Oblivious Pseudorandom Functions (OPRFs) using Prime-Order Groups

Alex Davidson, *Armando Faz Hernández*, Nick Sullivan, Christopher Wood

draft-irtf-cfrg-voprf@ietf.org

Sources: github.com/cfrg/draft-irtf-cfrg-voprf

Data Tracker: datatracker.ietf.org/doc/draft-irtf-cfrg-voprf

OPRF: Oblivious Pseudorandom Function

Two-party 1-RRT protocol between a server and a client

Client holds some input x

Server holds a private key k

y = PRF(k, x)

Oblivious

Client learns y,
but nothing about k.

Server does not learn
anything about x or y.

Verifiable

Client verifies proof that
PRF was computed with k.

Server commits to the
key k, and gives a proof.

Latest Changes

Issue #219 Folding Unblind into Finalize API

Finalize uses Unblind as a subroutine, no intermediate values are exposed

Issue #226 Removed the info parameter

Domain separation must now be provided as part of the input

Issue #239 Updates on proof generation

Improved interface for DLEQ proof of knowledge

Issue #234: Use SHAKE-256 for Decaf group

Implementations of Curve448 are likely accompanied by SHAKE

New Proposal

Issue #225: "Weakly" Verifiable construction without NIZK proofs

- Construction 4 eprint.iacr.org/2020/072
- No explicit proof is transmitted
- Only client-side changes (server does $R = k * T_0$)

$T_0 = \underline{r} * [\ H(x)\ ]$

$T_1 = (^1/\underline{r}) * R$

$T_0 = \underline{r} * [\ H(x) - \underline{s} * G\ ]$

$T_1 = (^1/\underline{r}) * R + \underline{s} * K_{pub}$

- Verify a linear combination of tokens at the cost of one issuance operation

New Issue

Unsafe blinding (eprint.iacr.org/2021/273)

Let G be a group in multiplicative (additive) notation

- Exponential (multiplicative) blinding is safe
- Some uses of multiplicative (additive) blinding are unsafe

Questions?

Sources: github.com/cfrg/draft-irtf-cfrg-voprf

Data Tracker: datatracker.ietf.org/doc/draft-irtf-cfrg-voprf

Issues: github.com/cfrg/draft-irtf-cfrg-voprf/issues