Zero Knowledge Proofs & ID

CONCORDIUM

Background
A central goal of the Concordium blockchain is to provide a built-in identity layer that on one hand allows entities to act privately on the chain and on the other hand enables identification of entities behaving suspiciously. To this end all entities (e.g., users or enterprises) acting on the chain must be identified and as part of this get an identity object issued by a **Identity Provider**. Now, the entity cannot simply show the **Identity Object** on the chain as this would disclose the identity and violate the privacy requirement. Instead, if the entity claims to have a certain identity or certain attributes (e.g., nationality or age) the user must prove that he has an Identity Object in which a trusted Identity Provider vouches for these particular attributes. Again, in order to protect the privacy of the entity these proofs must not reveal any other information than the claimed attributes. This is exactly what the zero-knowledge proofs achieve.

Furthermore, when creating an account it must be proved that the account is created in such a way that it can be de-anonymized by the Identity Provider in cooperation with the Anonymity Revokers in case the account is misused. Zero-knowledge techniques are applied here to ensure that these proofs do not leak information about the identity of the user.

To achieve all this, zero-knowledge proofs are specifically used in the following situations:

- Account holder needs to prove to the **ID provider** that the request for an **Identity Object** is correctly constructed without revealing details of cryptographic keys.
- When creating an account the **account holder** needs to prove knowledge of an **Identity Object** issued by a trusted **ID Provider** and that the account is created correctly based on that **Identity Object**.
- When disclosing properties of attributes of the account holder

Unrelated to the usage withinID, zero-knowledge proofs are also used in encrypted transfers of GTUs.

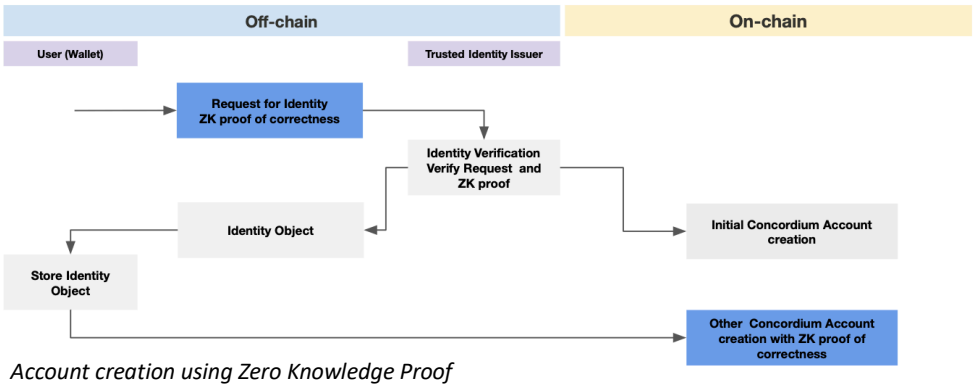
Technology
Concordium uses two types of non-interactive zero-knowledge proofs:

- Classic **Sigma protocols** are used to prove
  - knowledge of the value protected inside a commitment or an encryption
  - linear relationships of committed or encrypted values (equality is a special case).

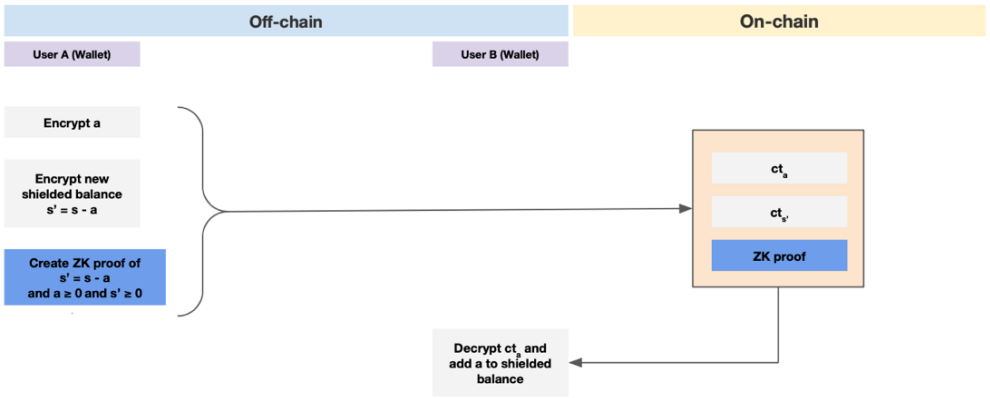
  The **Fiat-Shamir** transformation is used to make the Sigma protocols non-interactive.

- **Bulletproofs** are used to prove that a committed value is within a given range. This is for example used when proving that the entity's age is in a certain range (e.g., above 21).

In the future, SNARKs (such as SONIC) and other techniques will be investigated to efficiently prove more general properties of the attributes.



Account creation using Zero Knowledge Proof



Encrypted transfer (simplified) of amount a from user A to user B.
s= Shielded amount of user A before transaction