IETF Hackathon

Vector Commitment based Proof of Transit

IETF 118

4–5 November 2023

Prague, Czech Republic

Hackathon Plan

- We designed and implemented a new **Proof-of-Transit** mechanism
  - **What is Proof-of-Transit:** Proving that a packet has traversed a series of physical or virtual nodes, in a specific order.
  - **Drafts involved:** draft-ietf-sfc-proof-of-transit-08
  - **What we achieved:** Providing a working alternative, but more efficiency and security.

What got done

- **Result:** A working **Proof-of-Transit** solution
  - **New ideas:** It can help audit or monitor routing path.
  - **New code** **(demo inside):** https://github.com/liuchunchi/vcpot-demo
  - **New design:** Built on a newer cryptographic primitive:
    - **KZG polynomial commitment** (a construction to vector commitment)
    - As compared to: **Shamir Secret Sharing** in draft-ietf-sfc-proof-of-transit-08
  - **New results:**
    - **Constant size of transit proof** regardless of routing path length **(24Byte)**
    - **Constant computation time of transit proof** regardless of path length **(1-2ms)**

What we learned

- **Vector Commitment** is a interesting primitive to commit a routing path and verify actual execution result afterwards.
- **To OPSEC WG:**
  - Proof of Non Transit is hard, and we cannot do that.
  - **We re-distilled better use cases to be presented in SECDISPATCH**
- **To the concluded SFC WG:**
  - We developed a SFC proof of processing solution after you closed, sorry

Wrap Up

Team members:

Peter (Chunchi) Liu

First timers @ IETF/Hackathon:

Peter (Chunchi) Liu

Contacts:

liuchunchi@huawei.com