

Secret IBC setup

Two local secrets can Inter-blockchainly communicate with each other via a Hermes relay. For deployment of production level relaying services and config/guide information for both RLY and Hermes please refer to -->[IBC relayers documentation](#)

Secret IBC setup

In this demo, you will learn how to create two LocalSecret docker containers that can send IBC token transfers to each other using a Hermes relay. This tutorial assumes that you already have [SecretCLI](#) installed on your machine. In order to follow along, [clone the Secret-IBC-setup repo](#) before proceeding.

Next, install [Hermes relay](#). You can use the hermes configuration found in the [Secret Labs examples repository](#) for your configuration of Hermes.

If you are using a Mac, you may need to press Command + Shift + . (the period key) in order to see your Hermes configuration file.

The configuration file can be found in HOME/.hermes/bin

Bootstrapping LocalSecret

Make sure you have [Docker](#) installed and running. Then, cd into ./secret-ibc-setup/relay and run:

```
...
```

Copy docker compose up

```
...
```

This launches two LocalSecret containers:

- localsecret-1 on port 26657
- localsecret-2 on port 36657
-

Now let's establish an IBC channel with Hermes between the LocalSecret containers in order to make an IBC token transfer.

If you've never funded a Hermes wallet before [learn how to do so here](#).

If you would like to do further reading, refer to the [LocalSecret wallet docs](#) and the [Hermes wallet docs](#).

Creating the IBC transfer channel

To create an IBC transfer channel, make sure both LocalSecret chains are running in Docker, open a new terminal window in your code editor, and then run:

```
...
```

Copy hermescreatechannel--a-chainsecretdev-1--b-chainsecretdev-2--a-porttransfer--b-porttransfer--new-client-connection

```
...
```

SecretCLI will ask you to confirm the transaction before signing:

IBC transfer message in LocalSecret Query that the transaction was successful:

```
...
```

Copy secretcliquerytx

```
...
```

You have successfully created an IBC transfer channel on channel-0.

Next, start Hermes:

```
...
```

Copy hermes start

```
...
```

Send IBC Token transfers

Assuming you have the wallet key 'a' which is not the relayer's key, run:

...

Copy

be on the source network (secretdev-1)

secretcli config node http://localhost:26657

check the initial balance of a

secretcli q bank balances secret1ap26qrlp8mcq2pg6r47w43l0y8zkqm8a450s03

transfer to the destination network

secretcli tx ibc-transfer transfer transferchannel-0 secret1ap26qrlp8mcq2pg6r47w43l0y8zkqm8a450s03 1 uscr --from a

...

Query that the transaction was successful:

...

Copy secretcli query tx

...

If you run into the error `cannot send packet using client (07-tendermint-0) with status Expired: client is not active` it is [because the LocalSecret trusting period is 79 seconds](#).

To resolve this error you must launch the LocalSecret containers, create the IBC transfer channel, and start Hermes within 79 seconds of each other. Assuming you did not run into an error, switch to the destination network (secretdev-2) and query the bank balance of wallet a to confirm that the transaction was successful:

...

Copy

switch to the destination network (secretdev-2)

secretcli config node http://localhost:36657

check that you have an ibc-denom

secretcli q bank balances secret1ap26qrlp8mcq2pg6r47w43l0y8zkqm8a450s03

...

Congrats You should now see the ibc-denom returned in the transaction query:

IBC-denom transaction query

Last updated 1 month ago On this page * [Secret IBC setup](#) * [Bootstrapping LocalSecret](#)

Was this helpful? [Edit on GitHub](#) [Export as PDF](#)