RChain Node Validator Overview

RChain COOPERATIVE

August 2018

Copyright Notice

Audience

The distribution of this document is unlimited. It should be of interest to technical and executive members of the RChain community, who are currently planning or considering participating in the network as validators.

Table of Contents

RChain Node Validator Overview 1

Copyright Notice 2

Audience 2

Table of Contents 2

Executive Summary 4

How to Become an RChain Node Validator 4

Casper Proof-of-Stake in RChain 5

Parameters for RChain's Casper 5

Fee Structures on RChain 7

Transaction Fees 7

Validator Joining Fees 7

Slashing Conditions 9

Verification of Work 9

Ejection Criteria 10

The RChain Network 10

The RChain Root Shard 10

High Performance Shards 11

Peering in the RChain Network's Root Shard 12

Backbone Validation 12

Local Validation 12

Observers 13

The RChain TestNet 13

Approximate TestNetwork Specifications 13

The RChain MainNet 14

The RChain Backbone 14

Executive Summary

RChain is creating a next-generation blockchain with the goal of creating a protocol that allows for faster transactions on a public blockchain, the implementation of more complex smart contracts, and the building of more scalable functions. To accomplish its goal, RChain will use a proof-of-stake consensus mechanism. The proof-of-stake consensus mechanism requires a distributed group of node validators to secure the RChain network.

There are three levels of node validation: (1) Professional ($1,000,000 minimum staking requirement), (2) Business ($250,000 minimum staking requirement), and (3) Enthusiasts ($1,500 minimum staking requirement), which are based on the level of the node validator's RHOC stake (or REV after the MainNet launch). As a node validator, a node validator must validate transactions and add blocks to the RChain blockchain at a level that is commensurate with its node validation level.

In order to become a node validator, an interested party—which may be an individual, a group of individuals, or an entity—must be verified by undergoing Know Your Customer ("**KYC**"), Anti-Money Laundering ("**AML**"), and other national and international sanction list evaluations. Once verified, the prospective node validator must then (1) become a member of the RChain Cooperative, (2) agree to the terms of the Node Validator Agreement, and (3) purchase the RHOC/REV needed to stake on the RChain blockchain.

In purchasing RHOC/REV, a node validator receives an asset granting the node validator with the requisite cryptographic token to stake on the RChain blockchain in proportion to the amount of RHOC/REV it holds. The RHOC purchased in connection with the Node Validation Agreement shall be locked until RChain's MainNet launch, RChain's initial launch (also referred to as "**Mercury**"), contemplated to take place in December 2018, in a multi-signature cryptographic wallet, where Node Validator and RChain hold the two requisite private keys. This REV, into which the RHOC shall convert on a 1-for-1 basis at the MainNet Launch, will be staked for an additional Eighteen (18) months after the MainNet Launch (the "**Staking Period**"). During the Staking Period, Node Validators will still be able to earn and receive revenue from other node validation revenue opportunities as discussed further in this document.

To perform its validation duties, a node validator may use a variety of different hardware and storage options, but all of the hardware, server, storage, and other validating equipment (collectively, "**Node Validation Specs**") must be approved by RChain in writing before a node validator can use the Node Validation Specs in connection with staking and validating on the RChain blockchain.

How to Become an RChain Node Validator

If you are not yet a member of the RChain Cooperative:

1. Complete the RChain Validator Registration: https://staking.rchain.coop/signup
2. Undergo KYC, AML, and other identity verification
3. Become a member of the RChain Cooperative
4. Agree to the terms of the Node Validation Agreement
5. Purchase RHOC for staking purposes
6. Receive written approval for and purchase Node Validation Specs

If you are already a member of the RChain Cooperative:

1. Complete the RChain Validator Registration: https://staking.rchain.coop/signup
2. Agree to the terms of the Node Validation Agreement
3. Purchase RHOC for staking purposes
4. Receive written approval for and purchase Node Validation Specs

Casper Proof-of-Stake in RChain

RChain uses a pure proof-of-stake version of the correct-by-construction (“**CBC**”) Casper consensus framework.[1] Many details remain to be finalized; as a result, readers are encouraged to consider the following to be a DRAFT proposal and are politely asked to provide feedback and suggestions addressing any concerns they may have.

On RChain, each shard is an independent blockchain, all running their own Casper consensus protocols. This enables complete parallelism for operations local to different shards. All shards will share the same currency, called REV, for staking and purchasing computation. Transferring REV between shards is made possible by cross-shard relationships which form a tree data structure. [2]

Parameters for RChain’s Casper

1. A shard’s validator set is unlimited and may contain as many validators as wish to bond (and are accepted by the current validators).

2. Validators must deposit their stake by providing a purse that will be bonded for the entire time they are resident in the validator set.

3. Bonds are fixed for each shard such that all nodes must stake the same amount.

    a. There is complexity in managing how a shard’s stake amount changes over time, which needs to be resolved in future protocol designs.

4. Bond amounts vary between shards. The expected root shard bond amount is between $1,000 and $2,000 dollars· However, subsidiary shards may have arbitrarily high bonds.

5. Validators may attempt to bond to any shard at any time and the root shard should admit all validators that wish to bond. However, subsidiary shards may refuse validators that attempt to join. This can happen under various circumstances, e.g. if the shard requires specific hardware that the joining validator has not deployed.

6. There are no global minimum amounts for shard bonds.

7. Bonds for the root shard must be set by the community during genesis and are updated by the validators thereafter.

8. When a validator requests unbonding, the validator must wait between 3 and 6 months for their purse to be returned.

a. Validators may, from time to time, request “redemption” of funds within their bonded purse above the shard’s stake parameter. E.g., if the stake requirement is 1,000 REV and the validator’s bonded purse contains 1,500 REV, then the validator may request a “redemption” of 500 REV.

b. Redemptions are time-locked in the same way as funds being released during unbonding, so validators must wait an extended period to spend such funds.

9. Unbonded validators may re-apply for validation during their unbonding period using their original stake. E.g., if a validator is ejected for downtime, then when they come back online again they may re-apply to become a validator.

10. RChain’s Casper includes periodic monetary expansion.

a. It should be noted that while monetary expansion is desired by the RChain Cooperative, technical limitations exist that may prevent implementation in the Casper algorithm. Additional work is needed to determine how expansion can be made to work with RChain’s sharding architecture.

b. No specific rate of monetary expansion has been established.

11. RChain’s Casper includes sender-proposed transaction fees.

12. In MainNet, shards may not change their stake amounts, but in future versions we expect staking requirements to vary over time.

13. During bonding, a new validator must pay a “joining fee” that is distributed among the existing validators and deposited to their bonded purses.

Fee Structures on RChain

Transaction Fees

RChain uses the following transaction fee structure:

- Each transaction fee is proposed by the originator of the transaction, typically referred to as a "deployer."

- Fees are split among the validators according to the following equations:

- $b \coloneqq (1+k)(fee/(n+k)$
- $e \coloneqq fee/(n+k)$

  - Where:

    - **b** is the block proposer whose block introduces a transaction,

    - **e** is any validator that is not **b**

    - **n** is the number of validators,

    - **k** is a shard-specific "fee structure" parameter,

    - **fee** is the proposed transaction fee.

  - Notice: when **k=0**, all validators receive the same payment, and in the limit as **k** tends to infinity only the block proposer receives the entire fee.

- Calculating b and e will provide an Initial weight W_x For Each Validator X.

- For Each Validator X with stake equal to S_x, the final distributed fee will be calculated as $\text{fee}_x = \frac{\text{fee} * W_x * S_x}{\sum_{x=1}^{n} W_x * S_x}$

- It is expected that for RChain's root shard, the fee structure parameter k will be low, but that k may be larger in other shards

- 0.01% of all transaction fees are delivered to certain bonded wallets owned by the RChain Cooperative, to help fund maintenance and development of the software and the core network components.

Validator Joining Fees

One very serious attack on proof-of-stake networks occurs when a single investor deploys many nodes as validators, thus overwhelming the original validator set. Depending on the fault tolerance thresholds set by validators in the shard, it may be possible to launch this sort of attack by deploying

n/3 validators where n is the number of current validators. This is sometimes referred to a Sybil attack or an "ant army" attack. To avoid ant army attacks, the RChain fee structure requires that new validators pay a fee to existing validators. These fees are structured as follows:

- The joining fee f is equal to the stake for the shard equal to

- Each current validator is awarded a part of the fee, which is deposited in their bonded purse.

- Validators are ordered from 1 to n in order of their validation tenure (i.e., the amount of time they have been validating the shard) such that the oldest validator is known as $v_1$and the newest validator is v_n.

    - Validators that join at genesis will be ordered randomly.

- The amount awarded to validator k given by the function: $award(v_k) = 2f(n-k)/(n(n-1))$

Slashing Conditions

Validators in the RChain network that do not follow the protocol will correctly have their bonded stake revoked. This is called “slashing.” Slashing improves the security of the network because it imposes a cost for failing to follow the protocol. The initial RChain Casper algorithm has only a single slashing condition:

- Equivocation[3]

In the general case, equivocation is when a validator signs two incompatible blocks or signs a block containing an invalid update to the Rholang state. It is also considered equivocation if a validator uses two incompatible blocks or a block with invalid Rholang updates in the justification of a new block. The node software produced by RChain should never equivocate. Therefore, any equivocation is considered a deliberate attempt by a validator to manipulate the proof-of-stake algorithm to their own benefit and results in slashing of the entire contents of the validator’s bonded wallet.

Eventually, the RChain software will be updated to include the following additional slashing conditions:

- Producing an invalid block
  - ○ Not eventually linked to the genesis block
  - ○ Repeated transaction (“double spend”)
  - ○ Invalid Rholang computation (e.g. forging unforgeable name)
  - ○ Incorrectly executing the fork-choice rule (i.e. justification does not match choice of parent)
  - ○ Invalid data fields (e.g. hash, block number, etc.)
- Ignoring a slashable offense (i.e. not slashing when you’re supposed to is slashable).

Early in the network, the RChain node software is subject to a higher probability of bugs that result in validators producing invalid blocks. While RChain and its partners would like to identify and eliminate all such bugs during the TestNet phase, in practice we should account for bugs of this type at the protocol layer to ensure that validators do not lose stake as the result of identifiable problems in the software. Early in the main network’s lifetime, production of invalid blocks will not be considered a slashing offense. However, validators should expect that sometime between MainNet and subsequent releases, this policy will change, and invalid blocks will be considered a slashing offense.

Verification of Work

Validation is a resource intensive task, because the validator must store and compute the state of the Rholang tuplespace before and after every comm event, handle rollbacks correctly, and so forth. As a result, validators have an incentive to “cheat” by receiving blocks without ever proposing any. In this way, the validator avoids having to maintain the tuplespace.

To prevent this, RChain’s Casper algorithm will allow validators to produce “challenges” in the form of deliberately bad blocks which are sent only to a specific validator. The validator in question must reject the bad block, at which time the originating validator “proves” that it was a challenge rather than equivocation.

Ejection Criteria

Sometimes, validators fail to perform correctly, even though they haven’t been guilty of a slashable offense. In this case, the validator may be removed from the validator set automatically by the other validators. This is called “validator ejection.”

Validator ejection does not result in slashing but acts as if the validator itself requested unbonding. Eventually, the validator’s stake will be returned to an unbonded wallet just as if the validator had voluntarily left the shard.

Ejection criteria have not yet been determined, but are expected to include situations such as the following:

- Validators experiencing unusually high latency.
- Validators experiencing extended downtime.
- Validators who propose no new blocks for an extended period, even if they are online.
- Validators that send a block with an invalid signature, since this can be caused by a hardware or networking fault. Specific ejection criteria parameters are expected to vary from shard to shard.

The RChain Network

RChain’s MainNet is a proof-of-stake smart contracting blockchain with hierarchical sharding that operates at global scale. At the heart of the network is a highly decentralized “root shard” that provides robust economic security as the result of hundreds or thousands of individually staked transaction validators operating in parallel. RChain’s unique sharding architecture allows other shards to be deployed from the root shard, each with its own economic and technical parameters.

Unlike the smart contracting blockchains of today, RChain implements a next-generation processing architecture based on the rho-calculus[4], and can leverage significant computational power to allow large-scale distributed applications to run in and coordinate with a decentralized, economically secured blockchain.

RChain’s unique sharding architecture further allows both private and semi-private blockchainsto integrate directly with the public blockchain. This will allow a variety of enterprise and traditional finance applications to seamlessly interact with blockchain technology for the very firsttime.

The RChain Root Shard

The RChain network’s security begins and ends with the root shard. On its own, the root shard looks and performs much like current smart contracting blockchains. The root shard must have thousands of individual validators staking a sufficient amount to eliminate the incentive for validators to form coalitions.

The Root Shard will have the following general characteristics:

- Many global validators to ensure maximum decentralization, transaction, and wallet security.

- Slow block propagation times due to the large number of heterogeneous validators.

- Low total transaction throughput due to slow block propagation times (only around 10x current Ethereum throughputs, e.g.)

- Expensive transactions due to the large number of validators.

- Very low staking requirements to encourage large numbers of validators.

- Very low validator joining fees to encourage validators at various levels of participation.

High Performance Shards

In order to accomplish our performance goals, the RChain Cooperative and Pyrofex intend to roll out a number of high-performance shards with regional focus. The first of these shards will be deployed in the United States, but RChain encourages validator groups to form across the world and deploy their own regional shards.

A regional shard may have fewer individual validators who each stake larger amounts than is typical for the root shard. This makes these shards suitable for applications that need both acceptable levels of transaction security and performance. But, applications that are willing to sacrifice performance for security should consider the root shard, instead.

High Performance Shards will have the following general characteristics:

- Sufficient validators to ensure regional decentralization, transaction, and wallet security.

- Rapid block propagation times due to high levels of backbone bandwidth.

- High total transaction throughput due to rapid block propagation times (perhaps as much as 1,000x current Ethereum transaction rates)

- Moderate transaction fees due to the good balance of security and performance.

- Higher staking requirements to encourage validators to make effective infrastructure investments.

- Higher validator joining fees to protect against “ant army” attacks.

Peering in the RChain Network's Root Shard

RChain's network is intended to be extremely low latency, with block confirmation times on the order of just a few seconds. It is also intended to be extremely high capacity, targeting 40,000 total transactions per second. Unlike most existing blockchains, RChain has the convenience of a single governance organization that we can leverage to help split the difference between these competing, but equally valuable goals.

To accomplish this, the RChain's peer-to-peer network is split into the following parts:

Backbone Validators

- Local Validators

- Observers (or "Watchers")

Backbone Validation

RChain's backbone validators form a core network of extremely well-provisioned infrastructure that is capable of handling a large amount of global traffic at exceptional latency and throughput. Backbone validators SHOULD follow these rules:

1. Backbone validators deliver transactions to each other in a fully-connected mesh.

2. Backbone validators will accept transactions from any user.

3. Backbone validators will deliver current block updates to any user.

It is notable that backbone validators are not required to provide any user with a full copy of the entire validation history. This can be downloaded by users from the local validators as described below.

None of these rules are slashing conditions.

Local Validation

RChain's local validators form a peripheral network of locally provisioned infrastructure that is capable of handling regional traffic levels at acceptable latency and throughput. Local validators SHOULD follow these rules:

1. Local validators should provide complete chain history downloads to users in their own region.

2. Local validators deliver transactions to other local validators in the same region andto backbone validators with as high a graph connection level as is practical.

3. Local validators will deliver current block updates to any user in their region.

Local validators are not required to be promiscuous outside of their region and, due to the variation in local regional network architecture, may not be able to do so reliably in any case.

None of these rules are slashing conditions.

Observers

RChain Observers are nodes that receive and validate a shard's entire block graph. Observers must receive block updates from their upstream validators and may choose to provide local services, but are not required to do so. Observers may not participate in the Casper protocol and must use out of band mechanisms if they detect slashable offenses.

The RChain TestNet

The test network is slated to launch at the RCon3 conference in Berlin on September 5th. The test network will consist of the single root shard (a.k.a. "/") and, possibly, a single subordinate in the Western United States (a.k.a. "/us/west").

Approximate TestNetwork Specifications

The test network infrastructure is intended to provide sufficient capacity to test the RChain network across many small nodes as well as across fewer large nodes. Pyrofex intends to coordinate testing with the RChain Cooperative to measure performance details such as the consumption of CPU, RAM, and disk resources as well as the amount and variety of Casper traffic in various configurations and application environments. Actual rollout of the main network backbone capacity will vary based on actual test results. The following is included for reference only. Final specifications and acquisition of Node Validation Specs should be made based on real-world test results.

- 200 AMD Naples 7401 cores
- 4 TB DDR4-2666
- 200TB SATA HDD
- 25Gbps Ethernet between hosted nodes

The RChain MainNet

The RChain MainNet design is expected to change based on information gleaned during test net launch and burn in. The rollout process for RChain's main network will be completed in stages. Pyrofex and the RChain Cooperative will coordinate with validators large and small to ensure the main network rollout is smooth.

The RChain Backbone

A select group of large-scale validators will coordinate to create a backbone of high-performance infrastructure that handles the bulk of global traffic management, validation, and end-user services. This does not imply that RChain is centralized, merely that some network services can be obtained with higher performance from the backbone than from local providers.

The RChain backbone will MINIMALLY consist of nodes operated by RChain Cooperative, Refective Ventures, and Pyrofex Corporation with the following footprint. Please note that capacity is expressed in "cores" here without regard to RAM, disk, etc. Those wishing to perform capacity planning for initial deployments may take the suggested numbers from TestNet as described above but are encouraged to wait until test results have been obtained before making final calculations for their rollout.

The following table is provisional.

Exchange Point	City	Country	Capacity (in cores)
SIX	Seattle, Salt Lake City	USA	300
LINX	Northern VA	USA	300
LINX	London	UK	300
Equinix	Los Angeles	USA	300
Equinix	Singapore	Singapore	300

Validators are encouraged to coordinate with RChain, Reflective, or Pyrofex if they wish to deploy hardware in these locations. Service estimates are available upon request, if needed.

[1] https://github.com/ethereum/cbc-casper/wiki
[2] https://rchain.atlassian.net/wiki/spaces/CORE/pages/488243367/Cross-shard+transfers
[3] https://github.com/ethereum/cbc-casper/wiki/FAQ#wait-what-about-faulty-validators
[4] https://www.sciencedirect.com/science/article/pii/S1571066105051893