Investigation on the Effect of Frozen Heart Vulnerability on ZKFlow

Gamze Tillem

May 2022

Recently, researchers in Trail of Bits disclosed a vulnerability that can break implementations of several zero-knowledge proof schemes. The vulnerability, Frozen Heart, is caused by insecure implementation of Fiat-Shamir transformation [FS86] which might enable malicious parties to forge proofs.

The vulnerability affects several implementations of PlonK [GWC19] and Bulletproof [BBB+18] zero-knowledge proof systems. In PlonK, the vulnerability had existed in an earlier version of the protocol but was already fixed back in 2020. Thus, it only affects some implementations of the proof system. In bulletproofs, on the other hand, the vulnerability is caused by a mistake in the original paper and fixed after this disclosure.

After the disclosure of Frozen Heart, as ZKFlow team, we investigated whether it affects the security of our protocol. Our research shows that the vulnerability neither affects the proof system or the specific implementation of the proof system we use. In the following, we provide more information about our analysis.

In ZKFlow, currently we use Zinc ZKP compiler [1] from Matterlabs[2] to implement zero-knowledge proof circuits. Zinc uses Jens Groth's zk-SNARK system, a.k.a. Groth16 [Gro16], as the underlying ZKP scheme. Groth16 is a non-interactive proof system which uses a Common Reference String (CRS) to perform proof operations. Non-interactive zero-knowledge proofs can be obtained in two ways:

- Through trusted setup, where a trusted party generates the CRS that is going to be used during proving and verification.
- Through Fiat-Shamir transformation that is applied to an existing interactive proof system.

The two options can differ security guarantees, trust assumptions, and also theoretical security frameworks. In the case of Groth16, Fiat-Shamir transformation, which is the source of Frozen Heart vulnerability, is not required in the implementation of the protocol since non-interactivity is achieved via CRS

[1] `https://github.com/matter-labs/zinc`
[2] `https://matter-labs.io`

model. Therefore, we can confidently conclude that the vulnerability does not affect current version of the ZKFlow protocol which is built on Groth16.

References

[BBB+18] Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Gregory Maxwell. Bulletproofs: Short proofs for confidential transactions and more. In *IEEE Symposium on Security and Privacy*, pages 315–334. IEEE Computer Society, 2018.

[FS86] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194. Springer, 1986.

[Gro16] Jens Groth. On the size of pairing-based non-interactive arguments. In *EUROCRYPT (2)*, volume 9666 of *Lecture Notes in Computer Science*, pages 305–326. Springer, 2016.

[GWC19] Ariel Gabizon, Zachary J. Williamson, and Oana Ciobotaru. PLONK: permutations over lagrange-bases for oecumenical noninteractive arguments of knowledge. *IACR Cryptol. ePrint Arch.*, page 953, 2019.