Verifiable Oblivious PRFs

Armando Faz Hernandez
Research Engineer
ask-research@cloudflare.com

MPTS 2023:
NIST Workshop on Multi-party Threshold Schemes
September 28th, 2023
Verifiable Oblivious PRF

Agenda

- What is an OPRF?
- Construction
- Additional Properties
- Threshold Version
- Remarks
Oblivious Pseudorandom Function (PRF)

Two-party protocol between a Server holding a key $k$ and a Client holding input $x$ to compute a PRF.

$$y = \text{PRF}_k(x)$$

When protocol ends:

- Client learns the output $y$ of the PRF, and
- **Obliviousness**
  - Client learns nothing about the Server’s key.
  - Server learns nothing about the input nor the output.
Oblivious PRF – Applications

- Private Set Intersection
- Searchable Encryption
- Password-authentication Protocols
  - OPAQUE (uses OPRF as a subroutine)
    - draft-irtf-cfrg-opaque
- Authorization Protocols
  - Privacy Pass (uses VOPRF as a subroutine)
    - draft-ietf-privacypass-protocol
2HashDH – Construction

Jarecki, Kiayias, Krawczyk (2014)

G, an elliptic curve group of order q.

\( H_1 : \{0,1\}^* \rightarrow G \) (hash to curve function)

\( H_2 : \{0,1\}^* \rightarrow \{0,1\}^n \) (hash function)

Client (x)                      Server (k)

\[ r \leftarrow \mathbb{Z}_q \]

\[ a \leftarrow H_1(x)^r \]

\[ b \leftarrow a^k \]

\[ y \leftarrow H_2(x, b^{(1/r)}) \]

https://doi.org/10.1007/978-3-662-45608-8_13
Verifiable Oblivious PRF

Additional Properties

- **Verifiability**
  - Ensure the server used a committed key.

- **Partial-Obliviousness**
  - Additional public input.

- **Updatability**
  - Mechanisms to rotate the key.

- **Threshold Scheme**
  - Key distributed to several parties.
Verifiable Oblivious PRF

2HashDH-NIZK – Verifiable OPRF

Jarecki, Kiayias, Krawczyk (2014)

G, an elliptic curve group of order q.
H₁ : {0,1}ⁿ → G (hash to curve function)
H₂ : {0,1}ⁿ → {0,1}ⁿ (hash function)
DLEQ zk-proof

Client (x)

1. \( r \leftarrow \mathbb{Z}_q \)
2. \( a \leftarrow H_1(x)^r \)

if verify(p):

\( y \leftarrow H_2(x, b^{(1/r)}) \)

Server (k)

1. \( a \leftarrow \) Server

2. \( b \leftarrow a^k \)
3. \( b, p \leftarrow DLEQ_k(g, g^k, a, b) \)

https://doi.org/10.1007/978-3-662-45608-8_13
Additional Properties

- **Verifiability**
  - Ensure the server used a committed key.

- **Partial-Obliviousness**
  - Additional public input.

- **Updatability**
  - Mechanisms to rotate the key.

- **Threshold Scheme**
  - Key distributed to several parties.
**3HashSDHI – Partial Oblivious PRF**

*Tyagi, et al. (2022)*

G, an elliptic curve group of order q.

- \( H_1 : \{0,1\}^* \rightarrow G \) (hash to curve function)
- \( H_2 : \{0,1\}^* \rightarrow \{0,1\}^n \) (hash function)
- \( H_3 : \{0,1\}^* \rightarrow \{0,1\}^{2n} \) (hash function)
- DLEQ zk-proof

<table>
<thead>
<tr>
<th>Client ((x,t))</th>
<th>Server ((k,t))</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r \leftarrow \mathbb{Z}_q )</td>
<td>( a )</td>
</tr>
<tr>
<td>( a \leftarrow H_1(x)^r )</td>
<td>( w \leftarrow k + H_3(t) )</td>
</tr>
<tr>
<td>if verify(p): ( y \leftarrow H_2(t, x, b^{(1/r)}) )</td>
<td>( b \leftarrow a^{1/w} )</td>
</tr>
<tr>
<td></td>
<td>( p \leftarrow \text{DLEQ}_k(g, g^w, b, a) )</td>
</tr>
</tbody>
</table>

https://doi.org/10.1007/978-3-031-07085-3_23
Additional Properties

- **Verifiability**
  - Ensure the server used a committed key.

- **Partial-Obliviousness**
  - Additional public input.

- **Updatability**
  - Mechanisms to rotate the key.
  - See: [https://ia.cr/2019/1275](https://ia.cr/2019/1275)

- **Threshold Scheme**
  - Key distributed to several parties.
Additional Properties

- **Verifiability**
  - Ensure the server used a committed key.

- **Partial-Obliviousness**
  - Additional public input.

- **Updatability**
  - Mechanisms to rotate the key.

- **Threshold Scheme**
  - Key distributed to several parties.
Threshold OPRF

Jarecki, Krawczyk, Resch (2018)

$G$, an elliptic curve group of order $q$.

$H_1 : \{0,1\}^* \rightarrow G$ (hash to curve function)

$H_2 : \{0,1\}^* \rightarrow \{0,1\}^n$ (hash function)

---

Client ($x$)

- $r \leftarrow Z_q$
- $a \leftarrow H_1(x)^r$

i$th$ Server ($k_i$)

- $w \leftarrow k_i \lambda_i$
- $b_i \leftarrow a^w$

$y \leftarrow H_2(x, \Pi b_i^{(1/r)})$

---

https://doi.org/10.1007/978-3-662-45608-8_13
Specification of OPRFs

Work in progress at CFRG/IETF.

Document:

draft-irtf-cfrg-voprf

Describes:

OPRF, VOPRF, POPRF

Ciphersuites:

P-384 & P-521 & Decaf448

Implementations:

C, Go, rust, Typescript, SageMath

https://datatracker.ietf.org/doc/draft-irtf-cfrg-voprf/
Points to Consider

**Goal:** Raise interest in the research, application, as well as in the standardization of OPRFs.

Specification for Threshold OPRFs.
Alternatives and other constructions.
Use cases and applications.
Threshold OPRF as a gadget for other protocols:
  - t-PAKE.
  - t-Authorization Tokens.
Thanks!

Cloudflare Research

ask-research@cloudflare.com

https://research.cloudflare.com