https://text.ru/antiplagiat/68665618818bc

Title: Smart Contract Wallet: What Is It and Pros and Cons

What Are Smart Contract Wallets?

The development of the blockchain system has made **digital asset managemen**t safer and more efficient. A **smart contract wallet** changes the way owners of virtual money (typically Ethereum) deal with it and utilize the potential of programmable contracts. Unlike standard wallets, such storage provides maximum protection, automation, and flexibility. It makes them a mandatory element of the decentralized finance (DeFi) industry. This blog post explains the principles of operation, functionality, and risks associated with advanced storage.

Smart Contract Wallet Overview

It is a category of cryptocurrency wallets that rely on smart contracts – self-executing elements of code that perform predetermined actions when certain terms are met. While standard and custodial wallets only keep and transfer tokens, solutions with codes guarantee automation and extensive protection functions. They can fulfill the specified conditions without intermediaries, which ensures fast and secure operations.

Instead of controlling access to virtual assets with **private keys**, the participant deals with a programmable contract that stores **cryptocurrencies**. Innovative storage simplifies cooperation with blockchain, which enhances user experience and attracts new traders to the sector. They enable you to plan operations and rationally manage capital, tracking where and how virtual money is allocated.

Key Features of Smart Contract Wallets

Advanced wallets utilize flexible code settings to enhance their functionality. Let's consider the main profits and functions supported by such a storage.

- Automation and programmability. An essential feature of wallets is their significant programming potential. While standard storage has a basic set of functions, smart contact solutions enable participants to adjust and automate various financial procedures.
- Interaction with DeFi. Smart contracts are successfully synchronized with DeFi software. Owners can engage in lending, trading, exchanging and other DeFi activities, thereby expanding their financial potential.
- Colossal safety opportunities. Standard wallets are vulnerable to various dangers, whereas programmable storage utilize innovative security tools. Owners can adjust the security level to suit personal needs, adding multi-signatures and time locks to avoid capital loss.

- Decentralized interaction. Smart contract solutions rely on blockchain, which provides owners with decentralized management of their virtual money. This contrasts with centralized vaults, where users delegate asset management to a third party.
- Social recovery. Vaults may be programmed, and social recovery changes the wallet's public key if the owner does not remember the initial secret key. This can be achieved with multi-signature operations with pre-selected trusted persons. With such technologies, you do not have to worry about where to write down the recovery phrase.
- Multicall. It is the ability of a contract account to perform more than one procedure within a transaction. Such a configuration enhances the current wallet functionality, which requires individual confirmation for each action. It increases the processing period and worsens the user experience.
- Allowed and denied addresses. Vaults allow users to grant or restrict access to one or more addresses. Allowlisting enables the owner to grant access, while denylisting blocks access to profiles. Both options enhance security and offer users maximum account control.

Such wallets are code-driven, and you must pay for the computing resources required to run them. Operations in such an ecosystem are more expensive than using a standard vault, especially if it has additional functionality that requires additional code.

How Smart Contract Wallets Work

Such systems utilize a **blockchain**, which means that a network of machines records each transaction. This provides the safety of operations and prevents unauthorized changes to the records. When choosing such a storage, participants generate a non-standard version of the contract, known as a "smart contract".

Let's examine the primary components of the ecosystem.

- Code. It defines the operations' algorithm, security systems, and recovery methods.
- Implementation of operations on the chain. All procedures occur automatically if no conditions are violated. For example, users can limit expenditures on individual digital products.
- Support for **multisignature security**. Multiple signatures are required to confirm operations.
- Monitoring **gas fee**. Some storage solutions suggest gas-free operations with metatransactions.

The traditional workflow with such a storage occurs according to the following algorithm.

- The user launches an operation by selecting conditions through the wallet interface.
- The smart contract examines the sender's permissions and the conditions of the procedure.
- When the system is convinced that no conditions have been violated, it processes the transaction in the blockchain.
- Notification of completion. The user receives notifications after the transaction has been successfully completed.
- The recipient receives crypto coins and does not have to worry about the safety of the operation.

The Ethereum system operates with several account options: external accounts (EOAs), which are protected by a secret key or recovery words, and contract accounts that utilize code. Both solutions are crucial for performing transactions on Ethereum, so let's examine them in more detail.

- External accounts. You may utilize them to perform various tasks. The most famous solution is the **EOA wallet**, that combine a secret key with an address. Many popular non-custodial solutions offer accounts that belong to third parties.
- Contract accounts. Such systems do not have a secret key, but they have an address, a piece of code, and storage space.

When a virtual cash owner synchronizes a contract storage with a **decentralized application (DApp**), the system generates the wallet's smart account address. It is a multi-signature storage with many keys for performing operations.

Smart Contract Wallet Security Risks

As we can see, such storage offers flexibility, but this also creates some downsides. They are based on smart contracts (code created by humans) and face the same risks as self-executing programs. Let's analyze the main categories of dangers.

- Security risks. The code includes operational, implementation, and design risks. Typically, they manifest themselves in the form of illegal transactions, owner privilege escalation, asynchronous processing of operations, etc.
- Compromised secret keys. The marketing campaign for storage is often based on the fact that it does not require asset owners to work with keys. It is crucial to encrypt the keys stored in smart contracts. The code is publicly available, which increases the need for advanced encryption instruments. Regardless of the type of storage you choose, if your keys are compromised, the risk of losing your virtual assets increases. It is up to the participants to decide whether they want to handle their secret keys or delegate such procedures to the code.

- Wallet management outside the owner. When involving a middleman to launch a **smart contract wallet**, the owner may lose full control depending on the storage deployment algorithm.
- Synchronization with DApp. Incorrectly created interactions with decentralized applications make storage vulnerable to suspicious activity.
- Phishing. Criminals may attempt to obtain multi-signature information from storage owners through phishing attacks, potentially leading to the unauthorized withdrawal of virtual assets without the users' knowledge.

The **smart contract wallet** is the latest advancement in the evolution of virtual currency storage. It suggests programmable customization, DeFi interoperability, and adjustable safety. This is a good option for market participants seeking maximum control, **automated transactions**, and the ease of use of virtual cash.