CLI - Managing keys

Managing keys is an essential part of working with the Ledger, since all interactions are authenticated with these keys.

How to add keys

If you want tocreate a new local key , you would need to run the following command:

fetchd

keys

add

< your_key_nam e

> ℹ These keys are stored locally on your system. By default, these keys will be stored in the OS level keychain, however, in general these keys are considered less secure than using a hardware device After running the above command,fetchd will print out a summary of the new key being produced. An output example is shown below:

- name: test type: local address: fetch142tawq2rj397mctc3jtw9dfzf03ns0ze4swat0 pubkey: fetchpub1addwnpepqvtmze0ekffynnjx9n85g6sexzl49ze2vpgc2f52fteyyghjtvvqw682nkx mnemonic: "" threshold: 0 pubkeys: [] This will be followed by a 24-word mnemonic that can be used to re-generate the private key and address for the account.want

⚠ Keep this safe if ever use it to control Mainnet tokens!

How to look up for an address

A common operation that you may want to carry out is to research the address related to specified key. This can be performed quickly, by running following command:

fetchd

keys

show

-a

< name

of

ke y

> The expected output should be as shown below:

fetch142tawq2rj397mctc3jtw9dfzf03ns0ze4swat0 A less common operation, but still useful, would be to look for the public key related to a specified key. This can be achieved with the following command:

fetchd

keys

show

-p

< name

of

the

ke y

> The expected output should be similar to what shown below:

fetchpub1addwnpepqvtmze0ekffynnjx9n85g6sexzl49ze2vpgc2f52fteyyghjtvvqw682nkx

How to list keys

If you wish to research for more detailed information for all keys within your system, simply use the following command:

fetchd

keys

list This will show all your keys information in ayaml format similar to the one generated when you first created the key:

-

name:

test type : local address:

fetch142tawq2rj397mctc3jtw9dfzf03ns0ze4swat0 pubkey:

fetchpub1addwnpepqvtmze0ekffynnjx9n85g6sexzl49ze2vpgc2f52fteyyghjtvvqw682nkx mnemonic:

"" threshold:

0 pubkeys: []

How to recover a key

You can import a key from a 24-word mnemonic by running the following:

fetchd

keys

add

< nam e

--recover

> Enter your bip39 mnemonic < type or paste your mnemonic
>
> You will be prompted to enter themnemonic phrase , and it will then print the matching address and key details as when creating a new key.

Hardware wallets

Set up your hardware wallet

We recommend hardware wallets as a solution for managing private keys!

The Fetch ledger is compatible withLedger Nano hardware wallets . To use your Ledger Nano, you will need to go through the following steps:

1. Set-up your wallet by creating a PIN and passphrase, which must be stored securely to enable recovery if the device is lost or damaged.
2. Connect your device to your PC and update the firmware to the latest version using the Ledger Live application.
3. Install the Cosmos application using the software manager (Manager > Cosmos > Install).

How to add a new key

To use the hardware wallet address with the CLI, the user must first add it viafetchd . This process only records the public information about the key. You canimport the key by first plugging in the device and enter the device pin. Once you have unlocked the device, navigate to the Cosmos app on the device and open it.

Then, you can **add the key using the following command:

fetchd

keys

add

< name

for

the

ke y

--ledger

--index

0 ℹ The--ledger flag tells the command line tool to talk to the ledger device and the--index flag selects which HD index should be used. When running this command, the Ledger device will prompt you to verify the generated address. Once you have done this. you will get an output in the following form:

-

name:

hw1 type : ledger address:

fetch1xqqftqp8ranv2taxsx8h594xprfw3qxl7j3ra2 pubkey:

fetchpub1addwnpepq2dulyd9mly3xqnvfgdsjkqlqzsxldpdhd6cnpm67sx90zhfw2ragk9my5h mnemonic:

"" threshold:

0 pubkeys: []

Was this page helpful?

CLI - Introduction CLI - Managing tokens