Web3 Solana Blockchain - 08

ALBUS SECURITY

Accounts

Accounts are one of the core concepts in Solana. However, for those coming from an Ethereum (EVM) background, the way accounts function or the need for them can be a bit confusing.

Solana programs are essentially stateless. This means they do not store any information. So, where do all the variables and states get stored? - in the accounts.

Accounts are responsible for holding all types of data on the Solana blockchain. This data can be an executable smart contract as a whole or variables/data associated with the smart contract. Accounts also just refer to the accounts on your wallet that be used to store, send and receive tokens.

Types of accounts

- Non-executable account: Stores different types of data such as program variables, user balances, etc.
- Executable account: This is used only for storing program code.

Account Ownership

Every time we create an account using a wallet, the created account is initialized to be *owned* by a built-in program called the System program. Each account includes some metadata such as owner(to define access control over the account), is executable(y/n), and other such parameters.

Every transactions have own path, just like a full-duplex, but in other chains like eth so it's like a half-duplex which makes solana faster.

Everything is an account

We have accounts like data which includes things like wallets balances or other arbitrary information like that. All this information is usually in the form of arbitrary data structures programs.

Smart contracts are also accounts. these are just called program accounts and difference between program accounts and data accounts is at progra,

Accounts are executable while data accounts are not executable.

Rent fees are assessed approximately every two days (per epoch) and are based on the size of the account. Accounts that hold a sufficient balance to cover two years' worth of rent are exempt from these fees.

Given that rent fees can gradually deplete an account's balance, software programs should carefully consider whether accounts used for storage need to be rent-exempt. If accounts are not rent-exempt, there is a possibility that they may eventually deplete their lamports (the unit of account in the system) and be deleted by the runtime. Deleted accounts can later be recreated.

To ensure the integrity of certain data storage, it is advisable for accounts to enforce rent-exempt status before writing data. This measure helps to maintain the presence of critical data and prevents potential loss due to account deletion.

Progam Derived Address

In Solana, a program-derived address is an address generated based on the characteristics of a program and certain input data. It is used to uniquely identify accounts that are created and managed by a specific Solana program. Program-derived addresses are a fundamental concept in Solana's account model and are crucial for the proper functioning of smart contracts and decentralized applications (dApps) on the platform.

When a program creates or interacts with an account on Solana, it typically uses a program-derived address. This address is not simply a random address but is computed using a combination of the program's address (derived from the program's executable code) and some additional data or seeds provided by the program.

The purpose of using program-derived addresses is to ensure that specific accounts are controlled and managed only by a particular program, thereby enhancing security and isolation between different smart contracts and dApps running on the Solana blockchain.

Program-derived addresses play a significant role in various aspects of Solana's functionality, such as account ownership, account segregation, and access control, making them an essential building block for the platform's decentralized applications.