Table of Contents

1. Introduction

1. What is Lisk

Lisk is a next generation platform that allows for the development and distribution of JavaScript based decentralized applications using an easy to use, fully featured ecosystem. Through Lisk, developers can build, publish, distribute, and monetize their applications within a custom built cryptocurrency powered system that utilizes custom blockchains, smart contracts, cloud storage, and computing nodes; all from within one industry solution.

2. Technical Background

Lisk is written in Node.js[[1]](http://nodejs.org) on the backend, and HTML5 and CSS3 on the frontend. It works asynchronously and allows for fast processing of all functions such as network transactions. The database uses SQLite to allow the use and running of complex queries.

3. Key Innovation Factors

Lisk is the first decentralized application solution written entirely in Node.js. This opens up the Lisk ecosystem to thousands of current developers with no additional skills necessary. Any web developer who is already familiar with JavaScript and Node.js can immediately jump in and begin building decentralized applications from day one.

Our core goal with Lisk was to create an entire plug and play system that would allow developers to do everything from design, development, publication, and monetization, all from within one platform. By utilizing the Lisk ecosystem, developers can quickly deploy their JavaScript apps to Lisk Hosting & Storage Nodes, gain listing in the Lisk Dapp Store, and have immediate access to Lisk Compute Nodes for execution of the code. All while being backed by the integrity and security of the Lisk sidechain consensus

functionality.

To top it all off, all of these cloud functions are run by the users and Lisk delegates who are paid through a built in invoice system (or by the network itself in the case of delegates) and paid in LISK (Lisk's own cryptocurrency) or BTC. It truly is a one stop shop for application development that provides a cutting edge, affordable, and forward-thinking solution.

4. Lisk Components

- Decentralized P2P hosting of dapps
- Decentralized P2P storage for dapps
- Decentralized computing
- Sidechain consensus for every dapp
- Lisk and Bitcoin API interfaces
- Developer tools: Lisk CLI / Lisk Dapp SDK

2. Clients

1. Lisk

The full client is the best solution for super users, delegates and developers. It is available for Windows, Mac OS and Linux. Though it is only possible to be a delegate with the Linux client. Lite client users can connect to the full clients to access the network.

They can also use them to make API calls, if it is allowed by the full client owner. All full client users download the blockchain from each other through a peer-to-peer connection.

Dapp users can use the lite client for accessing their installed dapps as well. The dapps API and the peers API are available to developers. They make it possible to quickly and easily create JavaScript dapps using nw.js2 or Electron3.

2. Lisk Lite

The regular user will mostly use the lite client, a light-weight Lisk client, to access their Lisk account.

The lite client is available for Windows and Mac OS. It does not require an installation process, as it utilizes modern web technologies. It does not act as a network node, as it only connects to other peers which are online via an http connection. This brings several advantages.

The user does not have to download the blockchain anymore, which means the application itself stays very small. It does not broadcast secret keys through the network, all data is signed locally on your device. It is possible to make all types of transactions available.

If you want to run a delegate node, you can register a delegate account with the lite client. However, it is not possible to run a delegate from it, i.e. creating new blocks. For this you need the full client.

Dapp users can use the lite client for accessing their installed dapps as well. The dapps API and the peers API are available to developers.

They make it possible to create quickly and easily Javascript dapps with nw.js[[2]](https://github.com/nwjs/nw.js) or Electron[[3]](https://github.com/atom/electron).

The full client is the best solution for super users, delegates and developers. It is available for Windows, Mac OS and Linux. Though it is only possible to be a delegate with the Linux client. Lite client users can connect to the full clients to access the network.

They can also use them to make API calls, if it is allowed by the full client owner. All full client users download the blockchain from each other through a peer-to-peer connection.

3. Lisk Mobile

The mobile client allows the user to access their Lisk account while on the go. It will be available for both iOS and Android and featured in the Apple and Google Play app stores.

The backend infrastructure for the mobile client will mirror that of our desktop solution. The real change comes in the form of additions and tweaks to the user interface which will allow for a tailored experience on mobile devices. The app has been custom designed to provide a familiar and easy to use mobile interface, much like the Bitcoin or Banking apps you already use on a daily basis. It will also allow you to launch all of your favorite dapps from within the app itself. In the future, we plan to integrate device specific functionalities like the ability to utilize the fingerprint or retinal scan capabilities for added security on your account.

3. Consensus

Lisk is based on the DPoS[[4]](http://wiki.bitshares.org/index.php/BitShares) (Delegated Proof of Stake) consensus mechanism. This method of consensus was originally created by the BitShares team.

DPoS is based on delegates creating blocks. Delegates are trusted accounts which are elected to be “Active Delegates”. The 101 delegate accounts with the most votes create the blocks. Other delegates are listed as “Standby Delegates”, and can advance to the top 101 list by receiving votes from the other Lisk owners. All users of Lisk have 101 votes available to elect their favorite delegates into the top 101 list. The weight of each of the 101 votes is proportional to the amount of LISK the user has in the wallet the votes are cast from. This total amount is shown on the delegate list as an “Approval”, and is listed as a percentage of the 100 million LISK available that is voted for that delegate.

Delegate promotion to the top 101 or demotion to the standby list happens at the completion of the 101 block generation cycle. Each cycle of 101 blocks is created by the top 101 delegates in random order. The block time is 10 seconds. Newly created blocks are broadcast to the network and added to the blockchain. After 6 to 10 confirmations, a block, along with its transactions, can be considered as confirmed. A complete 101 block generation cycle takes approximately 16 minutes.

In DPoS, forks can occur, but the longest fork wins. Delegates must be online all of the time and have sufficient uptime. Uptime is used to catalogue the reliability of a node by logging each time that it misses a block that was assigned to it. Users vote for the top 101 delegates based on several factors, uptime being one key factor used to make a determination. If a delegate drops below a certain rating, users may remove votes from the delegate in question due to poor performance.

1. Delegates

The function of delegates is covered above in the Consensus section.

To be a delegate, a user needs to register a delegate account. This is accomplished from the client user interface in either the full or lite wallet. Keep in mind that block generation is only possible in the full wallet. This means that you can register a delegate in either version of the wallet but will only be able to perform the delegate functions from a full version of the client. The account number and username will be the same after the delegate registration. All Lisk accounts are eligible to become delegates.

New delegates start as standby delegates. Standby delegates begin with an approval

rating of 0% and will need to accrue votes from the Lisk community in order to advance to be one of the top 101 delegates. Block generation is performed by the top 101 delegates only. If you are in standby status, you will not forge any blocks.

2. Network fees

All valid transactions in the network must be processed. Delegates process transactions and store them in new blocks. For this work, the delegates receive a fee. All transactions in the network must contain some type of fee as a spam countermeasure.

The default network fee for sending an LISK transaction is 0.1 LISK. For example, a 100 LISK transaction includes an additional fee of 0.1 LISK for a total transaction cost of 100.1 LISK.

The following is a list of fees for different types of transactions:

- 0.1 LISK of amount sent for a spend transaction
- 5 LISK for registering a second passphrase
- 100 LISK for registering a username
- 100 LISK for registering as a delegate
- 1 LISK to add a contact
- 500 LISK to register a dapp
- 5 LISK per member for registering a multi-signature group.

Delegates receive the fees from all transactions of the last block cycle (101 blocks). Fees are split equally between all delegates who created a block in that cycle. Delegates who missed creating a block assigned to them during that cycle are not paid.

3. Peer-to-Peer

We are using a standard P2P network[[5]](https://en.wikipedia.org/wiki/Peer-to-peer), which works on top of the http protocol, and uses json formatted data as a method of data inter-change. The P2P module captures the following information about each peer:

- Version
- OS
- IP
- Port

4. Core Features

1. Usernames

Lisk allows users to register usernames. Which act as an alias to your account. Other users can send transactions to this username and the linked account will then receive it. This eliminates the need to remember long account addresses.

The network fee for username registration is 100 LISK. Usernames may contain the following characters:

- Traditional Alphabet (Upper & Lower Case): A-Z, a-z
- Numbers: 0-9
- Special Characters: !, @, $, &, and .

Each username is unique. The length is currently limited to 16 characters. Currently, it is not possible to remove a username from your account.

2. Contacts

Lisk allows users to maintain a contact or friends list. This feature can be used to store frequently used accounts, but can also be used as a reputation system. If an account has many confirmed contacts, it may be considered more reputable than one without.

Contacts work like followers on Twitter. A user is added to the contact list, which will then show as a pending contact request in the user's wallet. Regardless of whether or not the other user accepts the request, they will be shown in the contact list. Once the other user accepts the request, the requester will be added to his contact list as well. Both parties now have a new confirmed contact.

The network fee for adding a new contact or accepting an incoming request is 1 LISK.

3. Multi-signatures

Lisk allows users to create a multi-signature group. A multi-signature group consists of several Lisk users, called group members. Transactions from multi-signature groups can be configured to require some or all signatories for approval.

To achieve this a M of N multi-signature architecture is implemented. All members of a multi-signature group (N) are added, up to a maximum of 16 signatories, and then the required number (M) of signatures needed to approve a transaction is specified.

M must be greater than 1 and less than or equal than N. N is the number of members of the multi-signature group.

Once you initiate a transaction from the multi-signature group, all members will see this pending transaction and decide whether to approve or ignore it. Once the required number of confirmations has been collected, the group will allow the transaction and submit it to the blockchain.

The owners of a multi-signature group may change the rules of the group at any time with the approval of at least M of the signatories.

5. Decentralized Applications

1. Virtual Machine

Lisk Dapps are executed using Lisk Node, a specialized version of NodeJS that provides a sandboxed runtime environment in which to run individual dapps. Inter-process communication is achieved using Named Pipes, with no imposed limit on message size.

Upon launching a new Dapp, the Lisk client starts a new instance of Lisk Node as a child process. If a Dapp encounters a fatal error, then the child process is killed gracefully, leaving the parent Lisk client unaffected.

Please note, currently there is no protection against unauthorized system calls made from the running dapp. Therefore, running untrusted code is not yet advisable, and could potentially lead to loss of funds. Work is underway to provide a fully sandboxed environment in which to run untrusted code.

2. Dapps

A dapp is a decentralized application[[8]](https://github.com/DavidJohnstonCEO/DecentralizedApplications/blob/master/README.md) written in Node.js and JavaScript. It works with the Lisk VM using either the Lisk or soon the Bitcoin consensus algorithm. The Lisk VM is a scalable Node.js application that allows Node.js and JavaScript developers to write dapps. With current web technologies (HTML5/CSS3/JavaScript) the developer is able to create a powerful UI. Dapps can use custom Node.js packages from NPM (the Node.js package manager).

Regular users can launch the dapps on a Linux Lisk client or via the Lisk Lite client on Windows or Mac OS.

3. Dapps Development

Developers write dapps in JavaScript which allows the use of the full ecosystem of

Node.js packages powered by NPM. The Lisk VM is integrated with the Lisk API. This API interfaces with the Lisk Blockchain and even with the Bitcoin blockchain. Each dapp runs in the Lisk VM, which removes many possible attack vectors and thus makes it much safer for the end user to start dapps on their local machine. The Lisk API is accessible by the dapp.

To make the dapp development as easy as possible the Lisk Team released lisk-cli, a command line interface which creates your own testnet and dapp environment by answering a few simple questions. Additionally we prepared a Dapp Toolkit, which gives developers a reference implementation of the most important dapp functionalities, and serves as a solid foundation upon which they can start building their decentralized applications.

Many libraries have been written to provide the full Lisk API functionality for developers “straight out of the box”.

This API includes:

* Consensus API
* Lisk API
* Bitcoin API
* Database API

To open a dapp, the format: `http://ip:port/dapps/<dapp_id/username>` is used.

4. Dapps Computation

The Lisk Team is developing a system that allows for billing of CPU time. Where the Lisk VM uses its API to track how much CPU time is used to run a dapp. As a result, owners of nodes can run dapp master nodes in return for LISK or BTC payments.

The purpose of Lisk is to create a unique ecosystem, of which computation is one part. In the future, Lisk will have a submission manager to submit dapps to candidate nodes offering their service to run dapps and select the nodes meeting the specified resource requirements offering the best combination of price and performance. The node owners will earn revenue from providing computation, memory, storage and other resources.

This is referred to as Dapps Billing. You can compare it to the Heroku platform for deploying applications.

5. Dapps Consensus

Each Dapp has its own unique private side chain which operates in synchronization with the Lisk block time and current block height.

Dapp sidechains are managed by a group of up to 101 master nodes, each of which have block generation enabled specifically for an individual dapp. The primary role of each master node is to process transactions and signify the validity of each block generated on the sidechain.

The signing of blocks by a master node against a given dapp is restricted by the dapp owners. Whom then approve individual Lisk accounts as master nodes, which then are allowed to forge on the Dapp’s side chain.

Sidechain consensus is maintained among the 101 master nodes using the same Delegated Proof-of-Stake (DPOS) method used to secure the Lisk blockchain. This allows individual master nodes to collect fees from each transaction as reward for securing the dapp's side chain.

The motivations behind this form of consensus are to prevent unnecessary enlargement of the Lisk blockchain and to retain individual sidechain autonomy, while at the same time, ensuring the integrity of each side chain is constantly upheld.

It should be noted as an optional alternative in the soon future, Lisk dapps can

instead be secured by the Bitcoin blockchain using this same method.

6. Dapps Master Nodes

Dapp master nodes are Lisk nodes with an installed dapp and with block generation enabled specifically for that dapp. Dapp owners need to approve individual Lisk accounts to be permitted to be a master node. The nodes process transactions and generate new blocks which are then secured by the Lisk Blockchain, making them the core of the dapp system.

7. Dapps Storage

It is possible to host dapps on every kind of storage network, centralized and decentralized. However, the used storage network has to provide you with a download link to a ZIP package. This is necessary because you have to provide the download link at the dapp registration process so that once a consumer/user wants to install your dapp Lisk knows from where to download and unpack it.

At a later stage IPFS[[10]](https://ipfs.io) will be natively integrated into Lisk as a decentralized storage option.

8. Dapps Deposits / Withdrawals

Developers can use either LISK or BTC in their dapps11. Users of a dapp may deposit or withdraw funds from any given dapp. When LISK or BTC are sent to a dapp address, the funds appear in the dapp account. The funds will then become available for use within the dapp. This works the same way for BTC deposits as it does with LISK. BTC is sent to a special dapp address and then appears in the dapp Bitcoin wallet.

Dapp accounts are a special type of account created by the owner of a dapp. All deposited LISK or BTC will be stored in the associated addresses. For security reasons, only the use of multi-signature dapp accounts with trusted signers is recommended.

Withdrawals from dapps are processed by master nodes. When a withdrawal request is sent, the dapp master node processes it and moves the funds to the specified withdrawal address in the Lisk or Bitcoin blockchain.

9. Dapps Tokens

Developers may implement custom tokens in their dapps, and use these tokens as the main currencies within their dapps. These tokens may be used in the same way as LISK or BTC, but the tokens cannot be moved directly from one dapp sidechain to another dapp sidechain. They must only move via the Lisk main chain.

6. Appendix

Written by

* Max Kordek
* Oliver Beddows

Releases

* February 1st, 2016 (v1.0)

7. Sources

* [1] [node.js Organization.](https://nodejs.org)
* [2] [nw.js](https://github.com/nwjs/nw.js)
* [3] [Electron](https://github.com/atom/electron)
* [4] [Bitshares DPoS.](http://wiki.bitshares.org/index.php/BitShares)
* [5] [Peer-to-Peer Wikipedia Article](https://en.wikipedia.org/wiki/Peer-to-peer)
* [6] [Seccomp Wikipedia Article](https://en.wikipedia.org/wiki/Seccomp)

```
* [7] [npm.js](http://npmjs.org)
* [8] [David Johnston. Decentralized Applications.](https://github.com/DavidJohnstonCEO
/DecentralizedApplications/blob/master/README.md)
* [9] [Factom. Merkle tree.](https://github.com/FactomProject/FactomDocs/blob/master
/Factom_Whitepaper.pdf)
* [10] [IPFS. A decentralized storage solution.](https://ipfs.io)
* [11] [Sidechains. Deposit/withdrawal sidechain.](https://www.blockstream.com
/sidechains.pdf)

[**Based on the Crypti Whitepaper v2.1 by The Crypti Foundation**](https://crypti.me
/crypti.pdf)
```