Setting

Problem statement

- Client
- Mediator
- Issuer
Setting

Problem statement

Client

Mediator

Issuer

Fixed per-client private value

Fixed public value

Fixed private value
Requirements

Problem statement

Compute deterministic value \( y \) over private client input \( x \) and private Issuer input \( k \)

\[
y = F(k, x)
\]

Such that

- The Mediator only learns \( y \) if the client engages in the protocol with \( x \);
- The Client cannot engage in the protocol for private input \( x' \neq x \); and
- The Issuer does not learn \( x \), nor when two requests have the same \( x \).
Assume prime-order group with generator $G$ and order $q$, and $x$ and $k$ are private scalars, and $X = xG$ a non-hiding commitment to $x$.

$\pi = \text{NIZK}(DL(x, y) = z)$ is non-interactive Schnorr proof that $\log_z(x) = y$.

$\text{VerifyNIZK}(x, y, \pi)$ outputs 1 for $\pi = \text{NIZK}(DL(x, y) = z)$, and 0 otherwise.
Protocol Overview
Solution sketch

Client

\[ r \leftarrow Z_q \]
\[ P = rX \]

Mediator

**(r, P)**

Issuer

\[ V = kP = krX = krxG \]

\[ y = r^{-1}V = xG \]
Protocol Overview

Solution sketch

Malicious, acts as client to learn $y$

$r \leftarrow Z_q$
$P = rX$

$P$

$V = kP = krX = krXG$

$y = r^{-1}V = xG$
Protocol Overview
Solution sketch

\[ r \leftarrow \$Z_q \]
\[ P = rX \]
\[ \pi = \text{NIZK}(\text{DL}(P, x) = rG) \]
\[ (r, \pi) \rightarrow \text{VerifyNIZK}(rG, P, \pi) \]
\[ (rG, P, \pi) \rightarrow \text{VerifyNIZK}(rG, P, \pi) \]
\[ V = kP = krX = krxG \]
\[ y = r^{-1}V = xkG \]
Protocol Overview

Solution sketch

\[ r \leftarrow \$Z_q \]
\[ P = rX \]
\[ \pi = \text{NIZK}(\text{DL}(P, x) = rG) \]

\[ (r, \pi) \]

VerifyNIZK\((rG, P, \pi)\)

\[ (rG, P, \pi) \]

VerifyNIZK\((rG, P, \pi)\)

\[ V = kP = krX = krXG \]

\[ y = r^{-1}V = xkG \]

Only holder of \(x\) could have generated the request
Questions

Future work

Is the security model sensible?

Does the sketched protocol meet the desired security goals?

Does the protocol compute a PRF?