Towards RG last call for draft-irtf-cfrg-vdaf

IETF 118 - CFRG - Christopher Patton
Verifiable Distributed Aggregation Functions (VDAFs)

- Delegated multi-party computation based on (function) secret sharing
  - Privacy in the presence of malicious Collector and all but one Aggregator
  - Robustness in the presence of malicious Clients
- Prio3: Simple aggregate statistics from Fully Linear Proofs (FLPs)
  - Prio3Count, Prio3Sum, Prio3SumVec, Prio3Histogram, other unspecified variants
- Poplar1: Heavy hitters from Incremental Distributed Point Functions (IDPFs)
- Room for other MPC techniques (potentially requiring multiple rounds; Client as untrusted Dealer)
draft-02 (feature completeness)

- Initial version of Prio3 (based on [CGB17, BBCG+19]) and an implementation
- Initial version of Poplar1 (based on [BBCG+21])
draft-02 ⇒ draft-03

- Review from Henry Corrigan-Gibbs
  - Fix an attack on robustness of *Prio3* (joint randomness computation) (*)
  - Introduce codepoints to distinguish VDAFs (*)
Review from Hannah Davis et al. \cite{DPRS23}

- Align security considerations with the formal model
- Various robustness improvements (*)
- Restrict aggregation parameter usage (to mitigate attacks on privacy)
- Adopt cSHAKE128 (need random oracles most of the time) (*)
- Note: no direct proof for Poplar1 (Prio3 only)

Implementation of Poplar1 from David Cook (now co-editor)

- Guidance for constant-time implementations of IDPF
Feedback on IDPF from Xiao Wang

- IDPF optimization
  - Hashing operations dominate the runtime (cSHAKE128 is expensive)
  - Observation: Don't need a random oracle for privacy
  - Replace cSHAKE128 with a fixed-key mode of operation for AES from [GKWWY20] (*)
draft-05 ⇒ draft-06

- Review from Chris Wood, Eric Rescorla, and Shan Wang
  - Streamlined interface between DAP and VDAF ("ping-pong")
  - Editorial work
- Usability improvements to Prio3Histogram (*)
draft-06 ⇒ draft-07

- Replace cSHAKE128 with SHAKE128 (*)
  - Easier to implement (many libraries don't implement extensions of SHA-3)
- Define **Prio3SumVec**
- Optimize **Prio3Histogram** (*)
- Editorial work (Chris Wood review)
- Fix bugs in the ping-pong interface (*)
More optimization for IDPF (*)
  ○ ⅓ less AES calls, at the cost of 1 bit of security

Multi-proof mode for Prio3
  ○ Trade-off: Reduce communication cost (more CPU time)
  ○ Open question: How many proofs are required for the same level of robustness?

Replace SHAKE128 with TurboSHAKE128 (draft-irtf-cfrg-kangarootwelve) (*)
  ○ Up to 20% faster for Prio3

More sophisticated range check for Prio3Sum and Prio3SumVec (*)

Make Prio3Histogram multi-hot (*)

IANA considerations (registry for VDAF algorithm IDs)

Editorial work, solicit additional reviews