DESCRIPTION OF THE LIMEX PROTOCOL

The Limex protocol is a protocol for secure direct exchange by tokens and cryptocurrency between decentralized applications (DApp). It is open and public. Anyone can create their own DApp and become a member of the exchange.

The protocol works on Ethereum and its sidechain Limex and organizes an interface for connecting other blockchains. This allows you to add to the list of exchange any cryptocurrencies and tokens.

The exchange algorithms are contained in smart contracts pre-loaded into Ethereum and Limex, as well as in scripts of cross-source oracles. The codes of smart contracts and scripts are open.

At the network level, security is provided by the p2p protocol and the Ethereum and Limex consensus algorithms. At the application level, security is provided by the codes of smart contracts and scripts.

The Limex protocol was based on the following ideas: the account split technology [1], the concept of the Plasma platform proposed by V. Buterin and D. Poon [2], the smart contract BTC-relay [3], the Oraclize service [4], the Interleger protocol [5], 0x protocol [6], DYDX protocol [7] and others.

1. Introduction.

Secure direct exchange of tokens and coins means that the minimum required number of transactions is to be performed without intermediaries. Ideally, these are two transactions, one in each exchange direction. Only this method gives the maximum speed and minimum commission.

However, this is not safe. Due to the different time of recording and confirming transactions in various blockchains, there is a threat of "double waste" and fraud.

The use of multi-signature mechanisms does not solve the problem. The use of complex smart contracts significantly complicates, slows down the exchange and increases its cost. Integration of centralized services violates security and decentralization.

The Limex protocol offers a solution to the problem by providing insurance for each transaction.

The basis of the protocol is the smart contracts Lime (ERC20 token) and Teleport are the same in the blockchain Ethereum and its sidechain Limex.

Limex is a modified Ethereum fork and allows you to use identical smart contracts and a single address space.

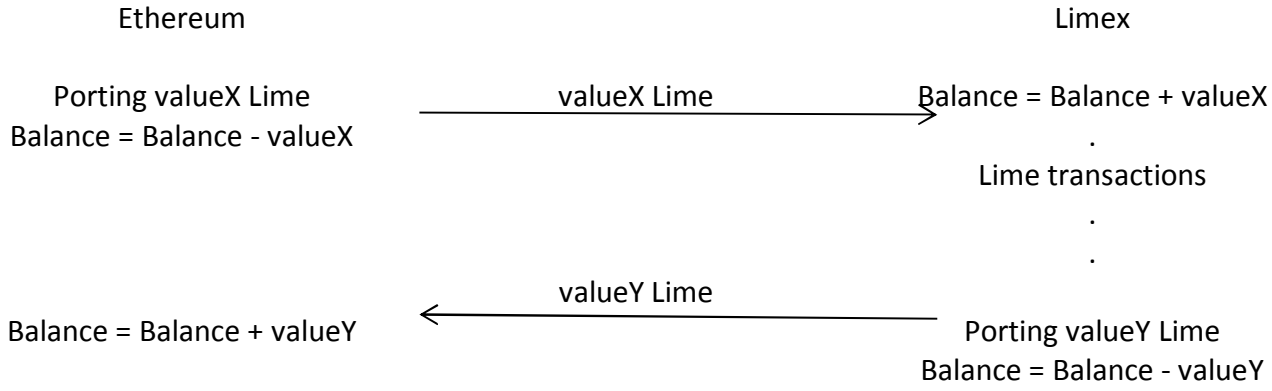
Smart contract Teleport serves to implement the interaction of the blockchain and sidechain. It contains functions that ensure the reliable transfer of Lime tokens between the blockchain and sidechain.

The transfer of the Lime token transactions to the sidechain Limex gives a reduction in their cost and speed increase.

2. Organization of the interaction of the blockchain and the sidechain.

Crosschain interaction is built by porting the Lime token from the blockchain to the sidechain and back. We call this “the account split technology”. Why, it becomes clear from the following scheme.

Scheme of interaction between the blockchain and the sidechain



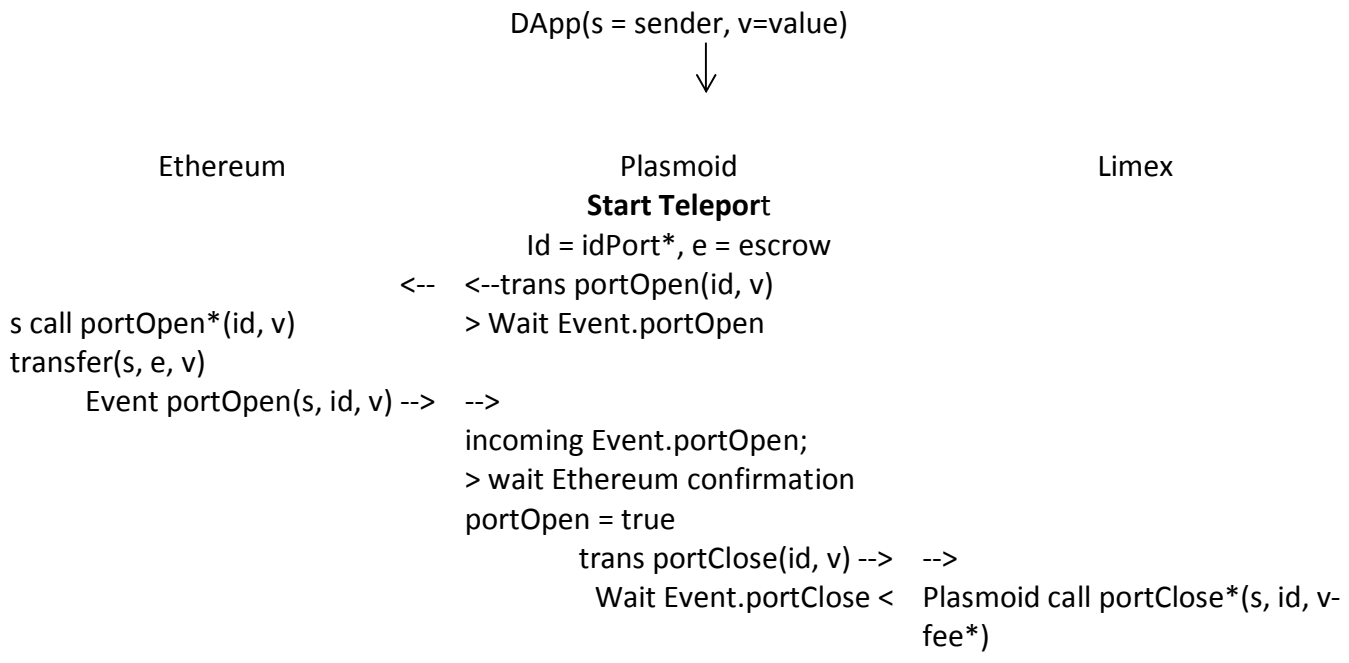
As can be seen from the above diagram, the user at any given time has a certain number of Lime tokens. Some of them are in Ethereum, and some are in Limex. Tokens are identical and can be collected at any time in any one account. This is the account split. More details on the account split technology are described in [1].

A special DApp (crosschain oracle Plasmoid) controls porting. It is a service with a set of logic scripts, providing transfer of signed transactions, requests and responses between DApp, blockchain and sidechain. Plasmoid generates necessary transactions between escrow accounts and user accounts with state control and error handling. It uses the Teleport script to port the tokens.

3. Porting Lime tokens.

Porting Lime tokens from blockchain to Limex is as follows. DApp initiates porting by running the Teleport script. In the blockchain, the amount of Lime is transferred to the escrow account, then, in the sidechain the amount of Lime transfers from the escrow account to the user account. When porting tokens from Limex to Ethereum, operations are performed in reverse order.

The scheme of porting Lime tokens



transfer(e, s, v-fee)
<-- <-- Event portClose(s, id)
incoming Event.portClose;
wait Limex confirmation <
portClose = true

if portOpen & portClose
Stop Teleport true

* - notes:

- idPort - unique port number, assigned automatically;
- portOpen, portClose - functions of the smart contract Teleport;
- fee - Plasmoid commission.

The protocol is synchronous, the transactions in the blockchain and sidechain are started in turn, one by one. This avoids attacks of "double waste", in the event of errors in the execution of the transaction, you can reversibly rollback.

The escrow accounts belong to Plasmoid. They are recorded in the table of the smart contract Teleport. The contract code of the contract and the service are open, anyone can create such an own service. The work of Plasmoid is paid by commissions. This will ensure competition and decentralization.

Plasmoid will provide an API for the integration of third-party services and applications.

This construction of interaction between the blockade and sidechain gives the following advantages:

- complete true decentralization;
- single address space;
- use of one pair of keys stored on the device;
- identical smart contracts;
- simple logic of scripts;
- simple securing of collateral liquidity in the sidechain;
- the possibility of creating DApp, using simultaneously/alternately blockchain and/or sidechain;
- unlimited opportunity to expand due to the integration of third-party services.

Next, schemes for organizing direct safe exchanges of Lime, ERC20, ETH in Ethereum, as well as cross-exchange of cryptocurrencies and tokens, will be described.

4. The scheme of safe exchange of tokens.

The availability of ported Lime tokens allows you to organize a secure, inexpensive and fast exchange of tokens ERC20 Ethereum. Security of exchange is provided by a scheme with a pledge. Smart contract DEx and script DExT (decentralized exchange of tokens) are used for exchange.

Minimization of commissions and acceleration of exchange is ensured by the fact that pledges are made in Limex. Exchange participants must have a sufficient number of Lime tokens in Limex as collateral. Otherwise, the missing amount of Lime must be ported.

The process of exchange is as follows. DApp1 (maker) translates into the p2p network an bid with the idOrder number for the exchange value1.Token1 at value2.Token2. DApp2 (taker) catches an bid and initiates an exchange. The crosschain oracle Plasmoid calculates the size of the collateral in Lime tokens based on cross-rates of tokens and Lime.

It is assumed that the cross-rate exchange of tokens is preliminarily agreed. In the coordination of the cross-rate, a service oracle that provides data from the market can participate. The protocol of coordination of the cross-rate is being developed.

Plasmoid manages a secure exchange. In doing so, he uses the DExT script and the smart contract Depo according to the following scheme.

Secure exchange scheme for ERC20 tokens in Ethereum

DApp1 (m = maker, v1.T1 = value1.Token1, id=IdOrder*)

↓

DApp2 (t = taker, v2.T2 = value2.Token2, id)

↓

Plasmoid	Limex	Ethereum
Start DExT		
// Init pledge		
e = escrow, p = pledge*		
Tx(DEx, id, m, t, p) -->	Plasmoid call DEx(id, m, t, p)	
permission <--	<-- Event startDex(id)	
trans Tx(m, inDepo(id,p)),		
Tx(t, inDepo(id,p))-->	-->	
Wait Event.inDepo(m, t) <	t call inDepo*(id, p)	
<	m call inDepo(id, p)	
	transfer(m, e, p)	
	transfer(t, e, p)	
<--	<-- Event inDepo(m, t)	
incoming Event.inDepo(m, t);		
wait Limex confirmation <		
// Start exchange		
Tx(startTransf, id) -->	-->	
confirmation <--	<-- Event startTransf(id)	
trans Tx(m, transfer, t, v1.T1),		
Tx(t, transfer, m, v2.T2)-->		-->
Wait Event.transer(m, t) <		m call transfer(m, t, v1.T1)
		t call transfer(t, m, v2.T2)
<--		<-- Event transfer(m)
incoming Event.transfer(m);		<-- Event transfer(t)
incoming Event.transfer(t);		
wait Ethereum confirmation <		
confirmation<--		
Tx(stopTransf, id) -->	-->Event stopTransf(id)	
// Cancel pledge		
v=value*		
Tx(outDepo*, m, t, id, v)-->	-->	
Wait Event.outDepo(m, t) <	Plasmoid call outDepo(m, t, id, v)	
	transfer(e, t, v)	
	transfer(e, m, v)	
<--	<--Event outDepo(m,t)	
incoming Event.outDepo(m, t);		
wait Limex confirmation <		
confirmation<--		
Stop DExT true		

* - notes:

- idOrder - unique order number, assigned automatically;
- inDepo, outDepo - functions of the smart contract DEx;
- pledge - the number of Lime tokens for the pledge, including all commissions;
- value - the number of Lime tokens to return to the maker and taker with Plasmoid fee.

This protocol is synchronous. Synchronicity of the protocol: first the collateral is formed (transactions in the sidechain are performed). Then there is an exchange (transactions in the blockchain). Then a security deposit is withdrawn (again transactions in the sidechain). Simultaneous execution of transactions in the blockchain and sidechain is not allowed.

Thus, the basic level of the Limex protocol is:

- blockchain Ethereum;
- sidechain Limex;
- smart contracts Lime (ERC20 token), Teleport, DEx;
- crosschain oracle Plasmoid.

The Limex protocol base layer also provides secure direct exchange of ERC20 to ETH in Ethereum.

To do this, Plasmoid uses a different script - DexT-ETH, which has some differences from the DexT script.

The first and third parts of the protocol of exchange (formation and withdrawal of pledge) are identical. Change is in the second part. Instead of one call to the transfer function of the smart contract of the ERC20 token, the signed ETH transfer transaction is broadcast to Ethereum.

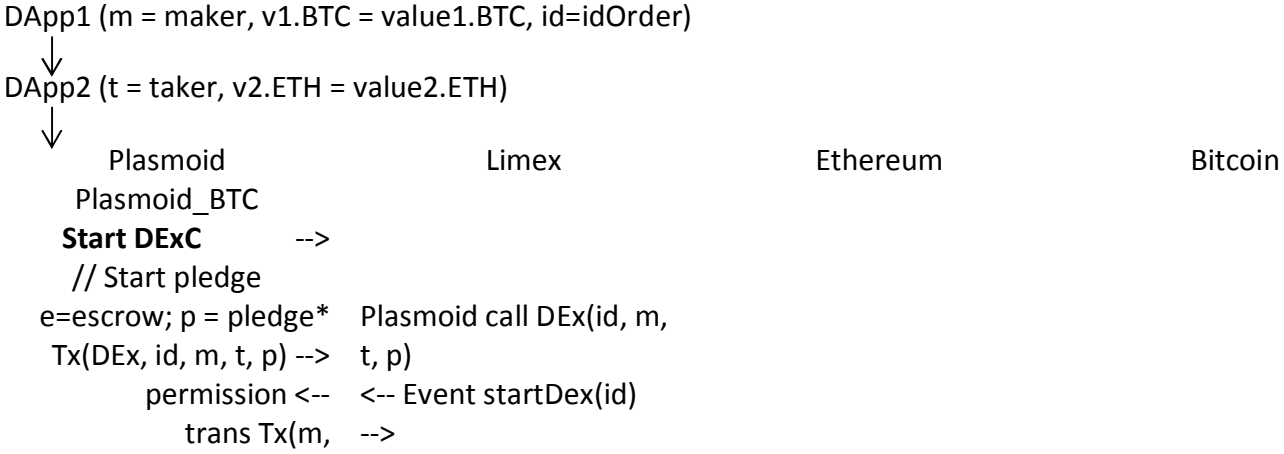
5. The scheme of secure exchange of cryptocurrencies

For the secure exchange of cryptocurrencies, the basic level of the protocol must be supplemented by cryptocurrency connectors. Connectors are extension modules for Plasmoid. Their purpose is to ensure the interaction with the blockchains of the cryptocurrency.

The names of the connectors are standardized and are formed by adding to the Plasmoid the symbol of the coin. Examples: Plasmoid_BTC, Plasmoid_ETC, Plasmoid_XRP, etc. So, for the organization of exchange BTC on ETH the connector Plasmoid_BTC should be added in Plasmoid.

The safe exchange of cryptocurrencies is controlled by the crosschain oracle Plasmoid and the DExC script (decentralized exchange of coins). The protocol for secure exchange of coins is similar to the protocol for secure exchange of tokens. Its scheme is shown in the example of the BTC-ETH exchange.

Secure Exchange Scheme BTC-ETH



inDepo(id,p)),	t call inDepo(id, p)		
Tx(t, inDepo(id,p))-->	m call inDepo(id, p)		
Wait Event.inDepo(m,t)<	transfer(m, e, p)		
	transfer(t, e, p)		
	<-- Event inDepo(m, t)		
<--			
incoming			
Event.inDepo(m, t);			
wait Limex			
confirmation<			
confirmation<--	-->		
// Start exchange	<-- Event startTransf(id)		
Tx(startTransf, id) -->			-->
confirmation <--		-->	Tx(m, t, v1.BTC)
trans Tx(m, t, v1.BTC) -->		Tx(t, m, v2.ETH)	
trans Tx(t, m, v2.ETH) -->			
wait Ethereum			
confirmation <			
confirmation ETH<--			
wait Bitcoin	-->		
confirmation<	<-- Event stopTransf(id)		
confirmation BTC<--			
Tx(stopTransf, id) -->			
confirmation <--	-->		
// Cancel pledge	Plasmoid call		
v = value	outDepo(m, t, id, v)		
Tx(outDepo*, m, t, id, v)-	transfer(e, t, v)		
->	transfer(e, m, v)		
Wait Event.outDepo(m,	<--Event outDepo(m,t)		
t) <			
<--			
incoming			
Event.outDepo(m, t);			
wait Limex confirmation			
<			
confirmation<--			
Stop DExC			

A small modification of this scheme provides a safe exchange of ERC20 tokens for coins. The difference is that Plasmoid calls the DexT_C script and the transaction is translated into Ethereum with the call of the ERC20 token transfer function.

6. Security

Secure direct exchange of tokens and coins without intermediaries implies the execution of the minimum required number of transactions. In the Limex protocol, there are two transactions, one in each exchange direction, secured by collateral. This gives the maximum speed and minimum commission, ensuring security.

The exchange of coins and tokens can be interrupted (accidentally or intentionally) in such a way that one of the parties will suffer damage. Smart contract DEx writes an exchange's log. The Limex protocol will provide for an arbitration procedure to voluntarily adjust the balance sheets of the parties. Balances will be automatically adjusted in Limex at the expense of pledges if the consent is not achieved.

An important element of protocol security is ensuring the security of crosschain interaction. For this, the smart contract Teleport is logging port. In the case of incidents, it is always possible to trace the progress of porting. This will restore the correct course of operations if necessary.

Another way to ensure the security of crosschain interaction is to ensure the work of crosschain oracles Plasmoid. Smart contracts controls the access to the oracles. Plasmoid is allowed to work if it has a sufficient number of Lime tokens in its account as collateral.

Security at the network level is provided by the consensus algorithm in Limex. It is built in such a way that it uses the Lime token as the native coin of sidechain Limex.