Password recovery for Account Abstraction Wallet

This topic proposes a new approach to password recovery for Account Abstraction Wallets, utilizing ZK-SNARKs (Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge).

The central idea behind this proposal is to store the hash(password, nonce)

on the contract wallet. In the event that a user loses their private key, which controls the contract wallet, they can employ their password to generate a zk-proof. This proof serves to verify that the user knows the password and requests a change of the private key. The confirmation process for this change will take approximately 3 days or more. Once confirmed, the contract wallet will update the hash(password, nonce + 1)

.

The zk-proof will contain public fields, including the newHash field. This field represents the updated hash value resulting from the password change process.

By utilizing ZK-SNARKs, the password recovery mechanism ensures that the user's privacy is maintained. The proof only discloses the necessary information to verify the password and update the private key, without revealing any sensitive details about the password itself. This enhances the overall security of the Account Abstraction Wallet system.

In addition to ZK-SNARKs. The extended confirmation period of 3 days or more serves as a safeguard against unauthorized access attempts. It adds an extra layer of protection by introducing a time-based delay, allowing users to regain control of their wallet while mitigating the risk of malicious actors attempting to exploit the recovery process.

Implementing this password recovery mechanism on the contract wallet enhances the user experience by providing an alternative solution to regain access to their wallet in case of a lost private key. It offers a robust and secure approach that balances convenience and privacy, maintaining the integrity of the Account Abstraction Wallet system.