# #

Staking

# #

Summary

This specification briefly introduces the functionality of stake module and what user should do with the provided commands.

# #

Concepts

# #

Voting power

Voting power is a consensus concept. IRIShub is a Byzantine-fault-tolerant POS blockchain network. During the consensus process, a set of validators will vote the proposal block. If a validator thinks the proposal block is valid, it will voteyes , otherwise, it will vote nil. The votes from different validator don't have the same weight. The weight of a vote is called the voting power of the corresponding validator.

# #

Validator

Validator is a full IRIShub node. If its voting power is zero, it is just a normal full node or a validator candidate. Once its voting power is positive, then it is a real validator.

# #

Delegator && Delegation

People that cannot, or do not want to run validator nodes, can still participate in the staking process as delegators. After delegating some tokens to validators, delegators will gain shares from corresponding validators. Delegating tokens is also called bonding tokens to validators. Later we will have detailed description on it. Besides, a validator operator is also a delegator. Usually, a validator operator not only has delegation on its own validator, but also can have delegation on other validators.

DANGER

It is strongly NOT recommended that validator operator COMPLETELY unbind self-delegation tokens, Cause the validator will be jailed (removed out of validator set) if he do so. The delegator who bonded tokens to this validator will also suffer losses. So, it is recommended that validator operator reserve at least 1 iris while unbonding tokens.

# #

Validator Candidates

The quantity of validators can't increase infinitely. Too many validators may result in low efficient consensus which slows down the blockchain TPS. So Byzantine-fault-tolerant POS blockchain network will have a limiation to the validator quantity. Usually, the value is 100. If more than 100 full nodes apply to join validator set. Then only these nodes with top 100 most bonded tokens will be real validators. Others will be validator candidates and will be descending sorted according to their bonded token amount. Once the one or more validators are kicked out from validator set, then the top candidates will be added into validator set automatically.

# #

Bond && Unbond && Unbonding Period

Validator operators must bond their liquid tokens to their validators. The validator voting power is proportional to the bonded tokens including both self-bonded tokens and tokens from other delegators. Validator operators can lower their own bonded tokens by sending unbond transactions. Delegators can also lower bonded token by sending unbond transactions. However, these unbonded token won't become liquid tokens immediately. After the unbond transactions are executed, the corresponding validator operators or delegators can't sending unbond transactions on the same validators again until the unbonding period is end. Usually the unbonding period is three weeks. Once the unbonding period is end, the unbonded

token will become liquid token automatically. The unbonding period mechanism makes great contribution to the security of POS blockchain network. Besides, if the self-bonded token equals to zero, then the corresponding validator will be removed out of validator set.

[#](#)

Redelegate

Delegators can transfer their delegation from one validator to another one. Redelegation can be devided into two steps: ubond from first validator and bond to another validator. As we have talked above, ubond operation can't be completed immediately until unbonding period is end, which means delegators can't send another redelegation transactions immediately.

[#](#)

Evidence && Slash

The Byzantine-fault-tolerant POS blockchain network assume that the Byzantine nodes possess less than 1/3 of total voting power. These Byzantine nodes must be punished. So it is necessary to collect the evidence of Byzantine behavior. According to the evidence, stake module will aotumatically slash a certain mount of token from corresponding validators and delegators. The slashed tokens are just burned. Besides, the Byzantine validators will be removed from the validator set and put into jail, which means their voting power is zero. During the jail period, these nodes are not event validator candidates . Once the jail period is end, they can send transactions to unjail themselves and become validator candidates again.

[#](#)

Rewards

As a delegator, the more bonded tokens it has on validator, the more rewards it will earn. For a validator operator, it will have extra rewards: validator commission. The rewards come from token inflation and transaction fee. As for how to calculate the rewards and how to get the rewards, please refer to[mint](mint) and[distribution](distribution) .

[#](#)

What Users Can Do

- Query your own validator
- Users can query their own validators by their wallet address. But firstly users have to convert their wallet addresses to validator operator address pattern:

iris keys show< key-name> Example Output:

- name: node0 type:local address: iaa1w9lvhwlvkwqvg08q84n2k4nn896u9pqx93velx pubkey: iap1addwnpepq03g7u43y3gwfz3pd4gkwz7d4mt600kzsc5cj2ysx58a5hp84qyduxtw28r mnemonic:"" threshold:0 pubkeys:[ ] Then, example command to query validator:

iris q staking validator iva14n9md3sq9xwscs96za8n85m0j9y2yu3cagxgke Example Output:

{ "operator_address" : "iva14n9md3sq9xwscs96za8n85m0j9y2yu3cagxgke" , "consensus_pubkey" : "icp1zcjduepq9meszzqu54gpxvs4vzvuv85qvv5ef0egz3sde0ps4dvktcv77uds0kkhgf" , "status" : 3 , "tokens" : "100000000" , "delegator_shares" : "100000000.000000000000000000" , "description" : { "moniker" : "node0" } , "unbonding_time" : "1970-01-01T00:00:00Z" , "commission" : { "commission_rates" : { "rate" : "1.000000000000000000" , "max_rate" : "1.000000000000000000" , "max_change_rate" : "1.000000000000000000" } , "update_time" : "2020-08-26T06:43:07.065305Z" } , "min_self_delegation" : "1" } * Edit validator

iris tx staking edit-validator--from

< key-name> --chain-id= irishub--fees = 0 .3iris --commission-rate= 0.15 --moniker = < new-name> * Delegate tokens to other validators If you just want to be a delegator, you can skip the above steps.

iris tx staking delegate iva14n9md3sq9xwscs96za8n85m0j9y2yu3cagxgke 1000iris --chain-id= irishub--from = < key-name> --fees = 0 .3iris * Unbond tokens from a validator

iris tx staking unbond iva14n9md3sq9xwscs96za8n85m0j9y2yu3cagxgke 1000iris --chain-id= irishub--from = < key-name> --fees = 0 .3iris * Redelegate tokens to another validator

iris tx staking redelegate iva14n9md3sq9xwscs96za8n85m0j9y2yu3cagxgke

iva1l2rsakp388kuv9k8qzq6lrm9taddae7fpx59wm 100iris--from mykey --chain-id= irishub--from = < key-name> --fees = 0 .3iris For other staking commands, please refer to[stake cli client](#)