Addresses

A Filecoin address is an identifier that refers to an actor in the Filecoin state. All actors (miner actors, the storage market actor, account actors) have an address.

All Filecoin addresses begin with anf to indicate the network (Filecoin), followed by any of the address prefix numbers (0 ,1 ,2 ,3 ,4) to indicate the address type. There are five address types:

Address prefix Description 0 An ID address. 1 ASECP256K1 public key address. 2 An actor address. 3 ABLS public key address. 4 Extensible, user-defined actor addresses.f410 addresses refers to Ethereum-compatible address space, eachf410 address is equivalent to an0x address. Each of the address types is described below.

Actor IDs

All actors have a short integer assigned to them byInitActor , a unique actor that can createnew actors. This integer that gets assigned is the ID of that actor. AnID address is an actor's ID prefixed with the network identifier and the address type.

Actor ID addresses are notrobust in the sense that they depend on chain state and are defined on-chain by theInitActor . Additionally, actor IDs can change for a brief time after creation if the same ID is assigned to different actors on different forks. Actor ID addresses are similar to monotonically increasing numeric primary keys in a relational database. So, when a chain reorganization occurs (similar to a rollback in a SQL database), you can refer to the same ID for different rows. The expected consensus algorithm will resolve the conflict. Once the state that defines a new ID reaches finality, no changes can occur, and the ID is bound to that actor forever.

For example, the mainnet burn account ID address,f099 , is structured as follows:

```

Copy Address type | f 0 9 9 | | | Actor ID | Network identifier

```

ID addresses are often referred to by their shorthandf0 .

Public keys

Actors managed directly by users, like accounts, are derived from a public-private key pair. If you have access to a private key, you can sign messages sent from that actor. The public key is used to derive an address for the actor. Public key addresses are referred to asrobust addresses as they do not depend on the Filecoin chain state.

Public key addresses allow devices, like hardware wallets, to derive a valid Filecoin address for your account using just the public key. The device doesn't need to ask a remote node what your ID address is. Public key addresses provide a concise, safe, human-readable way to reference actors before the chain state is final. ID addresses are used as a space-efficient way to identify actors in the Filecoin chain state, where every byte matters.

Filecoin supports two types of public key addresses:

- secp256k1 addresses
- that begin with the prefixf1
- .
- BLS addresses
- that begin with the prefixf3
- .
-

For BLS addresses, Filecoin usescurve bls12-381 for BLS signatures, which is a pair of two related curves,G1 andG2 .

Filecoin usesG1 for public keys, as G1 allows for a smaller representation of public keys andG2 for signatures. This implements the same design as ETH2 but contrasts with Zcash, which has signatures onG1 and public keys onG2 . However, unlike ETH2, which stores private keys in big-endian order, Filecoin stores and interprets private keys in little-endian order.

Public key addresses are often referred to by their shorthand,f1 orf3 .

Actors

Actor addresses provide a way to create robust addresses for actors not associated with a public key. They are generated by taking asha256 hash of the output of the account creation. The ZH storage provider has the actor addressf2plku564ddywnmb5b2ky7dhk4mb6uacsxuuev3pi and the ID addressf01248 .

Actor addresses are often referred to by their shorthand,f2 .

Extensible user-defined actors

Filecoin supports extensible, user-defined actor addresses through thef4 address class, introduced in Filecoin Improvement Proposal (FIP) 0048 . Thef4 address class provides the following benefits to the network:

- A predictable addressing scheme to support interactions with addresses that do not yet exist on-chain.
- User-defined, custom addressing systems without extensive changes and network upgrades.
- Support for native addressing schemes from foreign runtimes such as the EVM.
-

Anf4 address is structured asf4f , where is the actor ID of theaddress manager , andis the arbitrary actor ID chosen by that actor. Anaddress manager is an actor that can create new actors and assign anf4 address to the new actor.

Currently, perFIP 0048 ,f4 addresses may only be assigned by and in association with specific, built-in actors calledaddress managers . Once users are able to deploy custom WebAssembly actors, this restriction will likely be relaxed in a future FIP.

As an example, suppose an address manager has an actor ID (anf0 address)123 , and that address manager creates a new actor. Then, thef4 address of the actor created by the address manager isf4123fa3491xyz , wheref4 is the address class,123 is the actor ID of the address manager,f is a separator, anda3491xyz is the arbitrary chosen by that actor.

Previous Actors Next Blocks and tipsets

Last updated6 months ago