201772670	181585071	145261941	101677817
Theoretical packet loss		20177267	36317014.2	43578582.3
Calculated packet loss with AMM		20187599	36323130	43584124
Configured loss rate		0.1	0.2	0.3
Calculated loss rate with AMM		0.1000	0.2000	0.3000

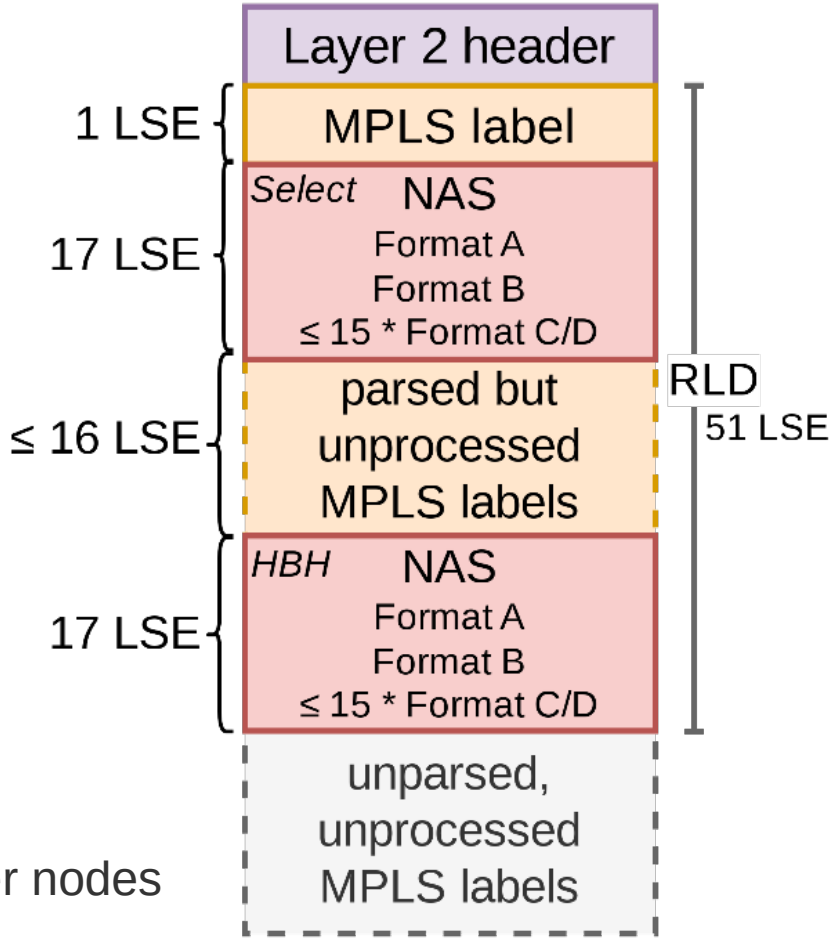


The RLD of P4-MNA

- A NAS has a maximum size of 17 LSEs[0]

- Two NAS are in-scope at an LSR
  - Select-scoped NAS below top-of-stack forwarding label
  - HBH-scoped NAS located deeper in the stack
  - ⇒ 34 LSEs for NAS must be accommodated in hardware

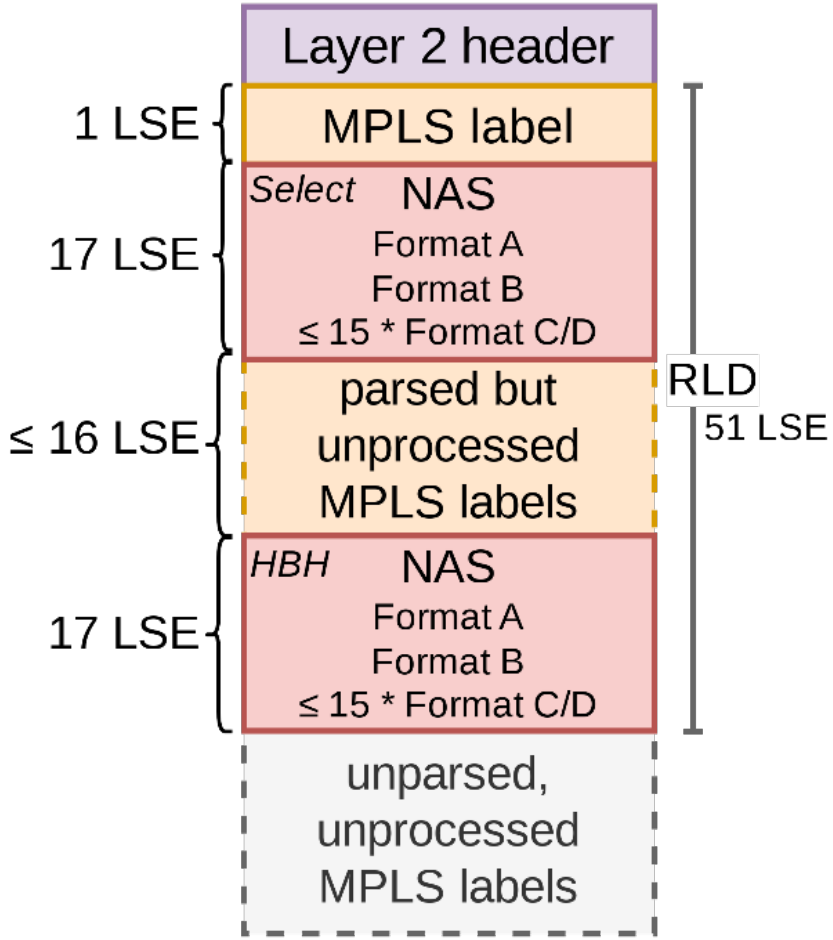
- RLD of P4-MNA: 51 LSE
  - Value results from the available resources in the ASIC
  - Parser must parse all LSEs until HBH-scoped NAS is found
    - Resources must be allocated for those LSEs
  - ⇒ 34 LSEs reserved for NAS
  - ⇒ 17 LSEs remain to encode SR paths or select-scoped NAS for other nodes
    - 1 top-of-stack forwarding label, 16 LSEs until HBH NAS is found
    - Although not relevant to the current LSR, we have to parse them



[0]: draft-ietf-mpls-mna-hdr-08

Impact of Maximum Supported NAS Size (1)

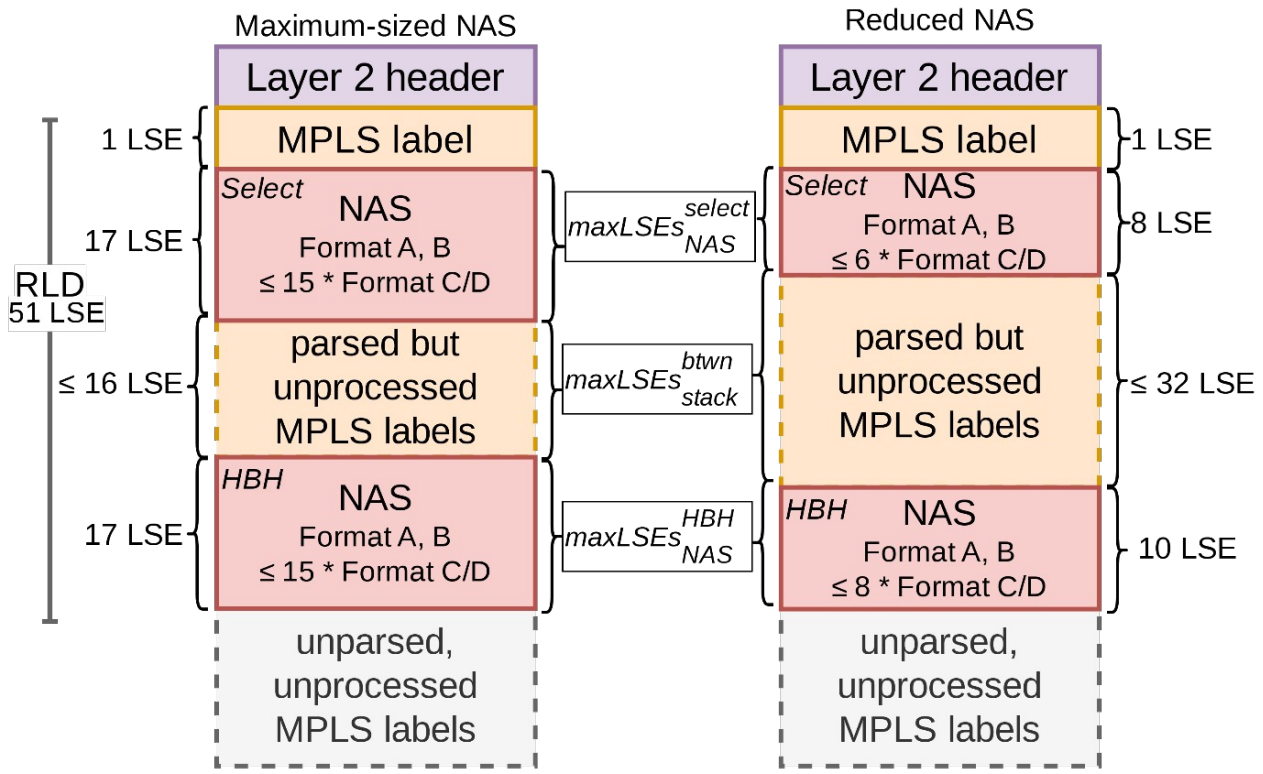
- Arrays are allocated in the P4 program for each LSE encoding type
  - Not just a „stack of LSEs“ but a „stack of differently encoded LSEs in different arrays“
  - Fixed array size prior to runtime, cannot be changed dynamically
  - Other hardware may be able to handle more LSEs or fewer LSEs
  - Observation
    - 16 LSEs for all other LSRs is not a lot given a maximum of 17 LSEs per NAS



- Do network actions really require 17 LSEs in a NAS?
  - Not the currently considered use cases
  - Reduce the maximum supported size of NAS?

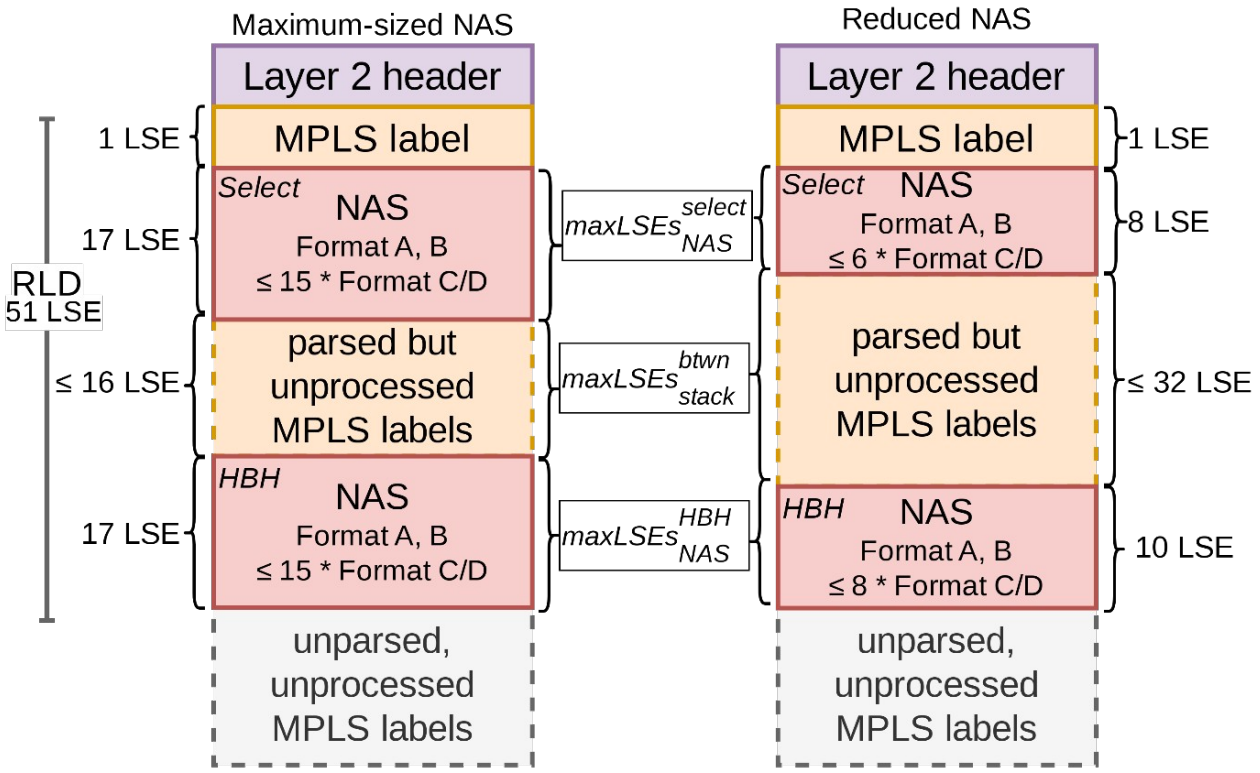
Impact of Maximum Supported NAS Size (2)

- Reducing the supported maximum size of a NAS leaves more room for other LSEs
  - Either: reduce the supported maximum size of a NAS in the encoding ☹
  - Or: signal the supported maximum size of a NAS ☺



Signaling the Maximum Supported NAS Size

- „The node responsible for selecting a path through the MPLS network needs to know and consider the MNA-capabilities and RLD of the transit nodes […][0].”
- What exactly are the „MNA-capabilities“?
- Proposal: Include supported sizes of NAS into signaling
  - $maxLSEs_{NAS}^{select}$ and $maxLSEs_{NAS}^{HBH}$
- Legacy hardware is not required to allocate 34 LSEs for NAS



[0]: draft-ietf-mpls-mna-hdr-08

Conclusion

- Successfully implemented the MNA framework using ISD in P4 on hardware with 400 Gb/s line speed
  - RLD of 51 LSEs containing up to 32 network actions
  - Implemented placeholder network actions to verify if actions are applied in line rate
  - Implemented and evaluated packet loss measurement using AMM
- Proposal to include the maximum supported size of NAS into signaling
  - Allows LSRs to support smaller NAS facilitating integration of less powerful hardware
- Future Work
  - Evaluate a PSD implementation
  - Implement more network actions
- GitHub: https://github.com/uni-tue-kn/P4-MNA
- Preprint: https://arxiv.org/abs/2410.20400

Questions?

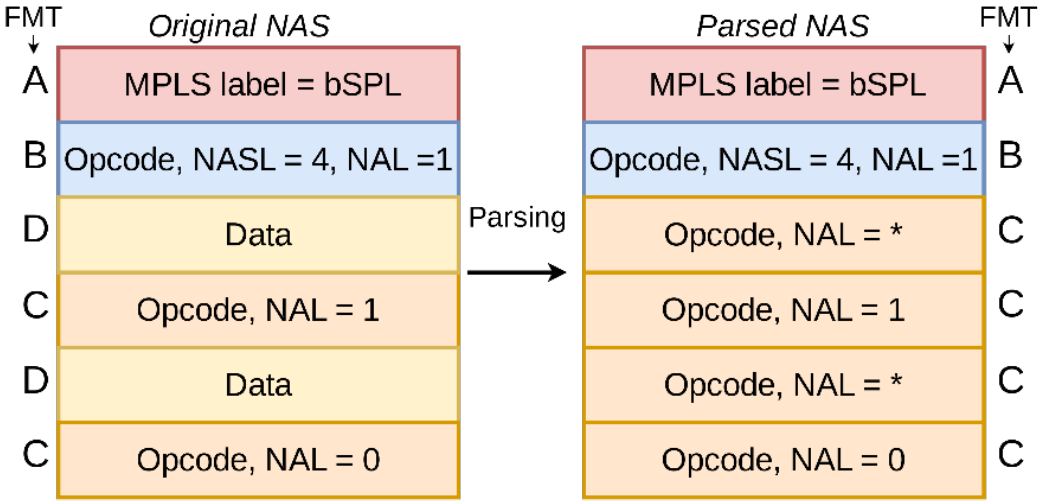
THANK YOU

BACKUP SLIDES

ISD Parsing

- Challenge: MNA header encoding is complex!
  - Parser in P4 is a simple finite state machine
  - Dynamic structure of different encodings requires a lot of states and transitions

- Solution: Parse all Format D LSEs as Format C LSEs using the NASL field
  - NAL field not used during parsing
  - Semantics between Format C and Format D are lost!
    - – restored later during pipeline processing using NAL field



ISD Processing – Restoring Semantics

- Network actions executed through application of match+action tables (MATs)
  - 1 MAT per LSE in a NAS ⇒ 16 MATs for 1 NAS
  - Matches opcode, NAL, `already_processed` metadata field
  - MAT holds up to 8 entries per opcode
    - Accessing different number of AD LSEs

- Semantics between Format B/C and D were lost during parsing
  - They are restored using MATs and the NAL field

❶ Match first LSE with action and corresponding number of AD LSEs

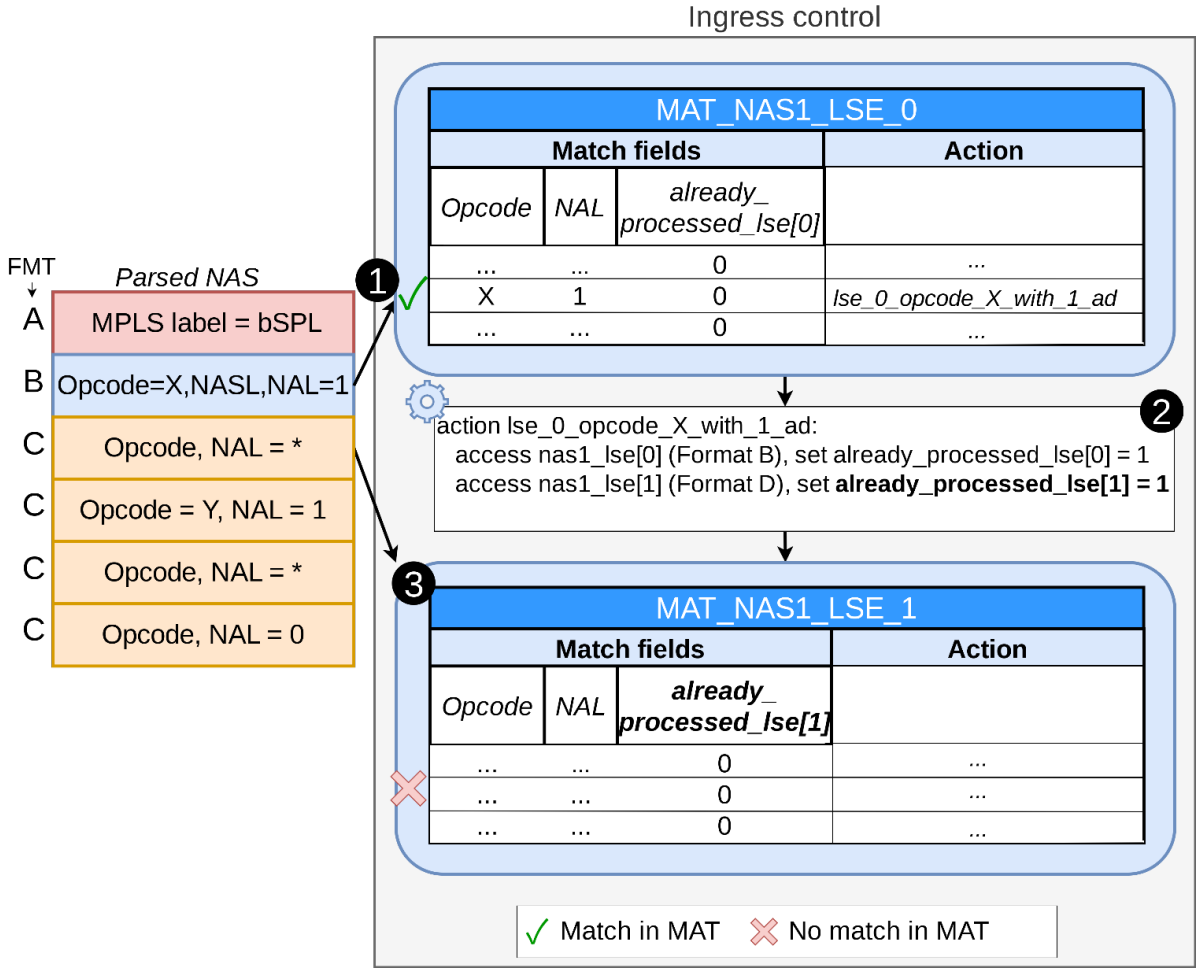
❷ Execute matched action

i. Access first LSE index as Format B

ii. Access all other LSEs indices as Format D

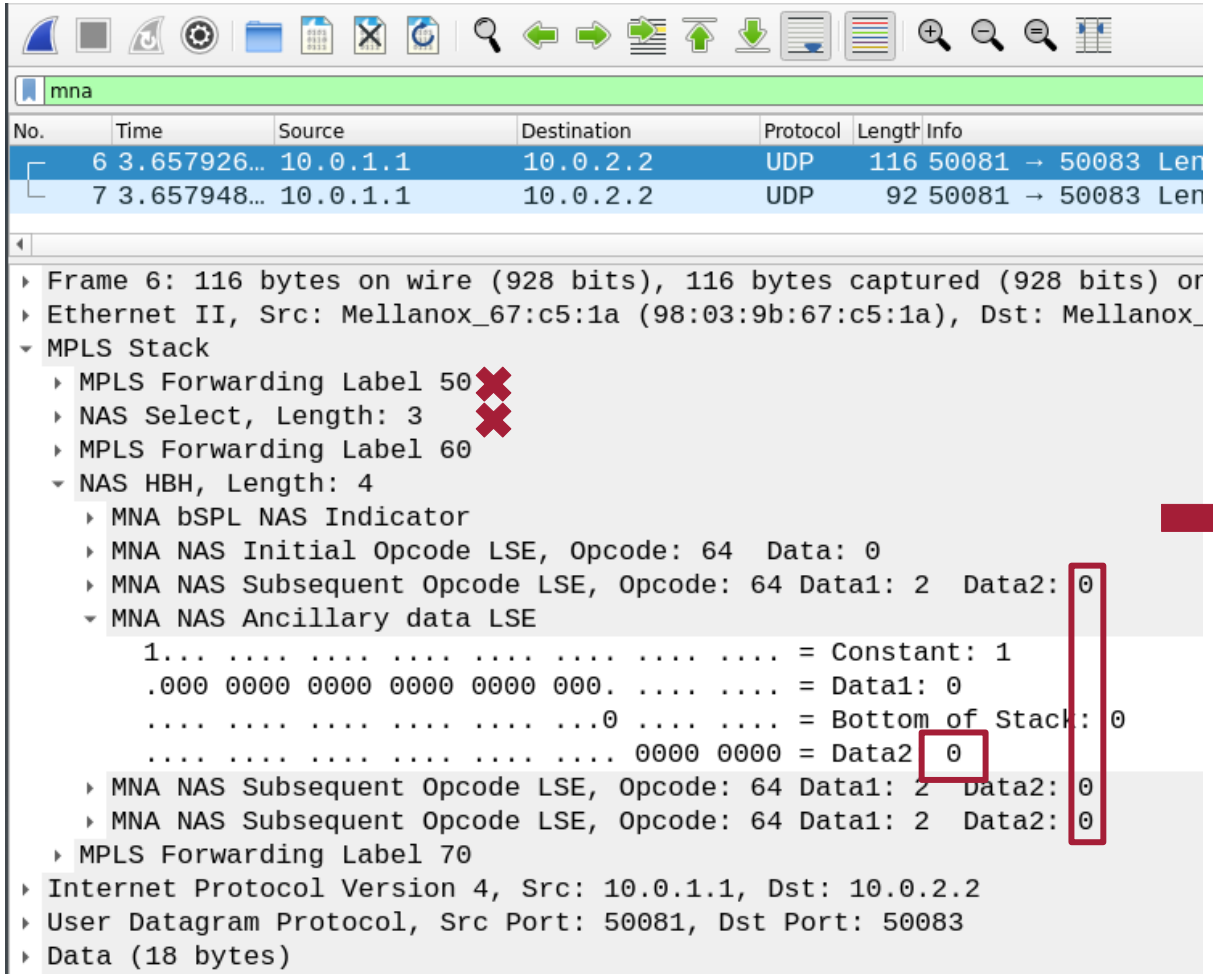
iii. Mark all accessed LSEs as `already_processed`

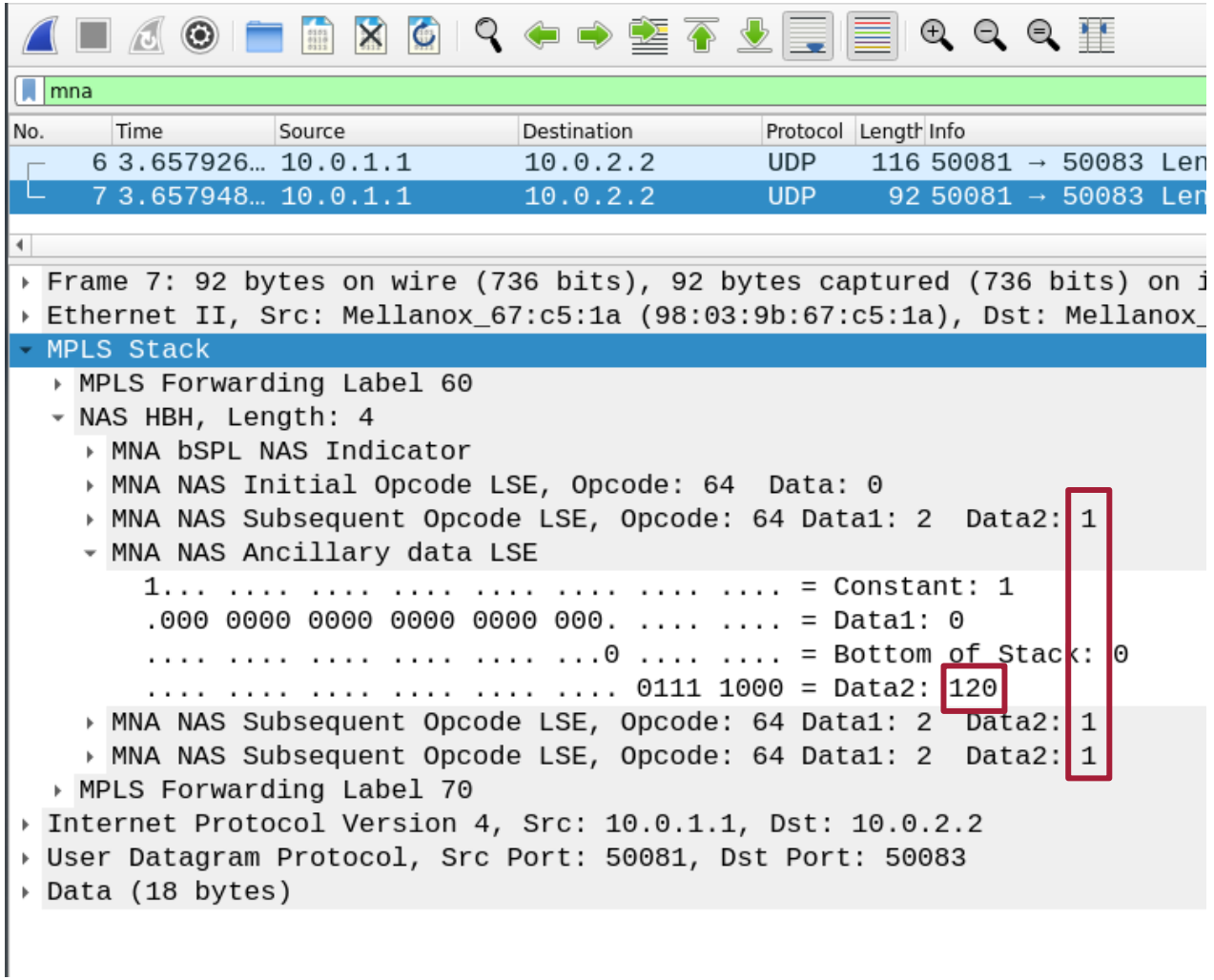
❸ Second LSE is marked as `already_processed` and does not match again!



EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Performance Evaluation – Wireshark

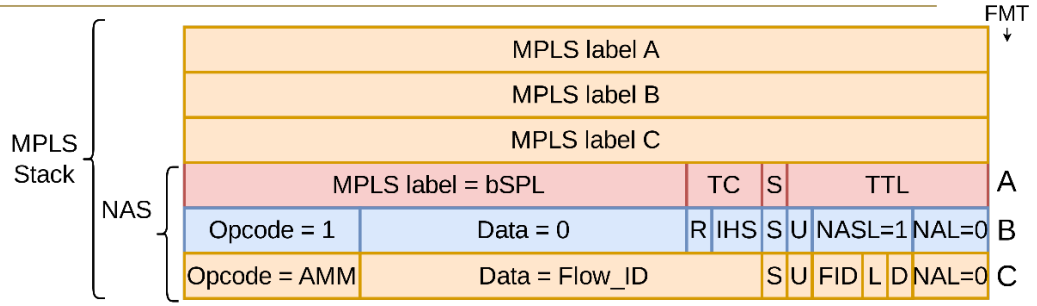




- Network actions were applied!

Evaluation – AMM Overhead

- 12 bytes required to encode AMM
  - 1 Format A LSE
  - 1 Format B LSE
  - 1 Format C (AMM) LSE



- Depending on RLD of LSR, a HBH copy must be placed in the stack
  - 24 bytes of overhead

- Maximum throughput at ingress LER reduced
  - 400 Gb/s of traffic + 12 bytes overhead leads to packet loss
  - How much ingress traffic is possible at ingress without packet loss?

- Not related to P4-MNA but to the MNA design!