The technical side of Sikoba: a short introduction

Eyal Ron

March 24, 2017

There are two things the Sikoba blockchain really needs to be. It needs to be *fast*, being able to serve many users with hardly any delay; It needs to have a *flexible* design, so that upgrading the economic logic won't be a huge deal.

These demands did not appear out of thin air. We don't want the Sikoba blockchain to become Bitcoin, with a large "how do we scale it?!" conference a few years down the road. We also don't want the system to stagnate, and yes SegWit vs. block-size debate, we're looking at you.

To face those challenges, we're designing the Sikoba blockchain as a *federated blockchain*. This is a permissioned mining model where only authorized nodes are allowed to add blocks to the blockchain. Each of these nodes is called a *federation member*. The collection of federation members is called the *Sikoba federation*.

The Sikoba federation will maintain the blockchain, enforcing its economic rules. The user trust factor in this case is in the choice of the federation members. No one member can cheat the system on its own, and as long as a large enough percentage of the members are honest, the blockchain remains intact.

The trust factor makes the selection of members a delicate task. It will be done in two stages: During the foundation period, the selection will be managed by Sikoba (the company) itself. However, once the system passes to production stage, the federation will start governing itself, simply meaning that members will be added or removed from the federation, based on a majority vote of the federation members.

Performance. Selection of federation members, both by Sikoba (the company) in the foundation period, and by the (Sikoba) federation itself later on, will be based on technical criteria. To ensure that the Sikoba blockchain can meet its performance requirements, candidate members would have to prove they possess high computational power, large storage, and advanced network connectivity capabilities. In return, members are compensated financially by users transaction fees.

Additionally, permissioned consensus algorithms are considerably faster than the existing permissionless ones, boosting the performance of the system.

Design flexibility. In order to tackle the flexibility, another feature is added to the blockchain. Namely, the economic logic of Sikoba works is implemented as a script within the blockchain.

This makes a change to the economic rules, which requires a dangerous fork in other blockchains, a simple a matter of a voting of the Sikoba federation within the blockchain. It allows upgrades

to be implemented within a few block cycles.

Anonymity. Sikoba blockchain will be pseudo-anonymous in the same manner that Bitcoin is. Users will be recognized by anonymous addresses. Those addresses may be connected to an identity management service outside the blockchain, but from the federation members point of view, users identities are unknown.

Privacy. It is obvious that a blockchain design must be such that users can authenticate the data related to them. In many projects, this requires downloading the full blockchain, which is a slow task consuming many resources. In the Sikoba blockchain, we want to avoid long synchronization times. In addition, for privacy reasons, we do not want to share with users the full state of the Sikoba blockchain.

How is this matter solved? By saving in each block a hash, which is the root of the Merkle tree generated by the state of the system. This means that it is sufficient for a user to get their own specific data, in addition to a few other nodes in the tree, in order to verify its authenticity. Federation members, on their side, would have to enforce that only the necessary information is sent to each user, based on its privileges.

Implementation. Sikoba is not meant to be built from scratch. Which existing technologies can support it?

One option is modular blockchains, such as Hyperledger's Fabric (supported by IBM) or Sawtooth Lake (supported by Intel). Those projects include built-in customization, allowing Sikoba to build its own economic rules and governing system on top of them. Another option is customizing Ethereum's codebase to allow federated blockchains. In this case, the economic rules would be written as a smart contract. There are other options, in different stages of development, such as Tezos or the Cosmos blockchains network.

We are currently in the process of evaluating those solutions, and choosing the right one for Sikoba.

Disclaimer. Federated blockchain architecture, governing model and the Sikoba blockchain is work in progress. This document is a "living" document, intended to continue evolving, and is in no way to be considered final.