cnt1-tokens-and-private-dex.md

Cutcoin Private DEX: use cases, mechanics, operations.

Introduction

Since Cutcoin 12 fork we have revolutionary opportunities for private Decentralized EXchange of assets (DEX) with the help of Cutcoin Liquidity pools. New token types are also available for different business cases. In the first part of the paper we investigate Cutcoin CryptoNote Tokens (CNT1), old and upcoming features, and look at DEX in detail in the next section.

Getting the Right Tools

Cutcoin Command Line tools (CLI) provide native support for Cutcoin CryptoNote Tokens and Private DEX. Cutcoin Command Line tools can be downloaded from GitHub. We have precompiled versions for popular platforms and open source code for people who need custom builds.

Cutcoin Testnet is a perfect sandbox for experiments with tokens and DEX. You may obtain your testnet cutcoins for free here.

Cutcoin CryptoNote Tokens

Token Types and Use Cases

At the time of publication there are 4 types of tokens supported on Cutcoin blockchain, and any issued token belongs to one of them:

1. Public token
2. Private token
3. Mintable token
4. Liquidity pool (LP) token

Public tokens are tokens with a fixed size supply intended to represent user assets. In creating a Public token, the user specifies its name and the supply size, and any other Cutcoin user can see this data right after the token genesis transaction is submitted.

Private tokens are pretty similar to Public tokens except that the token creator is the only person who knows the supply size. At the same time, the token name is public. This type of tokens can be used, for example, for representation of assets shared between several users with the share size not known for other participants.

Mintable tokens are also similar to Public tokens, but in addition to the basic properties allow the creator of a token to increase its supply. It can be increased an unlimited number of times for an arbitrary amount (but anyway, the summary supply cannot exceed the maximum allowed supply for tokens that is equal to 1844674407.0000000000).

Minting transactions contain a special message signed with the token private key that proves the issuer has the privilege to mint the supply. Token private key, in its turn, is a unique key that the token creator receives when creating the token. He can preserve this key in a secret place or share it with someone else; everybody who knows the key has an ability to mint the token supply.

Liquidity pool tokens are tokens of special kind intended to account liquidity volume provided by users to Liquidity pools. We describe Liquidity pools mechanics further on, for now it's important to note that each time a Liquidity pool is created a unique LP token is bined to it. When a user is adding to a pool liquidity he receives some amount of the corresponding LP token in the wallet. The exact amount depends on the pool liquidity and is proportional to the amount of the provided liquidity. When a user is taking from a pool liquidity he sends LP tokens from his wallet and receives liquidity from the pool.

One important difference between LP tokens and tokens of other types is who receives them after creation (submitting of token genesis transaction). Right after creation, public, private and mintable tokens come to an issuer's wallet. In contrast, LP tokens are stored on a special purpose LP Wallet. Initial LP token supply is always 1844674407 tokens (max allowed token supply). Cutcoin daemon guarantees that a user can receive LP tokens from this wallet only in return for pool liquidity that has been provided.

We emphasize that tokens of all listed types have equal capabilities regarding their accounting, transferring between wallets and usage as an attribute of Liquidity pool pairs. All manipulations with tokens happen as transactions that are explicitly stored in Cutcoin blockchain, so they are entirely decentralized. Transaction fees are always paid in cutcoins.

Transactions basically follow the CryptoNote protocol, therefore have a high level of privacy. We stick to conservative approach when designing new Cutcoin features, this means revealing minimal required information in transactions. For example, token genesis transaction for a Public token reveals its supply, but this is exactly what we need to know about a public token. All following transactions with this token have concealed amounts thanks to the privacy features of RingCT + Bulletproof. See Appendix C for more information.

Working with tokens

CLI Wallet has dedicated commands for creating, transferring and obtaining the information about tokens. The same commands are also available through the RPC interface.

Public token can be created using

```
create_token T101 100000 public
```

Here we create a Public token 'T101' with '100000' tokens supply.

In a similar way a Private token can be created:

```
create_token T102 100000 hidden
```

and mintable token:

```
create_token T103 100000 mintable
```

This command has an output that contains Token private key

```
b16a9595ac8ed2f38fbf6d5f036291df1704b8843f2cf3676233e2cf8df8630f
```

it must be provided each time you mint additional supply, so preserve it in a secret place for further minting.

```
mint_token_supply T103 2000 b16a9595ac8ed2f38fbf6d5f036291df1704b8843f2cf3676233e2cf8df8630f
```

We use a dedicated command 'create_lptoken' in order to create LP token. This is done to highlight their unique role (read more about LP tokens in the DEX section).

```
create_lptoken lpT104
```

Note that LP token names always start with an 'lp' prefix.

Cutcoin DEX

Overview

Cutcoin DEX was conceived as a private decentralized marketplace for asset exchange. The assets can be any CNT1 tokens (tokens that were emitted in the Cutcoin network) and Cutcoin itself. Decentralized exchanges usually use one of the two technology types: they can be based on a list of 'asks' and 'bids' provided by the users, with a matching engine that picks up matching positions and closes them; another possibility is a liquidity pool that has become popular recently. The latter is used in Cutcoin DEX.

Liquidity pool, in a nutshell, contains liquidity in two different currencies (tokens) that was provided by liquidity providers and locked in a special account. Liquidity providers are people who bring their funds and create a new pool or join an existing one. For a given pool, the relation between currency 1 volume and currency 2 volume defines the current exchange rate. Anybody who wants may exchange one of the pool currency for another one with the current exchange rate. When exchange operations are performed, pool applies a small fee (pool interest) Pools accumulate these fees and finally pay it when liquidity providers take their funds back, proportionally to their contribution volume.

It turns out that liquidity pools are profitable for different network participants: liquidity providers give their funds and return them with the accumulated pool interest. Other users can exchange their assets at any time with the defined exchange rate. Stakers benefit from the fact that some part of Cutcoin circulating supply happens to be locked in liquidity pools, therefore their chances to subscribe a block grow and their income does so.

It is worth to note that the newly created Binance network to Cutcoin network bridge allows cross-chain operations, direct liquidity flow into Cutcoin network, and then use Cutcoin private liquidity pools to obtain specific assets.

Liquidity pools in Cutcoin

In order to do bookkeeping and accounting of the funds in liquidity pools, we introduce a Cutcoin special purposes wallet (LP wallet). It is a public wallet, all nodes in the network know its balance and any user may build it from the keys and check its passing transactions. Still users can't just transfer funds from this wallet to their own wallets as Cutcoin daemons now contain strict rules for working with it. LP wallet deposits and withdrawals happen mechanically as a result of DEX operations.

Let's look in detail at Cutcoin liquidity pool life cycle and possible ways of its usage. The first thing to be done when creating a liquidity pool is a new LP token generation. This LP token will be bound to the pool. Its name must be unique, start with an 'lp' prefix and have 8 characters length at most.

The supply of the all newly created LP tokens (1844674407 tokens) is transferred to LP wallet balance, these funds are not on a user balance at this moment. The LP tokens are awarded to liquidity providers in return for the provided liquidity and required to account their shares in the current pool.

After the LP token has been created we can create the pool itself. We need to choose the pair of tokens for trading, for example T1 and T2 (or another pair T1 and Cutcoin) and the volume of the provided liquidity for each of them. Obviously the wallet balance should be enough for that. Liquidity volume of the first and the second token in the pair specify the initial exchange rate; as time passes, depending on the market, the rate may rise or fall. Current exchange rate is equal to the ratio between liquidity volumes in the pair:

```
R = V_T1 / V_T2                                          (1)
```

During the process of a pool creation the selected LP token is bound to the traded tokens pair, that's why one LP token can be used for creation of a single liquidity pool. In addition, in the pool creation transaction funds move in two opposite directions: the specified liquidity T1 and T2 from the current wallet go to the LP wallet, while the LP wallet sends some amount of LP tokens to the current wallet, their exact sum is evaluated as follows:

```
V_lp = sqrt(V_T1 * V_T2)                                 (2)
```

This way of pool liquidity accounting is currently the most popular, thanks to its simplicity and usability.

Each liquidity pool has a name, it is generated automatically according to the simple rules: /, in case with T1 and T2 the name is 'T1/T2'. Names of the tokens are ordered alphabetically, but if one of the currencies in the pool pair is Cutcoin, it is always placed at the second position (underlying currency).

Since a liquidity pool has been created, its creator has no special rights or privileges, in respect to other users. Anybody can add liquidity to the pool and get LP tokens in return, proportionally to the provided volume (2). When adding new liquidity to an existing pool the current exchange rate is used to determine the ratio between liquidity volumes of the first and second tokens in the pair.

Users can return the provided liquidity back from the pool at any time, for doing that they need to send all (or some part) of the owned LP tokens to LP wallet. In return, the funds and the accumulated pool interest come to the user's wallet. The exact amount is evaluated as:

```
T1 = V_lp_requested / V_lp_all_pool * V_T1_all_pool,
                                                         (3)
T2 = V_lp_requested / V_lp_all_pool * V_T2_all_pool,
```

i.e. user receives a fraction of the all funds in the pool equal to his share of all given to users' LP tokens of the current pool.

Working with a pool from a trader's point of view looks attractive. Purchase and sale operations are performed in respect to the first token in the trading pair. The second token is an underlying currency. In other words, the command 'buy T1/T2 50' means that you buy 50 tokens T1 using the current exchange rate and pay tokens T2 for it. Similarly, 'sell T1/T2 10' means selling 10 tokens T1 for T2. The required amount of the second token in the pair is evaluated automatically. Pool interest is also accounted automatically. When buying, pool interest is added to the required amount of the funds for token T2, and when selling, it is subtracted from the earnings.

All discussed operations technically were implemented as ordinary transactions in Cutcoin blockchain. There is an important restriction recently that you need to know about before starting working with liquidity pools. A single block can contain only one token transaction (any of 'create token', 'create lp token', 'mint token supply') or lp transaction (any of 'create liquidity pool', 'add liquidity', 'take liquidity', 'buy', 'sell'). It is related to stages we need to pass when implementing DEX in Cutcoin, but in the future we plan to relax this restriction.

Example.

Let's look at the specific example of a Liquidity pool usage and the required commands. You can reproduce this scenario in Cutcoin testnet, just choose different token names to make them unique.

Start with creation of the minimal set of tokens required to create a liquidity pool:

```
create_token T101 10000
create_token T102 10000
create_lptoken lpL104
```

Then you need to wait until the unlock time has passed for all tokens (10 blocks) and then you can create the pool:

```
create_liquidity_pool T101=200 T102=400 lpL104
```

The initial exchange rate of this pool is 0.5. Now you can run exchange operations:

```
buy T101/T102 2
```

```
sell T101/T102 1
```

And finally withdraw some part of the liquidity:

```
take_liquidity T101/T102 lpL104=50
```

Conclusion

We believe that a new CryptoNote based DEX marketplace should be highly demanded and continue developing it. You may ask questions and share ideas at info@cutcoin.org.

Appendix A. List of the Token and DEX Commands

`token_balance` Show the token(s) balance in the currently selected account.

`transfer_token` Transfer 'amount' of tokens with 'token_id' to 'address'.

`create_token` Create a token with the specified 'token_name' and 'token_supply'. The wallet balance in Cutcoins must be at least equal to 'TOKEN_GENESIS_AMOUNT' value + possible transaction fees. Supported token types are 'hidden' for tokens with the hidden supply, 'public' for tokens with publicly visible supply and 'mintable' for tokens with unlimited supply. The default type is 'public'.

`create_lptoken` Create liquidity pool token with the specified 'token_name'. The wallet balance in Cutcoins must be at least equal to 'TOKEN_GENESIS_AMOUNT' value + possible transaction fees.

`mint_token_supply` Mint additional token supply for the token with the specified 'token_name'. This token must be mintable and created in the current wallet (or 'token_secret_key' must be specified). 'new_token_supply' must be greater or equal to the current token supply and not exceed the maximal token supply.

`get_mintable_token_key` Print secret key for the token with 'token_name'.

`get_tokens` List tokens that have the names starting from the optional 'prefix'.

`create_liquidity_pool` Create liquidity pool for the specified pair of tokens(first_token_name, second_token_name) with the specified liquidity token. The wallet balance in Cutcoins must be at least equal to 'POOL_GENESIS_AMOUNT' value + possible transaction fees.

`add_liquidity` Add liquidity to the pool with the specified 'lp_name'. You must specify one of the tokens from the pool pair and its amount.

`take_liquidity` Take liquidity from the pool with the specified 'lp_name'. You must specify one of the tokens from the pool pair and its amount.

`get_liquidity_pools` List liquidity pools that have the names starting from the optional 'prefix'.

`buy` Buy the specified amount of the first token in the liquidity pool pair. Example: as the result of 'buy T1/cutcoin 1' command, you receive 1 T1 token from 'T1/cutcoin' pool using the current ratio.

`sell` Sell the specified amount of the first token in the liquidity pool pair. Example: as the result of 'buy T1/cutcoin 1' command, you send 1 T1 token from 'T1/cutcoin' pool using the current ratio.

Appendix B. Table of the Operation Costs

Operation name	Cost
Create token	100 Cut
Create LP token	100 Cut
Mint supply for Mintable token	100 Cut
Create Liquidity Pool	100 Cut
Add liquidity to Liquidity Pool	no fee
Take liquidity from Liquidity Pool	no fee
Buy	0.3% of the tx amount
Sell	0.3% of the tx amount

Appendix C. Token / DEX Commands and Privacy

Operation name	Disclosed parameters
Create Public token	Token name, token supply, the fact of the token creation
Create Private token	Token name, the fact of the token creation
Create Mintable token	Token name, token supply, the fact of the token creation
Create LP token	Token name, supply volume (it is constant for all), the fact of the token creation
Mint supply for Mintable token	Minted supply, the fact of the minting
Create Liquidity Pool	Disclose participating unspent output amounts, the fact of LP creation
Add liquidity to Liquidity Pool	Disclose participating unspent output amounts
Take liquidity from Liquidity Pool	Disclose participating unspent output amounts
Buy	Disclose participating unspent output amounts
Sell	Disclose participating unspent output amounts