IETF Hackathon -
IPFIX On-Path Telemetry with SRv6

IETF 118
4–5 November 2023
Prague, Czech Republic

Hackathon Plan

- **Collect hop-by-hop delay of from SRv6 nodes using following I-D/RFCs:**
  - **IPFIX IEs of SRv6 SRH** (draft-ietf-opsawg-ipfix-srv6-srh): Implemented in IETF 117 hackathon
  - **IPFIX IEs of On-Path Delay** (draft-opsawg-ipfix-on-path-telemetry)
  - **IOAM Edge-to-Edge Option-Type** (RFC 9197)

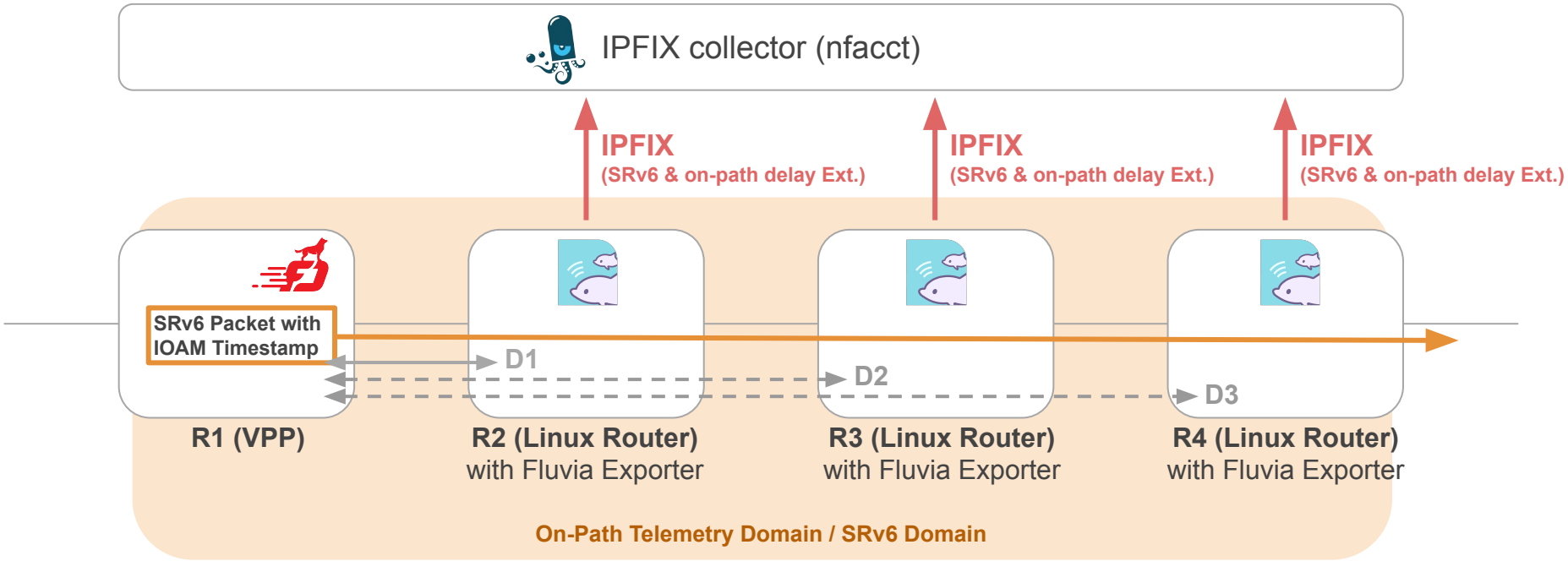
- **Goal: delay collection with IPFIX in SRv6 network with Linux Router**
  - Implement IEs to Fluvia Exporter, an IPFIX exporter with eBPF/XDP
  - Implement additional Wireshark dissectors

What got done

- **Fluvia Exporter:**
  - **COMPLETED: Decode the IOAM Edge-to-Edge Option-Type timestamp with eBPF/XDP**
    - RFC 9197
  - **COMPLETED: Export the on-path delay & SRv6 SRH information with IPFIX**
    - draft-ietf-opsawg-ipfix-on-path-telemetry-04
    - Interoperability test was succeeded with **nfacct and following Wireshark dissector**
- **Wireshark:**
  - **COMPLETED: IPFIX IEs for on-path telemetry**
    - draft-ietf-opsawg-ipfix-on-path-telemetry-04
  - **TODO: IOAM Edge-to-Edge Option-Type timestamp format**
    - RFC 9197

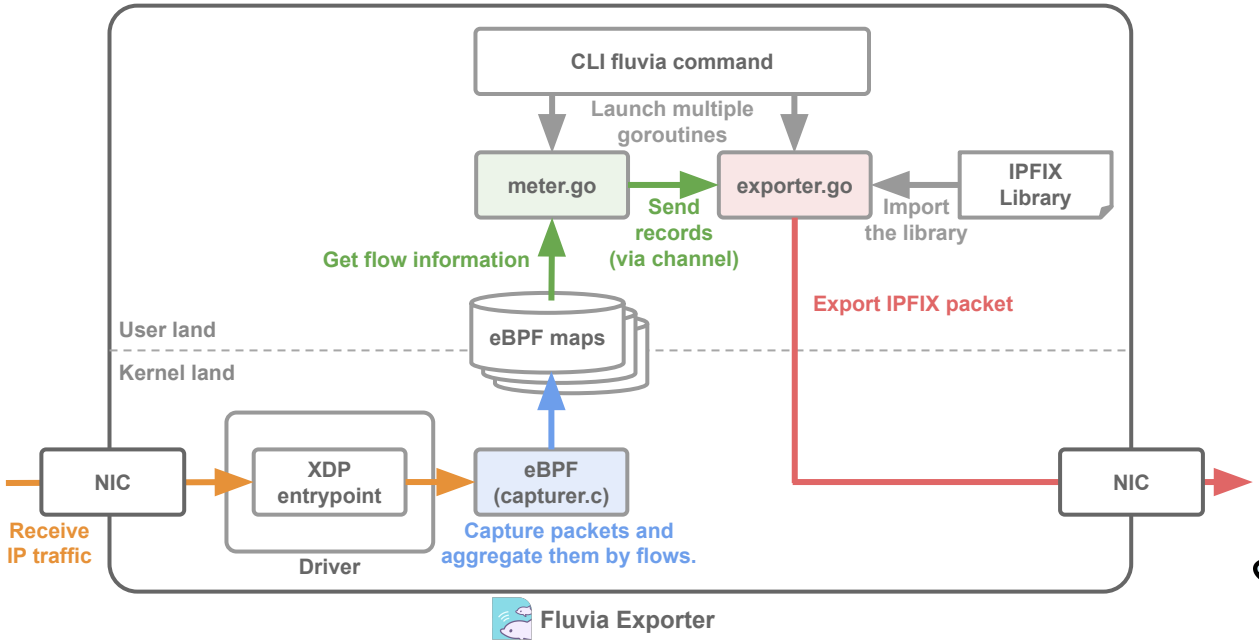
Fluvia Exporter

https://github.com/nttcom/fluvia



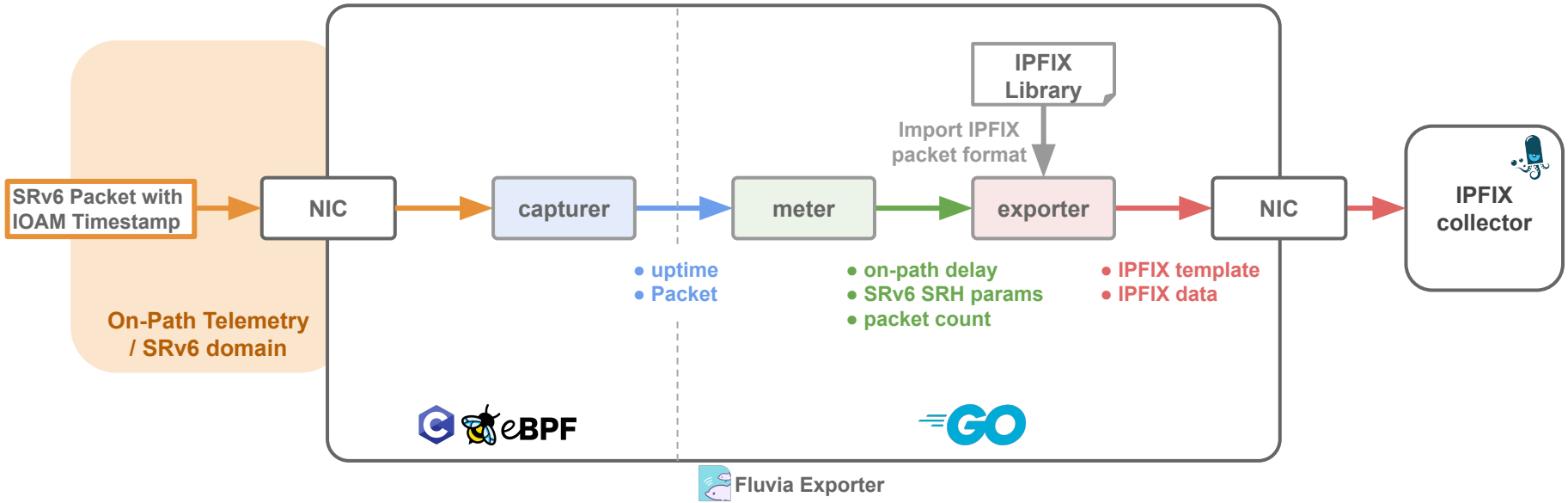
Fluvia Exporter Architecture

- **eBPF/XDP based Linux IPFIX exporter**
- **comprises 3 components (capturer, meter, exporter)**



On-Path Telemetry Pipeline

- capturer: collect IOAM timestamp and timestamp of when the packet was arrived
- meter: calculate the delay from each timestamp, parse the SRv6 SRH
- exporter: generate IPFIX packet, and send to a collector



Wrap-up / Our Code

- Fluvia Exporter: **https://github.com/nttcom/fluvia/pull/21**
  - https://github.com/Yuya9786/fluvia/tree/feature/rebased-ioam6-timestamp

Members

- Wataru Mishima,NTT Com, w.mishima@ntt.com
- Yuta Fukagawa, NTT Com, y.fukagawa@ntt.com
- Yuya Tajima, NTT Com, yuya.tajima@ntt.com
- Motoki Takenaka, NTT Com, m.takenaka@ntt.com