Another Composition Ownership and Unspentness Proof for Seraphis

coinstudent2048

June 21, 2022

Abstract

One component of Seraphis [4, 1] is the ownership and unspentness proof. A composition proving system for Seraphis is presented that is simply an instantiation of the Modified Chaum-Pedersen Proving System presented in Appendix A of [3]. Consequently, the proving system is complete, special sound, and special honest-verifier zero knowledge (SHVZK).

1 Public parameters

Let $\mathbb{G}$ be a prime order group where the Discrete Logarithm (DL) and Decisional Diffie-Hellman (DDH) problems are hard, and let $\mathbb{F}$ be its scalar field. Let G, X, U be generators of $\mathbb{G}$ with unknown DL relationship to each other. Note that these generators may be produced using public randomness. Let $\mathcal{H} : \{0,1\}^* \to \mathbb{F}$ be a cryptographic hash function. We assume that $\mathcal{H}$ is a random oracle, hence we work in the random oracle model.

The notation $\xleftarrow{\$}$ will be used to denote for a uniformly randomly chosen element, and $(1/x)$ for the modular inverse of $x \in \mathbb{F}$. Lastly, additive notation is used for group operations.

2 Composition Proving System

The composition proving system is a protocol for the relation:

$$\Big\{\big(G, X, U \in \mathbb{G}, \{K_i\}_{i=1}^n, \{\tilde{K}_i\}_{i=1}^n \in \mathbb{G}^n; \{x_i\}_{i=1}^n, \{y_i\}_{i=1}^n, \{z_i\}_{i=1}^n \in \mathbb{F}^n\big) : \\ \bigwedge_{i=1}^n \big(y_i \neq 0 \wedge K_i = x_i G + y_i X + z_i U \wedge \tilde{K}_i = (z_i/y_i)U\big)\Big\}$$

Observe that if $n = 1$, then the relation reverts back to the proving relation shown in Subsection 3.1 of [1], with differences only in notation. In this composition proving system, the Prover only needs to produce one ownership and unspentness proof transcript for all i instead of one proof transcript for each i.

The protocol proceeds as follows:

1. The prover generates $q \xleftarrow{\$} \mathbb{F}$ and $r_i, s_i \xleftarrow{\$} \mathbb{F}$, $\forall i \in \{1, \ldots, n\}$. The prover computes

$$A_1 = qG + \sum_{i=1}^n r_i X + \sum_{i=1}^n s_i U$$
$$A_{2,i} = r_i \tilde{K}_i - s_i U \ , \ \forall i \in \{1, \ldots, n\}$$

   and sends these values to the verifier.

2. The verifier sends a challenge $c \xleftarrow{\$} \mathbb{F}$ to the prover.

3. The prover computes the responses:

$$
\begin{aligned}
t_1 &= q + \sum_{i=1}^{n} c^i x_i \\
t_{2,i} &= r_i + c^i y_i \ , \ \forall i \in \{1, \ldots, n\} \\
t_3 &= \sum_{i=1}^{n} (s_i + c^i z_i)
\end{aligned}
$$

and sends these values to the verifier.

4. The verifier checks the following equalities. If any of them fail, then the prover has failed to satisfy the composition proof system.

$$
\begin{aligned}
A_1 + \sum_{i=1}^{n} c^i K_i &= t_1 G + \sum_{i=1}^{n} t_{2,i} X + t_3 U \\
\sum_{i=1}^{n} A_{2,i} &= \sum_{i=1}^{n} t_{2,i} \tilde{K}_i - t_3 U
\end{aligned}
$$

Using $\mathcal{H}$, it should be straightforward to apply Fiat-Shamir heuristic [2] to the above protocol to make it non-interactive.

The above protocol is basically the Modified Chaum-Pedersen Proving System presented in Appendix A of [3], except that the $U = x_i T_i + y_i G$ (this is in their notation) in the proving relation there becomes $0 = x_i T_i - y_i G$ here. Hence, the proof that the above protocol is complete, special sound, and SHVZK is essentially the same as in the original.

Lastly, to aid in cross-checking, the notation changes (indicated by $\rightarrow$) from the original in [3] to here is shown below.

$$
\begin{aligned}
S_i &\rightarrow K_i & \qquad T_i &\rightarrow \tilde{K}_i \\
x_i &\rightarrow y_i & F &\rightarrow X \\
y_i &\rightarrow z_i & G &\rightarrow U \\
z_i &\rightarrow x_i & H &\rightarrow G
\end{aligned}
$$

References

[1] coinstudent2048. A report on seraphis. `https://github.com/coinstudent2048/writeups/blob/main/seraphis.pdf`.

[2] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *Advances in Cryptology - CRYPTO '86, Santa Barbara, California, USA, 1986, Proceedings*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194. Springer, 1986.

[3] Aram Jivanyan and Aaron Feickert. Lelantus spark: Secure and flexible private transactions. Cryptology ePrint Archive, Report 2021/1173, 2021. `https://ia.cr/2021/1173`.

[4] UkoeHB. Seraphis: Privacy-focused tx protocol. `https://github.com/UkoeHB/Seraphis`.