openbankIT: a banking platform for e-money management based on blockchain technology

Dr. Pavel Kravchenko, Sergiy Vasilchuk, Bohdan Skriabin
pavel@atticlab.net, vsv@atticlab.net, bohdan@distributedlab.com

http://goo.gl/2DsgEd

Version 1.0.0 (March 25, 2017)

Abstract. Traditional banking technology has multiple problems that prevent its usage in emerging environment of internet payments. It has high down installment costs, solutions and APIs are proprietary and cannot be modified by a bank, security and reliability depends on expensive hardware and software modules. All these factors lead to high transaction cost, inability to interoperate and complex maintenance procedures.
OpenbankIT solves this problems by providing an open-source platform for e-money management that includes all the necessary modules for a bank, based on modern technology and security practices. Total cost of ownership of the banking platform therefore can be reduced 10 times compared to traditional technology while maintaining higher level of security, transparency and speed of transactions.

1 Introduction

OpenbankIT is an open-source banking platform for managing e-money that uses blockchain technology. We have developed a complete stack of technologies for banking industry, which purpose is to eliminate technological barriers between financial institutions. Transparency and reliability of the platform are guaranteed by crypto technologies.

1.1 Platform Principles

We didn't change the principles of the platform from the banker's perspective. Its primary goal is to significantly (at least 10 times) reduce the total cost of ownership for the core

banking system that processes e-money and to extend its functionality. Users, banks, merchants and financial institutions can customize the interface to suit their needs.
Through the use of blockchain technology the following properties of openbankIT platform were achieved:

- all transactions are signed by a user's private key;
- reliability and transparency of history storing and ensuring its integrity;
- processing and backup of transactions with common hardware and software;
- integration and implementation of smart contracts.

Important advantages of the platform are transparency, reliability and security of payments, provided by the cryptographic technologies. The same algorithms are used by banks and other financial institutions to protect their data. All transactions are signed by their initiators and processed by multiple independent servers. At the same time, each payment is an atomic transaction: if someone's balance is decreased, at the same time someone's is increased. The flexibility enables a deployment of various financial services directly on top of the openbankIT platform.

E-money issuance and distribution are fully controlled by the bank. The distribution process can be performed through the intermediary financial institutions that receive e-money from the bank and deliver it to the end users.

One of the features of openbankIT platform is an ability to integrate with other banking systems, which enables performing an exchange of e-money issued and processed by different banks and financial institutions. As a result, users can perform interbank payments as quickly as within an issuing bank.

OpenbankIT platform also supports features like imposition of restrictions on e-money flows between users, setting limits on accounts balances, setting maximum circulation limits of e-money for certain account types.

Infrastructure of openbankIT generally doesn't require any specialized equipment such as ATMs and checkout terminals. In the simplest case, it is enough for user to install an application on his smartphone. At the same time a smartphone, computer or a tablet can be used as a point of sale checkout terminal.

Any transaction conducted by the user or by the bank's management is timestamped and irrevocable. This means that one cannot cancel or modify the transaction that was confirmed in the past. At the same time the full history of the changes of all the balances can be provided to auditors.

1.2 Benefits

1. Through the use of modern technology and high level of automation, transaction processing costs in openbankIT are ten times less than traditional banking technologies.
2. Increased security of a bank ledger (insiders or hackers cannot change the ledger without knowing users' keys).
3. Increased transparency of all the transactions (all actions, including fee changes are transactions).
4. Increased speed of transactions - the clearing process from initiation to full completion takes 5 seconds.

2 Dictionary of Terms

Dictionary of terms is provided in accordance with the directive of National Bank of Ukraine about e-money (http://bank.gov.ua/doccatalog/document?id=72246).

E-money — the units of value, processed by means of electronic devices, which are considered as a liability of the issuing institution, and are accepted as means of payment.

Account — a set of data about the registered user, his balance etc, which is necessary for authentication of his actions and for displaying the results of these actions. Stored and processed by the core of the platform.

Balance — the amount of e-money units that corresponds to a user's account at some time.

Operation — a single action from a limited set of all possible actions of the core that determines changes of a certain account.

Transaction — a group of sequenced operations that change the state of accounts that can be atomically approved or rejected by the core (hereinafter the term "transaction" will be used instead of "operation" in cases where if it identical for the context provided, for example when transaction consists from one operation).

Payment — a successfully confirmed transaction which includes operation transfers e-money from the balance of one account to the balance of another account.

Recharge Card — a prepaid card with e-money. A special type of account that can receive only one payment and be used to top up user's balance. Recharge Cards are used as a tool of e-money distribution.

Key Pair — a public and a private key as parts of the digital signature scheme. Used by the core to perform the user authentication and payments processing.

Core — a decentralized network of validator nodes that stores and processes all accounts in the private blockchain. The core validates all transactions and agrees on the state of all the accounts according to the consensus protocol.

State of the Core — a set of states of accounts at a certain point of time.

Consensus — an agreement of the set of nodes about the state of the core at a certain point of time.

Node — a computer, that validates all transactions on the platform, transfers them to other network nodes and monitors changes of the state of the core. The computer runs a specialized software.

Gateway Node — a network node, which synchronizes the state of the core with validator nodes and broadcasts new transactions to the validator nodes in secure network that were received from parties from the public Internet.

Validator Node — a node, that participates in reaching the agreement on the state of the core with other validator nodes, using a consensus protocol.

Blockchain — an ordered sequence of blocks, where each block consists of a set of transactions and represents the result of consensus between the validator nodes.

Transaction Confirmation — a verification process performed by each validator independently, which ends with executing all of operations in the transaction or rejecting of it.

3 Roles in OpenbankIT Platform

Terms are provided according to the directive of National Bank of Ukraine about e-money (http://bank.gov.ua/doccatalog/document?id=72246). If necessary, the number of roles can be changed. For example, for a payment company such roles as General Agent, Fee Agent, Issuer, Administrator, Distribution Agent and Settlement Agent role can be performed by a Master.

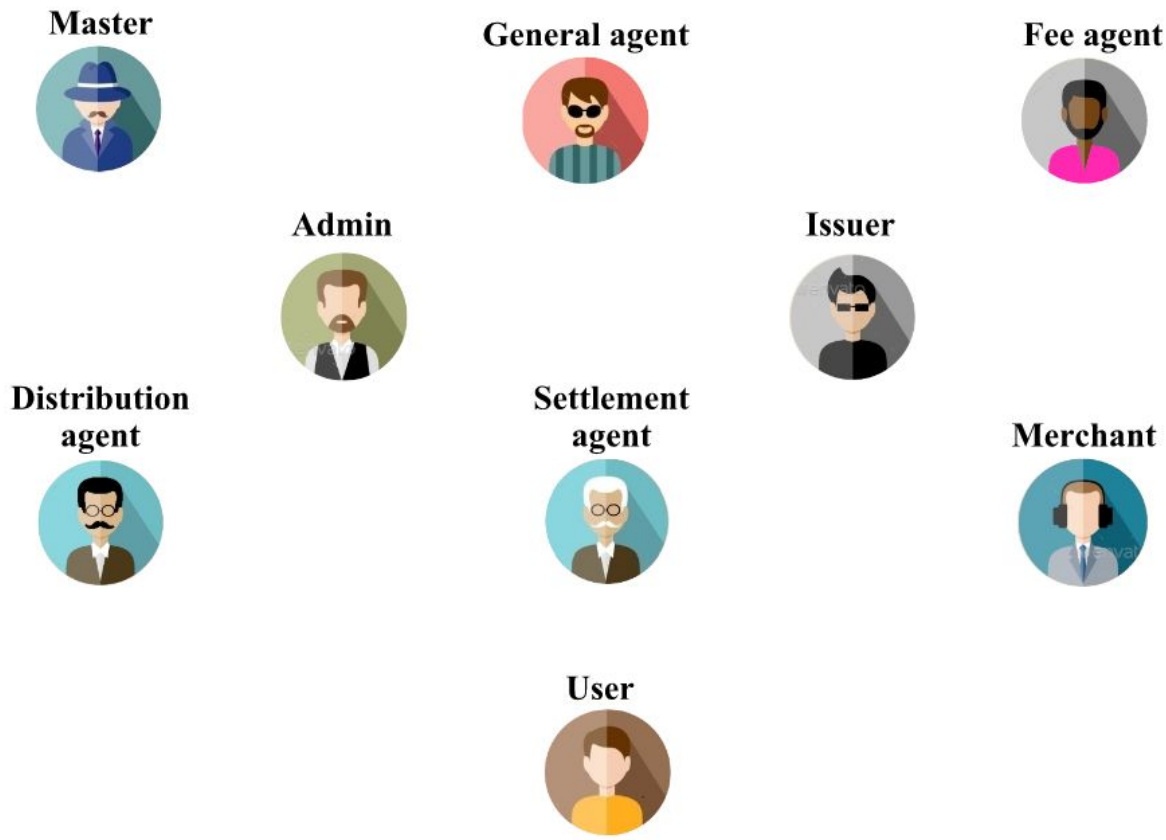

Figure 3.1 - List of openbankIT platform roles

Division of roles' permissions is necessary in order to perform effective management of the e-money accounting. Depending on his role, a participant can perform defined list of operations. For each account type special rules of keys generation and management are defined.

All types of roles in the openbankIT platform can be divided into two groups:

- **management roles** — roles that involved in e-money issuance and accounting (include Master, Issuer, Administrator, General Agent, Fee Agent roles);
- **non-management roles** — e-money users and some agent roles (include User, Merchant, Distribution Agent, Settlement Agent roles).

Master — the main responsible individual/organization. Only one Master is allowed on the openbankIT platform. Responsible for approval and revocation of the following roles: General Agent, Fee Agent, Administrators, Issuers.

Administrator — a Master's trustee. Responsible for appointment and revocation of Distribution Agents and Settlement Agents. Multiple Administrators are allowed to be registered on the platform.

Issuer — a Master's trustee. Responsible for the e-money issuance. Only one Issuer is allowed by the platform rules.

General Agent — an entity which holds the issued e-money units before distribution. Only one General Agent is allowed in the platform. Selected and approved during the openbankIT platform initialization. General Agent can be replaced by the Master.

Fee Agent — an entity which stores all the collected fees. Only one Fee Agent is allowed by the platform rules. Selected and approved during the openbankIT platform initialization. Fee Agent can be replaced by the Master.

Distribution Agent — an entity which is responsible for e-money distribution. Many Distribution Agents are allowed to operate simultaneously on the platform. Receives the e-money from the General Agent and transfers it to the end users. Can create recharge cards as an e-money distribution tool.

Settlement Agent — an entity which is responsible for e-money withdrawal from circulation. Few Settlement Agents are allowed to operate simultaneously on the platform. Can receive e-money from the end users and merchants and transfer it to the General Agent.

Merchant — an entity that accepts the e-money in exchange for goods or services. Many Merchants are allowed to operate simultaneously. A Merchant receives the e-money from the end users and is able to transfer it only to the Settlement Agent. A Merchant can also perform full or partial refund of payments received.

User — an entity who is the end user of the e-money. In case of Ukraine, the user is considered anonymous. In the case of the EU user must provide a name and a phone number to complete the registration. Creation of user account is performed automatically during the first incoming payment. User can receive payments from Distribution Agent, from another end user, from recharge cards, from Merchant as a refund transaction.

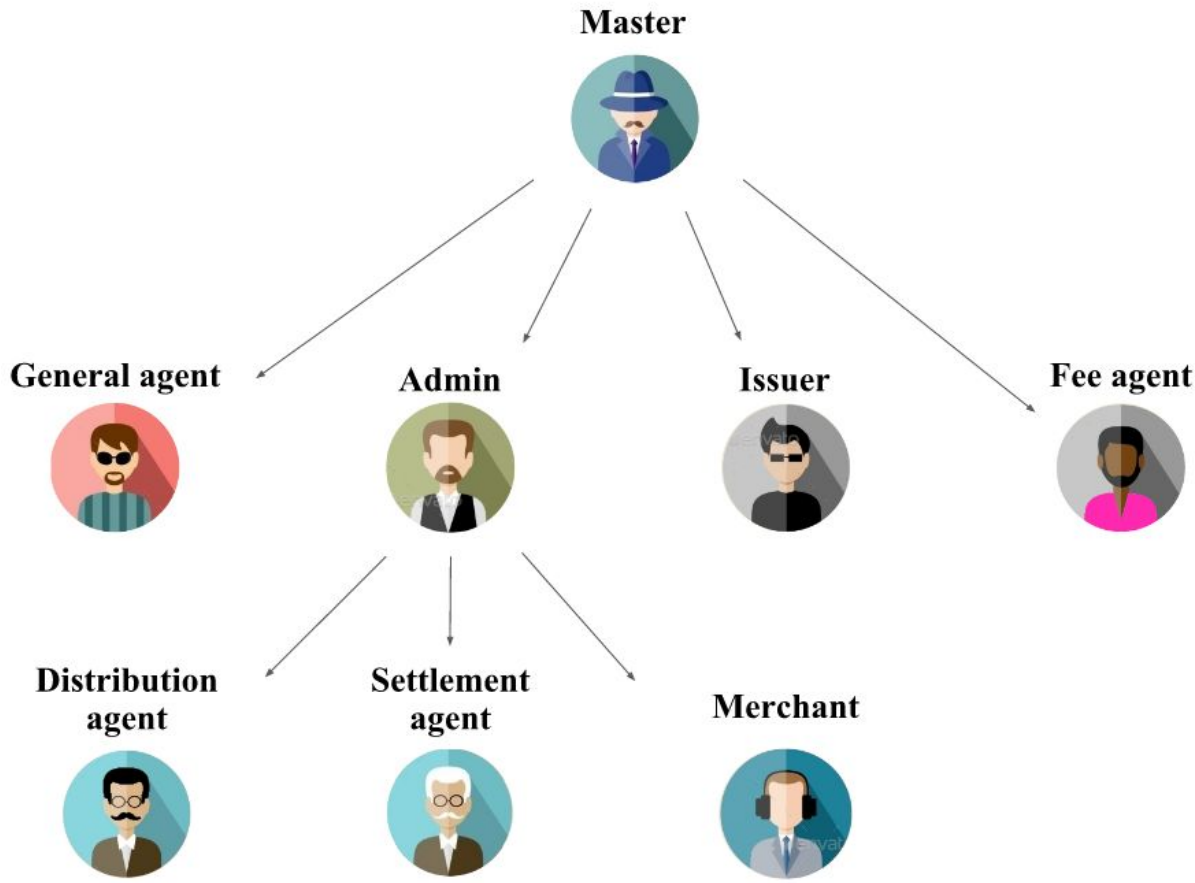

Figure 3.2 - Hierarchy of openbankIT platform roles

4 Architecture

4.1 Platform Components

The following is a description of the platform components and their functionality. OpenbankIT platform is built using a micro-service architecture where components interact via the RPC (Remote Procedure Call).

Backoffice — an interface which implements management tools for different roles like Administrators, Issuers, and Agents. Features like statistics monitoring and blocking the users by IP-addresses are accessible here.

Integration Module — provides monitoring and processing transactions in the core banking system, that are related to operations with e-money, and initiates transactions in the core banking system, which should be performed as a result of certain events in the openbankIT platform (such as e-money buying and selling).

Business Logic Module — module that implements a set of rules and restrictions related to e-money, such as the maximum balance of the wallet, the maximum daily/annual account turnover and others.

Identity Management Module — handles requests for registration of Agents, Users, Merchants. Supports limits impositions.

User App — web and mobile applications for end users which implements basic functions - such as displaying the account balance and transaction history, payments, invoice creation, editing the identification data, recharge cards scanning.

Merchant Module — a module that implements the IPN (Instant Payment Notification), web and mobile applications, test store and WordPress ecommerce plugin.

Blockchain Viewer Module — a module that provides information about transactions and statistics in an easy to view format.

Invoice Module — a module that enables managing invoices and their statistics.

Recharge Cards Module — enables features of recharge cards creation and their usage monitoring by the Distribution Agent.

Exchange Module — enables handling of e-money purchase requests by Distribution Agent with use of local payment systems.

Key Server — it is a separate service that is deployed on a remote server. Performs storage of private keys in a secure way. Can store encrypted private keys of platform members by their request. This option makes possible to access an account from different devices and to avoid manual transfer of personal keys from device to device. To use the Key Server a user has to register and to send his private key in a secure way.

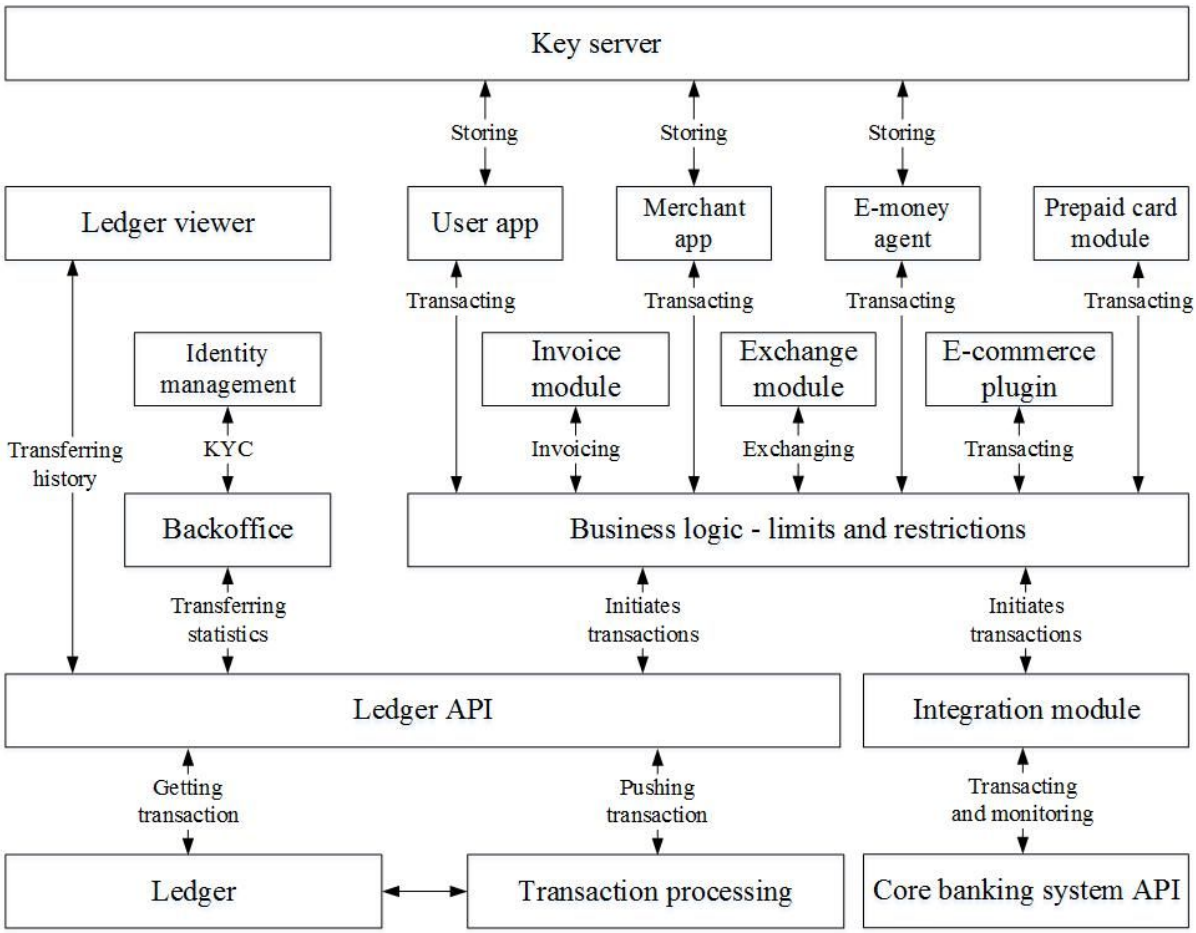

Figure 4.1 - Scheme of component relations in the openbankIT

User Application — is an application for end users which enables their interaction with openbankIT platform. Users can see the current state of their accounts in the application. In the same way users can create, sign and send transactions to the network. To start using the application a user has to complete a registration. A key pair – a public key and a private key – is generated during the registration process.

Invoice Module — is a module which enables convenient receiving and sending of payments. An invoice is created by the user's request and stored on the invoice server. Invoice contains the destination account ID and amount. During the invoice creation process the unique invoice number is associated. That number is sent to the party who wishes to make a payment. With this specific invoice number one can fetch the invoice data from the invoice server. Using invoice data a user can specify a payment destination and send a transaction to the network. The invoice will be disabled after payment confirmation.

Recharge Cards Module — is a module used for e-money distribution in physical locations. Recharge cards can be distributed both in physical and electronic forms. With recharge cards accounts can be refilled.

4.2 Core Structure

Blockchain is stored on every node in the network. By default, they are all controlled by the issuing bank. Some nodes can only store a copy of blockchain and do not take part in a transaction confirmation process (consensus protocol). Such nodes can act as gateway nodes or backup servers. Nodes that act as validators can be located in a safe network segment of the bank and be available through the gateway nodes.

The main database of the core implements a blockchain structure, where each block is a set of transactions. Every new block defines a new state of the core according to the previous block's state. Integrity of core`s database provided by the blockchain and consensus over it. Each block is cryptographically linked to the previous block. This feature ensures ability to validate the database and the history of transactions at any time in the future. The main database stores all the data passing through the core.

In addition to the main database, there is another database that stores only final state of the core. It is optimized for fast reading and writing of account data during validation of new transactions. Thus, each node in the network stores and processes the state of the core using two databases simultaneously. One is for fast searching and reading of transaction data, another supports the general history and synchronization with other nodes in the network.

4.3 Components Interaction

OpenbankIT architecture consists of different components, each solves a separate set of problems. This makes it easy to modify independently each of them by expanding or adding new features, as well as design and launch new platform components. In terms of geographical location, components can be located in a permanent place or non-permanent place. They also differ from each other by operational environment: network connection, physical control, hardware and software components. Interaction between platform components is performed through the message or request exchange, which is distributed over network channels. Methods for establishing these various channels depend on the type of components and safety requirements. In addition the platform has certain rules for authentication between components.

5 Entities Specification

5.1 Keys

Public keys can be published to anyone. Private keys have to be kept in secret. Users can choose the way of storing their private keys. They can store them locally on their own computers or on the remote key storage. Also users can manually store keys offline. In case

of keys loss the platform does not accept any other proof of account ownership. In this case account's balances become inaccessible and are actually lost.

5.2 Accounts

In the openbankIT platform each user has their own account. Each account necessarily contains at least one public key — main, and by default any transaction is signed by the main private key of account.

Management of roles' privileges is implemented via various accounts' permissions. The privilege of creating an account with specific type is defined by roles hierarchy (see. "Hierarchy of openbankIT platform roles"). In addition to the main key pair an account can contain auxiliary keys - so called signer's key. Signer's keys can be added by separate transactions, signed by the main private key. The use of signer's key enables such platform features as separation of roles' privileges, reserve keys creation and collective signing of transactions (multi-signature).

5.2.1 Account Data Structure

Account has the following set of values:

- main public key, also it is a unique account ID;
- account type;
- account balance in e-money units;
- the number of signed and confirmed operations;
- the number of signers;
- signers' public keys.
- signers' permissions.

5.2.2 Account ID

Main public key is an unique ID of the account. The account ID is used in operations performing e.g. to specify the sender and the receiver of the payment operation. To see all information about the current balance, history of transactions or full account status, a request with account ID has to be performed. Account ID can be encoded in various ways, which greatly improves its perception by a human or by means of electronic devices — QR or Base32 encoding.

Also a set of signers can be added to the account by linking an additional public key. Such signers can participate in common transactions signing by their private keys as well as by main private key. Any way the set of operations which may be signed by the signers keys can be limited. Operation of linking a new public key as a new account have to be signed only by the main key. Revocation of the signer can also be signed only by the main private key.

5.2.3 Account-Initiated Operations

There is a set of specific operations that can be performed different roles. Every operation has to be signed by its initiating account to be confirmed by the core.

Permitted Operations

Each account on the platform has a list of permitted operations that are defined by the core in accordance with its role. To perform any operation, the initiator has to create a transaction, sign it and distribute to the network. To be confirmed the transaction comes to the validator nodes. Each validator node checks each operation in transaction for having necessary permissions in order to be executed according to the type of initiator's account.

Payment Operation

User's account balance can be changed only by a payment operation. A transaction containing this operation reduces the sender's account balance and increases the recipient's account balance. This change is atomic, and cannot be canceled after confirmation.

5.2.4 Account Lifecycle

Account Creation

In the openbankIT platform the account is created when corresponding data structure is added to the ledger. The method of creation depends on the type of account. Master's, General Agent's and Fee Agent's accounts are created manually during the process of core initialization. All other types of accounts are created with confirmation of special transaction.

Methods of Account Initialization

- manual mode: Master's, Fee Agent's and General Agent's accounts;
- privileged mode: Distribution Agents', Settlement Agents', Merchants' accounts;
- automatic mode: Users' accounts.

Account Blocking

The process of account blocking means creation and confirmation of a transaction that contains the blocking operation for a specific account specified with its ID. Blocking operation can only be initiated by administrator of the platform. After confirmation of blocking transaction the core refuse to confirm all transactions that aim to change the state of a blocked account.

Period of Account Activity

Period of account's activity is period of time when a specific account is involved in initiating and/or authenticating transactions which change its current state.

5.2.5 Accounts' Privileges

Account's role defines the set of operations it can perform and these operations are specified in the code and therefore cannot be changed even by Master or Administrator. Using various types of accounts a roles' privileges management is implemented.

5.2.6 Account History

Each account has a unique version of its history in the core. Account history is a set of transactions that have changed its state from the moment of creation to the final state. The set of these transactions is clearly ordered and consists exclusively of the set of the transactions that have been confirmed by the core.

5.2.7 Accounts Created by Administrator

Settlement Agent's account can only be created by a platform Administrator. Settlement Agent provides his new public key to the Administrator, the latter creates a transaction with operation which initiates the creation of account of Settlement Agent's role and specifies the newly received public key. The account of Settlement Agent will be created after transaction confirmation. Accounts of Distribution Agent and Merchant are created in the same way.

5.3 Recharge Card

5.3.1 Recharge Card Lifecycle

Generation

Distribution Agent at any time generates a new key pair. For that key pair a recharge card account will then be created. A corresponding private key is stored in a digital form or printed on a physical card.

Creation

Recharge card account is created automatically when payment to its ID is confirmed. Distribution Agent creates a transaction with payment operation to recharge card account ID. After transaction confirmation the card will have a balance according to the amount of e-money sent.

Distribution

Recharge cards are sold by a Distribution Agent in digital or physical form.

Usage

Recharge cards are used as a tool of e-money distribution. Users can refill their wallets with use of recharge cards by creating a payment transaction which has to be signed by card's private key.

Termination

The lifecycle of recharge card is terminated when its balance is empty.

5.4 Transactions

5.4.1 Transaction Structure

One transaction can contain no more than one hundred of operations. After transaction is created it has to be signed. Signed transactions are pushed to a decentralized network for validation. The signing process is performed by linking of set of necessary digital signatures to transaction body. That set is defined by a set of operations, which is also a part of the transaction. Only in case when all of required signatures in the transaction are present it can be potentially confirmed by the core. The signatures correctness indicates that all initiators of operations actually have rights to perform them.

5.4.2 Transaction Lifecycle

Creation

The user creates an empty transaction. The user creates a set of operations and adds them to the transaction body. The user leads the transaction to a certain protocol format by filling all other transaction fields.

Signing

In order to create a transaction the user has to add a set of necessary signatures. The set of the signatures depends on a set of operations initiated. A transaction can be potentially confirmed only in case of all required signatures are present.

Pushing

Signed transaction can be sent for processing. In general, there is no possibility to cancel a transaction that was pushed to the network. If transaction is signed by non-management account it should be distributed through the public gateway node. A transaction signed by management account it can be pushed to the network directly.

Distribution

On the next step transaction reaches one of the decentralized network's node. The node which receives it first verifies its specific set of operations, its set of necessary signatures and ability of operations to be performed. In case of successful transaction verification, the node distributes it to all nodes it is connected to. Otherwise the transaction is rejected. All other nodes that received the new transaction perform the same actions.

Adding to Candidate

While consensus protocol is working, each validator node of the network combines all new transactions, that need to be confirmed. This set of transactions is called a candidate. The candidate contains transactions that do not contradict with each other. All validators perform several rounds of candidates exchange to expand consensus on a set of transactions being confirmed.

Confirmation

After forming a joint candidate validators begin voting on it that ultimately will determine whether the candidate will be confirmed. During the voting process each validator signs a joint candidate with its own private key and distributes the signature among all network nodes. A candidate is considered confirmed if it collects the majority of votes. After that the transaction is included to a block that defines a new state of the core and gets into the entire history of all transactions - the ledger.

6 Processes

6.1 Transaction Validation

A decentralized network of specialized computers validates and confirms transactions. In a general case these computers are under control of some organization - i.e bank. Each computer validates all transactions independently from other computers using special software. Exchanging of messages between network nodes is performed by means of closed or protected networks that are isolated from the Internet. Each computer in decentralized network keeps a core state and a history all transactions carried out. Any transaction which was confirmed by the core is irreversible, because it is stored in chronological order in the blockchain.

6.1.1 Transaction Validation Steps

1. Signed transaction is pushed to the gateway node with special request. Transaction has to be serialized with a common XDR format (External Data Representation).
2. Gateway node which receives the request checks it. Also the node validates the new transaction and distributes it to validator nodes.

3. Validator node also validates received transaction and distributes it to other nodes if they do not know about it yet.
4. If the majority of validator nodes agrees on confirmation of particular transaction, it is confirmed. Confirmation process is performed in accordance with consensus protocol rules.
5. Each confirmed transaction changes the current state of the core.

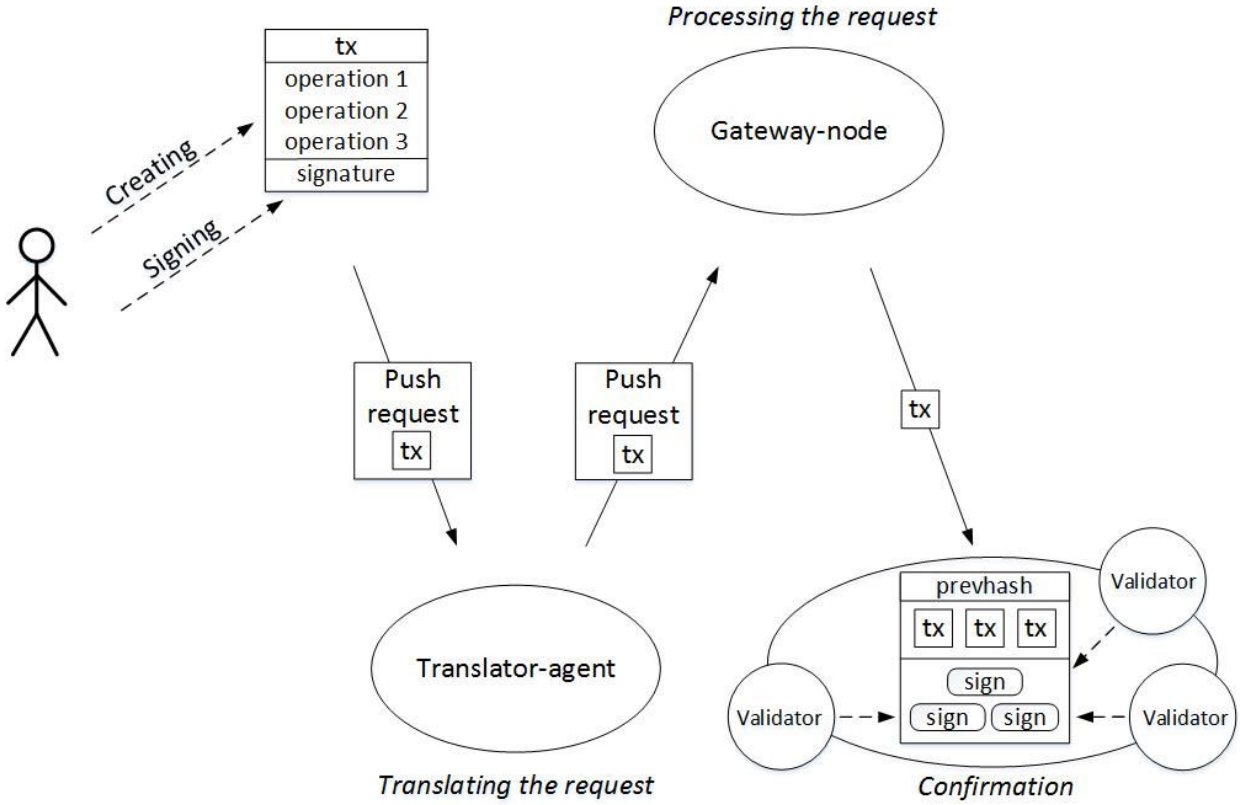

Figure 6.1 - Transaction processing in openbankIT platform

Any confirmed transaction cannot be changed or deleted, even if its initiator can prove that transaction was unwanted or mistaken. A payment sent to the false account will be confirmed and it will be impossible to undo this payment. However, in some cases it is possible to initiate another transaction in order to return the e-money accidentally sent to a wrong account.

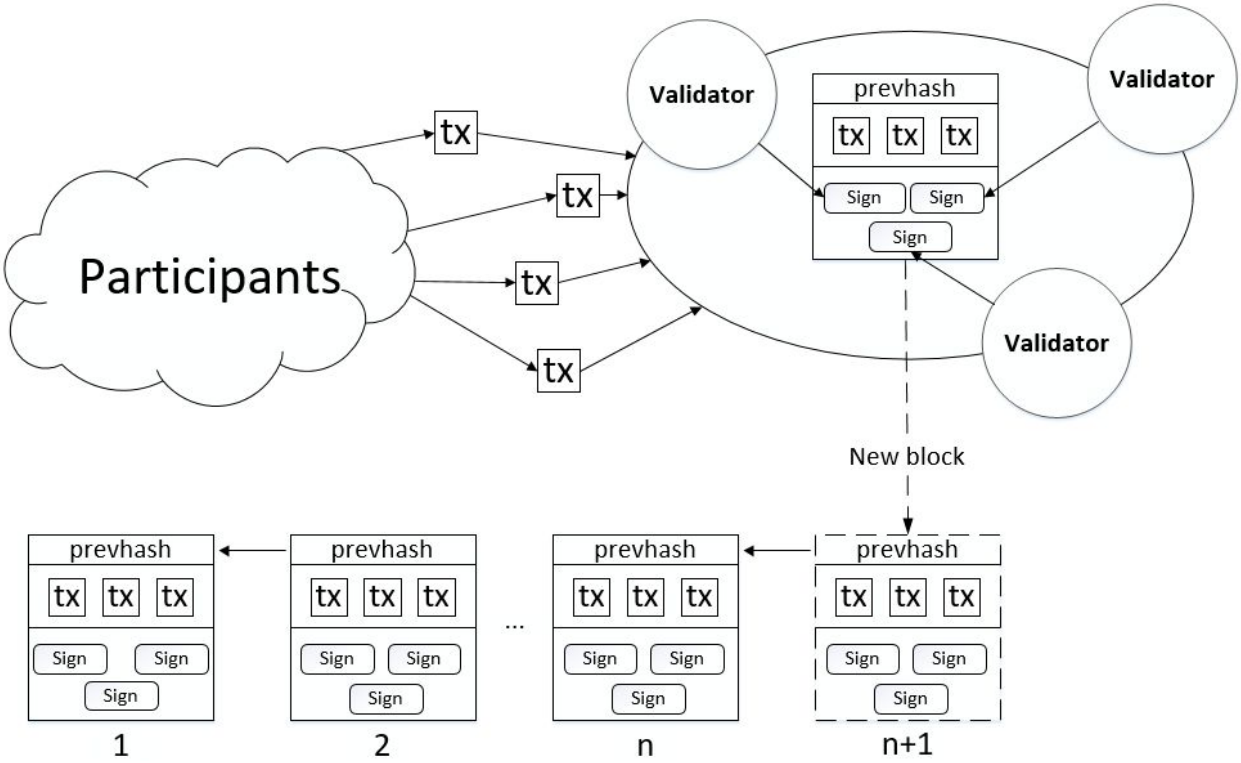

Figure 6.2 - Scheme of transaction processing and updating the main database (blockchain)

6.2 Fee collection

Fees can be collected from each transaction. OpenbankIT allows the issuing bank to configure transaction types and corresponding fee amounts. The fee can be fixed or it can be a percentage of payment amount. For example, the bank may set a 1% fee for transaction that transfers the e-money from user to merchant, but 0% fee for transfers between users. Every time the fee is charged, it increases the balance of the Fee Agent.

Fees can be imposed on:

- any transaction in specific currency;
- e-money flow between specific types of roles in specific currency;
- specific account for e-money transfers to specific types of roles in a specific currency.

6.3 E-money Lifecycle

6.3.1 E-money Issuance

Entering of e-money into market circulation is performed by emission process. Emission on openbankIT platform is centralized and fully controlled by the issuing bank/financial institution. Emission operation is defined as a special operation type, which makes a payment from Master's account to General Agent's account. Transaction which initiates the e-money emission operation can be signed only with an active emission key. After confirmation of emission transaction, the Master's account balance decreases, and the General Agent's

balance increases with corresponding amount of new e-money. Master's account balance is allowed to be negative, because it is responsible for the emission (which is debt). Distribution of issued e-money is performed by the General Agent. Settlement Agents perform e-money withdrawal from circulation using payments to the General Agent account. For a simplified case, the General Agent can be the Master itself.

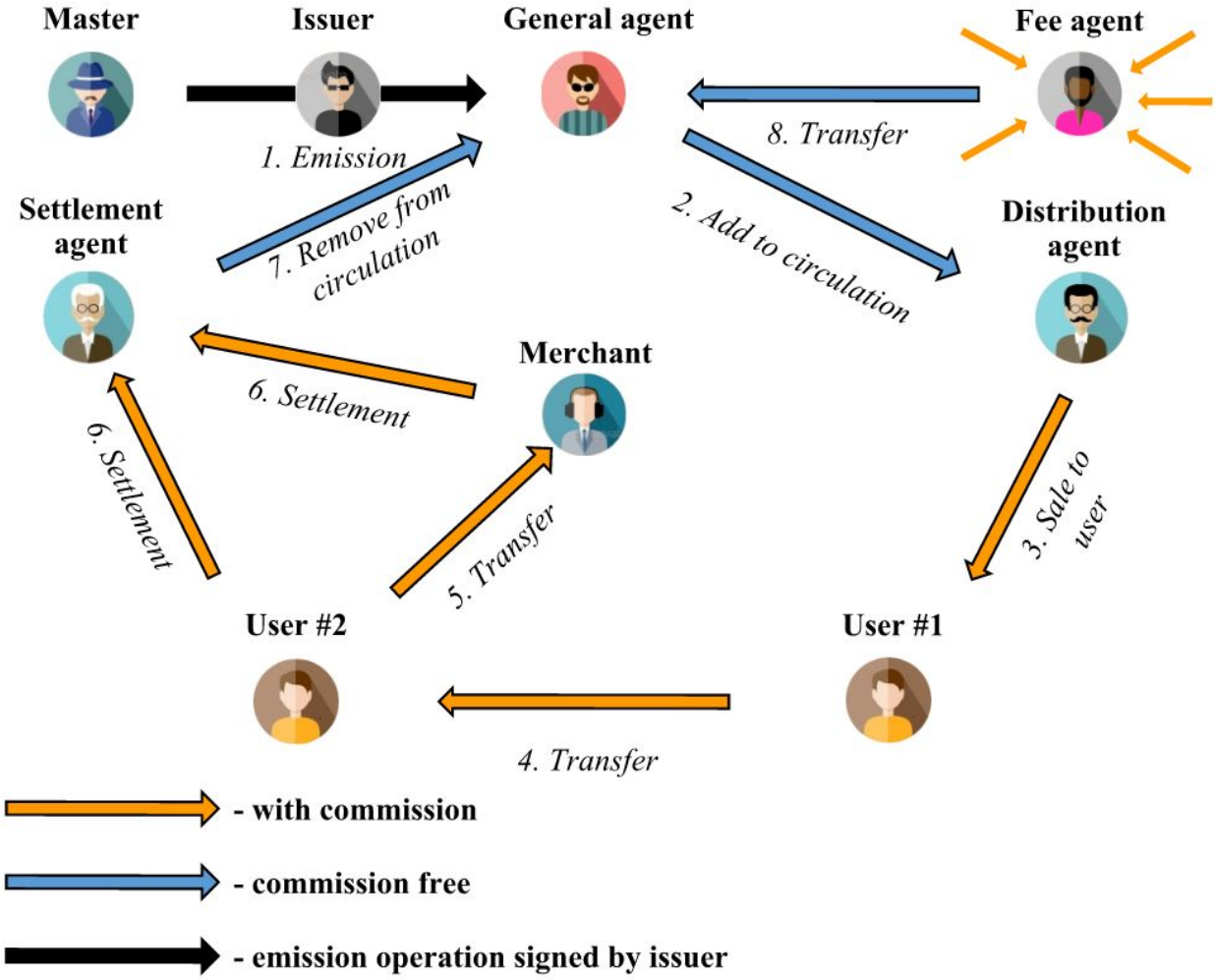

Figure 6.3 - Scheme of e-money lifecycle on the platform

6.3.2 E-money Transfers Between Users

Any user that has an account on the platform can publish his public key (account ID) to anyone for receiving payments. Similarly, a user who has a positive account balance can initiate a payment to another user. To make a payment, the sender generates a transaction with payment operation from own account to the account of recipient, which is determined by a unique identifier. Created transaction can be signed with sender's private key. Signed transaction can be distributed to the core. When transaction is confirmed by the core, the recipient can see a new state of his account and increased balance. After checking these changes recipient can be sure that payment is confirmed.

6.3.3 E-money Settlement and Withdrawal From Circulation

The process of e-money settlement is performed by payment operation to the Settlement Agent's account. This payment is permitted for end users and Merchant accounts only. When a payment is received, the Settlement Agent it pays the debt off to the sender.
The process of e-money withdrawal from circulation is performed by transferring of e-money form Settlement Agent's account to General Agent's account.

7 Security

OpenbankIT platform operates as a centralized e-money management system on top of decentralized network. In order to provide high level of assurance of payments processing authenticity, users manage balances and accounts themselves. User performs operations that are signed with their private key. Similar approach is used in Bitcoin protocol but in our case emission is controlled by the issuer. All validated transactions are irreversible, therefore one cannot change or remove any operation in the past. History of changes in the system can be provided for audit.

7.1 Threat Model

Threat analysis is an integral part of banking software development process. List of assumptions used during openbankIT security system development.

7.1.1 Assumptions

1. A user trusts the e-money collateral to the issuing bank.
2. A user trusts transactions processing to the bank only in transparent manner as they have a proof of execution of all actions.
3. A user trusts that bank provides him with properly working software and a public key certificate which is used in operations like balance check, transaction history monitoring, etc.
4. A user does not trust the bank in operations balance management.
5. Banks and users are interested in irrevocability of transaction processing as well as of account managing.

7.1.2 Threats Analyzed

1. Unauthorized e-money emission.
2. Obtaining of Administrator's privileges by malicious users.
3. Destruction or modification of account's actions log.
4. Denial of service as a result of external attack.
5. Destruction of all ledger copies.
6. Physical or logical substitution of ledger database (full or partial).
7. Theft of private keys of the users.

8. Transactions declining after their confirmation.
9. E-money double spending.
10. Change account's state without corresponding privileges.
11. Users and validator nodes MITM/impersonation attacks.

7.1.3 Threats Currently Out of Scope

1. Compromising of digital signature algorithms.
2. Compromising of key exchange and communication protocols (DHE/TLS).
3. Hardware and key storage compromising.

All these questions are covered in detail in "OpenbankIT Security Whitepaper".

8 Conclusion

OpenbankIT solves all the needs of banking industry for e-money management – accounting, internal and interbank e-money processing, management of administrators, fees, users, merchants etc. by providing an open-source platform built on top of modern technologies and security practices altogether with necessary modules pack. Total cost of ownership of banking platform can be reduced up to 10 times compared to traditional technology while maintaining higher level of security, transparency and speed of transactions.